FLICKCOM

Project Report Submitted By

ALEENA JOSEPH

Reg. No: AJC20MCA-2008

In Partial fulfillment for the Award of the Degree Of

MASTER OF COMPUTER APPLICATIONS (2 Year)
(MCA)
APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

AMAL JYOTHI COLLEGE OF ENGINEERING
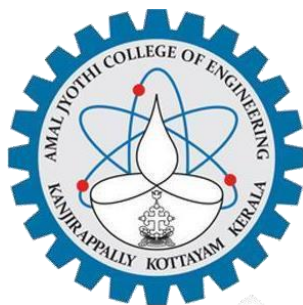## KANJIRAPPALLY

[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE, Accredited by NAAC with 'A' grade. Koovappally, Kanjirappally, Kottayam, Kerala –686518]

2021-2022

DEPARTMENT OF COMPUTER APPLICATIONS

AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY

CERTIFICATE

This is to certify that the Project report, **"FLICKCOM"** is the bonafide work of **ALEENA JOSEPH (Reg.No:AJC20MCA-2008)** in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications under APJ Abdul Kalam Technological University during the year 2021-2022.

Ms. Jetty Benjamin   **Rev. Fr. Dr. Rubin Thottupurathu Jose**
Internal Guide   **Coordinator**

Rev. Fr. Dr. Rubin Thottupurathu Jose

Head of the Department

DECLARATION

I hereby declare that the project report **"FLICKCOM"** is a bonafid work done at AmalJyothi College of Engineering, towards the partial fulfilment of the requirements for the award of the Degree of Master of Computer Applications (MCA) from APJ Abdul Kalam Technological University, during the academic year 2021-2022.

Date:01/03/2022   **ALEENA JOSEPH**

KANJIRAPPALLY   **Reg.No:AJC20MCA-2008**

ACKNOWLEDGEMENT

First and foremost, I thank God almighty for his eternal love and protection throughout the project. I take this opportunity to express my gratitude to all who helped me in completing this project successfully. It has been said that gratitude is the memory of the heart. I wish to express my sincere gratitude to our manager **Rev. Fr. Dr. Mathew Paikatt** and Principal **Dr. Lillykutty Jacob** for providing good faculty for guidance.

I owe a great depth of gratitude towards our Head of the Department **Rev. Fr. Dr. Rubin Thottupurathu Jose** for helping us. I extend my whole hearted thanks to the project coordinators **Rev. Fr. Dr. Rubin Thottupurathu Jose** for his valuable suggestions and for overwhelming concern and guidance from the beginning to the end of the project. I would also like to express sincere gratitude to my guide**, Ms. Jetty Benjamin** for her inspiration and helping hand.

I thank our beloved teachers for their cooperation and suggestions that helped me throughout the project. I express my thanks to all my friends and classmates for their interest, dedication, and encouragement shown towards the project. I convey my hearty thanks to my family for the moral support, suggestions, and encouragement to make this venture a success.

<div align="right">HARITHA KRISHNAN</div>

ABSTRACT

FlickCom is a web based system for film industry,developed in php,which is very helpful for the freshers who are interested in film field.The person ,seeking opportunities in film industry can make their profile and can upload in this site.

The Staffs(Film Personalities)can select matching profiles and accept them.They can schedule auditions and share vacancies also.

Normal users can login and view posters,upcoming films and it's releasing details ect,. They can make predictions about upcoming films,whether that becomes super hit/hit/flop.They can post their reviews about released films.

CONTENT

List of Abbreviation

IDE	-	Integrated Development Environment
PHP	-	Hyper Text Markup Language.
CSS	-	Cascading Style Sheet
SQL	-	Structured Query Language
UML	-	Unified Modeling Language

CHAPTER 1

INTRODUCTION

1. PROJECT OVERVIEW

FlickCom is a web based system for film industry,developed in php,which is very helpful for the freshers who are interested in film field.The person ,seeking opportunities in film industry can make their profile and can upload in this site.

The Staff(Film Personalities)can select matching profiles and accept them.They can schedule auditions and share vacancies also.

Normal users can login and view posters,upcoming films and it's releasing details ect,. They can make predictions about upcoming films,whether that becomes super hit/hit/flop.They can post their reviews about released films.

1.1 PROJECT SPECIFICATION

The proposed system is a website in which user can get reviews about the ongoing movies and they can predict the success of upcoming movies. It also helps the freshers to find vacancies in film industry.

The system mainly includes 4 modules. They are:

1. ADMIN:

- Login
- Manage featured members and normal members
- View reviews
- View predictios
- Add Triallers
- Add posters and Releasing details

2. NORMAL USERS:

- Sign Up
- View movie trailers, posters and releasing details
- Make predictions
- Post reviews

3. FRESHERS:

- Register and login
- Create profile
- View selection Notifications

4. STAFF:

- Login
- Add vacancies
- Schedule auditions
- View fresher's profile
- Select profiles

In this mini project mainly focusing only two modules, Admin module and Staff module. Remaining modules will do for the future scope.

CHAPTER 2

SYSTEM STUDY

2.1.INTRODUCTION

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minute's detail and analyzed. The system analyst plays the role of the interrogator and wells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system issue objected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal.

Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies, a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.

2.1 EXISTING SYSTEM

Existing system is not a fully automated system. Freshers are directly reached to the office and ask for the chance. And they need to search on multiple sites for get the auditions details

It is take more time to get information for their appropriate needs.

2.2 DRAWBACKS OF EXISTING SYSTEM

- Human effort is needed.
- It is difficult to maintain important information in user.
- More manual hours need to process application.

2.3 PROPOSED SYSTEM

The proposed system is defined to meets all the disadvantages of the existing system. It is necessary to have a system that is more user friendly and user attractive for those who are love film. The issues of existing system can be overcome by the proposed system .The users can get posters ,trailers, and the releasing date of the upcoming movies.They can also get reviews about the released movies and can make predictions about the upcoming movies. The freshers those who are seeking opportunities in film industry, they can create profiles and upload resumes.They can get notifications abut vacancies and auditions. The staffs can select the profiles also.

2.4 ADVANTAGES OF PROPOSED SYSTEM

The system is very simple in design and to implement. The system requires very low system resources, and the system will work in almost all configurations. It has got following features:

➢ **Better security:-**

For data to remain secure measures must be taken to prevent unauthorized access. Security means that data are protected from various forms of destruction. The system security problem can be divided into four related issues: security, integrity, privacy and confidentiality. Username and password requirement to sign in ensures security. It will also provide data security as we are using the secured databases for maintaining the documents.

➢ **Ensure data accuracy:-**

The proposed system eliminates the manual errors while entering the details of the users during the registration.

➢ **Better service: -**

The product will avoid the burden of hard copy storage. We can also conserve the time and human resources for doing the same task. The data can be maintained for longer period with no loss of data.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 **FEASIBILITY STUDY**

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its work ability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus, when a new application is proposed it normally goes through a feasibility study before it is approved for development.

The document provides the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Technical, Economic and Operational feasibilities. The following are its features:-

3.1.1 Economical Feasibility

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

> ➢ The costs conduct a full system investigation.
> ➢ The cost of the hardware and software.
> ➢ The benefits in the form of reduced costs or fewer costly errors.

The proposed system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication of the system is economically possible for development.

The cost of project, DREAMS was divided according to the system used, its development cost and cost for hosting the project. According to all the calculations the project was developed in a low cost. As it is completely developed using open source software.

3.1.2 Technical Feasibility

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment ,required method developing the system, of running the system once it has been designed.

Technical issues raised during the investigation are:

➢ Does the existing technology sufficient for the suggest done?
➢ Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project requires High Resolution Scanning device and utilizes Cryptographic techniques. Through the technology may become obsolete after some period of time, due to the fact that newer version of same software supports older versions, the system may still be used. So there are minimal constraints involved with this project. The system has been developed using PHP in front end and MySQL in server in back end, the project is technically feasible for development. The system has been developed using PHP in front end and MySQL in server in back end, the project is technically feasible for development. The System used was also of good performance of Processor Intel Pentium Gold; RAM 4GB and, Hard disk 500 GB

3.1.3 Behavioral Feasibility

The proposed system includes the following questions:

➢ Is there sufficient support for the users?
➢ Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

3.2 SYSTEM SPECIFICATION

3.1.4 Hardware Specification

Processor   -   Intel core i3

RAM   - 4 GB

Hard disk   - I TB

3.1.5 Software Specification

Front End   -   HTML, CSS

Back end   -   MYSQL

Client on PC   -   Windows 7 and above.

Technologies used -   JS, J Query, PHP, CSS, HTML

3.2 SOFTWARE DESCRIPTION

3.2.1 PHP

PHP is a server side scripting language designed for web development but also used as a general purpose programming language.PHP is now installed on more than 244 million websites and2.1million web servers. Originally created by RasmusLedorfin1995, the reference implementation of PHP is now produced by the PHP group. While PHP originally stood for personal Home page ,it now stands for PHP: Hypertext Preprocessor, a recursive acronym.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 a 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 stand alone incompatible with the GNU General Public License (GPL) due to restrictions on the usage of the term PHP.PHP can be deployed on most web servers and also as a stand alone shell on almost every operating system and platform, free of charge.

3.3.2 MySQL

MySQL, the most popular Open Source SQL database management system, is developed, distributed ,and supported by Oracle Corporation. The MySQL Website provides the latest information about MySQL software.

MySQL is a database management system.

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

*   **MySQL databases are relational.**

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You setup rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and "pointers "between different tables. The database enforces these rules, so that with a well-designed database, your application never sees in consistent, duplicate, orphan, out-of-date, or missing data. The SQL part of "MySQL" stands for "Structured Query Language". SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQLsyntax. SQL is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versionsexist.Inthismanual,"SQL92"referstothestandardreleasedin1992,

"SQL: 1999"refers to the standard released in1999,and"SQL:2003"referstothe current version of the standard. We use the phrase "the SQL standard "to mean the current version of the SQL Standard at anytime.

- **MySQL software is Open Source.**

    Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), to define what you may and may not down the software in different situations. If you feel uncomfortable with the GPL or need to embed MySQL code into a commercial application, you can buy a commercially licensed version from us. See the MySQL Licensing Overview for more information.

- **The MySQL Database Server is very fast, reliable, scalable, and easy to use.**

    If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, along side your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available.

- **MySQL Server works in client/server or embedded systems.**

    The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs). We also provide MySQL Server as an embedded multi-threaded library that you can link into your application to get a smaller, faster, easier-to-manage stand alone product.

CHAPTER 4

SYSTEM DESIGN

4.1 INTRODUCTION

Design is the first step into the development phase for any engineered product or system. Design is a creative process. A good design is the key to effective system. The term "design" is defined as "the process of applying various techniques and principles for the purpose of defining a process or a system in sufficient detail to permit its physical realization ". It may be defined as a process of applying various techniques and principles for the purpose of defining a device, a processor a system insufficient detail to permit its physical realization. Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm that is used. The system design develops the architectural detail required to build a system or product. As in the case of any systematic approach, this software too has undergone the best possible design phase fine tuning all efficiency, performance and accuracy levels. The design phase is a transition from a user oriented document to a document to the programmers or database personnel. System design goes through two phases of development: Logical and Physical design.

4.2 UML DIAGRAM

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. UML was created by the Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997.

