

PROJECT ON WEB APPLICATION FOR INTRA-COLLEGE COMMUNICATION SYSTEM

REPORT OF MAJOR PROJECT BY:

HIMANSHU TYAGI

ROLL NO. 2002300100074

CERTIFICATE

The report of the Project titled **Web Application for Intra-College Communication System** submitted by Himanshu Tyagi has been prepared for the partial fulfillment of the requirements for Dronacharya Group of Institutions. The report is hereby forwarded.

.....

ANTI PLAGIARISM DECLARATION

- 1.** I know that plagiarism means taking and using the ideas, writings, works or inventions of another as if they were one's own. I know that plagiarism not only includes verbatim copying, but also the extensive use of another person's ideas without proper acknowledgement (which includes the proper use of quotation marks). I know that plagiarism covers this sort of use material found in textual sources and from the Internet.
- 2.** I acknowledge and understand that plagiarism is wrong.
- 3.** I understand that my project work must be accurately referenced. I have followed the rules and conventions concerning referencing, citation and the use of quotations as set out in the Department Guide.
- 4.** This assignment is my own work, or my group's own unique group assignment. I acknowledge that copying someone else's assignment, or part of it, is wrong, and that submitting identical work to others constitutes a form of plagiarism.
- 5.** I have not allowed, nor will I in the future allow, anyone to copy my work with the intention of passing it off as their own work.

Student Name:

Himanshu Tyagi

TABLE OF CONTENTS

<u>Topics</u>	<u>Page No.</u>
1. Abstract.....	1
2. Introduction.....	2
3. Software Requirement Specification.....	4
4. System Planning & SDLC.....	11
5. Constraints of use.....	12
6. Technology used.....	12
7. Feasibility Study.....	14
8. Entity relationship diagram(ERD).....	15
9. Data flow diagram (DFD).....	15
10. Database designed	17
11.Unified Modeling Language (UML) Diagram.....	20
12.Source Codes.....	24
13. Screenshots of application.....	29
14.Aim and Objectives.....	33
15. Future Scope of Work.....	33
16. Conclusion.....	33
17. Bibliography.....	34

1. ABSTRACT

The purpose of Intra-College Communication System is to automate the existing manual system by the help of computerized equipment's and full- fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy assessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

Intra-College Communication System, as described above, can lead to error free, secure, reliable and fast networking system. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

2. INTRODUCTION

The “Intra-College Communication System” has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by this existing system. Moreover this system is effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it provides it is user-friendly. Intra-College Communication System, as described above, can lead to error free, secure, reliable and fast networking system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources.

Every institution, whether big or small, has challenges to overcome and managing the information's of friends, users, shares, videos, photos. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. These systems will ultimately allow you to better manage resources.

The project includes two main sections:

- ❖ User (College Campus Portal Member):
 - ❖ In this system user is only priority for accessing the “*Intra-College Communication System*” application.
 - ❖ Here user can create his/her own profile along with all the details and his/her profile photo.
 - ❖ Every users have individual & unique log in credential to go through the portal.
 - ❖ This website provides separate access for different sections of college (viz. Academic, Administration, Training & Placement, Library, Finance, Examination sections).
 - ❖ User can post including video, photo& any other kind of file and also can view the post in one place.
 - ❖ User can also like, dislike and comments with respect to the post.
 - ❖ There will be group facility which can be created by user.
 - ❖ Each user can see his/her own group's post as well as other groups' post if permission is given.
 - ❖ User can send message to another specific user.
 - ❖ User can also report abuse to the system for any unwanted post.

❖ Admin:

- ❖ Admin can modify any data and delete unwanted post and even can block users.
- ❖ Admin can add users and provide them user id and password via OTP confirmation which will be required by the users for accessing this website.
- ❖ Admin can block any user.

In this system developer is a part for developing the application. Every developer has a permission to access all data of database of Intra-College Communication System and also a permission for create, delete, update and many other features for betterment of security and future upgradation.

3. SOFTWARE REQUIREMENT SPECIFICATION (SRS)

3.1 Introduction

3.1.1 Purpose

The main objective of this document is to illustrate the requirements of the project *“Intra-College Communication System”*. The document gives the detailed description of both functional and non-functional requirements proposed by the client. The purpose of this project is to build a connectivity between students and faculties to reduce the manual work for conveying any information regarding to the college. It tracks all the details about the Posts, Shares and Photos etc.

3.1.2 Scope of Document Project

Intra-College Communication System is a system that can be easily use by various users(viz. Academic, Administration, Training & Placement, Library, Finance, Examination sections).

This project is specially designed for the use of students.

- Student can view study materials, Exam routine and important news related to their college, class, course in one place with a click of mouse.
- Student can ask for specific query for any subject through the post option and if anyone have the solution he / she can respond with the details.
- Students can have the messaging facility for any personal query to the portal members.
- There will be group facility which can be created by students or teachers for any specific discussion.
- Moreover any student can interact with any college persons (like students of any department, governing body, administrative section etc.)

It is especially useful for any educational institute where modifications in the content can be done easily according to requirements

The project can be easily implemented under various situations. We can add new features as and when we require, making reusability possible as there is flexibility in all the modules.

The language used for developing the project is PHP as it quite advantageous than other languages in terms of performance, tools available, cross platform compatibility and development process.

3.1.3 Definitions, Acronyms and Abbreviations

SQL -> Structured Query Language

ERD -> Entity Relationship Diagram

UML -> Unified Modeling Language

SRS -> Software Requirement Specification

SDLC -> Software Development Life Cycle

DFD -> Data Flow Diagram

3.2 Overall Descriptions

3.2.1 User Classes and Characteristics

The system provides different types of services based on the type of users. Here users are accessing the website after create his/her own account. When a user open his/her account then shows his/her posts and also show the all members. Users can also edit their profile along with all the details and his/her profile photo.

The features that are available to the Users are:

- Create his/her own profile along with all the details and his/her profile photo.
- Every users have individual & unique log in credential to go through the portal.
- User can post including video, audio & any other kind of file and also can view the post in one place.
- User can also like and comments with respect to the post.
- There will be group facility which can be created by user.
- Each user can see his/her own group's post as well as other groups' post if permission is given.
- User can send message to another specific user.
- User can also report spam to the portal for any unwanted post.
- This website provides separate access for different sections of college (viz. Academic, Administration, Training & Placement, Library, Finance, Examination sections).

3.2.2 Operating Environment

This application will be operating in webpages through browser. The Intra-College Communication System is a web application and can be operated through browser in web pages. The only requirement to use this website would be the internet connection and a device which can access high speed internet.

3.2.3 Assumptions and Dependencies

The assumptions are:

- The coding should be error free.
- The application should be user-friendly so that it is easy to use for the users.
- Valid information of every users must be stored in database that is accessible by the website.
- The system should provide more storage capacity and provide fast access to the database.
- The system should provide search facility and support quick transactions.
- The Intra-College Communication System is running 24 hours a day.
- Users may access from any browser that has Internet browsing capabilities and an Internet connection.
- Users must have their correct usernames and passwords to enter into their college campus accounts and do actions.

The dependencies are:

- The specific hardware and software due to which the product will be run.
- On the basis of listing requirements and specification the project will be developed and run.
- The end users should have proper understanding of the product.
- The information of all the users must be stored in a database that is accessible by the college campus system.
- Any update regarding the users profile, posts and any other updation is to be recorded to the database and the data entered should be correct.

3.2.4 Requirement

Software Configuration:

This software package is developed using HYML, CSS, Bootstrap as front end, JavaScript, jQuery as client side validation, Ajax as server side validation, PHP as business logic, MySQL as to store the database and Apache as web server.

Operating System: Windows XP, Windows 7, Windows 8, Windows 10

Front end: HTML, CSS, Bootstrap

Client side Validation: JavaScript, jQuery

Server side Validation: Ajax

Business Logic: PHP

Database: MySQL

Web Server: Apache

Hardware Configuration:

Processor: Core i3, 1.5MHz

Hard Disk: 150 GB

RAM: 2GB

Resolution: 480 X 800

3.2.5 Data Requirement

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts. In this project the inputs will be queries as fired by the users like create an account. Now the output will be visible when the user requests the server to get details of their own account and also accounts of the other members in the form of time, date.

