

A Project Report
On

STUDENT CONNECT

Submitted in partial fulfillment of the requirements for the award of the degree
Of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

BY

Mr. Putta Ravinder (2451 – 09 – 733 – 038)

Under the guidance of

T. Sujanavan

Asst. Professor

Department of CSE

M.V.S.R.E.C., Hyderabad.



Department of Computer Science and Engineering

M.V.S.R. ENGINEERING COLLEGE

(Affiliated to Osmania University & Recognized by AICTE)

Nadergul, Saroor Nagar Mandal, Hyderabad – 501 510

2012-13.

Department of Computer Science and Engineering
M.V.S.R. ENGINEERING COLLEGE
(Affiliated to Osmania University & Recognized by AICTE)
Nadergul, Saroor Nagar Mandal, Hyderabad – 501 510



CERTIFICATE

This is to certify that the project work entitled “**STUDENT CONNECT**” is bona fide work carried out by **Mr. Putta Ravinder (2451 – 09 – 733 – 038)** in partial fulfillment of the requirements for the award of degree of **BACHELOR OF ENGINEERING IN COMPUTER SCIENCE AND ENGINEERING** from M.V.S.R. Engineering College, affiliated to **OSMANIA UNIVERSITY, Hyderabad**, under our guidance and supervision.

The results embodied in this report have not been submitted to any other university or institute for the award of any degree or diploma.

Internal Guide
T.Sujanavan
Asst.professor
Department of CSE
MVSREC,Hyderabad.

Head of Department
J.Prasanna Kumar
Professor and Head
Department of CSE
MVSREC,Hyderabad.

DECLARATION

This is to certify that the work reported in the present project entitled “**STUDENT CONNECT**” is a record of bona fide work done by me in the Department of Computer Science and Engineering, M.V.S.R. Engineering College, Osmania University. The reports are based on the project work done entirely by us and not copied from any other source.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma to the best of my knowledge and belief.

Mr. Putta Ravinder
(2451 – 09 – 733 – 038).

ACKNOWLEDGEMENTS

We would like to express my sincere gratitude and indebtedness to my project guide **Mr.T. Sujanavan** for his/her valuable suggestions and interest throughout the course of this project.

We are also thankful to our principal **Mr. P. A. Sastry** and **Mr. J. Prasanna Kumar**, Professor and Head, Department of Computer Science and Engineering, MVSR Engineering College, Hyderabad for providing excellent infrastructure and a nice atmosphere for completing this project successfully as a part of our B.E. Degree (CSE).

We convey our heartfelt thanks to the lab staff for allowing me to use the required equipment whenever needed.

Finally, we would like to take this opportunity to thank my family for their support through the work. We sincerely acknowledge and thank all those who gave directly or indirectly their support in completion of this work.

Mr. Putta Ravinder (2451 – 09 – 733 – 038).

TABLE OF CONTENTS

	PAGE NOS
Certificate	i
Declaration.....	ii
Acknowledgements.....	iii
Table Of Contents.....	iv-vi
List of Tables.....	vii
List of Figures.....	viii
Abstract.....	ix

CHAPTER 1

INTRODUCTION	1
1.1 Existing system	1
1.2 Proposed system	1
1.3 Modules	1
1.3.1 Administrator Module	1
1.3.2 Student Module	1
1.3.3 Corporate Module	2
1.3.4 Groups Module	2
1.4 Scope	2

CHAPTER 2

TECHNOLOGIES AND TOOLS	3
2.1 HTML and CSS	3
2.2 JavaScript	3
2.3 AJAX	4
2.4 WAMP	4
2.5 Apache HTTP Server	4
2.6 MySQL	5
2.7 PHP	5

2.8 Net Beans IDE 7.3	5
2.9 Block Diagram	6
CHAPTER 3	
SYSTEM DESIGN	7
3.1 Requirements for system design	7
3.1.1 Star UML	7
3.1.2 Smart Draw	8
3.2 Logical Design	8
3.2.1 Application Design	8
3.2.2 Data Design	18
3.3 Physical design	21
3.3.1 System Architecture	22
3.3.2 System Components	23
3.4 Database tables	27
CHAPTER 4	
SYSTEM DEVELOPMENT	30
4.1 Requirements during Development	30
4.1.1 WAMP Server	30
4.1.2 Net Beans IDE	33
4.1.3 PHPmyAdmin	36
4.2 Sample Code	39
4.3 Output Screens	47
4.3.1 Common pages for all actors	47
4.3.2 Student page	50
4.3.3 Corporate page	52

CHAPTER 5	55
DEPLOYMENT	55
5.1 Requirements for Deployment	55
5.2 WAMP Directory Structure	55
5.3 Configuration files	
CHAPTER 6	
CONCLUSIONS	57
6.1 Conclusions	57
6.1.1 Limitations	57
6.1.2 Further Enhancements	57
REFERENCES	58

List of Tables

Table 3.1: Student Table	27
Table 3.2: Corporate Table	27
Table 3.3: Forum_question Table	28
Table 3.4: Forum_answer Table	28
Table 3.5: Opportunity Table	29

List of figures

Fig 3.1: Use Case Diagram of the System	09
Fig 3.2: Class Diagram of the System	10
Fig 3.3: State Chart Diagram of the Student and Corporate	11
Fig 3.4: Activity Diagram for Registration	12
Fig 3.5: Activity Diagram for Student	13
Fig 3.6: Activity Diagram for Corporate	13
Fig 3.7: Sequence Diagram	15
Fig 3.8: Collaboration Diagram	16
Fig 3.9: Component Diagram of the System	17
Fig 3.10: Deployment Diagram of the System	18
Fig 3.11: ER Diagram of the System	20
Fig 3.12: System Architecture	22
Fig 3.13: Component Flow Diagram	23
Fig 3.14: ER Diagram of the System	22
Fig 3.15: System Architecture	24
Fig 4.1: Getting Started in WAMP – 1	30
Fig 4.2: Getting Started in WAMP – 2	31
Fig 4.3: Getting Started in WAMP – 3	32
Fig 4.4: Getting Started in WAMP – 4	32
Fig 4.5: NetBeans Start Page	34
Fig 4.6: Setting Project in NetBeans – 1	35
Fig 4.7: Setting Project in NetBeans – 2	35
Fig 4.8: PhpMyAdmin – Administration	36
Fig 4.9: PhpMyAdmin – Database	38

ABSTRACT

Internet has become so indispensable in today's world. It is being used for varied purposes several blogs which now favor technical discussions, but they are not very authenticated or secure enough to share innovations. There are also several community portals like IEEE and ACM, but they do not support active communication between students like a chat or discussion forum as in social networking. Moreover they bore only highly complex technical research ideas and a novice in a field willing to enter may find it abstruse. So there is a severe lack in technical communication and there is a need for an effective platform for discussing and sharing technical knowledge between students and research people.

In the proposed system, we propose a novel system of technical networking where we provide authenticated channel of technical communication of innovations, guidance, resources between students, industries and research.

CHAPTER 1

INTRODUCTION

Internet has become so indispensable in today's world. It is being used for varied purposes several blogs which now favor technical discussions, but they are not very authenticated or secure enough to share innovations. There are also several community portals like IEEE and ACM, but they do not support active communication between students like a chat or discussion forum as in social networking. Moreover they bore only highly complex technical research ideas and a novice in a field willing to enter may find it abstruse. So there is a severe lack in technical communication and there is a need for an effective platform for discussing and sharing technical knowledge between students and research people..

1.1 Existing System

In the existing system, there is not a single portal which fulfills student needs. They cannot share ideas and cannot get enough guidance from the expert community. Few attempts have been made to develop such systems but no system serves the student's needs to the fullest.

1.2 Proposed System

In the proposed system, we propose a novel system of technical networking where we provide authenticated channel of technical communication of innovations, guidance, resources between students, industries and research.

1.3 Modules

1.3.1 Administrator

The Administrator is going to verify the registrations of students and corporates in order to make sure that the details provided at the time of registration are correct. Administrator also has the responsibility of spam filtering.

1.3.2 Student

The Student needs to register with correct details. Once the registration is verified the student can create a new topic, join in the discussion of any existing topic. Student can also view the opportunities posted by the various registered corporates.

1.3.3 Corporate

The Corporate needs to register with correct details. Once the registration is verified the corporate can post new opportunities, join in the discussion of any existing topic created by the students.

1.3.4 Groups

In order to facilitate the user we can maintain some groups related to exams preparation, idea

Sharing, and also for job searching. Each of this group will have moderators and users can create a topic or subscribe to the existing topic.

1.4 Scope

The scope of the system is to provide an automated solution for all the discussions, opportunities that pertain to Students and Corporates of various institutions and organizations respectively. The corporate post opportunities to the students and participate in the discussion. The student can view and create topics, the student can also comment and discuss on the topics.

. They are:

- a) Admin.
- b) Student.
- c) Corporate.

The following describes the Roles of all the users in this System.

- a) **Admin:** An admin in the system has the authority to verify the credentials of the students and corporates and also does spam filtering.
- b) **Student:** The role of student here are, creation of topics, join the discussions, and view opportunities.
- c) **Corporate:** The Corporates role is to view discussions, post opportunities and also resolve any queries pertaining to the student discussions.

CHAPTER – 2

TECHNOLOGIES AND TOOLS

We use the following technologies and tools to build the project.

2.1 HTML and CSS

HyperText Markup Language (HTML) is the main markup language for creating web pages and other information that can be displayed in a web browser.

HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets (like `<html>`), within the web page content. HTML tags most commonly come in pairs like `<h1>` and `</h1>`, although some tags, known as *empty elements*, are unpaired, for example ``. The first tag in a pair is the *start tag*, the second tag is the *end tag* (they are also called *opening tags* and *closing tags*). In between these tags web designers can add text, tags, comments and other types of text-based content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behavior of HTML web pages.

Web browsers can also refer to Cascading Style Sheets (CSS) to define the appearance and layout of text and other material. The W3C, maintainer of both the HTML and the CSS standards, encourages the use of CSS over explicit presentational HTML markup.

2.2 JavaScript

JavaScript (JS) is an interpreted computer programming language. It was originally implemented as part of web browsers so that client-side scripts could interact with the user, control the browser, communicate asynchronously, and alter the document content that was displayed.

JavaScript is a prototype-based scripting language that is dynamic, weakly typed, and has first-class functions. Its syntax was influenced by the language C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the self and Scheme programming languages. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles.

JavaScript's use in applications outside of web pages—for example, in PDF documents, site-specific browsers, and desktop widgets—is also significant. Newer and faster JavaScript VMs and frameworks built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications. JavaScript was formalized in the ECMAScript language standard and is primarily used as part of a web browser (client-side JavaScript). This enables programmatic access to computational objects within a host environment.

2.3 Ajax

Ajax (also **AJAX**; an acronym for **Asynchronous JavaScript and XML**) is a group of interrelated web development techniques used on the client-side to create asynchronous web applications. With Ajax, web applications can send data to, and retrieve data from, a server asynchronously (in the background) without interfering with the display and behavior of the existing page. Data can be retrieved using the XMLHttpRequest object. Despite the name, the use of XML is not required (JSON is often used instead), and the requests do not need to be asynchronous.

Ajax is not a single technology, but a group of technologies. HTML and CSS can be used in combination to mark up and style information. The DOM is accessed with JavaScript to dynamically display, and allow the user to interact with, the information presented. JavaScript and the XMLHttpRequest object provide a method for exchanging data asynchronously between browser and server to avoid full page reloads.

2.4 WAMP

WAMPs are packages of independently created programs installed on computers that use a Microsoft Windows operating system. WAMP is an acronym formed from the initials of the operating system Microsoft Windows and the principal components of the package: Apache, MySQL and one of PHP, Perl or Python. WAMP Server is a Windows web development environment. It allows you to create web applications with Apache2, PHP and a MySQL database. Alongside, PhpMyAdmin allows you to manage easily your database.

2.5 Apache

Apache is an open source web server platform. Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. Apache supports a variety of features, many implemented as compiled modules which extend the core functionality. These can range from server-side programming language support to authentication schemes. Apache provides a variety of MultiProcessing Modules (MPMs) which allow Apache to run in a process-based, hybrid (process and thread) or event-hybrid mode, to better match the demands of each particular infrastructure. Apache 2.0 differs from the previous versions by the much re-written code, which has considerably simplified its configuration and boosted its efficiency. It supports Ipv6, UNIX threading, other

protocols such as mod_echo. This version also offers a new compilation system and multi-language error messaging.

2.6 MySQL

MySQL (also known as My sequel) is the world's most widely used open source relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP (WAMP for Windows) open source web application software stack

2.7 PHP

PHP, which stands for "**PHP: Hypertext Preprocessor**" is a widely-used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. Its syntax draws upon C, Java, and Perl. **PHP** is a server-side scripting language designed for web development but also used as a general-purpose programming language. The main goal of the language is to allow web developers to write dynamically generated web pages quickly, but one can do much more with PHP.

What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server, generating HTML which is then sent to the client. The client would receive the results of running that script, but would not know what the underlying code was.

PHP code is interpreted by a web server with a PHP processor module which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical

2.8 NetBeans IDE 7.3

NetBeans IDE lets you quickly and easily develop Java desktop, mobile, and web applications, while also providing great tools for PHP and C/C++ developers. It is free and open source and has a large community of users and developers around the world. The NetBeans IDE is written in Java and can run on Windows, OS X, Linux, Solaris and other platforms supporting a compatible JVM.

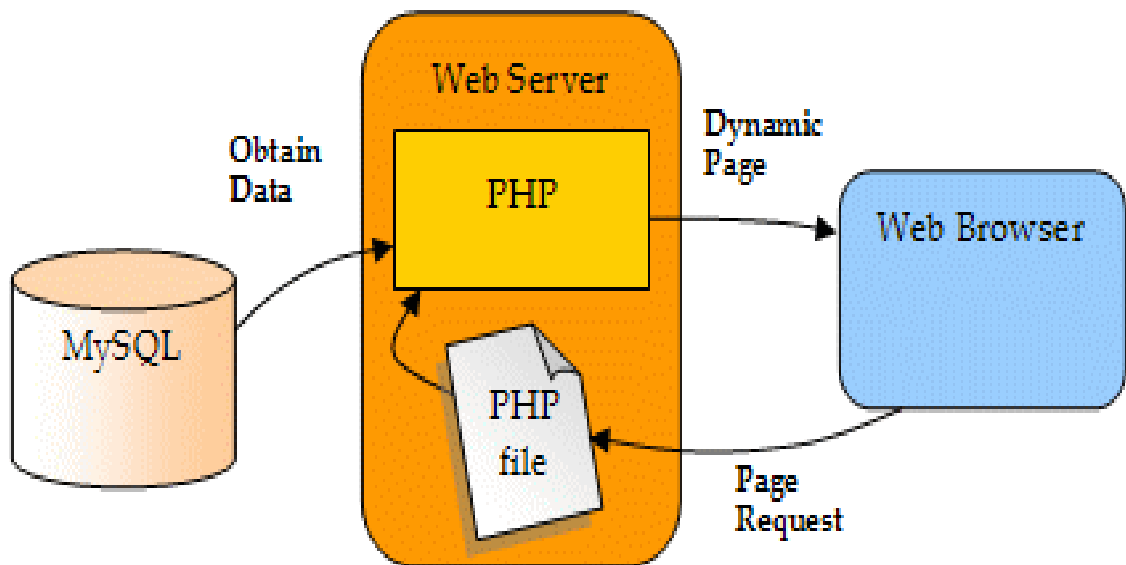
NetBeans supports PHP since version 6.5. The bundle for PHP includes:

- syntax highlighting, code completion, occurrence highlighting, error highlighting, CVS version control
- semantic analysis with highlighting of parameters and unused local variables
- PHP code debugging with xdebug.
- PHP Unit testing with PHPUnit and Selenium.