UML stands for **Unified Modeling Language**. UML is different from the other common programming languages such as C++, Java, COBOL, etc. UML is a pictorial language used to make software blueprints. UML can be described as a general purpose visual modeling language to visualize, specify, construct, and document software system. Although UML is generally used to model software systems, it is not limited within this boundary. It is also used to model non-software systems as well. For example, the process flow in a manufacturing unit, etc. UML is not a programming language but tools can be used to generate code in various languages using UML diagrams. UML has a direct relation with object oriented analysis and design. After some standardization, UML has become an OMG standard. All the elements, relationships are used to make a

complete UML diagram and the diagram represents a system. The visual effect of the UML diagram is the most important part of the entire process. All the other elements are used to make it complete. UML includes the following nine diagrams.

- Class diagram
- Object diagram
- Use case diagram
- Sequence diagram
- Collaboration diagram
- Activity diagram
- State chart diagram
- Deployment diagram
- Component diagram

4.2.1 USE CASE DIAGRAM

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system "refers to something being developed or operated, such as a mail-order product sales and service Web site. Use case diagrams are employed in UML (Unified Modeling Language), a standard notation for modeling of real-world objects and systems.
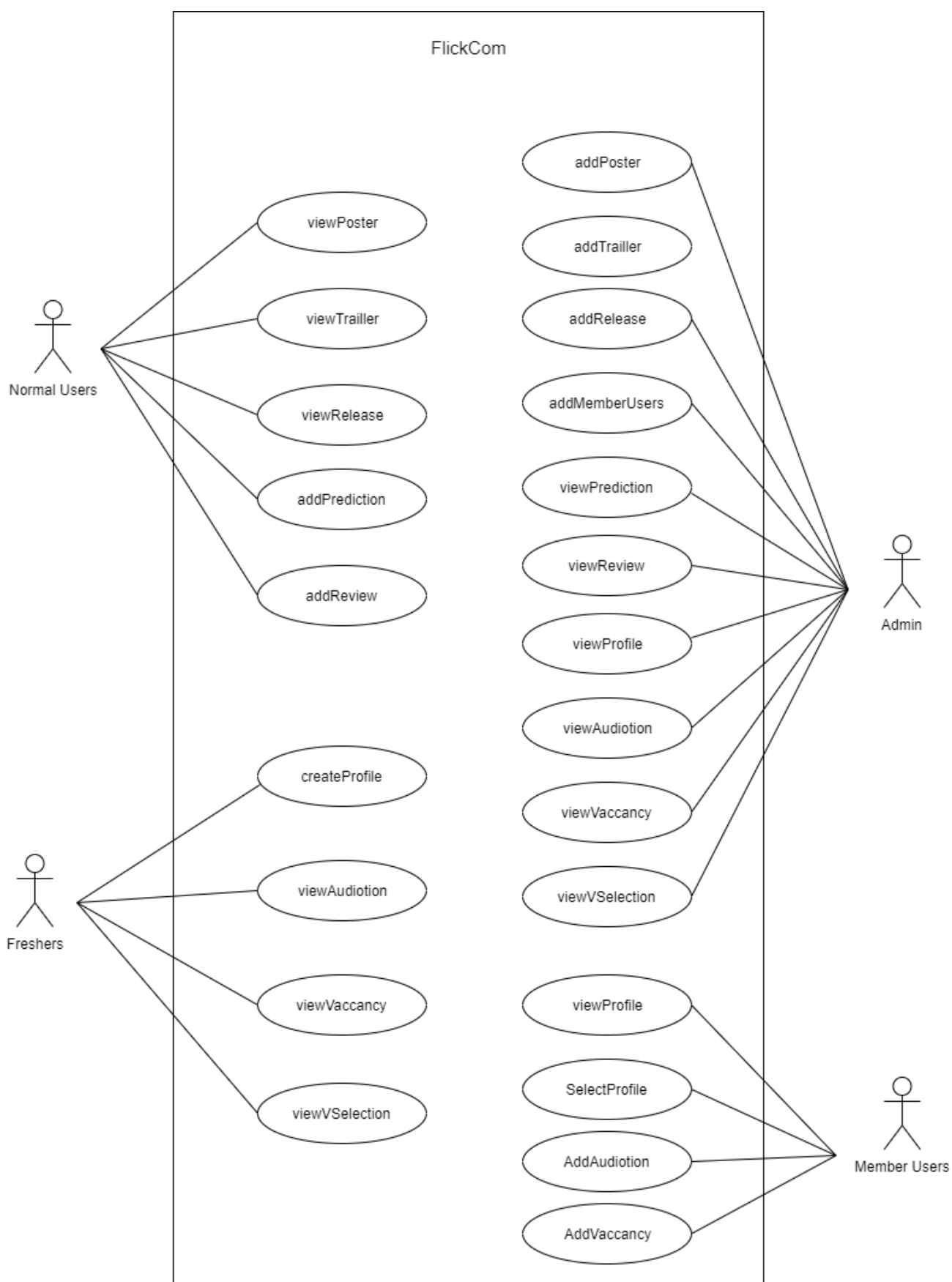
System objectives can include planning overall requirements, validating a hardware design, testing and debugging a software product under development, creating an online help reference, or performing a consumer-service-oriented task. For example, use cases in a product sales environment would include item ordering, catalog updating, payment processing, and customer relations. A use case diagram contains four components.

- The boundary, which defines the system of interest in relation to the world around it.
- The actors, usually individuals involved with the system defined according to their roles.

- The use cases, which are the specific roles are played by the actors within and around the system.

- The relationships between and among the actors and the use cases.

Use case diagrams are drawn to capture the functional requirements of a system. After identifying the above items, we have to use the following guidelines to draw an efficient use case diagram.

- The name of a use case is very important. The name should be chosen in such a way so that it can identify the functionalities performed.

- Give a suitable name for actors.

- Show relationships and dependencies clearly in the diagram.

- Do not try to include all types of relationships, as the main purpose of the diagram is to identify the requirements.

- Use notes whenever required to clarify some important points.

FlickCom

viewPoster

viewTrailler

viewRelease

addPrediction

addReview

Normal Users

addPoster

addTrailler

addRelease

addMemberUsers

viewPrediction

viewReview

viewProfile

viewAudiotion

viewVaccancy

viewVSelection

Admin

createProfile

viewAudiotion

viewVaccancy

viewVSelection

Freshers

viewProfile

SelectProfile

AddAudiotion

AddVaccancy

Member Users

4.2.2 SEQUENCE DIAGRAM

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems.

Sequence Diagram Notations –

- **Actors –** An actor in a UML diagram represents a type of role where it interacts with the system and its objects. It is important to note here that an actor is always outside the scope of the system we aim to model using the UML diagram. We use actors to depict various roles including human users and other external subjects. We represent an actor in a UML diagram using a stick person notation. We can have multiple actors in a sequence diagram.

- **Lifelines –** A lifeline is a named element which depicts an individual participant in a sequence diagram. So basically each instance in a sequence diagram is represented by a lifeline. Lifeline elements are located at the top in a sequence diagram

- **Messages –** Communication between objects is depicted using messages. The messages appear in a sequential order on the lifeline. We represent messages using arrows. Lifelines and messages form the core of a sequence diagram.

Messages can be broadly classified into the following categories:

- Synchronous messages

- Asynchronous Messages

- Create message

- Delete Message

- Self-Message

- Reply Message

- Found Message

- Lost Message

- **Guards –** To model conditions we use guards in UML. They are used when we need to restrict the flow of messages on the pretext of a condition being met. Guards play an important role in letting software developers know the constraints attached to a system or a particular process.

Uses of sequence diagrams –

- Used to model and visualize the logic behind a sophisticated function, operation or procedure.
- They are also used to show details of UML use case diagrams.
- Used to understand the detailed functionality of current or future systems.
- Visualize how messages and tasks move between objects or components.
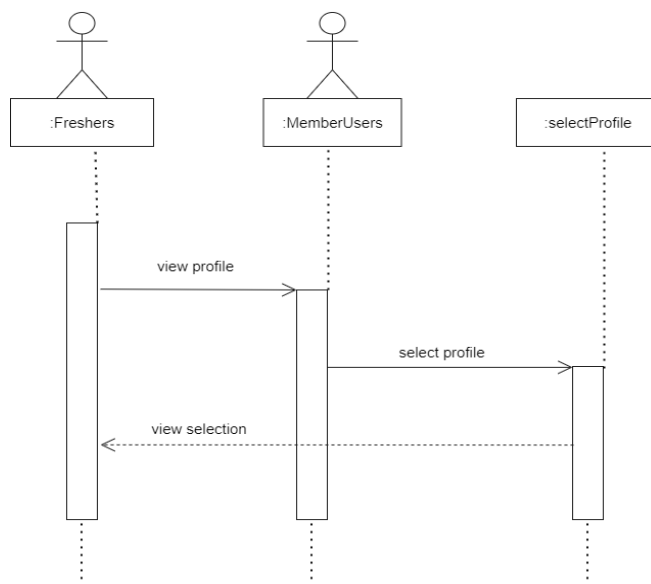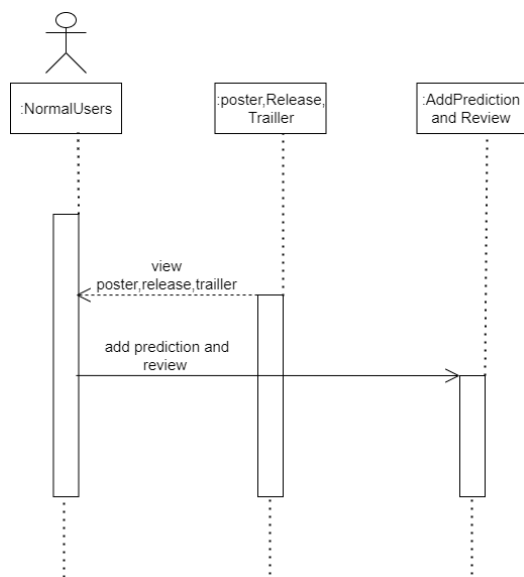
State Chart Diagram

State Diagram are used to capture the behavior of a software system. UML State machine diagrams can be used to model the behavior of a class, a subsystem, a package, or even an entire system. It is also called a State chart or State Transition diagram. State chart diagrams provide us an efficient way to model the interactions or communication that occur within the external entities and a system. These diagrams are used to model the event-based system. A state of an object is controlled with the help of an event. State chart diagrams are used to describe various states of an entity within the application system.
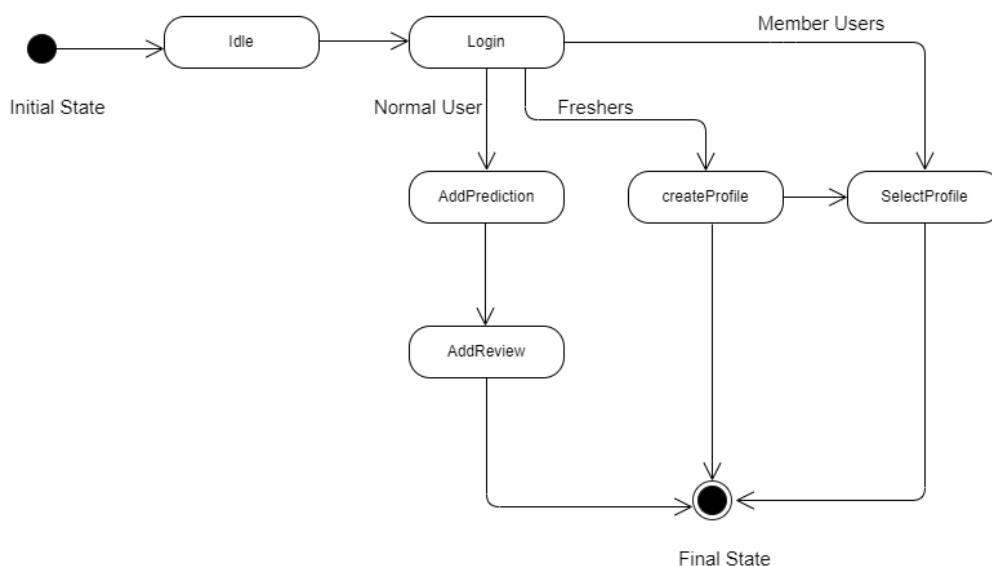
Activity Diagram

Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. Typically, an event needs to be achieved by some operations, particularly where the operation is intended to achieve a number of different things that require coordination, or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination. It is also suitable for
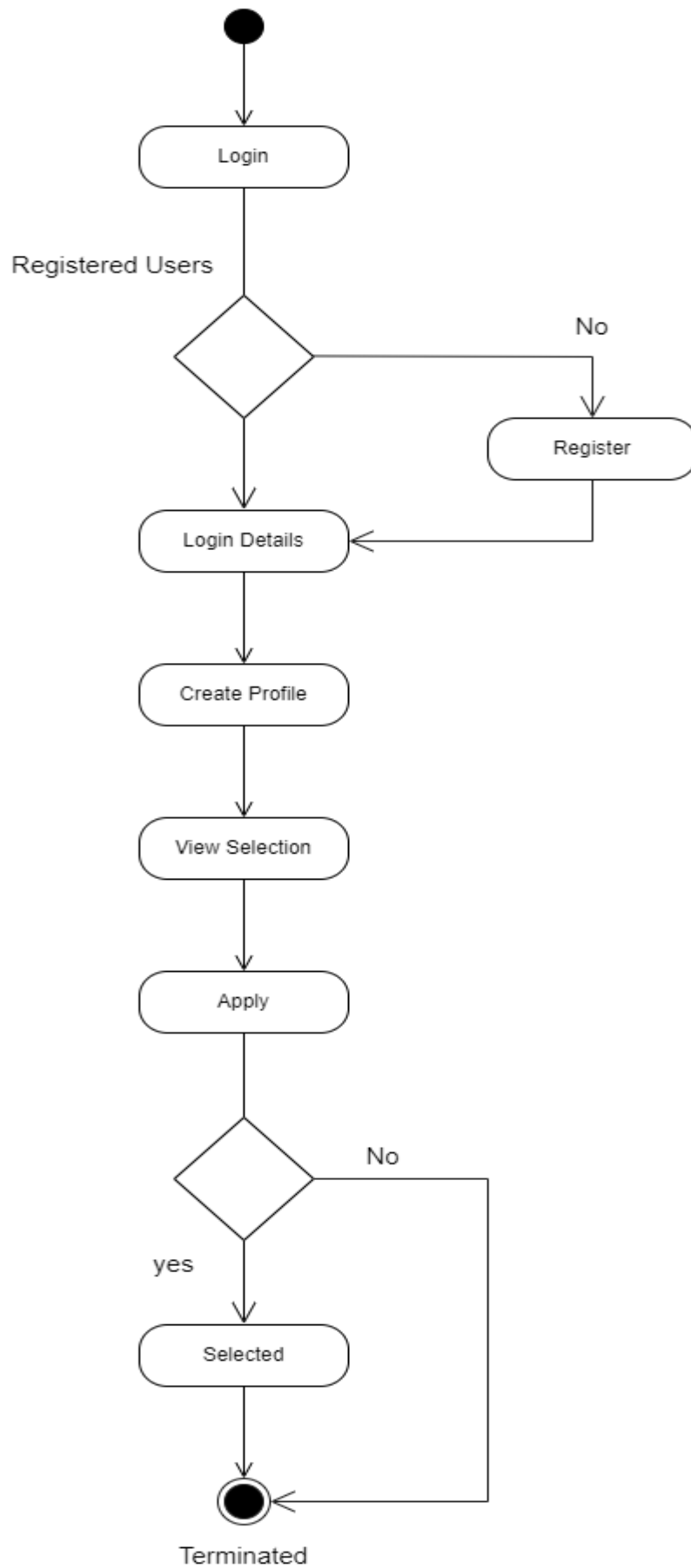
Class Diagram

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is alsoknown as a structural diagram.
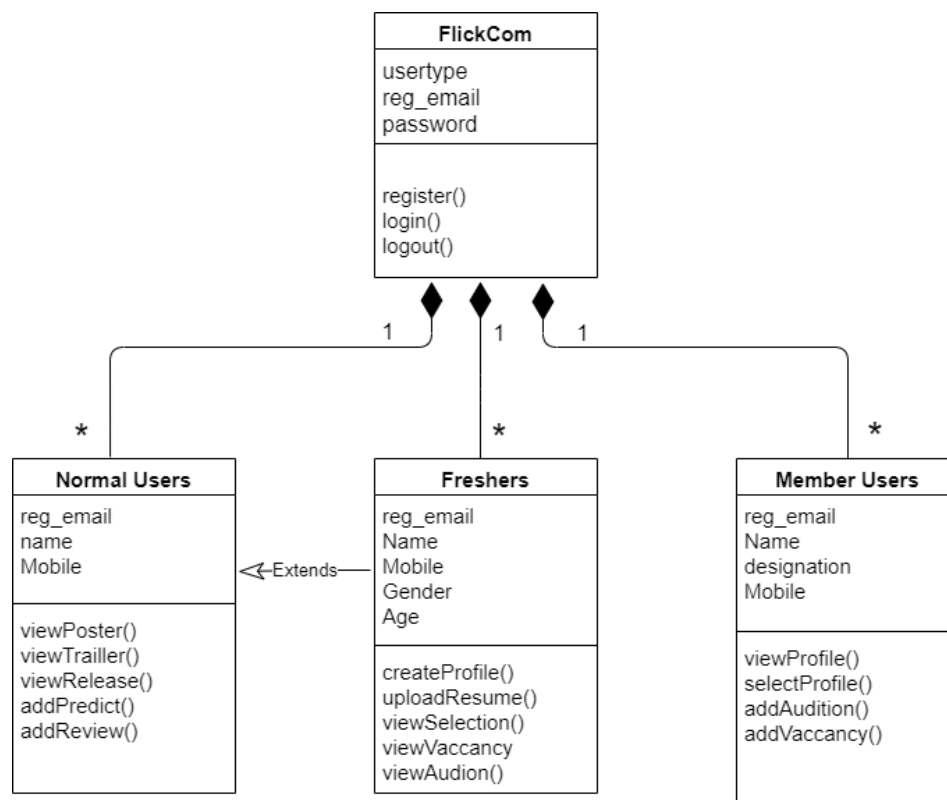
Object Diagram

Object diagrams are derived from class diagrams so object diagrams are dependent upon class diagrams. Object diagrams represent an instance of a class diagram. The basic concepts are similar for class diagrams and object diagrams. Object diagrams also represent the static view of a system but this static view is a snapshot of the system at a particular moment. Object diagrams are used to render a set of objects and their relationships as an instance.
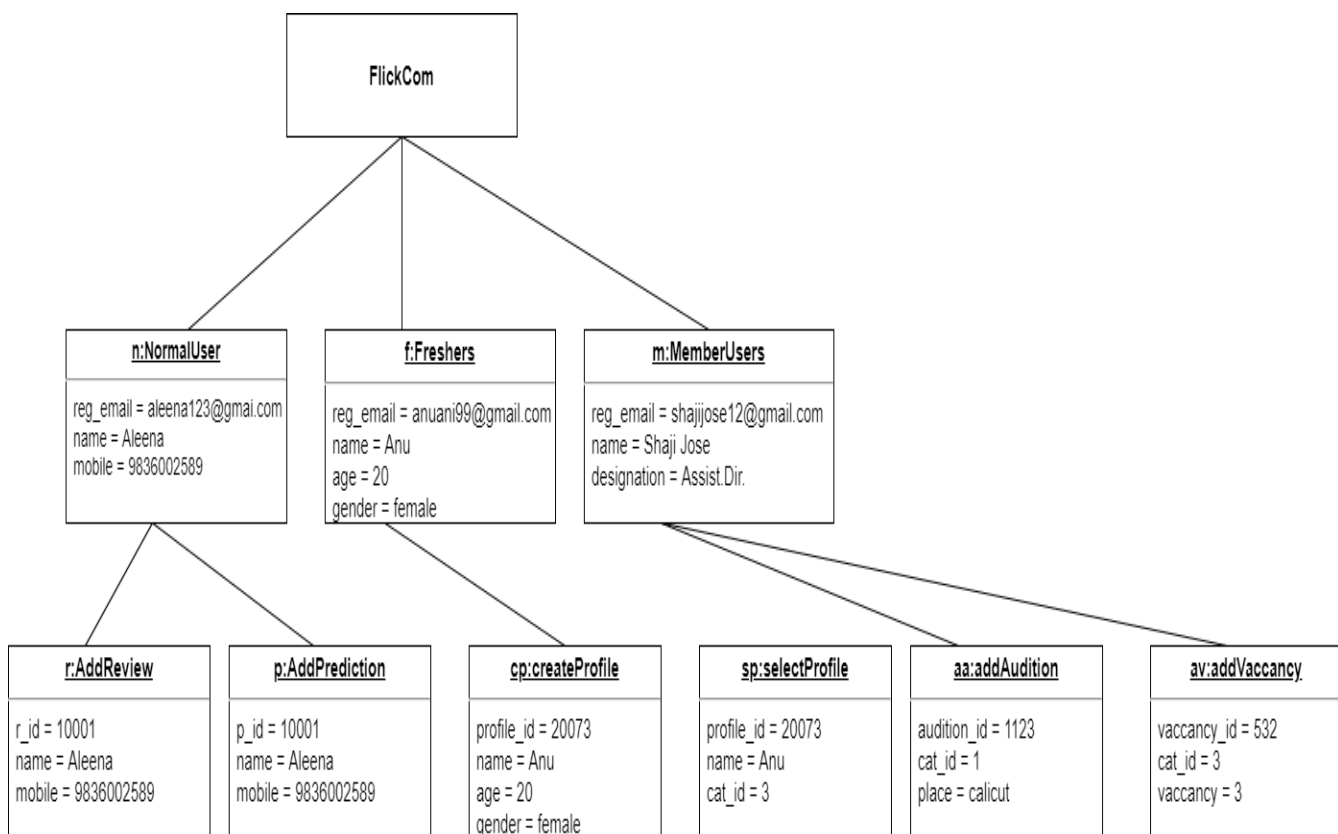
Component Diagram