3.3 External Interface Requirement

3.3.1 GUI

This application provides good graphical interface for the user and the administrator can operate on the system, performing the request task such as create, update, delete and also view the every details.

- The user interface must be customizable by the administrator.
- All the modules provided with the software must fit into this graphical user interface and accomplish to the standard defined.
- The design should be simple and all the different interfaces should follow a standard template.
- The user interface should be able to interact with the other users.
- Design of registration page, log in page and also otp verification page is easy to understand for every user.

3.4 System Features

This is possible by providing:-

- User accessibility available if he/she is a part of a particular college campus.
- Every users have individual & unique log in credential to go through the portal.
- User can register his/her profile along with all the details and his/her profile photo.
- This website provides separate access for different sections of college (viz. Academic, Administration, Training & Placement, Library, Finance, Examination sections)
- User can post including photo, audio documents, video documents & any other kind of file. User can also comments with respect to the post.
- Each user can see his/her own group's post as well as other groups' post if permission is given.
- User can also report spam to the portal for any unwanted post.
- Admin functionality will be added(admin can modify and delete unwanted post and even can block users)

3.5 Other Non-functional Requirements

3.5.1 Performance Requirement

The proposed system that we are going to develop will be used as the chief performance system within the different college campus and also users. Therefore, it is expected that the database would perform functionally all the requirements that are specified by the college and also for the users.

- The performance of the system should be fast and accurate.
- Intra-College Communication System shall handle expected and non-expected errors in way that prevent loss in information and long downtime period. Thus it should have inbuilt error testing to identify search or data check/fetch.
- The system should be able to handle large amount of data. Thus it should accommodate large number of data entry of a particular college campus without any fault.

3.5.2 Satisfy Requirement

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required take the database backup.

3.5.3 Security Requirement

- System will use secured database.
- Normal users can just read instruction and use this system but they cannot edit or modify anything except their personal and some other information.
- System will have two type of user and user has access constraints.
- Proper user authentication should be provided.
- No one should be able to hack user's details or any other information.
- There should be separate part for users that no users can access the database and only admin has the rights to update the database.

3.5.4 Requirement attributes

- There may be multiple admins creating the project, all of them will have the right to create changes to the system. But the users cannot do changes.
- The project should be open source.
- The quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database.
- The user be able to easily access this system from anywhere at any time.

3.5.5 Business Rule

A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, make a decision, or infer new data from existing data. This includes the rules and regulations that the System users should abide by. This includes the cost of the project and the discount offers provided. The users should avoid illegal rules and protocols. Neither admin nor member should cross the rules and regulations.

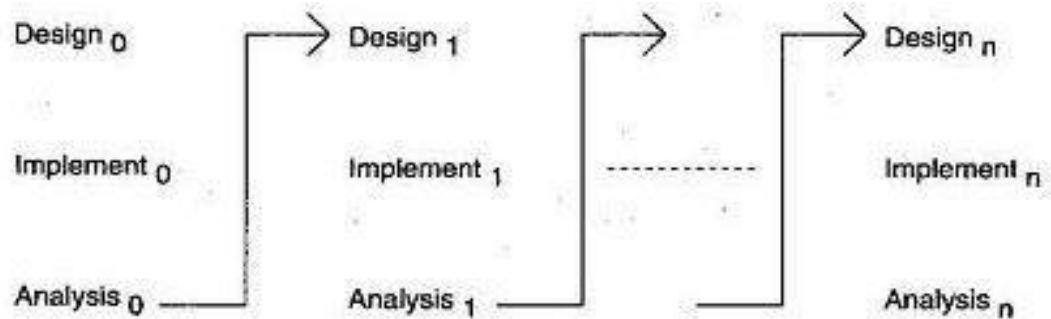
3.5.6 User Requirement

The users of the system are members of the *“Intra-College Communication System”*. The users are assumed to have basic knowledge of the computers and internet browsing. The administrators of the system should have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper user interface, user manual, online help and the guide to install and maintain the system must be sufficient to educate the users on how to use the system without any problems.

4 SYSTEM PLANNING & SYSTEMS DEVELOPMENT LIFECYCLE (SDLC)

We have chosen “*Iterative Life Cycle Model*” for developing this application, because an iterative life cycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which can then be reviewed in order to identify further requirements. This process is then repeated, producing a new version of the software for each cycle of the model.

Here is the diagram of the iterative life cycle model which depicts its working flow.



Few advantages for choosing the SDLC are –

- In iterative model we are building and improving the product step by step. Hence, we can track the defects at early stages. This avoids the downward flow of the defects.
- Testing and debugging in smaller iteration is easy.
- In iteration model we can get the reliable user feedback. When presenting sketches and blueprints of the product to user for their feedback, we are effectively asking them to imagine how the product will work.
- Progress can be measured.
- In iterative model less time is spent on documentation and more time is given for designing.
- Risk are identified and resolved during iteration; and each iteration is an easily managed milestone. It supports changing requirements.

5 CONSTRAINTS OF USE

- Admin and user must remember login id and password.
- User need to have a personal computer or cell phone with internet connection.

6 TECHNOLOGY USED

6.1 Front-end Design: HTML, CSS, Bootstrap

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Bootstrap is a free and open-source front-end library for designing websites and web applications. It contains HTML- and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many web frameworks, it concerns itself with front-end development only.

6.2 Client side validation: JavaScript, jQuery

JavaScript often abbreviated as JS, is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm. Alongside HTML and CSS, JavaScript is one of the three core technologies of the World Wide Web. JavaScript enables interactive web pages and thus is an essential part of web applications. The vast majority of websites use it, and all major web browsers have a dedicated JavaScript engine to execute it.

jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. It is free, open-source software using the permissive MIT License. Web analysis indicates that it is the most widely deployed JavaScript library by a large margin.

6.3 Server side validation: Ajax

Ajax is not a single technology. Ajax is a set of Web development techniques using many Web technologies on the client side to create asynchronous Web applications. With Ajax, Web applications can send and retrieve data from a server asynchronously (in the background) without interfering with the display and behavior of the existing page. By decoupling the data interchange layer from the presentation layer, Ajax allows Web pages, and by extension Web applications, to change content dynamically without the need to reload the entire page.

6.4 Business logic: PHP

PHP: Hypertext Preprocessor (or simply PHP) is a server-side scripting language designed for web development but also used as a general-purpose programming language. It was originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive acronym PHP: Hypertext Preprocessor. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

6.5 Database: MySQL

MySQL is an open source relational database management system. For proprietary use, several paid editions are available, and offer additional functionality. In this project MySQL has been used to store, update, retrieve and delete related to user's data and other additional data about projects.

6.6 Web Server: Apache

The Apache HTTP Server, colloquially called Apache, is a free and open-source cross-platform web server, released under the terms of Apache License 2.0. Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. The Apache HTTP Server is cross-platform; as of 1 June 2017 92% of Apache HTTP Server copies run on Linux distributions. Version 2.0 improved support for non-Unix operating systems such as Windows and OS/2. Old versions of Apache were ported to run on OpenVMS and NetWare.

