

Software Design Specifications

CONNECT2WORK

Version: [00.01]

Project Code	
Supervisor	Dr. Muhammad Rafi
Co Supervisor	Kashif Noor
Project Team	Umer Alam (18k-0164) Burhanuddin Shabbir (18k-0213) Ishaq Ahmed (18k-0295)
Submission Date	

Document History

[Revision history will be maintained to keep a track of changes done by anyone in the document.]

Version	Name of Person	Date	Description of change
01	Burhan	04-Dec-2021	Changes in Activity Diagram
02	Umer	05-Dec-2021	Changes in system Architecture

Distribution List

[Following table will contain list of people whom the document will be distributed after every sign-off]

Name	Role
Dr. Muhammad Rafi	Supervisor
Kashif Noor	Co Supervisor

Document Sign-Off

[Following table will contain sign-off details of document. Once the document is prepared and revised, this should be signed-off by the sign-off authority.

Any subsequent changes in the document after the first sign-off should again get a formal sign-off by the authorities.]

Version	Sign-off Authority	Project Role	Signature	Sign-off Date

Document Information

Category	Information
Customer	FAST-NU
Project	Connect2Work
Document	Software Design Specification
Document Version	1.0
Status	Draft
Author(s)	Ishaq Ahmed – k180295 Burhanuddin – k180213 Umer Alam – k180164
Approver(s)	Dr. Muhammad Rafi
Issue Date	
Document Location	
Distribution	Advisor Project Coordinator's Office (through Advisor)

Definition of Terms, Acronyms and Abbreviations

[This section should provide the definitions of all terms, acronyms, and abbreviations required to interpret the terms used in the document properly.]

Term	Description
ASP	Active Server Pages
DD	Design Specification
MVT	Model View Template

Table of Contents

1	Introduction	7
1.1	Purpose of Document	7
1.2	Intended Audience	7
1.3	Document Convention	7
1.4	Project Overview	7
1.5	Scope	8
2	Design Considerations	9
2.1	Assumptions and Dependencies	9
2.2	Risks and Volatile Areas	10
3	System Architecture	11
3.1	System Level Architecture	11
3.2	Software Architecture	11
4	Design Strategy	12
5	Detailed System Design	13
5.1	Database Design	13
5.1.1	ER Diagram	13
5.1.2	Data Dictionary	13
5.1.2.1	Data 1	13
5.1.2.2	Data 2	13
5.1.2.3	Data n	13
5.2	Application Design	15
5.2.1	Sequence Diagram	15
5.2.1.1	<Sequence Diagram 1>	15
5.2.1.2	<Sequence Diagram 2>	16
5.2.1.3	<Sequence Diagram n>	17
5.2.2	State Diagram	18
5.2.2.1	<State Diagram 1>	18
5.2.2.2	<State Diagram 2>	18
5.2.2.3	<State Diagram n>	18
6	References	19
7	Appendices	20

1 Introduction

1.1 Purpose of Document

The software design specification is a description of the software components and sub-systems to be provided as part of the product, in this case, it is Connect2Work. The guide includes an overview of the thought process, the nature of development, the workflow, the major conceptualizing concepts and driving force, and a list of business satisfying points for which the product attempted to fill gaps.

1.2 Intended Audience

This document is intended for anyone interested in the design and analysis decisions made by the developers, as well as future engineers who want to understand the nature of the project and how it became what it became, with an emphasis on what concepts and features were implemented. This report will go over the product's lower-level details as well as the higher-level business use cases.

1.3 Document Convention

Font style: Arial

Font size: 16 for main headings, 14 for sub-headings, 12 for sub-sub-headings and 10 for paragraphs

1.4 Project Overview

The blue-collar worker is perceived to make less than the white-collar worker, while getting their hands dirty doing manual labor or working in a division of manufacturing. Connect2Work is a two-sided marketplace that connects 'Recruiter' (people who need help), with Connect2Work's network, a network of pre-approved and background checked individuals who have the time and skills needed to complete the listed task. Hailed as godsend to people with more money than time, it allows blue-collar workers to get work on daily basis by empowering them to select or reject a job upon their own will and their own determined pay rate.

