

California State University, San Bernardino

CSUSB ScholarWorks

Theses Digitization Project

John M. Pfau Library

2010

Online administration system for small and medium-sized running events

Steinar Johansen

Follow this and additional works at: <https://scholarworks.lib.csusb.edu/etd-project>



Part of the [Databases and Information Systems Commons](#), and the [Software Engineering Commons](#)

Recommended Citation

Johansen, Steinar, "Online administration system for small and medium-sized running events" (2010).
Theses Digitization Project. 3836.
<https://scholarworks.lib.csusb.edu/etd-project/3836>

This Project is brought to you for free and open access by the John M. Pfau Library at CSUSB ScholarWorks. It has been accepted for inclusion in Theses Digitization Project by an authorized administrator of CSUSB ScholarWorks. For more information, please contact scholarworks@csusb.edu.

ONLINE ADMINISTRATION SYSTEM FOR SMALL AND
MEDIUM - SIZED RUNNING EVENTS

A Project
Presented to the
Faculty of
California State University,
San Bernardino

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
in
Computer Science

by
Steinar Johansen
March 2010

ONLINE ADMINISTRATION SYSTEM FOR SMALL AND
MEDIUM - SIZED RUNNING EVENTS

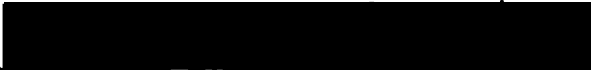
A Project
Presented to the
Faculty of
California State University,
San Bernardino

by
Steinar Johansen
March 2010

Approved by:



Dr. Ernesto Gomez, Chair,
Computer Science



Dr. Richard Botting



Dr. Josephine Mendoza

March 18, 2010
Date

ABSTRACT

The purpose of this project is to address inefficiency at small to medium-sized running events with regards to registration, race timing and result service. As a race coordinator or a participant, it is important that the race timing is correct and that the result service can be delivered as soon as possible after the event is finished. This project focuses on how an event coordinator can easily advertise his event online, collect online registration for the event and use the data to produce race results that are correct and delivered fast. This is done by providing the race coordinator with an online tool where a race flyer and payment can be defined, and where the race timing and result service is administrated, to make them available to the participants as early as possible after they have finished their race. This system will be a big improvement to all small to medium-sized running events because they do not have to develop their own online system. The system provides all the requirements to have a successful running event.

ACKNOWLEDGEMENTS

I thank the faculty at California State University San Bernardino, Department of Computer Science and Engineering for giving me the opportunity to pursue my Masters Degree in Computer Science. I express my sincere appreciation to my graduate advisor, Dr. Ernesto Gomez for inspiring me during the work with this project. I also like to express my sincere appreciation to the two other members of the board of advisors, Dr. Josephine Mendoza for advising me in the field of databases, and Dr. Richard Botting for his overall knowledge in the field of computer science.

TABLE OF CONTENTS

ABSTRACT iii

ACKNOWLEDGEMENTS iv

LIST OF TABLES vii

LIST OF FIGURES viii

CHAPTER ONE: INTRODUCTION

 1.1 Introduction. 1

 1.2 Purpose 2

 1.3 System Back End 3

 1.4 System Front End. 5

 1.5 Project Scope 6

 1.6 Acronyms and Abbreviations. 7

CHAPTER TWO: ARCHITECTURE

 2.1 The Overall Project Design. 9

 2.2 The Content Management System 9

 2.3 "Joomla!" File System 12

 2.4 "Joomla!" Components. 13

CHAPTER THREE: DATABASE DESIGN

 3.1 Introduction. 18

 3.2 Overall Design. 18

 3.3 Conceptual Design 19

 3.4 Data Dictionary 23

CHAPTER FOUR: PROJECT DESIGN AND SOLUTIONS

4.1 Project Analysis. 31
4.2 Online Advertisement. 32
4.3 Management of Registrants and People. 48
4.4 Race Timing 60
4.5 Event Advertisement and Registration. 64
4.6 Result Service. 79

CHAPTER FIVE: PROGRAM SOLUTIONS AND SECURITY

5.1 Introduction. 84
5.2 Solutions to Problems 84
5.3 Security. 87

CHAPTER SIX: SYSTEM TEST

6.1 Introduction. 89
6.2 Component Tests 89
6.3 System Tests. 90

CHAPTER SEVEN: CONCLUSION AND FURTHER DIRECTIONS

7.1 Conclusion. 91

APPENDIX A: PROJECT SOURCE CODE, BACK END. 92

APPENDIX B: PROJECT SOURCE CODE, FRONT END 172

REFERENCES 236

LIST OF TABLES

Table 1. Entity Description.	21
Table 2. Data Dictionary	23

LIST OF FIGURES

Figure 1. Use Case Diagram for the Web Site	4
Figure 2. Complete Joomla! System	10
Figure 3. Joomla! Diagram	11
Figure 4. Joomla! 1.5 File Structure.	13
Figure 5. The Back End Components Folder.	14
Figure 6. com_manualreg Component	15
Figure 7. com_manualreg Component and Files	17
Figure 8. Conceptual Schema	20
Figure 9. Table Layout.	22
Figure 10. Step 1. Register Organization.	33
Figure 11. Register New or Edit Organization.	34
Figure 12. Save Organization.	35
Figure 13. com_organizations.	36
Figure 14. List Event Flyers.	38
Figure 15. Create an Event Flyer Form	39
Figure 16. com_flyer.	41
Figure 17. Select Event View.	43
Figure 18. Define Fees View	44
Figure 19. List Dues and Payments Page.	45
Figure 20. Edit Dues and Payments Page.	46
Figure 21. com_payment.	46
Figure 22. Control Panel and List People Icon	48

Figure 23. List All Page.	49
Figure 24. Administrate People Window	51
Figure 25. com_registrations.	52
Figure 26. Select Event Page Manual Registration.	53
Figure 27. Add Participants Page.	54
Figure 28. Search Database Page	56
Figure 29. Using the Search Field	57
Figure 30. com_manualreg and com_mysearch	58
Figure 31. Control Panel, Race Timing Button.	60
Figure 32. The Select Event Page.	61
Figure 33. Enter Bib and Time Window.	62
Figure 34. com_timing Folders and Files	63
Figure 35. Front End, Start Page.	65
Figure 36. Local Running Events Page.	66
Figure 37. Race Flyer	68
Figure 38. Login Window Event Registration.	69
Figure 39. Register Page.	70
Figure 40. Race Registration Page	72
Figure 41. Check Out Page	73
Figure 42. Enter Payment Information Page	74
Figure 43. Payment Approved Page.	75
Figure 44. Already Registered Page.	76
Figure 45. Front End Components	77

Figure 46. Access to Event Results.	80
Figure 47. List Events with Results Service	81
Figure 48. Race Result Page	82
Figure 49. Result Search for Bib Number 1	83
Figure 50. Calculate Race Day Age	86

CHAPTER ONE

INTRODUCTION

1.1 Introduction

To have a successful running event for fundraising, many tasks have to be completed within a short time frame. A task of high priority is determining the race timing company to be used. There are several companies that provide professional race timing services for all types of running events. They can provide either electronic or manual timing services depending on the requirements of the event. However, none of these companies provides additional online services including event registration, advertisement and access to instantaneous results. Furthermore, those companies are costly and would result in decreased revenue from the event for the organization.

Runner's satisfaction with an event includes: easy web access to information regarding the event, reliable and accessible result postings. Small to medium-sized events don't have the resources to create an appealing informative web site to attract runners. Nor do they have the resources to hire a professional company to be responsible for the results.

Runners who want to participate in local events often browse the internet. There are several online sites where you can find information about running events. Often these sites cover large geographical areas making it difficult to find local events because of the number of postings. These postings usually give a minimal amount of information. Most do not have an online registration system. Once the event is finished, it is very important that the result service is timely and accurate. From a runner's standpoint, this is probably the most important issue and it often determines whether a runner will come back to another event held by the organization.

1.2 Purpose

The purpose of this project is to resolve the problems described above that are common for small to medium-sized running events. This web project will make it easy for a race coordinator to design an appealing "online posted" event flyer. It will also provide online registration for participants who want to register for a local event. In addition the site provides a very efficient way to handle on site, day of the event registrations. The system also provides a manual timing system that posts the results

online almost instantaneously. By browsing the web site the runner and family members will get almost immediate feedback regarding the runner's performance. Finally the system is generating a searchable result service where results can be searched for based on different parameters such as bib number, first name, last name, both first and last name, city or age group.

The purpose of the web site is to handle local events only. Race coordinators can define their specific event as they wish and the flyer will be searchable online immediately after the race coordinator has decided to publish it. This project will improve all the important functions for a small to medium-sized running event such as race advertisements, race registrations and result service. The project does not require any programming skills for the users, and the interaction with the system is based on basic computer skills.

1.3 System Back End

The project has two basic parts. A "Front End" and a "Back End". The "Back End" of the system is the administrator's part where a race coordinator will be interacting with the web system. In this part, the

coordinator can design the race flyer, manage registrations and time the event.

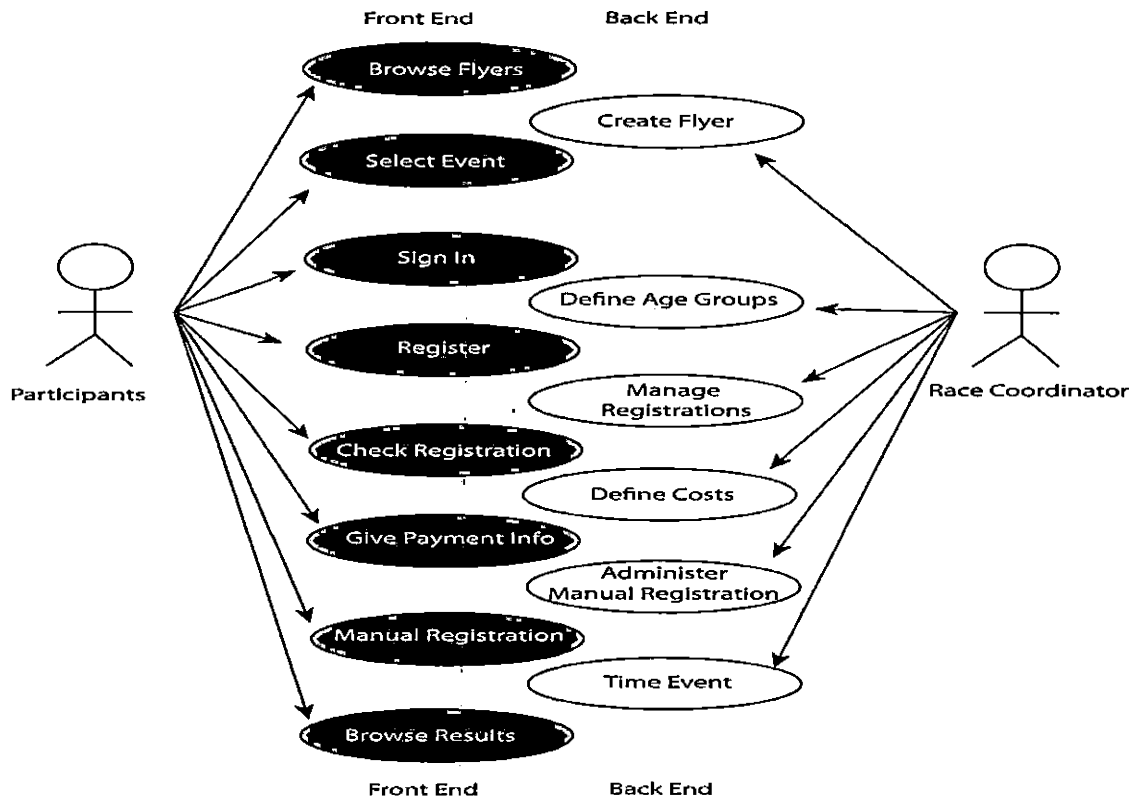


Figure 1. Use Case Diagram for the Web Site

There are three ways a participant can register for an event. First is through the system's online registration system. The second way is by mailing in the registration form and have the race coordinator enter the data into the

system. The third way is on site, same day registration. A race coordinator is given certain privileges to the system to administer an event. Figure 1 shows a use case diagram of the "Back End" and "Front End".

1.4 System Front End

The "Front End" of the system is the part that is accessible for anyone browsing the internet. This part provides a link to all the events posted, with the closest upcoming event on the top of the list. Participants can browse through the different events listed. Each flyer provides all the information needed about the event and a sign up link if they want to participate. Once the sign up link is clicked a person gets two options. If the person is a returning participant and has a username and a password, that information is entered and the participant will see information such as date, time and place for the event, the registrant's personal information and costs and fees for the event. If the registrant is new to the system, personal information such as username, password, name, address, date of birth, gender etc has to be entered. The user is taken to the next step which is the same as those who already

have log in information. From this step a person is just a few clicks away from registering for the event.

Participants and others can also browse results from previous events by following a link on the web site. By selecting an event, they will be able to see the overall results. They are also provided with different search options such as bib, age group, city, first name, last name or both first and last name.

A participant can also log in to the system and get information about which races they are currently signed up for and in which previous races they have participated. Figure 1 demonstrates with a use case diagram how the public interacts with the system.

1.5 Project Scope

The scope of this project is to create a web based system that will help small to medium-sized running events to advertise, manage registrants and to run a professional result service. The tool should be useable for any race coordinator and those given the access to manage the event with just basic computer skills. None of the tasks in the system requires any programming skills.

1.6 Acronyms and Abbreviations

Age Group:	The age span in which a participant belongs. E.g. a 46 year old will be in the 45-49 age group
Apache Server:	Open source web server.
Back End:	The area where the system is administered.
CMS:	Content Management System.
Component:	A MVC style expansion of "Joomla!"
Controller:	The part of a MVC design that processes data submitted by a user.
Editor:	Full featured HTML editor to create a web page.
Front End:	The part of the system visible to all browsing the internet.
HTML:	HyperText Markup Language
JavaScript:	Object Oriented script language used to utilize objects within a client
"Joomla!":	CMS system based on PHP and MySQL.
Model:	The component in MVC design that handles data transfer to and from the database.

MooTools: Compact super lightweight web 2.0
JavaScript framework

MVC: Model-View-Controller design.

MySQL: Open Source database engine

Pagination: A way to reduce or expand the number of
records per page.

PHP: PHP Hypertext Preprocessor.

Result Service: Ranking of participants by their finish
time.

View: The part of a MVC design that presents
options to a user.

Wampserver: Server bundle with Apache server, MySQL
server and PHP.

CHAPTER TWO

ARCHITECTURE

2.1 The Overall Project Design

The overall project design is a model-view-controller (MVC) design based on the "Joomla!" 1.5 Content Management System (CMS). The "Front End" and the "Back End" appear as two separate parts and are extended using the MVC style on each component. Extending the CMS this way will result in some redundant code but that is how the framework is constructed. A certain class might be exactly the same both in the "Front End" and the "Back End" of the system, but since there is no direct communication between those two layers, the system ends up with two identical classes.

2.2 The Content Management System

The "Joomla!" 1.5 Content Management System (CMS) that is used for the project has become the most popular CMS system to be used for more complex web systems using PHP programming. The words "content", "management" and "system" describe exactly what "Joomla!" does for the user. It is a framework that manages content and supports a project with many tools needed to create nearly any type of web systems.

The fact that "Joomla!" can be used to create nearly any type of web system doesn't necessarily mean that it is wise to use it for all systems. However, as open source CMS, one can add facilities to the framework if the framework does not provide a certain tool needed to complete a task. The normal way to create a "Joomla!" supported web site is to extend it.

Figure 2 shows a brief diagram of the four units that builds a complete "Joomla!" system. This is one of the most common setups for "Joomla!"

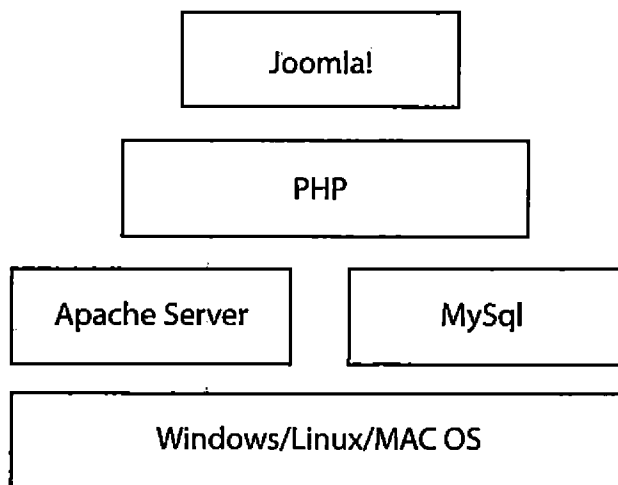


Figure 2. Complete Joomla! System

"Joomla!" follows a model-view-controller (MVC) design to control the flow in the system. Whether "Joomla!" is extended with a component or a module, the MVC design utilizes the functions built into "Joomla!" the best way. Joomla! allows the choice whether to use MVC or not, and the framework is flexible enough to allow other approaches. However, many of the functions in the system will not be available. This project is following a strict MVC style in all the components that are created in the project.

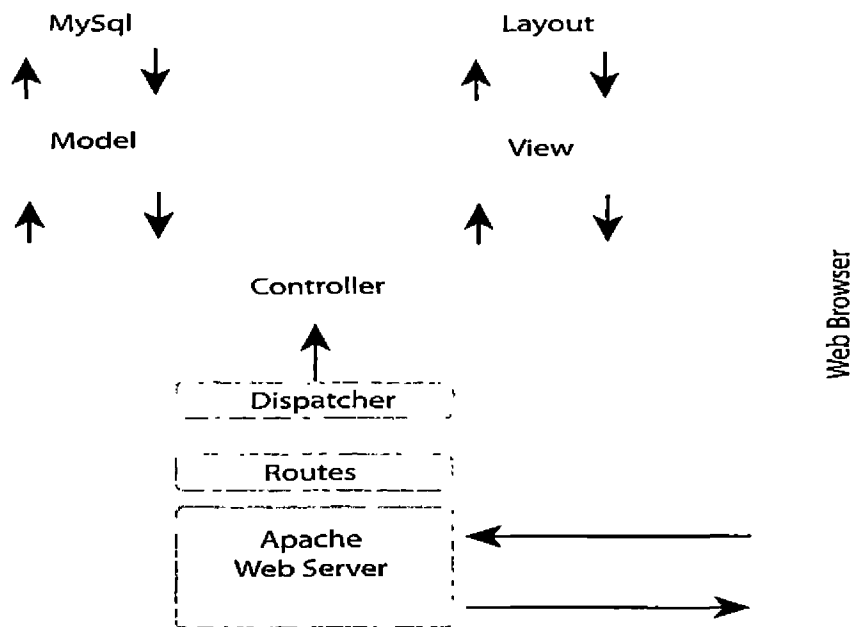


Figure 3. Joomla! Diagram

Figure 3 demonstrates a more detailed drawing of how the "Joomla!" system interacts with the different parts in the system and how the MVC style is obtained.

2.3 "Joomla!" File System

The file system in "Joomla!" is organized very well and it is very easy to navigate through it. The names of the folders and sub folders are self explanatory and at the first glance it might seem very simple. However, with more in-depth examination of the system we find that "Joomla!" 1.5 is very complex. Figure 4 gives a view of how the overall top level file structure is organized for the system. The front end file system is located on the very top level of the source files and the "Back End" file system is organized under the administrator folder. As indicated, many of the folders for the "Front End" are the same as for the "Back End".

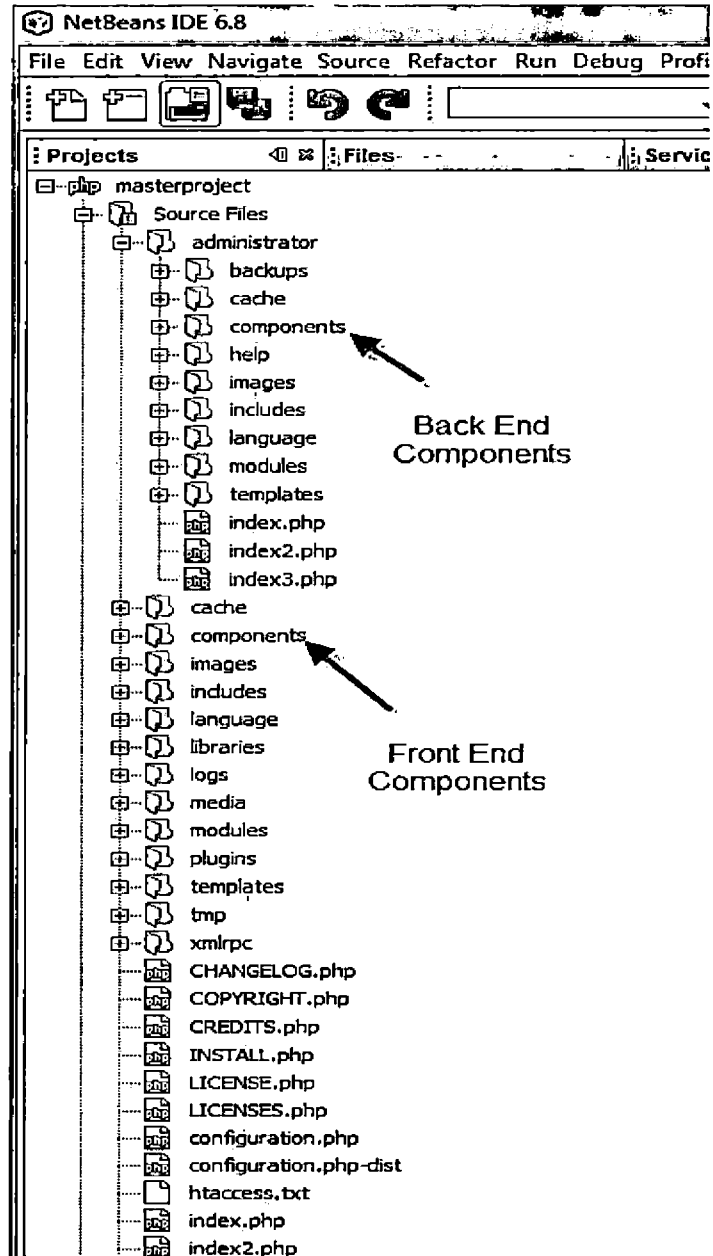


Figure 4. Joomla! 1.5 File Structure

2.4 "Joomla!" Components

This project has extended "Joomla!" 1.5 with components only. Adding a component either to the "Front

End" or the "Back End" follows the same procedure. First, the component has to be given a name following the com_something naming convention. Second, the component has to be registered within "Joomla!" by adding it to the components table in the database. Third, the component is created using MVC style, again following a given naming convention.

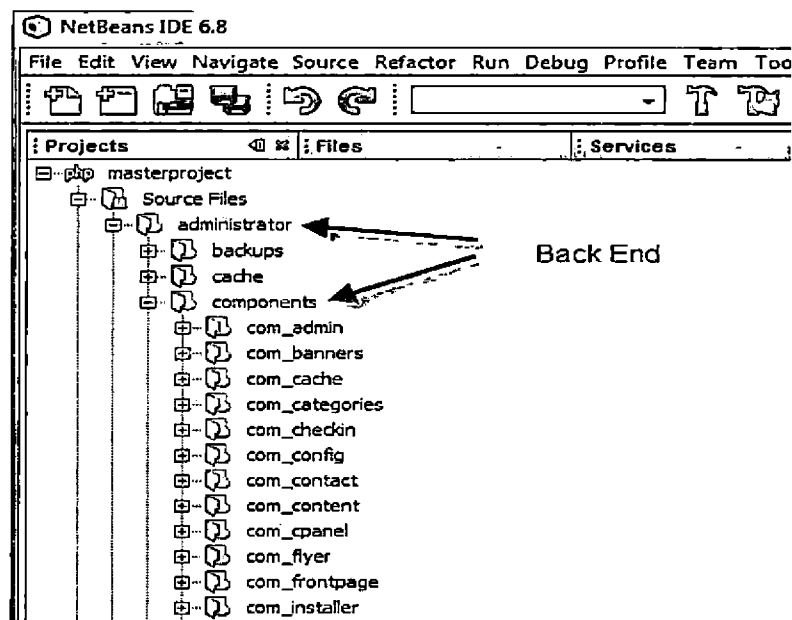


Figure 5. The Back End Components Folder

A component is called com_something e.g. com_login, com_timing, com_manualreg etc. The name that is being used

for a component has to be exactly the same when the component is registered in the components table within the database. Figure 5 demonstrates how components are named within the "Back End" components folder.

Within a component, a strict naming convention is used. There might be components that are not using a full featured MVC design because it does not directly interact with a database. A normal component will have a models folder, a views folder, and probably a tables folder. These folders have to be denoted in a plural form. In addition, to the folders a controller file and a file following the components name are required. For a component to work, the names have to be assigned as outlined. Figure 6 demonstrates how the `com_manualreg` is organized.

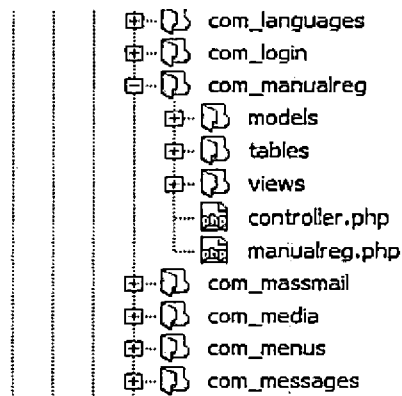


Figure 6. `com_manualreg` Component

The views folder contains the different views of the component. The names of the view can be anything but it is wise to use names that explain the view. Within the view a strict naming convention is used. The files that have to be present are a view.html.php, a folder named tmpl and inside the tmpl folder a file called default.php has to be present. For a component to take advantage of the features in "Joomla!" 1.5, the described folder and file structure must be followed. Further, if there is a model supporting a view, the model file has to bear the same name as the name of the view. Again, this is how the framework is set up to work in a MVC style programming. Figure 7 shows what a full featured component. It also shows the table folder that indicates this component has two tables in the database linked up to itself.

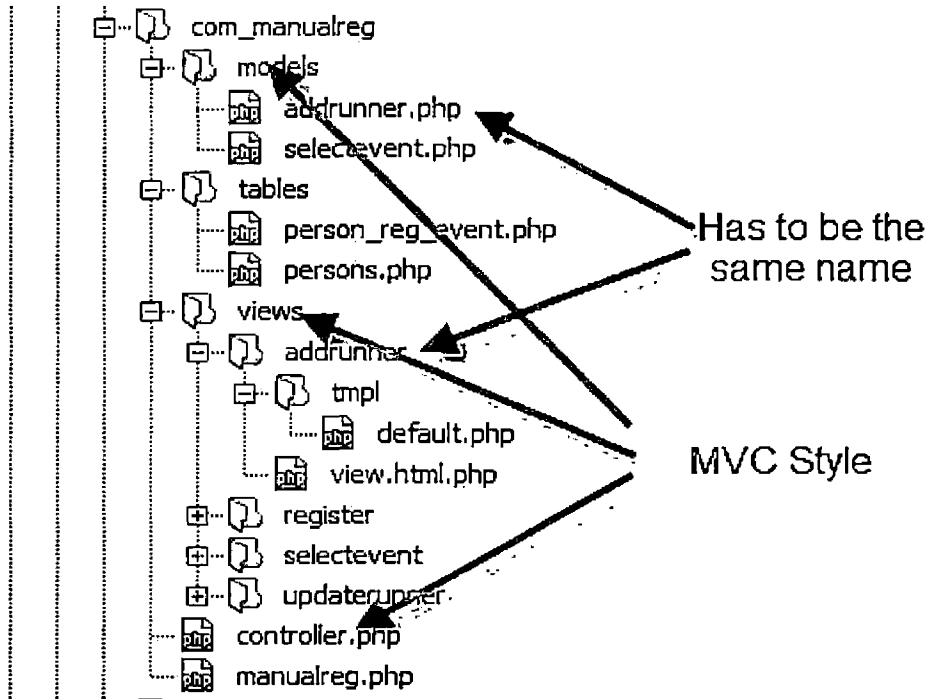


Figure 7. com_manualreg Component and Files

CHAPTER THREE

DATABASE DESIGN

3.1 Introduction

This project is using the MySQL version 5.1.33. It is the version that is embedded in the Wampserver bundle and the database that is commonly used together with "Joomla!" 1.5. The database contains a number of tables and most of them are part of the "Joomla!" 1.5 system. This documentation will only cover the tables that have been used to accomplish the different tasks for administration of a running event. It will not cover the connections that for instance a component uses in the component table of the database.

3.2 Overall Design

The system needs to track the following data: Information about an organization, an event, payments, timing, a registrant and information whether a registrant has paid or not. In addition, the system needs to track results and sort the results based on events, overall result, result among genders and result within an age group.

The tables used to organize the data are all in third normal form to reduce and control redundancy. However, there are exceptions to this rule. Information such as city, state and zip are not normalized. The reason is that the amount of data is not large enough to utilize a performance gain.

To utilize features in "Joomla!" 1.5 sequential assigned primary keys have been chosen for most of the tables. This has been used even on tables that implements many-to-many relationships, to take advantage of features in "Joomla!" 1.5 with regards to save and update of tables. The naming convention for tables connected to "Joomla!" is interesting. To utilize short cuts in the queries, all tables have to start with the name jos_something, e.g. jos_events or jos_persons. To access a table, #__events is used.

3.3 Conceptual Design

The database contains fourteen(14) tables. Ten(10) of those are tables containing information about something or someone while the other four(4) tables are there to implement many-to-many relationship between tables. Figure 8 shows the conceptual schema for the database.

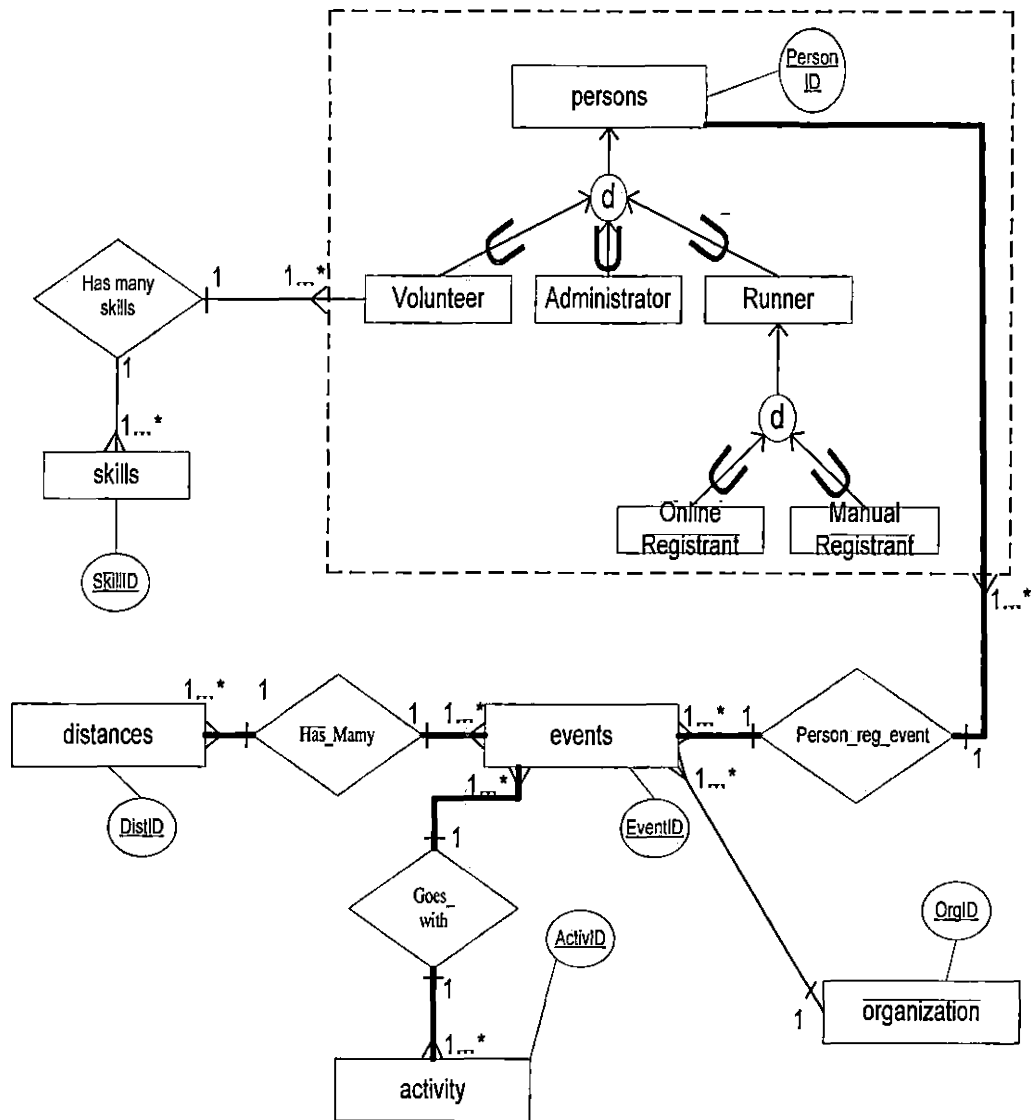


Figure 8. Conceptual Schema

As shown in the conceptual schema, entities are modeled as rectangular squares while the diamonds are the tables that implements many-to-many relationship. In the

implementation, each table name has added jos_ in front of the name to follow Joomla! 1.5 naming scheme, e.g. persons has gotten the name jos_persons, and person_reg_event is named jos_person_reg_event. Table 1 gives a brief overview of the data that each entities tracks.

Table 1. Entity Description

Entity Name	Description
jos_persons	This table contains information about a runner, an administrator, or a volunteer such as name, address, email, phone etc.
jos_events	This table contains information about an event such as where, when, type of event.
jos_organizations	This table contains information about an organization such as name, address, phone, contact person etc.
jos_distances	table contains information about the different distances such as 5k, 10k, half marathon etc.
jos_activity	This table contains information about different activities such as run, walk, run/walk etc.
jos_skills	This table contain information about different skills such as timing, refreshment, traffic regulations etc.
jos tshirt	This table contains t-shirt sizes.

Figure 9 shows the table layout and a full implementation of the database. The figure shows all the table names as they have been used in the implementation of the database. Connection between tables are also shown.

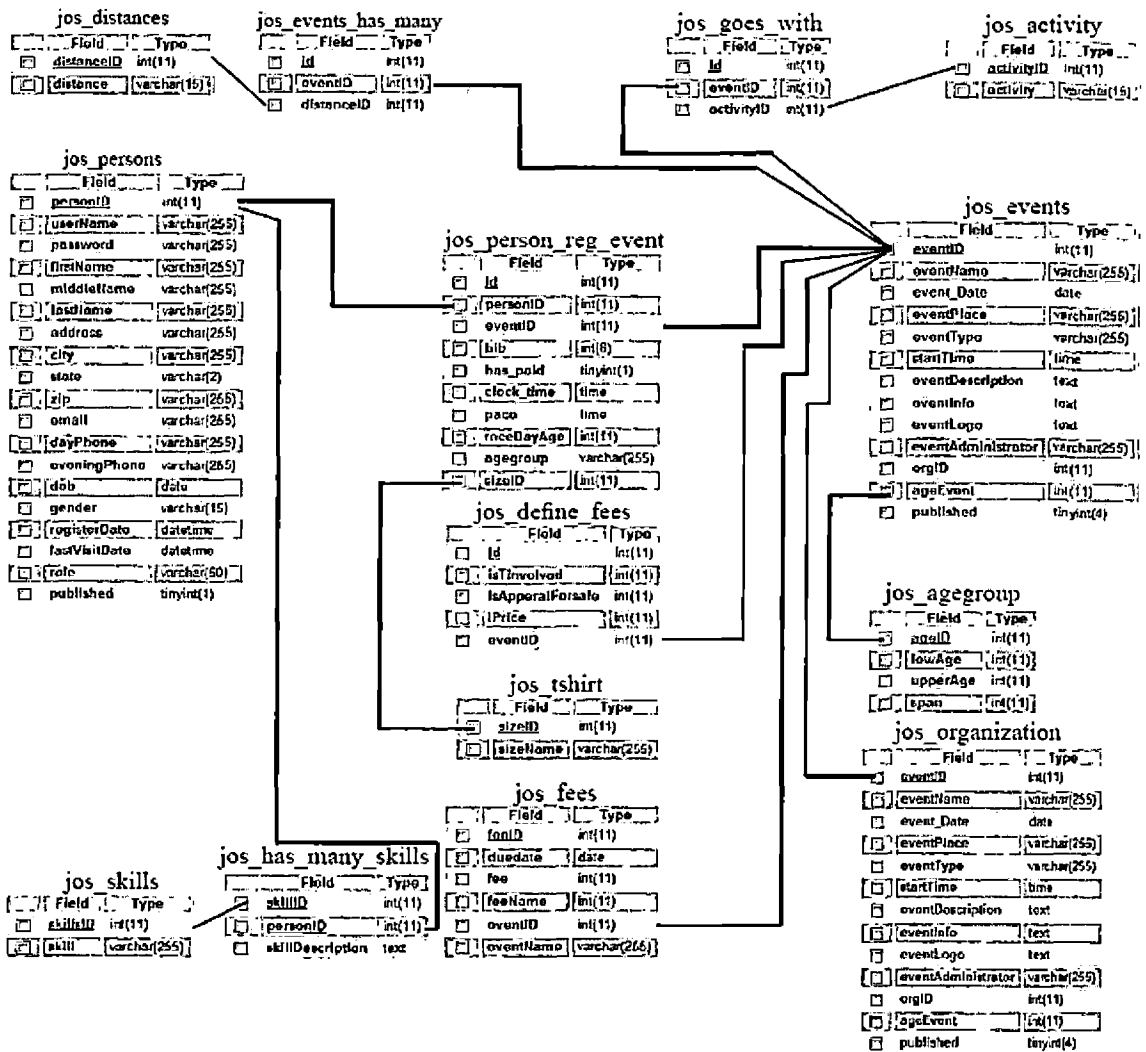


Figure 9. Table Layout

The table `jos_tshirt` is connected to `jos_person_reg_event` because participants might not choose the same size t-shirt every time they register. In the table `jos_events`, `ageEvent` is foreign key for `ageID` in the `jos_agegroup` table. The reason for this design is to link an event to a table where the data needed to calculate age groups for an event is found. The rest of the relationships are regular one-to-many relationships that connect the tables to make the data available through queries.

3.4 Data Dictionary

The data dictionary gives a short explanation about each table and each attribute in the database.

Table 2. Data Dictionary

Table	Attribute	Description
<u>jos_persons</u>		Contains information about a runner, an administrator, or a volunteer
	personID(PK)	System Generated Auto Inrement.
	name	Composite Attribute. FName, MName, LName.
	address	Composite Attribute, Street, City, State, Zip.

	email	Several people can have same email address (family-email).
	phone	Multi Valued. Day Phone, Evening Phone.
	registerDate	datetime format 0000-00-00 00:00:00 This is the date and time when an online registration was performed
	lastVisitDate	datetime format 0000-00-00 00:00:00 This is the date and time when a user last logged in to the system
	role	Varchar string. Indicates if a person is a volunteer, administrator or runner.
	published	Tinyint. Will be used to make participants data available for an administrator.
<u>Volunteer</u>		<i>Volunteer is of type Person and inherits all the attributes that a person has.</i>
	skill	Text. Describing a person's skill, e.g.

		a person has experience in administering the timing for an event
<u>Administrator</u>		<i>Administrator is of type Person and inherits all the attributes that a person has</i>
	access	Access level to the system, Defined in "Joomla!" 1.5 There are seven different user groups in "Joomla!" 1.5 where each group has different access level.
<u>Runner</u>	gender	Varchar string value Male or Female.
	dob	Date format YYYY-MM-DD
<u>Online registrant</u>		<i>Online_Registrant is of type Person and inherits all the attributes that a person has.</i>
	userName	Name of a person and can only belong to one person.
	password	Encrypted password using "Joomla!" 1.5 encryption algorithm.
<u>Manual registrant</u>		<i>manual_registrant is of type Person and inherits all the attributes that a person has.</i>

<u>Jos person reg event</u>		Represents the many-to-many relationship between jos_persons and jos_event.
	id (PK)	System Generated Auto Inrement.
	eventID	Foreign key refers to the event that the person is registered for
	personID	Foreign key refers to a person
	bib	Integer. A bib numbr and a participant can have different bib for different events.
	has_paid	Tinyint. true 1 meaning PAID, false 0 meaning NOT PAID.
	clock_time	Format hh:mm:ss. Race time for a person in one event.
	pace	Format hh:mm:ss. Pace per mile for a person in one event.
	raceDayAge	Int. Derived attribute. Calculated from dob. It is used in the result service.
	Agegroup	Varchar. Derived attribute. Calculated from dob. People are competing in age groups in a race.
	sizeID (FK)	T-shirt size.
<u>jos events</u>		Contains information about an event such as where, when, type of event

	eventID (PK)	System Generated Auto Inrement.
	eventName	Varchar. String name of the event.
	event_Date	Format YYYY-MM-DD. Date when the event occurs
	event_Place	Varchar. Where the event is held.
	event_Type	Varchar. Description of the event type.
	startTime	Time. hh:mm.
	eventDescription	Text. Any info about the event.
	event_logo	Text. Upload of logo.
	eventAdministrator	Varchar. Name of administrator.
	orgID (FK)	Foreign key from jos_organization table.
	ageEvent (FK)	Foreign key from jos_agegroup table.
	published	Tinyint. For publishing flyer online.
<u>Jos organization</u>		Contains information about an organization such as name, address, phone, contact person etc
	orgID (PK)	System Generated Auto Inrement.
	orgLogo	Text. logo upload.
	orgName	Varchar. Organization name.
	orgAddress	Varchar. Organization address.
	orgCity	Varchar. Organization city.

	orgState	Varchar. Organization state.
	orgZip	Varchar. Organization zip.
	orgPhone	Varchar. Organization phone.
	orgContact	Varchar. Contact person in the Organization.
	published	Tinyint. Organization published. yes 1 no 0.
<u>Jos tshirt</u>		This table contains t-shirt sizes.
	sizeID (PK)	System Generated Auto Inrement.
	sizeName	Varchar. Small,Medium, Large, X-Large, XX-large.
<u>Jos activity</u>		Contains information about different activities such as run, walk, run/walk etc.
	activityID (PK)	System Generated Auto Inrement.
	activity	Varchar. Indicates the activity involved in the event e.g.Run, Walk etc.
<u>Jos goes with</u>		<i>Represents the many- to-many relationship between jos_activity and jos event.</i>
	id (PK)	System Generated Auto Inrement.
	eventID (FK)	Refers to the jos_events table.
	activityID (FK)	Refers to jos_activity table.
<u>Jos distances</u>		contains information about the different

		distances such as 5k, 10k, half marathon etc.
	distanceID (PK)	System Generated Auto Inrement.
	distance	Varchar. 5k, 10k etc.
<u>Jos event</u> <u>Has many</u>		<i>Represents the many-to-many relationship between jos_distances and jos_events.</i>
	id (PK)	System Generated Auto Inrement.
	eventID (FK)	Connect to jos_event table.
	distanceID (FK)	Connect to jos_distances.
<u>jos define fees</u>		<i>Defines the fees for an event based on dates prior to the event.</i>
	id (PK)	System Generated Auto Inrement.
	isTinvolved	Int. 1 if t-shirt is involved in the fee, else 0.
	isApparelForsale	Int. 1 if yes, else 0.
	Tprice	Int. Price of t-shirt if not included in participation fee.
	eventID (FK)	Refers to jos_events table.
<u>Jos skills</u>		Contains information about different skills such as timing, refreshment, traffic regulations etc
	skillID (PK)	System Generated Auto Inrement.

	skill	Varchar. Type of skill. E.g. timing, refreshments, registration etc.
<u>Jos has</u> <u>Many skills</u>		<i>Is a relationship to take care of many-to-many relationship between jos_skills and jos_persons. A person has many skills and a skill can be possessed by more than 1 person</i>
	skillID (PK)	Int. Refers to jos_skills table.
	personID (PK)	Int. Refers to jos_persons.

CHAPTER FOUR

PROJECT DESIGN AND SOLUTIONS

4.1 Project Analysis

The background for this project is based on personal experience gained over several years either as a participant at local running events or as a race coordinator. All the small to medium-sized events have used a timing system based on writing the time on a slip from the bib number. Those slips are then placed on a board in the order a runner finishes the race. This system causes the results to be delayed several days before they are ready to be published. It also causes the award ceremony to be delayed because it is hard to single out winners in each age group. Another experience gained over the years with local small to medium-sized events is the way the race is announced. The sponsors mainly send out an event flyer and registration is done by mailing in the registration form together with a check to pay for the event.

Experience as a race coordinator has given me insight regarding how important the result service is to the participants. They require timely and correct timing and if there are faults with the service, runners are easily

ticked off and are most likely not to show up again. Finally, costs related to printing flyers and mailing them out to participants makes up a big part of the expenses for an event.

With all these things in mind, I decided that it is time to create a system that will handle all the issues related to a small to medium-sized event. The following points are taken into consideration. (APPENDIX A and APPENDIX B shows complete source code for the system.)

1. Online advertisement
2. Management of registered participants
3. Easy race timing of the event
4. Online registration and payment
5. Online searchable results
6. Ability to send email to previous registrants
7. Multi organizational system

4.2 Online Advertisement

Before an organization can advertize an event on the web, they have to register an organization that is in charge of the event. The registration of the organization is done at the "Back End's" control panel and is marked as

STEP 1. Figure 10 shows the "Back End's" control panel and where to start to register an organization.

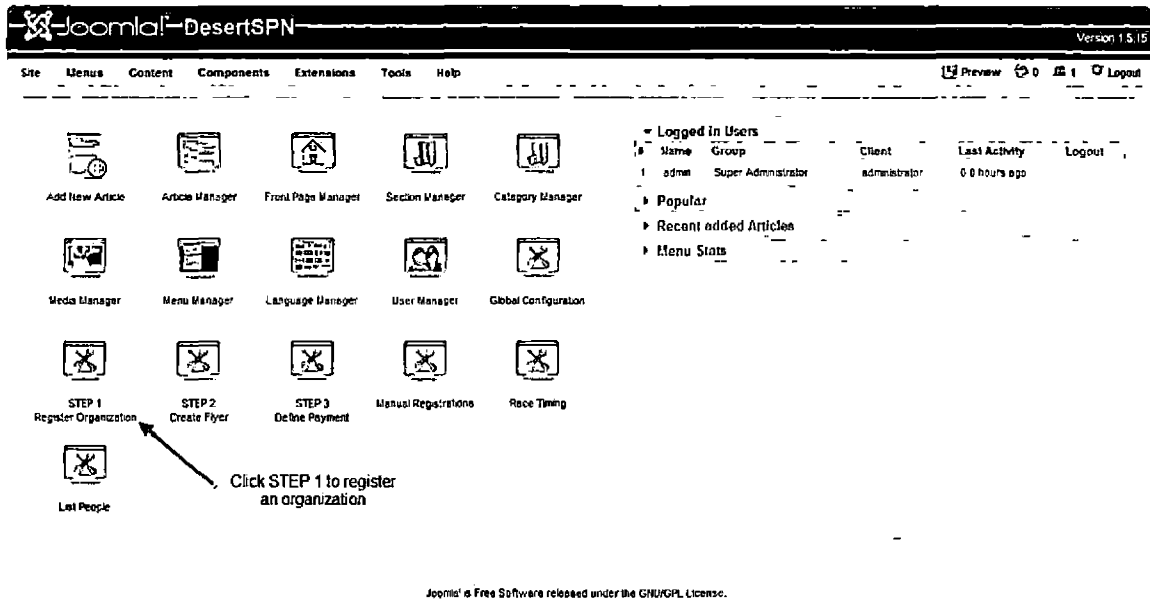


Figure 10. Step 1. Register Organization

When the icon for STEP 1 has been clicked, a new window appears on the screen. On this window a list of organizations is shown. To register a new organization click the "New" button in the upper right hand corner. To edit information of an already registered organization, click the "Edit" button up on the right hand corner, or

click on the organization's name in the list. The name has been converted to a link that performs the same task as if the "Edit" button is clicked. Figure 11 demonstrates the features in the "List Organization" window. To navigate back to the control panel, simply click on the "Site" link in the upper left hand corner and choose "Control Panel".

