



USER REQUIREMENTS DOCUMENT

COMMUTE BUDDY

TEAM ALWAYS IN VOGUE

Team Members:

Aastha Kedia | Maneesh Sharma
Rahul Goyal | Sarvesh Agrawal

Product Validation & Product Definition

Prof. Tony Wasserman

Faculty, CMU-SV

Description of the system in context:

Our solution, Commute Buddy is a subscription based college SSO ride sharing application, where only students and staff with College SSO will be able to sign up to the app. And have individual spaces (For example, like workspaces on Slack). Commute Buddy has the signed in users where people can put in their travel schedule, and can see other's travel schedule and can sign up for traveling with them. Since it is college SSO based, users have information about who they are traveling with from the college database. And once the 'ride-giver' accepts to give a ride to someone in need, the system logs it and puts it in the scheduled rides section. Additional security features like SOS, Sharing ride information and other basic features are implemented.

Benefits of our solution to the problem:

- Students don't have to worry about commute and safety
- Lesser cars on the road
- Students get to interact and bond with lot more people from the same university/ course
- Builds up the community at campus

Enumeration of key users and other product stakeholders:

1. Development Teams: Software developers, Q&A Engineers and Systems Engineers. URD is crucial for them to know what is to be built and verified
2. UI/UX designers: They build the User Interface designs to visualize what is being asked in the URD
3. Business Analysts: URD helps Business Analysts understand the application in levels that are deeper than normal understanding and Business Analysts understand the business needs and evaluate the processes that are in place
4. Product Owners / Program Managers: Product owners are responsible for articulating requirements and extract value out of the development team . Building Backlogs and prioritizing requirements is a primary responsibility of this stakeholder

Enumeration of possible use cases which correspond to “basic-level” (or “sea-level” to use Cockburn’s term) goals for each key user: user stories for each functionality

User Stories for Riders

EPIC	USER STORIES	P	SP
Registration and Authentication	As a rider, I want to sign up to the application using my college SSO so that I can avail the ride sharing service	H	3
	As a rider, I want to get authenticated from the commute buddy team so that I can be a verified member on the application	H	3
	As a rider, I want to update my profile so that I can update my contact and address information	M	2
Authorization	As a rider, I want to authorize the application to access my location, so that I can get picked up and dropped nearby	H	3
Scheduling	As a rider, I want to view my travel time, so that I can ensure I'll be on time	M	2
	As a rider, I want to schedule my rides in advanced so that I have a ride confirmation	H	3
	As a rider, I want to see the car owner's calendar/schedule so that I can request for rides in advanced accordingly	H	4
Location (Pickup and Drop)	As a rider, I want to enter my pickup location, so that I can ask for ride according to the rider's availability	H	3
	As a rider, I want to enter my destination, so that I can be dropped at my desired location	H	3
Map (Route)	As a rider, I want to view my ride on a map, so that I can follow my drivers route	M	4
	As a rider, I need to know the exact location of transportation options (stations, parking locations, etc.) relative to my	M	4

User Requirements Document - Always In Vogue

	location		
	As a rider, I need to know how to connect to another travel mode if I need to	H	4
Rating	As a rider, I want to rate my ride, so that I can provide feedback on my experience	H	2
	As a rider I want to filter car owners based on rating, so that I commute with car owners with high ratings	M	3
Payment	As a rider, I want a secure payment channel, so that I can pay	H	4
	As a rider, I want to receive confirmation of payment, so that I can use the ride sharing service	H	3
	As a rider, while taking the subscription I need to know that if there are other ways to pay for fees if I don't have a credit card	M	4
Privacy	As a rider, to protect my privacy, I need to know that my private information, my travel patterns and my location data will not be used for profit or resold without my express consent	M	5
	As a rider, to protect my privacy, I need to know that the data or information I generate or is generated about me is not used for surveillance purposes without legal authorization.	M	4
Notification	As a rider, I need to know the current state of sidewalks, roads, routes, and schedules (this should include current traffic, wait times and passenger volumes and this information must be accurate and dependable)	L	4
	As a rider, I need to know If there are areas or routes where the mode or vehicle I am traveling on is prioritized or is not allowed	L	3
	As a rider, I want a ride confirmation notification so that I can get my ride details	H	3
	As a rider, I want all ride related notifications so that I am updated with the ride status	H	3

User Requirements Document - Always In Vogue

Safety	As a rider, I need to know how to (and be able to) report conditions that affect my trip	H	3
	As a rider, I need to know how to (and be able to) request for help for emergencies	H	2

User Stories for Car-Owners

EPIC	USER STORY	P	SP
Registration and Authentication	As a car owner, I want to create my account on Commute Buddy so that I can give rides to students while commuting to and from school and earn money	H	3
	As a car owner, I want to be able to use my student Identification to login so that I can be a verified member on the application	H	3
Authorization	As a car owner, I want to be able to upload my Driver's license and car details so that I can be authorized and be listed on the app	H	4
	As a rider, I want to authorize the application to access my location, so that I can get picked up and dropped nearby	H	4
Ride Scheduling	As a car owner, I want to add times at which I will be commuting to school so that other students can schedule rides with me	H	3
	As a car owner, I want to change times when my schedule changes so that the riders can be notified about changes beforehand	H	3
	As a car owner, I want to be able to configure my schedule for a week, so that I do not have to configure it everyday	M	2
	As a car owner, I want to be able to see the list of users who are sending requests for a ride, so that I can approve their ride requests	H	4

