California State University, San Bernardino

2013

Online graduate application system

You Li

Recommended Citation

Li, You, "Online graduate application system" (2013). *Theses Digitization Project*. 4274.
https://scholarworks.lib.csusb.edu/etd-project/4274

ONLINE GRADUATE APPLICATION SYSTEM

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

You Li

December 2013

ONLINE GRADUATE APPLICATION SYSTEM

A Project

Presented to the

Faculty of

California State University,

San Bernardino

---

by

You Li

December 2013

Approved by:

Josephine G. Mendoza, Advisor, School
of Computer Science and Engineering

11/20/13
Date

Arturo I Concepcion

Haiyan Qiao

ABSTRACT

The Online Graduate Application System (OGAS) is
designed to improve on an earlier (2005) project 'Graduate
Advising System: Application Component' done by Mr. Kyle
Rotte. OGAS has four major interfaces - Applicants,
Recommenders, Graduate Admissions Coordinator/Assistant and
System Administrator and is developed using PHP, JQuery and
MySQL.

The features that are continued from Mr. Rotte's
project include: the ability for an applicant to submit and
track an application; a recommender to submit
recommendations; the Graduate Admissions Assistant to
monitor the status of applications; and the Graduate
Admissions Coordinator to review and evaluate both the
application and recommendations and then provide an
admission decision. OGAS provides the following
improvements: ability for an applicant to ask questions and
get automatic response either from the system or the
Graduate Admissions Assistant; an applicant to provide
changes to the references after submission of an
application; generate the forms in the same layout as the
paper-based forms; and more importantly, to make changes to
the interfaces with minimal programming.

ACKNOWLEDGEMENTS

First, I would like to thank my project advisor, Dr. Josephine Mendoza, for all her hard work on guiding my project. It is her persistent support that makes this project successful.

In addition, I would also like to thank Dr. Arturo Concepcion and Dr. Haiyan Qiao, my project committee members. Both of them provided critical and insightful feedback and suggestions.

Finally, I would like to thank all the professors in the School of Computer Science and Engineering at California State University, San Bernardino.

TABLE OF CONTENTS

CHAPTER FOUR: APPLICATION SYSTEM IMPLEMENTATION

CHAPTER FIVE: ADMINISTRATION SYSTEM IMPLEMENTATION

CHAPTER SIX: INSTALLATION

CHAPTER SEVEN: CONCLUSIONS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

1.1 Background

Currently the School of Computer Science and

Engineering (CSE) at California State University, San

Bernardino (CSUSB) does not have a Web-based graduate

application system to handle the graduate application

process and maintain graduate applicants' data.

There is an earlier project in 2005, "Graduate

Advising System: Application Component" by Kyle Rene Rotte

who designed an application component for the Graduate

Advising System (GRADS). Kyle's application component has

the following features: (1) allows applicants to submit and

track their applications; (2) allows references to submit

recommendations; and (3) allows the Graduate Admission

Assistant to monitor the status of the application and the

Graduate Admission Coordinator to review and evaluate the

application and recommendations.

However, this project was not implemented and launched

due to technical difficulties beyond the control of Mr.

Rotte.

My project aims to design, implement and launch a new application system – "Online Graduate Application System (OGAS)". In addition to keeping the features of the application component designed by Mr.Rotte, the OGAS will provide more features to improve the usability of the system.

Table 1. Rotte's Application System VS Online Graduate Application System

Features	Rotte's System	OGAS
Allows applicants to submit and track their applications	Yes	Yes
Allows references to submit the recommendations	Yes	Yes
Allows the Graduate Admission Assistant to monitor the status of the application	Yes	Yes
Allows the Graduate Admission Coordinator to review and evaluate the application and recommendations	Yes	Yes
Allows applicants to change the reference(s) after submitting the application	No	Yes
Allows applicants to ask questions about the application process	No	Yes
Generates the Application Form, the Letter of Reference Form, the Admission Evaluation Form, and the Graduate Decision Form in the same layout as the current paper-based forms	No	Yes

1.2 Significance

The OGAS simplifies the process of submitting the application by allowing applicants to submit and track their applications online and recommenders to submit and/or upload their recommendations online. The traditional paper-based application form and letter of reference form will no longer be sent via email or snail mail. Thus, it significantly improves the department's efficiency in processing an application.

All required information for an application can be saved before submission so that the applicants can work on the applications in multiple sessions. The Graduate Admissions Coordinator (GAC) and the Graduate Admissions Assistant (GAA) can monitor and update the status of an application. Updated information will be accessible to applicants, the GAC and the GAA. The system will track an application process and give appropriate feedback and responses to applicants, recommenders, the GAA, and the GAC automatically.

1.3 Purpose

The purpose of OGAS is to allow potential CSE graduate students to submit and track their online applications. In addition, applicants can change and modify their references via OGAS. As part of the application requirements, the system will allow recommenders to submit and/or upload their recommendations online. Also, the system will allow the GAA to monitor their applications and the GAC to evaluate application/recommendation and make admit decisions. Once an application is submitted, the system will be able to verify and validate the application by checking against admission requirements and validation rules.

1.4 Definitions

The following terms are defined as they apply to this project.

Table 2. Definitions

Term	Definition
Applicant	A prospective student applying for the MS CS degree in CSE.
Browser	A software application that accesses and displays data available on the Internet and local networks.
CSE	School of Computer Science and Engineering.
CSUSB	California State University, San Bernardino.
Graduate Admissions Assistant	A staff member who assists the Graduate Admissions Coordinator in processing graduate applications.
Graduate Admissions Coordinator	A faculty member who evaluates applications and admits students into the program.
Graphical User Interface (GUI)	A visual display of files, icons, and menus that allow users to interact with a computer.
Hypertext Markup Language (HTML)	A markup language for structuring text, images, and other data that can be displayed in a web browser.
JavaScript	An interpreted computer programming language for development functions that allow client-side scripts to interact with the user, control the browser, communicate asynchronously with a web server, and alter the document content that is displayed.
JQuery	An open-source cross-browser JavaScript library designed to simplify the client-side scripting of HTML.
Module	A component of a program.
MySQL	An open-source relational database management system.
PHP Hypertext Preprocessor (PHP)	A server-side scripting language designed for web development.
Recommender	A person who submits a letter of reference for applicants.
Status	The state to indicate that if the application is completed or not.
Uniform Resource Locator (URL)	A specific character string that constitutes a reference to a resource.

CHAPTER TWO

SOFTWARE REQUIREMENTS SPECIFICATION

2.1 Project Scope

The specific requirements for OGAS are based on requirements identified in Mr. Rotte's project as well as the current traditional paper-based application process. The following steps are required in this project:

1. Do a thorough analysis of Mr. Rotte's project and the current paper-based application process (including admission requirements and processing steps).

2. Work with the GAA and the GAC to figure out current limitations, problems, and possible new requirements.

3. Design a new database structure based on the results of 1 and 2.

4. Design new graphical user interfaces for applicants, recommenders, the GAC, and the GAA.

The following Computer Science core theories: database design (conceptual modeling, normalization and query processing), software engineering principles (testing),

computing algorithms, and system administration have been

applied in this project.

2.2 Project Perspective

2.2.1 Use-Case Diagram
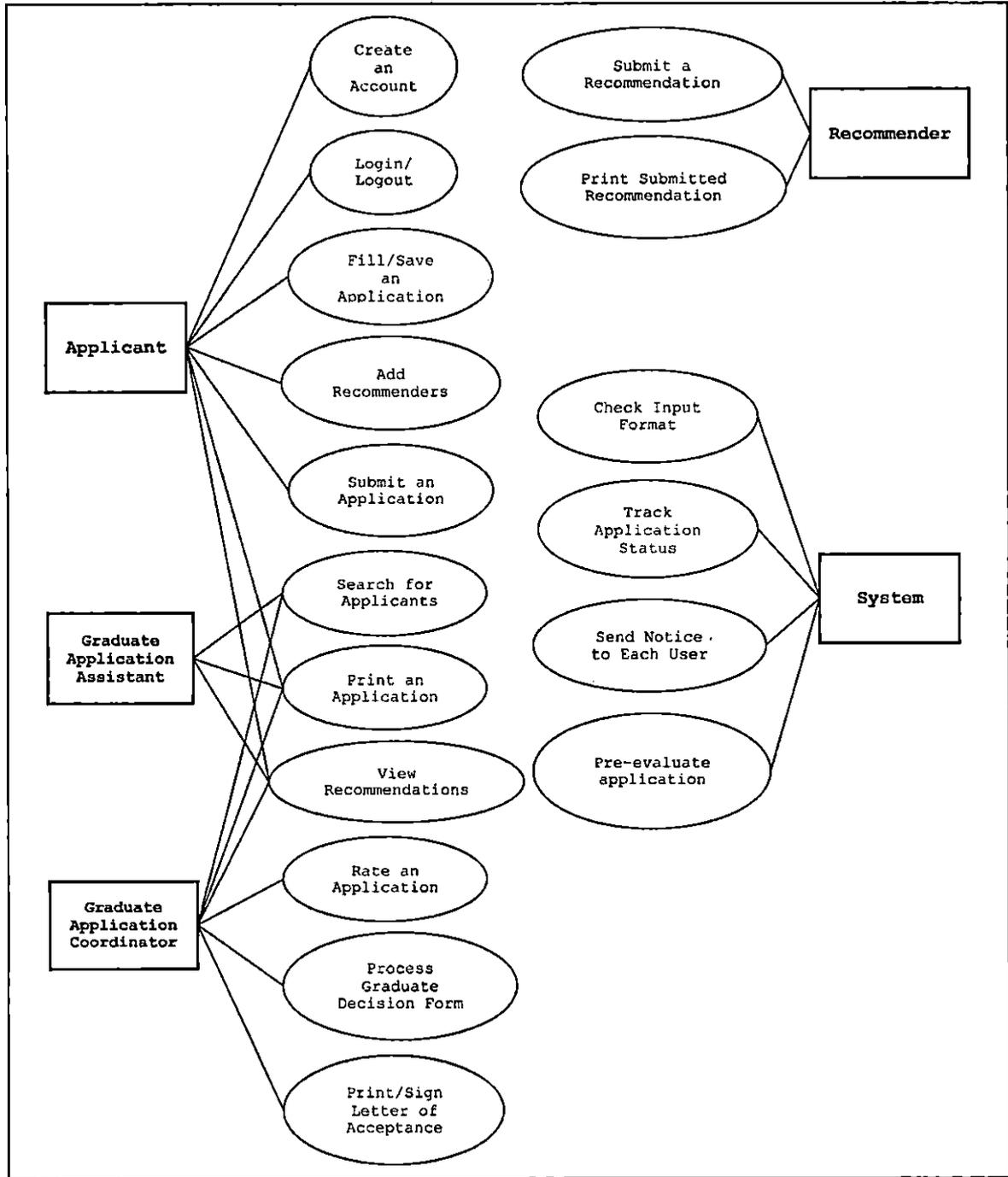
Figure 1. Use Case Diagram

2.2.2 System Interfaces

On the system level, OGAS will employ three major components as system interfaces: client x86 machines, a web server, and a database server.

The client x86 machines are computers used by applicants, the GAA, and the GAC. A browser is required on these computers to display the graphical user interface and data retrieved from the Web Server.

The web server hosts the application program files of OGAS and its supporting programs, such as HTML, JQuery, and PHP. The web server is the data processing unit between clients and the database server. Basically, it saves the data collected from client machines into the database server and passes to client machines the information queried from the database server.

The database server provides structured spaces for storing both the application and the system information. The application information includes all data related to an application. The system information includes the user interface settings and system settings.
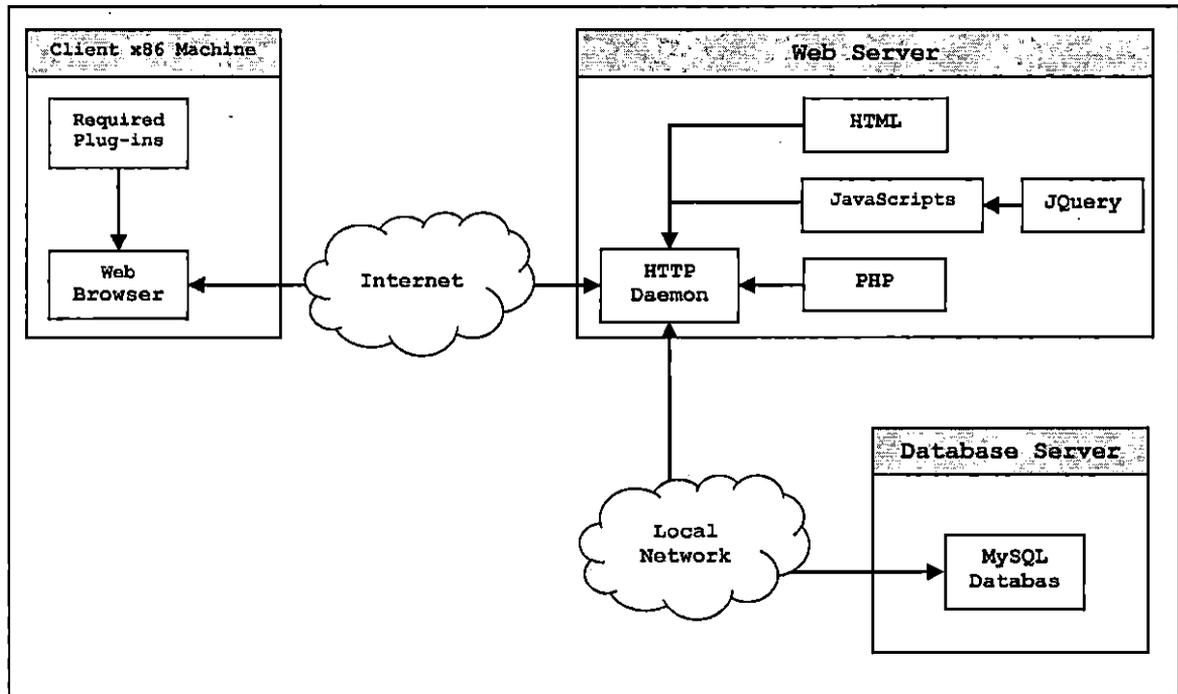
Figure 2. Deployment Diagram for Online Graduate
Application System

2.3 Users

The users of OGAS are classified into five types:

1.   Applicant: A prospective student applying for
       the MS CS degree in CSE.

2.   Recommender: A professor who provides
       recommendation for applicants.

3.   Graduate Admissions Coordinator: A faculty
       member who evaluates applications and makes
       admission decisions.

10

4.   Graduate Admissions Assistant: A staff member
        who assists the Graduate Admissions Coordinator
        in processing graduate applications.

5.   Database Administrator/Developer: A staff
        member who administers the database and
        develops the application system.

2.4 Product Functions

OGAS consists of two major components: the application component and the administration component.

A significant feature of both the application component and the administration component is the dynamic and flexible interface for all users. This feature allows changing the content displayed in the browser without altering the structure of the system.

The system validates input data before saving into the database. Validation checks the length, the value, and the format of inputted data according to the requirements of the input. The inputted data which passed validation checks will be saved into the database.

The system pre-evaluates an application by comparing the score of the GRE test with the minimum GRE score

requirement. For international applicants, the system also pre-evaluates their applications by comparing the score of the language test (TOEFL or IELTS) with the minimum score requirement.

To use the system, a login is required for applicants, the GAA, the GAC, and the System Administrator since there is a different interface for each of these users. To login, an account must first be created. Applicants create accounts by providing a valid e-mail address and a password that contain at least 8 characters, including at least one letter and one number. Accounts for the GAA and the GAC are created by the System Administrator. Recommenders are not required to login to use the system because they can open a unique URL to use the system. Each recommender's unique URL is generated by the system at the moment when the application is submitted. Then each URL will be sent automatically via email by the system to the recommender.

Table 3. User Roles and Privileges

User Role	Privileges
Applicant	▪ Save and update information before submitting ▪ Submit an application ▪ Provide References ▪ View submitted letters of reference with permissions
Recommender	▪ Submit/Upload letters of reference ▪ View his or her submitted letters of reference
Graduate Admissions Assistant	▪ View and update applications status ▪ View/Print applications and letters of reference
Graduate Admissions Assistant	▪ View and update applications status ▪ View and evaluate applications and letters of reference ▪ Admit applicants into the Master's program in CSE
Database Administrator/Developer	▪ View and update information for the Graduate Admissions Assistant and the Graduate Admissions Coordinator ▪ View and update Database structure ▪ Ensures security of the system

2.4.1 Applicant Module

This is the first module in the application component.

This module has three major parts: instructions, the

application form, and the status check list.

Instructions include how to apply to the CSE Master's Program at CSUSB as well as provide links for campus and department admission information.

The application form is the main part where applicants input application data. It has these seven web tabs.

Table 4. Tabs in the Application Form

Web Tab	Input Field
Biographical Information	- Student ID - Student Type (Citizen or Foreigner) - First Name - Middle Initial - Last Name - Date of Birth - Gender - Ethnicity
Contact Information	- Mailing Street Address - City - State - Country - Postal Code - Primary Phone Number - Secondary Phone Number
Program Information	- Program Name (default – Computer Science) - Start Term (includes quarter and year)
Academic Information	- GPA (based on 4.0 scale) - List of Schools Attended - Academic Honors - Activities - Publications
Test Information	- GRE - Date when GRE was taken - GRE Verbal Score - GRE Quantitative Score - GRE Analytical Writing Score - TOEFL (Required for International Students Only) - Date when the TOEFL was taken - TOEFL Score

14

	▪ IELTS (Required for International Students Only)
	    o The data when the IELTS is taken (Required for international students only)
	    o IELTS Score

Recommenders	▪ List of Recommenders
	    o Name
	    o Email
	    o Student indicating to waive the right to view the letter of reference

Statement of Purpose	● A brief statement (no more than 400 words) on applicant's reasons for pursuing the M.S. Degrees in Computer Science

After a successful login, applicants can start their applications from any of these tabs. At the end of each tab, applicants are allowed to save all inputted information within that tab. Each tab with saved information will be marked with a check mark to indicate the completion of that part. This feature will allow the applicants to finish the applications in multiple sessions. Then the applicants can continue working on another tab or logout and continue later. Applications with missing required information, such as the list of recommenders and the statement of purpose, will not be allowed to submit. Once an application is submitted, all information within that application cannot be changed. However, applicants can

delete, modify, or add a recommender after submitting an application.

