**“Interview Management system”**

A

**Project Report**

submission

of

**Software Engineering**



**SUBMITTED TO- SUBMITTED BY-**

Dr. Girdhari Singh Rahul Tuteja

2017ucp1479

# **1. Introduction**

This system manages the recruiting and hiring process, including job postings and job applications. It organizes and makes searchable information about job seekers. It helps scheduling, issues notification alerts .This systems maintain a database of new applicants and people who may be continuing prospects for future openings. These systems maintain a database of new applicants and people who may be continuing prospects for future openings.

# **1.1 PURPOSE**

The main purpose of this SRS for this project is to provide a basic idea of how the system will be designed and implemented. What are the functional and non-functional features that are to be put in the system? Client’s requirements are duly noted and documented to help design the system accordingly. It gives a comprehensive view of how the system is supposed to work and what is to be expected by the end user.

# **1.2 SCOPE**

The System is built to handle 3 kind of users, Admin, Candidate and the

Interviewer.

**1.2.1 Admin**

The recruiter of the company who will control all the functioning from adding jobs to selection or rejection of candidate.

**1.2.2 The Interviewer**

The person who will take the interview of the candidate.

**1.2.3 Candidate**

The general person who apply for the available jobs for the company.

**2. Specific Requirements**

**2.1 FUNCTIONAL REQUIREMENTS**

There must three different ends for the project, with the provided functionality.

**Admin**

1. He may add different Department and the corresponding positions to be held for the company.

2. He will add the interviewers for each position.

3. He will be the person who the put jobs for advertisements.

4. He will schedule the new interviews / next interview for each applicant.

5. He will assign interviewers for the interview.

6. In the end, he will be the one deciding whether person must be selected or not.

**Candidate**

1. Any new candidate can sign up for the job.

2. Any candidate can view/apply for jobs for which he/she is eligible.

3. Candidate must be able to see status for his current application.

4. Candidate should be able to see his profile.

**Interviewers**

1. He must be able to see his upcoming interviews.

2. He must be able to see candidate’s profile.

3. He shall reply with his thoughts about the candidate.

# **2.2 EXTERNAL INTERFACE REQUIREMENTS**

**User Interfaces:**

The user interface for system shall be compatible with any version of any browser.

**Software Interfaces (OS-Windows, Ubuntu Browser: chrome, torch etc):**

Operating System: Windows or Linux

# **2.3 PERFORMANCE REQUIREMENTS (NFR)**

* **Efficiency:** Enough resources implemented to achieve particular task efficiently.
* **Flexibility:** System should be flexible to provide space to add new features and maintain the old ones. The new feature integrated with minimum effort.
* **Integrity:** System should secure user information and not allow unauthorized access or updating to anyone.
* **Portability:** The system should run on any browser version, and any operating system.
* **Usability:** The system should have enough information to allow user to know the use of the displayed window. Any major changes made should be notified back to the user to allow them to know about the changes done by them.
* **Testability:** The system should be tested according to the requirements and confirm proper working of the software.
* **Maintainability:** Any errors found after the release of the system, the software must be maintainable to give required performance.