Connect2Work is a freelancing type site which empowers blue-collar to come forward and take their place in society along with others. It also provides a lot of freedom and flexibility which regular jobs cannot, by giving them freedom to choose. The system is divided into multiple sub modules. Users of the system are job poster (Recruiter) and job seeker (Worker). Input of the system are details of the Recruiter, Worker, Job.

Outputs of the system are rating for a job completed, Recruiter feedback about the job done, Notifications, Weekly Schedule maintenance. Processes of the system are collecting data, analyzing data and job posting, generate weekly schedule and complete the schedule.

Approach used will OBJECT-ORIENTED for its development.

1.5 Scope

Connect2Work is a freelancing type site which empowers blue-collar to come forward and take their place in society along with others. It also provides a lot of freedom and flexibility which regular jobs cannot, by giving them freedom to choose.

The system is divided into multiple sub modules. Users of the system are job poster (Recruiter) and job seeker (Worker). Input of the system are details of the Recruiter, Worker, Job. Outputs of the system are rating for a job completed, Recruiter feedback about the job done, Notifications, Weekly Schedule maintenance. Processes of the system are collecting data, analyzing data and job posting, generate weekly schedule and complete the schedule.

2 Design Considerations

[This section describes many of the issues which need to be addressed or resolved before attempting to devise a complete design solution. In other words, this section is used to formally set the groundwork for the system design.]

2.1 Assumptions and Dependencies

The list of assumptions at the moment are,

- A module for verifying the email address exists, but we did not manage the time to integrate this into the actual flow, so we're assuming that the email entered and given to us actually belongs to a user.
- A module to verify worker/recruiter identity already does not exist yet, so we're assuming that the information entered and given to us actually belongs to a verified citizen.

Dependencies:

- "react": "^17.0.1",
- "react-bootstrap": "^1.4.0",
- "react-dom": "^17.0.1",
- "react-icons": "^4.1.0",
- "react-password-strength": "^2.4.0",
- "react-password-strength-bar": "^0.3.2",
- "react-responsive": "^8.2.0",
- "react-responsive-cards": "^3.1.1",
- "react-router": "^5.2.0",
- "react-router-dom": "^5.2.0",
- "react-scripts": "4.0.1",
- "react-search": "^2.1.1",
- "react-search-autocomplete": "^1.0.6",
- "reactstrap": "^8.7.1",
- "cors": "^2.8.5",
- "express": "^4.17.1",
- "mysql": "^2.18.1",
- "nodemon": "^2.0.6",
- "styled-components": "^5.2.1",
- "@fortawesome/fontawesome-free": "^5.15.1",
- "@fortawesome/fontawesome-svg-core": "^1.2.32",
- "@fortawesome/free-solid-svg-icons": "^5.15.1",
- "@fortawesome/react-fontawesome": "^0.1.13",
- "@material-ui/core": "^4.11.2",
- "@material-ui/icons": "^4.11.2",
- "@testing-library/jest-dom": "^5.11.6",
- "@testing-library/react": "^11.2.2",
- "@testing-library/user-event": "^12.5.0",
- "axios": "^0.21.0",
- "bootstrap": "^4.5.3
- asgiref 3.4.1
- Django 3.2.7
- django-html5 1.0.0
- django-rest-framework 3.12.4
- gunicorn 20.1.0

- *pip* 21.2.4
- *pytz* 2021.1
- *setuptools* 57.4.0
- *sqlparse* 0.4.2
- *wheel* 0.37.0