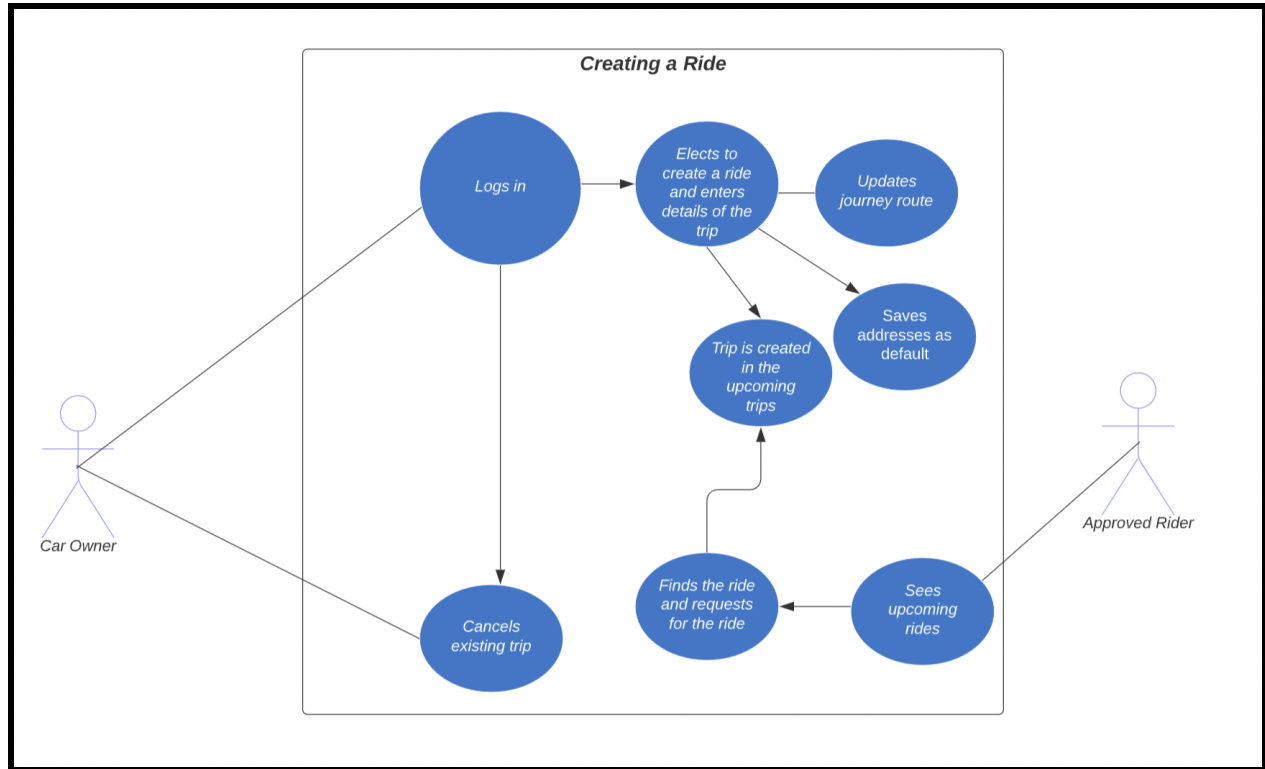
User Requirements Document - Always In Vogue

Integration	As a car owner, I want to talk to the riders on Slack so that we can communicate the details	M	3
Location (Pickup and Drop)	As a car owner, I want to add my address so that I can use that as my default address	M	3
	As a car owner, I want to select/add the source and destination so that riders can see my trip	H	4
	As a car owner, I want an intermediate stop to be added automatically as soon as I confirm a ride	H	4
Map (Route)	As a car owner, I need to know the exact drop off location (and pick up location if I am picking up someone enroute) so that I can navigate my way to that location through the maps.	H	4
	As a car owner, I need to be able to update the stops on maps which are visible to all the accepted riders so that they know how the trip is planned.	H	4
Rating	As a car owner, I want to have a rating to my profile, so that riders feel safe and comfortable commuting with me	H	2
	As a car owner, I want to be able to see other riders' ratings in the requests page so that I can make an informed decision before accepting their requests	M	3
Payment	As a car owner, I want to be able to save my bank account information securely, to be able to receive payments for the rides I am providing.	H	4
	As a car owner, I want to receive confirmation that the trip has been paid for before providing the service to the riders.	M	3
	As a car owner, I want to be able to see the history of payments made to me, so that I can keep track of all the transactions.	L	2
Privacy	As a car owner, to protect my privacy, I need to know that my private information, my travel patterns and my location data will not be used for profit or resold without my express	H	5

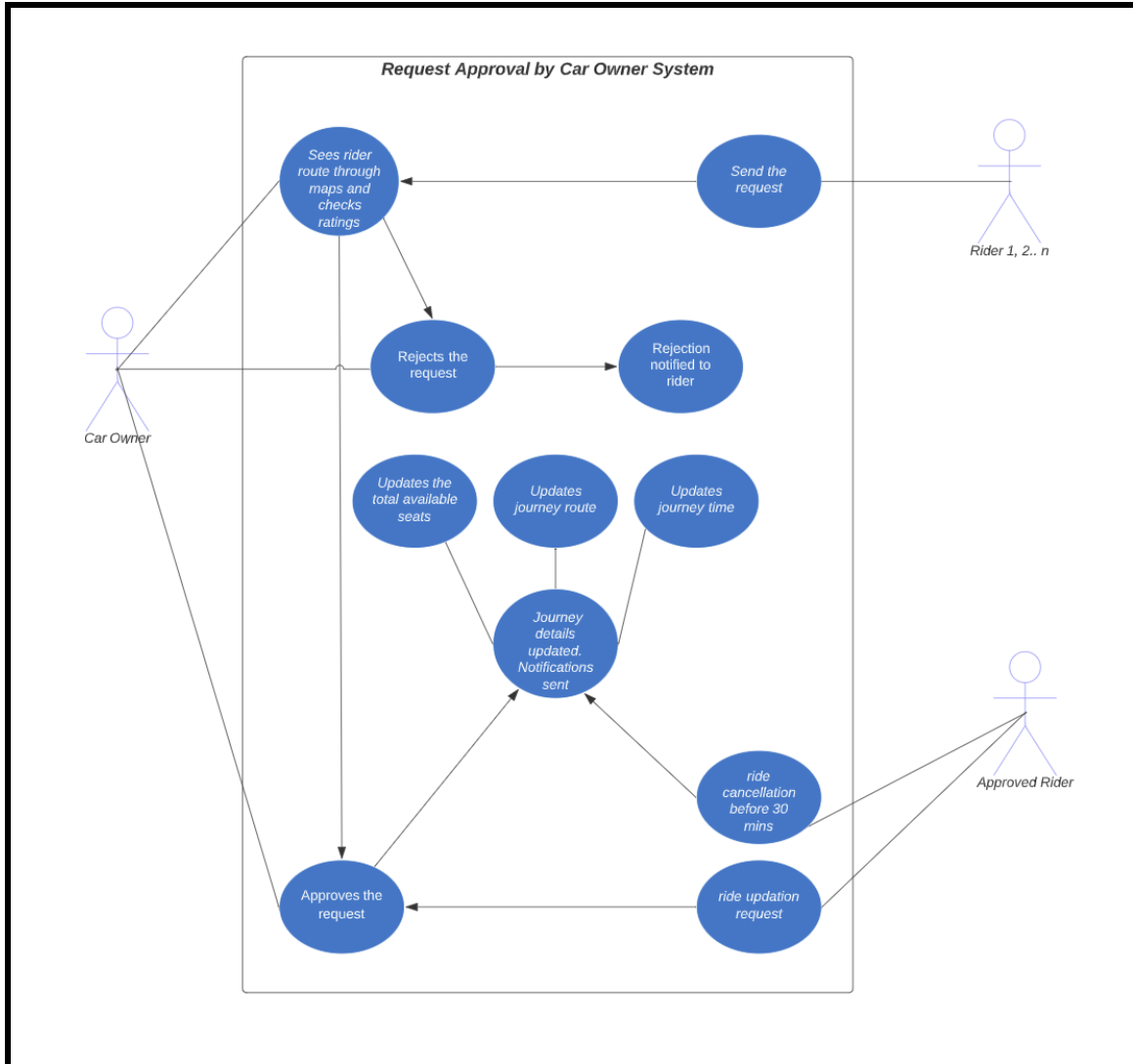
User Requirements Document - Always In Vogue

	consent.		
	As a car owner, to protect my privacy, I need to know that the data or information I generate (like driving license or other ID forms) or is generated about me is not used for surveillance purposes without legal authorization.	H	4
Notification	As a car owner, I need to know when a rider has booked a service with me so that I can accept the request and be ready to provide that service.	H	4
	As a car owner, I need to know when a rider cancels the trip or alters/updates the travel details so that I can accept the changes and plan the trip accordingly.	H	3
	As a car owner, I need to set alerts about a ride which has been booked with me and is very much outside my travel route, so that I can reject such bookings.	H	3
	As a car owner, I need to know about the ratings a rider is providing me so that I can maintain a good status quo.	M	2
	As a car owner, I need to know the current state of sidewalks, roads, routes, and schedules on my route so that I can plan my trip and update it accordingly.	L	4

