A Minor Project Mid-Term Report on

RojgarNepal

Submitted in Partial Fulfilment of the Requirements for

The Degree of **Bachelor of Engineering in Information Technology**

Under Pokhara University

Submitted by:

Aayusha Regmi, 201502

Abhinab Khanal, 201503

Raaz Gupta, 201545

Sandesh Khatiwada, 201547

Date:

15/06/2024

Department of Information Technology

NEPAL COLLEGE OF INFORMATION TECHNOLOGY

Balkumari, Lalitpur

ABSTRACT

RojgarNepal is a web-based freelancing platform that caters to the requirements of Nepali market. In response to the evolving needs of the Nepali market, RojgarNepal proposes to introduce a platform that addresses services often challenging to locate locally. There are two kinds of users in the system: clients and freelancers. Client can submit jobs on the platform. Freelancers can apply to the jobs posted by the clients. The client can directly communicate with the freelancers through an integrated chat system. Once client submits job, s/he can invite freelancers that are recommended to him/her based on preferences,else s/he can make a post that will be visible to every freelancers in the platform. Freelancers can respond to the posts by sending a short message to convince client to book their service. The client can have conversation with freelancers through the chat system and can book one freelancer based on his/her preference. The integration of online payment gateway will facilitate the payment process once the job is done. The frontend of the application is being built using ReactJS. Node and ExpressJS is being used for backend programming. RojgarNepal thus attempts to revolutionize the freelance landscape in Nepal, empowering individuals and businesses to thrive in the digital economy.

Keywords : Client,Freelancers,ReactJS,Node,ExpressJS

Table Of Contents

1. Introduction....1
1.1 Problem Statement....1
1.2 Project Objectives....1
1.3 Significance Of The Study....2
1.4 Scope And Limitations....3
2. Literature Review....4
3. Methodology....5
3.1 Software Development Lifecycle....5
3.1.1 Requirement Analysis....6
3.1.2 System Design....6
3.1.3 Coding....6
3.1.4 Testing And Evaluation....6
3.2 Tools Being Used....7
3.3 Technologies Being Used....7
4. System Design....9
4.1 Entity-Relationship (Er) Diagram:....9
4.2 Use Case Diagram....10
4.3 Class Diagram....11
4.4 Sequence Diagram....12
5. Task Done So Far....14
6. Results And Discussion....15
7. Task Remaining....16
8. Deliverables....17
9. Project Task And Time Schedule....18
9.2. Time Schedule....18

9.2 Gantt Chart..19

7. References..21

LIST OF FIGURES

Figure 1: Incremental Model of Software Engineering.....5

Figure 2: Entity-Relationship (ER) Diagram.....9

Figure 3: Use Case Diagram.....10

Figure 4: Class Diagram.....11

Figure 5: Sequence Diagram for User Registration.....12

Figure 6: Sequence Diagram for User Login.....13

Figure 7: Sequence Diagram for Posting a Job.....13

LIST OF TABLES

Table 1: Time Schedule.....18

Table 2: Iteration I.....19

Table 3: Iteration II.....19

Table 4: Iteration III.....20

1. INTRODUCTION

This project carries the motive of revolutionizing the freelancing landscape in Nepal. It addresses the challenges faced by both service providers and seekers by providing a user-friendly platform that offers a wide range of services. In this introduction, we'll explore the problems faced by freelancers and clients, our motivations, objectives, project scope, and any limitations to be considered.

1.1 PROBLEM STATEMENT

In today's rapidly evolving job market, accessing suitable freelance opportunities poses a significant challenge for individuals seeking for work arrangements. The traditional methods of finding freelance projects, such as relying on personal networks or engaging in direct negotiations with clients, often leads to being inefficient and time-consuming. This challenge is particularly active in urban hubs like Kathmandu, Pokhara, and Biratnagar, where the demand for freelance services is high, yet the platform for connecting freelancers with clients is limited.

For a country like Nepal, there are certain jobs that are high-in demand but are not incorporated in popular freelancing platforms. Some of these jobs include carpentry,laundry, cleaning service, aged care etc.

RojgarNepal carries the motive to revolutionize the freelancing landscape in Nepal by providing a user-friendly platform that connects freelancers with clients.

1.2 PROJECT OBJECTIVES

To address the challenges outlined in the problem statement for RojgarNepal, the platform is introduced with the following objectives:

i. To develop a web application that serves as an inclusive platform, specifically designed to cater to the Nepali market by incorporating low-skill but high-demand jobs.
ii. To facilitate direct communication between freelancers and client through an integrated chat system, to simplify the job-dealing process.
iii. To facilitate the payment process, through an integrated online payment system, once the job is completed.

1.3 SIGNIFICANCE OF THE STUDY

The advent of RojgarNepal represents a significant milestone in Nepal's digital evolution,introducing a bespoke freelancing platform tailored to local needs. Beyond mere convenience, RojgarNepal serves as a catalyst for economic empowerment, providing skilled professionals with newfound income opportunities, thereby mitigating unemployment and fostering economic growth. Moreover,the significance of the revolutionizing freelancing platform RojgarNepal are as follows:

i. RojgarNepal creates alternative income opportunities, reducing unemployment and stimulating economic growth, offering skilled professionals a new pathway to earn a living.

ii. With a clear understanding of the limitations of existing job search platforms, RojgarNepal distinguishes itself by incorporating jobs that are typically overlooked by other freelancing platforms. This inclusive approach ensures that individuals with diverse skill sets and from various industries have access to meaningful employment opportunities.

iii. The platform connects clients and freelancers across Nepal, fostering broader participation in the digital economy, including rural and underserved areas, promoting inclusivity.

iv. An online payment gateway ensures secure and efficient financial transactions, building trust between clients and freelancers, encouraging more users to engage with the platform.