2.2 Risks and Volatile Areas

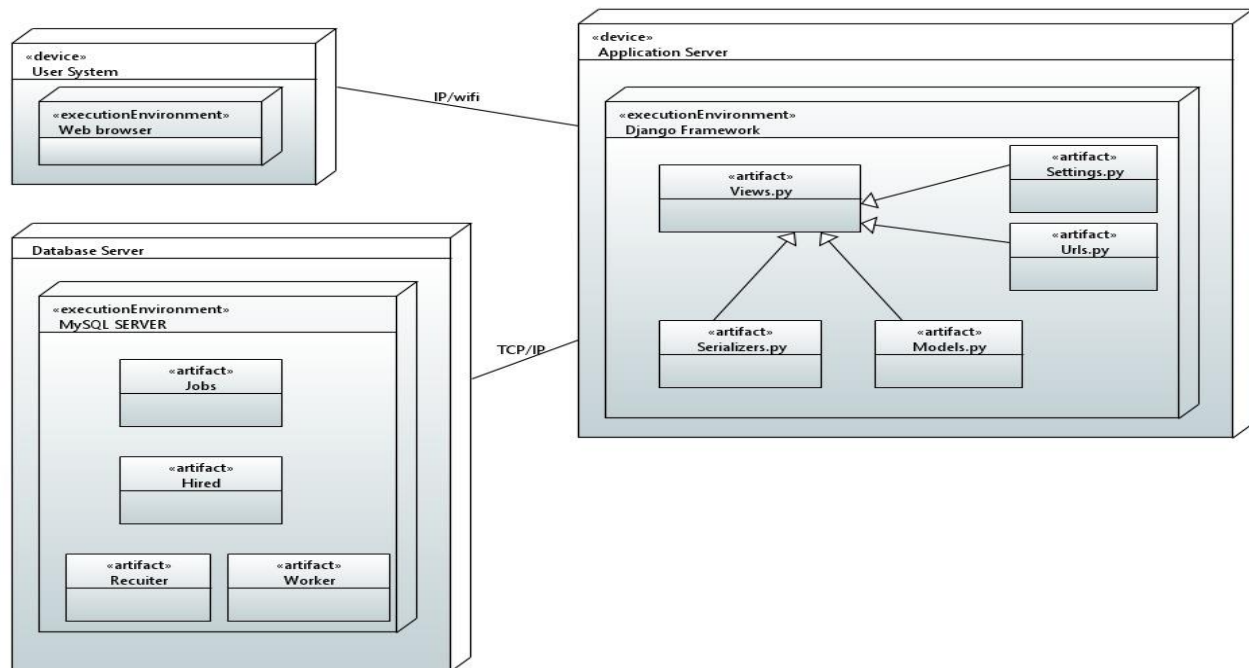
To minimize the risk factor, we will be using best practice of the industry known. Risks that are identified will be dealt at the time of construction for the software, new risks and threats will be dealt as system will work on.

We will be using an object-oriented modular approach in our system, so if a particular module or component technology becomes old it can be replaced with new technology and design pattern. We will be trying to achieve low coupling and strong cohesion.