**2.4 Technologies to used**

**2.4.1 Front End**

1) HTML

2) CSS

3) JavaScript

4) Angular.js

**2.4.2 Back End**

1) Node.js

2) Express.js

**2.4.3 Database**

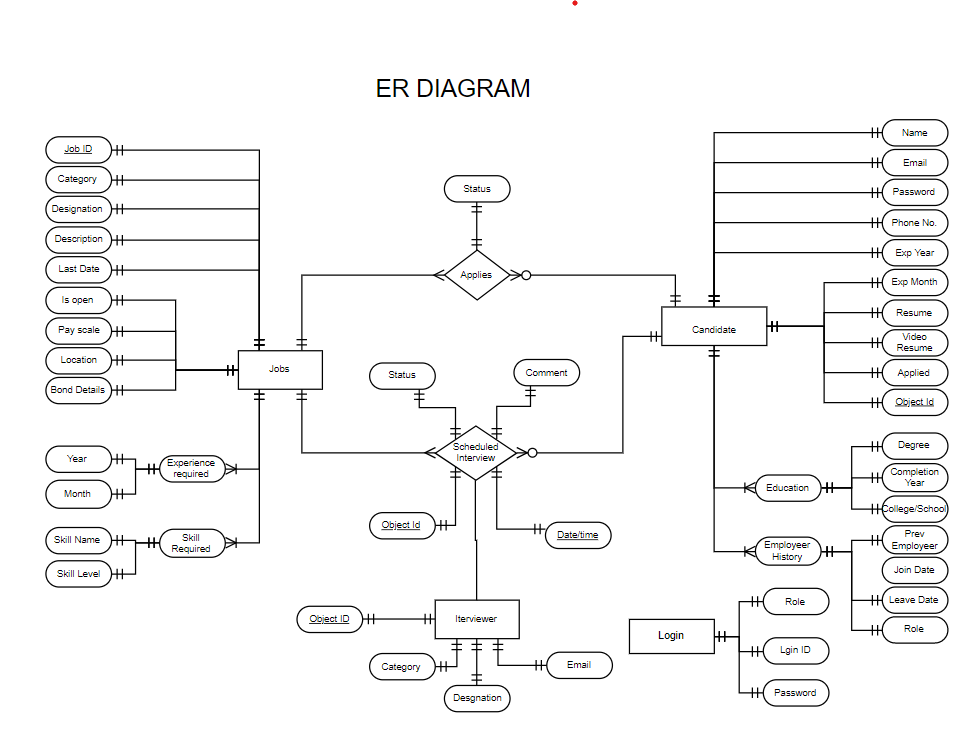
1) Mongo dB

**3. Design**

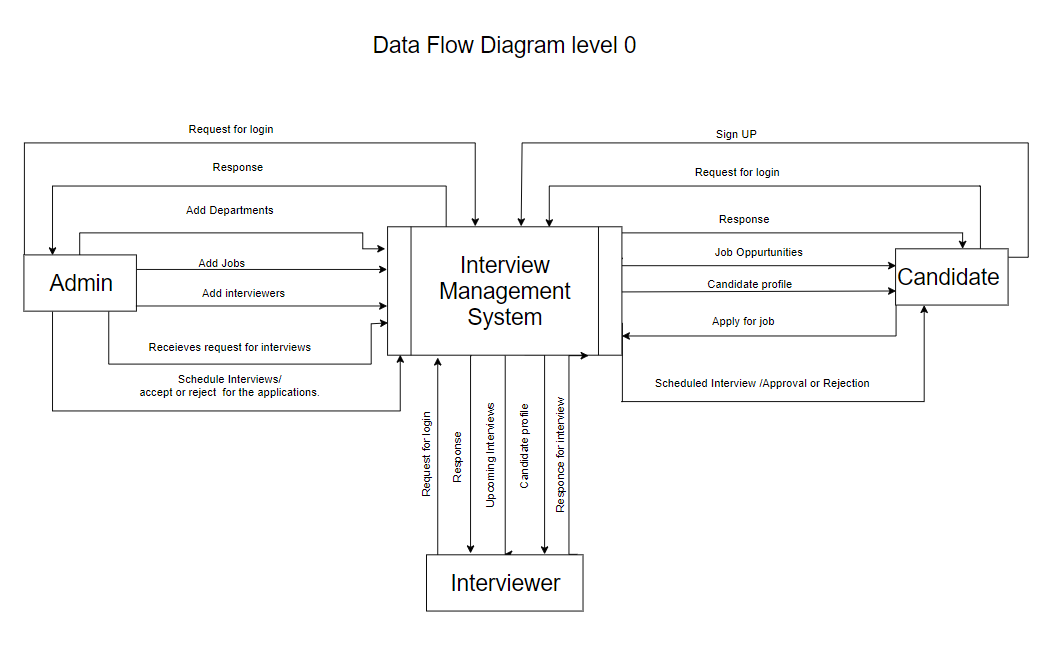
Software design is actually a multi-step process that focuses on four distinct attributes of a program: Data Structures, Software Architectures, Interface Representations, and Procedural (algorithm) details. The design process translates requirements into a representation of the software that can be assessed for quality before code generation begins.

