



**American International University-Bangladesh (AIUB)**  
**Department of Computer Science**  
**Faculty of Science & Technology (FST)**

**ShobKaaj – One App for Every Task Near You.**

A Software Engineering Project Submitted  
By

Semester: Summer_25_26		Section:	Group Number:	
SN	Student Name	Student ID	Contribution (CO3+CO4)	Individual Marks
1	Syed Al Mahmud	23-50168-1	40	
2	Tahmid Hasan Binoy	22-46511-1	20	
3	Tahmid Jawad Shafi	22-49127-3	20	
4	Md Jubair Hassan	21-45641-3	20	

The project will be Evaluated for the following Course Outcomes

<b>CO3:</b> Select appropriate software engineering models, project management roles and their associated skills for the complex software engineering project and evaluate the sustainability of developed software, taking into consideration the societal and environmental aspects	Total Marks	
	[5Marks]	
<b>CO4:</b> Develop project management plan to manage software engineering projects following the principles of engineering management and economic decision process	Total Marks	
	[5Marks]	
	[5Marks]	
Develop the project plan, its components of the proposed software products	[5Marks]	
Identify all the activities/tasks related to project management and categorize them within the WBS structure. Perform detailed effort estimation correspond with the WBS and schedule the activities with resources	[5Marks]	
Identify all the potential risks in your project and prioritize them to overcome these risk factors.	[5Marks]	

## Description of Student's Contribution in the Project work

Student Name: Syed Al Mahmud

Student ID: 23-50168-1

Contribution in Percentage (40%):

Contribution in the Project:

- Use Case diagram
- Class diagram
- Project Proposal
- Selection of Process model
- Requirements
- Software user interface
- Risk management
- Gantt Chart

---

Signature of the Student

Student Name: Tahmid Hasan Binoy

Student ID: 22-46511-1

Contribution in Percentage (20%):

Contribution in the Project:

- State diagram
- Project Proposal
- Selection of Process model
- Requirements
- Software user interface
- Risk management

---

Signature of the Student

Student Name: Tahmid Jawad Shafi

Student ID: 22-49127-3

Contribution in Percentage (20%):

Contribution in the Project:

- Activity diagram
- Project Proposal
- Selection of Process model
- Requirements
- Software user interface
- Risk management

---

Signature of the Student

Student Name: Md Jubair Hassan

Student ID: 21-45641-3

Contribution in Percentage (20%):

Contribution in the Project:

- Sequence diagram
- Project Proposal
- Selection of Process model
- Requirements
- Software user interface
- Risk management

\_\_\_\_\_  
Signature of the Student

## 1. PROJECT PROPOSAL

### **1.1 Background to the Problem**

Bangladesh has a large and diverse workforce, consisting of skilled laborers, students, freelancers, homemakers, and unemployed rural youth. However, many of them face challenges in finding steady employment or trustworthy clients. Similarly, homeowners, students, and small businesses often need quick, reliable help for everyday tasks like tutoring, delivery, repairs, event setup, etc., but struggle to find verified workers.

The root cause of this issue lies in the fragmented and informal nature of job opportunities and service access in Bangladesh. There is no unified, trusted platform that connects people who need services with skilled individuals available nearby. Existing apps and job boards are either too niche (e.g., only for tutors or freelancers) or lack verification and safety features. Rural workers especially remain disconnected from digital job markets due to low tech access or platform relevance.

This problem is important because it affects both economic growth and social well-being. Millions remain underemployed or unemployed, while others face delays, scams, or high costs due to the lack of reliable help. Addressing this can improve income distribution, digital inclusion, and overall societal efficiency.

### **1.2 Solution to the Problem**

The objective of the proposed project, “ShobKaaj”, is to develop a secure, user-friendly mobile platform that connects clients and workers in one place for all kinds of local services. It will bridge the gap between job seekers (especially rural and underrepresented groups) and job providers, ensuring verified, efficient, and trustworthy transactions.

## **Proposed Solutions:**

- GPS Matching: Clients and workers can find each other based on real-time locations.
- Verified Profiles: NID or mobile-based verification to ensure safety and trust.
- In-App Communication: Real-time chat to discuss job details.
- Digital Wallet Integration: Support for popular services like bKash, Nagad.
- Multilingual Support: Bangla-first interface with other options.
- Badges & Ratings: Encourage trust and reward quality service.

## **Basic Functionalities:**

- Clients: Can post jobs with title, description, budget, deadline, and location; browse worker profiles; chat and hire; give ratings.
- Workers: Can build profiles with photos, skills, and experience; receive instant job alerts; apply and communicate; get paid and rated.
- Universal: Location-matching, in-app wallet, ratings, badge system, NID/phone verification, rural accessibility features.

## **Target Users and Benefits:**

- Clients: Homeowners, students, event organizers, shopkeepers—get easy access to trustworthy local help.
- Workers: Students, women, freelancers, rural and semi-urban laborers—gain flexible work opportunities and verified payments.

## **Scientific and Societal Contribution:**

“ShobKaaj” contributes to the development of digital marketplaces by combining location-aware algorithms, secure identity checks, and real-time communication in a single mobile app. It has cultural and social impact by enabling flexible work for women, promoting financial inclusion, and reducing digital inequality, especially for rural and semi-urban populations

## **Literature Review:**

Global platforms like TaskRabbit, Upwork, and Fiverr focus on specific types of work such as freelancing or small household tasks. They work well in developed countries but do not support informal or local workers in places like Bangladesh. They also miss important features such as Bangla language, GPS-based worker search, mobile payments, and trust systems.

In Bangladesh, platforms like Bdjobs, Sheba.xyz, and Kormo Jobs are available, but each has limits. Bdjobs is mainly for office jobs, Sheba.xyz is for urban services like cleaning or repairs but lacks full support features, and Kormo Jobs gives entry-level work but does not include things like GPS, chat, or payments.

Because of these gaps, there is still no single platform in Bangladesh that connects both formal and informal workers with features like multi-job categories, real-time GPS matching, in-app chat, payments, and Bangla-first support.

### **How “ShobKaaj” Extends Existing Solutions:**

- Covers more job types (blue-collar, small jobs, tuition, freelance).
- Stronger focus on rural, underrepresented users with a simplified Bangla interface.
- Combines chat + job matching + verification + payment in one app.
- Promote safe, flexible work through trust badges, live ratings, and local language support.

