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

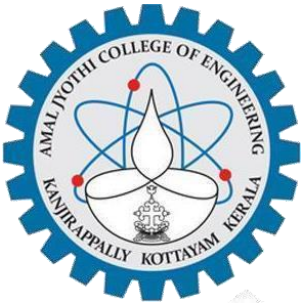
[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE, Accredited by NAAC with 'A' grade. Koovappally, Kanjirappally, Kottayam, Kerala –686518]

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 — **Internal Guide**

Ms. Nimmy Francis — **Coordinator**

Rev. Fr. Dr. Rubin Thottupurathu Jose — **Head of the Department**

External Examiner

DECLARATION

I hereby declare that the project report **"FLICKCOM"** is a bonafide 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: 21-07-2022 **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** and **Ms. Nimmy Francis** for their 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.

ALEENA JOSEPH

ABSTRACT

FlickCom is a web based system for film industry, developed in PHP, which is very helpful for the freshers those who are interested in film field. The person who need freshers for their project, they can request for joining to the site as a staff. After approve their request by the admin, staffs(film personalities) can schedule auditions for different fields, like acting, music, cinematography etc,. And also they can upload the details of various vacancies in film industry or related to their project.

The freshers those who are seeking opportunities in film industry, can create profile either for film industry jobs(vacancy) or for different auditions. Freshers can upload their resume, introductory video and their certificates or achievements in this site. They can apply for various auditions and vacancies. Each auditions has an amount of registration fees. Staffs will manually check their qualifications and select them, those who are exactly apt for different vacancies.

This system will also helps for finding details about the upcoming films and it's releasing details. Admin will upload the posters and trailer the latest movies, and details corresponds to them. Users can get details of very popular and upcoming movies details and they can rate each movies.

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

1.1 PROJECT OVERVIEW

FlickCom is a web based system for film industry, developed in PHP, which is very helpful for the freshers those who are interested in film field. This system will helps to find jobs and possibilities in film. Freshers can find out upcoming auditions and new vacancies in various fields in film industry. Freshers can apply for these, which are match with their profile. While applying auditions there has an amount of registration fees. The applicants can pay it by online.

The person those who are film personalities, they can join as a staff by request. After acceptance their request by admin, staffs can update new auditions and various vacancies. They will accept or reject each applications and download list of applicants.

Admin will upload latest and upcoming movie details like; poster, trailer, main details of movie (name of movie, writer, director, actors and actress) etc.. Users can rate each movies.

1.2 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 and new auditions in film industry.

The system mainly includes 4 modules. They are:

1. ADMIN:

First Admin will login to the system. Admin has the right to accept new staff's requests and send their new username and password by mail. Also can reject the request. He/she will get the number of new request in his/her dashboard. Admin will admit them by checking their uploaded Id proof in manually. Based on it admin has the permission to admit or reject them. And he/she can monitor activities of staffs.

Admin can post upcoming movie posters, trailers and its releasing details. Admin will add film category like; family, thriller, action, etc.. The posters of movies will be expire after it's releasing date. Admin is able to view the details about Vacancy, Audition, profile of freshers.

- Get number of new requests in his/her dashboard.
- Accept or reject staff's requests.
- Add film types and release platform.
- Upload new movie details.
- Expire the movie details after it's release.
- View list of freshers and users.

2. USERS:

The users can sign in to the site without registration. They can view the movie poster, trailers and releasing details. And also they can get the basic movie details. They can rate movies. To rate movies first they want to signup.

- Users can view upcoming movie posters, trailers, releasing details and other details related to the movie.
- Rate the movies after registration.

3. FRESHERS:

Freshers are the users those who are seeking new opportunities in film industry. Through this site they can create profiles in different fields. They can create profiles for either auditions or vacancy. They will get various vacancies and auditions which are match with their profiles. They can apply for these vacancies and auditions by uploading resume introductory video ,educational certificates, etc,. They can pay application fees for audition by online and can get bill of payment. And they get status about their applications.

- Register and login.
- Create profile for either audition or vacancy.
- Get new vacancy or audition which are matched with their profile.
- Apply for audition.
- Pay the registration fees and download the bill.
- Get the application status through email.
- Apply for vacancies.

- After application get it's status as pending, accepted or rejected.

4. STAFF:

The staffs are the persons those who are related to the film industry. They can request for join to this site as a staff . For that first they want to upload their Id proof. The admission is done by evaluating their proof in manually by the admin. After accepting their request by admin they will get their username and password through the given email. Using that they can sign in to the site. Then they can modify their username and password.

The staffs are responsible for update new vacancies and scheduling auditions for various fields. The scheduled vacancies and auditions will automatically unavailable after it's last date of registration. In the case of auditions, staff can view the list of applicants, from that he/she can verify their registration in manually, that who are paid the application fees. And from the applications for various vacancies ,they can select matchable profiles from the entire applications. They can accept or reject the applications and can download the list of applicants.

The staffs are able to view the number of applicants based on their age in a graphical representation . And also he/she can get the number of applicants in vacancies and auditions in a graphical manner. The main functionalities of Staffs is given in below:

- Schedule auditions.
- Verify the applications, that whose are paid the application fee.
- Download the list of applicants for auditions.
- Add latest vacancies in various fields.
- Approve or reject the application by verifying the qualifications and documents in manually.
- Automatically expire the auditions and vacancies after it's last date for apply.
- Get number of applicants for auditions and vacancies in a graphical manner.
- Get number of applicants based on age in a graphical representation.

CHAPTER 2

SYSTEM STUDY

2.1 INTRODUCTION

The process of gathering and evaluating data, identifying problems, and using the data to recommend system modifications is known as system analysis. During this problem-solving process, there must be considerable communication between the system users and the system developers. Any system development process should begin with a system analysis or research. The system is carefully inspected and evaluated. The system analyst assumes the position of the interrogator and enquires carefully about the operation of the current system. The system is viewed as a whole, and the system's input is identified. The various processes are linked to the outputs of the organizations. Identification of the issue, selection of the pertinent and consequential variables, analysis and synthesis of the numerous elements, and selection of the best or at least adequate course of action are the objectives of system analysis.

A thorough examination of the process must be conducted using various techniques such as interviews, questionnaires, and so on. The information obtained by various sources must be examined in order to draw a conclusion. Understanding the operation of the system is the end goal. This is referred to as the current system. The existing system issue is now being closely examined, and problem areas have been identified. The designer now acts as a problem solver, attempting to resolve the issues that the company is experiencing. The answers are offered as recommendations. The best option is then selected after being analytically compared to the current system. The user is made aware of the proposition and asked to approve it. The idea is examined and the necessary revisions are made in response to user requests. When the user accepts the idea, this cycle comes to an end.

The process of acquiring and analyzing data in advance of more in-depth system research is called preliminary research. Preliminary research necessitates ongoing interaction between system users and developers in order to overcome problems. It conducts a lot of feasibility studies. The results of these investigations can be used to select the most effective techniques for system study and analysis by providing a rough notion of the system activities.

2.2 EXISTING SYSTEM

The current system is not entirely automated. 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 takes more time to get information for their appropriate needs.

2.3 DRAWBACKS OF EXISTING SYSTEM

- Workforce effort is required.
- It is difficult to keep important information in the user's mind.
- More manual hours are required to process the application.

2.4 PROPOSED SYSTEM

All of the flaws in the current system are intended to be fixed by the proposed solution. For those who enjoy watching movies, a user-friendly and appealing system is essential. 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 post reviews about the movies and can rate movies. The freshers those who are seeking opportunities in film industry, they can create profiles and upload resume. They can get new vacancies and auditions details. The staffs can select the profiles also.

2.5 ADVANTAGES OF PROPOSED SYSTEM

The system is easy to use and set together. The system uses extremely little system resources and works in almost all environments. It has the following characteristics:

- **Ensure data accuracy:-**

Human error that occurs when entering user information during registration is eliminated by the offered approach.

- **Better service: -**

The product will make hard copy storage unnecessary. By carrying out the same task more than once, we can also save time and money this way. There won't be any data loss if the data is retained for a longer time.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDY

To ascertain whether the project will fulfil the objectives of the company for the amount of work, effort, and time put in it, a feasibility study is carried out. A feasibility study allows the developer to forecast the project's future and usefulness. A feasibility study of a system proposal is based on its work ability, which is the impact on the organization, ability to meet the needs of their users, and effective use of resources. As a result, when a new application is proposed, it is usually subjected to a feasibility study before being approved for development.

The document describes the project's viability and includes a number of factors that were carefully considered throughout the feasibility study, such as its technical, economic, and operational viability. It has the following features: -

3.1.1 Economical Feasibility

The cost and benefit of developing the system must be justified. Criteria to ensure that effort is focused on projects that will yield the best results and return the earliest. One of the factors influencing the development of a new system is its cost.

The following significant financial issues were raised during the preliminary investigation:

- The costs conduct a comprehensive system investigation.
- The price of the hardware and software.
- The advantages come in the form of lower costs or fewer costly errors.

Because the proposed system is being developed as part of a project, there will be no manual costs. Furthermore, the fact that all of the resources are already available suggests that the system can be developed affordably.

The FlickCom project's costs were divided into three categories: system costs, development costs, and hosting costs. All calculations show that the project was created at a low cost. It was entirely developed using open-source software.

3.1.2 Technical Feasibility

The system must first be evaluated from a technical standpoint. An outline design of the system needs in terms of input, output, programs, and procedures must serve as the foundation for this feasibility study. The inquiry must move on to recommend the sort of equipment, necessary way of developing the system, and method of operating the system once it has been designed once an outline system has been discovered.

Throughout the investigation, the following technical issues arose:

- Does the current technology allow for the suggested solutions?
- Is it possible for the system to grow if it is improved?

The project's design should take the limits into account while still achieving the necessary functionalities and performance. The project necessitates the use of a high-resolution scanning device as well as cryptographic techniques. Although the technology may become obsolete after a certain period of time, the system may still be used because newer versions of the same software support older versions. As a result, there are few constraints in this project. The system was created with PHP in the front end and MySQL in the back end; the project is technically feasible for development. The system was built with PHP for the front end and a MySQL server for the back end; the project is technically feasible. The system in question is equipped with an Intel Pentium Gold processor, 4GB of RAM, and a 500GB hard drive.

3.1.3 Behavioral Feasibility

The suggested system comprises the following questions:

- Is there enough help available for users?
- Will the proposed system cause any harm?

The project would be advantageous once it is created and put into action since it achieves the goals. After carefully weighing all behavioral factors, the project is determined to be behaviorally feasible.

3.2 SYSTEM SPECIFICATION

3.2.1. Hardware Specification

Processor - Intel core i3
RAM - 4 GB
Hard disk - I TB

3.2.2. Software Specification

Front End - HTML, CSS
Back end - MYSQL
Client on PC - Windows 7 and above.
Technologies used - JS, J Query, PHP, CSS, HTML

3.3 SOFTWARE DESCRIPTION

3.3.1 PHP

PHP is a server-side scripting language that can be used for both general-purpose programming and web development. PHP is currently used by over 244 million websites and 2.1 million web servers. The reference version of PHP, which Rasmus Ledorf initially developed in 1995, is now produced by the PHP group. Originally abbreviated as PHP: Personal Home Page, the term today refers to PHP: Hypertext Preprocessor. PHP code is interpreted by a web server's PHP processor module, which then creates the web page. Instead of calling an external file, PHP commands can be directly inserted into an HTML source document to process data. Due to trademark restrictions, PHP has evolved to include a command-line interface capability and can be used independently, rendering it incompatible with the GNU General Public License (GPL).On practically every operating system and platform, PHP may be installed without charge as a standalone shell and on the majority of web servers.

3.3.2 MySQL

MySQL, the most popular Open-Source SQL database management system, was created, distributed, and supported by Oracle Corporation. The MySQL website has the most recent information about MySQL software.

- **MySQL is a database management system.**

A database is a set of data that has been structured well. A simple shopping list, a photo gallery, or the enormous volumes of data in a business network might all be examples. Data saved in a computer database must be added to, accessed, and processed using a database management system, such as MySQL Server. Because computers are exceptionally good at processing vast volumes of data, database management systems play a significant role in computing, either as standalone utilities or as parts of other programs.

- **MySQL databases are relational.**

A relational database stores data in separate tables rather than in a single large storeroom. The database structures are organized into physical files that are optimized for performance. The logical model, which includes objects like databases, tables, views, rows, and columns, provides a versatile programming environment. You define the rules that govern the relationships between various data fields, such as one-to-one, one-to-many, unique, required, or optional, and "pointers" between different tables. The database enforces these rules, ensuring that your application never encounters inconsistent, duplicate, orphan, out-of-date, or missing data. SQL is an abbreviation for "Structured Query Language." SQL is the most widely used standardized language for accessing databases. Depending on your programming environment, you can either directly enter SQL (for instance, to generate reports), embed SQL statements in other language code, or use a language-specific API that conceals the SQL syntax. By way of the ANSI/ISO SQL Standard, SQL is defined. Since its inception in 1986, the SQL standard has undergone multiple revisions. In this document, "SQL: 1999" refers to the standard released in 1999, "SQL: 2003" refers to the most recent version of the standard, and "SQL:92" refers to the standard released in 1992. When we refer to "the SQL standard," we always mean the most recent revision of the SQL Standard.

- **MySQL software is Open Source.**

Oracle Corporation created, distributed, and supported MySQL, the most popular Open Source SQL database management system. The MySQL website contains the most up-to-date information on MySQL software. The GPL (GNU General Public License) is used by MySQL to define what you can and cannot do with the software in various situations. You can buy a commercially licensed version from us if you are uncomfortable with the GPL or need MySQL code to be embedded in a commercial application. For more information, see the MySQL Licensing Overview.

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

Give it a go if that's what you're going for. On a desktop or laptop, MySQL Server may run seamlessly with your other apps, web servers, and other devices while requiring little to no attention. You can set up MySQL to use all of the RAM, CPU, and I/O capabilities if you dedicate an entire machine to it.

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

A multi-threaded SQL server that supports various back ends, several client programs and libraries, administrative tools, and a wide range of application programming interfaces are all included in the client/server system known as MySQL Database Software (APIs). You can use MySQL Server's embedded multi-threaded library into your program to produce a more manageable, quicker, and more compact standalone solution.

CHAPTER 4

SYSTEM DESIGN

4.1 INTRODUCTION

Any engineered system or product's development process begins with design. The process of design is artistic. For a system to be effective, a proper design is necessary. The term "design" is defined as "the process of applying various techniques and principles to define a process or a system in sufficient detail to allow its physical realization". It can be defined as the process of applying various techniques and principles to define a device, a processor, or a system in sufficient detail to allow physical realization. Regardless of the development paradigm, software design is the technical foundation of the software engineering process. The system design creates the architectural detail needed to construct a system or product. This software has through the best design process imaginable, fine refining all efficiency, performance, and accuracy levels, as with any methodical methodology. A document for users becomes a document for programmers or database staff throughout the design phase. The two stages of system design are logical design and physical design.

4.2 UML DIAGRAM

UML is a standard language for describing, visualizing, constructing, and documenting software system artefacts. The Object Management Group (OMG) was responsible for developing UML, and a draught of the UML 1.0 definition was presented to the OMG in January 1997.

Unified Modeling Language is known as UML. Compared to other popular programming languages like C++, Java, COBOL, etc., UML is unique. Software designs are made using the visual language known as UML. A general-purpose visual modelling language called UML is used to visualize, design, construct, and document software systems. UML is not just used to represent software systems, despite the fact that this is where it is most frequently utilized. Systems that are not software are also modelled using it. For example, consider the process flow in a manufacturing unit. UML is not a programming language, but UML diagrams can be used to generate code in a variety of languages. UML is inextricably linked to object-oriented analysis and design. UML has become an OMG standard after some standardization. All of the elements and relationships are combined to form a complete UML diagram that represents a system The most important aspect of the

entire process is the visual effect of the UML diagram. All of the other elements are used to complete it.

The diagrams in UML are as follows:

- 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 visual representation of how system components interact. In system analysis, a use case is a tool for locating, explaining, and arranging system needs. In this context, the word "system" refers to something that is being created or run, such a website for mail-order product sales and services. Use case diagrams are used in UML (Unified Modeling Language), a standard language for modelling actual items and systems.

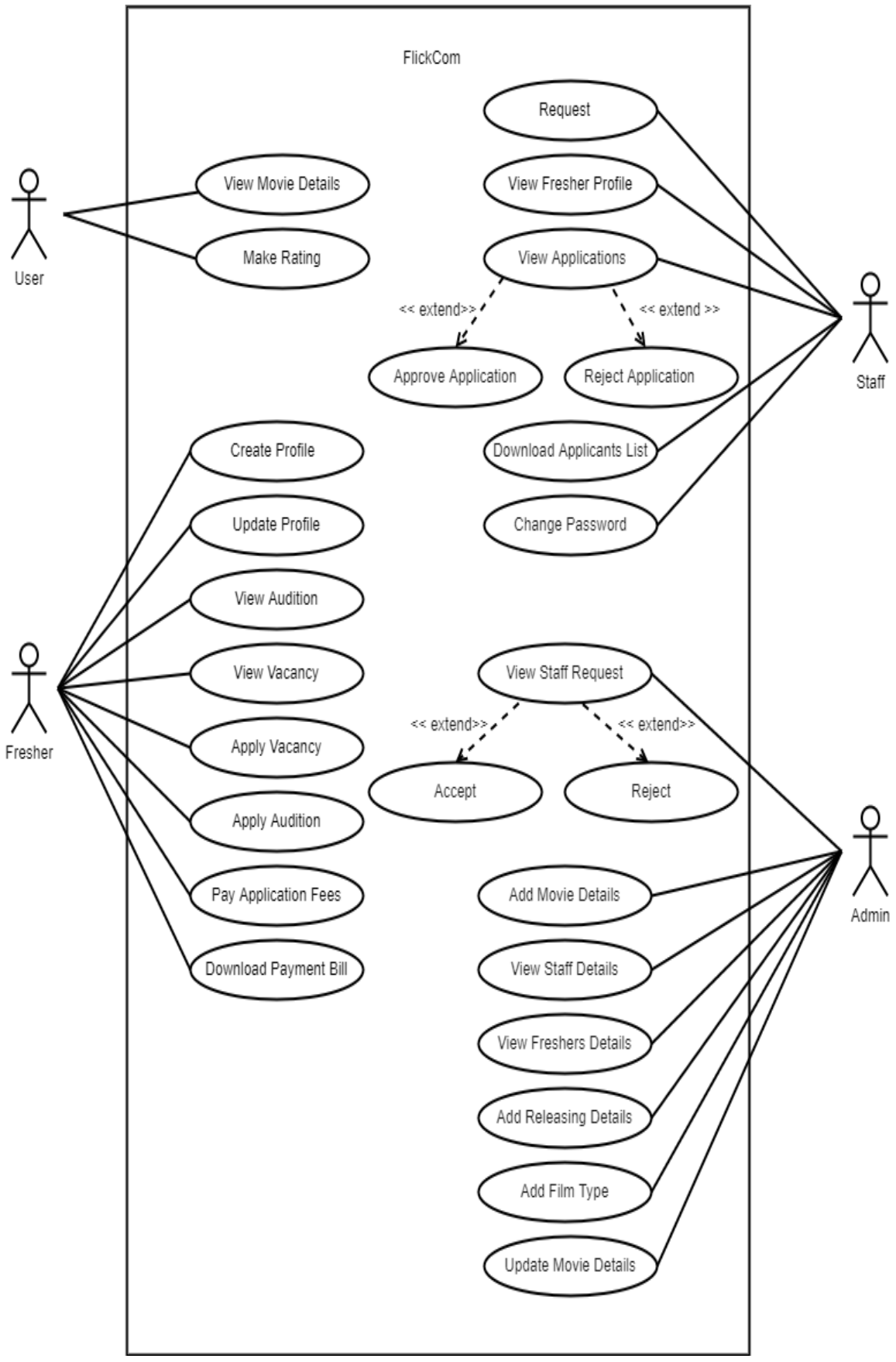
System objectives include things like organizing the overall requirements, validating a hardware design, testing and debugging a software product that is still in development, making an online help reference, or carrying out a task focused on customer service. For instance, use cases in a product sales context would be item ordering, catalogue updating, payment processing, and customer service. Four components make up a use case diagram.

- The line separating the system of interest from the environment around it.

- The actors are often participants in the system who are identified by their roles.
- The actors are often participants in the system who are identified by their roles.
- The connections and interdependencies between the actors and use cases.

Use case diagrams are used to capture a system's functional requirements. After identifying the items listed above, we must follow the guidelines below to create an effective use case diagram.

- A use case's naming is very significant. The name should be selected in a way that makes it clear what functions are being performed.
- Give the actors names that fit them.
- Clearly depict links and dependencies in the diagram.
- Keep in mind that the diagram's primary function is to indicate the needs; do not attempt to include all possible relationships.
- When necessary, take notes to help you remember some crucial details.


