

Chapter 1 – Introduction

1.1 Introduction

- MyLDRP is an contains *Automated Attendance system with SMS & Email Notification* which calculates Attendances and generate Notifications via different methods, also this system provides functionalities like Report Generation, SMS Notification.
- Using this system User can view Attendance, Timetable, Marks.
- The website is very useful to Students & Faculties of LDRP.
- The interface of this website is straight forward and takes you roughly a minute to get started.

1.2 Scope

- The web-application is titled Automated Attendance system with SMS & Email Notification.
- This web-application is intended for LDRP-ITR.
- The objective of this website is to save time of Faculty and Make easy access of details to students.
- Provides easy way to Faculty to Generate PDF, Excel or Direct Print Attendance Reports.

1.3 Project summary and Purpose

- The purpose of this web-application is to provide online information about Academic Details like Attendance, Time-Table, Marks.
- Faculties don't need to count each class's each subject Attendance, this system can automatically count and stores it.
- This web-application also provides Function of SMS & Email Notification.
- SMS & Email Notification also works automatically and bases on attendance and Threshold Percentage Students Parents are notified by SMS & Email (Automatically)

1.4 Overview of the project

- The web-application of Automated Attendance System is called MyLDRP.
- This system will mine data from Google Sheets and stores it inside Database.
- System will also generate Weekly reports automatically.
- Alongside weekly reports system will send Email & SMS notifications to Students Parents by it self.
- Front End web-portal for Faculty provided to generate reports manually in different file format.

1.5 Problem Definition

- Normally, Attendance are stored in books and it is time consuming to count attendance for each and every class.
- This system will count Attendance for each subject of each class and generate combined of subject's attendance weekly reports.
- The weekly reports will help Faculty to analyze Students attendance more easily and quickly.
- This system will mine data from Google Sheets and stores it inside Database.
- Faculty can view any student's attendance by inserting student's Enrollment number in Quick Search.
- Additionally, Faculty can view or generate pdf for whole class's Attendance for all subjects combined.
- More over, Time tables can be also view from this portal and PDF can also Downloaded for easy distribution among students.
- Before it was tough to notify parents about Students attendance, now with this system it is makes easy way to notify parents via SMS & Email.
- Student's Mid-semester Marks or Final Grad can be seen also from faculty-portal of this system.

Chapter 2 - Technology and Literature Review

2.1 About Tools and Technology

We have used PHP,HTML5,CSS3,Angular JS, JQuery for this project and we also used AJAX technology. This system is Structured in MVC Format.

AJAX

AJAX = Asynchronous JavaScript and XML.

AJAX is a technique for creating fast and dynamic web pages. AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page. Classic web pages, (which do not use AJAX) must reload the entire page if the content should change

AJAX is about updating parts of a web page, without reloading the whole page.

How AJAX work?

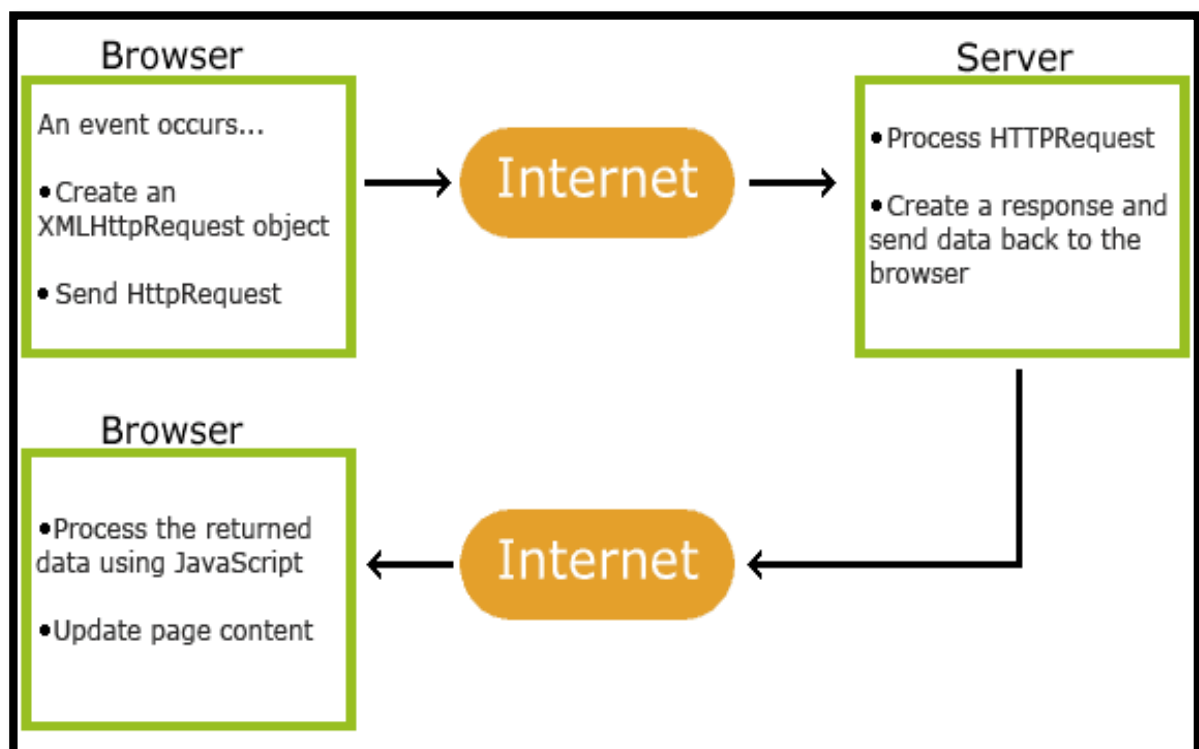


Fig 2(a): AJAX

MVC

MVC = Model View Controller.

MVC, or Model-View-Controller is a software architecture, or design pattern, that is used in software engineering, whose fundamental principle is based on the idea that the logic of an application should be separated from its presentation. Put simply, one would say that MVC is simply a better way of separating the logic of your application from the display.

How MVC work?