- Code coverage.
- Zend Framework support (since version 6.9)
- PHP 5.3 namespace and closure support (since version 6.8).
- Code Folding for Control Structures (since version 7.2 dev).

2.9 Block Diagram

Initially, a browser sends a HTTP request to a web server (Apache in this case) asking for a resource. The Web Server searches for the requested resource and if it is a php file then the server invokes the php interpreter. The interpreter then interprets the code between ‘<?php’ and ‘?>’. If it requires data to be fetched from the database, a connection will be established between the php interpreter and the database (MySQL in this case) and the corresponding execution takes place. The results will be forwarded to the web server and then to the browser



Block Diagram

CHAPTER – 3

SYSTEM DESIGN

Systems design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could see it as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering.

A good website/web application starts life in the design stage. There are several aspects of the site that are formed at this stage, including among other things, layout, color, sound, content, functionality and maintainability. No reasonable person would start to build a house without designing it first, no reasonable website builders should begin construction without a design either.

Object-oriented analysis and design methods are becoming the most widely used methods for computer systems design. The UML (Unified Modeling Language) has become the standard language in object-oriented analysis and design.

Unified Modeling Language (UML) combines techniques from data modeling (entity relationship diagrams), business modeling (work flows), object modeling, and component modeling. It can be used with all processes, throughout the software development life cycle, and across different implementation technologies.

There are two types of design variations. They are

- Logical Design
- Physical Design

3.1 Requirements for System Design

We have used the following to design the system for University Notification Management.

3.1.1 Star UML

StarUML was an open source UML tool, licensed under a modified version of GNU GPL. After being abandoned for some time, the project had a last revival to move from Delphi to Java/Eclipse and then stopped again. However, the community is still active.

The stated goal of the project was to replace larger, commercial applications such as Rational Rose and Borland Together.

StarUML supports most of the diagram types specified in UML 2.0. It is currently missing object, package, timing and interaction overview diagrams (though the first two can be adequately modeled through the class diagram editor). StarUML uses its own file format, with the *.uml* extension.

3.1.2 Smart Draw

SmartDraw is a visual processor used to create flowcharts, organization charts, mind maps, project charts, and other visuals. SmartDraw is built exclusively for Windows operating systems and works with Windows 7, Vista, XP, and 2000. Since version 7, it uses Microsoft's Fluent User Interface in conjunction with automated panels specific to each type of diagram.

It integrates with Microsoft Word, Excel, PowerPoint and Microsoft Project; it can export diagrams to common image formats and PDF format.

3.2 Logical Design

The logical design of a system pertains to an abstract representation of the data flows, inputs and outputs of the system. This is often conducted via modelling, using an over-abstract (and sometimes graphical) model of the actual system. In the context of systems design are included. Logical design includes ER Diagrams i.e. Entity Relationship Diagrams.

3.2.1 Application Design

3.2.1.1 Use Case Diagram

A **use case diagram** at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other types of diagrams as well.

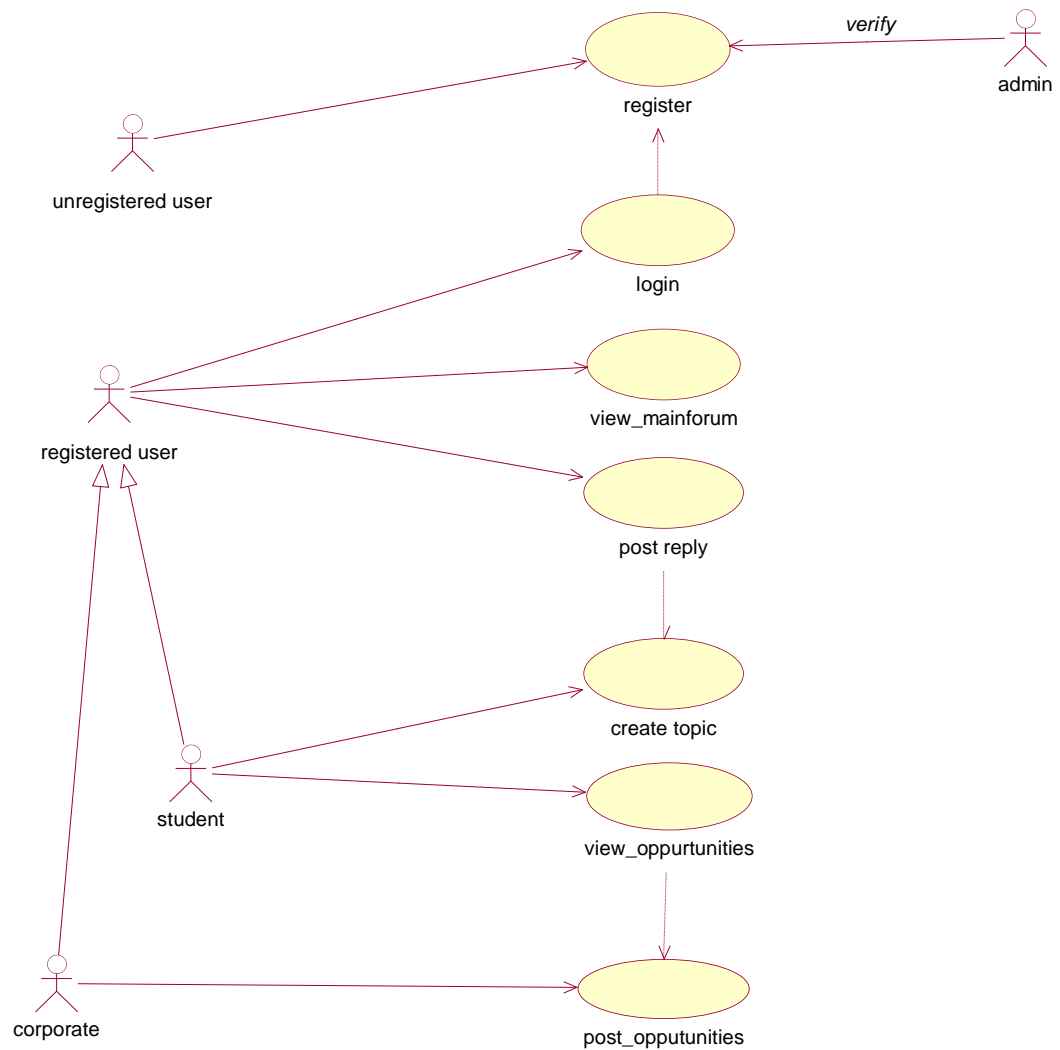


Fig 3.1: Use Case Diagram of the System

3.2.1.2 Class Diagram

A **class diagram** in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes.

The class diagram is the main building block of object oriented modeling. It is used both for general conceptual modeling of the semantics of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed.

In the diagram, classes are represented with boxes which contain three parts:

- The upper part holds the name of the class
- The middle part contains the attributes of the class
- The bottom part gives the methods or operations the class can take or undertake

In the design of a system, a number of classes are identified and grouped together in a class diagram which helps to determine the static relations between those objects. With detailed modeling, the classes of the conceptual design are often split into a number of subclasses.

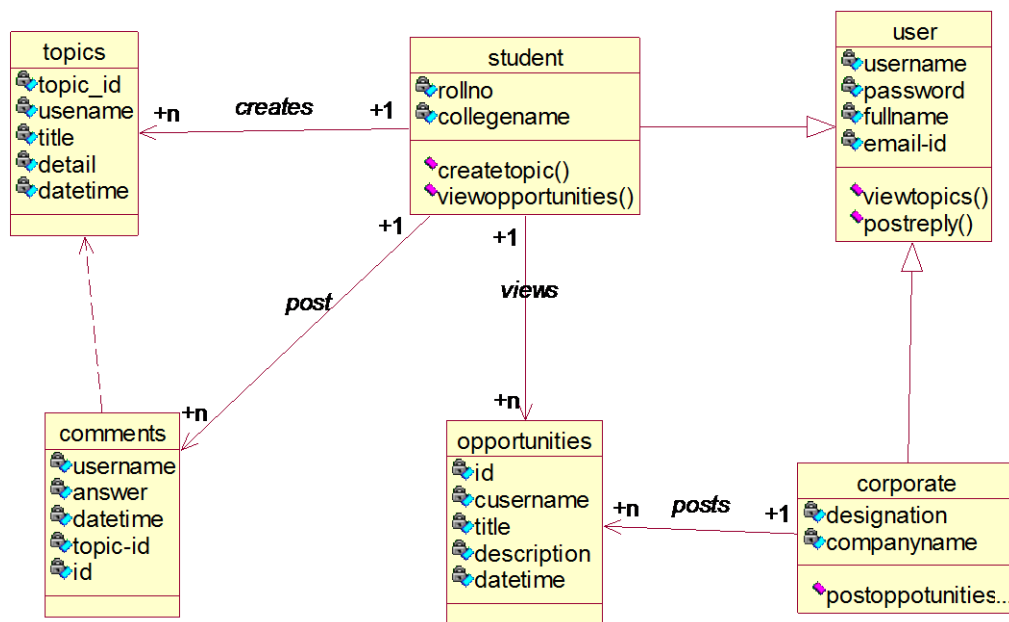


Fig: 3.2 Class Diagram of the System

3.2.1.3 State chart Diagram

A **state diagram** is a type of diagram used in computer science and related fields to describe the behavior of systems. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction. Many forms of state diagrams exist, which differ slightly and have different semantics.

State diagrams are used to give an abstract description of the behavior of a system. This behavior is analyzed and represented in series of events that could occur in one or more possible states. Hereby "each diagram usually represents objects of a single class and tracks the different states of its objects through the system". State diagrams can be used to graphically represent finite state machines.

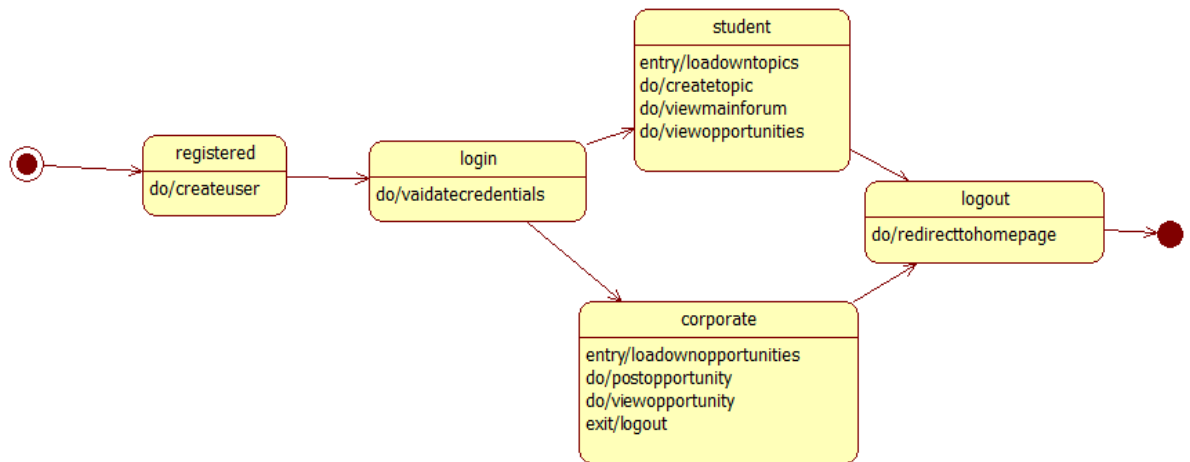


Fig 3.3: State Chart Diagram of Admin

3.2.1.4 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

Activity diagrams are constructed from a limited number of shapes, connected with arrows. The most important shape types:

- *rounded rectangles* represent *actions*;
- *diamonds* represent *decisions*;
- *bars* represent the start (*split*) or end (*join*) of concurrent activities;
- a *black circle* represents the start (*initial state*) of the workflow;
- *Encircled black circle* represents the end (*final state*).

Arrows run from the start towards the end and represent the order in which activities happen. Hence they can be regarded as a form of flowchart. Typical flowchart techniques lack constructs for expressing concurrency. However, the join and split symbols in activity diagrams only resolve this for simple cases; the meaning of the model is not clear when they are arbitrarily combined with decisions or loops.

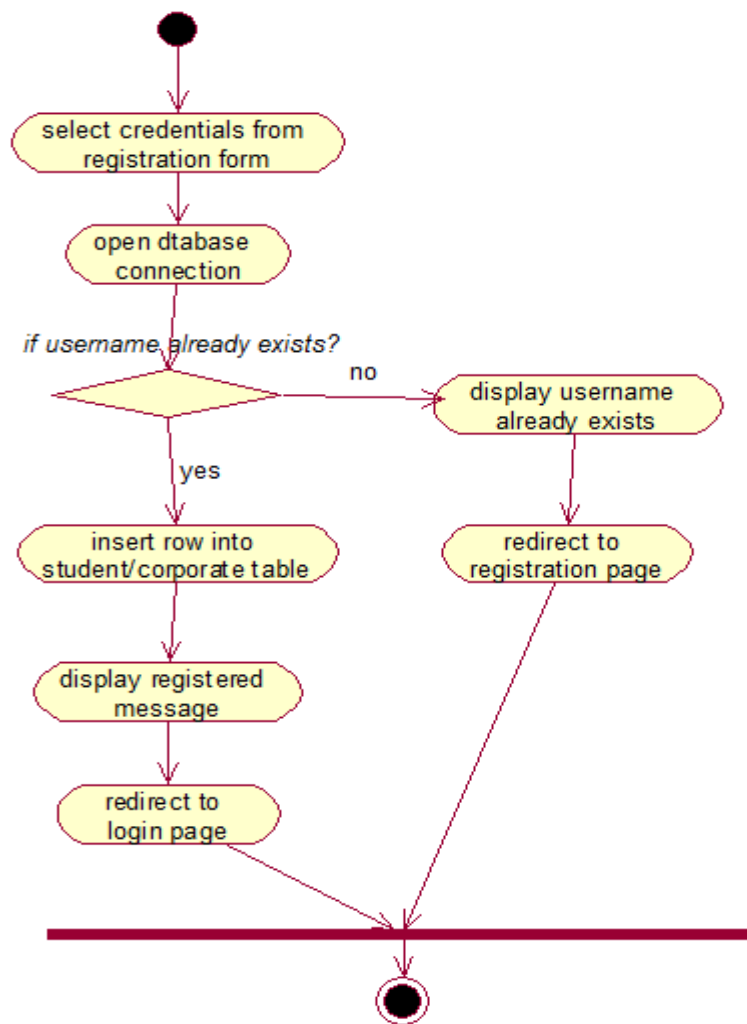


Fig 3.4: Activity diagram for registration

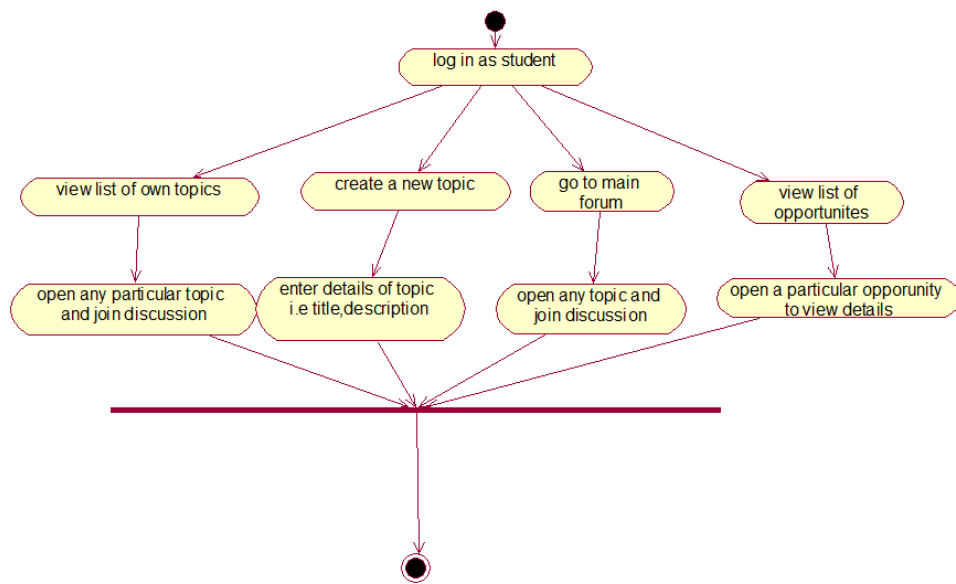


Fig 3.5: Activity diagram for student

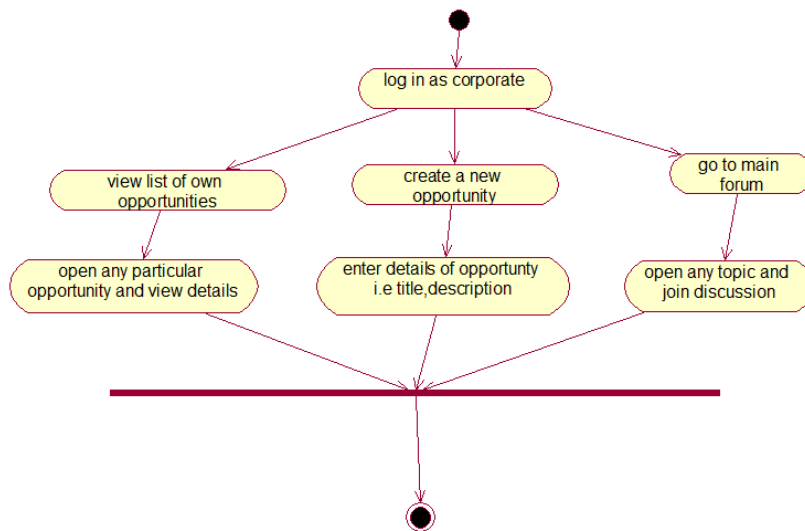


Fig 3.6: Activity diagram for corporate

3.2.1.5 Interaction Diagrams:

From the name *Interaction* it is clear that the diagram is used to describe some type of interactions among the different elements in the model. So this interaction is a part of dynamic behavior of the system.