FlickCom
Request
View Movie Details
View Fresher Profile
User
Make Rating
View Applications
<< extend>>
<< extend >>
Approve Application
Reject Application
Staff
Create Profile
Download Applicants List
Update Profile
Change Password
View Audition
View Vacancy
View Staff Request
Fresher
<< extend>>
<< extend>>
Apply Vacancy
Accept
Reject
Apply Audition
Admin
Pay Application Fees
Add Movie Details
Download Payment Bill
View Staff Details
View Freshers Details
Add Releasing Details
Add Film Type
Update Movie Details

4.2.2 SEQUENCE DIAGRAM

A sequence diagram simply depicts object interactions in a sequential order, i.e. the order in which these interactions occur. A sequence diagram can also be referred to using the terms event diagrams or event scenarios. Sequence diagrams show how and in what order objects in a system work. These diagrams are commonly used by businesspeople and software developers to document and comprehend requirements for both new and current systems.

Sequence Diagram Notations –

- **Actors –** An actor in a UML diagram indicates a certain kind of role that communicates with the system and its objects. An actor is always outside the purview of the system that we are seeking to depict with the UML diagram, which is a crucial point to remember. Actors take on a range of roles, including those of external subjects and human users. In a UML diagram, an actor is represented by a stick person. There may be several actors in a sequence diagram.
- **Lifelines –** A lifeline is a named element that stands in for a specific participant in a sequence diagram. Each incident in a sequence diagram is represented by a lifeline. The top of a sequence diagram contains the lifeline components.
- **Messages –** Messages are used to represent inter-object communication. The lifeline displays the messages in reverse chronological order. Arrows are used to represent messages. The construction of a sequence diagram revolves around messages and lifelines.

  The following categories serve as general classifications for messages:

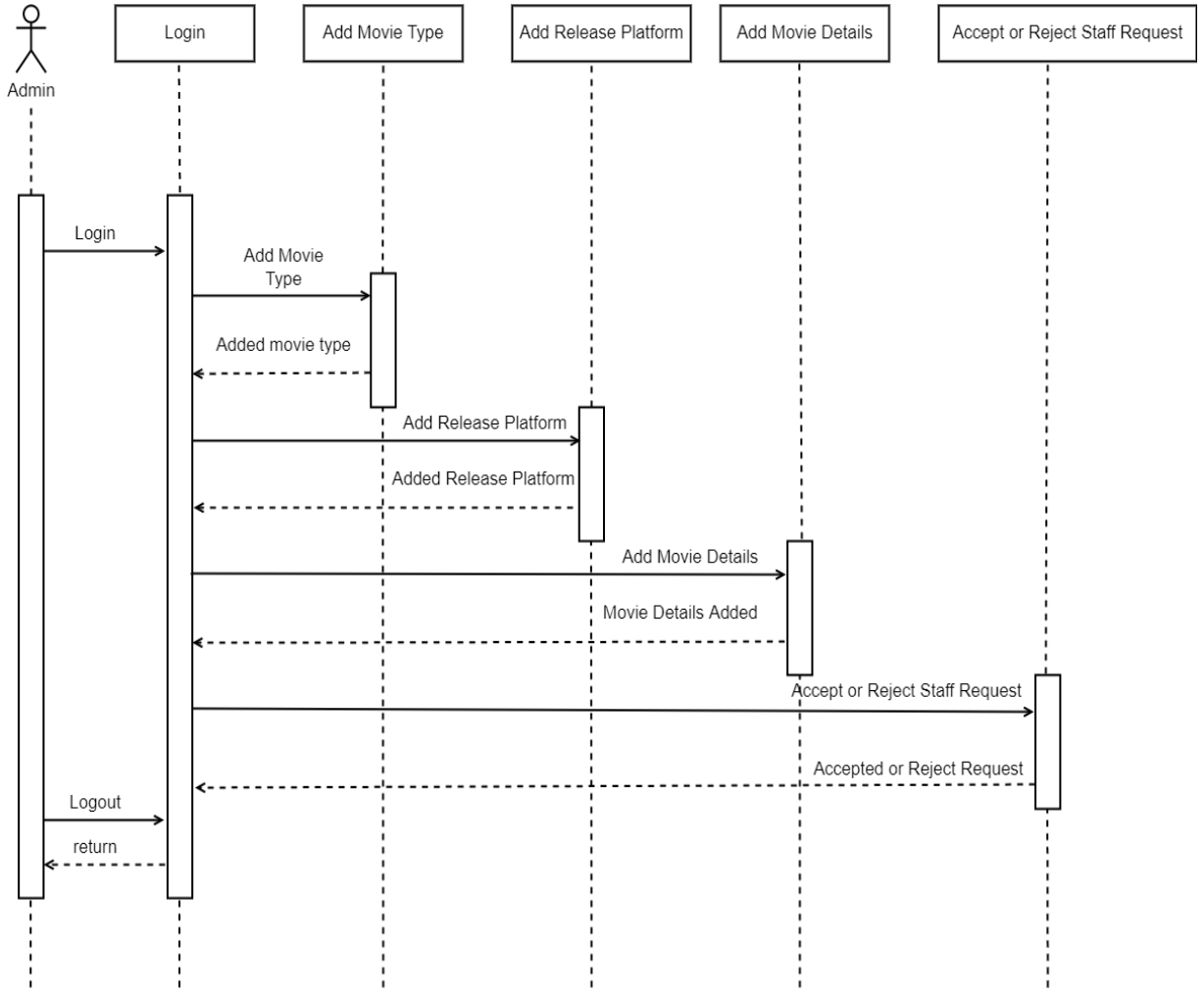
  - Synchronous messages
  - Asynchronous Messages
  - Create message
  - Delete Message

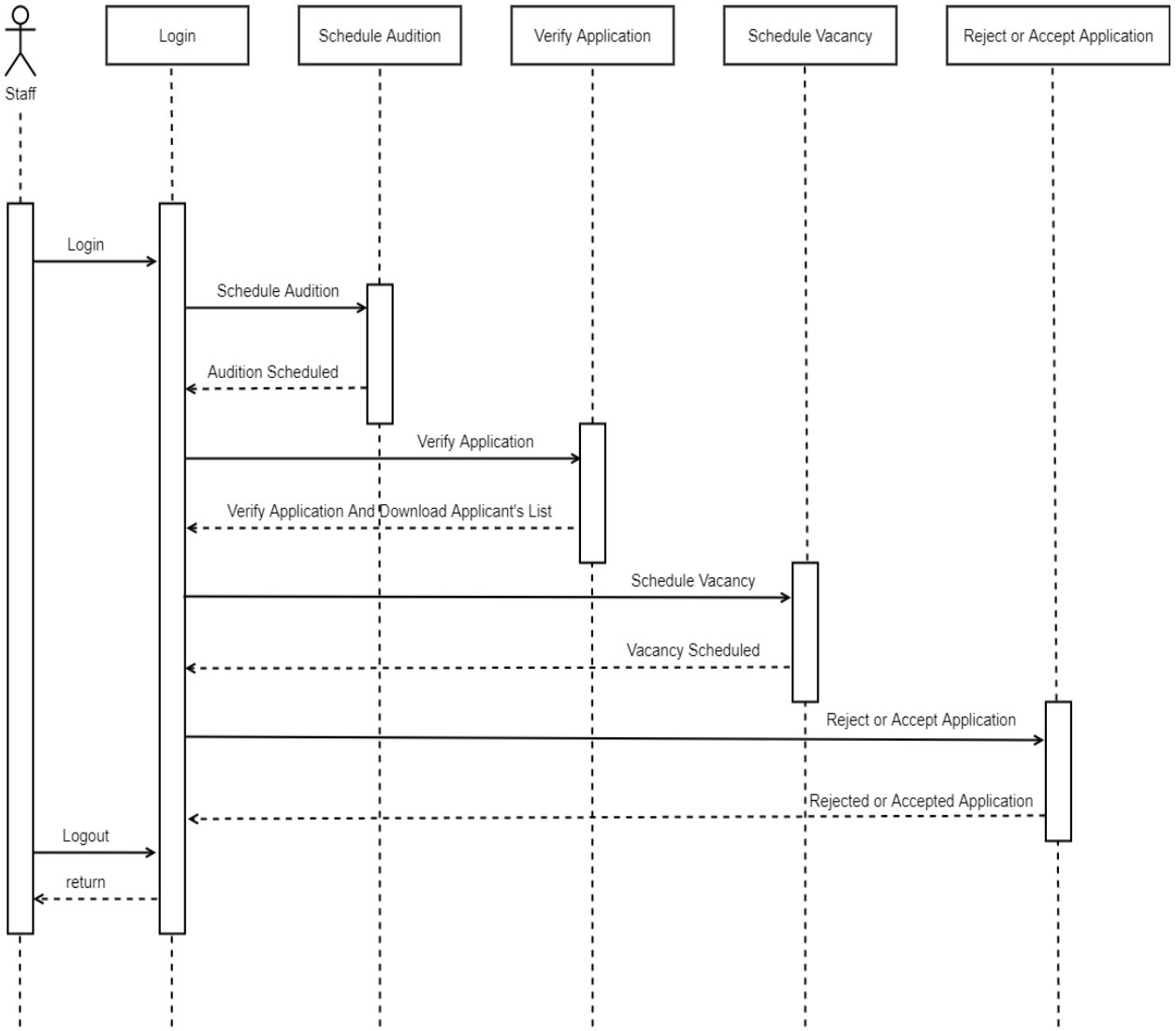
- Self-Message
- Reply Message
- Found Message
- Lost Message

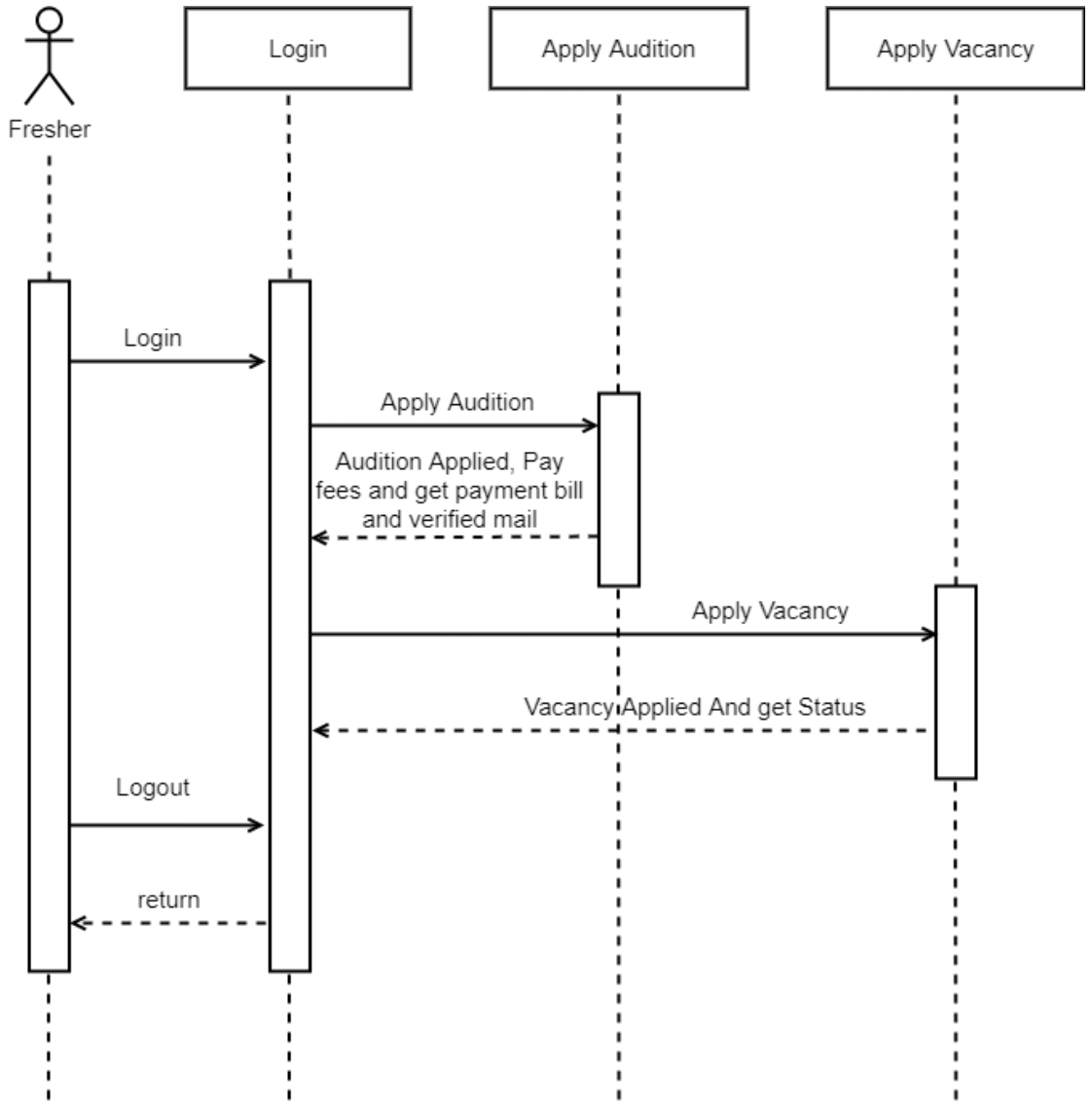
- **Guards –** In UML, we use guards to model conditions. They are employed to stop the flow of communications while pretending that a requirement has been satisfied. Software engineers rely on guards to warn them of the limitations imposed by a system or particular process.

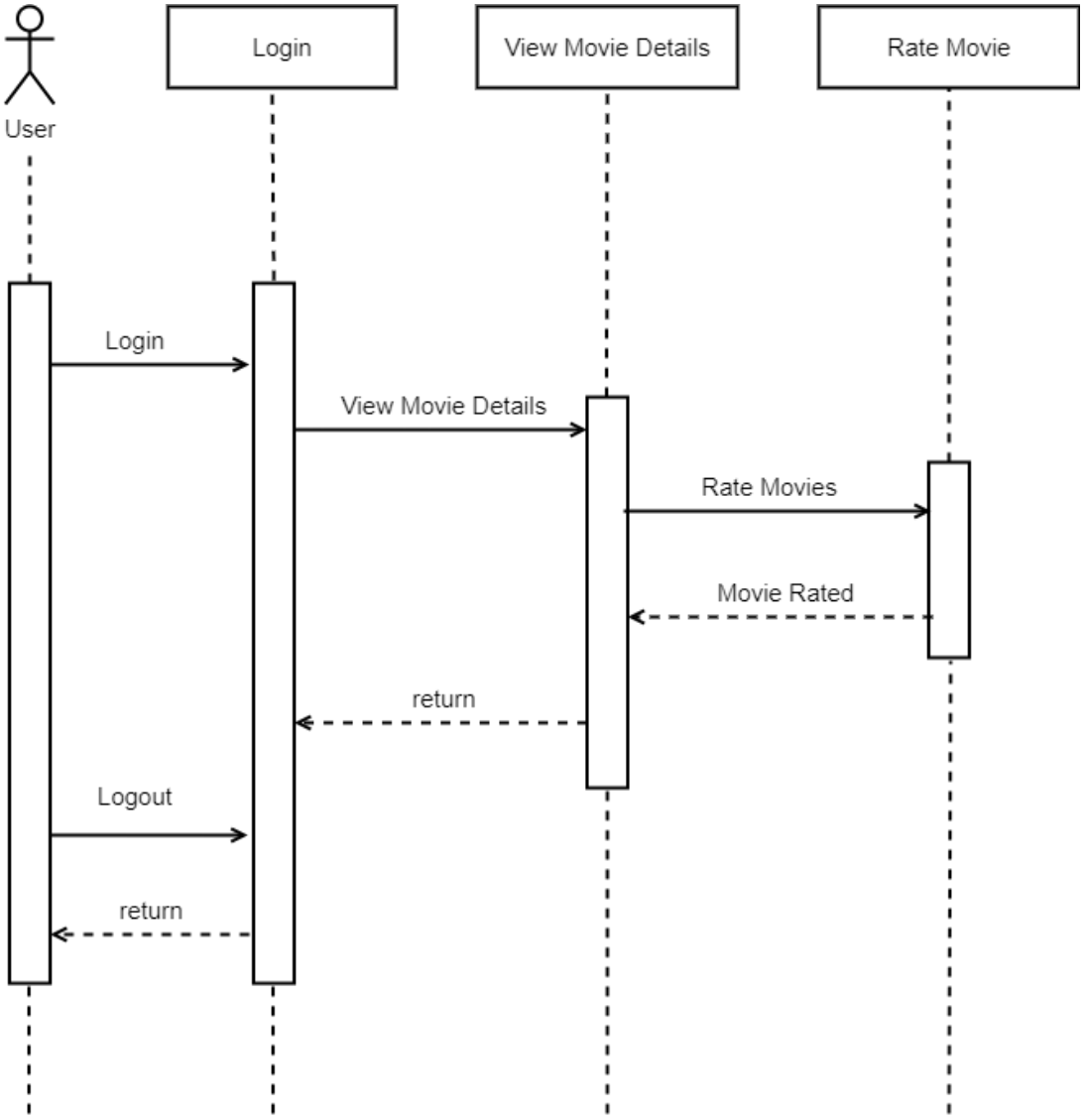
Uses of sequence diagrams –

- Used to represent and illustrate the reasoning behind a complex function, process, or procedure.
- They also serve to display the specifics of UML use case diagrams.
- Used to comprehend the precise operation of present or upcoming systems.
- Imagine the flow of tasks and messages between various objects or components.


Admin
Login
Add Movie Type
Add Release Platform
Add Movie Details
Accept or Reject Staff Request
Login
Add Movie Type
Added movie type
Add Release Platform
Added Release Platform
Add Movie Details
Movie Details Added
Accept or Reject Staff Request
Accepted or Reject Request
Logout
return

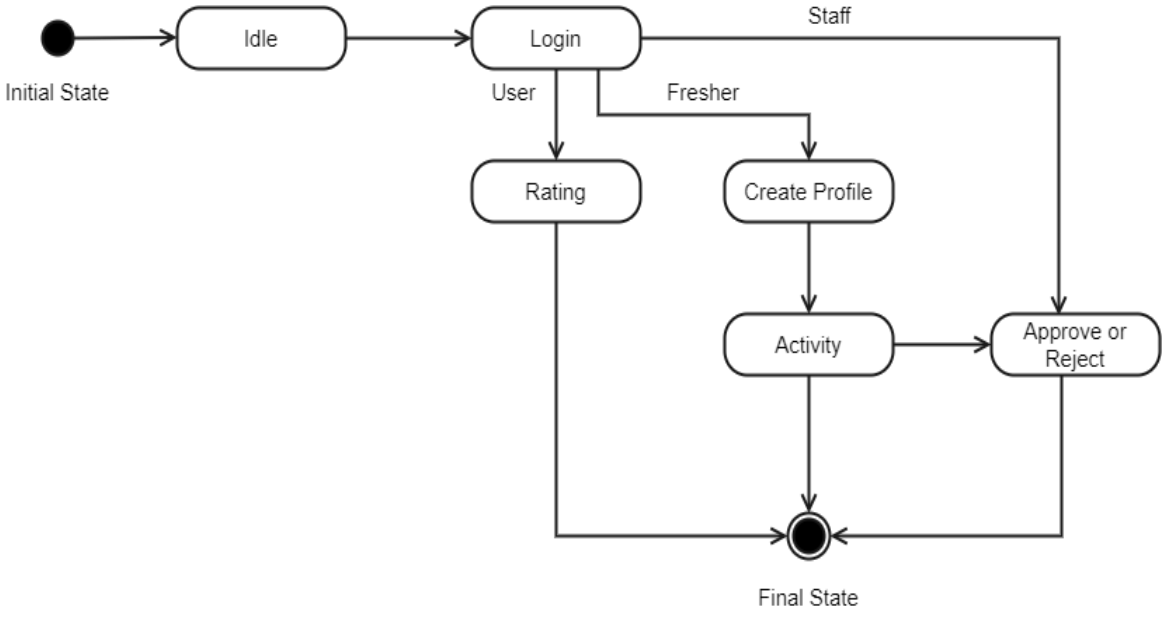

Staff
Login
Schedule Audition
Verify Application
Schedule Vacancy
Reject or Accept Application
Login
Schedule Audition
Audition Scheduled
Verify Application
Verify Application And Download Applicant's List
Schedule Vacancy
Vacancy Scheduled
Reject or Accept Application
Rejected or Accepted Application
Logout
return


Fresher
Login
Apply Audition
Apply Vacancy
Login
Apply Audition
Audition Applied, Pay fees and get payment bill and verified mail
Apply Vacancy
Vacancy Applied And get Status
Logout
return


User
Login
View Movie Details
Rate Movie
Login
View Movie Details
Rate Movies
Movie Rated
return
Logout
return