1.4 SCOPE AND LIMITATIONS

In the project, a system is being developed to create a centralized platform facilitating communication and deal-making between freelancers and clients. This platform will streamline the process of connecting freelancers with clients, enabling them to collaborate efficiently and securely in an online environment.

The scopes of this project are:

i. RojgarNepal will serve as a centralized platform connecting freelancers with clients, catering to various service needs in Nepal.
ii. The platform targets a broad audience, including freelancers seeking opportunities and clients requiring services across various industries.
iii. RojgarNepal aims to cater the needs of Nepali market by incorporating jobs that are hard to locate in other platforms.

The limitations of this project are:

i. Initially, RojgarNepal's services may be limited to specific locations, with a focus on major urban centres like Kathmandu.
ii. Users are expected to possess basic digital literacy to utilize RojgarNepal effectively, as the platform operates primarily online.
iii. Users are required to have access to internet connectivity and digital devices, limiting the accessibility of the platform to those with sufficient technological resources.

2. LITERATURE REVIEW

In this section, we have reviewed some of the existing platforms providing the similar service that are currently in use.

The first site we found was guru.com. It is a platform that connects employers and freelancers. One has to create an account first to get started with the platform. Users can register themselves as either employers or freelancers. Employers can post a job and freelancers provide quotation. Employer can select one among the freelancers that provide quotation to the posted job. However,this platform does not incorporate some jobs that are hard to find in digital platforms such as aged care,laundry,painting, carpentry,cleaning etc[01].

Secondly, we reviewed sriyog.com. It incorporates the low-skill jobs that were not found in the previous platform. However, the major hassle was there in the registration process itself. The freelancer has to register by filling up the form[02] and wait until the admin responds. So, the registration process itself was not so straightforward. Also, the platform does not include any recommendation system. This can make the overall process slower.

Thirdly, we reviewed a popular freelancing platform upwork.com. It is a globally popular platform specifically for high skill jobs such as Software Development and IT. However, the platform does not incorporate low-skill jobs that are high-in demand. Platforms like upwork might not be suitable for the needs of low-skill jobs in Nepali market[03].

In the this system, we are solving these issues by incorporating such low-skill but highly demanded jobs and providing an easy registration process and a recommendation system to speed-up the overall process for clients.

3. METHODOLOGY

We have deliberately chosen the following methodologies to effectively apply our expertise, skills, and techniques, ensuring precise alignment with project requirements.

3.1 SOFTWARE DEVELOPMENT LIFECYCLE

For the software development process of our project, we've opted for the Incremental Model, which involves building the system incrementally over multiple iterations. Each iteration encompasses the Analysis, Design, Coding, and Testing phases[04].

In the first iteration, our primary focus was on developing essential features including user authorization and authentication, job posting, adding freelancer's service details etc. These core components lay the foundation of our platform's functionality. Subsequent iterations will introduce additional features such as chat system, payment gateway integration, filter-based search and more, providing a step-by-step expansion of the platform's capabilities. The following subsection briefly describes various phases in the incremental model of SDLC that is being applied in the development of the system:

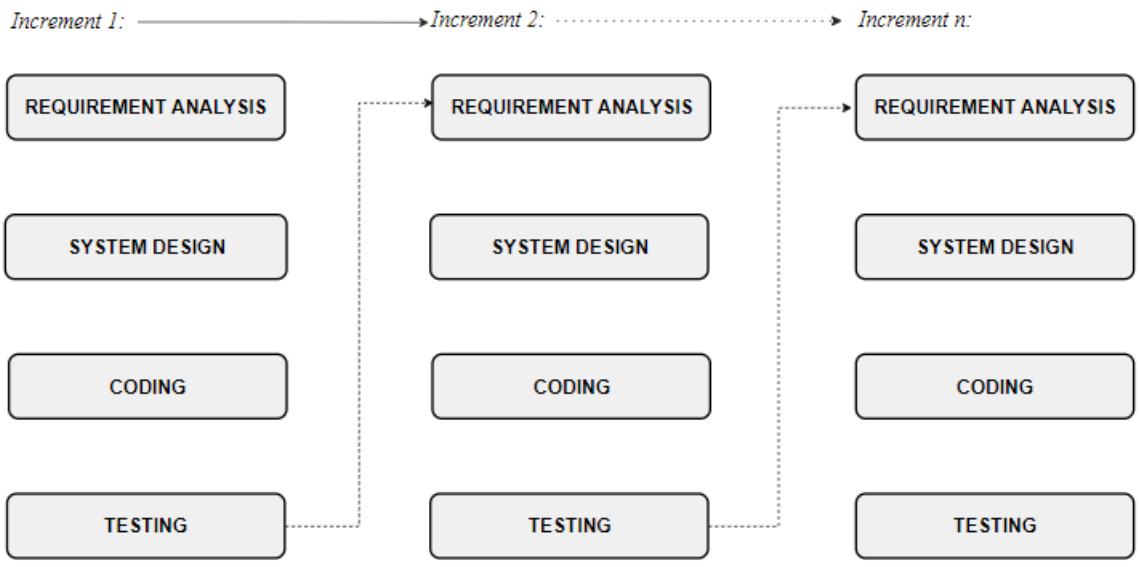

Figure 1: Incremental Model of Software Engineering

3.1.1 REQUIREMENT ANALYSIS

In the Requirement Analysis phase,requirement gathering and analysis was done, laying the groundwork for our project's success. In the first increment, we analysed the requirements for basic functionalities for the system such as authentication, authorization, job posting, freelancer’s service addition etc. In the subsequent increments, further polishing of the existing features and addition of new features is to be studied.