This interactive behavior is represented in UML by two diagrams known as *Sequence diagram* and *Collaboration diagram*. The basic purposes of both the diagrams are similar.

Sequence diagram emphasizes on time sequence of messages and collaboration diagram emphasizes on the structural organization of the objects that send and receive messages.

The purposes of interaction diagrams are to visualize the interactive behavior of the system. Now visualizing interaction is a difficult task. So the solution is to use different types of models to capture the different aspects of the interaction.

That is why sequence and collaboration diagrams are used to capture dynamic nature but from a different angle.

So the purposes of interaction diagram can be describes as:

To capture dynamic behavior of a system.

To describe the message flow in the system.

To describe structural organization of the objects.

To describe interaction among objects.

3.2.1.5.1 Sequence Diagram

A **sequence diagram** is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called **event diagrams**, **event scenarios**, and **timing diagrams**.

A sequence diagram shows, as parallel vertical lines (*lifelines*), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

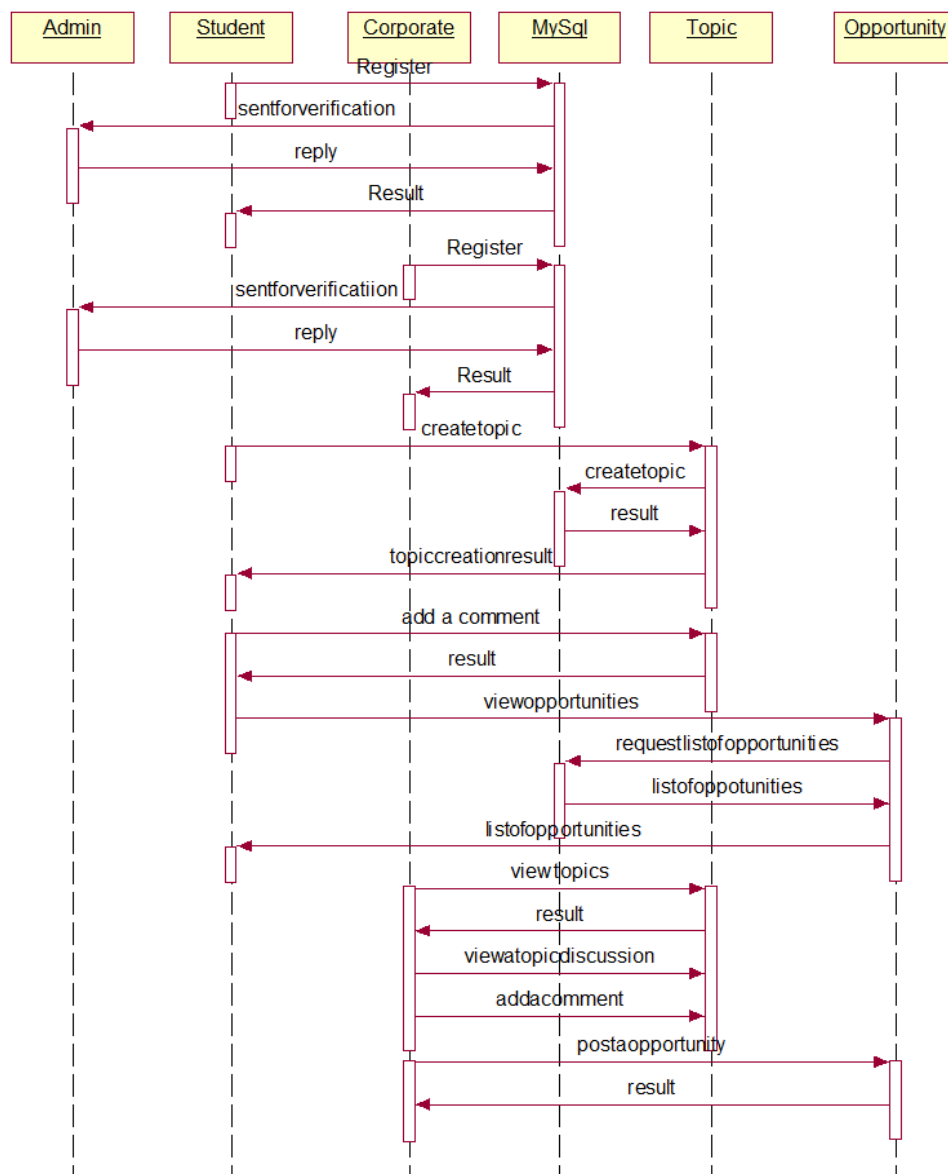
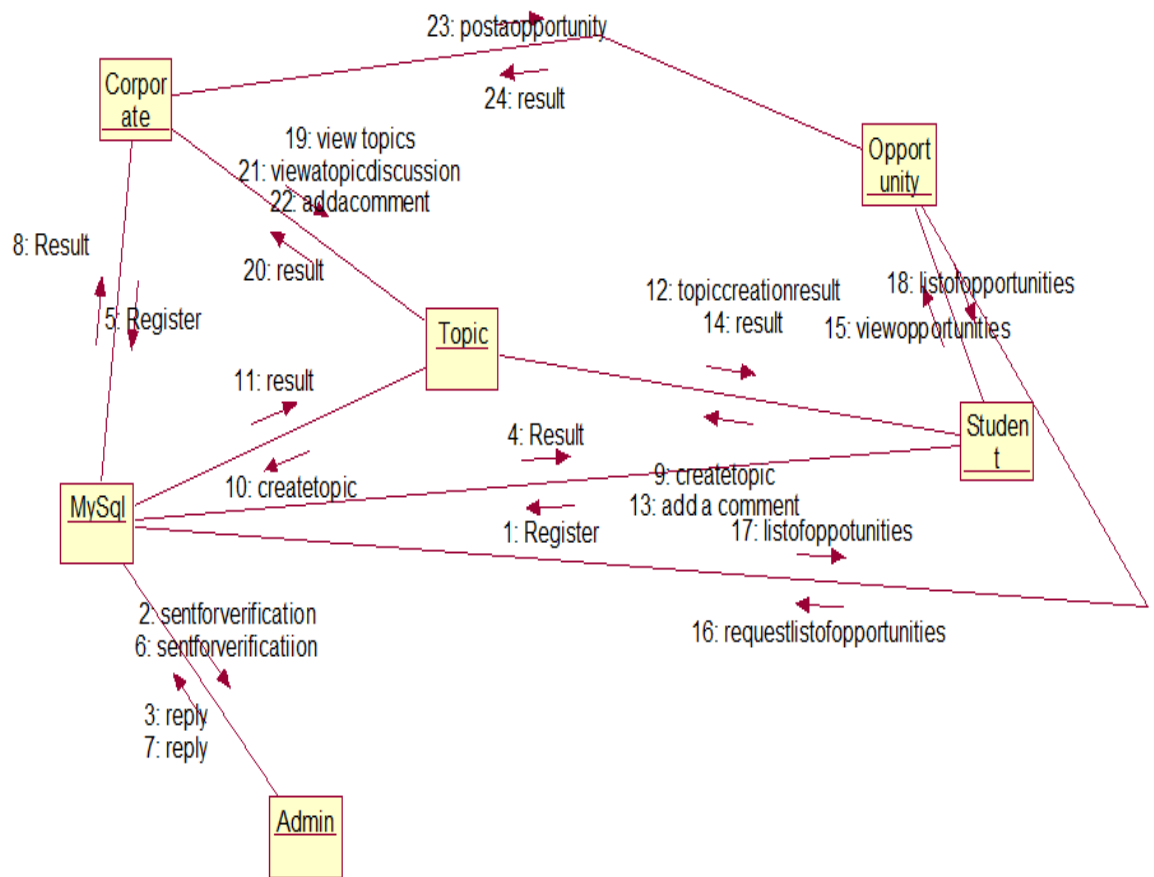


Fig 3.7: Sequence Diagram for student-corporate

3.2.1.6 Collaboration Diagram:

A Communication diagram models the interactions between objects or parts in terms of sequenced messages. Communication diagrams represent a combination of information taken from Class, Sequence, and Use Case Diagrams describing both the static structure and dynamic behavior of a system.



Collaboration Diagram for student-corporate

3.2.1.7 Implementation Diagram

3.2.1.7.1 Component Diagram:

In the Unified Modeling Language, a **component diagram** depicts how components are wired together to form larger components and or software systems. They are used to illustrate the structure of arbitrarily complex systems.

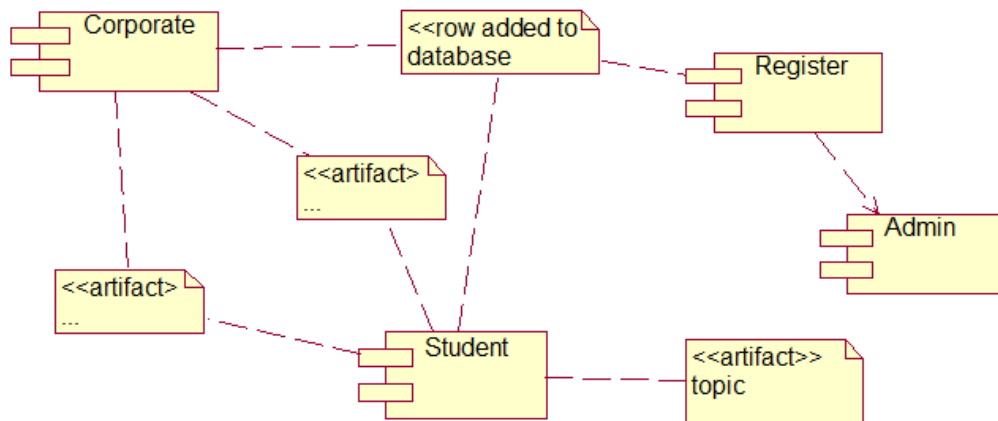
Components are wired together by using an *assembly connector* to connect the required interface of one component with the provided interface of another component. This illustrates the *service consumer* - *service provider* relationship between the two components.

An *assembly connector* is a "connector between two components that defines that one component provides the services that another component requires. An assembly connector is a connector that is defined from a required interface or port to a provided interface or port"

When using a component diagram to show the internal structure of a component, the provided and required interfaces of the encompassing component can delegate to the corresponding interfaces of the contained components.

A *delegation connector* is a "connector that links the external contract of a component (as specified by its ports) to the internal realization of that behavior by the component's parts."

The example above illustrates what a typical Insurance policy administration system might look like. Each of the components depicted in the above diagram may have other component diagrams illustrating their internal structure



Component diagram

3.2.1.7.2 Deployment Diagram

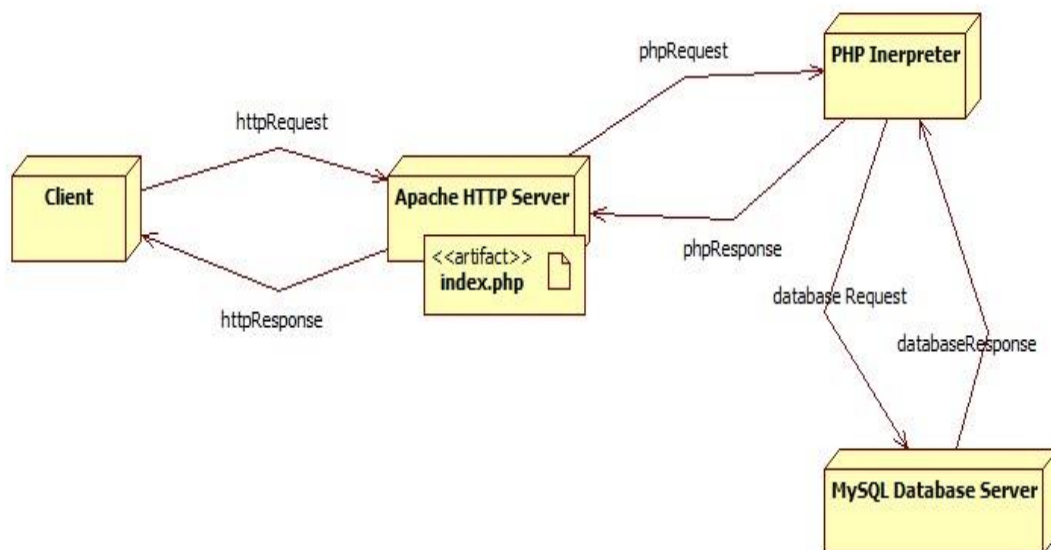
A **deployment diagram** in the Unified Modeling Language models the *physical* deployment of artifacts on nodes. To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have sub nodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

There are two types of Nodes.

1. Device Node
2. Execution Environment Node

Device nodes are physically computing resources with processing memory and services to execute software, such as typical computer or mobile phones. An execution environment node (EEN) is a software computing resource that runs within an outer node and which itself provides a service to host and execute other executable software elements.



Deployment Diagram

3.2.2 Data Design

3.2.2.1 Entity -Relationship Diagram

An ER model is an abstract way of describing a database. In the case of a relational database, which stores data in tables, some of the data in these tables point to data in other tables - for instance, your entry in the database could point to several entries for each of the phone numbers that are yours. The ER model would say that you are an entity, and each phone number is an entity, and the relationship between you and the phone numbers is 'has a phone number'. Diagrams created to design these entities and relationships are called entity–relationship diagrams or ER diagrams.

Using the three schema approach to software engineering, there are three levels of ER models that may be developed.

- Conceptual data model

This is the highest level ER model in that it contains the least granular detail but establishes the overall scope of what is to be included within the model set. The conceptual ER model normally defines master reference data entities that are commonly used by the organization. Developing an enterprise-wide conceptual ER model is useful to support documenting the data architecture for an organization.