3 System Architecture

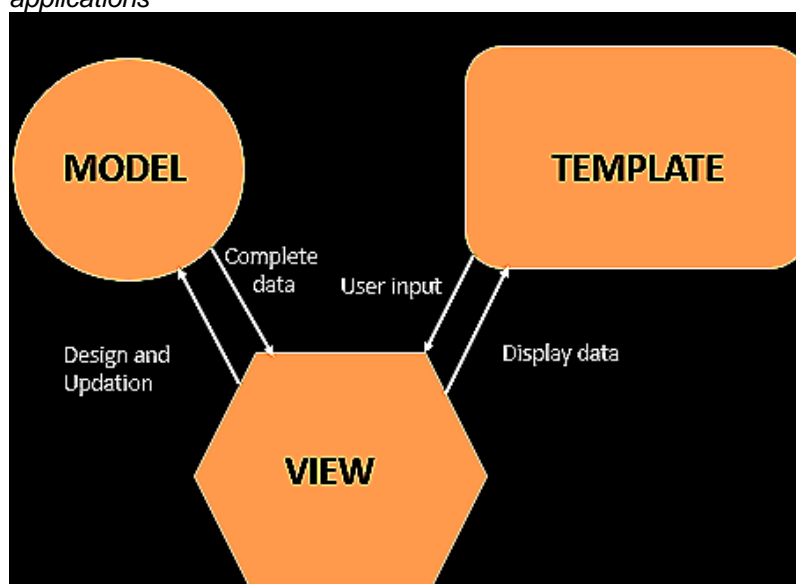
3.1 System Level Architecture

Deployment Diagram - Production



3.2 Software Architecture

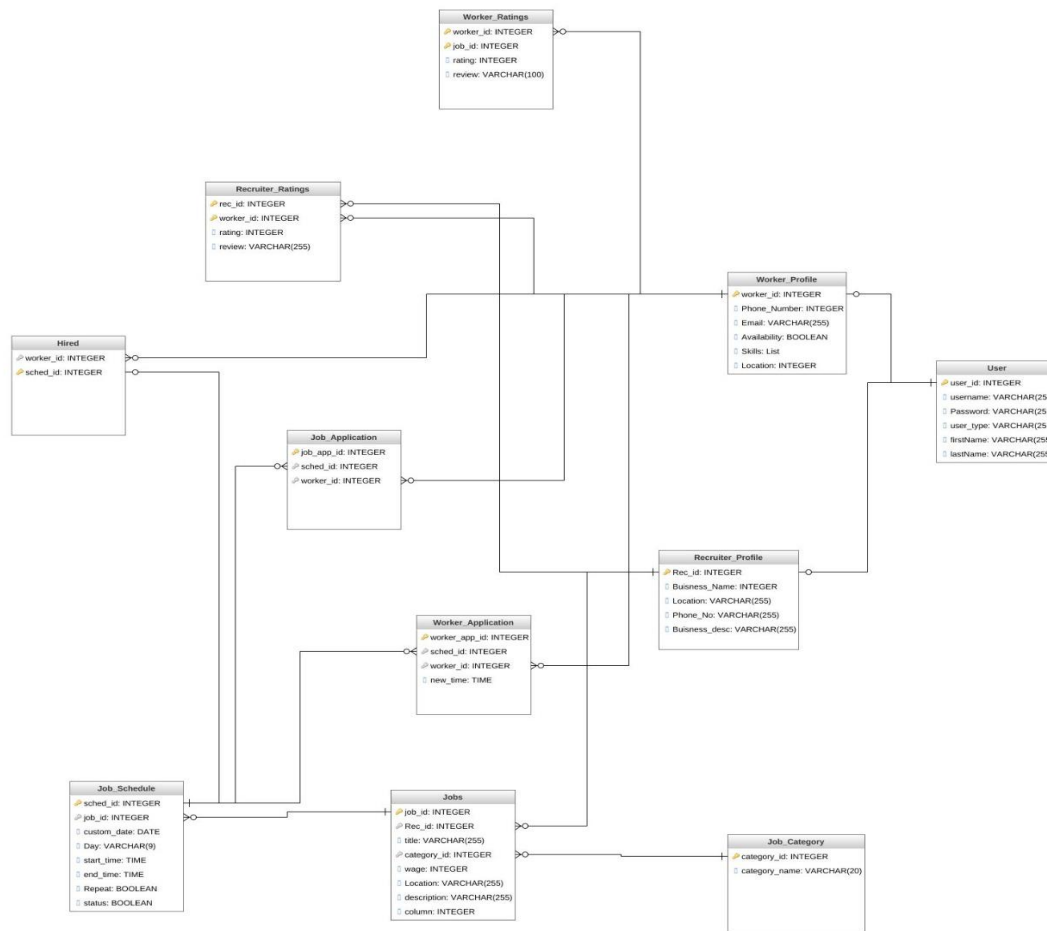
We are using MVT architecture because MVT has Views for receiving HTTP request and returning HTTP response. It is also Loosely coupled. Modifications are easy. Suitable for both small and large applications



4 Design Strategy

5 Detailed System Design

5.1 Database Design



5.1.1 ER Diagram

[Entity Relationship Diagram of the system with description]

5.1.2 Data Dictionary

[The convention recommended for writing the data dictionary is as follows.]