Component diagrams are different in terms of nature and behaviour. Component diagrams are used to model the physical aspects of a system. Physical aspects are the elements such as executables, libraries, files, documents, etc. which reside in a node. Component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.
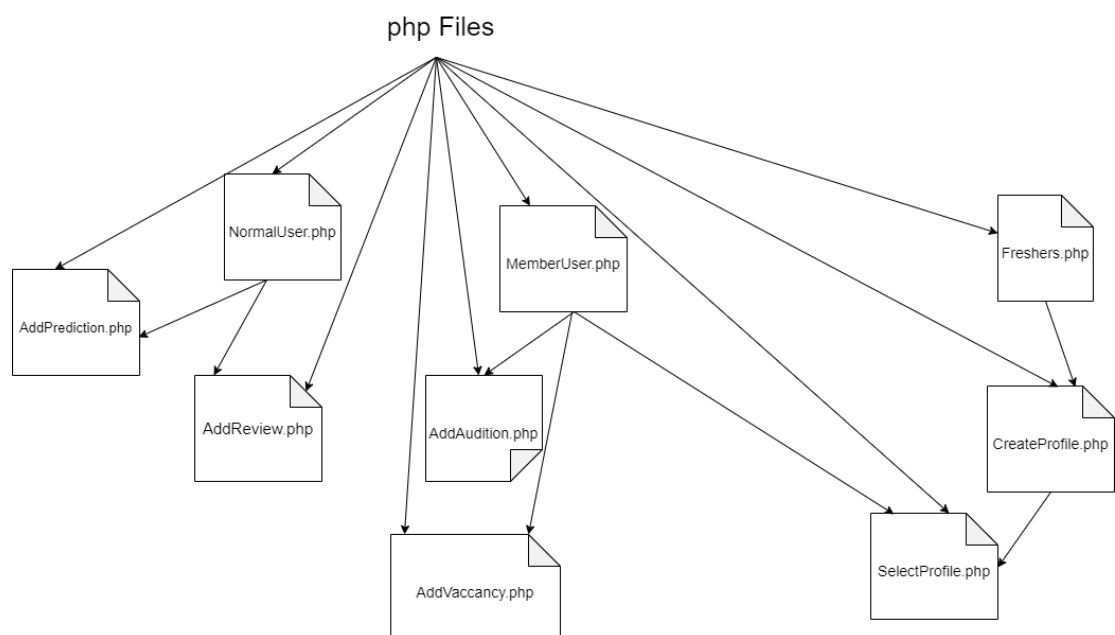
Deployment Diagram
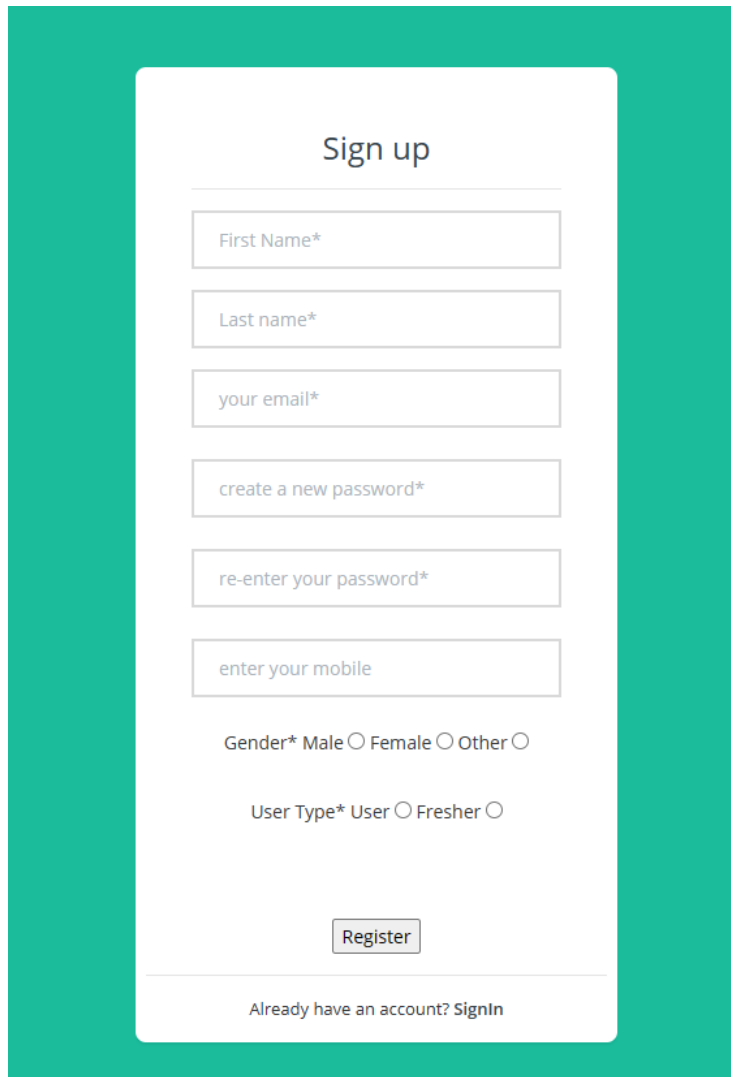
Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.
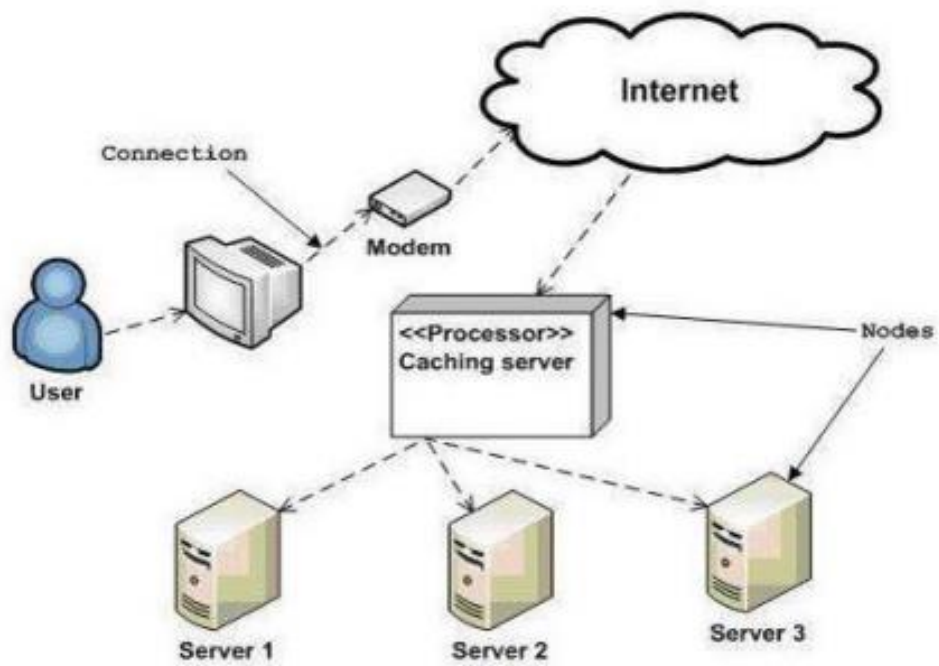
4.5 USER INTERFACE DESIGN

4.5.1-INPUT DESIGN

Form Name   : User Registration

Form Name   : User Login

Form Name   : Add Staff

Form Name   : Add Vaccancy

Form Name   : Schedule Audition

4.5.2 OUTPUT DESIGN

User Login

Staff Details

Staffs Record

FIRST NAME	LAST NAME	EMAIL	DESIGNATION	PASSWORD		
avani	hsagccc	23	shacn	pc	Delete	Edit
sdgds	sdg	sdfghj@gmail.com	Aleena12	ytjy	Delete	Edit
4w3	4	wfr@gmail.com	fAssss222	fetttwww	Delete	Edit
dxr	gfrssg	fbdh@gmail.com	Addd2sdcv	sdgsx	Delete	Edit
Alan	Jude	alanjude@gmail.com	alanjude5	dr	Delete	Edit

4.6 DATABASE DESIGN

A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is a two level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS.

In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

- Data Integrity
- Data independence

4.6.1 Relational Database Management System(RDBMS)

A relational model represents the database as a collection of relations. Each relation resembles a table of values or file of records. In formal relational model terminology, a row is called a tuple , a column header is called an attribute and the table is called a relation. A relational data base consists of a collection of tables, each of which is assigned a unique name. A row in a tale represents a set of related values.

Relations, Domains & Attributes

A table is a relation. The rows in a table are called tuples . A tuple is an ordered set of n elements. Columns are referred to as attributes. Relationships have been set between every table in the database. This ensures both Referential and Entity Relationship Integrity. A domain D is a set of atomic values. A common method of specifying a domain is to specify a data type from which the data values forming the domain are drawn. It is also useful to specify a name for the domain to help in interpreting its values. Every value in a relation is atomic, that is not decomposable.

Relationships

- Table relationships are established using Key. The two main keys of prime importance are Primary Key & Foreign Key. Entity Integrity and Referential Integrity Relationships can be established with these keys.

- Entity Integrity enforces that no Primary Key can have null values.

- Referential Integrity enforces that no Primary Key can have null values.

- Referential Integrity for each distinct Foreign Key value, there must exist a matching Primary Key value in the same domain. Other key are Super Key and Candidate Keys.

4.6.2 Normalization

Data are grouped together in the simplest way so that later changes can be made with minimum impact on data structures. Normalization is formal process of data structures in manners that eliminates redundancy and promotes integrity. Normalization is a technique of separating redundant fields and breaking up a large table into a smaller one. It is also used to avoid insertion, deletion, and updating anomalies. Normal form in data modelling use two concepts, keys and relationships. A key uniquely identifies a row in a table. There are two types of keys, primary key and foreign key. A primary key is an element or a combination of elements in a table whose purpose is to identify records from the same table. A foreign key is a column in a table that uniquely identifies record from a different table. All the tables have been normalized up to the third normal form.

As the name implies, it denotes putting things in the normal form. The application developer via normalization tries to achieve a sensible organization of data into proper tables and columns and where names can be easily correlated to the data by the user. Normalization eliminates repeating groups at data and thereby avoids data redundancy which proves to be a great burden on the computer resources. These include:

- ✓ Normalize the data.
- ✓ Choose proper names for the tables and columns.
- ✓ Choose the proper name for the data.

First Normal Form

The First Normal Form states that the domain of an attribute must include only atomic values and that the value of any attribute in a tuple must be a single value from the domain of that attribute. In other words 1NF disallows "relations within relations" or "relations as attribute values within tuples ". The only attribute values permitted by 1NF are single atomic or indivisible values. The first step is to put the data into First Normal Form. This can be donor by moving data into separate tables where the data is of similar type in each table. Each table is given a Primary Key or Foreign Key as per requirement of the project. In this we form new relations for each non-atomic attribute or nested relation. This eliminated repeating groups of data. A relation is said to be in first normal form if only if it satisfies the constraints that contain the primary key only.

Second Normal Form

According to Second Normal Form, for relations where primary key contains multiple attributes, no non-key attribute should be functionally dependent on a part of the primary key. In this we decompose and setup a new relation for each partial key with its dependent attributes. Make sure to keep a relation with the original primary key and any attributes that are fully functionally dependent on it. This step helps in taking out data that is only dependent on a part of the key. A relation is said to be in second normal form if and only if it satisfies all the first normal form conditions for the primary key and every non-primary key attributes of the relation is fully dependent on its primary key alone.