3.1.2 SYSTEM DESIGN

Transitioning from requirement analysis, the visualization of our project's architecture was done in the design phase. Here, we used the tools like MockFlow to craft intuitive user interfaces, leveraging insights from ER diagrams and Use Case diagrams. In the further increments, further design documents are to be prepared depending on the requirements specified.

3.1.3 CODING

In coding phase, the vision outlined in requirement documents and the design artifices was materialized into functional code. In this phase, we have implemented basic functionalities like user authentication and authorization, addition of service by freelancer, job postings etc. through code. Addition of new functionalities is to be done in subsequent increments.

3.1.4 TESTING AND EVALUATION

On the testing phase, we tested the API endpoints for basic functionalities through Thunderclient. The UI was also tested in this phase. The testing and evaluation will continue with the development of new features in next increments..

3.2 TOOLS BEING USED

We are using the following tools throughout this project to streamline the development process.

Git And GitHub:

Git is a distributed version control system that is being used in this project to manage different versions of the project and to collaborate with each other. GitHub is a platform that uses git for version control. We are using GitHub to host the project repository, to track changes and manage different versions of the project.

VS Code:

VS Code is a code editor that is being used as a primary code editor for development tasks.

THUNDER CLIENT:

Thunderclient is a tool that allows the testing of API endpoints. It is being used to test the REST APIs directly within the code editor, without switching to a separate application

3.3 TECHNOLOGIES BEING USED

We are using the following technologies during the project development task.

ReactJS:

ReactJS, often simply referred to as React, is an open-source JavaScript library primarily used for building user interfaces (UIs) for web applications[05]. We are using ReactJS to build the frontend of the application.

Node.js:

Node.js is being used as a server-side runtime environment[06] to build the backend of the system. It is an open-source runtime environment built on Chrome's V8 JavaScript engine.

Express.js :

Express.js, commonly referred to as Express, is a minimal and flexible web application framework for Node.js[07]. Express.js is being used to develop Rest API for the system.

MySQL:

We are using SQL (Structured Query Language) as a query language to interact with the database. MySQL is being used as a Relational Database Management System (RDBMS) for the project.

4. SYSTEM DESIGN

This section provides a comprehensive overview of the system's architecture focusing on key diagrams: Entity-Relationship (ER) diagram, Use Case diagram, Class Diagram, Sequence Diagram. These diagrams illustrate the core components and the interactions between them on the system.

4.1 ENTITY-RELATIONSHIP (ER) DIAGRAM:

An Entity-Relationship (ER) diagram is a visual representation of the entities(objects) within a system and the relationship between them. It is commonly used in database design to model the structure and the organization of data. The major entities involved in our system are User, Service,Payment,Conversation etc. The relationship between these entities is shown in the given ER diagram.

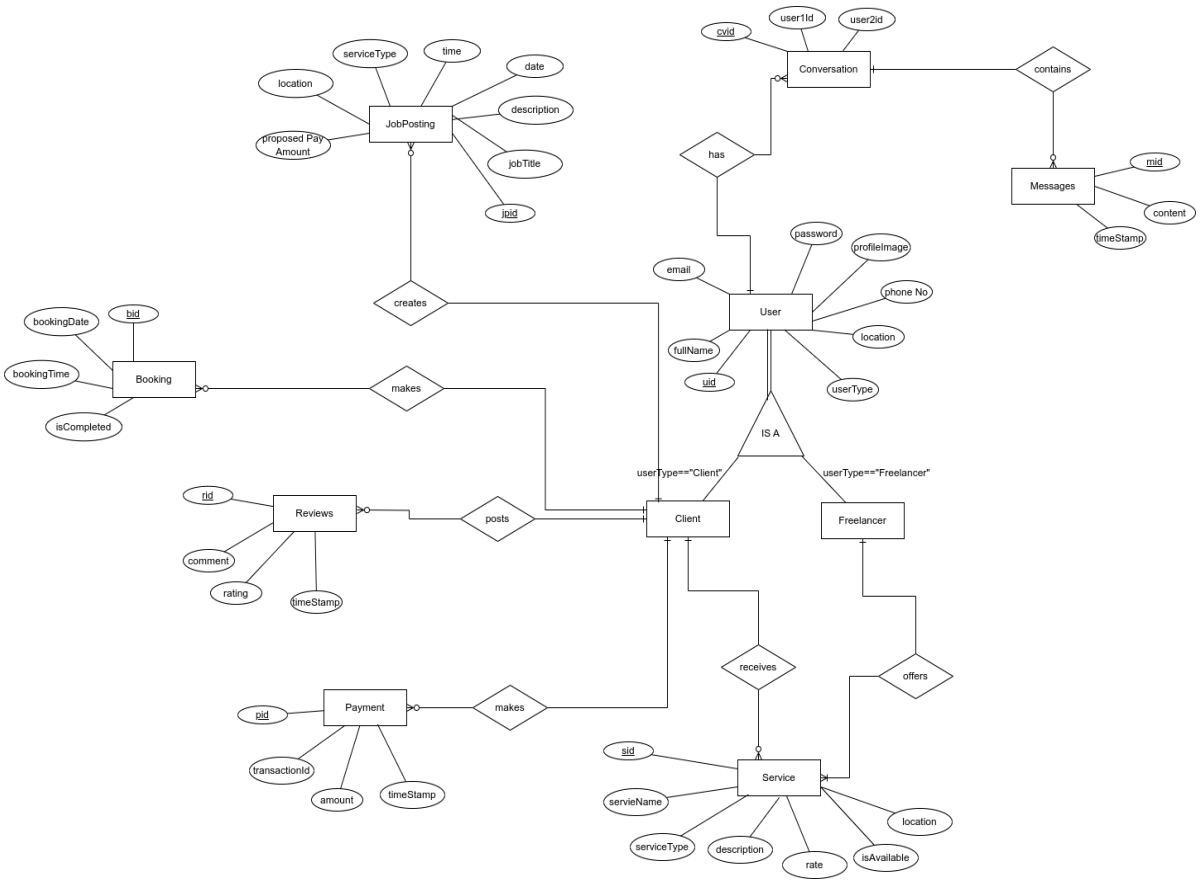

Figure 2: Entity-Relationship (ER) Diagram

4.2 USE CASE DIAGRAM

A use case diagram is a visual representation of the interactions between actors (In our case: Client, Freelancer and Admin) and a system under consideration. It shows the different use cases or functionalities provided by the system and the relationships between the actors and the use cases