A check list of requirements (online application form, three letters of reference, and official GRE score report) for an application is generated together with processing status (completed or missing). This checklist can be accessed by the applicants who will know what requirements are still missing and need to be satisfied. A check mark in the check list indicates the requirement is completed or satisfied.

2.4.2 Recommender Module

This second module in the application component provides recommenders two major functions. The first function enables recommenders to submit and/or upload their recommendations online. The identification of the related applicants will be automatically generated by the system and displayed on top of the page. The second function allows recommenders to view and print their submitted recommendations.

2.4.3 Coordinator/Assistant Module

The coordinator/assistant module is part of the administration component. The GAC and the GAA will share most functions of this module. The functions shared by the

GAC and GAA and special functions of the GAC are listed in
Figure 3.

GRADUATE ADMISSIONS COORDINATOR

- Evaluate an applications and
  recommendations
- Render admission decision (Deny,
  Admit (Without Condition, with
  Condition))

GRADUATE ADMISSIONS ASSISTANT

- Send/Receive Messages
- Search applicants
- View applicants' information
- View and print an application
  form
- View and print a letter of
  reference
- View and update the status of an
  application
- View the admission decision for
  applicants and send the
  admission decision letter
- Generate statistical reports
  based on the gender, the
  country, and the ethnicity of
  applicants

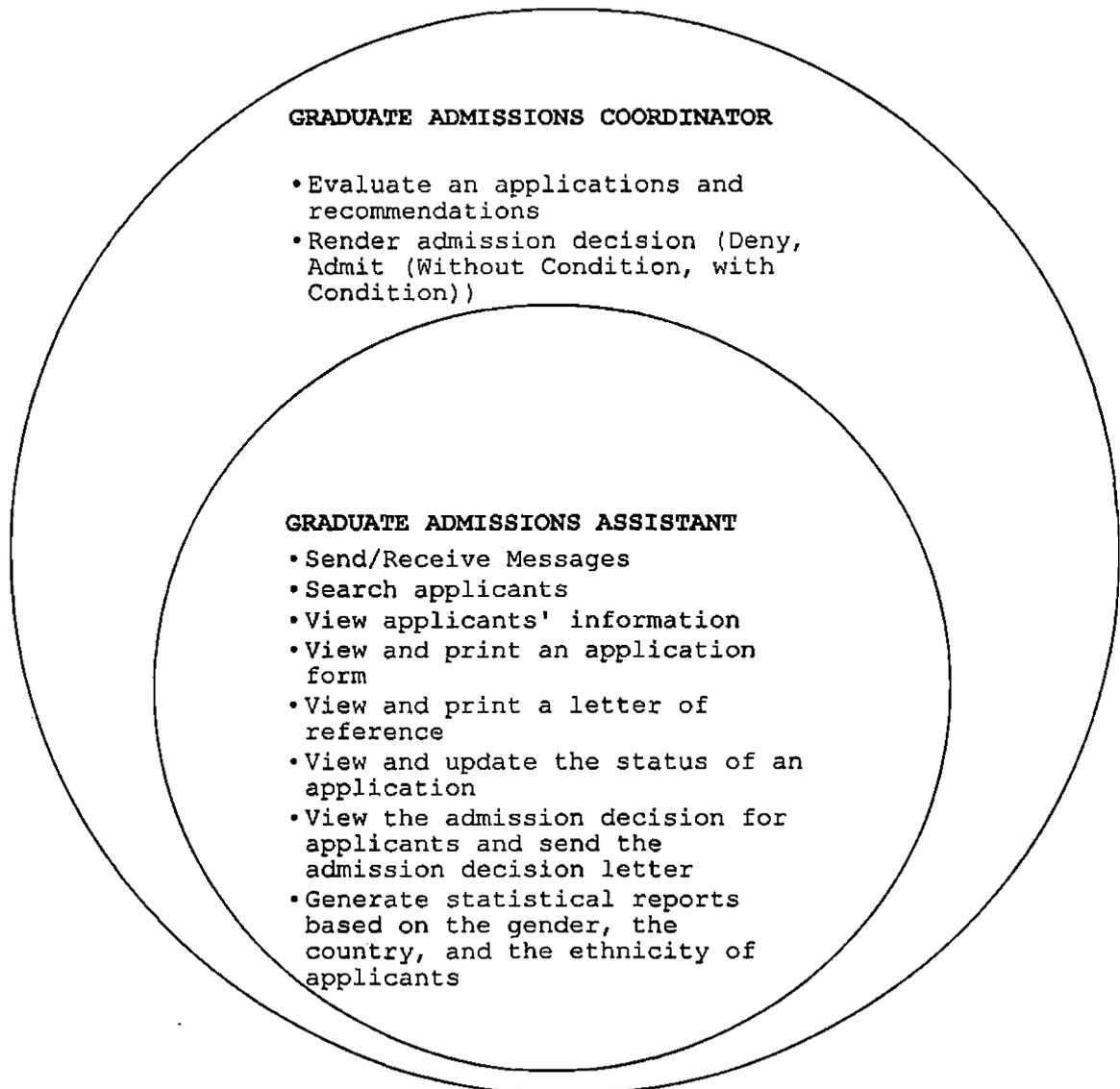Figure 3. Functions of the Graduate Application Assistant
and the Graduate Application Coordinator

An important feature of the coordinator/assistant module is the ability to automatically generate the application form, the letter of reference form, the evaluation form, and the graduate decision form in the same layout as the current paper-based forms. This module also provides the ability to print these forms.

CHAPTER THREE

DATABASE DESIGN AND IMPLEMENTATION

3.1 Design Process

The database provides structured spaces to store both the system information and the application information. A well-designed database ensures effective and efficient data queries. The activities within database analysis and design include:

1. Analyzing the requirements for the application component to find all types of inputs needed to be stored in the database.

2. Identifying attributes that are required to generate each type of input based on the results of 1.

3. Analyzing and deciding the roles that will be involved in completing an application.

4. Gathering all information required for each of the roles based on the results of 3.

5. Creating the conceptual model based on the results of 2 and 4.

6.   Creating the relational model based on the
       result of 5.

7.   Applying the data normalization rules to the
       relational model to identify update, insert and
       delete anomalies.

3.2 Database Table Schemas

There are two types of tables in OGAS - system
information tables and application information tables.

System information tables store the information of all
input fields and their attributes. This information
includes: application form sections, fields in each
application form section and their options, fields in the
recommendation form and their options, fields in the
evaluation form and their options, fields required to
create a new account, and the help information. This
information is maintained in the following tables:
APPLICATION_SECTIONS, APPLICATION_FIELDS,
APPLICATION_FIELDS_OPTIONS, TABLES, TABLES_OPTIONS,
RECOMMENDATION_FIELDS, RECOMMENDATION_FIELDS_OPTIONS,
EVALUATION_FIELDS, EVALUATION_FIELDS_OPTIONS, NEW_ACCOUNT,
and HELP_CENTER. All data that is stored in the system

information tables can be accessed by the application module.

The application form is divided into several sections. Each section may have sub-sections recursively. These information are saved in the APPLICATION_SECTIONS table. Each section or sub-section has several input fields where applicants can input corresponding information. These information are saved in the APPLICATION_FIELDS table. Some types of input fields, such as dropdown menu and radio button, have options for applicants to choose. These information are saved in the APPLICATION_FIELDS_OPTIONS table.

Table 5. APPLICATION_SECTIONS Table Schema

The APPLICATION_SECTIONS table stores information about each section in the application form.				
Column	Type	Optional	Default	Description/Comments
id	int(11)	No		Distinct ID (automatically generated by the system) that identifies this current section
name	varchar(100)	No		Title of this current section
parent	int(11)	No	-1	ID of the upper level section of this current section
content	varchar(500)	No		Instruction for filling fields in this current section
class	varchar(20)	No		Data types and features of current input field (textInput, required, etc.)

Table 6. APPLICATION_FIELDS Table Schema

Column	Type	Optional	Default	Description/Comments
The APPLICATION_FIELDS table stores information about all input fields and their attributes in each section of the application form.				
id	int(11)	No		Distinct ID (automatically generated by the system) that identifies this current field
section	int(11)	No		The ID of the section for this input field
name	varchar(50)	No		Title of this input field
type	varchar(20)	No		Type of this current input field (text, password, select, textarea, table, etc.)
class	varchar(300)	No		Data types and features of current input field (textInput, required, etc.)
hint	varchar(300)	No		Instruction and requirements for filling current field

Table 7. APPLICATION_FIELDS_OPTIONS Table Schema

Column	Type	Optional	Default	Description/Comments
The APPLICATION_FIELDS_OPTIONS table stores information of the options for each input field.				
id	int(11)	No		Distinct ID (automatically generated by the system) that identifies this current option
code	varchar(20)	No		Distinct string of character that represents the value of current option (Each code is unique among all tables in the database.)
name	varchar(100)	No		Display value of current option
parent	varchar(20)	No		The ID of the input filed option

Table 8. TABLES Table Schema

Column	Type	Optional	Default	Description/Comments
The TABLES table stores information of all input fields that display in a table.				
id	int(11)	No		Distinct ID that identifies this current field
parent	int(11)	No		The ID of the input field which current input field belongs to
name	varchar(20)	No		Title of the input field
type	varchar(20)	No		Type of this current input field (text, password, select, textarea, table, etc.)
class	varchar(300)	No		Data types and features of current input field (textInput, required, etc.)
hint	varchar(300)	No		Instruction for filling this current field

Table 9. TABLES_OPTIONS Table Schema

Column	Type	Optional	Default	Description/Comments
The TABLES_OPTIONS table stores information of the options for each input field that will be displayed in a table.				
id	int(11)	No		Distinct ID that identifies this current field option
code	varchar(20)	No		Distinct string of character that represents the value of current option (Each code is unique among all tables in the database.)
name	varchar(75)	No		Display value of current option
parent	int(11)	No		The ID of the input filed which current input field option belongs to

Table 10. RECOMMENDATION_FIELDS Table Schema

Column	Type	Optional	Default	Description/Comments
— The RECOMMENDATION_FIELDS table stores information about all input fields and their attributes in each section of the letter of reference form.				
id	int(11)	No		Distinct ID that identifies this current input field
section	varchar(20)	No		The ID of the section which current input field belongs to
name	varchar(70)	No		Title of the input field
type	varchar(20)	No		Type of this current input field (text, password, select, textarea, table, etc.)
class	varchar(200)	No		Data types and features of current input field (textInput, required, etc.)
hint	varchar(100)	No		Instruction for filling this current field

Table 11. RECOMMENDATION_FIELDS_OPTIONS Table Schema

Column	Type	Optional	Default	Description/Comments
— The RECOMMENDATION_FIELDS_OPTIONS table stores information about options for each input field that will be displayed in a letter of reference.				
id	int(11)	No		Distinct ID (automatically generated by the system) that identifies this current option
code	varchar(20)	No		Distinct string of character that represents the value of current option (Each code is unique among all tables in the database.)
name	varchar(100)	No		Display value of current option

The RECOMMENDATION_FIELDS_OPTIONS table stores information about options for each input field that will be displayed in a letter of reference.				
Column	Type	Optional	Default	Description/Comments
parent	int(11)	No		The ID of the input field which has this option

Table 12. EVALUATION_FIELDS Table Schema

The EVALUATION_FIELDS table stores information of all input fields and their attributes in evaluation form.				
Column	Type	Optional	Default	Description/Comments
id	int(11)	No		Distinct ID that identifies this current input field
name	varchar(70)	No		Title of the input field
type	varchar(20)	No		Type of this current input field (text, password, select, textarea, table, etc.)
class	varchar(200)	No		Data types and features of current input field (textInput, required, etc.)
hint	varchar(100)	No		Instruction for filling this current field

Table 13. EVALUATION_FIELDS_OPTIONS Table Schema

The EVALUATION_FIELDS_OPTIONS table stores information of the options for each input field that will be displayed in evaluation form.				
Column	Type	Optional	Default	Description/Comments
id	int(11)	No		Distinct ID (automatically generated by the system) that identifies this current option
code	varchar(20)	No		Distinct string of character that represents the value of current option (Each code is unique among all tables in the database.)
name	varchar(100)	No		Display value of current option
parent	int(11)	No		The ID of the input field which has this option

Table 14. NEW_ACCOUNT Table Schema

The NEW_ACCOUNT table stores information of all input fields required for creating a new account.				
Column	Type	Optional	Default	Description/Comments
id	int(11)	No		Distinct ID (automatically generated by the system) that identifies this current input field
name	varchar(20)	No		Title of the input field
type	varchar(20)	No		Type of this current input field (text, password, select, textarea, table, etc.)
class	varchar(300)	No		Data types and features of current input field (textInput, required, etc.)
hint	varchar(150)	No		Instruction for filling this current field

26

Table 15. HELP_CENTER Table Schema

The HELP_CENTER stores information about each link in the help center section. Help center section includes all links to the pages which have categorized information about possible issues about the application process.				
Column	Type	Optional	Default	Description/Comments
id	int(11)	No		Distinct ID (automatically generated by the system) that identifies this current input field
name	varchar(35)	No		Topic of the link
content	varchar(150)	No		Address of the page that contains information relevant or related to the current topic
class	varchar(20)	No		Data types and features of current input field (textInput, required, etc.)

The application process needs to store information about applicants, educational background, recommenders, recommendations (values in the letter of reference form or the name and location information of the uploaded letter of reference file), evaluation and admission decisions, and contact information of the GAA and the GAC.

To maintain these data, the following tables are required: APPLICANTS, SCHOOL_ATTENDED, RECOMMENDER, RECOMMENDATION, EVALUATIONS, and STAFF.

27

Table 16. APPLICANTS Table Schema

The APPLICANTS table stores the personal information of applicants and their application.			
Column	Type	Default	Description/Comments
id	int(11)		Distinct ID (automatically generated by the system) that identifies this current applicant
password	varchar(256)		A string of characters that represents the password entered by applicant. This will be stored as encrypted password
salt	varchar(16)		A bit of additional data which is used together with password during the hashing process
studenttype	varchar(20)		Indicates if the applicant is a resident ("resident" = yes and "non-resident" = no)
firstname	varchar(20)		Applicant first name
middleinitial	varchar(2)		Applicant middle initial
lastname	varchar(20)		Applicant last name
email	varchar(100)		Distinct email address provided by applicant (System checks the format and the existence before saving)
studentID	varchar(10)		Student Number (Coyote Number) obtained by the applicant after finishing the CSUSB University Online

			Application in CSU Mentor
before dateofbirth	varchar(12)		Date of birth (MM/DD/YYYY)
gender	varchar(6)		Gender (Female/Male)
ethnicity	varchar(20)		Ethnicity
mailingstreetaddressline1	varchar(200)		Address Line 1
mailingstreetaddressline2	varchar(100)		Address Line 2
country	varchar(50)		Country
state	varchar(50)		State
city	varchar(50)		City
postalcode	varchar(20)		Postal code
primaryphonenumber	varchar(20)		Primary phone number
secondaryphonenumber	varchar(20)		Secondary phone number
programname	varchar(50)		Name of the program (No default value)
startterm	varchar(20)		The quarter and year to start
GPA	varchar(7)		Grade point average in 4.0 scale
academichonors	varchar(2000)		Academic honors received by the applicant
activities	varchar(2000)		Academic conferences and other events the applicant attended
publications	varchar(2000)		Journal articles, papers, presentations, books authored by the applicant
GREdate	varchar(10)		Date when GRE exam was taken
verbal	varchar(5)		Score for the verbal part of the GRE exam
quantitative	varchar(5)		Score for the quantitative part of the GRE exam

analyticalwriting	varchar(5)		Score of analytical writing of the GRE exam
TOEFLdate	varchar(10)		Date when TOEFL was taken
TOEFLscore	varchar(5)		Total score for TOEFL exam
IELTSdate	varchar(10)		Date when IELTS was taken
IELTSscore	varchar(5)		Total score for IELTS exam
statementofpurpose	varchar(5000)		A brief statement on applicant's reasons for pursuing the M.S. Degree in Computer Science.
officialGRE	varchar(20)	not received	Indicates if the official GRE score report is received
officialTOEFL	varchar(20)	not received	Indicates if the official TOEFL score report is received
officialIELTS	varchar(20)	not received	Indicates if the official IELTS score report is received
visits	int(11)	1	Total number of visits which is used for statistic purpose
lastvisit	timestamp	CURRENT_TIMESTAMP	Date and time of last visit which is used to indicate how long this current applicant has not login
finished	varchar(20)	NULL	A string of IDs of completed sections which is separated by commas
status	int(11)	0	Indicates if the application is submitted ("Not submitted" = 0 and "submitted" = 1)

active	int(11)	0	Indicates if the account is active ("Active" = 1 and "Not active" = 0)
hash	varchar(32)		A series of random numbers generated by system to validate the email address

In order to improve the security of the system, the password is encrypted before saving into the database. A hash table combined with salt is used to encrypt the password. When a password is entered, the system generates a string of random characters, which is called "salt", and saves it into the database. Then the system hashes this string together with the inputted password to generate the encrypted password and saves the encrypted password to the database. When the applicants log in, the system retrieves the salt and hashes it together with the inputted password. Then the system compares the hashing result with the encrypted password saved in the database. If they match, then the system allows the applicants to successfully login.