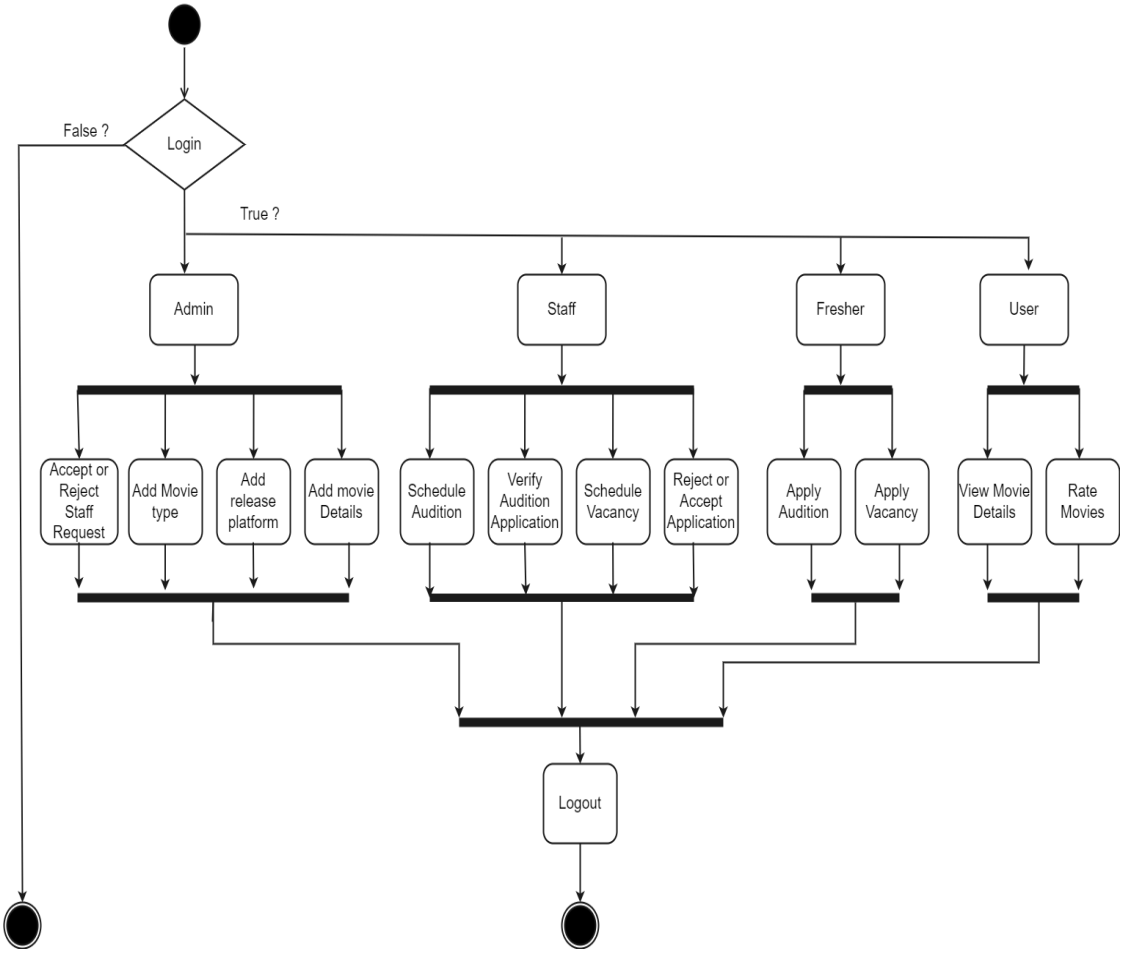
4.2.3 State Chart Diagram

A state diagram is used to represent the behavior of a software system. State machine diagrams in UML can be used to model the behavior of a class, a subsystem, a package, or an entire system. It's also referred to as a state chart or a state transition diagram. State chart diagrams are an efficient way to model the interactions or communication that occur between external entities and within a system. The event-based system is represented by these diagrams. An event is used to control the state of an object. In application systems, state chart diagrams are used to depict the multiple states of an entity.



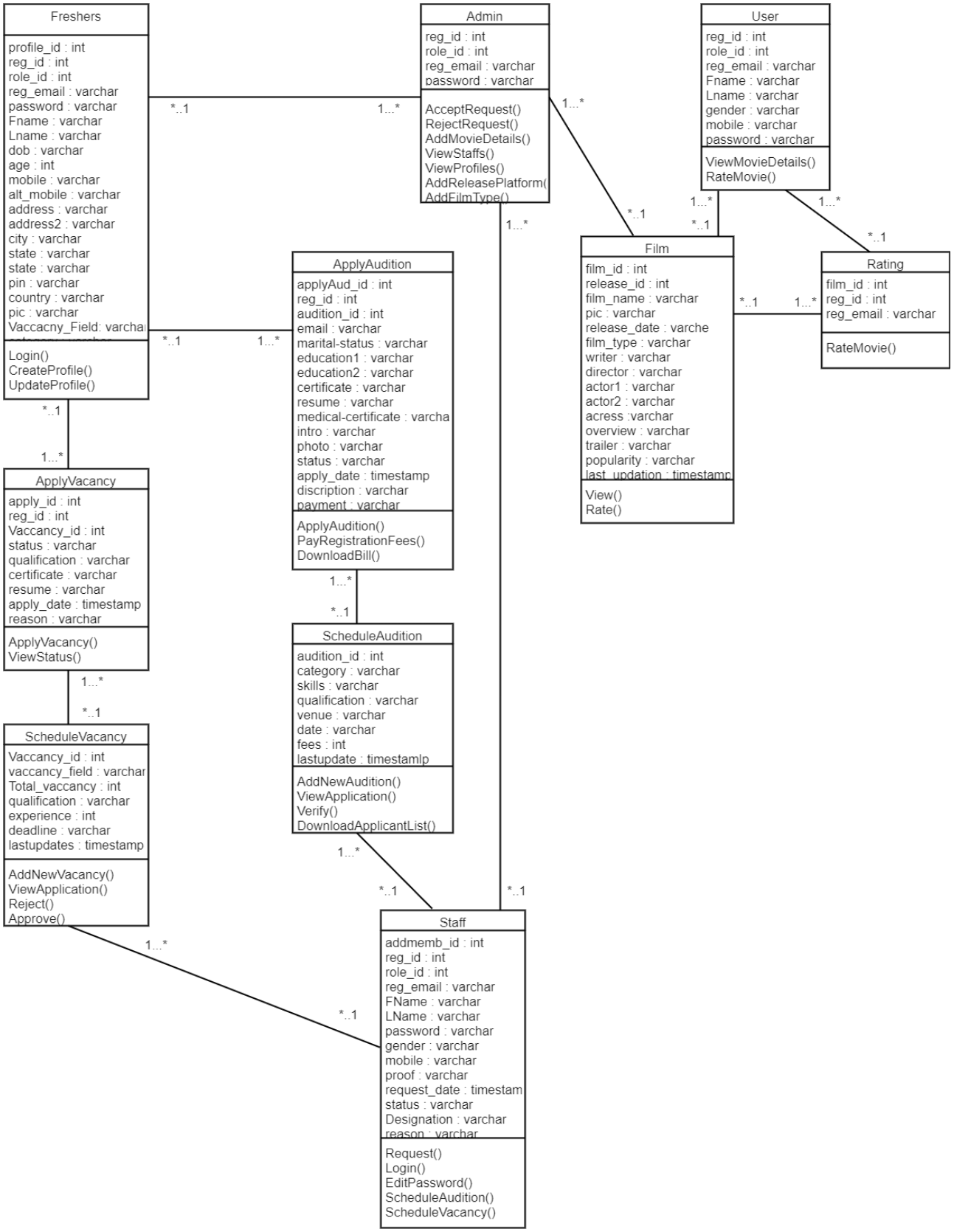
4.2.4 Activity Diagram

At different levels of abstraction, activity diagrams show how activities are organized to create a service. A typical event must be completed by some operations, especially when the operation is intended to complete several distinct tasks that call for coordination. Another typical requirement is how the events in a single use case relate to one another, especially in use cases where activities may overlap and require coordination.



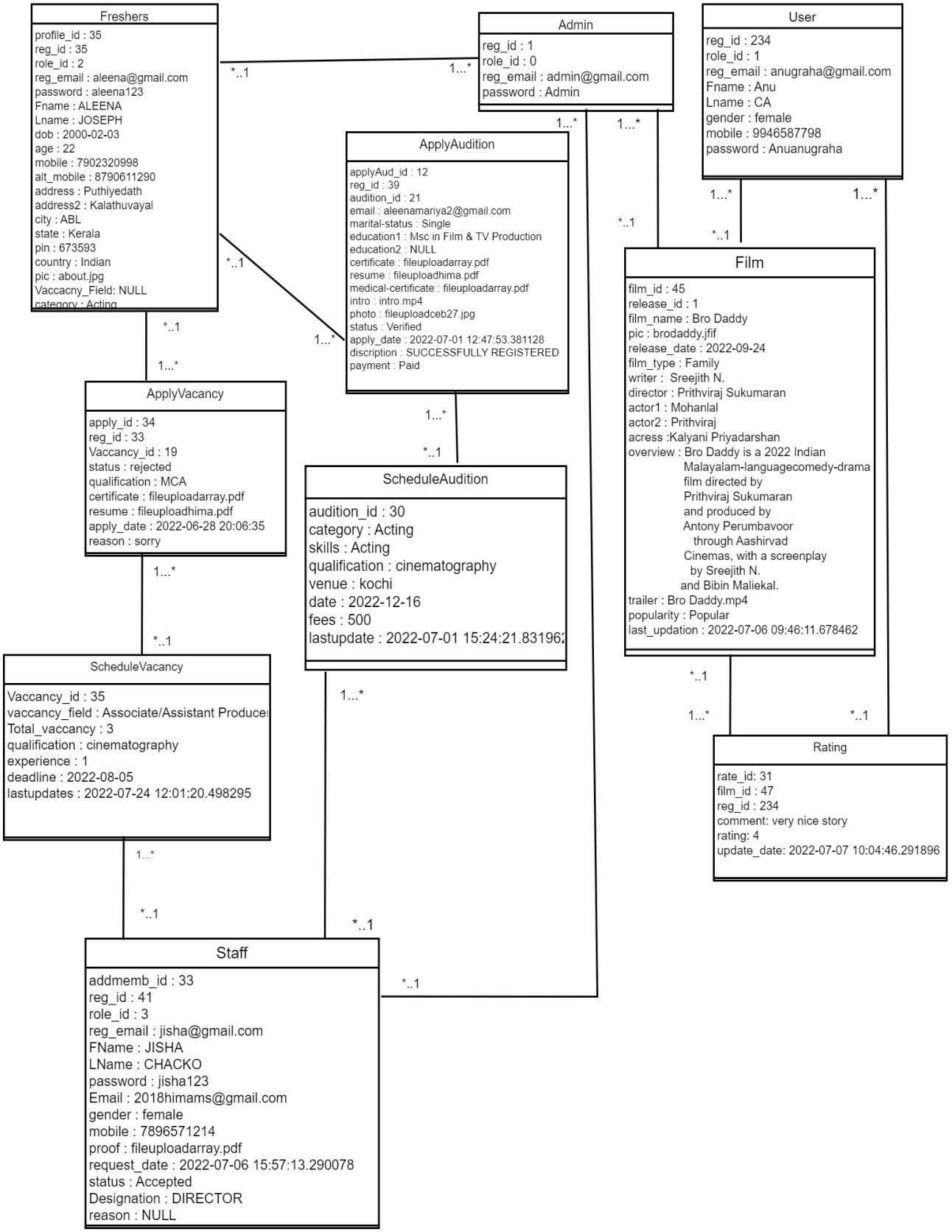
4.2.5 Class Diagram

Class diagrams are static diagrams. It depicts the static view of an application. A class diagram can be used to create executable code for software applications in addition to visualizing, explaining, and documenting different parts of a system. A class diagram outlines a class's characteristics, methods, and restrictions on the system. Class diagrams are frequently used in the modelling of object-oriented systems because they are the only UML diagrams that can be directly mapped to object-oriented languages. A collection of classes, interfaces, affiliations, collaborations, and constraints are represented visually in a class diagram. An alternative term for it is a structural diagram.


Freshers
profile_id : int
reg_id : int
role_id : int
reg_email : varchar
password : varchar
Fname : varchar
Lname : varchar
dob : varchar
age : int
mobile : varchar
alt_mobile : varchar
address : varchar
address2 : varchar
city : varchar
state : varchar
state : varchar
pin : varchar
country : varchar
pic : varchar
Vaccacny_Field: varcha
Login()
CreateProfile()
UpdateProfile()
Admin
reg_id : int
role_id : int
reg_email : varchar
password : varchar
AcceptRequest()
RejectRequest()
AddMovieDetails()
ViewStaffs()
ViewProfiles()
AddReleasePlatform(
AddFilmType()
User
reg_id : int
role_id : int
reg_email : varchar
Fname : varchar
Lname : varchar
gender : varchar
mobile : varchar
password : varchar
ViewMovieDetails()
RateMovie()
Film
film_id : int
release_id : int
film_name : varchar
pic : varchar
release_date : varche
film_type : varchar
writer : varchar
director : varchar
actor1 : varchar
actor2 : varchar
acress :varchar
overview : varchar
trailer : varchar
popularity : varchar
last_updation : timestamp
View()
Rate()
Rating
film_id : int
reg_id : int
reg_email : varchar
RateMovie()
ApplyAudition
applyAud_id : int
reg_id : int
audition_id : int
email : varchar
marital-status : varchar
education1 : varchar
education2 : varchar
certificate : varchar
resume : varchar
medical-certificate : varcha
intro : varchar
photo : varchar
status : varchar
apply_date : timestamp
discription : varchar
payment : varchar
ApplyAudition()
PayRegistrationFees()
DownloadBill()
ApplyVacancy
apply_id : int
reg_id : int
Vaccancy_id : int
status : varchar
qualification : varchar
certificate : varchar
resume : varchar
apply_date : timestamp
reason : varchar
ApplyVacancy()
ViewStatus()
ScheduleAudition
audition_id : int
category : varchar
skills : varchar
qualification : varchar
venue : varchar
date : varchar
fees : int
lastupdate : timestamp
AddNewAudition()
ViewApplication()
Verify()
DownloadApplicantList()
ScheduleVacancy
Vaccancy_id : int
vaccancy_field : varchar
Total_vaccancy : int
qualification : varchar
experience : int
deadline : varchar
lastupdates : timestamp
AddNewVacancy()
ViewApplication()
Reject()
Approve()
Staff
addmemb_id : int
reg_id : int
role_id : int
reg_email : varchar
FName : varchar
LName : varchar
password : varchar
gender : varchar
mobile : varchar
proof : varchar
request_date : timestam
status : varchar
Designation : varchar
reason : varchar
Request()
Login()
EditPassword()
ScheduleAudition()
ScheduleVacancy()

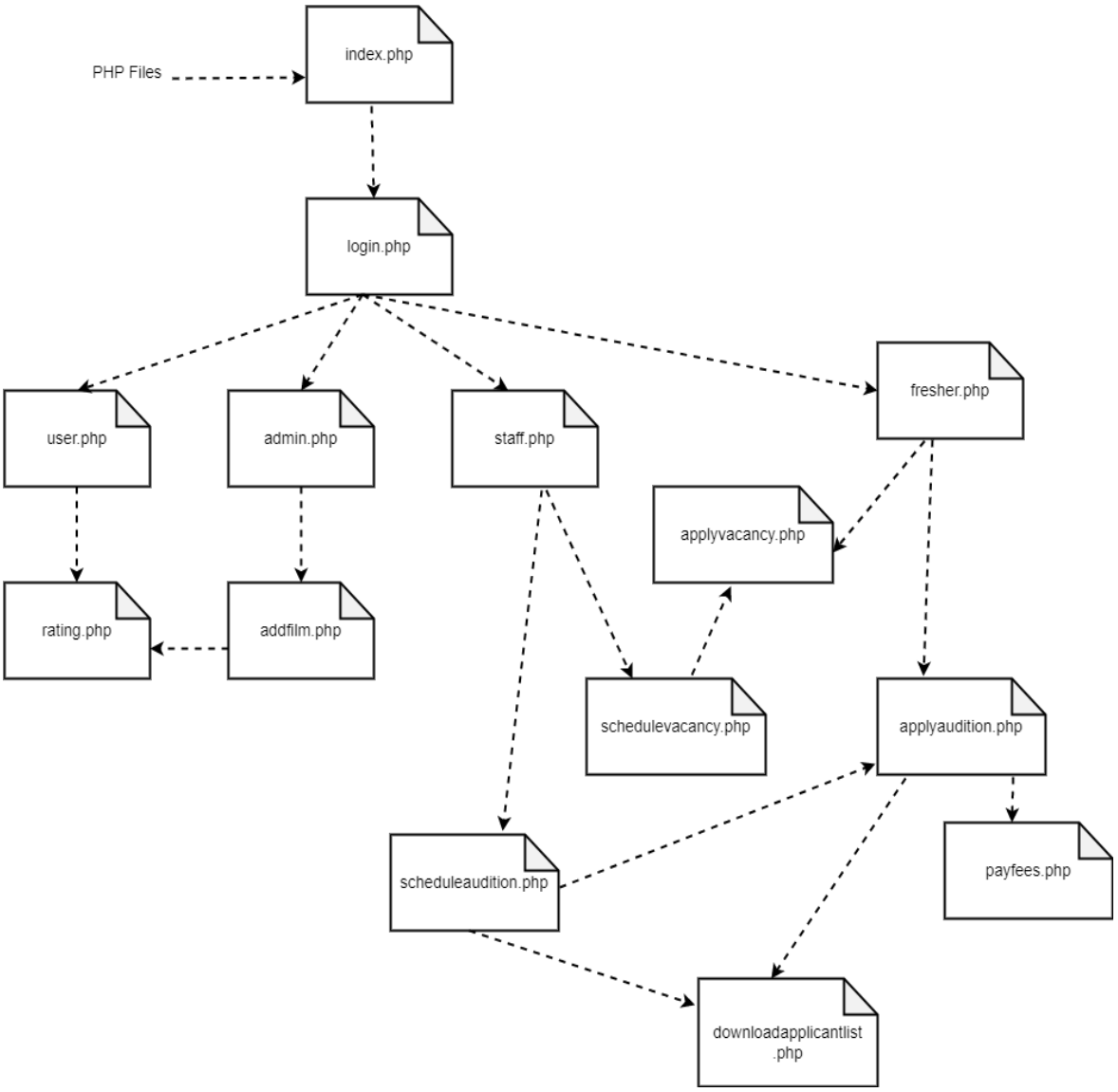
4.2.6 Object Diagram

Object diagrams are dependent on class diagrams because they are generated from them. An object diagram is a visual representation of a class diagram. Class and object diagrams share the same essential ideas. The static view of a system is also represented by object diagrams, but this static view represents a moment in time snapshot of the system. To represent a group of things and their connections, object diagrams are employed.


