

CS World

A PROJECT REPORT

Submitted By

Manish Kumawat	(170770107535)
Kandarp Vyas	(170770107622)
Dhruv Gajjar	(170770107521)
Muskaan Chaudhary	(170770107508)

In fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
In
Computer Engineering



SILVER OAK COLLEGE OF ENGINEERING AND TECHNOLOGY

**OPP. BHAGWAT VIDYAPITH, NEAR GOTAB CROSS ROAD,
AHMEDABAD- 382481.**

Candidate's Declaration

We hereby declare that project report titled “CS World” submitted towards the completion of project in 8th semester of bachelor of Computer Engineering in Silver Oak College Of Engineering & Technology, Ahmedabad is an authenticate record of our work carried out.

We further declare that to the best of our knowledge the report of C.E. 8th semester.

First Candidates Name : **Manish Kumawat**

Enrollment Number : **170770107535**

Candidate's signature :

Second Candidates Name : **Kandarp Vyas**

Enrollment Number : **170770107622**

Candidate's signature :

Third Candidates Name : **Dhruv Gajjar**

Enrollment Number : **170770107521**

Candidate's signature :

Fourth Candidates Name : **Muskaan Chaudhry**

Enrollment Number : **170770107508**

Candidate's signature :

Submitted to:

Silver Oak College of Engineering & Technology,

Ahmedabad Affiliated to:

Gujarat Technological University.

ABSTRACT

As a student everyone is different, everyone has their own dreams, their view and ideas but everyone struggles at some point. Here our effort is to minimize that struggle and help them to achieve their greatness. Our main objective is to create application to connect all students and create community for them to help them find their interest and give them proper guidance for better career advancement and over all life...

ACKNOWLEDGEMENT

We are heartily thankful to our supervisor, Mr. Vivek Shah, whose encouragement, supervision and support from the preliminary to the concluding level enabled us to develop an understanding of the subject. At the end, we offer our regards and blessings to all of those who supported us in any respect during the completion of the project and to our college for providing a resources and materials.

List of Figure

ID	Title	Page No.
2.1	Incremental Model	8
2.2	Pie Chart	9
4.1	Use-Case Diagram	22
4.2	Account Use-Case Diagram	23
4.3	Activity Diagram User	24
4.4	Activity Diagram Admin	25
4.5	Sequence Diagram	27
4.6	E-R Diagram	29
5.1	Empathy Mapping	31
5.2	Product Development	32
5.3	AEIOU	37
5.4	Ideation	40
5.5	Empathy Summery	42

Index

Chapter	Title	Page No.
1	Introduction	
	1.1 Project Summary	2
	1.2 Project Scope	2
	1.3 Objective	2
	1.4 Technologies and Literature Review	2
2	Software Project Management	
	2.1 Project Planning and Scheduling	6
	2.2 Project Development Approach	6
	2.3 Project Plan	9
	2.3.1 Milestone and Deliverables	10
	2.3.2 Roles and Responsibilities	11
3	System Requirement Study	
	3.1 User Characteristic	13
	3.2 Hardware and Software Characteristics	14
	3.3 Constraints	14
4	System Analysis	
	4.1 Study of Current System	17
	4.2 Requirement of this System	17
	4.3 Feasibility Study	19
	4.4 Requirement Validation	20
	4.5 Function of the System	21
	4.5.1 Use-Case Diagram	21
	4.5.2 Activity Diagram	24
	4.5.3 Sequence Diagram	26
	4.6 Data Modeling	28
	4.6.1 E-R Diagram	28

Chapter	Title	Page No.
5	Canvases	
5.1	Empathy Mapping	31
5.2	Product Development	32
5.3	AEIOU	37
5.4	Ideation	40
5.5	Empathy Summery	42
6	Implementation Planning and Details	
6.1	Implementation Environment	46
6.2	Program Modules and Specification	46
6.3	Security features	47
6.4	Coding Standards	47
7	Testing	
7.1	Test plan	50
7.2	Testing Strategy	50
7.3	Test methods	52
7.4	Test case	54
8	Screenshots and User Manual	
8.1	Login Screen	56
8.2	Live Chat Screen	57

CHAPTER: 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 Project Summary

- In this project we are going to develop android based application which involves Live Chatting and Guidance (and many more features to come). We will develop also the web-app for this Application.
- There are mainly only two modules, Admin and user .
- There is much functionality like notifications, security, ask the questions, give your suggestions, etc.
- We will make students aware about future technologies and give them suggestions.
- There are also Live Chatting option and share your ideas with anyone.

1.2 Purpose

- Our concept is to create a community for students to help them to solve their queries and get suggestions and answers to their endless questions.
- Everyone is learner so here experts and beginners all will be connected at same place and they will help each other by sharing knowledge and resources.

1.3 Scope

- Scope of our project is to provide backbone support to students to find their interest and help them by providing answers.(Aim is to make must have app for CS students.)
- Each update will bring new features in app which will be stable.

1.4 Technology and Literature Review

- The front-end used in our project.
- Tools used are.

JavaScript

- JavaScript supports the development of both client and server components of web-based applications.
- On the client side, it can be used to write programs that are executed by a web browser within the context of the web page.
- On the server side, it can be used to write web server programs that can process information submitted by a web browser and then update the web browser display accordingly.

Advantages:

- It can be used for server side and client-side scripting.
- It is more flexible than JavaScript.

React - Native

- React Native combines the best parts of native development with React, a best-in-class JavaScript library for building user interfaces.
- **Use a little—or a lot.** You can use React Native today in your existing Android and iOS projects or you can create a whole new app.
- React Native lets you create truly native apps and doesn't compromise your users' experiences.

React primitives render to native platform UI, meaning your app uses the same native platform APIs other apps do

Android Studio

- Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ.
- On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as,
 - i. A flexible Gradle-based build system
 - ii. A fast and feature-rich emulator
 - iii. A unified environment where you can develop for all Android devices
 - iv. Extensive testing tools and frameworks
 - v. Lint tools to catch performance, usability, version compatibility, and other problems
 - vi. C++ and NDK support

SQL:

- **SQL (Structured Query Language)** is a special-purpose programming language designed for managing data held in a relational database management system (RDBMS).
- Originally based upon relational algebra and tuple relational calculus, SQL consists of a data definition language and a data manipulation language.
- The scope of SQL includes data insert, query, update and delete, schema creation and modification, and data **access control**. Although SQL is often described as, and to a great extent is, a declarative language (4GL), it also includes procedural elements.
- **Data Definition:** Defining tables and structure in the database.
- **Data manipulation:** Used to manipulate the data within those schema objects.

CHAPTER: 2

SOFTWARE PROJECT MANAGEMENT

CHAPTER 2

SOFTWARE PROJECT MANAGEMENT

Project planning and scheduling

Project Planning

- Project planning is part of project management, which relates to the use of schedules such as Gantt charts to plan and subsequently report progress within the project environment.
- Initially, the project scope is defined and the appropriate methods for completing the project are determined. Following this step, the durations for the various tasks necessary to complete the work are listed and grouped into a work breakdown structure.
- Project planning is often used to organize different areas of a project, including project plans, workloads and the management of teams and individuals.

Project Scheduling

- Project Scheduling is the culmination of a planning activity that is primary component of any project management.
- When combined with estimation methods and risk analysis, scheduling, establishes a road map for the project management.
- Scheduling begins with the process composition. The characteristics of the project are used to adapt an appropriate task set for the work to be done.
- The task network is used to compute the critical project path, a time line chart and a variety of project information

Project Development Approach

The activities we followed for this project is listed below :

- Planning the work or objectives
- Analysis & Design of objectives

- Designing and Planning of features
- Allocation of resources
- Organizing the work
- Database Designing

The Process Paradigm we used for our project is Incremental Model.

The Incremental Process Model

- The Incremental Model combines elements of the linear sequential model with the iterative philosophy of prototyping. The incremental model applies linear sequences in a staged fashion as calendar time progresses.
- Each linear sequence produces a deliverable “increment” of the objectives.
- It should be noted that the process flow for any increment can incorporate the prototyping paradigm.
- When an incremental model is used, the first increment is often a core product. That is, basic requirements are addressed, but many supplementary features remain undelivered.
- The core product is used by the customer. As a result of use and/or evaluation, a plan is developed for the next increment. The plan addresses the modification of the core product to better meet the needs of the customer and the delivery of additional features and functionality.
- This process is repeated following the delivery of each increment, until the complete product is produced.
- The Incremental process model, like prototyping and other evolution approaches, is iterative in nature.
- But unlike prototyping, the incremental model focuses on the delivery of an operational product with each increment.
- Early increments are stripped down versions of the final product, but they do provide capability that serves the user and also provide a platform for evaluation by the user.