Table 17. SCHOOL_ATTENDED Table Schema

Column	Type	Default	Comments
The SCHOOL_ATTENDED table stores the education history information entered by the applicant during the online application process.			
Id	int(11)		Distinct ID that identifies this education history record
applicant	int(11)		The id of the applicant who has this education history record
schoolname	varchar(200)	.	Name of the school attended by the applicant
yearfrom	varchar(15)		Year when education was started at the school
Yearto	varchar(15)		Year when degree from the school was earned
GPA	varchar(8)		Grade point average on a 4.0 scale
degreeobtained	varchar(50)		Degree obtain from the school
Action	varchar(30)	delete edit	Functions that can be done on one entry of an education history by applicant

Table 18. RECOMMENDER Table Schema

The RECOMMENDER table stores the identification of recommenders and status information of the related recommendations.			
Column	Type	Default	Comments
applicant	int(11)		ID of the applicant who provided this recommender
nameofreference	varchar(100)		Name of the recommender
Email	varchar(100)		Email address of the recommender
waivetoreview	varchar(5)		Applicant's indication to waive the right to review the recommendation from this recommender (Yes/No)
Status	varchar(20)	not received	The status which indicates whether this recommendation has been submitted by the recommender
Action	varchar(30)	delete edit	Functions that can be done on one entry of a recommender by applicant
Hash	varchar(32)		A series of random numbers that is generated by the system when the application is submitted and used to identify and verify a recommender

Table 19. RECOMMENDATION Table Schema

The RECOMMENDATION table stores information submitted by the recommender.			
Column	Type	Default	Comments
Id	int(11)		Distinct ID that identifies this current record of recommendation
recommender	int(11)		The ID of the recommender who submitted this recommendation
applicant	int(11)		The ID of the applicant for whom this recommendation is submitted
Howlonghaveyou knowntheapplic ant	varchar(20)		Length of the time that the recommender has known the applicant
relationship	varchar(20)		The relationship of the recommender with the applicant
position	varchar(50)		The position of the recommender
institution	varchar(100)		The school, organization, or business where the recommender works
Phone	varchar(20)		Work phone number provided by the recommender
Address	varchar(200)		The address of the institution where the recommender works
nativeintellec tualability	varchar(5)		Rating on the applicant's native intellectual ability. ("Upper 1 to 2%" = 1, "Upper 10% but not upper 1 or 2%" = 2, "Upper 25% but not upper 10%" = 3, "Upper half but not upper 25%" = 4, "Lower half" = 5, "No basis for judgment" = 6)
imaginationand probablecreati vity	varchar(5)		Rating on the applicant's imagination and probable creativity. ("Upper 1 to 2%" = 1, "Upper 10% but not upper 1 or 2%" = 2, "Upper 25% but not upper 10%" = 3, "Upper half but not upper

			25%" = 4, "Lower half" = 5, "No basis for judgment" = 6)
breadthofgeneralknowledge	varchar(5)		Rating on the applicant's breadth of general knowledge. ("Upper 1 to 2%" = 1, "Upper 10% but not upper 1 or 2%" = 2, "Upper 25% but not upper 10%" = 3, "Upper half but not upper 25%" = 4, "Lower half" = 5, "No basis for judgment" = 6)
abilityinoralexpression	varchar(5)		Rating on the applicant's ability in oral expression. ("Upper 1 to 2%" = 1, "Upper 10% but not upper 1 or 2%" = 2, "Upper 25% but not upper 10%" = 3, "Upper half but not upper 25%" = 4, "Lower half" = 5, "No basis for judgment" = 6)
writingability	varchar(5)		Rating on the applicant's writing ability. ("Upper 1 to 2%" = 1, "Upper 10% but not upper 1 or 2%" = 2, "Upper 25% but not upper 10%" = 3, "Upper half but not upper 25%" = 4, "Lower half" = 5, "No basis for judgment" = 6)
Emotionalmaturity	varchar(5)		Rating on the applicant's native intellectual ability. ("Upper 1 to 2%" = 1, "Upper 10% but not upper 1 or 2%" = 2, "Upper 25% but not upper 10%" = 3, "Upper half but not upper 25%" = 4, "Lower half" = 5, "No basis for judgment" = 6)
Promiseasateacher	varchar(5)		Rating on the applicant's emotional maturity. ("Upper 1 to 2%" = 1, "Upper 10% but not upper 1 or 2%" = 2, "Upper 25% but not upper 10%" = 3, "Upper half but not upper 25%" = 4, "Lower half" = 5, "No basis for judgment" = 6)
overallrecommendationastopro	varchar(5)		Rating on the applicant's promise as a professional in the field. ("Upper 1 to 2%"

miseasaprofess ional			= 1, "Upper 10% but not upper 1 or 2%" = 2, "Upper 25% but not upper 10%" = 3, "Upper half but not upper 25%" = 4, "Lower half" = 5, "No basis for judgment" = 6)
Overallrecomme ndation	varchar(10)		Rating on the applicant's overall recommendation.
comments	varchar(8000)		Additional comments on the applicant by the recommender
timestamp	timestamp	CURRENT_ TIMESTAMP	Date and time of most recent update of this record (Indicates when the recommendation is submitted)

Table 20. EVALUATION Table Schema

The EVALUATION table stores information on decision by GAC about the application.			
Column	Type	Default	Comments
studentID	varchar(12)		Student identification number
classification	varchar(50)		Classification of an admission (Classified, Conditionally Classified, Probation Conditional, Probation Unclassified, Deny)
admitterm	varchar(20)		Quarter and Year of when the applicant is admitted to the program
deficiencies	varchar(150)		List of pre-requisite courses that must be satisfied by the student
comments	varchar(2000)		Additional remarks by the GAC

Table 21. STAFF Table Schema

The STAFF table stores information about the GAA or the GAC.			
Column	Type	Default	Comments
Id	int(11)		Distinct ID that identifies this current staff member
firstname	varchar(30)		First name
middleinitial	varchar(10)		Middle initial
lastname	varchar(30)		Last name
Title	varchar(30)		Title of the staff (Professor, Associate Professor, Assistant Professor, etc.)
position	varchar(80)		Position the staff (Graduate Admissions Assistant / Graduate Admissions Coordinator)
Email	varchar(100)		Email of the Graduate Admissions Assistant or the Graduate Admissions Coordinator
Phone	varchar(20)		Phone number of the GAA or the Graduate Admissions Coordinator
Salt	varchar(32)		A string of characters which is used together with password during the hashing process for encryption
password	varchar(32)		Password entered by the account holder (encrypted by the system)

CHAPTER FOUR

APPLICATION SYSTEM IMPLEMENTATION

This chapter explains the system functions for the application component of OGAS. Screen shots will be given to display the layouts, possible inputs, and expected outputs.

4.1 Data Validation

The data is validated to ensure the quality of the data that will be stored in the database. Validation is basically done by comparing the entered value of an input field with the input requirements associated with the field. An input field may have one or more requirements, each requirement has a corresponding validation function. All validation functions will be executed right after a value is entered in an input field. In this way, the user is alerted immediately about incorrect input and will have to provide the correct input. Thus, process of saving or submitting the data is speeded up. The table below shows all input requirements and the corresponding validation functions for each input field.

Table 22. Input Requirements and Corresponding Validation

Functions

Requirements	Validation Function Name	Class of Field
Value of the input field is required.	required()	required
Length of the data for the input field must be more than x.	validateMinLength()	validateMinLength
Value of the input field must be bigger than x.	validateMin()	validateMin
Length of the value must be less than x.	validateMaxLength()	validateMaxLength
Value of the input field must be smaller than x.	validateMax()	validateMax()
Value of the input field should contain only numbers and letters (without special characters).	validateAlphaNum()	validateAlphaNum
Value of the input field should be an email address.	validateEmail()	validateEmail
Value of the input field should be a valid URL.	validateUrl()	validateUrl
Value of the input field should be a number.	validateNumber()	validateNumber
Value of the input field should be a whole number.	validateInteger()	validateInteger
Value of the input field should contain only letters (without special characters or numbers).	validateAlpha()	validateAlpha
Value of the input field must contain at least one numeric and one alphabetic character.	validateAtLeast()	validateAtLeast
Value of the input field should contain only alphabetic characters, numbers, spaces, and the following characters: ".", ",", "-", "_", "(", ")", "*", "#", ":".	validatePhrase()	validatePhrase
Value of the input field should be a phone number.	validatePhone()	validatePhone
Value of the input field should be a date (mm/dd/yyyy).	validateDate()	validateDate
Value of the input field should match the value of	validateSameAs()	validateSameAs

another specified input field.		
Check if the entered email already exists in the database.	validateEmailExist()	validateEmailExist

4.2 Applicant: Applicant Login

The applicant login section is located on the starting page of the application component of OGAS. Applicants have to login to start new applications or continue their saved applications. A valid email address and a password are required to login.

Figure 4. Application Login Component

4.3 Applicant: Create an Account

Applicants can create their accounts via OGAS. The window for creating an account can be opened by clicking the button "Create an Account" on the welcome page. A valid email address is required to create an account. A link to activate the account will be generated by the system and sent to the applicant via email. The applicant who fails to activate the account will not be able to login into OGAS.

The system checks the existence of the email. If the email already exists in our system, the applicant has to use another email to create the account. The system also checks the password which must contain at least eight characters, including at least one letter and one number. No symbols are allowed to use in the password.

School of
Computer Science and Engineering

Create Your Account with Us

Login Information

Please enter a personal password to access your application in the future. Passwords must contain at least 8 characters, including at least one letter and one number. The allowed characters are a-z, A-Z, and 0-9. No symbols may be used. Here are some examples of passwords that aren't allowed: 123456, allalphas, l1ke, pi@ces, until, my_app.

Email

Password

Verify Password

                                 ☞ Submit

School of Computer Science and Engineering
California State University San Bernardino
5500 University Parkway
San Bernardino, CA 92407-2397

Figure 5. Window for Creating an Account

4.4 Applicant: Password Recovery

Applicants get a system generated password via email if they forgot the original password. The email address that is used to create the account is required for the system to generate a new password.

Once a request to get a new password is sent to the system, the system will check if the submitted email address exists in the system. If the email address exists, the system generates a random password and sends the new password to the email address provided by the applicant. Then the applicant can use the new password to log in the system.

Figure 6. Window for Password Recovery

4.5 Applicant: Application Instructions

This page shows applicants the step-by-step procedure to do the online application. Useful links for related information are provided on this page.

Figure 7. Application Instructions

4.6 Applicant: Ask Questions

This function provides applicants an interface to ask questions about the application process. Once a question is entered, the system will search an 'FAQ' database and list all related topics that might answer the proposed question. If no solution can be found from the related topics, the proposed question will be automatically sent by the system to the GAA. The GAA can answer the question via the admission component of OGAS. Then both the question and answer provided by the GAA will be added and saved in the 'FAQ' database. Therefore the 'FAQ' database can dynamically grow.

Questions on the Application Process

If you have any questions about the application process, please click here.
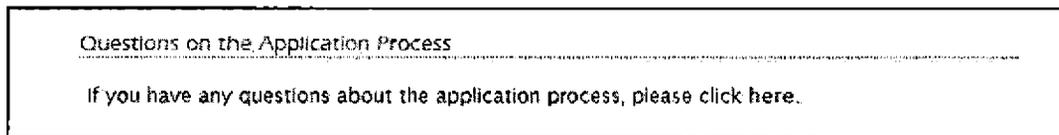
Figure 8. Link of Interface to Ask Questions

4.7 Applicant: Biographical Information

This page is designed to collect biographical information on applicants. The system checks all required input fields. If the value of a required input field is blank, then an error message shows up after the input field. (See Figure 9.)

Figure 9. Biographical Information Form with Validation Results

4.8 Applicant: Contact Information

This page is designed to collect personal information of applicants. The system checks the values of all required input fields including Mailing Street Address Line 1, City, State, Country, Postal Code, and Primary Phone Number. If the values are blank, an error message shows on top of the page.

Figure 10. Contact Information Form

4.9 Applicant: Program Information

This page is designed to collect what term and year the applicant intends to start the degree.

Figure 11. Program Information Form

4.10 Applicant: Academic Background

This page is designed to collect academic background information of applicants. The CSUSB GPA Calculator is a simple tool that can be used to calculate the numeric grade of a CSUSB course by providing the number of units and the letter grade of that course. Thus, it can convert GPAs not on a 4.0 scale to 4.0 scale.

Figure 12. Academic Background Information Form

All schools can be entered by clicking "plus" icon at the right bottom of the 'Colleges and Universities Attended' box. (See Figure 13.)

School Name	Year From	Year To	GPA	Degree Obtained	Actions
Riverside Community College	2009	2012	3.2	BS	🗁 🗑
				"Plus" icon ⟶	⊕

Figure 13. "Plus" Icon for Adding a School

attended. Make sure to include your current institution and any ESL studies (if applicable).

Purpose

& Subm

Add a new school attended

School Name

Year From

Year To

CPA

Degree Obtained

🖫 Submit

Activities

Figure 14. Add a New School Attended

The window to edit the information of a saved record
of an attended school can be opened by clicking the
"Folder" icon. (See Figure 15.)

Figure 15. "Folder" Icon for Editing a School

Figure 16. Edit a Saved School Attended

A record of an attended school can be removed by clicking the "Trash can" icon. (See Figure 17.)

School Name	Year From	Year To	GPA	Degree Obtained	Actions
Riverside Community College	2009	2012	3.2	BS	

Figure 17. "Trash" Icon for Deleting a School

4.11 Applicant: Test Information

This page is designed to collect information on GRE, TOEFL, and IELTS.

Students who got their undergraduate degrees from CSUSB will be presented the screen shown in Figure 18. Figure 19 depicts the screen displayed for international students who did not get their undergraduate degrees from CSUSB.

Figure 18. Test Information Form for Resident Students

| Biographical Information ✓ |
| Contact Information ✓ |
| Program Information ✓ |
| Academic Background ✓ |
| Test Information ✓ |
| Recommendations ✓ |
| Statement Of Purpose ✓ |

Preview & Submit

Test Information

Undergraduate Degree
Do you hold a Bachelor's degree from an English speaking institution?

○ yes
◉ no

Graduate Record Examination (GRE)

We accept GRE scores taken within the last *three* years only

If you have not taken GRE yet leave the fields blank. You can mail us your official GRE scores later. The institutional code for CSUSB is 4099

* Please send an official copy of the GRE test scores to the School of Computer Science & Engineering. Address is given on the bottom of this page.

GRE Date

07/10/2013

Verbal

132

Quantitative

170

Analytical Writing

3.5

English Requirement

TOEFL or IELTS is required for international students only. This requirement can be waived by completing ACLP/IEP level 6 language program and submitting proof of completion

Test Of English As A Foreign Language (TOEFL)

We accept TOEFL scores taken within the last *two* years only. You only need to report a Total Score

TOEFL Date

11/12/2012

TOEFL Score

91

TOEFL Test Type

Please select TOEFL Test T ▼

International English Language Testing System (IELTS)

We accept 6.0 band score taken within last two years only. IELTS is NOT required if you have a satisfactory TOEFL score

IELTS Date

IELTS Score

🖫 Save

Figure 19. Test Information Form for International Students

4.12 Applicant: Recommendations

This page is designed to collect information about recommenders provided by applicants.

Recommenders can be added by clicking the "Plus" icon at the bottom of the recommenders table. (See Figure 20.)

Recommenders

Please list up to 3 recommenders, who will submit letters of recommendation on your behalf. Recommendations should be provided from employers, supervisors, professors, or others who are able to comment on your professional and educational goals and potential.

Name Of Reference | Email | Waive To Review | Status | Actions

"Plus" Icon

Save | Add a new record
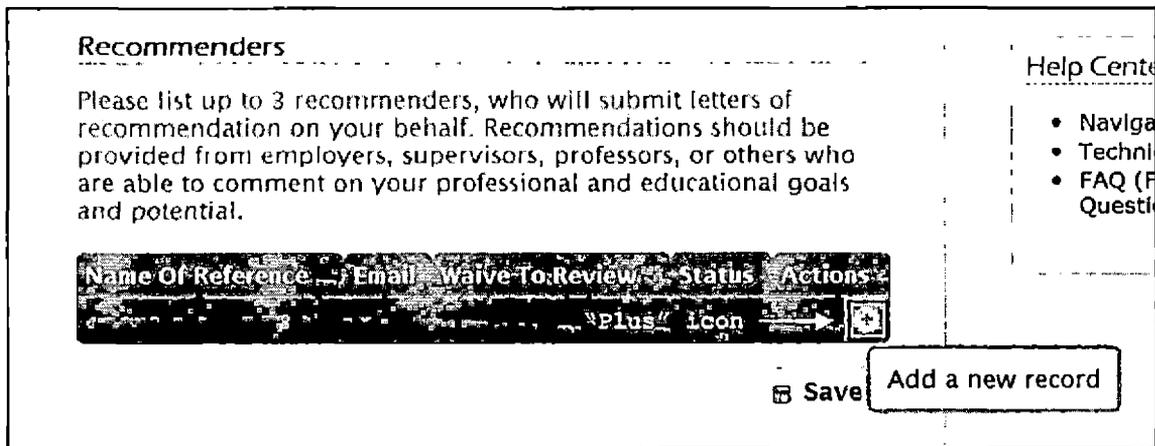
Help Cent

• Naviga
• Techni
• FAQ (F
  Questi

Figure 20. "Plus" Icon for Adding a Recommender

This brings up the 'Add a New Recommender' window

shown in Figure 21. Enter a recommender by entering the

name, email, waive to review in the appropriate fields.

Figure 21. Add a New Recommender

4.13 Applicant: Statement of Purpose

This page is for applicants to write a brief statement on their reasons for pursuing the M.S. Degree in Computer Science.

Figure 22. Form of Statement of Purpose

4.14 Applicant: Save and Submit

At the bottom of each tab, a "Save" button is provided for saving inputted data of all required fields within that tab. A successful save occurs only if all required data are able to pass the validation checks. A system notification of the result of save will be displayed on top of the form. (See Figure 23).

Figure 23. System Notification for a Successful Save

If a save fails, the input fields that failed the validation checks will be highlighted as errors. (See Figure 24.) This happens to all tabs in the application form.