Freshers
profile_id : 35
reg_id : 35
role_id : 2
reg_email : aleena@gmail.com
password : aleena123
Fname : ALEENA
Lname : JOSEPH
dob : 2000-02-03
age : 22
mobile : 7902320998
alt_mobile : 8790611290
address : Puthiyedath
address2 : Kalathuvayal
city : ABL
state : Kerala
pin : 673593
country : Indian
pic : about.jpg
Vaccacny_Field: NULL
category : Acting
Admin
reg_id : 1
role_id : 0
reg_email : admin@gmail.com
password : Admin
User
reg_id : 234
role_id : 1
reg_email : anugraha@gmail.com
Fname : Anu
Lname : CA
gender : female
mobile : 9946587798
password : Anuanugraha
ApplyAudition
applyAud_id : 12
reg_id : 39
audition_id : 21
email : aleenamariya2@gmail.com
marital-status : Single
education1 : Msc in Film & TV Production
education2 : NULL
certificate : fileuploadarray.pdf
resume : fileuploadhima.pdf
medical-certificate : fileuploadarray.pdf
intro : intro.mp4
photo : fileuploadceb27.jpg
status : Verified
apply_date : 2022-07-01 12:47:53.381128
discription : SUCCESSFULLY REGISTERED
payment : Paid
Film
film_id : 45
release_id : 1
film_name : Bro Daddy
pic : brodaddy.jfif
release_date : 2022-09-24
film_type : Family
writer : Sreejith N.
director : Prithviraj Sukumaran
actor1 : Mohanlal
actor2 : Prithviraj
acress :Kalyani Priyadarshan
overview : Bro Daddy is a 2022 Indian Malayalam-languagecomedy-drama film directed by Prithviraj Sukumaran and produced by Antony Perumbavoor through Aashirvad Cinemas, with a screenplay by Sreejith N. and Bibin Maliekal.
trailer : Bro Daddy.mp4
popularity : Popular
last_updation : 2022-07-06 09:46:11.678462
ApplyVacancy
apply_id : 34
reg_id : 33
Vaccancy_id : 19
status : rejected
qualification : MCA
certificate : fileuploadarray.pdf
resume : fileuploadhima.pdf
apply_date : 2022-06-28 20:06:35
reason : sorry
ScheduleAudition
audition_id : 30
category : Acting
skills : Acting
qualification : cinematography
venue : kochi
date : 2022-12-16
fees : 500
lastupdate : 2022-07-01 15:24:21.831962
ScheduleVacancy
Vaccancy_id : 35
vaccancy_field : Associate/Assistant Produce
Total_vaccancy : 3
qualification : cinematography
experience : 1
deadline : 2022-08-05
lastupdates : 2022-07-24 12:01:20.498295
Rating
rate_id: 31
film_id : 47
reg_id : 234
comment: very nice story
rating: 4
update_date: 2022-07-07 10:04:46.291896
Staff
addmemb_id : 33
reg_id : 41
role_id : 3
reg_email : jisha@gmail.com
FName : JISHA
LName : CHACKO
password : jisha123
Email : 2018himams@gmail.com
gender : female
mobile : 7896571214
proof : fileuploadarray.pdf
request_date : 2022-07-06 15:57:13.290078
status : Accepted
Designation : DIRECTOR
reason : NULL

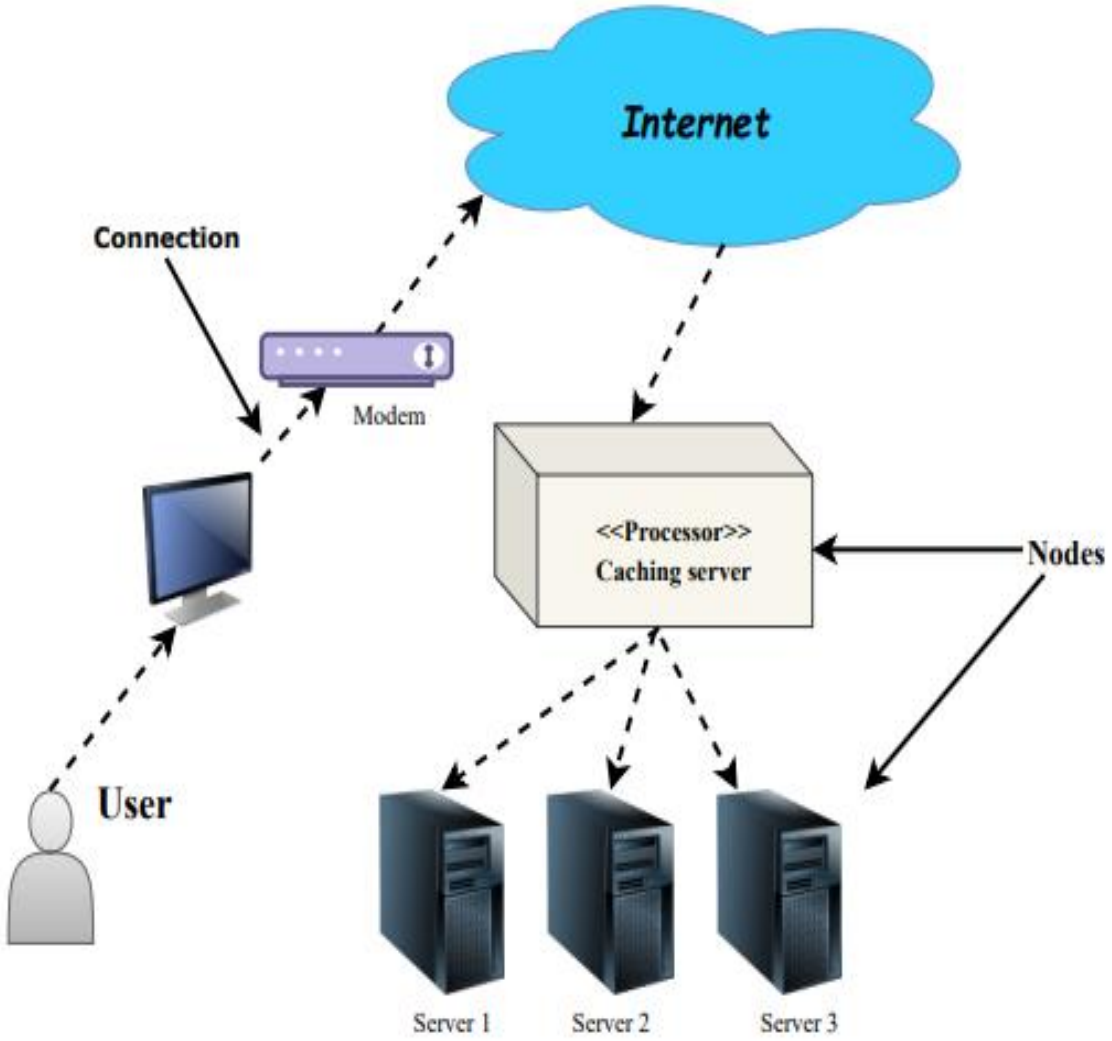
4.2.7 Component Diagram

Different component diagrams exhibit different characteristics. A system's physical properties are represented using component diagrams. The components that are physically present in a node include executables, libraries, files, documents, and so forth. Component diagrams are used to show how system components are arranged and related to one another. The development of executable systems also uses these representations.



4.2.8 Deployment Diagram

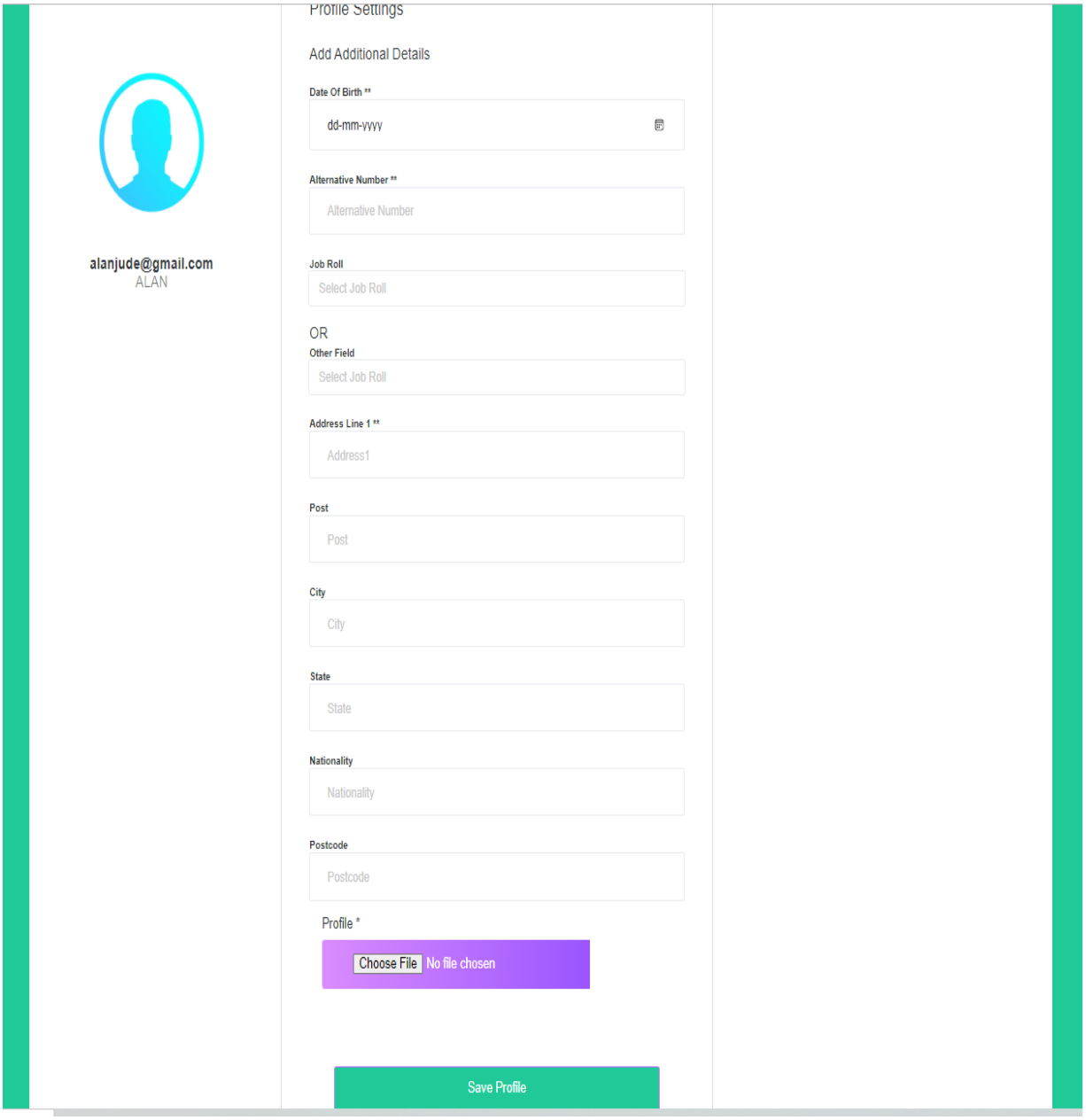
The topology of a system's physical components as well as the locations of its software components are shown in deployment diagrams. The static deployment view of a system is described using deployment diagrams. Diagrams for deployment show nodes and their connections.