Third Normal Form

According to Third Normal Form, Relation should not have a non-key attribute functionally determined by another non-key attribute or by a set of non-key attributes. That is, there should be no transitive dependency on the primary key. In this we decompose and set up relation that includes the non-key attributes that functionally determines other non-key attributes. This step is taken to get rid of anything that does not depend entirely on the Primary Key. A relation is said to be in third normal form if only if it is in second normal form and more over the non key attributes of the relation should not be depend on other non-key attribute.

TABLE DESIGN

1. Tbl_UserType

Primary key: UserType_id
Foreign key:

Field Name	Data Type	Size	Description
UserType_id	Int	10	Primary key of Tbl_UserType Table
Type_Name	varchar	20	Store user type

2. Tbl_Registration

Primary key: reg_email
Foreign key: UserType_id

Field Name	Data Type	Size	Description
reg_email	varchar	50	Primary key
UserType_id	int	10	Foreign key of Tbl_UserType table
FName	varchar	25	Store first name
LName	varchar	25	Store last name
Mobile	varchar	15	Store mobile number

3. Tbl_Login

Primary key: login_id
Foreign key: reg_email, addMember_id

Field Name	Data Type	Size	Description
login_id	int	10	Primary key
reg_email	varchar	50	Foreign key of Tbl_Registration table
addMember_id	int	10	Foreign key of Tbl_AddMembers table
password	varchar	30	Store password

4. Tbl_FeildCat

Primary key: cat_id
Foreign key:

Field Name	Data Type	Size	Description
cat_id	Int	10	Primary key
Feild_Name	varchar	20	tore feid category name

5. Tbl_Vaccancy

Primary key: vacancy_id
Foreign key: cat_id

Field Name	Data Type	Size	Description
vaccancy_id	int	10	Primary key
cat_id	int	10	Foreign key of Tbl_FeidCat table
vaccancy	int	4	Number of vaccancies
condact	varchar	15	Store condact number

6. Tbl_Audition
   Primary key: audition_id
   Foreign key: cat_id

Field Name	Data Type	Size	Description
audition_id	int	10	Primary key
cat_id	int	10	Foreign key of Tbl_FeidCat table
place	varchar	25	Store place of audition
date	date		ore date of audition
time	timestamp		Store time of audition

7. Tbl_AddMembers
   Primary key: addMember_id
    Foreign key:

Field Name	Data Type	Size	Description
addMember_id	int	10	Primary key
FName	varchar	25	Store first name
LName	varchar	25	Store last name
email	varchar	25	Store email
designation	varchar	25	tore designation of member users
Mmbr_password	varchar	25	Store password of member users

8. Tbl_poster

Primary key: poster_id
Foreign key:

Field Name	Data Type	Size	Description
poster_id	int	10	Primary key
poster	varchar	200	Store posters
Release_date	date		Store release date
Release_type	varchar	50	Store release type

9. Tbl_trailler

Primary key: trailler_id
Foreign key:

Field Name	Data Type	Size	Description
trailler_id	int	10	Primary key
trailler	varchar	200	Store trailler

10. Tbl_FresherProfile

Primary key:profile_id
Foreign key: reg_email, cat_id

Field Name	Data Type	Size	Description
profile_id	int	10	Primary key
cat_id	int	10	Foreign key of Tbl_FeidCat table
reg_email	varchar	50	Foreign key of Tbl_Registration table
gender	varchar	15	Store gender
age	int	3	Store age
lternative_mobile	int	15	Store alternative mobile number
addressLine_1	varchar	50	Store house/building name
city	varchar	25	Store city name
state	varchar	25	Store sate
country	varchar	25	Store name of the country
pin	int	10	Store pincode
qualification	varchar	25	tore qualification
experience	varchar	25	Store experience
Profile_pic	varchar	200	Strore profile picture of freshers
resume	varchar	200	Store resume of freshers
Intro_video	varchar	500	tore introduction video

11. Tbl_prediction
Primary key: p_id
Foreign key: reg_email

Field Name	Data Type	Size	Description
p_id	Int	10	Primary key
reg_email	varchar	50	Foreign key of Tbl_Registration table
prediction	varchar	200	Store predictions

12.Tbl_review
Primary key: r_id
Foreign key: reg_email

Field Name	Data Type	Size	Description
r_id	Int	10	Primary key
reg_email	varchar	50	Foreign key of Tbl_Registration table
review	varchar	200	Store review

CHAPTER 5

SYSTEM TESTING

5.1 INTRODUCTION

Software Testing is the process of executing software in a controlled manner, in order to answer the question-Does the software behave as specified ? Software testing is often used in association with the terms verification and validation. Validation is the checking or testing of items, includes software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques such as reviews, analysis, inspections, and walkthroughs. Validation is the process of checking that what has been specified is what the user actually wanted.

Other activities which are often associated with software testing are static analysis and dynamic analysis. Static analysis investigates the source code of software, looking for problems and gathering metrics without actually executing the code. Dynamic analysis looks at the behavior of software while it is executing, to provide information such as execution traces, timing profiles, and test coverage information.

Testing is a set of activity that can be planned in advanced and conducted systematically. Testing begins at the module level and work towards the integration of entire computers based system. Nothing is complete without testing, as it vital success of the system testing objectives, there are several rules that can serve as testing objectives. They are:

Testing is a process of executing a program with the intent of finding an error.

- A good test case is one that has high possibility of finding an undiscovered error.
- A successful test is one that uncovers an undiscovered error.

If a testing is conducted successfully according to the objectives as stated above, it would uncover errors in the software. Also testing demonstrate that the software function appear to be working according to the specification, that performance requirement appear to have been met.

There are three ways to test program.

- For correctness
- For implementation efficiency
- For computational complexity

Test for correctness are supposed to verify that a program does exactly what it was designed to do. This is much more difficult than it may at first appear, especially for large programs.

5.2 TEST PLAN

A test plan implies a series of desired course of action to be followed in accomplishing various testing methods. The Test Plan acts as a blue print for the action that is to be followed. The software engineers create a computer program, its documentation and related data structures. The software developers is always responsible for testing the individual units of the programs, ensuring that each performs the function for which it was designed. There is an independent test group (ITG) which is to remove the inherent problems associated with letting the builder to test the thing that has been built. The specific objective so testing should be stated in measurable terms. So that theme any time to failure, the cost to find and fix the defects, remaining defect density or frequency of occurrence and test work-hours per regression test all should be stated within the test plan.

The levels of testing include:

- ❖ Unit testing
- ❖ Integration Testing
- ❖ Data validation Testing
- ❖ Output Testing

5.2.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design–the software component or module. Using the component level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and uncovered scope established for unit testing. The unit testing is white-box oriented, and step can be conducted in parallel for multiple components. The modular interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution. Boundary conditions are tested to ensure that all statements in a module have been executed at least once. Finally, all error handling paths are tested.

Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other tests are moot. Selective testing of execution paths is an essential task during the unit test. Good design dictates that error conditions be anticipated and error handling paths setup to reroute or cleanly terminate processing when an error does occur. Boundary testing is the last task of unit testing step. Software often fails at its boundaries.

Unit testing was done in Sell-Soft System by treating each module as separate entity and testing each one of them with a wide spectrum of test inputs. Some flaws in the internal logic of the modules were found and were rectified. After coding each module is tested and run individually. All unnecessary code where removed and ensured that all modules are working, and gives the expected result.

5.2.2 Integration Testing

Integration testing is systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as whole. Correction is difficult because isolation of causes is complicated by vast expanse of entire program. Once these errors are corrected, new ones appear and the process continues in a seemingly endless loop. After performing unit testing in the System all the modules were integrated to test for any in consistence is in the interfaces. Moreover differences in program structures were removed and a unique program structure was evolved.

5.2.2 Validation Testing or System Testing

This is the final step in testing. In this the entire system was tested as a whole with all forms, code, modules and class modules. This form of testing is popularly known as Black Box testing or System tests.

Black Box testing method focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program.

Black Box testing attempts to find errors in the following categories; incorrect or missing functions, interface errors, errors in data structures or external data access, performance errors and initialization errors and termination errors.

5.2.3 Output Testing or User Acceptance Testing

The system considered is tested for user acceptance; here it should satisfy the firm's need. The software should keep in touch with perspective system; user at the time of developing and making changes whenever required. This done with respect to the following points:

> ➢ Input Screen Designs,
> ➢ Output Screen Designs,

The above testing is done taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system understudy is tested using that test data. While testing the system by which test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

CHAPTER 6

IMPLEMENTATION

6.1 INTRODUCTION

Implementation is the stage of the project where the theoretical design is turned into a working system. It can be considered to be the most crucial stage in achieving a successful new system gaining the users confidence that the new system will work and will be effective and accurate. It is primarily concerned with user training and documentation. Conversion usually takes place about the same time the user is being trained or later. Implementation simply means convening a new system design into operation, which is the process of converting a new revised system design into an operation alone. At this stage the main workload, the greatest up heavy land the major impact on the existing system shifts to the user department. If the implementation is not carefully planned or controlled, it can create chaos and confusion. Implementation includes all those activities that take place to convert from the existing system to the new system. The new system may be a totally new, replacing an existing manual or automated system or it may be a modification to an existing system. Proper implementation is essential to provide are liable system to meet organization requirements. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from the old system to the new system.The system can be implemented only after through testing is done and if it is found to be working according to the specifications. The system personnel check the feasibility of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required to implement the three main aspects: education and training, system testing and changeover.

The implementation state involves the following tasks:

- ☐ Careful planning.
- ☐ Investigation of system and constraints.
- ☐   Design of methods to achieve the changeover.

6.2 IMPLEMENTATION PROCEDURES

Implementation of software refers to the final installation of the package in its real environment, to the satisfaction of the intended uses and the operation of the system. In many organizations someone who will not be operating it, will commission the software development project. In the initial stage people doubt about the software but we have to

ensure that the resistance does not build up, as one has to make sure that:

- ☐ The active user must be aware of the benefits of using the new system.
- ☐ Their confidence in the software is built up.
- ☐ Proper guidance is imparted to the user so that he is comfortable in using the application.

Before going ahead and viewing the system, the user must know that for viewing the result, the server program should be running in the server. If the server object is not up running on the server, the actual process won't take place.

6.2.1 User Training

User training is designed to prepare the user for testing and converting the system. To achieve the objective and benefits expected from computer based system, it is essential for the people who will be involved to be confident of their role in the new system. As system becomes more complex, the need for training is more important. By user training the user comes to know how to enter data, respond to error messages, interrogate the database and call up routine that will produce reports and perform other necessary functions.

6.2.2 Training on the Application Software

After providing the necessary basic training on computer awareness the user will have to be trained on the new application software. This will give the underlying philosophy of the use of the new system such as the screen flow, screen design type of help on the screen, type of errors while entering the data, the corresponding validation check at each entry and the ways to correct the date entered. It should then cover information needed by the specific user/ group to use the system or part of the system while imparting the training of the program on the application. This training may be different across different user groups and across different levels of hierarchy

6.2.3 System Maintenance

Maintenance is the enigma of system development. The maintenance phase of the software cycle is the time in which a software product performs useful work. After a system is successfully implemented, it should be maintained in a proper manner. System maintenance is an important aspect in the software development life cycle. The need for system maintenance is for it to make adaptable to the changes in the system environment.

Software maintenance is of course, far more than "Finding Mistakes".