Figure 24. System Notification for a Failed Save

Once all required information in each section is successfully saved, applicants can preview and submit their application by clicking the "Preview & Submit" button below the tab box. (See Figure 25.)

Biographical Information ✓

Contact Information ✓

Program Information ✓

Academic Background ✓

Test Information ✓

Recommendations ✓

Statement Of Purpose ✓
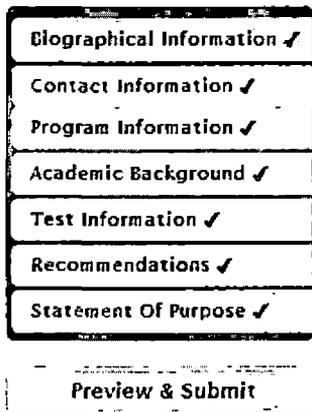
Preview & Submit

Figure 25. Preview and Submit Button

All saved information is shown in Figure 26. If there is an incorrect information entered, the applicant can click the "Resume my application" button at the bottom of the page to correct the errors. Otherwise, the applicant can click the "Submit my application" to submit the application.

Biographical Information

Student ID : *003638010*
Are you a citizen or permanent resident of the U.S.?   *No*
First Name: *Boa*
Middle Initial:
Last Name: *Kim*
Date Of Birth: *01/01/1983*
Gender: *Male*
Ethnicity: *Asian: Korean*

Contact Information

Mailing Street Address Line 1: *111 F St.*
Mailing Street Address Line 2:
City: *San Bernardino*
State: *Ca*
Country: *United States*
Postal Code: *92407*
Primary Phone Number: *9090000000*
Secondary Phone Number:

Program Information

Program Name: *Computer Science*
Start Term: *Spring 2014*

Academic Background

GPA: *3.6*
Academic Honors:

   Honor 1
   Honor 2
   Honor 3
   Honor 4
   Honor 5

Activities:

   Activity 1
   Activity 2
   Activity 3
   Activity 4
   Activity 5

Publications:

   Publication 1

School Name	Year From	Year To	GPA	Degree Obtained
Riverside Community College	2009	2012	3.2	BS

Test Information

GRE Date:
Verbal:
Quantitative:
Analytical Writing:
TOEFL Date:
TOEFL Score:
IELTS Date:
IELTS Score:

Recommendations

Name Of Reference	Email	Waive To Review	Status
Dr. Smith	Li.you.id@gmail.com	Yes	Not Received

Statement Of Purpose

Statement Of Purpose :

   My name is Leo.

Print my application   Resume my application   Submit my application

Figure 26. Preview and Submit an Application

4.15 Applicant: Print Application Form

Applicants can print out the application form after submitting. The form will be generated automatically by the system in the same layout as the current paper-based application form.

Figure 27. Application Form Generated by System

4.16 Applicant: Status Check

Once applicants have submitted their applications
online, they can find out the status on whether the
application form has been completed, whether GRE official
score has been received and whether the recommenders have
submitted their reference letters. A completed item will be
marked with a check mark.

| | | | | ☑ Message | 🔒 Password | ⏻ Logout |

| Home | Application | Status | | |

Welcome back, Leo Lee

Application Check List

Submission Status:
Submitted

Last Date Accessed:
2013-10-22 01:46:53

- Online application form ✓

- Three recommendation letters

Help Center

Name Of Reference	Email	Waive To Review	Status	Action
You Li	Li.you.Id@gmail.com	No	Received	✉ 🔍
David	Li.you.Id@gmail.com	Yes	Received	✉ 🔍
Dr. Jason	Youli222@gmail.com	No	Not Received	✉ 🔍

- Navigation Overview
- Technical Tips
- FAQ (Frequently Asked Questions)

- Official GRE scores report ✓

Figure 28. Application Checklist Status

Applicants can contact their recommenders via email by
clicking the "Mail" icon on the Action column to follow up
on the recommendation letter. Applicants can modify their

recommenders' information by clicking the "Folder" icon on

the Action column. Applicants can also delete a recommender

by clicking the "Trash" icon on the Action column. (See

Figure 29.)

Name Of Reference	Email	Waive To Review	Status	Actions
You Li	Li.you.id@gma	Send email to this recommender		
David	Li.you.id@gma	Edit this recommender's information		
Dr. Jason	Youli222@gma	Delete this recommender		

Figure 29. Icons on the Action Column

67

The 'Contact' window is shown in Figure 30. The contact information (name and email address) of the recommender is generated automatically by the system from the input information given by the applicant. Applicants can enter the Subject and Message in the appropriate fields.
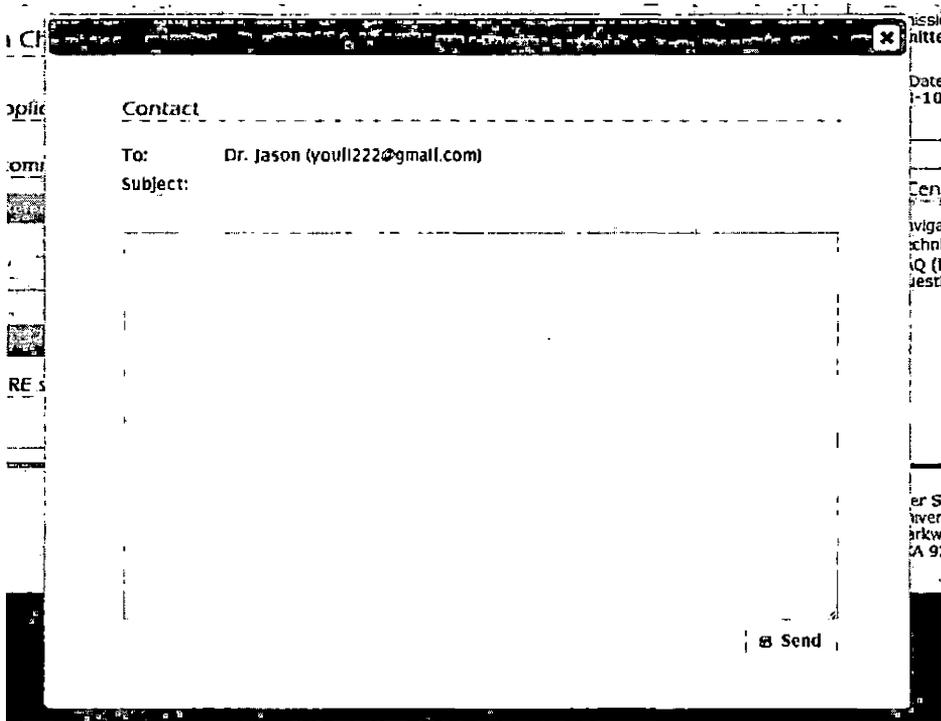
Figure 30. Contact a Recommender

The 'Edit your record' window in Figure 31 shows where the applicants can change the name, email, waive to review option.

Figure 31. Edit a Record of Recommender

4.17 Applicant: Recommendation Review

If the applicant did not waive the right to view a recommendation letter, then the applicant can view and print this letter of reference. The form will be generated by the system and displayed in the same layout as the current paper-based letter of reference form.

Figure 32. View and Print Letter of Reference

4.18 Applicant: Help Center

The Help Center is designed to provide applicants

assistance in completing an application. Sections of the

Help Center include: the Navigation Overview, the Technical

Tips, and the FAQ (Frequently Asked Questions).

Figure 33. Navigation Overview and Help Center

4.19 Recommender: Online Letter of Reference

Recommenders can click the link in the notification

email sent from OGAS system to start submitting a

recommendation.

Figure 34. Notification Email to Recommenders

Information about the applicant will be automatically

filled in by the system and displayed on top of the page.

Figure 35. Application Information Section on Letter of

Reference

Figure 36. Online Letter of Reference

4.20 Recommender: Print Letter of Reference

Recommenders can view and print a submitted letter of reference on the same page where they submitted that recommendation.

Figure 37. View and Print Letter of Reference

CHAPTER FIVE

ADMINISTRATION SYSTEM IMPLEMENTATION

This chapter explains the system functions of the administration component of OGAS. Screen shots will be given to display the layouts, possible inputs, and expected outputs.

5.1 User Login

This page is the starting page of the administration component. The GAC and the GAA will be required to enter their work email address and a valid password to log into the system.
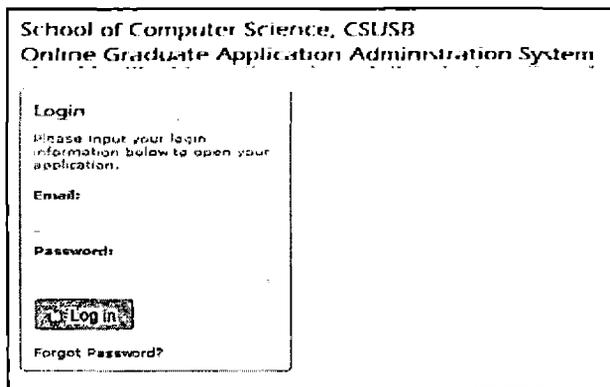
School of Computer Science, CSUSB
Online Graduate Application Administration System

Login

Please input your login
information below to open your
application.

Email:

Password:

Log in

Forgot Password?

Figure 38. User Login Interface for Administration System

5.2 Coordinator/Assistant: Message Box

This set of functions allows the GAA and the GAC to receive and send messages and emails. The messages sent to the GAA or the GAC via OGAS are shown in the 'inbox'. Similarly, the messages sent from the GAA or the GAC via OGAS are shown in 'outbox'. The Contact List saves the contact information of all staff who are involved in the campus and department application process.
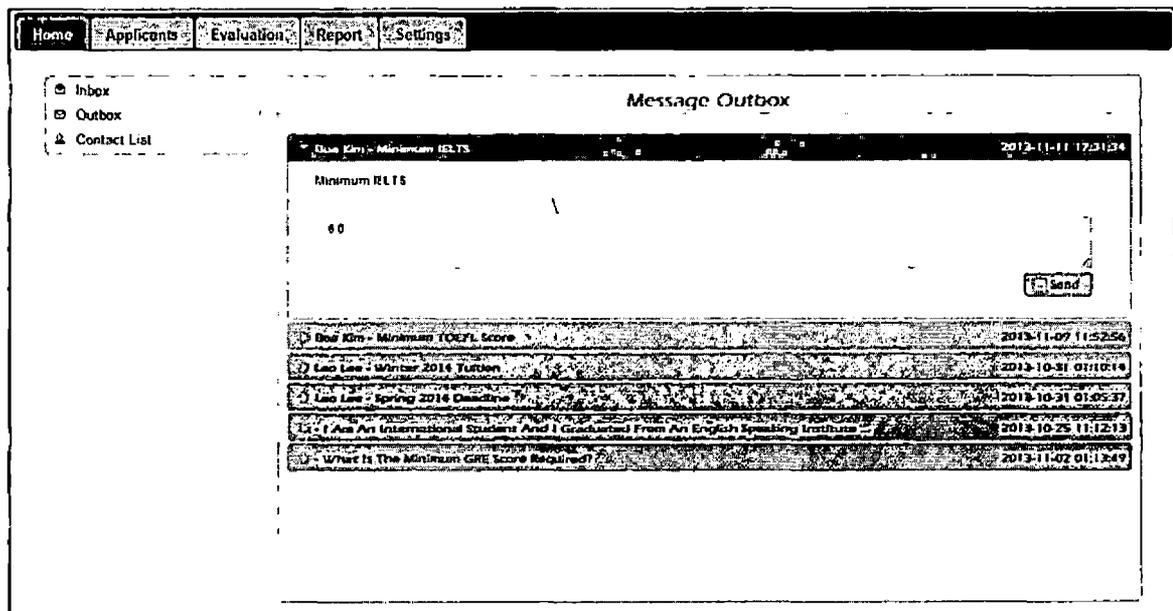
Figure 39. Message Box Interface

5.3 Coordinator/Assistant: Search Applicants

This set of functions allows the GAA and the GAC to look up applicants by first name, last name, email address, or phone number. Three buttons for quick search are provided on the left side: Unsubmitted Application, Submitted Application, and All application. Clicking "Unsubmitted Application" shows all applicants who have started their application but have not submitted. Clicking "Submitted Application" shows all applicants who have submitted their applications. Clicking "All Applicants" shows all applicants who have successfully created accounts on OGAS.

Figure 40. Interface of Searching Applicants

5.4 Coordinator/Assistant: Track Process Status

The most current status of each item required for an

application (Online Application, Three Recommendations,

Office GRE score report, Official TOEFL/IELTS score report)

is displayed using an appropriate icon. The "Question" icon

on a column means that the document (indicated as the title

of the column) has not been received. The "House" icon on a

column means that document has been received. The

"Prohibition" icon on a column means that document is not

required for this applicant.

First Name	M.I.	Last Name	Resident	Start Term	Application Received	Recommendation Received	Official GRE scores Received	Official Language scores Received	Actions
Boa		Kim	No	SP14		? Dr. Smith		TOEFL	
Leo	J	Lee	Yes	W14		? You Li David ? Dr. Jason			

Figure 41. List of Applicants with Status of Each Task

If a user hovers on the icon, the meaning of the icon will
be displayed.

Figure 42. Icon Tip

The meaning of each icon is provided in Figure 43.

Figure 43. Legend of Status Icon

5.5 Coordinator/Assistant: Update Status

The GAA and the GAC can change the "Not received"
required document to "Received" status by clicking the
"Question" icon. A message box will pop out to confirm the
change.
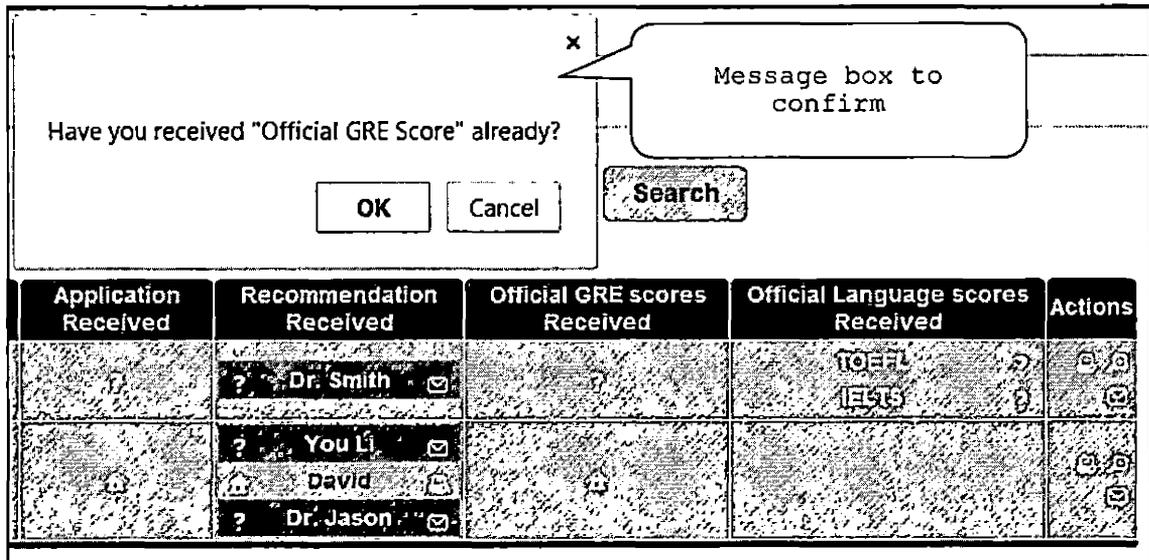
Figure 44. System Confirmation Message for Updating Status

5.6 Coordinator/Assistant: Contact Recommenders

Those recommenders who did not submit the letter of
reference will be highlighted with red background. (See
Figure 45.)

First Name	M.I.	Last Name	Resident	Start Term	Application Received	Reconumendation Received	Official GRE scores Received	Official Language scores Received	Actions
Boa		Kim	No	SP14			"Envelop" Icon	TOEFL CET6	
Leo	J	Lee	Yes	W14		? You Li ☑ David ☑ ? Dr. Jason ☑			

Figure 45. Highlighted Recommenders

The function to contact those recommenders who failed to
submit recommendations is available in OGAS. This function
can be invoked by clicking the "Envelop" icon which is
located at the right side of the highlighted recommender.
(See Figure 45.)