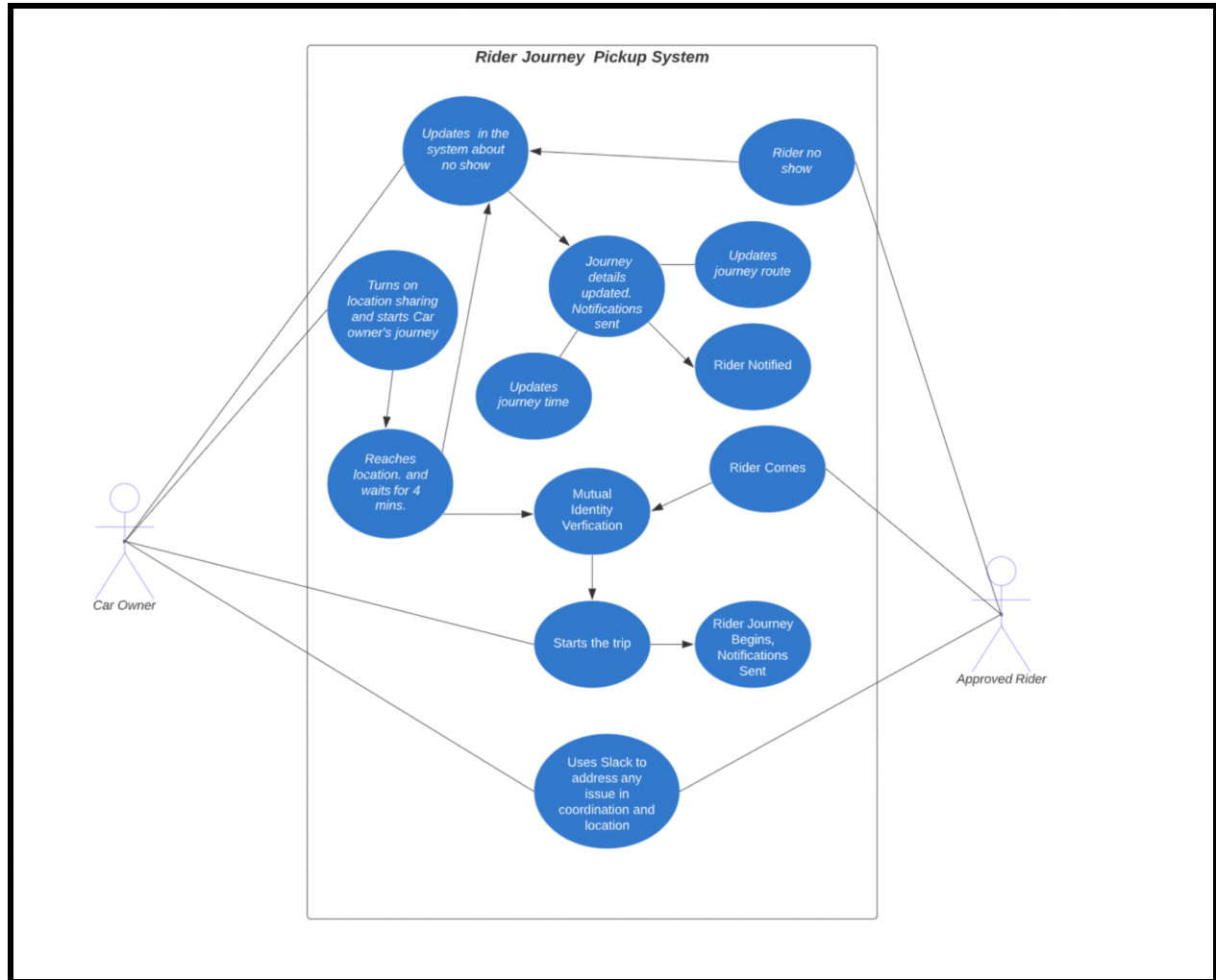
User Requirements Document - Always In Vogue