## **2. SOFTWARE DEVELOPMENT LIFE CYCLE**

### **2.1 Process Model**

#### **Nature and Environment of the Software**

**ShobKaaj** is a mobile application designed to connect clients with local workers across Bangladesh for various services. The software environment is:

- User-driven and evolving: especially with feedback from rural and underrepresented users.
- Feature-rich: including GPS-based matching, chat, verification, ratings, and mobile payments.
- Trust-sensitive: requiring secure identity checks and real-time communication.
- Cross-platform: with a focus on inclusivity, multilingual support, and usability on low-end devices.

This dynamic and feedback-heavy environment demands a flexible and iterative development process.

#### **Selected Model: Agile (Scrum)**

The Agile methodology, particularly the Scrum, is best suited due to:

- Flexibility in accommodating evolving requirements.
- Iterative delivery of features in sprints for faster feedback and improvement.
- User-centric development, with regular testing and adaptation based on real user input.
- Efficient collaboration among cross-functional teams.
- Early issue detection and continuous integration of improvements.

## Why Agile Over Other Models

Model	Limitation for ShobKaaj	Why Agile Is Better
Waterfall	Rigid, no scope for mid-development change	Agile allows adaptive planning and change
V-Model	Testing-focused but inflexible	Agile includes testing in every sprint
RAD	Resource-intensive and prototype-focused	Agile balances speed with structure

## Conclusion

Agile (Scrum) is the optimal choice for ShobKaaj due to its adaptability, incremental delivery, and strong support for user-centered, secure, and scalable mobile app development.

## 2.2 Project Role Identification and Responsibilities

The **ShobKaaj** project follows the **Scrum** methodology, where each role plays a vital part in ensuring the successful development and delivery of the platform. The following are the key roles and their responsibilities:

### **1. Scrum Master**

The Scrum Master ensures that the Scrum process is properly followed and that the project proceeds smoothly. They support both the development team and the Product Owner throughout the project.

#### **Responsibilities:**

- Facilitates Scrum events (Sprint Planning, Daily Scrum, Review, Retrospective).
- Supports the Product Owner in managing and prioritizing the Product Backlog.
- Coaches the development team in self-organization and productivity.
- Identifies and removes obstacles that hinder team progress.
- Serves as a communication link between the team, customer, and management.

### **2. Scrum Team**

The Scrum Team is a self-organizing, cross-functional group of professionals responsible for delivering product increments each sprint. Typically consisting of 3 to 9 members, the team works collaboratively without internal sub-divisions.

#### **Responsibilities:**

- Plans and completes sprint tasks and backlog items.

- Estimates effort and manages the Sprint Backlog.
- Delivery of high-quality, working product increments.
- Shares collective accountabilities for all outcomes.
- Continuously improves through team feedback and retrospectives.

### **3. Product Owner**

The Product Owner is responsible for maximizing the value of the product. They are the main decision-maker regarding the features and priorities of the application.

#### **Responsibilities:**

- Manages and prioritizes the Product Backlog.
- Defines and communicates clear product requirements.
- Makes final decisions on backlog changes.
- Represents business needs and user expectations.
- Collaborates with all stakeholders to ensure product alignment.

### **4. Customer**

The Customer is the end user or stakeholder who provides valuable input regarding product needs and usability. Their involvement ensures the final product aligns with real-world expectations.

#### **Responsibilities:**

- Contributes to backlog item discussions and refinements.
- Provides feedback during sprint reviews and testing phases.
- Validates that the delivered features meet their needs.

### **5. Management**

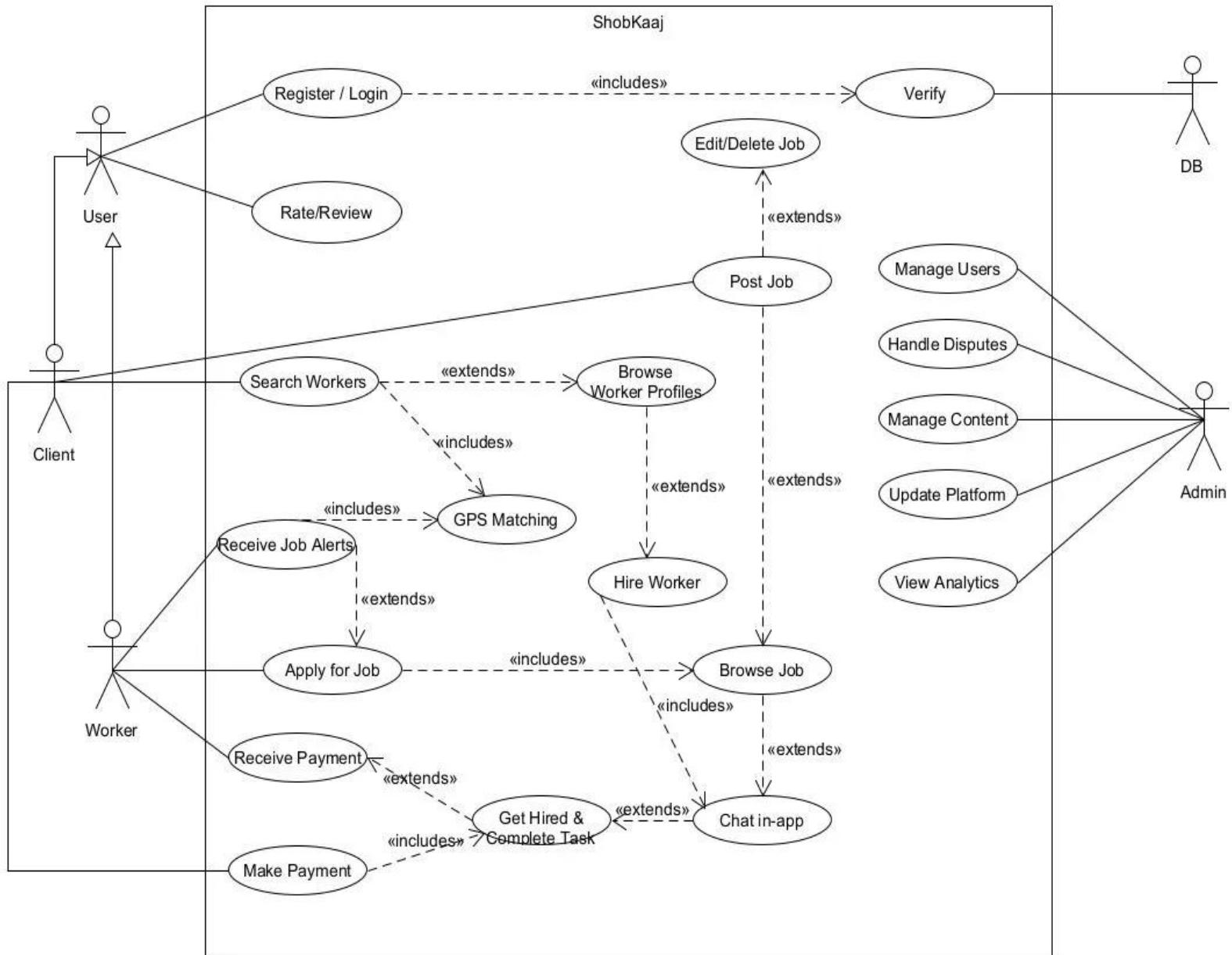
Management oversees the overall direction and resource allocation of the project. While not directly involved in Scrum ceremonies, their role supports the team's structure and success.