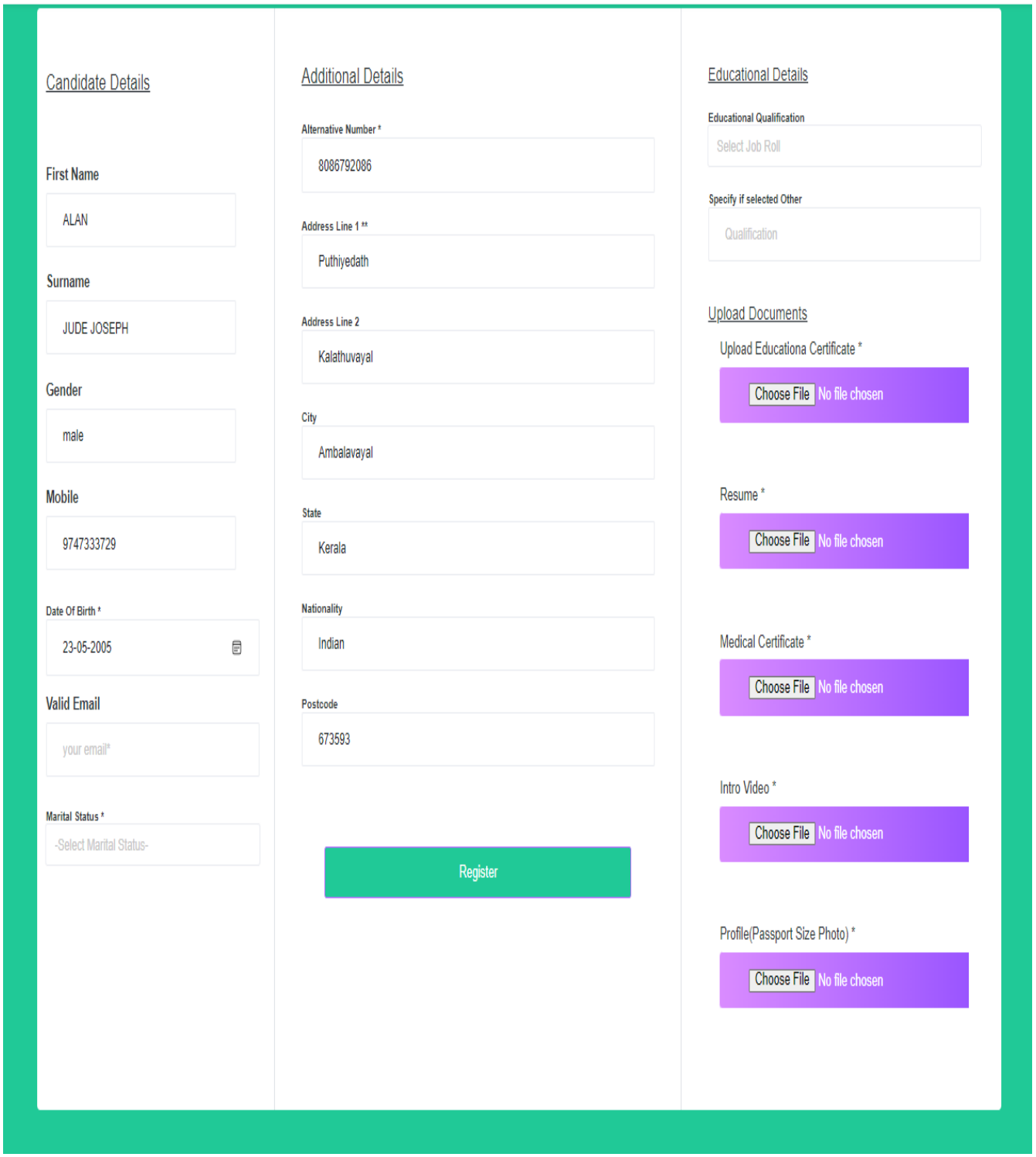
4.3 USER INTERFACE DESIGN

4.3.1 INPUT DESIGN

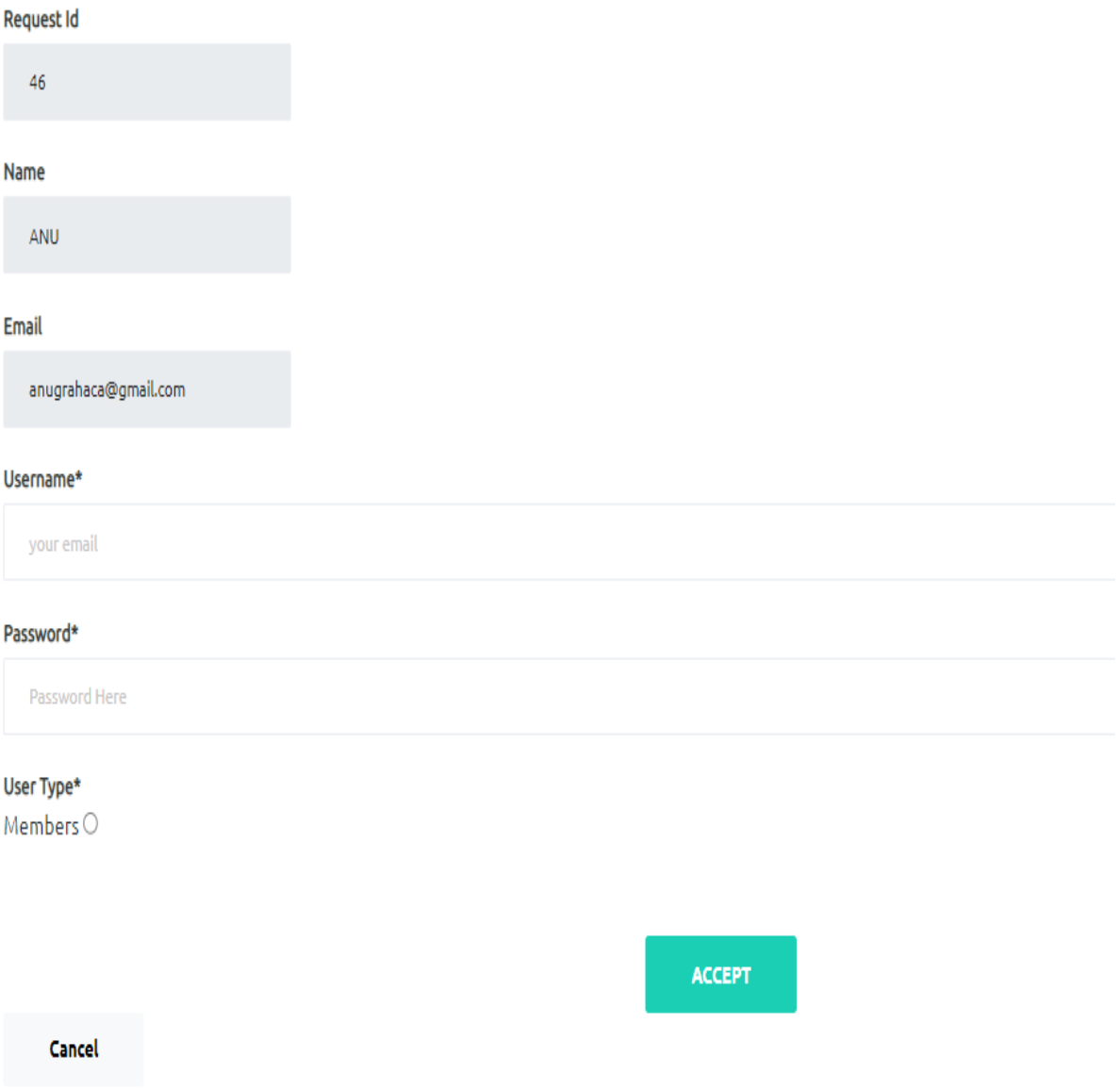
Form Name : Freshers Profile



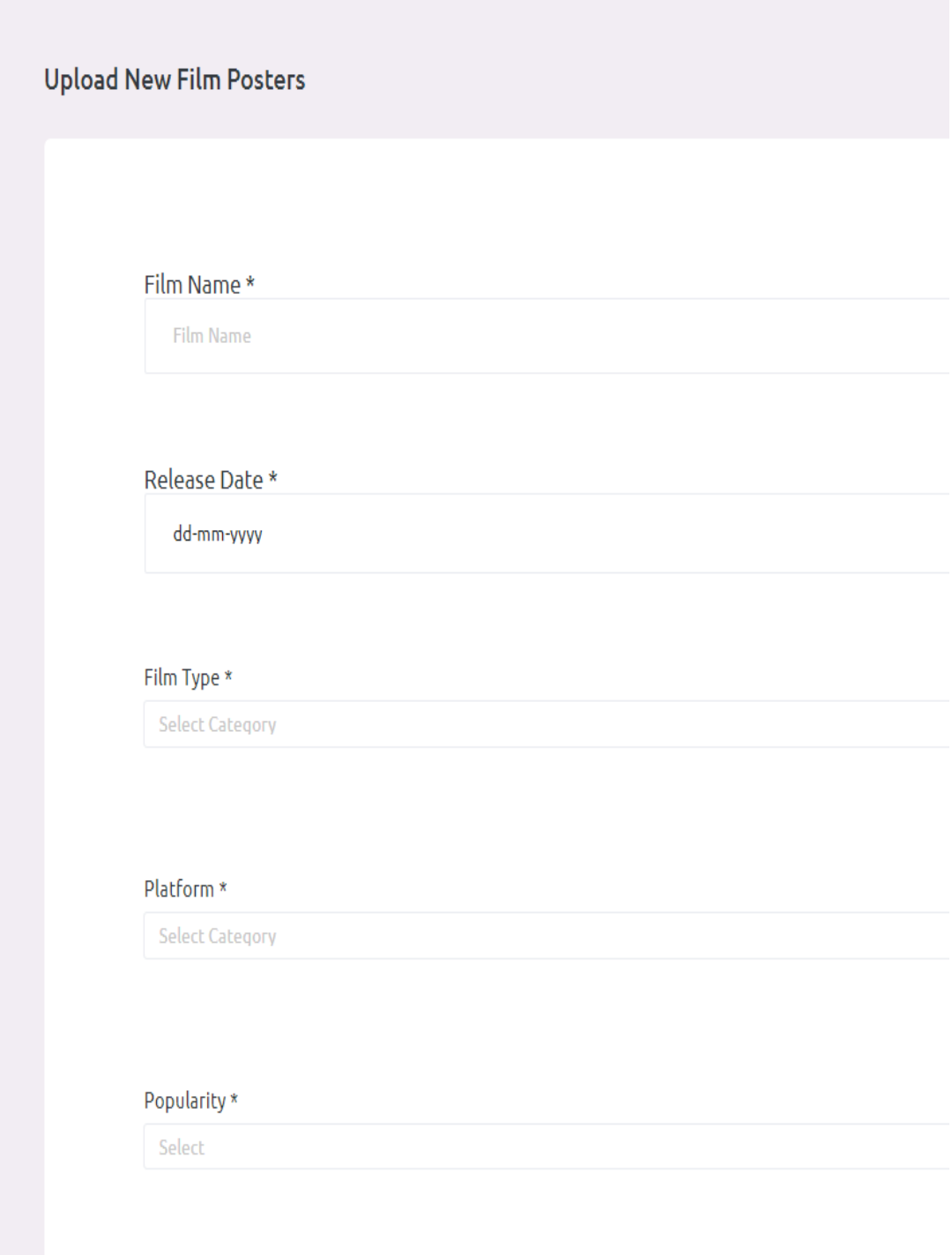
Form Name : Apply Audition

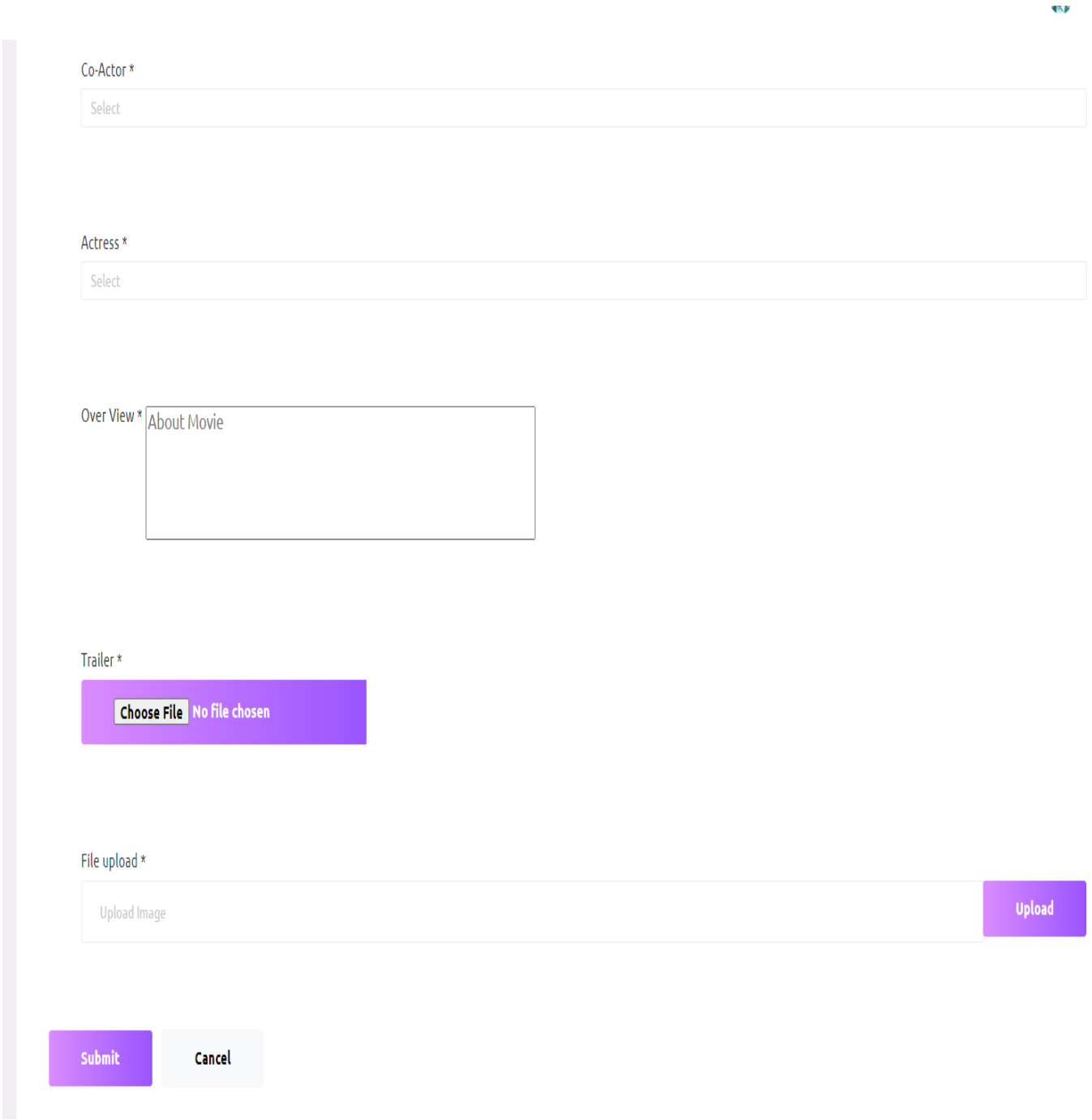


Form Name : Accept Staff request

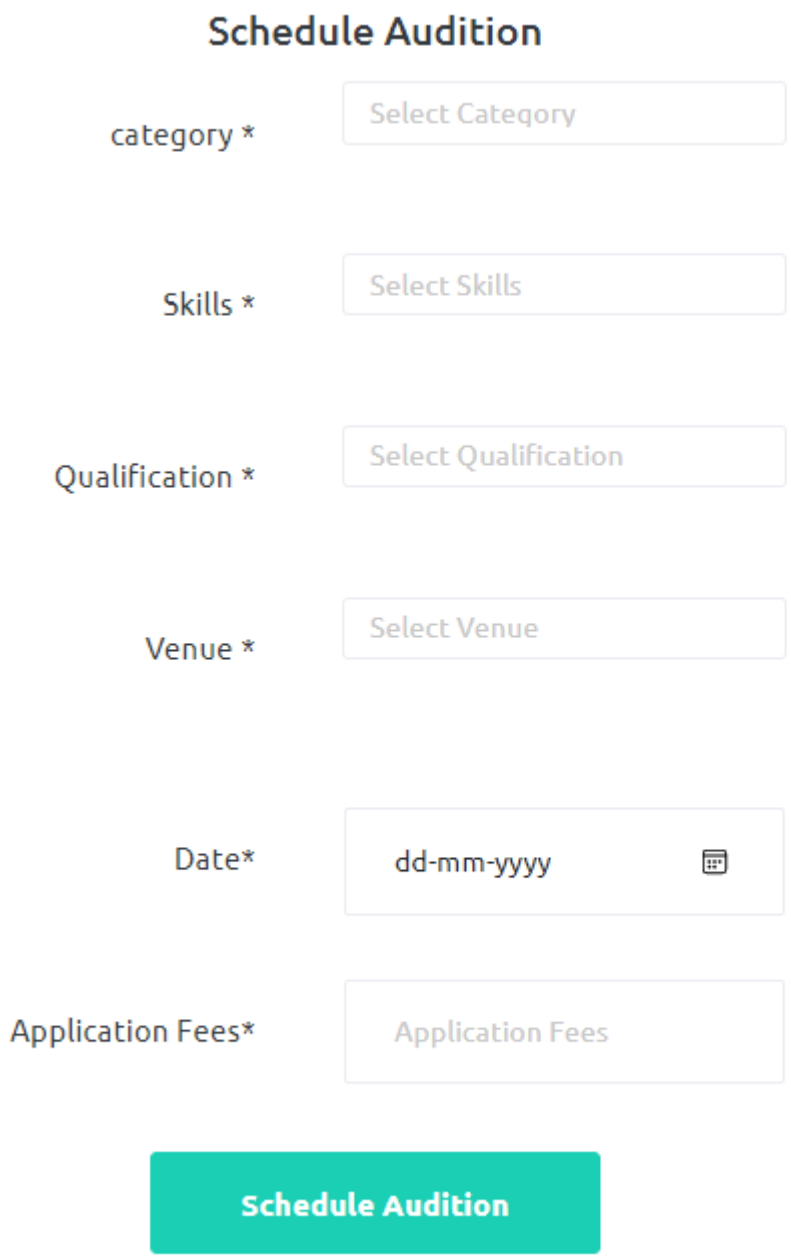


Form Name : Upload Film Details



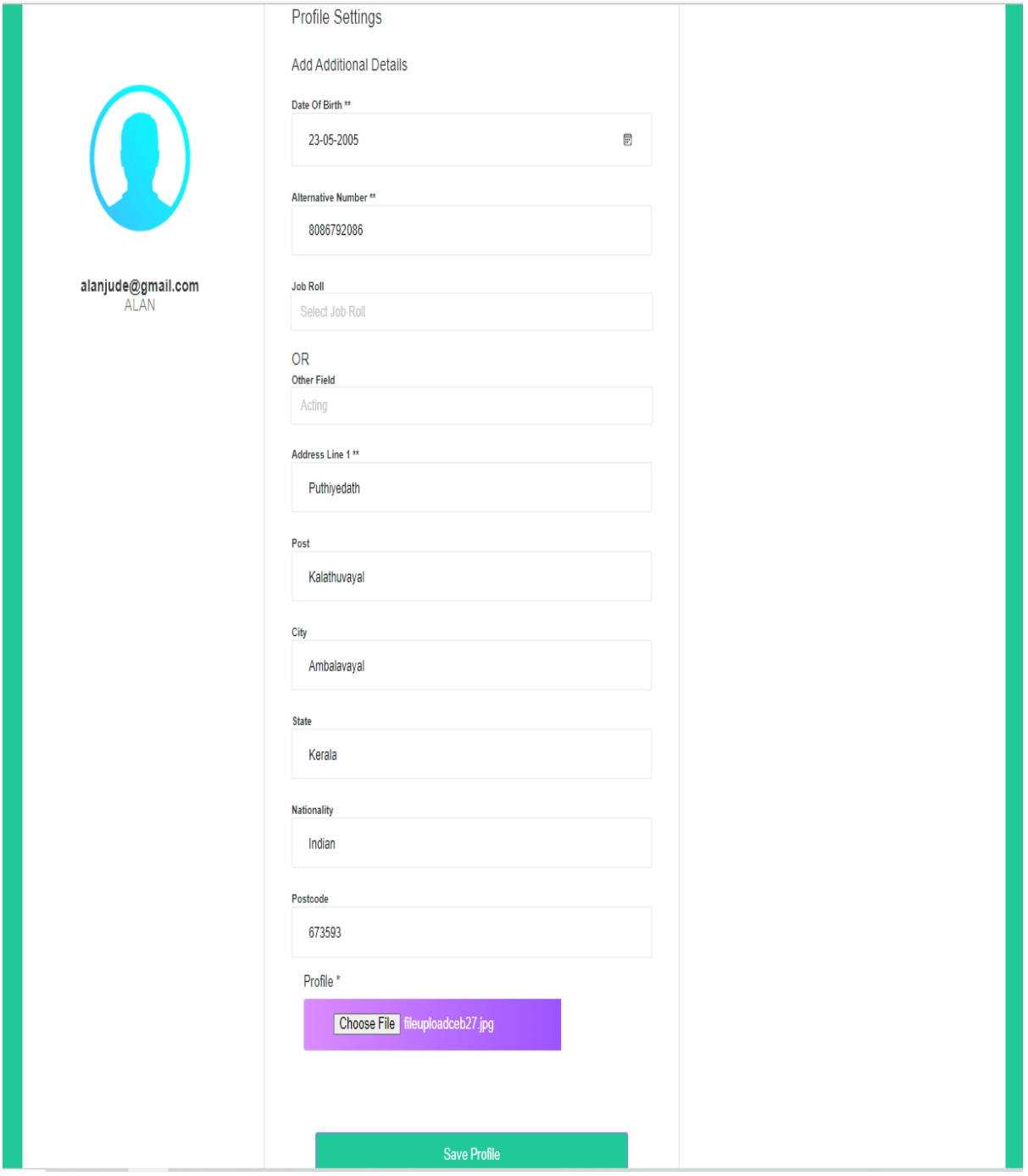

Co-Actor *
Select
Actress *
Select
Over View *
About Movie
Trailer *
Choose File
No file chosen
File upload *
Upload Image
Upload
Submit
Cancel

Form Name : Schedule Audition

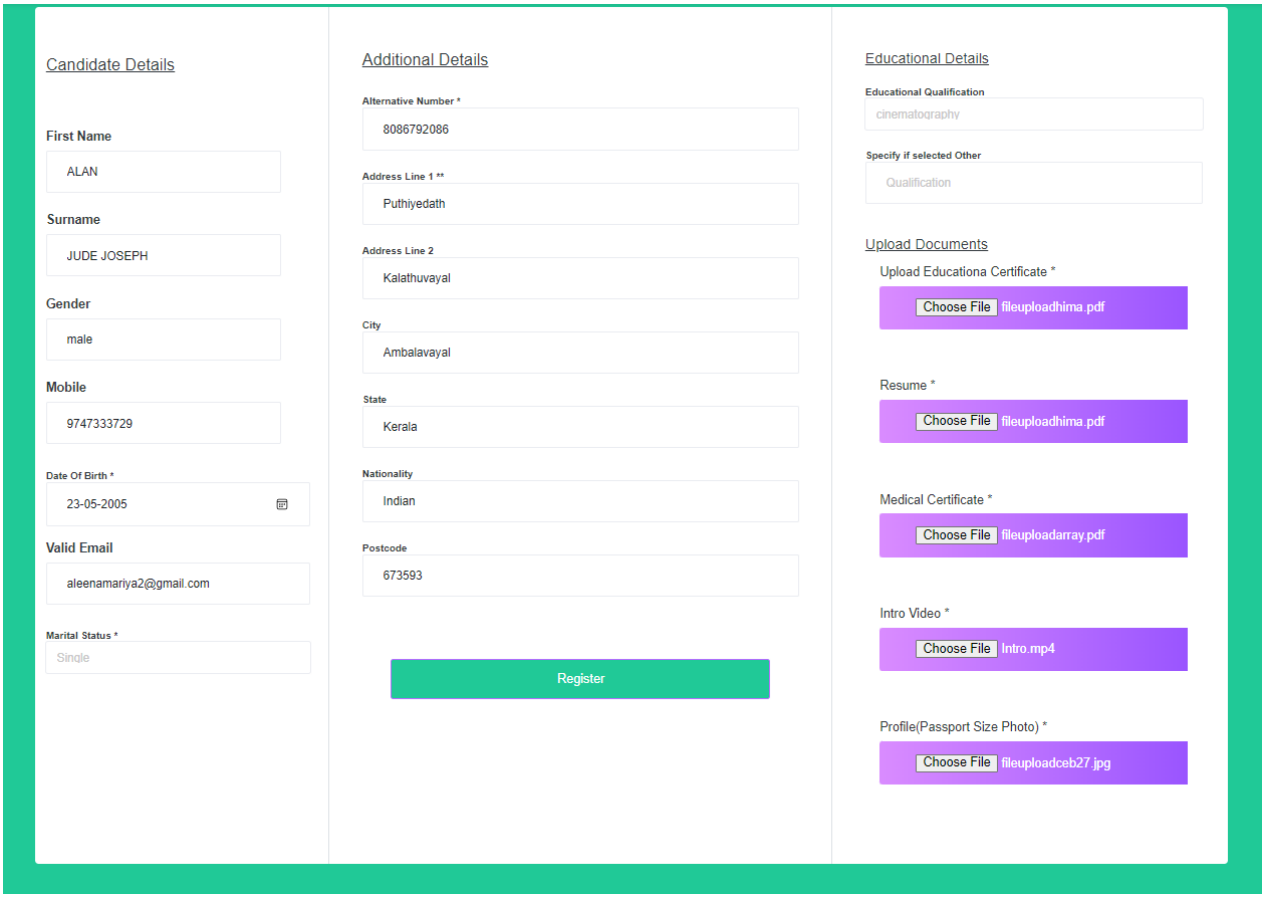


4.3.2 OUTPUT DESIGN

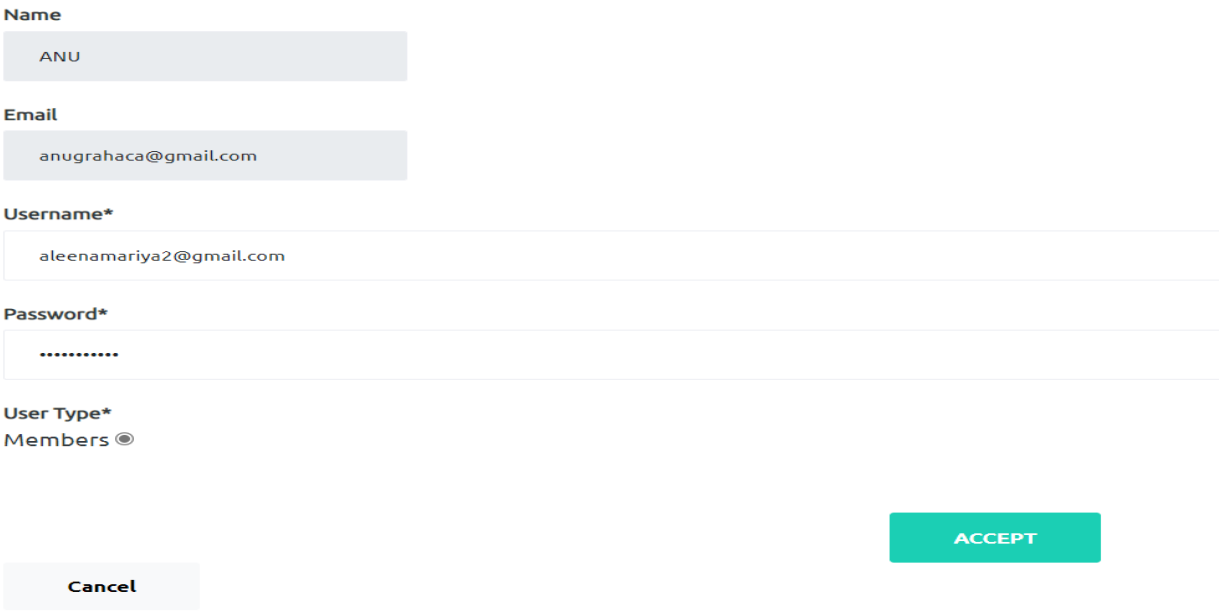
Freshers Profile



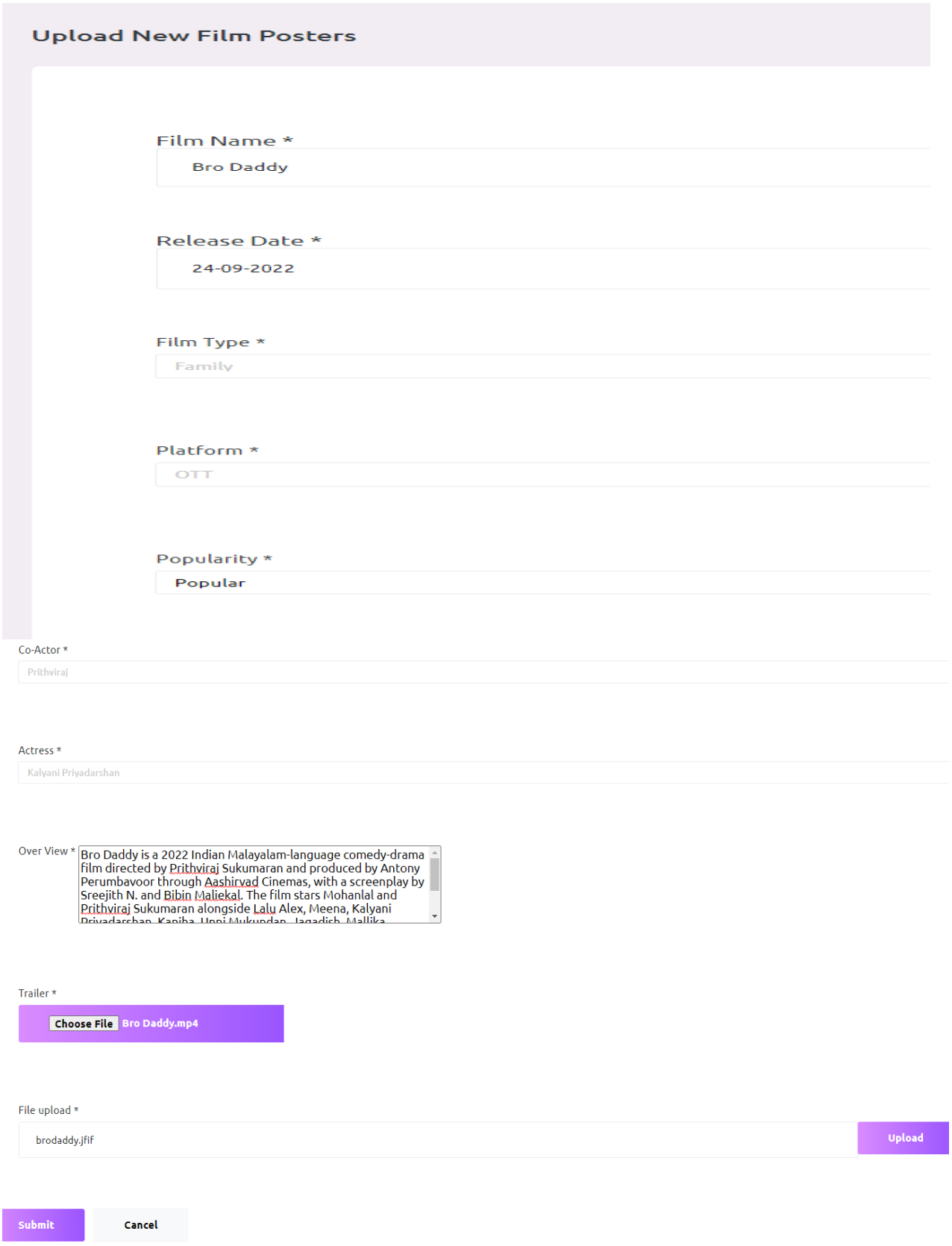
Apply Audition



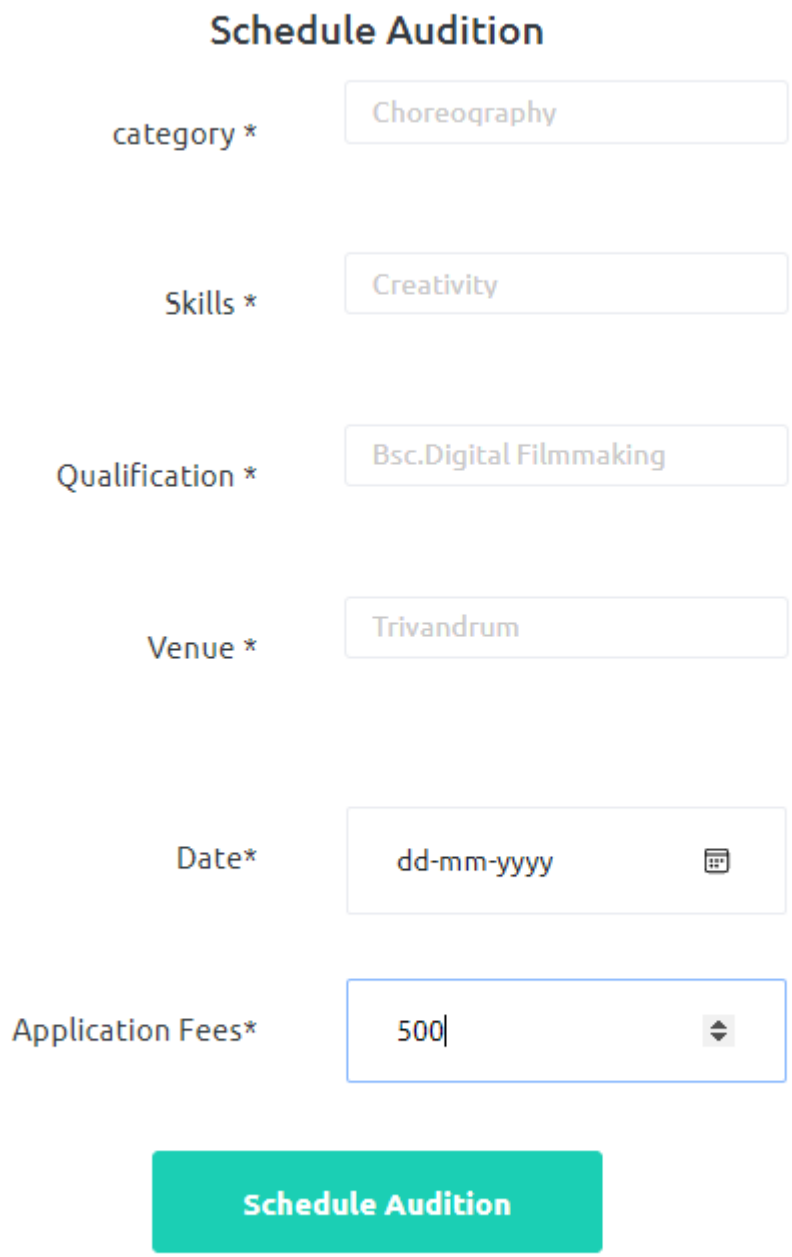
Accept Staff



Upload Film Details



Schedule Audition



4.4 DATABASE DESIGN

A database is a method for organizing information storage that makes it possible for users to retrieve stored data quickly and effectively. Any database's main function is to hold data, which must be protected.

