
Software Requirements Specification

for

CAB SHARING

Version <1.1>

Prepared by

Group 18:

Group Name: Code Closed

AAKASH LAWLA
ABHISHEK GURJAR
AKASH KUMAR BHOI
RISHABH MUKATI
PRINCE KUMAR AHIRWAR
HARIOM SHAKYAWAL
GOPAL AGGARWAL
UJJAWAL GOYAL
SOURABH MINA
SOURAV ANAND

190006
190037
190082
190704
190646
190354
200390
201058
200996
200997

aaklawala@iitk.ac.in
gurjara96@gmail.com
akashbhoi525@gmail.com
mukadirishabh02@gmail.com
princeprinceahirwar@gmail.com
hariomkoli306@gmail.com
gopalagggarwal5858@gmail.com
ujjawalgo247@gmail.com
sourabh2002wow@gmail.com
souravanand1982@gmail.com

Course: CS253

Mentor TA: Pinaki Chakraborty

Date: 31-01-2022

| | |
|---|-----------|
| CONTENTS | II |
| REVISIONS | II |
| 1 INTRODUCTION | 1 |
| 1.1 PRODUCT SCOPE | 1 |
| 1.2 INTENDED AUDIENCE AND DOCUMENT OVERVIEW | 1 |
| 1.3 DEFINITIONS, ACRONYMS AND ABBREVIATIONS | 1 |
| 1.4 DOCUMENT CONVENTIONS | 2 |
| 1.5 REFERENCES AND ACKNOWLEDGMENTS | 2 |
| 2 OVERALL DESCRIPTION | 4 |
| 2.1 PRODUCT OVERVIEW | 4 |
| 2.2 PRODUCT FUNCTIONALITY | 4 |
| 2.3 DESIGN AND IMPLEMENTATION CONSTRAINTS | 5 |
| 2.4 ASSUMPTIONS AND DEPENDENCIES | 5 |
| 3 SPECIFIC REQUIREMENTS | 5 |
| 3.1 EXTERNAL INTERFACE REQUIREMENTS | 5 |
| 3.2 FUNCTIONAL REQUIREMENTS | 7 |
| 3.3 USE CASE MODEL | 8 |
| 4 OTHER NON-FUNCTIONAL REQUIREMENTS | 16 |
| 4.1 PERFORMANCE REQUIREMENTS | 16 |
| 4.2 SAFETY AND SECURITY REQUIREMENTS | 16 |
| 4.3 SOFTWARE QUALITY ATTRIBUTES | 16 |
| 5 OTHER REQUIREMENTS | 18 |
| APPENDIX A – DATA DICTIONARY | 20 |
| APPENDIX B - GROUP LOG | 22 |

Revisions

| Version | Primary Author(s) | Description of Version | Date Completed |
|-----------------------|-------------------|---|----------------|
| Draft Type and Number | Full Name | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | 00/00/00 |

1 Introduction

1.1 Product Scope

We are making a website for the IIT Kanpur campus community. On our Cab Sharing website, one can share a cab or auto with another person and know if someone wants to go to the same destination and share the cab. The primary purpose of this is to minimize the cost of traveling to a particular destination. Benefits are:-

- 1.) First, travel expenses will be less.
- 2.) Pollution will be less if we share cabs.
- 3.) People can find a mate to chat with during their journey and make it joyful.

Objectives:-

Campus communities can share details like date, time, location, number of people preferred, etc. And according to their priorities, they'll get notifications. There will be a login interface also. If they find an appropriate match, then they can chat also.

1.2 Intended Audience and Document Overview

This document is intended for project developers, project managers, marketing staff, document writers, and test users. So that they can understand the functionalities of our website and can give their feedback so we can improve our product accordingly.

This document contains four more sections. In the first section, one can find the overall description of the product, like functionalities and design and implementation constraints. Specific functional requirements are mentioned in the second section, and non-functional requirements in the third section. And in the last section for other requirements if there are any.

1.3 Definitions, Acronyms and Abbreviations

Definition:-

- **Customers:** The user using the cab sharing service for travelling. (They are the target audience for our program, in this case, *the campus community*).
- **Match:** The match defines a prospective partner between users travelling via/to the exact location, giving the matched customers a chance to travel together.
- **Driver:** Driver refers to the cab drivers whing their cab to fulfil the journey.
- **Filter:** Filters give the customers, i.e., the users, an option to do a custom search for their partner for travel that matches their interest, etc.

Abbreviations and Acronyms:

| | |
|--------|-----------------------------|
| CSS | Cascading style sheets |
| etc | Et cetera |
| js | java script |
| HTML | Hypertext Markup Language |
| UI | User Interface |
| vscode | Visual studio code |
| HTTP | HyperText transfer Protocol |
| TA | Teaching Assistant |
| PC | Personal Computer |

1.4 Document Conventions

| | |
|-----------------------|-------------|
| Font | Arial |
| Text Font size | 11 |
| Heading Font size | 14 |
| Sub Heading Font size | 12 |
| Sub Heading Font | Arial(Bold) |
| Heading Font | Arial(Bold) |
| Commenting | Italic |

1.5 References and Acknowledgments**References:**

- [SRSTemplate-CS253-2022.docx](#)

- <https://developer.ibm.com/articles/the-sequence-diagram/>
- <https://app.creately.com/diagram/lKfcic53S6z/edit>
- https://en.wikipedia.org/wiki/System_context_diagram

Acknowledgements:

We would like to express our sincere gratitude to our TA, Mr Pinaki Chakraborty, for his constant support, assistance and instruction during this project. His feedback and suggestions were invaluable, and this helped us complete this document and get deep insight into our project.