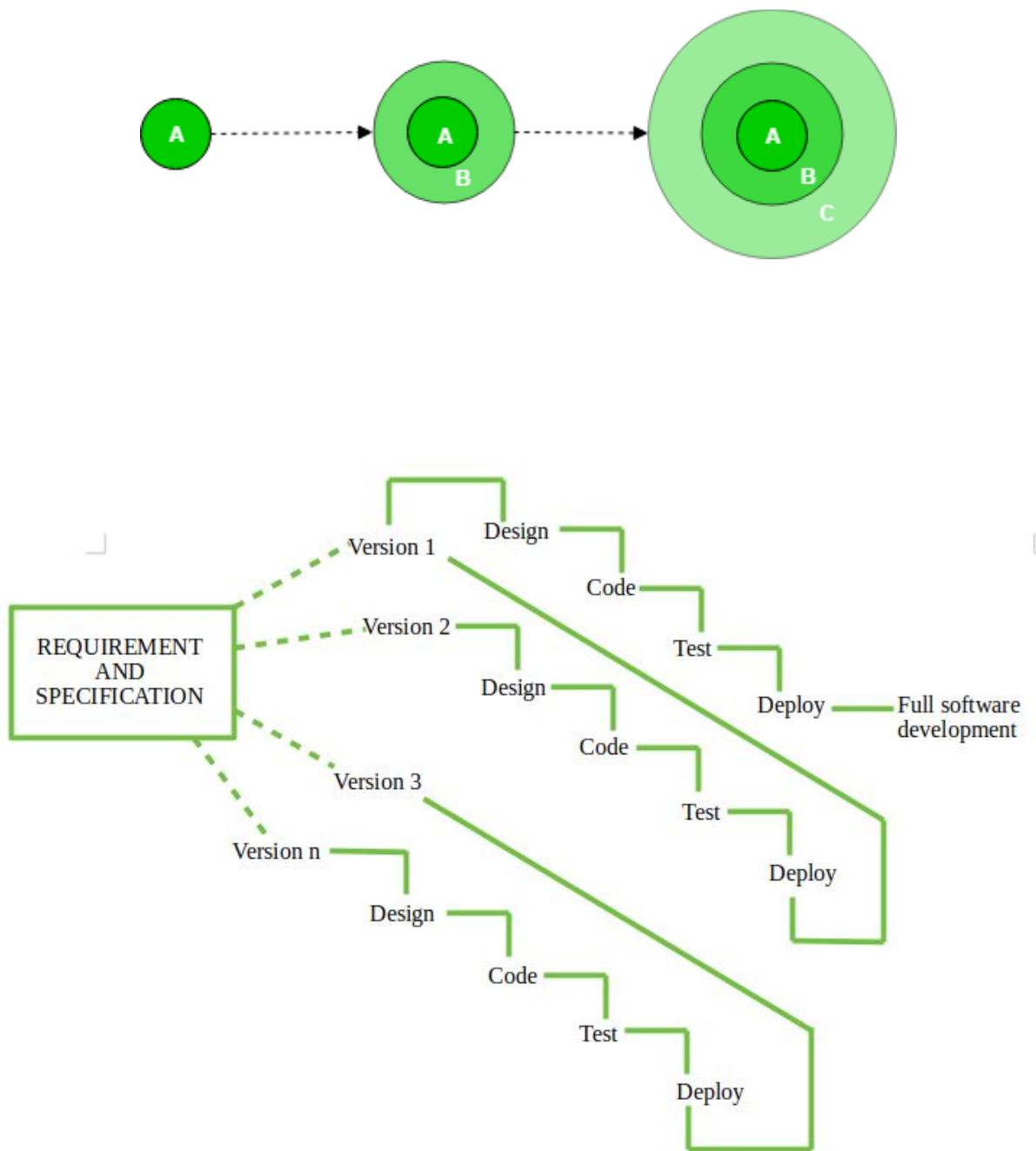


Figure 2.1 Incremental Model

Project Plan

	From Date	To Date
1. Preliminary Research and Analysis	2/6/2020	4/9/2020
2. Prototype Designing	5/9/2020	15/12/2020
3. Implementation	22/12/2020	28/2/2021
4. Review and Testing	5/3/2021	~

[Table 2. 1 Project Plan]

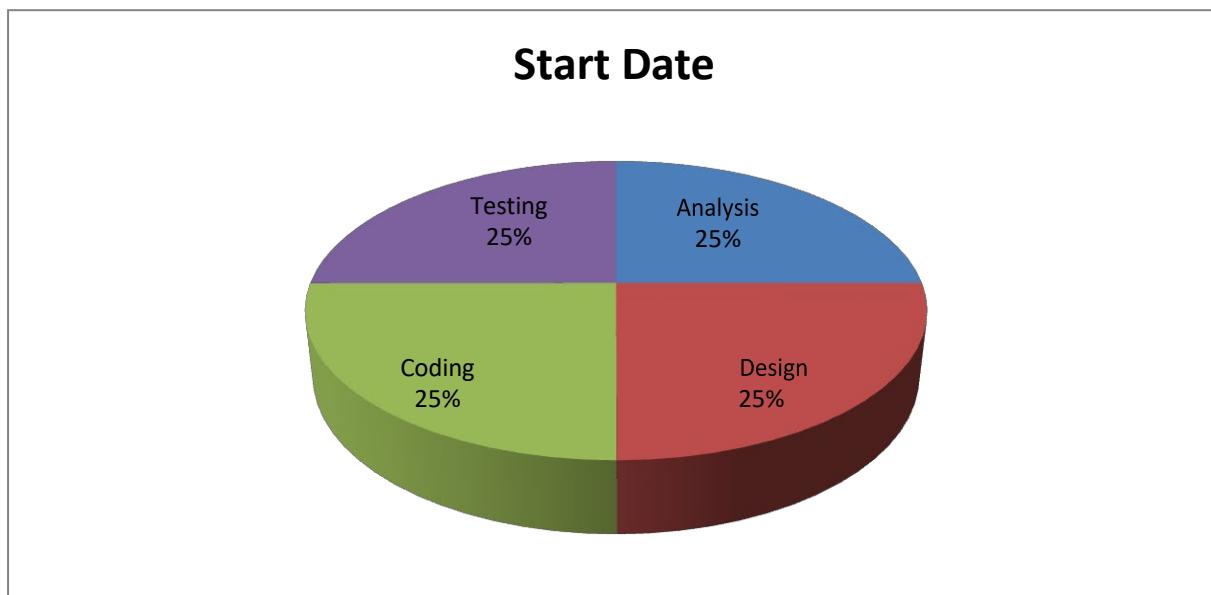


Figure 2.2 Pie Chart

Milestone and Deliverables

- For the start, we researched about the tool and technologies that are going to be the support of our project.
- After analysis and enough research, product planning and design planning were made in action.
- After completing it we started with the design of all forms of this module. The prototype design was selected and approved by all the members.
- After designing comes the feature implementation part, where we worked on the main features first, keeping the space for future add-ons.
- After that we completed User Module. Between this we did our research on the similar existing models.
- There was continuous research on the target users. And what more functions and features we could add.

Software Process Activity	Milestones
Project Plan	Project Schedule
Requirement Collection	User requirements
Data Flow Analysis	System Flow
Design <ol style="list-style-type: none"> 1. Database Design 2. User Interface Design 3. System Design 	System Design Document
Implementation <ol style="list-style-type: none"> 1. Code For giving security 2. Code for reports 	Access Reports
Testing	Setting validation and error message

Table 2. 2 Milestones and Deliverables

Roles and Responsibilities

Manish – Lead Developer

- Project Management
- Presentation
- Report Writing
- Requirement Analysis

Kandarp – Lead Backend Coder

- Content Creator
- Work Review
- Work progress analysis
- Report and Canvases Content

Dhruv – Tester and Developer

- Canvases
- Work Review
- Report Content
- Prototype

Muskaan – UI/UX Designer

- Content Creator
- Presenter
- Helper

CHAPTER: 3

SYSTEM

REQUIREMENT

STUDY

CHAPTER 3

SYSTEM REQUIREMENT STUDY

3.1 User Characteristics

- It describes the type of user which deals with the applications. Basically, this application has four types of users as given below:
 1. Administrator
 2. Moderator
 3. Mentor
 4. Users

1. Administrator

- Responsibility of manager is to manage the servers and to ensure that everything works according to its work.
- Coordinating activities and operations to secure efficiency and compliance.
- Manage guideline rules and check for guidelines violation.
- Policy management

2. Moderator

- The main function of a moderator is to promote interaction among the members.
- To ensure that the discussion stays within the thematic parameters.
- A moderator has authority to determine the content and blog, in case of violation moderator can delete that content.
- Moderator should be posting new threads and adding new content. They should be helping out members with their queries and they should be keeping threads alive by asking questions.

3. Mentor

- Helps you see the destination
- One of a mentor's primary roles is to encourage.
- A good mentor will clarify the big picture by offering suggestions as to what efforts will be necessary to make intended goals a reality.

4. Users

- Ask for help whenever required to solve doubts and questions.
- Help others to clear their doubts and questions.
- Share knowledge and get knowledge.

3.2 Hardware and Software Characteristics

- **Hardware requirement**

- Computer with 2.5Ghz processor
- 8gb of RAM
- Internet connection
- Server or Compatible cloud service
- Android or IOS mobile device

- **Software requirement**

- Android Studio / XCode
- Visual Studio Code
- MySQL or any other DATABASE
- React Native CLI / Expo CLI
- Designing tools (Adobe XD, Adobe illustrator)

3.3 Constraints

3.3.1 Regulatory Policies

- These policies are generally thought to be best applied when good behaviour can be easily defined and bad behaviour can be easily regulated and avoided.
- It limits individuals behaviour against community guidelines and rules.

3.3.2 Hardware limitation

- The smooth functionality of the portal mainly depends on the speed of hardware and then on speed of the internet.
- Hardware limitation will occur if user still use very outdated mobile device or outdated system.

CHAPTER: 4

SYSTEM

ANALYSIS

CHAPTER 4

SYSTEM ANALYSIS

4.1 Study of Current System

- React Native is used for the front-end development for this system which makes the user interaction with the application easier.

4.2 Requirement of this System

- AI auto detection, word press, react native, MySQL are the basic requirements for the given system.
- User response and feedback plays an important part for this system as the main concept is to provide an interactive platform.

Functional Requirement

User Requirement

- User needs to login to the system to access the platform which will undergo the process for authentication and validation.
- User should have basic knowledge about how the app works to gain maximum advantage.

Identification of Functional Requirement

- The high-level functional requirement often needs to be identified from an informal problem description document or from a conceptual understanding of the problem.
- Each high-level requirement characterizes away of system usage by some users to perform some meaningful piece of work.

Documentation of functional requirement

- For documenting the functional requirement, we need to specify the set of functionalities supported by the system.
- A function can be identified the state at which the data to input to the system, its input data domain, the output data domain, and the type of possessing to be carried out on the input data to obtain the output data.

Non-Functional Requirement

Usability

- The interface should use terms and concepts, which are drawn from the experience of people who will make most out of the system.

Efficiency

- The platform must be easy to access and fast in speed.

Readability

- User should never be surprised by the behavior of the system and it should also provide meaningful feedback when error occurs so that user can recover from the error.

Accuracy

- The user should require that data are obtained from database and stored in database must be accurate.

Security

- The user provided information must be secured and the privacy should always be maintained.

Maintainability

- User wants that the system should be maintained easily that is, if there are some changes required in the system that can be done easily.

4.3 Feasibility Study

- Feasibility is the measure of how beneficial the development of information system will be to an organization.
- The feasibility analysis is categorized under four different types.
 1. Operational Feasibility
 2. Technical Feasibility
 3. Scheduled Feasibility
 4. Economic Feasibility

1. Operational Feasibility

- The System is to be developed for any user who wants to use it. We want our system user friendly and easy to use.
- The administrator also may be non-technical, so the user interface will be designed in such a way that it gets comfortable for non-technical person to operate easily.

2. Technical Feasibility

- It is a partially measurement of specific technical solution and the availability of technical resorts and expertise.
- The analyst must find out whether the current technical resources, which are available in the system is capable of handling the job.
- If not, then the analyst with the help of developer should confirm whether the technology is available and capable or not.

Better Considering

- Here we have to consider those tools which are required for developing the project.
- AI auto detection, word press, react native, MySQL are the basic requirements for the given system.

