Spring 2017

Digitized Engineering Notebook

Geetha R. Bavanam
Governors State University

Himaja Chowdary Jampani
Governors State University

Mohaseen Baig Mirza
Governors State University

Satya Priya Sriramoju
Governors State University

ABSTRACT

Digitized Engineering Notebook is a web based application of the Engineering Notebook that can be used by an individual or team which can track and record all the activities during specific design process. This application can create a new project or select an existing project, add team members to the project, it also enables team members to access the portal at the same time there to synchronize their work. The substantial applicability of this portal is creation of dated log entries for ideas, details, and observations, chronological listing of log entries and enabling uploads of text documents, diagrams, sketches, figures, etc.

By designing the digitized engineering notebook in a user-friendly way, we can save lot of time for an individual or team by enabling to access and navigate from anywhere using their own devices.

Table of Content

1 Project Description

This is online portal which is used to record the daily work of a particular individual. It allows user to create project and can give access to the team members who are working on that project so that the all team share ideas files and comments on a particular item.

It has additional features like Events Creation, Event Calendar, Log Entry Complete Display, File Uploading functionality and displaying all uploaded files in a single place.

1.1 Competitive Information

Here the existing system is nothing but manual system using which they are maintaining their daily work in a note book. But the current system allows user to maintain their notes in online and can share with other team mates if required online.

1.2 Relationship to Other Applications/Projects

This application is not related to another application/project

1.3 Assumptions and Dependencies

There are no dependencies on this project.

1.4 Future Enhancements

Including of task completion module which shows how much percentage of the task has completed till that data. Another enhancement is that whenever team member commented on any post or uploaded any file for a project, the alert message will be sent to the users immediately on any update.

2 Project Technical Description

Development Environment

- **Windows System with OS of Windows 8**
- **Minimum 500 GB hard disk**
- **4 GB ram is preferred for better performance**
- **IIS 8 installation is required to deploy the project in the system.**
- **Visual studio must be installed for development purpose**
- **SQL Management studio need to install in the system for db.**

System Design:

SYSTEM DESIGN phase follows system analysis phase. Design is maintaining record proof design divisions and providing a blueprint for the implementation phase. The bridge which is between system analysis and system implementation is called design.

System design is transition from a user oriented, document oriented to programmers or database personnel. The design is a solution, a "how to" approach to the creation a new system. This is composed of several steps. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. The logical and physical stages are involved in the design stage in the areas of development and logical design reviews which prepares input and output specification including the detailed implementation plan and can prepare a walkthrough the logical design.

Objectives of Design:

System design is like a blue print for a building, it specifies all the features that are to be in the finished product. Objectives of analysis phase are accomplished by design.

Module Description:

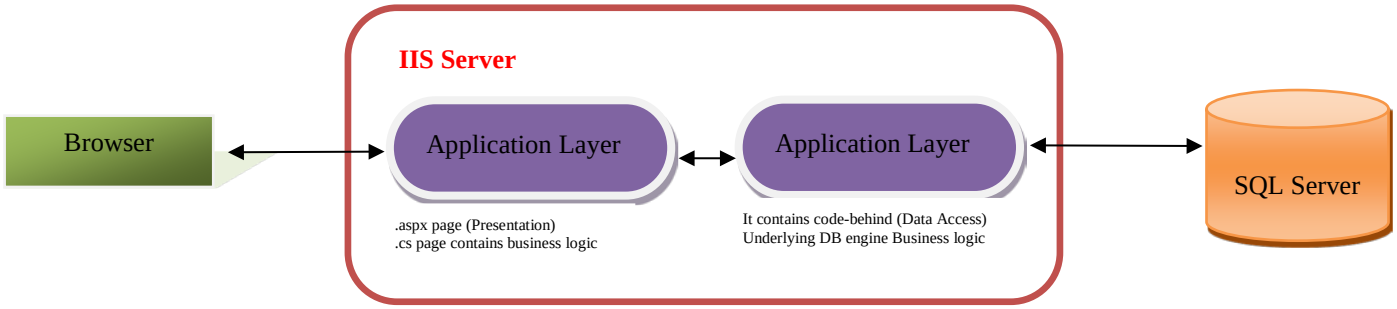
The entire application is based on the following modules:

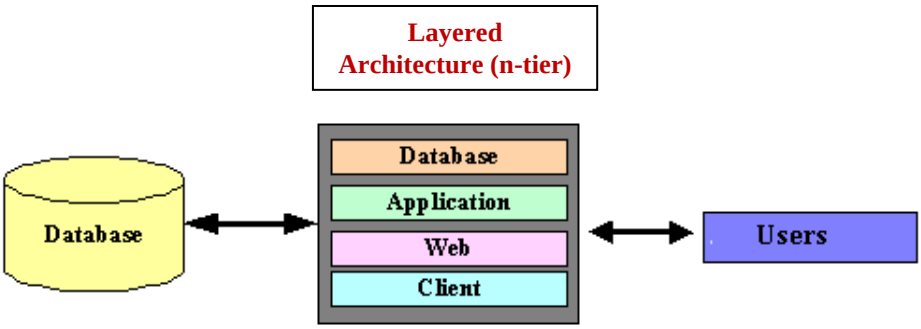
1. Administrator Module
2. User Module

Hierarchy of Users:

1. Administrator
2. User

2.1 Application Architecture

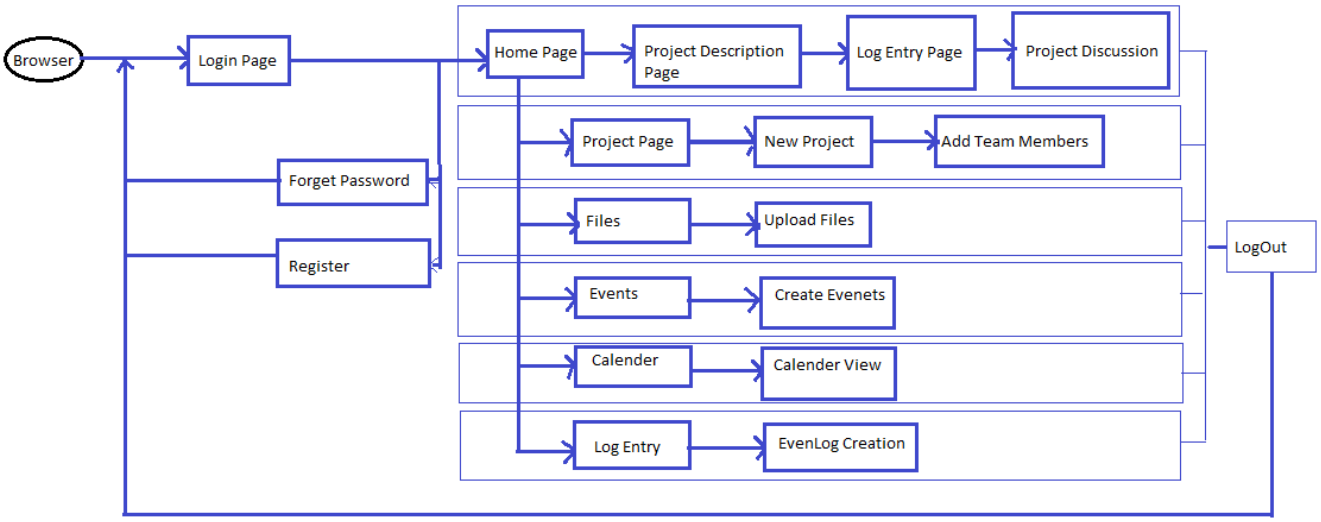




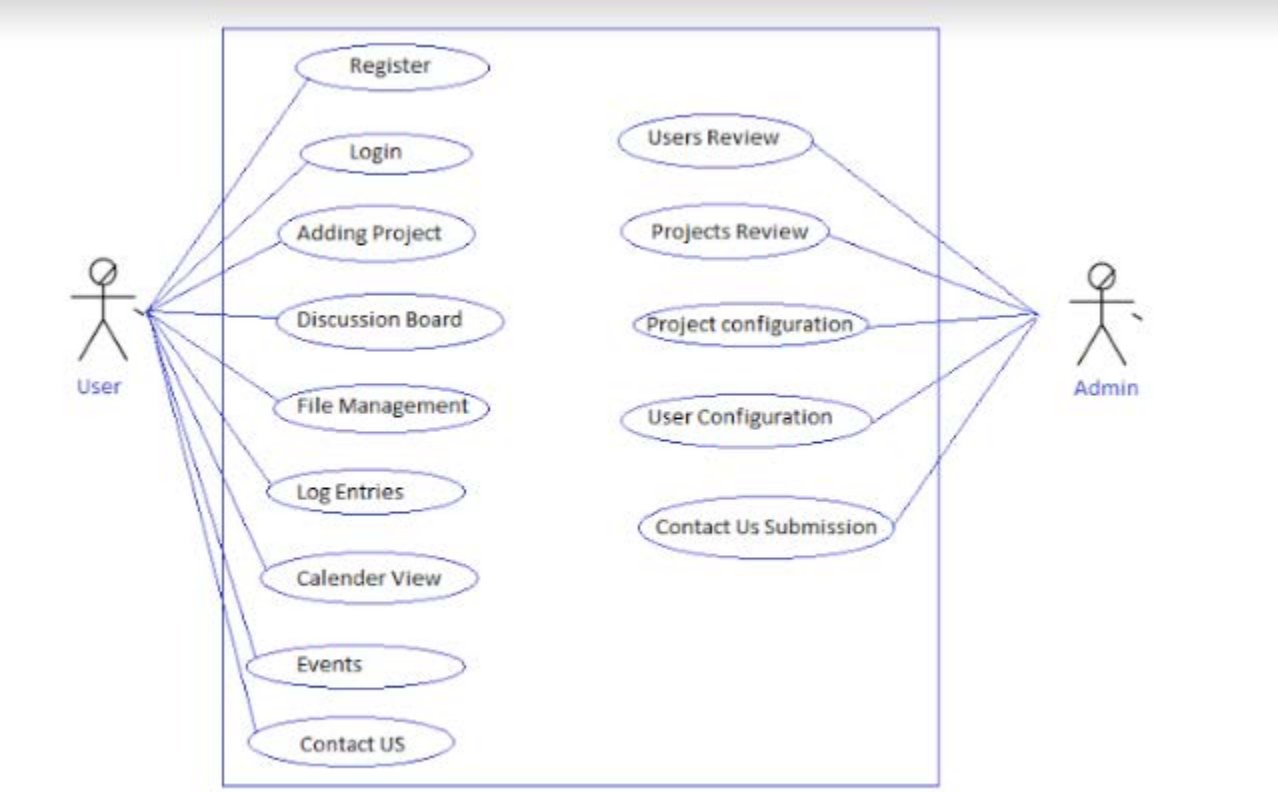
- **Database Layer:** Contains the data and database-related objects like stored procedures, triggers, packages, etc.