Also, we are thankful to our course instructor Prof. Indranil Saha for giving us this excellent opportunity to explore our creativity and hone our skills.

2 Overall Description

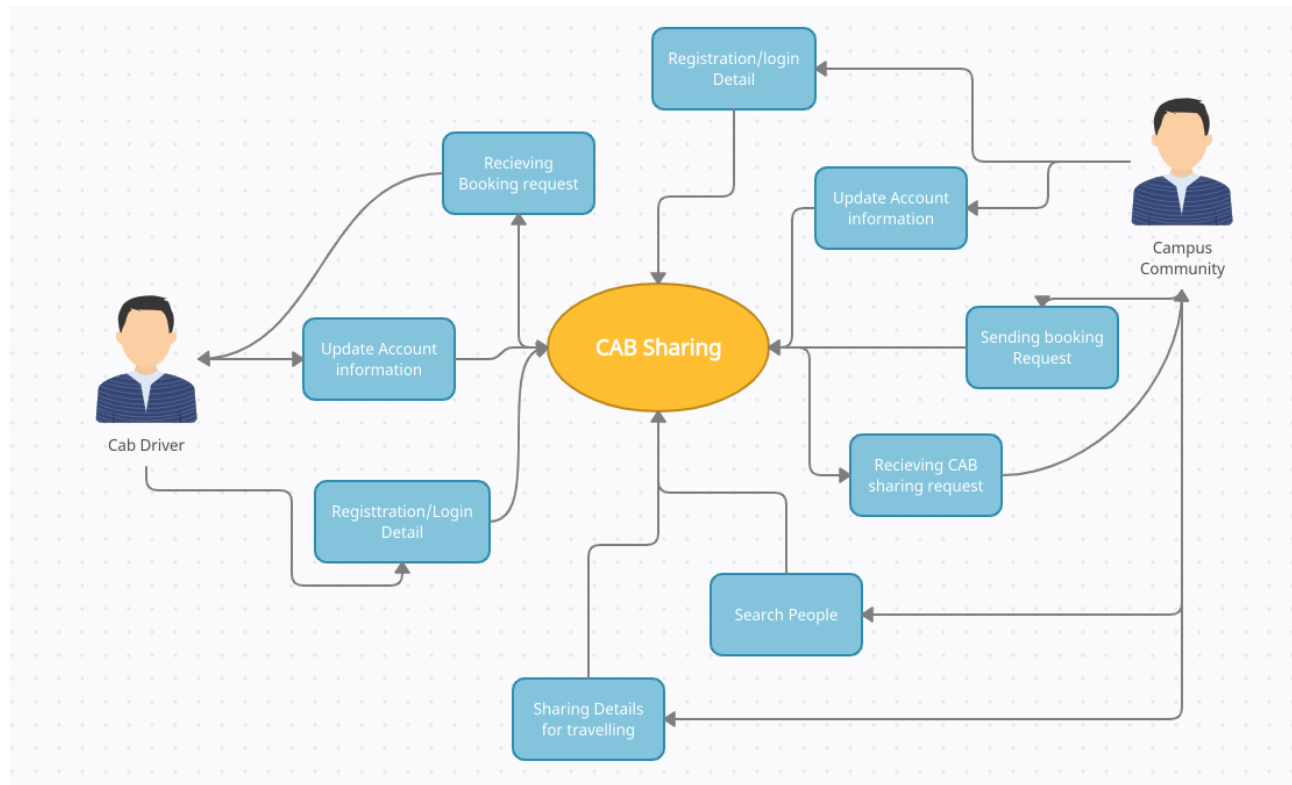
2.1 Product Overview

Cab Sharing aims to provide user-friendly software to the campus community who seek their travelling partner to go outside from the campus or come into the campus and to the cab drivers who seek their customers.

People can register themselves as drivers or travellers using their credentials.

The registered user will find or request their partner/s to share a cab according to their preferences like the destination, cab size, rent, etc. After that, they will contact each other and send a booking request to the drivers.

On the other hand, drivers will accept or reject the requests and update their availability, charges, profile, etc.



2.2 Product Functionality

Campus Community

- Campus communities can share their details like date, time, location, number of people preferred, etc.
- Users can search for another user and send a request if the details of another user are comfortable.

- Users can accept or decline the request.
- Users can see the status of cab driver availability and contact details.
- Users can chat through the chat window about the journey, pick-up location, etc.
- Users can send a request to the cab driver for a ride with the pickup location, destination, and timing details.

Cab driver

- Cab drivers can register themselves.
- Cab drivers can update their Availability details like Booked, Available, Not functional.
- Cab drivers can update their charges.
- Cab drivers can accept or decline the request.

2.3 Design and Implementation Constraints

The technology used in the project is up to date with current standards to run smoothly on all browsers and operating systems.

Technologies tools required are:

- Frontend - HTML, CSS, Bootstrap, Javascript
 - Backend - node.js, express
 - DBMS - MongoDB
 - editor - vscode
-
- Passwords of the user shall be encrypted in DBMS for security purposes.
 - When the system crashes, it will return at most one hour for maintainability purposes.
 - The system will work 24*7.

2.4 Assumptions and Dependencies

- It is assumed that users should be familiar with computers and have an internet connection.
- It is assumed that information given by the user is correct.
- It is assumed that users are familiar with the English language.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

Registration/Login Page

On this page, users can register themselves. If the user is already registered, they can log in using their mobile no. or email id and a generated password. After the registration, the user goes to their respective interface.