3. Schedule Feasibility

- Schedule feasibility corresponds to whether sufficient time is available to complete the project.

Factor Considering

- Schedule of the project
- Time by which project has to be completed
- Reporting period

4. Economic Feasibility

- Economic feasibility is a measure of cost effectiveness of a project or solution.
- For declaring that the system is economically feasible, the benefits from the project should exceed or at least to the equal to the cost of development.

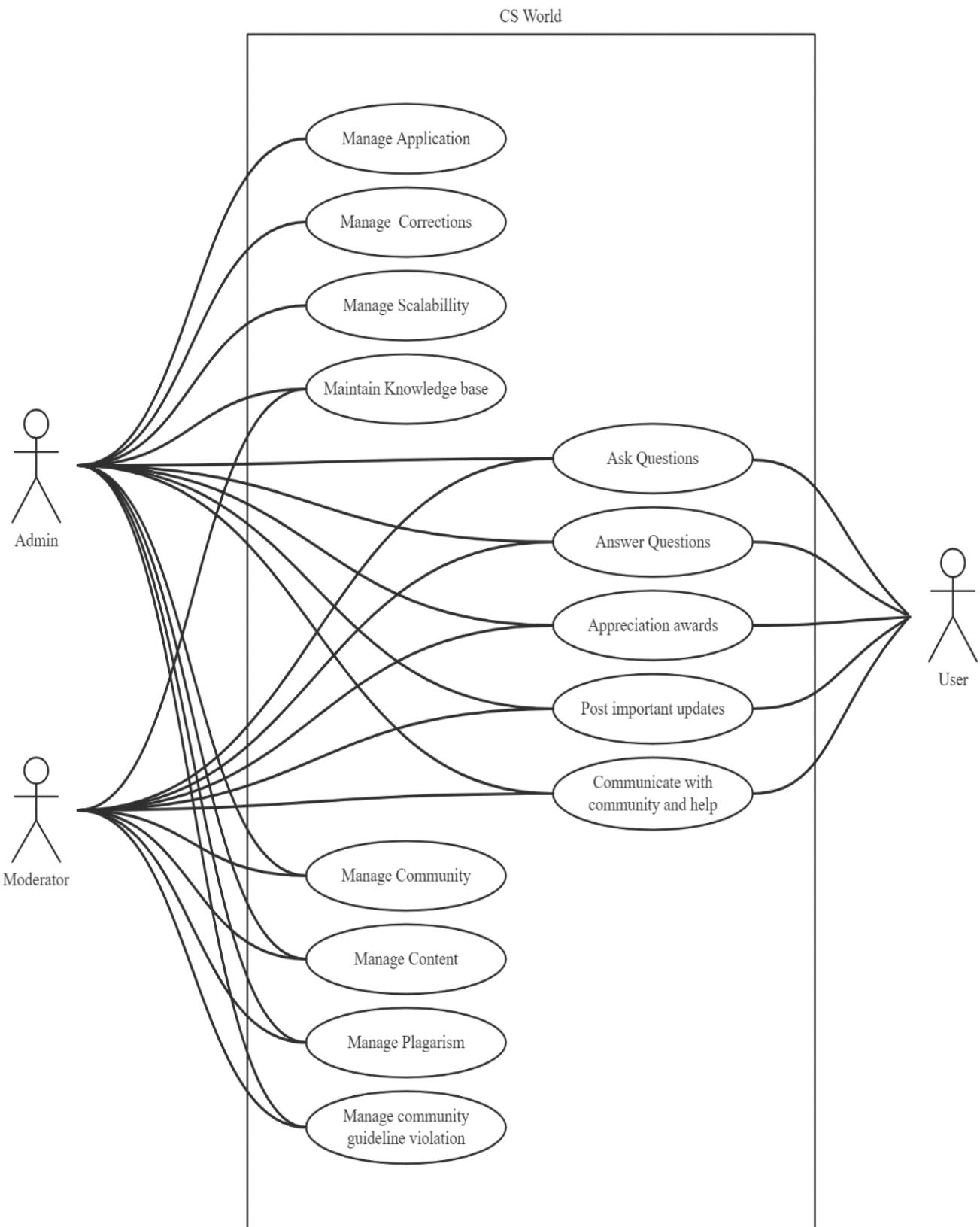
4.4 Requirement Validation

- Requirement validation examines this specification to ensure that all the system requirements have been stated unambiguously.
- These inconsistent, error have been detected and corrected and the work products confirmed to the standard.
- Source of the requirement are identified; final Statement of requirement has been examined by original source.
- Requirements related to main requirements are founded.
- Requirements are clarifying stated and are not misinterpreted.
- All sources of requirements are covered to get a maximum requirement.
- All method of finding requirements is applied.

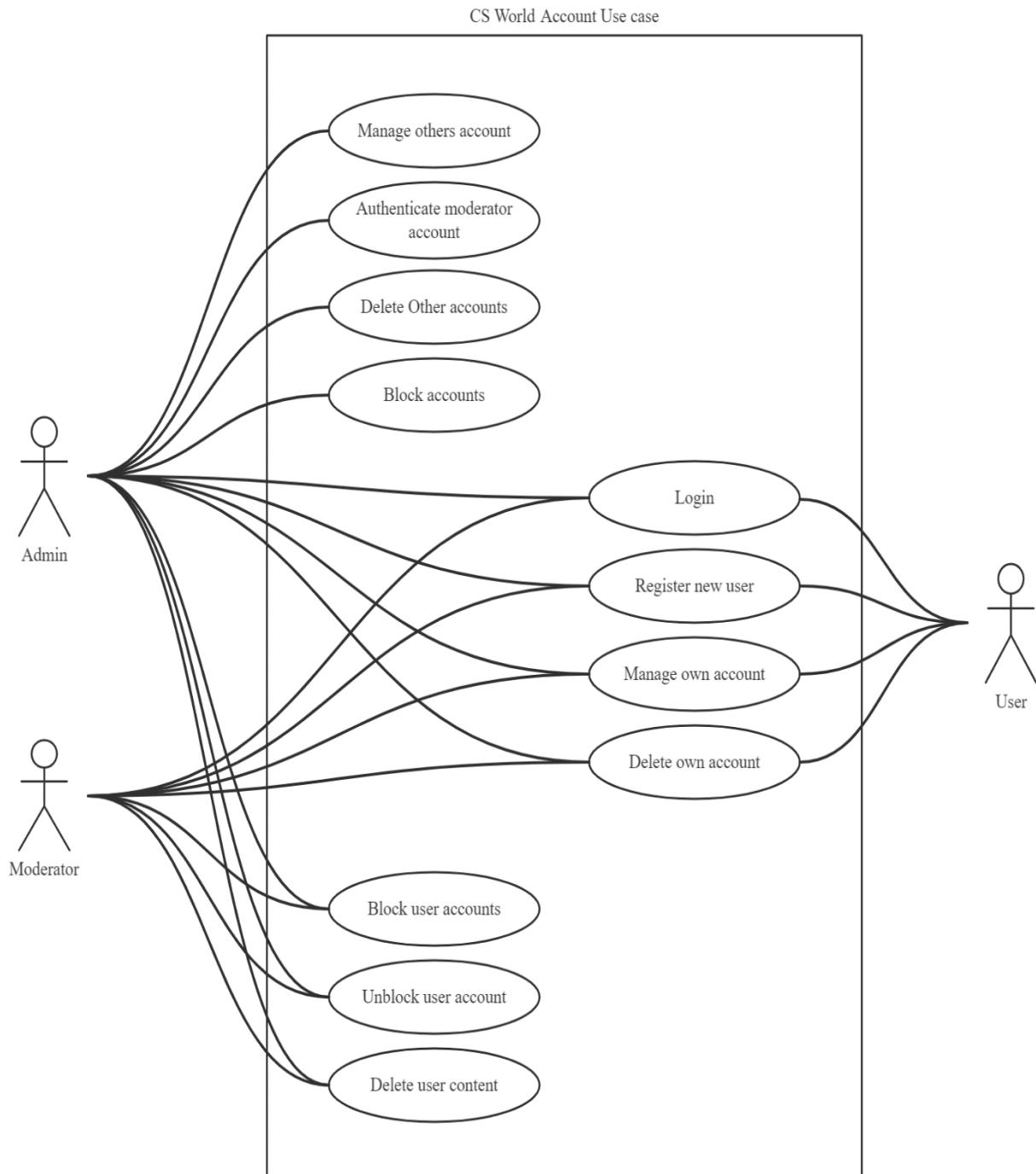
4.5 Function of the System

4.5.1 Use-Case

- In software and systems engineering, a use case is a list of steps, typically defining interactions between actor and a system, to achieve a goal.
- The actor can be a human, an external system, or time.
- In systems engineering, use cases are used at a higher level than within software engineering, often representing missions or stakeholder goals.
- The detailed requirements may then be captured in Systems Modeling Language or as contractual statements.
- As an important requirement technique, use cases have been widely used in modern software engineering over the last two decades.
- Use case driven development is a key characteristic of process models and frameworks.
- With its iterative and evolutionary nature, use case is also a good fit for agile development.