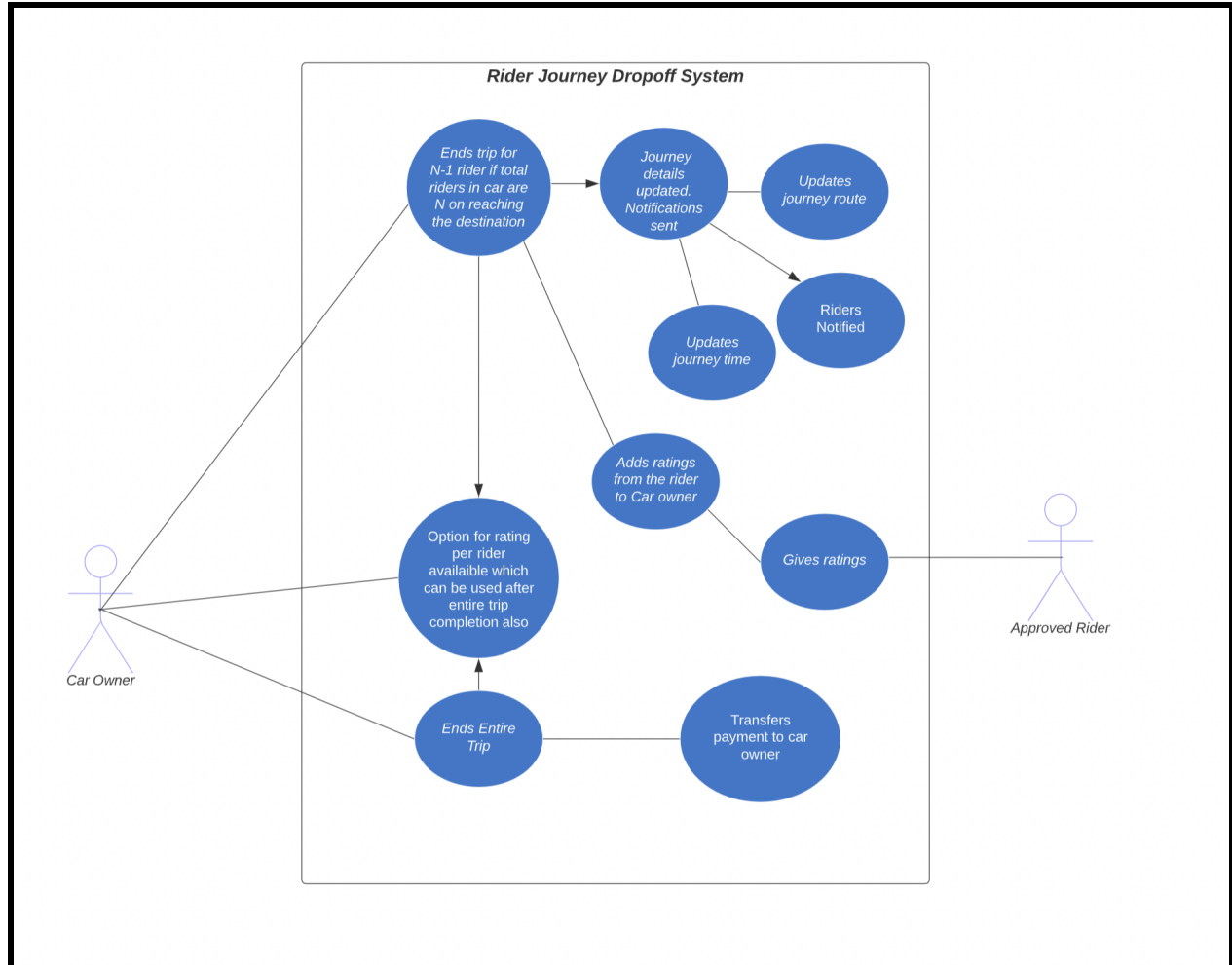


User Requirements Document - Always In Vogue



User Requirements Document - Always In Vogue





USE CASE FOR RIDERS:

1. Use-Case Specification: Registration and Authentication

- I. Participating Actors: This use case is intended for those who are registering themselves for the first time in the platform
- II. Brief Description: This use case allows the first time users to sign up to the application using their college SSO credentials and fill in their personal details.

Users will be asked to fill in the below mentioned information to create an account and get authenticated by the admin team upon verification:

- A. Full Name

User Requirements Document - Always In Vogue

- B. University Name
 - C. University ID
 - D. E-mail ID
 - E. Password
 - F. Rider or Car Owner
 - G. Document upload field to upload the university Id card, driving license etc
 - H. Address
 - I. Contact Information
 - J. Emergency Contact
- III. Assumption:
- A. The user is a first time user
 - B. The user belongs to some educational institution
- IV. Flow of Events:
- A. Basic Flow
 1. First time user opens the application and selects sign up option
 2. The user is asked to input various information fields as mentioned above
 3. Once the user inputs all the desired information, the application shows them a pop up of successful input of data
 4. The user is put in the waitlist
 5. Application admin checks in the data input and validates it with the university admin
 6. Upon successful verification, user is given the access of the platform
 7. Once the user enters the platform, they are asked to subscribe for the services with various plans available
 8. User is able to access the riders calendar and book rides accordingly
 - B. Alternative Flows
 1. Wrong credentials - If the user enters wrong credentials then they will be unable to sign up to the platform and will be redirected to sign up again with the correct credentials as per the university database.
 2. Govt id invalid or doesn't match with the input data - If the user's govt id data doesn't match then the authentication team will reject the request of sign up and will be asked to again go through the sign up process with the correct govt id or any relevant data.
 3. Red flagged by the university admin - During the screening process, if the user is red flagged by the university then they will not be able to access the platform till university accepts their sign up request and allows them to avail the services.
- V. Business Rules:
- A. All the entries in the signup form is mandatory
 - B. User can only sign up as either a rider or a car owner

- C. Background check is mandatory to approve the users to access the platform
- D. User has to be affiliated to a university or educational institution

2. Use-Case Specification: Notifications

- I. Participating Actors: This use case is intended for both the riders and car-owners to get notifications about the ride details.
- II. Brief Description: This use case will allow users to get real time notifications of the ride status and if there are any changes in the ride request or updates from the car owners. It will help riders get updates about how long the ride will take, what is the departure and arrival time. It will also notify the car owners about the riders status and what is the best route to pick them.
- III. Assumptions:
 - A. The users are registered and logged in to the application to receive the real time updates
 - B. Users have an active internet connection
- IV. Flow of Events:
 - A. Basic Flow:
 - 1. Riders -
 - i) As the rider requests a ride and it gets accepted, the rider receives a notification with the vehicle and owner details updates
 - ii) As the ride time approaches, the rider will receive a notification with a reminder and estimated time of arrival
 - iii) During the ride, the rider will receive notification with the estimated time of travel
 - 2. Car Owners -
 - i) Owners receive notifications whenever a rider requests for a ride.
 - ii) Owners get notification about the riders pickup and drop location and the number of riders who are there in the same route.
 - iii) Riders get the notification once the ride is completed about the amount awarded for a successful ride completion.
- V. Business Rules:
 - A. Users must be logged in to receive the notifications.
 - B. Users must give access to receive notifications.
 - C. Users will receive ride details notifications at the time of booking and a reminder notification an hour before the ride.

3. Use-Case Specification: Chat feature

- I. Participating Actors: This use case is intended for both the riders and car owners to have smooth communication through third party integration.
- II. Brief Description: This feature will help riders and owners to communicate with each other through the official channel of communication in their educational institution like slack, MS-teams etc. Users can exchange their time preferences and also build a stronger community using this feature which will eventually result in stronger connections and networking opportunities.
- III. Assumption:
 - A. Users have an established communication channel in their universities.
 - B. Users are a part of the communication channel.
 - C. Users are logged in to the application.
 - D. Users have taken the application subscription.
- IV. Flow of Events:
 - A. Basic Flow
 1. Users while looking for the riders can check with the owners through a third party integrated chatting application
 2. Users will click on the chat option and will be redirected to the application
 3. Users can look for the person's profile and start a communication
 4. After having a conversation they can make the desired booking in the application as per their convenience.
 - B. Alternative Flows
 1. If the user is unable to use the chat option then they can access the contact details and mail id of the other person and reach out to confirm the ride booking or for any other communication.
- V. Business Rules:
 - A. Users must be logged in to the application and the third part communication application.
 - B. Users will have to give access to integrate the external application.
 - C. Users must adhere to the application's guidelines of communication to be able to access the features.

4. Use-Case Specification: Ride Scheduling

I. Participating Actors:

The use case is initiated by a local or an international student, whoever wants a ride from or to school

II. Brief Description:

This use case deals with the riders scheduling rides on the commute buddy application. A trip is created when a registered verified rider requests a ride or schedules it beforehand as per the week schedule in the application.