In the 'contact a recommender' window (See Figure 46),
the GAA or the GAC can send emails to the recommenders.
When the "Send" button is clicked, the system reads the
values of the subject and the message. The system queries
the contact information of the GAA or GAC saved in the
database and combines it together with the content of the

83

subject and the message. Then the system sends the combined

information to the recommender automatically.
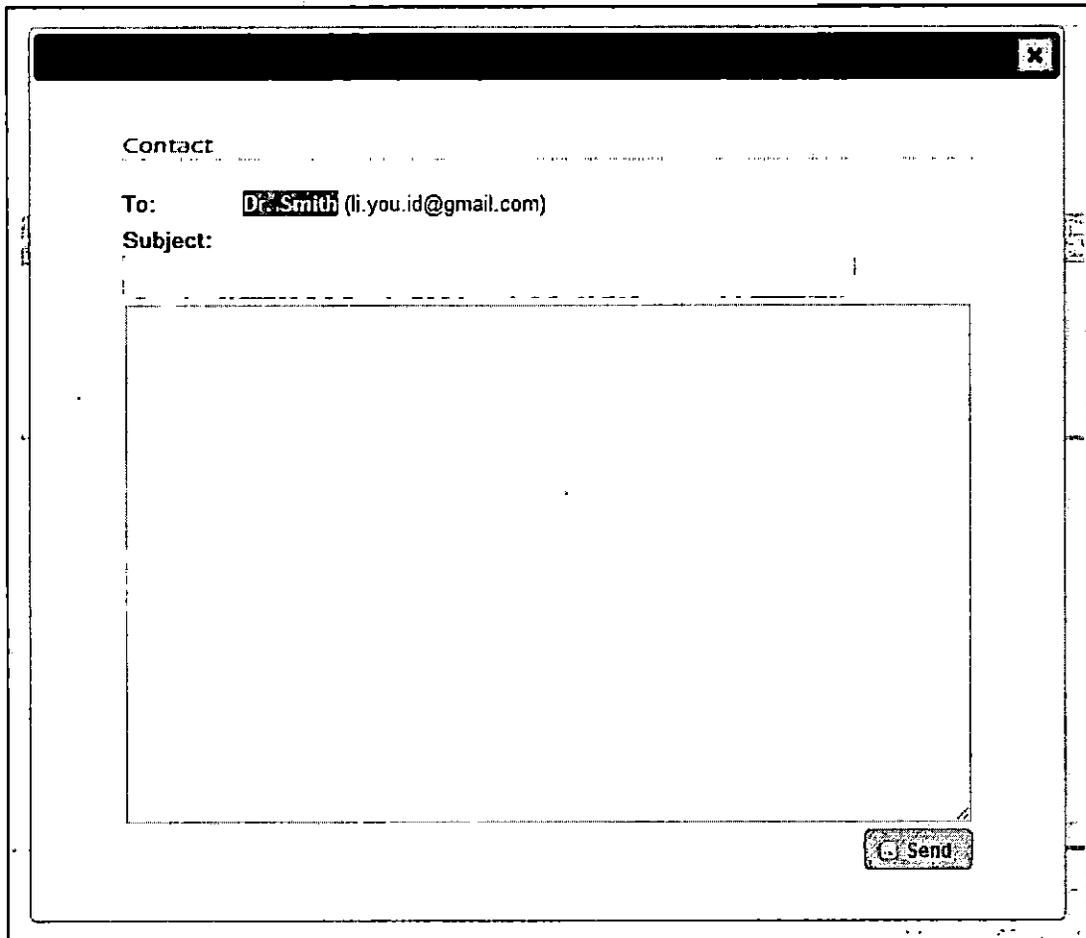
Figure 46. Interface to Contact a Recommender

5.7 Coordinator/Assistant: Print Letters of Reference and Application Form

This set of functions allows the GAA and the GAC to print submitted letter of reference forms and submitted application forms. (See Figure 47.)

First Name	M.I.	Last Name	Resident	Start Term	Application Received	Recommendation	Official GRE scores ived	Official Language scores Received	Actions
Boa		Kim	No	SP14		Print Letter of Reference			
Leo	J	Lee	Yes	W14		David Dr. Jason Yes		Print Application Form	

Figure 47. Icons to Print Letters of Reference and Application Form

Both forms will be automatically generated in the same
layout as the current paper-based forms.

Figure 48. Print a Letter of Reference

Figure 49. Print an Application Form

5.8 Coordinator: Evaluation

This function is available to the GAC only. It allows the GAC to evaluate an application and the related recommendations. Five quick search buttons are provided on the left side: Admit Classified, Conditionally Classified, Probation Conditional, Probation Unclassified, and Deny. These buttons will indicate the admit decision made by the GAC
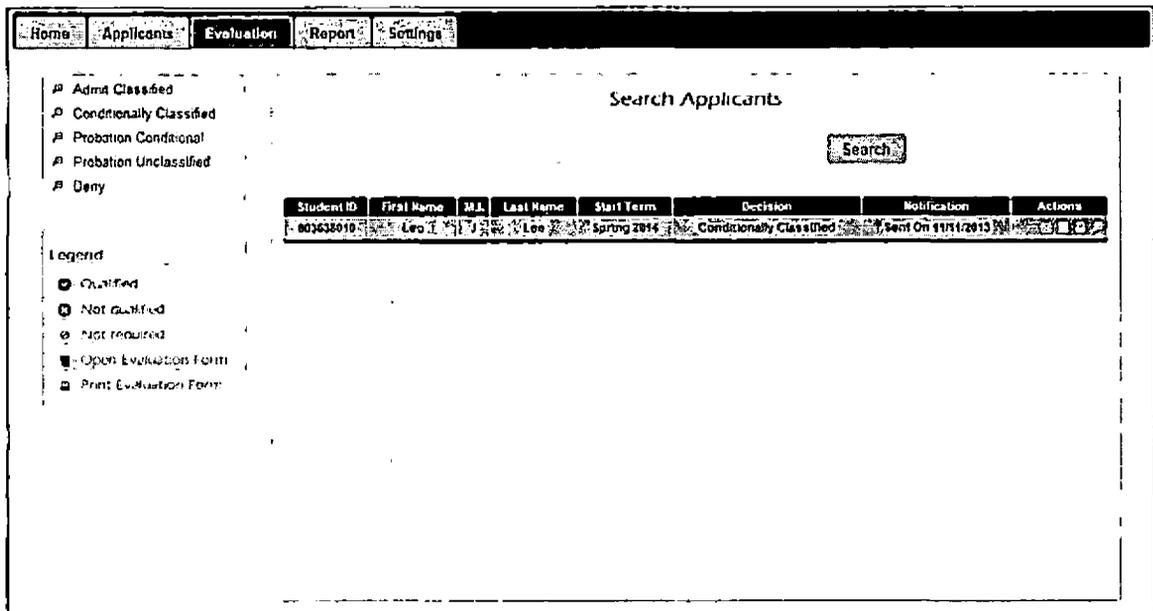
Figure 50. Interface to Evaluate Each Document

5.9 Coordinator: Admit Decision

This set of function is available to the GAC only. It allows the GAC to submit admission decision, generate graduate decision form, and send admission decision letter. The window for submitting admission decision can be opened by clicking the "Note" icon on the Action column. (See Figure 51.)

Figure 51. "Note" Icon to Open the Admission Evaluation Form

The Student Id input field in the Admission Evaluation

Form is automatically filled by the system.

Figure 52. Interface to Submit Admit Decisions

5.10 Coordinator/Assistant: Statistic Reports

This set of functions provides statistic information by generating corresponding reports.

Figure 53. Interface to View Statistic Report

5.11 Coordinator/Assistant: System Settings

This page provides the GAA and the GAC interfaces to set up the admission requirements (the minimum GRE score requirement, the minimum TOEFL score requirement for each test type, and the minimum IELTS score for 6.0 band), the application deadlines, and the templates of emails and letters.
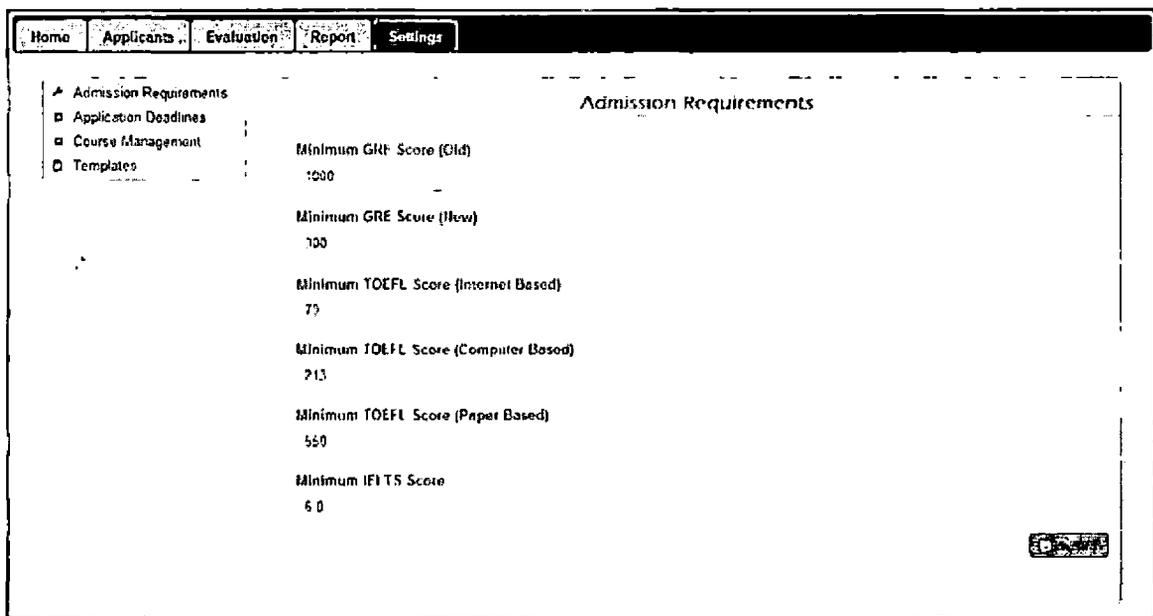
Figure 54. Interface to Change System Settings

CHAPTER SIX

INSTALLATION

6.1 System Requirements

OGAS requires two types of servers: the application

server and the database server. The application server runs

Apache 2.4.4 as the web server and PHP 5.4.19 as the

application engine while the database server runs the MySQL

5.5.32 database system.

6.2 Installation

The installation of OGAS is done in two major steps:

the application installation and the database installation.

6.2.1 Application Installation

The following steps must be done to install the application

program of OGAS on the application server:

1.   Copy all the files in the root directory on the

       CD to the web root directory.

2.   Open the "database_settings.php" file in the

       \system directory on the web server.

3.   Edit the database host name, the database name,

       the database user name, and the database

password in the "database_settings.php" file so it points to the correct database server.

6.2.2 Database Installation

The steps below show the sequence to install and configure the database on the database server.

1. Install MySQL database system on the database server.

2. Login into the database administration system using the username and password set in the step 1.

3. Add a new user and set up the privileges for this user.

4. Import the "create_database.sql" file in the root directory on the CD to create tables and import data to the newly created database.

CHAPTER SEVEN

CONCLUSIONS

7.1 Project Accomplishments

OGAS will provide benefits for both applicants and the CSE graduate admission staff. For the applicants, it facilitates submitting and managing an application to the Master's Program in the CSE. For the staff, it will significantly simplify processing an application. The most significant features and advantages are:

1. An incomplete application can be saved so applicants can finish their application in multiple sessions.

2. The system generates different interfaces for domestic/resident students and international students.

3. The status of the application process is accessible online by the applicants.

4. If applicants submitted incorrect email address for a recommender, or need to replace a reference that has already been submitted, the

applicants can update these information online at any time.

5. A recommender can submit or upload a letter of reference online. A submitted letter of reference can be printed out by the recommender, the CSE graduate admissions staff, and by the applicant if did not waive the right to read.

6. The GAA and the GAC can monitor and update the status of an application.

7. The GAC can evaluate an application and related recommendations and give an admit/deny decision for applicants into the Master's Program in the CSE.

8. A pre-validation mechanism is employed to ensure the quality of input data before being stored in the database. Unlike validating at the time of submitting, the pre-validation functions will check each input value right after the value is entered. This will skip the step to validate each input data at the time of save or submission.

9.   The system will track incomplete applications
      and send appropriate notices to the applicant
      and/or recommenders if further action is
      required.

10.   The system will be able to generate all forms
      in the same layout as the current paper-based
      forms.

7.2 Future Directions

The Online Graduate Application System was designed
based on the idea of a content-free structure which
dynamically saves data into as well as loads content from
the database. This design will enable OGAS to easily be
adapted by other schools or departments. A user interface
will need to be configured for the application component of
OGAS to comply with the requirements by another school or
department. The following are suggestions for future
implementations:

- Provide an interface to capture the signature for
  documents and forms. The signature will be done on
  a browser window using the mouse and the interface
  will save the signature image.

- Improve the usability of the system by

    o interfacing OGAS with the Student Management

    System maintained by CSUSB using PeopleSoft,

    o allowing recommenders to be able to replace an

    previously uploaded file,

    o combining the recommender's uploaded file with

    the system-generated letter of recommendation.

- Allow the ability to attach a file to system-

    generated emails.

- Add the ability to search the questions database

    using phrases (e.g. minimum GRE score) in addition

    to standard keywords that are currently searchable

    through the question title.

APPENDIX A

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

MASTERS APPLICATION

Application for Admission to
the Master of Science in Computer Science Degree Program
at California State University San Bernardino

Complete and return this form to the following address:

> School of Computer Science and Engineering
> California State University San Bernardino
> 5500 University Parkway
> San Bernardino, CA 92407-2397

Note: In addition to submitting this application, you must also apply for admission to the University.

For which term and year are you applying for admission? Fall Winter Spring (Circle one)

Year _____

Name: _____

Mailing Address: _____

_____

_____

Primary Phone: _____ Secondary Phone: _____

E-Mail Address: _____

Are you a citizen or permanent resident of the U.S.? Yes No (Circle one)

Schools Attended	Years		GPA	Degree Obtained/Expected
	From	To		

GRE scores, if already taken:

Exam Date: _____ Verbal: _____ Quantitative: _____ Analytical: _____

Send a copy of GRE test scores to the School of Computer Science and Engineering (address given above).

TOEFL Score (required for international students): _____ Date Taken: _____

Signature _____ Date _____

Attach a brief statement (no more than 400 words) on your reasons for pursuing the M.S. Degree in Computer Science. Include any additional information concerning your preparation pertinent to the M.S. Degree.

Revision: August 5, 2010 - ml

100

Name: _____

Write a brief statement (no more than 400 words) on your reasons for pursuing the M.S. Degree in Computer Science. Include any additional information concerning your preparation pertinent to the M.S. Degree.

LETTER OF REFERENCE
School of Computer Science and Engineering
California State University, San Bernardino
5500 University Parkway, San Bernardino, CA 92407-2397

To the Applicant: Please complete information requested below and give this form to a person familiar with your educational/professional background and abilities (Please type).
Name of Applicant (Last, First, MI)
Address   Phone (   )
Choose one: I do not waive my right to review this letter of reference_____   I do waive my right to review this letter of reference _____ Waiver of access to statements of recommendation is optional. I understand that letters and statements of recommendation concerning me are to be received and maintained by CSUSB, School of Computer Science and Engineering, and I hereby expressly and voluntarily waive any and all access rights I might have to such recommendations under the Federal Family Rights and Privacy Act, the California Information Act, or the laws, regulations, or policies. Signature _____ Date _____

To the Recommender: Persons who have applied for admissions to CSUSB, whether or not they have been accepted for admission and enrolled therein, shall be permitted to inspect and review letters and statements of recommendations to which they have not waived their right of access (see above student waiver of access).

How long have you known the Applicant?
In what relationship?
Name of Reference: Position/Institution   Phone   (   )
Address   E-mail:
Signature _____ Date _____

Please rate the applicant in comparison with others of the same age and position whom you have known.

	Upper 1 or 2%	Upper 10% but not upper 1 or 2%	Upper 25% but not upper 10%	Upper half but not upper 25%	Lower half	No basis for judgement
Native Intellectual Ability						
Imagination and Probable Creativity						
Breadth of General Knowledge						
Ability in Oral Expression						
Writing Ability						
Emotional Maturity						
Promise as a Teacher						
Overall recommendation as to promise as a professional in the field						

Overall Recommendation (check one): (use reverse side for comments if necessary)
_____ I recommend the applicant without reservation as an excellent prospect.
_____ I have some reservations, but would recommend the applicant as a good prospect.
_____ I have substantial doubts, but think the applicant should be given a chance to prove him/herself.
_____ I feel the applicant is not suited for the program.

RETURN THIS FORM TO THE ADDRESS AT THE TOP OF THIS REFERENCE LETTER
Revision: August 5, 2010 - ml

APPENDIX B

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

APPLICATION EVALUATION FORM

Admission Evaluation Form
M.S. in Computer Science
School of Computer Science and Engineering
California State University, San Bernardino

Name: _____ SID#: _____ - ____ - _____ Date: _____

Address: _____ City: _____

State: _____ Zip Code: _____ E-Mail: _____

Home Phone: _____ Work Phone: _____

International Student [   ]   CA Resident [   ]

1. Degrees Earned:

Degree	Institution	Graduation Date	GPA

2. Letters of Recommendation:

Recommender	Institution	Position	Comment

3. GRE:

Date: _____ Verbal: _____ Quantitative: _____ Analytical: _____

4. TOEFL: (International Students ONLY)

Date: _____ Score: _____ (Minimum 550/213)

Updated: 2011

104

5. Statement of Purpose

    Excellent: _____   Very Good: _____   Good: _____   Fair: _____   Poor: _____

6. Transfer Courses: (If Any)

Institution	Course Number and Title	Term/Year

7. Graduate Committee Decision:

    A. Accept as:

        Graduate Classified: _____

        Graduate Conditionally Classified: _____

            Remove deficiencies in:

                CSE 201: _____   CSE 202: _____   CSE 330: _____

                Math 211: _____   Math 212: _____   Math 272: _____

                GRE: _____

                Other:

                    CSE 310: _____   CSE 455: _____

                    CSE 313: _____   CSE 460: _____

                    CSE 401: _____

                    NSci 306: _____   (Graduate Entrance Writing Requirement)

                    [WREE >= 5.0 GMAT Writing Score]

    B. Reject: _____

        Reason: _____

                  _____

                  _____

                  _____

Graduate Coordinator: _____   Date: _____

APPENDIX C

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

GRADUATE DECISION FORM

CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO
GRADUATE DECISION FORM

TO:_____ _____
                                                                    STUDENT NAME

GRADUATE PROGRAM:_____ _____
                                                          STUDENT'S SSN

   FALL____WINTER____SPRING____ TODAY'S DATE:_____

____ADMIT CLASSIFIED

____ADMIT CONDITIONALLY CLASSIFIED
     Conditions to be met for classified status:

     _____

                                   .

     _____

     _____

SPECIAL ADMISSION (major code will be of the graduate program)

____PROBATION/CONDITIONAL ____PROBATION/UNCLASSIFIED

     _____

     _____

     _____

____DENY REASON_____

     _____

     _____

STUDENTS WITH LESS THAN A 2.50 IN THEIR LAST 90 QUARTER UNITS HAVE BEEN DENIED BY THE ADMISSIONS OFFICE
AND SENT A LETTER OF DENIAL. THE FILE HAS BEEN SENT TO YOU FOR POSSIBLE SPECIAL ADMISSION.