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

The current system working technology is old fashioned and there is no usage of commonly used technologies like internet. The proposed system introduces facility for user to service and schemes online and view all information.

FUTURE SCOPE

Future Scope of this project is that completing the functionality of remaining three modules and Automating the system.

CHAPTER 8

BIBLIOGRAPHY

REFERENCES:

- Gary B. Shelly, Harry J. Rosenblatt, "*System Analysis and Design*",2009.

- Roger S Pressman, "*Software Engineering*",1994.

- PankajJalote, "So*ftware engineering*: a precise approach",2006.

- James lee and Brent ware Addison, "Open source web development with LAMP", 2003

- IEEE Std 1016 Recommended Practice for Software DesignDescriptions.

WEBSITES:

- www.w3schools.com

- www.jquery.com

- http://homepages.dcc.ufmg.br/~rodolfo/es-1-03/IEEE-Std-830-1998.pdf

- www.agilemodeling.com/artifacts/useCaseDiagram.html

CHAPTER 9

APPENDIX

9.1 Sample Code

signup.php

```php
  <?php
session_start();
include('connectivity.php');
?>

<!DOCTYPE html>
<html lang="en">

<head>

 <title>FlickCom:SignUp</title>
 <!-- HTML5 Shim and Respond.js IE11 support of HTML5 elements and media queries -->
 <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
 <!--[if lt IE 11]>
   <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
   <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
   <![endif]-->
 <!-- Meta -->
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=0, minimal-ui">
 <meta http-equiv="X-UA-Compatible" content="IE=edge" />
 <meta name="description" content="" />
 <meta name="keywords" content="">
 <meta name="author" content="Phoenixcoded" />
 <!-- Favicon icon -->
 <link rel="icon" href="assets/images/favicon.ico" type="image/x-icon">

 <!-- vendor css -->
 <link rel="stylesheet" href="assets/css/style.css">




</head>
<body>
<?php
include('connectivity.php');
?>




   <form method="post" action="adding.php" name="registration" class="registartion-form"
onsubmit="return validateForm()">
<!-- Enter first name -->
```

```php
<?php
if(isset($_REQUEST['message']))
{
   $msg=$_REQUEST['message'];
   echo'<tr><td colspan="5" style="color:#FF0000" align="center">'.$mgs.'</td></tr>';
}
?>
```

```html
<!-- [ auth-signup ] start -->
<div class="auth-wrapper">
  <div class="auth-content text-center">

    <div class="card borderless">
      <div class="row align-items-center text-center">
        <div class="col-md-12">
          <div class="card-body">
            <h4 class="f-w-400">Sign up</h4>
            <hr>
            <div class="form-group mb-3">
              <input type="text" class="form-control" id="fname" name="fname" placeholder="First
Name*" title="Start with a Capital letter & Only alphabets without space are allowed!!" value = ""
required onchange="Validate();"> <span id = "blankMsg" style="color:red"> </span>
              <script>
function Validate()
{
   var val = document.getElementById('fname').value;

   if (!val.match(/^[A-Z][A-Za-z]{3,}$/))
   {
      document.getElementById('blankMsg').innerHTML="Start with a Capital letter & Only alphabets
without space are allowed!!";
         document.getElementById('fname').value = "";
      return false;
   }
document.getElementById('blankMsg').innerHTML=" ";
   return true;
}
</script>
         </div>

         <div class="form-group mb-3">
           <input type="text" class="form-control" name="lname" id = "lname" title="Start with a
Capital letter & Only alphabets without space are allowed!!" placeholder="Last name*" value = ""
required onchange="Val();"><span id = "charMsg" style="color:red"> </span>
           <script>
function Val()
{
   var val = document.getElementById('lname').value;

   if (!val.match(/^[A-Z][A-Za-z]{3,}$/))
   {
      document.getElementById('charMsg').innerHTML="Start with a Capital letter & Only alphabets
without space are allowed!!";
         document.getElementById('lname').value = "";
```

```
            return false;
        }
document.getElementById('charMsg').innerHTML=" ";
    return true;
}
</script>




        </div>

        <div class="form-group mb-4">
         <input type="text" class="form-control" name="email" id="email" title="Enter a Valid
Email" placeholder="your email*" required onchange="return Validata();"
                /><span id="msg15" style="color:red;"></span>
                <script>
function Validata()
{
    var val = document.getElementById('email').value;

    if (!val.match(/([A-z0-9_\-\.]){1,}\@([A-z0-9_\-\.]){1,}\.([A-Za-z]){2,4}$/))
    {
        document.getElementById('msg15').innerHTML="Enter a Valid Email";

         document.getElementById('email').value = "";
        return false;
    }
document.getElementById('msg15').innerHTML=" ";
    return true;
}
</script>
        </div>




        <div class="form-group mb-4">
         <input type="password" class="form-control" name="pswd1" id ="pswd1" title="**Password
must be at least 6 characters long." placeholder="create a new password*" value = "" required
onchange="return Validp();"
                /><span id="msg9" style="color:red;"></span>
                <script>
function Validp()
{
    var val = document.getElementById('pswd1').value;

    if (!val.match(/^[A-Za-z0-9!-*]{6,15}$/))
    {
        document.getElementById('msg9').innerHTML="Password should contain atleast 6 characters";

         document.getElementById('pswd1').value = "";
        return false;
    }
document.getElementById('msg9').innerHTML=" ";
    return true;
}
```

```
</script>
        </div>



        <div class="form-group mb-4">
          <input type="password" class="form-control" name="pswd2" id = "pswd2" title="Must be
match with password" placeholder="re-enter your password*" value = "" required=" onchange="return
check();"><span id="msg17" style="color:red;"></span>
          <script>
 function check()
{
var pas1=document.getElementById("pswd1");
        var pas2=document.getElementById("pswd2");

        if(pas1.value=="")
 {
  document.getElementById('msg17').innerHTML="Password can't be null!!";
  pas1.focus();
  return false;
 }
 if(pas2.value=="")
 {
  document.getElementById('msg17').innerHTML="Please confirm password!!";
  pass2.focus();
  return false;
 }
 if(pas1.value!=pas2.value)
 {
  document.getElementById('msg17').innerHTML="Passwords does not match!!";
  pas1.focus();
  return false;
 }
   document.getElementById('msg17').innerHTML=" ";
 return true;

}
 </script>
        </div>



        <div class="form-group mb-4">
          <input type="text" class="form-control" name="phoneNumber" id="phoneNumber"
title="Enter 10 digit mobile number" placeholder="enter your mobile" required
onchange="Validphn();">
          <span id="msg4" style="color:red;"></span>
        <script>
function Validphn()
{
   var val = document.getElementById('phoneNumber').value;

   if (!val.match(/^[7-9][0-9]{1,9}$/))
   {
```

```
        document.getElementById('msg4').innerHTML="Only Numbers are allowed and must contain 10
number";


            document.getElementById('phoneNumber').value = "";
        return false;
    }
document.getElementById('msg4').innerHTML=" ";
    return true;
}


</script>
        </div>



        <div class="form-group mb-4">
          <label for="gender">Gender*</label>
          Male <input type="radio" name="gender" value="male" required>
          Female <input type="radio" name="gender" value="female" required>
          Other <input type="radio" name="gender" value="other" required>
        </div>



        <div class="form-group mb-4">
          <label>User Type*     </label>
          User <input type="radio" name="type" id="type" value="1" required>
        Fresher <input type="radio" name="type" id="type" value="2" required>
        </div>
      </div>



        <input type = "submit" name="submit" id="submit" class="submit"  value = "Register">
        <hr>
        <p class="mb-2">Already have an account? <a href="login.php" class="f-w-
400">SignIn</a></p>
      </div>
    </div>
   </div>
  </div>
 </div>



</form>


<!-- [ auth-signup ] end -->

<!-- Required Js -->
<script src="assets/js/vendor-all.min.js"></script>
<script src="assets/js/plugins/bootstrap.min.js"></script>
```

```
<script src="assets/js/pcoded.min.js"></script>
```

```
</body>
```

```
</html>
```

Index.php

```php
<?php
session_start();
include('connectivity.php');
?>
```

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>FlickCom</title>
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/shorthandcss@1.1.1/dist/shorthand.min.css" />
    <link rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Muli:200,300,400,500,600,700,800,900&display=swap
" />
    <link rel="stylesheet" type="text/css"
        href="https://cdnjs.cloudflare.com/ajax/libs/slick-carousel/1.9.0/slick.min.css" />
    <link rel="stylesheet" type="text/css" href="//cdn.jsdelivr.net/npm/slick-carousel@1.8.1/slick/slick-
theme.css" />
</head>

<body class="bg-black muli">
    <nav class="w-100pc flex flex-column md-flex-row md-px-10 py-5 bg-black">
        <div class="flex justify-between">

            <a data-toggle="toggle-nav" data-target="#nav-items" href="#"
                class="flex items-center ml-auto md-hidden indigo-lighter opacity-50 hover-opacity-100
ease-300 p-1 m-3">
                <i data-feather="menu"></i>
            </a>
        </div>
        <div id="nav-items" class="hidden flex sm-w-100pc flex-column md-flex md-flex-row md-
justify-end items-center">

            <a href="signup.php" class="button bg-white black fw-600 no-underline mx-5">Register</a>
        </div>
    </nav>

    <!-- hero section -->
    <section id="home" class="min-h-100vh flex justify-start items-center">
        <div class="mx-5 md-mx-l5">

            <div>
```

```
            <h2 class="white fs-l3 lh-2 md-fs-xl1 md-lh-1 fw-900 "><i>"Filmmaking is a chance to live
many lifetimes."</i> <br /><i>"Filmmaking is teamwork."</i>
                <br/><i>Let find your team here..</i></h2>



        </div>
    </section>

    <!-- features -->


    <!-- slider -->


    <!-- testimonials -->

    <!-- featured blog -->


    <!-- blog -->
    <section class="p-0 md-p-5">
        <div class="flex flex-wrap">
            <div class="w-100pc md-w-33pc p-10">
                <a href="#" class="block no-underline p-5 br-8 hover-bg-indigo-lightest-10 hover-scale-up-1
ease-300">
                    <img class="w-100pc" src="assets/images/yoann-siloine-dyaxQ-aoGWY-unsplash.jpg"
alt="">

                </a>
            </div>
            <div class="w-100pc md-w-33pc p-10">
                <a href="#" class="block no-underline p-5 br-8 hover-bg-indigo-lightest-10 hover-scale-up-1
ease-300">
                    <img class="w-100pc" src="assets/images/fabrice-villard-Jrl_UQcZqOc-unsplash.jpg"
alt="">


                </a>
            </div>
            <div class="w-100pc md-w-33pc p-10">
                <a href="#" class="block no-underline p-5 br-8 hover-bg-indigo-lightest-10 hover-scale-up-1
ease-300">
                    <img class="w-100pc" src="assets/images/florian-klauer-mk7D-4UCfmg-unsplash.jpg"
alt="">

                </a>
            </div>

        </div>
    </section>

    <!-- subscribe -->
```

```
<!-- footer -->

</div>
<script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>
<script src="https://unpkg.com/feather-icons"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/slick-carousel/1.9.0/slick.min.js"></script>
<script src="https://cdn.jsdelivr.net/gh/cferdinandi/smooth-scroll@15.0.0/dist/smooth-
scroll.polyfills.min.js"></script>
<script src="assets/js/script.js"></script>
</body>