A conceptual ER model may be used as the foundation for one or more *logical data models* (see below). The purpose of the conceptual ER model is then to establish structural metadata commonality for the master data entities between the set of logical ER models. The conceptual data model may be used to form commonality relationships between ER models as a basis for data model integration.

- Logical data model

A logical ER model does not require a conceptual ER model, especially if the scope of the logical ER model is to develop a single disparate information system. The logical ER model contains more detail than the conceptual ER model. In addition to master data entities, operational and transactional data entities are now defined. The details of each data entity are developed and the entity relationships between these data entities are established. The logical ER model is however developed independent of technology into which it will be implemented.

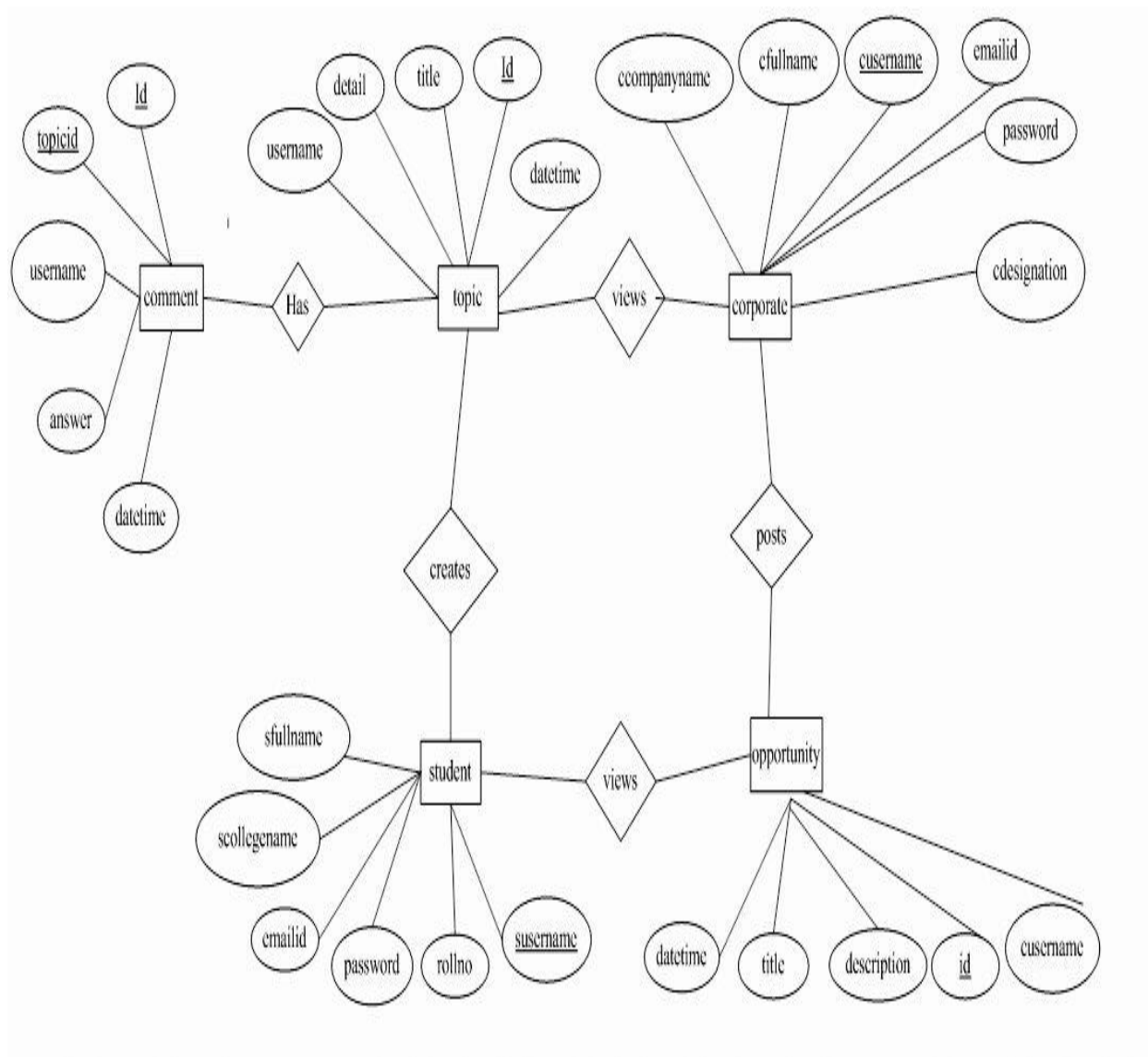
- Physical model

One or more physical ER models may be developed from each logical ER model. The physical ER model is normally developed to be instantiated as a database. Therefore, each physical ER model must contain enough detail to produce a database and each physical ER model is technology dependent since each database management system is somewhat different.

The physical model is normally forward engineered to instantiate the structural metadata into a database management system as relational database objects such as database tables, database indexes such as unique key indexes, and database constraints such as a foreign key constraint or a commonality constraint. The ER model is also normally used to design modifications to the relational database objects and to maintain the structural metadata of the database.

The first stage of information system design uses these models during the requirements analysis to describe information needs or the type of information that is to be stored in a database. The data modeling technique can be used to describe any ontology (i.e. an overview and classifications of used terms and their relationships) for a certain area of interest. In the case of the design of an information system that is based on a database, the conceptual data model is, at a later stage (usually called logical design), mapped to a logical data model, such as the relational model; this in

turn is mapped to a physical model during physical design. Note that sometimes, both of these phases are referred to as "physical design". It is also used in database management system.



E-R diagram

3.3 Physical Design

The physical design relates to the actual input and output processes of the system. This is laid down in terms of how data is input into a system, how it is verified/ authenticated, how it is processed, and how it is displayed as output. In Physical design, following requirements about the system are decided.

1. Input requirement,
2. Output requirements,
3. Storage requirements,
4. Processing Requirements,
5. System control and backup or recovery.

Put another way, the physical portion of systems design can generally be broken down into three sub-tasks:

1. User Interface Design
2. Data Design
3. Process Design

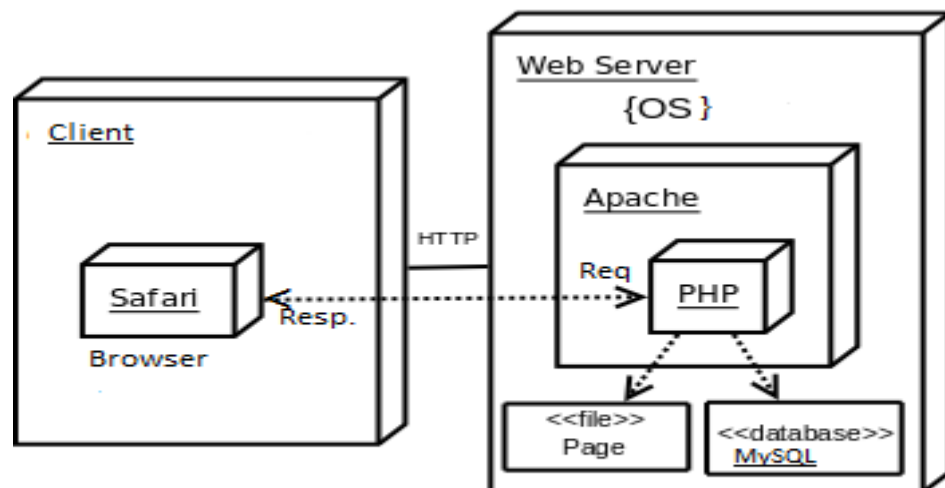
User Interface Design is concerned with how users add information to the system and with how the system presents information back to them. Data Design is concerned with how the data is represented and stored within the system. Finally, Process Design is concerned with how data moves through the system, and with how and where it is validated, secured and/or transformed as it flows into, through and out of the system. At the end of the systems design phase, documentation describing the three sub-tasks is produced and made available for use in the next phase.

Physical design, in this context, does not refer to the tangible physical design of an information system. To use an analogy, a personal computer's physical design involves input via a keyboard, processing within the CPU, and output via a monitor, printer, etc. It would not concern the actual layout of the tangible hardware, which for a PC would be a monitor, CPU, motherboard, hard drive, modems, video/graphics cards, USB slots, etc. It involves a detailed design of a user and a product database structure processor and a control processor. The H/S personal specification is developed for the proposed system

3.3.1 System Architecture

The Architecture of the system comprises of a web server, Apache in this case, a PHP Interpreter and a MySQL database. The client uses a web browser to connect to the system. The client requests for a resource in the web server. The request follows HTTP – HyperText Transfer Protocol. The client initiates the request by sending a HttpRequest to the web server. The server fetches the required resource and sends it back to the client by using HttpResponse. If the client requests for any PHP file, then the server processes the PHP script in the file and sends the HTML document to the client. Whenever the server encounters PHP Script, that is, whenever it encounters `<?php /*code*/ ?>`, it invokes the PHP interpreter. The PHP interpreter executes the PHP code inside the script tag and returns a HTML response to the server.

Occasionally the PHP code might request data to be fetched from the database. In such situations, the PHP Interpreter establishes a connection with the specified database, MySQL in our application, and fetches the result. The Interpreter then includes the results in the HTML document that is later returned to the web server. The following block diagram shows the typical architecture of the system. The figure contains a client that sends a HTTP Request and a Web Server that sends a HTTP Response back to the client. The web Server has PHP interpreter and has connections to MySQL database. We use WAMP as a web server for the development of our system. WAMP contains Apache web server, PHP Interpreter and MySQL database server embedded into its package that is suitable to run in Windows Environment.

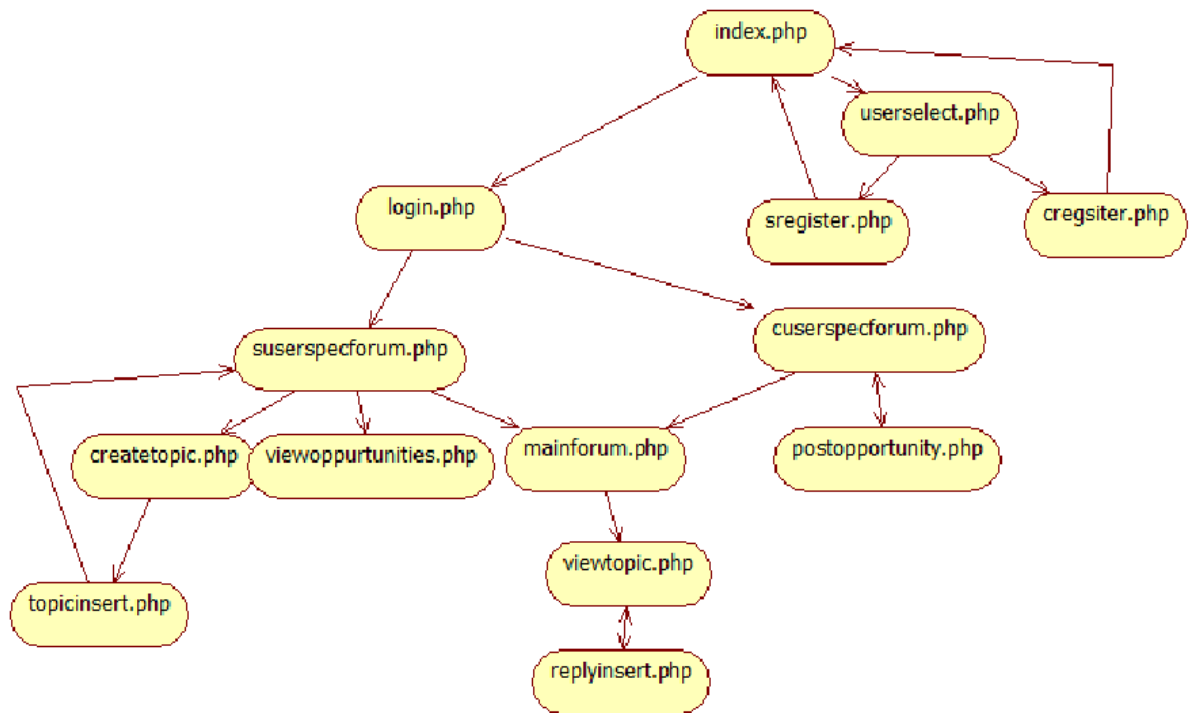


Apache PHP MySQL architecture

Fig 3.15: System Architecture

3.3.2 System Components

3.3.2.1 Component Flow Diagram



Component flow diagram

3.3.2.2 Component Description

3.3.2.2.1 Index.php

(This is the default Home page of the Student Connect web application)

Description: This Home page contains the Login option to both the registered users ie., the student as well as the corporate. Else, if the candidate is a new user, clicking on the newuser button redirects to the registration page.

Input: Username, Password.

Processing: The user after entering the username and password clicks login button and the details will be forwarded to `login.php` page which is responsible for validation.

3.3.2.2.2 **userselect.php**

Input: Selection of User type ie., Student or Corporate.

Processing: This is First page of registration where the user has an option of selecting the nature of candidature. This page contains Student and Corporate buttons. On selecting the Student the page redirects to sregister.php and user fills the credentials. On selecting corporate the page redirects to cregister.php and user fills all the credentials.

Output: The page is redirected to /sregister.php if the user is a student.

The page is redirected to /cregister.php if the user is a corporate.

3.3.2.2.3 **sregister.php**

Input: Full Name, College Name, Roll No, Email id, Username, Password

Processing: The Student enters the above mentioned details to register .On registering a new row entry is made in the student table in the name of the user (uname). And the details stored here will be used as a reference in the later stages.

Output: On successful insertion a message will be displayed to the user *registration successful*. If the user doesn't enter valid details or leave any details unfilled, then an error message is displayed.

3.3.2.2.4 **cregister.php**

Input: Full Name, Company Name, Designation, Email id, Username, Password

Processing: The Corporate enters the above mentioned details to register .On registering a new row entry is made in the corporate table in the name of the user (uname). And the details stored here will be used as a reference in the later stages.

Output: On successful insertion a message will be displayed to the user *registration successful*. If the user doesn't enter valid details or leave any details unfilled, then an error message is displayed.

3.3.2.2.5 **login.php**

Input: username, password forwarded from index.php.

Processing: Then the process of validation starts as follows: Firstly, the entered username and password are checked in the student table. If the record exists then the user will be logged in as student, otherwise the entered username and password are checked in the corporate table. If the record exists then the user will be logged in as corporate. It will collect the username and password from the user and the data will be forwarded to the validate.php for verification.

Output: The user will be redirected to the respective home page of the student/corporate.

3.3.2.2.6 config.php

Input:username,password.

Processing: This file will try to establish a connection to the database, if failed then *failed to connect MySQL* message will be displayed. This file is included in almost all other files where the database connection is required. This also has additional functionality to check whether the user is authorized to the particular page or not.

Output: Successful connection establishment/ Error in connection establishment

3.3.2.2.7 suserspecforum.php?id=<username>

Input: susername, password

Processing: The student has permission to his/her own list of own topics created. The user can also view the topics created by the other students. The other functionality of the student is to view other topics, involve in discussion by commenting on the topics. The user also has an option to view opportunities posted by corporates. The date and time are also displayed.

Output: The corresponding pages will be displayed as per the users (student) action.

3.3.2.2.8 createtopic.php

Inputs:Title,Description

Processing: The create topic page allows the student in creating new topic with a title and a description. After completion and submission of the topic, a corresponding table row gets added to the forum_question table. The table contains a unique id to the topic, which gets auto incremented on creating a new topic. The student can also reset the topic.

Output:The topic entered will be displayed in the list of topics.

3.3.2.2.9 mainforum.php

Processing:The mainforum page contains the list of all the topics created by the students. Any student or the corporate can view and post comments on the topics created by a specific student.

The page displays the topics in a sorted manner, the most recently created topics appears on the top. The page also displays time and date of creation of the topic.

3.3.2.2.10 viewtopic.php

Input: User (Student) clicking on the hyperlink of the respective topic.