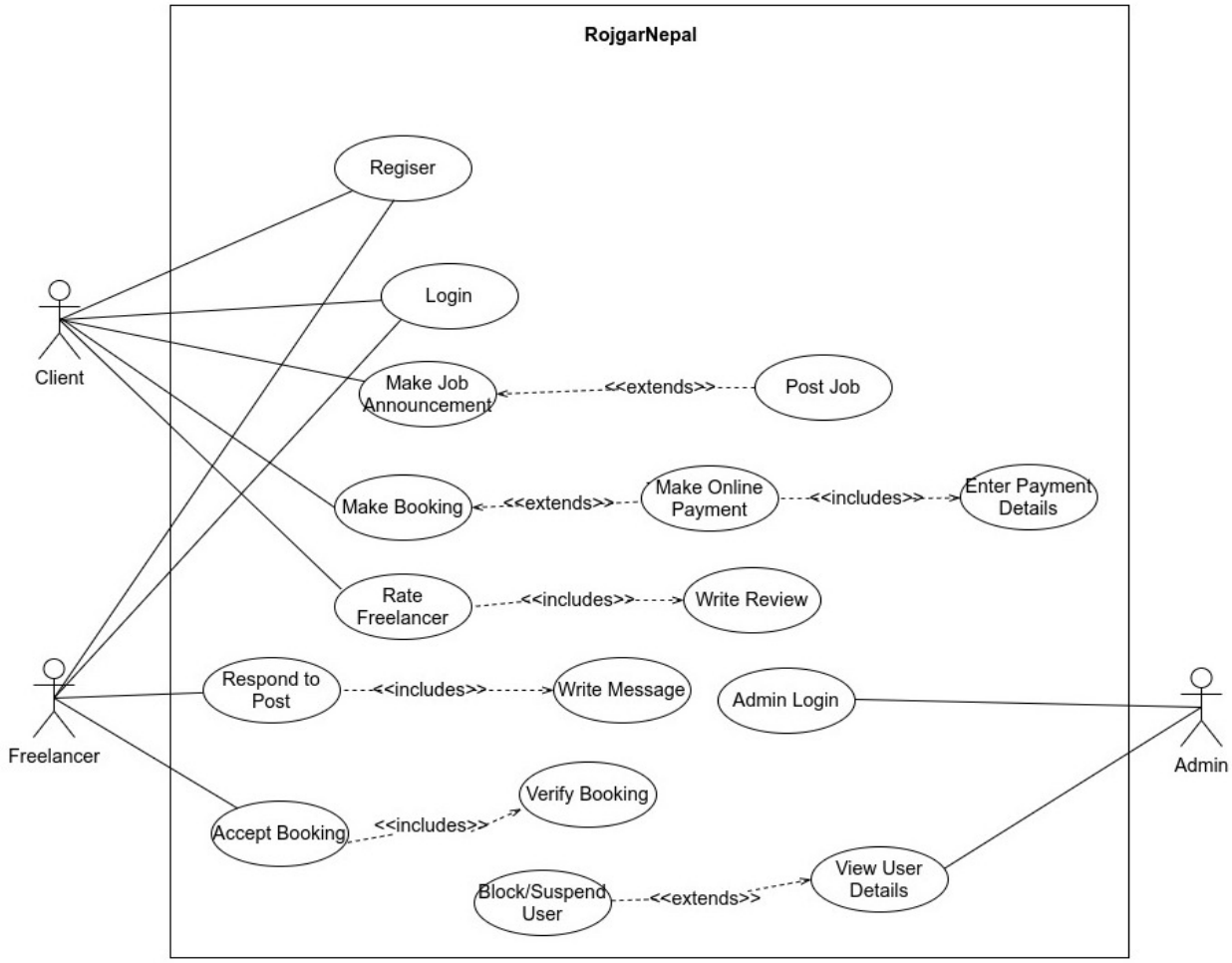

Figure 3: Use Case Diagram

4.3 CLASS DIAGRAM

Class diagram describes the structure of the system by showing its classes, attributes, operations (or methods), and the relationships among objects. The major classes involved in our system includes : User, Service, Booking, JobPosting, Reviews etc.