5.1.2.1 Data 1

[Description (Refer to Template on next page).]

5.1.2.2 Data 2

[Description (Refer to Template on next page).]

.

5.1.2.3 Data n

[Description (Refer to Template on next page).]

< Data 1>						
Name	Give primary name of the data or control item, the data store or an external entity.					
Alias	Write other names used for the first entry.					
Where-used/how-used	List all processes that use the data or control item and how it is used (e.g., input to process, output from the process, as a store, as n external entity)					
Content description	Notation for representing content.					
Column Name	Description	Type	Length	Null able	Default Value	Key Type
[Column1 Name]	[Description of the column]	[Type of column]	[Length of column]	[Is Column Null able]	[Default Value]	[If Primary Key than write PK, if Foreign Key then FK, if not a key leave it blank]
[Column2 Name]	[Description of the column]	[Type of column]	[Length of column]	[Is Column Null able]	[Default Value]	[If Primary Key than write PK, if Foreign Key then FK, if not a key leave it blank]

[Make similar tables for all the data items.

The notation to develop content description is given below:

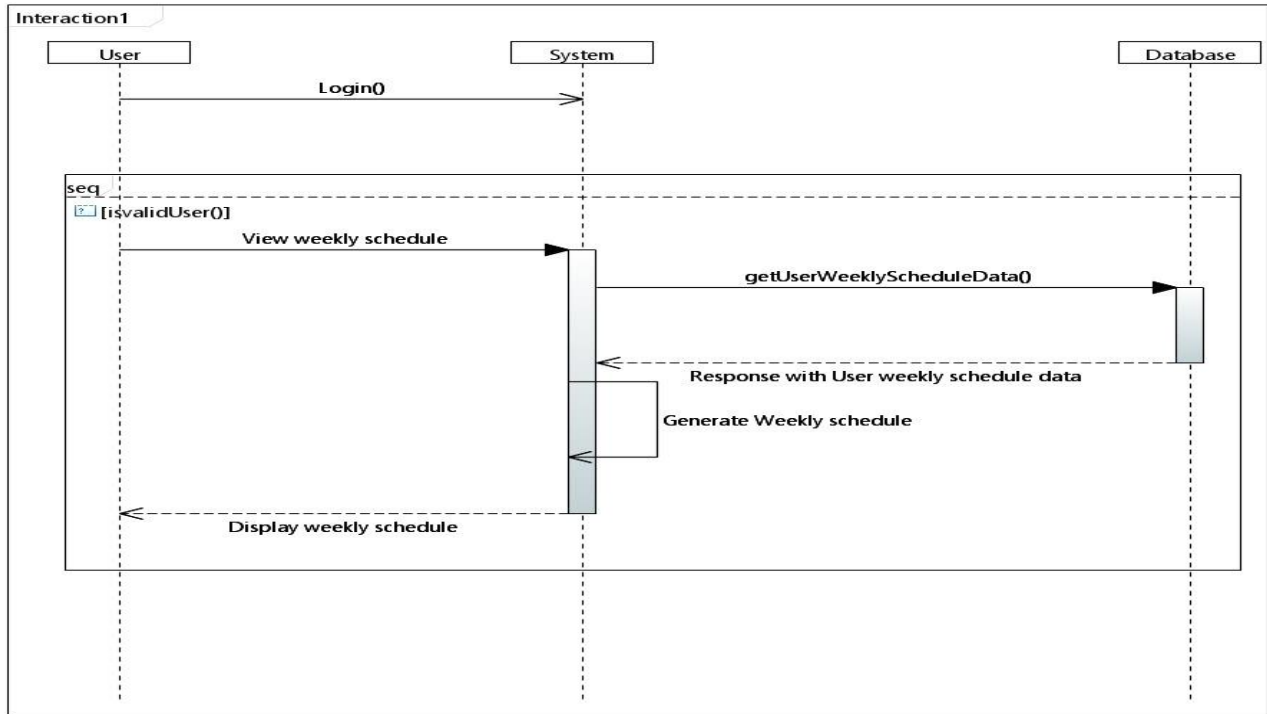
Data construct	Notation	Meaning
	=	is composed of
Sequence	+	And
Selection	[[]]	either-or
Repetition	{ }n	n repetitions of
	()	optional data
	* ... *	delimits comments
]		