Processing: The selected topic displays the following: Title, Description, Date Time and the author of the topic. The Title contains the////////

Output: Display of the topic and comments posted by all the users.

3.3.2.2.11 viewopportunities.php

Input: User (Student) clicking on the hyperlink of the view opportunities.

Processing: The list of opportunities displays the following: Title, Description, Date Time and the author of the opportunity. The Title contains the opportunity posted by the corporate. The description details about the opportunity posted. If any job vacancy is provided in opportunity, then the user (student) only can apply for the post through offline..

Output: Display of the topic and comments posted by all the users.

3.3.2.2.12 cuserspecforum.php

Input: The corporate after login in gets redirected to the *cuserspecforum.php*

Processing: The opportunities posted by the respective corporate are retrieved from database

The list of opportunities along with the order and the date, time are displayed in the page. Only the corporate can view his/her opportunities and restricted in viewing opportunities posted by other corporates

Output: The page redirects to post opportunities or main forum based on the user selection.

3.3.2.2.13 postopportunities.php

Input: The Title, Description about the opportunities.

Processing: The post opportunities page allows the corporate in posting new opportunities with a title and a description. After completion and submission of the opportunity, a corresponding table row gets added to the opportunity table. The table contains a unique id to each opportunity, which gets auto incremented on posting new opportunity. The corporate can also reset the opportunity.

Output: The opportunity entered will be displayed in the list of opportunities.

.

Database Table Description

3.4 Database Tables

3.4.1 Table-1: student

sfullname	scollegename	srollno	semailid	susername	spassword	sverify
Ravinder Putta	MVSR	245109733038	ravi060791ster@gmail	ravi060791	135621	1
o paramesh	mvsr	245109733014	paramesh014@gmail.co	paramesh014@gmail.co	paramesh	1
Lohith Kumar Menchu	MVSR	11	lovemenchu@gmail.com	lkmenchu	135621	1

The student table contains the details of students ie., full name, college name,rollno,email id,username,password.The attributes sfullname,scollegename,srollno,semail id are stored in the database for each student.Whereas,the susername,spassword are also stored for login purpose. . “susername” is considered as the primary key of the table.susername once assigned can’t be reassigned again. The sverify is assigned by admin to authenticate the details of user.

3.4.2 Table-2: corporate

cfullname	ccompanyname	cdesignation	cemail	cusername	cpassword	cverify
Purnima	MVSR Engineering College	HR	purnima.janaki@gmail.com	purnima.janaki@gmail.com	puri	1
sandeep	MVSR Engineering College	HR	sandeepragila@gmail.com	sandeep	sandeep	1
Pavan	MVSR Engineering College	System Anlayst	pavan@gmail.com	pvn	98480	1
Lohith Kumar Menchu	MVSR	HR	lkmenchu@gmail.com	Clkmenchu	135621	1

The corporate table contains the details of corporate ie., full name, company name,designation,emailid,username,password.The attributes cfullname,ccompanyname,cdesignation,cemail id are stored in the database for each corporate.Whereas,the cusername,cpassword are also stored as well used for login purpose. . “cusername” is considered as the primary key of the table.usernameonce assigned can’t be reassigned again. The cverify is assigned by admin to authenticate the details of corporate.

3.4.3 Table-3: forum_question

id	topic	detail	username	datetime
11	ajax	Acronym for Asynchronous JavaScript and XML)[1] is...	ravi060791	13/03/13 09:45:55
12	bigdata	Big data[1][2] is a collection of data sets so lar...	ravi060791	15/03/13 08:16:28
13	lol	header("Location:/studentconnect/suserspecforum.p...	ravi060791	15/03/13 08:24:19
16	Report	ABSTRACT: In this presentation we present Google, ...	ravi060791	23/03/13 01:56:26
17	University Campus Social Network System for Knowle...	Introduction: Online social network services...	ravi060791	23/03/13 04:14:17
21	DBMS	A database is an organized collection of data. The...	ravi060791	05/04/13 06:14:45
22	Software as Service	Software as Service Software as Service Software a...	lkmenchu	01/05/13 05:07:46

The forum question table contains the details of any topic created by a student. The fields are id, topic,detail,datetime.the id is unique identifier assigned to each topic which is auto incremented on creation of a new this filed gets auto incremented. Topic field contains the name of the project, detail field contains the description of the of the project. The username filed references the susername field in the student table to keep track of the author of the table.datetime field is useful in sorting the topics in temporial order.

3.4.4 Table-4: forum_answer

question_id	id	username	answer
1	2	ravi060791	i do not know
1	3	ravi060791	i learned few things since my reply
1	4	ravi060791	i learned few things since my reply
1	5	ravi060791	helloo

The forum_answer table contains the following fields question_id,id, username, answer fileds.the question_id references id filed in forum_question tableand useful in keeping track of answers/comments corresponding to a particular topic. The id filed is unique number given to each answer/comment and is the primary key.username field is used to display the author of the comment/answer.

3.4.5 Table-5: opportunity

id	title	description	username	datetime
1	assistant engineer	assistant engineer capgemini salary:1000000000000...	purima.janaki@gmail.com	2015-03-13 08:34:19
2	sandeep	sandeepsandeepsandeepsandeepsandeepsandeepsandeeps...	sandeep	2015-03-13 09:35:33
3	sandeep	sandeepsandeepsandeepsandeepsandeepsandeepsandeeps...	sandeep	2015-03-13 10:51:06
4	System Engineer	Vacnacy for the post of software system engineer. ...	jo	2001-05-13 05:10:40

The opportunity table contains the fields like id, title, description, username,datetime. The id field is used to uniquely identify an opportunity and is the primary key.The username field in this table references username filled in the coporate table. Title and description fields describe the details of the opportunities. The opportunities can only be edited by the corporate who has created the opportunity. This can be viewed by all the students and the corporate himself.

CHAPTER 4

SYSTEM DEVELOPMENT

4.1 Requirements during Development

We used the following Tools to develop University Notification Management system.

- WAMP Server
- NetBeans IDE
- Browser

4.1.1 WAMP Server

WampServer is a Windows web development environment. It allows you to create web applications with Apache2, PHP and a MySQL database. Alongside, PhpMyAdmin allows you to manage easily your database.

WAMPs are packages of independently created programs installed on computers that use a Microsoft Windows operating system.

WAMP is an acronym formed from the initials of the operating system Microsoft Windows and the principal components of the package: Apache, MySQL and one of PHP, Perl or Python. Apache is a web server. MySQL is an open-source database. PHP, Perl and Python are scripting languages that can manipulate information held in a database and generate web pages dynamically each time content is requested by a browser. Other programs may also be included in a package, such as phpMyAdmin which provides a graphical user interface for the MySQL database manager.

4.1.1.1 Getting Started in WAMP

If the installation went well, you should have a new icon in the bottom right, where the clock is:

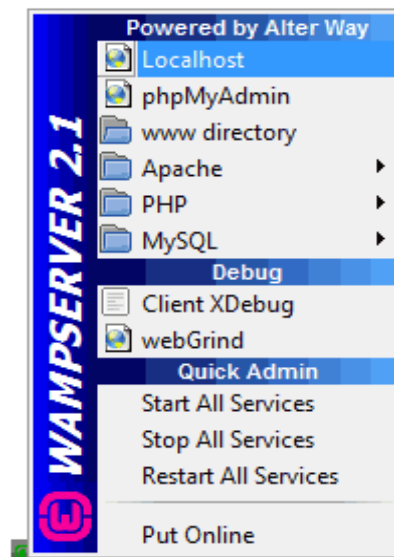


Fig 4.1: Getting started in WAMP-1

Click the icon to see the menu above.

From here, you can stop the server, exit it, view help files, and see the configuration pages.

Click on localhost, though, and you'll see this page appear: (Localhost just refers to the server running on your own computer. Another way to refer to your server is by using the IP address 127.0.0.1.)

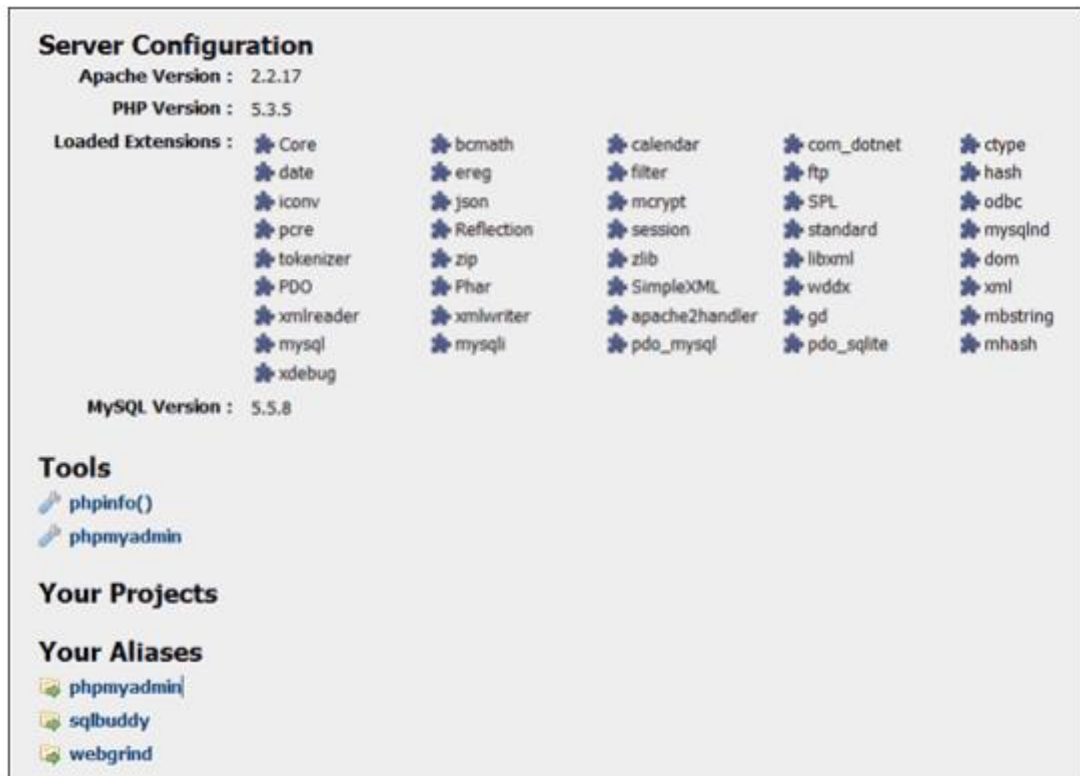


Fig 4.2: Getting started in WAMP - 2

Click the link under Tools that says `phpinfo()`. If all went well, you should be looking at the following page (The one below is a different php version, but don't worry about this - as long as you see something): The `info.php` page (click to open in a new window 66K)

If you saw the above page, then congratulations! Your PHP server is up and running, and you can make a start scripting PHP pages.

If you don't see the `info.php` page, then you will need to refer to the wampserver forums. The page you need is here: [Wampserver Forums](#)

Hopefully, your question will already have been asked and answered. (Unfortunately, we can't answer questions about Wampserver, as it's not our software.)

Saving your PHP files

Whenever you create a new PHP page, you need to save it in your WWW directory. You can see where this is by clicking its item on the menu:

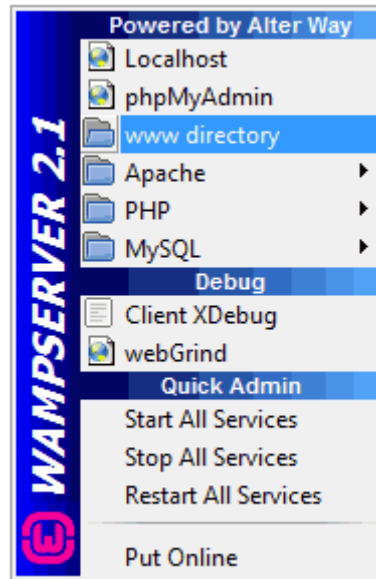


Fig 4.3: Getting started in WAMP – 3

When you click on www directory, you should see an explorer window appear. This one is from Windows Vista: (You'll probably have only two files, index and testmysql.)

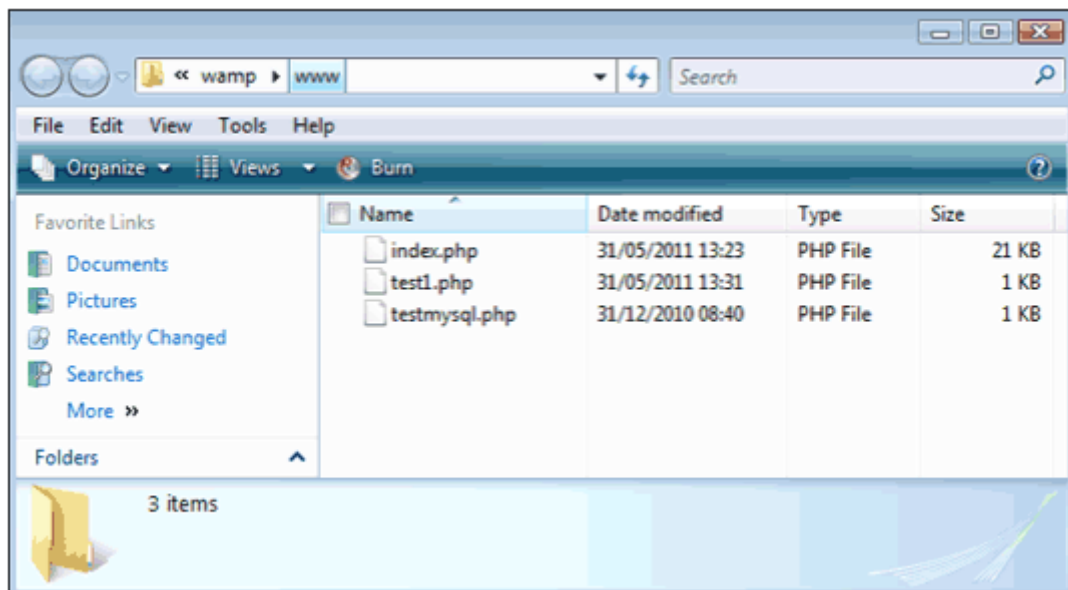


Fig 4.4: Getting started in WAMP – 4

This www folder for Wampserver is usually at this location on your hard drive:

c:/wamp/www/

Bear this in mind when you click File > Save As to save your PHP scripts.

4.1.1.2 Launching your PHP scripts

Suppose you have created a php script called test1.php. To launch this script, you need to add the script name after localhost in your browser. So instead of this:

`http://localhost/index.php`

You would type this:

`http://localhost/test1.php`

You don't type the name of the wamp folder, however. This would be wrong, for example:

`c:/wamp/www/test1.php`

As too would this:

`http://localhost/www/test1.php`

Your server knows where the www folder is, so you don't have to type it out: just add the script name to localhost. Likewise, if you create a folder under www then you'd just type this:

`http://localhost/folder_name/script_name.php`

4.1.2 NetBeans IDE

NetBeans is an integrated development environment (IDE) for developing primarily with Java, but also with other languages, in particular PHP, C/C++, and HTML5. It is also an application platform framework for Java desktop applications and others.

The NetBeans IDE is written in Java and can run on Windows, OS X, Linux, Solaris and other platforms supporting a compatible JVM. The NetBeans Platform allows applications to be developed from a set of modular software components called modules. Applications based on the NetBeans Platform (including the NetBeans IDE itself) can be extended by third party developers.

In this project we used Netbeans as an IDE to develop PHP code along with frontend languages like Html, Javascript and JQuery and CSS.

The following is the start page the start page that we will see when we open Netbeans IDE, Where we find all tutorials and demo, video tutorials and sample projects to work with Netbeans IDE.