#### **Responsibilities:**

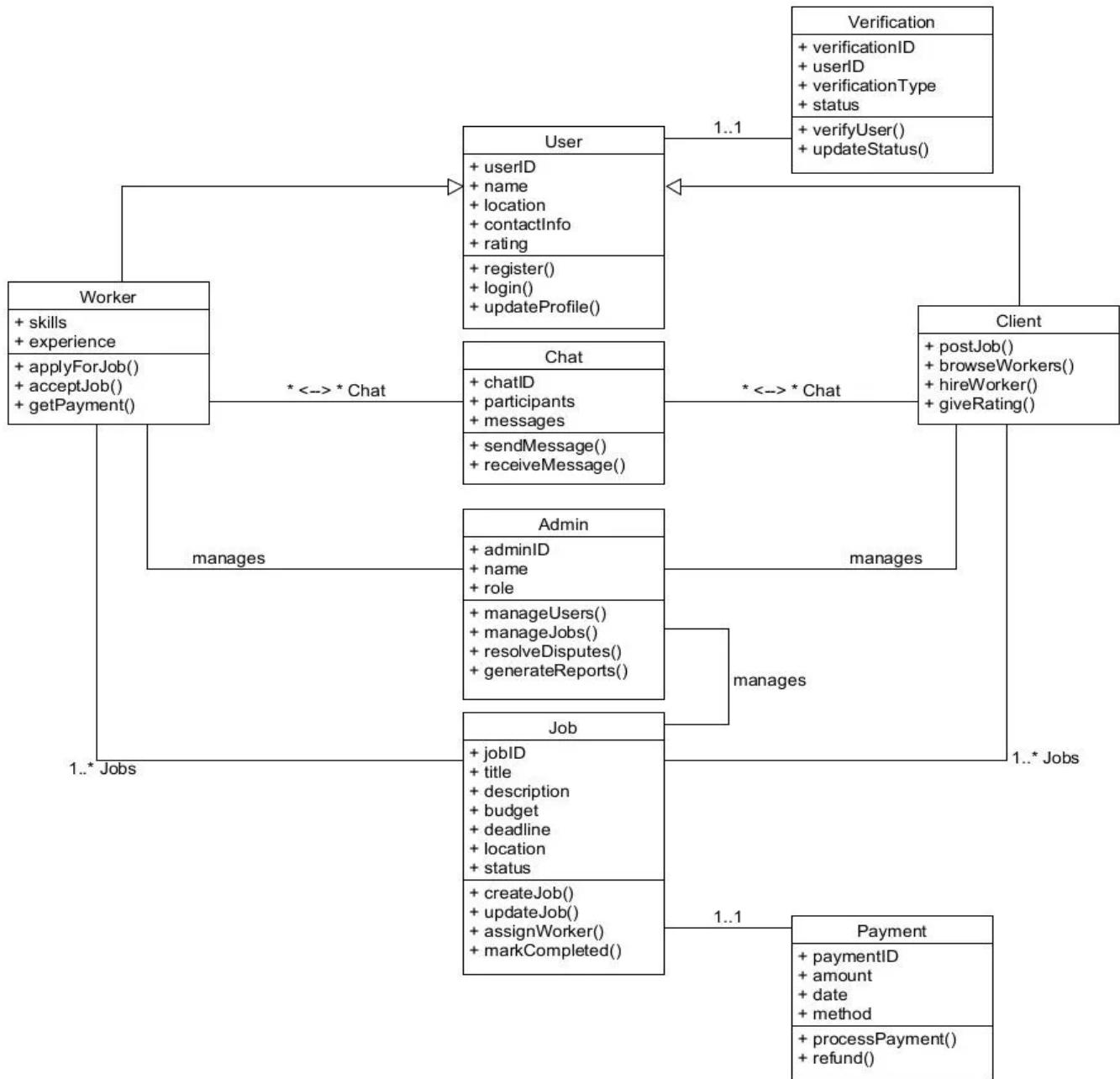
- Sets goals, standards, and expectations for the project.
- Approves key decisions and monitors progress.
- Ensures the team has the tools, time, and support to deliver.
- Promotes an Agile culture and supports Scrum adoption.