- **Application Layer:** Contains the objects addressing the business logic; All the business objects will be here in application layer.
- **Web Interface Layer**: It will be on the web server; It contains the web pages (ASPs) of the application which will interact with the front-end browsers
- **Client Layer:** Contains the web browser which interacts with web server

2.2 Application Information flows



Use case diagram:



2.3 Interactions with other Projects (if Any)

No interaction with other projects

2.4 Interactions with other Applications

No interaction with other applications

2.5 Capabilities

It will retrieve data from Sql server and the data entered in the front end will be saved in backend.

3 Project Requirements

3.1 Identification of Requirements

The main purpose of functional requirements within the requirement specification document is to define all the activities or operations that take place in the system. Through interactions with the users of the system we derived these.

Functional Requirements:

User login functionality in the main index page.

Any user can browse the site and contact the site owners without registration in contact us page.

< GSU-Spring 2017 DB-Admin-000001>

This project must allow Admin to access only the admin pages.

Implementation: Mandatory

< GSU-Spring 2017 DB-Admin-000002>

This project must allow Admin to access all internal projects created by user.

Implementation: Mandatory

< GSU-Spring 2017 DB-Admin-000003>

This project must allow Admin to access all user information.

Implementation: Mandatory

< GSU-Spring 2017 DB-Admin-000004>

This project must allow Admin to deactivate project. So that users cannot see those deactivated projects.

Implementation: Mandatory

< GSU-Spring 2017 DB-Admin-000005>

This project should contain a functionality that Admin can deactivate the user access to that site

Implementation: Mandatory

< GSU-Spring 2017 DB-Admin-000006>

This project must allow Admin to access all contact us submitted details.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-000007>

This project must allow users for registration.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-000008>

This project must allow users for contacting admin using contact us page.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-000009>

Activation link must be sent to user's email.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000010>

Password retrieval option must be provided to users

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000011>

This project should allow unauthenticated user to submit contact us form.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000012>

This project must allow users to create project

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000013>

This project must allow users to give access to team members for the created project.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000014>

This project must allow users to submit their daily work.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000015>

This project must allow users to comments on other works.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000016>

This project must allow users to create events.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000017>

This project must allow users to upload files.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000018>

This project must allow users to edit their personal details.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000019>

This project must allow users to view events in a calendar.

Implementation: Mandatory

< GSU-Spring 2017 DEN-Users-0000020>

This project must allow users to remove project access.

Implementation: Mandatory

Non-Functional Requirements:

The systems user interface aspects are being described as Nonfunctional requirements which are indirectly connected to the system application functionality.

About is the nonfunctional page which display about the site information.

User Interface and Human factors:

The proposed system should provide a user- friendly interface to customers and candidates as well as Administrator with ease of use. The user interface must be suitable for easy and fast data entry. With the help of this interface, customers and candidate even without adequate knowledge of systems would complete their tasks. Administrator needs little training about the system in-order to utilize the facilities being provided by the system efficiently.

Documentation:

The proposed system requires three levels of documentation, user level documentation which helps the customers, candidates and administrator how to interact with the system. This documentation includes how to fill the forms provided and get reports that can be generated by the system. In the proposed system help link provides the user level documentation. The RAD prepared during analysis phase helps the developer in-order to develop the system as per client's requirements. The system design documentation prepared during development process provides information regarding design goals and about subsystems into the proposed system which also

helps in testing process. In addition to user level and development level documentation proposed system also requires technical documentation for maintainers of the system. This technical documentation includes the port number on which server is running and listening client requests and includes any other configuration required for maintainers.

Hardware Consideration:

Hardware Considerations includes the virtual machine on which the system should be built. Virtual machine includes operating system and any software components needed. Virtual machine minimum required for web server is WINDOWS 7 and Above system and web browser is IE 9.0, Mozilla or Chrome.

Performance Characteristics: The proposed system is server applications and server applications are inherently multi-threaded. Every request creates new thread hence the system supports any no of concurrent users. Hence this system offers superior performance and easy solutions to problems. This is the static requirement. The Dynamic requirement is system response. As the proposed system developed using ASP Pages which reduces response problems.

Error handling and Extreme conditions: The proposed system should handle exceptions that originate at low level components and exceptions at elevated level components. The high-level components in proposed system should handle exceptions that occur while connecting to database Server, null reference etc. The end user should not be bothered about exceptions at low-level. When low level Exceptions arises, user should be shown with appropriate message. Errors that arise during data entry should be handled by performing client side validations. In the proposed system, all client side validations will be done using asp.net predefined validation controls.

Quality Issues:

Quality issues refer to how reliable and robust should the system be? While developing the proposed system the developer must be able to guarantee the reliability transactions so that they will be processed completely and accurately.

The ability of system to detect failures and recovery from those failures refers to the availability of system. Robustness of system refers to the capability of system providing information when concurrent users requesting for information.

As the proposed system's capability of handling various exception, it is reliable and it will be developed using ASP which supports multithreading. Hence it satisfies the requests from concurrent users. So, it is robust.

System modification:

Updating can be done by any developer familiar with specified hardware and software constraints followed for development of proposed system.

3.2 Operations, Administration, Maintenance and Provisioning (OAM&P)

3.3 Security and Fraud Prevention

Security and confidentiality are the top most concerns of the client. The proposed system should provide the following.

- Administrator should be provided with id and password for secured access of information regarding the users.
- Each User should also be provided with code and password for controlled access of information regarding their work log entries and projects.

- Pseudo Requirements: No design and implementation constraints imposed by the client before the development of this application.

3.4 Release and Transition Plan

The proposed system can be deployed and withstand in any physical environment.

4 Project Design Description

Project design phase follows system analysis phase. Design is maintaining record proof design divisions and providing a blueprint for the implementation phase. The bridge between system analysis and system implementation is called design.

Project design is transition from a user oriented, document oriented to programmers or database personnel. The design is a solution, a "how to" approach to the creation a new system. This is composed of several steps. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study.

Objectives of Design

Project design is like a blue print for a building, it specifies all the features that are to be in the finished product. The objectives determined in the analysis phase are accomplished by design.

Module Description:

The entire application is based on the following modules:

- Administrator Module
- User Module
- Hierarchy of Users:

- Administrator
- Users

5 Internal/external Interface Impacts and Specification

It is an internet application. The user who has credentials can login anywhere through internet.

Users: It's provides authentication and authorization of the users. The user can create the project and can write comments on that project and able to upload the files related to the project. User can maintain the daily status of the work related to project and the project owner has an extra feature of adding user to that project.

Admin: Admin has a role to activate or deactivate the project and able to restrict the user access to the site.

Most of the time the database server is busy providing authentication.

Future Enhancements:

Extendibility: Provides elevated level extendibility. It means it provides all the basic features and allows us to extend their features very easily without disturbing the existing code.

We can make this application is suitable to work on any application just by changing the deployment files.

6 Design Units Impacts

A Data Dictionary is a collection of metadata, that is, data about data. In addition to storing catalog information about schemas and constraints, the data dictionary stores other information, such as design decisions, usage standards, application program descriptions, and user information.

A Database is a collection of interrelated data stored with a minimum of redundancy to serve many applications. The database design is used to group data into several tables and minimizes the artificiality embedded in using separate files. The tables are organized to:

- Reduced duplication of data.
- Simplify functions like adding, deleting, modifying data etc.,
- Retrieving data
- Clarity and ease of use
- More information at low cost

Normalization is built around the concept of normal forms. A relation is said to be in a normal form if it satisfies a certain specified set of constraints on the kind of functional dependencies that could be associated with the relation. The normal forms are used to ensure that several types of anomalies and inconsistencies are not introduced into the database.

First Normal Form:

If all underlying domains contained atomic values then the relation R is in first normal form.

Second Normal Form:

A relation R is said to be in second normal form if and only if it is in first normal form and every non-key attribute is fully dependent on the primary key.

Third Normal Form:

A relation R is said to be in third normal form if and only if it is in second normal form and every non-key attribute is non-transitively depend on the primary key.