PLEASE RETAIN ATTACHED DOCUMENTS. YOU MAY NEED TO REFER TO THESE DOCUMENTS AT A LATER TIME.
NOTE: IF THERE IS AN OPTION-SPECIFY_____

_____   _____
DATE   SIGNATURE OF COORDINATOR

_____   _____
DATE   SIGNATURE OF STUDENT-I UNDERSTAND THE CONDITIONS OF MY SPECIAL ADMISSION

DISTRIBUTION: WHITE-ADMISSIONS/ CANARY-GRAD STUDIES/ PINK-COORDINATOR/ GOLDENROD-STUDENT
REVISED 10/96

107

APPENDIX D

DATABASE CREATION SCRIPTS

```sql
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";

--
-- Database: `csega`

--
CREATE DATABASE IF NOT EXISTS `csega` DEFAULT CHARACTER
SET latin1 COLLATE latin1_swedish_ci;
USE `csega`;
--


-- ------------------------------------------------------------

-

--
-- Table structure for table `applicants`
--

CREATE TABLE `applicants` (
  `id` int(11) NOT NULL auto_increment,
  `password` varchar(256) NOT NULL,
  `salt` varchar(16) NOT NULL,
  `studenttype` varchar(20) NOT NULL,
  `firstname` varchar(20) NOT NULL,
  `middleinitial` varchar(2) NOT NULL,
  `lastname` varchar(20) NOT NULL,
  `email` varchar(100) NOT NULL,
  `studentID` varchar(10) NOT NULL,
  `dateofbirth` varchar(12) NOT NULL,
  `gender` varchar(6) NOT NULL,
  `ethnicity` varchar(20) NOT NULL,
  `mailingstreetaddressline1` varchar(200) NOT NULL,
  `mailingstreetaddressline2` varchar(100) NOT NULL,
  `country` varchar(50) NOT NULL,
  `state` varchar(50) NOT NULL,
  `city` varchar(50) NOT NULL,
  `postalcode` varchar(20) NOT NULL,
  `primaryphonenumber` varchar(20) NOT NULL,
  `secondaryphonenumber` varchar(20) NOT NULL,
  `programname` varchar(50) NOT NULL,
  `startterm` varchar(20) NOT NULL,
  `GPA` varchar(7) NOT NULL,
  `academichonors` varchar(2000) NOT NULL,
```

```
  `activities` varchar(2000) NOT NULL,
  `publications` varchar(2000) NOT NULL,
  `GREdate` varchar(10) NOT NULL,
  `verbal` varchar(5) NOT NULL,
  `quantitative` varchar(5) NOT NULL,
  `analyticalwriting` varchar(5) NOT NULL,
  `TOEFLdate` varchar(10) NOT NULL,
  `TOEFLscore` varchar(5) NOT NULL,
  `IELTSdate` varchar(10) NOT NULL,
  `IELTSscore` varchar(5) NOT NULL,
  `statementofpurpose` varchar(5000) NOT NULL,
  `transcripts` varchar(20) NOT NULL default 'not
received',
  `officialGRE` varchar(20) NOT NULL default 'not
received',
  `officialTOEFL` varchar(20) NOT NULL default 'not
received',
  `officialIELTS` varchar(20) NOT NULL default 'not
received',
  `visits` int(11) NOT NULL default '1',
  `lastvisit` timestamp NOT NULL default CURRENT_TIMESTAMP
on update CURRENT_TIMESTAMP,
  `finished` varchar(20) default NULL,
  `status` int(11) NOT NULL default '0',
  `active` int(11) NOT NULL default '0',
  `hash` varchar(32) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=28 ;


--
-- Dumping data for table `applicants`
--

-- ------------------------------------------------------------
-


--
-- Table structure for table `application_fields`
--

CREATE TABLE `application_fields` (
  `id` int(11) NOT NULL auto_increment,
  `section` int(11) NOT NULL,
  `name` varchar(50) NOT NULL,
```

```
    `type` varchar(20) NOT NULL,
    `class` varchar(300) NOT NULL,
    `hint` varchar(300) NOT NULL,
    PRIMARY KEY  (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=35
;


--
-- Dumping data for table `application_fields`
--


INSERT INTO `application_fields` VALUES(1, 7, 'first
name', 'text', 'textInput required', '');
INSERT INTO `application_fields` VALUES(2, 7, 'middle
initial', 'text', 'textInput', '');
INSERT INTO `application_fields` VALUES(3, 7, 'last name',
'text', 'textInput required', '');
INSERT INTO `application_fields` VALUES(4, 1, 'student ID
', 'text', 'textInput required validateInteger ', 'Your
coyote ID number.');
INSERT INTO `application_fields` VALUES(5, 8, 'date of
birth', 'text', 'textInput required datepicker',
'MM/DD/YYYY');
INSERT INTO `application_fields` VALUES(6, 8, 'gender',
'select', 'required', '');
INSERT INTO `application_fields` VALUES(7, 8, 'ethnicity',
'select', 'required', '');
INSERT INTO `application_fields` VALUES(8, 9, 'mailing
street address line 1', 'text', 'textInput required', '');
INSERT INTO `application_fields` VALUES(9, 9, 'mailing
street address line 2', 'text', 'textInput', '');
INSERT INTO `application_fields` VALUES(10, 9, 'city',
'text', 'textInput required', '');
INSERT INTO `application_fields` VALUES(11, 9, 'state',
'text', 'textInput required', '');
INSERT INTO `application_fields` VALUES(12, 9, 'country',
'select', 'required', '');
INSERT INTO `application_fields` VALUES(13, 9, 'postal
code', 'text', 'textInput required', '');
INSERT INTO `application_fields` VALUES(14, 10, 'primary
phone number', 'text', 'textInput required validatePhone',
'');
INSERT INTO `application_fields` VALUES(15, 10, 'secondary
phone number', 'text', 'textInput validatePhone', '');
```

```
INSERT INTO `application_fields` VALUES(16, 11, 'program
name', 'info', 'textInput required', 'computer science');
INSERT INTO `application_fields` VALUES(17, 11, 'start
term', 'select', 'required', '');
INSERT INTO `application_fields` VALUES(18, 14, 'GRE
date', 'text', 'textInput gredate', 'MM/DD/YYYY');
INSERT INTO `application_fields` VALUES(19, 14, 'verbal',
'text', 'textInput', '');
INSERT INTO `application_fields` VALUES(20, 14,
'quantitative', 'text', 'textInput', '');
INSERT INTO `application_fields` VALUES(21, 14,
'analytical writing', 'text', 'textInput', '');
INSERT INTO `application_fields` VALUES(22, 15, 'TOEFL
date', 'text', 'textInput toefldate internationalOnly',
'MM/DD/YYYY');
INSERT INTO `application_fields` VALUES(23, 15, 'TOEFL
score', 'text', 'textInput internationalOnly', '');
INSERT INTO `application_fields` VALUES(24, 16, 'IELTS
date', 'text', 'textInput ieltsdate internationalOnly',
'MM/DD/YYYY');
INSERT INTO `application_fields` VALUES(25, 16, 'IELTS
score', 'text', 'textInput internationalOnly', '');
INSERT INTO `application_fields` VALUES(26, 12, 'GPA',
'text', 'textInput required', '');
INSERT INTO `application_fields` VALUES(27, 12, 'academic
honors', 'textarea', 'ui-corner-all mh80', '');
INSERT INTO `application_fields` VALUES(28, 12,
'activities', 'textarea', 'ui-corner-all mh80', '');
INSERT INTO `application_fields` VALUES(29, 13,
'school_attended', 'table', '', '');
INSERT INTO `application_fields` VALUES(30, 17,
'recommender', 'table', '', '');
INSERT INTO `application_fields` VALUES(31, 18, 'statement
of purpose ', 'textarea', 'required mh500', '');
INSERT INTO `application_fields` VALUES(32, 1, 'student
type', 'radio', 'required citizen', 'Are you a citizen or
permanent resident of the U.S.? ');
INSERT INTO `application_fields` VALUES(34, 12,
'publications', 'textarea', 'ui-corner-all mh80', '');


-- ------------------------------------------------------
-
--
```

```
-- Table structure for table `application_fields_options`
--

CREATE TABLE `application_fields_options` (
  `id` int(11) NOT NULL auto_increment,
  `code` varchar(20) NOT NULL,
  `name` varchar(100) NOT NULL,
  `parent` varchar(20) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=284
;


--
-- Dumping data for table `application_fields_options`
--


INSERT INTO `application_fields_options` VALUES(1, 'AIAN',
'American Indian or Alaskan Native', '7');
INSERT INTO `application_fields_options` VALUES(2, 'B',
'Black (not of Hispanic origin)', '7');
INSERT INTO `application_fields_options` VALUES(3, 'HMA',
'Hispanic: Mexican-American/Mexican/Chicano', '7');
INSERT INTO `application_fields_options` VALUES(4, 'HCA',
'Hispanic: Central American', '7');
INSERT INTO `application_fields_options` VALUES(5, 'HAS',
'Hispanic: South American', '7');
INSERT INTO `application_fields_options` VALUES(6, 'HPR',
'Hispanic: Puerto Rican', '7');
INSERT INTO `application_fields_options` VALUES(7, 'HC',
'Hispanic: Cuban', '7');
INSERT INTO `application_fields_options` VALUES(8, 'HO',
'Hispanic: Other', '7');
INSERT INTO `application_fields_options` VALUES(9, 'ACH',
'Asian: Chinese', '7');
INSERT INTO `application_fields_options` VALUES(10, 'AJ',
'Asian: Japanese', '7');
INSERT INTO `application_fields_options` VALUES(11, 'AK',
'Asian: Korean', '7');
INSERT INTO `application_fields_options` VALUES(12, 'AL',
'Asian: Laotian', '7');
INSERT INTO `application_fields_options` VALUES(13, 'ACA',
'Asian: Cambodian', '7');
INSERT INTO `application_fields_options` VALUES(14, 'AI',
'Asian: Indian', '7');
```

```
INSERT INTO `application_fields_options` VALUES(15, 'ASA',
'Asian: Southeast Asian', '7');
INSERT INTO `application_fields_options` VALUES(16, 'AT',
'Asian: Thai', '7');
INSERT INTO `application_fields_options` VALUES(17, 'AV',
'Asian: Vietnamese', '7');
INSERT INTO `application_fields_options` VALUES(18, 'AO',
'Asian: Other', '7');
INSERT INTO `application_fields_options` VALUES(19, 'PIG',
'Pacific Islanders: Guamanian', '7');
INSERT INTO `application_fields_options` VALUES(20, 'PIH',
'Pacific Islanders: Hawaiian', '7');
INSERT INTO `application_fields_options` VALUES(21, 'PIS',
'Pacific Islanders: Samoan', '7');
INSERT INTO `application_fields_options` VALUES(22, 'PIO',
'Pacific Islanders: Other', '7');
INSERT INTO `application_fields_options` VALUES(23, 'W',
'White (not of Hispanic origin)', '7');
INSERT INTO `application_fields_options` VALUES(24, 'F',
'Filipino', '7');
INSERT INTO `application_fields_options` VALUES(25, 'O',
'Other', '7');
INSERT INTO `application_fields_options` VALUES(26, 'N',
'No response', '7');
INSERT INTO `application_fields_options` VALUES(27, 'D',
'Decline to state', '7');
INSERT INTO `application_fields_options` VALUES(28,
'Female', 'Female', '6');
INSERT INTO `application_fields_options` VALUES(29,
'Male', 'Male', '6');
INSERT INTO `application_fields_options` VALUES(30, 'F13',
'Fall 2013', '17');
INSERT INTO `application_fields_options` VALUES(31, 'W14',
'Winter 2014', '17');
INSERT INTO `application_fields_options` VALUES(32,
'SP14', 'Spring 2014', '17');
INSERT INTO `application_fields_options` VALUES(33,
'SU14', 'Summer 2014', '17');
INSERT INTO `application_fields_options` VALUES(42, 'ABW',
'Aruba', '12');
INSERT INTO `application_fields_options` VALUES(43, 'AFG',
'Afghanistan', '12');
INSERT INTO `application_fields_options` VALUES(44, 'ALB',
'Albania', '12');
```

```
INSERT INTO `application_fields_options` VALUES(45, 'DZA',
'Algeria', '12');
INSERT INTO `application_fields_options` VALUES(46, 'ASM',
'American Samoa', '12');
INSERT INTO `application_fields_options` VALUES(47, 'AND',
'Andorra', '12');
INSERT INTO `application_fields_options` VALUES(48, 'AGO',
'Angola', '12');
INSERT INTO `application_fields_options` VALUES(49, 'AIA',
'Anguilla', '12');
INSERT INTO `application_fields_options` VALUES(50, 'ATA',
'Antarctica', '12');
INSERT INTO `application_fields_options` VALUES(51, 'ATG',
'Antigua and Barbuda', '12');
INSERT INTO `application_fields_options` VALUES(52, 'ARG',
'Argentina', '12');
INSERT INTO `application_fields_options` VALUES(53, 'ARM',
'Armenia', '12');
INSERT INTO `application_fields_options` VALUES(54, 'AUS',
'Australia', '12');
INSERT INTO `application_fields_options` VALUES(55, 'AUT',
'Austria', '12');
INSERT INTO `application_fields_options` VALUES(56, 'AZE',
'Azerbaijan', '12');
INSERT INTO `application_fields_options` VALUES(57, 'BHS',
'Bahamas', '12');
INSERT INTO `application_fields_options` VALUES(58, 'BHR',
'Bahrain', '12');
INSERT INTO `application_fields_options` VALUES(59, 'BGD',
'Bangladesh', '12');
INSERT INTO `application_fields_options` VALUES(60, 'BRB',
'Barbados', '12');
INSERT INTO `application_fields_options` VALUES(61, 'BLR',
'Belarus', '12');
INSERT INTO `application_fields_options` VALUES(62, 'BEL',
'Belgium', '12');
INSERT INTO `application_fields_options` VALUES(63, 'BLZ',
'Belize', '12');
INSERT INTO `application_fields_options` VALUES(64, 'BEN',
'Benin', '12');
INSERT INTO `application_fields_options` VALUES(65, 'BMU',
'Bermuda', '12');
INSERT INTO `application_fields_options` VALUES(66, 'BTN',
'Bhutan', '12');
```

```sql
INSERT INTO `application_fields_options` VALUES(67, 'BOL',
'Bolivia', '12');
INSERT INTO `application_fields_options` VALUES(68, 'BIH',
'Bosnia and Herzegovina', '12');
INSERT INTO `application_fields_options` VALUES(69, 'BWA',
'Botswana', '12');
INSERT INTO `application_fields_options` VALUES(70, 'BVT',
'Bouvet Island', '12');
INSERT INTO `application_fields_options` VALUES(71, 'BRA',
'Brazil', '12');
INSERT INTO `application_fields_options` VALUES(72, 'IOT',
'British Indian Ocean Territory', '12');
INSERT INTO `application_fields_options` VALUES(73, 'BRN',
'Brunei', '12');
INSERT INTO `application_fields_options` VALUES(74, 'BGR',
'Bulgaria', '12');
INSERT INTO `application_fields_options` VALUES(75, 'BFA',
'Burkina Faso', '12');
INSERT INTO `application_fields_options` VALUES(76, 'BDI',
'Burundi', '12');
INSERT INTO `application_fields_options` VALUES(77, 'CIV',
'C??te d?', '12');
INSERT INTO `application_fields_options` VALUES(78, 'KHM',
'Cambodia', '12');
INSERT INTO `application_fields_options` VALUES(79, 'CMR',
'Cameroon', '12');
INSERT INTO `application_fields_options` VALUES(80, 'CAN',
'Canada', '12');
INSERT INTO `application_fields_options` VALUES(81, 'CPV',
'Cape Verde', '12');
INSERT INTO `application_fields_options` VALUES(82, 'CYM',
'Cayman Islands', '12');
INSERT INTO `application_fields_options` VALUES(83, 'CAF',
'Central African Republic', '12');
INSERT INTO `application_fields_options` VALUES(84, 'TCD',
'Chad', '12');
INSERT INTO `application_fields_options` VALUES(85, 'CHL',
'Chile', '12');
INSERT INTO `application_fields_options` VALUES(86, 'CHN',
'China', '12');
INSERT INTO `application_fields_options` VALUES(87, 'CXR',
'Christmas Island', '12');
INSERT INTO `application_fields_options` VALUES(88, 'CCK',
'Cocos (Keeling) Islands', '12');
```

```
INSERT INTO `application_fields_options` VALUES(89, 'COL',
'Colombia', '12');
INSERT INTO `application_fields_options` VALUES(90, 'COM',
'Comoros', '12');
INSERT INTO `application_fields_options` VALUES(91, 'COG',
'Congo', '12');
INSERT INTO `application_fields_options` VALUES(92, 'COD',
'Congo, The Democratic Republic of the', '12');
INSERT INTO `application_fields_options` VALUES(93, 'COK',
'Cook Islands', '12');
INSERT INTO `application_fields_options` VALUES(94, 'CRI',
'Costa Rica', '12');
INSERT INTO `application_fields_options` VALUES(95, 'HRV',
'Croatia', '12');
INSERT INTO `application_fields_options` VALUES(96, 'CUB',
'Cuba', '12');
INSERT INTO `application_fields_options` VALUES(97, 'CYP',
'Cyprus', '12');
INSERT INTO `application_fields_options` VALUES(98, 'CZE',
'Czech Republic', '12');
INSERT INTO `application_fields_options` VALUES(99, 'DNK',
'Denmark', '12');
INSERT INTO `application_fields_options` VALUES(100,
'DJI', 'Djibouti', '12');
INSERT INTO `application_fields_options` VALUES(101,
'DMA', 'Dominica', '12');
INSERT INTO `application_fields_options` VALUES(102,
'DOM', 'Dominican Republic', '12');
INSERT INTO `application_fields_options` VALUES(103,
'TMP', 'East Timor', '12');
INSERT INTO `application_fields_options` VALUES(104,
'ECU', 'Ecuador', '12');
INSERT INTO `application_fields_options` VALUES(105,
'EGY', 'Egypt', '12');
INSERT INTO `application_fields_options` VALUES(106,
'SLV', 'El Salvador', '12');
INSERT INTO `application_fields_options` VALUES(107,
'GNQ', 'Equatorial Guinea', '12');
INSERT INTO `application_fields_options` VALUES(108,
'ERI', 'Eritrea', '12');
INSERT INTO `application_fields_options` VALUES(109,
'EST', 'Estonia', '12');
INSERT INTO `application_fields_options` VALUES(110,
'ETH', 'Ethiopia', '12');
```

```
INSERT INTO `application_fields_options` VALUES(111,
'FLK', 'Falkland Islands', '12');
INSERT INTO `application_fields_options` VALUES(112,
'FRO', 'Faroe Islands', '12');