Campus Community Interface

Other than the menu bar, news and advertisements appear on the dashboard.

The things on the dashboard menu of the campus community are

- **Profile:** It contains all the user's details. They can edit their profile.
- **Travel detail:** In these sections, you can apply for cab sharing and fill in the information like date, destination, luggage info, time, etc. The user can also update this info.
- **Match:** These buttons match the info which you give on the Travel detail page with other users and provide all the matches (if available).
- **Search:** In this, users can search for others by their initial and final destination, dates, etc.
- **Notification:** If someone sends a request or accepts/rejects the request, the notification comes in these sections.
- **Requests:** There are two sections in this
 - **Sent:** All the requests the user sends appear in this section.
 - **Received:** In this section, all the requests which are received appear.
- **Chat:** In this section, the user can chat with other users.
- **Driver Detail:** This section will show all the details of cab drivers.

CAB Driver Interface

At the dashboard, news and advertisements appear other than the menu bar.

The things that are on the dashboard menu of the cab driver are

- **Profile:** It contains all the details of the user. They can edit their profile and update their status here.
- **Notification:** The notification appears on these sections if someone sends a request.
- **Request:** In these sections, all the requests come which are sent by other users.

3.1.2 Hardware Interfaces

The Cab Sharing project has no specific hardware requirement.

3.1.3 Software Interfaces

- MongoDB database management system is used for storing all data about a system, for example, user, journey time, and destination. It has an excellent open-source **UI** that displays tables and rows in well-formatted form for developers to manage the whole database.
- For **UI**, HTML and CSS will illustrate the system attractively. And Javascript and HTML languages will handle client and server-side requirements with HTTP requests.

3.2 Functional Requirements

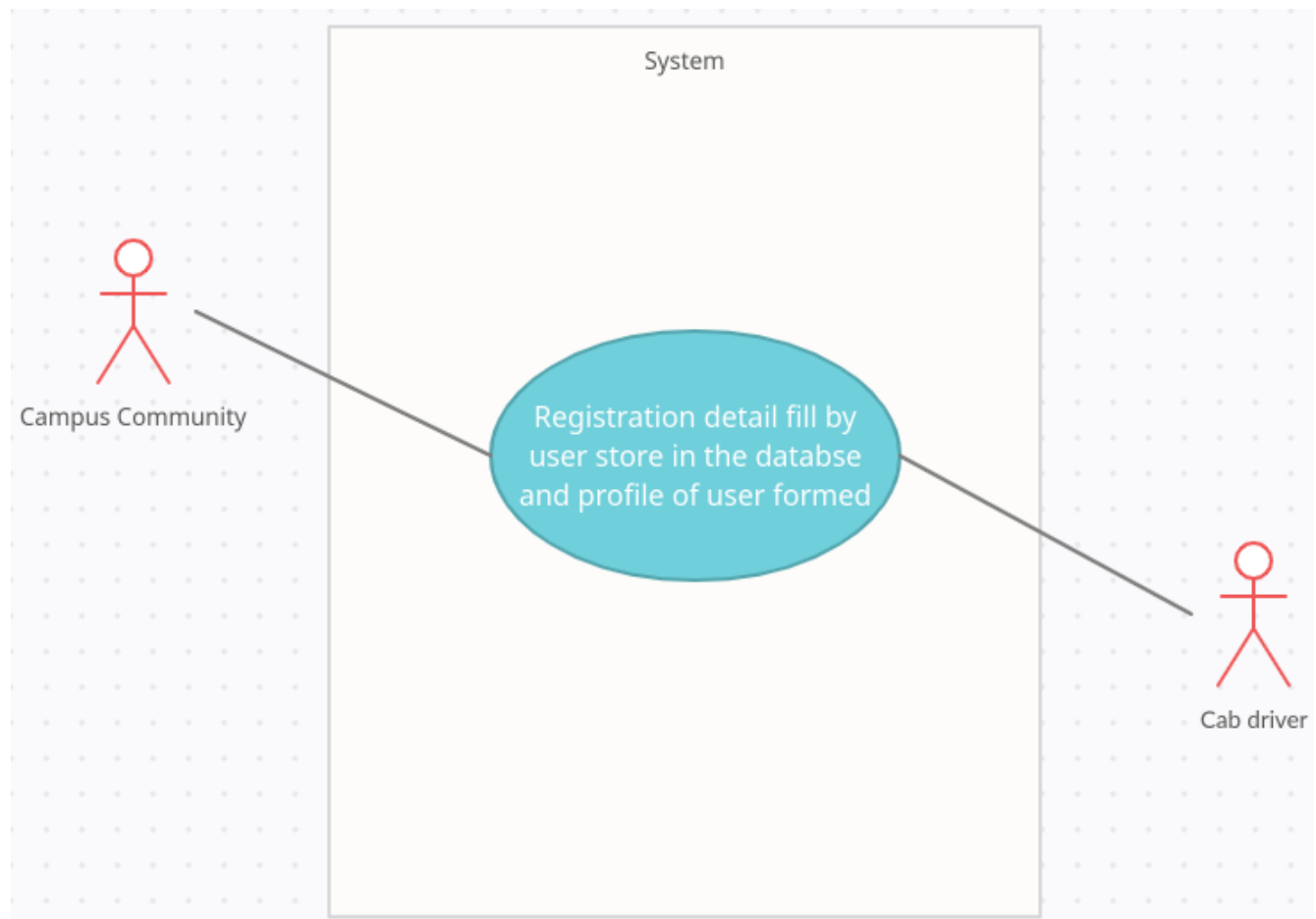
- **Registration**
 - Users can register them as a student or as a driver.
 - Users have to enter their details like Name, Address, Email, Phone Number etc.
 - If the user's Email ID is found un-registered, at the , then the registration process will continue, else the user will be redirected to the Login Page.
 - Users have to pick a username and set a password.
 - Username has to be a unique Username and Password has to be a minimum of 8 characters including both case letters and numbers and a special symbol and max 16 characters.
 - After successful registration the user will be redirected to the login page.
- **Login**
 - Users have to enter the username and password.
 - If login credentials are found valid then they enter the main page.
 - Otherwise It will show a warning message and a forgot password option.
- **Forgot Password**
 - Users have to enter the email ID and submit it.
 - If the email is found registered then a reset password link will be sent to that email.
 - Otherwise it will show a warning and redirect to the registration page.
- **Profile**
 - Users will be able to see their details.
 - They will be able to update their details.
 - They will be able to Logout and after logout it will redirect to the login page.
- **Update Profile**
 - Users have to enter new details like Phone number, Email etc.
 - If a user wants to update the email id or mobile no., it must be unregistered and a verification link will be sent to verify their new email id/mobile no.
- **Travel detail and Match**
 - Users will select a date and time and fill in the required information like starting location and final destination.
 - They will get the list of the available people based on their input information.
 - They will have to send a request to the most matched person.
- **Search**
 - Users can search for matches related to his\her interests.
 - They can filter the result according to their needs.