</html>
```

Addmembers.php

```
<!DOCTYPE html>
<html lang="en">

<head>
    <title>Flick:Admin-AddMembers</title>
    <!-- HTML5 Shim and Respond.js IE11 support of HTML5 elements and media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
    <!--[if lt IE 11]>
        <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
        <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
        <![endif]-->
    <!-- Meta -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=0, minimal-
ui">
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="description" content="" />
    <meta name="keywords" content="">
    <meta name="author" content="Phoenixcoded" />
    <!-- Favicon icon -->
    <link rel="icon" href="assets/images/favicon.ico" type="image/x-icon">

    <!-- vendor css -->
    <link rel="stylesheet" href="assets/css/style.css">



</head>
<body class="">
    <!-- [ Pre-loader ] start -->
    <div class="loader-bg">
        <div class="loader-track">
            <div class="loader-fill"></div>
        </div>
    </div>
    <!-- [ Pre-loader ] End -->
    <!-- [ navigation menu ] start -->
    <nav class="pcoded-navbar ">
        <div class="navbar-wrapper ">
```

```
          <div class="navbar-content scroll-div " >

              <div class="">
                <div class="main-menu-header">
                  <img class="img-radius" src="assets/images/user/avatar-2.jpg" alt="User-Profile-
Image">

                  <div class="user-details">
                    <span>Admin</span>

                  </div>

              </div>

              <ul class="nav pcoded-inner-navbar ">
                <li class="nav-item">
                  <a href="index.html" class="nav-link "><span class="pcoded-micon"><i class="feather
icon-home"></i></span><span class="pcoded-mtext">Dashboard</span></a>
                </li>

              </ul>



          </div>
        </div>
    </nav>
    <!-- [ navigation menu ] end -->
    <!-- [ Header ] start -->

    <header class="navbar pcoded-header navbar-expand-lg navbar-light header-dark">


          <div class="m-header">
            <a class="mobile-menu" id="mobile-collapse" href="#!"><span></span></a>
            <a href="#!" class="b-brand">
              <!-- ========   change your logo hear   ============ -->
              <img src="assets/images/logo.png" alt="" class="logo">
              <img src="assets/images/logo-icon.png" alt="" class="logo-thumb">
            </a>
            <a href="#!" class="mob-toggler">
              <i class="feather icon-more-vertical"></i>
            </a>
          </div>



    </header>
    <!-- [ Header ] end -->

    <br><br><br><br>

<!-- [ Main Content ] start -->
<div class="pcoded-main-container">
```

```
    <div class="pcoded-content">
      <!-- [ breadcrumb ] start -->
          <div class="container-fluid">
        <div class="row">
          <div class="col-md-7">
            <div class="card">
              <form class="form-horizontal" method="post" action="memberadd.php" name="myform"
class="registartion-form" onsubmit="return validateform()>

            <?php
if(isset($_REQUEST['message']))
{
    $msg=$_REQUEST['message'];
    echo'<tr><td colspan="2" style="color:#FF0000" align="center">'.$mgs.'</td></tr>';
}
?>

                <div class="card-body">


                <center>
                  <h4 class="card-title">Add New Staff</h4>
                </center>

                  <div class="form-group row">
                   <label
                     for="fname"
                     class="col-sm-3 text-end control-label col-form-label"
                     >First Name*</label
                   >
                   <div class="col-sm-7">
                    <input
                      type="text"
                      class="form-control"
                      id="fname" name="fname"
                      placeholder="First Name Here" title="Start with a Capital letter & Only alphabets
without space are allowed!!" required onchange="Validate();"
                    /><span id="msg2" style="color:red;"></span>
                   </div>
                   <script>
function Validate()
{
    var val = document.getElementById('fname').value;

    if (!val.match(/^[A-Z][A-Za-z]{3,}$/))
    {
        document.getElementById('msg2').innerHTML="Start with a Capital letter & Only alphabets
without space are allowed!!";
            document.getElementById('fname').value = "";
        return false;
    }
document.getElementById('msg2').innerHTML=" ";
    return true;
```

```
      }
    </script>



              </div>
              <div class="form-group row">
               <label
                 for="lname"
                 class="col-sm-3 text-end control-label col-form-label"
                 >Last Name*</label
               >
               <div class="col-sm-7">
                <input
                  type="text"
                  class="form-control"
                  id="lname" name="lname"
                  placeholder="Last Name Here" title="Start with a Capital letter & Only alphabets
without space are allowed!!" required onchange="Val();"
                  /><span id="msg1" style="color:red;"></span>
               </div>
               <script>
function Val()
{
    var val = document.getElementById('lname').value;

    if (!val.match(/^[A-Z][A-Za-z]{3,}$/))
    {
       document.getElementById('msg1').innerHTML="Start with a Capital letter & Only alphabets
without space are allowed!!";
            document.getElementById('lname').value = "";
       return false;
    }
document.getElementById('msg1').innerHTML=" ";
    return true;
}
</script>

              </div>


               <div class="form-group row">
               <label
                 for="email"
                 class="col-sm-3 text-end control-label col-form-label"
                 >User Name*</label
               >
               <div class="col-sm-7">
                <input
                  type="text"
                  class="form-control"
                  id="email" name="email"
                  placeholder="your email" title="Enter a Valid Email" required onchange="return
Validata();"
                  /><span id="msg15" style="color:red;"></span>
               </div>
```

```
<script>
function Validata()
{
   var val = document.getElementById('email').value;

   if (!val.match(/([A-z0-9_\-\.]){1,}\@([A-z0-9_\-\.]){1,}\.([A-Za-z]){2,4}$/))
   {
      document.getElementById('msg15').innerHTML="Enter a Valid Email";

       document.getElementById('email').value = "";
      return false;
   }
document.getElementById('msg15').innerHTML=" ";
   return true;
}
</script>
</div>

                <div class="form-group row">
                 <label
                   for="password"
                   class="col-sm-3 text-end control-label col-form-label"
                   >Password</label
                 >
                 <div class="col-sm-7">
                  <input
                    type="password"
                    class="form-control"
                    id="password" name="password"
                    placeholder="Password Here"
  title="Password must be at least 6 characters long." required onchange="return Validp();"
                  /><span id="msg9" style="color:red;"></span>
                 </div>
                 <script>
function Validp()
{
   var val = document.getElementById('password').value;

   if (!val.match(/^[A-Za-z0-9!-*]{6,15}$/))
   {
      document.getElementById('msg9').innerHTML="Password should contain atleast 6 characters";

       document.getElementById('password').value = "";
      return false;
   }
document.getElementById('msg9').innerHTML=" ";
   return true;
}

</script>

                </div>
                <div class="form-group row">
                 <label
```

```
                    for="Designation"
                    class="col-sm-3 text-end control-label col-form-label"
                    >Designation*</label
                  >
                  <div class="col-sm-7">
                   <input
                    type="text"
                    class="form-control"
                    id="feild" name="feild"
                    list="defaultEmails" size="64" maxlength="256" multiple placeholder="Designation"
required
                   />
                   <datalist id="defaultEmails">
                      <option value="Editing">
                      <option value="Cinematographer">
                      <option value="Casting">
                      <option value="Camera Operator">
                      <option value="Film Crew">
                      <option value="Production Assistant">
                      <option value="Sound Engineer">
                   </datalist>
                  </div>
                </div>

                <div class="form-group row">
                  <label
                    for="type"
                    class="col-sm-3 text-end control-label col-form-label"
                    >User Type*</label
                  >
                  <div class="col-sm-7">
                    User <input type="radio" name="type" value="1" required>
                    Fresher <input type="radio" name="type" value="2" required>
                    Members <input type="radio" name="type" value="3" required>
                  </div>
                </div>
              </div>
              <center><button type = "submit" name="submit" class="btn btn-success" ><i
class="feather mr-2 icon-check-circle"></i>ADD</button>
              </center>
              </form>
          </div>
        </div>
        </div>
        </div>
      </div>
     </div>
    </div>



    <!-- Required Js -->
    <script src="assets/js/vendor-all.min.js"></script>
    <script src="assets/js/plugins/bootstrap.min.js"></script>
```

```
<script src="assets/js/pcoded.min.js"></script>

<!-- Apex Chart -->
<script src="assets/js/plugins/apexcharts.min.js"></script>


<!-- custom-chart js -->
<script src="assets/js/pages/dashboard-main.js"></script>
</body>

</html>
```

AddVaccancy.php

```
<!DOCTYPE html>
<html lang="en">

<head>
  <title>Flick:Vaccancy</title>
  <!-- HTML5 Shim and Respond.js IE11 support of HTML5 elements and media queries -->
  <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
  <!--[if lt IE 11]>
    <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
    <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
    <![endif]-->
  <!-- Meta -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=0, minimal-ui">
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="description" content="" />
  <meta name="keywords" content="">
  <meta name="author" content="Phoenixcoded" />
  <!-- Favicon icon -->
  <link rel="icon" href="assets/images/favicon.ico" type="image/x-icon">

  <!-- vendor css -->
  <link rel="stylesheet" href="assets/css/style.css">




</head>
<body class="">
  <!-- [ Pre-loader ] start -->
  <div class="loader-bg">
    <div class="loader-track">
      <div class="loader-fill"></div>
    </div>
  </div>
  <!-- [ Pre-loader ] End -->
  <!-- [ navigation menu ] start -->
  <nav class="pcoded-navbar ">
    <div class="navbar-wrapper ">
      <div class="navbar-content scroll-div " >
```

```html
        <div class="">
          <div class="main-menu-header">
            <img class="img-radius" src="assets/images/user/avatar-2.jpg" alt="User-Profile-Image">
            <div class="user-details">
              <span>Me</span>
              <div id="more-details"><i class="fa fa-chevron-down m-l-5"></i></div>
            </div>
          </div>
          <div class="collapse" id="nav-user-link">
            <ul class="list-unstyled">
              <li class="list-group-item"><a href="auth-normal-sign-in.html"><i class="feather icon-log-
out m-r-5"></i>Logout</a></li>
            </ul>
          </div>
        </div>

        <ul class="nav pcoded-inner-navbar ">
          <li class="nav-item">
            <a href="Members-index.php" class="nav-link "><span class="pcoded-micon"><i
class="feather icon-home"></i></span><span class="pcoded-mtext">Home</span></a>
          </li>

        </ul>



      </div>
    </div>
  </nav>
  <!-- [ navigation menu ] end -->
  <!-- [ Header ] start -->

  <header class="navbar pcoded-header navbar-expand-lg navbar-light header-dark">


        <div class="m-header">
          <a class="mobile-menu" id="mobile-collapse" href="#!"><span></span></a>
          <a href="#!" class="b-brand">
            <!-- ========   change your logo hear   ============ -->
            <img src="assets/images/logo.png" alt="" class="logo">
            <img src="assets/images/logo-icon.png" alt="" class="logo-thumb">
          </a>
          <a href="#!" class="mob-toggler">
            <i class="feather icon-more-vertical"></i>
          </a>
        </div>



  </header>
  <!-- [ Header ] end -->


<br><br><br><br><br><br><br>
<!-- [ Main Content ] start -->
```

```
<div class="pcoded-main-container">



   <div class="pcoded-content">
      <!-- [ breadcrumb ] start -->
          <div class="container-fluid">
       <div class="row">
        <div class="col-md-5">
          <div class="card">
           <form class="form-horizontal" method="post" action="AddingVaccancy.php" name="registration"
class="registartion-form" onsubmit="return function()>