The database design process is divided into two stages. The first step is to collect user requirements and design a database that meets these requirements as clearly as possible. This is known as Information Level Design, and it is performed independently of any specific DBMS.

The process of converting this information-level design into a design for the particular DBMS that will be used to implement the system in issue is the second stage. The properties of the DBMS that will be utilized are addressed in this phase, which is referred to as Physical Level Design. Concurrent with a system design is a database design. The two primary objectives described below are what the database's data organization intends to accomplish.

- Data Integrity
- Data independence

4.4.1 Relational Database Management System(RDBMS)

A relational model represents a database as a set of relationships. Each relationship is similar to a table of values or a file of records. In the formal language of the relational model, a row is known as a tuple, a column header as an attribute, and the table as a relation. A collection of tables, each with a unique name, make up a relational data base. In a tale, a row stands for a group of connected values.

Relations, Domains & Attributes

A table is a type of relation. A table's rows are referred to as tuples. A tuple is a set of n elements that is ordered. Columns are known as attributes. Every table in the database has relationships set up. This ensures the integrity of both the referential and entity relationships. Atomic value collections make up D domains. The selection of a data type from which the domain's data values are derived is a typical method of domain specification. The domain should have a name to make it easier to understand its values.

Since each value in a relation is atomic, it cannot be divided.

Relationships

- Key is used to create table relationships. Primary Key and Foreign Key are the two principal keys that are most crucial. With the use of these keys, relationships for entity integrity and referential integrity can be created.
- Entity Integrity forbids the use of null values for any Primary Key.
- No Primary Key may contain null values, according to Referential Integrity.
- Referential Integrity: A Primary Key value in the same domain must correspond to each unique Foreign Key value. Super Key and Candidate Keys are additional keys.

4.4.2 Normalization

To minimize the impact of any changes on data structures, data are organized as simply as feasible. The formal technique of structuring data structures in a way that minimizes redundancy and supports integrity is known as normalization. The process of normalization involves breaking up a big table into smaller ones and eliminating unnecessary fields. It also helps to prevent erroneous insertion, deletion, and updating. Keys and relationships are the two ideas that are typically employed in data modelling. A row in a table can only be uniquely identified by a key. The two sorts of keys are primary keys and foreign keys. A table element or combination of elements is known as a primary key and is used to distinguish between records in the same table. A table column known as a foreign key is one that specifically identifies records from another table. All the tables have been normalized up to the third normal form.

It refers to restoring anything to its original state, as the name suggests. Through normalizing, the application developer aims to produce a logical arrangement of the data into appropriate tables and columns with names that may be quickly associated with the data by the user. Normalization eliminates recurring data groups, preventing data redundancy, a big resource hog for computers. These include:

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

First Normal Form

The value of any attribute in a tuple must be a single value from that attribute's domain in accordance with the First Normal Form. An attribute's domain must only contain atomic values. In other words, "relations within relations" or "relations as attribute values within tuples" are not permitted in 1NF. For attribute values, 1NF only permits single atomic or indivisible values. The initial step is to convert the data to First Normal Form. This can be accomplished by separating data into separate tables with similar data types in each table. Each table is assigned a Primary Key or a Foreign Key based on the project's needs. For each non-atomic attribute or nested relation, we create a new relationship. Repeated data groupings were removed as a result. If just the constraints containing the primary key are satisfied, a relation is said to be in first normal form.

Second Normal Form

In relationships when the primary key comprises numerous attributes, no non-key attribute should be functionally dependent on a portion of the primary key, according to Second Normal Form. We decompose and create a new relation for each partial key and its dependent attributes in this step. Keep in touch with the original main key and any properties that are entirely dependent on it in terms of functionality. This procedure helps with data extraction that only needs a small amount of the key. If and only if all of the requirements for first normal form are met for the primary key and all of the relation's non-primary key qualities are completely dependent on the primary key alone, then the relation is said to be in second normal form.

Third Normal Form

Relationships should not have a non-key attribute that is functionally determined by another non-key attribute or a set of non-key attributes, according to Third Normal Form. No transitive dependence on the primary key should exist, in other words. We break down the relationship into non-key attributes that effectively control other non-key attributes. This procedure is used to get rid of everything that is not entirely rely on the Primary Key. When solely in second normal form and the non-key qualities of the relation are independent of one another, a relation is said to be in third normal form.

TABLE DESIGN

1. tbl_registration

Description: To store registration details.

Primary key: reg_id

Foreign key: addmemb_id

Field Name	Data Type	Size	Description
reg_id	int	4	Primary key
role_id	int	2	Role of Users
addmemb_id	int	4	Foreign key of table tbl_addmembers
reg_email	varchar	40	Store user name
Fname	varchar	25	Store first name
Lname	varchar	35	Store last name
Mobile	varchar	12	Store mobile number
gender	varchar	10	Store gender of users
password	varchar	15	Store password

2. Release

Description: To store release platforms.

Primary key: release_id

Foreign key: Null

Field Name	Data Type	Size	Description
release_id	int	5	Primary key
category	varchar	25	Store release platform category

3. tbl_addmembers

Description: To store details of new staffs.

Primary key: addmemb_id

Foreign key: reg_id

Field Name	Data Type	Size	Description
addmemb_id	int	10	Primary key
reg_id	int	4	Foreign key of table tbl_registration
FName	varchar	25	Store first name of staffs
LName	varchar	35	Store last name of staffs
Email	varchar	45	Store email of staff
password	varchar	25	Store password
Designation	varchar	25	Store designation
gender	varchar	10	Store gender of staff
mobile	varchar	11	Store mobile number
proof	varchar	250	Store Id proof of staff
request_date	timestamp	6	Store date of request
role	varchar	1	Store user role
reason	varchar	100	Store reason of rejection

status	varchar	10	Store status of request

4. tbl_apply

Description: To store details of vacancy applications.

Primary key: apply_id

Foreign key: reg_id, vaccancy_id

Field Name	**Data Type**	**Size**	**Description**
apply_id	int	3	Primary key
reg_id	int	3	Foreign key of table tbl_registration
vaccancy_id	int	3	Foreign key of table tbl_vaccancy
status	varchar	15	Store status of application
qualification	varchar	30	Store qualification
certificate	varchar	200	Store certificate
resume	varchar	200	Store resume
apply_date	timestamp	6	Store date of application
reason	varchar	150	Store reason for rejection

5. tbl_apply_audition

Description: To store details of audition applications.

Primary key: applyAud_id

Foreign key: reg_id, audition_id

Field Name	Data Type	Size	Description
applyAud_id	int	5	Primary key
reg_id	int	5	Foreign key of table tbl_registration
audition_id	int	5	Foreign key of table tbl_audition
email	varchar	45	Store email
marital-status	varchar	20	Store marital status
education1	varchar	30	Store first education
education2	varchar	30	Store second education
certificate	varchar	300	Store certificate
resume	varchar	300	Store resume
medical-certificate	varchar	300	Store medical certificate
intro	varchar	300	Store introductory video
photo	varchar	300	Store profile photo
status	varchar	20	Store status of application
apply_date	timestamp	6	Store apply date
discription	varchar	200	Store description
payment	varchar	10	Store payment status

6. tbl_audition

Description: To store auditions.

Primary key: audition_id

Foreign key: Null

Field Name	Data Type	Size	Description
audition_id	int	10	Primary key
category	varchar	25	Store audition category
skills	varchar	25	Store skills for audition
qualification	varchar	35	Store qualification for audition
date	varchar	15	Store deadline for apply audition
venue	varchar	25	Store venue for audition
fees	int	3	Store application fees
lastupdates	timestamp	6	Store last updation

7. tbl_aud

Description: To store categories of audition.

Primary key: aud_id

Foreign key: Null

Field Name	Data Type	Size	Description
aud_id	int	10	Primary key
category	varchar	25	Store audition category

8. tbl_films

Description: To store film details.

Primary key: poster_id

Foreign key: release_id

Field Name	Data Type	Size	Description
film_id	int	10	Primary key
release_id	int	10	Foreign key of table release
film_name	varchar	25	Store film name
pic	varchar	300	Store poster of film
release_date	varchar	10	Store release date of film
film_type	varchar	25	Store type o film
writer	varchar	30	Store name of film writer
director	varchar	30	Store name of film director
actor1	varchar	30	Store name of main actor
actress	varchar	30	Store name of actress
overview	varchar	2000	Store overview of film
trailer	varchar	1000	Store trailer of film
popularity	varchar	20	Store popularity of film
Last_updation	timestamp	6	Store last update date

9. tbl_filmtype

Description: To store film types.

Primary key: filmtype_id

Foreign key: Null

Field Name	Data Type	Size	Description
filmtype_id	int	10	Primary key
types	varchar	25	Store type of film

10. tbl_freshers

Description: To store details of freshers.

Primary key: profile_id

Foreign key: reg_id

Field Name	Data Type	Size	Description
profile_id	int	3	Primary key
reg_id	int	5	Foreign key of table tbl_registration
dob	varchar	10	Store date of birth of fresher
age	int	2	Store age
alt_mobile	varchar	11	Store alternative mobile number
address	varchar	20	Store house/building name
address2	varchar	20	Store address line 2
city	varchar	15	Store city name
state	varchar	20	Store sate
country	varchar	20	Store name of the country
pin	varchar	10	Store pincode
pic	varchar	200	Store profile picture of freshers
Vaccancy_feild	varchar	35	Store vacancy field
category	varchar	30	Store audition category

11. tbl_qualification

Description: To store qualifications.

Primary key: qual_id

Foreign key: Null

Field Name	Data Type	Size	Description
qual_id	int	10	Primary key
qualification	varchar	35	Store qualification

12. tbl_skills

Description: To store skills.

Primary key: skill_id

Foreign key: Null

Field Name	Data Type	Size	Description
skill_id	int	10	Primary key
skills	varchar	25	Store skills

13. tbl_staffdesignation

Description: To store designations of staff

Primary key: des_id

Foreign key: Null

Field Name	Data Type	Size	Description
des_id	int	10	Primary key
designation	varchar	25	Store designation of staff

14. tbl_vacancy-field

Description: To store vacancy fields.

Primary key: vacfield_id

Foreign key: Null

Field Name	Data Type	Size	Description
vacancy_id	int	2	Primary key
field	varchar	30	Store vacancy field

15. tbl_vaccancy

Description: To store vacancies.

Primary key: vaccancy_id

Foreign key: Null

Field Name	Data Type	Size	Description
vaccancy_id	int	10	Primary key
Vaccancy_field	varchar	35	Store vacancy field
Total_vaccancy	int	10	Store total vacancy
qualification	varchar	25	Store qualification of vacancy
experience	int	10	Store experience
deadline	varchar	10	Store deadline
lastupdates	timestamp	6	Store date of last updation

16. tbl_venue

Description: To store venues of auditions.

Primary key: venue_id

Foreign key: Null

Field Name	Data Type	Size	Description
venue_id	int	2	Primary key
venue	varchar	15	Store venue of audition

17. tbl_rating

Description: To store ratings of movies.

Primary key: rate_id

Foreign key: reg_id, film_id

Field Name	Data Type	Size	Description
rate_id	int	4	Primary key
reg_id	int	5	Foreign key of table tbl_registration
film_id	int	4	Foreign key of table tbl_films
comment	varchar	200	Store comment
rating	int	3	Store rating for films
update_date	timestamp	6	Store date of updation

CHAPTER 5

SYSTEM TESTING

5.1 INTRODUCTION

Software testing is the controlled execution of software to determine whether it behaves according to specifications. Software testing, as well as verification and validation, are frequently used synonymously. Validation is the process of examining or evaluating a product, including software, to determine whether it complies with all relevant specifications. Verification techniques other than software testing include reviews, analyses, inspections, and walkthroughs. Validation is the process of making sure that the specification matches the user's intent.

Two other actions that are typically related to software testing are static and dynamic analysis. Without actually running the code, static analysis looks for issues and metrics in the source code of software. With the help of dynamic analysis, information like execution traces, timing profiles, and test coverage can be obtained by looking at the behavior of software as it is being used.

A series of tasks known as testing can be organized in advance and completed in a methodical way. Modules are tested first, followed by the integration of the full computer-based system. Testing is a necessity because testing goals are essential to the system's performance. A number of rules can be used as test objectives. They are:

A program is tested by being run with the goal of identifying any errors.

- A test case with a high likelihood of detecting an unknown fault qualifies as a good test case.
- A test that finds an error that has not yet been found is successful.

If the testing is successfully completed in accordance with the aforementioned objectives, it will identify software flaws. Additionally, testing reveals that it seems like the software functionalities are functioning as intended and that the performance specifications have been reached.

Three methods exist for software testing:

- For correctness.
- For implementation efficiency.
- For computational complexity.

Correctness tests are meant to guarantee that a software operates exactly as intended. This is much harder than it seems, particularly for big programs.

5.2 TEST PLAN

A test plan denotes a series of desired actions to be taken in order to complete various testing methods. The Test Plan serves as a blueprint for the action to be taken. Software engineers create computer programs, documentation, and data structures. The software developers are always in charge of testing the individual units of the programs to ensure that they perform the function for which they were designed. An independent test group (ITG) exists to eliminate the inherent problems associated with allowing the builder to test the finished product. The specific goal of testing should be stated in quantifiable terms. Defect density or frequency of occurrence, duration to failure, cost to locate and correct the problems, and test effort hours per regression test should all be mentioned in the test plan. The testing levels include:

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

5.2.1 Unit Testing

Unit testing focuses on the software component or module, which is the smallest unit of software design. Within the module border, significant control pathways are tested using the component level design description as a guide. We determine the relative difficulty and uncovered extent of unit testing. Multiple components can be checked simultaneously during unit testing because it is white-box in nature. To guarantee that data enters and exits the software unit under test properly, the modular interface is tested. To make sure that

information temporarily stored retains its integrity during each step of an algorithm's execution, the local data structure is inspected. Boundary conditions are verified to make sure that each statement in a module has been run at least once. All error handling paths are then tested to the limit.

The data flow over a module interface needs to be tested before moving on to other tests. If data does not enter and exit properly, all other tests are invalidated. Selective testing of execution pathways must be done during the unit test. Error handling pathways should be set up to cleanly reroute or terminate processing when an error occurs and error situations should be anticipated. Boundary testing is the last activity in the unit testing process. At its boundaries, software frequently fails.

When testing individual modules in the Sell-Soft System, each module was treated as a distinct entity and subjected to a variety of test inputs. The modules' internal logic had a few bugs, which were found and repaired. Each module is tested and run independently after coding. Each module was checked to make sure it worked well and delivered the desired results before any extraneous code was removed.

5.2.2 Integration Testing

Integration testing is a methodical approach for developing the program's architecture and carrying out tests to find interface flaws. The objective is to construct a program structure that is determined by design using unit-tested components. The entire program is put through testing. Because of the program's extensive reach, correcting the problem is challenging. After these mistakes are fixed, new ones arise, and it seems like the process never ends.

All modules were integrated once the system's unit testing was finished in order to check for any interface discrepancies. A single program structure also developed as disparities in program architectures were eliminated.

5.2.3 Validation Testing or System Testing

The testing has reached its conclusion. The complete system, including all forms, code, modules, and class modules, was tested in its entirety. Common names for this kind of testing include System testing and Black Box testing.

The functional requirements of the software are the main emphasis of the Black Box testing approach. Black Box testing, in other words, enables a software engineer to create sets of input circumstances that fully exercise all functional requirements for a program. Correct or missing functions, interface mistakes, data structure or external data access faults, performance errors, initialization errors, and termination errors are the types of errors that black box testing looks for.

5.2.4 Output Testing or User Acceptance Testing

The system under consideration has been tested for user acceptance; it should meet the needs of the firm. The software should maintain contact with the prospective system and user while developing and making changes as needed. This is done in relation to the following points:

- Input Screen Designs,
- Output Screen Designs,

A variety of test data types are used to conduct the aforementioned testing. In system testing, the preparation of test data is crucial. The system understudy is tested using the test data after the test data preparation. During system testing, test data mistakes are found and fixed using the aforementioned testing procedures, and the fixes are also recorded for later use.

Automation Testing

Automation testing is the process of testing software and other tech products to ensure it m eetsstrict requirements. Essentially, it’s a test to double-check that the equipment or software doesexactly what it was designed to do. It tests for bugs, defects, and any other issues that canarisewith product development. Automation testing can be run at any time of the day. It usesscripted sequences to examine the software. It then reports on what’s

been found, and this information can be compared with earlier test runs.

Benefits of Automation Testing

- Detailed reporting capabilities - Automation testing uses well-crafted test cases for various scenarios. These scripted sequences can be incredibly in-depth, and provide detailed reportsthat simply wouldn't be possible when done by a human.
- Improved bug detection - One of the main reasons to test a product is to detect bugs and other defects. Automation testing makes this process an easier one. It's also able to analyze a wider test coverage than humans may be able to.
- Simplifies testing - Testing is a routine part of the operations of most SaaS and techcompanies. Making it as simple as possible is key. Using automation is extremely beneficial. When automating test tools, the test scripts can be reused.
- Speeds up the testing process - Machines and automated technology work faster than humans. Along with improved accuracy, this is why we use them. In turn, this shortensyour software development cycles.
- Reduces human intervention - Tests can be run at any time of day, even overnight, without needing humans to oversee it. Plus, when it's conducted automatically, this can also reducethe risk of human error.

5.2.5 Selenium Testing

Selenium is an open-source tool that automates web browsers. It provides a single interface that lets you write test script in programming language like Ruby, Java, NodeJS, PHP, Perl, Python and C#, among others. The Selenium testing tool is used to automate test across browsers for web applications. It's used to ensure high-quality web applications whether they are responsive, progressive, or regular. Selenium is an open-source tool.