# Diagrams

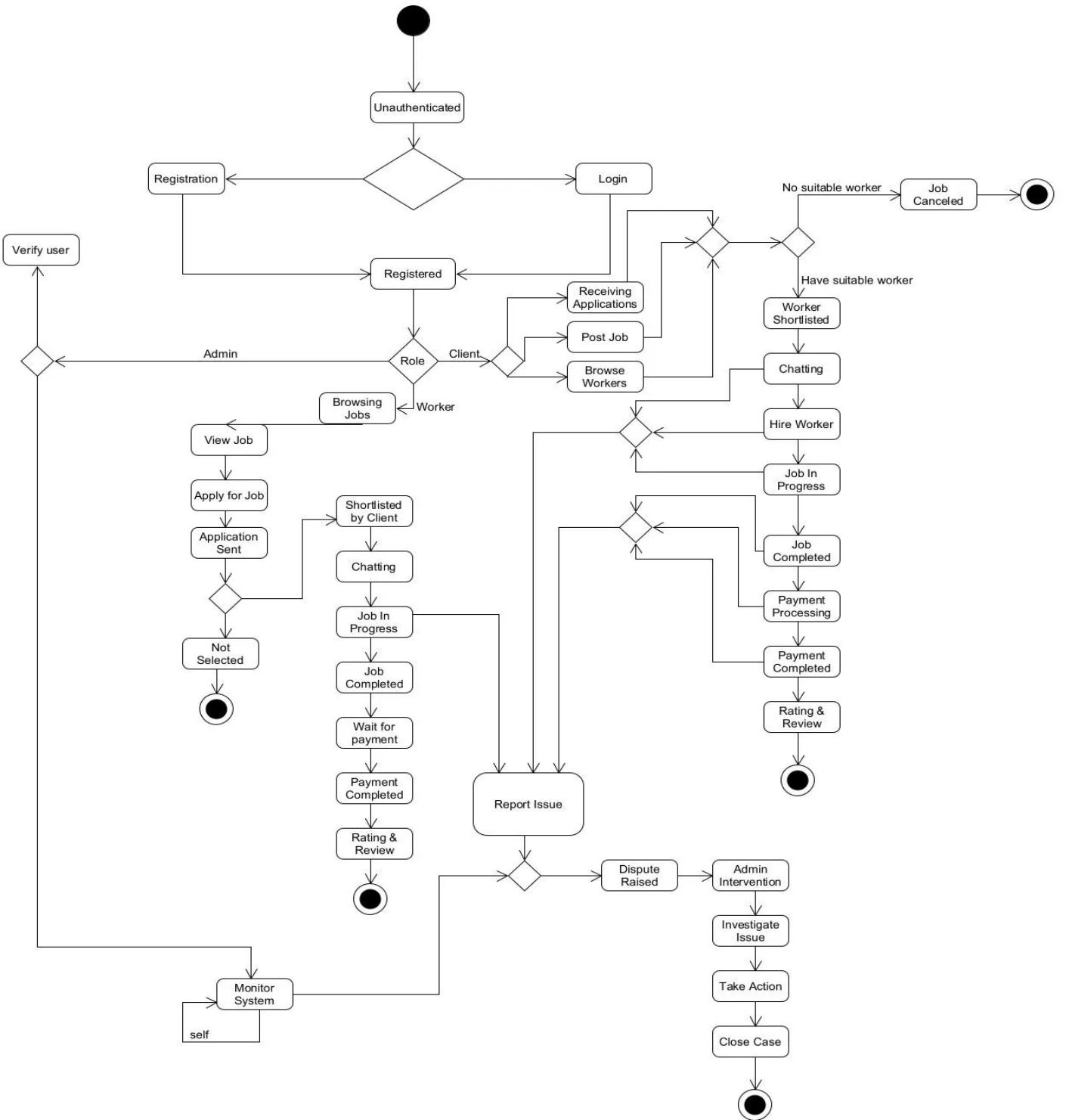
## 1.Usecase diagram



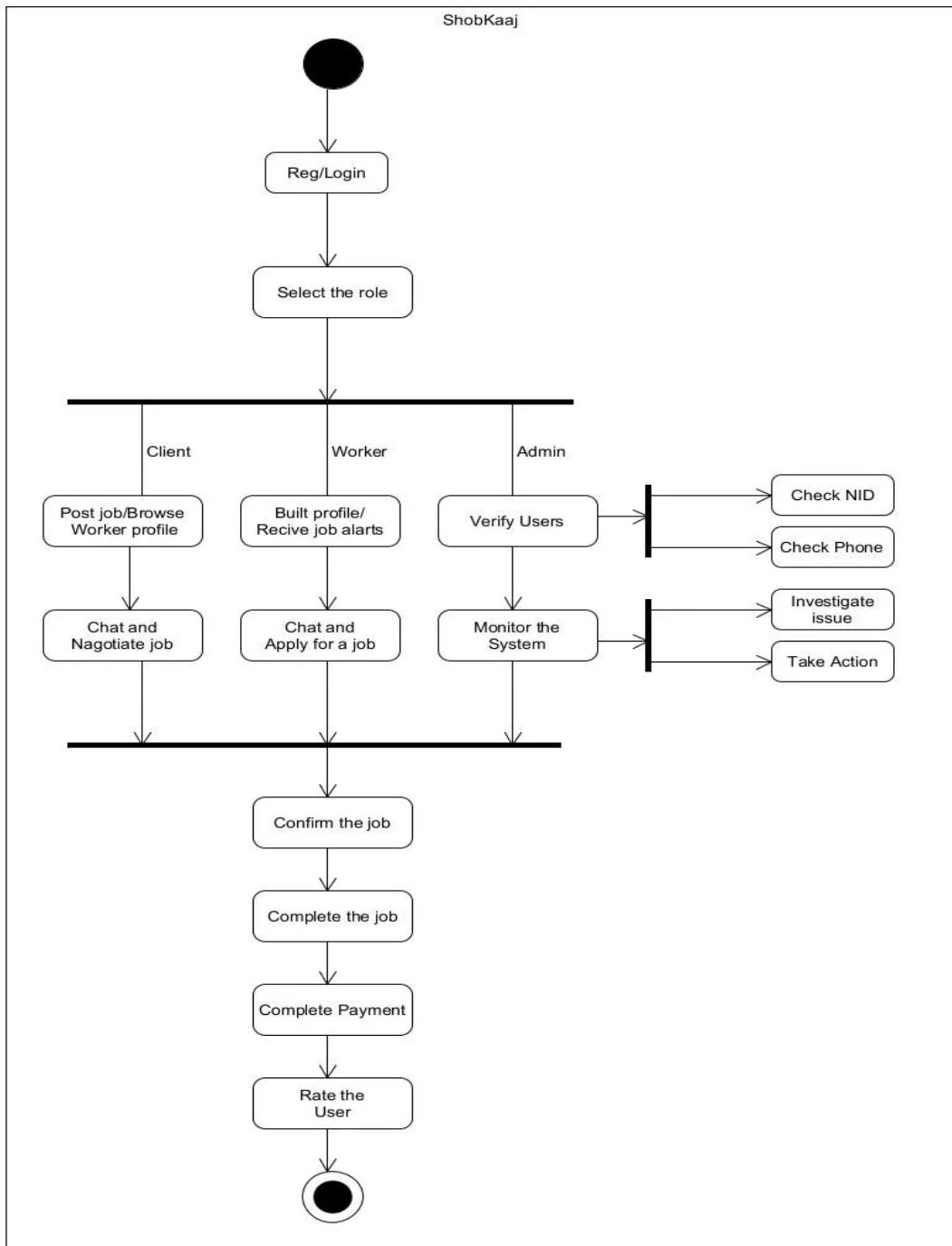
## 2. Class diagram



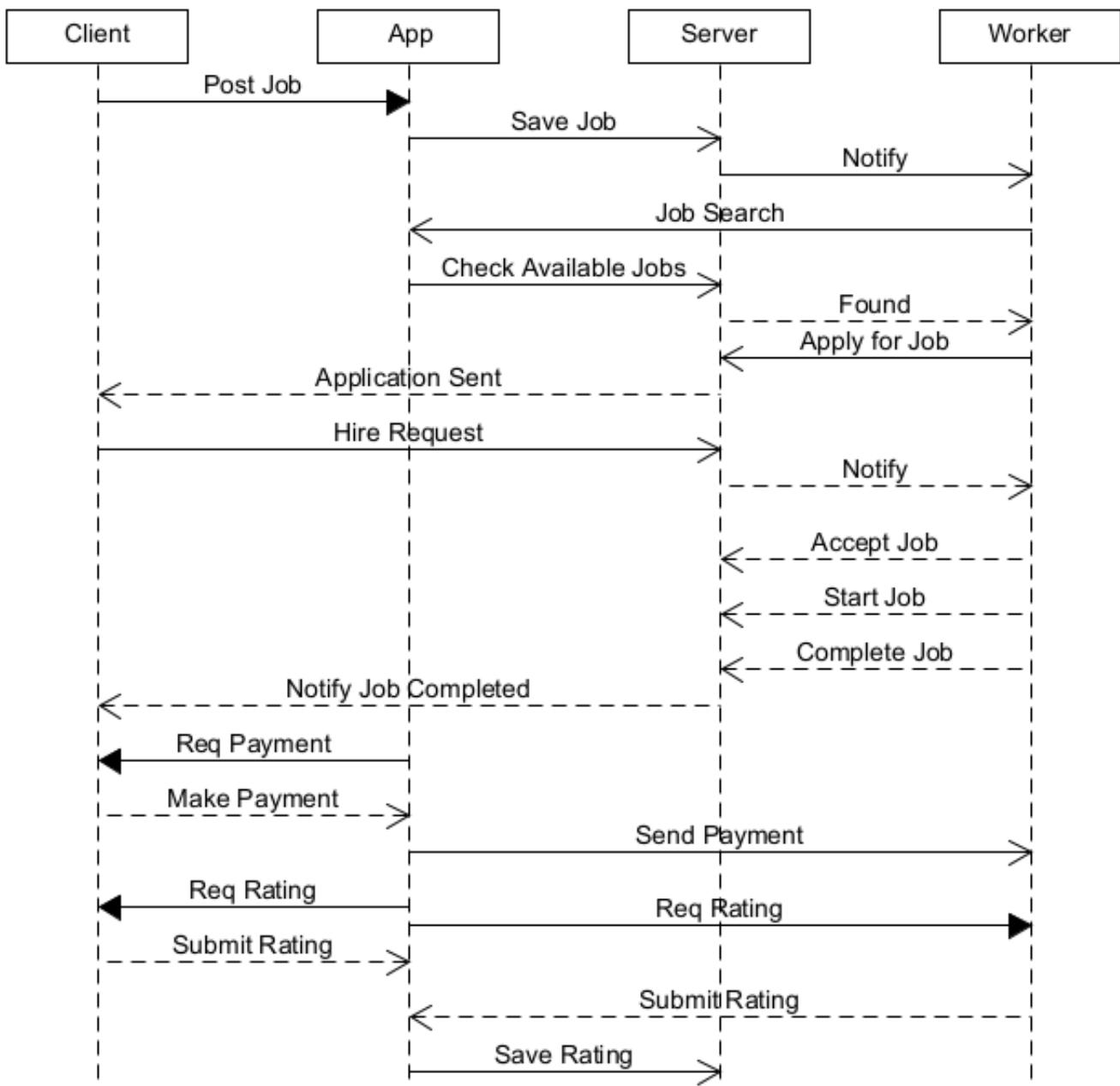
### 3.State diagram



#### 4. Activity diagram



## 5. Sequence diagram



# Requirements

## 1. Functional Requirements

ID	Requirement Description
FR1	Users must be able to register and log in using phone number or NID-based verification.
FR2	Clients can create job posts with a title, description, location, budget, and deadline.
FR3	Workers can browse, filter, and apply to jobs based on location, category, and skills.
FR4	GPS-based matching must connect clients with nearby available workers in real time.
FR5	Real-time in-app chat must be available between clients and workers.
FR6	Workers can create and manage profiles with photo, skills, and experience.
FR7	Clients and Workers must be able to rate and review after job completion.
FR8	The platform must support in-app digital payments through bKash and Nagad.
FR9	The app must support Bangla-first UI with optional rural dialect and English fallback.
FR10	Workers must receive real-time job alerts and earn trust badges based on performance.

## 2. Non-Functional Requirements

ID	Requirement Description
NFR1	The app must be optimized for smooth performance on low-end Android devices.
NFR2	The system must be scalable to support high traffic and a growing user base.
NFR3	All user data must be securely stored with encryption and secure authentication.