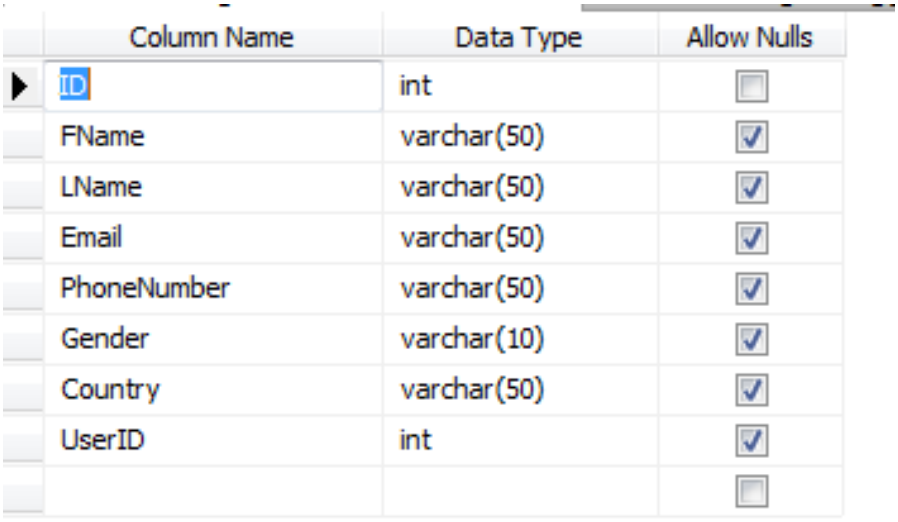
1. **Roles**

Column Name	Data Type	Allow Nulls
ID	int	☐
RoleName	varchar(30)	☑
		☐

2. **Users**

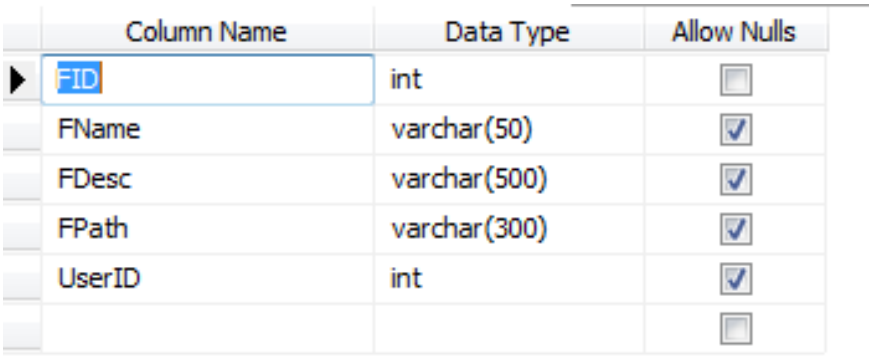
Column Name	Data Type	Allow Nulls
UserID	int	☐
UserName	varchar(50)	☑
Password	varchar(50)	☑
UInfoID	int	☑
RoleID	int	☑
UStatus	bit	☑
		☐

3. **User Info**



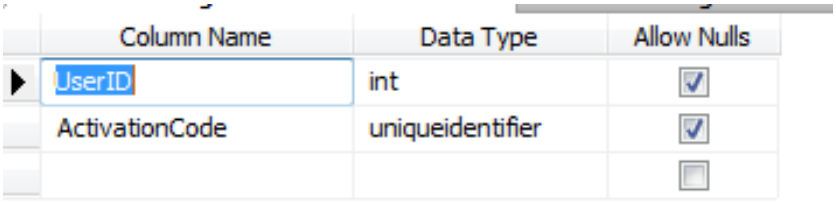
Column Name	Data Type	Allow Nulls
ID	int	☐
FName	varchar(50)	☑
LName	varchar(50)	☑
Email	varchar(50)	☑
PhoneNumber	varchar(50)	☑
Gender	varchar(10)	☑
Country	varchar(50)	☑
UserID	int	☑
		☐

4. User Files



Column Name	Data Type	Allow Nulls
FID	int	☐
FName	varchar(50)	☑
FDesc	varchar(500)	☑
FPath	varchar(300)	☑
UserID	int	☑
		☐

5. **User Activation**



Column Name	Data Type	Allow Nulls
UserID	int	☑
ActivationCode	uniqueidentifier	☑
		☐

6. Project Log

Column Name	Data Type	Allow Nulls
Id	int	☐
Comment	varchar(300)	☑
Commented	datetime	☑
UserId	int	☑
ProjectID	int	☑
FID	int	☑
		☐

7. **Project Access**

Column Name	Data Type	Allow Nulls
Id	int	☐
ProjectID	int	☑
UserID	int	☑
		☐

8. **Project**

Column Name	Data Type	Allow Nulls
PID	int	☐
ProName	varchar(50)	☑
ProDuration	varchar(50)	☑
startDate	varchar(50)	☑
ProDesc	varchar(500)	☑
ProNotes	varchar(300)	☑
UserId	int	☑
Created	datetime	☑
PStatus	bit	☑
		☐

Column Properties

9. **Log Entries**

Column Name	Data Type	Allow Nulls
LogID	int	☐
LogName	varchar(50)	☑
LogDate	varchar(30)	☑
CreatedBy	int	☑
PID	int	☑
LogNotes	varchar(500)	☑
FilePath	varchar(250)	☑
		☐

10. File details

Column Name	Data Type	Allow Nulls
FID	int	☐
FName	varchar(50)	☑
FPath	varchar(250)	☑
		☐

11. Events

Column Name	Data Type	Allow Nulls
EID	int	☐
EName	varchar(50)	☑
EDate	varchar(30)	☑
CreatedBy	int	☑
		☐

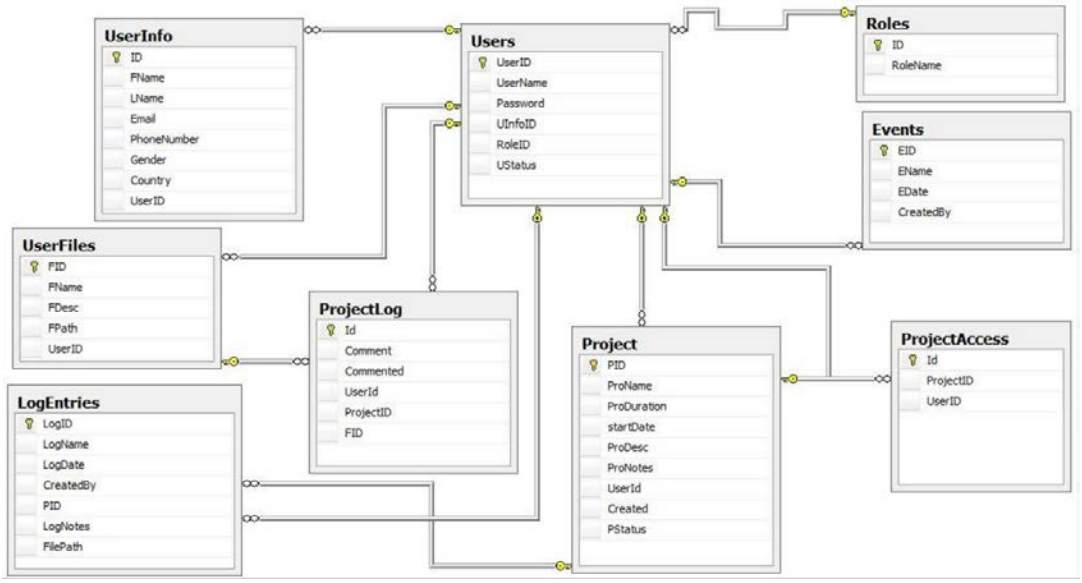
12. Contact Message

Column Name	Data Type	Allow Nulls
MsgID	int	☐
FName	varchar(50)	☑
LName	varchar(30)	☑
Email	varchar(40)	☑
Message	varchar(500)	☑
		☐

Stored Procedures List:

- Emplogin
- ProjectLogInsert
- ProjectSubmit
- UserActivationCheck
- UserRegister
- UserUpdate

ER diagram:



6.1 Functional Area /Design Unit

6.1.1 Functional Overview

Front end or User Interface Design

The entire user interface is planned to be developed in browser specific environment with a touch of internet-Based Architecture for achieving the Distributed Concept. The browser specific components are designed by using the HTML standards, and the dynamism of the designed by concentrating on the constructs of the ASP Server Pages. We have used bootstrap for UI design.

Communication or Database Connectivity Tier

The Communication architecture is designed by ADO.NET and the database connectivity is established by using the ADO.NET Connectivity object.

The standards of three-tier architecture are given major concentration to keep the standards of higher cohesion and limited coupling for effectiveness of the operations.

About C#:

C# is an elegant and type-safe object-oriented language that enables developers to build a variety of secure and robust applications that run on the .NET Framework. You can use C# to create Windows client applications, XML Web services, distributed components, client-server applications, database applications, and much, much more. Visual C# provides an advanced code editor, convenient user interface designers, integrated debugger, and many other tools to make it easier to develop applications based on the C# language and the .NET Framework.

About ASP.NET:

ASP.NET is a web development platform, which provides a programming model, a comprehensive software infrastructure and various services required to build up robust web applications for PC, as well as mobile devices.

ASP.NET works on top of the HTTP protocol, and uses the HTTP commands and policies to set a browser-to-server bilateral communication and cooperation.

ASP.NET is a part of Microsoft .Net platform. ASP.NET applications are compiled codes, written using the extensible and reusable components or objects present in .Net framework. These codes can use the entire hierarchy of classes in .Net framework.

An ASP.NET application codes can be written in any of the following languages:

- C#
- Visual Basic.Net
- Jscript
- J#

ASP.NET is used to produce interactive, data-driven web applications over the internet. It consists of many controls such as text boxes, buttons, and labels for assembling, configuring, and manipulating code to create HTML pages.