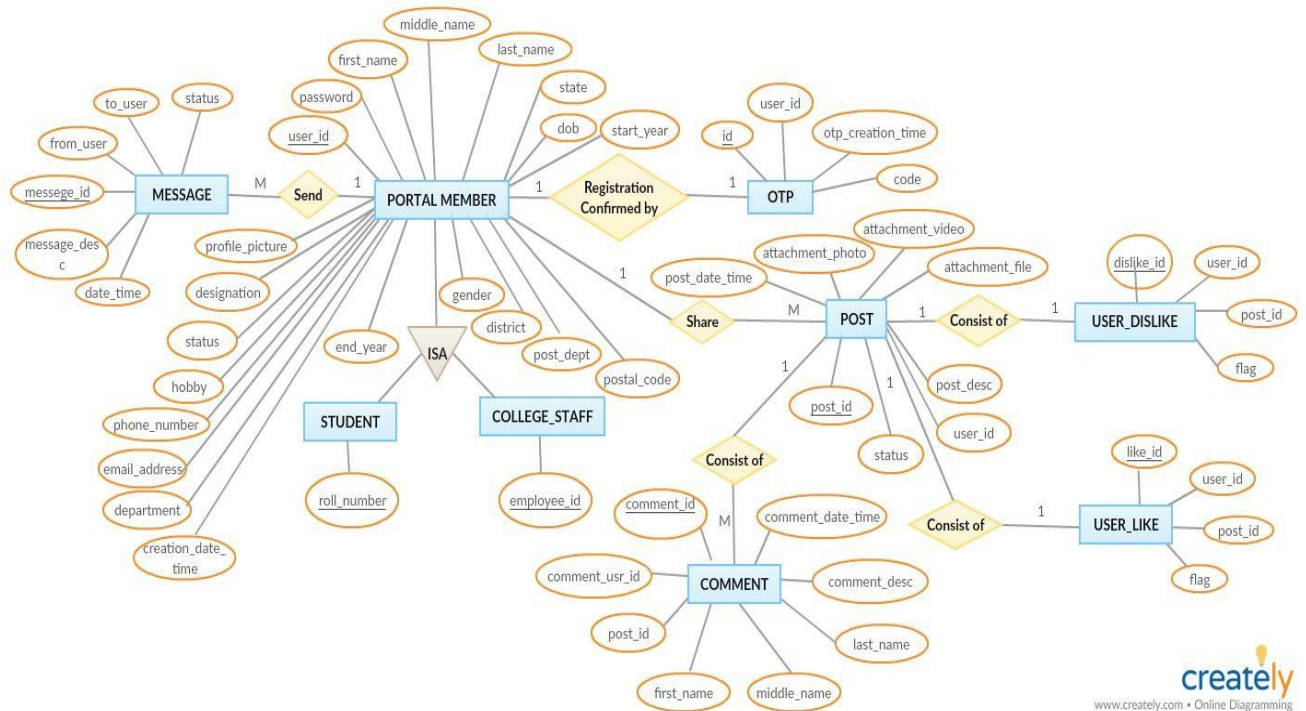
7 FEASIBILITY STUDY

A feasibility study is an analysis of how successfully a project can be completed, accounting for factors that affect it such as economic, technological and operational. Project managers use feasibility studies to determine potential positive and negative outcomes of a project before investing a considerable amount of time and money into it.

During the stage of our feasibility study, we had to undergo the following steps as described under:

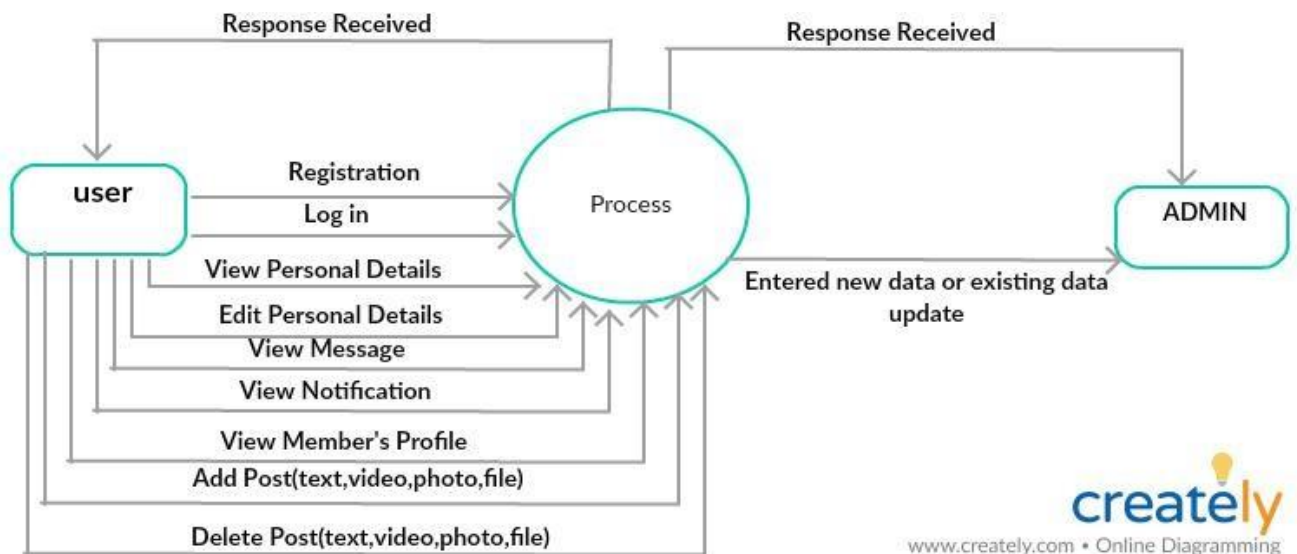
- Identify the origin of data at different levels of the system.
 - Identify the expectation of end user from the finished product/system.
 - Analyze the drawback(s) of the existing system.
-
- **Technical feasibility study:** It lays out details on how a good or service will be delivered, which includes transportation, business location, technology needed, materials and labour.
 - **Financial feasibility study:** It is a projection of the amount of funding or startup capital needed, what sources of capital can and will be used and what kind of return can be expected on the investment.
 - **Organizational feasibility study:** It is a definition of the corporate and legal structure of the business; this may include information about the founders, their professional background and the skills they possess necessary to get the company off the ground and keep it operational.