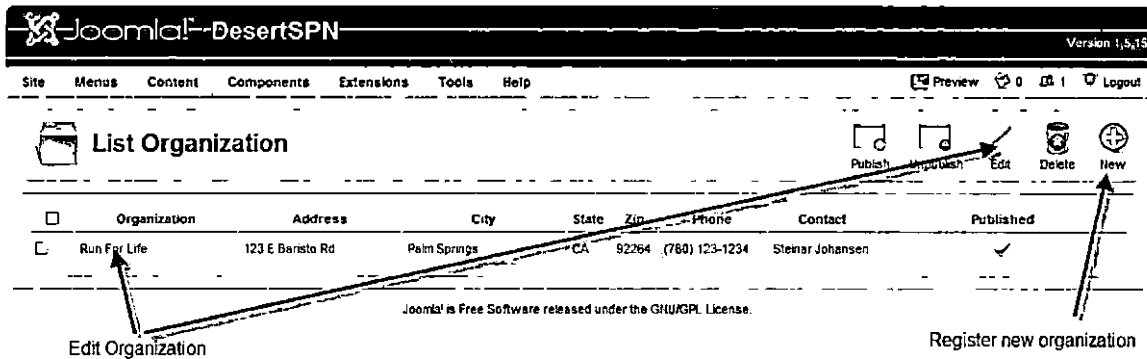
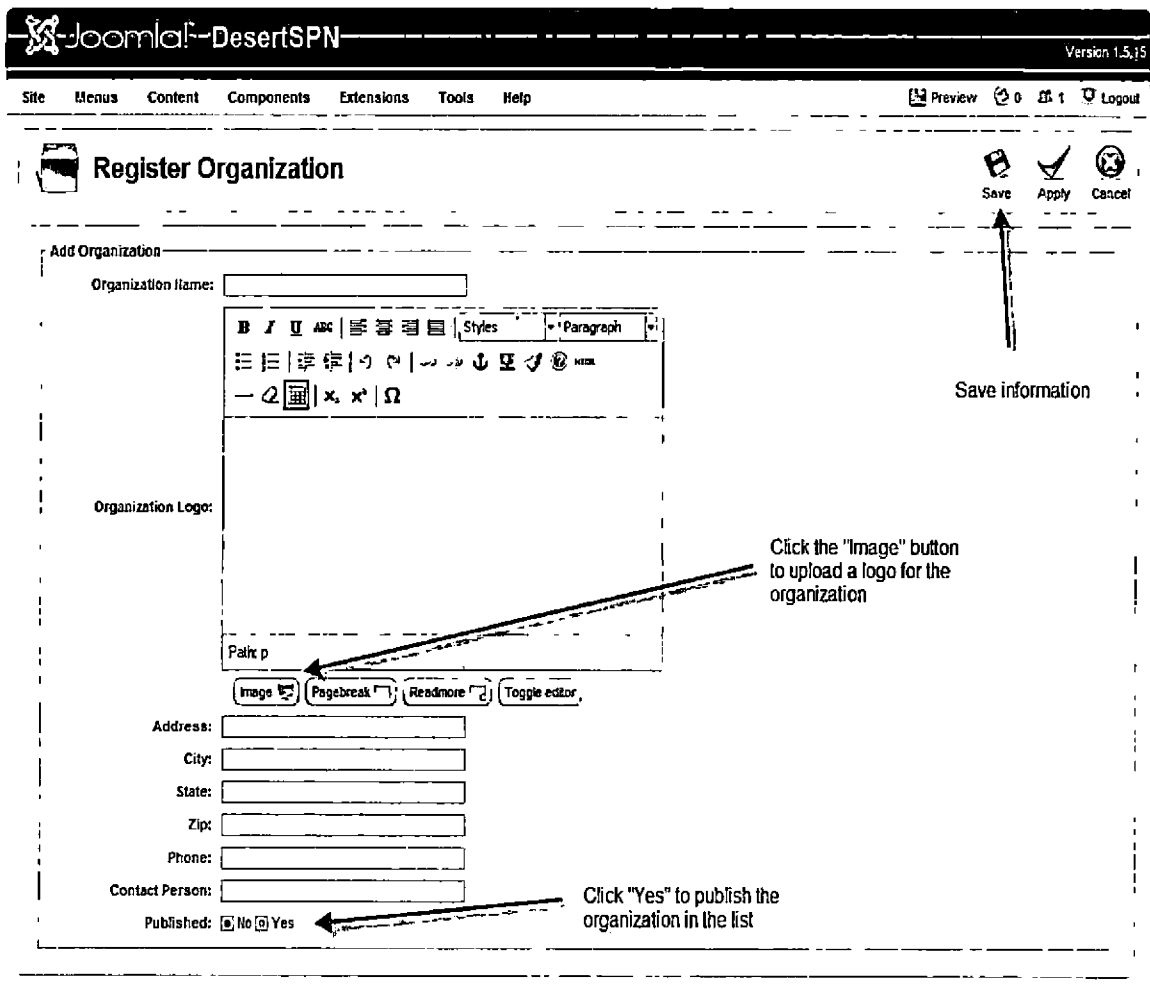


Figure 11. Register New or Edit Organization

When you click on the "New" button the "Register Organization" window appear. In this window, information such as organization name, address, city, state, zip, phone, contact person has to be entered into the form. The user can also upload a logo and choose to publish the organization in this window as well. Once all the

information is entered into the form, click the "Save" button in the upper right hand corner to save the information. Figure 12 shows what the "Register Organization" window looks like.



Joomla! is Free Software released under the GNU/GPL License.

Figure 12. Save Organization

Figure 13 shows the folder structure of the organization component. The component demonstrates a full featured MVC style component with two views.

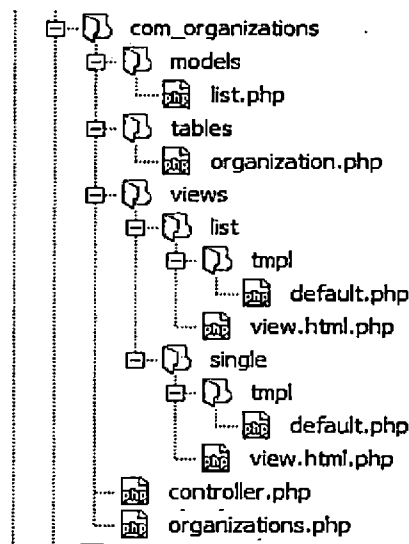


Figure 13. com_organizations

The entry point to the component is in the display function in the controller. If the controller doesn't detect a view value, the default view "list" is set. As seen from figure 13, a model named list is present. That model has the same name as one of the two views, and "Joomla!" 1.5 will automatically connect them. The list view's view.html.php file will retrieve the data directly

from the model by using a `get('variable name')` function. This data is transferred to the presentation file, `default.php` to show a list of organizations registered in the system.

The toolbar buttons present are laid out in the `organization.php` file and when for instance the "Edit" button is pressed, the task `edit` is sent to the controller that sets the view to `single` and populates the form with data from the record that has been checked. If the "Add" button is pressed, the task value `add` is sent which sets the view to `single` and shows an empty form.

When the form has been filled out and the "Save" button on the toolbar is pressed, the controller's `save` function first checks if this is a record that has an `id`, meaning it is an update and not a save. If it is an update, the post data will be "binded" and saved. If there is no `id` value with the post data, the next sequential `id` is picked and the data are stored. The `bind()` function and `store()` function are built-in functions in "Joomla!" 1.5

When the data about the organization are stored in the system, proceed to STEP 2, which is the design of the flyer for the web site. From the control panel, click the STEP 2 button. Figure 14 shows the "List Event Flyers"

Joomla! - DesertSPN Version 1.5.15

Site Menus Content Components Extensions Tools Help Preview 0 1 Logout

List Event Flyers Publish Unpublish Edit Delete New

<input type="checkbox"/>	Event Name	Event Date	Event Administrator	Event Organization	Span	Lower Age	Upper Age	Distance	Activity	Published
<input type="checkbox"/>	Run For Life	2010-05-01	Stenar Johansen	Run For Life	5	15	80	5k (3.1 Miles)	Run	✓

Joomla! is Free Software released under the GNU/GPL License

Figure 14. List Event Flyers

From this window there are two options. Either add a new flyer, or edit a flyer that is already in the system. Figure 15 shows the form used to design an event flyer.

Joomla! - DesertSPN Version 1.5.15

Site | Menu | Content | Components | Extensions | Tools | Help Preview | 0 | 1 | Logout

Create an Event Flyer

Save | Apply | Cancel

Create Flyer

Path: p

Event Logo:

Event Name:

Event Date:

Event Place:

Event Type:

Event Description:

Event Intro:

Organizer:

Published: No Yes

Add Age Group

Define Lower Age: Define Upper Age: Define Age Span: Years

Define event type

1 Mile 2 Miles 5k (3.1 Miles) 10k (6.2 Miles) Half Marathon Marathon

Define activity

Run Walk Bicycling Triathlon Run/Walk Swimming

Joomla! is Free Software released under the GNU/GPL License

Figure 15. Create an Event Flyer Form

On the "Create an Event Flyer form", the information entered will be stored in a few different tables. Together

with the regular information about the event, information needs to be entered for use to calculate age groups and pace for a race. The age group information is based on a low age, an upper age and an age span. The information used to calculate pace is to specify what kind of race it is.

The event description field and the event info field is a text editor with many options on how to format the information for the event. It accepts images and text can be wrapped around images. The flyer is saved by clicking on the "Save" button in the upper right hand corner.

Let's take a look at how the flyer component is designed. This component is a full featured MVC component. It has two views, two models, a controller and it supports access to four different tables in the database. Figure 16 illustrates the folders and files involved in this component.

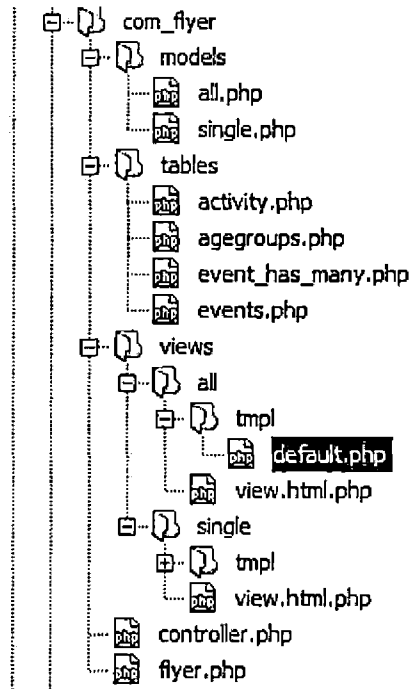


Figure 16. com_flyer

The entry point is in the controller. Since no view variable is set, the controller sets the view to all. This view shows a list of the flyers that has been designed for the web. The "all" model is creating an array of data that is used to populate the table in the view. If the "New" button is clicked, the single view is invoked which shows the form used to create the flyer. If you check the box in front of a flyer name and click edit, that specific single view will appear with all the fields populated ready to be edited.

The radio buttons in the form are created from database queries and populated using the "Joomla!" 1.5 built in function to create radio buttons. There are different types of radio buttons and in "Joomla!" 1.5 and in this form "select.booleanlist" and "select.radiolist" are used.

Another variable that is used in almost every components is the \$cid variable. That variable is used when a record is updated or edited. It stores the id from a record that is present in the view and keeps track of it. When the controller has to make a decision whether to update or save a record it checks if there is a cid present.

Once the flyer is designed, proceed to STEP 3 which is to define the payment for the event. The first view in this component instructs the user to select an event. Figure 17 shows what the select event view looks like.

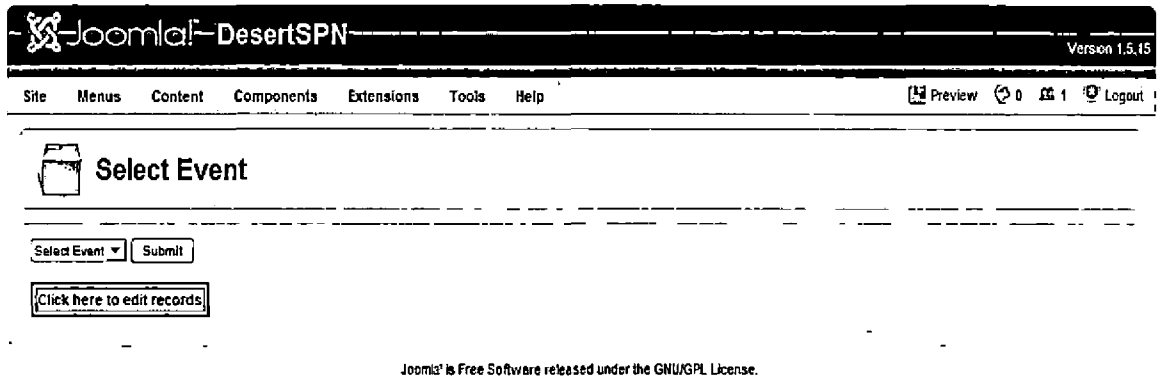


Figure 17. Select Event View

Clicking submit takes the user to the define fees page. In this view they have to specify the registration fees. Normally there are different fees depending on when a participant signs up. On site, same day registration is normally more expensive than the regular fee. Furthermore, some events might have an early bird special to motivate participants to sign up early. Even a very early bird special might be available but this does not happen very often. This system is prepared for four different fees if an event chooses to use them all. Figure 18 shows what the define fees page looks like.

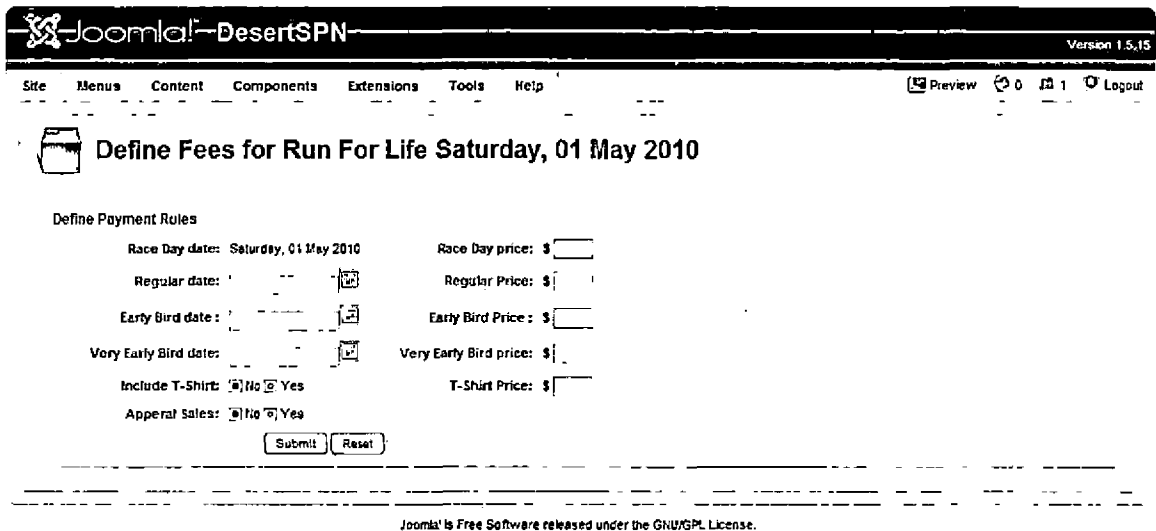


Figure 18. Define Fees View

Also, in this page the user has to specify whether a t-shirt is included in the fees or not. In case a t-shirt is not included in the fee, the user has to specify the price for the t-shirt. This is used later in the registration where a participant can select whether to buy a t-shirt or not.

To edit payment information, click on the button "Click here to edit records". The "all" view which is called "List Dues and Payments" is shown. This page allows the user to either select an event and click the edit button or to click on the event name for the record the user wants to edit. This takes the user to the edit page.

Figure 19 shows the layout for the "List Dues and Payments" page.

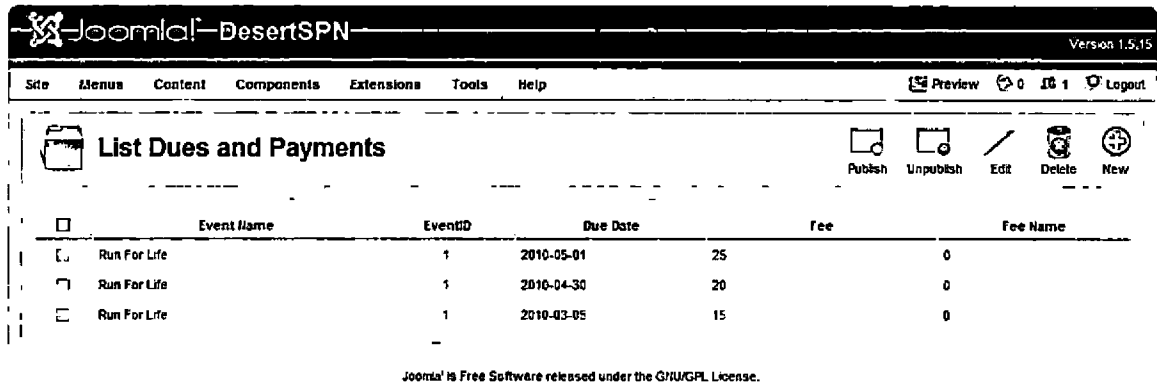


Figure 19. List Dues and Payments Page

When selecting the payment to edit, the user is taken to the "Edit Dues and Payments page. The only part that can be edited is the date and the price field. The other fields are not editable. The calendar used in this page and many other pages is a MooTools calendar. MooTools is a light weight JavaScript based on object oriented design. Figure 20 shows the "Edit Dues and payments" page.

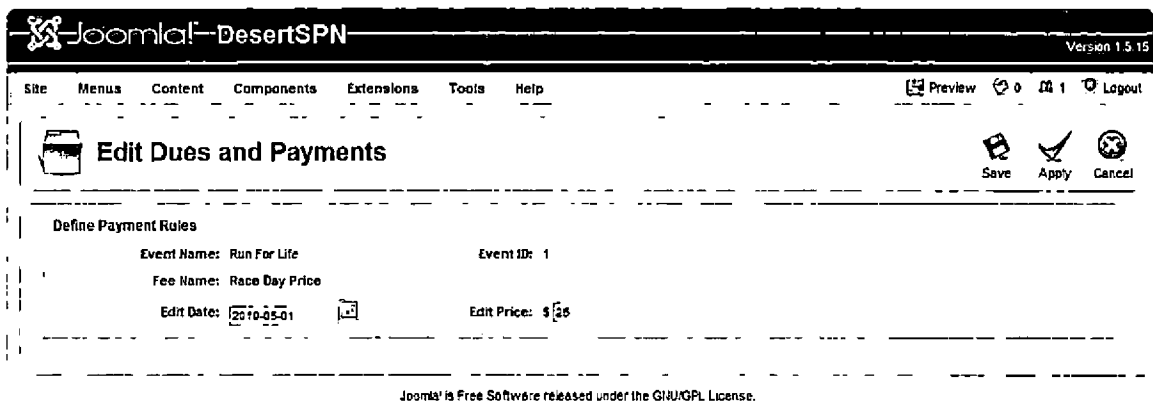


Figure 20. Edit Dues and Payments Page

The "payments" component is designed as follows. This component has five views, three models, a controller and it supports two different tables. Figure 21 illustrates the design of the "payments" component

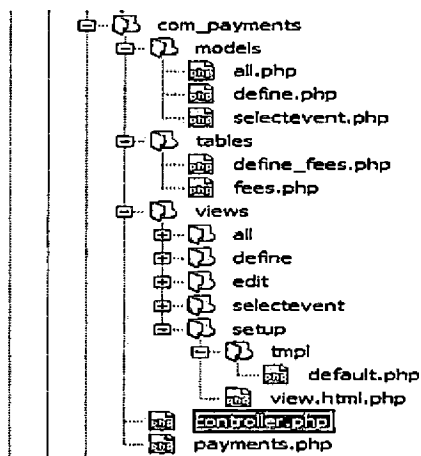


Figure 21. com_payment

The entry point in the controller sets the view to "selectevent" because the view variable is not set. This view features a dropdown list from which the user can choose which event to define the payments. There is also a button to click to "edit" payments. When event is chosen and the "submit" button clicked, the select function in the controller will set the view to "define". It also checks that "eventID" is present. If it is not, a redirection back to "selectevent" is performed so an event can be chosen. The "define view" has a form where the event date is filled out. The rest of the fields are filled out by the administrator. The define() function in the controller gets the post data and saves it, eventually updates it and when the operation is finished the user is redirected back to the "selectevent" page.

If the update button is pressed, the view variable is set to "all" and a list of the events payment dates is shown. From the list, either click on the event name or check the record that is to be updated. The view is set to edit and when the change is submitted by clicking the save button, the save() in the controller will update the record based on the cid value. After the update, a redirect to the "selectevent" view is executed.

4.3 Management of Registrants and People

Management of people in the database can be done in several ways. There is one component, com_registrations that handles management of all categories of people in the database. Then there is another component, com_manualreg that handles management of registrants to a certain event. The "persons" component handles entries of people to the database that are not participants in an event. Those people are administrators and volunteers. From the control panel at the back, click on the icon called "List People". Figure 22 shows the control panel layout.

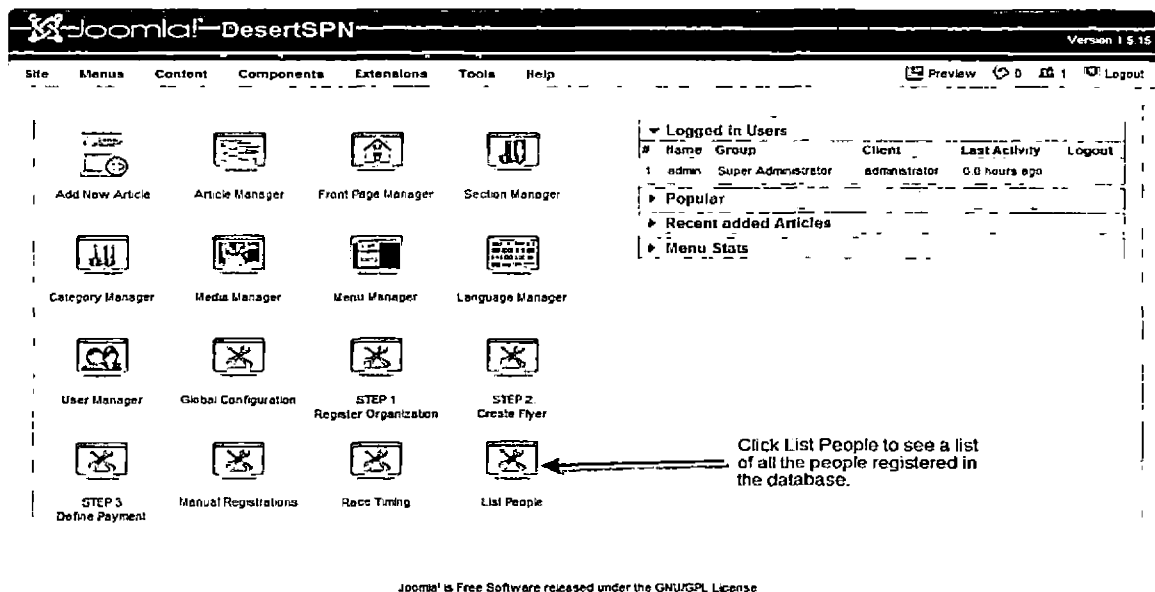


Figure 22. Control Panel and List People Icon

When the icon is clicked, the window listing all people appears. This page is using pagination and based on the number of records chosen in the pagination, that number of records will be shown. Figure 23 shows the "List All" page.

Joomla! - DesertSPN Version 1.5.13

Site Menus Content Components Extensions Tools Help Preview 0 1 Logout

List All **Edit Record** Publish Unpublish Edit Delete New

<input type="checkbox"/>	First Name	Last Name	Address	City	State	Zip	Gender	Email	Phone	Day of birth	Role	Last Visit	Published
<input type="checkbox"/>	Run	Runningsen	111 Somewhere	Rancho Mirage	CA	92270	male	asa@asa.as	(111) 111-1111	1953-06-06	Athlete		1
<input type="checkbox"/>	Areyou	Running	111 Somewhere	Santa Ana	CA	92701	female	asa@asa.as	(111) 111-1111	1957-01-18	Athlete		2
<input type="checkbox"/>	KidRun	Fast	111 Somewhere	Palm Springs	CA	92262	male	asa@asa.as	(111) 111-1111	1953-06-16	Athlete		3
<input type="checkbox"/>	HowFast	DidYouRun	111 Somewhere	Palm Springs	CA	92264	male	asa@asa.as	(111) 111-1111	1953-07-09	Athlete		4
<input type="checkbox"/>	IamAslow	Runner	111 Somewhere	Palm Springs	CA	92264	male	asa@asa.as	(111) 111-1111	1930-02-03	Athlete		5
<input type="checkbox"/>	HowOften	DoYouRun	111 Somewhere	Palm Desert	CA	92260	male	asa@asa.as	(111) 111-1111	1937-01-30	Athlete		6
<input type="checkbox"/>	Sily	Runner	111 Somewhere	Palm Springs	CA	92262	male	asa@asa.as	(111) 111-1111	1979-02-06	Athlete		7
<input type="checkbox"/>	Ihate	Running	111 Somewhere	Palm Springs	CA	92264	female	asa@asa.as	(111) 111-1111	1959-04-28	Athlete		8
<input type="checkbox"/>	RunningFor	Monney	Alcova Lake	Alcova	WV	67543	male	asa@asa.as	(111) 111-1111	1987-01-30	Athlete		9
<input type="checkbox"/>	Marathon	Runner	111 Somewhere	Palm Desert	CA	92260	male	asa@asa.as	(111) 111-1111	1975-08-22	Athlete		10

Select Display 10 Start Prev 1 2 3 4 Next End Page 1 of 4 **Create New Record**

Figure 23. List All Page

From this page a user can either edit a record or create a new record. The choices in the pagination are 5, 10, 15, 20, 25, 30, 50, 100 and all. The pagination is based on MooTool JavaScript, and the date for the pagination is created in the model. If a new record is to be added to the system, the "New" button in the upper right hand corner is clicked. A regular form is presented where standard data is collected. There is no restriction on what fields have to be filled out. The default role for the form is an athlete. But from the select list, a volunteer or administrator can be selected. Figure 24 shows the form.

Joomla! - DesertSPN Version 1.5.15

Site Menus Content Components Extensions Tools Help Preview 0 1 Logout

Administrare People

Save Apply Cancel

Create Login

Username:

Password:

Verify Password:

Create User Information

First Name:

Middle Name:

Last Name:

Address:

City:

State:

Zip:

E-Mail:

Day Phone:

Evening Phone:

Day of Birth (yyyy-mm-dd):

Gender: Male: Female:

Role:

Published: No Yes

Joomla! is Free Software released under the GNU/GPL License.

Figure 24. Administrare People Window

As figure 24 shows, there is nothing special other than a form to be filled out, and to save the record, click the "Save" button up in the right hand corner. If editing a record was selected, the form would be filled out with the information from the record chosen to be edited.

Let's take a look to what's going on in the registrations components. Figure 25 shows the folder structure for com_registrations.

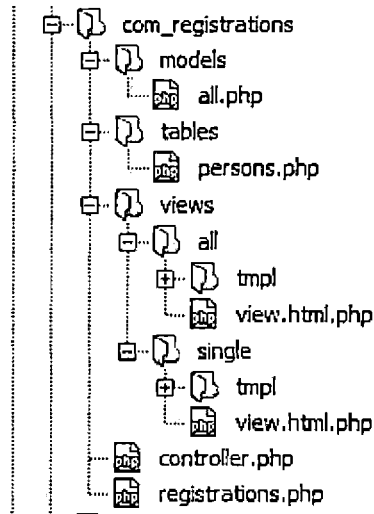


Figure 25. com_registrations

This figure is a full featured MVC component. It supports two views, "all" and "single". The all view has support from the "all" model and it operates the same way as the previous explained organization component. The only difference is that the "single" view supports a generic list to create a drop down for selecting role. This generic list is created as an array that is submitted to the parameter list to create a "Joomla!" drop down. A radio list for gender is also featured. This is also created using "Joomla!" 1.5 features. These features are created in the view.html.php file and are transferred to the default file using "Joomla!" 1.5 assignRef() function.

The other way to administer people is through the "manualreg" component when a participant wants to manually register for an event, or a bib number is assigned to a participant for a specific event. This administration includes a second component named mysearch. So to perform this administration, two components are involved. The manual registration component is accessible from the control panel on the "Back End" of the system. From the "Back End" control panel, click the "Manual Registration" button. This will take the user to the select event page. Figure 26 shows a picture of the "Select Event" page.

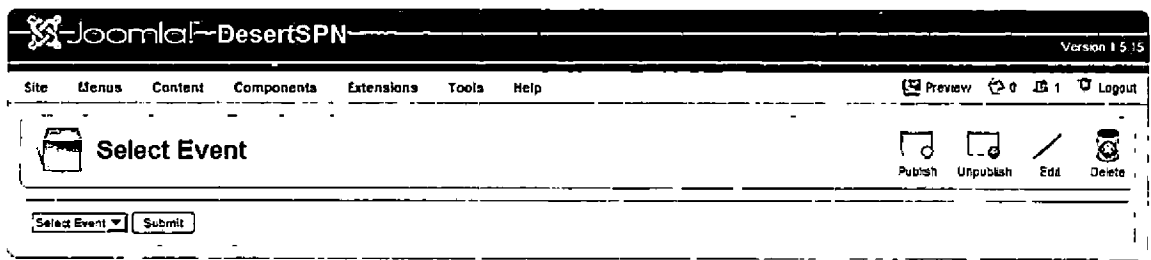


Figure 26. Select Event Page Manual Registration

To administer participants, select an event and click the submit button. This takes the user to the "Add

Participants" page. Figure 27 shows the layout for the page.

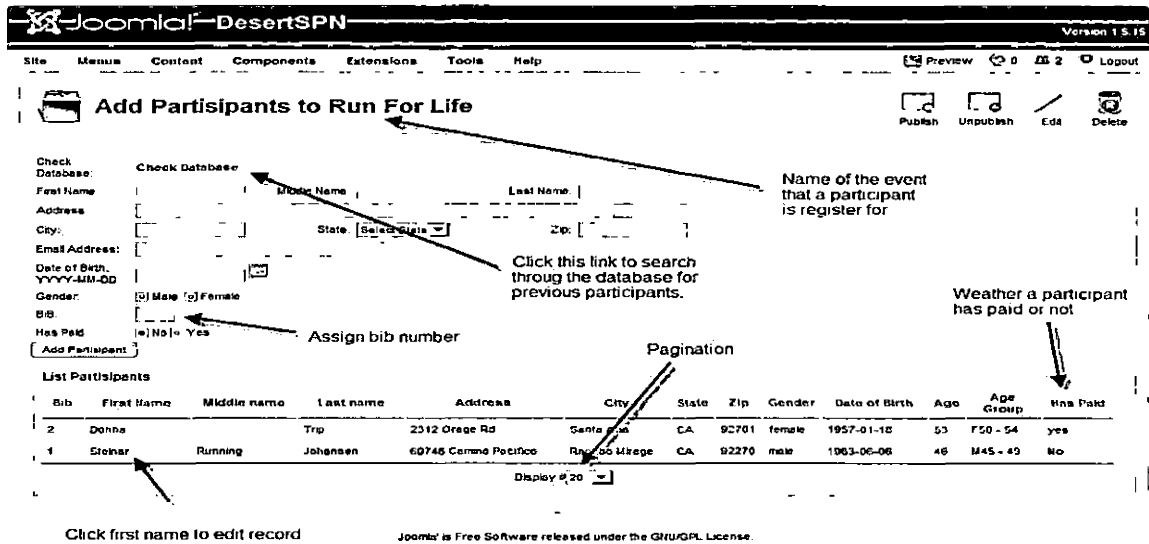


Figure 27. Add Participants Page

There are several things going on in this window. First it is the window that is being used to collect on site, same day registration, or registration of participants that have mailed in a registration form to register for an event. This is also the page that is being used to assign bib number to a participant who has registered online. Further, this is also the page where a

search through the database can be done to see if a registrant has been registered in the system before. If the person is found, the form will be filled in and a bib number is assigned and the registration is done. The list of registrants is listed in descending order of bib numbers. Normally, the next bib number assigned is higher than the previous one. By sorting the list in descending order each person registered can easily be controlled. The page is also using pagination to set up the number of records that want to be viewed. To submit the registration, click the button called "Add Participant"

To perform a search through the database, the "Check database" has to be clicked. Figure 28 shows the search database page.

Search: Click the name that is to be registered

<input type="checkbox"/>	Username	First Name	Last Name	Address	City	State	Zip	Gender	Email	Day of Birth
<input type="checkbox"/>	jp60nar	Steinar	Johansen	69746 Camino Pacifico	Rancho Mirage	CA	92270	male	jp60nar@dc.rr.com	1963-06-06
<input type="checkbox"/>	donna@trp	Donna	Trp	2312 CRAIGE Rd	Santa Ana	CA	92701	female	donna@trp.com	1957-01-18
<input type="checkbox"/>	ph@catala	Phil	Catala	2475 N Via Monte Vis	Palm Springs	CA	92262	male	ph@catala.com	1958-06-16
<input type="checkbox"/>	don@peterson	Don	Peterson	123 Baristo Way	Palm Springs	CA	92264	male	don@peterson.net	1958-07-09
<input type="checkbox"/>	john@johnson	John	Johnson	222 E Ramon Rd	Palm Springs	CA	92264	male	john@johnson.net	1930-02-03

Display # 5 Start Prev 1 2 3 4 5 6 Next End Page 1 of 6

Joomla! is Free Software released under the GNU/GPL License.

Figure 28. Search Database Page

From the list in this page, click on the first name to add the participant to the event. As the database grows bigger, it might be harder to find a person. That's where the search field comes in that is located in the upper left hand corner. This search field can search for any one attribute at the time. That is, search can be completed by first name, last name, address, city, zip etc. But only one field at the time. In most cases by searching by last name for example, this narrows the search to easily find what the user is looking for. Figure 29 shows a picture of a search for people living in San Bernardino.

Joomla! - DesertSPN Version 1.5.15

Site Menus Content Components Extensions Tools Help Preview 0 1 Logout

Search:

<input type="checkbox"/>	Username	First Name	Last Name	Address	City	State	Zip	Gender	Email	Day of Birth
<input type="checkbox"/>	jonpeters	John	Peters	2843 Cliff Dr	San Bernardino	CA	93503	male	john@peters.com	1927-01-02
<input type="checkbox"/>	peterhanson	Peter	Hanson	3432 42nd Ave.	San Bernardino	CA	92503	male	peter@hanson.com	1965-06-06
<input type="checkbox"/>	petjohnson	Pet	Johnson	123 Somewhere	San Bernardino	CA	93405	male	pet@johnson.com	1979-01-09

Display # 3

Joomla! is Free Software released under the GNU/GPL License

Figure 29. Using the Search Field

As seen from figure 29 the search was able to reduce the table to show only three possible records. It should then be easy to single out the record the user is looking for and whether the person is a previous registrant. Figure 30 shows the two components that are involved to perform the management of a participant.

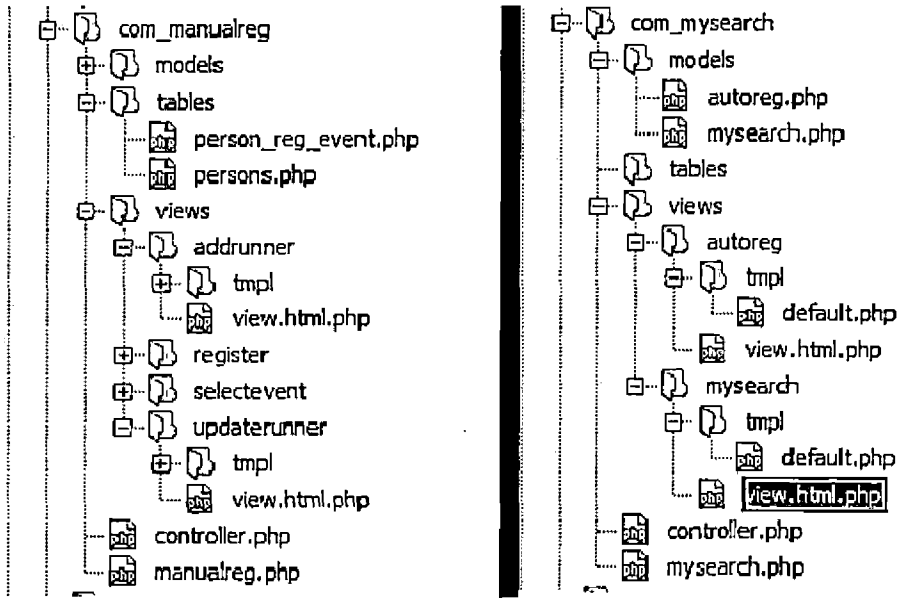


Figure 30. com_manualreg and com_mysearch

The entry point is in the controller of the com_manualreg. Since no view variable has been set, the "selectevent" view is set. The controller's select() function checks if there is an "eventID" present. If not, the "selectevent" view is set. If there is an "eventID" variable present, the view is set to "addrunner". The POST data from the "addrunner" is handled in the saverunner() in the controller. This function also calls the getAgegroup() function which calculate the age group for a person based on the date of birth variable.

If a record is being updated, the `edit()` function in the controller sets the view to "updaterunner" and the `cid` variable selects which record to use to fill in the form. If the action is to register a new record, the `add()` is invoked which set the view to "updaterunner". The `save()` in the controller handles the update of a record.

If the link "Check database" is clicked, a redirect to the `com_mysearch` component is performed using the "Joomla!" 1.5 built in `JRoute()` function. The entry point is in the controller of this component. The view is set to "mysearch". The model's `buildSearch()` is setting up the fields that the search field will accept. For this search, the search can be done for all the fields in the form related to a person. When the first name is clicked, the control is moved to the "auroreg" view which will populate the form and when the "Save" button is clicked, the control is moved back to the "manualreg" component. When there is a transition between the two components, the "eventID" variable is also sent to keep track of the event that is being managed.

4.4 Race Timing

The system has a race timing component built in to provide correct timing and almost spontaneous results for an event. From the "Back End's" control panel click on the "Race Timing" button to enter into the timing component. Figure 31 shows a picture of the control panel and where the "Race Timing" button is located.

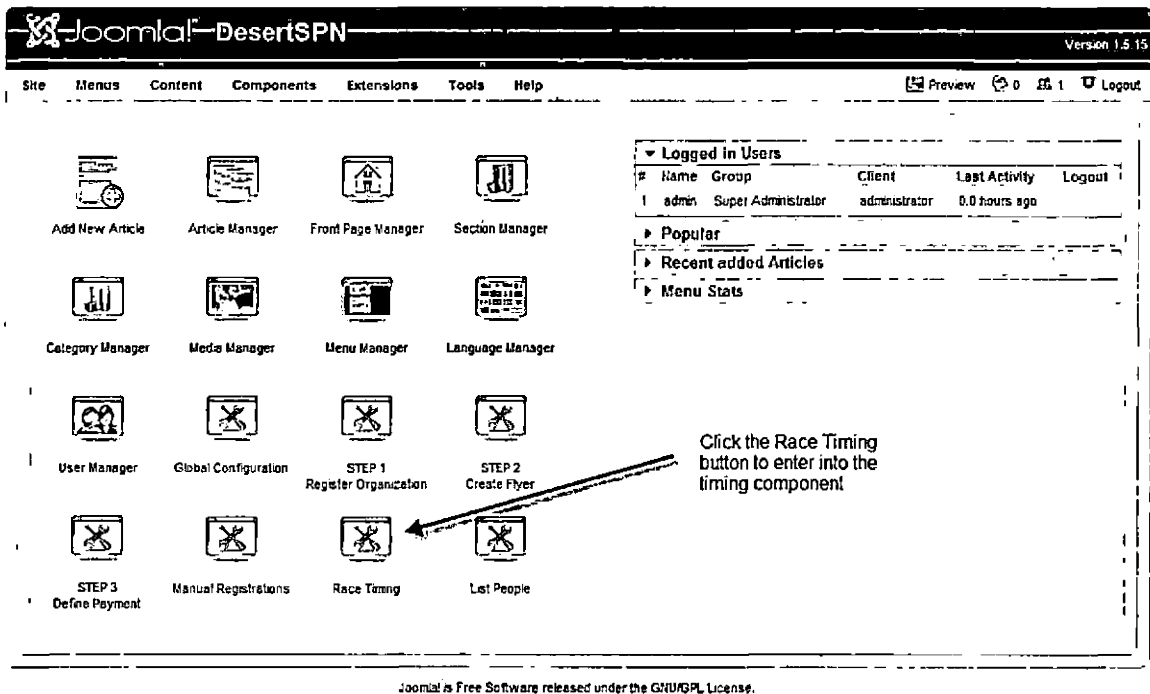


Figure 31. Control Panel, Race Timing Button

When the "Race Timing" button has been clicked, a new page appears to select an event for which to perform the race timing. This page features only a drop down select component and a submit button. Figure 32 shows a picture of the Select Event page.

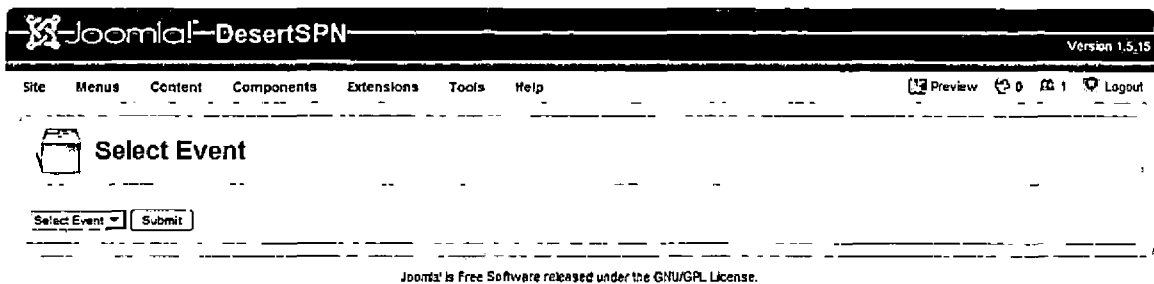
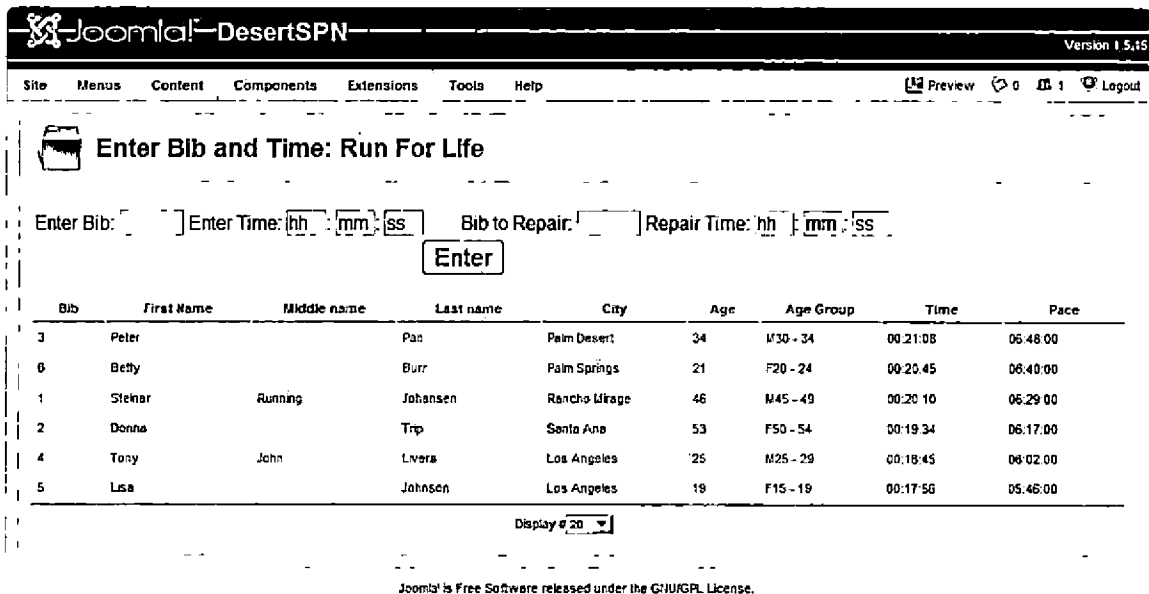


Figure 32. The Select Event Page

When an event has been selected, and the "Submit" button is pressed, the user is transferred to the "Enter Bib and Time" page. This page has three parts to it. Part one is the area where to enter the bib and time. Part two is where to repair the time for a bib if there was a wrong time entered for a bib. Part three is where there is a list of all the participants in this event who have gotten a time, with the slowest time first. The last bib that got a

time is more likely to be a slower time than the previous time entered to the system. By sorting the list in descending order, the last time given to a bib can easily be controlled that it was entered into the system. Figure 33 shows the layout and how the page is organized.



Enter Bib and Time: Run For Life

Enter Bib: Enter Time: : : Bib to Repair: Repair Time: : :

Bib	First Name	Middle name	Last name	City	Age	Age Group	Time	Pace
3	Peter		Pat	Palm Desert	34	M30 - 34	00:21:08	06:48:00
6	Betty		Burr	Palm Springs	21	F20 - 24	00:20:45	06:40:00
1	Steinar	Running	Johansen	Rancho Mirage	46	M45 - 49	00:20:10	06:29:00
2	Donna		Trp	Santa Ana	53	F50 - 54	00:19:34	06:17:00
4	Tony	John	Lvera	Los Angeles	25	M25 - 29	00:18:45	06:02:00
5	Lisa		Johnson	Los Angeles	19	F15 - 19	00:17:56	05:46:00

Display 0/20

Joomla! is Free Software released under the GNU/GPL License.

Figure 33. Enter Bib and Time Window

Figure 33 demonstrates a list organized in descending order based on the race time. The pace is calculated based on the given hours, minutes and seconds by converting the distances into meters and converting the time into seconds.

Using the metric system gives the pace a little bit more accurate time.

Figure 34 illustrates the folder structure of the timing component. It is a full featured MVC component with two views, two models and a controller. This component does not have any direct support of any tables.

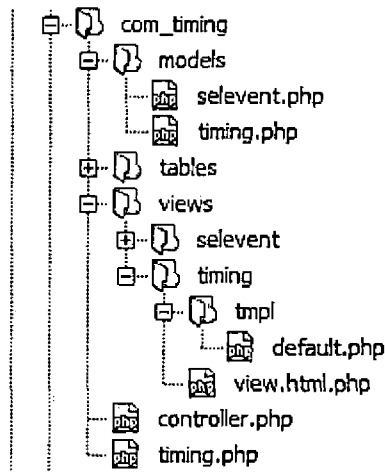


Figure 34. com_timing Folders and Files

The entry point is in the controller where the view variable is set to "selectevent". When the user has selected an event and the submit button is pressed, the select() checkid an "eventID" is present. If not, the Select Event page is shown again. If an "eventID" is

present, the timing view is selected. The POST data from this view is all handled in the save() function in the controller. The first check is to see if a time 00:00:00 has been entered. If so an error message is sent back saying that the time cannot be 0. The controller is supported with several functions that for instance check if a valid bib is entered, whether minutes and seconds are higher than 59, whether a bib has already gotten a time, calculating pace, and finally a function that saves the time. When the return to the enter bib page occurs, the "eventID" is also sent to maintain the info about the event for which the user is entering the times.

In case an update of a time is necessary, the system performs the same tests to detect illegal values.

4.5 Event Advertisement and Registration

Two of the better improvements this system provides for small to medium-sized events are the ability to advertise the events online and collect online registrations to the event. This component is located in the system's "Front End". To access the event advertisements and online registration a registrant will have to click the link on the left hand corner saying "List

Running Events". Figure 35 shows the outline of the start page and where the link is to the events advertised on the web.