- **Notification**
 - Users will get a notification if their request will be accepted.
 - They will also receive information of matched person
- **Requests**
 - User can send a request for cab sharing to the suitable person
- **Chat**
 - Users get an option to chat with their matches.
 - They can send and receive messages from his/her partner.
- **About**
 - Users will get the contact details of developers.
- **Driver Details**
 - Users will find the details of drivers and their contacts.

3.3 Use Case Model

3.3.1 Registration

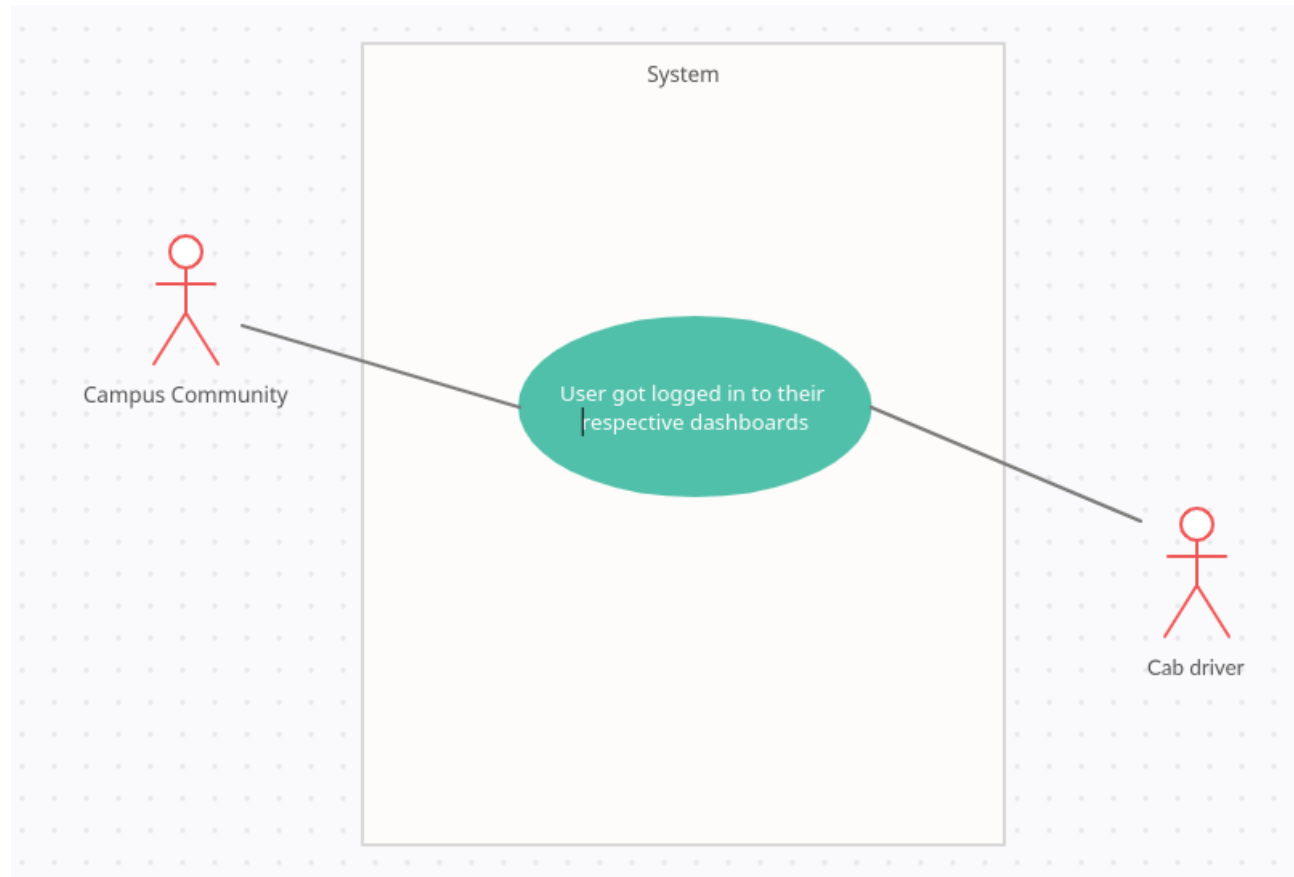
| | |
|--------------------------|--|
| Use case | U1 |
| Author | Hariom Shakyawal |
| Purpose | User registration and profile making |
| Requirement Traceability | The User has access to the registration page. |
| Priority | High |
| Preconditions | None |
| Postconditions | User can sign in any time and can use the system. |
| Actor(s) | User |
| Exception | If given username not available user have to refill details. |
| Note | |