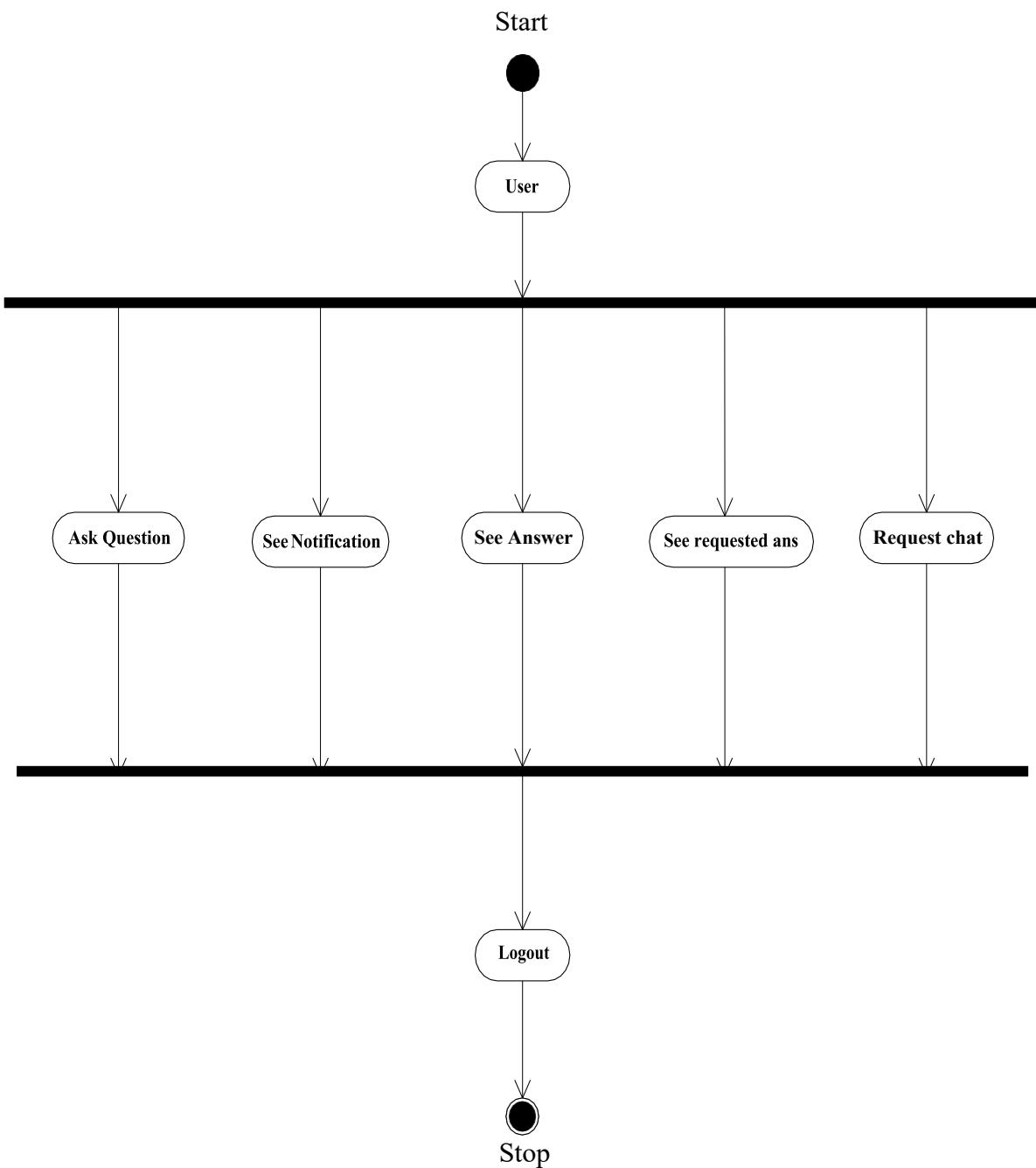


[Figure 4.1 Use-Case Diagram]



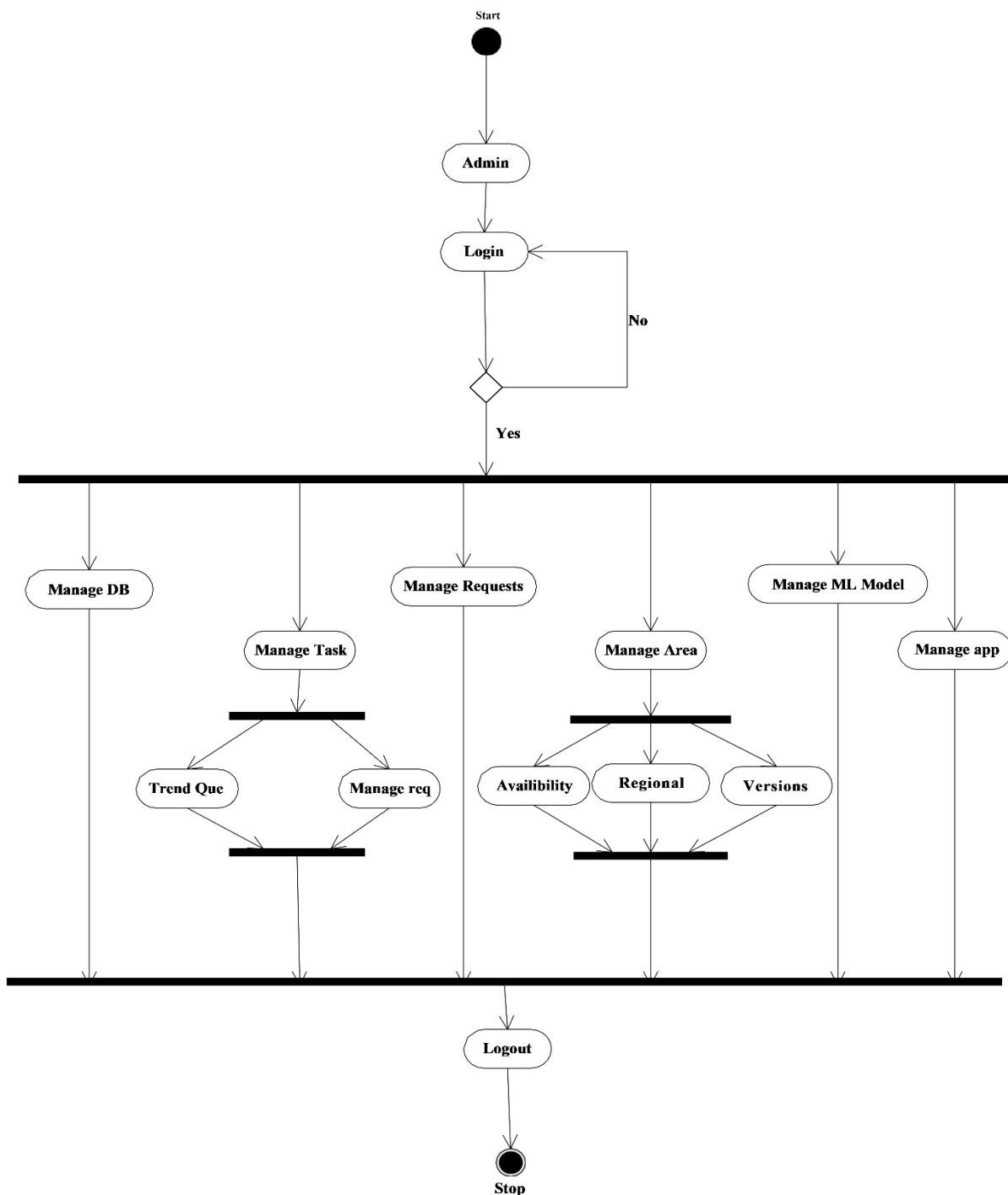
[Figure 4.2 Account Use-Case Diagram]

4.5.2 Activity diagram



[Figure 4.3 Activity Diagram User]

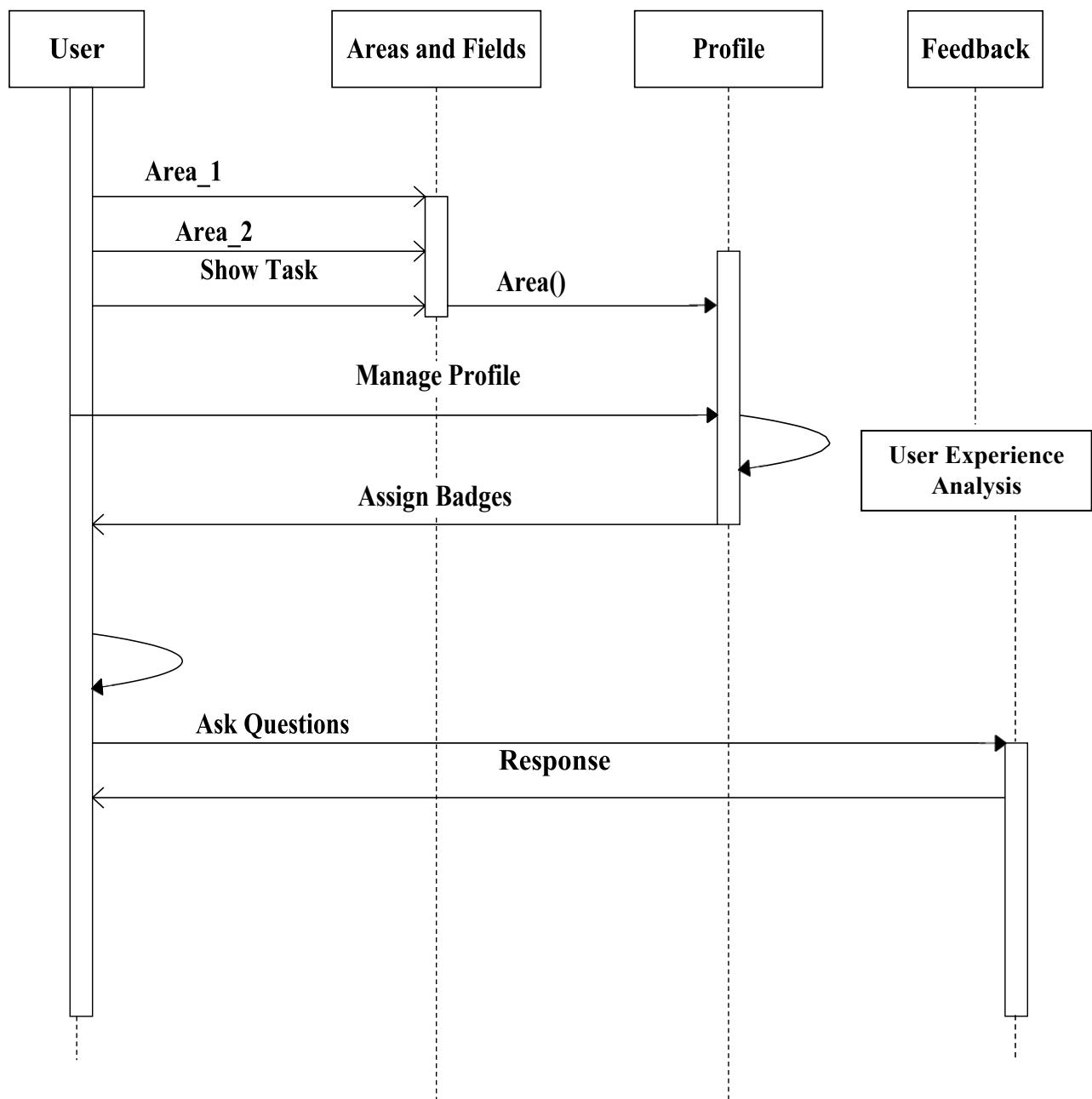
Admin:



[Figure 4.4 Activity Diagram Admin]