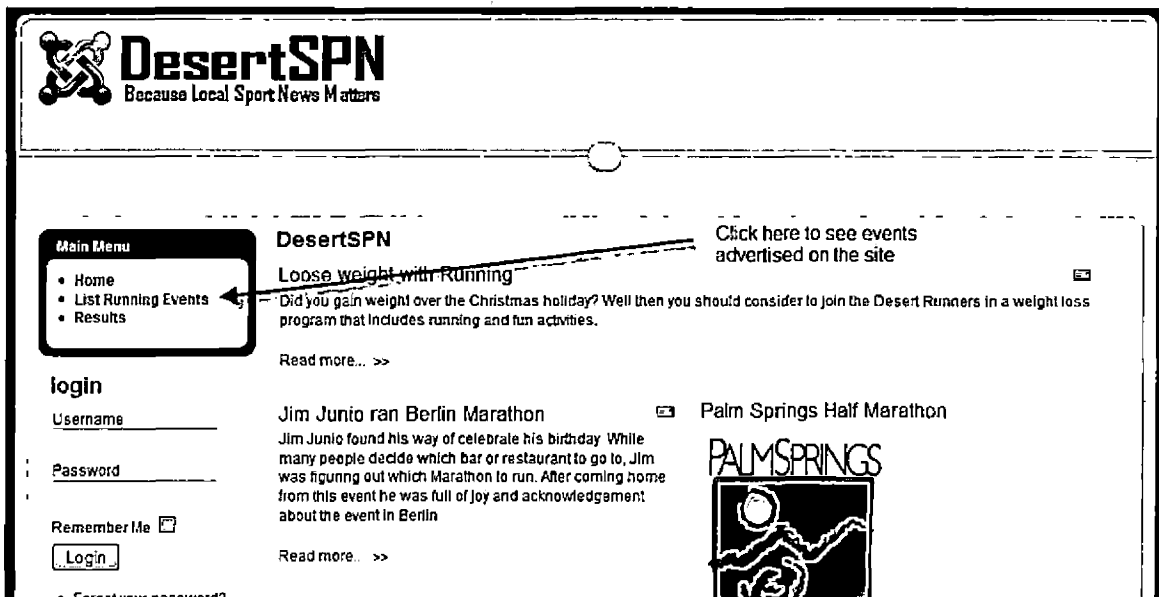


Figure 35. Front End, Start Page

The entry page to the web features a news portal as well as links where participants can browse event flyers or check out results for events that have been held. By clicking on the link "List Running Events" a list of events will show up. The list shows the logo for the upcoming events, with the event coming up in the shortest time on

the top of the list. Figure 36 shows how the Local Running Events page looks. It is suggested that the race coordinators create an event logo that is 500 X 118 pixels to fit a standard, and that the logo has the information about the time and where the event is being held.



Figure 36. Local Running Events Page

The logos that represent an event are turned into a clickable link. So by clicking on one of the logos the

event flyer will appear which is also where a participant can choose to sign up. The main menu to the left in the window will be present in all the pages through the web. The flyer that appears has the same standard. However the person creating the flyer has a lot of tools to use in the text editor provided during the step of creating the flyer. Text can be formatted. Pictures can be inserted. Text can be wrapped around pictures. It is an advanced text editor that "Joomla!" 1.5 provides. Figure 37 shows an example of a race flyer.

Main Menu

- Home
- List Running Events

Palm Springs Half Marathon Thursday, 01 April 2010

Palm Springs Half Marathon
Palm Springs April 1, 2010



Event Type: Running event
Date: Thursday, 01 April 2010
Event Place: Ruth Hardy Park
Event Administrator: Greg Klein
Event Organization: Greg Klein Sport
Sign up for Event: [Click Here](#)

Fast, Scenic USATF Certified Course

New Half Marathon Course

Due to the growth of the Half Marathon, we had to change the course. We kept all the beautiful parts and both the Half Marathon and 5K courses have been USA Track and Field certified.

Take the challenge and run 13.1 miles on a mostly flat and very fast course which starts at Ruth Hardy Park and winds through the beautiful Las Palmas area of Palm Springs. With the awesome San Jacinto Mountains looming above, you will pass by many celebrities' homes such as EMS Presley's and Marilyn Monroe's hideaway.

Half Marathon Relay

The first leg 5.4 miles (this is the hardest and longest of the three legs), the second leg 3.5 miles and the third leg 4.2 miles. The relay tag area for both tags will take place at the corner of Alejo and Vine at the Jewish Community Center north lot. Two person teams can either have one person run two legs or each person run one leg and then run the final leg together.

Sweetheart Division

In addition to the other relay team divisions, we will be offering a sweetheart division for the relay portion of the event. In order to be in the sweetheart division, you must be a two person team of either husband/wife, boyfriend/girlfriend, father/daughter or mother/son. First person will run the first leg 5.4 miles, the other person will run the second leg 3.5 miles and then both persons will run the final 4.2 mile leg and cross the finish line together.

Directions

To Ruth Hardy Park take I-10 to the Hwy 111/Palm Springs exit, go 12 miles to Tachevah, go left to Miralaste and then right. The park is on the corner of Miralaste and Tamarisk.

Registration & Fees

You can register by filling out the entry form in this application and mailing it with a check or you can register online with a credit card at [adve.com](#). Race morning registration will be available at Ruth Hardy Park starting at 5:30 a.m.

Download the event flyer: [here](#)

Half Marathon

\$47* before Jan. 15, 2010

\$55* Jan. 16 to Feb. 13

\$50* Race day

*Seniors 55+ and Kids 14 & under; \$5 discount

[Sign Up](#)

[return to Event List](#)

Figure 37. Race Flyer

In the race flyer, the logo picture from the list of events is inserted to make registrants confident that they are seeing the wanted flyer. A sign up link is provided both on the top and the bottom of the flyer. By clicking on either one, the login window will appear. On this window a participant can login using a username and password if they have registered for an event before. Otherwise the "REGISTER" link is clicked to register in the system.

Figure 38 shows the login window

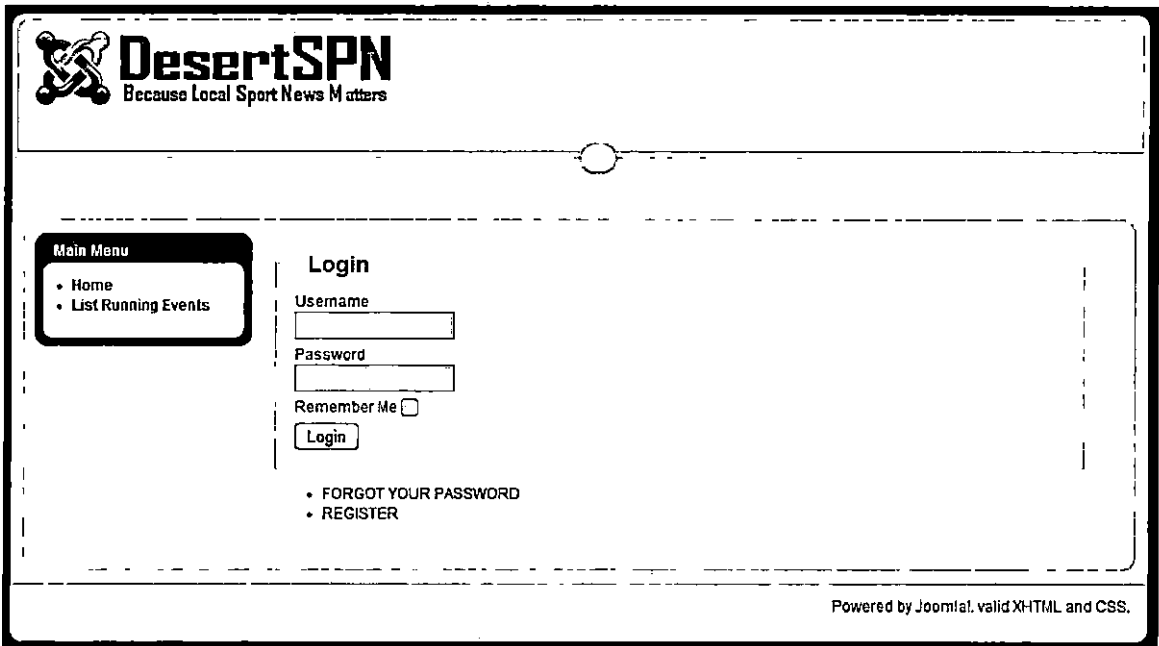
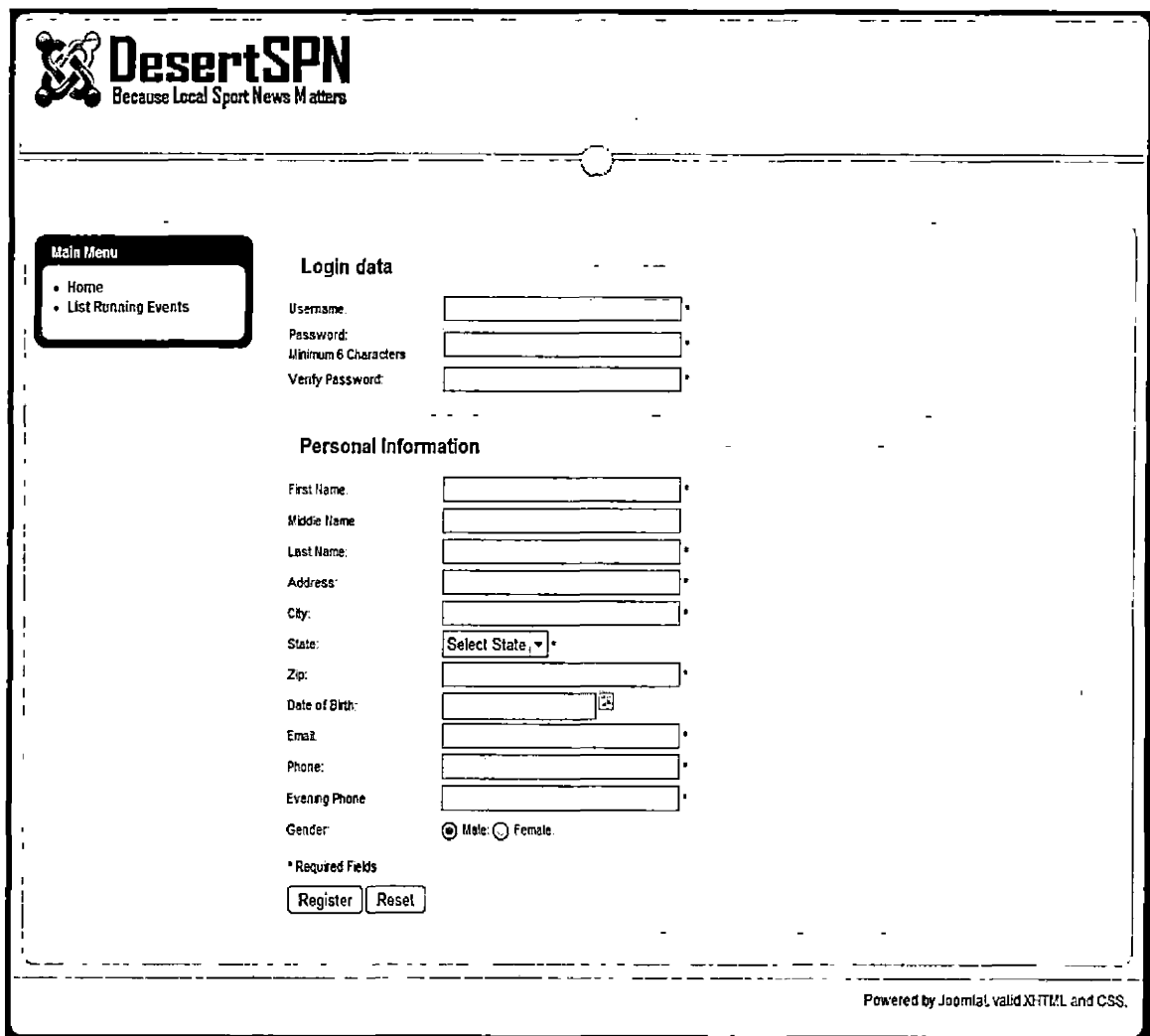


Figure 38. Login Window Event Registration

There is also a "forgot your password" link which does what it says. By clicking there, a username has to be provided. If the username is in the database, a code is sent in an email to the registrant to be used to reset the password. Figure 39 shows the register page.



The image shows a web registration form for DesertSPN. At the top left is the logo with the text "DesertSPN Because Local Sport News Matters". Below the logo is a "Main Menu" box containing "Home" and "List Running Events". The form is divided into two main sections: "Login data" and "Personal Information".

Login data

- Username:
- Password: (Minimum 6 Characters)
- Verify Password:

Personal Information

- First Name:
- Middle Name:
- Last Name:
- Address:
- City:
- State:
- Zip:
- Date of Birth:
- Email:
- Phone:
- Evening Phone:
- Gender: Male Female

* Required Fields

Powered by Joomla!, valid XHTML and CSS.

Figure 39. Register Page

This page uses JavaScript to control that the number of characters or fields that is marked with a * are being filled out correctly. The form will not be submitted unless those fields are filled out. Fields that are not filled out correctly will appear as red indicating that the field has an error. When the field is filled out correctly, the registrant is taken to the race registration page. This page shows information such as race name, event date, the registrants' personal information, etc. It also shows the entry fee and it is possible to enter a deductible donation on this page. Figure 40 shows how this page is laid out.

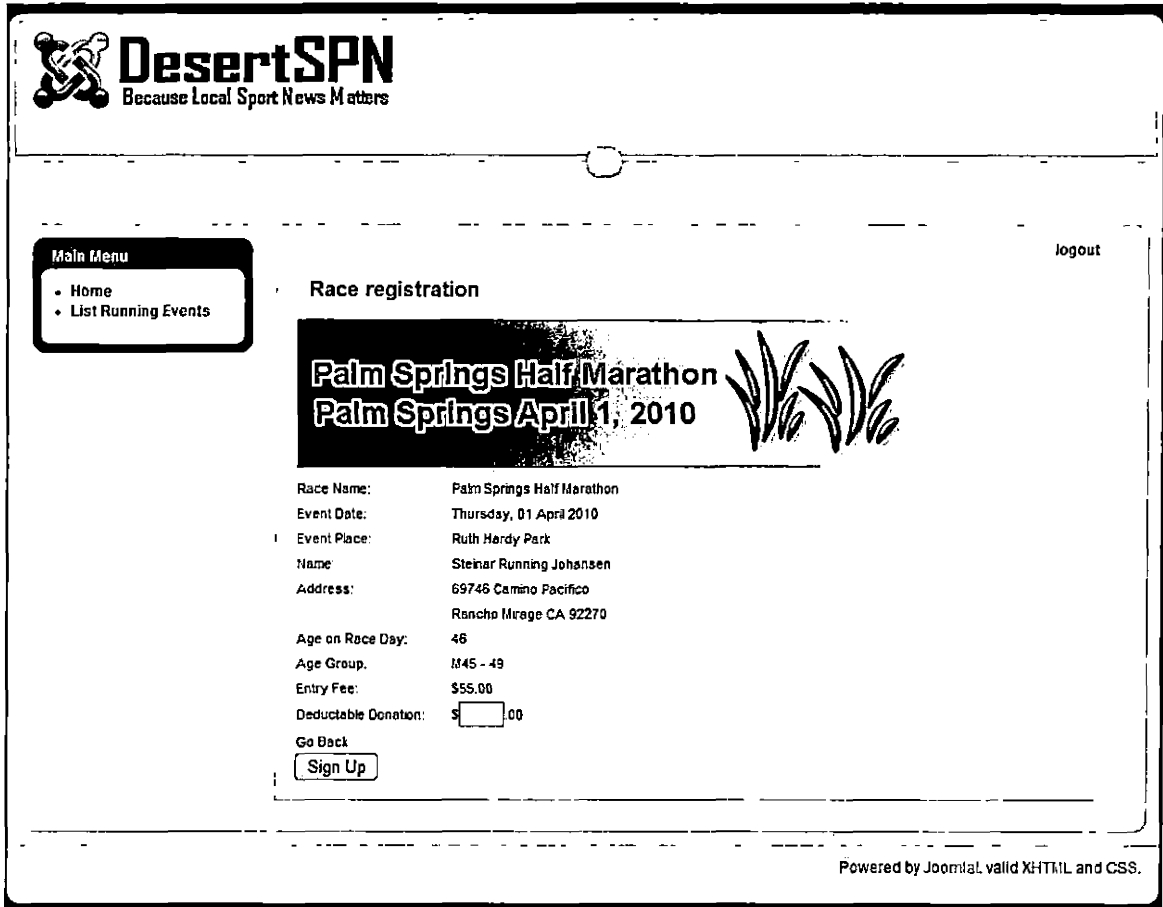


Figure 40. Race Registration Page

At this stage in the registration process a registrant can either logout by clicking on the "Logout" link, or they can proceed by clicking on the "Sign Up" button, that will take the registrant to the page where total cost for the event is listed. Personal information is listed as well as the race organization's information. Figure 41 shows the layout of the page.

Main Menu

- Home
- List Running Events

logout

Race Organization

PALMSPRINGS



Half Marathon
Half Marathon, 5K, 10K, 1/2 Mar & More

Greg Klein Sport
123 Vista China Rd.
Palm Springs CA 92262
(760) 555-6666

Personal Information

Name: Stenor Running Johansen
Address: 69746 Caminito Pacifico
Rancho Mirage CA 92270
Race Day Age: 46
Gender: male
Age Group: M45 - 49

Registration Fees

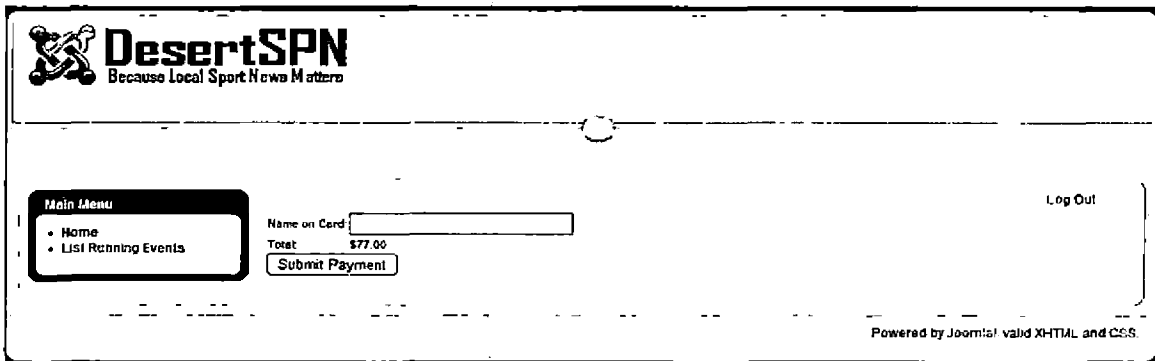
Registration Fee:	\$55.00
Deductible Donation:	\$20.00
Process Fee:	\$2.00
Total:	\$77.00

Check Out

Figure 41. Check Out Page

The options on this pages are either to logout to stop the registration or click the "Check Out" button. If the "Check Out" button is clicked, the entry point where a credit card company takes over is reached. For this documentation just a simple illustration of the process is

shown. The page shows the total cost for the event and it shows a box where the name of the credit card is entered. When this form is submitted, a return value is stored in the database indicating that the registrants payment was successful. A new window appears which confirms the payment and that an email is being sent to the registrant confirming the charge for the event. Figure 42 shows the payment page where the name on the credit card is entered.



DesertSPN
Because Local Sport News Matters

Main Menu

- Home
- List Running Events

Name on Card:

Total: \$77.00

Log Out

Powered by Joomla! valid XHTML and CSS

Figure 42. Enter Payment Information Page

When the "Submit Payment" button is clicked, the payment approved page appear. The payment approved page shows information whether the payment was approved or not. Figure 43 shows the layout of the payment approved page.

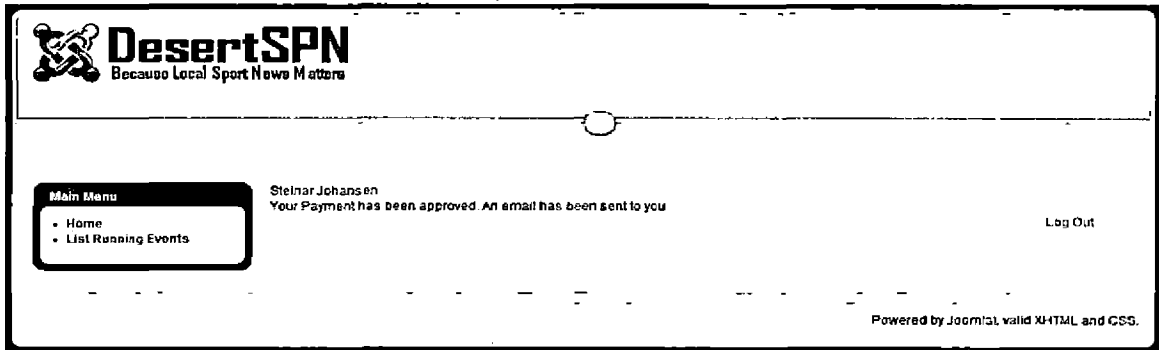


Figure 43. Payment Approved Page

If a registrant with a login information tries to sign up for an event that he has already signed up for, the registrant is routed to a page showing that he has already signed-up for the event. Additional information about the event such as where the event is held and the date and time for the event is presented. That page will also show previous events that the registrant has participated in where he registered using the system. Figure 44 shows the layout of that page.

DesertSPN
Because Local Sport News Matters

logout

Main Menu

- Home
- List Running Events
- Search Event Results

Login

Username

Password

Remember Me

- Forgot your password?

Personal Information

Name: Steinar Running Johansen
 69748 Camino Pacifico
 Rancho Mirage CA 92270

Upcoming Events You Are Signed Up For

Event Date	Event Name	Event Place
Thursday, 01 April 2010	Palm Springs Half Marathon	Ruth Hardy Park

Old Events You Joined

Event Date	Event Name	Event Place	See Results
Friday, 01 May 2009	Run For Life	Ruth Hardy Park, Palm Springs	Click here

Click there to see results from previous events

Powered by Joomla!, valid XHTML and CSS.

Figure 44. Already Registered Page

To perform the navigation between pages and to handle the data entered, four different components are working together. Figure 45 shows the layout of those components.

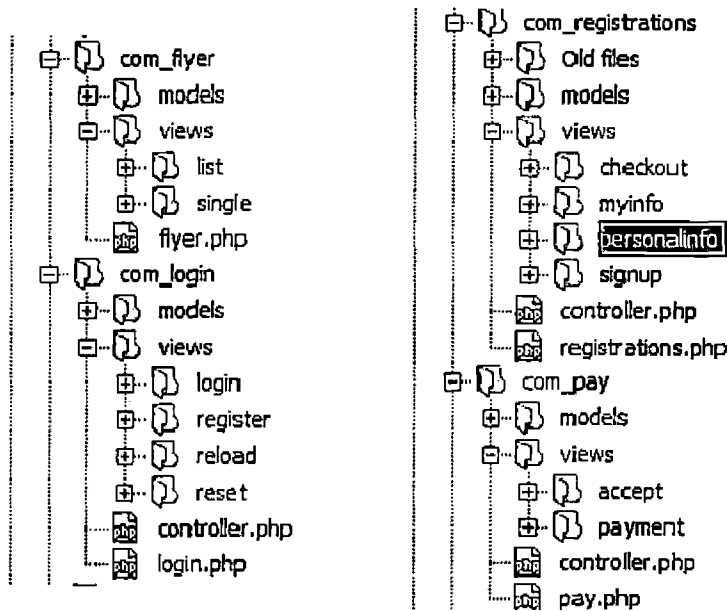


Figure 45. Front End Components

These components follow the MVC style as did the components in the "Back End". However the "Front End" components do not feature a table folder. For security reasons the table object is only directly accessible from the "Back End". However, all tables are accessible using the "Joomla!" 1.5 getInstance() function using the addIncludePath() function. "Joomla!" 1.5 provides the security.

The entry point is in the com_flyer controller class in the flyer.php file. Since the view variable is not set, the list view is set. This view provides a list of flyers

where the event date is later than today's date. Each flyers logo is made clickable using the JRoute object and the <a href> tag. When a flyer logo is clicked the view is set to single and the flyer that was created in the "Back End" is presented. The flyer presents another JRoute object for sign up. By clicking the signup link, the JRoute object routes the request to the login component. The request provides a view variable that sets the view to login. In the login window there are three options. They are login, register or forgot password. The login component has four views and two models to handle the login, register or forgot password request. If either a login or a registration is used, a session is started that stores the eventID, username and personID. These variables are used to retrieve data from the database further into the registration. The transition from the login component to the registration component is done by setting the view to signup. The display() function in the controller routes the request to the checkout view. The save() function routes the request to the pay component that handles the payment and is the entry point where a connection to a credit card company is featured. There are a few checks in the registration that will route the registrant to an

information page in case data like eventID is getting lost. The view myinfo and personal info handles the redirection. These two views are also involved in case a registrant tries to register for an event that the registrant has already signed up for. In that case the personalinfo view will show the information about the event.

The register page is using JavaScript to make sure that form fields are correctly filled out. Each field is checked with regards to a regular expression for the field. If a field has to be checked, the class name "inputbox required" is added to the form field and "Joomla!" will check the field. The form cannot be submitted unless all fields are correct filled out.

4.6 Result Service

The result service is accessed from the web's start page. On the Main Menu section a click on the link named "Search Event Results" will bring up a list of events. Figure 46 shows where to access the event results.

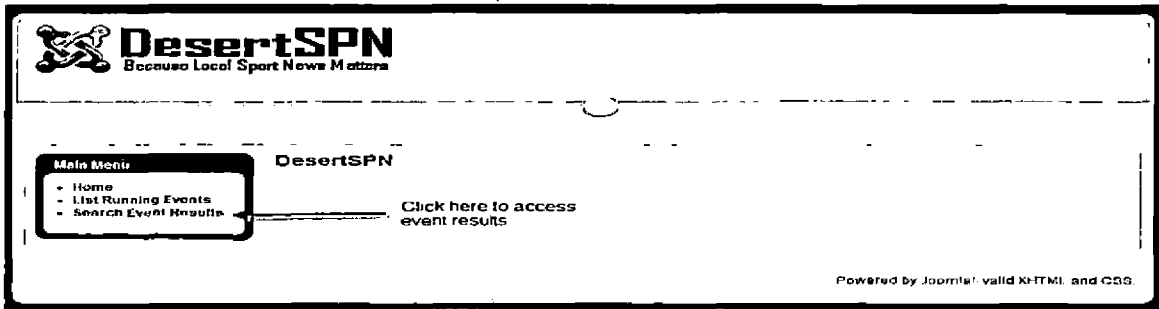


Figure 46. Access to Event Results

When the link is clicked, a list of events that have results connected will show up. The list will only show events that are older than today's date. By clicking on one of the event icons a page with search options will appear. Figure 47 shows how the list is laid out.

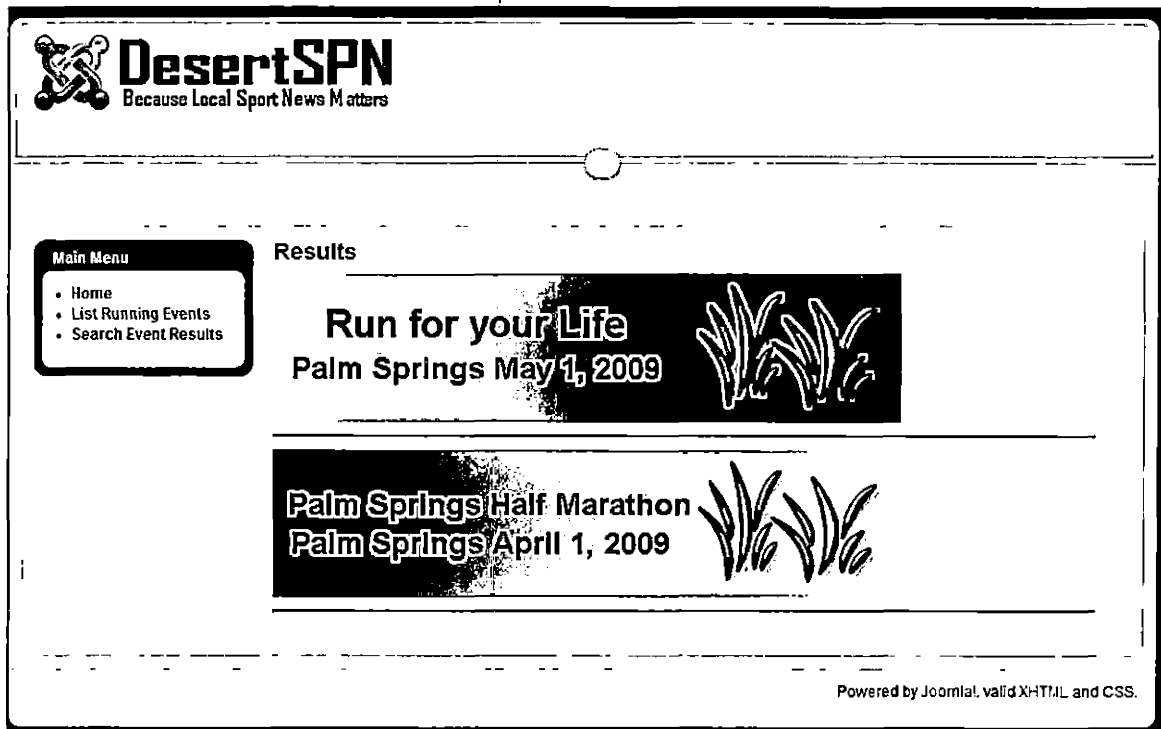


Figure 47. List Events with Results Service

The layout for this is the same as the layout used to show event flyers. For consistency issues, the same design is being used so registrants, participants and others should feel confident that they are on the same site. Figure 48 shows how the results are laid out and what the search options are when searching for results.

Main Menu

- Home
- List Running Events
- Search Event Results

Login

 Username

 Password

 Remember Me

- Forgot your password?

Results for Run For Life at Friday, 01 May 2009

Search for runner

 Bib

Search for runner

 First Name: Last Name:

Search for runner

 City

Search for runner

 Age Group

Bib	First Name	Middle name	Last Name	City	State	Age	Age Group	Sex	Time	Pace	ABP	SexPI	AgePI
5	NoSeeth		Forkie	Los Angeles	CA	19	F15 - 19	F	00:17:56	05:48:00	1	1	1
4	Running	John	JyLife	Los Angeles	CA	25	M25 - 29	M	00:18:45	06:02:00	2	1	1
2	Arejou		Running	Santa Ana	CA	53	F50 - 54	F	00:19:34	06:17:00	3	2	1
1	Run	Running	Runningsen	Rancho Mirage	CA	46	M45 - 49	M	00:20:10	06:29:00	4	2	1
6	Thalshly		RunEver	Palm Springs	CA	21	F20 - 24	F	00:20:45	06:40:00	5	3	1
3	Marathon		Runner	Palm Desert	CA	34	M30 - 34	M	00:21:03	06:48:00	6	3	1

Powered by Joomla!, valid XHTML and CSS.

Figure 48. Race Result Page

The layout for the results that are used is a standard that many event and race providers are using. The list shows data about a person like bib number, name, city and state, age, age group, gender, time, pace per mile, and then a count for overall result, results among gender and results among age group. The list is sorted in ascending order based on race time. The page also provides several ways to search for a participant, or a group of

participants. Searching by bib number only the person with the bib will show up with exactly the same data as represented in the overall list. A search can be executed based on first name, last name or both. A search for all participants from one city can be done, or a search based on age group is possible. The number of age groups refers to the actual number of age groups present in the event. Figure 49 shows the results when a search is done for bib number 1.

DesertSPN
Because Local Sport News Matters

Main Menu

- Home
- List Running Events
- Search Event Results

Results for Run For Life at Friday, 01 May 2009

Bib	First Name	Middle name	Last Name	City	State	Age	Age Group	Sex	Time	Pace	AKPI	SexPI	AgePI
1	Steiner	Running	Johansen	Rancho Mirage	CA	45	M45-49	M	00:20:10	08:29:00	4	2	1

Powered by Joomla!, valid XHTML and CSS.

Figure 49. Result Search for Bib Number 1

The amount of data used on this documentation are low, but it illustrates the concept for how this is working.

CHAPTER FIVE

PROGRAM SOLUTIONS AND SECURITY

5.1 Introduction

This chapter will discuss solutions to some problems that occurred during the programming as well as discussing security issues related to especially POST data sent with forms. SQL injection attack is always something that has to be taken seriously when form data is sent to be processed. Other issues that this web system has to deal with are "Joomla!" 1.5 programming style and how to take advantage of built in features in the framework. (See APPENDIX C and APPENDIX D for full source code for the system).

5.2 Solutions to Problems

The "Joomla!" 1.5 framework is using a built in feature to update a record. "Joomla!" 1.5 uses a form variable called cid. This variable is an array of records and it helps the system to load the first record which is the record that is being updated. The call used to initiate the variable is `$cid = JRequest::getVar('cid', array(0), '', 'array');`

```
$id = $cid[0];
```

```
$row = load($id);
```

The `$id` is set to the first record in the `cid` array which is the record that is ready to be edited. This `id` is assigned to an instance of the table object being used. When the controller gets POST data from the form, the `$id` value is set and the `bind()` function is invoked instead of the `save()` function. The `bind()` function will update the record that the `id` value carries.

"Joomla!" 1.5 feature a WYSIWYG editor that this project uses in several places. This gives an editor many opportunities when for instance designing an event flyer. The following code will utilize the use of the editor.

```
$editor =& JFactory::getEditor(); //Create an instance  
$this->assignRef('editor', $editor);
```

In the `default.php` the editor is initialized with this code.

```
<?php echo $this->editor->display('eventDescription',  
$this->row->eventDescription, '100%', '250', '40', '10');?>
```

The parameters for the HTML editor are name, content, width, height, columns, rows and buttons in that order.

"Joomla!" will strip off all HTML coding when the form data is submitted. To handle this problem in the

`JRequest::getVar()` function, the last parameter past in to

the function has to be JREQUEST_ALLOWRAW which will reinstall the code after the Bind() has stripped it off.

An important issue of discussion is the calculation of age based on the date of birth info submitted to the system by a registrant. There are several things that have to be considered such as leap years and when the race takes place. Figure 50 shows the function created to calculate race day age.

```
//calculate Race Day Age based on DOB
function birthday ($bday, $rday)
{
    list($year1,$month1,$day1) = explode("-", $bday);
    list($year2,$month2,$day2) = explode("-", $rday);
    $year_diff = $year2 - $year1;
    $month_diff = $month2 - $month1;
    $day_diff = $day2 - $day1;
    if ($month_diff < 0) $year_diff--;
    elseif (($month_diff==0) && ($day_diff < 0)) $year_diff--;
    return $year_diff;
}
```

Figure 50. Calculate Race Day Age

The explode() function will break up the date by year, month and day. The year difference is the result of race date year minus the birth date year. The month difference

is the result of race date month minus birth date month. The day difference is the result of race date day minus birth date day. As a result of the month difference or day difference the year difference is decremented.

5.3 Security

Security is always an issue when a web system is dealing with form data. SQL injection attack is a common problem and has to be taken seriously. The first issue to discuss is how the code is invoked. All files starts with a statement like this:

```
defined( '_JEXEC' ) or die ('Restricted Access');
```

This code makes sure that the code can only be invoked within "Joomla!" 1.5 itself. Absolutely all the files start with this statement except for the index.php file. Another security feature that is normally applied is that there is an index.html file within all the folders. This solution is preferred to avoid showing the file structure in case something goes wrong during browsing the web.

To deal with form data, whenever a form is submitted, the form contains the statement

```
<?php echo JHTML::_ ( 'form.token' ); ?> This is a randomly generated form token that will be checked by the function
```

that will handle the data in the controller. In the controller the following code is present to handle the form token:

```
JRequest::checkToken('request') or jexit( 'Invalid Token' );
```

If the token is wrong, an error message is sent to the user telling the user that it is an invalid token.

There are several ways to handle submitted form data. The first thing to do is to force variables to their expected values. Here is an example.

```
$userName = JRequest::getString('username', '', 'post', 'string');
```

This statement will strip off all white spaces, and characters that are not letters. In case you want to allow special characters, add the `JREQUEST_ALLOWRAW` as the fifth parameters. "Joomla!" will help with the security. It is possible to even increase the security by using special queries if the application requires top safety. In many cases, what the `JRequest` object does with regards to security, it is considered enough to prevent SQL injection attack.

CHAPTER SIX

SYSTEM TEST

6.1 Introduction

The system has been tested thoroughly and errors discovered have been repaired as they have been discovered. Testing has been done by the programmer of the system, a potential user of the system as well as the test that was done during two demos at the university. The testing has been done on two levels, component tests and system test.

6.2 Component Tests

All the different modules have been tested individually with test data to check that they behave as anticipated. Unexpected data have been entered to see if the component behaves correctly with regards to rejecting values that are not expected for a field. Each component has been thoroughly checked with regards to response to unexpected access to a component from outside the "Joomla!" system to see that access to a file is denied.

6.3 System Tests

The system has been tested as a whole system to see that the transition between components is working as it is supposed to. These tests have been done by the programmer, a potential user and through two demonstrations for the board of advisors. Errors detected have been corrected and new tests has been performed. The tests has been done with regard to several issues such as correct outcome of a specific task, clearness of what data to enter, correct calculations and clear presentation of the results of an operation.

CHAPTER SEVEN

CONCLUSION AND FURTHER DIRECTIONS

7.1 Conclusion

This project will be a good asset for small to medium-sized events with regards to advertisement, administer registrations, management of participants and running result service for an event. It contains all the factors that can make an event successful and it will reduce the costs leaving more revenue to the cause the event is supposed to support. The system is easy to use and there is no requirement for programming skills to successfully administer an event.

Future improvements will be to add electronic timing to the system. That will make it a complete system with the possibility to manage even large events. Another improvement to the system would be to add some JavaScript components such as automatic fill in for form elements, automatic search through the database for previous participants. Another improvement would be the ability to sell apparel such as running gear and event memorabilia as part of the check out procedure when registered for an event.

APPENDIX A
PROJECT SOURCE CODE, BACK END

Source code for com_organization

com_organizations\controller.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-05 21:14:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This controller will responde accordinly to user actions. The controller will
 * detect which request has been made and respond appropriately by triggering the
 * model to manipulate the data appropriately and passing the model into the view
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Include the path to the TableEvents class in the events.php
JTable::addIncludePath(JPATH_COMPONENT_DS.'tables');

// Import Joomla! 1.5 controller from the framework
jimport('joomla.application.component.controller');

// This class implements the edit(), add() and save() funtions
class OrganizationsController extends JController
{
    // Activates the edit button on the default view
    function edit()
    {
        // we set the view environment variable to single
        JRequest::setVar('view', 'single');
        // load the view.html.php for single records
        $this->display();
    }

    // Activates the new button on the default view
    function add()
    {
        // we set the view environment variable to single
        JRequest::setVar('view', 'single');
        // load the view.html.php for single records
        $this->display();
    }

    function save()
    {
        // Include global member functions like redirect for instance
        global $option;

        JRequest::checkToken('request') or jexit( 'Invalid Token' );

        // Set Row as an instance of the TableFlyer Class
        $row =& JTable::getInstance('organization', 'Table');

        // Bind() is a function of $row and loads all the variables from the form
        // If unsuccessful, an error is casted. Bind() matches the variables to
        // to the object. JRequest::get() reduses the ability of SQL Attack
        if ( !$row->bind(JRequest::get('post')) )
        {
            JError::raiseError(500, $row->getError());
        }

        $row->orgLogo = JRequest::getVar('orgLogo', '', 'post', 'string',
            JREQUEST_ALLOWRAW);

        // All member variables are turned into update or store statements depending
        // on the id. In this case it will be an insert statement
    }
}
```



```

        if(!$row->store())
        {
            JError::raiseError(500, $row->getError());
        }

        // redirect to the main view for the extension
        $this->setRedirect('index.php?option=' . $option, 'Organization Saved');
    }

    function display()
    {
        $view = JRequest::getVar('view');
        if (!$view)
        {
            JRequest::setVar('view', 'list');
        }
        parent::display();
    }
}

```

com_organizations\organization.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-05 10:40:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This is the entry point to the extension to create information about an
 * organization for the system.
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// This is a call to include the controller, and make sure it is only called once
require_once( JPATH_COMPONENT_DS.'controller.php' );

// This code creates the toolbar for the different windows we are using. The
// JToolBarHelper contains functions that produces all the HTML code necessary
switch ( $task )
{
    case 'edit':
    case 'add':
        JToolBarHelper::save();
        JToolBarHelper::apply();
        JToolBarHelper::cancel();
        break;
    default:
        JToolBarHelper::title( JText::_ ( 'List Organizations' ), 'generic.png' );
        JToolBarHelper::publishList();
        JToolBarHelper::unpublishList();
        JToolBarHelper::editList();
        JToolBarHelper::deleteList();
        JToolBarHelper::addNew();
        break;
}

// Create an instance of the controller
$controller = new OrganizationsController();

// Execute the task variable chosen
$controller->execute( $task );

// Redirect after the task is finish its execution.
$controller->redirect();

```

com_organizations\models\list.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-05 18:59:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport( 'joomla.application.component.model' );

class OrganizationsModelList extends JModel
{
    // initiate the data variable
    var $data = null;

    function getData()
    {
        // Check if the data variable is empty
        if( empty( $this->data ) )
        {
            // Create a query to load all the records from jos_events
            $query = "SELECT * FROM #__organization";

            // get a list of data based on the query
            $this->data = $this->_getList( $query );
        }

        // Return the data
        return $this->data;
    }
}
}
```

com_organizations\views\list view.html.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-05 19:02:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the view for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

class OrganizationsViewList extends JView
{
    // Looking for a PHP file in the models directory
    function display( $tpl = null )
    {
        // Calls the getData
        $rows =& $this->get( 'data' );

        // Assign the data to the rows
        $this->assignRef( 'rows', $rows );

        // Looks for output in the tmpl directory
        parent::display( $tpl );
    }
}
```

```
}
```

com_organizations\views\list\tmpl\default.php .