8 ENTITY RELATIONSHIP DIAGRAM(ERD)

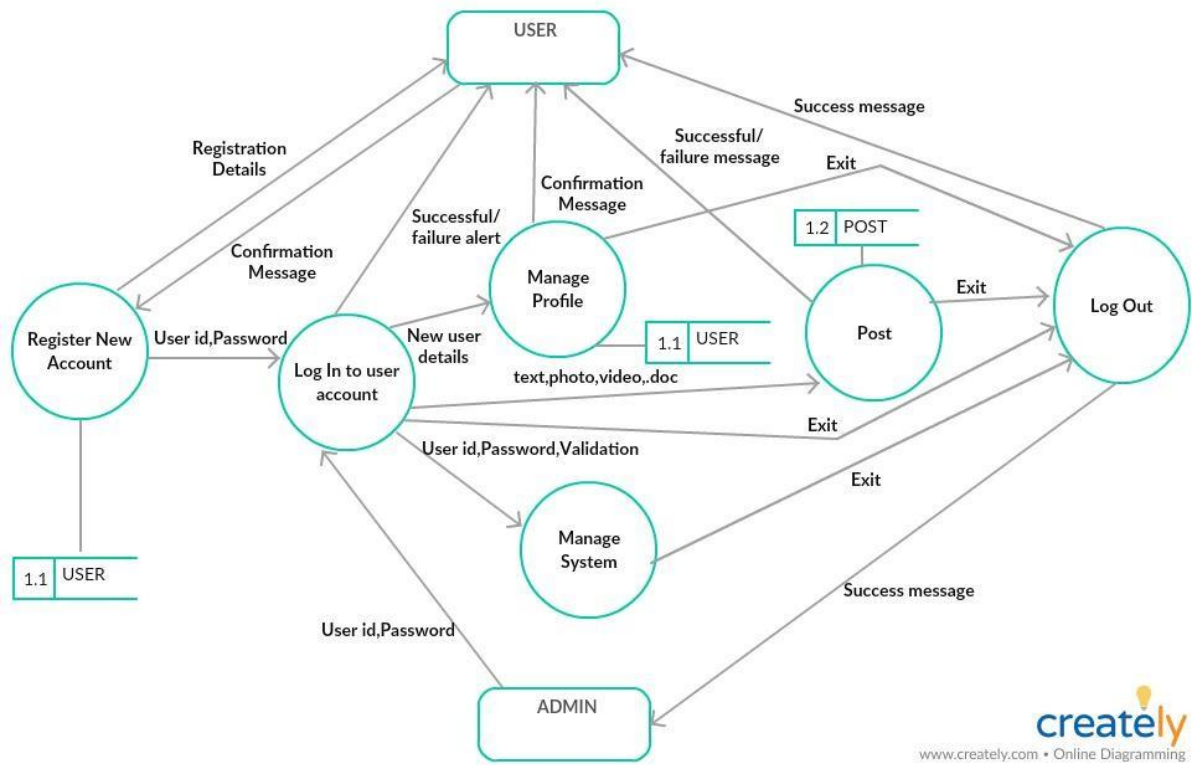


9 DATA FLOW DIAGRAM(DFD)

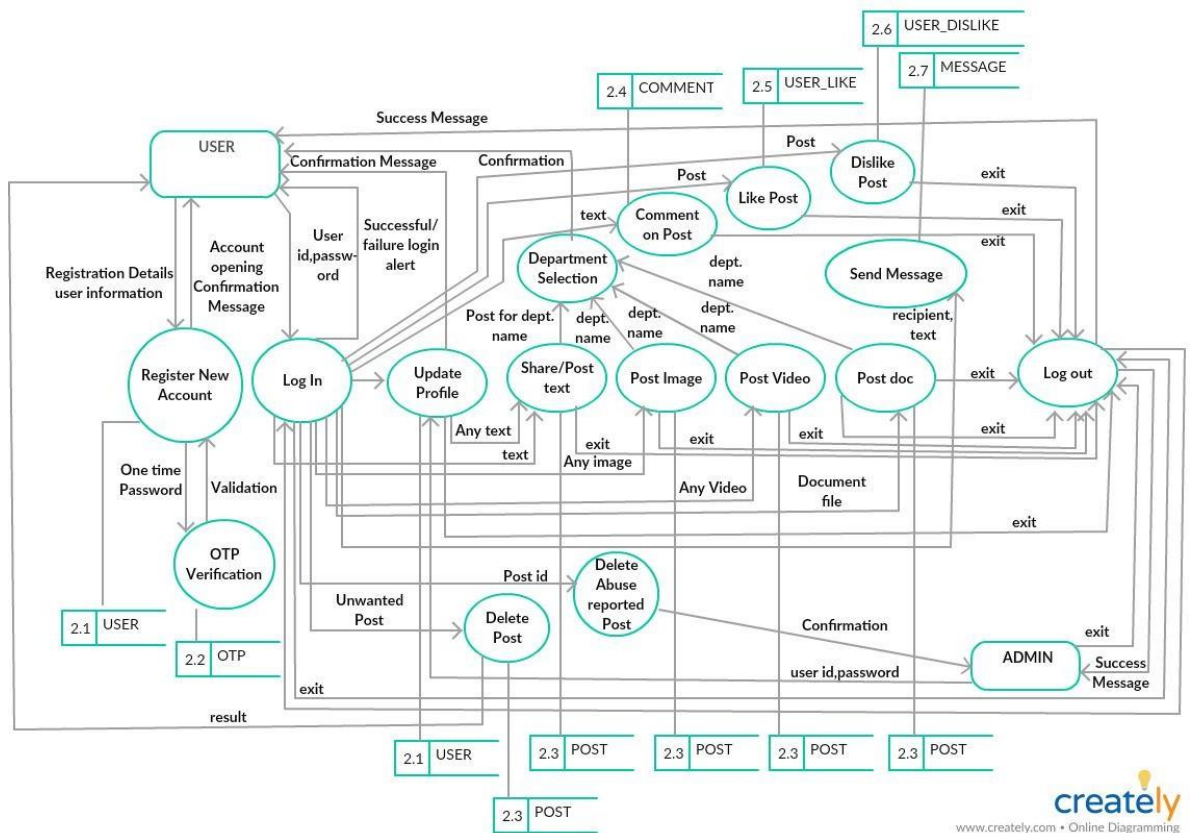
- level 0



- level 1



- level 2



10 DATABASE DESIGNED

10.1 Database: college_campus

Table	Action	Rows	Type	Collation	Size	Overhead
comment	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
message	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 KiB	-
otp	Browse Structure Search Insert Empty Drop	31	InnoDB	latin1_swedish_ci	16 KiB	-
post	Browse Structure Search Insert Empty Drop	11	InnoDB	latin1_swedish_ci	16 KiB	-
state	Browse Structure Search Insert Empty Drop	36	InnoDB	latin1_swedish_ci	16 KiB	-
user	Browse Structure Search Insert Empty Drop	22	InnoDB	latin1_swedish_ci	16 KiB	-
user_dislike	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 KiB	-
user_like	Browse Structure Search Insert Empty Drop	16	InnoDB	latin1_swedish_ci	16 KiB	-
user_master	Browse Structure Search Insert Empty Drop	200	InnoDB	latin1_swedish_ci	16 KiB	-
9 tables	Sum	318	InnoDB	latin1_swedish_ci	144 KiB	0 B

10.2 Database table:

1. User

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	first_name	char(50)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
2	middle_name	char(50)			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext More
3	last_name	char(50)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
4	user_id	varchar(100)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
5	password	varchar(100)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
6	gender	char(10)			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext More
7	hobby	varchar(500)			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext More
8	district	varchar(100)			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext More
9	state	varchar(100)			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext More
10	postal_code	bigint(20)			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext More
11	phone_number	bigint(30)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
12	email_address	varchar(100)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
13	dob	date			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext More
14	department	varchar(100)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
15	designation	varchar(100)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
16	start_year	varchar(100)			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext More
17	end_year	varchar(100)			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext More
18	profile_picture	varchar(500)			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext More
19	status	varchar(50)			No	None		Change Drop Primary Unique Index Spatial Fulltext More
20	creation_date_time	timestamp			on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	Change Drop Primary Unique Index Spatial Fulltext More

2. user_master

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	id	int(11)		No	None	AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext Distinct values
<input type="checkbox"/>	2	userid	varchar(50)		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values

3. state

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	state_id	int(100)		No	None	AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext More
<input type="checkbox"/>	2	state_name	varchar(500)		No	None		Change Drop Primary Unique Index Spatial Fulltext More

4. post

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	post_id	int(11)		No	None	AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	user_id	varchar(100)		No	None		Change Drop More
<input type="checkbox"/>	3	post_desc	longtext		Yes	NULL		Change Drop More
<input type="checkbox"/>	4	attachment_photo	varchar(500)		Yes	NULL		Change Drop More
<input type="checkbox"/>	5	attachment_video	varchar(500)		Yes	NULL		Change Drop More
<input type="checkbox"/>	6	attachment_file	varchar(500)		Yes	NULL		Change Drop More
<input type="checkbox"/>	7	post_dept	varchar(50)		No	None		Change Drop More
<input type="checkbox"/>	8	post_date_time	timestamp	on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP	Change Drop More
<input type="checkbox"/>	9	status	varchar(50)		No	None		Change Drop More

5. comment

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	comment_id	int(11)		No	None	AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	post_id	int(11)		No	None		Change Drop More
<input type="checkbox"/>	3	comment_user_id	varchar(100)		No	None		Change Drop More
<input type="checkbox"/>	4	first_name	varchar(50)		No	None		Change Drop More
<input type="checkbox"/>	5	middle_name	varchar(50)		No	None		Change Drop More
<input type="checkbox"/>	6	last_name	varchar(50)		No	None		Change Drop More
<input type="checkbox"/>	7	comment_desc	mediumtext		No	None		Change Drop More
<input type="checkbox"/>	8	cmnt_date_time	timestamp	on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP	Change Drop More

6. message

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	messege_id	int(11)		No	None	AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	from_user	varchar(100)		No	None		Change Drop More
<input type="checkbox"/>	3	to_user	varchar(100)		No	None		Change Drop More
<input type="checkbox"/>	4	message_desc	text		No	None		Change Drop More
<input type="checkbox"/>	5	status	varchar(50)		No	None		Change Drop More
<input type="checkbox"/>	6	date_time	timestamp	on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP	Change Drop More

7. otp

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	id			No	None	AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	user_id			No	None		Change Drop More
<input type="checkbox"/>	3	code			No	None		Change Drop More
<input type="checkbox"/>	4	otp_creation_time		on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP	Change Drop More