           <?php
if(isset($_REQUEST['message']))
{
   $msg=$_REQUEST['message'];
   echo'<tr><td colspan="2" style="color:#FF0000" align="center">'.$mgs.'</td></tr>';
}
?>

              <div class="card-body">


              <center>
                <h4 class="card-title">Add New Vaccancy</h4>
              </center>

                <div class="form-group row">
                 <label
                   for="feild"
                   class="col-sm-3 text-end control-label col-form-label"
                   >Vaccancy Feild*</label
                 >
                 <div class="col-sm-7">
                  <input
                    type="text"
                    class="form-control"
                    id="feild" name="feild"
                    list="defaultEmails" size="64" maxlength="256" multiple placeholder="Vaccancy Feilds"
required
                  />
                  <datalist id="defaultEmails">
                     <option value="Editing">
                     <option value="Cinematographer">
                     <option value="Casting">
                     <option value="Camera Operator">
                     <option value="Film Crew">
                     <option value="Production Assistant">
                     <option value="Sound Engineer">
                   </datalist>
                 </div>
               </div>
               <div class="form-group row">
```

```
                    <label
                     for="novaccancy"
                     class="col-sm-3 text-end control-label col-form-label"
                     >Total Vaccancy*</label
                    >
                    <div class="col-sm-7">
                     <input
                      type="text"
                      class="form-control"
                      id="novaccancy" name="novaccancy"
                      placeholder="Number of Vaccancies" required
                     onchange="Validatevac();"
                     /><span id="msg7" style="color:red;"></span>
                    </div>
                    <script>
function Validatevac()
{
   var val = document.getElementById('novaccancy').value;

   if (!val.match(/^[0-9][0-9]{0,9}$/))
   {
      document.getElementById('msg7').innerHTML="Only Numbers are allowed and must contain
number";


            document.getElementById('novaccancy').value = "";
      return false;
   }
document.getElementById('msg7').innerHTML=" ";
   return true;
}

</script>
                 </div>

                 <div class="form-group row">
                  <label
                    for="qualification"
                    class="col-sm-3 text-end control-label col-form-label"
                    >Qualification*</label
                  >
                  <div class="col-sm-7">
                   <input
                     type="text"
                     class="form-control"
                     id="qualification" name="qualification" list="defaultEmails" size="64" maxlength="256"
multiple
                     placeholder="Select qualification" required
                   />
                   <datalist id="defaultEmails">
                      <option value="Electrical Engineering">
                      <option value="Bsc.Digital Filmmaking">
                      <option value="cinematography">
                      <option value="MSc in Cinema.">
                      <option value="MA in Film Studies.">
```

```
                    <option value="MSc in Film & TV Production">
                    <option value="BA Animation">
                      <option value="BA (Hons)Digital Content Creation">
                  </datalist>
                </div>
              </div>
              <div class="form-group row">
                <label
                  for="experience"
                  class="col-sm-3 text-end control-label col-form-label"
      >Experience*</label
                  >
                <div class="col-sm-7">
                  <input
                    type="text" maxlength="2" minlength="1"
                    class="form-control"
                    id="experience" name="experience"
                    placeholder="Experience" required onchange="Validatexep();"
                  /><span id="msg8" style="color:red;"></span>

                </div>
```

```
<script>
function Validatexep()
{
   var val = document.getElementById('experience').value;

   if (!val.match(/^[0-9][0-9]{0,9}$/))
   {
      document.getElementById('msg8').innerHTML="Only Numbers are allowed and must contain
number";


            document.getElementById('experience').value = "";
      return false;
   }
document.getElementById('msg8').innerHTML=" ";
   return true;
}

</script>
```

```
              </div>
              <div class="form-group row">
                <label
                  for="deadline"
                  class="col-sm-3 text-end control-label col-form-label"
                  >Deadline for Application*</label
                  >
                <div class="col-sm-7">
                  <input
                    type="date"
                    class="form-control"
                    id="deadline" name="deadline"
                    placeholder="Date of last submition" min='2022-01-01' max='2023-01-01' required
                  /><script type="text/javascript"
```

```
src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
<script type="text/javascript">
   $(function () {
      var today = new Date();
      var month = ('0' + (today.getMonth() + 1)).slice(-2);
      var day = ('0' + today.getDate()).slice(-2);
      var year = today.getFullYear();
      var date = year + '-' + month + '-' + day;
      $('[id*=deadline]').attr('min', date);
   });
</script>

                </div>

               </div>
             </div>
             <center><button type = "submit" name="submit" class="btn btn-success" ><i class="feather mr-2
icon-check-circle"></i>ADD VACCANCY</button>
             </center>
             </form>
          </div>
        </div>
        </div>
        </div>




        </div>
        <!-- Latest Customers end -->
      </div>




     <!-- [ Main Content ] end -->
   </div>
</div>

   <!-- Required Js -->
   <script src="assets/js/vendor-all.min.js"></script>
   <script src="assets/js/plugins/bootstrap.min.js"></script>
   <script src="assets/js/pcoded.min.js"></script>

<!-- Apex Chart -->
<script src="assets/js/plugins/apexcharts.min.js"></script>


<!-- custom-chart js -->
<script src="assets/js/pages/dashboard-main.js"></script>
</body>
```

```
</html></html>
    <?php
    include('connectivity.php');
    if(isset($_POST['submit']))
    {
        $a=$_POST['feild'];
        $b=$_POST['novaccancy'];
        $c=$_POST['qualification'];
        $d=$_POST['experience'];
        $e=$_POST['deadline'];
        $query3=mysqli_query($query,"INSERT INTO
        `tbl_vaccancy`(`Vaccancy_feild`,
        `Total_vaccancy`,`qualification`,`experience`,`deadline`) VALUES
        ('$a','$b','$c','$d',$e)");

        if($query3)
        {
        header('location:vaccancy.php');
        }
    }
?>
```
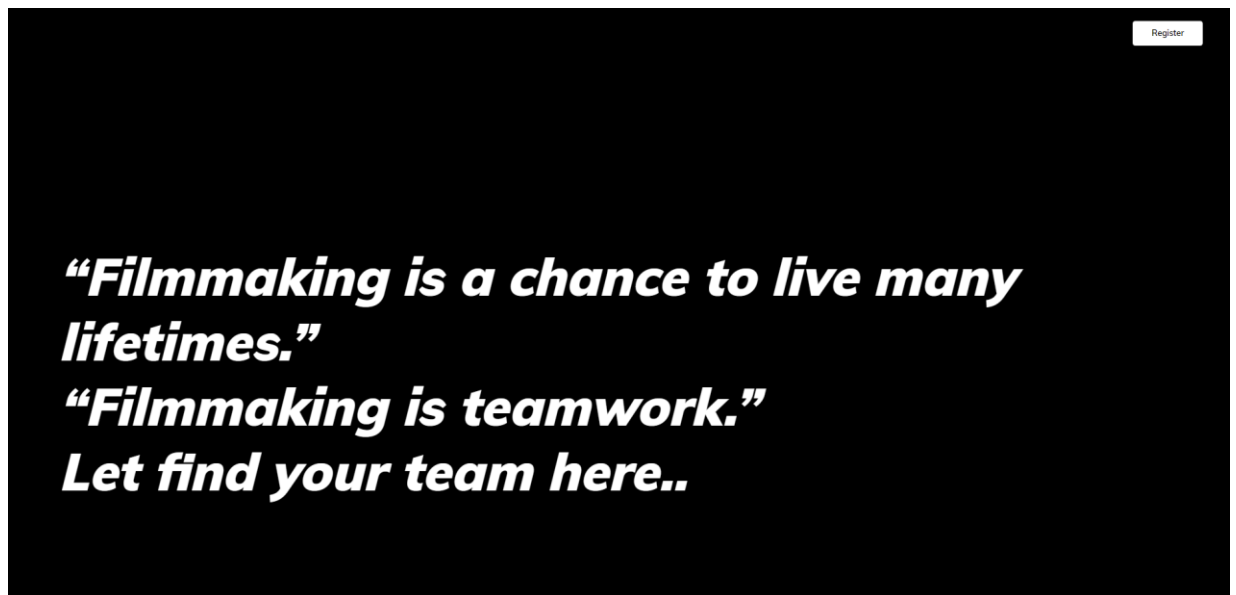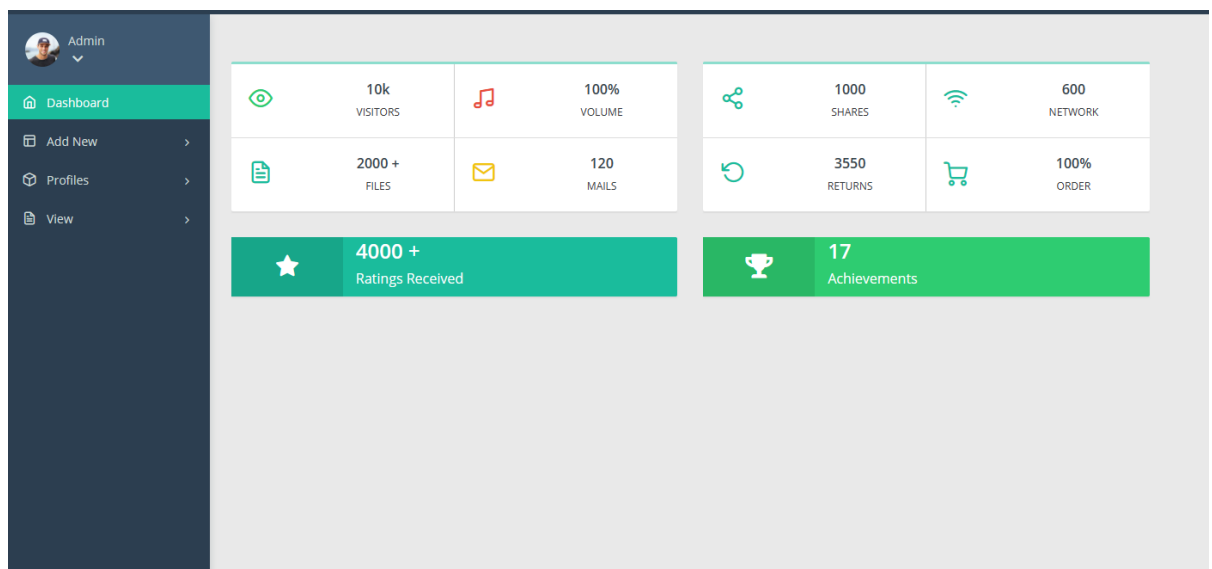
9.2 ScreenShots

Home page

Admin Dashboard

Staff Details

Staffs Record

FIRST NAME	LAST NAME	EMAIL	DESIGNATION	PASSWORD		
avani	hsagccc	23	shacn	pc	Delete	Edit
sdgds	sdg	sdfghj@gmail.com	Aleena12	ytjy	Delete	Edit
4w3	4	wfr@gmail.com	fAssss222	fetttwww	Delete	Edit
dxr	gfrssg	fbdh@gmail.com	Addd2sdcv	sdgsx	Delete	Edit
Alan	Jude	alanjude@gmail.com	alanjude5	dr	Delete	Edit