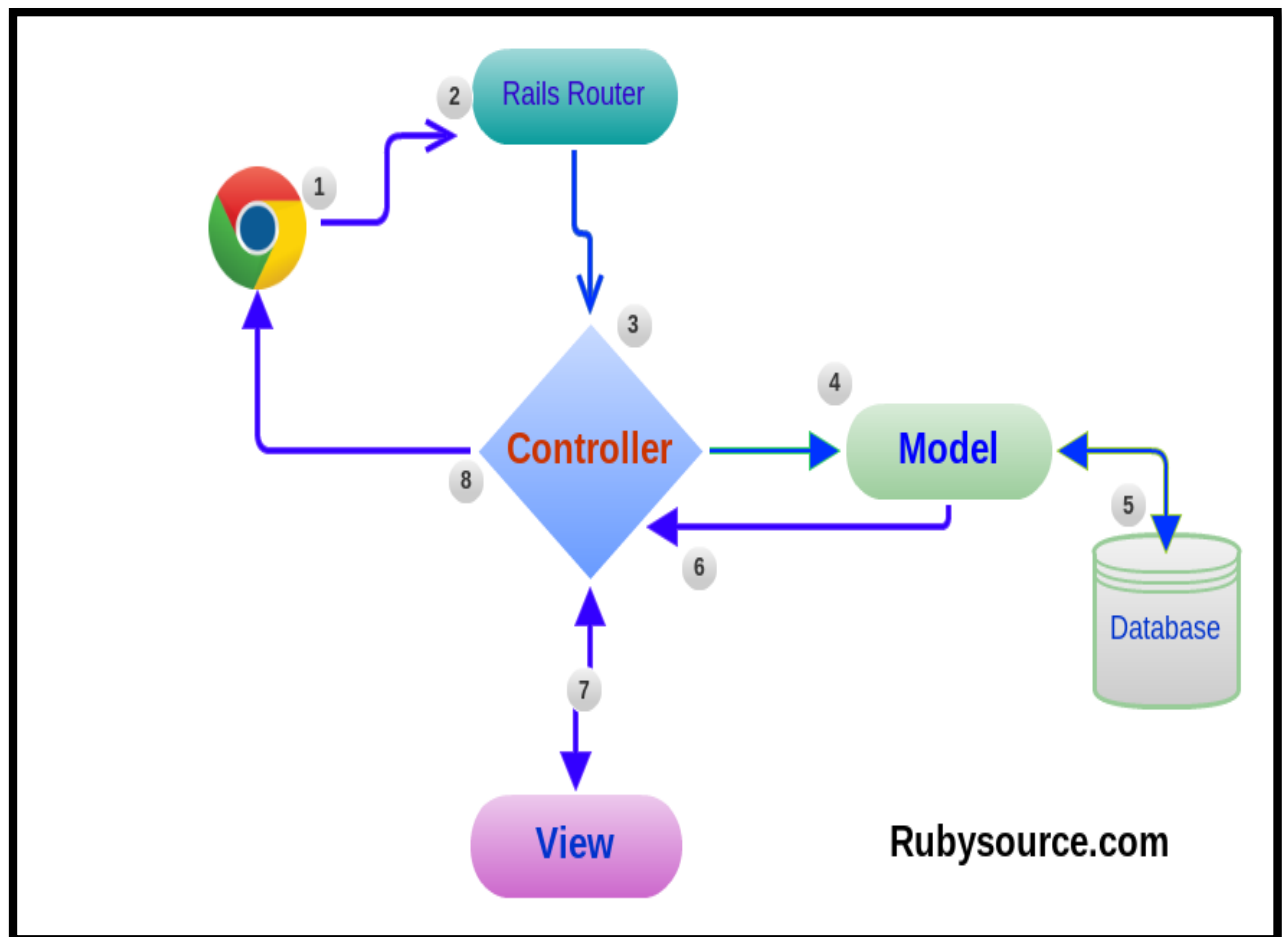


Fig 2(b): MVC

PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the **PHP** reference implementation is now produced by The **PHP** Group.

The best things in using PHP are that it is extremely simple for a newcomer, but offers many advanced features for a professional programmer. Don't be afraid reading the long list of PHP's features. You can jump in, in a short time, and start writing simple scripts in a few hours.

Angular JS

Angular JS is a structural framework for dynamic web apps. It lets you use HTML as your template language and lets you extend HTML's syntax to express your application's components clearly and succinctly. Angular's data binding and dependency injection eliminate much of the code you would otherwise have to write.

Angular is what HTML would have been, had it been designed for applications. HTML is a great declarative language for static documents. It does not contain much in the way of creating applications, and as a result building web applications is an exercise in what do I have to do to trick the browser into doing what I want?

Angular takes another approach. It attempts to minimize the impedance mismatch between document centric HTML and what an application needs by creating new HTML constructs. Angular teaches the browser new syntax through a construct we call directives. Examples include:

- ❖ Data binding, as in `{{}}`.
- ❖ DOM control structures for repeating, showing and hiding DOM fragments.
- ❖ Support for forms and form validation.
- ❖ Attaching new behavior to DOM elements, such as DOM event handling.
- ❖ Grouping of HTML into reusable components.

jQuery

jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers.

Java Scripts

JavaScript is an interpreted programming or script language from Netscape. It is somewhat similar in capability to Microsoft's Visual Basic, Sun's Tcl, the UNIX-derived Perl, and IBM's REXX. In general, script languages are easier and faster to code in than the more structured and compiled languages such as C and C++. Script languages generally take longer to process than compiled languages, but are very useful for shorter programs.

2.2 Brief History of Work Done

❖ Project Plan

No.	Activity	Days	Approx. Weeks
1	Defining Project	7	1
2	Understanding Problem Definition and System Requirements	13	2
3	System Design	30	4
4	Diagrams	12	2
5	Data Dictionary	15	2
6	Form Design	13	2
7	Project Coding	65	8
8	Testing & Debugging	10	2

Chapter 3 - System Requirements Study

3.1 User Characteristics

- **Admin:** It can perform various operations.
 - ✓ Insert Link of Google Sheets for each Subjects
 - ✓ Create Faculty User
 - ✓ Send SMS Notification.
 - ✓ Manage Time-Table
 - ✓ Maintain System
- **Faculty:** It can perform various operations.
 - ✓ Login
 - ✓ Student Attendance Search Using Quick Search.
 - ✓ Class wise attendance between two specific dates.
 - ✓ Generate PDF or Excel reports.
 - ✓ Send Message Directly to Parents using enrolment number.
- **Student:** It can perform various operations. (Future Development)
 - ✓ View Mid Semester Marks.
 - ✓ Can view Final Examination Grads.
 - ✓ Can see time-table
 - ✓ Search for their attendance for specific dates.

3.2 Hardware and Software Requirements

This product is developed mainly using open source technologies like PHP, JQuery, Angular JS, MVC, MySQL etc.

Software Requirements

- **Server Side: -**
 - Database :- MySQL
 - Framework :- None (Larval For future purposes)
- **Client Side: -**
 - Operating System :- Any (Windows, IOS, Linux)
 - Browser :- Any compatible browser

Hardware Requirements

- **Server Side: -**
 - Processor :- Intel core i5, 3.1GHz
 - RAM :- 2 GB or More
 - Storage :- 10-15GB (with Rapid Backup)
- **Client Side: -**
 - Processor :- Core @ duo processor 1.8 GHz
 - RAM :- 512 MB or More
 - HDD :- 100-150 MB (to store cash Files of App)

3.3 Assumptions and Dependencies

- **Assumptions**
 - ✓ The code should be free with compilation errors/syntax errors.
 - ✓ The system must have an interface which is simple enough to understand.
- **Dependencies**
 - ✓ The proposed system would be designed, developed and implemented based on the software requirements specifications document.
 - ✓ High speed internet is required.
 - ✓ Google API & Google Sheets Access required.

Chapter 4 - System Analysis

4.1 Study of Current System

- ❖ When the process of working on this website started, a lot of research was done about how to mine data from Google Sheets or how to make automate work with different sheets formats .
- ❖ It is very complex to understand the whole process of system.
- ❖ Also, some statistical studies and reports were studied before and during the development of this website, to understand the PHP Market trends, how other websites of this type are behaving in the real market.

4.2 Requirements of New System

4.2.1 Developer Requirements

In order to make this System functional, Developer require the following:

- ❖ Access to Corn Job manager to setup Corn jobs of System.
- ❖ Links of Google Sheets Containing Attendance.
- ❖ SMS API to Provide SMS Functionality.
- ❖ SMTP Server to generate mails.

4.2.2 System Requirements

- ❖ PHP 5.6+ is used.
- ❖ PHPStorm, Con job Manager is to be installed.
- ❖ MySQL should be used for database.