8. user_like

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	like_id			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext More
<input type="checkbox"/>	2	post_id			No	None		Change Drop Primary Unique Index Spatial Fulltext More
<input type="checkbox"/>	3	user_id			No	None		Change Drop Primary Unique Index Spatial Fulltext More
<input type="checkbox"/>	4	flag			No	None		Change Drop Primary Unique Index Spatial Fulltext More

9. user_dislike

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	dislike_id			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext More
<input type="checkbox"/>	2	user_id			No	None		Change Drop Primary Unique Index Spatial Fulltext More
<input type="checkbox"/>	3	post_id			No	None		Change Drop Primary Unique Index Spatial Fulltext More
<input type="checkbox"/>	4	flag			No	None		Change Drop Primary Unique Index Spatial Fulltext More

11 UNIFIED MODELING LANGUAGE (UML) DIAGRAM

11.1 Use Case Diagram

- Welcome Page

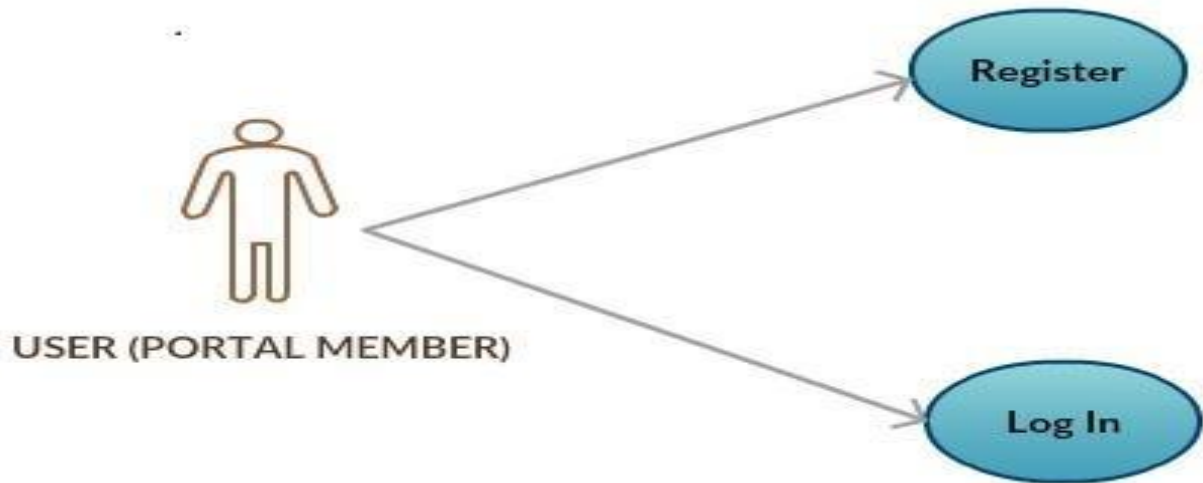


Fig: USE CASE DIAGRAM OF WELCOME PAGE