3.3.2 Sign In/ Log In

| | |
|--------------------------|---|
| Use case | U2 |
| Author | Hariom Shakyawal |
| Purpose | For checking that user is registered and get his page |
| Requirement Traceability | The User has access to the login page and has login details |
| Priority | High |
| Preconditions | User must be registered. |
| Postconditions | User will be able to use the system. |
| Actor(s) | User |

| | |
|-----------|---|
| Exception | If the username or password is found invalid user can not enter the system. |
| Includes | U1 |



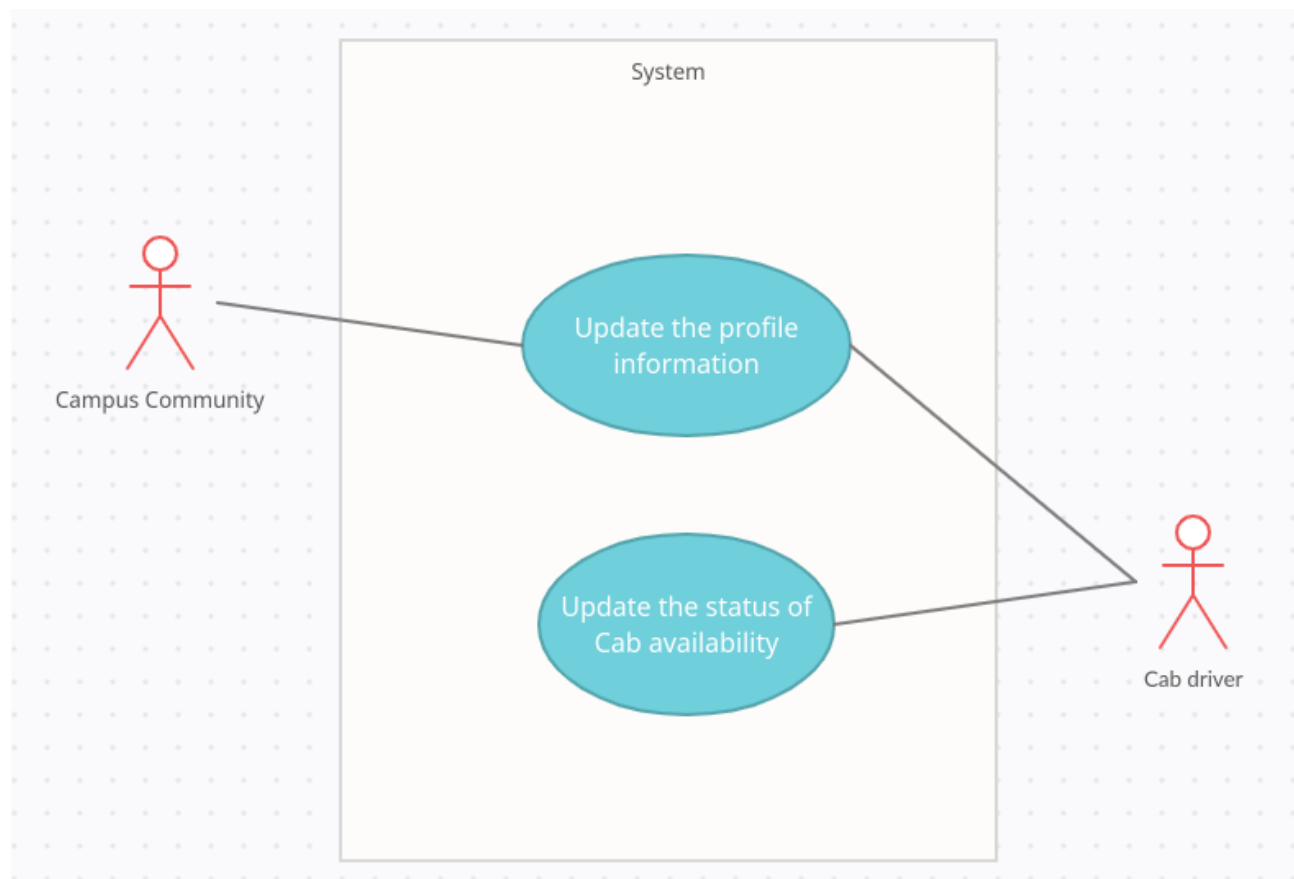
3.3.3 Forgot Password

| | |
|--------------------------|---------------------------------------|
| Use case | U3 |
| Author | Gopal Aggarwal |
| Purpose | For resetting users password |
| Requirement Traceability | The User has access to the login page |
| Priority | Medium |
| Preconditions | User should be registered |

| | |
|----------------|---|
| Postconditions | User will be able to login with a new password |
| Actor(s) | User |
| Exceptions | If the user is found unregistered, the user will be prompted to register on the page. |
| Includes | U1 |