5.2 Application Design

5.2.1 Sequence Diagram

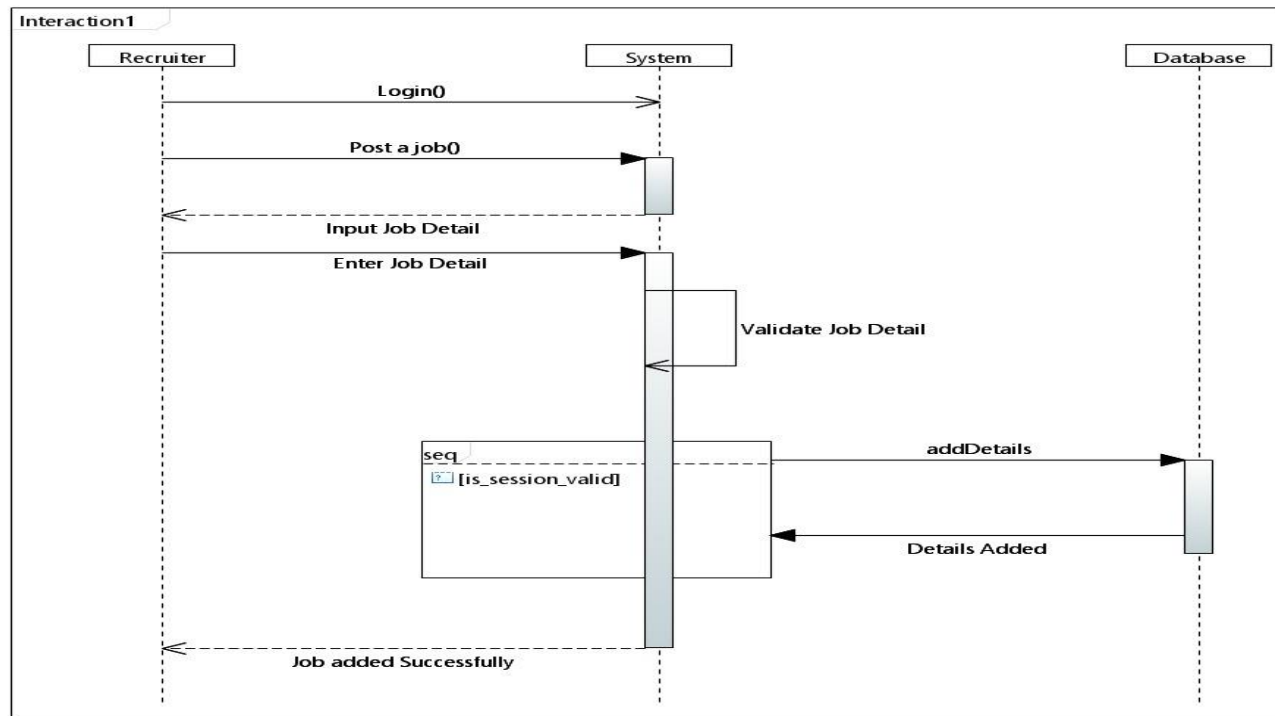
5.2.1.1 <Weekly Schedule Interaction>

This diagram defines the sequence of a user viewing weekly schedule maintained by system.