Riders will be asked to provide the following input:

- A. Source of the trip
- B. Destination of the trip
- C. Time of leaving the trip
- D. Date of the trip

III. Assumptions:

Rider is registered, logged in and is verified to be a rider.

IV. Flow of Events:

A. Basic Flow:

- 1. The use case starts when a rider elects to create a scheduling
- 2. The system loads a modal which consists of an input form
- 3. The rider enters the source address
- 4. The rider enters the destination address
- 5. The rider enters the time and date
- 6. The rider submits the details
- 7. The system sends a trip request to the car owner and once it is accepted it gets added to the upcoming trips list

B. Alternate Flow:

- 1. In step 2, the system loads the modal with predefined default information of the user if the user has default address and car and the use case continues to step 6
- 2. In step 6, the rider saves the entered details as their default trip details and the use case continues to step 7
- 3. In step 7, the user can modify the ride or opt to delete the ride

V. Business Rules:

- A. The rider has to be logged in to the system
- B. The rider has to be a verified user on the application
- C. All the entries of the input form are mandatory

5. Use-Case Specification: Privacy and safety

- I. Participating Actors: This use case is intended for all the users of the Commute Buddy application.
- II. Brief Description: This use case is very crucial from users point of view where we make sure that data remains safe and is not accessible to outside applications for any purpose and also all the users have their safety measures covered while using our application to avail the ride sharing services. Users need to update the following details before the ride:
 - A. Access to privacy and data
 - B. Emergency contact and address
 - C. Any health conditions to be taken care of
- III. Assumptions:
Users are registered, logged in and are verified to be a user.
- IV. Flow of Events:
 - A. Basic Flow:
 1. The use case starts when the users give permission to access their data.
 2. Users have the right to revoke the permission at any time.
 3. Users can check the detailed privacy policy in place.
 4. Users will be asked to avail the safety packages and updates while subscribing for the services.
 5. Users will be asked to provide an emergency contact in case of emergencies.
 - B. Alternate Flow:
 1. If the users don't give permission access to their data then they will have limited access to the functionalities which the application can offer.
 2. If the user doesn't give the emergency contact then in case of emergencies the university will be updated so as to avail any help from the existing database.
- V. Business Rules:
 - A. Users must be logged into the application.
 - B. Users will be asked for their consent and need to approve it.
 - C. Users need to update their profile and should update their health conditions.

USE CASE FOR CAR OWNERS:

1. Use-Case Specification: Car Owner Registration and Authentication

User Requirements Document - Always In Vogue

I. Participating Actors:

The use case is initiated by a car owner and involves interaction between the University system, CommuteBuddy's Registrations and Verification Department or a partnered external verification agency.

II. Brief Description:

This use case deals with verification of car owners. If a car owner is found fit and completes all requirements, then they are registered in the system.

Car owners will be asked to provide the following input:

A. Driving Related-

1. Driver's License
2. Car Ownership Documents
3. Car Insurance
4. Health Insurance
5. University ID
6. Driving Background and any incidents/accidents

B. Personal Information-

All personal information like Name, Student ID. Contact information, Address, Emergency Contact etc. will be retrieved from the University SIO Database itself. But the car owner can fill in different information for some fields like Emergency Contact.

C. Acceptance to Privacy policies and Terms and Conditions Agreement.

D. Car owners will be required to complete an online training module and submit a quiz.

III. Assumptions:

The user is a first time user.

The user belongs to some educational institution.

The institution has some SSO and SIO similar infrastructure.

IV. Flow of Events:

A. Basic Flow:

1. First time user opens the application from SIO (the name depends on university).
2. The user is asked to input various information fields- A(Driving Related), B(Personal Information) and C(Agreement to Privacy policies and T&Cs) as mentioned in the description.
3. Once the user inputs all the desired information, the application shows them a pop up of successful input of data.
4. The user is put in the waitlist.
5. CommuteBuddy's Registrations and Verification Department or partnered external verification agency checks in the Personal data input and validates it with the university admin. Car Owner has to approve this step.

User Requirements Document - Always In Vogue

6. CommuteBuddy's Registrations and Verification Department or partnered external verification agency verifies the authenticity of the Driving Documents provided and does a background check.
7. The training Module has to be completed at self pace and get acceptable results in the quiz.
8. Upon successful completion of Steps 5, 6, and 7 the car owner is given access to the platform.
9. The registration and verification is complete.
10. Once the car owner enters the platform, they are asked to subscribe for the services with various plans available.
11. The car owner selects one plan and pays for it.
12. The car owner can now use the platform.

B. Alternate Flow:

1. In step 5 or 6, if due to any reason, the request is denied, the person will be denied registration. The reasons may be that government id is invalid or input data doesn't match with the university records. Another reason can be something is flagged in background search.
2. In step 7, if the training module is not passed, the car owner can have 2 more chances with different questions. But if the car owner still fails, he will not be provided another chance for 2 months and the user will be unregistered.
3. In step 11, if the car owner does not pay for the platform, he can come back again and pay it in future.

V. Business Rules:

- A. All the entries of the input form are mandatory

2. Use-Case Specification: Creating a Ride

I. Participating Actors:

The use case is initiated by a local or an international student, whoever wants a ride from/to school

II. Brief Description:

This use case deals with the car owners creating rides on the commute buddy application. A trip is created when a registered verified driver provides his schedule or sets up his schedule for the week in the application.

Car owners will be asked to provide the following input:

- A. Source of the trip
- B. Destination of the trip
- C. Time of leaving the trip

- D. Car details (fetched from the system if user has added a default car)
- E. Number of available seats

III. Assumptions:

Car owner is registered, logged in and is verified to be a car owner

IV. Flow of Events:

A. Basic Flow:

1. The use case starts when a car owner elects to create a ride
2. The system loads a modal which consists of an input form
3. The car owner enters the source address
4. The car owner enters the destination address
5. The car owner provides his car details
6. The car owner enters the total available seats
7. The car owner submits the details
8. The system creates a trip and adds it to the upcoming trips list

B. Alternate Flow:

1. In step 2, the system loads the modal with predefined default information of the user if the user has default address and car and the use case continues to step 7
2. In step 2, the car owner elects against creating a trip and clicks cancel and the use case ends
3. In step 7, the car owner saves the entered details as their default trip details and the use case continues to step 8
4. In step 8, the user can modify the ride or opt to delete the ride

V. Business Rules:

- A. The car owner has to be logged in to the system
- B. The car owner has to be a verified user on the application
- C. All the entries of the input form are mandatory

3. Use-Case Specification: Managing a Ride

I. Participating Actors:

The use case is initiated by a car owner who has created the trip and involves interaction between riders, system and the car owners. Corner Cases will also include partnered insurance agencies or Student's health insurance, emergency services.

II. Brief Description:

This use case deals with the car owners managing the trip from accepting or rejecting

the request to dropping off the rider. This Use case has two parts- Ride Acceptance/Rejection and Actual Ride which is a multi step workflow involving maps, GPS, payments and ratings in general.