- Registration Page

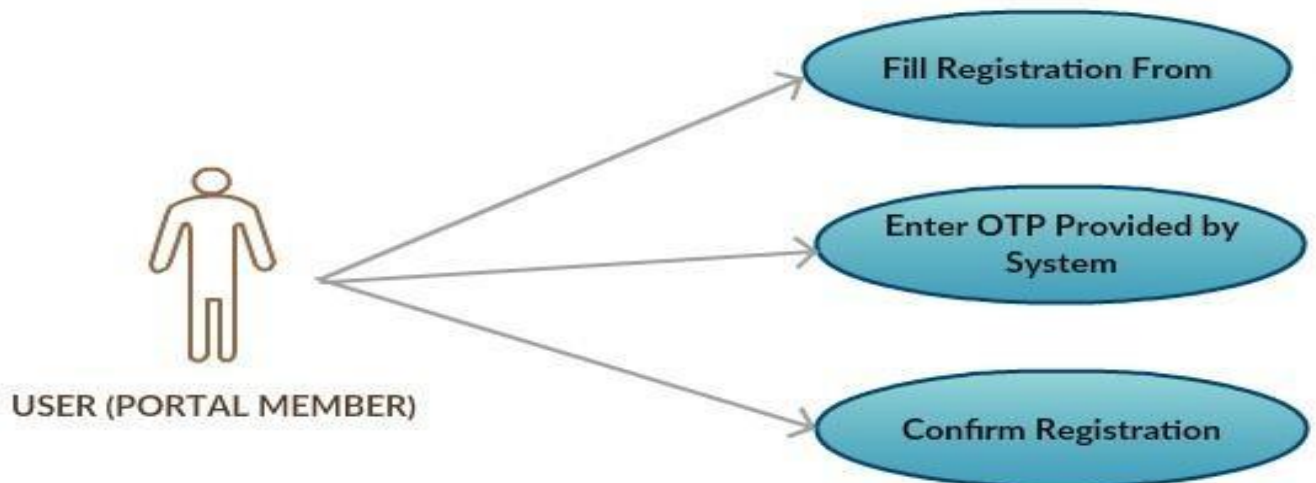


Fig: USE CASE DIAGRAM OF REGISTRATION PAGE

- After Login Page

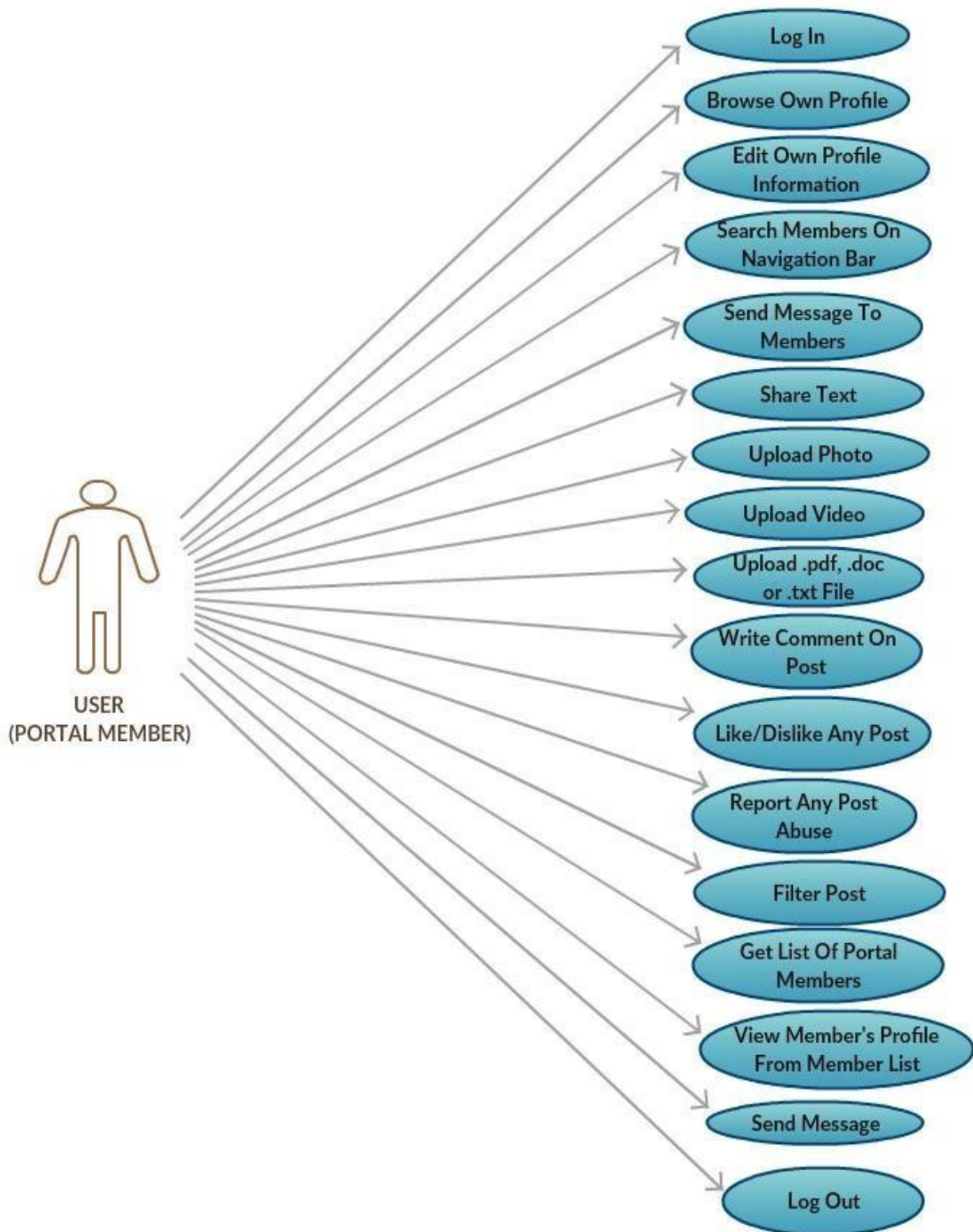


FIG: USE CASE DIAGRAM OF AFTER LOGIN PAGE

- Admin

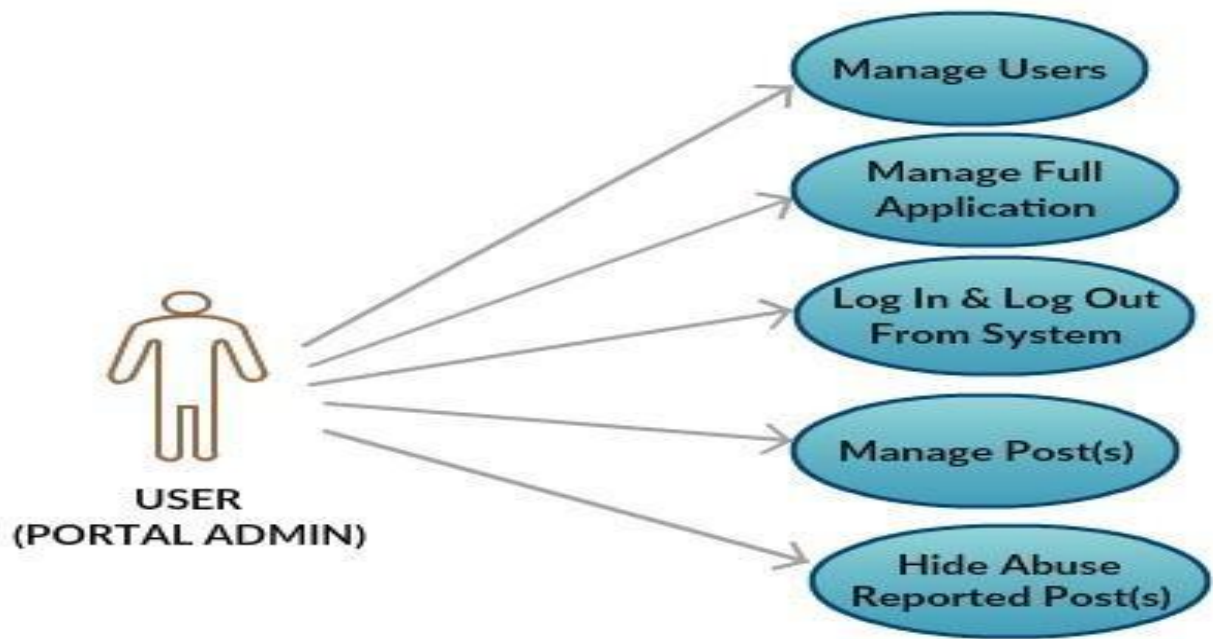


FIG: USE CASE DIAGRAM OF ADMIN PAGE

11.2 Activity Diagram

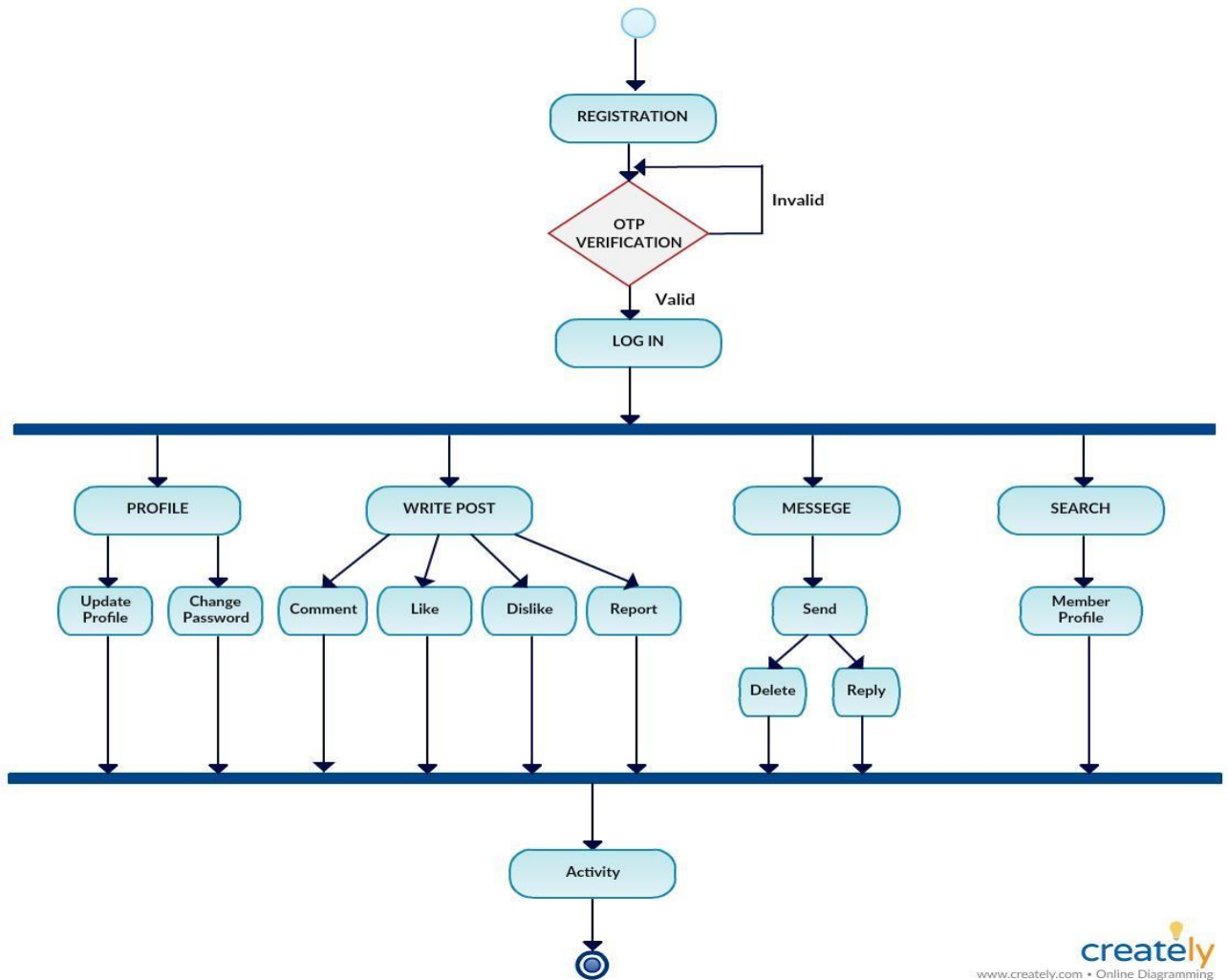


FIG:ACTIVITY DIAGRAM

12 SOURCE CODE

12.1 DBconnect.php (Creating Database Connection)

```
1  <?php
2  class DBconnect{
3      function __construct(){
4      }
5      function connect(){
6          $servername = "localhost";
7          $username = "root";
8          $password = "";
9          $dbname = "college_campus";
10         $conn = new mysqli($servername,$username,$password,$dbname);
11         // Check for database connection error
12         if (mysqli_connect_errno()) {
13             echo "Failed to connect to MySQL: " . mysqli_connect_error();
14             exit;
15         }
16         return $conn;
17     }
18 }
19 ?>
```