3.3.4 Update Information

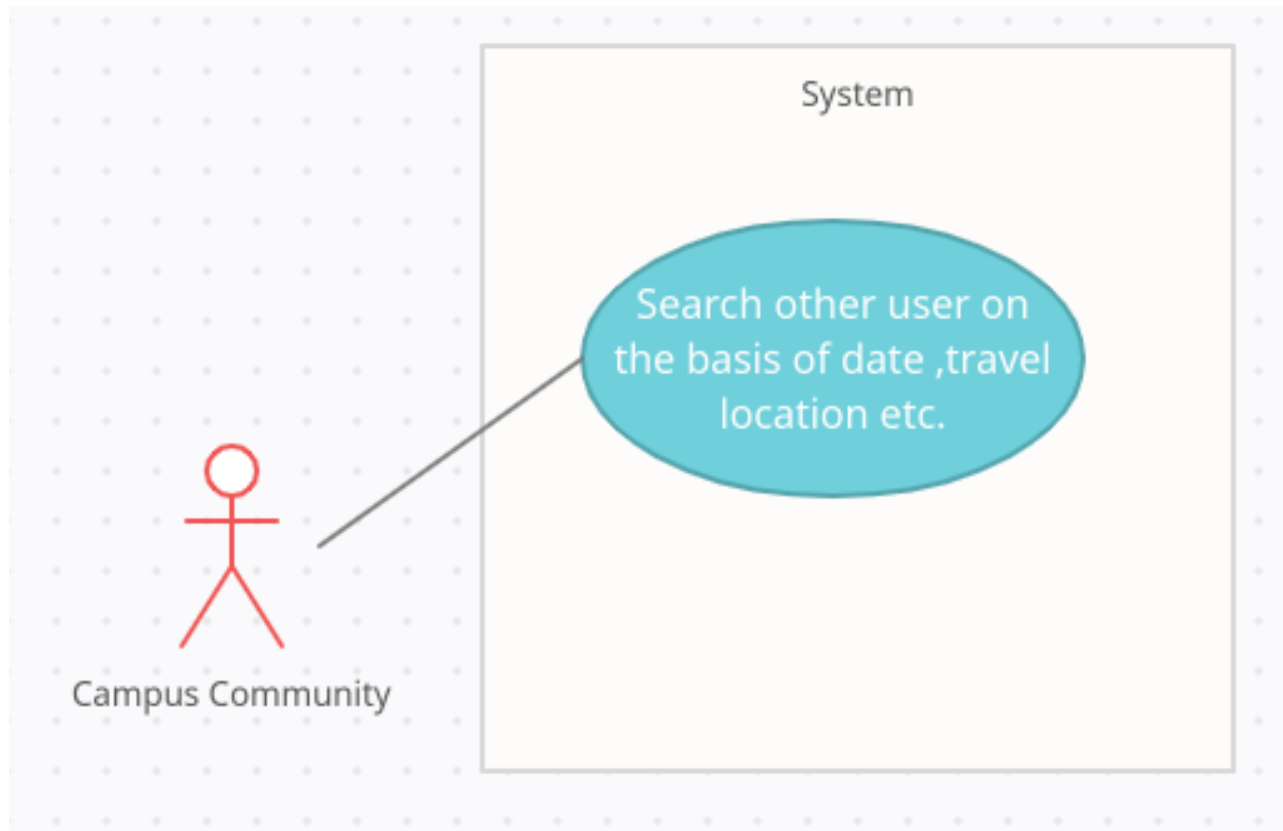
| | |
|--------------------------|---|
| Use case | U4 |
| Author | Rishabh Mukati |
| Purpose | For updating the profile information of the user and cab driver can also update that they are Available/Not Available |
| Requirement Traceability | The User has access to the login page and has login details and access to the profile section. |
| Priority | High |
| Preconditions | User should be a registered member and logged in to the system |
| Postconditions | User profile gets updated, and cab driver status gets updated |
| Actor(s) | Cab Availability status- Cab driver, Update profile - Users |
| Exception | None |
| Includes | U2 |



3.3.5 Search

| | |
|--------------------------|--|
| Use case | U5 |
| Author | Sourav Anand |
| Purpose | To help users searching for vacant seats in cabs |
| Requirement Traceability | The User has access to the login page and has login details and also has access to the search section. |
| Priority | High |
| Preconditions | User should be signed in |
| Postconditions | User will be able to see a list of available seats |
| Actor(s) | Campus Community |
| Exception | none |