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-05 19:04:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is an interface that produces the view for all records.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );
?>

<!-- This creates the header for the output screen -->
<form action="index.php" method="post" name="adminForm">
  <table class="adminlist">
    <thead>
      <tr>
        <th width="4%">
          <input type="checkbox" name="toggle" value=""
            onclick="checkAll(<?php echo count ( $this->rows ); ?>);"
          />
        </th>
        <th width="15%">Organization</th>
        <th width="15%">Address</th>
        <th width="15%">City</th>
        <th width="4%">State</th>
        <th width="4%">Zip</th>
        <th width="10%">Phone</th>
        <th width="10%">Contact</th>
        <th width="5"nowrap="nowrap">Published</th>
      </tr>
    </thead>
    <?php
    jimport('joomla.filter.output');
    // Variabe that creates different CSS. 1 or 0 to distinguish rows
    $k = 0;

    // Loop through the database as long as there are records
    for($i=0, $n=count($this->rows); $i < $n; $i++)
    {
      // reference to the current row
      $row =&$this->rows[$i];

      // HTML code for a check box that is recognized by the back end
      //
      // JavaScript
      $checked = JHTML::_('grid.id', $i, $row->orgID);

      // Publish an icon telling whether the record is published or not
      $published = JHTML::_('grid.published', $row, $i);

      //Pick the row to edit
      $link = JFilterOutput::ampReplace( 'index.php?option=' .
        $option . '&task=edit&cid[]=' . $row->orgID );
      ?>
      <tr class="<?php echo "row$k"; ?>">
        <td>
          <?php echo $checked; ?>
        </td>
        <td>
          <a href="<?php echo $link; ?>"> <?php echo $row->orgName; ?></a>
        </td>
        <td>
          <?php echo $row->orgAddress; ?>
        </td>
        <td>

```

```

        <?php echo $row->orgCity; ?>
    </td>
    <td>
        <?php echo $row->orgState; ?>
    </td>
    <td>
        <?php echo $row->orgZip; ?>
    </td>
    <td>
        <?php echo $row->orgPhone; ?>
    </td>
    <td>
        <?php echo $row->orgContact; ?>
    </td>
    <td align="center">
        <?php echo $published; ?>
    </td>
</tr>
<?php

    // Switch k between 1 and 0 for different CSS
    $k = 1 - $k;
    }
    ?>
</table>

<!-- Hidden variables for option -->
<input type="hidden" name="option" value="<?php echo $option; ?>" />
<input type="hidden" name="task" value="" />
<input type="hidden" name="checked" value="0" />
</form>

```

com_organizations\views\single\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-05 14:35:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This file will define the toolbar for the administrator needed to easily manage
 * organization records for the system.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

// The Joomla! 1.5 convention is component name, view and the name of the view
class OrganizationsViewSingle extends JView
{
    function display($tpl = null)
    {
        JToolBarHelper::title( JText::_ ( 'Register Organization' ), 'generic.png' );
        // Create an instance of the TableFlyer object
        $row =& JTable::getInstance( 'organization', 'Table' );

        // Pull in an array of record id.
        $cid = JRequest::getVar( 'cid', array(0), '', 'array' );

        // Since we want to edit one record at the time, we select the first ID in
// array
        $id = $cid[0];
        $row->load($id);

        // Instance of the object is returned as a row
        $this->assignRef( 'row', $row );
        $editor =& JFactory::getEditor();
        $this->assignRef( 'editor', $editor );
    }
}

```

```

// Create radio buttons for published
$this->assignRef('published', JHTML::_('select.booleanlist',
    'published', 'class="inputbox"', $row->published));

// Loads an output file based on the value of $tpl. Since $tpl = null
// default.php will be loaded
parent::display($tpl);
}
}

```

com_organizations\views\single\tmpl\default.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-05 14:38:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is an interface to contains the form data for the addOrganization to the system.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// This prepare the JavaScript Calendar available to be used in the form
JHTML::_('behavior.calendar');
?>

<form action="index.php" method="post" name="adminForm" id="adminForm">
  <fieldset class="adminform">

    <!-- Creates the line around the form that is being shown -->
    <legend>Add Organization</legend>

    <!-- admintable is a preset class for Joomla! 1.5 to format the table -->
    <table class="admintable">
      <tr>
        <td width="100" align="right" class="key">
          Organization Name:
        </td>
        <td>
          <input class="text_area" type="text" name="orgName"
            id="eventName" size="50" maxlength="250"
            value="<?php echo $this->row->orgName; ?>" />
        </td>
      </tr>
      <tr>
        <td width="100" align="right" class="key">
          Organization Logo:
        </td>
        <td>
          <!-- This code display an instance the admin HTML rich text editor.
          -->
          <td>
            <?php
              echo $this->editor->display('orgLogo',
                $this->row->orgLogo, '100%', '250', '40', '10');
            ?>
          </td>
        </tr>
      <tr>
        <td width="100" align="right" class="key">
          Address:
        </td>
        <td>
          <!-- Creates the HTML code to present a JavaScript Calendar -->
          <input class="text_area" type="text" name="orgAddress"
            id="eventName" size="50" maxlength="250"
            value="<?php echo $this->row->orgAddress; ?>" />
        </td>
      </tr>
    </table>
  </fieldset>
</form>

```

```

        <td width="100" align="right" class="key">
            City:
        </td>
        <td>
            <input class="text_area" type="text" name="orgCity"
                id="eventPlace" size="50" maxlength="250"
                value="<?php echo $this->row->orgCity; ?>" />
        </td>
    </tr>
    <tr>
        <td width="100" align="right" class="key">
            State:
        </td>
        <td>
            <input class="text_area" type="text" name="orgState"
                id="eventType" size="50" maxlength="250"
                value="<?php echo $this->row->orgState; ?>" />
        </td>
    </tr>
    <tr>
        <td width="100" align="right" class="key">
            Zip:
        </td>
        <!-- This code display an instance the admin HTML rich text editor.
    -->
        <td>
            <input class="text_area" type="text" name="orgZip"
                id="eventType" size="50" maxlength="250"
                value="<?php echo $this->row->orgZip; ?>" />
        </td>
    </tr>
    <tr>
        <td width="100" align="right" class="key">
            Phone:
        </td>
        <!-- This code display an instance the admin HTML rich text editor.
    -->
        <td>
            <input class="text_area" type="text" name="orgPhone"
                id="eventType" size="50" maxlength="250"
                value="<?php echo $this->row->orgPhone; ?>" />
        </td>
    </tr>
    <tr>
        <td width="100" align="right" class="key">
            Contact Person:
        </td>
        <td>
            <input class="text_area" type="text" name="orgContact"
                id="eventAdministrator" size="50" maxlength="250"
                value="<?php echo $this->row->orgContact; ?>" />
        </td>
    </tr>
    <tr>
        <td width="100" align="right" class="key">
            Published:
        </td>
        <td>
            <?php echo $this->published; ?>
        </td>
    </tr>
</table>
</fieldset>
<input type="hidden" name="orgID" value="<?php echo $this->row->orgID; ?>" />
<input type="hidden" name="option" value="<?php echo $option; ?>" />
<input type="hidden" name="task" value="" />
<?php echo JHTML::_ ( 'form.token' ); ?>
</form>

```

com_organizations\tables\organization.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-05 10:49:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is an extension of the JTable specific to the jos_organization table
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

class TableOrganization extends JTable
{
    var $orgID = null;
    var $orgLogo = null;
    var $orgName = null;
    var $orgAddress = null;
    var $orgCity = null;
    var $orgState = null;
    var $orgZip = null;
    var $orgPhone = null;
    var $orgContact = null;
    var $published = null;

    function __construct( &$db )
    {
        parent::__construct( '#_organization', 'orgID', $db );
    }
}
```

Source Code for com_flyer

com_flyer\controller.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-05 21:14:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This controller will responde accordinly to user actions. The controller will
 * detect which request has been made and respond appropriately by triggering the
 * model to manipulate the data appropriately and passing the model into the view
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Include the path to the TableEvents class in the events.php
JTable::addIncludePath( JPATH_COMPONENT_DS . 'tables' );

// Import Joomla! 1.5 controller from the framework
jimport( 'joomla.application.component.controller' );

// This class implements the edit(), add() and save() funtions
class FlyerController extends JController
{
    // Activates the edit button on the default view
    function edit()
    {
        // we set the view environment variable to single
        JRequest::setVar( 'view', 'single' );
        // load the view.html.php for single records
        $this->display();
    }
}
```

```

}

// Activates the new button on the default view
function add()
{
    // we set the view environment variable to single
    JRequest::setVar('view', 'single');
    // load the view.html.php for single records
    $this->display();
}

// Activates the save button on the add flyer form view
function save()
{
    // Include global member functions like redirect for instance
    global $option;

    // Set Row as an instance of the table agegroups Class
    $row2=& JTable::getInstance('agegroups', 'Table');

    /*Bind() is a function of $row and loads all the variables from the form
    If unsuccessful, an error is casted. Bind() matches the variables to
    to the object. JRequest::get reduces the ability of SQL Attack */
    if (!$row2->bind(JRequest::get('post')))
    {
        JError::raiseError(500, $row->getError());
    }

    // INSERT or UPDATE depending whether an ID is present
    if (!$row2->store())
    {
        JError::raiseError(500, $row->getError());
    }

    // Get the ID from agegroups and use it as FK in events
    $myID = $row2->ageID;

    // Set Row as an instance of the table events Class
    $row =& JTable::getInstance('events', 'Table');

    /*Bind() is a function of $row and loads all the variables from the form
    If unsuccessful, an error is casted. Bind() matches the variables to
    to the object. JRequest::get reduces the ability of SQL Attack */
    if (!$row->bind(JRequest::get('post')))
    {
        JError::raiseError(500, $row->getError());
    }

    // Since these fields accepts HTML content, it needs special handling.
    // Bind() strips out HTML and getVar will re-install it again
    $row->eventDescription = JRequest::getVar('eventDescription', '', 'post',
        'string', JREQUEST_ALLOWRAW);
    $row->eventInfo = JRequest::getVar('eventInfo', '', 'post', 'string',
        JREQUEST_ALLOWRAW);
    $row->eventLogo = JRequest::getVar('eventLogo', '', 'post', 'string',
        JREQUEST_ALLOWRAW);

    // Insert FK from agegroups table
    $row->ageEvent = $myID;
    $organization = JRequest::getInt('organization', '', 'post', 'string');
    if($organization != '' || $organization != 0)
    {
        $row->orgID = $organization;
    }
    // $eventids = $row->eventID;

    $eventname =& JRequest::getVar('eventName', '', 'post', 'string');

    // If no date is given, a default value is inserted
    if (!$row->event_Date)
    {

```



```

    $row->event_Date = date( 'Y-m-d H:i:s' );
}

// INSERT or UPDATE depending whether an ID is present
if(!$row->store())
{
    JError::raiseError(500, $row->getError());
}

//get eventID from the last saved row
$eventids = $row->eventID;
//get the distance from the radio list. whether it is a 5k, 10k etc
$event_type = JRequest::getInt('distances', '', 'post', 'string');
//get the type of event. Is it run, walk, triathlon etc
$activity = JRequest::getInt('activ', '', 'post', 'string',
    JREQUEST_ALLOWRAW);
//queries for different id's to be used for to update or store event data
$sql1 = "SELECT id FROM #__event_has_many WHERE eventID='$eventids'";
$sql2 = "SELECT distanceID FROM #__event_has_many WHERE distanceID =
    '$event_type'";
$sql3 = "SELECT id FROM #__goes_with WHERE eventID='$eventids'";
//Get id from jos_event_has_many table if it exists
$db1 = JFactory::getDBO();
$db1->setQuery($sql1);
$result1 = $db1->loadResult();
//get the id from jos_goes_with if it exists
$db3 = JFactory::getDBO();
$db3->setQuery($sql3);
$result2 = $db3->loadResult();
//If data exists for id in jos_has_many, then go ahead and update the record
if($result1)
{
    $db11 = JFactory::getDBO();
    $update = new stdClass;
    $update->id = $result1;
    $update->distanceID = $event_type;
    $db11->updateObject('#__event_has_many', $update, 'id' );
    if(!$db11)
    {
        JError::raiseError(500, 'No update happened');
    }
}
//if id does not exist for jos_event_has_many, then go ahead and insert it
else if(!$result1)
{
    $sql5 = "INSERT INTO #__event_has_many VALUES ('', '$eventids',
        '$event_type')";
    $db5 = JFactory::getDBO();
    $db5->setQuery($sql5);
    $result3 = $db5->loadResult();
    if(!$db5)
    {
        JError::raiseError(500, 'what is this');
    }
}
//If id exist for jos_goes_with, go ahead and update that record
if($result2)
{
    $db12 = JFactory::getDBO();
    $update2 = new stdClass;
    $update2->id = $result2;
    $update2->activityID = $activity;
    $db12->updateObject('#__goes_with', $update2, 'id');
    if(!$db12)
    {
        JError::raiseError(500, 'No update happened');
    }
}
//If id does not exist, go ahead and insert a new record
else if(!$result2)

```

```

    {
        $sql4 = "INSERT INTO #__goes_with VALUES ('', 'Seventids',
'$activity')";
        $db4 = JFactory::getDBO();
        $db4->setQuery($sql4);
        $result4 = $db4->loadResult();
        if(!$db4)
        {
            JError::raiseError(500, $row->getError());
        }
    }

    // redirect to the main view for the extension
    $this->setRedirect('index.php?option=' . $option, 'Flyer Saved');
}

// If the single view is not requested, the all view is displayed.
function display()
{
    $view = JRequest::getVar('view');
    if (!$view)
    {
        JRequest::setVar('view', 'all');
    }
    parent::display();
}
}
}

```

com_flyer\flyer.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-06-29 08:14:PST
 *             Last Update: 2009-07-05 21:30:00:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This is the entry point to the extension to create a flyer for the system. The
 * extension will allow administrators to create event flyers for their events.
 * The administrators will also decide when the flyer will be displayed, and for
 * for how long. They can also edit the flyers they have previouslu created.
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// This is a call to include the controller, and make sure it is only called once
require_once( JPATH_COMPONENT_DS.'controller.php' );

// This code creates the toolbar for the different windows we are using. The
// JToolBarHelper contains functions that produces all the HTML code necessary

switch ( $task )
{
    case 'edit':

    case 'add':
        JToolBarHelper::save();
        JToolBarHelper::apply();
        JToolBarHelper::cancel();
        break;
    default:
        JToolBarHelper::title( JText::_ ( 'List Event Flyers' ), 'generic.png' );
        JToolBarHelper::publishList();
        JToolBarHelper::unpublishList();
        JToolBarHelper::editList();
        JToolBarHelper::deleteList();
        JToolBarHelper::addNew();
}

```

```

        break;
    }

    // Create an instance of the controller
    $controller = new FlyerController();

    // Execute the task variable chosen
    $controller->execute( $task );

    // Redirect after the task is finish its execution.
    $controller->redirect();

```

com_flyer\models\all.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-05 18:10:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport( 'joomla.application.component.model' );

class FlyerModelAll extends JModel
{
    // initiate the data variable
    var $event = null;

    function getEvent()
    {
        // Check if the data variable is empty
        if( empty( $this->event ) )
        {
            // Create a query to load all the records from jos_events

            $query = 'SELECT * FROM #__events AS a, #__agegroups AS b,
                #__distances AS c, #__activity AS d, #__event_has_many AS e,
                #__goes_with AS f, #__organization as g
                WHERE a.ageEvent = b.ageID AND a.orgID = g.orgID
                AND a.eventID = e.eventID AND c.distanceID = e.distanceID
                AND a.eventID = f.eventID AND d.activityID = f.activityID
                ';

            // get a list of data based on the query
            $this->event = $this->_getList( $query );
        }

        // Return the data
        return $this->event;
    }
}

```

com_flyer\models\single.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-24 18:58:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

```

```

*/
// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC') or die ('Restricted Access');

// Import the Joomla! 1.5 framework model functions
jimport('joomla.application.component.model');

class FlyerModelSingle extends JModel
{
    var $activity = null;
    var $distance = null;
    var $organization = null;
    function getActivity()
    {
        if(empty($this->activity))
        {
            $query = "SELECT activityID, activity FROM #__activity";
            $this->activity = $this->_getList($query);
        }
        return $this->activity;
    }

    function getDistance()
    {
        if(empty($this->distance))
        {
            $query = "SELECT distanceID, distance FROM #__distances";
            $this->distance = $this->_getList($query);
        }
        return $this->distance;
    }

    function getorganization()
    {
        if(empty($this->organization))
        {
            $sql = "SELECT orgID AS value, orgName as text FROM #__organization";
            $this->organization = $this->_getList($sql);
        }
        return $this->organization;
    }
}

```

com_flyer\views\all\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-05 18:10:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the view for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC') or die ('Restricted Access');

// bring in the functions related to JView
jimport( 'joomla.application.component.view');

class FlyerViewAll extends JView
{
    // Looking for a PHP file in the models directory
    function display($tpl = null)
    {
        // Calls the getEvent
        $rows =& $this->get('event');

        // Assign the data to the rows
        $this->assignRef('rows', $rows);
    }
}

```

```

    // Looks for output in the tmpl directory
    parent::display($tpl);
}
}

```

com_flyer\views\all\tmpl\default.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-06-29 11:04:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is an interface that produces the view for all records.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );
?>

<!-- This creates the header for the output screen -->
<form action="index.php" method="post" name="adminForm">
  <table class="adminlist">
    <thead>
      <tr>
        <th width="3%">
          <input type="checkbox" name="toggle" value=""
            onclick="checkAll(<?php echo count ( $this->rows; ?>);" />
          </th>
        <th width="15%">Event Name</th>
        <th width="7%">Event Date</th>
        <th width="15%">Event Administrator</th>
        <th width="12%">Event Organization</th>
        <th width="4%">Span</th>
        <th width="6%">Lower Age</th>
        <th width="6%">Upper Age</th>
        <th width="10%">Distance</th>
        <th width="6%">Activity</th>
        <th width="5" nowrap="nowrap">Published</th>
      </tr>
    </thead>
    <?php
    jimport('joomla.filter.output');
    // Variabe that creates different CSS. 1 or 0 to distinqush rows
    $k = 0;

    // Loop through the database as long as there are records
    for($i=0, $n=count($this->rows); $i < $n; $i++)
    {
      // reference to the current row
      $row =&$this->rows[$i];
      $row2 =&$this->rows1[$i];

      // HTML code for a check box that is recognized by the back end
      javascript
      $checked = JHTML::_('grid.id', $i, $row->eventID);

      // Publish an icon telling whether the record is published or not
      $published = JHTML::_('grid.published', $row, $i);

      //Pick the row to edit
      $link = JFilterOutput::ampReplace( 'index.php?option=' .
        $option . '&task=edit&cid[]=' . $row->eventID);

    ?>
    <tr class="<?php echo "row$k"; ?>">
      <td align="center">
        <?php echo $checked; ?>

```

```

</td>
<td>
  <a href="<?php echo $link; ?>"> <?php echo $row->eventName; ?></a>
</td>
<td align="center">
  <?php echo $row->event_Date; ?>
</td>
<td>
  <?php echo $row->orgContact ?>
</td>
<td>
  <?php echo $row->orgName ?>
</td>
<td align="center">
  <?php echo $row->span ?>
</td>
<td align="center">
  <?php echo $row->lowAge ?>
</td>
<td align="center">
  <?php echo $row->upperAge ?>
</td>
<td align="center">
  <?php echo $row->distance ?>
</td>
<td align="center">
  <?php echo $row->activity ?>
</td>
<td align="center">
  <?php echo $published; ?>
</td>
</tr>
<?php
  // Switch k between 1 and 0 for different CSS
  $k = 1 - $k;
  }
  ?>
</table>

<!-- Hidden variables for option -->
<input type="hidden" name="option" value="<?php echo $option; ?>" />
<input type="hidden" name="task" value="" />
<input type="hidden" name="checked" value="0" />
</form>

```

com_flyer\views\single\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-02 14:25:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This file will define the toolbar for the administrator needed to easily manage
 * event records for the system.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

// The Joomla! 1.5 convention is component name, view and the name of the view
class FlyerViewSingle extends JView
{
  function display($tpl = null)
  {
    JToolBarHelper::title( JText::_('Create an Event Flyer'), 'generic.png');
    // Create an instance of the TableFlyer object

```

```

$row = & JTable::getInstance('events', 'Table');
// Create an instance of the agegroups table object
$row2 = & JTable::getInstance('agegroups', 'Table');
$row3 = & JTable::getInstance('event_has_many', 'Table');
// Pull in an array of record id.
$cid = JRequest::getVar('cid', array(0), '', 'array' );
// Since we want to edit one record at the time, we select the first ID in
array
$id = $cid[0];
$row->load($id);
//get data for the dropdown list for organizations
$org = & $this->get('organization');
//build the dropdown list for organizations
$organization[] = JHTML::_('select.option', '0',
    JText::_('Select Organization'), 'value', 'text' );
$organization = array_merge($organization, $org );
$lists['organization'] = JHTML::_('select.genericlist',
$organization, 'organization', 'class="inputbox" size="1" ', 'value',
'text');
//Assign the organisation list to the template
$this->assignRef('lists', $lists);
//Query one to connect events to agegroups
$sql = "SELECT ageEvent FROM #__events WHERE eventID = '$id'";
$db = & JFactory::getDBO();
$db->setQuery($sql);
$result = $db->loadResult();
//Query two to make sure that ageID and ageEvent is the same
$sql2 = "SELECT ageID FROM #__agegroups WHERE ageID = '$result'";
$db2 = & JFactory::getDBO();
$db2->setQuery($sql2);
$id2 = $db->loadResult();
$row2->load($id2);
//get the value for the radio list of activity used for updates
$sql3 = "Select activityID from #__goes_with where eventID = '$id'";
$db3 = & JFactory::getDBO();
$db3->setQuery($sql3);
$id3 = $db3->loadResult();
//get the value for the radio list of distances used for updates
$sql4 = "Select distanceID from #__event_has_many where eventID = '$id'";
$db4 = & JFactory::getDBO();
$db4->setQuery($sql4);
$id4 = $db4->loadResult();

// Instance of the object is returned as a row
$this->assignRef('row', $row);
$this->assignRef('row2', $row2);
// make a WYSIWYG editor available for the form
$editor = & JFactory::getEditor();
$this->assignRef('editor', $editor);

// Create radio buttons for published
$this->assignRef('published', JHTML::_('select.booleanlist',
    'published', 'class="inputbox"', $row->published));

// Activate a JavaScript calendar for the form
$this->assignRef('event_date', JHTML::_('calendar',
    $row->event_date, 'event_date', 'event_date'));

//Get data for the activity radio list
$activ = & $this->get('activity');
//populate the activity radio list
$this->assignRef('activ', JHTML::_('select.radiolist', $activ, 'activ',
    '', 'activityID', 'activity', $id3 ));
//get the data for the distance radio list
$distances = & $this->get('distance');
//populate the distances radio list.
$this->assignRef('distances', JHTML::_('select.radiolist', $distances,
'distances', '', 'distanceID', 'distance', $id4));
// Loads an output file based on the value of $tpl. Since $tpl = null
// default.php will be loaded
parent::display($tpl);
}

```

```
}
```

com_flyer\views\single\tmpl\default.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-06-29 11:04:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is an interface to contains the form data for the addEvents to the system.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// This prepare the JavaScript Calendar available to be used in the form
JHTML::_('behavior.calendar');
?>

<form action="index.php" method="post" name="adminForm" id="adminForm">
  <fieldset class="adminform">

    <!-- Creates the line around the form that is being shown -->
    <legend>Create Flyer</legend>

    <!-- admintable is a preset class for Joomla! 1.5 to format the table -->
    <table class="admintable">
      <tr>
        <td width="100" align="right" class="key">
          Event Logo:
        </td>
        <!-- This code display an instance the admin HTML rich text editor.
        -->
        <td>
          <?php
            echo $this->editor->display('eventLogo',
              $this->row->eventLogo, '100%', '250', '40', '10');
          ?>
        </td>
      </tr>

      <tr>
        <td width="100" align="right" class="key">
          Event Name:
        </td>
        <td>
          <input class="text_area" type="text" name="eventName"
            id="eventName" size="50" maxlength="250"
            value="<?php echo $this->row->eventName; ?>" />
        </td>
      </tr>

      <tr>
        <td width="100" align="right" class="key">
          Event Date:
        </td>
        <!-- Creates the HTML code to present a JavaScript Calendar -->
        <td>
          <?php echo $this->event_Date; ?>
        </td>
      </tr>

      <tr>
        <td width="100" align="right" class="key">
          Event Place:
        </td>
        <td>
          <input class="text_area" type="text" name="eventPlace"
            id="eventPlace" size="50" maxlength="250"
            value="<?php echo $this->row->eventPlace; ?>" />
        </td>
      </tr>
    </table>
  </fieldset>
</form>
```



```

</tr>
<tr>
  <td width="100" align="right" class="key">
    Event Type:
  </td>
  <td>
    <input class="text_area" type="text" name="eventType"
      id="eventType" size="50" maxlength="250"
      value="<?php echo $this->row->eventType; ?>" />
  </td>
</tr>
<tr>
  <td width="100" align="right" class="key">
    Event Description:
  </td>
  <td>
    <!-- This code display an instance the admin HTML rich text editor.
-->
    <td>
      <?php
        echo $this->editor->display('eventDescription',
          $this->row->eventDescription, '100%', '250', '40',
          '10');
      ?>
    </td>
  </tr>
<tr>
  <td width="100" align="right" class="key">
    Event Info:
  </td>
  <td>
    <!-- This code display an instance the admin HTML rich text editor.
-->
    <td>
      <?php
        echo $this->editor->display('eventInfo',
          $this->row->eventInfo, '100%', '250', '40', '10');
      ?>
    </td>
  </tr>
<tr>
  <td width="100" align="right" class="key">
    Organizer:
  </td>
  <td>
    <?php echo $this->lists['organization']; ?>
  </td>
</tr>
<tr>
  <td width="100" align="right" class="key">
    Published:
  </td>
  <td>
    <?php echo $this->published; ?>
  </td>
</tr>
</table>
</fieldset>
<fieldset class="adminform">
  <legend>Add Age Group </legend>
  <table>
    <tr>
      <td width="100" align="right" class="key">
        Define Lower Age:
      </td>
      <td>
        <input class="text_area" type="text" name="lowAge"
          id="lowAge" size="10" maxlength="250"
          value="<?php echo $this->row2->lowAge; ?>" />
      </td>
    </tr>
  </table>
</fieldset>

```

```

        <td width="100" align="right" class="key">
            Define Upper Age:
        </td>
        <td>
            <input class="text_area" type="text" name="upperAge"
                id="upperAge" size="10" maxlength="250"
                value="<?php echo $this->row2->upperAge; ?>" />
        </td>

        <td width="100" align="right" class="key">
            Define Age Span:
        </td>
        <td>
            <input class="text_area" type="text" name="span"
                id="span" size="10" maxlength="250"
                value="<?php echo $this->row2->span; ?>" /> Years
        </td>
    </tr>
</table>
</fieldset >
<fieldset class="adminform">
    <legend>Define event type</legend>
    <table>
        <tr>
            <td>
                <?php echo $this->distances; ?>
            </td>
        </tr>
    </table>
</fieldset>
<fieldset class="adminform">
    <legend>Define activity</legend>
    <table>
        <tr>
            <td>
                <?php echo $this->activ; ?>
            </td>
        </tr>
    </table>
</fieldset>

<input type="hidden" name="eventID" value="<?php echo $this->row->eventID; ?>" />
<input type="hidden" name="ageID" value="<?php echo $this->row2->ageID; ?>" />

<input type="hidden" name="option" value="<?php echo $option;?>" />
<input type="hidden" name="task" value="" />
<?php echo JHTML::_ ( 'form.token' ); ?>
</form>

```

com_flyer\tables\activity.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-13 09:01:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is an extension of the JTable specific to the jos_agegroups table
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC')or die ('Restricted Access');
class TableActivity extends JTable
{
    var $activityID = null;
    var $activity = null;

```

```

// Override the class constructor. it takes a database object as a parameter
// and will call the parent constructor using the name of the database table,
// the primary key and database object
function __construct(&$db)
{
    parent::__construct( '#_activity', 'activityID', $db);
}
}

```

com_flyer\tables\agegroups.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-13 09:01:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is an extension of the JTable specific to the jos_agegroups table
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC') or die ('Restricted Access');
class TableAgegroups extends JTable
{
    var $ageID = null;
    var $lowAge = null;
    var $upperAge = null;
    var $span = null;

    // Override the class constructor. it takes a database object as a parameter
    // and will call the parent constructor using the name of the database table,
    // the primary key and database object
    function __construct(&$db)
    {
        parent::__construct( '#_agegroups', 'ageID', $db);
    }
}

```

com_flyer\tables\event_has_many.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-08-28 6:36:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is an extension of the JTable specific to the jos_agegroups table
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC') or die ('Restricted Access');
class TableEvent_has_many extends JTable
{
    var $id = null;
    var $eventID = null;
    var $distanceID = null;

    // Override the class constructor. it takes a database object as a parameter
    // and will call the parent constructor using the name of the database table,
    // the primary key and database object
    function __construct(&$db)
    {
        parent::__construct( '#_event_has_many', 'id', $db);
    }
}

```

```
}
```

com_flyer\tables\events.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-06-29 08:22:PST
 * @updated 1   2009-06-29 10:04:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is an extension of the JTable specific to the jos_events table
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );
class TableEvents extends JTable
{
    var $eventID = null;
    var $eventLogo = null;
    var $eventName = null;
    var $eventPlace = null;
    var $eventType = null;
    var $eventDescription = null;
    var $startTime = null;
    var $eventInfo = null;
    var $eventDate = null;
    var $eventAdministrator = null;
    var $orgID = null;
    var $published = null;

    // Override the class constructor. it takes a database object as a parameter
    // and will call the parent constructor using the name of the database table,
    // the primary key and database object
    function __construct( &$db )
    {
        parent::__construct( '#_events', 'eventID', $db );
    }
}
}
```

Source Code for com_manualreg

com_manualreg\controller.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-05 18:14:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This controller will responde accordinly to user actions. The controller will
 * detect which request has been made and respond appropriately by triggering the
 * model to manipulate the data appropriately and passing the model into the view
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Include the path to the TableEvents class in the events.php
JTable::addIncludePath( JPATH_COMPONENT_DS . 'tables' );

// Import Joomla! 1.5 controller from the framework
jimport( 'joomla.application.component.controller' );

// This class implements the edit(), add() and save() funtions
```

```

class ManualregController extends JController
{
    var $_Agegroup = null;

    function edit()
    {
        // we set the view environment variable to single
        JRequest::setVar('view', 'updaterunner');
        // load the view.html.php for single records
        $this->display();
    }

    // Activates the new button on the default view
    function add()
    {
        // we set the view environment variable to single
        JRequest::setVar('view', 'updaterunner');
        // load the view.html.php for single records
        $this->display();
    }

    // This function sets the manual registration page.
    function select()
    {
        $eventID = JRequest::getInt('race$', '', 'post', 'string');

        if ($eventID != '' || $eventID != 0)
        {
            JRequest::setVar('view', 'addrunner');
            $this->display();
        }
        else
        {
            JRequest::setVar('view', 'selectevent');
            $this->display();
        }
    }

    function save()
    {
        global $option;
        $row=& JTable::getInstance('person_reg_event', 'Table');
        if (!$row->bind(JRequest::get('post')))
        {
            JRequest::raiseError(500, $row->getError());
        }
        if (!$row->store())
        {
            JRequest::raiseError(500, $row->getError());
        }
        JTable::addIncludePath(DS.'com_registrations'.DS.'tables');
        $row2=& JTable::getInstance('persons', 'Table');
        if (!$row2->bind(JRequest::get('post')))
        {
            JRequest::raiseError(500, $row2->getError());
        }
        if (!$row2->store())
        {
            JRequest::raiseError(500, $row2->getError());
        }
        $eventID = JRequest::getInt('eventID');
        $link = 'index.php?option=com_manualreg&view=addrunner&event_ID=' .
$eventID;
        $this->setRedirect($link, $msg, '');
        return;
    }

    function saverunner()
    {
        JTable::addIncludePath(JPATH_COMPONENT.DS.'tables');
        $row =& JTable::getInstance('persons', 'Table');
        $row2 =& JTable::getInstance('person_reg_event', 'Table');
    }
}

```

```

$post =& JRequest::get('post');
$eventID = JRequest::getInt('eventID');
$event_ID = JRequest::getInt('event_ID');
$bib = JRequest::getVar('bib', '', 'post', 'string');
$firstName = JRequest::getVar('firstName', '', 'post', 'string');
$lastName = JRequest::getVar('lastName', '', 'post', 'string');

if($firstName == '' || $lastName == '')
{
    $msg = 'Make sure that the form is filled out correctly. Try Again';
    $link = 'index.php?option=com_manualreg&view=addrunner&eventID=' .
$eventID;
    $this->setRedirect($link, $msg, 'error');
    return;
}

$sql1 = "SELECT bib from #__person_reg_event WHERE bib='$bib' AND
eventID='$eventID'";
$db1 =& JFactory::getDBO();
$db1->setQuery($sql1);
$result = $db1->loadResult();
if($result == $bib)
{
    $msg = 'bib is already in use. Try Again';
    $link = 'index.php?option=com_manualreg&view=addrunner&eventID=' .
$eventID;
    $this->setRedirect($link, $msg, 'error');
    return;
}
if(($eventID == 0 || $eventID == '') && ($event_ID == 0 || $event_ID == ''))
{
    $msg = 'You need to pick an event. Try Again';
    $link = 'index.php?option=com_manualreg&view=selectevent&eventID=' .
$eventID;
    $this->setRedirect($link, $msg, 'error');
    return;
}

else
{
    if (!$row->bind(JRequest::get('post')))
    {
        JError::raiseError(500, $row->getError());
    }
    if (!$row->store())
    {
        JError::raiseError(500, $row->getError());
    }

    if($eventID != '' || $eventID != 0)
    {
        //$gend = null;
        $personID = $row->personID;
        $sql2 = "SELECT dob from #__persons WHERE personID = '$personID'";
        $sql3 = "SELECT event_Date from #__events WHERE eventID =
'$eventID'";
        $db2 =& JFactory::getDBO();
        $db2->setQuery($sql2);
        $res1 = $db2->loadResult();
        $bday = (string)$res1;
        $db3 =& JFactory::getDBO();
        $db3->setQuery($sql3);
        $res2 = $db3->loadResult();
        $rday = (string)$res2;
        $age = $this->birthday($bday, $rday);
        $gen = JRequest::getVar('gender', '', 'post', 'string');
        //JError::raiseError(500, $gen);
        $gend = null;
        if($gen == 'male')
        {
            $gend = 'M';
        }
    }
}

```

```

    }
    else if($gen == 'female')
    {
        $gend = 'F';
    }
    $mygroup = $this->getAgegroup($eventID, $age, $gend);
    $row2->eventID = $eventID;
    $row2->personID = $personID;
    $row2->bib = $bib;
    $row2->agegroup = $mygroup;
    $row2->raceDayAge = $age;
    $row2->has_paid = JRequest::getInt('paid');

    if (! $row2->bind(JRequest::get('post')))
    {
        JError::raiseError(500, $row->getError());
    }
    if (! $row2->store())
    {
        JError::raiseError(500, $row2->getError());
    }
    $link = 'index.php?option=com_manualreg&view=addrunner&event_ID=' .
        $eventID;
    $this->setRedirect($link, $msg, '');
    return;
}
else
{
    // $gend = null;
    $eventID = JRequest::getInt('event_ID');
    $sql4 = "SELECT dob from #__persons WHERE personID = '$personID'";
    $sql5 = "SELECT event_Date from #__events WHERE eventID =
        '$eventID'";
    $db4 =& JFactory::getDBO();
    $db4->setQuery($sql4);
    $res1 = $db4->loadResult();
    $bday = (string)$res1;
    $db5 =& JFactory::getDBO();
    $db5->setQuery($sql5);
    $res2 = $db5->loadResult();
    $rday = (string)$res2;
    $age = $this->birthday($bday, $rday);
    $gen = JRequest::getInt('gender', '', 'post', 'string');
    $gend = null;
    if($gen == 'male')
    {
        $gend = 'M';
    }
    else
    {
        $gend = 'F';
    }
    $mygroup = $this->getAgegroup($eventID, $age, $gend);

    // $mygroup = $gend . $mygroup1;
    JError::raisWarning(500, JText::_(' $mygroup = ' . $mygroup));
    $personID = $row->personID;
    $row2->eventID = $eventID;
    $row2->personID = $personID;
    $row2->bib = $bib;
    $row2->agegroup = $mygroup;
    $row2->has_paid = JRequest::getInt('paid');
    if (! $row2->store())
    {
        JError::raiseError(500, $row2->getError());
    }
    $link = 'index.php?option=com_manualreg&view=addrunner&event_ID=' .
        $event_ID;
    $this->setRedirect($link, $msg, '');
}
}
}

```

```

}
function &getAgegroup($eventid, $age, $gend)
{
    $sql3 = "SELECT lowAge, upperAge, span FROM #__agegroups
            WHERE ageID='$eventid'";
    $db2 =& JFactory::getDBO();
    $db2->setQuery($sql3);
    $data = $db2->loadRow();
    $low = $data['0'];
    $high = $data['1'];
    $span = $data['2'];

    if ($age < $low)
    {
        $strng1 = "0 - ";
        $strng2 = $low -1;
        $this->_agegroup = $gend.$strng1.$strng2;
        return $this->_agegroup;
        break;
    }
    if ($age >= $high)
    {
        $strng3 = $high;
        $strng4 = " - 100+";
        $this->_agegroup = $gend.$strng3.$strng4;
        return $this->_agegroup;
        break;
    }
    if ($age > $low)
    {
        for($i = $low; $i <= $high; $i += $span -1)
        {
            if($i < $age)
            {
                $i++;
            }
            else if($i == $age)
            {
                $strng5 = $i;
                $strng6 = " - ";
                $strng7 = $i + $span-1;
                $this->_agegroup = $gend.$strng5.$strng6.$strng7;
                return $this->_agegroup;
                break;
            }
            else
            {
                $strng8 = $i-$span;
                $strng9 = " - ";
                $strng10 = $i -1;
                $this->_agegroup = $gend.$strng8.$strng9.$strng10;
                return $this->_agegroup;
                echo '<br />';
                break;
            }
        }
    }
}
//calculate Race Day Age based on DOB
function birthday ($bday, $rday)
{
    list($year1,$month1,$day1) = explode("-", $bday);
    list($year2,$month2,$day2) = explode("-", $rday);
    $year_diff = $year2 - $year1;
    $month_diff = $month2 - $month1;
    $day_diff = $day2 - $day1;
    if ($month_diff < 0) $year_diff--;
    elseif (($month_diff==0) && ($day_diff < 0)) $year_diff--;
    return $year_diff;
}

```



```

    }

function display()
{
    $view = JRequest::getVar('view');
    if (! $view)
    {
        JRequest::setVar('view', 'selectevent');
    }

    parent::display();
}
}

```

com_manualreg\manualreg.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-06-29 08:14:PST
 *             Last Update: 2009-07-05 21:30:00:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This is the entry point to the extension to create a flyer for the system. The
 * extension will allow administrators to create event flyers for their events.
 * The administrators will also decide when the flyer will be displayed, and for
 * for how long. They can also edit the flyers they have previouslu created.
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// This is a call to include the controller, and make sure it is only called once
require_once( JPATH_COMPONENT_DS.'controller.php' );

// This code creates the toolbar for the different windows we are using. The
// JToolBarHelper contains functions that produces all the HTML code necessary

switch ( $task )
{
    case 'edit':

    case 'add':
        JToolBarHelper::save();
        JToolBarHelper::apply();
        JToolBarHelper::cancel();
        break;
    default:

        JToolBarHelper::publishList();
        JToolBarHelper::unpublishList();
        JToolBarHelper::editList();
        JToolBarHelper::deleteList();
        //JToolBarHelper::addNew();

        break;
}

// Create an instance of the controller
$controller = new ManualregController();

// Execute the task variable chosen
$controller->execute( $task );

// Redirect after the task is finish its execution.
$controller->redirect();

```

com_manualreg\models\addrunner.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-05 18:10:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport( 'joomla.application.component.model' );

class ManualregModelAddrunner extends JModel
{
    // initiate the data variable

    var $data = null;
    var $_pagination = null;
    var $_total = null;
    var $_agegroup = null;
    var $_eventID = null;

    function buildSearch()
    {
        $eventID = JRequest::getInt( 'eventID', '', 'post', 'string' );
        if( $eventID == 0 || $eventID == '' )
        {
            $eventID = JRequest::getInt( 'event_ID' );
            if( $eventID == 0 || $eventID == '' )
            {
                $eventID = JRequest::getInt( 'eventID' );
            }
        }
        return "SELECT * FROM #__persons AS a, #__events as b,
                #__person_reg_event AS c
                WHERE a.personID = c.personID AND b.eventID = c.eventID
                AND c.eventID = '$eventID' ORDER BY bib DESC";
    }

    function getTotal()
    {
        if( !$this->_total )
        {
            $query = $this->buildSearch();
            $this->_total = $this->getListCount( $query );
        }
        return $this->_total;
    }

    function &getPagination()
    {
        if( !$this->_pagination )
        {
            jimport( 'joomla.html.pagination' );
            global $mainframe;
            $this->_pagination = new JPagination( $this->getTotal(),
            JRequest::getVar( 'limitstart', 0 ), JRequest::getVar( 'limit',
            $mainframe->getConfig( 'list_limit' ) );
        }
        return $this->_pagination;
    }

    function &getData()
    {

```

```

$pagination =& $this->getPagination();
if(empty($this->_data))
{
    $query = $this->buildSearch();
    $this->_data = $this->getList($query, $pagination->limitstart,
        $pagination->limit);
}
return $this->_data;
}

function &getAgegroup()
{
    if (empty( $this->_agegroup ))
    {
        //$eventID = JRequest::getInt('eventID');
        //$myTest = JRequest::getVar('usrName');
        //$myTest = 'jo60nor';
        $sql1 = "SELECT eventID FROM #__person_req_event WHERE
            eventID='$eventID'";
        $sql2 = "SELECT raceDayAge from #__Persons WHERE userName='$myTest'";

        $db =& $this->getDBO();
        $db->setQuery($sql2);
        $age = $db->loadResult();
        $db1 =& $this->getDBO();
        $db1->setQuery($sql1);
        $ageEvent = $db1->loadResult();
        $db2 =& $this->getDBO();
        $sql3 = "SELECT lowAge, upperAge, span FROM #__agegroups
            WHERE ageID='$ageEvent'";
        $db2->setQuery($sql3);
        $data = $db2->loadRow();
        $low = $data['0'];
        $high = $data['1'];
        $span = $data['2'];
        //$this->_agegroup = 44;
        //return $this->_agegroup;
        if ($age < $low)
        {
            $strng1 = "0 - ";
            $strng2 = $low -1;
            $this->_agegroup = $strng1.$strng2;
            return $this->_agegroup;
            break;
        }
        if ($age > $high)
        {
            $strng3 = $high;
            $strng4 = " - 100+";
            $this->_agegroup = $strng3.$strng4;
            return $this->_agegroup;
            break;
        }
        if ($age > $low)
        {
            for($i = $low; $i <= $high; $i += $span -1)
            {
                if($i < $age)
                {
                    $i++;
                }
                else if($i == $age)
                {
                    $strng5 = $i;
                    $strng6 = " - ";
                    $strng7 = $i + $span-1;
                    $this->_agegroup = $strng5.$strng6.$strng7;
                    return $this->_agegroup;
                }
            }
        }
    }
}

```



```

class ManualregViewAddrunner extends JView
{
    function display($tpl = null)
    {
        $eventID = JRequest::getInt('races', '', 'post', 'string');
        if($eventID == '' || $eventID == 0)
        {
            $eventID = JRequest::getInt('eventID');
            if($eventID == '' || $eventID == 0)
            {
                $eventID = JRequest::getInt('event_ID');
            }
        }
        $query = "SELECT eventName from #__events WHERE eventID = '$eventID'";
        $db = & JFactory::getDBO();
        $db->setQuery($query);
        $eventName = $db->loadResult();
        JToolBarHelper::title(JText::_('Add Participants to ' . $eventName),
            'generic.png');

        $this->assignRef('eventID', $eventID);

        // Calls the getEvent
        $rows =& $this->get('data');

        $state = array(
            JHTML::_('select.option', null, JText::_('Select State')),
            JHTML::_('select.option', 'AL', JText::_('AL')),
            JHTML::_('select.option', 'AK', JText::_('AK')),
            JHTML::_('select.option', 'AR', JText::_('AR')),
            JHTML::_('select.option', 'AZ', JText::_('AZ')),
            JHTML::_('select.option', 'AB', JText::_('AB')),
            JHTML::_('select.option', 'BC', JText::_('BC')),
            JHTML::_('select.option', 'CA', JText::_('CA')),
            JHTML::_('select.option', 'CO', JText::_('CO')),
            JHTML::_('select.option', 'CT', JText::_('CT')),
            JHTML::_('select.option', 'DE', JText::_('DE')),
            JHTML::_('select.option', 'FL', JText::_('FL')),
            JHTML::_('select.option', 'GA', JText::_('GA')),
            JHTML::_('select.option', 'HI', JText::_('HI')),
            JHTML::_('select.option', 'IA', JText::_('IA')),
            JHTML::_('select.option', 'ID', JText::_('ID')),
            JHTML::_('select.option', 'IL', JText::_('IL')),
            JHTML::_('select.option', 'IN', JText::_('IN')),
            JHTML::_('select.option', 'KS', JText::_('KS')),
            JHTML::_('select.option', 'KY', JText::_('KY')),
            JHTML::_('select.option', 'LA', JText::_('LA')),
            JHTML::_('select.option', 'MA', JText::_('MA')),
            JHTML::_('select.option', 'MB', JText::_('MB')),
            JHTML::_('select.option', 'MD', JText::_('MD')),
            JHTML::_('select.option', 'ME', JText::_('ME')),
            JHTML::_('select.option', 'MI', JText::_('MI')),
            JHTML::_('select.option', 'MN', JText::_('MN')),
            JHTML::_('select.option', 'MO', JText::_('MO')),
            JHTML::_('select.option', 'MS', JText::_('MS')),
            JHTML::_('select.option', 'MT', JText::_('MT')),
            JHTML::_('select.option', 'NB', JText::_('NB')),
            JHTML::_('select.option', 'NC', JText::_('NC')),
            JHTML::_('select.option', 'ND', JText::_('ND')),
            JHTML::_('select.option', 'NE', JText::_('NE')),
            JHTML::_('select.option', 'NH', JText::_('NH')),
            JHTML::_('select.option', 'NJ', JText::_('NJ')),
            JHTML::_('select.option', 'NL', JText::_('NL')),
            JHTML::_('select.option', 'NM', JText::_('NM')),
            JHTML::_('select.option', 'NS', JText::_('NS')),
            JHTML::_('select.option', 'NT', JText::_('NT')),
            JHTML::_('select.option', 'NU', JText::_('NU')),
            JHTML::_('select.option', 'NV', JText::_('NV')),
            JHTML::_('select.option', 'NY', JText::_('NY')),
            JHTML::_('select.option', 'OH', JText::_('OH')),
            JHTML::_('select.option', 'OK', JText::_('OK')),
        );
    }
}

```

```

        JHTML::_('select.option', 'ON', JText::_('ON')),
        JHTML::_('select.option', 'OR', JText::_('OR')),
        JHTML::_('select.option', 'PA', JText::_('PA')),
        JHTML::_('select.option', 'PE', JText::_('PE')),
        JHTML::_('select.option', 'QC', JText::_('QC')),
        JHTML::_('select.option', 'RI', JText::_('RI')),
        JHTML::_('select.option', 'SC', JText::_('SC')),
        JHTML::_('select.option', 'SD', JText::_('SD')),
        JHTML::_('select.option', 'SK', JText::_('SK')),
        JHTML::_('select.option', 'TN', JText::_('TN')),
        JHTML::_('select.option', 'TX', JText::_('TX')),
        JHTML::_('select.option', 'UT', JText::_('UT')),
        JHTML::_('select.option', 'VA', JText::_('VA')),
        JHTML::_('select.option', 'VT', JText::_('VT')),
        JHTML::_('select.option', 'WA', JText::_('WA')),
        JHTML::_('select.option', 'WI', JText::_('WI')),
        JHTML::_('select.option', 'WV', JText::_('WV')),
        JHTML::_('select.option', 'WY', JText::_('WY')),
        JHTML::_('select.option', 'YT', JText::_('YT')),
        JHTML::_('select.option', 'XX', JText::_('No State'))
    );
    $gender[] = JHTML::_('select.option', 'male', 'Male:');
    $gender[] = JHTML::_('select.option', 'female', 'Female:');
    $lists['gender'] = JHTML::_('select.radiolist', $gender, 'gender',
        'class="inputbox" ', 'value', 'text', '', 'gender');
    $this->assignRef('lists', $lists);
    $this->assignRef('state', JHTML::_('select.genericList',
        $state, $state, 'class="inputbox" ', 'value',
        'text'));
    $paid = null;
    $pagination = & $this->get('pagination');

    // Assign the data to the rows
    $this->assignRef('rows', $rows);
    //$this->assignRef('eventID', $eventID);
    $this->assignRef('pagination', $pagination);
    $this->assignRef('paid', JHTML::_('select.booleanlist',
        'paid', 'class="inputbox"', $paid));
    parent::display($tpl);
}
}

```

com_manualreg\views\addrunner\tmpl\default

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

JHTML::_('behavior.calendar');
$event_ID = (int) JRequest::getVar('event_ID', 0);
$eventID = $this->eventID;
?>
<form action="index.php?option=com_manualreg&task=saverunner"
method="post" id="josForm" name="josForm" style="font-size: 8pt;">
<table cellpadding="2">
<tr>
<td>
<td>
        Check Database:
    </td>
<td>
        <?php
            $link = JRoute::_('index.php?option=com_mysearch&eventID=' .
                $eventID);
            echo '<a href="' . $link . '>' . '<strong>' . 'Check Database'
                . '</strong>' . '</a>';
        ?>
    </td>
</tr>
</table>

```

```

<tr>
  <td width="80" align="left" class="key">
    First Name:
  </td>
  <td>
    <input class="text_area" type="text" name="firstName"
      id="firstName" size="20" maxlength="50"
      value="" />
  </td>
  <td width="80" align="right" class="key">
    Middle Name:
  </td>
  <td>
    <input class="text_area" type="text" name="middleName"
      id="middleName" size="20" maxlength="50"
      value="" />
  </td>
  <td width="80" align="right" class="key">
    Last Name:
  </td>
  <td>
    <input class="text_area" type="text" name="lastName"
      id="lastName" size="20" maxlength="50"
      value="" />
  </td>
</tr>
<tr>
  <td width="80" align="left" class="key">
    Address:
  </td>
  <td colspan="5" align="left" class="key">
    <input class="text_area" type="text" name="address"
      id="address" size="116" maxlength="250"
      value="" />
  </td>
</tr>
<tr>
  <td width="80" align="left" class="key">
    City:
  </td>
  <td>
    <input class="text_area" type="text" name="city"
      id="city" size="20" maxlength="50"
      value="" />
  </td>
  <td width="80" align="right" class="key">
    State:
  </td>
  <td>
    <?php echo $this->state; ?>
  </td>
  <td width="80" align="right" class="key">
    Zip:
  </td>
  <td>
    <input class="text_area" type="text" name="zip"
      id="zip" size="20" maxlength="50"
      value="" />
  </td>
</tr>
<tr>
  <td>
    Email Address:
  </td>
  <td colspan="3">
    <input class="text_area" type="text" name="email"
      id="email" size="50" maxlength="50"
      value="" />
  </td>
</tr>
<tr>
  <td width="80" align="left" class="key">

```

```

        Date of Birth:<br>
        YYYY-MM-DD
    </td>
    <td colspan="2">
        <input type="text" name="dob" id="dob" value="" />
        
    </td>
</tr>

<tr>
    <td>
        Gender:
    </td>
    <td>
        <?php echo $this->lists['gender']; ?>
    </td>
</tr>

<tr>
    <td>
        BiB:
    </td>
    <td>
        <input class="text_area" type="text" name="bib"
            id="bib" size="5" maxlength="5"
            value="" />
    </td>
</tr>

<tr>
    <td>
        Has Paid:
    </td>
    <td>
        <?php echo $this->paid; ?>
    </td>
</tr>
</table>

<button class="button validate" type="submit"><?php echo JText::_('Add
    Partisipant'); ?></button>

<input type="hidden" name="option" value="com_manualreg" />
<input type="hidden" name="task" value="saverunner" />
<input type="hidden" name="eventID" value="<?php echo $eventID; ?>" />
<input type="hidden" name="event_ID" value="<?php echo $event_ID; ?>" />
<?php echo JHTML::_('form.token'); ?>
</form>
<br />
<fieldset>
    <legend>List Partisipants</legend>
    <table class="adminlist">
        <thead>
            <tr>
                <th width="4%">Bib</th>
                <th width="10%">First Name</th>
                <th width="10%">Middle name</th>
                <th width="10%">Last name</th>
                <th width="15%">Address</th>
                <th width="10%">City</th>
                <th width="3%">State</th>
                <th width="3%">Zip</th>
                <th width="5%">Gender</th>
                <th width="10%">Date of Birth</th>
                <th width="3%">Age</th>
                <th width="7%">Age Group</th>
                <th width="5" nowrap="nowrap">Has Paid</th>
            </tr>
        </thead>
        <?php
        jimport('joomla.filter.output');
        // Variabe that creates different CSS. 1 or 0 to distinguish rows

```



```

$k = 0;

// Loop through the database as long as there are records
for($i=0, $n=count($this->rows); $i < $n; $i++)
{
    // reference to the current row
    $row =&$this->rows[$i];
    $published = JHTML::_('grid.published', $row, $i);
    $link = JFilterOutput::ampReplace( 'index.php?option=' .
        $option . '&task=edit&cid[]=' . $row->id);

?>

<tr class="<?php echo "row$k"; ?>">
    <td>
        <?php echo $row->bib; ?>
    </td>
    <td>
        <a href="<?php echo $link; ?>"><?php echo $row->firstName;
            ?></a>
    </td>
    <td>
        <?php echo $row->middleName; ?>
    </td>
    <td>
        <?php echo $row->lastName; ?>
    </td>
    <td>
        <?php echo $row->address; ?>
    </td>
    <td>
        <?php echo $row->city; ?>
    </td>
    <td>
        <?php echo $row->state; ?>
    </td>
    <td>
        <?php echo $row->zip; ?>
    </td>
    <td>
        <?php echo $row->gender; ?>
    </td>
    <td>
        <?php echo $row->dob; ?>
    </td>
    <td>
        <?php echo $row->raceDayAge; ?>
    </td>
    <td>
        <?php echo $row->agegroup; ?>
    </td>
    <td>
        <?php $pay = $row->has_paid;
            if($pay == 0)
            {
                ?>
                <strong style="color: red">No</strong>
                <?php }

                else
                {
                    echo 'yes';
                }
                ?>
    </td>
</tr>
<?php

// Switch k between 1 and 0 for different CSS
$k = 1 - $k;
}
?>

```

```

        <tfoot>
            <tr>
                <td colspan="14"><?php echo $this->pagination->getListFooter();
                    ?></td>
            </tr>
        </tfoot>
    </table>
</fieldset>