III. Assumptions:

Car owner is registered, logged in and is verified to be a car owner

IV. Flow of Events:

A. Basic Flow:

1. The use case starts after the ride is created and is visible to all the riders using the CommuteBuddy.
2. The car owner receives a notification when the rider is willing to book the ride.
3. The car owner can click on the profile of the person and check the ratings the rider has. He can also see the route requested by the rider on the map along his previous map and see how much off route he will have to go and the stops he will have to make.
4. The car owner accepts or rejects the rider's request.
5. On rejection, nothing happens on the car owner's side. On approval, system will update-
 - a) The path taken by the rider so that the other riders will be able to the updated journey route.
 - b) The available seats in the car.
 - c) The time for the journey (through traffic, approx wait times etc.).
6. As more riders are approved by the car owner, the system updates the above details and all approved riders and the car owner are notified of the same along with car and car rider details.
7. The car owner will allow and start the location (GPS) sharing to the app at the start of the journey so that the riders can track the car and get ready to reach the first pickup spot.
8. The car owner will wait for 4 minutes which will be started by the system on the pick up point. Any discrepancies due to errors on the GPS or maps can be discussed on Slack.
9. If the rider comes, the car owner and riders show their identity and then the journey will go as planned.
10. At drop-off, the car owner ends the journey for a particular rider. The car owner will have the journey for that rider marked complete. He can provide rating at that time or after completing the whole ride or in future.
11. The car owner will receive the payment at the end of the entire journey.

B. Alternate Flow:

1. In step 5 or step 6, the car owner can cancel the service entirely or for a particular rider. In that case, the car owner has to provide a reason. If only

a rider is removed, the journey is updated accordingly. If the whole trip is canceled, the riders are notified and the use case ends.

2. In step 5 or step 6, an approved rider can ask for an update in pickup or drop off location. The car owner is notified and can check the path he will have to take for the updated request. If he feels it will not be possible or will be problematic for other riders, he can reject the request. If he approves the request, the journey is updated accordingly and the approved riders are notified of the same.
3. In step 5 or step 6, an approved rider can cancel the journey. In this case, the journey is updated accordingly and the approved riders are notified of the same.
4. In step 9, the car owner will leave if the rider is not on time and updates this on the application. The system will update the journey details as mentioned in step 5 and notifications will be sent to the approved riders and car owner.
5. In step 11, if there is some issue with the bank account details of the car owner, he will be asked by the system to check and add the details again or enter new details. The payment will be sent after that.

V. Business Rules:

- A. The car owner has to be logged in to the system.
- B. The car owner has to be a verified user on the application.
- C. As soon as the car owner reaches the pickup location, the system will start the 4 minute clock.
- D. If any cancellation happens from car owners, valid reasons are mandatory. Numerous cancellations on riders will be flagged.
- E. The rider can cancel only until 30 mins before the journey start time.

Enumeration of key nonfunctional requirements, including a product-specific definition of each requirement and how each will be measured or observed with respect to the product

Accessibility:

- Users should be able to reach their profile data from any page within three clicks. Our key focus will be to enhance the user experience and provide the functionalities at ease.

Security:

User Requirements Document - Always In Vogue

- The system should require users to enter their passwords every 60 days and will follow a dual authentication SSO based login process. We aim to keep the application very secured and safe since it has a lot of users' personal information.

Scalability:

- The system must accommodate a minimum of five thousand concurrent users. As we scale up the features and functionalities, we aim to increase the traffic on the application without any glitches and dysfunctionalities.

Performance:

- All the pages should be able to load within two seconds. This will be a very crucial measure for us to make sure user experience remains intact and the application is very responsive.

Compatibility:

- The iOS/ android application must support devices running on any OS versions. Since we plan to build a mobile application, it is very important to integrate it with all the operating systems to increase the user base.

Reliability:

- The system must perform without failure in 95 percent of use cases during a month. During our MVP, we aim to achieve 95% availability and later we will aim to achieve high-high availability and should run 99% of the use cases without failure.

Maintainability:

- The mean time to restore the system (MTTRS) following a system failure must not be greater than 10 minutes. MTTRS includes all corrective maintenance time and delay time.

User Requirements Document - Always In Vogue

Availability:

- The app must be available to the users 99.98 percent of the time every month throughout the day. In the case of unplanned system downtime, all features will be available again after one working day.

Data integrity:

- The system shall maintain data integrity by keeping backups of all updates to the database for every record transaction and will be encrypted so that the user data remains safe.

Modularity:

- We will use a micro-service architecture to limit the downtime of functionalities at times of failure or updates.

Interoperability:

- A key aspect of our application is the ability to integrate with third party applications to provide the user with a wide range of critical decision making information.

Corner Cases Addressed

Insurance: Since all of the users on our site are associated with educational institutions, we want to take advantage of their health insurance program. In the long run, we'll collaborate with healthcare organizations to offer more services to individuals connected to our app. We'll also make sure that throughout the registration process, the verification stage requests that car owners update their government-issued proof of car insurance. Once the user database increases, we also plan to take a minimal insurance fee linked to the subscription fee to improve the coverage and facilities linked to the insurance plans.

Different campus size and location: We will have precise drop-off alternatives in our application because every college will have various regulations and campus layouts, and whenever the vehicle owners update the drop-off location, passengers will be matched to the car owners who are headed to their closest drop spot. We want to make it easier for riders to go from their homes to university, however users may utilize the chat feature to get around the precise drop-off location on campus. We'll use the main automobile parking space on campus as a normal drop-off location. If cars are not permitted on campus, we will urge drivers to utilize the drop off location, which is the closest parking area.

Safety measures: In the event of an emergency with varying levels of severity, users will have access to an SOS button in the application. When a user presses the SOS button, their emergency contacts are called, and if there is no response, the college administrator is contacted to discover alternative contacts from the school's database. If no one answers, the



situation will escalate to the highest level of severity, and the police department will be notified of the situation's location and present state.

Users will also be able to share their current drive status with their friends and family to let them track the journey.

Driver's skill sets: To be qualified for safe driving, all vehicle owners who register with the application must update their driver's license and other official documentation. We also have a rating system where users may leave comments. This system will be regularly monitored to provide ratings to both parties and determine how much they can use the platform. We will contact car owners to encourage them to finish certain driving courses in order to continue accessing the services in the case that we receive regular input on how to improve driving abilities.

Appendix

Tool Analysis findings:

- User story: Lucid Charts
- No code/ low code: Webflow.io
- UI Mockups: Figma
- Storyboards: storyboardthat.com

Validation Interviews:

Validation Interview 1 (Sarvesh Agrawal)

Interviewee: Harshal Gautam

Interviewee Occupation: Product Manager @ Microsoft

Location: Seattle, WA

1. I and my team are working on a project as a part of my Product Definition and Validation course at Carnegie Mellon University. I would like to share it with you and pick your brain on it. Is it fine?