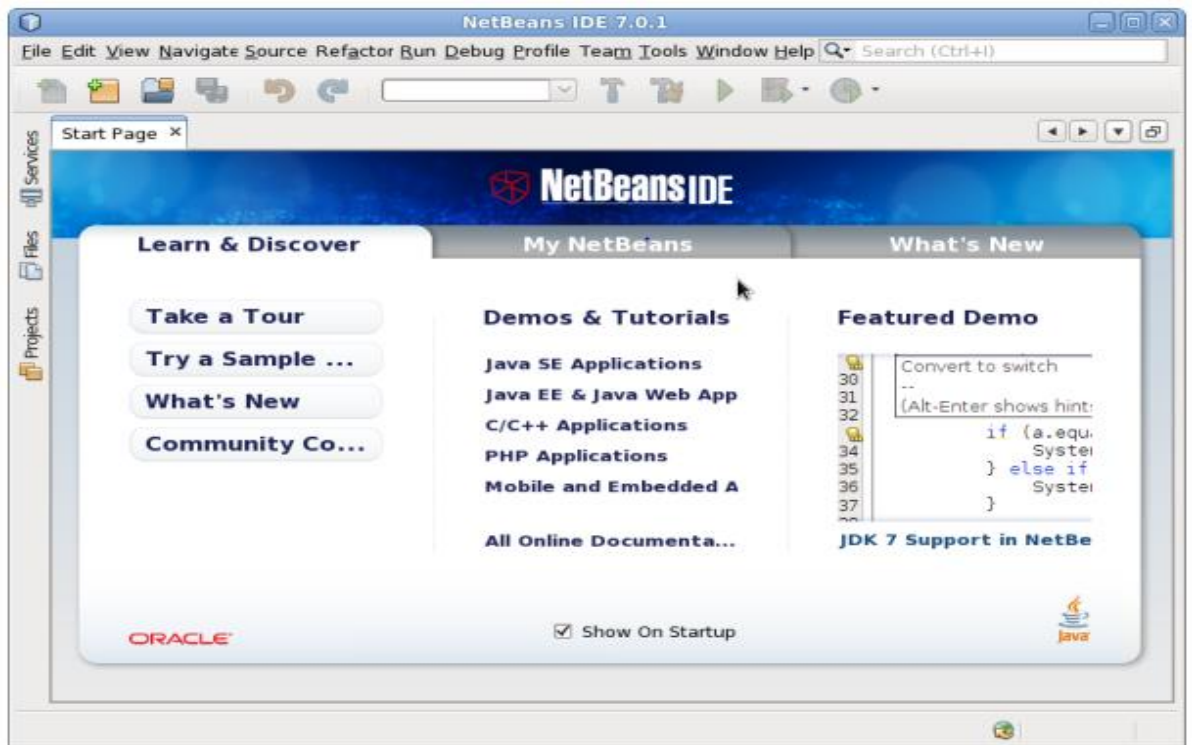


Fig 4.5: NetBeans Start Page

4.1.2.1 Setting up a PHP Project in the NetBeans IDE for PHP

For help in installing and starting NetBeans IDE, please see the installation documentation.

To start PHP development in the NetBeans IDE for PHP, you first need to create a project. A project contains the information on the location of the project files and the way you want to run and debug your application (run configuration).

1. Start the IDE, switch to the Projects window, and choose File > New Project. The Choose Project panel opens.
2. In the Categories list, choose PHP.
3. In the Projects area, choose PHP Application and click Next. The New PHP Project > Name and Location panel opens.
4. In the Project Name text field, enter NewPHPPProject.
5. In the Sources Folder field, browse for your PHP document root and create a subfolder there called NewPHPPProject. The document root is the folder where the web server looks for files to open in the browser. The document root is specified in the web server configuration file. For example, on Xampp, the document root is XAMPP_HOME/htdocs.

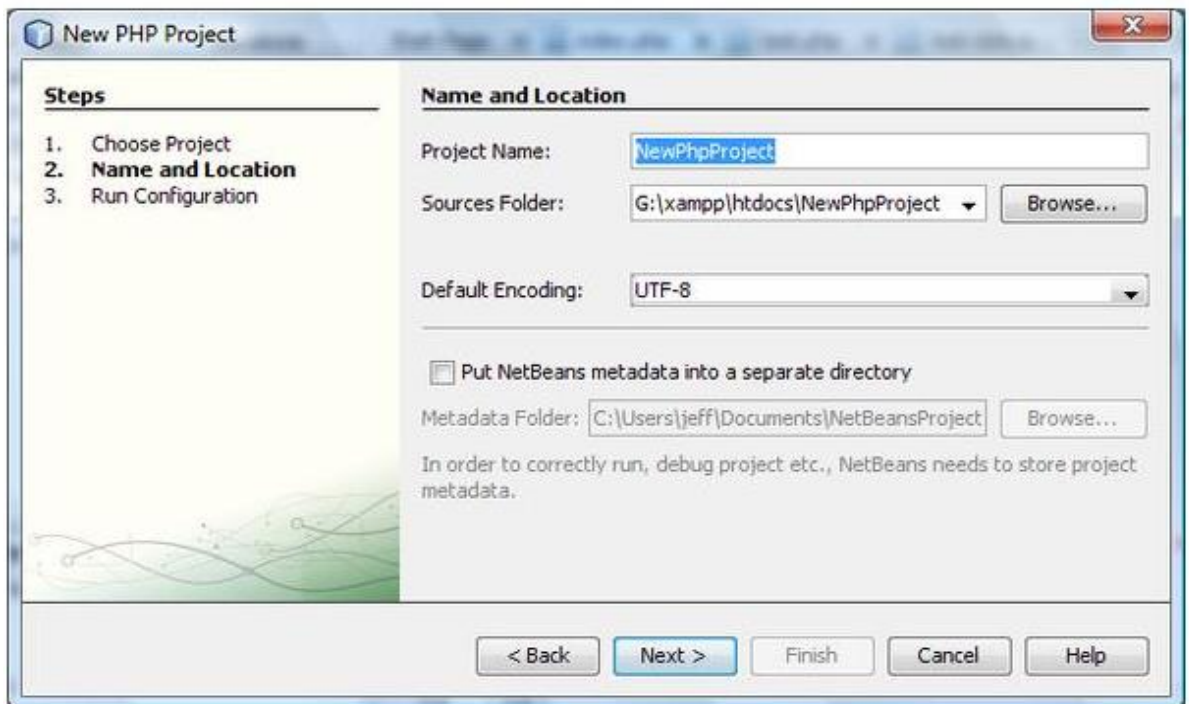


Fig 4.6: Setting Project in NetBeans-1

6. Leave all other fields with their default values. Click Next. The Run Configuration window opens.
7. In the Run As drop-down list, select Local Web Site. The project will run on your local Apache server. Your other options are to run the project remotely via FTP and to run it from the command line.
8. Leave the Project URL at default.
9. Click Finish. The IDE creates the project.

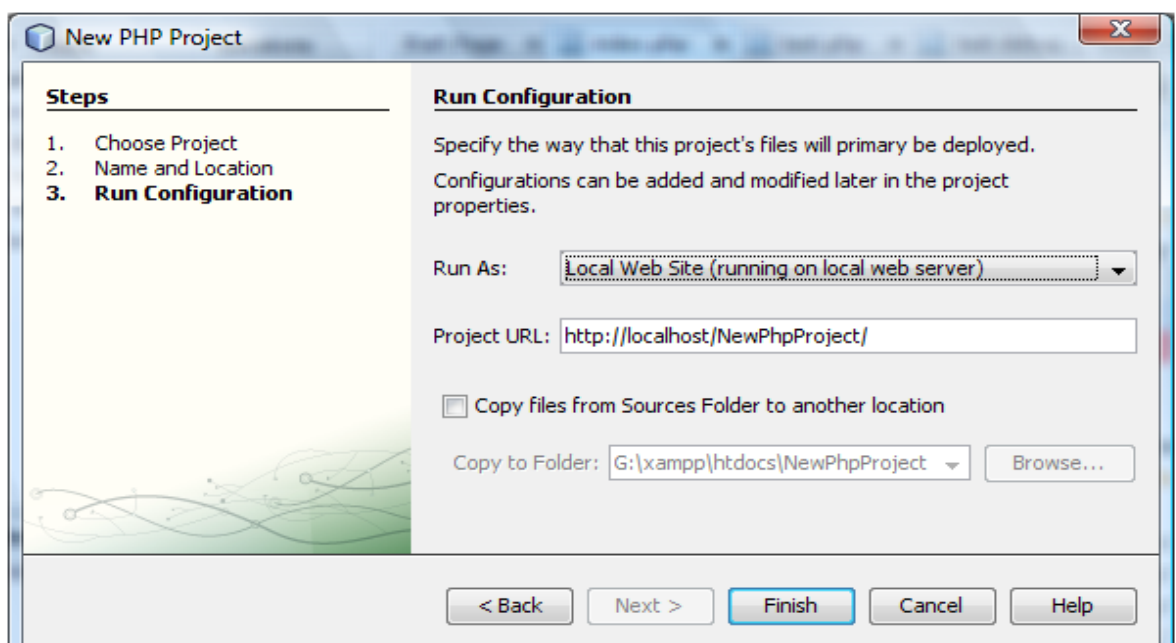


Fig 4.7: Setting project in NetBeans -2

4.1.3 PHPmyAdmin

phpMyAdmin is a tool written in PHP intended to handle the administration of MySQL servers over the Web. Currently it can create and drop databases, create/drop/alter tables and views, delete/edit/add fields, execute any SQL statement, manage keys on fields, manage privileges, manage triggers and stored procedures, export data into various formats and is available in 62 languages. Development is backed up by the phpMyAdmin team.

PhpMyAdmin is one of the most popular applications for MySQL databases management. It is a free tool written in PHP. Through this software you can create, alter, drop, delete, import and export MySQL database tables. You can run MySQL queries, optimize, repair and check tables, change collation and execute other database management commands.

All the SiteGround clients can manage their MySQL databases through the preinstalled PhpMyAdmin software which is integrated in Control Panel.

4.1.3.1 PhpMyAdmin: Administration

Here we will describe the functionality of the software, integrated in Control Panel.

Once you enter your PhpMyAdmin application, you will see different areas.

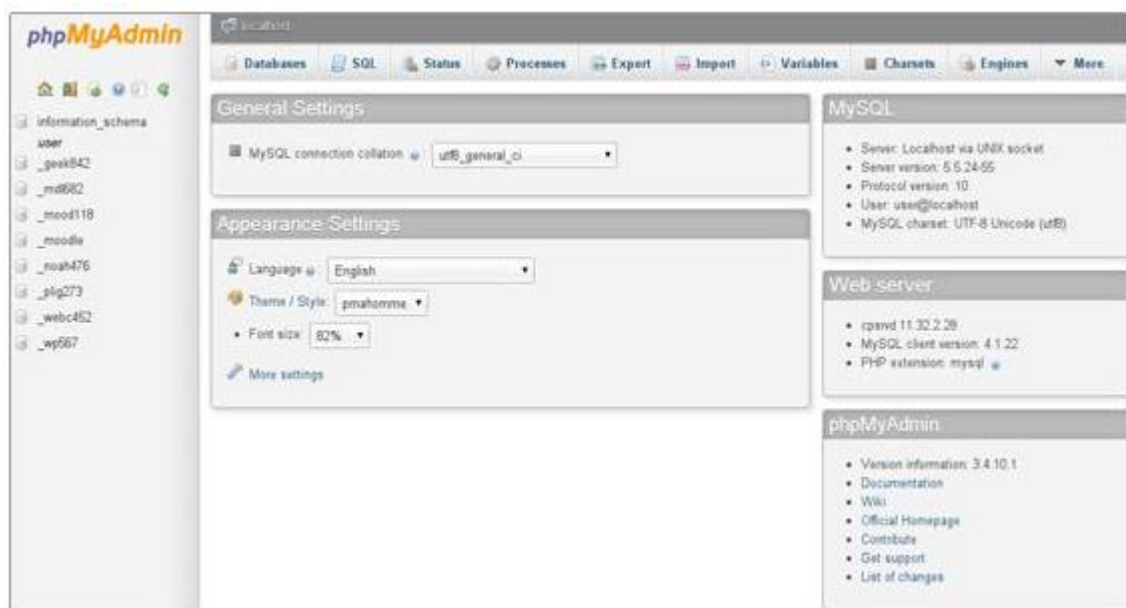


Fig 4.8: PhpMyAdmin : Administration

In the upper part you will find the server hostname. The databases which you will manage are stored on the same server as the software and the hostname is: localhost.

Under it there is information regarding the MySQL server, the MySQL client and the PhpMyAdmin version.

Next, you will see the MySQL charset and you will be able to define the MySQL connection collation.

In the right column you can change the default language, alter the style, customize the theme color and the font size. Also there you will notice links to PhpMyAdmin resources.

Please note that you can not create a database directly through cPanel->PhpMyAdmin. However, you can easily create a new database from your cPanel->MySQL Databases. Navigate to the Create New Database box. Enter the database name in the New Database text field and click on the Create Database button. The database name will start with the cPanel user name. For example if your cPanel user name is user and you want to have a database named test, the actual database name will be user_test. You can check our MySQL tutorial for more details.

In the left panel you will see the list of the databases for which your Control Panel user has privileges.

4.1.3.2 User Management

Within phpMyAdmin, user management is controlled via the Privileges link from the main page. Users can be created, edited, and removed.

4.1.3.2.1 Creating a new user

To create a new user, click the Add a new user link near the bottom of the Privileges page (you must be a "superuser", e.g., user "root"). Use the textboxes and drop-downs to configure the user to your particular needs. You can then select whether to create a database for that user and grant specific global privileges. Once you've created the user (by clicking Go), you can define that user's permissions on a specific database (don't grant global privileges in that case). In general, users do not need any global privileges (other than USAGE), only permissions for their specific database.

4.1.3.2.2 Editing an existing user

To edit an existing user, simply click the pencil icon () to the right of that user in the Privileges page. You can then edit their global- and database-specific privileges, change their password, or even copy those privileges to a new user.

4.1.3.2.3 Deleting a user

From the Privileges page, check the checkbox for the user you wish to remove, select whether or not to also remove any databases of the same name (if they exist), and click Go.

Assigning privileges to user for a specific database

Users are assigned to databases by editing the user record (from the Privileges link on the home page) not from within the Privileges link under the table. If you are creating a user specifically for a given table you will have to create the user first (with no

global privileges) and then go back and edit that user to add the table and privileges for the individual table.

4.1.3.3 Databases

In the Databases tab you will find a list with all the databases which can be managed through the cPanel user.



Fig 4.9: PhpMyAdmin Database

Once you click on a chosen database, you can start its management.

If you enable the statistics for the databases, you will see a table with the used collations, the number of the tables and the rows, the size of the data and the indexes, the total size and the overhead.

Please note that the enabling of the statistics will increase the load caused by your account on the server. Thus it is not recommended when you have many large databases.