4.2.3 User Requirements

- ❖ An Internet connection
- ❖ Any one of Most Updated Web Browser

4.3 Feasibility Study

Depending on the results of the initial investigation, the survey is expanded to a more detailed feasibility study. Feasibility study is a test of system proposal.

According to its work ability, impact on the organization, ability to meet user needs, and effective use of resources. The objective for this phase is not to solve the problem but to acquire a sense of scope. During the study, the problem definition is crystallized and aspects of the problem to be included in the system are determined.

We started the project by listing the possible queries that the user might want to be satisfied. And on these lines we guided the project further.

The three main points, kept in mind at the time of project, are:

- Possible (To build it with the given technology and resources)
- Affordable (given the time and cost constraints of the organization)
- Acceptable (for use by the eventual users of the system)

The three major areas to be considered while determining the feasibility of a project are:

1. Technical Feasibility

The technical issue usually raised during the feasibility stage of the investigation includes the following:

- Does the necessary technology exist to do what is suggested?
- Do the proposed equipment's have the technical capacity to hold the data required to use the new system?
- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of Secure Infrastructure Implementation System. The current system developed is technically feasible. It is a web based user interface. Thus it provides an easy access to the users. The databases purpose is to create, establish and maintain a work- flow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The software and hardware requirements for the development of this project are not many and are already available as free as open source. The work for the project is done with the current equipment and existing software technology. Necessary bandwidth exists for providing a fast feed- back to the users irrespective of the number of users using the system.

2. Operational Feasibility

Under this category of service, we conduct a study to analysis and determine whether your need can be fulfilled by using a proposed solution. The result of our operational feasibility Study will clearly outline that the solution proposed for your business is operationally workable and conveniently solves your problems under consideration after the proposal is implemented.

I would precisely describe how the system will interact with the systems and persons around. Our feasibility report would provide results of interest to all stakeholders. We will do as per the needs of the business requirements.

3. Timeline Feasibility

It is important to understand that a need must be fulfilled when it has to be. Some otherwise feasible and highly desirable projects can become non-feasible due to very restrictive timeline constraints. This fact makes it imperative that milestones are clearly linked to the timeline and projects are well conceived with safe unforeseen margins. We made sure that we strictly follow what has been stated above.

4.4 Characteristics of the Proposed System

- **Easiness in modification of data:**

The proposed system provides managing of huge data effectively and efficiently for efficient results, storing the details of the Students, Faculties etc. in such a way that the database can be modified.

- **User friendly:**

The proposed system is user friendly because the retrieval and storing of data is fast and data is maintained efficiently. Moreover, the graphical user interface is provided in the proposed system, which provides user to deal with the system very easily.

- **Reports are easily generated:**

Reports can be easily generated in a proposed system. So any type of reports can be generated in a proposed system, which helps the managers in a decisions- making activity.

- **Computer operator control:**

Computer operator control will be there no errors. Moreover, storing and retrieving of information is easy. So work can be done speedily and in time.

4.5 Features of New System

❖ Faculty Login Page

- ✓ This page displays the login with validation. If the user name is invalid then it will display the validation with the use of Ajax control tool kit.

❖ Dashboard

- ✓ After successful login Faculty can log in to their dashboard section.
- ✓ Faculty can use quick search to find any specific user's Attendance.
- ✓ PDF or Excel Reports can be Generated.

❖ View Class Attendance between specific dates

- ✓ Faculty can view Attendance of Class.
- ✓ PDF or Excel Reports can be Generated.

❖ View Time Table

- ✓ Faculty can view Time-table.

❖ View Marks Page

- ✓ Faculty can view Marks of Students.

❖ Send Notification Page

- ✓ SMS & Email Notification can be send.