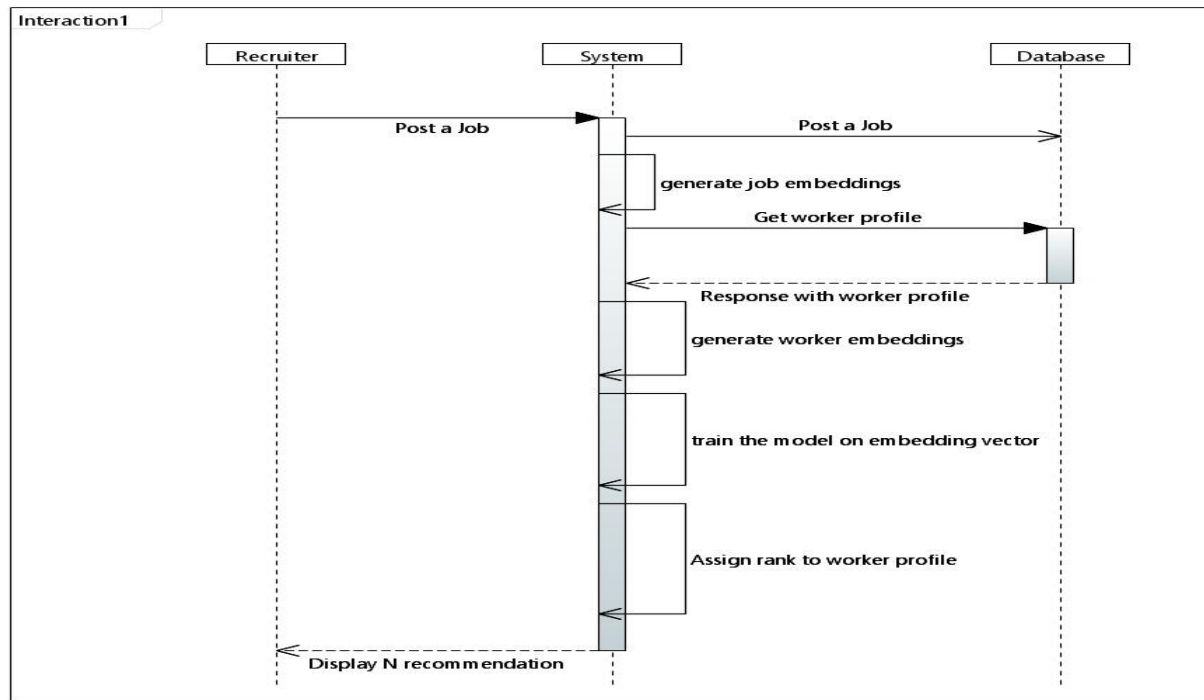
5.2.1.2 <Job Posting Interaction>

This diagram the sequence of a job posting by a recruiter and its validation in database.