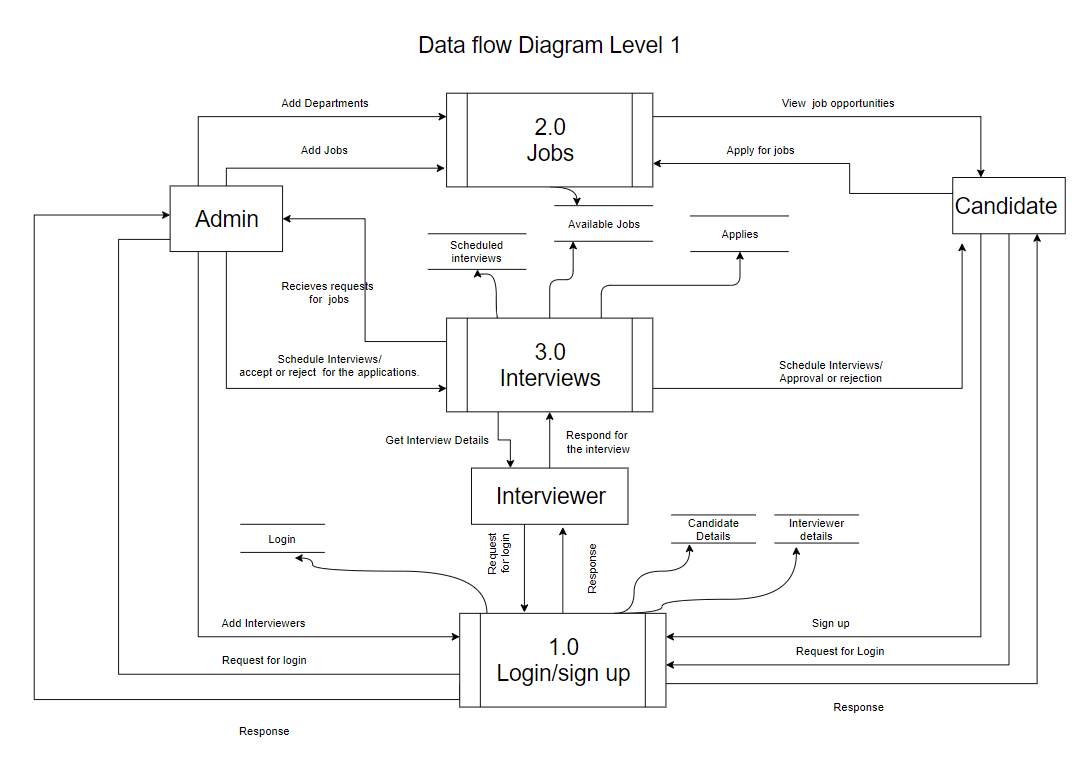
3.1 Structural Design

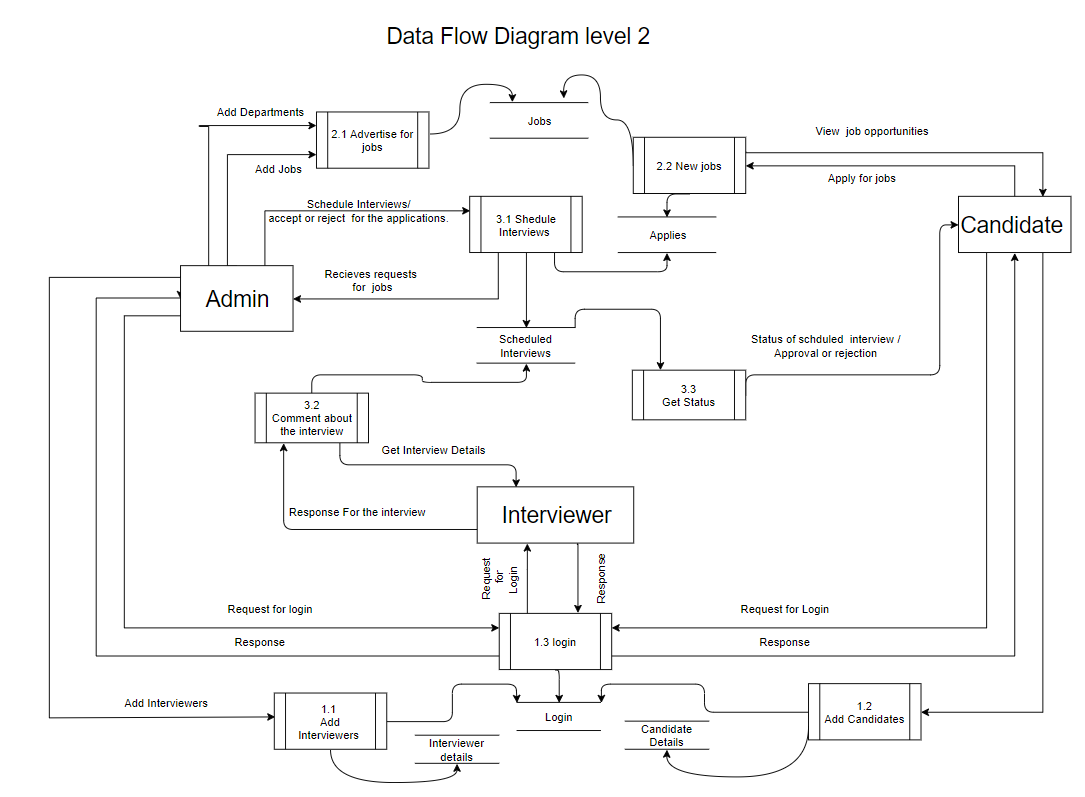
3.1.1 Entity Relationship Diagram (For database Design)



3.1.2 Data Flow Diagrams







A brief discussion about the modules used in Dfd

**1.0 Login/sign up**

Module name itself defines its function this module will be used to

Authenticate the different users as well as their signing up.

**1.1 Login**

This section will check the validity of the input credentials through database named logins.

**1.2 Add Candidates**

In this module, we will help us add new entry of candidates in the db named candidate as well as the logins.

**1.3Add Interviewers**

In this module, we will help us add new entry of interviewers by the admin in the db named interviewers as well as the logins.

**2.0 JOBS**

This module will help us to advertise as well as see /apply new opportunities for the candidates.

**2.1 Advertise jobs**

In this module, we will add the jobs Entered by the admin.

**2.2 New Jobs**

In these modules candidates can see as well as apply for the new job opportunities.

**3.0 Interviews**

This module will be the core of our project the main functioning will be done here.