NFR4	The UI/UX must be intuitive and accessible to users with minimal tech literacy.
NFR5	The system should ensure 99% uptime and handle offline mode gracefully where needed.

### 3. User Requirements

User Type	Capabilities
Client	Register/Login, Post jobs, Browse worker profiles, Chat, Hire, Pay, Rate & Review
Worker	Register/Login, Create profile, Receive alerts, Apply to jobs, Chat, Get paid, Earn badges
Admin	(Optional) Verify users, moderate content, resolve disputes, manage reports

### 4. System Requirements

Component	Requirement
Platform	Android (initial), extendable to iOS and Web
Backend	RESTful API with secure data handling
Database	Centralized database for users, jobs, messages, ratings, payments
Payment Gateway	Integration with bKash and Nagad APIs
GPS Services	Location tracking and geo-matching logic
Chat System	Real-time communication (Firebase, <a href="#">Socket.IO</a> or equivalent)
Notification System	Push notifications for job status and message

## COCOMO Estimation

We estimated the project size at 10,000 SLOC. Using the COCOMO Basic Model for an Semi-detached project:

$$\text{Effort} = PM = \text{Coefficient} \times (\text{SLOC}/1000)^P$$

$$Development\ time = DM = 2.50(PM)^T$$

$$\text{Required number of people} = ST = PM/DM$$

Where:

- 10K SLOC

- Constants:

Software Project Type	Coefficient<Effort Factor>	P	T
Organic	2.4	1.05	0.38
Semi-detached	3.0	1.12	0.35
Embedded	3.6	1.20	0.32