4.6 Class Diagram

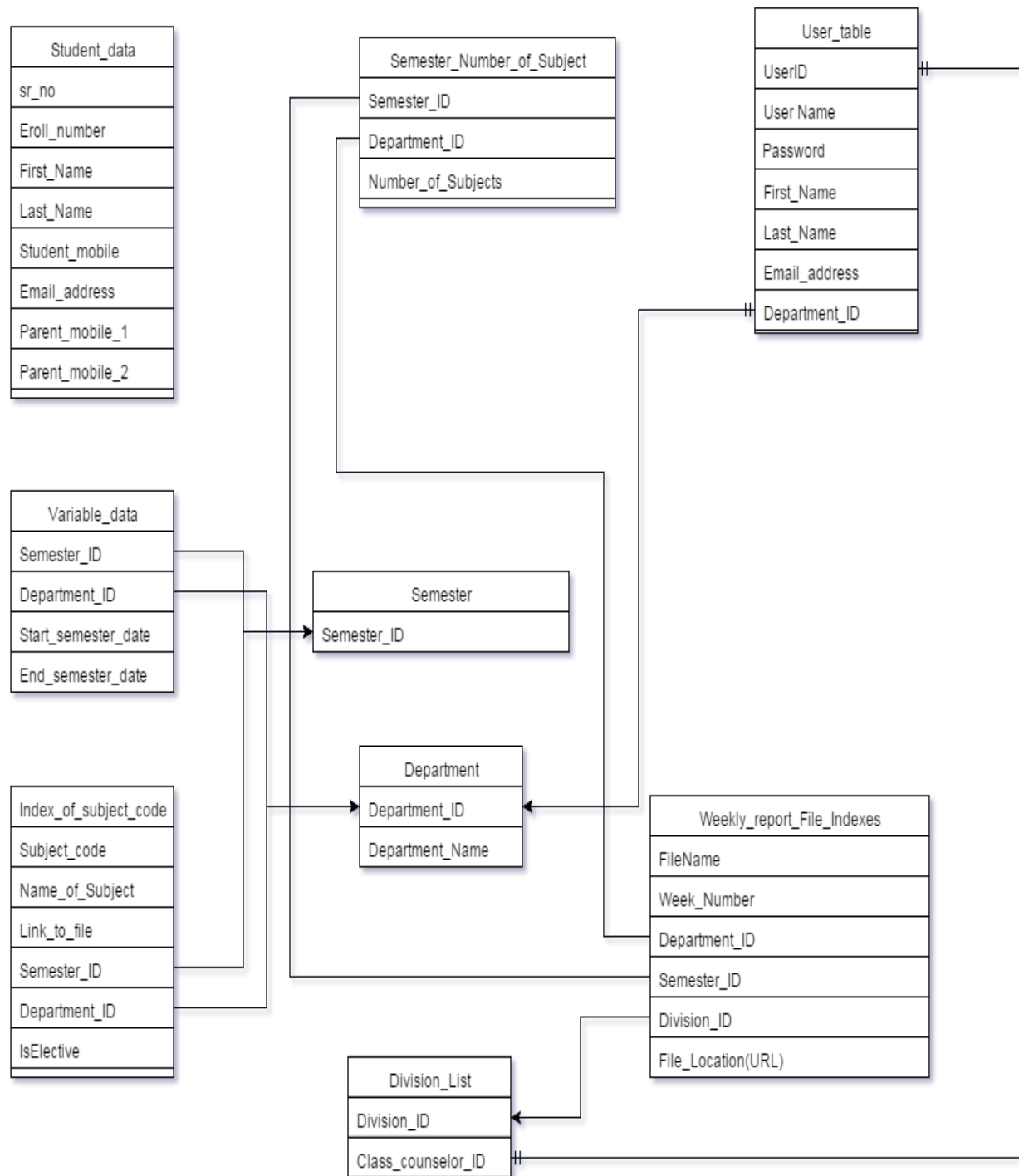


Fig 4(a): Class Diagram

4.7 Activity Diagram

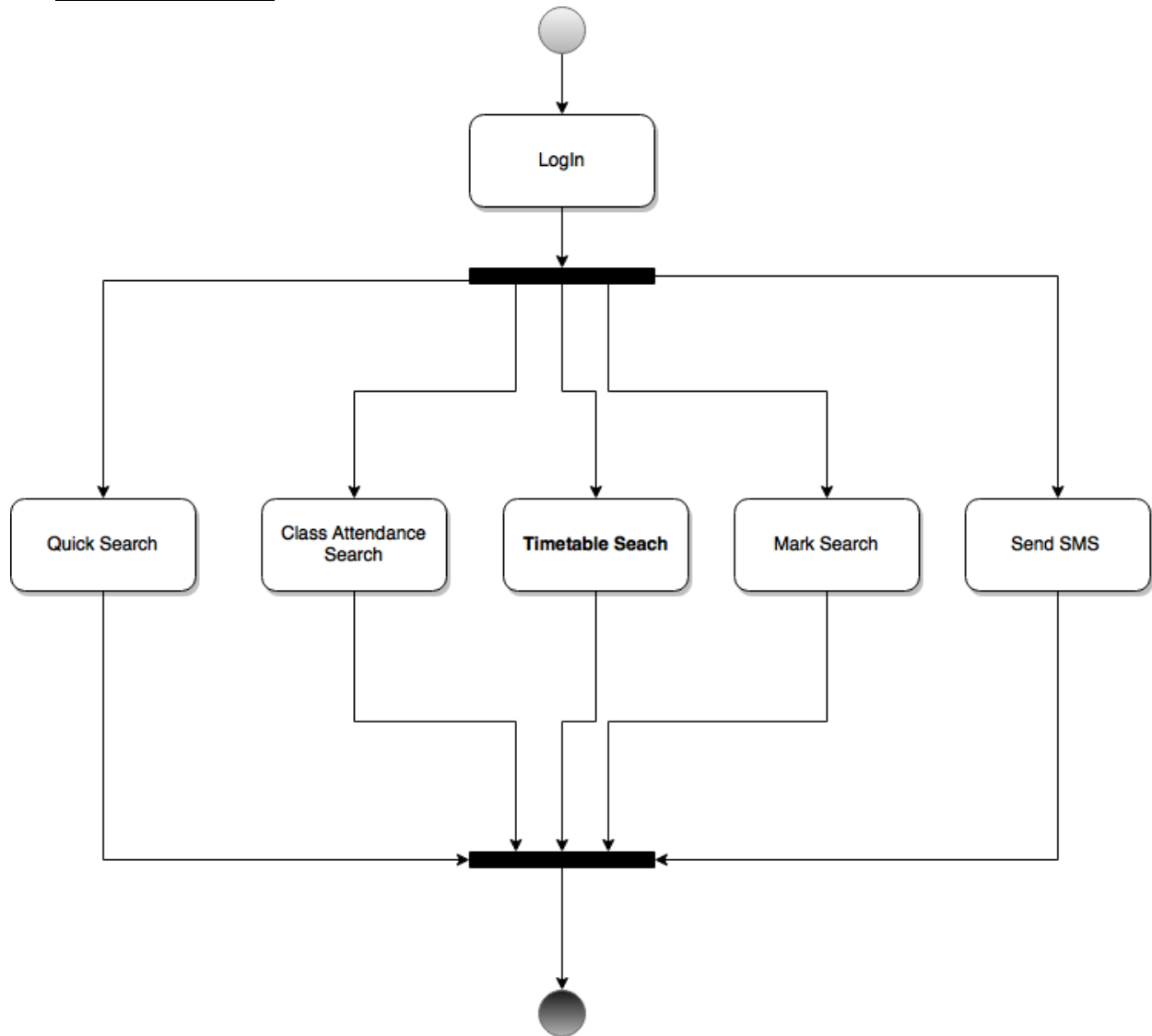


Fig 4(b): User Activity Diagram

4.8 Use Case Diagram

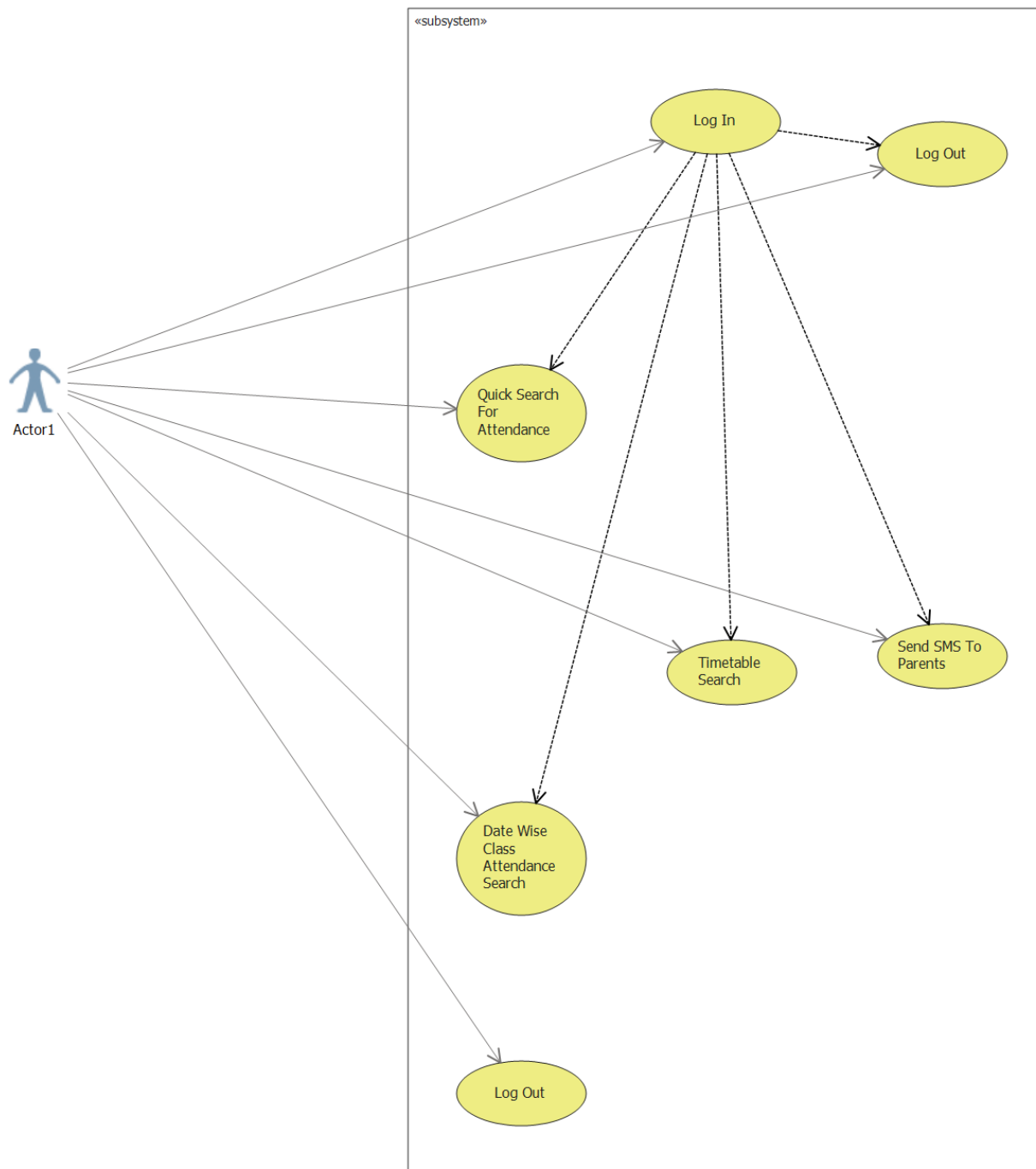


Fig 4(c): Use Case Diagram

4.9 Sequence Diagram

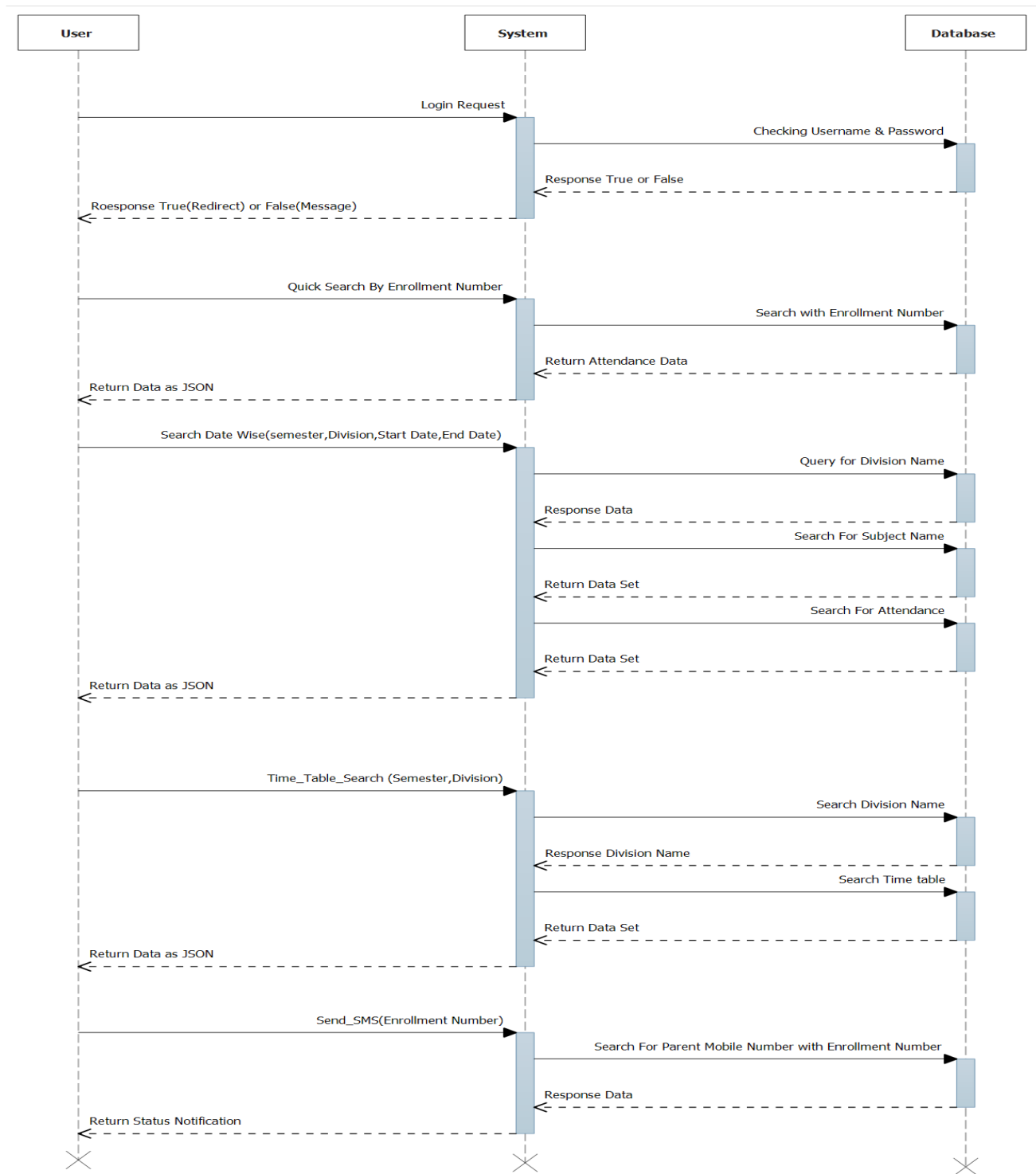


Fig 4(d): Sequence Diagram

4.11 Flow Diagram

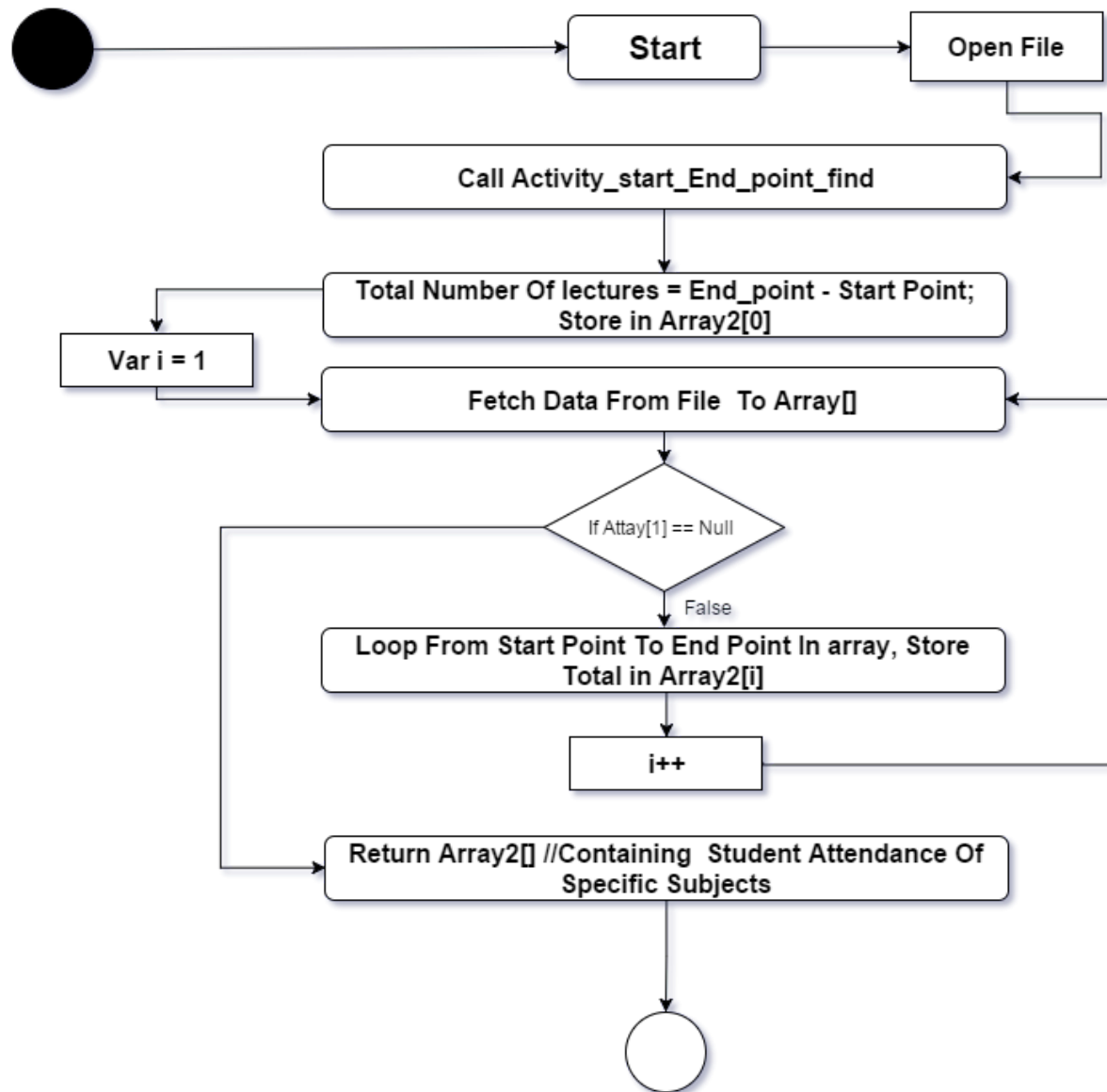


Fig 4(e): Flow Diagram

Chapter 5 - System Design

5.1 Database Design

➤ **Table Name:** Att_testing_struct

Description: This table is used to store Attendance.

Primary Key: Sr no.

att_testing_struct

Column	Type	Null	Default	Comments
Lecture_Date	date	No		
Subject_Code	varchar(6)	No		Format={LLNNNX} where L= letter, N=Number, X=Extra Flag
Sr no (Primary)	int(11)	No		
Erol	varchar(20)	No		
DepartmentID	int(11)	No		FK-->"DepartmentId@att_testing_struct"
Semester	int(11)	No		FK-->"Semester@att_testing_struct"
DivisionID	int(11)	No		FK-->"Division@att_testing_struct"
BatchID	int(11)	No		FK-->"BatchID@att_testing_struct"
Present	tinyint(1)	No		
SDLN	int(10)	No	1	Same_Day_Lecture_Number

➤ **Table Name:** Batch_Year

Description: This table is used to store Batch Years.

Primary Key: Batch_YearID

batch_year

Table comments: Like Batch Of 2012-2016, or 2014-2018, etc....

Column	Type	Null	Default	Comments
Batch_YearID (Primary)	int(11)	No		
Starting_Year	int(4)	No	2012	
Ending_Year	int(4)	No	2016	
Comments	int(11)	Yes	NULL	

- **Table Name:** Department_Data
Description: This table is used to store Department_data.