12.2 DBlibrary.php (Create object of DBconnect Class & Opening DB Connection)

```
1  <?php
2  class DBlibrary {
3      private $conn;
4      function __construct() {
5          include 'DBconnect.php';
6          // opening db connection
7          $db = new DBconnect();
8          $this->conn=$db->connect();
9      }
10 }
11 ?>
```

12.3 selectState()

(function which will provide the options in the state input of Registration Form)

```
49 public function selectState()  
50 {  
51     $stmt = "select state_name from state";  
52     $result=mysqli_query($this->conn,$stmt);  
53     return $result;  
54 }
```

12.4 registerUser()

```
57 public function  
registerUser($first_name,$middle_name,$last_name,$user_id,$password,$district,$state,$postal_code,$email,  
$phone,$dob,$gender,$hobby,$department,$designation,$start_year,$end_year,$img,$status,$otp){  
    $stmt="select user_id from user where user_id='$user_id'";  
58     $result=mysqli_query($this->conn,$stmt);  
59     $num_rows=mysqli_num_rows($result);  
60     if($num_rows!=0){  
61         return 2;  
62     }  
63     else{  
64         $password=md5($password);  
65         $stmt="insert into  
user(first_name,middle_name,last_name,user_id,password,gender,hobby,district,state,postal_code,ph  
one_number,email_address,dob,department,designation,start_year,end_year,profile_picture,status)va  
lues('$first_name','$middle_name','$last_name','$user_id','$password','$gender','$hobby','$distri  
ct','$state','$postal_code','$phone','$email','$dob','$department','$designation','$start_year','  
$end_year','$img','$status')";  
66         $result=mysqli_query($this->conn,$stmt);  
67         if($result){  
68             //delete old OTP  
69             $stmt="delete from otp where user_id='$user_id'";  
70             $result=mysqli_query($this->conn,$stmt);  
71             $stmt="insert into otp(user_id,code)values('$user_id','$otp')";  
72             $result=mysqli_query($this->conn,$stmt);  
73             if($result){ return 0; //success  
74                 }  
75             else  
76                 return 1; //error  
77         } } }  
78
```

12.5 loginUser()

```
78
79 public function loginUser($user_id,$password){
80     $password=md5($password);
81     $stmt="select * from user where user_id='$user_id' and password='$password'";
82     $result=mysqli_query($this->conn,$stmt);
83     $num_rows=mysqli_num_rows($result);
84     if($num_rows==0){
85         return 0; //Failure
86     }
87     else{
88         return 1; //Success
89     }
90 }
91
```

12.6 activate User()

(To make user status=1 after entering One Time Password at the time of User Registration)

```
91
92 public function activateUser($otp){
93     $stmt="select otp.user_id from user, otp where otp.code='$otp' and otp.user_id=user.user_id";
94     $result=mysqli_query($this->conn,$stmt);
95     $row=mysqli_fetch_array($result);
96     $num_rows=mysqli_num_rows($result);
97     if ($num_rows!=0)
98     {
99         $user_id=$row['user_id'];
100         $stmt="update user set status='active' where user_id='$user_id'";
101         $result=mysqli_query($this->conn,$stmt);
102         return 1;
103     }
104     else{
105         return 0;
106     }
107 }
108
```


12.7 resend Otp ()

```
108
109 ▼ public function resendOtp($user_id){
110     $stmt="select otp.code from user, otp where user.user_id = otp.user_id and user.user_id =
        ' $user_id'";
111     $result=mysqli_query($this->conn,$stmt);
112     $row=mysqli_fetch_array($result);
113     $otp=$row['code'];
114     return $otp;
115 }
```

12.8 get Post()

(Show post(s) of all user(s) on the Home Page)

```
123
124 ▼ public function getPost(){
125     $stmt="select user.user_id, user.first_name, user.middle_name, user.last_name, user.designation,
        user.profile_picture, post.post_id, post.post_desc, post.attachment_photo, post.attachment_video,
        post.attachment_file, post.post_dept, post.post_date_time from user, post where user.user_id=post.user_id
        order by post_date_time desc";
126     $row_post=mysqli_query($this->conn,$stmt);
127     return $row_post;
128 }
129
```

12.9 my Post()

(Show only the post(s) of logged In user on the Profile Page)

```
129
130 ▼ public function myPost($user_id){
131     $stmt="select user.user_id, user.first_name, user.middle_name, user.last_name, user.designation,
        user.profile_picture, post.post_id, post.post_desc, post.attachment_photo, post.attachment_video,
        post.attachment_file, post.post_dept, post.post_date_time from user, post where
        user.user_id=post.user_id and user.user_id=' $user_id' order by post_date_time desc";
132     $row_mypost=mysqli_query($this->conn,$stmt);
133     return $row_mypost;
134 }
135
```

12.10 comment() (Show Comment(s) on the Post(s))

```
154
155 ▼ public function comment($post_id){
156     $stmt="SELECT comment_user_id, first_name, middle_name, last_name, comment_desc, cmnt_date_time FROM
        comment WHERE post_id='$post_id' ORDER BY cmnt_date_time desc";
157     $cmnt=mysqli_query($this->conn,$stmt);
158     return $cmnt;
159 }
160
```

12.11 count Comment() (Count the Number of total Comment(s) of each Post(s) & Show beside of the Comment Button)

```
160
161 ▼ public function countComment($post_id){
162     $stmt="SELECT COUNT(*) AS count_comment FROM comment WHERE post_id='$post_id'";
163     $result=mysqli_query($this->conn, $stmt);
164     $count=mysqli_fetch_array($result);
165     return $count;
166 }
167
```

12.12 post Upload() (Upload a new Post(may be text, photo, video or document file))

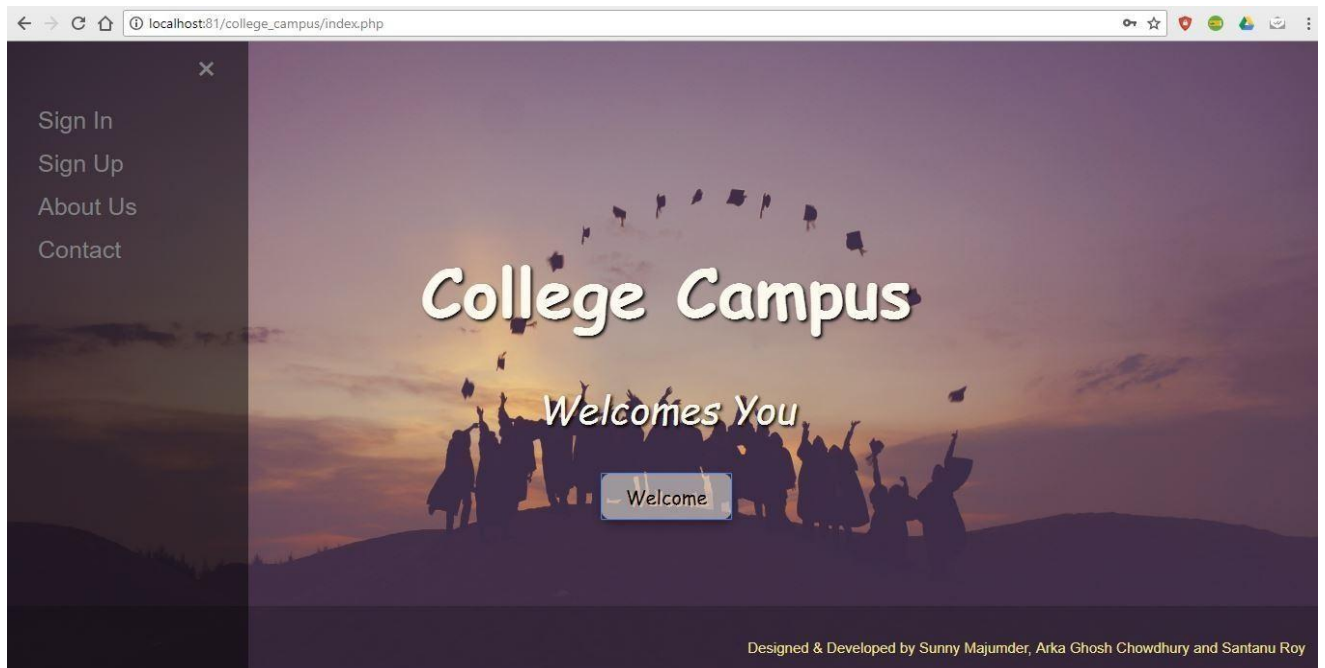
```
184
185 ▼ public function postUpload($user_id, $text_post, $img, $vid, $attch, $deptfilter, $status){
186     $stmt="insert into post(user_id, post_desc, attachment_photo, attachment_video, attachment_file,
        post_dept, status)values('$user_id','$text_post','$img','$vid','$attch','$deptfilter','$status')";
187     $result=mysqli_query($this->conn, $stmt);
188     return $result;
189 }
190
```