4.5.3 Sequence diagram

- The well-known Message Sequence Chart technique has been incorporated into the Unified Modeling Language (UML) diagram under the name of Sequence Diagram.
- A sequence diagram shows, as parallel vertical lines, different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur.
- This allows the specification of simple runtime scenarios in a graphical manner.
- The well-known Message Sequence Chart technique has been incorporated into the Unified Modeling Language (UML) diagram under the name of Sequence Diagram. A sequence diagram shows, as parallel vertical lines, different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

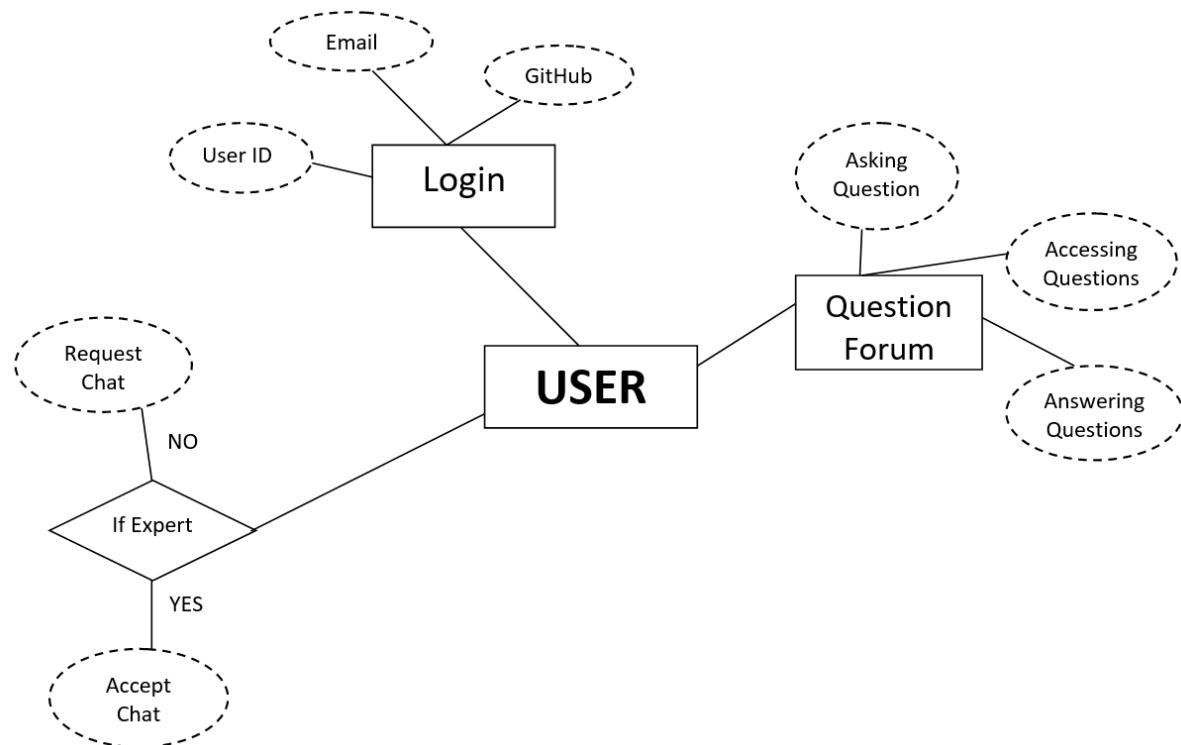


[Figure 4.5 Sequence Diagram]

4.6 Data Modeling

4.6.1 ER-Diagram

- In software engineering, an entity–relationship model (ER model) is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database.
- The main components of ER models are entities and the relationships that can exist among them, and databases.
- An entity-relationship model is a systematic way of describing and defining a business process.
- The process is modelled as components (entities) that are linked with each other by relationships that express the dependencies and requirements between them, such as: one building may be divided into zero or more apartments, but one apartment can only be located in one building. Entities may have various properties (attributes) that characterize them.
- Diagrams created to represent these entities, attributes, and relationships graphically are called entity–relationship diagrams.



[Figure 4.6 E.R. Diagram]

CHAPTER: 5

CANVASES

CHAPTER 5

CANVASES

5.1 Empathy Mapping

Design For : CS World

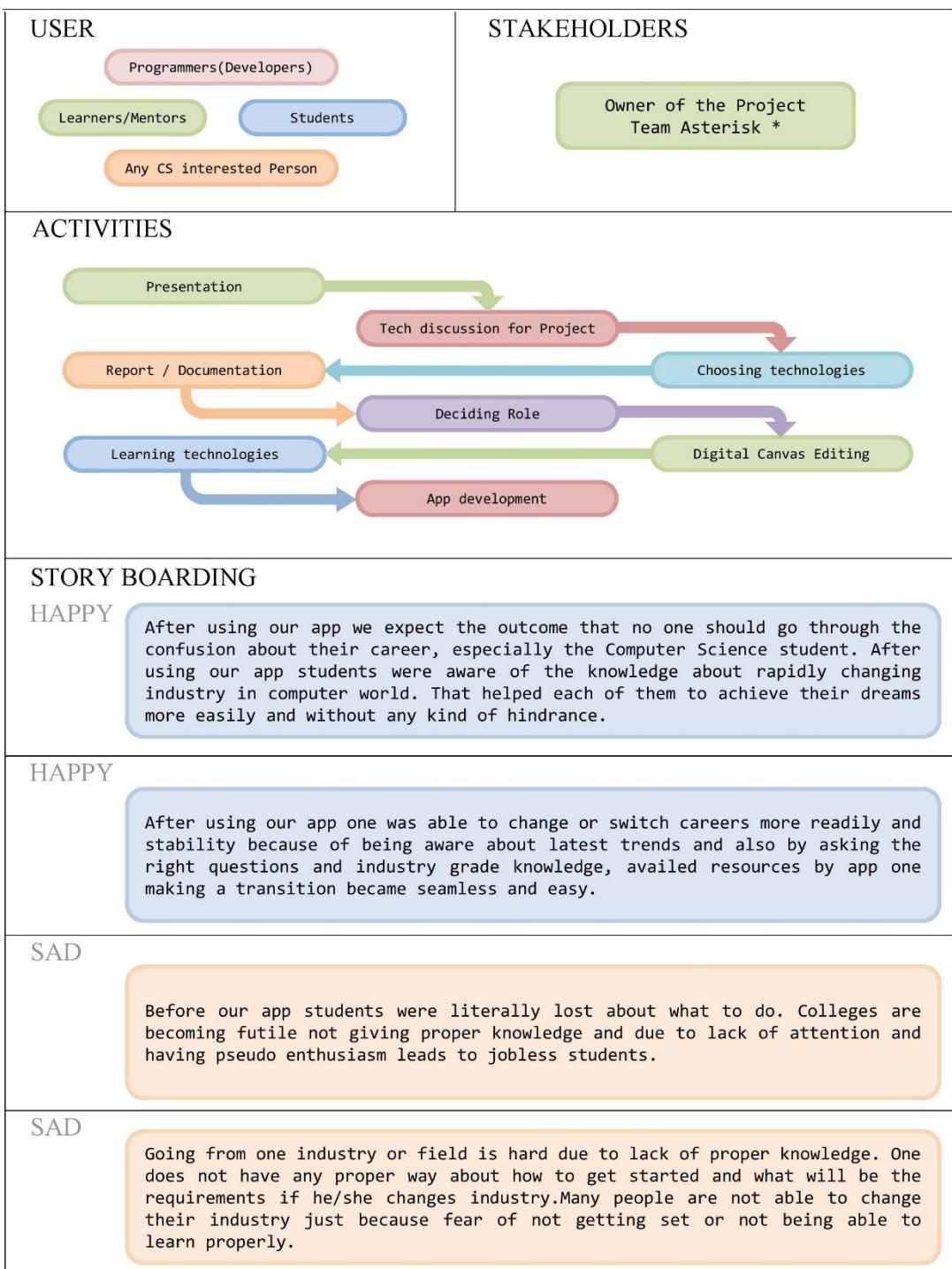
Date : 09/10/2020

Design By :

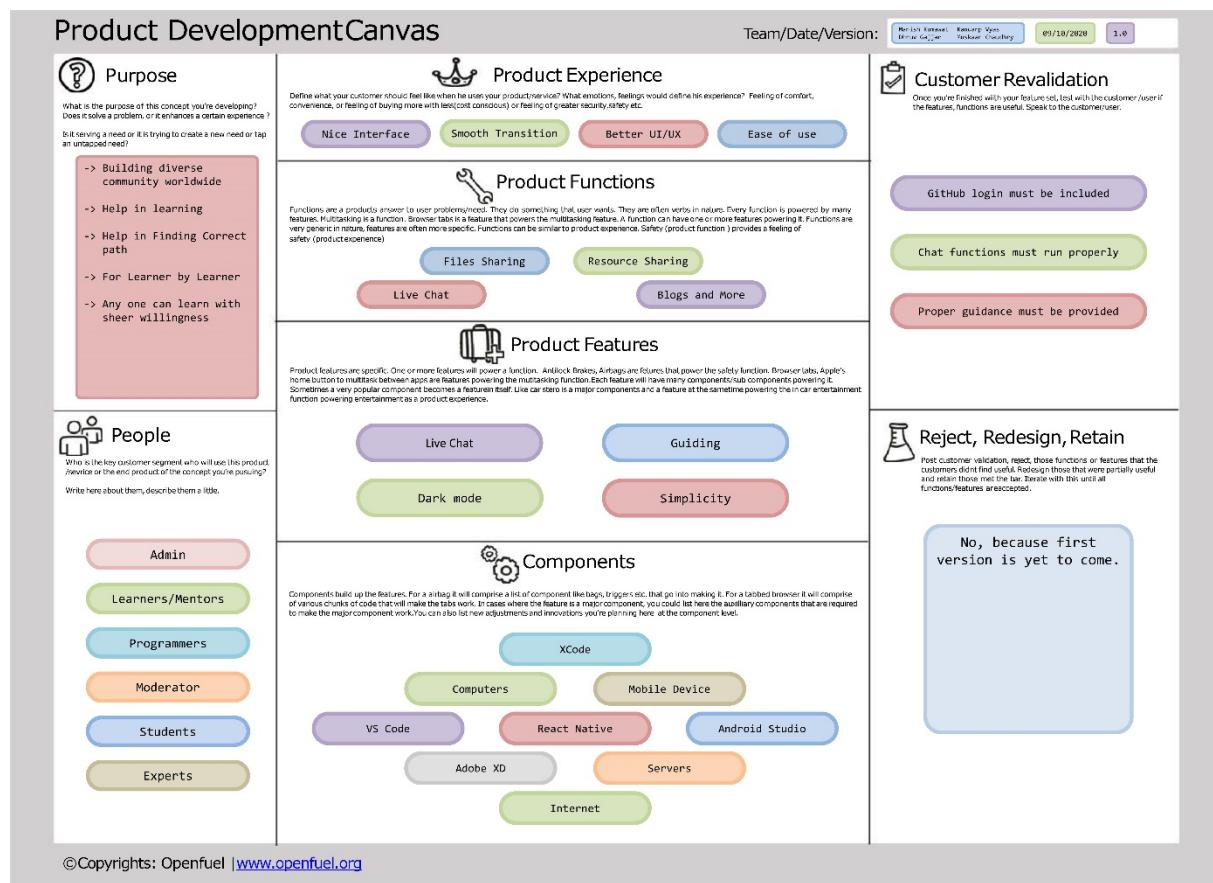
Manish Kumawat
Dhruv Gajjar

Kandarp Vyas
Muskaan Chaudhry

Version : 1.0



5.2 Product Development





Purpose

What is the purpose of this concept you're developing?
Does it solve a problem, or it enhances a certain experience ?