Primary Key: Department_ID

department_data

Column	Type	Null	Default	Comments
DepartmentID (Primary)	int(11)	No		
Department_Name	varchar(32)	No		
Head_Of_Department	varchar(32)	Yes	NULL	Optional

- **Table Name:** Division_Data
Description: This table is used to store Division Data.
Primary Key: DivisionID

division_data

Column	Type	Null	Default	Comments
DivisionID (Primary)	int(11)	No		
Division_Name	varchar(10)	No		Division Letter
DepartmentID	int(11)	No		FK--->

- **Table Name:** Faculty Data
Description: This table is used to store Faculty Data.
Primary Key: FacultyID

faculty_data

Table comments: This table is made to store Faculty Data for login purpose

Column	Type	Null	Default	Comments
FacultyID (Primary)	int(11)	No		
User_Name	varchar(50)	No		
Password	varchar(32)	No		
DepartmentID	int(11)	No		FK--->"DepartmentId@faculty_data"
First_Name	varchar(32)	No		
Middle_Name	varchar(32)	Yes	NULL	Optional
Last_Name	varchar(32)	No		
designation	varchar(50)	Yes	NULL	

- **Table Name:** Index of Google Sheet
Description: This table is used to store Index of Google Sheet.
Primary Key: sr no.

index_of_google_sheet

Column	Type	Null	Default	Comments
Sr_No (Primary)	int(11)	No		
Subject_Code	varchar(6)	No		FK-->"Subject_Code@index_of_ooogle_sheet"
DivisionID	int(11)	No		FK-->"Division@index_of_ooogle_sheet"
Link_Part_1	varchar(500)	No		Still To Change
GID	varchar(100)	No		Still To Change

- **Table Name:** Semester Start End Date
Description: This table is used to store Semester Start End Date.
Primary Key: Start End Date

semester_start_end_date