### Calculation

$$PM = 3.0 \times (10) ^{1.05} \approx 39.55$$

$$DM = 2.5 \times (39.55) ^{0.35} \approx 9 \text{ months}$$

$$ST = 39.55 / 9 = 4.39 \text{ Persons} \approx 5 \text{ Persons}$$

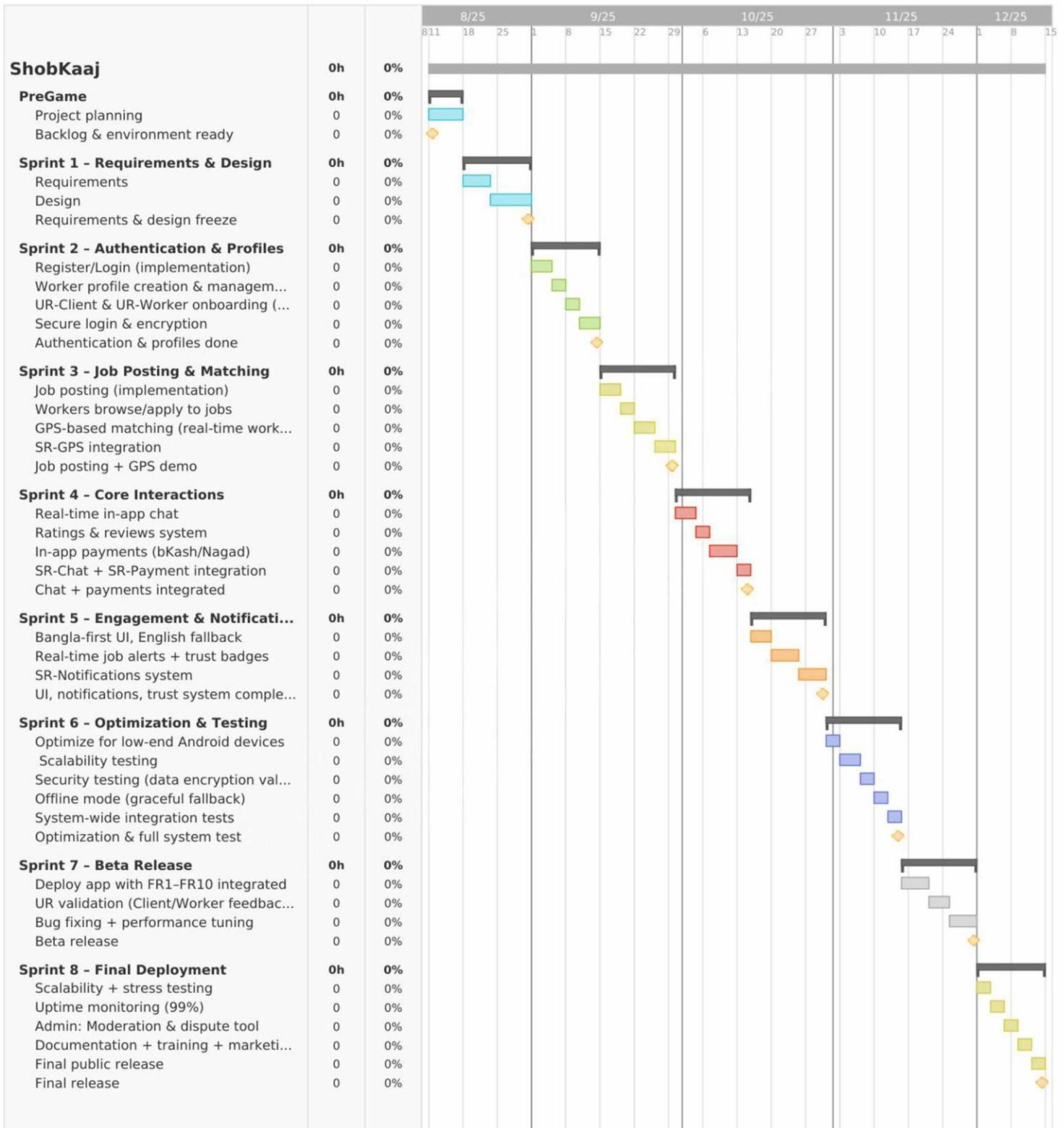
Adjustment to complete in 17 weeks

$$\text{Effort (BAC)} = 39.55 \text{ PM}$$

$$\text{Duration (compressed)} = 17 \text{ weeks} (\approx 4.25 \text{ months})$$

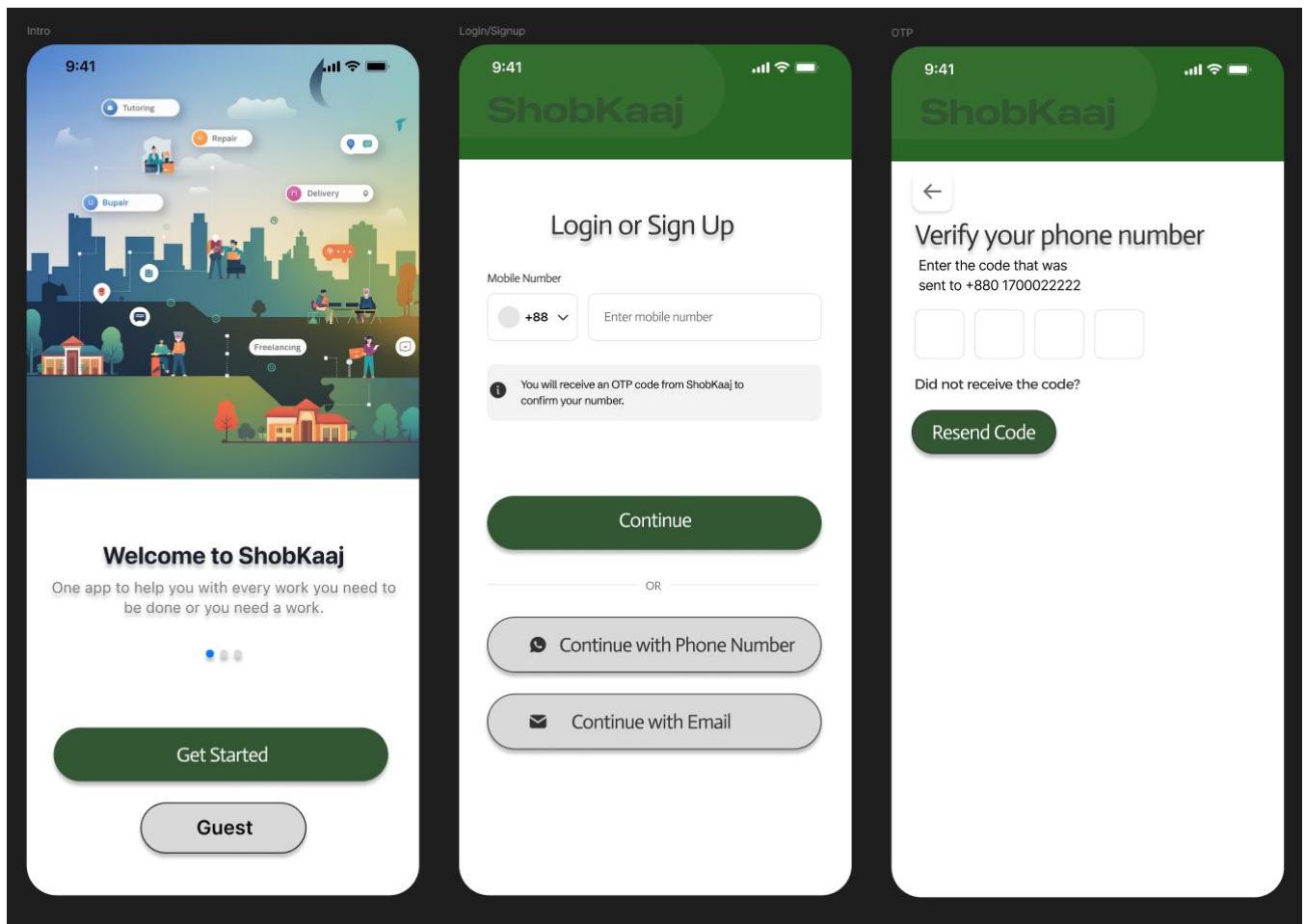
$$\text{Team Size} = 39.55 / 4.25 \approx 9-10 \text{ Person}$$