About ADO.NET:

ADO.NET provides a bridge between the front-end controls and the back end database. The ADO.NET objects encapsulate all the data access operations and the controls interact with these objects to display data, thus hiding the details of movement of data.

It has a import objects like Data Reader, Data Command and Data Adapter etc.

Java Script

JavaScript is a script-based programming language that was developed by Netscape Communication Corporation. JavaScript was originally called Live Script and renamed as JavaScript to indicate its relationship with Java. JavaScript supports the development of both client and server components of Web-based applications. On the client side, it can be used to write programs that are executed by a Web browser within the context of a Web page. On the server side, it can be used to write Web server programs that can process information submitted by a Web browser and then updates the browser's display accordingly.

Even though JavaScript supports both client and server Web programming, we prefer JavaScript at Client side programming since most of the browsers supports it. JavaScript is almost as easy to learn as HTML, and JavaScript statements can be included in HTML documents by enclosing the statements between a pair of scripting tags. <SCRIPTS>. </SCRIPT>.

<SCRIPT LANGUAGE = "JavaScript">

JavaScript statements

</SCRIPT>

Here are a few things we can do with JavaScript:

- Validate the contents of a form and make calculations.
- Add scrolling or changing messages to the Browser's status line.
- Animate images or rotate images that change when we move the mouse over them.

- Detect the browser in use and display different content for different browsers.
- Detect installed plug-ins and notify the user if a plug-in is required.
- We can do much more with JavaScript, including creating entire application.

Hyper Text Markup Language

Hypertext Markup Language (HTML), the languages of the World Wide Web (WWW), allows users to produce Web pages that include text, graphics and pointer to other Web pages (Hyperlinks).

HTML is not a programming language but it is an application of ISO Standard 8879, SGML (Standard Generalized Markup Language), but specialized to hypertext and adapted to the Web. The idea behind Hypertext is that instead of reading text in rigid linear structure, we can easily jump from one point to another point. We can navigate through the information based on our interest and preference. A markup language is simply a series of elements, each delimited with special characters that define how text or other items enclosed within the elements should be displayed. Hyperlinks are underlined or emphasized works that load to other documents or some portions of the same document.

HTML can be used to display any type of document on the host computer, which can be geographically at a different location. It is a versatile language and can be used on any platform or desktop.

HTML provides tags (special codes) to make the document look attractive. HTML tags are not case-sensitive. Using graphics, fonts, varied sizes, color, etc., can enhance the presentation of the document. Anything that is not a tag is part of the document itself.

Basic HTML Tags:

- <! -- --> Specifies comments
- <A>……. </A> Creates hypertext links
- <B>……. </B> Formats text as bold
- <BIG>……. </BIG> Formats text in large font.
- <BODY>…</BODY> Contains all tags and text in the HTML document
- <CENTER>...</CENTER> Creates text
- <DD>…</DD> Definition of a term
- <DL>...</DL> Creates definition list
- <FONT>…</FONT> Formats text with a font
- <FORM>...</FORM> Encloses a fill-out form
- <FRAME>...</FRAME> Defines a frame in a set of frames
- <H#>…</H#> Creates headings of various levels
- <HEAD>...</HEAD> Contains tags that specify information about a document
- <HR>...</HR> Creates a horizontal rule
- <HTML>…</HTML> Contains all other HTML tags
- <META>...</META> Provides meta-information about a document
- <SCRIPT>…</SCRIPT> Contains client-side or server-side script
- <TABLE>…</TABLE> Creates a table
- <TD>…</TD> Indicates table data in a table
- <TR>…</TR> Designates a table row
- <TH>…</TH> Creates a heading in a table

Advantages

- A HTML document is small and hence easy to send over the net. It is small because it does not include formatted information.
- HTML is platform independent.
- HTML tags are not case-sensitive.

Bootstrap:

Bootstrap is a powerful front-end framework for faster and easier web development. It includes HTML and CSS based design templates for common user interface components like Typography, Forms, Buttons, Tables, Navigations, Dropdowns, Alerts, Modals, Tabs, Accordion, Carousel and many other as well as optional JavaScript extensions. Bootstrap also gives you ability to create responsive layout with much less efforts.

Advantages of Bootstrap

The main advantage of Bootstrap is free set of tools by which one can create flexible and responsive web layouts and common interface components.

Additionally, using the Bootstrap data APIs you can create advanced interface components like Scroll spy and Typeaheads without writing a single line of JavaScript.

Reason for Bootstrap:

Save lots of time — you can save lots of time and efforts using the Bootstrap predefined design templates and classes and concentrate on other development work.

- **Responsive features** — Using Bootstrap you can easily create responsive designs. Bootstrap responsive features make your web pages to appear more appropriately on different devices and screen resolutions without any change in markup.
- **Consistent design** — All Bootstrap components share the same design templates and styles through a central library, so that the designs and layouts of your web pages are consistent throughout your development.
- **Easy to use** — Bootstrap is very easy to use. Basic working knowledge of HTML and CSS is enough to start development with Bootstrap.
- **Compatible with browsers** — Bootstrap is created with modern browsers in mind and it is compatible with all modern browsers such as Mozilla Firefox, Google Chrome, Safari, Internet Explorer, and Opera.
- **Open Source** — and the best part is it is completely free to download and use.

Testing

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later during maintenance also.

Testing Objectives:

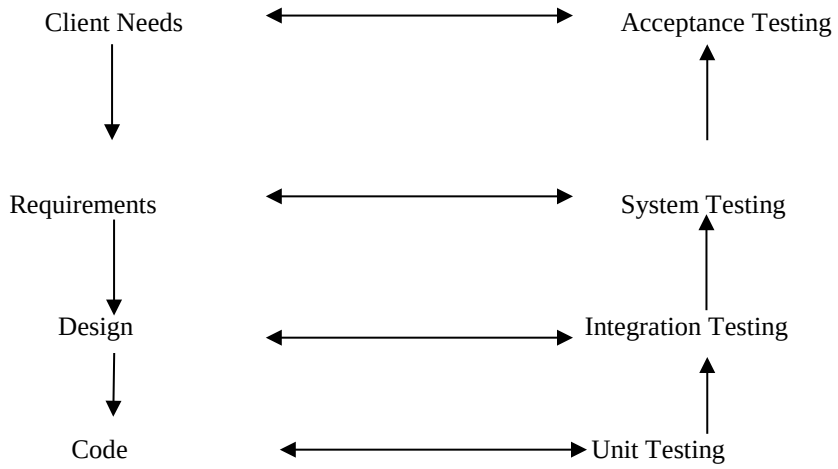
The main objective of testing is to uncover a host of errors, systematically and with minimum effort and time. Stating formally, we can say,

- Testing is a process of executing a program with the intent of finding an error.
- A successful test is one that uncovers a yet undiscovered error.

- The tests are inadequate to detect possibly present errors.
- The software confirms to the quality and reliable standards.

Level of Testing

To uncover the errors, present in distinct phases we have the concept of levels of testing. The basic levels of testing are



Unit testing:

Unit testing centers check exertion around the littlest unit of programming i.e. the module. Using the detailed design and the process specifications testing is done to uncover errors within the boundary of the module. All modules should be successful in the unit test before the begin of the reconciliation testing starts In this project, each service can be thought of a module. There are so many modules like Admin, Tracking, and Inventor y. Each module has been tested by giving different sets of input. When building up the module and also completing the advancement so that every module works with no mistake. By accepting from the user inputs are validated.

Integration Testing:

After the unit testing we must perform integration testing. The goal here is to see if modules can be integrated properly, the emphasis being on testing interfaces between modules. This testing

activity can be considered as testing the design and hence the emphasis on testing module interactions.

In this project, the main system is formed by integrating all the modules. When integrating all the modules I have checked whether the integration effects working of any of the services by giving different combinations of inputs with which the two services run perfectly before Integration.

System Testing

Here the entire software system is tested. The reference document for this process is the requirements document, and the goal is to see if software meets its requirements.

Here the entire Digitized Engineering Notebook website has been tested against requirements of project and it is checked whether all requirements of project have been satisfied or not.

Acceptance Testing

To demonstrate the software to the client, acceptance test is performed with the given data. Testing here is focused on external behavior of the system the internal logic of program is not emphasized. In this project Digitized Engineering Notebook website, I have loaded some data and tested whether project is working correctly or not.

Test cases should be selected so that the largest number of attributes of an equivalence class is exercised at once. In software development, testing phase is important part. It is the process of finding errors and missing operations and a complete verification to determine whether the objectives are met and the user requirements are satisfied.

Test Cases

S.N (	TESTING OBJECT	EXPECTED VALUE	SIMULATED VALUE	EXPLANATION	REMARKS
1	Login/UserID	"satyapriyagsu@gmail.com"	"satyapriyagsu@gmail.com"	Pass	Expected Value=Simulated Value
2	Login/UserID	"satyapriyagsu@gmail.com"	"satyapriya2010@gmail.com"	Fail	UserID Incorrect
3	Login/UserID	"name"	""	Fail	Log infield Empty
4	Login/Password	"password"	"password"	Pass	Expected Value=Simulated Value
5	Login/Password	"password"	""	Fail	Password Field Empty
6	Login/Password	"password"	"PASSWORD"	Fail	Case Sensitive

6.1.2 Impacts

ASP.net pages are simple, yet powerful technology for creating and maintaining dynamic-content web pages. Based on the Microsoft Programming languages, ASP Pages offers proven portability, open standards, and a mature re-usable component model. The ASP architecture enables the separation of content generation from content presentation. This separation not eases maintenance headaches, it also allows web team members to focus on their areas of expertise. Now, web page designer can concentrate on layout, and web application designers on programming, with minimal concern about impacting each other's work.