Column	Type	Null	Default	Comments
Semester	int(11)	No		FK---->
DepartmentID	int(11)	No		Fk--->
Start_Date	date	No		
End_Date	date	Yes	NULL	

➤ **Table Name:** Student Data

Description: This table is used to store Student Data.

Primary Key: StudentId

student_data

Table comments: Student's Data Storage

Column	Type	Null	Default	Comments
StudentID (<i>Primary</i>)	int(11)	No		System Wide Unique Identification Number
Eroll_Number	varchar(20)	No		Erollment Number
First_Name	varchar(32)	No		
Middle_Name	varchar(32)	Yes	NULL	
Last_Name	varchar(32)	No		
Email_Student	varchar(100)	No		
Email_Parent	varchar(100)	Yes	NULL	Parent's Email ID
Mobile_Student	bigint(11)	No		Without Any Area Code
Mobile_Parent	bigint(11)	No		Without Any Area Code
DepartmentID	int(11)	No		FK-->
Semester	int(11)	No		FK--->
DivisionID	int(11)	No		FK--->
Batch_YearID	int(11)	No		FK--->
IsDetained	tinyint(1)	No	0	0=No, 1=Yes

➤ **Table Name:** Subject_data

Description: This table is used to Subject_data.

Primary Key: Sr No.

subject_data

Column	Type	Null	Default	Comments
Sr No (<i>Primary</i>)	int(11)	No		
Subject_Code	varchar(6)	No		Format={LLNXX} where L= letter, N=Number, X=Extra Flag
Subject_Name	varchar(100)	No		
Semester	int(11)	No		FK---->"Semester@Subject_Data"
DepartmentID	int(11)	No		FK--->"DepartmentID@Department_Data"
IsElective	tinyint(1)	No		0= No, 1=Yes
Extra_Flag_Code	int(11)	Yes	NULL	Like CE 601-X Here is value in this text Field

➤ **Table Name:** Subject_Teching_Faculty_record

Description: This table is used to store Subject Teching Faculty record.