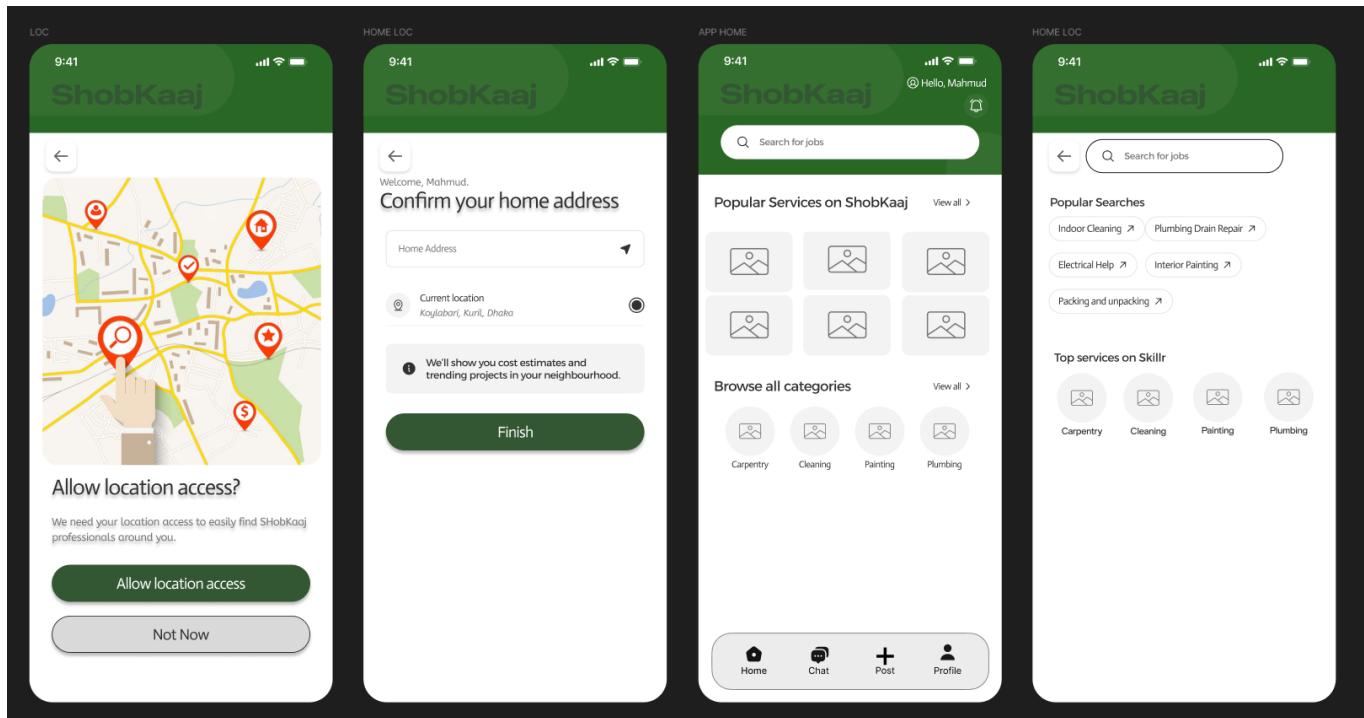
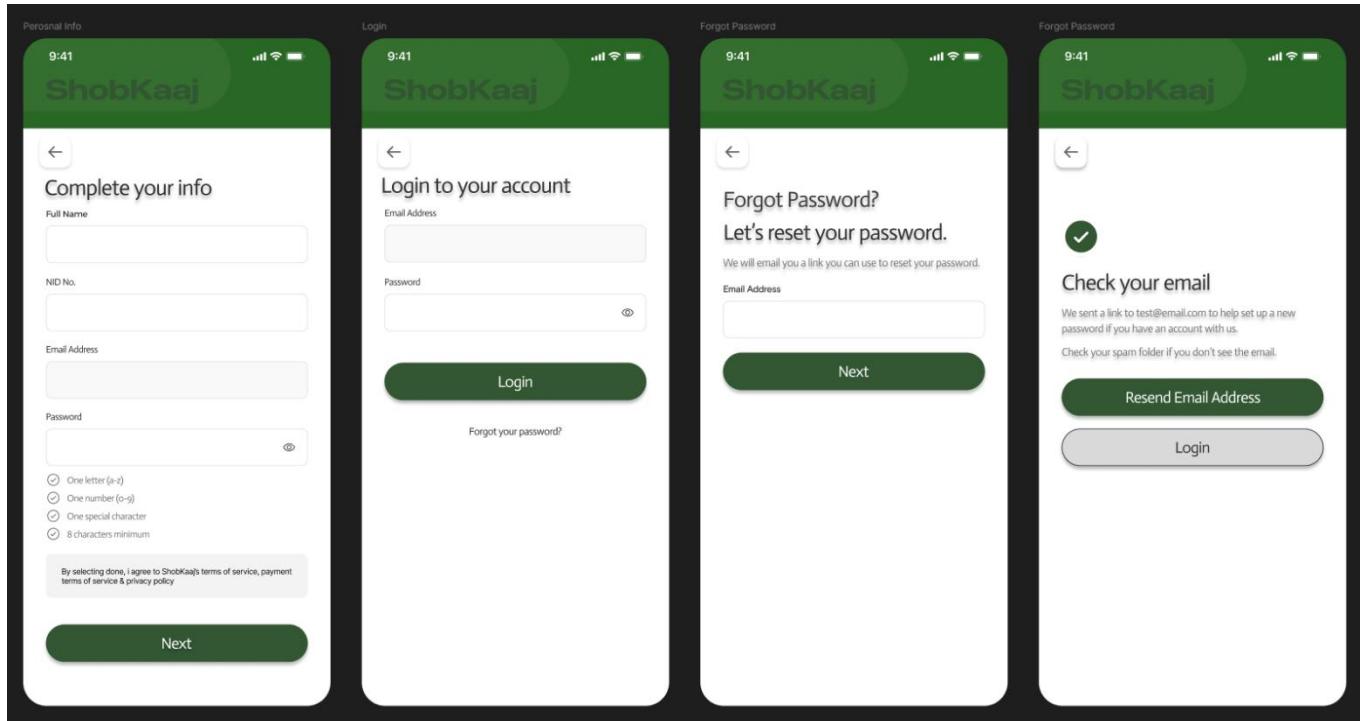
# Gantt Chart