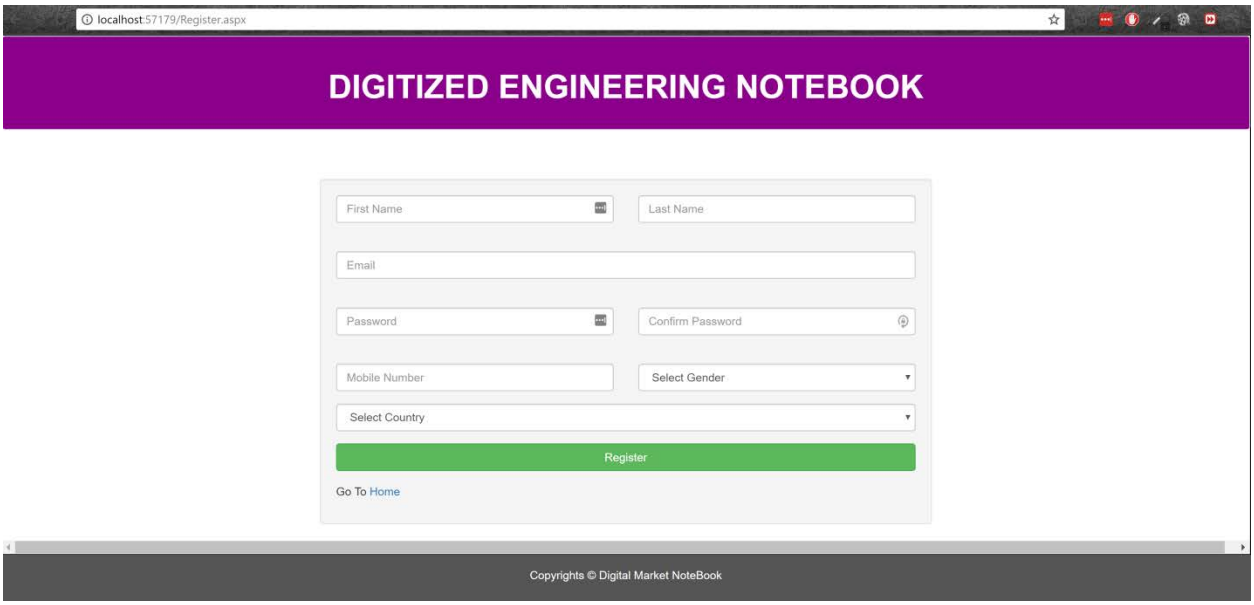
6.1.3 Requirements

- This system should allow the administrator to manage the users.

- This system should allow the administrator to edit/view/deactivate the Users and Projects.
- This system should allow the administrator to manage user's information.
- This system should allow the administrator to manage projects information.
- This system should allow the user to create projects.
- This system should allow the user to add members to the projects created.
- This system should allow the user to create an event related to the project.
- This system should allow the user to upload files related to the project.
- This system should allow the user to create Log Entry for a project.
- This system should allow the user to create multiple projects.
- This system should allow the multiple users to share/comment/upload files across the common projects.
- This system should allow the user to view the list of his projects.
- This system should allow the user to view the list of his Events.
- This system should allow the user to view the his Events in a calendar view by day/week/month.
- This system should allow the user to Update his/her profile information.
- This system should allow the user to Register for an account.
- This system should send an automatic e-mail to all its users for account activation.

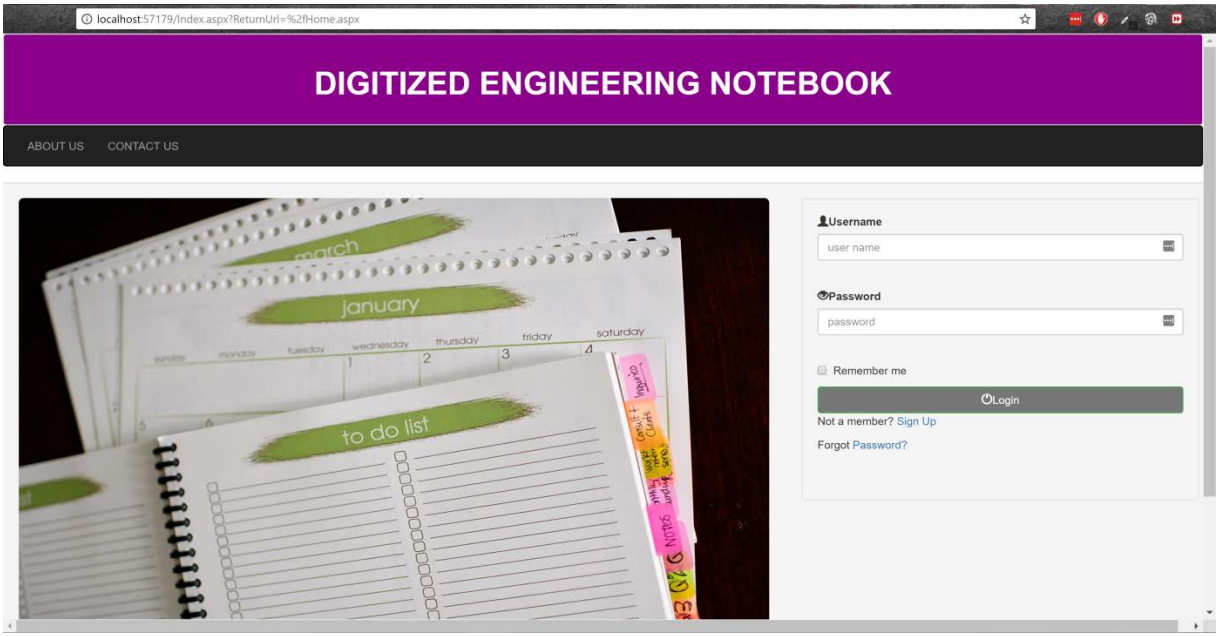
6.2 Functional Area B/Design Unit B

Registration Page:



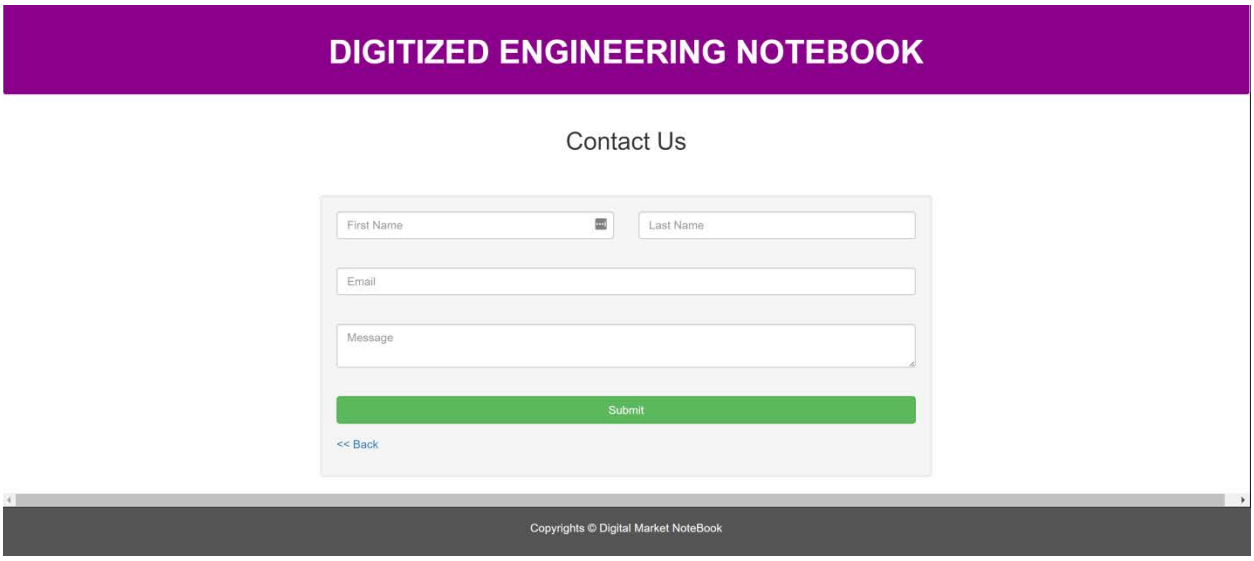
Registration page allows the user to register as a member of the project.