**3.1Schedule interviews**

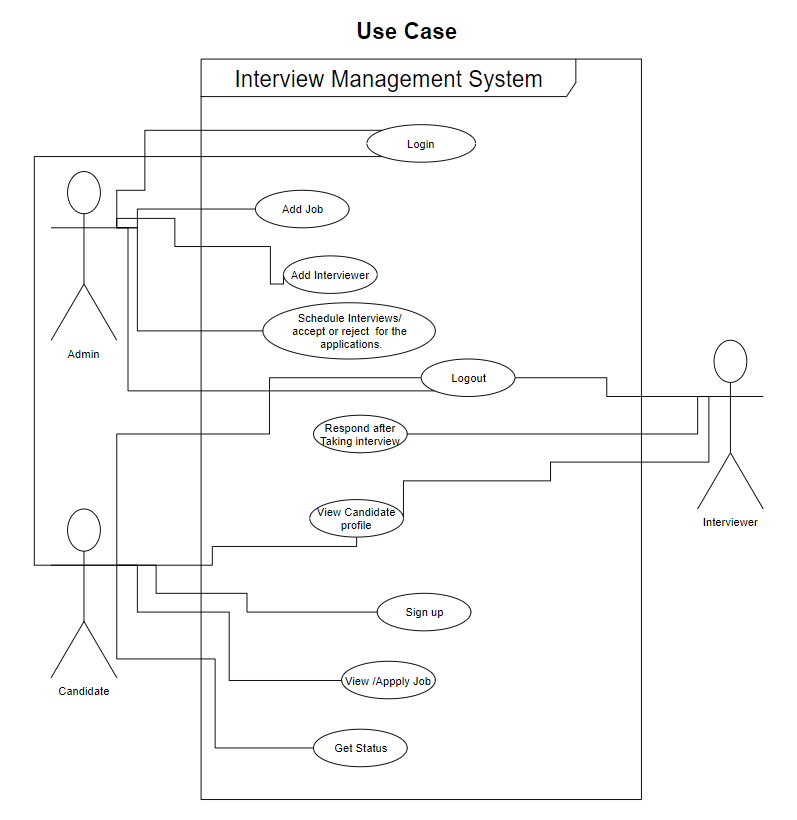
Admin will schedule new as well as the next interviews for the candidates and in the end select or reject the application.

**3.2 Comment about the interviews**

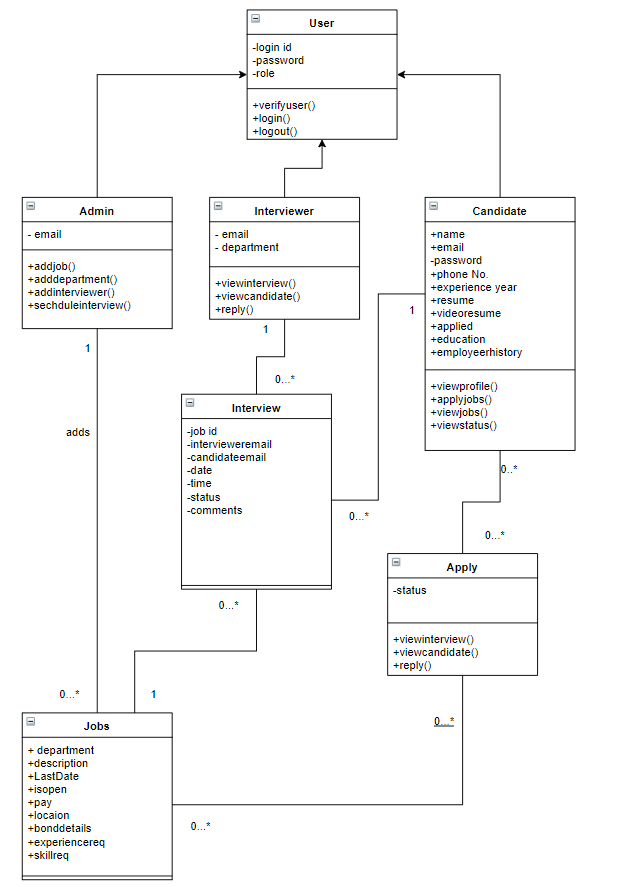
Here interviewer will see his upcoming interviews as well as reply after taking the interviews.