4.2 Sample Code

Cuserspecforum.php

```
<head>
  <title>Corporate Specific Forum</title>
  <link rel="stylesheet" href="style.css" type="text/css">
  <link media="screen" href="header.css" type="text/css" rel="stylesheet">
</head>
<?php
include('config.php');
$connection=mysql_connect("localhost", "root", "")or die("cannot connect");
mysql_select_db("studentconnect",$connection)or die("cannot select DB");
$cn=$_SESSION['cusername'];
$sql="SELECT * FROM opportunities where cusername='$cn'";
include 'header.php';
?>
<div id="viewtopic">
<table width="90%" border="0" align="center" cellpadding="3" cellspacing="1"
bgcolor="#CCCCCC">
<tr>
<td width="6%" align="center" bgcolor="#E6E6E6"><strong>#</strong></td>
<td width="53%" align="center" bgcolor="#E6E6E6"><strong>Topic</strong></td>
<td width="13%" align="center"
bgcolor="#E6E6E6"><strong>Date/Time</strong></td>
</tr>

<?php
$result=mysql_query($sql);
if($result)
{
// Start looping table row
while($rows=mysql_fetch_array($result)){
?>
<tr>
<td bgcolor="#FFFFFF"><?php echo $rows['id']; ?></td>
<td bgcolor="#FFFFFF"><a href="cviewopportunity.php?id=<?php echo
$rows['id'];?>"><?php echo $rows['title'];?></a> </td>
<td align="center" bgcolor="#FFFFFF"><?php echo $rows['datetime']; ?></td>
</tr>

<?php
// Exit looping and close connection
}
}
else{ }
mysql_close();
?>
<tr>
```

```

<td width="50%" align="left" bgcolor="#E6E6E6"><input type="button"
name="btnSubmit" value="Post Opportunities"
onclick="window.location.href='postopportunities.php';" class="button" /></td>
<td width="50%" align="right" bgcolor="#E6E6E6"><input type="button"
name="btnSubmit" value="Main Forum"
onclick="window.location.href='mainstudentforum.php';" class="button" /></td>
</tr>
</table>
</div>

```

Mainstudentforum.php

```

<head>
<link rel="stylesheet" href="style.css" type="text/css">
<link media="screen" href="header.css" type="text/css" rel="stylesheet">
</head>
<?php
include('config.php');
// Connect to server and select database.
$connection=mysql_connect("localhost", "root", "")or die("cannot connect");
mysql_select_db("studentconnect",$connection)or die("cannot select DB");
$sql="SELECT * FROM forum_question ORDER BY id DESC";
// OREDER BY id DESC is order result by descending

$result=mysql_query($sql);
?>
<?php include('header.php'); ?>
<div id="viewtopic">
<table border="0" align="center" cellpadding="3" cellspacing="1"
bgcolor="#CCCCCC">
<tr>
<td width="10%" align="center" bgcolor="#E6E6E6"><strong>#</strong></td>
<td width="63%" align="center" bgcolor="#E6E6E6"><strong>Topic</strong></td>
<td width="13%" align="center"
bgcolor="#E6E6E6"><strong>Date/Time</strong></td>
</tr>

<?php
// Start looping table row
while($rows=mysql_fetch_array($result)){
?>
<tr>
<td bgcolor="#FFFFFF"><?php echo $rows['id']; ?></td>
<td bgcolor="#FFFFFF"><a href="viewtopic.php?id=<?php echo
$rows['id'];?>"><?php echo $rows['topic'];?></a> </td>
<td align="center" bgcolor="#FFFFFF"><?php echo $rows['datetime']; ?></td>
</tr>

```

```

<?php
// Exit looping and close connection
}
mysql_close();
?>
</table>
</div>

```

createtopic.php

```

<?php include('header.php'); ?>
<?php include('config.php'); ?>
<head>
    <title>Create Topic</title>
    <link rel="stylesheet" href="style.css" type="text/css">
    <link media="screen" href="header.css" type="text/css" rel="stylesheet">
</head>
<div id="content">
    <table width="400" border="0" align="center" cellpadding="0" cellspacing="1"
    bgcolor="#CCCCCC">
    <tr>
    <form id="form1" name="form1" method="post" action="topicinsert.php">
    <td>
    <table width="100%" border="0" cellpadding="3" cellspacing="1"
    bgcolor="#FFFFFF">
    <tr>
    <td colspan="3" bgcolor="#E6E6E6"><strong>Create New Topic</strong> </td>
    </tr>
    <tr>
    <td width="14%"><strong>Topic</strong></td>
    <td width="2%"></td>
    <td width="84%"><input name="topic" type="text" id="topic" size="50" /></td>
    </tr>
    <tr>
    <td valign="top"><strong>Detail</strong></td>
    <td valign="top"></td>
    <td rows="10"><textarea name="detail" cols="50" rows="10"
    id="detail"></textarea></td>
    </tr>
    <tr>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
    <td><input type="submit" name="Submit" value="Submit" /> <input type="reset"
    name="Submit2" value="Reset" /></td>
    </tr>
    </table>
    </td>
</form>

```

```
</tr>
</table>
</div>
```

Cviewopportunity.php

```
<link rel="stylesheet" href="style.css" type="text/css">
<link media="screen" href="header.css" type="text/css" rel="stylesheet">
<?php
include('config.php');
$conn=mysql_connect("localhost","root","");
mysql_select_db("studentconnect",$conn);
$urname=$_SESSION['cusername'];
$id=$_GET['id'];
$_SESSION['oid']=$id;
$result=mysql_query("select * from oppurtunities where cusername='$urname'");
$rows=mysql_fetch_array($result);
include 'header.php';
?>
<div id="viewtopic">
<table class="author" align="center" border="0" cellpadding="2" cellspacing="0">
  <tr>
    <th>Title</th>
    <td><?php echo $rows['title']; ?></td>
  </tr>
  <tr>
    <th>Description</th>
    <td><?php echo $rows['description']; ?></td>
  </tr>
  <tr>
    <th>By</th>
    <td><?php echo $rows['cusername']; ?></td>
  </tr>
  <tr>
    <th>Date&Time</th>
    <td><?php echo $rows['datetime']; ?></td>
  </tr>
</table>
</div>
<?php
mysql_close();
?>
<a href="cuserspecforum.php">Home</a>
```

Listopportunities.php

```
<head>
  <title>List Opportunities</title>
  <link rel="stylesheet" href="style.css" type="text/css">
<link media="screen" href="header.css" type="text/css" rel="stylesheet">
</head>
<?php
include('config.php');
// Connect to server and select database.
$connection=mysql_connect("localhost", "root", "")or die("cannot connect");
mysql_select_db("studentconnect",$connection)or die("cannot select DB");
//$sn=$_SESSION['susername'];
//echo "$sn";
$sql="SELECT * FROM oppurtunities";
// OREDER BY id DESC is order result by descending
include('header.php');
?>
<div id="viewtopic">
<table width="90%" border="0" align="center" cellpadding="3" cellspacing="1"
bgcolor="#CCCCCC">
<tr>
<td width="6%" align="center" bgcolor="#E6E6E6"><strong>#</strong></td>
<td width="53%" align="center" bgcolor="#E6E6E6"><strong>Title</strong></td>
<td width="13%" align="center"
bgcolor="#E6E6E6"><strong>Date/Time</strong></td>
</tr>

<?php
$result=mysql_query($sql);
if($result)
{
while($rows=mysql_fetch_array($result)){
?>
<tr>
<td bgcolor="#FFFFFF"><?php echo $rows['id']; ?></td>
<td bgcolor="#FFFFFF"><a href="viewopportunities.php?id=<?php echo
$rows['id'];?>"><?php echo $rows['title'];?></a> </td>
<td align="center" bgcolor="#FFFFFF"><?php echo $rows['datetime']; ?></td>
</tr>

<?php
// Exit looping and close connection
}
}
else{ }
mysql_close();
```



```

</td>
</form>
</tr>
</table>
</div>

```

Viewopportunities.php

```

<head>
  <title>View Opportunities</title>
  <link rel="stylesheet" href="style.css" type="text/css">
  <link media="screen" href="header.css" type="text/css" rel="stylesheet">
</head>
<?php
include('config.php');
$conn=mysql_connect("localhost","root","");
mysql_select_db("studentconnect",$conn);
//$urname=$_SESSION['susername'];
$id=$_GET['id'];
$_SESSION['oid']=$id;
$result=mysql_query("select * from oppurtunities where id='$id'");
$rows=mysql_fetch_array($result);
include 'header.php';
?>
<div id="viewtopic">
<table width="900" border="0" align="center" cellpadding="0" cellspacing="1"
bgcolor="#CCCCCC">
<tr>
<td><table width="100%" border="0" cellpadding="3" cellspacing="1"
bordercolor="1" bgcolor="#FFFFFF">
<tr>
<td bgcolor="#F8F7F1"><strong><?php echo $rows['title']; ?></strong></td>

<tr>
<td bgcolor="#F8F7F1"><?php echo $rows['description']; ?></td>

<tr>
<td bgcolor="#F8F7F1"><strong>By :</strong> <?php echo $rows['cusername']; ?>
</td>

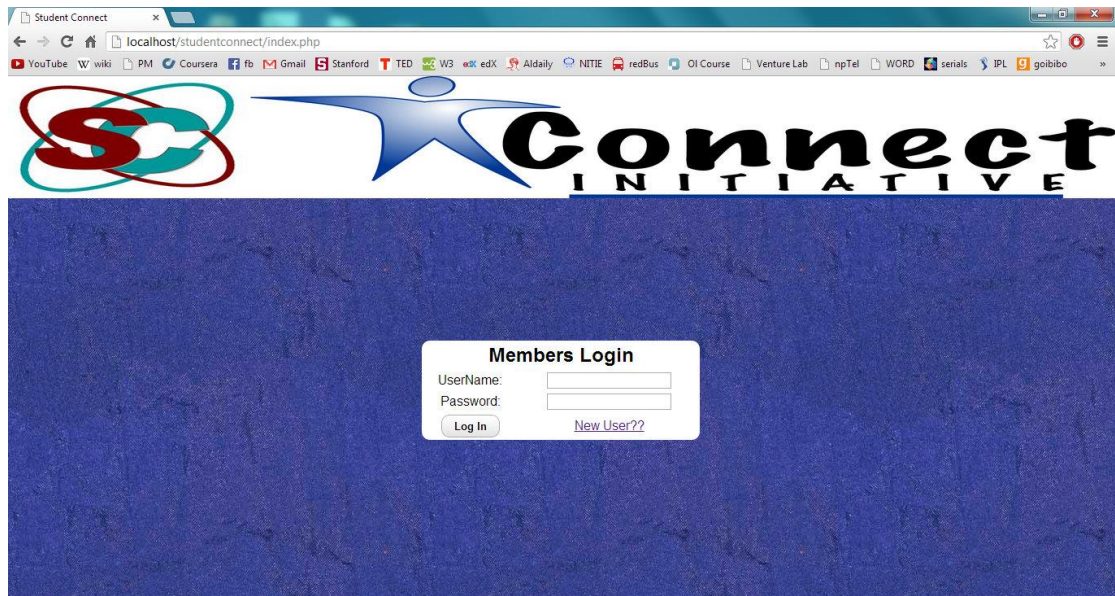
<tr>
<td bgcolor="#F8F7F1"><strong>Date/time :</strong><?php echo $rows['datetime'];
?></td>
</tr>
</table></td>
</tr>

```

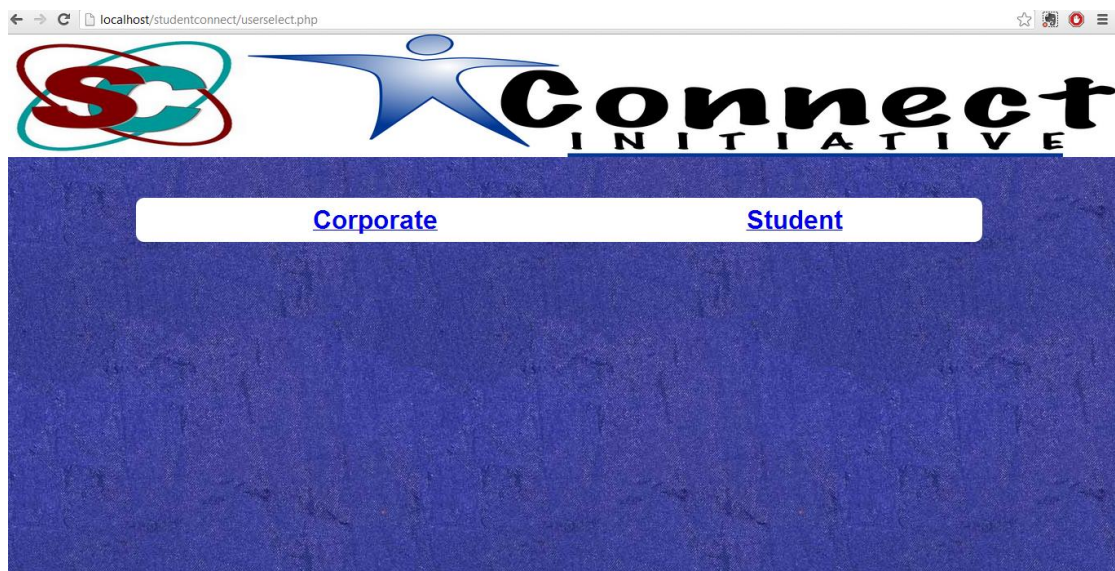
```
</table>
<?php
// Exit looping and close connection
mysql_close();
/*if (!isset($_SESSION['prs'])) {
echo "pears were not set";
session_register ("pears");
$pears = dmondindex("pear");
}*/
?>
</div>
```


4.3 Output Screens

4.3.1 Common pages for all the actors (student, corporate, admin)




Home page



Corporate, student registration-usersselect page

4.3.1.1 Registration page (student, corporate)

localhost/studentconnect/sregister.php



Registration Form


Full Name
College Name
Roll No
Email id
Username
Password

Register

sfullname	scollegename	srollno	semailid	susername	spassword	sverify
Ravinder Putta	MVSR	245109733038	ravi060791ster@gmail	ravi060791	135621	1
o paramesh	mvsr	245109733014	paramesh014@gmail.co	paramesh014@gmail.co	paramesh	1
Lohith Kumar Menchu	MVSR	11	lovemenchu@gmail.com	lkmenchu	135621	1

Student registration form

localhost/studentconnect/register.php



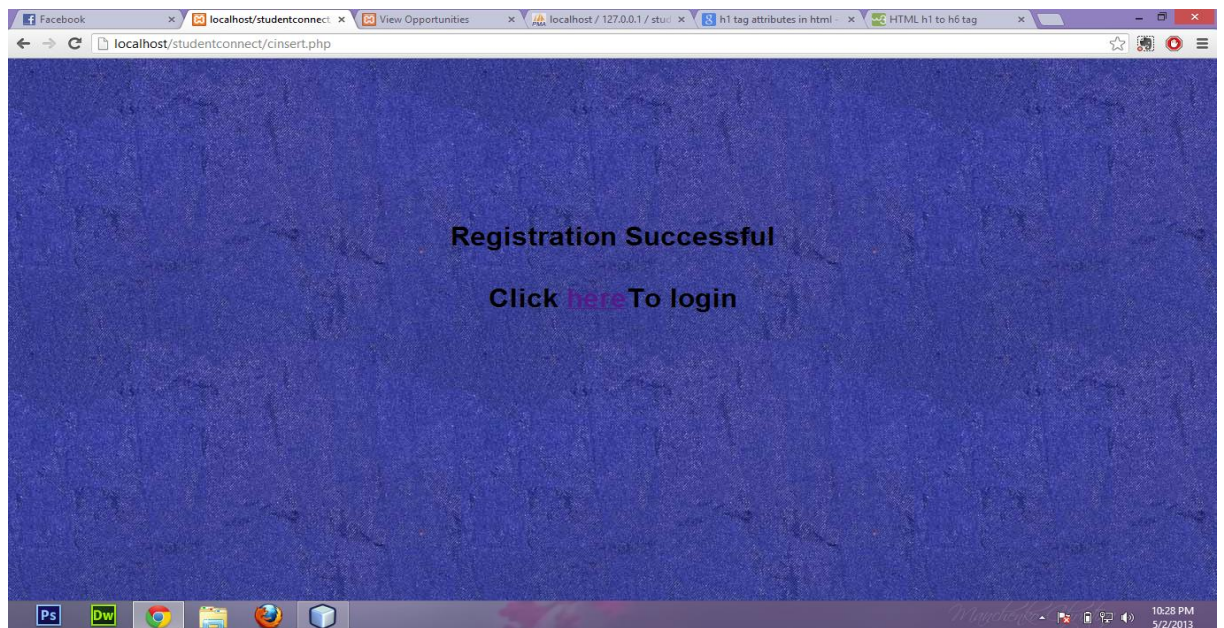
Corporate Registration Form

Full Name
Company Name
Designation
Emailid
Username
Password

Register

cfullname	ccompanyname	cdesignation	cemail	cusername	cpassword	cverify
Purnima	MVSR Engineering College	HR	purnima.janaki@gmail.com	purnima.janaki@gmail.com	puri	1
sandeep	MVSR Engineering College	HR	sandeeppragila@gmail.com	sandeep	sandeep	1
Pavan	MVSR Engineering College	System Anlayst	pavan@gmail.com	pvn	98480	1
Lohith Kumar Menchu	MVSR	HR	lkmenchu@gmail.com	Clkmenchu	135621	1