Login Page:



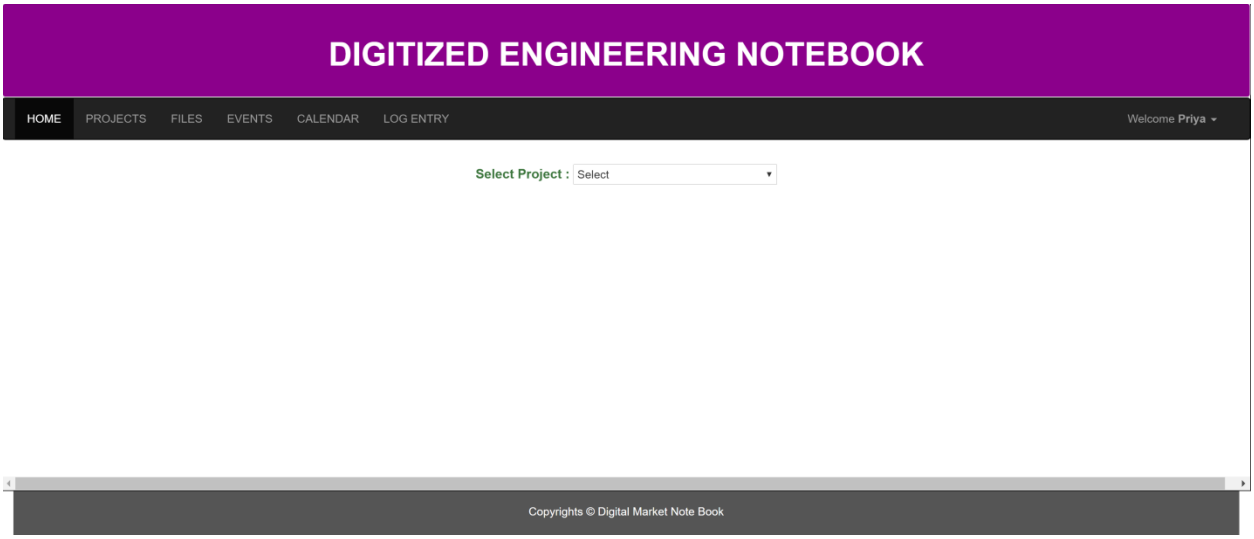
Login page allows the user to enter the credentials and also gives access to the website.

Contact Us Page:



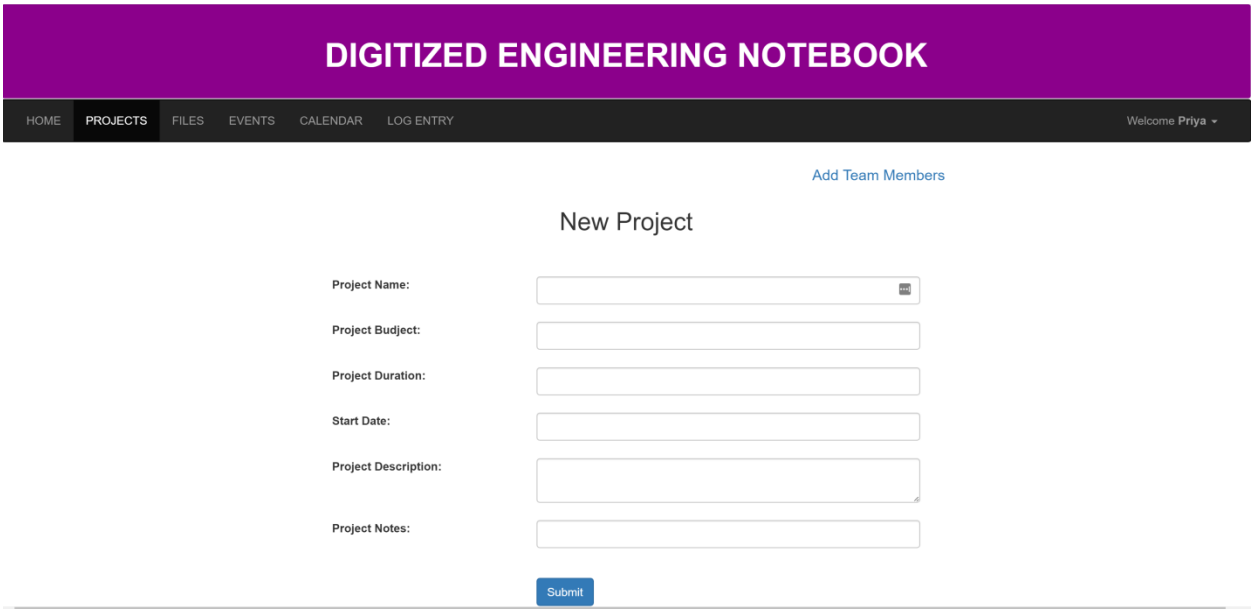
This page allows any user to request information about the site.

Home Page:



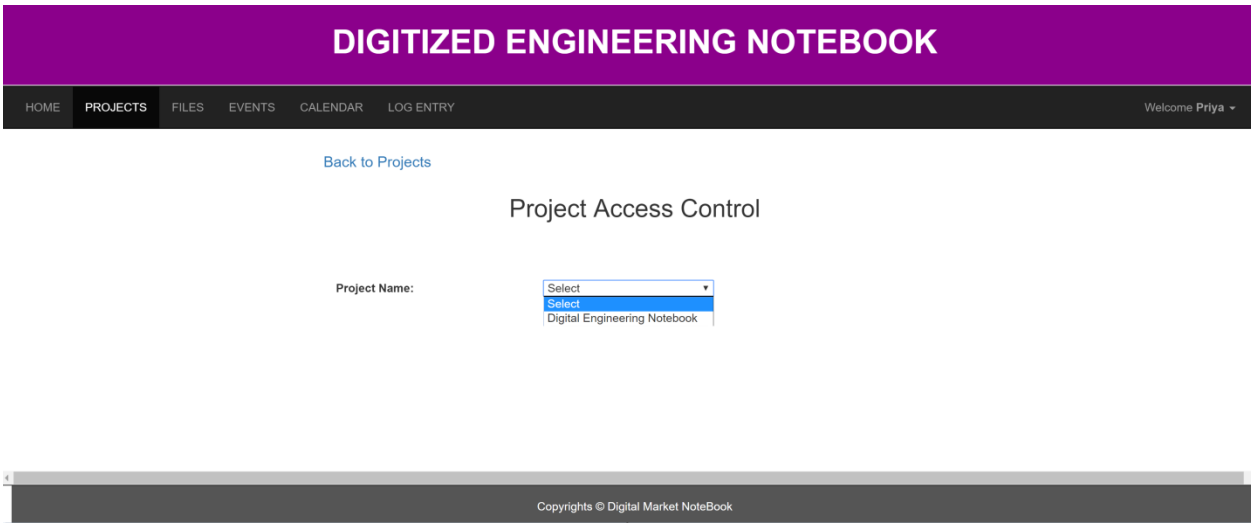
This page allows the user to select the project that he/she created.

Projects Page:

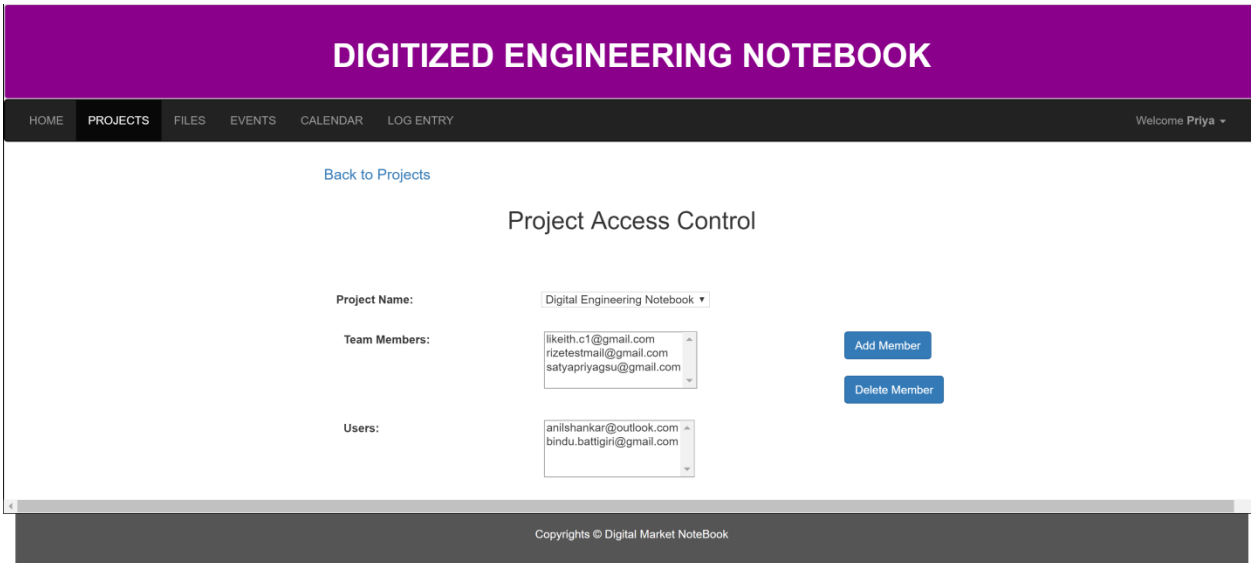


This page allows the user to create a project.

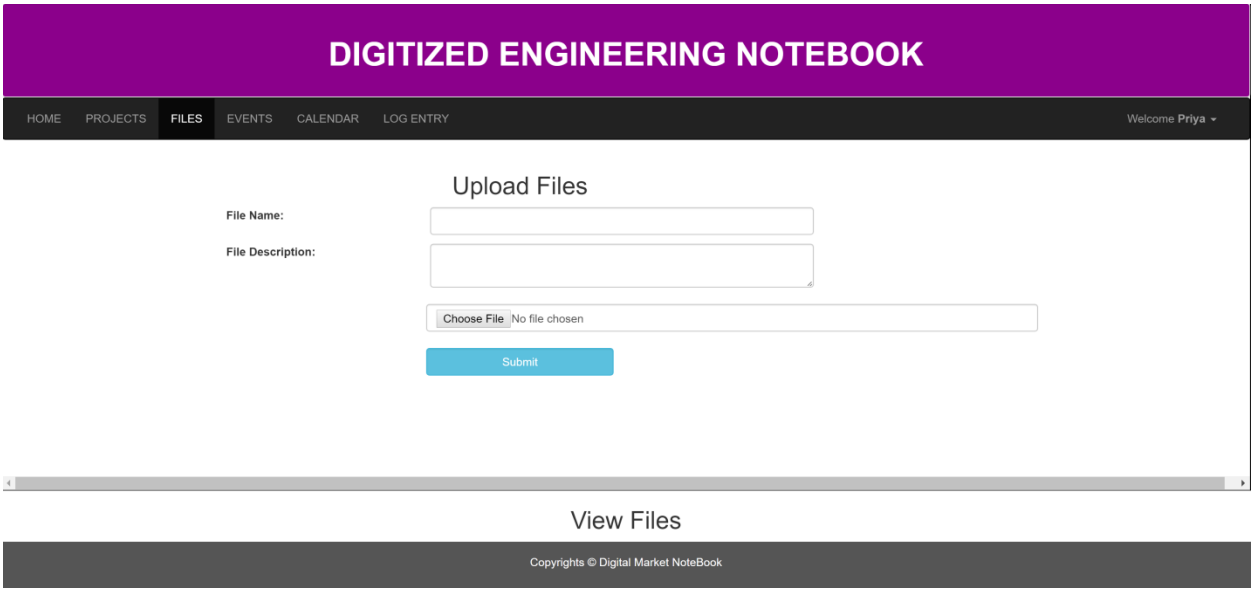
Project Access Control Page:



This page allows the user to Add Team Members to the selected project from the drop-down.