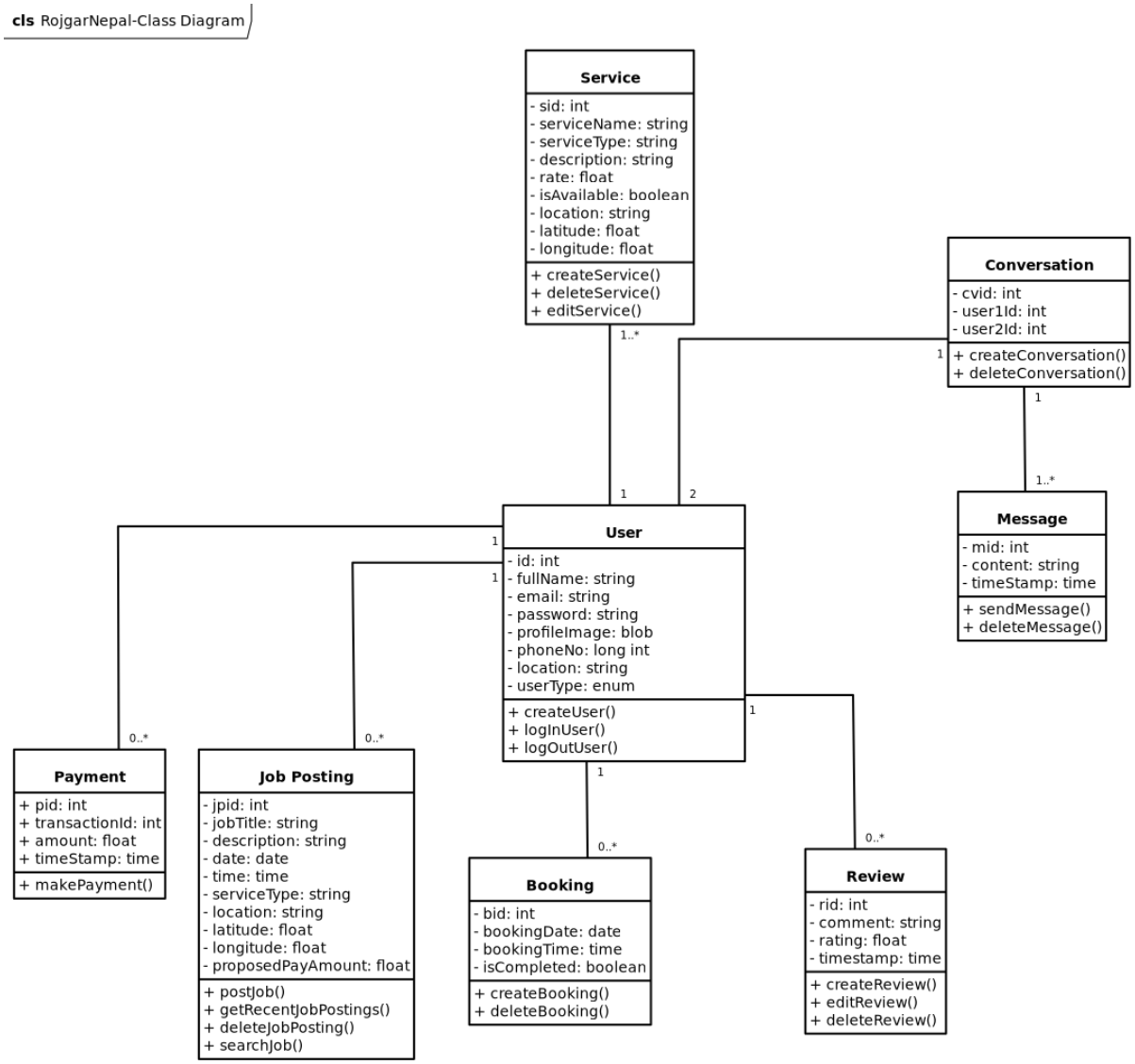

Figure 4: Class Diagram

4.4 SEQUENCE DIAGRAM

Sequence diagram illustrates how objects interact in a particular scenario of the system. Three scenarios that have been implemented till now are illustrated through sequence diagram in this section.

i. User Registration:

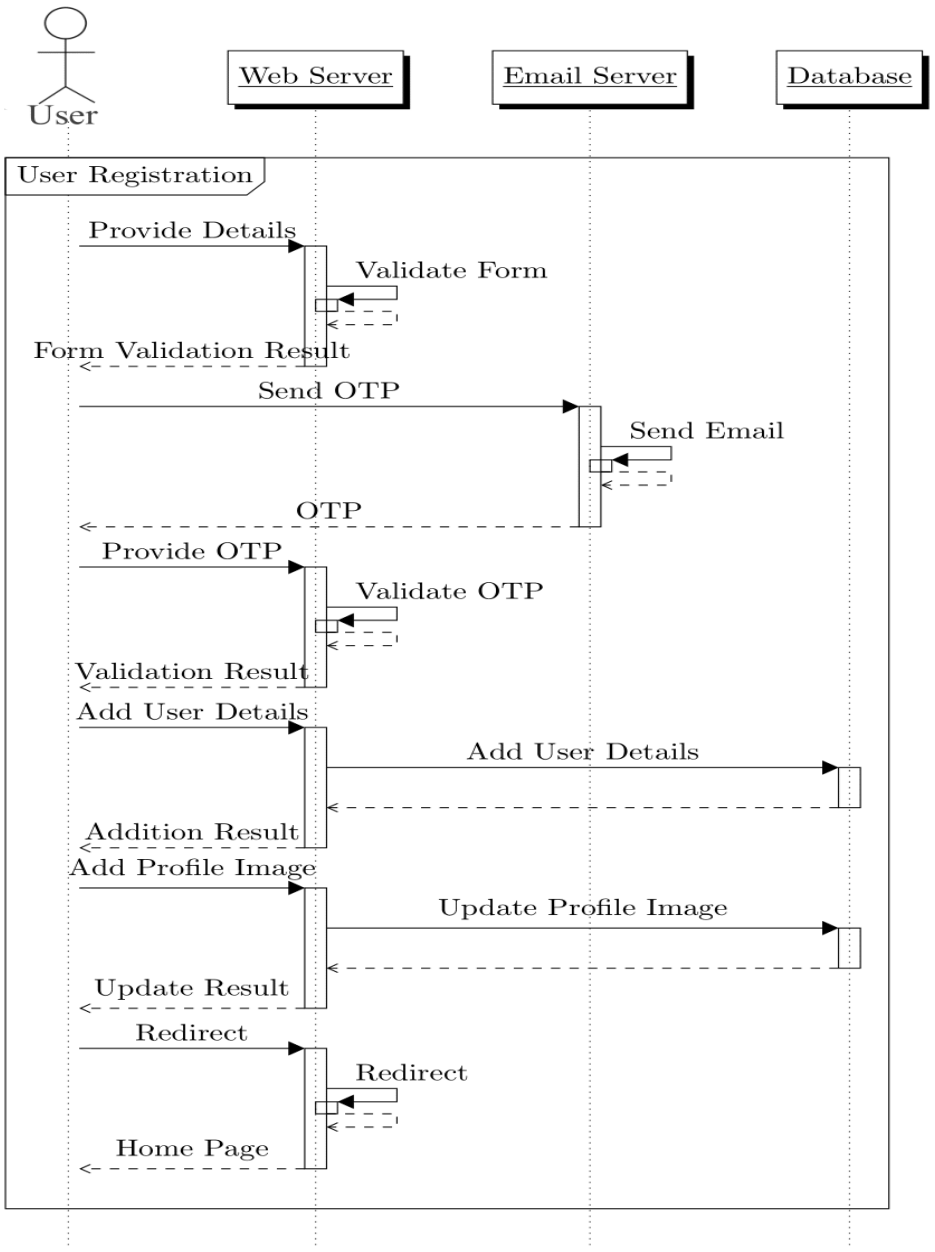

Figure 5: Sequence Diagram for User Registration

ii. User Login:

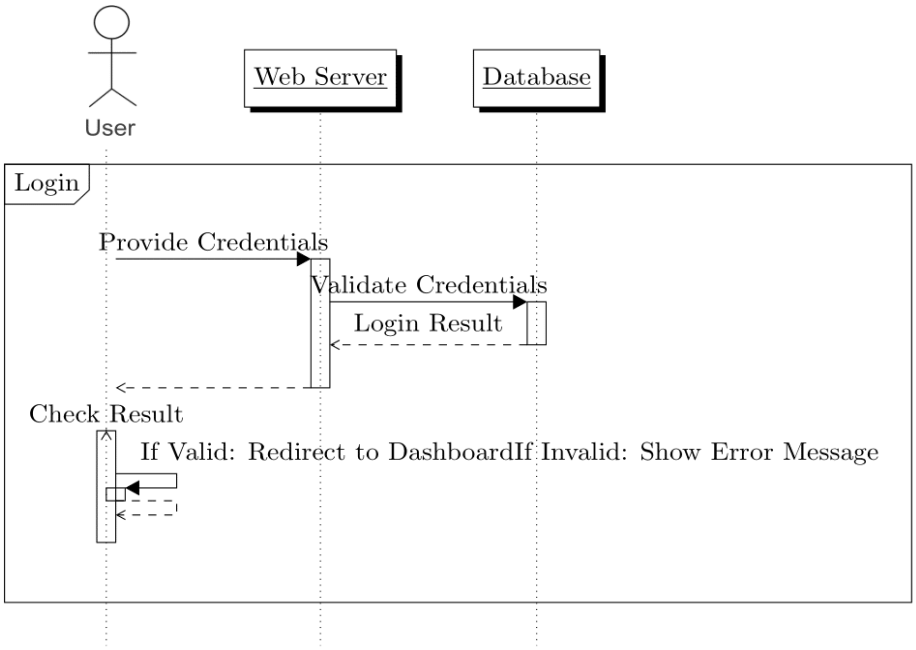

Figure 6: Sequence Diagram for User Login

iii. Posting Job

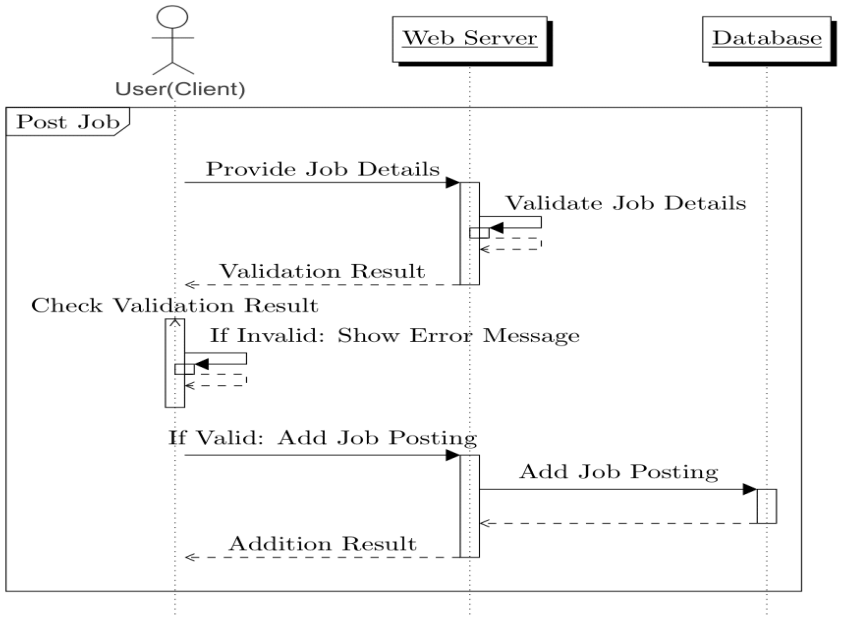

Figure 7: Sequence Diagram for Posting a Job

5. TASK DONE SO FAR

We have worked on basic functionalities of the system so far. These functionalities include:

i. User Authentication and Authorization:

We have developed User Authentication and Authorization functionality using stateless authentication approach. We have used jsonwebtoken for the authentication of users. So far, a user can signup up as a client or a freelancer and can login with the correct credentials. Proper validation logic is implemented in signup process.