Corporate Registration form



Registration successful

4.3.2 Student activities

id	topic	detail	username	datetime
11	ajax	Acronym for Asynchronous JavaScript and XML.[1] is...	ravi060791	13/03/13 09:45:55
12	bigdata	Big data[1][2] is a collection of data sets so lar...	ravi060791	15/03/13 08:16:28
13	lol	header("Location./studentconnect/suserspecforu.p...	ravi060791	15/03/13 08:24:19
16	Report	ABSTRACT: In this presentation we present Google. ...	ravi060791	23/03/13 01:56:26
17	University Campus Social Network System for Knowle...	Introduction: Online social network services...	ravi060791	23/03/13 04:14:17

Creating a new topic by the registered student

#	Topic	Date/Time
22	Software as Service	01/05/13 05:07:46
21	DBMS	05/04/13 08:14:45
17	University Campus Social Network System for Knowledge Sharing	23/03/13 04:14:17
16	Report	23/03/13 01:56:26
12	bigdata	15/03/13 08:16:28
11	ajax	13/03/13 09:45:55

Student main forum-view of own topics of a student

localhost/studentconnect/viewtopic.php?id=21



Logout

Title	DBMS
Description	A database is an organized collection of data. The data is typically organized to model relevant aspects of reality (for example, the availability of rooms in hotels), in a way that supports processes requiring this information (for example, finding a hotel with vacancies). A general-purpose database management system (DBMS) is a software system designed to allow the definition, creation, querying, update, and administration of databases. Well-known DBMSs include MySQL, PostgreSQL, SQLite, Microsoft SQL Server, Microsoft Access, Oracle, Sybase, dBASE, FoxPro, and IBM DB2. A database is not generally portable across different DBMS, but different DBMSs can inter-operate by using standards such as SQL and ODBC or JDBC to allow a single application to work with more than one database.
By	ravi060791
Date&Time	05/04/13 06:14:45


question_id	id	username	answer
1	2	ravi060791	i do not know
1	3	ravi060791	i learned few things since my reply
1	4	ravi060791	i learned few things since my reply
1	5	ravi060791	hello

Comment

Submit Reset

View topic page

localhost/studentconnect/suserspecforum.php?id=ravi060791



Logout

#	Topic	Date/Time
11	ajax	13/03/13 09:45:55
12	bigdata	15/03/13 08:16:28
13	lol	15/03/13 08:24:19
16	Report	23/03/13 01:56:26
17	University Campus Social Network System for Knowledge Sharing	23/03/13 04:14:17
21	DBMS	05/04/13 06:14:45

Create Topic Main Forum View Opportunities

Student page-suserspec forum page

4.3.3 Corporate activities



The screenshot displays a web application interface for posting opportunities. At the top, there is a navigation bar with a logo, the text "Connect INITIATIVE", and a "Logout" button. The main content area is titled "Create New Opportunities". It contains a form with two input fields: "Title" and "Description". Below the form are "Submit" and "Reset" buttons. A table below the form lists existing opportunities. Arrows point from the "Title" and "Description" fields to the corresponding columns in the table.

id	title	description	cusername	datetime
1	assistant engineer	assistant engineer capgemini salary:1000000000000...	pumima.janaki@gmail.com	2015-03-13 08:34:1
2	sandeep	sandeepsandeepsandeepsandeepsandeepsandeeps...	sandeep	2015-03-13 09:35:3
3	sandeep	sandeepsandeepsandeepsandeepsandeepsandeeps...	sandeep	2015-03-13 10:51:0

Post opportunities page



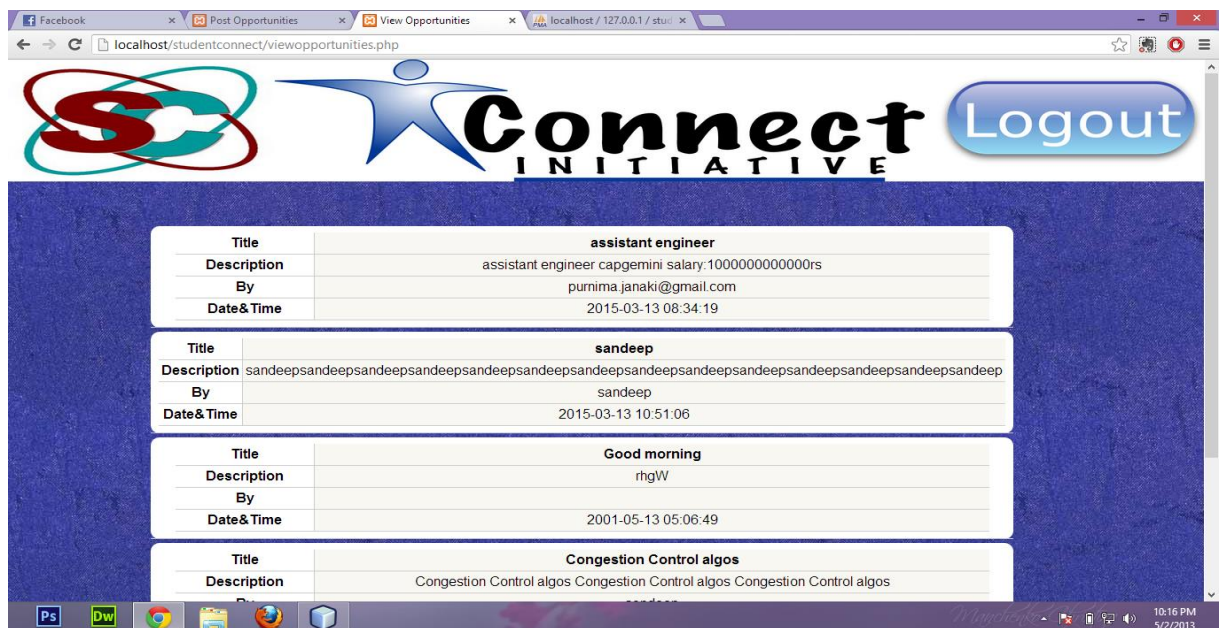
The screenshot shows a web browser window with the URL `localhost/studentconnect/viewtopic.php?id=21`. The page features the 'Connect INITIATIVE' logo and a 'Logout' button. The main content area displays a topic titled 'DBMS' with a detailed description of database management systems. Below the topic information, there is a section for replies, showing one reply from user 'ravi060791' on 05/04/13. At the bottom, there is a 'Comment' form with a text input field and 'Submit' and 'Reset' buttons.

Title	DBMS
Description	A database is an organized collection of data. The data is typically organized to model relevant aspects of reality (for example, the availability of rooms in hotels), in a way that supports processes requiring this information (for example, finding a hotel with vacancies). A general-purpose database management system (DBMS) is a software system designed to allow the definition, creation, querying, update, and administration of databases. Well-known DBMSs include MySQL, PostgreSQL, SQLite, Microsoft SQL Server, Microsoft Access, Oracle, Sybase, dBASE, FoxPro, and IBM DB2. A database is not generally portable across different DBMS, but different DBMSs can inter-operate by using standards such as SQL and ODBC or JDBC to allow a single application to work with more than one database.
By	ravi060791
Date & Time	05/04/13 06:14:45

ravi060791	Reply One	30/04/13 08:00:10
------------	-----------	-------------------

Comment

View of topics-by student and corporate.



The screenshot shows a web browser window with the URL `localhost/studentconnect/viewopportunities.php`. The page features the 'Connect INITIATIVE' logo and a 'Logout' button. The main content area displays a list of opportunities, each with a table showing details like Title, Description, By, and Date & Time.

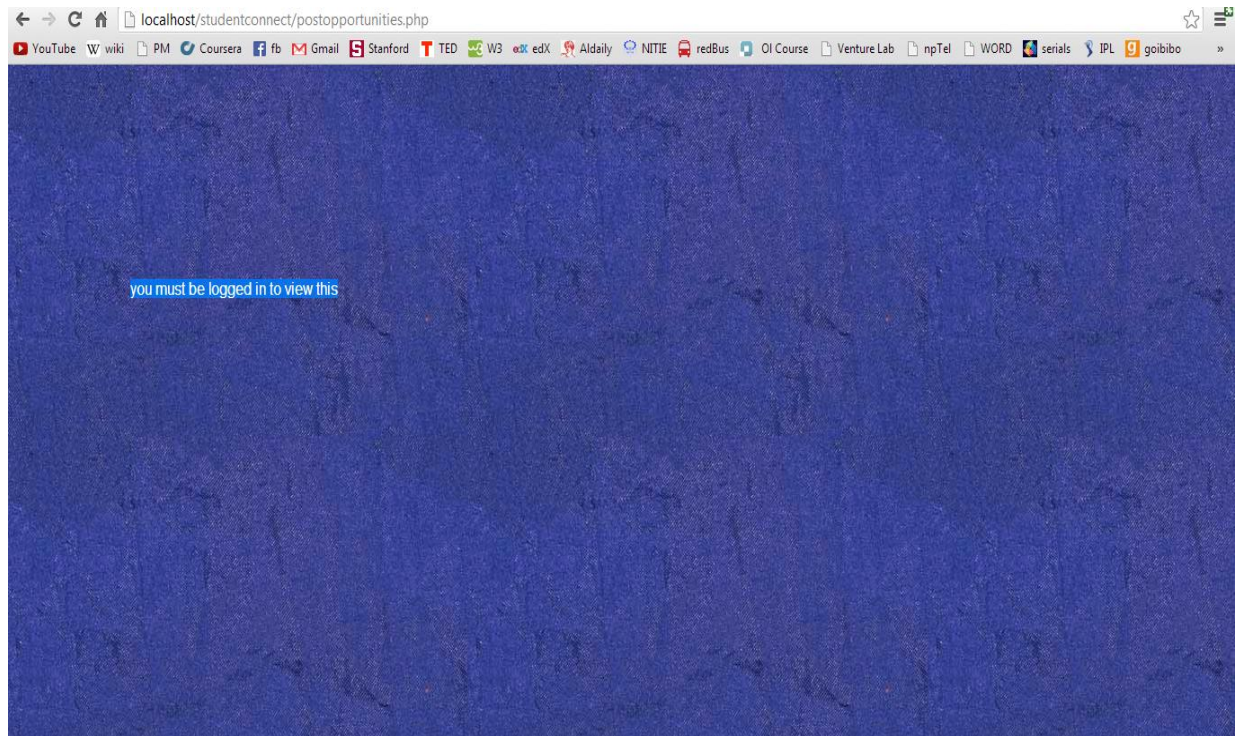
Title	assistant engineer
Description	assistant engineer capgemini salary:1000000000000rs
By	purnima.janaki@gmail.com
Date & Time	2015-03-13 08:34:19

Title	sandeep
Description	sandeepsandeepsandeepsandeepsandeepsandeepsandeepsandeepsandeepsandeepsandeepsandeepsandeepsandeepsandeepsandeeps
By	sandeep
Date & Time	2015-03-13 10:51:06

Title	Good morning
Description	rhgW
By	
Date & Time	2001-05-13 05:06:49

Title	Congestion Control algos
Description	Congestion Control algos Congestion Control algos Congestion Control algos

View opportunities



User must be logged in to view the page

CHAPTER 5

DEPLOYMENT

5.1 Requirements for Deployment

We require the following to deploy the system.

- Apache Server
- MySQL database
- PHP Engine

We used WAMP Server to deploy the System on Windows. The following illustrates the description about the deployment in WAMP Server

5.2 WAMP Directory Structure

WAMP Directory contains the following subdirectories. The description of each directory is as follows:

- “**bin**” directory contains binaries for Apache, MySQL, and PHP, in their respectively sub-directory.
- “**apps**” directory contains server-side tools such as PhpMyAdmin, SQL Buddy, and Web Grind.
- “**tools**” contains client-side tool such as xdc (XDebug Client).
- “**www**” is the apache server’s root directory.
- “**logs**” contains apache and MySQL log files.
- “**alias**” contains the apache’s alias configuration for PhpMyadmin, SQL Buddy, and Web Grind.

To deploy the web application in WAMP, we should first create a directory in “www” folder under wamp directory.

The URL <http://localhost/MyWebApp> will redirect to the home page of the application. By default Apache takes the files with the name “**index.*** like **index.htm**, **index.html**, **index.php**” as the Home Page.

5.3 Configuration Files

The following describe the configuration files present under WAMP

- **http.conf** is a configuration file which is used by the Apache HTTP Server. It stores information on various functions of the server, which can be edited by removing or adding a number sign “#” at the beginning of the line, thus setting values for each directive. The Apache’s configuration file “**http.conf**” is

located at

`"c:\wamp\bin\apache\Apache2.2.17\conf"`

- PHP.ini is a configuration file that is used to customize the behavior of PHP at runtime. This enables easy administration in the way one administers's Apache web server using configuration files. The Settings in which upload directory, register global variables, display errors, log errors, max uploading size setting, maximum time to execute a script and other configurations is written in this file. The PHP's configuration file "**php.ini**" is located at `"c:\wamp\bin\apache\Apache2.2.17\bin"`
- my.ini is a configuration file that connects Apache and MySQL. The MySQL's configuration file "**my.ini**" is located at `"c:\wamp\bin\mysql\mysql5.5.8"`

The Database tables needed for the Application will be created using phpMyAdmin in WAMP. **phpMyAdmin** is a free and open source tool written in PHP intended to handle the administration of MySQL with the use of a web browser. It can perform various tasks such as creating, modifying or deleting databases, tables, fields or rows; executing SQL statements; or managing users and permissions.

CHAPTER 6

CONCLUSIONS

6.1 Conclusions

The Application is constructed in a manner that requires the least installation of software in the clients. This solves most of the problems encountered when the notification management is performed manually.

6.1.1 Limitations

The application has the following limitations:

- Each branch is assumed to have five subjects which limit the user to choose a maximum of five subjects during registration.
- Each subject is implemented using its name rather than by its subject code.

6.1.2 Future Enhancements

The System can be enhanced with the inclusion of following features thereby increasing the scope of the system.

- Inter College Communication can be made by using an additional column in the notification table that has the college name specifying to which college the notification has been sent for.
- An additional University table can be included that has the columns with university name and college id's of the colleges under the university thereby making the inclusion of many universities to use the system.

REFERENCES

Bibliography

- [1] Programming PHP by Rasmus Lerdorf, Kevin Tatroe, Peter MacIntyre, 2nd edition.
- [2] Beginning MySQL by Robert Sheldon, Geoff Moes.
- [3] The Complete Reference Ajax by Thomas A. Powell, 2008.
- [4] The Unified Modeling Language User Guide by Grady Booch, James Rumbaugh, Ivar Jacobson, 2006.
- [5] Apache: The Definitive Guide by Ben Laurie, Peter Laurie, 3rd edition.

Web Links

- [1] <http://projects.apache.org/docs/index.html>
- [2] <http://www.w3schools.com/php/default.asp>
- [3] <http://php.net/manual/en/index.php>
- [4] <http://www.wampserver.com/en/>
- [5] <http://www.easyphp.org/introduction.php>
- [6] <http://www.htmlgoodies.com/primers/html/>
- [7] <http://en.wikipedia.org/>