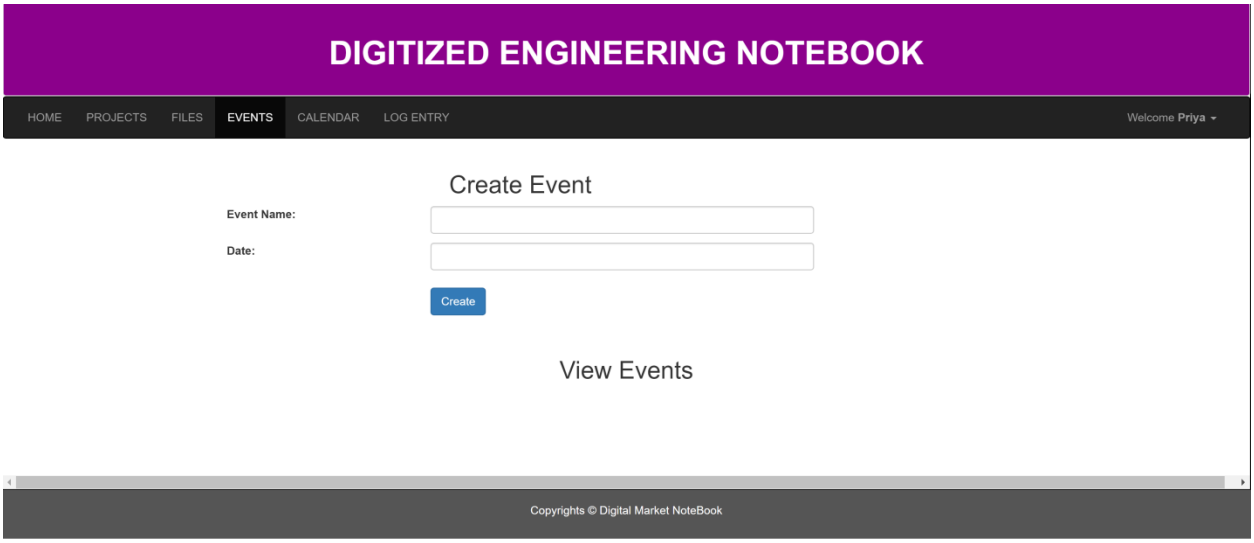


Files:



This page allows the user to upload files related to his/her project.

Events:

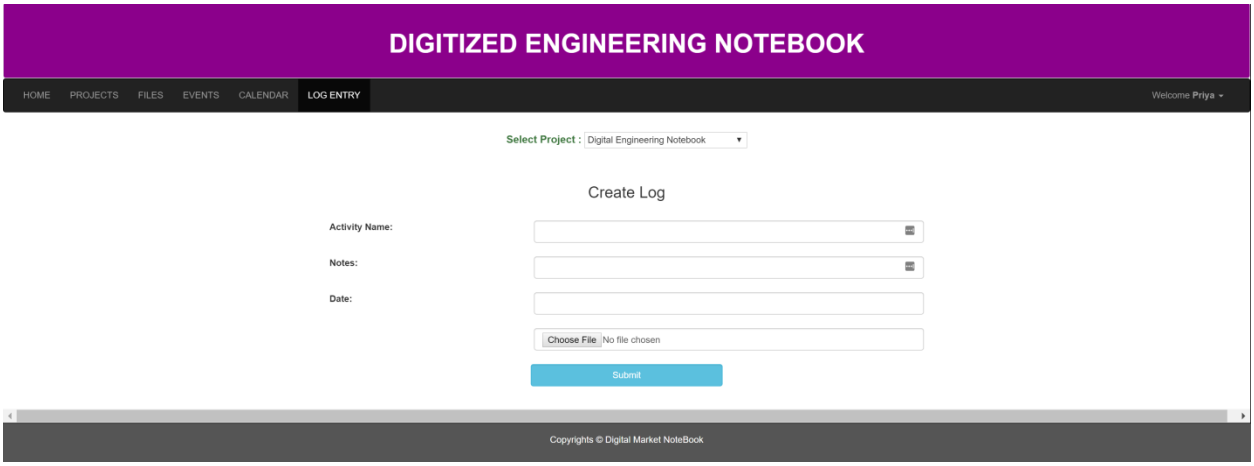


This page allows the user to create events for the project.

Event Viewer:

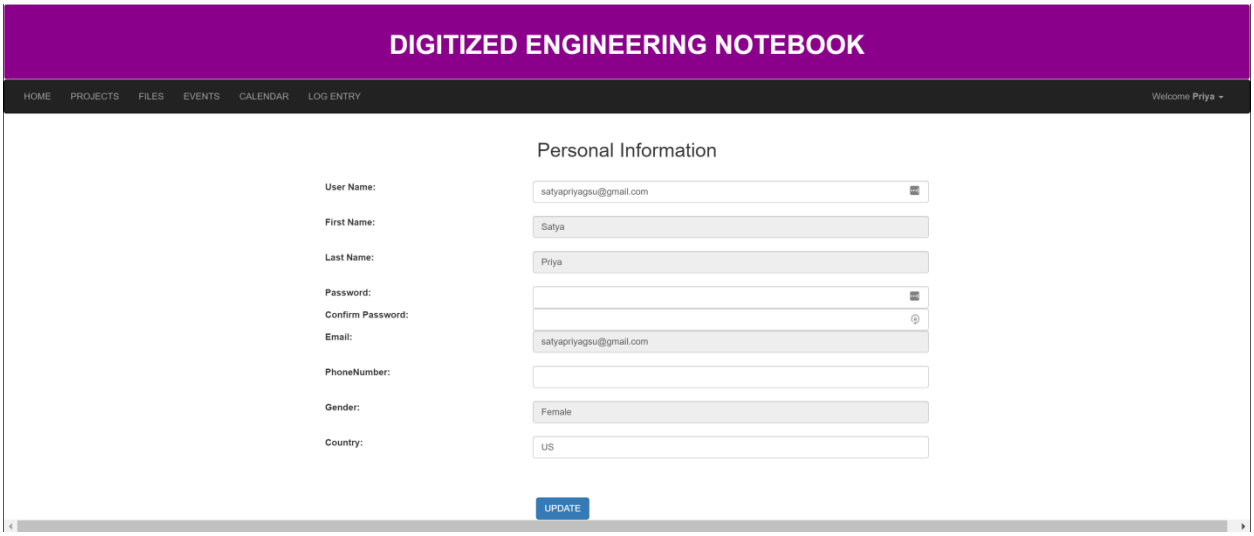
This page allows the user to view the events created for the project.

Log Entry:



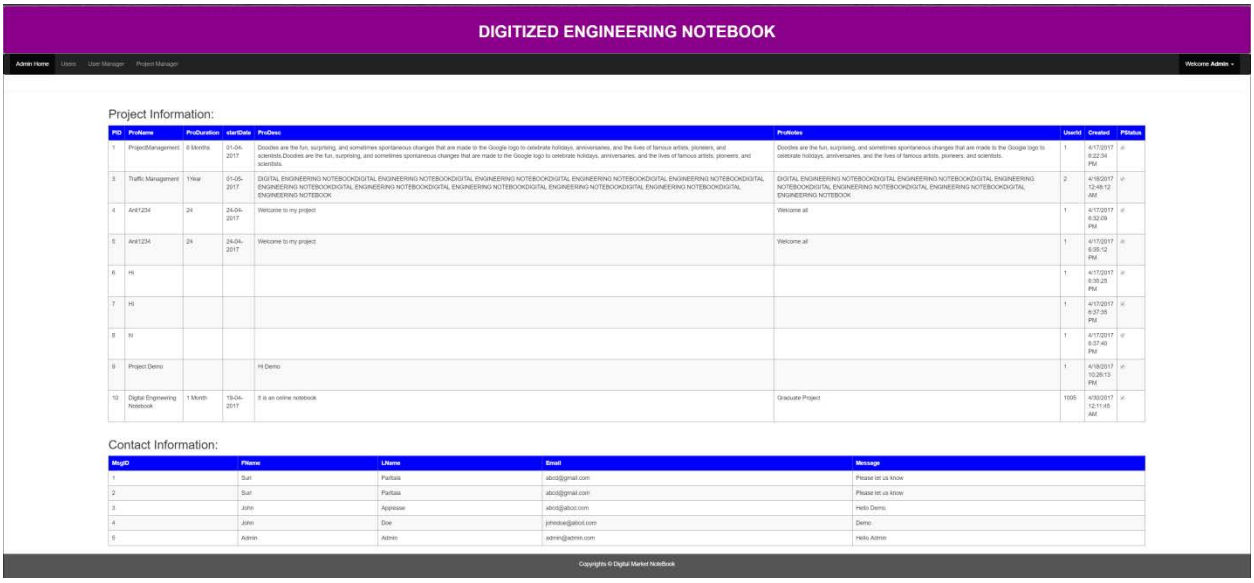
This page allows the user to create the log entries for the selected project.