Ans. Yes, for sure. Go ahead.

2. We came across students' pain-point regarding ways to travel to the University. In response to that we plan on making Commute Buddy, a college SSO-based application, where only students and staff logged with the College SSO will be able to sign up for the app and have individual spaces (For example, like workspaces on Slack). Commute Buddy is going to have signed-in users where people can put in their travel plans and can see others' travel schedules and can sign up for travel with them. Since it is college SSO-based, users will have verified information about whom they are traveling with from the college database. And once the 'ride-giver' accepts to give a ride to someone in need of a ride, the system will log it and will put it in the scheduled rides section. Additional security features like SOS and sharing ride information among others will be implemented. Does this sound good to you?

Ans. I hear you. The overview you gave is great, but I would like more information. My first question is can you tell me something more about this being a pain point?

3. So, I and my team came across this pain point from talking and interviewing our peers in our University which is CMU Silicon Valley Campus, some other friends in other Universities like Duke, ASU, NYU, and other contacts. Weather issues, different stops, university timings, and time management are some of the concerns for traveling. We are not saying that the best transportation would be through our platform, it is a communal and feasible way to travel. Also, many students cannot afford Commercial ride-sharing services like Uber and lyft on a daily basis when they are getting an education on loan.

User Requirements Document - Always In Vogue

Ans. Oh, this was what was going to be my next question. I understand it will be some sort of carpooling but how is it different from Uber, Lyft, etc.? Are you not going to charge? Can you please explain this?

4. Yes, we are planning a Semester based subscription plan at minimal costs for access to the platform. And then the riders will pay the drivers/owners a fee that would be way less than these existing platforms. We would have to figure out how to reach that price as we plan it to be affordable for students but keep the drivers motivated to use the application. Another way we are thinking to go about it is letting the drivers set the prices, but we are aware that then we will have to come up with a mechanism for the prices to be monitored and in check because it will make no sense if they are keeping the commercial prices and providing no real benefit to students. Secondly, the trust and safety factors of the driver being from the University community and peer-to-peer two-way feedback comes as a differentiation here.

Ans. This is interesting. I like the factor of prices being balanced to be affordable by students yet sufficient to keep the student motivated. And I am happy that you are considering the angle of the driver selecting the prices and thinking about that factor too which is different from the existing services. I get the goal, value, and differentiation from the big commercial services here. Although I think that carpooling services are out of scope in your project, I would also like you to consider local carpooling services if any. My next question is what about insurance which is a big factor when it comes to this service?

5. Touching upon the point of carpooling services, yes, one of our differentiation factors is University communal trust, but still, we talked to students if they know about any existing services for them. We also searched and didn't get any great results for university transport. Coming to your question about insurance, we are yet to figure that out. On one hand, we were discussing that since this is a service for feasibility, insurance is out of the scope of our project, but on the other hand, as you mentioned insurance plays a big part in this industry. Do you have any ideas or tips on how we can approach this?

Ans. I feel that for it to become a successful service, the safety of riders, cars, and drivers must be considered. Riders can damage the owner's car, riders or drivers can be irresponsible and there are other scenarios also. So, you will have to come up with some solutions for these scenarios also along with your main idea.

6. Got it. I will bring this up with the team and we will brainstorm on this point. Thank you for your input on insurance. Are there any other inputs you would want to give?

Ans. I think you are doing good. The idea has its strength, and you are definitely trying to solve a pain point here. There are other features that come to mind like wait time for drivers and riders and payments which you can get from Uber and others but that is not of very much importance here. Insurance was a major direction I thought of. And other than that, you are going in the right direction.

Validation Interview 2 (Rahul Goyal)

Previous Interview Context:

During the user research interview with Anamika, we tried understanding what all pain points she had in terms of commuting from home to campus as she just shifted to the US and how was she managing the commute currently. She emphasized on a few pointers like safe and reliable commute options which can be economical as she needs to use the service on a daily basis. She also mentioned that currently for normal timings she walks for close to 35 minutes to reach the campus or has to plan her schedule according to the VTA bus timings. For odd hours she mostly relies on people from the same class (if any) else she takes the cab which is putting a lot of load on her pockets.

Validation Interview Script (Edited):

Rahul: Hi Anamika, thank you so much for taking out the time again to help us with the interview. In this interview the primary focus is to present to you the solution which we have built taking into account the pain points you mentioned in the last interview. We would highly appreciate your feedback on the same and if you can suggest a few improvement areas which we might have missed upon.

Anamika: Hi Rahul, I'm more than happy to help you with the interview and I'm really looking forward to giving my inputs to help you build something which can help us with this challenging commute problem.

Rahul: So Anamika, last time you mentioned a few pain points in the current commute options like long walking time, planning day according to the VTA schedule or taking cab services which are not very economical and safety is always a concern. After a lot of brainstorming and understanding the perspective of other students from our campus we have come up with a SSO based student commute service model.

Anamika: That sounds interesting, Rahul!

Rahul: Yeah, so we are envisioning the whole solution to work. We will build a platform where students and faculties who have a car with them and use it for commute purposes can register to the platform to help those who want to do carpooling. The other end of the platform will be the users who want to avail this service and are in need of a better commute option.

Anamika: Oh this sounds like Uber, Lyft. But I find these options very costly though because I need to go for classes daily and the financial situation is a huge constraint for me here. Are you planning on doing something different here?

Rahul: Yeah, so keeping all that in mind we have come up with a subscription based carpooling service which will be SSO based meaning that it is meant only for CMU community and will be very cheap when compared to the outside cab services and will be a very safe and secure

User Requirements Document - Always In Vogue

options when you need to use it at odd times. Plus we will also validate the riders and owners in our platform with all the documentation to avoid any challenges later.

Anamika: This sounds nice! It feels like you are automating what I generally do on a daily basis, roaming around and asking if someone is going on that route so that I can ask for a ride. But my challenge is how can I see when a car owner is planning to travel because I can't rely on them for last minute updates.

Rahul: I totally understand your concern here, Anamika! In our solution we are planning to integrate the CMU calendar so that riders can see when different owners have classes so that they can reach out to the owners accordingly and also the owners can manually update the timings on which they will be traveling so that the riders can send requests accordingly. Plus we also are planning to push a lot of notifications for the users so that they are updated regularly and plan the day accordingly.

Anamika: Oh this solves one of my major concerns, but I have one more issue of finding the right communication channel to find out who the owner is because currently at times it becomes very awkward to walk up to someone and ask them to drop you. I tend to use my friends' references if I don't know someone.

Rahul: We have got you covered, Anamika! We are planning to integrate our app with a chat option linked with Slack where we can easily find all our fellow mates. Plus you don't have to feel shy now because it's not just a favor anymore, they are also incentivized to help people and make money in return.

Anamika: Totally makes sense, Rahul. Looks like you have thought it through very well. I'm really excited to see how your team will execute the whole plan.