# Software user interface

Figma:<https://www.figma.com/design/jJarzBaAAiTopQT5vsmdqY/Final-Project?node-id=0->





# Risk Table

Risk ID	Risk Name	Risk Probability	Risk Level	Risk Impact	Mitigation Plan	Notes
Risk-1	Low Adoption by Clients & Workers	Medium	High	Without enough users, ShobKaaj won't gain traction	Early marketing campaigns, referral bonuses, village/union partnerships, campus drives	Focus on rural & semi-urban outreach
Risk-2	App Bugs & Crashes on Low-End Devices	High	High	Workers may abandon app due to instability	Continuous QA testing on budget smartphones, beta releases, bug tracking	Bangladesh has high % of low-end phone users
Risk-3	Scalability Issues During Rapid Growth	Medium	High	App may slow down as user base increases	Cloud-based backend, load balancing, modular microservices	Plan phased rollout in major cities first
Risk-4	Payment Gateway Downtime (bKash/Nagad)	Medium	High	Failed transactions cause distrust	Integrate multiple gateways + fallback cash-on-delivery option	Maintain API monitoring & alerts
Risk-5	User Data Leak / NID Verification Breach	Low	Very High	Legal risks + loss of trust	Strong encryption, two-factor authentication, regular audits	Store minimal personal data
Risk-6	Inaccurate GPS Matching in Rural Areas	Medium	Medium	Wrong job-worker pairing leads to frustration	Allow manual location input if GPS fails	Test in low-connectivity areas
Risk-7	Regulatory Compliance with Govt. Digital Laws	Low	High	Risk of legal restrictions or shutdown	Consult legal experts, ensure NID verification aligns with govt. rules	Monitor ICT ministry updates
Risk-8	Limited Rural Accessibility (Network/Language)	Medium	Medium	Rural workers struggle with app usability	Bangla-first UI, offline-friendly features, local dialect support	Pilot in rural districts
Risk-9	Fake Profiles & Scams by Workers/Clients	Medium	High	Reduces trust in platform	Strict NID verification, admin moderation, report & block feature	Introduce trust badges & rating
Risk-10	Competition from Sheba.xyz / Kormo Jobs	Medium	Medium	Users may shift to alternatives	Differentiate via GPS-matching + rural inclusion + Bangla-first	Focus on underserved informal job market

Risk ID	Risk Name	Risk Probability	Risk Level	Risk Impact	Mitigation Plan	Notes
Risk-11	Delayed App Development (Sprint Slippage)	Medium	High	Late delivery increases cost & risks	Strict sprint planning, buffer time, focus on MVP first	Track tasks with Scrum board
Risk-12	Server Downtime Affecting Job Matching	Medium	High	Clients & workers lose confidence	Cloud hosting, backup servers, auto-scaling, 24/7 monitoring	Maintain 99% uptime target

All files of our project **ShobKaaj** are uploaded on GitHub

**GitHub Link** = <https://github.com/mahito-0/Software-Engineering-Project.git>

## Rubric for Project Assessment (CO3)

Criteria	Marks distribution (Max 3X5= 15)				Acquired Marks
	Inadequate (1-2)	Satisfactory (3)	Good (4)	Excellent (5)	
<b>Selection of Software Engineering Models</b>	Does not articulate a position or argument of choosing appropriate model. Does not present any evidence to support the arguments for the choice of the model	Articulates a position or argument for choosing models that is unfocused or ambiguous. Presents incomplete/vague evidence to support argument for model choice	Articulates a position or argument of choosing models that is limited in scope. Does not present enough evidence to support the argument for the choice of the model	Clearly articulates a position or argument for the choosing software engineering models. Presents sufficient amount of evidence to support argument for the model selection	
<b>Role identification and Responsibility Allocation</b>	The project has poor project management plans for identifying roles and assigning the responsibilities	Identify few roles in the project management where some of the roles are left alone with any project responsibilities	Identify most of the roles in the project management and assign their responsibilities	Well planned project with proper role identification and responsibility allocation in the project management activities	
<b>Impact identification</b>					
<b>Formatting and Submission</b>	Project report is not complete and Several errors in spelling and grammar. Present a Confusing	Some errors in spelling and grammar. Some problems	Few errors in spelling and grammar. Presents most of the details in	Project report is complete and No errors in spelling and grammar. Consistently	

	organization of concepts, supporting arguments, and real-life example. Sentences rambling, and details are repeated.	of organizing the answer in a logical order of defining, elaborating, and providing real-life examples.	a logical flow of organization in definition, details, and example.	presents a logical and effective organization of definition, details, and real-life example of the topic.	
<b>Acquired marks:</b>					
<b>CO Pass / Fail:</b>					

## Rubric for Project Assessment (CO4)

Marking Criteria	Marks Distribution (Maximum 3X5=15)				Acquired Marks
	Inadequate (1-2)	Satisfactory (3)	Good (4)	Excellent (5)	
Project Planning	No background information regarding the project is given; project goals and benefits are missing.	Insufficient background information is given; project goals and benefits are poorly stated	Sufficient background information is given; the purpose and goals of the project are explained.	Thorough and relevant background information is given; project goals are clear and easy to identify.	
Effort Estimation and Scheduling	Student vaguely discuss the impact of societal, health, safety, legal and cultural issues in their project	Student provided with partial relevance to the impact of societal, health, safety, legal and cultural issues in their project	Student fairly provided the analysis to the impact of societal, health, safety, legal and cultural issues in their project	Student comprehensively provided the analysis to the impact of societal, health, safety, legal and cultural issues in their project	
Risk Management	Ambiguous representative example.	Partially identify / indicate towards real-life example.	Real-life example is fairly connected towards the definition.	Comprehensively defend with real life example.	
<b>Acquired Marks:</b>					
<b>CO Pass / Fail:</b>					