**3.2 Object Oriented Designing** :

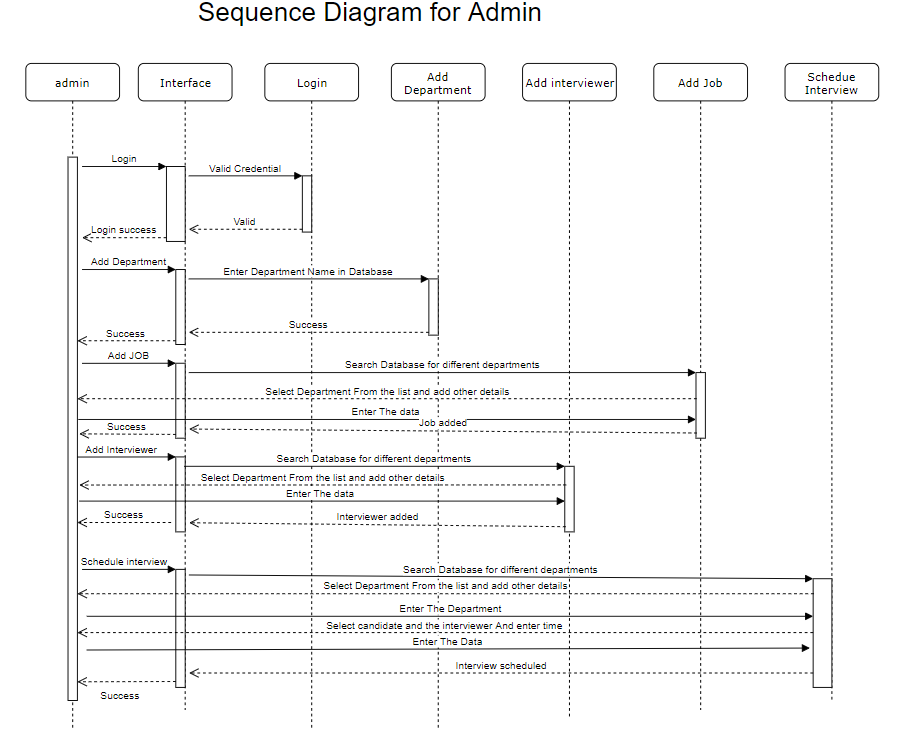
3.2.1 Use Case Diagram **:**

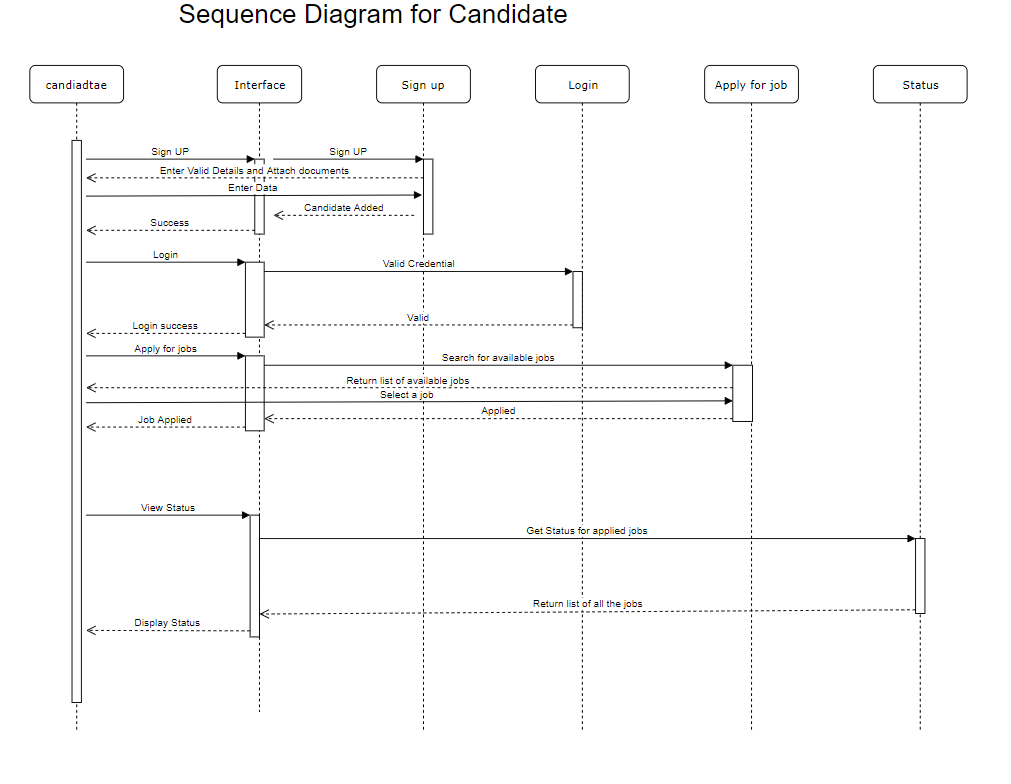


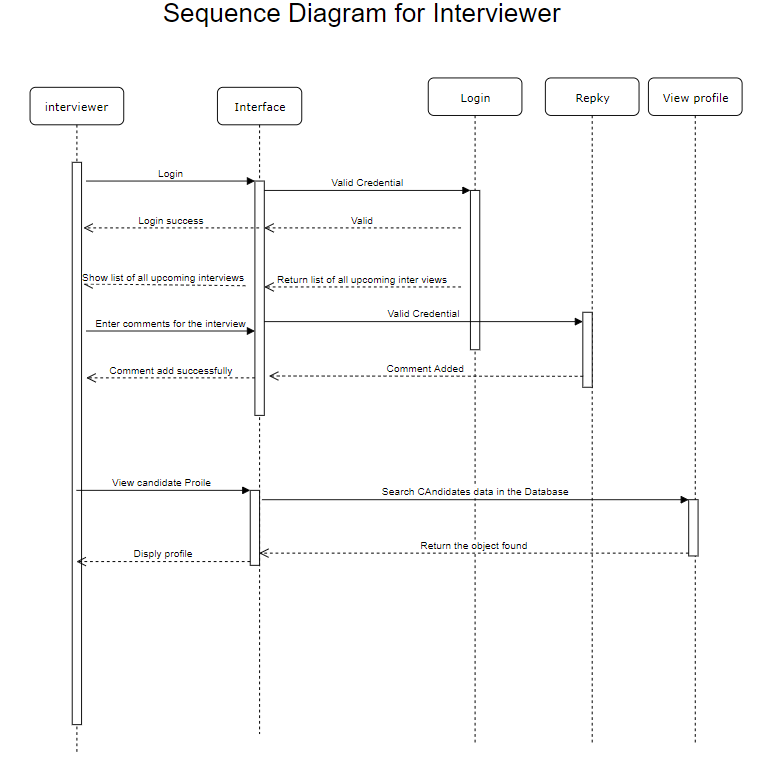
3.2.2 Class Diagrams



3.2.3 Sequence Diagrams







**4. System Testing**

**4.1 Unit Testing:**

Unit testing is undertaken when a module has been created and successfully reviewed. In order to test a single module we need to provide a complete environment i.e.

-The procedures belonging to other modules that the module under test calls

- Non local data structures that module accesses

-A procedure to call the functions of the module under test with appropriate parameters Unit

testing was done on each and every module

1. Log-in Module:

This module is used for candidate /interviewer/admin’s sign-in using his email-id and password. If both details entered by the user are correct then he will be redirected to homepage else an error saying invalid credentials will show on login page and ask again for email-id and password.

This test was done using dummy user details and it worked fine.

2. Sign-up Module:

This module is for for candidate/ admin registration. Here, candidate /admin should enter asked details and if no other user/restaurant with same credentials existed then registration will be completed else it will generate saying that user with these details already exists.

This test was done by registering fake user/restaurant details and it worked well.

Like this, all others modules were tested and they all worked fine.

**4.2 Integrated Testing :**

In this testing we test various integration of the project module by providing the input .The primary objective is to test the module interfaces in order to ensure that no errors are occurring when one module invokes the other module.

In this testing first we check a single module then we check it’s integrations with other modules i.e., the modules which are being invoked by it one by one.

For this project first I did testing for each module individually in the unit testing.

After that I did test for each module along with the modules invoked by it and it did work fine.

For e.g.

I did integration testing on signup module. At first, I entered details of a user who already existed, so it generated an error saying that user with these details already existed.

Then I entered new details and my registration was done and then I was redirected to LogIn page which was supposed to happen.

So, it worked fine.

Same way, I tested all other modules and they all worked fine.