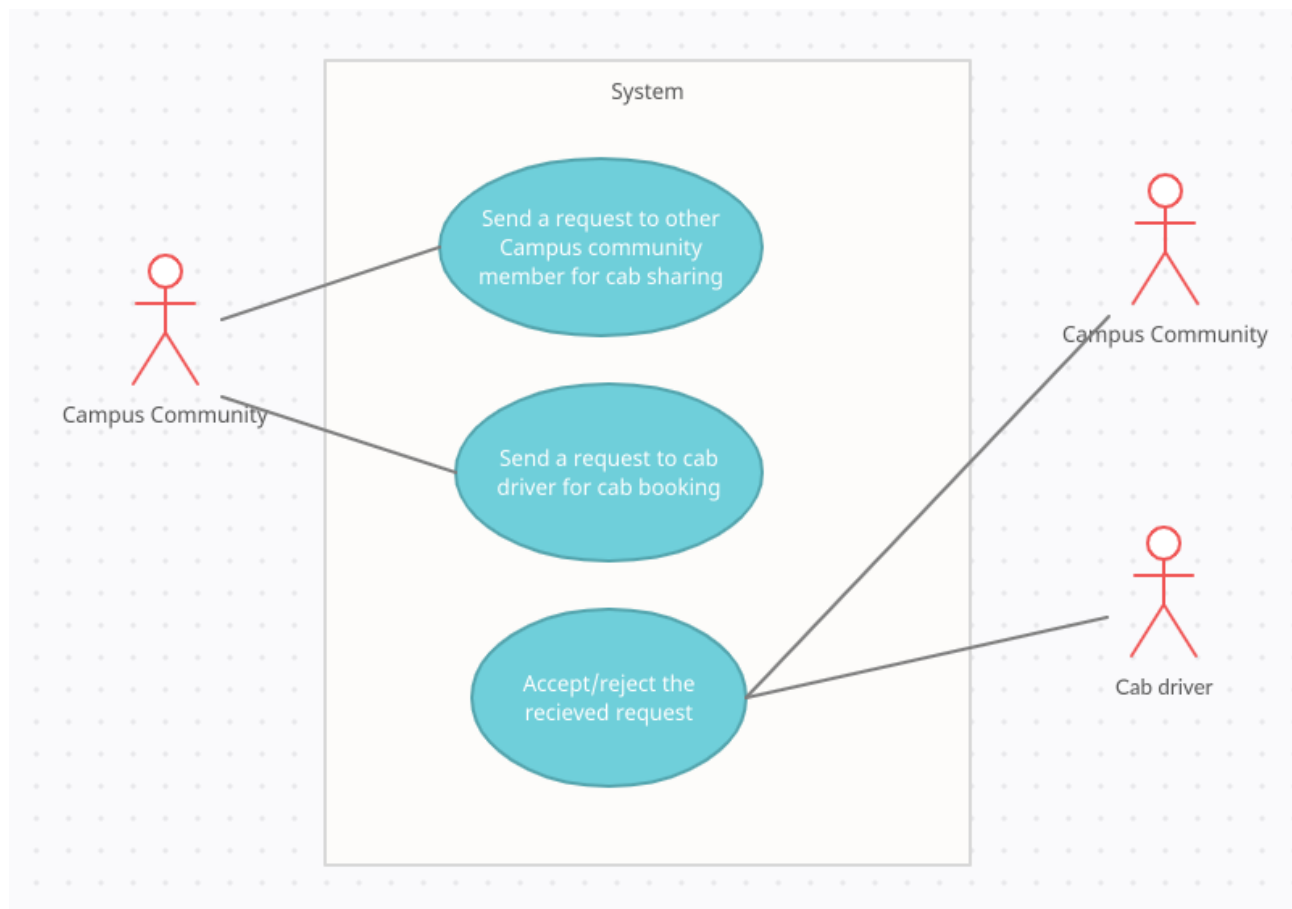
| | |
|----------|----|
| Includes | U2 |
|----------|----|



3.3.6 Sending and Accepting/Rejecting Request

| | |
|--------------------------|--|
| Use case | U6 |
| Author | Rishabh Mukati |
| Purpose | The Campus community can send a request to other Campus community members for cab sharing or to the cab driver for booking cabs, and they can accept or reject the request. |
| Requirement Traceability | The User has access to Search, or the user has access to Travel detail and matches pages and Cab details page to send requests to other users. The campus community has access to the request page to see the request and accept that. |
| Priority | High |
| Preconditions | User should be a registered member and logged in to the system |