Personal Information:



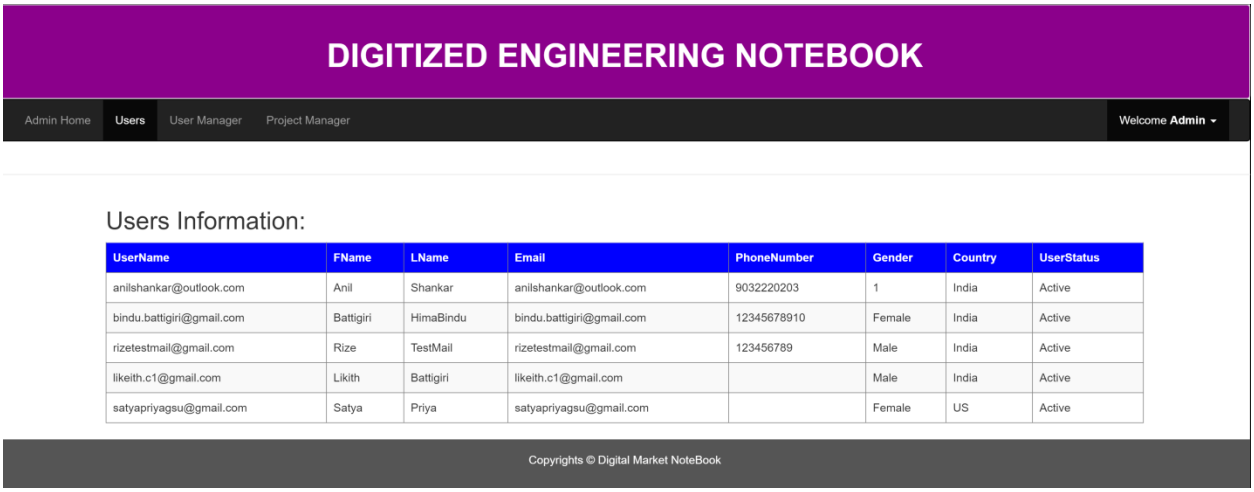
This page allows the user to update his/her personal information.

Admin Home:



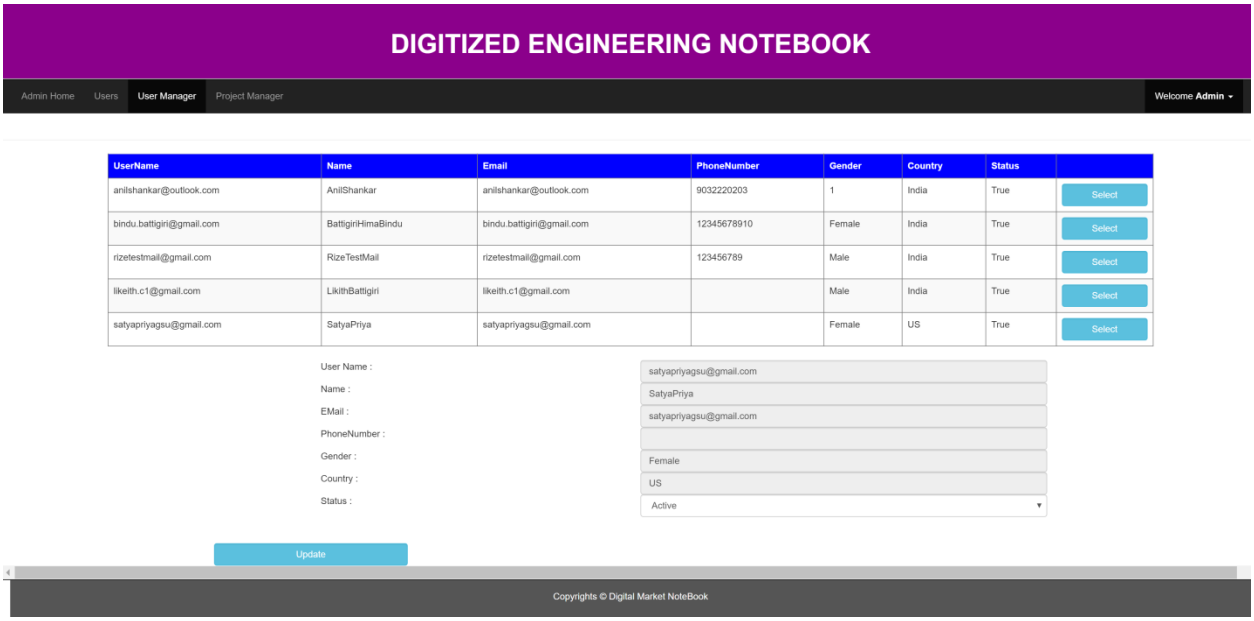
This page contains the information about the projects created by all the registered users along with the Messages (Requested by users).

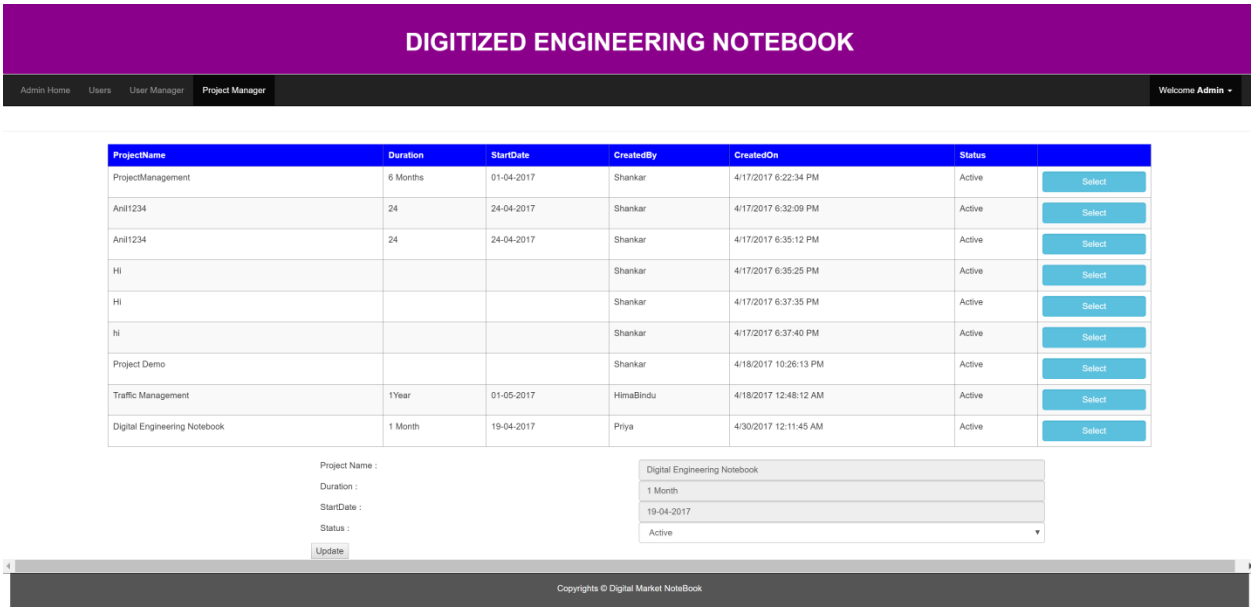
Admin/Users:



This page displays the information of the Registered Users.

Admin User Manager/Project Manager:





Both the User Management and Project Management pages are designed to manage the users and projects respectively.

6.2.1 *Functional Overview*

The “Digitized Engineering Notebook” is web-based application. This application software has been computed successfully and was also tested successfully by taking “test cases”. It is user friendly, and has required options, which can be utilized by the user to perform the desired operations. The software is developed using C#.Net as front end and MS-SQL as back end in Windows environment. The goals that are achieved by the software are:

- ✓ Instant access.
- ✓ Improved productivity.
- ✓ Optimum utilization of resources.
- ✓ Efficient management of records.
- ✓ Simplification of the operations.
- ✓ Less processing time and getting required information.
- ✓ User friendly.

7 Open Issues

In our Project, we don't have any open issues. Everything is completed and running successfully.

8 Acknowledgements

- *Software Engineering practice and principles* 6th edition by *Roger Pressmen* (Tata McGraw Hill).
- *Internet & World Wide Web How to program* 3rd edition by *Deitel&Deitel and Goldberg* (Pearson education).
- *Data base System Concepts* 4th edition by *Silbershatz, Korth, and Sudharshan* (Tata McGraw Hill).
- *Fundamentals of Data base systems* 4th edition by Ramez*Elmasri and ShamkantB.Navathe*(Pearson education).

9 References

https://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx

https://msdn.microsoft.com/en-us/library/aa187916.aspx

https://msdn.microsoft.com/en-us/library/e80y5yhx(v=vs.110).aspx

http://www.itp.uzh.ch/~suzanne/ebooks/The%20Web%20Book-A4-HM.pdf

http://www.tutorialrepublic.com/twitter-bootstrap-tutorial/bootstrap-introduction.php

https://www.w3schools.com/bootstrap/