INSERT INTO `application_fields_options` VALUES(113,
'FJI', 'Fiji Islands', '12');
INSERT INTO `application_fields_options` VALUES(114,
'FIN', 'Finland', '12');
INSERT INTO `application_fields_options` VALUES(115,
'FRA', 'France', '12');
INSERT INTO `application_fields_options` VALUES(116,
'GUF', 'French Guiana', '12');
INSERT INTO `application_fields_options` VALUES(117,
'PYF', 'French Polynesia', '12');
INSERT INTO `application_fields_options` VALUES(118,
'ATF', 'French Southern territories', '12');
INSERT INTO `application_fields_options` VALUES(119,
'GAB', 'Gabon', '12');
INSERT INTO `application_fields_options` VALUES(120,
'GMB', 'Gambia', '12');
INSERT INTO `application_fields_options` VALUES(121,
'GEO', 'Georgia', '12');
INSERT INTO `application_fields_options` VALUES(122,
'DEU', 'Germany', '12');
INSERT INTO `application_fields_options` VALUES(123,
'GHA', 'Ghana', '12');
INSERT INTO `application_fields_options` VALUES(124,
'GIB', 'Gibraltar', '12');
INSERT INTO `application_fields_options` VALUES(125,
'GRC', 'Greece', '12');
INSERT INTO `application_fields_options` VALUES(126,
'GRL', 'Greenland', '12');
INSERT INTO `application_fields_options` VALUES(127,
'GRD', 'Grenada', '12');
INSERT INTO `application_fields_options` VALUES(128,
'GLP', 'Guadeloupe', '12');
INSERT INTO `application_fields_options` VALUES(129,
'GUM', 'Guam', '12');
INSERT INTO `application_fields_options` VALUES(130,
'GTM', 'Guatemala', '12');
INSERT INTO `application_fields_options` VALUES(131,
'GIN', 'Guinea', '12');
INSERT INTO `application_fields_options` VALUES(132,
'GNB', 'Guinea-Bissau', '12');
```

```
INSERT INTO `application_fields_options` VALUES(133,
'GUY', 'Guyana', '12');
INSERT INTO `application_fields_options` VALUES(134,
'HTI', 'Haiti', '12');
INSERT INTO `application_fields_options` VALUES(135,
'HMD', 'Heard Island and McDonald Islands', '12');
INSERT INTO `application_fields_options` VALUES(136,
'VAT', 'Holy See (Vatican City State)', '12');
INSERT INTO `application_fields_options` VALUES(137,
'HND', 'Honduras', '12');
INSERT INTO `application_fields_options` VALUES(138,
'HKG', 'Hong Kong', '12');
INSERT INTO `application_fields_options` VALUES(139,
'HUN', 'Hungary', '12');
INSERT INTO `application_fields_options` VALUES(140,
'ISL', 'Iceland', '12');
INSERT INTO `application_fields_options` VALUES(141,
'IND', 'India', '12');
INSERT INTO `application_fields_options` VALUES(142,
'IDN', 'Indonesia', '12');
INSERT INTO `application_fields_options` VALUES(143,
'IRN', 'Iran', '12');
INSERT INTO `application_fields_options` VALUES(144,
'IRQ', 'Iraq', '12');
INSERT INTO `application_fields_options` VALUES(145,
'IRL', 'Ireland', '12');
INSERT INTO `application_fields_options` VALUES(146,
'ISR', 'Israel', '12');
INSERT INTO `application_fields_options` VALUES(147,
'ITA', 'Italy', '12');
INSERT INTO `application_fields_options` VALUES(148,
'JAM', 'Jamaica', '12');
INSERT INTO `application_fields_options` VALUES(149,
'JPN', 'Japan', '12');
INSERT INTO `application_fields_options` VALUES(150,
'JOR', 'Jordan', '12');
INSERT INTO `application_fields_options` VALUES(151,
'KAZ', 'Kazakstan', '12');
INSERT INTO `application_fields_options` VALUES(152,
'KEN', 'Kenya', '12');
INSERT INTO `application_fields_options` VALUES(153,
'KIR', 'Kiribati', '12');
INSERT INTO `application_fields_options` VALUES(154,
'KWT', 'Kuwait', '12');
```

```
INSERT INTO `application_fields_options` VALUES(155,
'KGZ', 'Kyrgyzstan', '12');
INSERT INTO `application_fields_options` VALUES(156,
'LAO', 'Laos', '12');
INSERT INTO `application_fields_options` VALUES(157,
'LVA', 'Latvia', '12');
INSERT INTO `application_fields_options` VALUES(158,
'LBN', 'Lebanon', '12');
INSERT INTO `application_fields_options` VALUES(159,
'LSO', 'Lesotho', '12');
INSERT INTO `application_fields_options` VALUES(160,
'LBR', 'Liberia', '12');
INSERT INTO `application_fields_options` VALUES(161,
'LBY', 'Libyan Arab Jamahiriya', '12');
INSERT INTO `application_fields_options` VALUES(162,
'LIE', 'Liechtenstein', '12');
INSERT INTO `application_fields_options` VALUES(163,
'LTU', 'Lithuania', '12');
INSERT INTO `application_fields_options` VALUES(164,
'LUX', 'Luxembourg', '12');
INSERT INTO `application_fields_options` VALUES(165,
'MAC', 'Macao', '12');
INSERT INTO `application_fields_options` VALUES(166,
'MKD', 'Macedonia', '12');
INSERT INTO `application_fields_options` VALUES(167,
'MDG', 'Madagascar', '12');
INSERT INTO `application_fields_options` VALUES(168,
'MWI', 'Malawi', '12');
INSERT INTO `application_fields_options` VALUES(169,
'MYS', 'Malaysia', '12');
INSERT INTO `application_fields_options` VALUES(170,
'MDV', 'Maldives', '12');
INSERT INTO `application_fields_options` VALUES(171,
'MLI', 'Mali', '12');
INSERT INTO `application_fields_options` VALUES(172,
'MLT', 'Malta', '12');
INSERT INTO `application_fields_options` VALUES(173,
'MHL', 'Marshall Islands', '12');
INSERT INTO `application_fields_options` VALUES(174,
'MTQ', 'Martinique', '12');
INSERT INTO `application_fields_options` VALUES(175,
'MRT', 'Mauritania', '12');
INSERT INTO `application_fields_options` VALUES(176,
'MUS', 'Mauritius', '12');
```

```
INSERT INTO `application_fields_options` VALUES(177,
'MYT', 'Mayotte', '12');
INSERT INTO `application_fields_options` VALUES(178,
'MEX', 'Mexico', '12');
INSERT INTO `application_fields_options` VALUES(179,
'FSM', 'Micronesia, Federated States of', '12');
INSERT INTO `application_fields_options` VALUES(180,
'MDA', 'Moldova', '12');
INSERT INTO `application_fields_options` VALUES(181,
'MCO', 'Monaco', '12');
INSERT INTO `application_fields_options` VALUES(182,
'MNG', 'Mongolia', '12');
INSERT INTO `application_fields_options` VALUES(183,
'MSR', 'Montserrat', '12');
INSERT INTO `application_fields_options` VALUES(184,
'MAR', 'Morocco', '12');
INSERT INTO `application_fields_options` VALUES(185,
'MOZ', 'Mozambique', '12');
INSERT INTO `application_fields_options` VALUES(186,
'MMR', 'Myanmar', '12');
INSERT INTO `application_fields_options` VALUES(187,
'NAM', 'Namibia', '12');
INSERT INTO `application_fields_options` VALUES(188,
'NRU', 'Nauru', '12');
INSERT INTO `application_fields_options` VALUES(189,
'NPL', 'Nepal', '12');
INSERT INTO `application_fields_options` VALUES(190,
'NLD', 'Netherlands', '12');
INSERT INTO `application_fields_options` VALUES(191,
'ANT', 'Netherlands Antilles', '12');
INSERT INTO `application_fields_options` VALUES(192,
'NCL', 'New Caledonia', '12');
INSERT INTO `application_fields_options` VALUES(193,
'NZL', 'New Zealand', '12');
INSERT INTO `application_fields_options` VALUES(194,
'NIC', 'Nicaragua', '12');
INSERT INTO `application_fields_options` VALUES(195,
'NER', 'Niger', '12');
INSERT INTO `application_fields_options` VALUES(196,
'NGA', 'Nigeria', '12');
INSERT INTO `application_fields_options` VALUES(197,
'NIU', 'Niue', '12');
INSERT INTO `application_fields_options` VALUES(198,
'NFK', 'Norfolk Island', '12');
```

```
INSERT INTO `application_fields_options` VALUES(199,
'PRK', 'North Korea', '12');
INSERT INTO `application_fields_options` VALUES(200,
'MNP', 'Northern Mariana Islands', '12');
INSERT INTO `application_fields_options` VALUES(201,
'NOR', 'Norway', '12');
INSERT INTO `application_fields_options` VALUES(202,
'OMN', 'Oman', '12');
INSERT INTO `application_fields_options` VALUES(203,
'PAK', 'Pakistan', '12');
INSERT INTO `application_fields_options` VALUES(204,
'PLW', 'Palau', '12');
INSERT INTO `application_fields_options` VALUES(205,
'PSE', 'Palestine', '12');
INSERT INTO `application_fields_options` VALUES(206,
'PAN', 'Panama', '12');
INSERT INTO `application_fields_options` VALUES(207,
'PNG', 'Papua New Guinea', '12');
INSERT INTO `application_fields_options` VALUES(208,
'PRY', 'Paraguay', '12');
INSERT INTO `application_fields_options` VALUES(209,
'PER', 'Peru', '12');
INSERT INTO `application_fields_options` VALUES(210,
'PHL', 'Philippines', '12');
INSERT INTO `application_fields_options` VALUES(211,
'PCN', 'Pitcairn', '12');
INSERT INTO `application_fields_options` VALUES(212,
'POL', 'Poland', '12');
INSERT INTO `application_fields_options` VALUES(213,
'PRT', 'Portugal', '12');
INSERT INTO `application_fields_options` VALUES(214,
'PRI', 'Puerto Rico', '12');
INSERT INTO `application_fields_options` VALUES(215,
'QAT', 'Qatar', '12');
INSERT INTO `application_fields_options` VALUES(216,
'REU', 'R??union', '12');
INSERT INTO `application_fields_options` VALUES(217,
'ROM', 'Romania', '12');
INSERT INTO `application_fields_options` VALUES(218,
'RUS', 'Russian Federation', '12');
INSERT INTO `application_fields_options` VALUES(219,
'RWA', 'Rwanda', '12');
INSERT INTO `application_fields_options` VALUES(220,
'SHN', 'Saint Helena', '12');
```

```sql
INSERT INTO `application_fields_options` VALUES(221,
'KNA', 'Saint Kitts and Nevis', '12');
INSERT INTO `application_fields_options` VALUES(222,
'LCA', 'Saint Lucia', '12');
INSERT INTO `application_fields_options` VALUES(223,
'SPM', 'Saint Pierre and Miquelon', '12');
INSERT INTO `application_fields_options` VALUES(224,
'VCT', 'Saint Vincent and the Grenadines', '12');
INSERT INTO `application_fields_options` VALUES(225,
'WSM', 'Samoa', '12');
INSERT INTO `application_fields_options` VALUES(226,
'SMR', 'San Marino', '12');
INSERT INTO `application_fields_options` VALUES(227,
'STP', 'Sao Tome and Principe', '12');
INSERT INTO `application_fields_options` VALUES(228,
'SAU', 'Saudi Arabia', '12');
INSERT INTO `application_fields_options` VALUES(229,
'SEN', 'Senegal', '12');
INSERT INTO `application_fields_options` VALUES(230,
'SYC', 'Seychelles', '12');
INSERT INTO `application_fields_options` VALUES(231,
'SLE', 'Sierra Leone', '12');
INSERT INTO `application_fields_options` VALUES(232,
'SGP', 'Singapore', '12');
INSERT INTO `application_fields_options` VALUES(233,
'SVK', 'Slovakia', '12');
INSERT INTO `application_fields_options` VALUES(234,
'SVN', 'Slovenia', '12');
INSERT INTO `application_fields_options` VALUES(235,
'SLB', 'Solomon Islands', '12');
INSERT INTO `application_fields_options` VALUES(236,
'SOM', 'Somalia', '12');
INSERT INTO `application_fields_options` VALUES(237,
'ZAF', 'South Africa', '12');
INSERT INTO `application_fields_options` VALUES(238,
'SGS', 'South Georgia and the South Sandwich Islands',
'12');
INSERT INTO `application_fields_options` VALUES(239,
'KOR', 'South Korea', '12');
INSERT INTO `application_fields_options` VALUES(240,
'ESP', 'Spain', '12');
INSERT INTO `application_fields_options` VALUES(241,
'LKA', 'Sri Lanka', '12');
```

```
INSERT INTO `application_fields_options` VALUES(242,
'SDN', 'Sudan', '12');
INSERT INTO `application_fields_options` VALUES(243,
'SUR', 'Suriname', '12');
INSERT INTO `application_fields_options` VALUES(244,
'SJM', 'Svalbard and Jan Mayen', '12');
INSERT INTO `application_fields_options` VALUES(245,
'SWZ', 'Swaziland', '12');
INSERT INTO `application_fields_options` VALUES(246,
'SWE', 'Sweden', '12');
INSERT INTO `application_fields_options` VALUES(247,
'CHE', 'Switzerland', '12');
INSERT INTO `application_fields_options` VALUES(248,
'SYR', 'Syria', '12');
INSERT INTO `application_fields_options` VALUES(249,
'TWN', 'Taiwan', '12');
INSERT INTO `application_fields_options` VALUES(250,
'TJK', 'Tajikistan', '12');
INSERT INTO `application_fields_options` VALUES(251,
'TZA', 'Tanzania', '12');
INSERT INTO `application_fields_options` VALUES(252,
'THA', 'Thailand', '12');
INSERT INTO `application_fields_options` VALUES(253,
'TGO', 'Togo', '12');
INSERT INTO `application_fields_options` VALUES(254,
'TKL', 'Tokelau', '12');
INSERT INTO `application_fields_options` VALUES(255,
'TON', 'Tonga', '12');
INSERT INTO `application_fields_options` VALUES(256,
'TTO', 'Trinidad and Tobago', '12');
INSERT INTO `application_fields_options` VALUES(257,
'TUN', 'Tunisia', '12');
INSERT INTO `application_fields_options` VALUES(258,
'TUR', 'Turkey', '12');
INSERT INTO `application_fields_options` VALUES(259,
'TKM', 'Turkmenistan', '12');
INSERT INTO `application_fields_options` VALUES(260,
'TCA', 'Turks and Caicos Islands', '12');
INSERT INTO `application_fields_options` VALUES(261,
'TUV', 'Tuvalu', '12');
INSERT INTO `application_fields_options` VALUES(262,
'UGA', 'Uganda', '12');
INSERT INTO `application_fields_options` VALUES(263,
'UKR', 'Ukraine', '12');
```

```
INSERT INTO `application_fields_options` VALUES(264,
'ARE', 'United Arab Emirates', '12');
INSERT INTO `application_fields_options` VALUES(265,
'GBR', 'United Kingdom', '12');
INSERT INTO `application_fields_options` VALUES(266,
'USA', 'United States', '12');
INSERT INTO `application_fields_options` VALUES(267,
'UMI', 'United States Minor Outlying Islands', '12');
INSERT INTO `application_fields_options` VALUES(268,
'URY', 'Uruguay', '12');
INSERT INTO `application_fields_options` VALUES(269,
'UZB', 'Uzbekistan', '12');
INSERT INTO `application_fields_options` VALUES(270,
'VUT', 'Vanuatu', '12');
INSERT INTO `application_fields_options` VALUES(271,
'VEN', 'Venezuela', '12');
INSERT INTO `application_fields_options` VALUES(272,
'VNM', 'Vietnam', '12');
INSERT INTO `application_fields_options` VALUES(273,
'VGB', 'Virgin Islands, British', '12');
INSERT INTO `application_fields_options` VALUES(274,
'VIR', 'Virgin Islands, U.S.', '12');
INSERT INTO `application_fields_options` VALUES(275,
'WLF', 'Wallis and Futuna', '12');
INSERT INTO `application_fields_options` VALUES(276,
'ESH', 'Western Sahara', '12');
INSERT INTO `application_fields_options` VALUES(277,
'YEM', 'Yemen', '12');
INSERT INTO `application_fields_options` VALUES(278,
'YUG', 'Yugoslavia', '12');
INSERT INTO `application_fields_options` VALUES(279,
'ZMB', 'Zambia', '12');
INSERT INTO `application_fields_options` VALUES(280,
'ZWE', 'Zimbabwe', '12');
INSERT INTO `application_fields_options` VALUES(281, 'cs',
'Computer Science', '16');
INSERT INTO `application_fields_options` VALUES(282,
'yes', 'yes', '32');
INSERT INTO `application_fields_options` VALUES(283, 'no',
'no', '32');


-- ------------------------------------------------------------
-
```

```
--                      .                .
-- Table structure for table `application_sections`
--

CREATE TABLE `application_sections` (
  `id` int(11) NOT NULL auto_increment,
  `name` varchar(100) NOT NULL,
  `parent` int(11) NOT NULL default '-1',
  `content` varchar(500) NOT NULL,
  `class` varchar(20) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=20
;


--
-- Dumping data for table `application_sections`
--


INSERT INTO `application_sections` VALUES(1, 'biographical
information', 0, 'You will get your student ID after
applying to the university. If you have not done that yet,
please click <a
href="http://www.csumentor.edu/admissionapp/grad_apply.asp
" target="_blank" style="text-decoration:underline;font-
style:italic">here</a> to finish it first.', '');
INSERT INTO `application_sections` VALUES(2, 'contact
information', 0, '', '');
INSERT INTO `application_sections` VALUES(3, 'program
information', 0, '', '');
INSERT INTO `application_sections` VALUES(4, 'academic
background', 0, '', '');
INSERT INTO `application_sections` VALUES(5, 'test
information', 0, 'Please provide us your test information
the same as they are on the official transcripts. GRE is
required for all applicants. If you have not taken the
test or received the score yet, please leave them blank.
You can mail us your official transcripts later. <div
class="ui-state-default ui-corner-all"
style="padding:5px;">* Please Send a copy of each test
scores to the School of Computer Science and Engineering
(address given on the bottom of the page).</div>', '');
INSERT INTO `application_sections` VALUES(6,
'recommendations', 0, '', '');
```