Is it serving a need or it is trying to create a new need or tap an untapped need?

- > Building diverse community worldwide
- > Help in learning
- > Help in Finding Correct path
- > For Learner by Learner
- > Any one can learn with sheer willingness



People

Who is the key customer segment who will use this product /service or the end product of the concept you're pursuing?

Write here about them, describe them a little.

Admin

Learners/Mentors

Programmers

Moderator

Students

Experts

Product Experience

Define what your customer should feel like when he uses your product/service? What emotions, feelings would define his experience? Feeling of comfort, convenience, or feeling of buying more with less(cost conscious) or feeling of greater security,safety etc.

Nice Interface Smooth Transition Better UI/UX Ease of use

Product Functions

Functions are a products answer to user problems/need. They do something that user wants. They are often verbs in nature. Every function is powered by many features. Multitasking is a function. Browser tabs is a feature that powers the multitasking feature. A function can have one or more features powering it. Functions are very generic in nature, features are often more specific. Functions can be similar to product experience. Safety (product function) provides a feeling of safety (product experience)

Files Sharing Resource Sharing
Live Chat Blogs and More

Product Features

Product features are specific. One or more features will power a function. Antilock Brakes, Airbags are features that power the safety function. Browser tabs, Apple's home button to multitask between apps are features powering the multitasking function. Each feature will have many components/sub components powering it. Sometimes a very popular component becomes a feature in itself. Like car stereo is a major component and a feature at the same time powering the in car entertainment function powering entertainment as a product experience.

Live Chat Guiding
Dark mode Simplicity

Components

Components build up the features. For a airbag it will comprise a list of component like bags, triggers etc. that go into making it. For a tabbed browser it will comprise of various chunks of code that will make the tabs work. In cases where the feature is a major component, you could list here the auxiliary components that are required to make the major component work. You can also list new adjustments and innovations you're planning here at the component level.

XCode Mobile Device
Computers React Native Android Studio
VS Code Adobe XD Servers
Internet



Customer Revalidation

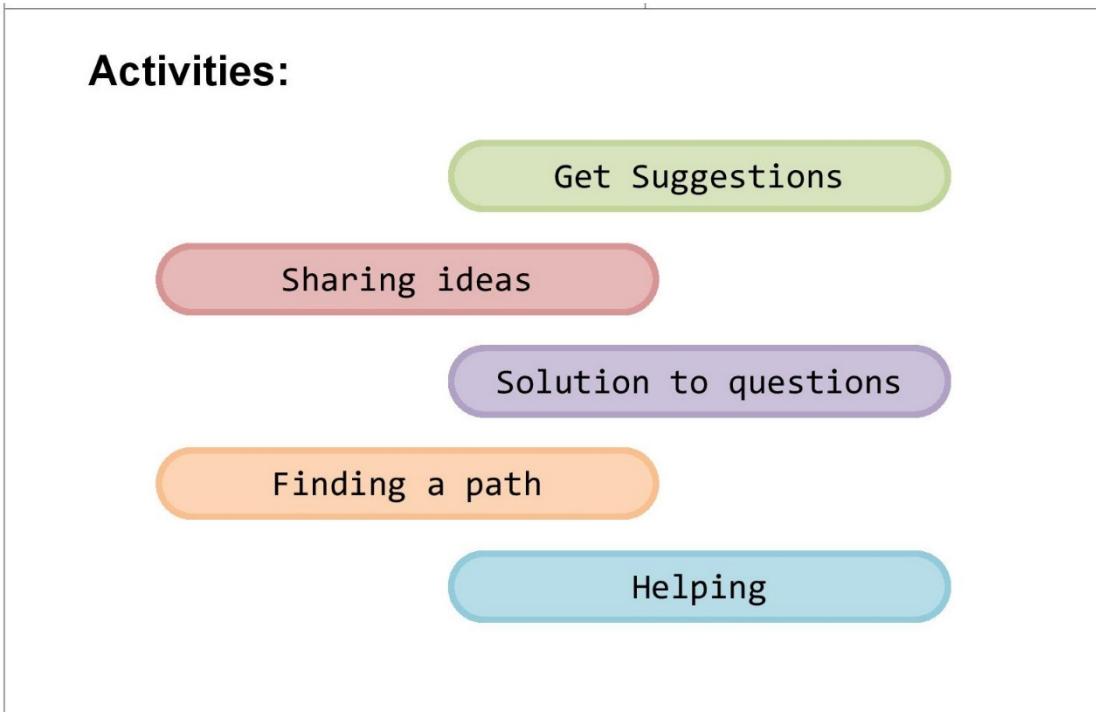
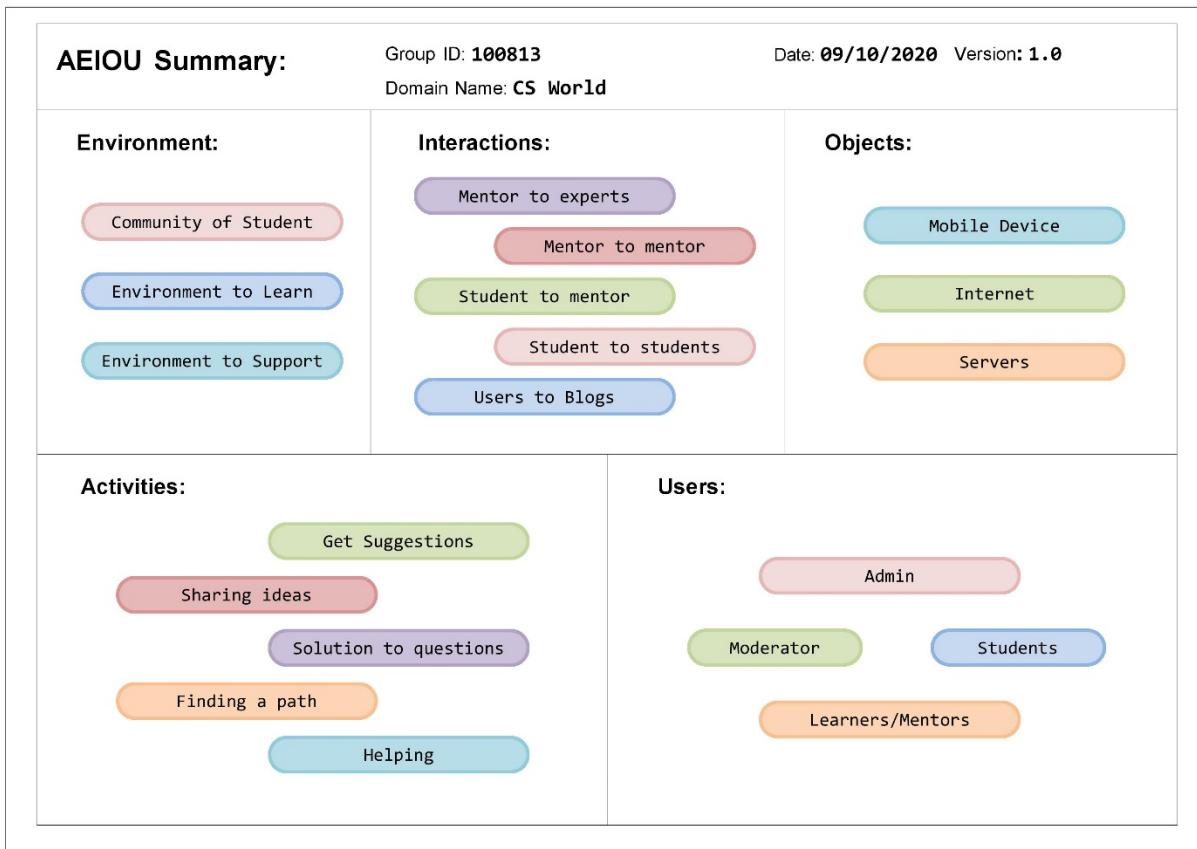
Once you're finished with your feature set, test with the customer / user if the features, functions are useful. Speak to the customer / user.