```

com_manualreg\views\selectevent\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-07 07:40:PST
 * @package    Joomla! 1.5
 * @subpackage Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This file will define view for manual registrations application.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

// The Joomla! 1.5 convention is component name, view and the name of the view
class ManualregViewSelectevent extends JView
{
    function display($tpl = null)
    {
        JToolBarHelper::title( JText::_('Select Event'), 'generic.png');
        $events =& $this->get('event');
        $races[] = JHTML::_('select.option', '0',
            JText::_('Select Event'), 'value', 'text' );
        $races = array_merge( $races, $events );
        $lists['races'] = JHTML::_('select.genericlist',
            $races, 'races', 'class="inputbox" size="1" ', 'value', 'text');
        $this->assignRef('lists', $lists);
        parent::display($tpl);
    }
}

```

com_manualreg\views\selectevent\tmpl\default.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
?>

<form action="index.php?option=com_manualreg&task=select"
    method="post" id="josForm" name="josForm" style="font-size: 8pt;">

    <?php echo $this->lists['races']; ?>

    <button class="button validate" type="submit"><?php echo JText::_('Submit'); ?></button>
    <input type="hidden" name="option" value="com_manualreg" />
    <input type="hidden" name="task" value="select" />
    <?php echo JHTML::_('form.token'); ?>

</form>

```

com_manualreg\views\updaterunner\view.html.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-12-05 08:17:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This file will define addrunner view for manual registrations application.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

// The Joomla! 1.5 convention is component name, view and the name of the view
class ManualregViewUpdaterunner extends JView
{
    function display($tpl = null)
    {
        // Create an instance of the TableFlyer object
        $row =& JTable::getInstance('person_reg_event', 'Table');
        // Create an instance of the agegroups table object
        $row2=& JTable::getInstance('persons', 'Table');
        $cid = JRequest::getVar( 'cid', array(0), '', 'array' );
        // Since we want to edit one record at the time, we select the first ID in
array
        $id = $cid[0];
        $row->load($id);
        $sql1 = "SELECT eventID, personID FROM #__person_reg_event WHERE id =
'$id'";
        $db1 =& JFactory::getDBO();
        $db1->setQuery($sql1);
        $result1 = $db1->loadRow();
        $eventID = $result1['0'];
        $personID = $result1['1'];
        $sql2 = "SELECT * from #__persons WHERE personID = '$personID'";
        $db2 =& JFactory::getDBO();
        $db2->setQuery($sql2);
        $row2 = $db2->loadObject();
        JToolBarHelper::title( JText::_('Update ' . $row2->firstName . ' ' .
        $row2->lastName), 'generic.png');
        $gender[] = JHTML::_('select.option', 'male', 'Male:');
        $gender[] = JHTML::_('select.option', 'female', 'Female:');
        $lists['gender'] = JHTML::_('select.radiolist', $gender, 'gender',
        'class="inputbox" ', 'value', 'text', $row2->gender, 'gender');
        $this->assignRef('lists', $lists);
        $this->assignRef('row', $row);
        $this->assignRef('eventID', $eventID);
        $this->assignRef('row2', $row2);
        $this->assignRef('has_paid', JHTML::_('select.booleanlist',
        'has_paid', 'class="inputbox"', $row->has_paid));

        parent::display($tpl);
    }
}
}
```

com_manualreg\views\updaterunner\tmpl\default.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-12-14 09:07:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is an interface to contains the form data for the addEvents to the system.
 */
```

```

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// This prepare the JavaScript Calendar available to be used in the form
JHTML::_('behavior.calendar');
?>

<form action="index.php" method="post" name="adminForm" id="adminForm">
  <fieldset class="adminform">
    <legend>Update Registration Data</legend>
    <table class="admintable">
      <tr>
        <td width="50" align="left" class="key">
          First Name:
        </td>
        <td>
          <input class="text_area" type="text" name="firstName"
            id="firstName" size="20" maxlength="50"
            value="<?php echo $this->row2->firstName; ?>" />
        </td>
        <td width="50" align="right" class="key">
          Middle Name:
        </td>
        <td>
          <input class="text_area" type="text" name="middleName"
            id="middleName" size="20" maxlength="50"
            value="<?php echo $this->row2->middleName; ?>" />
        </td>
        <td width="50" align="right" class="key">
          Last Name:
        </td>
        <td>
          <input class="text_area" type="text" name="lastName"
            id="lastName" size="20" maxlength="50"
            value="<?php echo $this->row2->lastName; ?>" />
        </td>
      </tr>
      <tr>
        <td width="50" align="left" class="key">
          Address:
        </td>
        <td colspan="2">
          <input class="text_area" type="text" name="address"
            id="address" size="40" maxlength="250"
            value="<?php echo $this->row2->address; ?>" />
        </td>
      </tr>
      <tr>
        <td width="80" align="left" class="key">
          City:
        </td>
        <td>
          <input class="text_area" type="text" name="city"
            id="city" size="20" maxlength="50"
            value="<?php echo $this->row2->city; ?>" />
        </td>
        <td width="80" align="right" class="key">
          State:
        </td>
        <td>
          <input class="text_area" type="text" name="state"
            id="state" size="20" maxlength="50"
            value="<?php echo $this->row2->state; ?>" />
        </td>
        <td width="80" align="right" class="key">
          Zip:
        </td>
        <td>
          <input class="text_area" type="text" name="zip"
            id="zip" size="20" maxlength="50"
            value="<?php echo $this->row2->zip; ?>" />
        </td>
      </tr>
    </table>
  </fieldset>
</form>

```

```

        </td>
</tr>
<tr>
    <td width="80" align="right" class="key">
        Email Address:
    </td>
    <td colspan="3">
        <input class="text_area" type="text" name="email"
            id="email" size="50" maxlength="50"
            value="<?php echo $this->row2->email; ?>" />
    </td>
</tr>
<tr>
    <td width="80" align="left" class="key">
        Date of Birth:<br>
        YYYY-MM-DD
    </td>
    <td colspan="2">
        <input type="text" name="dob" id="dob" value="<?php echo
            $this->row2->dob; ?>" />
        
    </td>
</tr>
<tr>
    <td width="80" align="left" class="key">
        Gender:
    </td>
    <td>
        <?php echo $this->lists['gender']; ?>
    </td>
</tr>
<tr>
    <td width="80" align="left" class="key">
        BiB:
    </td>
    <td>
        <input class="text_area" type="text" name="bib"
            id="bib" size="5" maxlength="5"
            value="<?php echo $this->row->bib; ?>" />
    </td>
</tr>
<tr>
    <td width="80" align="left" class="key">
        Has Paid:
    </td>
    <td>
        <?php echo $this->has_paid; ?>
    </td>
</tr>
</table>
</fieldset>

    <input type="hidden" name="eventID" value="<?php echo $this->eventID; ?>" />
    <input type="hidden" name="id" value="<?php echo $this->row->id; ?>" />
    <input type="hidden" name="personID" value="<?php echo $this->row2->personID;
        ?>" />
    <input type="hidden" name="option" value="<?php echo $option;?>" />
    <input type="hidden" name="task" value="" />
    <?php echo JHTML::_('form.token'); ?>
</form>

```

com_manualreg\tables\person_reg_event.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-07 15:22:PST
 * @updated 1   2009-06-29 10:04:PST

```

```

* @package      Joomla! 1.5
* @subpackage   Manualreg
* @Copyright    Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
* This is an extention of the JTable specific to the jos_events table
*/

```

```

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC')or die ('Restricted Access');

```

```

class TablePerson_reg_event extends JTable
{
    var $id = null;
    var $eventID = null;
    var $personID = null;
    var $bib = null;
    var $raceDayAge = null;
    var $agegroup = null;
    var $has_paid = null;
    var $clock_time = null;
    var $pace = null;
    var $sizeID = null;

    // Override the class constructor. it takes a database object as a parameter
    // and will call the parent constructor using the name of the database table,
    // the primary key and database object
    function __construct(&$db)
    {
        parent::__construct( '#__person_reg_event', 'id', $db);
    }
}

```

com_manualreg\tables\person.php

```

<?php
/**
* @Author      Steinar Johansen
* @version     Version 1 created 2009-06-29 08:22:PST
* @updated 1   2009-06-29 10:04:PST
* @package     Joomla! 1.5
* @subpackage  Events
* @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
* This is an extention of the JTable specific to the jos_events table
*/

```

```

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC')or die ('Restricted Access');

```

```

class TablePersons extends JTable
{
    var $personID = null;
    var $userName = null;
    var $password = null;
    var $firstName = null;
    var $middleName = null;
    var $lastName = null;
    var $address = null;
    var $city = null;
    var $state = null;
    var $zip = null;
    var $email = null;
    var $dayPhone = null;
    var $eveningPhone = null;
    var $dob = null;
    var $gender = null;
    var $registerDate = null;
    var $lastVisitDate = null;
    var $ageGroupID = null;
    var $role = null;
    var $published = null;

    // Override the class constructor. it takes a database object as a parameter
    // and will call the parent constructor using the name of the database table,

```

```

// the primary key and database object
function __construct(&$db)
{
    parent::__construct( '#__persons', 'personID', $db);
}

/**
 * Validation and filtering
 *
 * @return boolean True is satisfactory
 */
function check()
{
    jimport('joomla.mail.helper');

    // Validate user information
    if (trim( $this->firstName) == '' ) {
        $this->setError( JText::_ ( 'Please enter your first name.'));
        return false;
    }

    if (trim( $this->lastName) == '' ) {
        $this->setError( JText::_ ( 'Please enter your last name'));
        return false;
    }

    if (trim( $this->address) == '' ) {
        $this->setError( JText::_ ( 'Please enter your address'));
        return false;
    }

    if (trim( $this->city) == '' ) {
        $this->setError( JText::_ ( 'Please enter a city'));
        return false;
    }

    if (trim( $this->zip) == '' ) {
        $this->setError( JText::_ ( 'Please enter a zip code'));
        return false;
    }

    if (trim( $this->dob) == '' ) {
        $this->setError( JText::_ ( 'Please enter a date of birth'));
        return false;
    }

    if (trim( $this->dayPhone) == '' ) {
        $this->setError( JText::_ ( 'Please enter a phone number'));
        return false;
    }

    if (trim( $this->eveningPhone) == '' ) {
        $this->setError( JText::_ ( 'Please enter an evening phone
            number'));
        return false;
    }

    if (trim( $this->username ) == '' ) {
        $this->setError( JText::_ ( 'Please enter a user name.' ) );
        return false;
    }

    if (preg( "[<>\"'%;()&]", $this->username) ||
        strlen(utf8_decode($this->username )) < 2) {
        $this->setError( JText::sprintf( 'VALID_AZ09', JText::_ (
            'Username' ), 2 ) );
        return false;
    }

    if ((trim($this->email) == "") || ! JMailHelper::isEmailAddress(
        $this->email) ) {
        $this->setError( JText::_ ( 'WARNREG_MAIL' ) );
    }
}

```

```

        return false;
    }

    if ($this->registerDate == null) {
        // Set the registration timestamp
        $now =& JFactory::getDate();
        $this->registerDate = $now->toMySQL();
    }

    // check for existing username
    $query = 'SELECT id'
        . ' FROM #__persons '
        . ' WHERE username = ' . $this->_db->Quote($this->username)
        . ' AND id != ' . (int) $this->id;
    ;
    $this->_db->setQuery( $query );
    $xid = intval( $this->_db->loadResult() );
    if ($xid && $xid != intval( $this->id )) {
        $this->setError( JText::_('WARNREG_INUSE'));
        return false;
    }

    // check for existing email
    $query = 'SELECT id'
        . ' FROM #__persons '
        . ' WHERE email = ' . $this->_db->Quote($this->email)
        . ' AND id != ' . (int) $this->id;
    ;
    $this->_db->setQuery( $query );
    $xid = intval( $this->_db->loadResult() );
    if ($xid && $xid != intval( $this->id )) {
        $this->setError( JText::_('WARNREG_EMAIL_INUSE' ) );
        return false;
    }

    return true;
}

function setLastVisit( $timeStamp=null, $id=null )
{
    // check for User ID
    if (is_null( $id )) {
        if (isset( $this )) {
            $id = $this->id;
        } else {
            // do not translate
            jexit( 'WARNMOSUSER' );
        }
    }

    // if no timestamp value is passed to function, than current time is
    used
    $date =& JFactory::getDate($timeStamp);

    // updates user lastvistdate field with date and time
    $query = 'UPDATE ' . $this->_tbl
        . ' SET lastvisitDate = ' . $this->_db->Quote($date->toMySQL())
        . ' WHERE id = ' . (int) $id;
    ;
    $this->_db->setQuery( $query );
    if (!$this->_db->query()) {
        $this->setError( $this->_db->getErrorMsg() );
        return false;
    }

    return true;
}
}

```


Source Code for com_mysearch

com_mysearch\controller.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-05 18:14:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This controller will responde accordinly to user actions. The controller will
 * detect which request has been made and respond appropriately by triggering the
 * model to manipulate the data appropriately and passing the model into the view
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Include the path to the TableEvents class in the events.php
JTable::addIncludePath(JPATH_COMPONENT.DS.'tables');

// Import Joomla! 1.5 controller from the framework
jimport('joomla.application.component.controller');

// This class implements the edit(), add() and save() funtions
class MysearchController extends JController
{
    var $_Agegroup = null;
    function edit()
    {
        $eventID = JRequest::getVar('eventID');

        // we set the view environment variable to single
        JRequest::setVar('view', 'autoreg');
        // load the view.html.php for single records
        $this->display();
        // $link = 'index.php?option=com_mysearch&view=autoreg&eventID=' . $eventID;
        // $this->setRedirect($link, $msg, '');
    }

    function saverunner()
    {
        //GET table jos_person_reg_event
        JTable::addIncludePath(JPATH_ADMINISTRATOR.
        DS.'components'.DS.'com_manualreg'.DS.'tables');
        $row =& JTable::getInstance('person_reg_event', 'Table');
        //GET post data
        $eventID = JRequest::getInt('eventID');
        $personID = JRequest::getInt('personID');
        $bib = JRequest::getVar('bib', '', 'post', 'string');

        //Check if bib is already in use
        $sql1 = "SELECT bib from #__person_reg_event WHERE bib='$bib' AND
        eventID='$eventID'";
        $db1 =& JFactory::getDBO();
        $db1->setQuery($sql1);
        $result = $db1->loadResult();

        if($result == $bib)
        {
            $msg = 'bib is already in use. Try Again';
            $link = 'index.php?option=com_manualreg&view=addrunner&eventID=' .
            $eventID;
            $this->setRedirect($link, $msg, 'error');
            return;
        }
        //Check to see if an eventID is present
    }
}
```

```

if(($eventID == 0 || $eventID == ''))
{
    $msg = 'You need to pick an event. Try Again';
    $link = 'index.php?option=com_manualreg&view=selectevent&eventID=' .
        $eventID;
    $this->setRedirect($link, $msg, 'error');
    return;
}
else
{
    if (!$row->bind(JRequest::get('post')))
    {
        JError::raiseError(500, $row->getError());
    }
    $row->eventID = $eventID;
    $row->personID = $personID;
    $sql2 = "SELECT dob from #__persons WHERE personID = '$personID'";
    $sql3 = "SELECT event Date from #__events WHERE eventID = '$eventID'";
    $db2 =& JFactory::getDBO();
    $db2->setQuery($sql2);
    $res1 = $db2->loadResult();
    $bday = (string)$res1;
    $db3 =& JFactory::getDBO();
    $db3->setQuery($sql3);
    $res2 = $db3->loadResult();
    $rday = (string)$res2;
    $age = $this->birthday($bday, $rday);
    $gen = JRequest::getVar('gender', '', 'post', 'string');
    //JError::raiseError(500, $gen);
    $gend = null;
    if($gen == 'male')
    {
        $gend = 'M';
    }
    else if($gen == 'female')
    {
        $gend = 'F';
    }
    $mygroup = $this->getAgegroup($eventID, $age, $gend);
    $row->agegroup = $mygroup;
    $row->bib = $bib;
    $row->raceDayAge = $age;
    $row->has_paid = JRequest::getInt('paid');
    if(!$row->store())
    {
        JError::raiseError(500, $row->getError());
    }
    $link = 'index.php?option=com_manualreg&view=addrunner&eventID=' .
        $eventID;
    $this->setRedirect($link, $msg, '');
    return;
}
}

function &getAgegroup($eventid, $age, $gend)
{
    $sql3 = "SELECT lowAge, upperAge, span FROM #__agegroups
        WHERE ageID='$eventid'";
    $db2 =& JFactory::getDBO();
    $db2->setQuery($sql3);
    $data = $db2->loadRow();
    $low = $data['0'];
    $high = $data['1'];
    $span = $data['2'];

    if ($age < $low)
    {
        $strng1 = "0 - ";
        $strng2 = $low - 1;
        $this->_agegroup = $gend.$strng1.$strng2;
        return $this->_agegroup;
        break;
    }
}

```

```

    }
    if ($age > $high)
    {
        $strng3 = $high;
        $strng4 = " - 100+";
        $this->_agegroup = $gend.$strng3.$strng4;
        return $this->_agegroup;
        break;
    }
    if ($age > $low)
    {
        for($i = $low; $i <= $high; $i += $span -1)
        {
            if($i < $age)
            {
                $i++;
            }
            else if($i == $age)
            {
                $strng5 = $i;
                $strng6 = " - ";
                $strng7 = $i + $span-1;
                $this->_agegroup = $gend.$strng5.$strng6.$strng7;
                return $this->_agegroup;
                break;
            }
            else
            {
                $strng8 = $i-$span;
                $strng9 = " - ";
                $strng10 = $i -1;
                $this->_agegroup = $gend.$strng8.$strng9.$strng10;
                return $this->_agegroup;
                echo '<br />';
                break;
            }
        }
    }
}
}

//calculate Race Day Age based on DOB
function birthday ($bday, $rday)
{
    list($year1,$month1,$day1) = explode("-", $bday);
    list($year2,$month2,$day2) = explode("-", $rday);
    $year_diff = $year2 - $year1;
    $month_diff = $month2 - $month1;
    $day_diff = $day2 - $day1;
    if ($month_diff < 0) $year_diff--;
    elseif (($month_diff==0) && ($day_diff < 0)) $year_diff--;
    return $year_diff;
}

function display()
{
    $view = JRequest::getVar('view');
    if (!$view)
    {
        JRequest::setVar('view', 'mysearch');
    }

    parent::display();
}
}

```

com_mysearch\mysearch.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-12-15 08:36:PST

```

```

* @package Joomla! 1.5
* @subpackage Events
* @Copyright Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
*
* This is the entry point to the extension to create a search for the system. The
* extension will allow administrators to search for people in the database.
*/

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// This is a call to include the controller, and make sure it is only called once
require_once( JPATH_COMPONENT_DS.'controller.php' );

// Create an instance of the controller
$controller = new MysearchController();

// Execute the task variable chosen
$controller->execute( $task );

// Redirect after the task is finish its execution.
$controller->redirect();

```

com_mysearch\models\mysearch.php

```

<?php
/**
 * @Author Steinar Johansen
 * @version Version 1 created 2009-12-14 12:56:PST
 * @package Joomla! 1.5
 * @subpackage Events
 * @Copyright Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport('joomla.application.component.model');

class MysearchModelMysearch extends JModel
{
    var $_search = null;
    var $_query = null;
    var $_data = null;
    var $_pagination = null;
    var $_total = null;

    function &getData()
    {
        $pagination =& $this->getPagination();
        if(empty($this->_data))
        {
            $query = $this->buildSearch();
            $this->_data = $this->getList($query, $pagination->limitstart,
                $pagination->limit);
        }
        return $this->_data;
    }

    function buildSearch()
    {
        if (!$this->_query)
        {
            $search = $this->getSearch();
            $this->_query = "SELECT * FROM #__persons";
            if ($search != '')
            {

```

```

        $fields = array('username','firstName', 'middleName', 'lastName',
            'address','city', 'state', 'zip', 'dob', 'gender',
            'email');
        $where = array();
        $search = $this->_db->getEscaped( $search, true );
        foreach ($fields as $field)
        {
            $where[] = $field . " LIKE '%{$search}%'";
        }
        $this->_query .= ' WHERE ' . implode(' OR ', $where);
    }
}
return $this->_query;
}

function getTotal()
{
    if(!$this->_total)
    {
        $query = $this->buildSearch();
        $this->_total = $this->getListCount($query);
    }
    return $this->_total;
}

function &getPagination()
{
    if(!$this->_pagination)
    {
        jimport('joomla.html.pagination');
        global $mainframe;
        $this->_pagination = new JPagination($this->getTotal(),
            JRequest::getVar('limitstart', 0), JRequest::getVar('limit',
            $mainframe->getConfig('list_limit')));
    }
    return $this->_pagination;
}

function getSearch()
{
    if (!$this->_search)
    {
        global $mainframe, $option;
        $search = $mainframe->getUserStateFromRequest
            ( "$option.search", 'search', '', 'string' );
        $this->_search = JString::strtolower($search);
    }
    return $this->_search;
}
}
}

```

com_mysearch\models\autoreg.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-05 18:10:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport('joomla.application.component.model');

class MysearchModelAutoreg extends JModel
{
    // initiate the data variable

```

```

var $data = null;
var $_pagination = null;
var $_total = null;
var $_agegroup = null;
var $eventID = null;

function buildSearch()
{
    $eventID = JRequest::getInt('races', '', 'post', 'string');
    if($eventID == 0 || $eventID == '')
    {
        $eventID = JRequest::getInt('event_ID');
    }
    return "SELECT * FROM #__persons AS a, #__events as b,
            #__person_reg_event AS c
            WHERE a.personID = c.personID AND b.eventID = c.eventID
            AND c.eventID = '$eventID' ORDER BY bib DESC";
}

function getTotal()
{
    if(!$this->_total)
    {
        $query = $this->buildSearch();
        $this->_total = $this->getListCount($query);
    }
    return $this->_total;
}

function &getPagination()
{
    if(!$this->_pagination)
    {
        jimport('joomla.html.pagination');
        global $mainframe;
        $this->_pagination = new JPagination($this->getTotal(),
            JRequest::getVar('limitstart', 0), JRequest::getVar('limit',
                $mainframe->getConfig('list_limit')));
    }
    return $this->_pagination;
}

function &getData()
{
    $pagination =& $this->getPagination();
    if(empty($this->_data))
    {
        $query = $this->buildSearch();
        $this->_data = $this->getList($query, $pagination->limitstart,
            $pagination->limit);
    }
    return $this->_data;
}

function &getAgegroup()
{
    if (empty( $this->_agegroup ))
    {
        //$eventID = JRequest::getInt('eventID');
        //$myTest = JRequest::getVar('usrName');
        //$myTest = 'jo60nor';
        $sql1 = "SELECT eventID FROM #__person_reg_event WHERE
                eventID='$eventID'";
        $sql2 = "SELECT raceDayAge from #__Persons WHERE userName='$myTest'";

        $db =& $this->getDBO();
        $db->setQuery($sql2);
    }
}

```



```

* @subpackage Events
* @Copyright Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
* This file will define addrunner view for manual registrations application.
*/

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

// The Joomla! 1.5 convention is component name, view and the name of the view
class MysearchViewMysearch extends JView
{
    function display($tpl = null)
    {
        $eventID = JRequest::getInt('eventID');
        $session =& JFactory::getSession();
        $session->set('eventID', $eventID);
        $rows =& $this->get('data');
        $search =& $this->get('search');
        $pagination =& $this->get('pagination');

        $this->assignRef('rows', $rows);
        $this->assignRef('pagination', $pagination);
        $this->assignRef('search', $search);
        $this->assignRef('eventID', $eventID);
        parent::display($tpl);
    }
}

```

com_mysearch\views\mysearch\tmpl\default.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
?>
<!-- This creates the header for the output screen -->
<form action="index.php" method="post" name="adminForm">
<table>
<tr>
<td align="left">
    Search:
    <input type="text" name="search" value=
    "<?php echo $this->search ?>" id="search" />
    <button type="submit">Go</button>
</td>
</tr>
</table>

<table class="adminlist">
<thead>
<tr>
<th width="5">
    <input type="checkbox" name="toggle" value=""
    onclick="checkAll(<?php echo count ($this->rows); ?>);"
    />
</th>
<th width="5%">Username</th>
<th width="7%">First Name</th>
<th width="7%">last name</th>
<th width="15%">Address</th>
<th width="10%">City</th>
<th width="2%">State</th>
<th width="3%">Zip</th>
<th width="5%">Gender</th>
<th width="11%">Email</th>
<th width="20%">Day of Birth</th>
</tr>

```



```

</thead>
<?php
jimport('joomla.filter.output');
// Variabe that creates different CSS. 1 or 0 to distingush rows
$k = 0;

// Loop through the database as long as there are records
for($i=0, $n=count($this->rows); $i < $n; $i++)
{
    // reference to the current row
    $row =&$this->rows[$i];

    // HTML code for a check box that is recognized by the back end
    javascript
    $checked = JHTML::_('grid.id', $i, $row->personID);

    // Publish an icon telling whether the record is published or not
    $published = JHTML::_('grid.published', $row, $i);

    //Pick the row to edit
    $link = JFilterOutput::ampReplace( 'index.php?option=' .
        $option . '&task=edit&cid[]=' . $row->personID);
    ?>

    <tr class="<?php echo "row$k"; ?>">
        <td>
            <?php echo $checked; ?>
        </td>
        <td>
            <?php echo $row->userName; ?>
        </td>
        <td>
            <a href="<?php echo $link; ?>"><?php echo $row->firstName;
                ?></a>
        </td>
        <td>
            <?php echo $row->lastName; ?>
        </td>
        <td>
            <?php echo $row->address; ?>
        </td>
        <td>
            <?php echo $row->city; ?>
        </td>
        <td>
            <?php echo $row->state; ?>
        </td>
        <td>
            <?php echo $row->zip; ?>
        </td>
        <td>
            <?php echo $row->gender; ?>
        </td>
        <td>
            <?php echo $row->email; ?>
        </td>
        <td>
            <?php echo $row->dob; ?>
        </td>
    </tr>
<?php

// Switch k between 1 and 0 for different CSS
$k = 1 - $k;
}
?>

<tfoot>
    <tr>
        <td colspan="14"><?php echo $this->pagination->getListFooter();
            ?></td>
    </tr>

```

```

        </tr>
    </tfoot>
</table>

<!-- Hidden variables for option -->
<input type="hidden" name="option" value="<?php echo $option; ?>" />
<input type="hidden" name="eventID" value="<?php echo $this->eventID; ?>" />
<input type="hidden" name="view" value="mysearch" />
<input type="hidden" name="task" value="" />
<input type="hidden" name="checked" value="0" />
</form>

```

com_mysearch\views\autoreg\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-05 18:38:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This file will define addrunner view for manual registrations application.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

// The Joomla! 1.5 convention is component name, view and the name of the view
class MysearchViewAutoreg extends JView
{
    function display($tpl = null)
    {
        $session =& JFactory::getSession();
        $eventID = $session->get('eventID', '');
        JTable::addIncludePath(JPATH_ADMINISTRATOR.
            DS.'components'.DS.'com_manualreg'.DS.'tables');
        $row1 = & JTable::getInstance('persons', 'Table');
        $cid = JRequest::getVar( 'cid', array(0), '', 'array' );
        // Since we want to edit one record at the time, we select the first ID in
array
        $sid = $cid[0];
        $row1->load($sid);
        $query1 = "SELECT gender, state from #__persons WHERE personID = '$sid'";
        $db1 = & JFactory::getDBO();
        $db1->setQuery($query1);
        $result1 = $db1->loadRow();
        $query2 = "SELECT eventName from #__events WHERE eventID = '$eventID'";
        $db2 = & JFactory::getDBO();
        $db2->setQuery($query2);
        $eventName = $db2->loadResult();
        JToolBarHelper::title( JText::_('Add Participants to ' . $eventName),
            'generic.png' );

        $this->assignRef('eventID', $eventID);

        $state = array(
            JHTML::_('select.option', null, JText::_('Select State')),
            JHTML::_('select.option', 'AL', JText::_('AL')),
            JHTML::_('select.option', 'AK', JText::_('AK')),
            JHTML::_('select.option', 'AR', JText::_('AR')),
            JHTML::_('select.option', 'AZ', JText::_('AZ')),
            JHTML::_('select.option', 'AB', JText::_('AB')),
            JHTML::_('select.option', 'BC', JText::_('BC')),
            JHTML::_('select.option', 'CA', JText::_('CA')),
            JHTML::_('select.option', 'CO', JText::_('CO')),
            JHTML::_('select.option', 'CT', JText::_('CT')),
            JHTML::_('select.option', 'DE', JText::_('DE')),

```

```

JHTML::_('select.option', 'FL', JText::_('FL')),
JHTML::_('select.option', 'GA', JText::_('GA')),
JHTML::_('select.option', 'HI', JText::_('HI')),
JHTML::_('select.option', 'IA', JText::_('IA')),
JHTML::_('select.option', 'ID', JText::_('ID')),
JHTML::_('select.option', 'IL', JText::_('IL')),
JHTML::_('select.option', 'IN', JText::_('IN')),
JHTML::_('select.option', 'KS', JText::_('KS')),
JHTML::_('select.option', 'KY', JText::_('KY')),
JHTML::_('select.option', 'LA', JText::_('LA')),
JHTML::_('select.option', 'MA', JText::_('MA')),
JHTML::_('select.option', 'MB', JText::_('MB')),
JHTML::_('select.option', 'MD', JText::_('MD')),
JHTML::_('select.option', 'ME', JText::_('ME')),
JHTML::_('select.option', 'MI', JText::_('MI')),
JHTML::_('select.option', 'MN', JText::_('MN')),
JHTML::_('select.option', 'MO', JText::_('MO')),
JHTML::_('select.option', 'MS', JText::_('MS')),
JHTML::_('select.option', 'MT', JText::_('MT')),
JHTML::_('select.option', 'NB', JText::_('NB')),
JHTML::_('select.option', 'NC', JText::_('NC')),
JHTML::_('select.option', 'ND', JText::_('ND')),
JHTML::_('select.option', 'NE', JText::_('NE')),
JHTML::_('select.option', 'NH', JText::_('NH')),
JHTML::_('select.option', 'NJ', JText::_('NJ')),
JHTML::_('select.option', 'NL', JText::_('NL')),
JHTML::_('select.option', 'NM', JText::_('NM')),
JHTML::_('select.option', 'NS', JText::_('NS')),
JHTML::_('select.option', 'NT', JText::_('NT')),
JHTML::_('select.option', 'NU', JText::_('NU')),
JHTML::_('select.option', 'NV', JText::_('NV')),
JHTML::_('select.option', 'NY', JText::_('NY')),
JHTML::_('select.option', 'OH', JText::_('OH')),
JHTML::_('select.option', 'OK', JText::_('OK')),
JHTML::_('select.option', 'ON', JText::_('ON')),
JHTML::_('select.option', 'OR', JText::_('OR')),
JHTML::_('select.option', 'PA', JText::_('PA')),
JHTML::_('select.option', 'PE', JText::_('PE')),
JHTML::_('select.option', 'QC', JText::_('QC')),
JHTML::_('select.option', 'RI', JText::_('RI')),
JHTML::_('select.option', 'SC', JText::_('SC')),
JHTML::_('select.option', 'SD', JText::_('SD')),
JHTML::_('select.option', 'SK', JText::_('SK')),
JHTML::_('select.option', 'TN', JText::_('TN')),
JHTML::_('select.option', 'TX', JText::_('TX')),
JHTML::_('select.option', 'UT', JText::_('UT')),
JHTML::_('select.option', 'VA', JText::_('VA')),
JHTML::_('select.option', 'VT', JText::_('VT')),
JHTML::_('select.option', 'WA', JText::_('WA')),
JHTML::_('select.option', 'WI', JText::_('WI')),
JHTML::_('select.option', 'WV', JText::_('WV')),
JHTML::_('select.option', 'WY', JText::_('WY')),
JHTML::_('select.option', 'YT', JText::_('YT')),
JHTML::_('select.option', 'XX', JText::_('No State'))
);
$gender[] = JHTML::_('select.option', 'male', 'Male:');
$gender[] = JHTML::_('select.option', 'female', 'Female:');
$lists['gender'] = JHTML::_('select.radiolist', $gender, 'gender',
    'class="inputbox" ', 'value', 'text', $result1[0], 'gender');
$this->assignRef('lists', $lists);
$this->assignRef('state', JHTML::_('select.genericList',
    $state, 'state', 'class="inputbox" ', 'value',
    'text', $result1[1]));
$paid = null;
$pagination = & $this->get('pagination');

// Assign the data to the rows

$this->assignRef('row1', $row1);
$this->assignRef('eventName', $eventName);
$this->assignRef('pagination', $pagination);

```

```

        $this->assignRef('paid', JHTML::_('select.booleanlist',
            'paid', 'class="inputbox"', $paid));
        parent::display($tpl);
    }
}

```

com_mysearch\views\autoreg\tmpl\default.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

JHTML::_('behavior.calendar');
$event_ID = (int) JRequest::getVar('event_ID', 0);
$eventID = $this->eventID;
$personID = $this->row1->personID;
?>
<form action="index.php?option=com_mysearch&task=saverunner"
    method="post" id="josForm" name="josForm" style="font-size: 8pt;">
    <table cellpadding="2">
        <tr>
            <td>
                Event:
            </td>
            <td>
                <?php echo $this->eventName; ?>
            </td>
        </tr>
        <tr>
            <td>
                PersonID:
            </td>
            <td>
                <?php echo $this->row1->personID; ?>
            </td>
        </tr>
        <tr>
            <td width="80" align="left" class="key">
                First Name:
            </td>
            <td>
                <input class="text_area" type="text" name="firstName"
                    id="firstName" size="20" maxlength="50"
                    value="<?php echo $this->row1->firstName; ?>" />
            </td>
            <td width="80" align="right" class="key">
                Middle Name:
            </td>
            <td>
                <input class="text_area" type="text" name="middleName"
                    id="middleName" size="20" maxlength="50"
                    value="<?php echo $this->row1->middleName; ?>" />
            </td>
            <td width="80" align="right" class="key">
                Last Name:
            </td>
            <td>
                <input class="text_area" type="text" name="lastName"
                    id="lastName" size="20" maxlength="50"
                    value="<?php echo $this->row1->lastName; ?>" />
            </td>
        </tr>
        <tr>
            <td width="80" align="left" class="key">
                Address:
            </td>
            <td colspan="5" align="left" class="key">

```

```

        <input class="text_area" type="text" name="address"
            id="address" size="116" maxlength="250"
            value="<?php echo $this->row1->address; ?>" />
    </td>
</tr>
<tr>
    <td width="80" align="left" class="key">
        City:
    </td>
    <td>
        <input class="text_area" type="text" name="city"
            id="city" size="20" maxlength="50"
            value="<?php echo $this->row1->city; ?>" />
    </td>
    <td width="80" align="right" class="key">
        State:
    </td>
    <td>
        <?php echo $this->state; ?>
    </td>
    <td width="80" align="right" class="key">
        Zip:
    </td>
    <td>
        <input class="text_area" type="text" name="zip"
            id="zip" size="20" maxlength="50"
            value="<?php echo $this->row1->zip; ?>" />
    </td>
</tr>
<tr>
    <td>
        Email Address:
    </td>
    <td colspan="3">
        <input class="text_area" type="text" name="email"
            id="email" size="50" maxlength="50"
            value="<?php echo $this->row1->email; ?>" />
    </td>
</tr>
<tr>
    <td width="80" align="left" class="key">
        Date of Birth:<br>
        YYYY-MM-DD
    </td>
    <td colspan="2">
        <input type="text" name="dob" id="dob" value="<?php echo
            $this->row1->dob; ?>" />
        
    </td>
</tr>
<tr>
    <td>
        Gender:
    </td>
    <td>
        <?php echo $this->lists['gender']; ?>
    </td>
</tr>
<tr>
    <td>
        BiB:
    </td>
    <td>
        <input class="text_area" type="text" name="bib"
            id="bib" size="5" maxlength="5"
            value="" />
    </td>
</tr>
<tr>
    <td>

```

```

        Has Paid:
    </td>
    <td>
        <?php echo $this->paid; ?>
    </td>
</tr>
</table>

<button class="button validate" type="submit"><?php echo JText::_('Add Partisipant');
?></button>

<input type="hidden" name="option" value="com_mysearch" />
<input type="hidden" name="task" value="saverunner" />
<input type="hidden" name="eventID" value="<?php echo $eventID; ?>" />
<input type="hidden" name="event_ID" value="<?php echo $event_ID; ?>" />
<input type="hidden" name="personID" value="<?php echo $personID; ?>" />
<?php echo JHTML::_('form.token' ); ?>
</form>

```

Source Code for com_payments

com_payments\controller.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-09-01 09:39:FST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This controller will responde accordinly to user actions. The controller will
 * detect which request has been made and respond appropriately by triggering the
 * model to manipulate the data appropriately and passing the model into the view
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Include the path to the TableEvents class in the events.php
JTable::addIncludePath(JPATH_COMPONENT_DS.'tables');

// Import Joomla! 1.5 controller from the framework
jimport('joomla.application.component.controller');

// This class implements the edit(), add() and save() funtions
class PaymentsController extends JController
{
    // Activates the edit button on the default view
    function edit()
    {
        // we set the view environment variable to single
        JRequest::setVar('view', 'edit');
        // load the view.html.php for single records
        $this->display();
    }

    // Activates the new button on the default view
    function add()
    {
        // we set the view environment variable to single
        JRequest::setVar('view', 'define');
        // load the view.html.php for single records
        $this->display();
    }

    function select()
    {
        $eventID = JRequest::getInt('eventID', '', 'post', 'string');
    }
}

```

```

if ($eventID != '' || $eventID != 0)
{
    JRequest::setVar('view', 'define');
    $this->display();
}
else
{
    JRequest::setVar('view', 'selectevent');
    $this->display();
}
}

function define()
{
    $row =& JTable::getInstance('fees', 'Table');
    $row2 = & JTable::getInstance('define_fees', 'Table');
    $eventID = JRequest::getInt('eventID');
    //$raceDate = JRequest::getString('racedate', '', 'post', 'date');
    //$raceDatePrice = JRequest::getInt('racedateprice', '', 'post', 'string');

    $regDate = JRequest::getString('regdate', '', 'post', 'date');
    $normPrice = JRequest::getInt('normalprice', '', 'post', 'string');

    $earlyBird = JRequest::getString('earlybird', '', 'post', 'date');
    $earlyPrice = JRequest::getInt('earlyprice', '', 'post', 'string');

    $vyEarlyBird = JRequest::getString('vyearlybird', '', 'post', 'date');
    $vyEarlyPrice = JRequest::getInt('vyearlyprice', '', 'post', 'string');

    $sql = "SELECT eventName, event_Date from #__events WHERE eventID =
    '$eventID'";
    $db = JFactory::getDBO();
    $db->setQuery($sql);
    $result = $db->loadRow();
    $eventname = $result[0];
    $eventdate = $result[1];
    $row->feeID = JRequest::getInt('feeID', 0);
    $row->eventID = $eventID;
    $row->eventName = $eventname;
    $row->duedate = $eventdate;
    $row->fee = JRequest::getString('racedateprice', '', 'post', 'string');
    $row->feeName = 'Race Day Price';
    if(!$row->store())
    {
        JError::raiseError(500, $row->getError());
    }
    if ($normPrice != '')
    {
        $row->feeID = JRequest::getInt('feeID', 0);
        $row->eventID = $eventID;
        $row->eventName = $eventname;
        $row->duedate = $regDate;
        $row->fee = $normPrice;
        $row->feeName = 'Normal price';
        if(!$row->store())
        {
            JError::raiseError(500, $row->getError());
        }
    }
    if($earlyPrice != '')
    {
        $row->feeID = JRequest::getInt('feeID', 0);
        $row->eventID = $eventID;
        $row->eventName = $eventname;
        $row->duedate = $earlyBird;
        $row->fee = $earlyPrice;
        $row->feeName = 'Early Bird';
        if(!$row->store())
        {
            JError::raiseError(500, $row->getError());
        }
    }
}

```

```

        if ($vyEarlyPrice != '')
        {
            $row->feeID = JRequest::getInt('feeID', 0);
            $row->eventID = $eventID;
            $row->eventName = $eventname;
            $row->duedate = $vyEarlyBird;
            $row->fee = $vyEarlyPrice;
            $row->feeName = 'Very Early Bird';
            if(!$row->store())
            {
                JError::raiseError(500, $row->getError());
            }
        }
    }
}

$row2->id = JRequest::getInt('id', 0);
$row2->isTInvolved = JRequest::getInt('tshirt', 'integer');
$row2->isApperalForsale = JRequest::getInt('apperal', 'integer');
$row2->tPrice = Jrequest::getInt('tshirtPrice', 'integer');
$row2->eventID = $eventID;

if(!$row2->store())
{
    JError::raiseError(500, $row2->getError());
}

$link = 'index.php?option=com_payments';
$this->setRedirect($link, $msg, 'error');
}

function save()
{
    // Include global member functions like redirect for instance
    global $option;
    //JError::raiseError(500, 'this is inside the save function');

    // Set Row as an instance of the table agegroups Class
    $row=& JTable::getInstance('fees', 'Table');

    /*Bind() is a function of $row and loads all the variables from the form
    If unsuccessful, an error is casted. Bind() matches the variables to
    to the object. JRequest::get reduces the ability of SQL Attack */
    if (!$row->bind(JRequest::get('post')))
    {
        JError::raiseError(500, $row->getError());
    }

    // INSERT or UPDATE depending whether an ID is present
    if(!$row->store())
    {
        JError::raiseError(500, $row->getError());
    }

    $this->setRedirect('index.php?option=' . $option, 'Data Updated');
}

function display()
{
    $view = JRequest::getVar('view');
    if (!$view)
    {
        JRequest::setVar('view', 'selectevent');
    }
    parent::display();
}
}

```



```
}
```

com_payments\payments.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-09-01 09:38:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This is the entry point to the payment system.
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// This is a call to include the controller, and make sure it is only called once
require_once( JPATH_COMPONENT_DS.'controller.php' );

// This code creates the toolbar for the different windows we are using. The
// JToolBarHelper contains functions that produces all the HTML code necessary

// Create an instance of the controller
$controller = new PaymentsController();

// Execute the task variable chosen
$controller->execute( $task );

// Redirect after the task is finish its execution.
$controller->redirect();
```

com_payments\models\all.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-09-13 07:14:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport('joomla.application.component.model');

class PaymentsModelAll extends JModel
{
    // initiate the data variable
    var $fees = null;

    function getfees()
    {
        // Check if the data variable is empty
        if(empty($this->fees))
        {
            // Create a query to load all the records from jos_events
            $query = "SELECT * FROM #__fees ORDER BY eventID";

            // get a list of data based on the query
            $this->fees = $this->getList($query);
        }
    }
}
```

```

    // Return the data
    return $this->fees;
}
}

```

com_payments\models\define.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-09-02 10:08:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model used to prepare for payment set up
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ('Restricted Access');

// Import the joomla! 1.5 framework model functions
jimport('joomla.application.component.model');

class PaymentsModelDefine extends JModel
{
    var $event = null;

    function getEvent()
    {
        if (empty($this->event))
        {
            $sql1 = "SELECT eventID AS value, EventName as text FROM #__events";
            $this->event = $this->getList($sql1);
        }
        return $this->event;
    }
}
}

```

com_payments\models\selectevent.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-07 07:10:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ('Restricted Access');

// Import the joomla! 1.5 framework model functions
jimport('joomla.application.component.model');

class PaymentsModelSelectevent extends JModel
{
    // initiate the data variable
    var $event = null;

    function getEvent()
    {
        if (empty($this->event))
        {
            $sql = "SELECT eventID AS value, eventName as text FROM #__events";

```

```

        $this->event = $this->_getList($sql);
    }
    return $this->event;
}
}

```

com_payments\views\all\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-09-01 09:51:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This view will process data that will be shown to the user
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

class PaymentsViewAll extends JView
{
    function display($tpl = null)
    {
        //Create Toolbar
        JToolBarHelper::title( JText::_ ( 'List Dues and Payments' ), 'generic.png' );
        JToolBarHelper::publishList();
        JToolBarHelper::unpublishList();
        JToolBarHelper::editList();
        JToolBarHelper::deleteList();
        JToolBarHelper::addNew();

        // Calls the getEvent
        $rows =& $this->get('fees');

        // Assign the data to the rows
        $this->assignRef('rows', $rows);

        parent::display($tpl);
    }
}

```

com_payments\views\all\tmpl\default.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-09-01 09:52:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This is the actual view that is being presented to the user
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

?>

<form action="index.php" method="post" name="adminForm">
    <table class="adminlist">
        <thead>

```

```

        <tr>
            <th width="3%">
                <input type="checkbox" name="toggle" value=""
                    onclick="checkAll(<?php echo count ($this->rows); ?>);"
                />
            </th>
            <th width="15%">Event Name</th>
            <th width="7%">EventID</th>
            <th width="10%">Due Date</th>
            <th width="12%">Fee</th>
            <th width="10%">Fee Name</th>
        </tr>
    </thead>
    <?php
    jimport('joomla.filter.output');
    // Variabe that creates different CSS. 1 or 0 to distingush rows
    $k = 0;

    // Loop through the database as long as there are records
    for($i=0, $n=count($this->rows); $i < $n; $i++)
    {
        // reference to the current row
        $row =&$this->rows[$i];
        //$row2 =&$this->rows1[$i];

        // HTML code for a check box that is recognized by the back end
    //
    javascript
    $checked = JHTML::_('grid.id', $i, $row->feeID);

    //Pick the row to edit
    $link = JFilterOutput::ampReplace( 'index.php?option=' .
        $option . '&task=edit&cid[]=' . $row->feeID);
    ?>
    <tr class="<?php echo "row$k"; ?>">
        <td align="center">
            <?php echo $checked; ?>
        </td>
        <td>
            <a href="<?php echo $link; ?>"> <?php echo $row->eventName; ?></a>
        </td>
        <td align="center">
            <?php echo $row->eventID; ?>
        </td>
        <td>
            <?php echo $row->duedate; ?>
        </td>
        <td>
            <?php echo $row->fee; ?>
        </td>
        <td>
            <?php echo $row->feeName; ?>
        </td>
    </tr>
    <?php

    // Switch k between 1 and 0 for different CSS
    $k = 1 - $k;
    }
    ?>
</table>

<!-- Hidden variables for option -->
<input type="hidden" name="option" value="<?php echo $option; ?>" />
<input type="hidden" name="task" value="" />
<input type="hidden" name="boxchecked" value="0" />
</form>

```

com_payments\views\define\view.html.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-09-01 09:43:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This view will process data that will be shown to the user
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

class PaymentsViewDefine extends JView
{
    function display($tpl = null)
    {
        $eventID = JRequest::getInt( 'aces', '', 'post', 'string' );
        $sql1 = "SELECT event_Date, eventName FROM #__events WHERE eventID =
        '$eventID' ";
        $db =& JFactory::getDBO();
        $db->setQuery( $sql1 );
        $result = $db->loadRow();
        $eventDate = $result[0];
        $eventName = $result[1];
        $date = JHTML::date( $eventDate );
        JToolBarHelper::title( JText::_ ( 'Define Fees for ' . $eventName . ' ' .
        $date ), 'generic.png' );