ii. Verification of user's email by sending OTP:

We have developed a functionality to verify user's email address by sending One Time Password (OTP) to user's email address. This ensures only valid and correct email address is being used during sign-up process..

iii. Addition of Service by Freelancers:

A freelancer can add the service s/he provides in the platform through a form. S/he can provide service details such as service name, service type, description, service location etc. The details provided by the user is validated properly to ensure correctness of the data.

iv. Posting of Jobs:

A client can post jobs to the platform depending on his requirements. S/he can provide job details such as service he is seeking for, date, time, location etc. Interactive map allows client to provide the exact location so that freelancer can easily find out where client is seeking the service.

v. Integration of Interactive Map:

We have integrated interactive map in the service and job posting page, where client can mark his/her location while making a job post and freelancer can mark his/her location while adding service details.

vi. Client and Freelancer's Dashboard:

We have included dashboard for both clients and freelancers with the dynamic data showing the job posts, no of bookings, no of active projects etc.

6. RESULTS AND DISCUSSION

The development of the basic functionalities of our system has yielded significant progress towards achieving the project's objectives.

i. The implementation of user authentication and authorization using JSON Web Tokens (JWT) has been successful. Users can now sign up as either clients or freelancers and log in with their credentials. The validation logic during the signup process ensures that only correctly formatted and complete data is accepted.

ii. Freelancers can now add their services to the platform through a detailed form. The validation logic implemented ensures data integrity and accuracy. This feature is vital for building a robust service catalogue that clients can browse.

iii. Clients are able to post jobs specifying their requirements, including service type, date, time, and location. The integration of an interactive map allows clients to pinpoint exact locations, facilitating better coordination with freelancers. This feature significantly enhances user experience by providing a straightforward way for clients to communicate their needs.

iv. The interactive map has been successfully integrated into the service and job posting pages. Both clients and freelancers can now mark their respective locations, aiding in precise job coordination. The map integration has improved the usability of the platform, making it easier for users to specify and find locations.

v. Dashboards have been developed for both clients and freelancers, displaying dynamic data such as job posts, number of bookings, and number active projects. More details on job posts and active projects will be added in the future.

7. TASK REMAINING

In the subsequent increments, following major tasks are to be done :

i. Integration of Chat System:

A robust chat system is to be integrated in the platform. We will be using socket.io for real time communication between client and freelancers. This would make the deal-making process between client and freelancers easier and hassle-free.

ii. Integration of Online Payment Gateway:

Online payment gateway is to be integrated that allows seamless transaction between client and freelancers once a job is completed. We are researching on which service to integrate for payment system so that the payment process becomes easy,secure and reliable.

iii. Freelancer Booking System:

Once a deal is made between client and freelancers through the chat system, client needs to book the freelancer. This is done by a booking system that is to be implemented in the platform.

iv. Recommendation System:

Recommendation System is to be implemented that recommends freelancers to clients based on their preferences. A content-based recommendation will be integrated to the platform that allows client to locate the freelancer that matches his/her job requirements.

v. Reviews and Rating:

Reviews and Rating system is another functionality that is to be implemented. Once a job is completed, client will be allowed to rate freelancer based on his/her experience with the freelancer and his service.

8. DELIVERABLES

Our system is envisioned as a responsive, platform-independent web application tailored specifically for service-based freelancing. At its core, the platform serves three primary user roles: clients, freelancers, and administrators.

Clients, as key stakeholders, enjoy a seamless experience within our interface. They are empowered to effortlessly make job announcements and bookings. The recommendation system recommends the freelancers using content based recommendation technique. They can directly message the preferred freelancer through the integrated chat system or post the job details in the platform.

Freelancers can promptly respond to job posts with short message and accept bookings with ease, enhancing productivity and facilitating smoother interactions between clients and service providers. Administrators play a crucial role in maintaining the integrity and security of the platform. Moreover, our platform integrates seamless online payment capabilities, providing clients and freelancers with a secure means of transaction.

Additionally, clients have the opportunity to rate freelancers based on their experiences, fostering transparency and accountability within the platform and ultimately enhancing user satisfaction and trust.

9. PROJECT TASK AND TIME SCHEDULE

The Project Time Schedule and Gantt Chart outlines our project's timeline across three iterations. The first iteration (May 12 - June 1) focuses on requirement analysis, UI design, and core feature implementation. The second iteration (June 2 - June 23) adds additional features, dynamic dashboards,recommendation system and embedded chat system. The final iteration (June 24 - July 14) focuses on system refinement, comprehensive testing, and finalizing documentation.

9.2. TIME SCHEDULE

TASK	APPROX DURATION IN DAYS
Requirement Analysis and Specification	26
System Design and Wire framing	16
Update Design	7
Coding and Implementation	42
Testing and Debugging each module	26
Overall System Test	12
Develop Documents	61