5.2.1.3 <Matchmaking Algorithm >

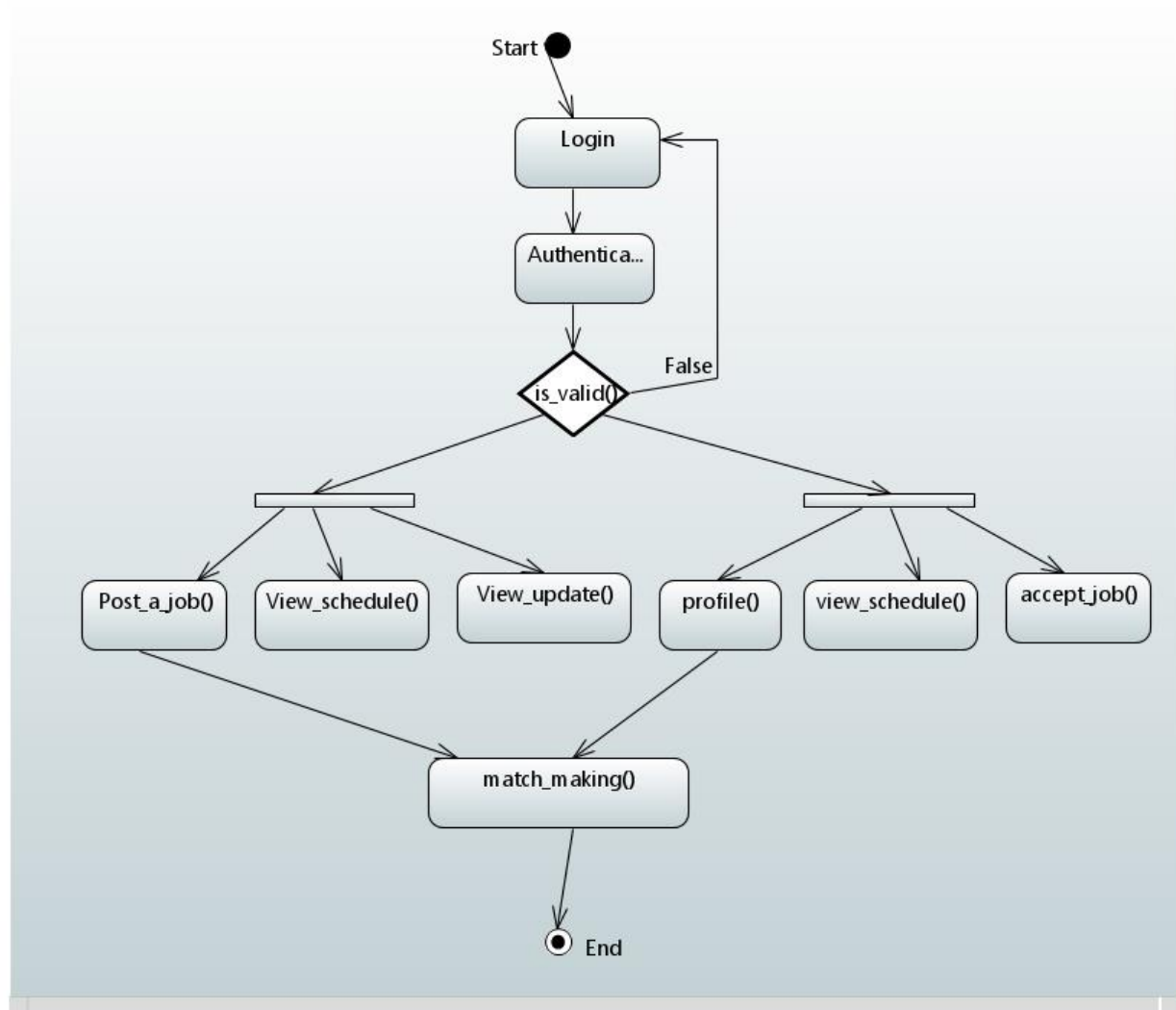
This diagram explains the working of the matchmaking algorithm, once a job is posted this algorithm will run and recommend the recruiter the top workers for the job



5.2.2 State Diagram

5.2.2.1 <System>

This diagram shows the working of the whole system, its start points, its functional separation according to the user and then ending at a point



5.2.2.2 <State Diagram 2>

[Diagram & Explanation of diagram]

.
.
.

5.2.2.3 <State Diagram n>

[Diagram & Explanation of diagram]

6 References

For reference purposes we visited many similar products currently available on website and a research paper similar to our app.

[1] WORKSAPP, May 2020, *International Research Journal of Engineering and Technology (IRJET)*, <https://www.irjet.net/archives/V5/i1/IRJET-V5I1143.pdf>

2. www.taskrabbit.com

3. www.zaarly.com

4. www.needto.com

7 Appendices

[Include supporting detail that would be too distracting to include in the main body of the document.]