12.13 delete Post() (Delete own post of logged In User)

```
190
191 ▼ public function deletePost($post_id){
192     $stmt="delete from post where post_id='$post_id'";
193     $delpost=mysqli_query($this->conn, $stmt);
194     return $delpost;
195 }
196
```


13 SCREENSHOTS OF APPLICATION

13.1 Welcome Page (index. php)



13.2 Registration Interface (registration.php)

CCampus About Us Contact Us Help Sign In

Registration Form

First Name * **Middle Name** **Last Name ***

First Name Middle Name Last Name

Department / Section * **Designation ***

Select Department / Section Select Designation

Starting Year (Only for Student) **Ending Year (Only for Student)**

Select Starting Year Select Ending Year

Student Roll No / Employee Id * **New Password *** **Confirm Password ***

It will be your user id Min 8 character Re-enter new password

☐ Show Password ☐ Show Password

Email Address * **Phone Number ***

Enter Active Email Address +91 10-Digit Phone Number

District **State** **Postal Code**

Select District Select State ZIP Code

Date of Birth **Gender** **Hobby(s)**

mm/dd/yyyy I am... multiple-separated by ,

Select Profile Photo to Upload

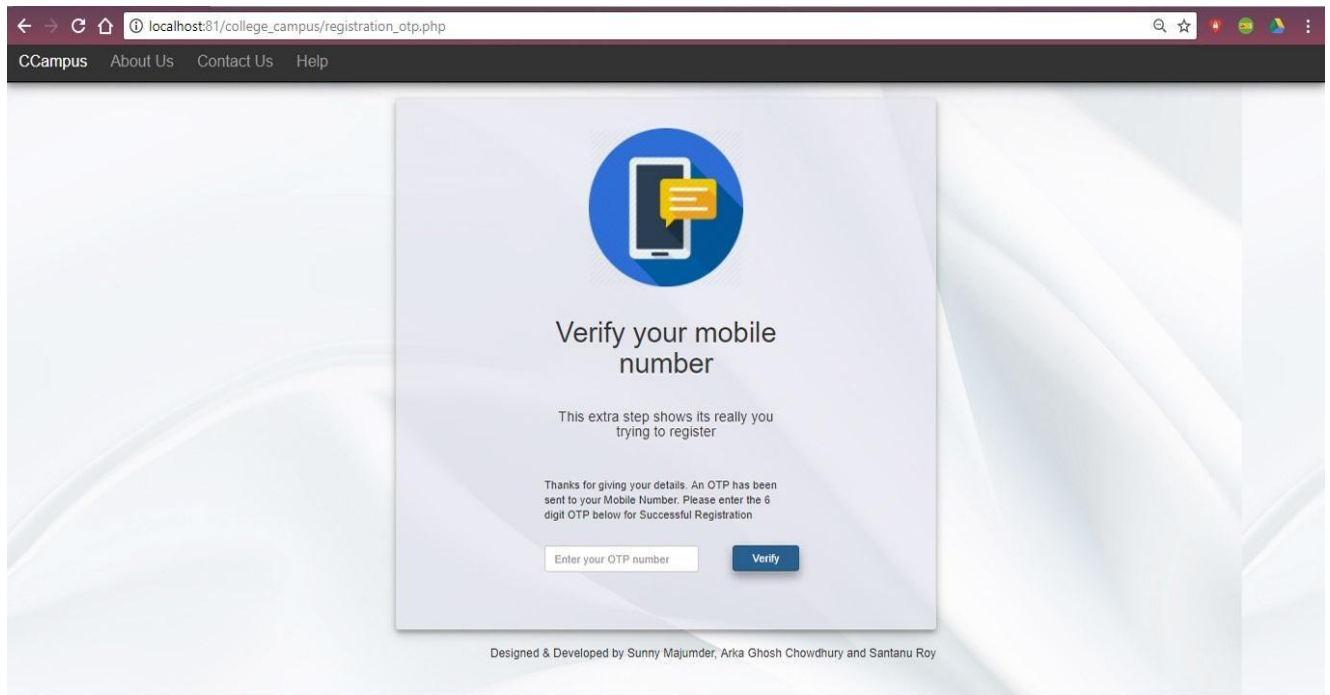
Choose File No file chosen

(*) indicates mandatory fields

Submit Clear

Designed & Developed by Sunny Majumder, Arka Ghosh Chowdhury and Santanu Roy


13.3 OTP Verification Page (registration_otp.php)



The screenshot shows a web browser window with the address bar displaying `localhost:81/college_campus/registration_otp.php`. The browser's navigation bar includes links for **CCampus**, **About Us**, **Contact Us**, and **Help**. The main content area features a light blue background with a central white card. At the top of the card is a circular icon containing a smartphone and a speech bubble. Below the icon, the heading **Verify your mobile number** is displayed. Underneath the heading, a message states: "This extra step shows its really you trying to register". Further down, a thank-you message reads: "Thanks for giving your details. An OTP has been sent to your Mobile Number. Please enter the 6 digit OTP below for Successful Registration". At the bottom of the card, there is a text input field labeled "Enter your OTP number" and a blue button labeled "Verify". The footer of the page, located below the card, credits the design and development to Sunny Majumder, Arka Ghosh Chowdhury, and Santanu Roy.

localhost:81/college_campus/registration_otp.php

CCampus About Us Contact Us Help



Verify your mobile number

This extra step shows its really you trying to register

Thanks for giving your details. An OTP has been sent to your Mobile Number. Please enter the 6 digit OTP below for Successful Registration

Enter your OTP number

Verify

Designed & Developed by Sunny Majumder, Arka Ghosh Chowdhury and Santanu Roy

14 AIM & OBJECTIVES

- ❖ The very first thing keeping in our mind we are developing this project is to gather all the college members, starting from administrative level to the basic clerical section into a single frame.
- ❖ Our aim is to convey all the notice(s), event(s), corrigendum(s) etc. and moreover what is going around the college.
- ❖ To grow the teacher and student relationship stronger.
- ❖ Last but not the least 24 X 7 connectivity with college campus.

15 FUTURE SCOPE OF WORK

The tremendous popularity of using the social media networking could never have been realized before. In fact, social media have become an important tool of marketing in true sense of customer orientation. But this kind of social networking site which can access only for college campus not from outside the college will make a dramatic changes inside the college campus.

- ❖ The scope of Intra-College Communication System is widening and today it offers a strong support to the college campus in providing the much desired touch of concern.
- ❖ The bright future prospect of college networking is also proven with the fact that the technology is integrated in mobile phones as well.
- ❖ The project is developing with the power of interpersonal communication on a globalized outlook.

16 CONCLUSION

Our project is only a humble venture to satisfy the needs to manage their project work. Several user friendly coding have also adopted. This package shall prove to be a powerful package in satisfying all the requirements. The objective of software planning is to provide a frame work that enables the manager to make reasonable estimates made within a limited time frame at the beginning of the software project and should be updated regularly as the project progresses.

17 BIBLIOGRAPHY

- ✓ www.wikipedia.com
- ✓ www.w3school.com
- ✓ www.stackoverflow.com
- ✓ www.tutorialspoint.com
- ✓ www.javapoint.com
- ✓ www.youtube.com
- ✓ <http://php.net/manual/en/function.ucfirst.php>
- ✓ https://www.w3schools.com/howto/howto_google_fonts.asp
- ✓ <https://www.iconfinder.com/>