126

```
INSERT INTO `application_sections` VALUES(7,
'identification', 1, 'Please use your legal name. Do not
use nicknames or abbreviations. <em><a>International
students:</a> Type your name exactly as it appears on your
passport.</em>', '');
INSERT INTO `application_sections` VALUES(8, 'demographic
information', 1, 'An international student is either
holding one of the following visas: F1, B1, B2, H4, or
L2.', '');
INSERT INTO `application_sections` VALUES(9, 'current
address', 2, 'For addresses outside of the United States,
please use all available space on both street address
lines and the city line to include the province, territory
and postal code.', '');
INSERT INTO `application_sections` VALUES(10, 'Telephone
', 2, '', '');
INSERT INTO `application_sections` VALUES(11, 'Program of
Study', 3, '', '');
INSERT INTO `application_sections` VALUES(12, 'general
academic information', 4, 'We report GPA (grade point
average) on a 4.0 scale. The top grade is an A, which
equals 4.0. <a
href="http://rre.csusb.edu/misc/gpcalc.html"
target="_blank">CSUSB GPA Calculator</a>', '');
INSERT INTO `application_sections` VALUES(13, 'colleges
and universities attended', 4, 'Please list all colleges
and universities you have attended in chronological order,
beginning with the first college or university you
attended. Make sure to include your current institution
and any ESL studies (if applicable).', '');
INSERT INTO `application_sections` VALUES(14, 'graduate
record examination (GRE)', 5, 'We only accept GRE taken
within <strong>3 years</strong>. If you have taken the GRE
multiple times, please report the best score received in
each skill category. ', '');
INSERT INTO `application_sections` VALUES(15, 'Test of
English as a Foreign Language (TOEFL)', 19, 'We only
accept TOEFL taken within 2 years. You only need to report
a Total Score. ', 'internationalOnly');
INSERT INTO `application_sections` VALUES(16,
'International English Language Testing System (IELTS)',
19, 'We only accept 6.0 band score taken within 2 years.
IELTS is NOT required If you have TOEFL score. ',
'internationalOnly');
```

```
INSERT INTO `application_sections` VALUES(17,
'recommenders', 6, 'Please list up to 3 recommenders, who
will submit letters of recommendation on your behalf.
Recommendations should be provided from employers,
supervisors, professors, or others who are able to comment
on your professional and educational goals and
potential.', '');
INSERT INTO `application_sections` VALUES(18, 'statement
of purpose', 0, 'Write a brief statement (no more than 400
words) on your reasons for pursuing the M.S. Degree in
Computer Science. Include any additional information
concerning your preparation pertinent to the M.S.
Degree.', '');
INSERT INTO `application_sections` VALUES(19, 'English
Requirement', 5, 'TOEFL or IELTS is required for
international students only. This requirement can be
waived by completing ACLP/IEP level 6 (language program).
', 'internationalOnly');


-- ---------------------------------------------------------
-


--
-- Table structure for table `evaluations`
--

CREATE TABLE `evaluations` (
  `studentID` varchar(12) NOT NULL,
  `classification` varchar(50) NOT NULL,
  `admitterm` varchar(20) NOT NULL,
  `deficiencies` varchar(150) NOT NULL,
  `comments` varchar(2000) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=utf8;


--
-- Dumping data for table `evaluations`
--



-- ---------------------------------------------------------
-


--
-- Table structure for table `evaluation_fields`
```

```
--

CREATE TABLE `evaluation_fields` (
  `id` int(11) NOT NULL auto_increment,
  `name` varchar(20) NOT NULL,
  `type` varchar(20) NOT NULL,
  `class` varchar(100) NOT NULL,
  `hint` varchar(200) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

--

-- Dumping data for table `evaluation_fields`
--


-- --------------------------------------------------------------
-

--

-- Table structure for table `help_center`
--

CREATE TABLE `help_center` (
  `id` int(11) NOT NULL auto_increment,
  `name` varchar(35) NOT NULL,
  `content` varchar(150) NOT NULL,
  `class` varchar(20) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;

--

-- Dumping data for table `help_center`
--

INSERT INTO `help_center` VALUES(1, 'Navigation Overview',
'./system/modules/navigation_overview.php',
'open_dialog');
INSERT INTO `help_center` VALUES(2, 'Technical Tips',
'./system/modules/tech_tips.php', 'open_dialog');
INSERT INTO `help_center` VALUES(3, 'FAQ (Frequently Asked
Questions)', './system/modules/question_answers.php',
'open_dialog');
```

```
--  -------------------------------------------------------------
-
--
-- Table structure for table `message_templates`
--

CREATE TABLE `message_templates` (
  `id` int(11) NOT NULL auto_increment,
  `code` varchar(30) NOT NULL,
  `subject` varchar(200) NOT NULL,
  `content` varchar(5000) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;

--
-- Dumping data for table `message_templates`
--

INSERT INTO `message_templates` VALUES(1, 'letter of
reference', 'letter of reference for CSE in CSUSB', 'has
applied CSE in CSUSB and has listed you as reference.
Please click the link below to fill your recommendation
letter. \\n To the Recommender: Persons who have applied
for admissions to CSUSB, whether or not they have been
accepted for admission and enrolled therein, shall be
permitted to inspect and review letters and statements of
recommendations to which they have not waived their right
of access (see above student waiver of access). \\n Please
click the link below to start you recommendation: ');

--  -------------------------------------------------------------
-


--
-- Table structure for table `new_account`
--

CREATE TABLE `new_account` (
  `id` int(11) NOT NULL auto_increment,
  `name` varchar(20) NOT NULL,
  `type` varchar(20) NOT NULL,
  `class` varchar(300) NOT NULL,
  `hint` varchar(150) NOT NULL,
```

```
   PRIMARY KEY  (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;


--
-- Dumping data for table `new_account`
--

INSERT INTO `new_account` VALUES(1, 'email', 'text',
'textInput required validateEmail validateEmailExist',
'Please input your valid email address. You will need it
to activate your account.');
INSERT INTO `new_account` VALUES(2, 'password',
'password', 'textInput required validateAlphaNum
validateAtLeast validateMinLength val-8', '');


-- ------------------------------------------------------
-


--
-- Table structure for table `recommendation`
--

CREATE TABLE `recommendation` (
  `id` int(11) NOT NULL auto_increment,
  `recommender` int(11) NOT NULL,
  `applicant` int(11) NOT NULL,
  `howlonghaveyouknowntheapplicant` varchar(20) NOT NULL,
  `relationship` varchar(20) NOT NULL,
  `position` varchar(50) NOT NULL,
  `institution` varchar(100) NOT NULL,
  `phone` varchar(20) NOT NULL,
  `address` varchar(200) NOT NULL,
  `nativeintellectualability` varchar(5) NOT NULL,
  `imaginationandprobablecreativity` varchar(5) NOT NULL,
  `breadthofgeneralknowledge` varchar(5) NOT NULL,
  `abilityinoralexpression` varchar(5) NOT NULL,
  `writingability` varchar(5) NOT NULL,
  `emotionalmaturity` varchar(5) NOT NULL,
  `promiseasateacher` varchar(5) NOT NULL,
  `overallrecommendationastopromiseasaprofessional`
varchar(5) NOT NULL,
  `overallrecommendation` varchar(10) NOT NULL,
  `comments` varchar(8000) NOT NULL,
```

```sql
  `timestamp` timestamp NOT NULL default
CURRENT_TIMESTAMP,
  PRIMARY KEY  (`id`)`
) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=9 ;


--
-- Dumping data for table `recommendation`
--



-- ---------------------------------------------------------------
-


--
-- Table structure for table `recommendation_fields`
--


CREATE TABLE `recommendation_fields` (
  `id` int(11) NOT NULL auto_increment,
  `name` varchar(70) NOT NULL,
  `section` varchar(20) NOT NULL,
  `type` varchar(20) NOT NULL,
  `class` varchar(200) NOT NULL,
  `hint` varchar(100) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=112 ;


--
-- Dumping data for table `recommendation_fields`
--


INSERT INTO `recommendation_fields` VALUES(6, 'how long
have you known the applicant', 'info', 'text', 'textInput
required', 'How long have you known the Applicant?      ');
INSERT INTO `recommendation_fields` VALUES(5,
'relationship', 'info', 'text', 'textInput required', 'In
what relationship?');
INSERT INTO `recommendation_fields` VALUES(2, 'position',
'info', 'text', 'textInput required', 'position');
INSERT INTO `recommendation_fields` VALUES(1,
'institution', 'info', 'text', 'textInput required',
'institution');
```

132

```sql
INSERT INTO `recommendation_fields` VALUES(3, 'phone',
'info', 'text', 'textInput required validatePhone', 'your
primary phone number');
INSERT INTO `recommendation_fields` VALUES(4, 'address',
'info', 'text', 'textInput required', 'street line, city,
state, country');
INSERT INTO `recommendation_fields` VALUES(7, 'native
intellectual ability ', 'comparison', 'radio',
'required', '');
INSERT INTO `recommendation_fields` VALUES(8, 'imagination
and probable creativity ', 'comparison', 'radio',
'required', '');
INSERT INTO `recommendation_fields` VALUES(9, 'breadth of
general knowledge ', 'comparison', 'radio', '', '');
INSERT INTO `recommendation_fields` VALUES(10, 'ability in
oral expression', 'comparison', 'radio', '', '');
INSERT INTO `recommendation_fields` VALUES(11, 'writing
ability ', 'comparison', 'radio', 'required', '');
INSERT INTO `recommendation_fields` VALUES(12, 'emotional
maturity', 'comparison', 'radio', 'required', '');
INSERT INTO `recommendation_fields` VALUES(13, 'promise as
a teacher', 'comparison', 'radio', 'required', '');
INSERT INTO `recommendation_fields` VALUES(14, 'overall
recommendation as to promise as a professional',
'comparison', 'radio', 'required', 'overall recommendation
as to promise as a professional in the field ');
INSERT INTO `recommendation_fields` VALUES(15, 'overall
recommendation ', 'overall', 'radio', 'required', '');


-- -------------------------------------------------------
-

--
-- Table structure for table
`recommendation_fields_options`
--

CREATE TABLE `recommendation_fields_options` (
  `id` int(11) NOT NULL auto_increment,
  `code` varchar(20) NOT NULL,
  `name` varchar(100) NOT NULL,
  `parent` int(11) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=11 ;
```

```
--
-- Dumping data for table `recommendation_fields_options`
--

INSERT INTO `recommendation_fields_options` VALUES(1, '1',
'Upper 1  or 2%', 0);
INSERT INTO `recommendation_fields_options` VALUES(2, '2',
'upper 10%  but not  upper 1 or  2%', 0);
INSERT INTO `recommendation_fields_options` VALUES(3, '3',
'upper 25%  but not  upper 10% ', 0);
INSERT INTO `recommendation_fields_options` VALUES(4, '4',
'upper half  but not upper  25% ', 0);
INSERT INTO `recommendation_fields_options` VALUES(5, '5',
'lower  half', 0);
INSERT INTO `recommendation_fields_options` VALUES(6, '6',
'no basis  for  judgement ', 0);
INSERT INTO `recommendation_fields_options` VALUES(7,
'excellent', 'I recommend the applicant without
reservation as an excellent prospect. ', 15);
INSERT INTO `recommendation_fields_options` VALUES(8,
'good', 'I have some reservations, but would recommend the
applicant as a good prospect. ', 15);
INSERT INTO `recommendation_fields_options` VALUES(9,
'doubts', 'I have substantial doubts, but think the
applicant should be given a chance to prove him/herself',
15);
INSERT INTO `recommendation_fields_options` VALUES(10,
'not', 'I feel the applicant is not suited for the
program. ', 15);


-- --------------------------------------------------------
-


--
-- Table structure for table `recommender`
--

CREATE TABLE `recommender` (
  `id` int(11) NOT NULL auto_increment,
  `applicant` int(11) NOT NULL,
  `nameofreference` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
  `waivetoreview` varchar(5) NOT NULL,
```

```sql
  `status` varchar(20) NOT NULL default 'not received',
  `action` varchar(30) NOT NULL default 'delete edit',
  `hash` varchar(32) NOT NULL,
  PRIMARY KEY  (`id`).
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=24
;


--
-- Dumping data for table `recommender`
--


-- ----------------------------------------------------------
-


--
-- Table structure for table `school_attended`
--


CREATE TABLE `school_attended` (
  `id` int(11) NOT NULL auto_increment,
  `applicant` int(11) NOT NULL,
  `schoolname` varchar(200) NOT NULL,
  `yearfrom` varchar(15) NOT NULL,
  `yearto` varchar(15) NOT NULL,
  `GPA` varchar(8) NOT NULL,
  `degreeobtained` varchar(50) NOT NULL,
  `action` varchar(30) NOT NULL default 'delete edit',
  PRIMARY KEY  (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=16
;


--
-- Dumping data for table `school_attended`
--


-- ----------------------------------------------------------
-


--
-- Table structure for table `staff`
--


CREATE TABLE `staff` (
  `id` int(11) NOT NULL auto_increment,
```

```
  `firstname` varchar(30) NOT NULL,
  `middleinitial` varchar(10) NOT NULL,
  `lastname` varchar(30) NOT NULL,
  `title` varchar(30) NOT NULL,
  `position` varchar(80) NOT NULL,
  `email` varchar(100) NOT NULL,
  `phone` varchar(20) NOT NULL,
  `salt` varchar(32) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;


--
-- Dumping data for table `staff`
--


INSERT INTO `staff` VALUES(1, 'Haiyan', '', 'Qiao',
'Associate Professor', 'Graduate Coordinator for
Admissions', 'hqiao@csusb.edu', '1-(909) 537-5415', '');


-- --------------------------------------------------------
-


--
-- Table structure for table `tables`
--


CREATE TABLE `tables` (
  `id` int(11) NOT NULL auto_increment,
  `parent` int(11) NOT NULL,
  `name` varchar(20) NOT NULL,
  `type` varchar(20) NOT NULL,
  `class` varchar(200) NOT NULL,
  `hint` varchar(500) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=10
;


--
-- Dumping data for table `tables`
--


INSERT INTO `tables` VALUES(1, 29, 'school name', 'text',
'textInput required ', '');
```

```sql
INSERT INTO `tables` VALUES(2, 29, 'year from', 'text',
'textInput required ', '');
INSERT INTO `tables` VALUES(3, 29, 'year to', 'text',
'textInput required ', '');
INSERT INTO `tables` VALUES(4, 29, 'GPA', 'text',
'textInput required ', '');
INSERT INTO `tables` VALUES(5, 29, 'degree obtained',
'text', 'textInput required ', '');
INSERT INTO `tables` VALUES(6, 30, 'name of reference',
'text', 'textInput required ', 'first name MI last name');
INSERT INTO `tables` VALUES(7, 30, 'email', 'text',
'textInput required validateEmail', '');
INSERT INTO `tables` VALUES(8, 30, 'waive to review',
'radio', 'required ', 'Waiver of access to statements of
recommendation is optional. I understand that letters and
statements of recommendation \nconcerning me are to be
received and maintained by CSUSB, School of Computer
Science and Engineering, and I hereby expressly and
voluntarily waive any and all access rights I might have
to such recommendations under the Federal Family Rights
and Privacy Act, the California Information Act, or the
laws, regulations, or policies.');
INSERT INTO `tables` VALUES(9, 30, 'status', 'info',
'hide', 'letter not sent');


-- ----------------------------------------------------------
-


--
-- Table structure for table `tables_options`
--

CREATE TABLE `tables_options` (
  `id` int(11) NOT NULL auto_increment,
  `code` varchar(20) NOT NULL,
  `name` varchar(75) NOT NULL,
  `parent` int(11) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=3 ;


--
-- Dumping data for table `tables_options`
--
```

```
INSERT INTO `tables_options` VALUES(1, 'yes', 'I do waive
my right to review this letter of reference.', 8);
INSERT INTO `tables_options` VALUES(2, 'no', 'I do not
waive my right to review this letter of reference', 8);
```

REFERENCES

[1] JQuery, "API Documentation", Internet: www.jquery.com,
     [October 22, 2013]

[2] JQuery User Interface, "API Documentation", Internet:
     www.jqueryui.com, [October 22, 2013]

[3] K. Rotte, "Graduate Advising System : Application
     Component", M.S. thesis, California State University,
     San Bernardino, San Bernardino, CA, Spring 2005.

[4] PHP Hypertext Preprocessor, "PHP Manual", Internet:
     www.php.net, [October 18, 2013]

[5] Y. Chiang, "Advising Module: Graduate Application
     System", M.S. thesis, California State University, San
     Bernardino, San Bernardino, CA, Spring 2005.