Test cases for a Login Page

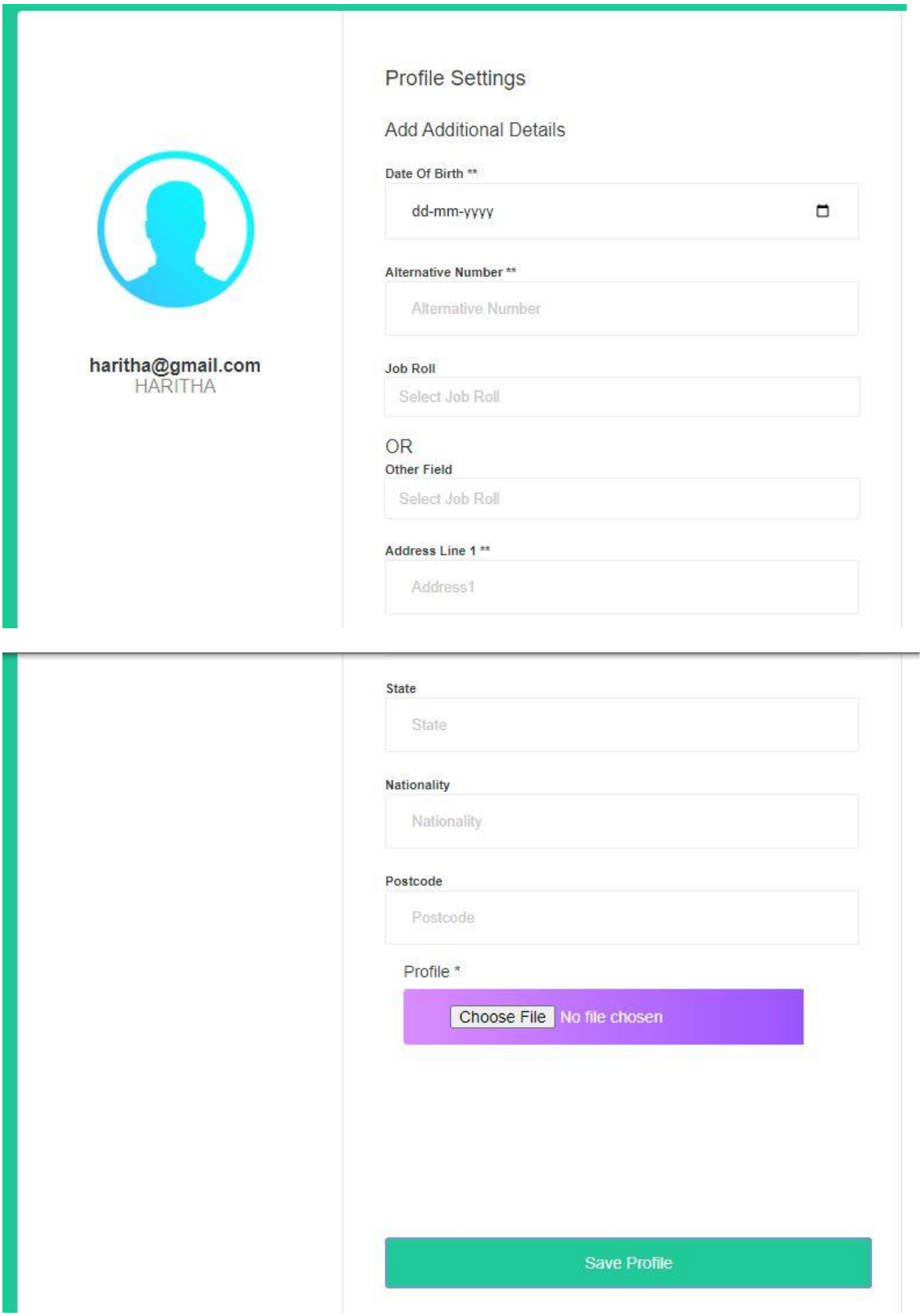
Test Case ID: Fun_1			**Test Designed By: Aleena Joseph**		
Test Priority (Low/Medium/High): High			**Test Designed Date:** 18-07-2022		
Module Name: Login Screen			**Test Executed By: Ms. Jetty Benjamin**		
Test Title: Verify login with valid username and password			**Test Execution Date:** 18-05-2022		
Description: Test the Login Page					
Pre-Condition: User has valid username and password					
Step	**Test Step**	**Test Data**	**Expected Result**	**Actual Result**	**Status(Pass/Fail)**
1	Navigation to Login Page		Login Page should be displayed	Login page displayed	Pass
2	Provide Valid User Name	User Name: haritha@gmail.com	User should be able to Login	User Logged in and navigated to the dashboard with records	Pass
3	Provide Valid Password	Password: haritha123			
4	Click on Sign In button				
5	Provide Invalid user name or Password	Email Id: haritha@gmail.com Password: haritha123	User should not be able to Login	Message for enter valid username or password displayed	Pass
6	Provide Null Username Id or Password	Email Id: null Password: null			
7	Click on Sign In button				
Post-Condition: User is validated with database and successfully login into account. The Account session details are logged in database					

Login

```
package test;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
public class flickcom {
public static void main(String[] args) {
System.setProperty("webdriver.chrome.driver","C:\\Users\\Teena\\Downloads\\chrome
driver_win32\\chromedriver.exe" );
WebDriver driver=new ChromeDriver();
driver.get("http://localhost/FlickCom/login.php");
driver.findElement(By.id("email")).sendKeys("haritha@gmail.com");
driver.findElement(By.id("pswd1")).sendKeys("haritha123");
driver.findElement(By.id("submit")).click();
String actualUrl="http://localhost/FlickCom/Freshers/about.php";
String expectedUrl= driver.getCurrentUrl();
if(actualUrl.equalsIgnoreCase(expectedUrl))
{
System.out.println("Test passed");
}
else
{
System.out.println("Test failed");
}
}
}
```

```
package test;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
public class flickcom {
 public static void main(String[] args) {
     System.setProperty("webdriver.chrome.driver","C:\\Users\\Teena\\Downloads\\chromedriver_win32\\chromedriver.exe" );
  WebDriver driver=new ChromeDriver();
  driver.get("http://localhost/FlickCom/login.php");
  driver.findElement(By.id("email")).sendKeys("haritha@gmail.com");
  driver.findElement(By.id("pswd1")).sendKeys("haritha123");
  driver.findElement(By.id("submit")).click();
  String actualUrl="http://localhost/FlickCom/Freshers/about.php";
  String expectedUrl= driver.getCurrentUrl();
  if(actualUrl.equalsIgnoreCase(expectedUrl))
  {
  System.out.println("Test passed");
  }
  else
  {
  System.out.println("Test failed");
  }
 }
}
```

```
Starting ChromeDriver 103.0.5060.53 (a1711811edd74ff1cf2150f36ffa3b0dae40b17f-refs/branch-heads/5060@{#853}) on port 60439
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Jul 21, 2022 9:19:34 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Jul 21, 2022 9:19:34 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Test passed
```


haritha@gmail.com
HARITHA
Profile Settings
Add Additional Details
Date Of Birth **
dd-mm-yyyy
Alternative Number **
Alternative Number
Job Roll
Select Job Roll
OR
Other Field
Select Job Roll
Address Line 1 **
Address1
State
State
Nationality
Nationality
Postcode
Postcode
Profile *
Choose File
No file chosen
Save Profile

Test cases for Fresher Registration

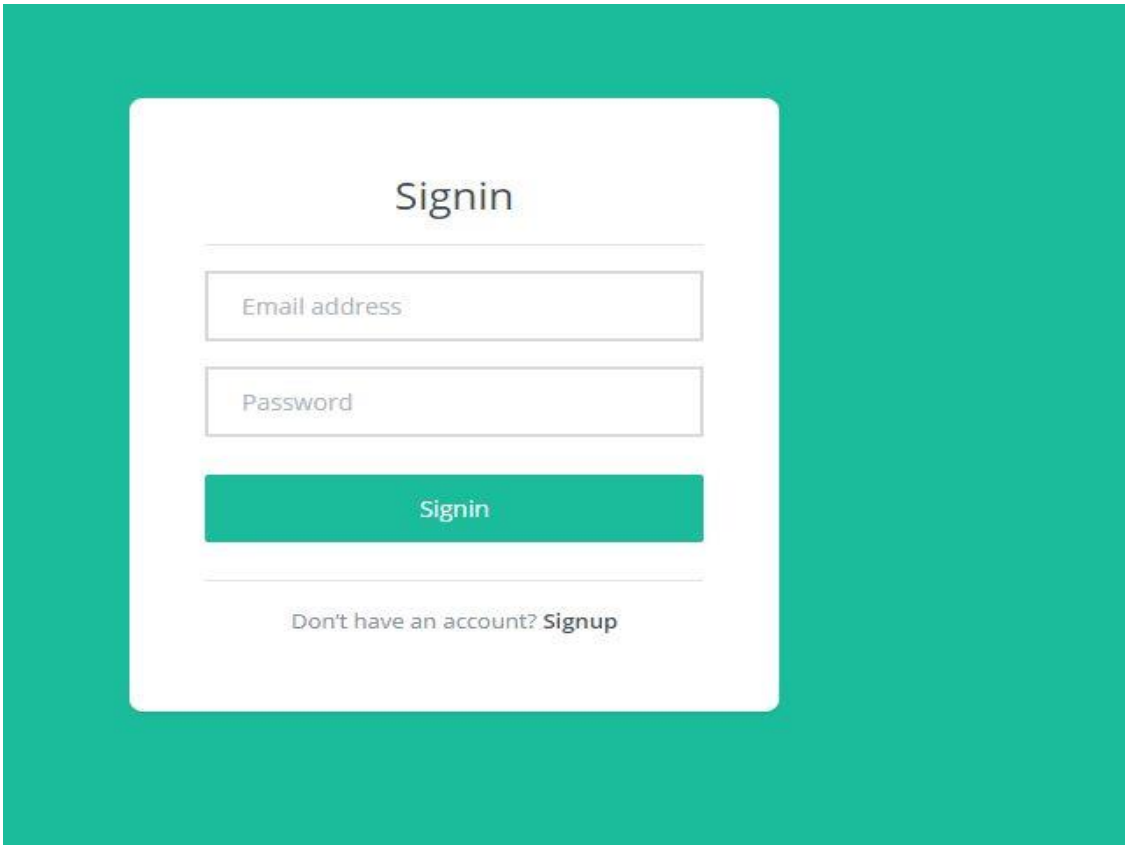
Project Name: FlickCom					
			Updation Test Case		
Test Case ID: registration			**Test Designed By: Aleena Joseph**		
Test Priority (Low/Medium/High): High			**Test Designed Date:** 18-07-2022		
Module Name: Register Screen			**Test Executed By: Ms. Jetty Benjamin**		
Test Title: User Registration Details			**Test Execution Date:** 18-07-2022		
Description: Register to system and Registration is completed then login , if some error occurs, test will fail					
Pre-Condition: User has valid user name and password					
Step	**Test Step**	**Test Data**	**Expected Result**	**Actual Result**	**Status (Pass/Fail)**
1	Navigation to Register Page		RegisterPage Should be displayed	Registration page displayed	Pass
2	Provide Valid Registration details	User Name: tv@gmail .com	User should be able to Register	User registration Completed after go to the login page	Pass
3					
4	Click on Login button				
5	Provide profile details	Input profile details	User will be redirected to Login page	Use will be redirected to Login page	Pass
7	Click onregister button				
8	Provide invalid information	Input invalid profile details.	User will bestay in Register page	User will be stay on that page showing error message	Pass
9	Click onregister button				
Post-Condition: User is validated with database and successfully login into account. TheAccount session details are logged in database.					

Registration

Register

```
package test;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
public class flick_register {
public static void main(String[] args) {
System.setProperty("webdriver.chrome.driver","C:\\Users\\Teena\\Downloads\\chromed
river_win32\\chromedriver.exe" );
WebDriver driver=new ChromeDriver();
driver.get("http://localhost/FlickCom/signup.php");
driver.findElement(By.id("fname")).sendKeys("AMRUTHA");
driver.findElement(By.id("lname")).sendKeys("JOSEPH");
driver.findElement(By.id("email")).sendKeys("tv@gmail.com");
driver.findElement(By.id("pswd1")).sendKeys("amrutha123");
driver.findElement(By.id("pswd2")).sendKeys("amrutha123");
driver.findElement(By.id("phoneNumber")).sendKeys("9967341098");
driver.findElement(By.id("gender")).sendKeys("Female");
driver.findElement(By.id("type")).sendKeys("Fresher");
driver.findElement(By.id("submit")).click();
String actualUrl="http://localhost/FlickCom/login.php";
String expectedUrl= driver.getCurrentUrl();
if(actualUrl.equalsIgnoreCase(expectedUrl))
{
System.out.println("Test passed");
}
else
{
System.out.println("Test failed");
}
}}
```

```
1  package test;
2  import org.openqa.selenium.By;
3  import org.openqa.selenium.WebDriver;
4  import org.openqa.selenium.chrome.ChromeDriver;
5  public class flick_register {
6   public static void main(String[] args) {
7       System.setProperty("webdriver.chrome.driver","C:\\Users\\Teena\\Downloads\\chromedriver_win32\\chromedriver.exe" );
8    WebDriver driver=new ChromeDriver();
9    driver.get("http://localhost/FlickCom/signup.php");
0    driver.findElement(By.id("fname")).sendKeys("AMRUTHA");
1    driver.findElement(By.id("lname")).sendKeys("JOSEPH");
2    driver.findElement(By.id("email")).sendKeys("amrutha@gmail.com");
3    driver.findElement(By.id("pswd1")).sendKeys("amrutha123");
4    driver.findElement(By.id("pswd2")).sendKeys("amrutha123");
5    driver.findElement(By.id("phoneNumber")).sendKeys("9967341098");
6    driver.findElement(By.id("gender")).sendKeys("Female");
7    driver.findElement(By.id("type")).sendKeys("Fresher");
8
9    driver.findElement(By.id("submit")).click();
0    String actualUrl="http://localhost/scholarshipportal/loginpage/login.html";
1    String expectedUrl= driver.getCurrentUrl();
2    if(actualUrl.equalsIgnoreCase(expectedUrl))
3    {
4    System.out.println("Test passed");
5    }
6    else
7    {
8    System.out.println("Test failed");
9    }
0   }
1  }
2
3
```



CHAPTER 6

IMPLEMENTATION

6.1 INTRODUCTION

The implementation stage of a project is when the theoretical design is transformed into a working system. It is the most important stage in achieving a successful new system because it gives users confidence that the new system will work, be effective, and accurate. Documentation and user training are its main priorities. Conversion typically happens either later, or at the same time that the user is being trained. Implementation is the process of turning a new, revised system design into a standalone operation, and it simply refers to bringing a new system design into operation. At this point, the user department is responsible for the majority of the workload, the most heavy lifting, and the most impact on the existing system. If the implementation is not well planned and controlled, chaos and confusion may occur. Implementation encompasses all activities undertaken to convert from the existing system to the new system. The new system may replace an existing human or automated system, be entirely new, or involve modifications to an existing system. To develop a dependable system that satisfies the needs of the organization, proper implementation is necessary. The process of putting the created system into practical use is referred to as system implementation. This covers all procedures necessary to switch from the old to the new system. Only after thorough testing and verification that it satisfies the requirements can the system be put into use. The system staff assesses the system's viability. The effort necessary for system analysis and design to implement the three key components of education and training, system testing, and changeover increases with the complexity of the system being implemented. The following tasks are included in the implementation state:

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

6.2 IMPLEMENTATION PROCEDURES

Software implementation is the process of installing a package in its final form so that it functions as intended and meets all other requirements for the system. Many businesses will hire someone to design the software but not operate it themselves.

People have early reservations about the software, but we must watch out that they do not become more resistant by making 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 proceeding to view the system, the user should be aware that in order to view the results, the server program must be running on the server. The actual process will not take place if the server object is not up and running on the server.

6.2.1 User Training

The goal of user training is to get the user ready for system conversion and testing. The individuals engaged must have faith in their ability to contribute to the new system in order for it to provide the desired results and advantages. As the system gets more complicated, training becomes more important. The user is trained on how to enter data, reply to error alerts, query the database, and call routines that produce reports and carry out other essential tasks.

6.2.2 Training on the Application Software

The user will need to be instructed on the new application software after receiving the appropriate basic computer awareness training. This will outline the guiding principles for using the new system, including how the screens work, how they are designed, what kind of support is displayed, what kinds of errors occur when entering data, what validation checks are performed for each entry, and how to change the date that was entered. Then, while providing program training on the application, it should cover the information needed by the particular user/group to operate the system or a certain area of the system. Depending on the user group and hierarchical level, this training may vary.

6.2.3 System Maintenance

The maintenance of systems is a mystery. When a software product is at the maintenance stage of its life cycle, it is valuable. A system needs to be properly maintained once it has

been effectively implemented. The life cycle of software development must include system maintenance.

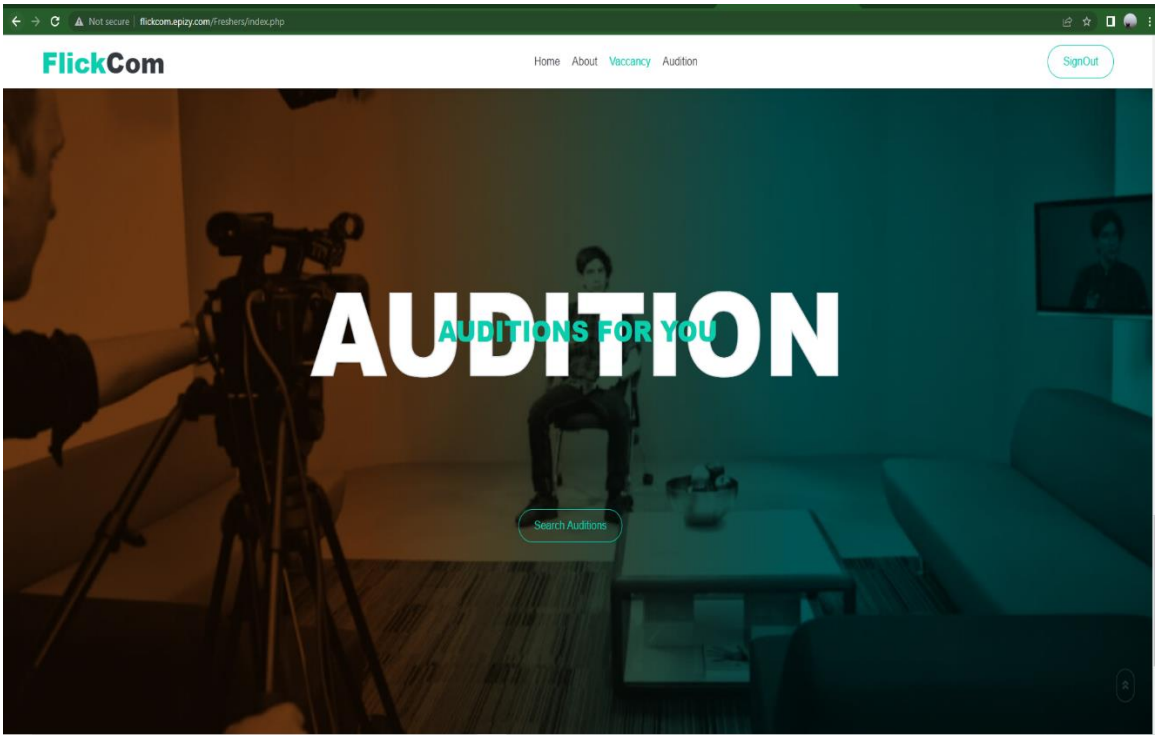
Making the system more adaptive to changes in the system environment is the goal of system maintenance. Of all, maintaining software involves much more than just "Finding Mistakes."

6.2.4 Hosting

The activity of housing, serving, and managing files for one or more Web sites is known as hosting (also known as Web site hosting, Web hosting, and Webhosting). The quick Internet connection is more crucial than the computer capacity allotted for Web site files. The majority of hosting providers provide connectivity over T-carrier system lines. A similar link would typically be necessary and pricey for a single company running its own website.

Hosting Method Steps:

- Create an account or login into your account.
- Create a hosting account.
- Create a custom domain or a subdomain provided by infinityFree.
- Manage your account.
- Upload your Files.
- Creating your Database.
- Choose a platform for your website.
- Changing your PHP connection file configuration.


Not secure
flickcom.epizy.com/Freshers/index.php
FlickCom
Home
About
Vaccancy
Audition
SignOut
AUDITION
AUDITIONS FOR YOU
Search Auditions

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.

This system is very helpful for the users for getting details of latest movies with its posters and trailers. They can rate each movies based on their likes. The person those who are searching opportunities in film industry or those who are aimed for starting their carrier in film fields, they can find ways to their goal in a single platform. They can get various vacancies and auditions for film fields. They can choose and apply for auditions and vacancies as their wish. So that, this site is really helpful for the freshers for finding scopes in film industry.

7.2 FUTURE SCOPE

In future we can expect the modified version of "FlickCom". The extended version of this site may include the selection process and scoring details of auditions. In future it includes movie success prediction and fans based club.

CHAPTER 8

BIBLIOGRAPHY

REFERENCES:

- Gary B. Shelly, Harry J. Rosenblatt, "*System Analysis and Design*",2009.
- Roger S Pressman, "*Software Engineering*",1994.
- Pankaj Jalote, "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 Design Descriptions.