GitHub login must be included

Chat functions must run properly

Proper guidance must be provided

5.3 AEIOU



Environment:

Community of Student

Environment to Learn

Environment to Support

Interactions:

Mentor to experts

Mentor to mentor

Student to mentor

Student to students

Users to Blogs

Objects:

Mobile Device

Internet

Servers

Users:

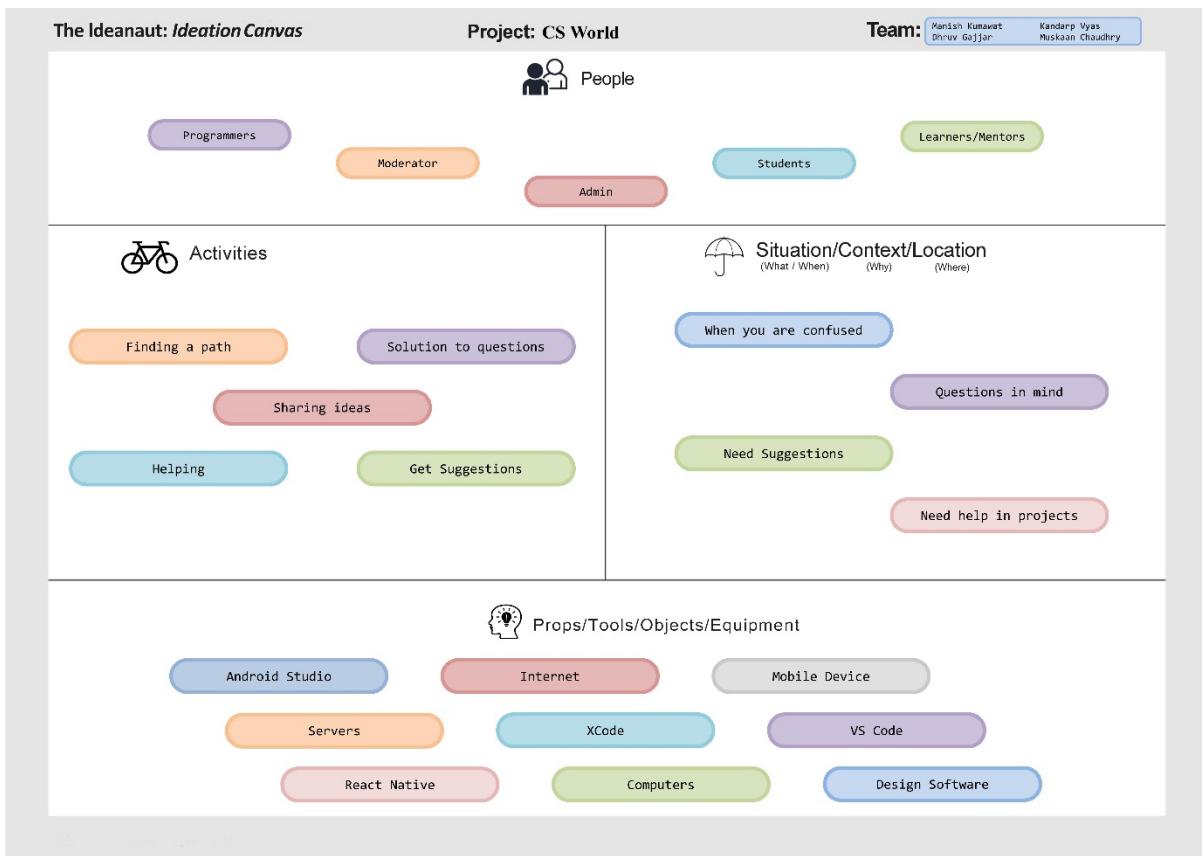
Admin

Moderator

Students

Learners/Mentors

5.4 Ideation



Activity

Finding a path

Solution to questions

Sharing ideas

Helping

Get Suggestions



Situation/Context/Location

(What / When)

(Why)

(Where)

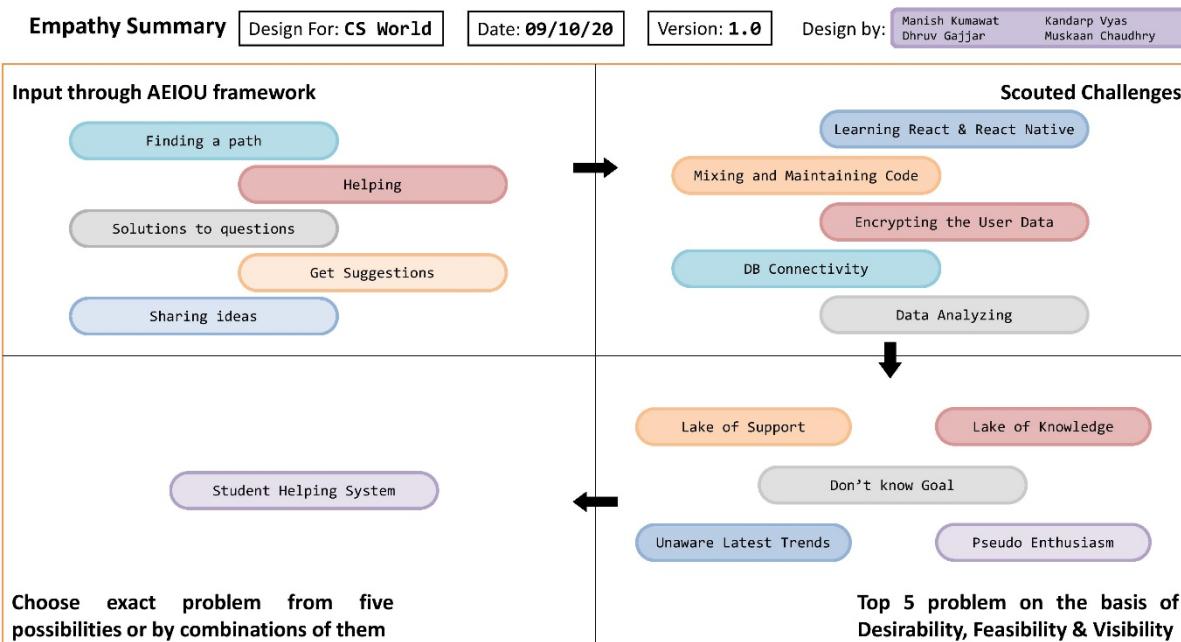
When you are confused

Questions in mind

Need Suggestions

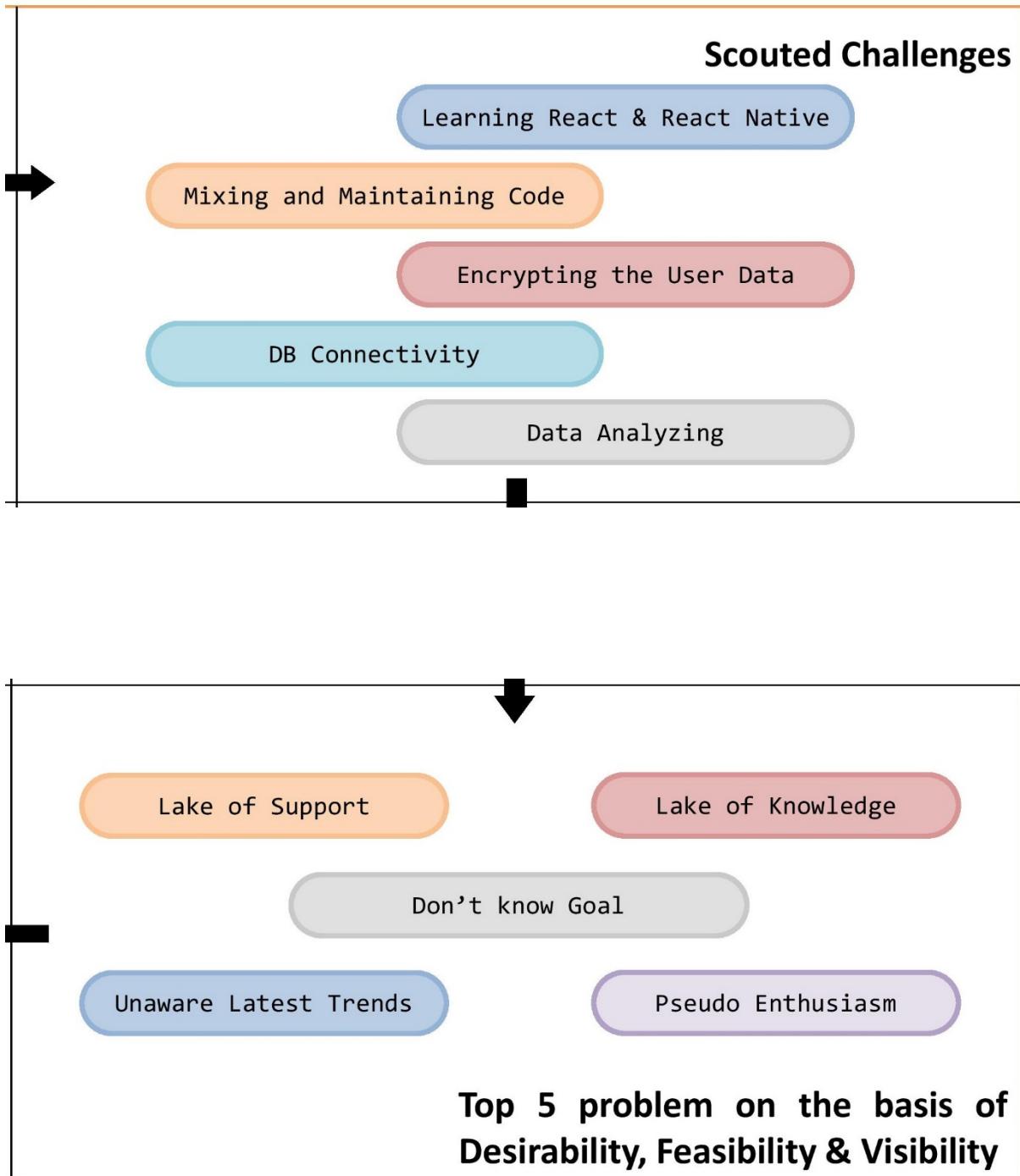
Need help in projects

5.5 Empathy Summary



Input through AEIOU framework

- Finding a path
- Helping
- Solutions to questions
- Get Suggestions
- Sharing ideas



Student Helping System



Choose exact problem from five possibilities or by combinations of them

CHAPTER: 6

IMPLEMENTATION PLANNING AND DETAILS

CHAPTER 6

IMPLEMENTATION PLANNING AND DETAILS

6.1 Implementation Environment

- This describes the technological and physical environment in which the product is to be installed. It includes automated, mechanical, organizational, and other devices, along with the nonhuman adjacent systems.
- Here product is for students and mentors so it will make it easier to for them to connect each other and make it better.
- It contains two main devices that are Servers and Mobile devices. Majorly it contains software components.
- As it will be used by students and mentors it is multi user software and it is based on GUI.

6.2 Program Modules and Specification

- Module is the way to improve the structure design by break down the problem for solving it into independent task.

Advantages of Module

- It breakdown the problem into independent modules so the complexity of the problem can be minimized.
 - Each independent module can be easily assigned to the various members of the development team.
 - Module can be easily run and tested independently from another.
-
- In this software we divided all the parts into multiple subparts so that work on particular part can be easier.

- Some of major modules are Login, Signup, Chat, Blogs. For backend there are also modules for database connection, control the data flow, process done by all modules.

6.3 Security features

- We are using Google Firebase as a Database which is one of the most powerful, reliable and secure servers in the world. Our motive is to provide secure and reliable software.
- For Security purpose there are multiple privileges given to the users. As per their permission they can make modifications to specific part for which permission is given to the that particular user.
- Two factor authentications can be enabled by the user if needed. Also pin for the application will be enabled on new device.

6.4 Coding Standards

- Coding standards are a set of guidelines, best practices, programming styles and conventions that developers adhere to when writing source code for a project. All big software companies have them.