Table 1: Time Schedule

9.2 GANTT CHART

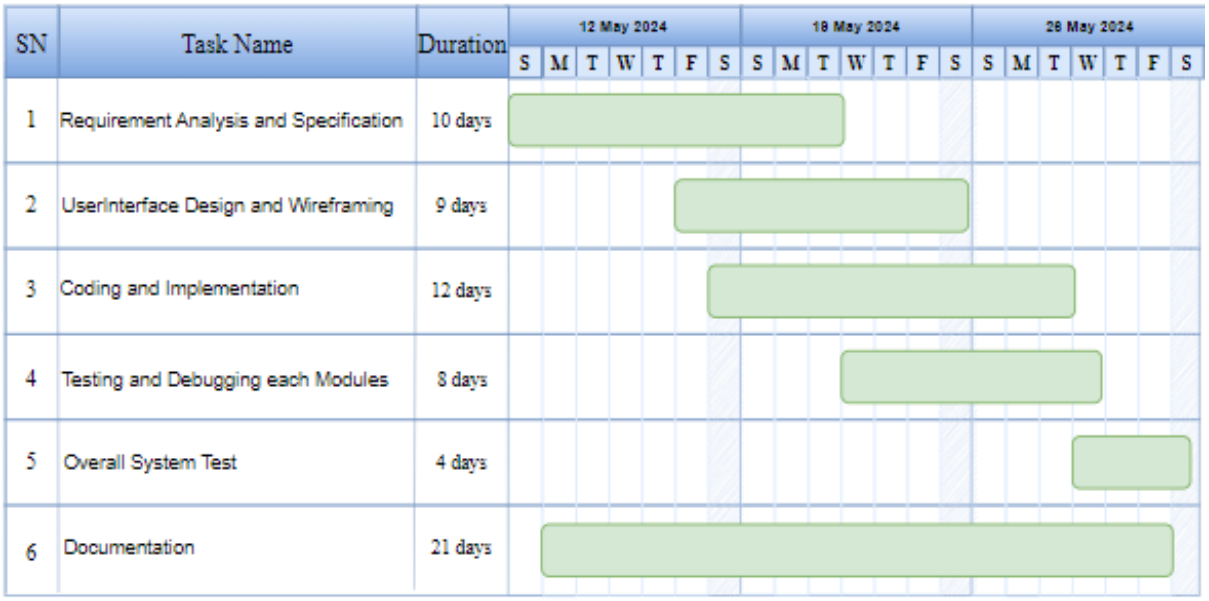

Table 2: Iteration I

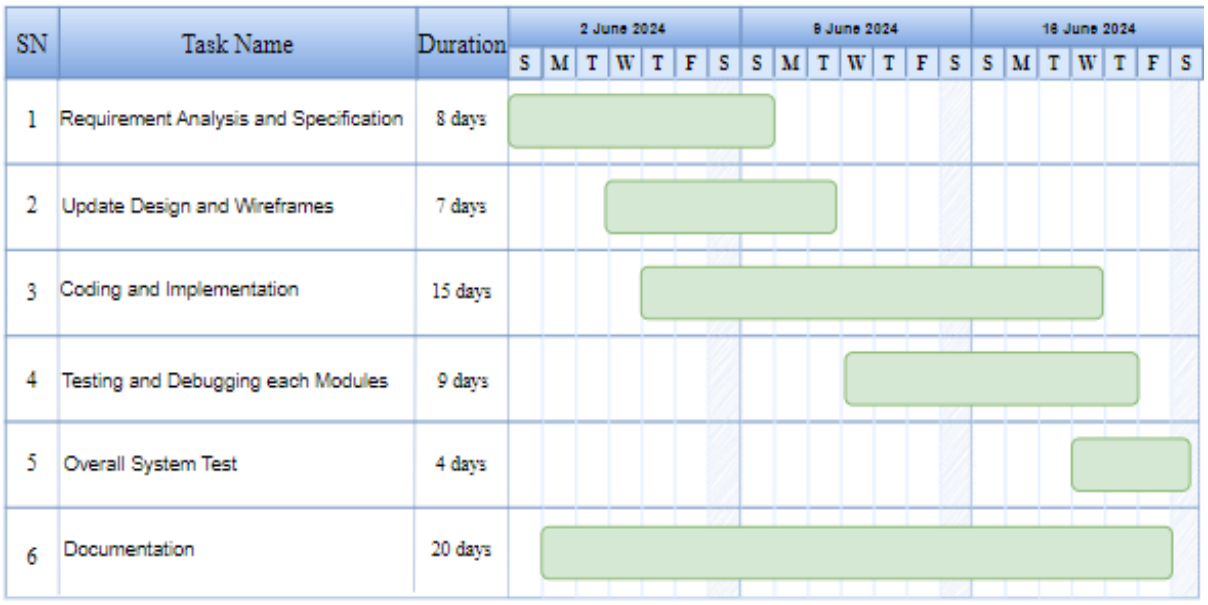

Table 3: Iteration II

SN	Task Name	Duration	23 June 2024							30 June 2024							7 July 2024						
			S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S
1	Requirement Analysis and Specification	8 days																					
2	Project Review and Update Design	7 days																					
3	Coding and Implementation	15 days																					
4	Module Testing and Debugging	9 days																					
5	Overall System Test	4 days																					
6	Final Documentation	20 days																					

Table 4: Iteration III

7. REFERENCES

[01] "Find and Hire Freelancers by Skills - Guru." Guru.com. Accessed: Jun. 12,2024. [Online] Avialable: https://www.guru.com/m/hire/freelancers/all-skills/

[02] "SRIYOG Join | Professional Marketplace." Sriyog.com. Accessed: Jun. 12,2024. [Online] Avialable: https://sriyog.com/join

[03] "Upwork." upwork.com. Accessed: Jun. 12, 2024. [Online] Avialable: https://www.upwork.com/nx/search/jobs/?location=Nepal&sort=recency

[04] R S. Pressman, “The Process” in , Software Engineering: A Practitioner's Approach, McGraw-Hill (Inc.), New York, NY, USA, 2000,

[05] "React Reference Overview." React.dev. Accessed: Jun. 12, 2024. [Online] Avialable: https://react.dev/reference/react

[06] “Node.js — About Node.js®." nodejs.org. Accessed: Jun. 12,2024. [Online] Avialable: https://nodejs.org/en/about

[07] "Express 4.x - API Reference." . Accessed: Jun. 12, 2024. [Online] Avialable: https://expressjs.com/en/4x/api.html