        $yearlybird = null;
        $earlybird = null;
        $regdate = null;
        $racedate = null;
        $tshirt = null;
        $apperal = null;
        $this->assignRef( 'regdate', JHTML::_ ( 'calendar',
        $regdate, 'regdate', 'regdate' ) );
        $this->assignRef( 'earlybird', JHTML::_ ( 'calendar',
        $earlybird, 'earlybird', 'earlybird' ) );
        $this->assignRef( 'vyearlybird', JHTML::_ ( 'calendar',
        $vyearlybird, 'vyearlybird', 'vyearlybird' ) );
        $this->assignRef( 'tshirt', JHTML::_ ( 'select.booleanlist',
        'tshirt', 'class="inputbox"', $tshirt ) );
        $this->assignRef( 'apperal', JHTML::_ ( 'select.booleanlist',
        'apperal', 'class="inputbox"', $apperal ) );
        $this->assignRef( 'date', $date );
        $this->assignRef( 'eventID', $eventID );

        parent::display( $tpl );
    }
}
}
```

com_payments\views\define\tmpl\default.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-09-01 09:43:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This is the actual view that is being presented to the user
 */
```

```

// This make sure that the code can only be invoked within Joomla! 1.5
defined( 'JEXEC' ) or die ( 'Restricted Access' );
//JHTML::_('behavior.calendar');
$eventID = JRequest::getInt('races', '', 'post', 'integer');

?>

<form action="index.php?option=com_payments&task=define"
method="post" name="adminForm" id="adminForm">
<fieldset class="adminform">

<!-- Creates the line around the form that is being shown -->
<legend>Define Payment Rules</legend>

<!-- admintable is a preset class for Joomla! 1.5 to format the table -->
<table class="admintable">

<tr>
<td width="80" align="right" class="key">
Race Day date:
</td>
<td>
<?php echo $this->date ?>
</td>
<td width="100" align="right" class="key">
Race Day price:
</td>
<td>
<strong>${</strong>
<input class="text_area" type="text" name="racedateprice"
id="racedateprice" size="5" maxlength="5"
/>

</td>
</tr>
<tr>
<td width="80" align="right" class="key">
Regular date:
</td>
<td >
<?php echo $this->regdate ?>
</td>
<td width="100" align="right" class="key">
Regular Price:
</td>
<td>
<strong>${</strong>
<input class="text_area" type="text" name="normalprice"
id="normalprice" size="5" maxlength="5"
value="" />

</td>
</tr>
<tr>
<td width="80" align="right" class="key">
Early Bird date :
</td>
<td>
<?php echo $this->earlybird ?>
</td>
<td width="100" align="right" class="key">
Early Bird Price :
</td>
<td>
<strong>${</strong>
<input class="text_area" type="text" name="earlyprice"
id="earlyprice" size="5" maxlength="5"
value="" />

</td>
</tr>
<tr>
<td width="80" align="right" class="key">

```

```

        Very Early Bird date:
    </td>
    <td>
        <?php echo $this->vyearlybird ?>
    </td>
    <td width="100" align="right" class="key">
        Very Early Bird price:
    </td>
    <td>
        <strong>${</strong>
        <input class="text_area" type="text" name="vyearlyprice"
            id="vyearlyprice" size="5" maxlength="5"
            value="" />
    </td>
</tr>
<tr>
    <td width="100" align="right" class="key">
        Include T-Shirt:
    </td>
    <td>
        <?php echo $this->tshirt; ?>
    </td>
    <td width="100" align="right" class="key">
        T-Shirt Price:
    </td>
    <td>
        <strong>${</strong>
        <input class="text_area" type="text" name="tshirtPrice"
            id="tshirtPrice" size="5" maxlength="5"
            value="" />
    </td>
</tr>
<tr>
    <td width="100" align="right" class="key">
        Apperal Sales:
    </td>
    <td>
        <?php echo $this->apperal; ?>
    </td>
</tr>
<tr>
    <td colspan="4" align="center">
        <button class="button validate" type="submit"><?php echo
        JText::_('Submit'); ?></button>
        <input type="hidden" name="option" value="com_payments" />
        <input type="hidden" name="task" value="define" />
        <input type="hidden" name="eventID" value="<?php echo $eventID;
        ?>" />
        <button type="reset"><?php echo JText::_('Reset'); ?></button>
    </td>
</tr>
</table>
</fieldset>
</form>

```

com_payments\views\edit\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-09-13 09:37:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This file will define the toolbar for the administrator needed to easily manage
 * payment records for the system.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

```

```

// bring in the functions related to JView
jimport( 'joomla.application.component.view');

// The Joomla! 1.5 convention is component name, view and the name of the view
class PaymentsViewEdit extends JView
{
    function display($tpl = null)
    {
        //Create Toolbar
        JToolBarHelper::title( JText::_('Edit Dues and Payments'), 'generic.png');
        JToolBarHelper::save();
        JToolBarHelper::apply();
        JToolBarHelper::cancel();
        // Create an instance of the TableFlyer object
        $row =& JTable::getInstance('fees', 'Table');
        $cid = JRequest::getVar( 'cid', array(0), '', 'array' );
        // Since we want to edit one record at the time, we select the first ID in
// array
        $id = $cid[0];
        $row->load($id);
        $this->assignRef('row', $row);
        $this->assignRef('duedate', JHTML::_('calendar',
            $row->duedate, 'duedate', 'duedate'));
        parent::display($tpl);
    }
}

```

com_payments\views\edit\tmpl\default.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-09-13 09:40:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This is the actual view that is being presented to the user
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );
JHTML::_('behavior.calendar');
?>

<form action="index.php?option=com_payments" method="post" name="adminForm" id="adminForm">
  <fieldset class="adminform">

    <!-- Creates the line around the form that is being shown -->
    <legend>Define Payment Rules</legend>

    <!-- admintable is a preset class for Joomla! 1.5 to format the table -->
    <table class="admintable">
      <tr>
        <td width="80" align="right" class="key">
          Event Name:
        </td>
        <td>
          <?php echo $this->row->eventName; ?>
        </td>
        <td width="80" align="right" class="key">
          Event ID:
        </td>
        <td>
          <?php echo $this->row->eventID; ?>
        </td>
      </tr>
      <tr>
        <td width="80" align="right" class="key">
          Fee Name:

```



```

        </td>
        <td>
            <strong> <?php echo $this->row->feeName; ?> </strong>
        </td>
    </tr>
    <tr>
        <td width="80" align="right" class="key">
            Edit Date:
        </td>
        <td>
            <?php echo $this->duedate; ?>
        </td>
        <td width="100" align="right" class="key">
            Edit Price:
        </td>
        <td>
            <strong> $ </strong>
            <input class="text_area" type="text" name="fee"
                id="fee" size="5" maxlength="5"
                value="<?php echo $this->row->fee; ?>" />
        </td>
    </tr>
</table>
<input type="hidden" name="feeID" value="<?php echo $this->row->feeID; ?>" />
<input type="hidden" name="option" value="<?php echo $option; ?>" />
<input type="hidden" name="task" value="" />
<?php echo JHTML::_ ( 'form.token' ); ?>
</fieldset>
</form>

```

com_payments\views\selectevent\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-07 07:40:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This file will define view for manual registrations application.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

// The Joomla! 1.5 convention is component name, view and the name of the view
class PaymentsViewSelectevent extends JView
{
    function display($tpl = null)
    {
        JToolBarHelper::title( JText::_ ( 'Select Event' ), 'generic.png' );
        $events =& $this->get( 'event' );
        $races[] = JHTML::_ ( 'select.option', '0',
            JText::_ ( 'Select Event' ), 'value', 'text' );
        $races = array_merge( $races, $events );
        $lists['races'] = JHTML::_ ( 'select.genericlist',
            $races, 'races', 'class="inputbox" size="1" ', 'value', 'text' );
        $this->assignRef( 'lists', $lists );
        parent::display( $tpl );
    }
}

```

com_payments\views\selectevent\tmpl\default.php

```
<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
?>

<form action="index.php?option=com_payments&task=select"
      method="post" id="josForm" name="josForm" style="font-size: 8pt;">

    <?php echo $this->lists['races']; ?>

    <button class="button validate" type="submit"><?php echo JText::_('Submit'); ?></button>
    <input type="hidden" name="option" value="com_payments" />
    <input type="hidden" name="task" value="select" />
    <?php echo JHTML::_('form.token'); ?>

</form>
<br />
<table border="2">
    <tr>
        <td>
            <?php
                $link = JRoute::_("index.php?option=com_payments&view=all");
                echo '<a href="' . $link . '">' . '<strong>' . 'Click here to edit
                records' . '</strong>' .
                '</a>' . '<br />';
            ?>
        </td>
    </tr>
</table>
```

Source Code for com_timing

com_timing\controller.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-24 16:31:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This controller will responde accordinly to user actions. The controller will
 * detect which request has been made and respond appropriately by triggering the
 * model to manipulate the data appropriately and passing the model into the view
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Include the path to the TableEvents class in the events.php
JTable::addIncludePath(JPATH_COMPONENT_DS.'tables');

// Import Joomla! 1.5 controller from the framework
jimport('joomla.application.component.controller');

// This class implements the edit(), add() and save() funtions
class TimingController extends JController
{
    //This function handles the request data from select event form
    //It check's to see if an event was chosen before timing data cn be
    //entered
    function select()
    {
        $seventID = JRequest::getInt('races', '', 'post', 'string');
```

```

if ($eventID != '' || $eventID != 0)
{
    JRequest::setVar('view', 'timing');
    $this->display();
}
else
{
    //$session =& JFactory::getSession();
    //$session->set('eventID', $eventID);
    JRequest::setVar('view', 'selevent');
    $this->display();
    //$link = 'index.php?option=com_timing&view=timing&eventID=' . $eventID;
    //$this->setRedirect($link, $msg, '');
    return;
}
}

//This function handles the request from the timing form
function save()
{
    //Get post data from the timing form
    $eventID = JRequest::getInt('eventID');
    $event_ID = JRequest::getInt('event_ID');
    $bib = JRequest::getInt('bib', '', 'post', 'integer');
    $hours = JRequest::getInt('hours', '', 'post', 'integer');
    $minutes = JRequest::getInt('minutes', '', 'post', 'integer');
    $secs = JRequest::getInt('secs', '', 'post', 'integer');
    $bib1 = JRequest::getInt('bib1', '', 'post', 'integer');
    $hours1 = JRequest::getInt('hours1', '', 'post', 'integer');
    $minutes1 = JRequest::getInt('minutes1', '', 'post', 'integer');
    $secs1 = JRequest::getInt('secs1', '', 'post', 'integer');

    //Error Checking for eventID
    if($eventID == 0)
    {
        $msg = 'Select an Event';
        $link = 'index.php?option=com_timing&view=selevent';
        $this->setRedirect($link, $msg, '');
        return;
    }
    if($bib != 0)
    {
        if($hours == 0)
        {
            if($minutes == 0)
            {
                if($secs == 0)
                {
                    if($bib1 == 0)
                    {
                        $msg = 'Hours, Minutes and Seconds cannot be zero';
                        $link =
                            'index.php?option=com_timing&view=timing&eventID=' .
                                $eventID;
                        $this->setRedirect($link, $msg, '');
                        return;
                    }
                }
            }
        }
    }
    //Check to see if the bib is valid for a particular race
    $bibCheck1 = $this->checkBib($bib, $eventID);
    //JError::raiseWarning(500, JText::_('bibCheck = ' . $bibCheck1));
    if($bibCheck1 == -1)
    {
        $msg = 'Bib is not valid for this race';
        $link = 'index.php?option=com_timing&view=timing&eventID=' .
            $eventID;
        $this->setRedirect($link, $msg, '');
        return;
    }
}

```

```

    }

    //Check to see if minutes entered is lower than 60
    $minutesCheck = $this->checkMinutes($minutes);
    if($minutesCheck == -1)
    {
        $msg = 'Minutes cannot exceed 60';
        $link = 'index.php?option=com_timing&view=timing&eventID=' .
            $eventID;
        $this->setRedirect($link, $msg, '');
        return;
    }

    //Check to see if Seconds entered is lower than 60
    $secondCheck = $this->checkseconds($secs);
    if($secondCheck == -1)
    {
        $msg = 'Seconds cannot exceed 59';
        $link = 'index.php?option=com_timing&view=timing&eventID=' .
            $eventID;
        $this->setRedirect($link, $msg, '');
        return;
    }

    $checkBibTime = $this->bibTime($bib, $eventID);
    if($checkBibTime == -1)
    {
        $msg = 'Bib already has a time!';
        $link = 'index.php?option=com_timing&view=timing&eventID=' .
            $eventID;
        $this->setRedirect($link, $msg, '');
        return;
    }
    $this->addTime($eventID, $bib, $hours, $minutes, $secs);
    return;
}

if($bib1 != 0)
{
    $this->addTime($eventID, $bib1, $hours1, $minutes1, $secs1);
    return;
}
else
{
    $msg = 'You Need to enter a Bib and Time OR repair Time for a bib';
    $link = 'index.php?option=com_timing&view=timing&eventID=' . $eventID;
    $this->setRedirect($link, $msg, '');
    return;
}
}

function display()
{
    $view = JRequest::getVar('view');
    if (!$view)
    {
        JRequest::setVar('view', 'selevent');
    }
    parent::display();
}

function addTime($eventID, $bib, $hours, $minutes, $secs)
{
    $time = $hours . ':' . $minutes . ':' . $secs;
    $sql2 = "SELECT distanceID from #__event_has_many WHERE eventID=
    '$eventID'";
    $db2 =& JFactory::getDBO();
    $db2->setQuery($sql2);
    $distanceID = $db2->loadResult();
    $dist = $this->distance($distanceID);
}

```

```

$space = $this->calcpace($dist, $hours, $minutes, $secs);
//Database request to retrieve id from jos_person_reg_event table
//Based on bib and eventID
$sql = "SELECT id from #__person_reg_event WHERE eventID = '$eventID'
      AND bib = '$bib'";
$db = & JFactory::getDBO();
$db->setQuery($sql);
$result = $db->loadResult();

//This is the database request to update records in the
//jos_person_reg_event table
$db1 = JFactory::getDBO();
$update = new stdClass;
$update->id = $result;
$update->clock_time = $time;
$update->pace = $space;
$db1->updateObject('#__person_reg_event', $update, 'id' );
if(!$db1)
{
    JError::raiseError(500, 'No update happened');
}

//redirect back to the timing for for new record to be entered
$link = 'index.php?option=com_timing&view=timing&eventID=' . $eventID;
$this->setRedirect($link, '', '');
return;
}

//This function is checking if the bib is valid for the race
function checkBib($bib, $eventID)
{
    $sql1 = "SELECT id from #__person_reg_event WHERE eventID = '$eventID'
          AND bib = '$bib'";
    $db1 = & JFactory::getDBO();
    $db1->setQuery($sql1);
    $result1 = $db1->loadResult();
    if(!$result1)
    {
        return -1;
    }
    else
    {
        return 1;
    }
}

//This function check if a bib has a time already
function bibTime($bib, $eventID)
{
    $sql2 = "SELECT clock_time FROM #__person_reg_event
          WHERE eventID = '$eventID' AND bib= '$bib'
          AND clock_time > 0";
    $db2 = & JFactory::getDBO();
    $db2->setQuery($sql2);
    $result2 = $db2->loadResult();
    if($result2)
    {
        return -1;
    }
    else
    {
        return 1;
    }
}

//This function check if minutes entered is less than 60
function checkMinutes($minutes)
{
    if($minutes > 59)
    {
        return -1;
    }
}

```

```

        else
        {
            return 1;
        }
    }

//This function check to see if seconds entered is less than 60
function checkseconds($secs)
{
    If($secs > 59)
    {
        return -1;
    }
    else
    {
        return 1;
    }
}

function calcpace($dist, $hours, $minutes, $secs)
{
    $mile = 1609.344;
    $s1 = $hours * 3600;
    $s2 = $minutes * 60;
    $s3 = $secs;
    $total1 = $s1 + $s2 + $s3;

    $total2 = $total1/$dist;

    $total3 = (int)($total2 * $mile);

    $total4 = $total3%60;

    $total5 = ($total3 - $total4)/60;

    $space = $total5 . ':' . $total4;
    return $space;
}

function distance($distanceID)
{
    $length = null;
    switch($distanceID)
    {
        case 1:
            $length = 1609.344;
            break;
        case 3:
            $length = 5000;
            break;
        case 4:
            $length = 10000;
            break;
        case 5:
            $length = 15000;
            break;
        case 6:
            $length = 21097.5;
            break;
        case 7:
            $length = 42195;
            break;
    }
    return $length;
}
}

```

com_timing\timing.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-24 16:29:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This is the entry point to the extension to do the timing for the system. The
 * extension will allow administrators to do timing for their events.
 */

// This make sure that the code can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// This is a call to include the controller, and make sure it is only called once
require_once( JPATH_COMPONENT_DS.'controller.php' );

// This code creates the toolbar for the different windows we are using. The
// JToolBarHelper contains functions that produces all the HTML code necessary

// Create an instance of the controller
$controller = new TimingController();

// Execute the task variable chosen
$controller->execute( $task );

// Redirect after the task is finish its execution.
$controller->redirect();
```

com_timing\models\selectevent.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-07 07:10:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport('joomla.application.component.model');

class TimingModelSelevent extends JModel
{
    // initiate the data variable
    var $event = null;

    function getEvent()
    {
        if(empty($this->event))
        {
            $sql = "SELECT eventID AS value, eventName as text FROM #__events";
            $this->event = $this->_getList($sql);
        }
        return $this->event;
    }
}
```

com_timing\models\timing.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-05 18:10:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport( 'joomla.application.component.model' );

class TimingModelTiming extends JModel
{
    // initiate the data variable

    var $data = null;
    var $_pagination = null;
    var $_total = null;
    var $_agegroup = null;
    var $eventID = null;
    var $_allpl = null;
    var $_sexpl = null;
    var $_agepl = null;

    function buildSearch()
    {
        $eventID = JRequest::getInt( 'aces', '', 'post', 'string' );
        if( $eventID == 0 || $eventID == '' )
        {
            $eventID = JRequest::getInt( 'eventID' );
        }
        return "SELECT * FROM #__persons AS a, #__events as b,
                #__person_reg_event AS c
                WHERE a.personID = c.personID AND b.eventID = c.eventID
                AND c.eventID = '$eventID' AND c.clock_time > 0
                ORDER BY clock_time DESC";
    }

    function getTotal()
    {
        if( !$this->_total )
        {
            $query = $this->buildSearch();
            $this->_total = $this->getListCount( $query );
        }
        return $this->_total;
    }

    function &getPagination()
    {
        if( !$this->_pagination )
        {
            jimport( 'joomla.html.pagination' );
            global $mainframe;
            $this->_pagination = new JPagination( $this->getTotal(),
                JRequest::getVar( 'limitstart', 0 ), JRequest::getVar( 'limit',
                    $mainframe->getConfig( 'list_limit' ) );
        }
        return $this->_pagination;
    }

    function &getData()
    {
        $pagination =& $this->getPagination();
        if( empty( $this->_data ) )
    }
}
```



```

    {
        $query = $this->buildSearch();
        $this->_data = $this->getList($query, $pagination->limitstart,
            $pagination->limit);
    }
    return $this->_data;
}

function &getAgegroup()
{
    if (empty( $this->_agegroup ))
    {
        //$eventID = JRequest::getInt('eventID');
        //$myTest = JRequest::getVar('usrName');
        //$myTest = 'jo60nor';
        $sql1 = "SELECT eventID FROM #__person_reg_event WHERE
            eventID='$eventID'";
        $sql2 = "SELECT raceDayAge from #__Persons WHERE userName='$myTest'";

        $db =& $this->getDBO();
        $db->setQuery($sql2);
        $age = $db->loadResult();
        $db1 =& $this->getDBO();
        $db1->setQuery($sql1);
        $ageEvent = $db1->loadResult();
        $db2 =& $this->getDBO();
        $sql3 = "SELECT lowAge, upperAge, span FROM #__agegroups
            WHERE ageID='$ageEvent'";
        $db2->setQuery($sql3);
        $data = $db2->loadRow();
        $low = $data['0'];
        $high = $data['1'];
        $span = $data['2'];
        //$this->_agegroup = 44;
        //return $this->_agegroup;
        if ($age < $low)
        {
            $strng1 = "0 - ";
            $strng2 = $low -1;
            $this->_agegroup = $strng1.$strng2;
            return $this->_agegroup;
            break;
        }
        if ($age > $high)
        {
            $strng3 = $high;
            $strng4 = " - 100+";
            $this->_agegroup = $strng3.$strng4;
            return $this->_agegroup;
            break;
        }
        if ($age > $low)
        {
            for($i = $low; $i <= $high; $i += $span -1)
            {
                if($i < $age)
                {
                    $i++;
                }
                else if($i == $age)
                {
                    $strng5 = $i;
                    $strng6 = " - ";
                    $strng7 = $i + $span-1;
                    $this->_agegroup = $strng5.$strng6.$strng7;
                    return $this->_agegroup;
                    break;
                }
            }
        }
    }
}

```



```

<input type="hidden" name="option" value="com_timing" />
<input type="hidden" name="task" value="select" />
<?php echo JHTML::_('form.token'); ?>
</form>

```

com_timing\views\timing\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-24 16:35:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This file will define view for manual registrations application.
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die( 'Restricted access' );

// bring in the functions related to JView
jimport( 'joomla.application.component.view' );

// The Joomla! 1.5 convention is component name, view and the name of the view
class TimingViewTiming extends JView
{
    function display($tpl = null)
    {
        $eventID = JRequest::getInt('eventID', '', 'post', 'string');
        if($eventID == '' || $eventID == 0)
        {
            $eventID = JRequest::getInt('eventID');
            if($eventID == '' || $eventID == 0)
            {
                $eventID = JRequest::getInt('event_ID');
            }
        }
        $query = "SELECT eventName from #__events WHERE eventID = '$eventID'";
        $db = & JFactory::getDBO();
        $db->setQuery($query);
        $eventName = $db->loadResult();
        JToolBarHelper::title( JText::_('Enter Bib and Time: ' . $eventName),
            'generic.png');
        $rows = & $this->get('data');
        $pagination = & $this->get('pagination');
        $this->assignRef('rows', $rows);
        $this->assignRef('pagination', $pagination);
        $this->assignRef('eventID', $eventID);
        parent::display($tpl);
    }
}

```

com_timing\views\timing\tmpl\default.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-24 16:47:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This file will define view for manual registrations application.
 */
$event_ID = (int) JRequest::getVar('event_ID', 0);
$eventID = $this->eventID;
?>
<form action="index.php?option=com_timing&task=save"
    method="post" id="josForm" name="josForm" style="font-size: 12pt;">
<table>
<tr>

```



```

</table>
</form>
<br />
<table class="adminlist" width="50%" align="center">
  <thead >
    <tr>
      <th width="3%">Bib</th>
      <th width="6%">First Name</th>
      <th width="6%">Middle name</th>
      <th width="6%">Last name</th>
      <th width="6%">City</th>
      <th width="3%">Age</th>
      <th width="5%">Age Group</th>
      <th width="5%">Time</th>
      <th width="5%">Pace</th>
    </tr>
  </thead>
  <?php
  jimport('joomla.filter.output');
  // Variabe that creates different CSS. 1 or 0 to distingush rows
  $k = 0;

  // Loop through the database as long as there are records
  for($i=0, $n=count($this->rows); $i < $n; $i++)
  {
    // reference to the current row
    $row =&$this->rows[$i];
    $has_paid = JHTML::_('grid.published', $row, $i);

  ?>

    <tr class="<?php echo "row$k"; ?>">
      <td>
        <?php echo $row->bib; ?>
      </td>
      <td>
        <a href="<?php echo $link; ?>"><?php echo $row->firstName;
?></a>
      </td>
      <td>
        <?php echo $row->middleName; ?>
      </td>
      <td>
        <?php echo $row->lastName; ?>
      </td>
      <td>
        <?php echo $row->city; ?>
      </td>
      <td>
        <?php echo $row->raceDayAge; ?>
      </td>
      <td>
        <?php echo $row->agegroup; ?>
      </td>
      <td>
        <?php echo $row->clock_time; ?>
      </td>
      <td>
        <?php echo $row->pace; ?>
      </td>
    </tr>
  <?php

  // Switch k between 1 and 0 for different CSS
  $k = 1 - $k;
  }
  ?>

  <tfoot>

```

```
<tr>
  <td colspan="14"><?php echo $this->pagination->getListFooter();
  ?></td>
</tr>
</tfoot>
</table>
```

APPENDIX B
PROJECT SOURCE CODE, FRONT END

Source Code com_flyer

com_flyer\flyer.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-19 10:46:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the entry point to view flyers on the web
 * and open the template in the editor.
 */

// Make sure that the class can only be invoked inside Joomla
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Make the Joomla framework core controller code available
jimport( 'joomla.application.component.controller' );

class FlyerController extends JController
{
    function display()
    {
        // get the view
        $view = JRequest::getVar('view');

        // If the view is not set, set it to list
        if ( !$view )
        {
            JRequest::setVar('view', 'list');
        }
        parent::display();
    }
}

// Create a new instance of the controller
$controller = new FlyerController();

// Execute the task variable
$controller->execute($task);

// Redirect after the task variable has been executed
$controller->redirect();
```

com_flyer\models\list.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-19 10:46:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model that creates the list of flyer logos shown on the web
 * and open the template in the editor.
 */

//Make sure the the file can only be invoked within Joomla!
defined( '_JEXEC' ) or die ( 'Restricted Control' );

//Make Joomla! view functionality available
jimport( 'joomla.application.component.model' );
class FlyerModelList extends JModel
{
    var $_data = null;
    function &getData()
    {
        if (empty($this->_data))
        {
```



```

        $query = "SELECT * FROM #__events WHERE published = '1'
                ORDER BY event_Date";
        $this->_data = $this->_getList($query);
    }
    return $this->_data;
}
}

```

com_flyer\views\list\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-19 10:46:PST
 * @package    Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * this file gets the list of records from the model and passes it to the default.php
 */

//Make sure the the file can only be invoked within Joomla!
defined( '_JEXEC' ) or die( 'Restricted access' );

//Make Joomla! view functionality available
jimport( 'joomla.application.component.view' );

class FlyerViewList extends JView
{
    function display($tpl = null)
    {
        //gets the list of data from the model
        $rows =& $this->get('data');
        //assign the list of data to the default page
        $this->assignRef('rows', $rows);
        parent::display($tpl);
    }
}

```

com_flyer\views\list\tmpl\default.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-19 10:48:PST
 * @package    Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the view for the list that is shown on the web
 */

//make sure that this file can only be invoked within Joomla!
defined( '_JEXEC' ) or die( 'Restricted access' ); ?>

<div class="componentheading">Local Running Events</div>

<?php
    foreach ($this->rows as $row)
    {
        $link = JRoute::_('index.php?option=com_flyer&eventID=' . $row->eventID .
            '&view=single');
        echo '<a href="' . $link . '">';
        $row->eventLogo . '</a>';
        echo '<hr>';
    }
?>

```

com_flyer\views\single\view.html.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-08 07:15:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the interface for the single review of events. It extends JView
 * from Joomla! 1.5 framework.
 */

// Make sure the file can only be invoked within Joomla!
defined( '_JEXEC' ) or die( 'Restricted access' );

// Make Joomla! view functionality available for the file
jimport( 'joomla.application.component.view' );

// Access tables in the back end of the system
JTable::addIncludePath(JPATH_ADMINISTRATOR.
    DS.'components'.DS.'com_flyer'.DS.'tables');

class FlyerViewSingle extends JView
{
    function display($tpl = null)
    {
        $id = (int) JRequest::getVar('eventID', 0);
        $events =& JTable::getInstance('events', 'table');
        $events->load($id);
        if ($events->published == 0)
        {
            JError::raiseError(404, 'Your requested Event is not Available');
        }
        $date = JHTML::date($events->event_Date);
        $backlink = 'index.php?option=com_flyer';
        $org = $events->orgID;
        $sql = "SELECT orgName, orgContact FROM #__organization WHERE orgID='$org'";
        $db =& JFactory::getDBO();
        $db->setQuery($sql);
        $result = $db->loadRow();
        $orgName = $result[0];
        $orgContact = $result[1];
        $this->assignRef('events', $events);
        $this->assignRef('date', $date);
        $this->assignRef('backlink', $backlink);
        $this->assignRef('orgName', $orgName);
        $this->assignRef('orgContact', $orgContact);

        parent::display($tpl);
    }
}
}
```

com_flyer\views\single\tmpl\default.php

```
<?php
//Make sure that the file can only be invoked within Joomla!
defined( '_JEXEC' ) or die( 'Restricted access' );
?>

<p align="left" style="font-size: 15pt;" ><strong><?php echo htmlspecialchars
($this->events->eventName) . ' ' . $this->date; ?></strong></p><br />

<p><?php echo $this->events->eventLogo; ?></p><br />
<table style="font-size: 9pt;">
    <tr>
        <td width="150">
            <strong>Event Type :</strong>
        </td>
        <td>

```

```

        <strong><?php echo htmlspecialchars($this->events->eventType);
        ?></strong>
    </td>
</tr>
<tr>
    <td width="150">
        <strong>Date :</strong>
    </td>
    <td>
        <strong><?php echo $this->date; ?></strong>
    </td>
</tr>
<tr>
    <td width="150">
        <strong>Event Place :</strong>
    </td>
    <td>
        <strong><?php echo htmlspecialchars($this->events->eventPlace);
        ?></strong>
    </td>
</tr>
<tr>
    <td width="150">
        <strong>Event Administrator :</strong>
    </td>
    <td>
        <strong><?php echo htmlspecialchars($this->orgContact); ?></strong>
    </td>
</tr>
<tr>
    <td width="150">
        <strong>Event Organization :</strong>
    </td>
    <td>
        <strong><?php echo htmlspecialchars($this->orgName); ?></strong>
    </td>
</tr>
<tr>
    <td width="150">
        <strong> Sign up for Event</strong>
    </td>
    <td>
        <?php
        $link = JRoute::_('index.php?option=com_login&view=login&eventID=' .
            $this->events->eventID);
        echo '<a href="' . $link . '>' . '<strong>' . 'Click Here' .
            '</strong>' . '</a>' ;
        ?>
    </td>
</tr>
</table>

<p><?php echo($this->events->eventDescription); ?></p>

<p><?php echo($this->events->eventInfo); ?></p>

<p>
<?php
$link = JRoute::_('index.php?option=com_login&view=login&eventID=' . $this->events->eventID);
echo '<a href="' . $link . '>' . '<strong>' . 'Sign Up' . '</strong>' . '</a>' ;
?>
</p>
<a href="<?php echo htmlspecialchars($this->backlink); ?>"><strong> return to Event
List</strong></a>

```

Source Code com_login

com_login\controller.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-19 10:46:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

// Make sure that the class can only be invoked inside Joomla
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Make the Joomla framework core controller code available
jimport( 'joomla.application.component.controller' );
date_default_timezone_set( 'America/Los_Angeles' );
class LoginController extends JController
{

function login()
{
    // Check for request forgeries
    JRequest::checkToken( 'request' ) or jexit( 'Invalid Token' );
    global $mainframe;
    $code = JRequest::getInt( 'code' );

    $eventID = JRequest::getInt( 'event_id' );
    $usrname = trim( $_POST[ 'username' ] );
    $password = trim( $_POST[ 'passwd' ] );
    $passwd1 = md5( $password );
    $query = "SELECT * FROM #__persons WHERE username = '$usrname' AND password = '$passwd1'";
    $db =& JFactory::getDBO();
    $db->setQuery( $query );
    $result = $db->loadResult();
    if ( !$result )
    {
        $msg = 'Username or Password is incorrect. Try again';
        $link = 'index.php?option=com_login&eventID=' . $eventID;
        $this->setRedirect( $link, $msg, 'error' );
    }
    if ( $code == -1 )
    {
        $rows = $db->loadObjectList();
        $session =& JFactory::getSession();
        foreach ( $rows as $row )
        {
            $session->set( 'id', $row->personID );
            $session->set( 'username', $row->userName );
        }
        $this->setRedirect( 'index.php?option=com_registrations&code=' . $code );
        return;
    }
    else
    {
        $rows = $db->loadObjectList();
        $session =& JFactory::getSession();
        $session->set( 'eventID', $eventID );
        foreach ( $rows as $row )
        {
            $session->set( 'id', $row->personID );
            $session->set( 'username', $row->userName );
        }
        // $this->setRedirect( 'index.php?option=com_registrations&eventID='
        // . $eventID . '&usrName=' . $usrname );
    }
}
}

```

```

        $this->setRedirect('index.php?option=com_registrations');
    }
}

function register()
{
    // Check for request forgeries
    JRequest::checkToken('request') or jexit( 'Invalid Token' );
    global $mainframe;
    $post =& Jrequest::get('post');
    $eventID = JRequest::getInt('event_id');
    $userName = JRequest::getVar('username', '', 'post', 'string');
    $password = JRequest::getVar('password', '', 'post', 'string',
    JREQUEST_ALLOWRAW);
    $password2 = JRequest::getVar('password2', '', 'post', 'string',
    JREQUEST_ALLOWRAW);
    $firstName = JRequest::getVar('firstName', '', 'post', 'string');
    $middleName = JRequest::getVar('middleName', '', 'post', 'string');
    $lastName = JRequest::getVar('lastName', '', 'post', 'string');
    $address = JRequest::getVar('address', '', 'post', 'string');
    $city = JRequest::getVar('city', '', 'post', 'string');
    $state = JRequest::getVar('state', '', 'post', 'string');
    $zip = JRequest::getVar('zip', '', 'post', 'string');
    $dob = JRequest::getVar('dob', '', 'post', 'date');
    $email = JRequest::getVar('email', '', 'post', 'string');
    $dayPhone = JRequest::getVar('dayPhone', '', 'post', 'string');
    $seveninPphone = JRequest::getVar('eveningPhone', '', 'post', 'string');

    $query = "SELECT * FROM #__persons WHERE username = '$userName'";
    $db =& JFactory::getDBO();
    $db->setQuery($query);
    $result = $db->loadResult();
    if ($result)
    {
        $msg = 'Username is Taken. Choose another username';
        // $link = 'index.php?option=com_login&view=register&javascript:history.go(-1)';
        $link = 'index.php?option=com_login&view=reload&v1=' . $firstName .
            '&v2=' . $middleName . '&v3=' . $lastName . '&v4=' . $address .
            '&v5=' . $city . '&v6=' . $state . '&v7=' . $zip . '&v8=' . $dob
            . '&v9=' . $dayPhone . '&v10=' . $seveninPphone . '&v11='
            . $email . '&eventID=' . $eventID;
        $this->setRedirect($link, $msg, 'error');
        return;
    }

    else if ($password == '' || $password2 == '')
    {
        $msg = 'You have to enter a password. Use the Browser Back Button to
        correct this problem';
        $link = 'index.php?option=com_login&view=reload&v1=' . $firstName .
            '&v2=' . $middleName . '&v3=' . $lastName . '&v4=' . $address .
            '&v5=' . $city . '&v6=' . $state . '&v7=' . $zip . '&v8=' . $dob
            . '&v9=' . $dayPhone . '&v10=' . $seveninPphone . '&v11=' . $email
            . '&eventID=' . $eventID;
        $this->setRedirect($link, $msg, 'error');
        Echo 'password 1 equal zero';
        return;
    }

    else if($password != $password2)
    {
        $msg = 'Passwords do not match';
        $link = 'index.php?option=com_login&view=reload&v1=' . $firstName .
            '&v2=' . $middleName . '&v3=' . $lastName . '&v4=' . $address .
            '&v5=' . $city . '&v6=' . $state . '&v7=' . $zip . '&v8=' . $dob
            . '&v9=' . $dayPhone . '&v10=' . $seveninPphone . '&v11=' .
            $email . '&eventID=' . $eventID;
        $this->setRedirect($link, $msg, 'error');
        return;
    }

    else if($dob == '')
    {

```

```

        $msg = 'You have to enter a Date of Birth';
        $link = 'index.php?option=com_login&view=reload&v1=' . $firstName .
            '&v2=' . $middleName . '&v3=' . $lastName . '&v4=' . $address .
            '&v5=' . $city . '&v6=' . $state . '&v7=' . $zip . '&v8=' . $dob
            . '&v9=' . $dayPhone . '&v10=' . $eveningPhone . '&v11=' .
            $email . '&eventID=' . $eventID;
        $this->setRedirect($link, $msg, 'error');
        return;
    }
    else
    {
        JFactory::addIncludePath(JPATH_ADMINISTRATOR.
            DS.'components'.DS.'com_registrations'.DS.'tables');
        $row =& JFactory::getInstance('persons', 'Table');
        $row->personID = JRequest::getInt('personID', 0);
        $row->userName = JRequest::getString('username', '');
        $row->password = md5(JRequest::getString('password', ''));
        $row->firstName = JRequest::getString('firstName', '');
        $row->middleName = JRequest::getString('middleName', '');
        $row->lastName = JRequest::getString('lastName', '');
        $row->address = JRequest::getString('address', '');
        $row->city = JRequest::getString('city', '');
        $row->state = JRequest::getString('state', '');
        $row->zip = JRequest::getString('zip', '');
        $row->email = JRequest::getString('email', '');
        $row->dob = JRequest::getString('dob', '');
        $row->dayPhone = JRequest::getString('dayPhone', '');
        $row->eveningPhone = JRequest::getString('eveningPhone', '');
        $date =& JFactory::getDate();
        $row->registerDate = JRequest::getString('registerDate',
            $date->toMySQL());
        $row->gender = JRequest::getString('gender', '');
        $row->role = 'Athlete';
        if(!$row->store())
        {
            JError::raiseError(500, $row->getError());
        }
        $sql = "SELECT * FROM #__persons WHERE username = '$userName' ";

        $db =& JFactory::getDBO();
        $db->setQuery($sql);
        $result = $db->loadResult();
        if (!$result)
        {
            JError::raiseError(500, 'Something went wrong!');
        }
        else
        {
            $rows = $db->loadObjectList();
            $session =& JFactory::getSession();
            $session->set('eventID', $eventID);
            foreach ($rows as $row)
            {
                $session->set('id', $row->personID);
                $session->set('city', $row->city);
                $session->set('username', $row->userName);
            }
            $this->setRedirect('index.php?option=com_registrations&eventID='
                . $eventID . '&usrName=' . $row->userName);
        }
    }
}
function requestreset()
{
    $username = JRequest::getVar('userName', null, 'post', 'string');

    $model = &$this->getModel('Reset');
    if ($model->requestReset($username) === false)
    {
        $message = JText::sprintf('The username you entered is not valid.
            Try again.', $model->getError());
    }
}

```

```

        $this->setRedirect('index.php?option=com_login&view=reset', $message);
        return false;
    }
    $this->setRedirect('index.php?option=com_login&view=reset&layout=confirm');
}

function confirmreset()
{
    JRequest::checkToken() or jexit( 'Invalid Token' );

    // Get the input
    $token = JRequest::getVar('token', null, 'post', 'alnum');

    // Get the model
    $model = &$this->getModel('Reset');

    // Verify the token
    if ($model->confirmReset($token) === false)
    {
        $message = JText::sprintf('Token is not correct.
        Make sure you entered the Token that was sent to your
        email.',
        , $model->getError());
        $this->setRedirect('index.php?option=com_login&view=reset
        &layout=confirm', $message);
        return false;
    }
    $id = $model->confirmReset($token);
    $this->setRedirect('index.php?option=com_login&view=reset
    &layout=complete&id=' . $id);
}

function completereset()
{
    JRequest::checkToken() or jexit( 'Invalid Token' );
    $id = JRequest::getVar('id');

    // Get the input
    $password1 = JRequest::getVar('password1', null, 'post', 'string',
    JREQUEST_ALLOWRAW);
    $password2 = JRequest::getVar('password2', null, 'post', 'string',
    JREQUEST_ALLOWRAW);

    // Get the model
    $model = &$this->getModel('Reset');

    // Reset the password
    if ($model->completereset($password1, $password2, $id) === false)
    {
        $message = JText::sprintf('Reset Password Failed. Try Again',
        $model->getError());
        $this->setRedirect('index.php?
        option=com_login&view=reset&layout=complete', $message);
        return false;
    }

    $message = JText::_('Reset of Password was a Success. ');
    $this->setRedirect('index.php?option=com_flyer', $message);
}

function display()
{
    // get the view
    $view = JRequest::getVar('view');
    if(!$view)
    {
        JRequest::setVar('view', 'login');
    }
    parent::display();
}
}
}

```

com_login\login.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-19 10:40:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

require_once( JPATH_COMPONENT_DS.'controller.php' );

// Create a new instance of the controller
$controller = new LoginController();

// Execute the task variable
$controller->execute($task);

// Redirect after the task variable has been executed
$controller->redirect();
```

com_login\models\reset.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-05 18:10:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model used to reset the password
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport( 'joomla.application.component.model' );

class LoginModelReset extends JModel
{
    var $_namespace      = 'com_user.reset.';
    var $pid = null;

    function requestReset($username)
    {
        jimport( 'joomla.mail.helper' );
        jimport( 'joomla.user.helper' );
        $sql = "SELECT personID FROM #__persons WHERE userName = '$username'";
        $db =& JFactory::getDBO();
        $db->setQuery($sql);
        if ( !($sid = $db->loadResult()) )
        {
            return false;
        }
        $token = JUtility::getHash( JUserHelper::genRandomPassword() );
        $query = "UPDATE #__persons SET token = '$token' WHERE personID = '$sid'";
        $db->setQuery($query);
        if ( !$db->query() )
        {
            $this->setError( JText::_ ( 'DATABASE_ERROR' ) );
            return false;
        }
        $sql2 = "SELECT email FROM #__persons WHERE personID = '$sid' ";
        $db->setQuery($sql2);
        if ( !($email = $db->loadResult()) )
```



```

    {
        return false;
    }
    if (!JMailHelper::isEmailAddress($email))
    {
        return false;
    }

    /*****
    * Use the JUtility::sendMail() so send the token to the email that is
    * given fr the user.
    * if (!$this->_sendConfirmationMail($email, $token))
    * {
    *     return false;
    * }
    */
    return true;
}

function confirmReset($token)
{
    global $mainframe;

    if(strlen($token) != 32)
    {
        $this->setError(JText::_('INVALID_TOKEN'));
        return false;
    }
    $db = &JFactory::getDBO();
    $db->setQuery("SELECT personID FROM #__persons WHERE token = '$token'");

    // Verify the token
    if (!$id = $db->loadResult())
    {
        $this->setError(JText::_('INVALID_TOKEN'));
        return false;
    }
    //Return true;
    $this->pid = $id;
    return $this->pid;
}

function completeReset($password1, $password2, $id)
{
    if (! $password1 )
    {
        $this->setError(JText::_('You Must Enter a Password'));
        return false;
    }

    // Verify that the passwords match
    if ($password1 != $password2)
    {
        $this->setError(JText::_('Passwords do not Match'));
        return false;
    }
    $password = md5(JRequest::getString('password1', ''));
    //JError::raiseError(500, $password);
    $query = "UPDATE #__persons SET password = '$password', token = ''
        WHERE personID = '$id'";
    $db =& JFactory::getDBO();
    $db->setQuery($query);
    if (!$db->query())
    {
        $this->setError(JText::_('DATABASE_ERROR'));
        return false;
    }
    Return true;
}
}
}

```

com_login\views\login\view.html.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-05 18:10:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved
 * this is the login view
 */

// Make sure that the file can only be invoked within Joomla!
defined( '_JEXEC' ) or die ( 'Access Restricted' );
jimport( 'joomla.application.component.view' );
JTable::addIncludePath(JPATH_ADMINISTRATOR.
    DS.'components'.DS.'com_flyer'.DS.'tables');

class LoginViewLogin extends JView
{
    function display($tpl = null)
    {
        $id = (int) JRequest::getVar('eventID', 0);
        $events =& JTable::getInstance('events', 'table');
        $events->load($id);
        $this->assignRef('events', $events);
        parent::display($tpl);
    }
}
```

com_login\views\login\tmpl\default.php

```
// Make sure that the file can only be invoked within Joomla!
<?php defined( '_JEXEC' ) or die( 'Restricted access' );

?>

<form action="index.php?option=com_login&task=login" method="post" name="adminForm"
class="inputbox" >

<fieldset class="input">
    <legend>Login</legend>
    <p id="com-form-login-username">
        <label for="username"><?php echo JText::_('Username') ?></label><br />
        <input name="username" id="username" type="text" class="inputbox"
            accept="" alt="username" size="18" />
    </p>
    <p id="com-form-login-password">
        <label for="passwd"><?php echo JText::_('Password') ?></label><br />
        <input type="password" id="passwd" name="passwd" class="inputbox"
            size="18" alt="password" />
    </p>
    <?php if(JPluginHelper::isEnabled('system', 'remember')) : ?>
    <p id="com-form-login-remember">
        <label for="remember"><?php echo JText::_('Remember me') ?></label>
        <input type="checkbox" id="remember" name="remember" class="inputbox"
            value="yes" alt="Remember Me" />
    </p>
    <?php endif; ?>
    <input type="submit" name="Submit" class="button" value="<?php echo
JText::_('LOGIN') ?>"
        onclick="gotocontact( '<?php echo $this->contact[0]->id; ?>' )" />
</fieldset>

<ul>
<li>
    <a href="<?php echo JRoute::_('index.php?option=com_login&view=reset'
); ?>">
        <?php echo JText::_('FORGOT YOUR PASSWORD'); ?></a>
</li>
```

```

<li>
    <a href="<?php echo JRoute::_(
'index.php?option=com_login&view=register&eventID=
    . $this->events->eventID ); ?>">
        <?php echo JText::_('REGISTER'); ?></a>
</li>
</ul>

<input type="hidden" name="option" value="com_login" />
<input type="hidden" name="task" value="login" />
<input type="hidden" name="event_id" value="<?php echo $this->events-
>eventID ?>" />
<input type="hidden" name="return" value="<?php echo $this->return; ?>" />
<?php echo JHTML::_('form.token'); ?>
</form>

```

com_login\views\register\view.html.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
defined( '_JEXEC' ) or die ( 'Access Restricted' );
jimport( 'joomla.application.component.view' );
JTable::addIncludePath(JPATH_ADMINISTRATOR.
    DS.'components'.DS.'com_flyer'.DS.'tables');
JTable::addIncludePath(JPATH_ADMINISTRATOR.
    DS.'components'.DS.'com_registrations'.DS.'tables');

class LoginViewRegister extends JView
{
    function display($tpl = null)
    {

        $event_id = (int) JRequest::getVar('eventID', 0);
        $person_id = (int) JRequest::getVar('personID', 0);
        $rowEvent =& JTable::getInstance('events', 'Table');
        $rowEvent->load($event_id);
        $rowPerson = & JTable::getInstance('persons', 'Table');
        $rowPerson->load($person_id);
        $this->assignRef('dob', JHTML::_('calendar',
            $rowPerson->dob, 'dob', 'dob'));
        $this->assignRef('rowPerson', $rowPerson);
        $this->assignRef('rowEvent', $rowEvent);
        $gender[] = JHTML::_('select.option', 'male', 'Male:');
        $gender[] = JHTML::_('select.option', 'female', 'Female:');
        $lists['gender'] = JHTML::_('select.radiolist', $gender, 'gender',
            'class="inputbox"', 'value', 'text', 'male', 'gender');
        $this->assignRef('lists', $lists);
        $state = array(
            JHTML::_('select.option', null, JText::_('Select State')),
            JHTML::_('select.option', 'AL', JText::_('AL')),
            JHTML::_('select.option', 'AK', JText::_('AK')),
            JHTML::_('select.option', 'AR', JText::_('AR')),
            JHTML::_('select.option', 'AZ', JText::_('AZ')),
            JHTML::_('select.option', 'AB', JText::_('AB')),
            JHTML::_('select.option', 'BC', JText::_('BC')),
            JHTML::_('select.option', 'CA', JText::_('CA')),
            JHTML::_('select.option', 'CO', JText::_('CO')),
            JHTML::_('select.option', 'CT', JText::_('CT')),
            JHTML::_('select.option', 'DE', JText::_('DE')),
            JHTML::_('select.option', 'FL', JText::_('FL')),
            JHTML::_('select.option', 'GA', JText::_('GA')),
            JHTML::_('select.option', 'HI', JText::_('HI')),
            JHTML::_('select.option', 'IA', JText::_('IA')),
            JHTML::_('select.option', 'ID', JText::_('ID')),

```