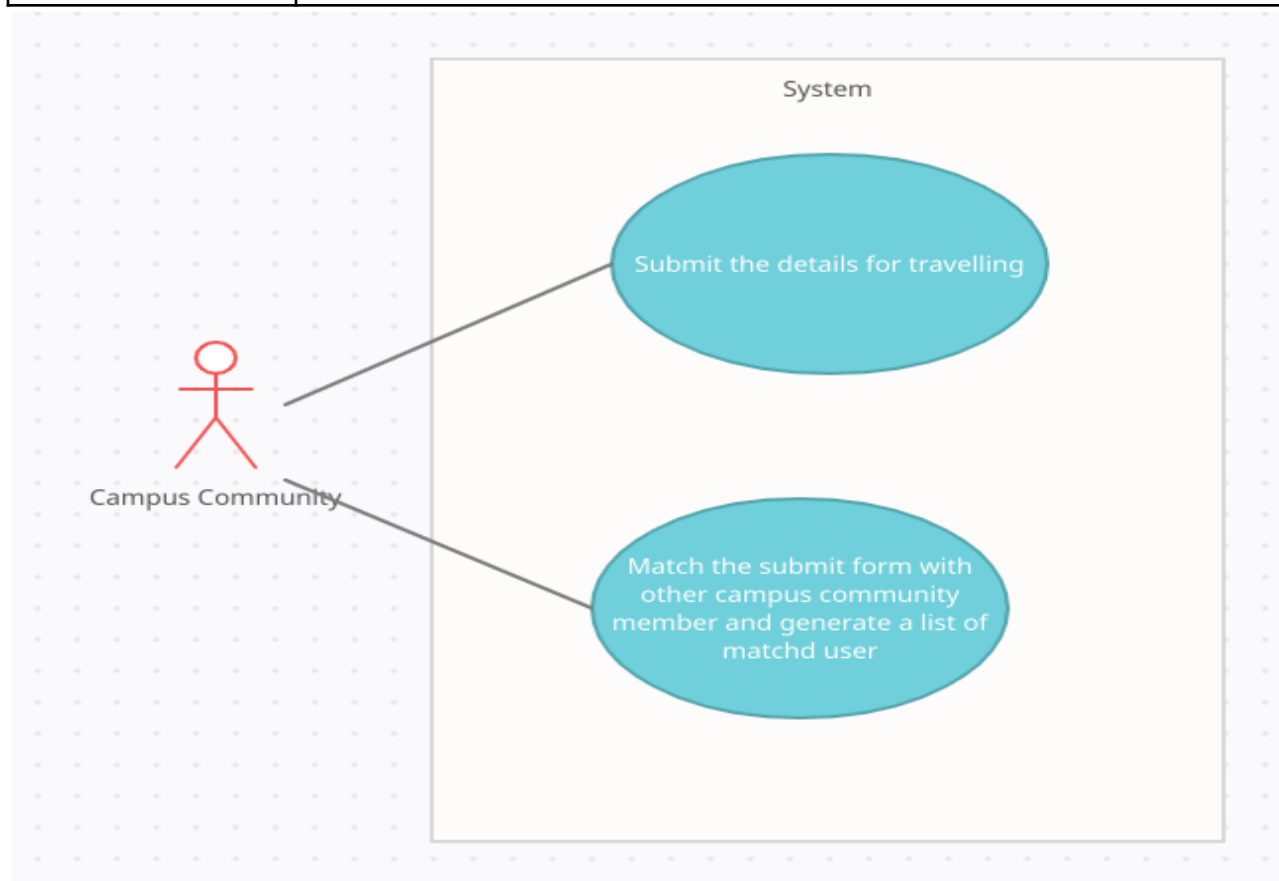
| | |
|----------------|--|
| Postconditions | The user who receives a request will be able to see the request in the request section and can accept or reject the request. |
| Actor(s) | Sending request - Campus Community Accepting/Rejecting request - Users |
| Exception | None |
| Includes | U2, U5 |



3.3.7 Travel detail and Match

| | |
|----------|---|
| Use case | U7 |
| Author | Rishabh Mukati |
| Purpose | To fill the details about travel and match with other users |

| | |
|--------------------------|---|
| Requirement Traceability | The User has access to the login page and has login details and also has access to the search section. |
| Priority | High |
| Preconditions | Users should be registered members and logged into the system and have access to the Travel details and matches . |
| Postconditions | The travel details will be stored in the database. The user will be matched with another user and the list of match users will appear. |
| Actor(s) | Campus Community |
| Exception | None |
| Includes | U2 |



4 Other Non-functional Requirements

4.1 Performance Requirements

- A stable connection is recommended for proper use of the website.
- The system shall save the user information (for further logins) of around 3 thousand users.
- The system must be able to function with a maximum of 100 simultaneous logins.

4.2 Safety and Security Requirements

- **Safety:-**
 - The users are required to fill in the correct details if they don't do that it will create problems.
 - The data of all the users must be kept confidential.
- **Security:-**
 - The system will have a login system (Username & Password) to prevent unauthorized users from accessing/damaging the data or the website.
 - Only the admin of the system has the access to update and delete the data from the user database.

4.3 Software Quality Attributes

- **Flexibility:**

The user gets an option to reset the password, if the user forgets his/her password, the password gets sent to his registered email-id.
- **Comfortability**

The system has a chat feature, making it comfortable for the users to interact within the system. Thus, making them familiar with the environment of the user interface while chatting with their pals.
- **Availability:**

The system could be accessed from PC's, laptops and mobiles. Moreover, the system shall run on every browser (Chrome, Safari, Mozilla Firefox etc.) and operating system (Windows, Linux, Mac Mavericks etc.). The system would be made available for 24*7

• Other Requirements

<This section is **Optional**. Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

Appendix A – Data Dictionary

The list of various actors and their attributes like Student, Cab, Driver and their name, age, etc. The details of the section will be updated in the future once the outline for the class diagram is prepared.

Appendix B - Group Log

<Please include here all the minutes from your group meetings, your group activities, and any other relevant information that will assist in determining the effort put forth to produce this document>

| DATE | TIME | Discussion |
|--------------|---------------------|--|
| 27th January | 5.00 pm - 5.20 pm | Tell every member to read the SRS document and decide to meet on 28th Jan at 4.00 pm. |
| 28th January | 4.00 pm - 4.40 pm | In zoom meet, discuss SRS among us, clear some doubts, and discuss unclear doubts with TA. |
| 29th January | 12.00 pm - 12.45 pm | Discuss whether to register drivers as a user or not, then TA recommended adding new features like chat messaging and making the system more user-friendly. Make a discord group for instant doubt solving with TA and members. |
| 29th January | 4.00 pm - 5.10 pm | Discussion on zoom meet regarding adding new features and deleting some features. Also, divide work and decide to meet on the next day. |
| 30th January | 11.00 am - 12.30 pm | Evaluate work among us and help each other where anybody is stuck. |
| 30th January | 4.00 pm - 4.40 pm | Discuss about use cases and other minor errors |
| 31st January | 1.00 pm - 1.45 pm | Finalise document and make log entry, fix minor error |
| 1st January | 2.30 pm - 4.00 | Document reviewed by TA |

| | | |
|--|--|---------------------------------|
| | | and make changes as he suggests |
|--|--|---------------------------------|