Write Comments and Documentation

- Perhaps one of the first things you learn as a developer is to comment your code. At first it may seem like a waste of time, following the mentality of 'If they are a developer too – they can understand it'. While it is true some of the time, commenting your code and providing proper documentation will guide the other developers through the algorithm and logic that you implemented. But don't get carried away and comment every line of code! Obvious code should be left as is.

Write readable yet efficient code

- Readable codes are easy to follow, yet use optimal space and time. When writing code, you may often want to write it in as little lines as possible. Perhaps you can write an entire method or function in one line, but that only makes it harder to read and understand.

If avoidable, do NOT hard-code!

- The only thing that should be hard-coded are constants. Functions must be kept generic for further and future use.

Write readable yet efficient code. Confirm to the coding standards of your current project

- *"DISCIPLINED PERSONAL PRACTICES CAN REDUCE DEFECT INTRODUCTION RATES BY UP TO 75 PERCENT"*
- Every project/company has their own coding standards. Some might prefer one style over the other when it comes to things such as naming conventions, file structure spacing.
- There are IDEs where you can set the preferred style, which will auto-correct you when you save. It is easier to read and, therefore understand, when all the files of the projects use the same style, naming convention, spacing, etc.

CHAPTER: 7

TESTING

CHAPTER 7

TESTING

7.1 Test plan

As it is the first release, an approved plan with approved test bed, testing types etc., should be in place.

- The application is planned to work on the android devices and is said to be suitable for various network interferences.
- Further the UI is designed to suit all kind of mobile sizes or is said to be adjustable as per individual screens.
- Beta testing is going to be carried out before proceeding with the actual model release.
- Feedback from the end user holds most importance in the testing and the releasing of the application.
- The automation tools required for the application is with all the necessary permissions and certifications.

7.2 Testing Strategy

Following are the testing strategies to be focused onto:

Device Selection:

- This is one of the most critical steps before starting the android application testing process.
- Decide which devices are to be taken into account for the testing process.
- The selection is to be done so as to maximize the number of target customers.
- Factors such as OS version, Screen resolution and Form factors [Tablet or smart phones] play a vital role in the selection phase.
- If required, even the help of Emulators can be taken into account.
- But emulators should not replace the physical device testing process.
- Device emulators are cost effective and they come in handy during the initial development phase.

- But, to test the real-life scenarios, physical devices are the must. Both emulators and physical devices are to be used in a balanced manner for an optimized result.

Beta Testing of the Application:

- Beta testing is very effective in testing with the real-world users, real devices, actual networks and applications installed in a wide geography.
- This gives a clear picture of the network density, network variations [Wi-Fi, 4G, 3G, and 2G] and the impact on the application.
- Beta testing in the real world is one of its kind and cannot be replicated in a controlled environment.

Connectivity:

- Normally, Android applications are connected to the internet for various requirements.
- The connectivity on different devices plays a key role in putting up the strategy.
- Mostly the connectivity is controlled by simulation software which helps in regulating the network speed, latency, and limited connectivity while testing.
- It is said that testing under real network connections is always advisable for real-time result/data.

Manual or Automated Testing:

- Though automation testing takes ample amount of time for the first run, it comes in handy when the testing has to be repeated. This also reduces the overall time span of testing during the different development stages.
- Android Automation should be clubbed with Manual testing when the regression testing repetition is high in the application development phase, compatibility testing has to be done for the same application on different OS versions, backward compatibility checkpoints etc.

7.3 Test methods

The following tests are to be performed on the application:

1) Functional Testing:

Testing is normally achieved by user interface-initiated test flows. Not just the flow of a use case is tested, but the various business rules are also tested. This testing will be done by certifying the requirements. i.e., whether the application is working based on the requirements.

2) Android UI Testing:

This is an user-centric testing of the application. In this test phase, items such as visibility of text in various screens of the app, interactive messages, alignment of data, the look and feel of the app for different screens, size of fields etc are tested.

3) Compatibility Testing:

This testing is done mostly in the form of two matrices of OS Vs app and Device model Vs app. Usually, a list of supported OS (and sometimes devices) will be provided by us.

4) Interface Testing:

In other words, it is also termed as Integration testing. This testing is done after all the modules of the app are completely developed, tested individually and all the bugs are fixed verified.

Interface testing includes tests like a complete end to end testing of the app, interaction with other apps like Maps, social apps etc, usage of Microphone to enter text, usage of Camera to scan a barcode or to take a picture etc.

5) Network Testing:

The app should talk to the intermediate service so as to carry out the process. During this testing, request/response to/from the service is tested for various conditions. This test is mainly done to verify the response time in which the activity is performed like refreshing data after sync or loading data after login etc.

6) Performance Testing:

Performance of the application under some peculiar conditions are checked.

Those conditions include:

- Low memory in the device.
- The battery is extremely at a low level.
- Poor/Bad network reception.

7) Installing Testing:

This is to ensure smooth installation of the application without ending up in errors, partial installation etc. Upgrade and uninstallation testing are carried out as part of Installation testing.

8) Security Testing:

Testing of the data flow for encryption and decryption mechanism is to be tested in this phase. Access to stored data is also tested in this phase.

9) Field Testing:

Field testing is done specifically for the mobile data network and not in-house but by going out and using the app as a normal user. This testing is done 'only' after the whole app is developed, tested and regressed (for bugs and test cases). It is basically done to verify the behaviour of the app when the phone has a 2G or 3G connection. Field testing verifies if the app is crashing under slow network connection or if it is taking too long to load the information.

10) Interrupt Testing:

This type of testing is also known as Offline Scenario Verification. Conditions where the communication breaks in the middle are called as offline conditions.

7.4 Test case

The purpose of the application is to provide interactive as well as user friendly platform that is easy to use and does not possess any harmful threat to the privacy. As this being stated, the application is supposed to pass all the testing methods and beta test period, before being launched in the market.

Required output:

- UI: An aesthetically pleasing UI with easy-to-use functionality, Screen fits for all the android devices, non-changing fonts and figures.
- Network: Capable of running smoothly in network variations, doesn't necessarily requires high speed internet.
- Privacy: Good login setup, back up availability, authenticity and verification
- Compatibility: Capable of running on various OS setups, can work on newest OS updation.

Expected output:

- UI: A good UI that fits the screen
- Network: Network issues won't necessarily affect the performance.
- Privacy: powerful login and authentication system
- Compatibility: Will be able to work on several OS

CHAPTER: 8

SCREENSHOTS

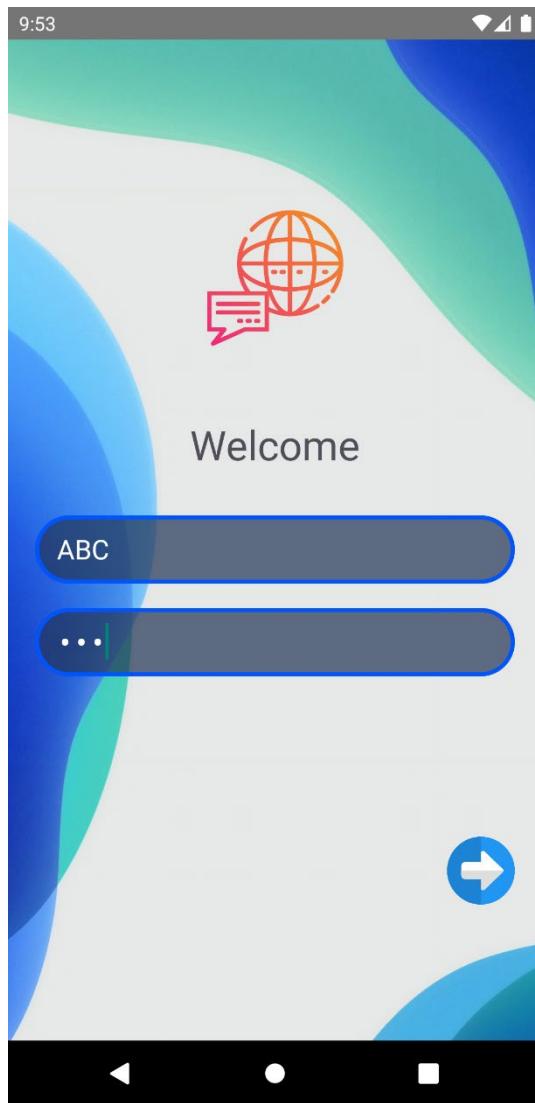
AND

USER MANUAL

CHAPTER 8

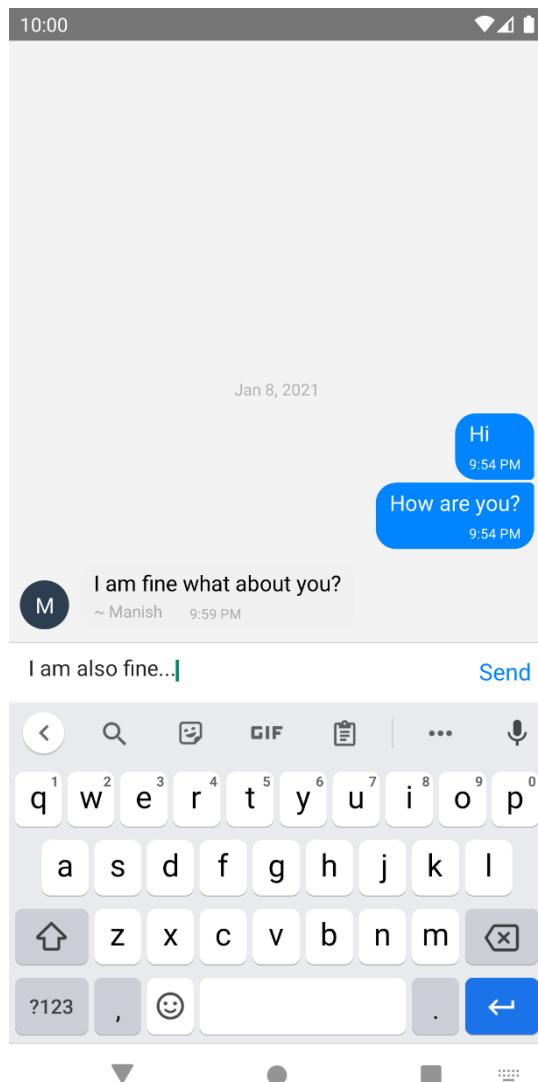
SCREENSHOTS AND USER MANUAL

8.1 Login Screen



This is the Login page where user can use their id and password to login into the system to access various resources.

8.2 Live Chat Screen



Here user can discuss with all the users who are online at that moment.