```

JHTML::_('select.option', 'IL', JText::_('IL')),
JHTML::_('select.option', 'IN', JText::_('IN')),
JHTML::_('select.option', 'KS', JText::_('KS')),
JHTML::_('select.option', 'KY', JText::_('KY')),
JHTML::_('select.option', 'LA', JText::_('LA')),
JHTML::_('select.option', 'MA', JText::_('MA')),
JHTML::_('select.option', 'MB', JText::_('MB')),
JHTML::_('select.option', 'MD', JText::_('MD')),
JHTML::_('select.option', 'ME', JText::_('ME')),
JHTML::_('select.option', 'MI', JText::_('MI')),
JHTML::_('select.option', 'MN', JText::_('MN')),
JHTML::_('select.option', 'MO', JText::_('MO')),
JHTML::_('select.option', 'MS', JText::_('MS')),
JHTML::_('select.option', 'MT', JText::_('MT')),
JHTML::_('select.option', 'NB', JText::_('NB')),
JHTML::_('select.option', 'NC', JText::_('NC')),
JHTML::_('select.option', 'ND', JText::_('ND')),
JHTML::_('select.option', 'NE', JText::_('NE')),
JHTML::_('select.option', 'NH', JText::_('NH')),
JHTML::_('select.option', 'NJ', JText::_('NJ')),
JHTML::_('select.option', 'NL', JText::_('NL')),
JHTML::_('select.option', 'NM', JText::_('NM')),
JHTML::_('select.option', 'NS', JText::_('NS')),
JHTML::_('select.option', 'NT', JText::_('NT')),
JHTML::_('select.option', 'NU', JText::_('NU')),
JHTML::_('select.option', 'NV', JText::_('NV')),
JHTML::_('select.option', 'NY', JText::_('NY')),
JHTML::_('select.option', 'OH', JText::_('OH')),
JHTML::_('select.option', 'OK', JText::_('OK')),
JHTML::_('select.option', 'ON', JText::_('ON')),
JHTML::_('select.option', 'OR', JText::_('OR')),
JHTML::_('select.option', 'PA', JText::_('PA')),
JHTML::_('select.option', 'PE', JText::_('PE')),
JHTML::_('select.option', 'QC', JText::_('QC')),
JHTML::_('select.option', 'RI', JText::_('RI')),
JHTML::_('select.option', 'SC', JText::_('SC')),
JHTML::_('select.option', 'SD', JText::_('SD')),
JHTML::_('select.option', 'SK', JText::_('SK')),
JHTML::_('select.option', 'TN', JText::_('TN')),
JHTML::_('select.option', 'TX', JText::_('TX')),
JHTML::_('select.option', 'UT', JText::_('UT')),
JHTML::_('select.option', 'VA', JText::_('VA')),
JHTML::_('select.option', 'VT', JText::_('VT')),
JHTML::_('select.option', 'WA', JText::_('WA')),
JHTML::_('select.option', 'WI', JText::_('WI')),
JHTML::_('select.option', 'WV', JText::_('WV')),
JHTML::_('select.option', 'WY', JText::_('WY')),
JHTML::_('select.option', 'YT', JText::_('YT')),
JHTML::_('select.option', 'XX', JText::_('No State'))
);
$this->assignRef('state', JHTML::_('select.genericList',
    $state, 'state', 'class="inputbox" ' . ' ', 'value',
    'text', $rowPerson->state));
JHTML::_('behavior.formvalidation');
parent::display($tpl);
}
}

```

com_login\views\register\tmpl\default.php

```

<?php // no direct access
defined('_JEXEC') or die('Restricted access');
JHTML::_('behavior.calendar');
?>
<script type="text/javascript">
<!--
    Window.onload(function(){
        document.formvalidator.setHandler('passverify', function (value)
            { return $('password').value == value; }
        );
    });

```

```

// -->
</script>

<?php
    if(isset($this->message)){
        $this->display('message');
    }
?>
<?php $eventID=JRequest::getVar('eventID'); ?>
<form action="<?php echo JRoute::_('index.php?option=com_login' ); ?>"
    method="post" id="josForm" name="josForm" class="form-validate"
    style="font-size: 8pt;">
<fieldset>
    <legend>Login data</legend>
<table cellpadding="0" cellspacing="0" border="0" width="100%" class="contentpane">
<tr>
    <td width="20%" height="25">
        <label id="usernamemsg" for="username">
            <?php echo JText::_('Username' ); ?>:
        </label>
    </td>
    <td>
        <input type="text" id="username" name="username" size="40"
            value="<?php echo $this->escape($this->rowPerson->get('userName'
        ));?>"
            class="inputbox required validate-username" maxlength="25"
            style="font-size: 8pt" /> *
    </td>
</tr>
<tr>
    <td height="25">
        <label id="pwmsg" for="password">
            <?php echo JText::_('Password' ); ?>:<br /> Minimum 6
            Characters
        </label>
    </td>
    <td>
        <input class="inputbox required validate-password" type="password"
            id="password" name="password" size="40" value="" style="font-size:
            8pt" /> *
    </td>
</tr>
<tr>
    <td height="25">
        <label id="pw2msg" for="password2">
            <?php echo JText::_('Verify Password' ); ?>:
        </label>
    </td>
    <td>
        <input class="inputbox required validate-passverify" type="password"
            id="password2" name="password2" size="40" value="" style="font-size:
            8pt" /> *
    </td>
</tr>
</table>
</fieldset>
<fieldset>
    <legend>Personal Information</legend>
<table cellpadding="0" cellspacing="0" border="0" width="100%" class="contentpane">
<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="firstName">
            <?php echo JText::_('First Name' ); ?>:
        </label>
    </td>
    <td>
        <input type="text" name="firstName" id="firstName" size="40"
            value="<?php echo $this->escape($this->rowPerson->get('firstName' ));?>"
            class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

```

```

<tr>
  <td width="20%" height="25">
    <label id="namemsg" for="middleName">
      <?php echo JText::_('Middle Name' ); ?>
    </label>
  </td>
  <td>
    <input type="text" name="middleName" id="middleName" size="40"
      value="<?php echo $this->rowPerson->middleName;?>"
      class="" maxlength="50" style="font-size: 8pt" />
    </td>
</tr>

<tr>
  <td width="20%" height="25">
    <label id="namemsg" for="lastName">
      <?php echo JText::_('Last Name' ); ?>
    </label>
  </td>
  <td>
    <input type="text" name="lastName" id="lastName" size="40"
      value="<?php echo $this->rowPerson->lastName;?>"
      class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
  <td width="20%" height="25">
    <label id="namemsg" for="address">
      <?php echo JText::_('Address' ); ?>
    </label>
  </td>
  <td>
    <input type="text" name="address" id="address" size="40"
      value="<?php echo $this->rowPerson->address;?>"
      class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
  <td width="20%" height="25">
    <label id="namemsg" for="city">
      <?php echo JText::_('City' ); ?>
    </label>
  </td>
  <td>
    <input type="text" name="city" id="city" size="40"
      value="<?php echo $this->rowPerson->city;?>"
      class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
  <td width="20%" height="25">
    <label id="namemsg" for="state">
      <?php echo JText::_('State' ); ?>
    </label>
  </td>
  <td>
    <?php echo $this->state; ?> *
  </td>
</tr>

<tr>
  <td width="20%" height="25">
    <label id="namemsg" for="zip">
      <?php echo JText::_('Zip' ); ?>
    </label>
  </td>
  <td>
    <input type="text" name="zip" id="zip" size="40"
      value="<?php echo $this->rowPerson->zip;?>"
    </td>
</tr>

```

```

        class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="dob">
            <?php echo JText::_{ 'Date of Birth' }; ?>:
        </label>
    </td>
    <td>
        <?php echo $this->dob ?>
    </td>
</tr>

<tr>
    <td height="25">
        <label id="emailmsg" for="email">
            <?php echo JText::_{ 'Email' }; ?>:
        </label>
    </td>
    <td>
        <input type="text" id="email" name="email" size="40"
            value="<?php echo $this->rowPerson->email;?>"
            class="inputbox required validate-email"
            maxlength="100" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="dayPhone">
            <?php echo JText::_{ 'Phone' }; ?>:
        </label>
    </td>
    <td>
        <input type="text" name="dayPhone" id="dayPhone" size="40"
            value="<?php echo $this->escape{$this->rowPerson->get('dayPhone')}; ?>"
            class="inputbox required validate-phone" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="eveningPhone">
            <?php echo JText::_{ 'Evening Phone' }; ?>:
        </label>
    </td>
    <td>
        <input type="text" name="eveningPhone" id="eveningPhone" size="40"
            value="<?php echo $this->rowPerson->eveningPhone;?>"
            class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="gender">
            <?php echo JText::_{ 'Gender' }; ?>:
        </label>
    </td>
    <td>
        <?php
            echo $this->lists['gender'];
        ?>
    </td>
</tr>

```

```

<tr>
  <td colspan="2" height="30">
    <?php echo JText::_(' * Required Fields' ); ?>
  </td>
</tr>
</table>
<table>
  <button class="button validate" type="submit"><?php echo JText::_('Register'); ?></button>
  <button type="reset"><?php echo JText::_('Reset'); ?></button>
  <input type="hidden" name="task" value="register" />
  <input type="hidden" name="event_id" value="<?php echo $eventID; ?>" />
  <input type="hidden" name="gid" value="0" />
  <?php echo JHTML::_('form.token' ); ?>
</fieldset>
</form>

```

com_login\views\reload\view.html.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

defined( 'JEXEC' ) or die ( 'Access Restricted' );
jimport( 'joomla.application.component.view' );

JTable::addIncludePath(JPATH_ADMINISTRATOR.
  DS.'components'.DS.'com_flyer'.DS.'tables');
JTable::addIncludePath(JPATH_ADMINISTRATOR.
  DS.'components'.DS.'com_registrations'.DS.'tables');

class LoginViewReload extends JView
{
  function display ($tpl = null)
  {
    $event_id = (int) JRequest::getVar('eventID', 0);
    $person_id = (int) JRequest::getVar('personID', 0);
    $rowEvent =& JTable::getInstance('events', 'Table');
    $rowEvent->load($event_id);
    $rowPerson = & JTable::getInstance('persons', 'Table');
    $cid = JRequest::getVar( 'cid', array(0), '', 'array' );
    $person_id = $cid[0];
    $rowPerson->load($person_id);
    $this->assignRef('dob', JHTML::_('calendar',
      $rowPerson->dob, 'dob', 'dob'));
    $this->assignRef('rowPerson', $rowPerson);
    $this->assignRef('rowEvent', $rowEvent);
    $gender[] = JHTML::_('select.option', 'male', 'Male:');
    $gender[] = JHTML::_('select.option', 'female', 'Female:');
    $lists['gender'] = JHTML::_('select.radiolist', $gender, 'gender',
      'class="inputbox" ', 'value', 'text', 'male', 'gender');
    $this->assignRef('lists', $lists);
    $msg = JRequest::getVar('msg');
    $this->assignRef('msg', $msg);
    $state = array(
      JHTML::_('select.option', null, JText::_('Select State')),
      JHTML::_('select.option', 'AL', JText::_('AL')),
      JHTML::_('select.option', 'AK', JText::_('AK')),
      JHTML::_('select.option', 'AR', JText::_('AR')),
      JHTML::_('select.option', 'AZ', JText::_('AZ')),
      JHTML::_('select.option', 'AB', JText::_('AB')),
      JHTML::_('select.option', 'BC', JText::_('BC')),
      JHTML::_('select.option', 'CA', JText::_('CA')),
      JHTML::_('select.option', 'CO', JText::_('CO')),
      JHTML::_('select.option', 'CT', JText::_('CT')),
      JHTML::_('select.option', 'DE', JText::_('DE')),
      JHTML::_('select.option', 'FL', JText::_('FL')),
      JHTML::_('select.option', 'GA', JText::_('GA')),
      JHTML::_('select.option', 'HI', JText::_('HI')),
      JHTML::_('select.option', 'IA', JText::_('IA')),
      JHTML::_('select.option', 'ID', JText::_('ID')),

```



```

JHTML::_('select.option', 'IL', JText::_('IL')),
JHTML::_('select.option', 'IN', JText::_('IN')),
JHTML::_('select.option', 'KS', JText::_('KS')),
JHTML::_('select.option', 'KY', JText::_('KY')),
JHTML::_('select.option', 'LA', JText::_('LA')),
JHTML::_('select.option', 'MA', JText::_('MA')),
JHTML::_('select.option', 'MB', JText::_('MB')),
JHTML::_('select.option', 'MD', JText::_('MD')),
JHTML::_('select.option', 'ME', JText::_('ME')),
JHTML::_('select.option', 'MI', JText::_('MI')),
JHTML::_('select.option', 'MN', JText::_('MN')),
JHTML::_('select.option', 'MO', JText::_('MO')),
JHTML::_('select.option', 'MS', JText::_('MS')),
JHTML::_('select.option', 'MT', JText::_('MT')),
JHTML::_('select.option', 'NB', JText::_('NB')),
JHTML::_('select.option', 'NC', JText::_('NC')),
JHTML::_('select.option', 'ND', JText::_('ND')),
JHTML::_('select.option', 'NE', JText::_('NE')),
JHTML::_('select.option', 'NH', JText::_('NH')),
JHTML::_('select.option', 'NJ', JText::_('NJ')),
JHTML::_('select.option', 'NL', JText::_('NL')),
JHTML::_('select.option', 'NM', JText::_('NM')),
JHTML::_('select.option', 'NS', JText::_('NS')),
JHTML::_('select.option', 'NT', JText::_('NT')),
JHTML::_('select.option', 'NU', JText::_('NU')),
JHTML::_('select.option', 'NV', JText::_('NV')),
JHTML::_('select.option', 'NY', JText::_('NY')),
JHTML::_('select.option', 'OH', JText::_('OH')),
JHTML::_('select.option', 'OK', JText::_('OK')),
JHTML::_('select.option', 'ON', JText::_('ON')),
JHTML::_('select.option', 'OR', JText::_('OR')),
JHTML::_('select.option', 'PA', JText::_('PA')),
JHTML::_('select.option', 'PE', JText::_('PE')),
JHTML::_('select.option', 'QC', JText::_('QC')),
JHTML::_('select.option', 'RI', JText::_('RI')),
JHTML::_('select.option', 'SC', JText::_('SC')),
JHTML::_('select.option', 'SD', JText::_('SD')),
JHTML::_('select.option', 'SK', JText::_('SK')),
JHTML::_('select.option', 'TN', JText::_('TN')),
JHTML::_('select.option', 'TX', JText::_('TX')),
JHTML::_('select.option', 'UT', JText::_('UT')),
JHTML::_('select.option', 'VA', JText::_('VA')),
JHTML::_('select.option', 'VT', JText::_('VT')),
JHTML::_('select.option', 'WA', JText::_('WA')),
JHTML::_('select.option', 'WI', JText::_('WI')),
JHTML::_('select.option', 'WV', JText::_('WV')),
JHTML::_('select.option', 'WY', JText::_('WY')),
JHTML::_('select.option', 'YT', JText::_('YT')),
JHTML::_('select.option', 'XX', JText::_('No State'))
);
$this->assignRef('state', JHTML::_('select.genericList',
    $state, 'state', 'class="inputbox" ' . ' ', 'value',
    'text', $rowPerson->state));
JHTML::_('behavior.formvalidation');

parent::display($tpl);
}
}

```

com_login\views\reload\tmpl\default.php

```

<?php // no direct access
defined('_JEXEC') or die('Restricted access');
JHTML::_('behavior.calendar');
?>
<script type="text/javascript">
<!--
    Window.onDomReady(function(){

```

```

        document.formvalidator.setHandler('passverify', function (value) { return
        ($('#password').value == value); } );
    });
// -->
</script>

<?php
    if(isset($this->message)){
        $this->display('message');
    }
?>
<?php echo $eventID=JRequest::getVar('eventID');
echo 'eventID = ' . $eventID; ?>
<form action="<?php echo JRoute::_('index.php?option=com_login' ); ?>"
method="post" id="josForm" name="josForm" class="form-validate"
style="font-size: 8pt;">
<fieldset>
    <legend>Login data</legend>
<table cellpadding="0" cellspacing="0" border="0" width="100%" class="contentpane">
<tr>
    <td width="20%" height="25">
        <label id="usernamemsg" for="username">
            <?php echo JText::_('Username' ); ?>:
        </label>
    </td>
    <td>
        <input type="text" id="username" name="username" size="40"
value="<?php echo $this->escape($this->rowPerson->get( 'userName'
));?>"
class="inputbox required validate-username" maxlength="25"
style="font-size: 8pt" /> *
    </td>
</tr>
<tr>
    <td height="25">
        <label id="pwmsg" for="password">
            <?php echo JText::_('Password' ); ?>:<br /> Minimum 6
            Characters
        </label>
    </td>
    <td>
        <input class="inputbox required validate-password" type="password"
id="password" name="password" size="40" value="" style="font-size:
8pt" /> *
    </td>
</tr>
<tr>
    <td height="25">
        <label id="pw2msg" for="password2">
            <?php echo JText::_('Verify Password' ); ?>:
        </label>
    </td>
    <td>
        <input class="inputbox required validate-passverify" type="password"
id="password2" name="password2" size="40" value="" style="font-size:
8pt" /> *
    </td>
</tr>
</table>
</fieldset>
<fieldset>
    <legend>Personal Information</legend>
<table cellpadding="0" cellspacing="0" border="0" width="100%" class="contentpane">
<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="firstName">
            <?php echo JText::_('First Name' ); ?>:
        </label>
    </td>
    <td>
        <input type="text" name="firstName" id="firstName" size="40"
value="<?php echo $firstName = JRequest::getVar('v1');?>"

```

```

        class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="middleName">
            <?php echo JText::_('Middle Name' ); ?>:
        </label>
    </td>
    <td>
        <input type="text" name="middleName" id="middleName" size="40"
            value="<?php echo $middleName = JRequest::getVar('v2');?>"
            class="" maxlength="50" style="font-size: 8pt" />
    </td>
</tr>

<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="lastName">
            <?php echo JText::_('Last Name' ); ?>:
        </label>
    </td>
    <td>
        <input type="text" name="lastName" id="lastName" size="40"
            value="<?php echo $lastName = JRequest::getVar('v3');?>"
            class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="address">
            <?php echo JText::_('Address' ); ?>:
        </label>
    </td>
    <td>
        <input type="text" name="address" id="address" size="40"
            value="<?php echo $address = JRequest::getVar('v4');?>"
            class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="city">
            <?php echo JText::_('City' ); ?>:
        </label>
    </td>
    <td>
        <input type="text" name="city" id="city" size="40"
            value="<?php echo $city = JRequest::getVar('v5');?>"
            class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
    </td>
</tr>

<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="state">
            <?php echo JText::_('State' ); ?>:
        </label>
    </td>
    <td>
        <?php echo $this->state; ?> *
    </td>
</tr>

<tr>
    <td width="20%" height="25">
        <label id="namemsg" for="zip">
            <?php echo JText::_('Zip' ); ?>:
        </label>

```

```

        </td>
        <td>
            <input type="text" name="zip" id="zip" size="40"
                value="<?php echo $zip = JRequest::getVar('v7');?>"
                class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
        </td>
    </tr>

    <tr>
        <td width="20%" height="25">
            <label id="namemsg" for="dob">
                <?php echo JText::_('Date of Birth' ); ?>:
            </label>
        </td>
        <td>
            <?php echo $this->dob?>
        </td>
    </tr>

    <tr>
        <td height="25">
            <label id="emailmsg" for="email">
                <?php echo JText::_('Email' ); ?>:
            </label>
        </td>
        <td>
            <input type="text" id="email" name="email" size="40"
                value="<?php echo $email = JRequest::getVar('v11');?>"
                class="inputbox required validate-email"
                maxlength="100" style="font-size: 8pt" /> *
        </td>
    </tr>

    <tr>
        <td width="20%" height="25">
            <label id="namemsg" for="dayPhone">
                <?php echo JText::_('Phone' ); ?>:
            </label>
        </td>
        <td>
            <input type="text" name="dayPhone" id="dayPhone" size="40"
                value="<?php echo $dayPhone = JRequest::getVar('v9');?>"
                class="inputbox required validate-phone" maxlength="50" style="font-size: 8pt" /> *
        </td>
    </tr>

    <tr>
        <td width="20%" height="25">
            <label id="namemsg" for="eveningPhone">
                <?php echo JText::_('Evening Phone' ); ?>:
            </label>
        </td>
        <td>
            <input type="text" name="eveningPhone" id="eveningPhone" size="40"
                value="<?php echo $eveningPhone = JRequest::getVar('v10');?>"
                class="inputbox required" maxlength="50" style="font-size: 8pt" /> *
        </td>
    </tr>

    <tr>
        <td width="20%" height="25">
            <label id="namemsg" for="gender">
                <?php echo JText::_('Gender' ); ?>:
            </label>
        </td>
        <td>
            <?php
                echo $this->lists['gender'];
            ?>
        </td>
    </tr>

```

```

</tr>
<tr>
  <td colspan="2" height="30">
    <?php echo JText::_(' * Required Fields' ); ?>
  </td>
</tr>
</table>
  <button class="button validate" type="submit"><?php echo JText::_('Register'); ?></button>
  <button type="reset"><?php echo JText::_('Reset'); ?></button>
  <input type="hidden" name="task" value="register" />
  <input type="hidden" name="event_id" value="<?php echo $eventID; ?>" />
  <input type="hidden" name="gid" value="0" />
  <?php echo JHTML::_('form.token' ); ?>
</fieldset>
</form>

```

com_login\views\reset\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2010-01-07 08:21: PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the interface for the reset password view. It extends JView
 * from Joomla! 1.5 framework.
 */

defined( '_JEXEC' ) or die( 'Restricted access' );
jimport( 'joomla.application.component.view' );

class LoginViewReset extends JView
{
    function display($tpl = null)
    {
        parent::display($tpl);
    }
}

```

com_login\views\reset\metadata.xml

```

<?xml version="1.0" encoding="utf-8"?>
<metadata>
  <view title="Reset">
    <message><![CDATA[TYPERESETDESC]]></message>
  </view>
</metadata>

```

com_login\views\reset\tmpl\default.php

```

<?php defined( '_JEXEC' ) or die; ?>

<div class="componentheading">
  <?php echo JText::_('Enter Username'); ?>
</div>

<form action="<?php echo JRoute::_('index.php?option=com_login&task=requestreset'
); ?>" method="post" class="">
  <table cellpadding="0" cellspacing="0" border="0" width="100%"
    class="contentpane">
    <tr>

```

```

        <td colspan="2" height="40">
            <p><?php echo JText::_('Enter the Username for your
            account and a token will be e-mailed to you. '); ?></p>
        </td>
    </tr>
    <tr>
        <td width="110px" height="40">
            Enter User Name:
        </td>
        <td>
            <input id="userName" name="userName" type="text"
            class="" />
        </td>
    </tr>
</table>

<button type="submit" class="validate"><?php echo JText::_('Submit');
?></button>
<?php echo JHTML::_('form.token'); ?>
</form>

```

com_login\views\reset\templ\default.xml

```

<?xml version="1.0" encoding="utf-8"?>
<metadata>
    <layout title="DEFAULT RESET LAYOUT">
        <message>
            <![CDATA[DEFAULT RESET LAYOUT DESC]]>
        </message>
    </layout>
    <state>
        <name>DEFAULT RESET LAYOUT</name>
        <description>DEFAULT RESET LAYOUT DESC</description>
    </state>
</metadata>

```

com_login\views\reset\templ\confirm.php

```

<?php defined('_JEXEC') or die; ?>

<div class="componentheading">
    <?php echo JText::_('Confirm your Account'); ?>
</div>

<form action="<?php echo JRoute::_('index.php?option=com_login&task=confirmreset'
); ?>" method="post" class="">
    <table cellpadding="0" cellspacing="0" border="0" width="100%"
    class="contentpane">
        <tr>
            <td colspan="2" height="40">
                <p><?php echo JText::_('Copy the Token that has been
                sent to you into the field under here'); ?></p>
            </td>
        </tr>
        <tr>
            <td height="40">
                <label for="token" class="hasTip" title="<?php echo
                JText::_('RESET_PASSWORD_TOKEN_TIP_TITLE'); ?>:<?php echo
                JText::_('RESET_PASSWORD_TOKEN_TIP_TEXT'); ?>"><?php echo
                JText::_('Token'); ?></label>
            </td>
            <td>
                <input id="token" name="token" type="text"
                class="required" size="36" />
            </td>
        </tr>
    </table>

```

```

        <button type="submit" class="validate"><?php echo JText::_('Submit'); ?></button>
        <?php echo JHTML::_('form.token' ); ?>
</form>

```

com_login\views\reset\tmpl\confirm.xml

```

<?xml version="1.0" encoding="utf-8"?>
<metadata>
    <layout hidden="true" />
</metadata>

```

com_login\views\reset\tmpl\complete.php

```

<?php defined('_JEXEC') or die( 'Restricted Access' );
$id = JRequest::getInt('id');

?>
<div class="componentheading">
    <?php echo JText::_('Reset your Password'); ?>
</div>

<form action="<?php echo JRoute::_('index.php?option=com_login&task=completereset' );
?>" method="post" class="josForm form-validate">
    <table cellpadding="0" cellspacing="0" border="0" width="100%"
        class="contentpane">
        <tr>
            <td colspan="2" height="40">
                <p><?php echo JText::_('Complete the Password Reset.
                Enter your new Password'); ?></p>
            </td>
        </tr>
        <tr>
            <td height="40">
                <label for="password1" class="hasTip" title="<?php echo
                JText::_('RESET_PASSWORD_PASSWORD1_TIP_TITLE');?>:<?php echo
                JText::_('RESET_PASSWORD_PASSWORD1_TIP_TEXT'); ?>"><?php echo
                JText::_('Password'); ?></label>
            </td>
            <td>
                <input id="password1" name="password1" type="password"
                class="required validate-password" />
            </td>
        </tr>
        <tr>
            <td height="40">
                <label for="password2" class="hasTip" title="<?php echo
                JText::_('RESET_PASSWORD_PASSWORD2_TIP_TITLE');?>:<?php echo
                JText::_('RESET_PASSWORD_PASSWORD2_TIP_TEXT'); ?>"><?php echo
                JText::_('Verify Password'); ?></label>
            </td>
            <td>
                <input id="password2" name="password2" type="password"
                class="required validate-password" />
            </td>
        </tr>
    </table>
    <button type="submit" class="validate"><?php echo JText::_('Submit');
    ?></button>
    <input type="hidden" name="id" value="<?php echo $id; ?>" />
    <?php echo JHTML::_('form.token' ); ?>
</form>

```

com_login\views\reset\tmpl\complete.xml

```

<?xml version="1.0" encoding="utf-8"?>
<metadata>

```

```
<layout hidden="true" />
</metadata>
```

Source Code com_registrations

com_registrations\controller.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-09 06:46:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

// Make sure that the class can only be invoked inside Joomla
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Make the Joomla framework core controller code available
jimport( 'joomla.application.component.controller' );

class RegistrationsController extends JController
{
    function logout()
    {
        $session =& JFactory::getSession();
        $session->destroy();

        $this->setRedirect('index.php?option=com_flyer');
        return;
    }
    function info()
    {
        JRequest::setVar('view', 'personalinfo');
        parent::display();
    }
    function save()
    {
        JTable::addIncludePath(JPATH_ADMINISTRATOR.
            DS.'components'.DS.'com_manualreg'.DS.'tables');
        $row =& JTable::getInstance('person_reg_event', 'Table');
        $row->eventID = JRequest::getInt('eventID');
        $row->personID = JRequest::getInt('personID');
        $row->raceDayAge = JRequest::getInt('age');
        $row->agegroup = JRequest::getVar('agegr');
        if(!$row->store())
        {
            JError::raiseError(500, $row->getError());
        }
        $id = $row->id;
        $eventID = JRequest::getInt('eventID');
        $fee = JRequest::getVar('total');
        $personID = JRequest::getVar('personID');
        $this->setRedirect('index.php?option=com_pay&eventID='
            . $eventID . '&id=' . $id . '&fee=' . $fee . '&nid=' .
            $personID);
    }
    function display()
    {
        $session =& JFactory::getSession();
        $eventID = $session->get('eventID', '');
        $username = $session->get('username', '');
        $model = &$this->getModel('signup');
        $code = JRequest::getInt('code', '');
    }
}
```



```

        if($code == -1)
        {
            JRequest::setVar('view', 'personalinfo');
            parent::display();
            return;
        }

        //$eventID = JRequest::getInt('eventID');
        if ($eventID == '' || $eventID == 0)
        {
            //JRequest::setVar('view', 'myinfo');
            $msg = 'Session Expired. holy shit !! Select an Event';
            $this->setRedirect('index.php?option=com_flyer', $msg);
            return;
        }
        else if(!$username)
        {
            $msg = 'you have to log in again!';
            $link = 'index.php?option=com_login&eventID='.$eventID;
            $this->setRedirect($link, $msg, 'error');
            return;
        }
        else if ($model->checksignup($eventID, $username) === true)
        {
            //$message = JText::sprintf('You have already signed up for this
            event');
            JRequest::setVar('view', 'myinfo');
            //$this->setRedirect('index.php?option=com_registrations&
            view=signup&layout=regfinish', $message);
            // return ;
        }
        else if ($model->checksignup($eventID, $username) === false)
        {
            $view = JRequest::getVar('view');
            if(!$view)
            {
                //JError::raiseError(500, $username);
                JRequest::setVar('view', 'signup');
            }
            else
            {
                JRequest::setVar('view', 'checkout');
            }
        }
        parent::display();
    }
}

```

com_registrations\registrations.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-07-09 06:40:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

require_once( JPATH_COMPONENT_DS.'controller.php' );

// Create a new instance of the controller
$controller = new RegistrationsController();

// Execute the task variable
$controller->execute($task);

```

```
// Redirect after the task variable has been executed
$controller->redirect();
```

com_registrations\models\checkout.php

```
<?php
```

```
jimport('joomla.application.component.model');
class RegistrationsModelCheckout extends JModel
{
    var $_eventdata = null;
    var $_persondata = null;
    var $_totalprice = null;
    var $_pricing = null;
    var $_organization = null;

    function &geteventdata()
    {
        if (empty( $this->_eventdata ))
        {
            $eventID = JRequest::getVar('eventID');
            $sql1 = "SELECT * FROM #__events WHERE eventID='$eventID'";
            $db1 =& $this->getDBO();
            $db1->setQuery($sql1);
            $this->_eventdata = $db1->loadObject();
            return $this->_eventdata;
        }
    }

    function &getpersondata()
    {
        if (empty( $this->_persondata ))
        {
            $personID = JRequest::getVar('personID');
            $sql2 = "SELECT * FROM #__persons WHERE personID='$personID'";
            $db2 =& $this->getDBO();
            $db2->setQuery($sql2);
            $this->_persondata = $db2->loadObject();
            return $this->_persondata;
        }
    }

    function getpricing()
    {
        if (empty( $this->_pricing))
        {
            $eventID = JRequest::getVar('eventID');
            $today = & JFactory::getDate();
            $myFormat = $today->toFormat('%Y-%m-%d');
            $sql3= "SELECT min(fee) FROM #__fees WHERE eventID='$eventID'
                and dueDate >= '$myFormat'";
            $db3 =& $this->getDBO();
            $db3->setQuery($sql3);
            $price = $db3->loadResult();
            $this->_pricing = $price;
            return $this->_pricing;
        }
    }

    function gettotalprice()
    {
        if (empty( $this->_totalprice))
        {
            $eventID = JRequest::getVar('eventID');
            $racefee = JRequest::getVar('fee');
            $tsT = JRequest::getVar('tees');

            $sql5 = "SELECT tPrice FROM #__define_fees WHERE eventID='$eventID'";
            $db5 =& $this->getDBO();
        }
    }
}
```

```

$db5->setQuery($sql5);
$tshirtprice = 0;
if($isT == 1)
{
    $tshirtprice = $db5->loadResult();
}

$deductable_donation = JRequest::getInt('deductfee', 0);

$surcharge = 2;

$total = $racefee + $tshirtprice + $deductable_donation + $surcharge;
$this->_totalprice = $total;
return $this->_totalprice;
}
}

function getorganization()
{
    if (empty( $this->_organization))
    {
        $eventID = JRequest::getVar('eventID');
        $sql6 = "SELECT orgID FROM #__events WHERE eventID='$eventID' ";
        $db6 =& $this->getDBO();
        $db6->setQuery($sql6);
        $orgID = $db6->loadResult();

        $sql7 = "SELECT * FROM #__organization WHERE orgID='$orgID'";
        $db7 =& $this->getDBO();
        $db7->setQuery($sql7);
        $this->_organization = $db7->loadObject();
        return $this->_organization;
    }
}
}
}

```

com_registrations\models\myinfo.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2010-01-11 09:10:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport('joomla.application.component.model');

class RegistrationsModelMyinfo extends JModel
{
    var $info = null;
    function getinfo($eventID)
    {
        $db =& $this->getDBO();
        $sql = "SELECT eventName FROM #__events WHERE eventID = '$eventID'";
        $db->setQuery($sql);
        $this->info = $db->loadResult();
        return $this->info;
    }
}
}

```

com_registrations\models\personalinfo.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2010-01-11 14:04:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 * This is the model for all the records that has been entered into the database
 */

// Make sure that that the class can only be invoked within Joomla! 1.5
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Import the joomla! 1.5 framework model functions
jimport( 'joomla.application.component.model' );

class RegistrationsModelPersonalinfo extends JModel
{
    var $personID = null;
    var $signups = null;

    function getpersonid($username)
    {
        $db = & $this->getDBO();
        $sql = "SELECT personID from #__persons WHERE userName = '$username'";
        $db->setQuery($sql);
        $this->personID = $db->loadResult();
        return $this->personID;
    }

    function getsignups($eventID, $personID)
    {
        $db = & $this->getDBO();
        $sql = "SELECT * FROM #__events AS a, #__persons AS b,
                #__person_reg_event AS c
                WHERE a.eventID = c.eventID AND b.personID = c.personID
                AND b.personID = '$personID' ";
        //$db->setQuery($sql);
        $this->signups = $this->_getList($sql);
        return $this->signups;
    }
}
}
```

com_registrations\models\signup.php

```
<?php
/**
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

jimport( 'joomla.application.component.model' );
class RegistrationsModelSignup extends JModel
{
    var $_events = null;
    var $_persons = null;
    var $_agegroup = null;
    var $_price = null;
    var $_tshirtprice = null;
    var $_raceDayAge = null;

    function getraceDayage($eventID, $username)
    {
        if (empty( $this->_raceDayAge ))
        {
            //$eventID = JRequest::getInt('eventID');
            //$myTest = JRequest::getVar('usrName');
        }
    }
}
```

```

    $sql1 = "SELECT event_Date FROM #__events WHERE eventID='$eventID'";
    $db1 =& $this->getDBO();
    $db1->setQuery($sql1);
    $rday = $db1->loadResult();
    $sql2 = "SELECT dob from #__Persons WHERE userName='$username'";
    $db2 =& $this->getDBO();
    $db2->setQuery($sql2);
    $bday = $db1->loadResult();
    $this->_raceDayAge = $this->birthday($bday, $rday);
    return $this->_raceDayAge;
}
}

function getagegroup($eventID, $username)
{
    if (empty( $this->_agegroup ))
    {
        //$session =& JFactory::getSession();
        //$age = $session->get('raceDayAge', '');
        //$age = JRequest::getInt('raceDayAge');
        //$eventID = JRequest::getInt('eventID');
        //$myTest = JRequest::getVar('usrName');
        //$myTest = 'jo60nor';
        $sql1 = "SELECT ageEvent, event_Date FROM #__events WHERE
eventID='$eventID'";
        $sql2 = "SELECT dob, gender from #__Persons WHERE
userName='$username'";

        $db =& $this->getDBO();
        $db->setQuery($sql2);
        //Getting DOB from person table
        $result = $db->loadRow();
        $bday = $result[0];
        $gender = $result[1];
        $gend = null;
        if($gender == 'male')
        {
            $gend = 'M';
        }
        else if($gender == 'female')
        {
            $gend = 'F';
        }
        $db1 =& $this->getDBO();
        $db1->setQuery($sql1);
        $result1 = $db1->loadRow();
        $ageEvent = $result1[0];
        //Getting Race day date from events table
        $rday = $result1[1];
        //Calculate Race Day Age based on Race Day and DOB

        $age = $this->birthday($bday, $rday);
        //$this->_raceDayAge = $age;
        // return $this->_raceDayAge;
        $db2 =& $this->getDBO();
        $sql3 = "SELECT lowAge, upperAge, span FROM #__agegroups
WHERE ageID='$ageEvent'";
        $db2->setQuery($sql3);
        $data = $db2->loadRow();
        $low = $data['0'];
        $high = $data['1'];
        $span = $data['2'];
        //$this->_agegroup = 44;
        //return $this->_agegroup;
        if ($age < $low)
        {
            $strng1 = "0 - ";
            $strng2 = $low - 1;
            $this->_agegroup = $gend.$strng1.$strng2;
            return $this->_agegroup;
            break;
        }
    }
}

```

```

    }
    if ($age >= $high)
    {
        $strng3 = $high;
        $strng4 = " - 100+";
        $this->_agegroup = $gend.$strng3.$strng4;
        return $this->_agegroup;
        break;
    }
    if ($age > $low)
    {
        for($i = $low; $i <= $high; $i += $span - 1)
        {
            if($i < $age)
            {
                $i++;
            }
            else if($i == $age)
            {
                $strng5 = $i;
                $strng6 = " - ";
                $strng7 = $i + $span-1;
                $this->_agegroup = $gend.$strng5.$strng6.$strng7;
                return $this->_agegroup;
                break;
            }
            else
            {
                $strng8 = $i-$span;
                $strng9 = " - ";
                $strng10 = $i - 1;
                $this->_agegroup = $gend.$strng8.$strng9.$strng10;
                return $this->_agegroup;
                echo '<br />';
                break;
            }
        }
    }
}
}
}
}

function getpersons($username)
{
    if (empty( $this->_persons ))
    {
        //JError::raiseError(500, $username);
        //$userName = JRequest::getVar('usrName');
        $sql4 = "SELECT * FROM #__persons WHERE userName='$username'";
        $db3 =& $this->getDBO();
        $db3->setQuery($sql4);
        $this->_persons = $db3->loadObject();
    }
    return $this->_persons;
}

function getevents($eventID)
{
    if (empty( $this->_events ))
    {
        //JError::raiseError(500, $eventID);
        //$eventID = JRequest::getVar('eventID');
        $sql5 = "SELECT * FROM #__events WHERE eventID='$eventID'";
        $db4 =& $this->getDBO();
        $db4->setQuery($sql5);
        $this->_events = $db4->loadObject();
        return $this->_events;
    }
}
}
}
}

```

```

function &getprice($eventID)
{
    if (empty( $this->_price))
    {
        //$eventID = JRequest::getVar('eventID');
        $today = & JFactory::getDate();
        $myFormat = $today->toFormat('%Y-%m-%d');
        $sql6= "SELECT min(fee) FROM #__fees WHERE eventID='$eventID'
                and dueDate >= '$myFormat'";
        $db5 =& $this->getDBO();
        $db5->setQuery($sql6);
        $price = $db5->loadResult();
        $this->_price = $price;
        return $this->_price;
    }
}

function &gettshirtprice($eventID)
{
    if (empty( $this->_tshirtprice ))
    {
        //$eventID = JRequest::getVar('eventID');
        $sql7 = "SELECT isTInvolved, tPrice FROM #__define_fees
                WHERE eventID='$eventID'";
        $db6 =& $this->getDBO();
        $db6->setQuery($sql7);
        $this->_tshirtprice = $db6->loadObject();
    }
    return $this->_tshirtprice;
}

function birthday ($bday, $rday)
{
    list($year1,$month1,$day1) = explode("-", $bday);
    list($year2,$month2,$day2) = explode("-", $rday);
    $year_diff = $year2 - $year1;
    $month_diff = $month2 - $month1;
    $day_diff = $day2 - $day1;
    if ($month_diff < 0) $year_diff--;
    elseif (($month_diff==0) && ($day_diff < 0)) $year_diff--;
    return $year_diff;
}

function checksignup($eventID, $username)
{
    $db8 =& $this->getDBO();
    $sql8 = "SELECT personID FROM #__persons where userName = '$username'";
    $db8->setQuery($sql8);
    $personID = $db8->loadResult();
    $sql9 = "SELECT id FROM #__person_reg_event WHERE eventID = '$eventID'
            AND personID = '$personID'";
    $db8->setQuery($sql9);
    if (!$id = $db8->loadResult())
    {
        return false;
    }
    return true;
}
}

```

com_registrations\views\checkout\view.html.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
jimport( 'joomla.application.component.view' );
class RegistrationsViewCheckout extends JView
{

```

```

function display($tpl = null)
{
    $person =& $this->get('persondata');
    $events =& $this->get('eventdata');
    $total_price =& $this->get('totalprice');
    $reg_fee =& $this->get('pricing');
    $organization =& $this->get('organization');
    // $raceFee =& $this->get('pricing');
    $agegroup = JRequest::getVar('agegr');
    $age = JRequest::getVar('age');

    $deductfee = JRequest::getInt('deductfee', 0);
    $isTextra = JRequest::getInt('tees');
    $tPrice = JRequest::getVar('tPrice');
    // $tPrice = 'check this';
    $registrationfee = JRequest::getInt('fee');

    $this->assignRef('person', $person);
    $this->assignRef('events', $events);
    $this->assignRef('agegroup', $agegroup);
    $this->assignRef('age', $age);
    $this->assignRef('genders', $genders);
    $this->assignRef('deductfee', $deductfee);
    $this->assignRef('isTextra', $isTextra);
    $this->assignRef('tPrice', $tPrice);
    $this->assignRef('registrationfee', $registrationfee);
    $this->assignRef('total_price', $total_price);
    $this->assignRef('organization', $organization);
    $this->assignRef('reg_fee', $reg_fee);

    parent::display($tpl);
}
}
?>

```

com_registrations\views\checkout\tmpl\default.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
$link = JRoute::_('index.php?option=com_registrations&task=logout');
echo '<p align="right">' . '<a href="' . $link . '" >' . '<strong>' .
    'logout' . '</strong>' . '</a>' . '</p>' ;

?>

<form action="index.php?option=com_registrations&task=save"
    method="post" id="josForm" name="josForm" style="font-size: 8pt;">
<fieldset>
    <legend>Race Organization</legend>

    <table>
        <tr>
            <td>
                <?php echo $this->organization->orgLogo ; ?>
            </td>
            <td>
                <?php
                echo $this->organization->orgName . '<br />' ;
                echo $this->organization->orgAddress . '<br />' ;
                echo $this->organization->orgCity . ' ' ;
                echo $this->organization->orgState . ' ' ;
                echo $this->organization->orgZip . '<br />' ;
                echo $this->organization->orgPhone . '<br />' ;
                ?>
            </td>
        </tr>
    </table>

```



```

    </table>
</fieldset>

<fieldset>
  <legend>Personal Information</legend>
  <table>
    <tr>
      <td width="120px" height="20">
        Name:
      </td>
      <td>
        <?php
          echo $this->person->firstName . ' ' ;
          echo $this->person->middleName . ' ' ;
          echo $this->person->lastName;
        ?>
      </td>
    </tr>
    <tr>
      <td width="120px" height="20">
        Address:
      </td>
      <td>
        <?php
          echo $this->person->address ;
        ?>
      </td>
    </tr>
    <tr>
      <td width="120px" height="20">
      </td>
      <td>
        <?php
          echo $this->person->city . ' ' ;
          echo $this->person->state . ' ' ;
          echo $this->person->zip;
        ?>
      </td>
    </tr>
    <tr>
      <td width="120px" height="20">
        Race Day Age:
      </td>
      <td>
        <?php
          echo $this->age ;
        ?>
      </td>
    </tr>
    <tr>
      <td width="120px" height="20">
        Gender:
      </td>
      <td>
        <?php
          echo $this->person->gender ;
        ?>
      </td>
    </tr>
    <tr>
      <td width="120px" height="20">
        Age Group:
      </td>
      <td>
        <?php
          echo $this->agegroup ;
        ?>
      </td>
    </tr>
  </table>
</fieldset>

```

```

</table>
</fieldset>

<fieldset>
  <legend>Registration Fees</legend>
  <table>
    <tr>
      <td width="120px" height="20">
        Registraration Fee:
      </td>
      <td>
        $<?php echo $this->reg_fee; ?>.00
      </td>
    </tr>
    <?php
      if($this->isTextra == 1)
      {
        echo '<tr>';
        echo '<td>';
        echo 'Extra T-Shirt: ';
        echo '</td>';
        echo '<td>';
        echo '$'. $this->tPrice . '.00';
        echo '</td>';
        echo '</tr>';
      }
    >
    <tr>
      <td>
        Deductable Donation:
      </td>
      <td>
        $<?php echo $this->deductfee; ?>.00
      </td>
    </tr>
    <tr>
      <td>
        Process Fee:
      </td>
      <td>
        $ 2.00
      </td>
    </tr>
    <tr>
      <td>
        <strong>Total</strong>
      </td>
      <td>
        <strong>$<?php echo $this->total_price; ?>.00</strong>
      </td>
    </tr>
  </table>
</fieldset>
<?php
  $eventID = $this->events->eventID;
  $agegr = $this->agegroup;
  $age = $this->age;
  $personID = $this->person->personID;
  $total = $this->total_price
  ?>
  <button class="button validate" type="submit"><?php echo JText::_('Check Out'); ?></button>
  <input type="hidden" name="task" value="save" />
  <input type="hidden" name="eventID" value="<?php echo $eventID; ?>" />
  <input type="hidden" name="personID" value="<?php echo $personID; ?>" />
  <input type="hidden" name="agegr" value="<?php echo $agegr; ?>" />
  <input type="hidden" name="age" value="<?php echo $age; ?>" />
  <input type="hidden" name="total" value="<?php echo $total; ?>" />
  <input type="hidden" name="gid" value="0" />
  <?php echo JText::_('form.token'); ?>
</form>

```

com_registrations\views\myinfo\view.html.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-10-18 07:50:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the interface for the single review of events. It extends JView
 * from Joomla! 1.5 framework.
 */
defined( '_JEXEC' ) or die( 'Restricted Access' );
jimport( 'joomla.application.component.view' );

class RegistrationsViewMyinfo extends JView
{
    function display($tpl = null)
    {
        $session =& JFactory::getSession();
        $eventID = $session->get('eventID', '');
        $model = &$this->getModel('myinfo');
        $message = $model->getInfo($eventID);
        $this->assignRef('message', $message);
        parent::display($tpl);
    }
}
?>
```

com_registrations\views\myinfo\tmpl\default.php

```
<?php
/**
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

$link = JRoute::_('index.php?option=com_registrations&task=logout');
echo '<p align="right">' . '<a href="' . $link . '" >' . '<strong>' . 'logout' . '</strong>'
. '</a>' . '</p>';

echo 'you have already signed up for ' . $this->message . ' ';
$link = JRoute::_('index.php?option=com_registrations&task=info');
echo '<a href="' . $link . '" >' . '<strong>' . 'Click here' . '</strong>' . '</a>';
echo ' to move on';

?>
```

com_registrations\views\personalinfo\view.html.php

```
<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-10-18 07:50:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the interface for the single review of events. It extends JView
 * from Joomla! 1.5 framework.
 */
defined( '_JEXEC' ) or die( 'Restricted Access' );
jimport( 'joomla.application.component.view' );
JTable::addIncludePath(JPATH_ADMINISTRATOR.
    DS.'components'.DS.'com_registrations'.DS.'tables');

```

```

class RegistrationsViewPersonalinfo extends JView
{
    function display($tpl = null)
    {
        $session =& JFactory::getSession();
        $username = $session->get('username', '');
        $eventID = $session->get('eventID', '');
        $model = &$this->getModel('personalinfo');
        $personID = $model->getpersonid($username);
        $persons =& JTable::getInstance('persons', 'table');
        $persons->load($personID);
        $rows = $model->getsignups($eventID, $personID);
        $this->assignRef('persons', $persons);
        $this->assignRef('rows', $rows);

        parent::display($tpl);
    }
}

```

com_registrations\views\personalinfo\tmpl\default.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
$link = JRoute::_('index.php?option=com_registrations&task=logout');
echo '<p align="right">' . '<a href="' . $link . '" >' . '<strong>' . 'logout' . '</strong>'
. '</a>' . '</p>' ;
?>
<fieldset>
    <legend>Personal Information</legend>

<table cellpadding="0" cellspacing="0" border="0">
    <tr>
        <td width="100px">
            <h4> Name:</h4>
        </td>
        <td>
            <h4>
                <?php echo $this->persons->firstName . ' ' ; ?>
                <?php echo $this->persons->middleName . ' ' ; ?>
                <?php echo $this->persons->lastName; ?><br>
                <?php echo $this->persons->address; ?><br>
                <?php echo $this->persons->city . ' ' ; ?>
                <?php echo $this->persons->state . ' ' ; ?>
                <?php echo $this->persons->zip . ' ' ; ?>
            </h4>
        </td>
    </tr>
</table>
</fieldset>
<fieldset>
    <legend>Sign-Up Information</legend>
    <table class="adminlist">
        <thead>
            <tr>
                <th width="10%" align="center" bgcolor="#e3e3e6" >Event Date</th>
                <th width="10%" align="center" bgcolor="#e3e3e6">Event Name</th>
                <th width="10%" align="center" bgcolor="#e3e3e6">Event Place</th>
            </tr>
        </thead>
        <?php
        /*$k = 0;
        for($i=0, $n=count($this->rows); $i < $n; $i++)
            $row =&$this->rows[$i];*/
        foreach ($this->rows as $row)
        {
            ?>
        <tr>

```