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
session_start();
include('connectivity.php');
?>
<!DOCTYPE html>
<html lang="en">
<head>
<title>FlickCom:SignUp</title>
<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>
<!-- 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="css/style1.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
if(isset($_REQUEST['message']))
{
$msg=$_REQUEST['message'];
echo'<tr><td colspan="9" style="color:#FF0000" align="center">'.$mgs.'</td></tr>';
}
?>
<!-- [ 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-500">Sign up</h4>
<hr>
<div class="form-group mb-3">
<label>First Name</label>
<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]{2,}$/))
{
```

```
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">
<label>Last Name</label>
<input type="text" class="form-control" name="lname" id = "lname" title="Start with a
Capital letter & Only alphabets and 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-Za-z][A-Za-z\s]*$/))
{
document.getElementById('charMsg').innerHTML="Start with a Capital letter & Only
alphabets and space are allowed!!";
document.getElementById('lname').value = "";
return false;
}
document.getElementById('charMsg').innerHTML=" ";
return true;

}
```

```
</script>
</div>

<div class="form-group mb-4">
<label>User Name</label>
<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">
<label>Password</label>
<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 Valipswd();"/><span id="msg9"
style="color:red;"></span>
```

```
<script>
function Valipswd()
{
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">
<label>Confirm Password</label>
<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!!";
pas2.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">
<label>Mobile</label>
<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]{9,10}$/))
{
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="btn btn-block btn-primary
mb-3"  value = "Register">
<hr>
<p class="mb-2">Already have an account? <a href="login.php" class="f-w-
400">SignIn</a></p>
```

```
<p class="mb-2"><a href="index.php" class="f-w-400">Back</a></p>
</div>
</div>
</div>
</div>
</div>
</form>
<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>
```

login.php

```
<?php
session_start();
include('connectivity.php');
?>
<?php
if (isset($_POST['email']) && isset($_POST['pswd1']))
{
$myuser=$_POST['email'];
$mypass=$_POST['pswd1'];

$sql="SELECT * FROM `tbl_registration` where reg_email='$myuser' and
password='$mypass'";
$result=mysqli_query($query,$sql);

if(mysqli_num_rows($result) === 1)
{
```

```
while($row=mysqli_fetch_array($result)){
if ($row['reg_email'] == $myuser && $row['password'] == $mypass)
{
if($row['role_id']== 0){
session_start();

$_SESSION['id']=$row['reg_id'];

header("location:./admin/index.php");
}
else if($row['role_id']== 1)
{
session_start();

$_SESSION['id']=$row['reg_id'];
header("location:rating.php");
}
else if($row['role_id']== 3)
{
session_start();
$_SESSION['id']=$row['reg_id'];
header("location:./staffs/index.php");
}
else if($row['role_id']== 2)
{
session_start();
$_SESSION['id']=$row['reg_id'];
header("location:./Freshers/about.php");
}
else
{
}
```

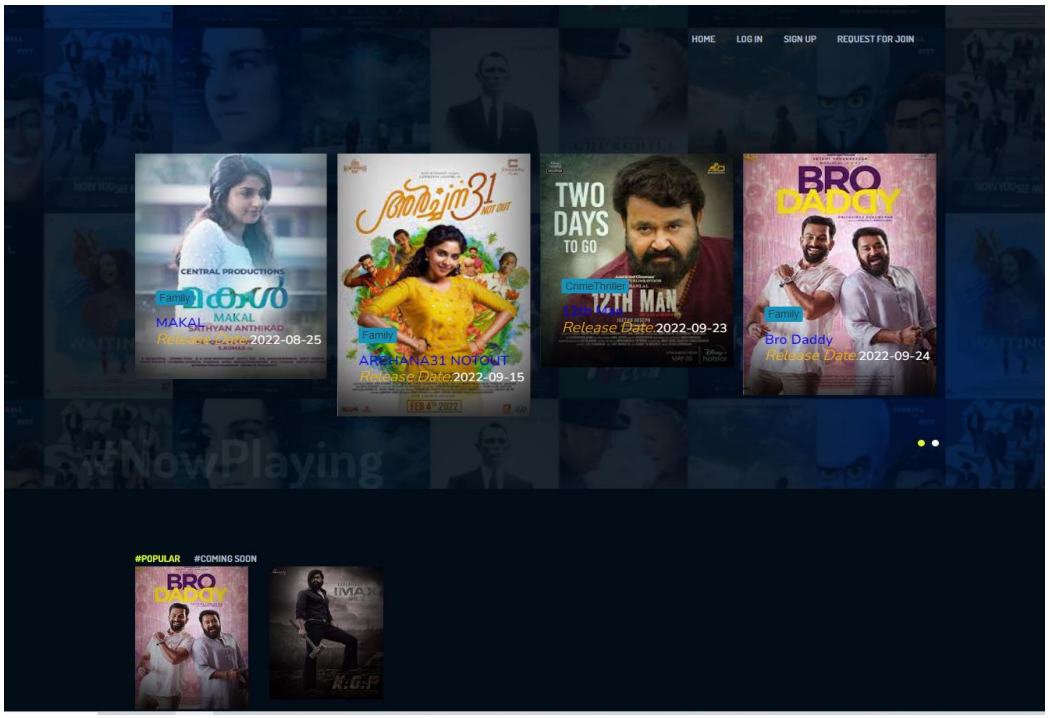
```
}
else{
?>
<script type="text/javascript">
alert("Invalid Username Or Password!!!.");
window.location = "login.php";
</script>
<?php
exit();
}
}
}
}
?>
<!DOCTYPE html>
<html lang="en">

<head>
<title>FlickCom:Signin</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 -->

<!-- vendor css -->

<link rel="stylesheet" href="css/style1.css">
</head>
<body>
<!-- [ auth-signin ] start -->
<div class="auth-wrapper">
<div class="auth-content text-center">

<div class="card borderless">
<div class="row align-items-center ">
<div class="col-md-12">
<div class="card-body">
<form method="post" action="" name="login" class="registartion-form"
onsubmit="return (valid())">
<h4 class="mb-3 f-w-400">Signin</h4>
<hr>
<div class="form-group mb-3">
<input type="text" class="form-control" name="email" id="email" placeholder="Email
address">
</div>
<div class="form-group mb-4">
<input type="password" class="form-control" id="pswd1" name="pswd1"
placeholder="Password">
</div>

<button class="btn btn-block btn-primary mb-4">Signin</button>
<br>
```

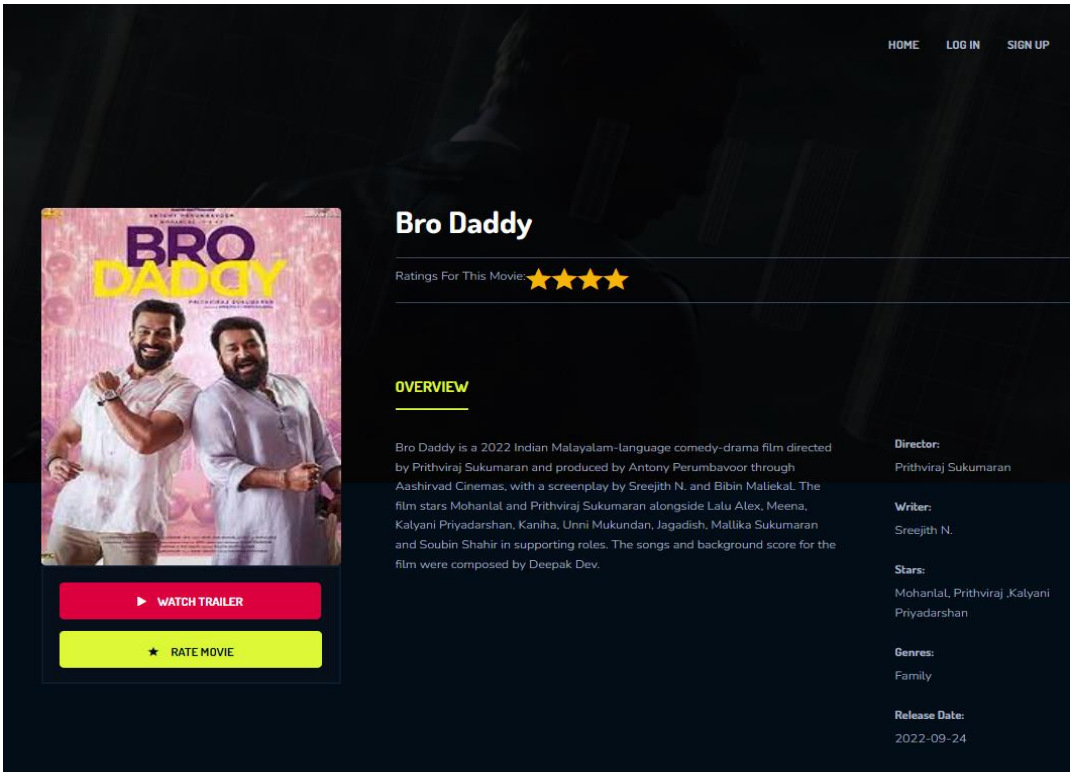
```
<p class="mb-0 text-muted">Don’t have an account? <a href="signup.php" class="f-w-
400">Signup</a></p>
<p class="mb-0 text-muted"> <a href="index.php" class="f-w-400">Back</a></p>

</form>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- [ auth-signin ] 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>
```

9.2 SCREEN SHOTS

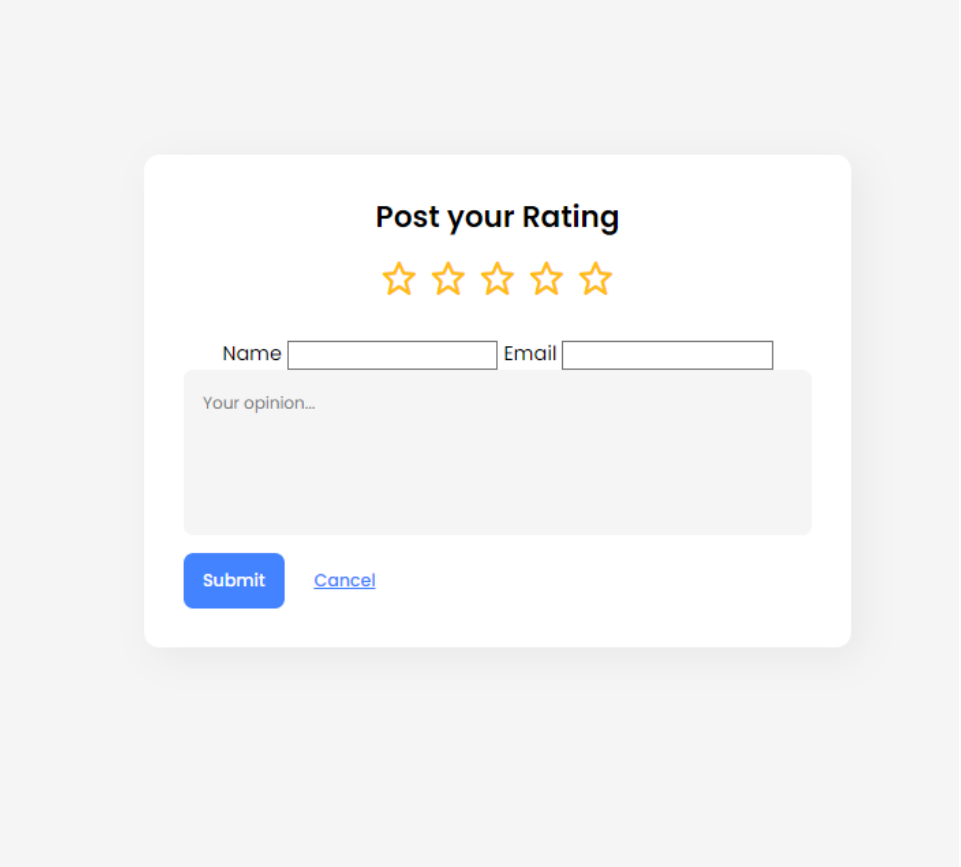
Home page



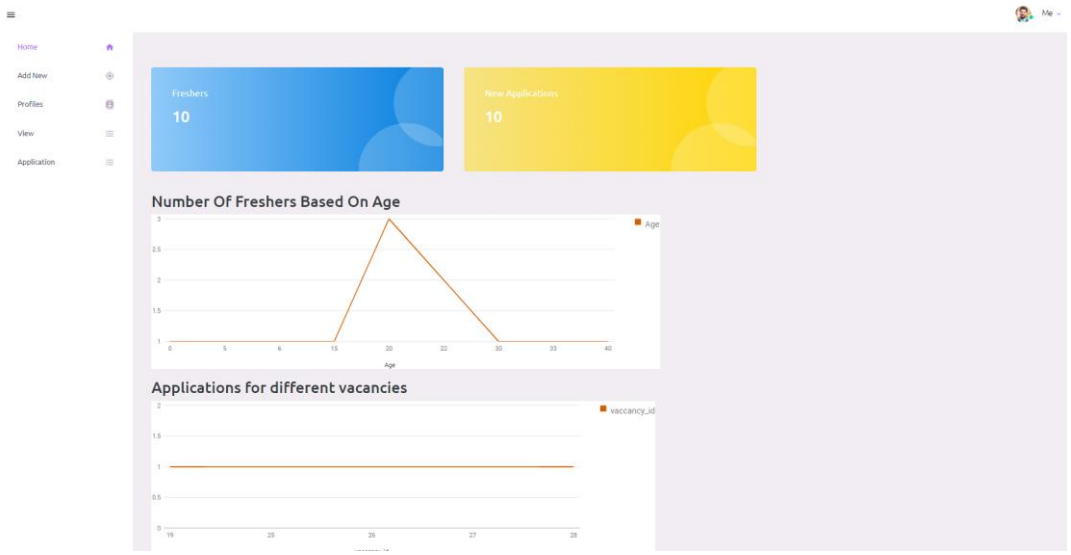
Movie Details



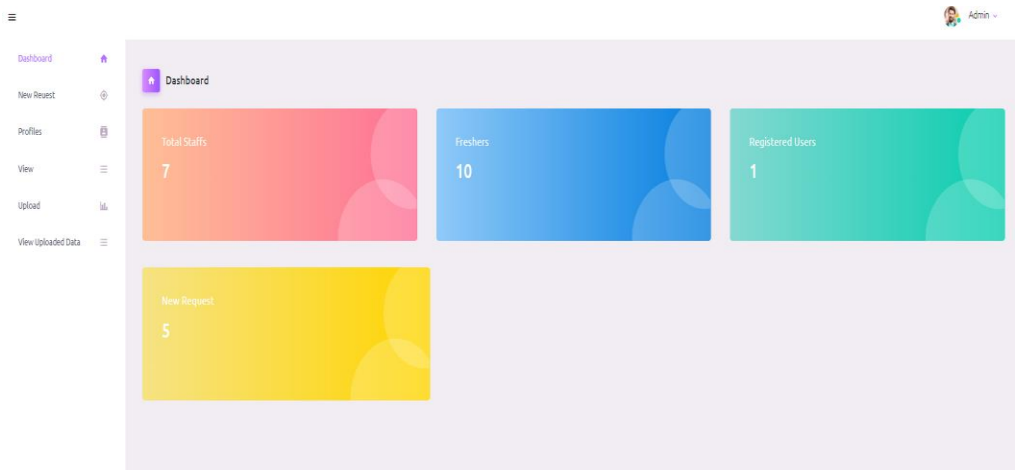
Rating



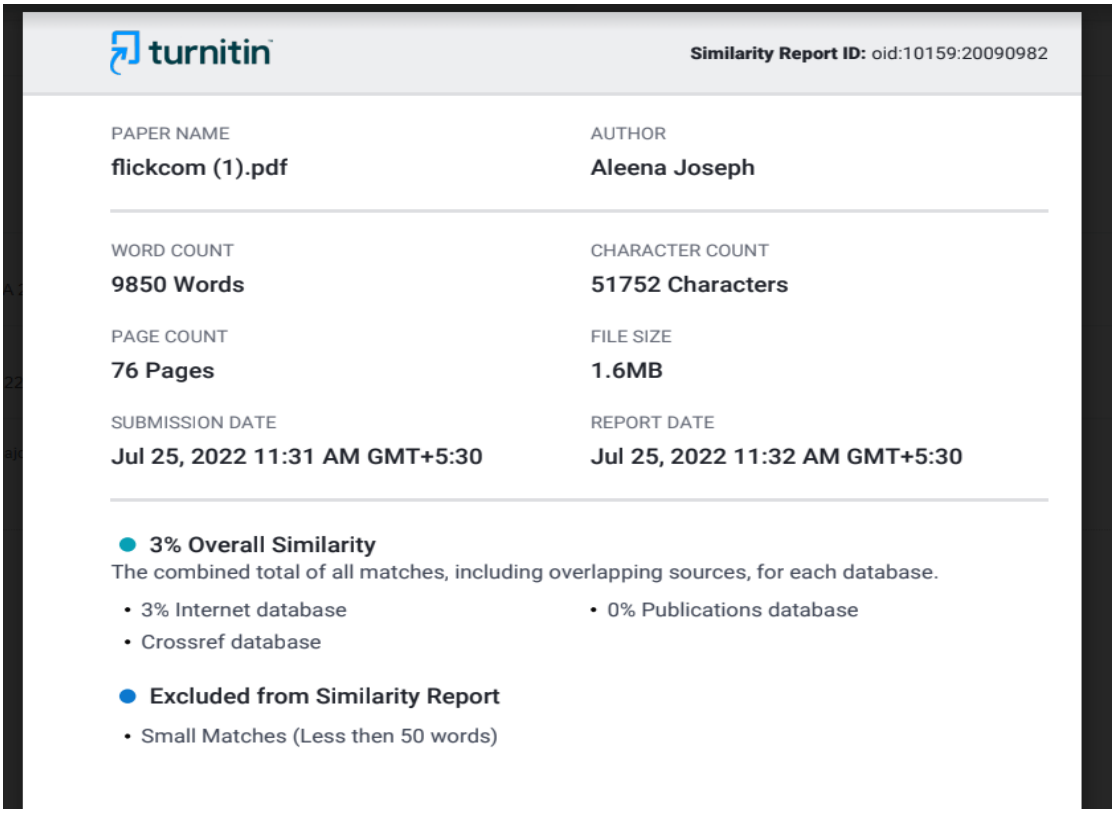
Staff Home Page



Admin Dashboard



9.3 PLAGIARISM REPORT



turnitin

Similarity Report ID: oid:10159:20090982

PAPER NAME
flickcom (1).pdf

AUTHOR
Aleena Joseph

WORD COUNT
9850 Words

CHARACTER COUNT
51752 Characters

PAGE COUNT
76 Pages

FILE SIZE
1.6MB

SUBMISSION DATE
Jul 25, 2022 11:31 AM GMT+5:30

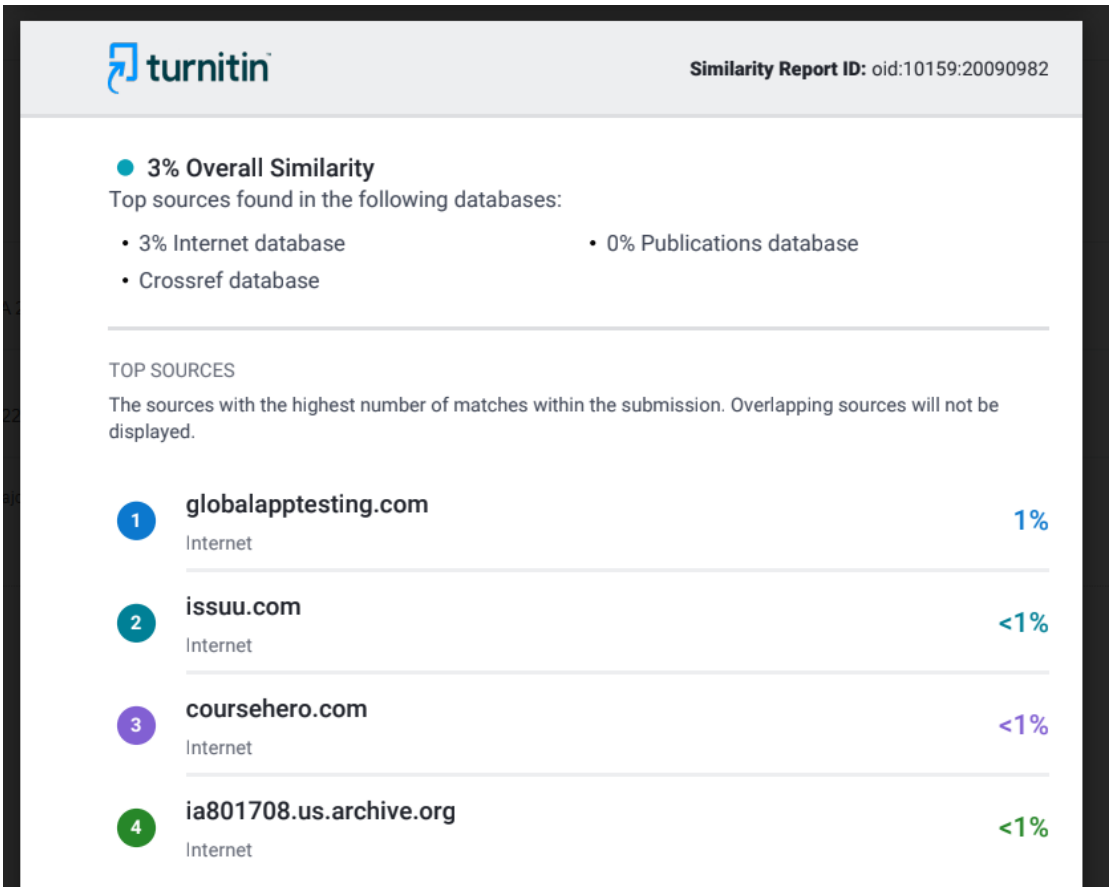
REPORT DATE
Jul 25, 2022 11:32 AM GMT+5:30

3% Overall Similarity
The combined total of all matches, including overlapping sources, for each database.

- 3% Internet database
- 0% Publications database
- Crossref database

Excluded from Similarity Report

- Small Matches (Less then 50 words)



turnitin

Similarity Report ID: oid:10159:20090982

3% Overall Similarity
Top sources found in the following databases:

- 3% Internet database
- 0% Publications database
- Crossref database

TOP SOURCES
The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	globalapptesting.com Internet	1%
2	issuu.com Internet	<1%
3	coursehero.com Internet	<1%
4	ia801708.us.archive.org Internet	<1%