Primary Key: sr_no

subject_teaching_faculty_record

Column	Type	Null	Default	Comments
Sr_No (Primary)	int(11)	No		
Subject_Code	varchar(6)	No		FK--->"Subject_Code@subject_teaching_faculty_record"
FacultyID	int(11)	No		FK-->"FacultyID@subject_teaching_faculty_record"
DepartmentID	int(11)	No		FK-->"DepartmentID@subject_teaching_faculty_record"
Semester	int(11)	No		FK-->"Semester@subject_teaching_faculty_record"
DivisionID	int(11)	No		FK-->"DivisionID@subject_teaching_faculty_record"
Batch_YearID	int(11)	No		FK-->"Batch_YearID@subject_teaching_faculty_record"

➤ **Table Name:** Status_Type_Master

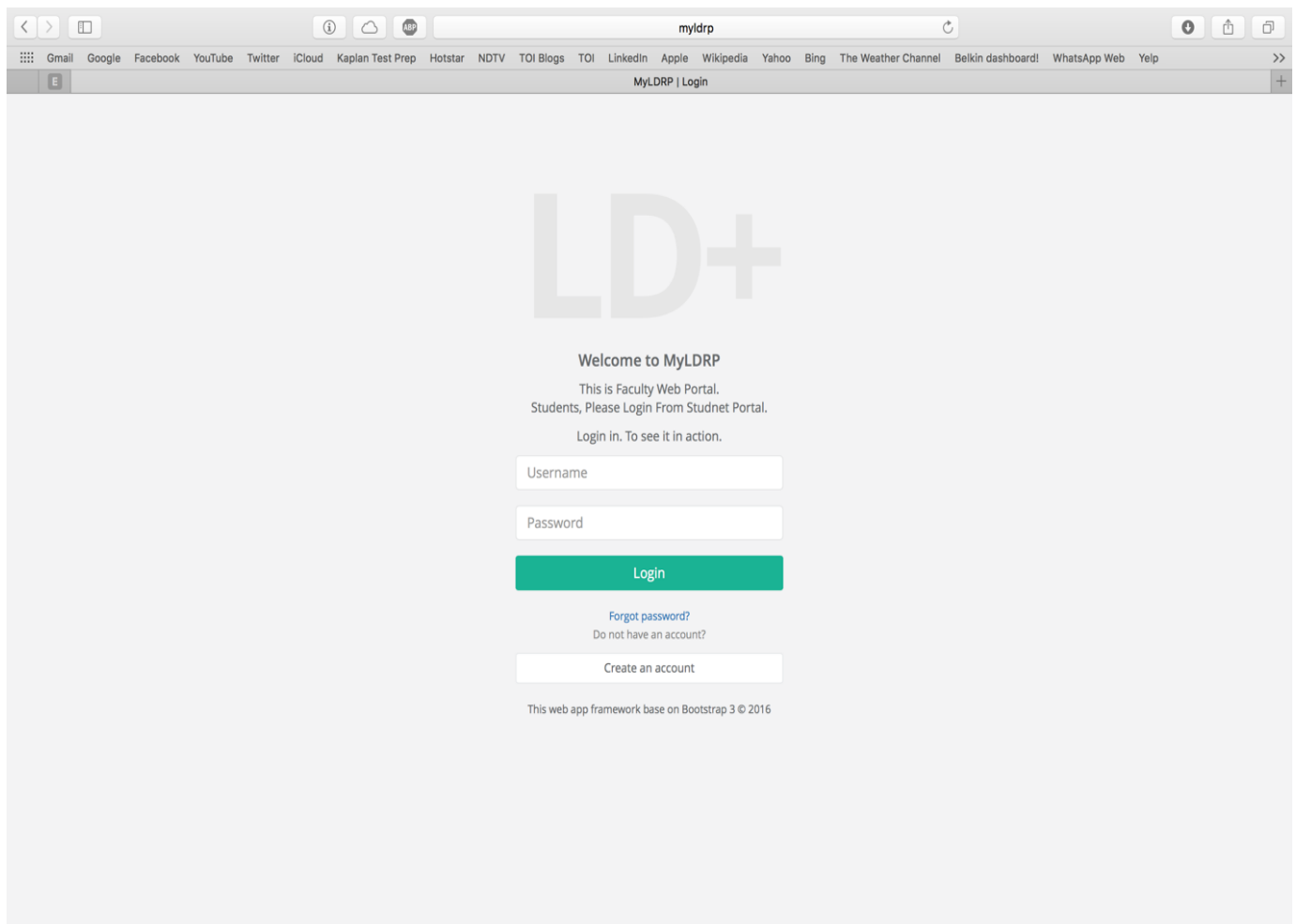
Description: This table is used to store status detail.

Primary Key: Status_Type_Id

time_table_slot_numbers

Column	Type	Null	Default	Comments
slot_number (Primary)	int(2)	No		

5.2 Input/Output and Interface Design



The image shows a web browser window displaying the MyLDRP login page. The browser's address bar shows 'myldrp'. The page features a large 'LD+' logo in the background. The login form includes fields for 'Username' and 'Password', a green 'Login' button, and links for 'Forgot password?' and 'Do not have an account?'. Below these is a 'Create an account' button. At the bottom, a small text line reads 'This web app framework base on Bootstrap 3 © 2016'.

myldrp

MyLDRP | Login

LD+

Welcome to MyLDRP

This is Faculty Web Portal.
Students, Please Login From Studnet Portal.

Login in. To see it in action.

Username

Password

Login

[Forgot password?](#)
[Do not have an account?](#)