```

        <td><?php echo JHTML::date($row->event_Date); ?></td>
        <td><?php echo $row->eventName; ?></td>
        <td><?php echo $row->eventPlace; ?></td>
    </tr>
    <?php
    }
    ?>
</table>
</fieldset>

```

com_registrations\views\signup\view.html.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
defined( '_JEXEC' ) or die( 'Restricted Access' );
jimport( 'joomla.application.component.view' );

JTable::addIncludePath( JPATH_ADMINISTRATOR .
    DS . 'components' . DS . 'com_flyer' . DS . 'tables' );
JTable::addIncludePath( JPATH_ADMINISTRATOR .
    DS . 'components' . DS . 'com_registrations' . DS . 'tables' );
class RegistrationsViewSignup extends JView
{
    function display($tpl = null)
    {
        $session =& JFactory::getSession();
        $username = $session->get('username', '');
        $model = &$this->getModel('signup');
        $person = $model->getpersons($username);
        $eventID = $session->get('eventID', '');
        $events = $model->getevents($eventID);
        $fee = $model->getprice($eventID);
        $raceDage = $model->getracedayage($eventID, $username);
        $agegroup = $model->getagegroup($eventID, $username);
        $tshirt = $model->gettshirtprice($eventID);
        $date = JHTML::date($events->event_Date);
        $tees = null;
        $this->assignRef('date', $date);
        $this->assignRef('raceDage', $raceDage);
        $this->assignRef('person', $person);
        $this->assignRef('events', $events);
        $this->assignRef('agegroup', $agegroup);
        $this->assignRef('ID', $ID);
        $this->assignRef('genders', $genders);
        $this->assignRef('fee', $fee);
        $this->assignRef('tshirt', $tshirt);
        $this->assignRef('tees', JHTML::_('select.booleanlist',
            'tees', 'class="inputbox"', $tees));

        parent::display($tpl);
    }
}

```

com_registrations\views\signup\tmpl\default.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
$link = JRoute::_('index.php?option=com_registrations&task=logout');
echo '<p align="right">' . '<a href="' . $link . '" >' . '<strong>' . 'logout' . '</strong>'
. '</a>' . '</p>';
?>

```

```

<form action="<?php echo JRoute::_( 'index.php?option=com_registrations' ); ?>"
  method="post" id="josForm" name="josForm" style="font-size: 8pt;">
<fieldset>
  <legend>Race registration</legend>
  <table cellpadding="1" cellspacing="3" border="0" width="100%" class="contentpane">
    <tr>
      <td colspan="2">
        <?php echo $this->events->eventLogo; ?>
      </td>
    </tr>
    <tr>
      <td width="20%" height="20">
        Race Name:
      </td>
      <td>
        <?php echo $this->events->eventName;?>
      </td>
    </tr>
    <tr>
      <td width="20%" height="20">
        Event Date:
      </td>
      <td>
        <?php echo $this->date;?>
      </td>
    </tr>
    <tr>
      <td width="20%" height="20">
        Event Place:
      </td>
      <td>
        <?php echo $this->events->eventPlace;?>
      </td>
    </tr>
    <tr>
      <td width="20%" height="20">
        Name:
      </td>
      <td>
        <?php echo $this->person->firstName;?>
        <?php echo $this->person->middleName;?>
        <?php echo $this->person->lastName;?>
      </td>
    </tr>
    <tr>
      <td width="20%" height="20">
        Address:
      </td>
      <td>
        <?php echo $this->person->address;?>
      </td>
    </tr>
    <tr>
      <td width="20%" height="20">
      </td>
      <td>
        <?php echo $this->person->city;?>
        <?php echo $this->person->state;?>
        <?php echo $this->person->zip;?>
      </td>
    </tr>
    <tr>
      <td width="20%" height="20">
        Age on Race Day:
      </td>
      <td>
      </td>
    </tr>
  </table>

```

```

        <?php echo $this->raceDAGE;?>
    </td>
</tr>

<tr>
    <td width="20%" height="20">
        Age Group:
    </td>
    <td>
        <?php echo $this->agegroup;?>
    </td>
</tr>

<tr>
    <td width="20%" height="20">
        Entry Fee:
    </td>
    <td>
        $<?php echo $this->fee;?>.00
    </td>
</tr>

<tr>
    <td width="20%" height="20">
        Deductable Donation:
    </td>
    <td>
        $<input type="text" id="deductfee" name="deductfee" size="4"
        value="" maxlength="4" style="font-size: 8pt" />.00
    </td>
</tr>
<tr>
    <td>
    <?php
        $shirt = $this->tshirt->isTInvolved;
        if($shirt == 0)
        {
            echo 'Add T-Shirt ' . '$' . $this->tshirt->tPrice . '.00';
        }
    ?>
    </td>
    <td>
        <?php
            $shirt = $this->tshirt->isTInvolved;
            if($shirt == 0)
            {
                echo $this->tees;
            }
        ?>
    </td>
</tr>
<tr>
    <td>
        <a href="javascript:history.go(-1)"> Go Back</a>
    </td>
</tr>
</table>
<?php
$eventID = $this->events->eventID;
$agegr = $this->agegroup;
$sage = $this->raceDAGE;
$personID = $this->person->personID;
$tPrice = $this->tshirt->tPrice;
$racefee = $this->fee;
$checkout = -1;

?>
<button class="button validate" type="submit"><?php echo JText::_('Sign Up'); ?></button>

<input type="hidden" name="task" value="" />
<input type="hidden" name="eventID" value="<?php echo $eventID; ?>" />

```

```



```

Source Code com_pay

com_pay\controller.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-03 07:33:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

// Make sure that the class can only be invokes inside joomla
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Make the joomla framework core controller code available
jimport( 'joomla.application.component.controller' );

class PayController extends JController
{
    function pays()
    {
        $name = JRequest::getVar('name', '', 'post', 'string');
        $eventID = JRequest::getInt('eventID');
        $payID = JRequest::getInt('payID');
        $db = JFactory::getDBO();
        $update = new stdClass;
        $update->id = $payID;
        $update->has_paid = 1;
        $db->updateObject('#__person_reg_event', $update, 'id' );
        if(!$db)
        {
            JError::raiseError(500, 'No update happened');
        }

        $this->setRedirect('index.php?option=com_pay&view=accept&name=' . $name
            . '&eventID=' . $eventID);

        parent::display();
    }
    function display()
    {
        $session =& JFactory::getSession();
        // $eventID = JRequest::getInt('eventID');
        $eventID = $session->get('eventID', '');
        //JError::raiseError(500, $eventID);
        if ($eventID == '' || $eventID == 0)
        {
            $msg = 'you have to log in again 1';
            $link = 'index.php?option=com_login&eventID=' . $eventID;
            $this->setRedirect($link, $msg, 'error');
        }
        else
        {
            $test = $session->get('username', '');

```



```

    $myTest = JRequest::setVar('test');
    if($test == '')
    {
        $msg = 'you have to log in again 2';
        $link = 'index.php?option=com_login&eventID='. $seventID;
        $this->setRedirect($link, $msg, 'error');
    }
    // get the view
    $view = JRequest::getVar('view');

    // If the view is not set, set it to list
    if (!$view)
    {
        JRequest::setVar('view', 'payment');
    }
}
parent::display();
}
}

```

com_pay\pay.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-03 07:31:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

require_once( JPATH_COMPONENT_DS.'controller.php' );

// Create a new instance of the controller
$controller = new PayController();

// Execute the task variable
$controller->execute($task);

// Redirect after the task variable has been executed
$controller->redirect();

```

com_pay\views\accept\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-03 07:40:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the interface for the single review of events. It extends JView
 * from Joomla! 1.5 framework.
 */

defined( '_JEXEC' ) or die( 'Restricted access' );
jimport( 'joomla.application.component.view' );
JTable::addIncludePath(JPATH_ADMINISTRATOR.
    DS.'components'.DS.'com_flyer'.DS.'tables');

class PayViewAccept extends JView
{
    function display($tpl = null)
    {
        echo JRequest::getVar('nam');
        echo '<br /> Your Payment has been approved. An email has been sent to you';
    }
}

```

```

    parent::display($tpl);
}
}

```

com_pay\views\accept\tmpl\default.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
$link = JRoute::_('index.php?option=com_logout&task=logout');
?>
<p align="right"><a href=" <?php $link ?>" onclick="true" <strong> Log Out </strong> </a></p>

```

com_pay\views\payment\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-11-03 07:40:PST
 * @package     Joomla! 1.5
 * @subpackage  Events
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the interface for the single review of events. It extends JView
 * from Joomla! 1.5 framework.
 */

defined( '_JEXEC' ) or die( 'Restricted access' );
jimport( 'joomla.application.component.view' );
JTable::addIncludePath(JPATH_ADMINISTRATOR.
    DS.'components'.DS.'com_flyer'.DS.'tables');

class PayViewPayment extends JView
{
    function display($tpl = null)
    {
        $eventID = JRequest::getInt('eventID');
        $id = JRequest::getInt('id');
        $fee = JRequest::getInt('fee');
        $personID = JRequest::getInt('nid');
        parent::display($tpl);
    }
}

```

com_pay\views\payment\tmpl\default.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
$payID = JRequest::getInt('id');
$personID = JRequest::getInt('nid');
$eventID = JRequest::getInt('eventID');
setlocale(LC_MONETARY, 'en_US');
$fee = JRequest::getInt('fee');
$link = JRoute::_('index.php?option=com_logout&task=logout');
?>
<p align="right"><a href=" <?php $link ?>" onclick="true" <strong> Log Out </strong> </a></p>

<form action="index.php?option=com_pay&view=accept&task=pays"
    method="post" id="josForm" name="josForm" style="font-size: 8pt;">
    <table>
        <tr>
            <td>
                Name on Card:
            </td>
            <td>

```

```

                <input class="text_area" type="text" name="name"
                    id="name" size="30" maxlength="100"
                    value="" />
            </td>
        </tr>
        <tr>
            <td>
                Total:
            </td>
            <td>
                <?php echo "$" . number_format($fee, 2); ?>
            </td>
        </tr>
    </table>
    <button class="button validate" type="submit"><?php echo JText::_('Submit
        Payment'); ?></button>

    <input type="hidden" name="option" value="com_pay" />
    <input type="hidden" name="task" value="pays" />
    <input type="hidden" name="eventID" value="<?php echo $eventID; ?>" />
    <input type="hidden" name="payID" value="<?php echo $payID; ?>" />
    <input type="hidden" name="personID" value="<?php echo $personID; ?>" />
</form>

```

Source Code com_results

com_results\controller.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-12-24 08:40:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

// Make sure that the class can only be invoked inside Joomla
defined( '_JEXEC' ) or die ( 'Restricted Access' );

// Make the Joomla framework core controller code available
jimport( 'joomla.application.component.controller' );

class ResultsController extends JController
{
    function bibsearch()
    {
        JRequest::checkToken('request') or jexit( 'Invalid Token' );
        $bib = JRequest::getInt('bib', '', 'post', 'integer');
        $eventID = JRequest::getInt('eventID');
        if( (!$bib) || (!$eventID) )
        {
            $message = JText::sprintf('Something went wrong. Try again. ');
            $this->setRedirect('index.php?option=com_results&view=single&eventID=' .
                $eventID, $message);
            return;
        }
        //$model = &$this->getModel('showresults');
        //$model->getResult($bib);
        JRequest::setVar('view', 'showresults');
        parent::display();
    }

    function fnamesearch()
    {
        JRequest::checkToken('request') or jexit( 'Invalid Token' );
        $eventID = JRequest::getInt('eventID');
    }
}

```

```

    $fname = JRequest::getVar('firstName', '', 'post', 'string');
    $lname = JRequest::getVar('lastName', '', 'post', 'string');
    if(!($lname) && !($fname))
    {
        $message = JText::sprintf('Something went wrong. Try again. ');
        $this->setRedirect('index.php?option=com_results&view=single&eventID=' .
        $eventID, $message);
        return;
    }
    JRequest::setVar('view', 'showresults');
    parent::display();
}

function citysearch()
{
    JRequest::checkToken('request') or jexit('Invalid Token');
    $eventID = JRequest::getInt('eventID');
    $city = JRequest::getVar('city', '', 'post', 'string');
    if(!$city)
    {
        $message = JText::sprintf('Something went wrong. Try again. ');
        $this->setRedirect('index.php?option=com_results&view=single&eventID=' .
        $eventID, $message);
        return;
    }
    JRequest::setVar('view', 'showresults');
    parent::display();
}

function agegrsearch()
{
    JRequest::checkToken('request') or jexit('Invalid Token');
    $eventID = JRequest::getInt('eventID');
    $agegroup = JRequest::getInt('agegroup', '', 'post', 'integer');
    if(!$agegroup)
    {
        $message = JText::sprintf('Something went wrong. Try again. ');
        $this->setRedirect('index.php?option=com_results&view=single&eventID=' .
        $eventID, $message);
        return;
    }
    JRequest::setVar('view', 'showresults');
    parent::display();
}

function display()
{
    // get the view
    $view = JRequest::getVar('view');

    // If the view is not set, set it to list
    if (!$view)
    {
        JRequest::setVar('view', 'all');
    }

    parent::display();
}
}
?>

```

com_results/results.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-12-24 08:45:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

```

```

require_once( JPATH_COMPONENT.DS.'controller.php' );

// Create a new instance of the controller
$controller = new ResultsController();

// Execute the task variable
$controller->execute($task);

// Redirect after the task variable has been executed
$controller->redirect();

```

com_results\models\all.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-12-24 08:33:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

// Make sure that the class can only be invoked inside Joomla
defined( '_JEXEC' ) or die ( 'Restricted Control' );

// Import Joomla Application Model
jimport('joomla.application.component.model');

class ResultsModelAll extends JModel
{
    var $_data = null;
    function &getData()
    {
        if (empty($this->_data))
        {
            $query = "SELECT * FROM #__events WHERE published = '1'
                    ORDER BY event_Date";
            $this->_data = $this->getList($query);
        }
        return $this->_data;
    }
}

```

com_results\models\showresults.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2010-01-12 14:57:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

// Make sure that the class can only be invoked inside Joomla
defined( '_JEXEC' ) or die ( 'Restricted Control' );

// Import Joomla Application Model
jimport('joomla.application.component.model');

class ResultsModelShowresults extends JModel
{
    var $_result = null;
    var $_eventname = null;
    var $_sexnum = null;
    var $_totalnum = null;

```

```

var $_agenum = null;
var $n = 0;
var $n1 = 0;
var $n2 = 0;

function &getResult($eventID, $bib)
{
    if (empty($this->_result))
    {
        $sql3 = "SELECT * FROM #__persons AS a, #__person_reg_event AS b
                WHERE a.personID = b.personID AND b.eventID = '$eventID' AND
                b.bib = '$bib' AND b.clock_time > '00:00:00' ORDER BY clock_time";
        $this->_result = $this->_getList($sql3);
        return $this->_result;
    }
}

function &getEventname($eventID)
{
    if (empty($this->_eventname))
    {
        $sql = "SELECT * from #__events WHERE eventID = '$eventID'";
        //$this->_eventname = $this->_getList($sql);
        $db =& JFactory::getDBO();
        $db->setQuery($sql);
        $this->_eventname = $db->loadObject();
        return $this->_eventname;
    }
}

function &getTotalnum($eventID, $bib)
{
    if (empty($this->_totalnum))
    {
        $db1 =& JFactory::getDBO();

        $sql1 = "SELECT count(clock_time)
                FROM #__person_reg_event
                WHERE clock_time <= (SELECT clock_time FROM #__person_reg_event
                WHERE eventID = '$eventID' AND bib = '$bib')";

        $db1->setQuery($sql1);
        $this->_totalnum = $db1->loadResult();
        return $this->_totalnum;
    }
}

function &getSexnum($eventID, $bib)
{
    if (empty($this->_sexnum))
    {
        $sql2 = "SELECT count(LEFT (agegroup, 1)) FROM #__person_reg_event
                WHERE clock_time <=
                (SELECT clock_time FROM #__person_reg_event
                WHERE eventID = '$eventID' AND bib = '$bib')
                AND LEFT (agegroup, 1) =
                (SELECT LEFT (agegroup, 1) FROM #__person_reg_event
                WHERE eventID = '$eventID' AND bib = '$bib')";

        $db2 =& JFactory::getDBO();
        $db2->setQuery($sql2);
        $this->_sexnum = $db2->loadResult();
        return $this->_sexnum;
    }
}

function &getAgenum($eventID, $bib)
{
    if (empty($this->_agenum))
    {
        $sql3 = "SELECT count(agegroup) FROM #__person_reg_event

```

```

WHERE clock_time <=
    (SELECT clock_time FROM #__person_reg_event
     WHERE eventID = '$eventID' AND bib = '$bib')
AND agegroup =
    (SELECT agegroup FROM #__person_reg_event
     WHERE eventID = '$eventID' AND bib = '$bib')";
$db3 =& JFactory::getDBO();
$db3->setQuery($sql3);
$this->_agenum = $db3->loadResult();
return $this->_agenum;
}
}

function &getResfname($eventID, $fname, $lname)
{
    if (empty($this->_result))
    {
        if($fname && $lname)
        {
            $sql4 = "SELECT * FROM #__persons AS a, #__person_reg_event AS b
            WHERE a.personID = b.personID AND b.eventID = '$eventID' AND
            a.firstName = '$fname' AND a.lastName = '$lname'
            AND b.clock_time > '00:00:00' ORDER BY clock_time";
            $this->_result = $this->_getList($sql4);
            return $this->_result;
        }
        if($fname)
        {
            $sql5 = "SELECT * FROM #__persons AS a, #__person_reg_event AS b
            WHERE a.personID = b.personID AND b.eventID = '$eventID' AND
            a.firstName = '$fname'
            AND b.clock_time > '00:00:00' ORDER BY clock_time";
            $this->_result = $this->_getList($sql5);
            return $this->_result;
        }
        if($lname)
        {
            $sql6 = "SELECT * FROM #__persons AS a, #__person_reg_event AS b
            WHERE a.personID = b.personID AND b.eventID = '$eventID' AND
            a.lastName = '$lname'
            AND b.clock_time > '00:00:00' ORDER BY clock_time";
            $this->_result = $this->_getList($sql6);
            return $this->_result;
        }
    }
}

function &getTotalfnm($eventID, $fname, $lname)
{
    if (empty($this->_totalnum))
    {
        $db7 =& JFactory::getDBO();
        // $persID = null;
        if($fname && !$lname)
        {
            $sql9 = "SELECT personID FROM #__persons
            WHERE firstName = '$fname' ";
            $db7->setQuery($sql9);
            $persID = $db7->loadResultArray();
        }
        if($lname && !$fname)
        {
            $sql19 = "SELECT personID FROM #__persons
            WHERE lastName = '$lname' ";
            $db7->setQuery($sql19);
            $persID = $db7->loadResultArray();
        }
        else if($fname && $lname)
        {
            $sql29 = "SELECT personID FROM #__persons
            WHERE firstName = '$fname' AND lastName = '$lname' ";
            $db7->setQuery($sql29);
        }
    }
}

```

```

        $persID = $db7->loadResultArray();
    }
    $n1 = count($persID);
    for($i = 0; $i < $n1; $i++)
    {
        $sql10 = "SELECT personID from #__person_reg_event
            WHERE eventID = '$eventID' AND personID = '$persID[$i]';";
        $db7->setQuery($sql10);
        $persID2[$i] = $db7->loadResult();
    }
    $n2 = count($persID2);
    for($i = 0; $i < $n2; $i++)
    {
        $sql11 = "SELECT count(clock_time)
            FROM #__person_reg_event
            WHERE clock_time <= (SELECT clock_time FROM #__person_reg_event
            WHERE eventID = '$eventID' AND personID = '$persID2[$i]')";
        $db7->setQuery($sql11);
        $allpl[$i] = $db7->loadResult();
    }
    $this->_totalnum = $allpl;
    return $this->_totalnum;
}
}

function &getSexfnm($eventID, $fname, $lname)
{
    if(empty($this->_sexnum))
    {
        $db8 =& JFactory::getDBO();
        if($fname && !$lname)
        {
            $sql12 = "SELECT personID FROM #__persons
                WHERE firstName = '$fname' ";
            $db8->setQuery($sql12);
            $persID = $db8->loadResultArray();
        }
        if($lname && !$fname)
        {
            $sql13 = "SELECT personID FROM #__persons
                WHERE lastName = '$lname' ";
            $db8->setQuery($sql13);
            $persID = $db8->loadResultArray();
        }
        else if($fname && $lname)
        {
            $sql14 = "SELECT personID FROM #__persons
                WHERE firstName = '$fname' AND lastName = '$lname' ";
            $db8->setQuery($sql14);
            $persID = $db8->loadResultArray();
        }
        $n1 = count($persID);
        for($i = 0; $i < $n1; $i++)
        {
            $sql15 = "SELECT personID from #__person_reg_event
                WHERE eventID = '$eventID' AND personID = '$persID[$i]';";
            $db8->setQuery($sql15);
            $persID2[$i] = $db8->loadResult();
        }
        $n2 = count($persID2);
        for($i = 0; $i < $n2; $i++)
        {
            $sql6 = "SELECT count(LEFT (agegroup, 1)) FROM #__person_reg_event
                WHERE clock_time <=
                (SELECT clock_time FROM #__person_reg_event
                WHERE eventID = '$eventID' AND personID =
                '$persID2[$i]')
                AND LEFT (agegroup, 1) =
                (SELECT LEFT (agegroup, 1) FROM #__person_reg_event
                WHERE eventID = '$eventID' AND personID =
                '$persID2[$i]')";
            $db8->setQuery($sql6);
        }
    }
}

```



```

        $sexpl[$i] = $db8->loadResult();
    }
    $this->_sexnum = $sexpl;
    return $this->_sexnum;
}

function &getAgefnum($eventID, $fname, $lname)
{
    if(empty($this->_agenum))
    {
        $db9 =& JFactory::getDBO();
        if($fname && ! $lname)
        {
            $sql17 = "SELECT personID FROM #__persons
                WHERE firstName = '$fname' ";
            $db9->setQuery($sql17);
            $persID = $db9->loadResultArray();
        }
        if($lname && ! $fname)
        {
            $sql18 = "SELECT personID FROM #__persons
                WHERE lastName = '$lname' ";
            $db9->setQuery($sql18);
            $persID = $db9->loadResultArray();
        }
        else if($fname && $lname)
        {
            $sql20 = "SELECT personID FROM #__persons
                WHERE firstName = '$fname' AND lastName = '$lname' ";
            $db9->setQuery($sql20);
            $persID = $db9->loadResultArray();
        }
        $n1 = count($persID);
        for($i = 0; $i < $n1; $i++)
        {
            $sql21 = "SELECT personID from #__person_reg_event
                WHERE eventID = '$eventID' AND personID = '$persID[$i]'";
            $db9->setQuery($sql21);
            $persID2[$i] = $db9->loadResult();
        }
        $n2 = count($persID2);
        for($i = 0; $i < $n2; $i++)
        {
            $sql22 = "SELECT count(agegroup) FROM #__person_reg_event
                WHERE clock_time <=
                (SELECT clock_time FROM #__person_reg_event
                WHERE eventID = '$eventID' AND personID = '$persID2[$i]')
                AND agegroup =
                (SELECT agegroup FROM #__person_reg_event
                WHERE eventID = '$eventID' AND personID = '$persID2[$i]')";
            $db9->setQuery($sql22);
            $ageP1[$i] = $db9->loadResult();
        }
        $this->_agenum = $ageP1;
        return $this->_agenum;
    }
}

function &getRescity($eventID, $city)
{
    if(empty($this->_result))
    {
        $sql23 = "SELECT * FROM #__persons AS a, #__person_reg_event AS b
            WHERE a.personID = b.personID AND b.eventID = '$eventID' AND
            a.city = '$city'
            AND b.clock_time > '00:00:00' ORDER BY clock_time";
        $this->_result = $this->_getList($sql23);
        return $this->_result;
    }
}

```

```

function &getAllcity($eventID, $city)
{
    if(empty($this->_totalnum))
    {
        $db10 =& JFactory::getDBO();
        $sql24 = "SELECT personID FROM #__person_reg_event WHERE
                eventID = '$eventID' AND clock_time > '00:00:00'
                ORDER BY clock_time";
        $db10->setQuery($sql24);
        $persID = $db10->loadResultArray();
        $n1 = count($persID);
        for($i = 0; $i < $n1; $i++)
        {
            $sql25 = "SELECT personID from #__persons
                    WHERE city = '$city' AND personID = '$persID[$i]'";
            $db10->setQuery($sql25);
            $test = $db10->loadResult();
            if($test != 0)
            {
                $persID2[$this->n] = $test;
                $this->n++;
            }
        }
        $n2 = count($persID2);
        for($i = 0; $i < $n2; $i++)
        {
            $sql26 = "SELECT count(clock_time)
                    FROM #__person_reg_event
                    WHERE clock_time <= (SELECT clock_time FROM #__person_reg_event
                    WHERE eventID = '$eventID' AND personID = '$persID2[$i]')";
            $db10->setQuery($sql26);
            $allpl[$i] = $db10->loadResult();
        }
        $this->_totalnum = $allpl;
        return $this->_totalnum;
    }
}

function &getSexcity($eventID, $city)
{
    if(empty($this->_sexnum))
    {
        $db11 =& JFactory::getDBO();
        $sql27 = "SELECT personID FROM #__person_reg_event WHERE
                eventID = '$eventID' AND clock_time > '00:00:00'
                ORDER BY clock_time";
        $db11->setQuery($sql27);
        $persID = $db11->loadResultArray();
        $n1 = count($persID);
        for($i = 0; $i < $n1; $i++)
        {
            $sql28 = "SELECT personID from #__persons
                    WHERE city = '$city' AND personID = '$persID[$i]'";
            $db11->setQuery($sql28);
            $test = $db11->loadResult();
            if($test != 0)
            {
                $persID2[$this->n1] = $test;
                $this->n1++;
            }
        }
        $n2 = count($persID2);
        for($i = 0; $i < $n2; $i++)
        {
            $sql30 = "SELECT count(LEFT (agegroup, 1)) FROM #__person_reg_event
                    WHERE clock_time <=
                    (SELECT clock_time FROM #__person_reg_event
                    WHERE eventID = '$eventID' AND personID =
                    '$persID2[$i]')
                    AND LEFT(agegroup, 1) =
                    (SELECT LEFT (agegroup, 1) FROM #__person_reg_event

```

```

        WHERE eventID = '$eventID' AND personID =
        '$persID2[$i]';
        $db11->setQuery($sql30);
        $sexpl[$i] = $db11->loadResult();
    }
    $this->_sexnum = $sexpl;
    return $this->_sexnum;
}
}

function &getAgecity($eventID, $city)
{
    if(empty($this->_agenum))
    {
        $db12 =& JFactory::getDBO();
        $sql31 = "SELECT personID FROM #__person_reg_event WHERE
            eventID = '$eventID' AND clock_time > '00:00:00'
            ORDER BY clock_time";
        $db12->setQuery($sql31);
        $persID = $db12->loadResultArray();
        $n1 = count($persID);
        for($i = 0; $i < $n1; $i++)
        {
            $sql32 = "SELECT personID from #__persons
                WHERE city = '$city' AND personID = '$persID[$i]'";
            $db12->setQuery($sql32);
            $stest = $db12->loadResult();
            if($stest != 0)
            {
                $persID2[$this->n2] = $stest;
                $this->n2++;
            }
        }
        $n2 = count($persID2);
        for($i = 0; $i < $n2; $i++)
        {
            $sql33 = "SELECT count(agegroup) FROM #__person_reg_event
                WHERE clock_time <=
                (SELECT clock_time FROM #__person_reg_event
                WHERE eventID = '$eventID' AND personID = '$persID2[$i]')
                AND agegroup =
                (SELECT agegroup FROM #__person_reg_event
                WHERE eventID = '$eventID' AND personID = '$persID2[$i]')";
            $db12->setQuery($sql33);
            $agePl[$i] = $db12->loadResult();
        }
        $this->_agenum = $agePl;
        return $this->_agenum;
    }
}

function &getResagegr($eventID, $agegroup)
{
    if(empty($this->_result))
    {
        $db13 =& JFactory::getDBO();
        $sql34 = "SELECT * FROM #__person_reg_event AS a, #__persons AS b
            WHERE a.personID = b.personID AND a.eventID = '$eventID' AND
            a.agegroup =
            (SELECT agegroup FROM #__person_reg_event
            WHERE id = '$agegroup' AND eventID = '$eventID')
            AND a.clock_time > '00:00:00' ORDER BY clock_time";
        $this->_result = $this->_getList($sql34);
        return $this->_result;
    }
}

function &getAllagegr($eventID, $agegroup)
{
    if(empty($this->_totalnum))
    {
        $db13 =& JFactory::getDBO();
    }
}

```

```

    $sql35 = "SELECT agegroup FROM #__person_reg_event
              WHERE eventID = '$eventID' AND id = '$agegroup'";
    $db13->setQuery($sql35);
    $agegroup = $db13->loadResult();

    $sql36 = "SELECT id FROM #__person_reg_event
              WHERE agegroup = '$agegroup'
              AND clock_time > '00:00:00'";
    $db13->setQuery($sql36);
    $id = $db13->loadResultArray();
    $idcount = count($id);
    for($i = 0; $i < $idcount; $i++)
    {
        $sql37 = "SELECT count(clock_time)
                  FROM #__person_reg_event
                  WHERE clock_time <= (SELECT clock_time FROM #__person_reg_event
                  WHERE id = '$id[$i]')";
        $db13->setQuery($sql37);
        $allpl[$i] = $db13->loadResult();
    }
    $this->_totalnum = $allpl;
    return $this->_totalnum;
}
}

function &getSexagegr($eventID, $agegroup)
{
    if(empty($this->_sexnum))
    {
        $db14 =& JFactory::getDBO();
        $sql38 = "SELECT agegroup FROM #__person_reg_event
                  WHERE eventID = '$eventID' AND id = '$agegroup'";
        $db14->setQuery($sql38);
        $agegroup = $db14->loadResult();

        $sql39 = "SELECT id FROM #__person_reg_event
                  WHERE agegroup = '$agegroup'
                  AND clock_time > '00:00:00'";
        $db14->setQuery($sql39);
        $id = $db14->loadResultArray();
        $idcount = count($id);
        for($i = 0; $i < $idcount; $i++)
        {
            $sql40 = "SELECT count(LEFT (agegroup, 1)) FROM #__person_reg_event
                      WHERE clock_time <=
                      (SELECT clock_time FROM #__person_reg_event
                      WHERE id = '$id[$i]')
                      AND LEFT(agegroup, 1) =
                      (SELECT LEFT (agegroup, 1) FROM #__person_reg_event
                      WHERE id = '$id[$i]')";
            $db14->setQuery($sql40);
            $sexpl[$i] = $db14->loadResult();
        }
        $this->_sexnum = $sexpl;
        return $this->_sexnum;
    }
}

function &getAgeagegr($eventID, $agegroup)
{
    if(empty($this->_agenum))
    {
        $db15 =& JFactory::getDBO();
        $sql41 = "SELECT agegroup FROM #__person_reg_event
                  WHERE eventID = '$eventID' AND id = '$agegroup'";
        $db15->setQuery($sql41);
        $agegroup = $db15->loadResult();

        $sql42 = "SELECT id FROM #__person_reg_event
                  WHERE agegroup = '$agegroup'
                  AND clock_time > '00:00:00'";
        $db15->setQuery($sql42);
    }
}

```

```

        $id = $db15->loadResultArray();
        $idcount = count($id);
        for($i = 0; $i < $idcount; $i++)
        {
            $sql43 = "SELECT count(agegroup) FROM #__person_reg_event
                WHERE clock_time <=
                (SELECT clock_time FROM #__person_reg_event
                WHERE id = '$id[$i]')
                AND agegroup =
                (SELECT agegroup FROM #__person_reg_event
                WHERE id = '$id[$i]')";
            $db15->setQuery($sql43);
            $agePl[$i] = $db15->loadResult();
        }
        $this->_agenum = $agePl;
        return $this->_agenum;
    }
}
}

```

com_results\models\single.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-12-24 08:33:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

// Make sure that the class can only be invoked inside Joomla
defined( '_JEXEC' ) or die( 'Restricted Control' );

// Import Joomla Application Model
jimport('joomla.application.component.model');

class ResultsModelSingle extends JModel
{
    var $_data = null;
    var $_result = null;
    var $_agegroups = null;
    var $_agenum = null;
    var $_agepl = null;
    var $_sexpl = null;

    function &getData()
    {
        $eventID = JRequest::getVar('eventID', 0);
        if (empty($this->_data))
        {
            $query = "SELECT * FROM #__events WHERE eventID = '$eventID'";
            $db =& $this->getDBO();
            $db->setQuery($query);
            $this->_data = $db->loadObject();
        }
        return $this->_data;
    }

    function &getResult()
    {
        $eventID = JRequest::getVar('eventID', 0);
        if (empty($this->_result))
        {
            $sql3 = "SELECT * FROM #__persons AS a, #__person_reg_event AS b
                WHERE a.personID = b.personID AND b.eventID = '$eventID' AND
                b.clock_time > '00:00:00' ORDER BY clock_time";

```

```

        // $sql2 = "SELECT * from #__persons AS a, #__events AS b,
        #__person_reg_event AS c
        // WHERE b.eventID = c.eventID AND a.personID = c.personID AND
        eventID = '$eventID' ";
        $this->_result = $this->_getList($sql3);
    }
    return $this->_result;
}

function &getAgegroups()
{
    $eventID = JRequest::getVar('eventID', 0);
    if (empty($this->_agegroups))
    {
        $sql3 = "SELECT id AS value, agegroup AS text from #__person_reg_event
        WHERE eventID = '$eventID' AND clock_time > '00:00:00'
        GROUP BY agegroup ORDER BY agegroup";
        $this->_agegroups = $this->_getList($sql3);
    }
    return $this->_agegroups;
}

function &getAgenum()
{
    $eventID = JRequest::getVar('eventID', 0);
    if (empty($this->_agenum))
    {
        $sql4 = "SELECT count(bib)
        WHERE eventID = '$eventID'
        GROUP BY agegroup";
        $this->_agenum = $this->_getList($sql4);
    }
    return $this->_agenum;
}

function &getAgepl()
{
    $eventID = JRequest::getInt('eventID', 0);
    if (empty($this->_agepl))
    {
        $sql5 = "SELECT clock_time FROM #__person_reg_event
        WHERE eventID = '$eventID'
        AND clock_time > '00:00:00'
        ORDER BY clock_time";
        $db5 =& JFactory::getDBO();
        $db5->setQuery($sql5);
        $time = $db5->loadResultArray();
        $num = count($time);
        for($i = 0; $i < $num; $i++)
        {
            $sql6 = "SELECT id from #__person_reg_event
            WHERE clock_time = '$time[$i]'";
            $db5->setQuery($sql6);
            $id[$i] = $db5->loadResult();
        }
        $num2 = count($id);
        for($i = 0; $i < $num2; $i++)
        {
            $sql7 = "SELECT count(agegroup) FROM #__person_reg_event
            WHERE clock_time <=
            (SELECT clock_time FROM #__person_reg_event
            WHERE id = '$id[$i]')
            AND agegroup =
            (SELECT agegroup FROM #__person_reg_event
            WHERE id = '$id[$i]')";
            $db5->setQuery($sql7);
            $this->_agepl[$i] = $db5->loadResult();
        }
    }
    return $this->_agepl;
}
}

```



```

<?php
    foreach ($this->rows as $row)
    {
        $link = JRoute::_('index.php?option=com_results&eventID=' . $row->eventID .
'&view=single');
        echo '<a href="' . $link . '"> ' .
        $row->eventLogo . '</a>';
        echo '<hr>';
    }
?>

```

com_results\views\showresults\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2010-01-12 14:50:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

// Make sure that the class can only be invoked inside Joomla
defined( '_JEXEC' ) or die( 'Restricted access' );
jimport( 'joomla.application.component.view' );
class ResultsViewShowresults extends JView
{
    function display($tpl = null)
    {
        $bib = JRequest::getInt('bib', '', 'post', 'integer');
        $fname = JRequest::getVar('firstName', '', 'post', 'string');
        $lname = JRequest::getVar('lastName', '', 'post', 'string');
        $city = JRequest::getVar('city', '', 'post', 'string');
        $agegroup = JRequest::getInt('agegroup', '', 'post', 'integer');
        $eventID = JRequest::getInt('eventID');
        $code = JRequest::getInt('code');
        $model = &$this->getModel('showresults');
        switch ($code)
        {
            case 1:
                $results = $model->getResult($eventID, $bib);
                $rows = $model->getEventname($eventID);
                $totalnum = $model->getTotalnum($eventID, $bib);
                $sexnum = $model->getSexnum($eventID, $bib);
                $agenum = $model->getAgenum($eventID, $bib);
                break;

            case 2:
                $results = $model->getResfname($eventID, $fname, $lname);
                $rows = $model->getEventname($eventID);
                $totalnum = $model->getTotalnum($eventID, $fname, $lname);
                $sexnum = $model->getSexfnum($eventID, $fname, $lname);
                $agenum = $model->getAgefnum($eventID, $fname, $lname);
                break;

            case 3:
                $results = $model->getRescity($eventID, $city);
                $rows = $model->getEventname($eventID);
                $totalnum = $model->getTotalnum($eventID, $city);
                $sexnum = $model->getSexcity($eventID, $city);
                $agenum = $model->getAgecity($eventID, $city);
                break;

            case 4:
                $results = $model->getResagegr($eventID, $agegroup);
                $rows = $model->getEventname($eventID);
                $totalnum = $model->getTotalnum($eventID, $agegroup);

```



```

        $sexnum = $model->getSexagegr($eventID, $agegroup);
        $agenum = $model->getAgeagegr($eventID, $agegroup);
        break;
    }
    $this->assignRef('totalnum', $totalnum);
    $this->assignRef('sexnum', $sexnum);
    $this->assignRef('agenum', $agenum);
    $this->assignRef('results', $results);
    $this->assignRef('rows', $rows);
    parent::display($tpl);
}
}
}

```

com_results\views\showresults\tmpl\default.php

```

<?php
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
$m = 1; //counter for Male
$f = 1; //counter for Female
?>
<div class="result1">
    <p align="center">Results for <?php echo $this->rows->eventName; ?> at
        <?php echo JHTML::date($this->rows->event_Date); ?></p>
</div>
<div class="result6">
<fieldset>
<table class="adminlist">
    <thead>
        <tr >

            <th id="res" width="4%" align="center">Bib</th>
            <th id="res" width="12%" align="center">First Name</th>
            <th id="res" width="13%" align="center">Middle name</th>
            <th id="res" width="12%" align="center">Last Name</th>
            <th id="res" width="15%" align="center">City</th>
            <th id="res" width="5%" align="center">State</th>
            <th id="res" width="5%" align="center">Age</th>
            <th id="res" width="12%" align="center">Age Group</th>
            <th id="res" width="3%" align="center">Sex</th>
            <th id="res" width="8%" align="center">Time</th>
            <th id="res" width="8%" align="center">Pace</th>
            <th id="res" width="7%" align="center">AllPI</th>
            <th id="res" width="7%" align="center">SexPI</th>
            <th id="res" width="5" align="center">AgePI</th>
        </tr>
    </thead>
    <?php
    jimport('joomla.filter.output');
    // Variabe that creates different CSS. 1 or 0 to distinguish rows
    $k = 0;
    $no = 1;

    // Loop through the database as long as there are records
    for($i=0, $n=count($this->results); $i < $n; $i++)
    {
        $row =&$this->results[$i];
        $totalnum1 = $this->totalnum[$i];
        $sexPl = $this->sexnum[$i];
        $agePl = $this->agenum[$i];
        ?>
        <tr class="<?php echo "row$k"; ?>">
            <td align="center">
                <?php echo $row->bib; ?>
            </td>
            <td>
                <?php echo $row->firstName; ?></a>
            </td>

```

```

        </td>
        <td align="center">
            <?php echo $row->middleName; ?>
        </td>
        <td>
            <?php echo $row->lastName; ?>
        </td>
        <td>
            <?php echo $row->city; ?>
        </td>
        <td align="center">
            <?php echo $row->state; ?>
        </td>
        <td align="center">
            <?php echo $row->raceDayAge; ?>
        </td>
        <td align="center">
            <?php echo $row->agegroup; ?>
        </td>
        <td align="center">
            <?php $gen = $row->gender;
            echo strtoupper($gen[0]);
            ?>
        </td>
        <td align="center">
            <?php echo $row->clock_time; ?>
        </td>
        <td align="center">
            <?php echo $row->pace; ?>
        </td>
        <td align="center">
            <?php echo $totalnum1; ?>
        </td>
        <td align="center">
            <?php echo $sexPl;?>
        </td>
        <td align="center">
            <?php echo $agePl; ?>
        </td>
    </tr>
<?php

    // Switch k between 1 and 0 for different CSS
    $k = 1 - $k;
    }
    ?>

</table>
</fieldset>
</div>

```

com_results\views\single\view.html.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-12-26 18:33:PST
 * @package     Joomla! 1.5
 * @subpackage  Registration
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */

// Make sure that the class can only be invokes inside joomla
defined( '_JEXEC' ) or die( 'Restricted access' );
jimport( 'joomla.application.component.view' );
class ResultsViewSingle extends JView
{
    function display($tpl = null)
    {

```

```

// Get the data from the model to extract event info
$rows =& $this->get('data');
// Get result data from the model
$results =& $this->get('result');
//Get the age groups from the model
$age =& $this->get('agegroups');
$agenum =& $this->get('agenum');
$pl =& $this->get('agepl');
$sex =& $this->get('sexpl');
$agegroup[] = JHTML::_('select.option', '0',
    JText::_('Age Group'), 'value', 'text');
$agegroup = array_merge($agegroup, $age);
$lists['agegroup'] = JHTML::_('select.genericlist',
    $agegroup, 'agegroup', 'class="inputbox" size="1" ', 'value', 'text');
//Assign the organisation list to the template
$this->assignRef('lists', $lists);
//Get the event date and convert it to a more friendly form
$date = JHTML::date($rows->event_Date);

//Pass the array of event data to the view
$this->assignRef('rows', $rows);

$this->assignRef('agenum', $agenum);
//Pass result data to the view
$this->assignRef('results', $results);
//pass the date to the view
$this->assignRef('date', $date);

$this->assignRef('pl', $pl);
$this->assignRef('sex', $sex);
parent::display($tpl);
}
}
?>

```

com_results\views\single\tmpl\default.php

```

<?php
/**
 * @Author      Steinar Johansen
 * @version     Version 1 created 2009-12-27 09:31:PST
 * @package     Joomla! 1.5
 * @subpackage  Results
 * @Copyright   Copyright (C) 2005 - 2008 Open Source Matters. All rights reserved.
 *
 * This file is the entry to registration system
 */
$m = 1; //counter for Male
$f = 1; //counter for Female
$eventID = JRequest::getVar('eventID', 0);
?>
<div class="result1">
    <p align="center">Results for <?php echo $this->rows->eventName; ?> at <?php echo $this->date; ?></p>
</div>

<div class="result2">
    <fieldset>
        <form action="index.php?option=com_results&task=hibsearch"
            method="post" id="josForm" name="josForm" style="font-size: 8pt;">
            <table>
                <tr>
                    <td colspan="3" align="center"><h3>Search for runner</h3></td>
                </tr>
                <tr>
                    <td width="50px">
                        Bib:
                    </td>
                    <td width="100px">

```

```

        <input class="text_area" type="" name="bib" id="bib"
        size="10" maxlength="5" value="" />
    </td>
    <td width="50px" align="right">
        <button class="button validate" type="submit" >
        <?php echo JText::_({'Submit'}); ?></button>
        <input type="hidden" name="task" value="bibsearch" />
        <input type="hidden" name="eventID" value="<?php echo $eventID;
        ?>" />
        <input type="hidden" name="code" value="1" />
        <?php echo JHTML::_('form.token'); ?>
    </td>
</tr>
</table>
</form>
</fieldset>
</div>

```

```

<div class="result3">
    <fieldset>
        <form action="index.php?option=com_results&task=fnamesearch"
        method="post" id="josForm" name="josForm" style="font-size: 8pt;">
            <table>
                <tr>
                    <td colspan="4" align="center"><h3>Search for runner</h3></td>
                </tr>
                <tr>
                    <td width="70px">
                        First Name:
                    </td>
                    <td width="100px">
                        <input class="text_area" type="" name="firstName" id="firstName"
                        size="13" maxlength="50" value="" />
                    </td>
                    <td width="70px">
                        Last Name:
                    </td>
                    <td width="100px">
                        <input class="text_area" type="" name="lastName" id="lastName"
                        size="13" maxlength="50" value="" />
                    </td>
                    <td align="right">
                        <button class="button validate" type="submit" ><?php echo
                        JText::_({'Submit'}); ?></button>
                        <input type="hidden" name="task" value="fnamesearch" />
                        <input type="hidden" name="eventID" value="<?php echo $eventID;
                        ?>" />
                        <input type="hidden" name="code" value="2" />
                        <?php echo JHTML::_('form.token'); ?>
                    </td>
                </tr>
            </table>
        </form>
    </fieldset>
</div>

```

```

<div class="result5">
    <fieldset>
        <form action="index.php?option=com_results&task=citysearch"
        method="post" id="josForm" name="josForm" style="font-size: 8pt;">
            <table>
                <tr>
                    <td colspan="3" align="center"><h3>Search for runner</h3></td>
                </tr>
                <tr>
                    <td width="50px">
                        City:
                    </td>
                    <td width="100px">
                        <input class="text_area" type="" name="city" id="city"
                        size="10" maxlength="50" value="" />
                    </td>
                </tr>
            </table>
        </form>
    </fieldset>
</div>

```



```

// Variabe that creates different CSS. 1 or 0 to distingush rows
$k = 0;
$no = 1;

// Loop through the database as long as there are records
for($i=0, $n=count($this->results); $i < $n; $i++)
{
    $row =&$this->results[$i];
    $agepl = &$this->pl[$i];
    $sexpl = &$this->sex[$i];
    ?>
    <tr class="<?php echo "row$k"; ?>">
    <td align="center">
        <?php echo $row->bib; ?>
    </td>
    <td>
        <?php echo $row->firstName; ?></a>
    </td>
    <td align="center">
        <?php echo $row->middleName; ?>
    </td>
    <td>
        <?php echo $row->lastName; ?>
    </td>
    <td>
        <?php echo $row->city; ?>
    </td>
    <td align="center">
        <?php echo $row->state; ?>
    </td>
    <td align="center">
        <?php echo $row->raceDayAge; ?>
    </td>
    <td align="left">
        <?php echo $row->agegroup; ?>
    </td>
    <td align="center">
        <?php $gen = $row->gender;
            echo strtoupper($gen[0]);
            ?>
    </td>
    <td align="center">
        <?php echo $row->clock_time; ?>
    </td>
    <td align="center">
        <?php echo $row->pace; ?>
    </td>
    <td align="center">
        <?php echo $no++; ?>
    </td>
    <td align="center">
        <?php echo $sexpl; ?>
    </td>
    <td align="center">
        <?php echo $agepl; ?>
    </td>

</tr>
<?php

// Switch k between 1 and 0 for different CSS
$k = 1 - $k;
}
?>

</table>
</fieldset>
<

```

REFERENCES

- [1] H. Graf, *Building Websites with Joomla! 1.5*. Birmingham, B27 6PA, UK: Packet Publishing Ltd, 2008.
- [2] J. Kennard, *Mastering Joomla! 1.5 Extension and Framework Development: The Professional Guide to Programming Joomla!*. Birmingham, B27 6PA, UK: Packet Publishing Ltd, 2007.
- [3] M Kifer, A. Bernstein and P. M. Lewis, *Database Systems: An Application-Oriented Approach*, 2nd ed. Stony Brook, NY: PEARSON Addison Wesley, 2006.
- [4] J. L. LeBlanc, *Learning Joomla! 1.5 ExtensionDevelopment: Creating Moduls, Components, and Plug-Ins with PHP*. Birmingham, B27 6PA, UK: Packet Publishing Ltd, 2008.
- [5] V. M. Manning, *Database Design, Application Developement & Administration*. Denver, CO: McGraw-Hill/Irwin, 2004
- [6] D. Powers, *PHP Object-Oriented Solutions*, New York, NY: Springer-Verlag, 2008.
- [7] S. Sarkar, *Joomla! with Flash: Build stunning, content-rich, and interactive websites with Joomla! 1.5 and Flash CS4*. Birmingham, B27 6PA, UK: Packet Publishing Ltd, 2009.