Create an account

This web app framework base on Bootstrap 3 © 2016

Fig 5(a): Login Page

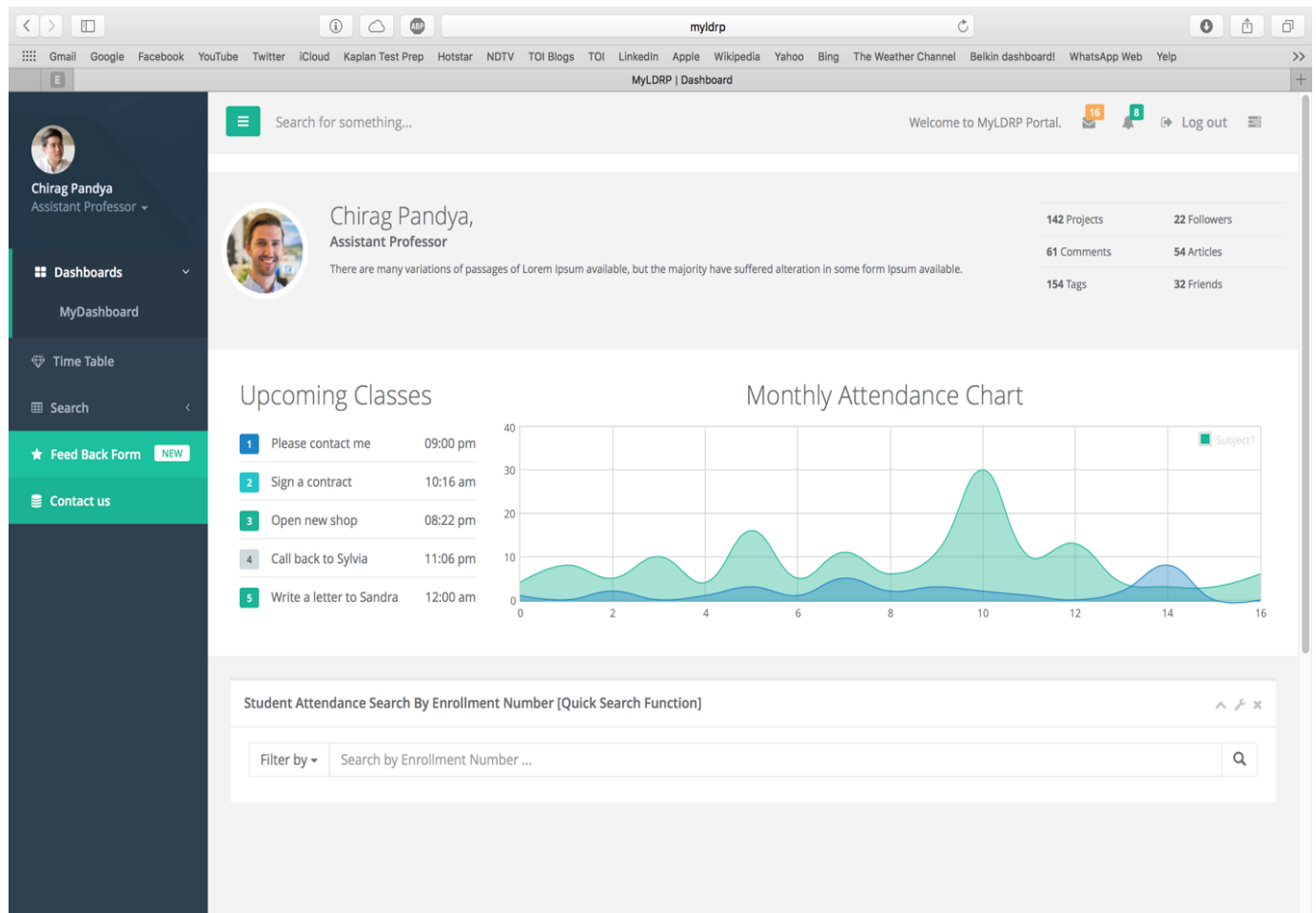


Fig 5(b): Home Page

The screenshot displays the MyLDRP web portal interface. At the top, there is a navigation bar with links to various services like Gmail, Google, Facebook, YouTube, Twitter, iCloud, Kaplan Test Prep, Hotstar, NDTV, TOI Blogs, TOI, LinkedIn, Apple, Wikipedia, Yahoo, Bing, The Weather Channel, Belkin dashboard!, WhatsApp Web, and Yelp. Below this, the main content area is titled "Student Attendance Search By Enrollment Number [Quick Search Function]". A search filter is set to "1418BECE30022". The results section, titled "Attendance of 1418BECE30022", shows a table with 10 entries. The table columns are Enrollment Number, Subject Name, Subject Code, Date, and Present/Absent. All entries for this enrollment number show "Absent" status. The table is paginated, showing 1 to 10 of 20 entries. At the bottom, there is a footer that reads "For Project Show Perpose Only © 2015-2016".

Enrollment Number	Subject Name	Subject Code	Date	Present/Absent
1418BECE30022	Maths2	CE201	2016-03-14	Absent
1418BECE30022	Physics	CE202	2016-03-14	Absent
1418BECE30022	CS	CE203	2016-03-14	Absent
1418BECE30022	FOP	CE204	2016-03-14	Absent
1418BECE30022	MOS	CE205	2016-03-14	Absent
1418BECE30022	EG	CE206	2016-03-14	Absent
1418BECE30022	Physics	CE202	2016-03-15	Absent
1418BECE30022	Maths2	CE201	2016-03-17	Absent
1418BECE30022	Maths2	CE201	2016-03-17	Absent
1418BECE30022	Physics	CE202	2016-03-17	Absent

Fig 5(c): Result Of Quick Search

The screenshot displays the MyLDRP web portal interface. The browser's address bar shows 'myldrp'. The page title is 'MyLDRP | Dashboard'. The user profile on the left is 'Chirag Pandya, Assistant Professor'. The main content area is titled 'Data Tables' and includes a search bar. The search criteria are set to 'Date Wise Search' with a dropdown menu. The search results show a table of attendance data for 'ABC'.

Class Attendance search [Semester Division And Dates]

Search criteria: semester: 2, Division: E, Start Date: 14/03/2016, End Date: 17/03/2016. Search button is present.

Attendance of ABC

Showing 1 to 10 of 62 entries. Search: [input field]. Copy CSV PDF Print buttons are present.

Enrollment Number	Maths2	Physics	CS	FOP	MOS	EG
1418BECE30022	0 / 3	0 / 4	0 / 3	0 / 3	0 / 3	0 / 3
1519BECE30001	3 / 3	3 / 4	3 / 3	3 / 3	3 / 3	3 / 3
1519BECE30002	2 / 3	2 / 4	2 / 3	2 / 3	2 / 3	2 / 3
1519BECE30003	3 / 3	3 / 4	3 / 3	3 / 3	3 / 3	3 / 3
1519BECE30004	3 / 3	3 / 4	3 / 3	3 / 3	3 / 3	3 / 3

Fig 5(d): Date Wise Search

The screenshot displays the MyLDRP web portal interface. At the top, there is a navigation bar with links to various services like Gmail, Google, Facebook, etc. Below this, a sidebar on the left contains links to 'Data Tables', 'Foo Tables', 'jqGrid', 'Feed Back Form', and 'Contact us'. The main content area shows the 'Attendance of ABC' for the date '17/03/2016'. It includes a search bar, a 'Show 10 entries' dropdown, and a table of attendance records. The table has columns for Enrollment Number, Maths2, Physics, CS, FOP, MOS, and EG. The data shows attendance for 10 students, with scores ranging from 0/3 to 4/4. At the bottom, there is a pagination bar with 'Previous', '1', '2', '3', '4', '5', '6', '7', and 'Next' buttons. The footer contains copyright information and storage usage details.

Copyright Example Company © 2014-2015

10GB of 250GB Free.

Fig 5(e): Date Wise Search Results

The screenshot displays a web application interface. At the top, there is a navigation bar with a menu icon, a notification badge showing '16', another notification badge showing '8', a 'Log out' button, and a settings icon. Below the navigation bar, the page title 'Data Tables' is prominently displayed. Underneath the title, a breadcrumb trail reads 'Home / Tables / Data Tables'. The main content area features a search form titled 'Class Attendance search [Semester Division And Dates]'. This form includes four input fields: 'Semester' (with a dropdown arrow), 'Start Date' (with a calendar icon), 'Division' (with a dropdown arrow), and 'End Date' (with a calendar icon). A green 'Search' button is positioned at the bottom right of the form. The interface is clean and modern, with a light gray background and distinct color-coded buttons and labels.

Fig 5(f): Responsive Web Application

Chapter 6 – System Testing

6.1 Testing Method

1) Black-Box Testing

The technique of testing without having any knowledge of the interior workings of the application is called black-box testing. The tester is oblivious to the system architecture and does not have access to the source code. Typically, while performing a black-box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

2) White-Box Testing

White-box testing is the detailed investigation of internal logic and structure of the code. White-box testing is also called **glass testing** or **open-box testing**. In order to perform **white-box** testing on an application, a tester needs to know the internal workings of the code. The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

3) Grey-Box Testing

Grey-box testing is a technique to test the application with having a limited knowledge of the internal workings of an application. In software testing, the phrase the more you know, the better carries a lot of weight while testing an application. Mastering the domain of a system always gives the tester an edge over someone with limited domain knowledge. Unlike black-box testing, where the tester only tests the application's user interface; in grey-box testing, the tester has access to design documents and the database. Having this knowledge, a tester can prepare better test data and test scenarios while making a test plan.

Chapter 7-Conclusion

- ❖ At the end we would like to conclude that we spent a lot of time on implementing the various features in the website.
- ❖ We would like to mention that, it was a not an easy job to co-ordinate with a team which is located at a different geographical location, far away from my current location.
- ❖ We would like to mention that this website has a couple of nice features like Report Generation.
- ❖ At the end we conclude by saying that apart from its limitations, the features existing in the website as of now are interesting and will satisfy the basic demands of the users.

Chapter 8 –Bibliography

- www.youtube.com/user/aspnet/mvc
- www.w3schools.com/asp.net/mvc_publish.asp
- www.asp.net/get-started/websites
- Microsoft ASP.NET MVC Step-by-Step