Rahul: We are thrilled that you liked the idea and really hope that it solves all your concerns and provides an efficient way to commute on a regular basis. Also, we are working on the business model to figure out the pricing strategy for the same. We will also integrate a SOS option, in case of emergency because we want our community's security and safety first.

Anamika: SOS looks like a great value addition.

Rahul: We are also planning to have a peer review mechanism set up in the app so that the owners and riders with higher ratings will have a high probability of getting a match and so we can ensure that the riders and owners respects each other and build that bond and connections in the longer run to make this model a success.

Anamika: Wow, this feature really helps in Uber too where I prefer to have high rated drivers and I recently observed that it works the opposite way too where riders are preferred on the basis of the ratings they have received from the drivers.

Rahul: Yes, that's our inspiration model. So, overall this is the final solution we have come up with to solve the current challenges in the system. I would highly appreciate it if you can give us some feedback if you have any and also help us test the application once we develop it.

Anamika: I think your team has thought of the product really well, it does cater to all the pain points I face regularly. If I have something I'll definitely get back to you on this and feel free to reach out if you have any more clarifying questions. I'm always there to help.

Rahul: Thank you so much Anamika! Really appreciate all your contribution and time commitment. Have a great day ahead.

Validation Interview 3 (Aastha Kedia) -

The transcript below is an edited version with introductory greetings and other unrelated parts not included as mentioned in the prompt.

A quick recap of the last interview

Niharika lives in San Jose and for most of her travels to CMU, she depends on VTA buses, light rail, and Caltrain. Since the busses coming to CMU run Monday through Friday from 9 AM to 6 PM, on the days she has classes in the late evenings (post 5:30 PM), she takes an Uber or Lyft back home and it is burning a hole in her pocket or someone from her cohort who owns a car and is willing to drop her drops her to nearest transit center; for taking a Caltrain back. She shifted to the States from India in the month of August for her Masters degree and is living with an undergraduate college friend who studies at SJSU. She has talked about feeling unsafe sometimes leaving for home around 9 PM alone due to the unfamiliarity of streets and strangers driving.

Moving on to explaining a product idea

We plan on making Commute Buddy - a college SSO-based application, where only students and staff with College SSO will be able to sign up for the app. Since it is college SSO-based, users will have information about who they are traveling with from the college database. And once the 'ride-giver' accepts to give a ride to someone in need, the system will log it and will put it in the scheduled rides section.

Aastha: So Niharika, what are your first thoughts on the plan of attack?

Niharika: Firstly, I like that it is SSO-based so safety concerns are kind of eliminated. And, because I have a weekly once a late evening class, I would like there could be a way to travel back home with someone from college, whom I could potentially know or maybe at least trust. My general mode of transport is Uber or Lyft, but there is always a safety factor with them. So in that aspect, I am inclined.

Aastha: You are correct! Since it is SSO based, it is meant for the CMU community only, so not anyone and everyone would be eligible to sign up to be a ride-giver or asker. It is community-driven, safe, and secure.

Niharika: But I have a question, how would I know someone's schedule and request a ride? Will I be the one to put out a request and then a ride-giver would see and according to his/her calendar accept?

Aastha: To answer your question, we will have the option to integrate the calendar, this way, the ride-givers, and the riders would be able to put up their class schedules, and also manually put up the travel times, perhaps giving better visibility as to who is available to drop and when. So the rider would search for a time range and whichever ride-giver falls in that range, will be visible to the rider as an option to ask for a ride. The app will be enabled with notifications as well so that updates are sent to the riders and ride-givers.

Niharika: Sounds good. What about the ride-giver, why would they give us or any rider a ride? This would just add to his/her time to reach home! I feel like once I would know that some XYZ person also has a class ending when mine does and they live near me or can drop me - why would I go through the hassle of booking through the app?

Aastha: That is true, and perhaps we have decided to have a small fee that riders will need to pay (which would be lower than Uber or Lyft) so that the ride-givers are motivated to give rides and earn money alongside.

Niharika: Ok that makes sense! So basically it is like an Uber or Lyft of, by, and for the CMU community, and perhaps it will have a chat interface too.

Aastha: Yes, there would be Slack Integration, along with Google or Apple maps integration and a payment portal! The riders and rider-givers will chat through the Slack app, select a location on the map, and pay via the app. Also, we are planning to add an SOS button to the interface too so that we can be extra secure and safe. We are trying to eliminate the need to juggle emails for planning or go through awkward interactions asking for a ride.

Niharika: Nice thing! I like the idea of an SOS button.

Aastha: Okay, I would like to know what you like most about our product idea?

Niharika: The safety - I mean, late-night travel in a new country can be a little tricky and if I can find someone, who is from the college, to drop me home from college - Yeah!

Aastha: And, What do you like least about our product idea?

Niharika: Right now, all I think of is this is so much like Uber but with less transportation cost and more reliability and safety. So there isn't a feature that I liked less than the other!

Aastha: That sounds positive! By the way, we will also have a feedback option for ride-givers and riders, so that we know everything is going well. And users have the option of expressing gratitude, appreciation, or despair/complaints.

Niharika: That is a good-to-have feature, love the overall idea! I hope this comes into action soon, as I am struggling with finding a safe and comfortable means of going back home late in the evenings. I would definitely suggest my friends here at CMU to use it. It is not only safety that I am looking at, it will help me build connections with the college community.

Aastha: Thank you, Niharika. This is one of our value propositions too, networking. You will meet new people from the college community and will have the option to build connections and expand your network. Yeah. Do you have any more questions for me?

Niharika: Nope, just that I am really excited to use this!

Aastha: Thank you for your time and patience, Niharika. Have a good day.

Validation Interview 4-(Maneesh Sharma)

Date: 11/14/2022

Interviewee: Eva

Interviewer: Maneesh

Edited Transcript:

Q: Hi Eva, thank you for taking time for a short follow-up interview! As a car owner and someone who drives to school everyday, what do you think of the Storyboard?

I think it is pretty intuitive and explains to me what I will be able to achieve out of using Commute Buddy. However, I want maybe a little more explanation.

Q: Can you elaborate on the kind of explanation you seek?

For instance, I want to know and see maybe some of the interfaces where I would be able to sign up for rides and how would I get to know if the person I am traveling with, is safe or not.

Q: So that is part of the prototype and storyboards are usually to represent experiences of a user. I would share the prototype with you as well. Do you think this storyboard addresses your pain points?

Well, I have been looking for people to join me while I go to school. Having got a new car very recently, I would ideally like for people to join me while I am commuting as it saves me gas money and also, I have had few of my female friends asking me to drop them home in the evenings because it has become too dark these days. Also, there are days I do not have classes but I have to come to campus just to drop a few of my friends. So I think if there was a systematic way of addressing this, it would be very beneficial and yes, to answer your question, the storyboard strip addresses my pain points. But I had a question- I would sometimes maybe feel like traveling with new people instead of traveling with the same people. Is there a way you can randomize traveling people?

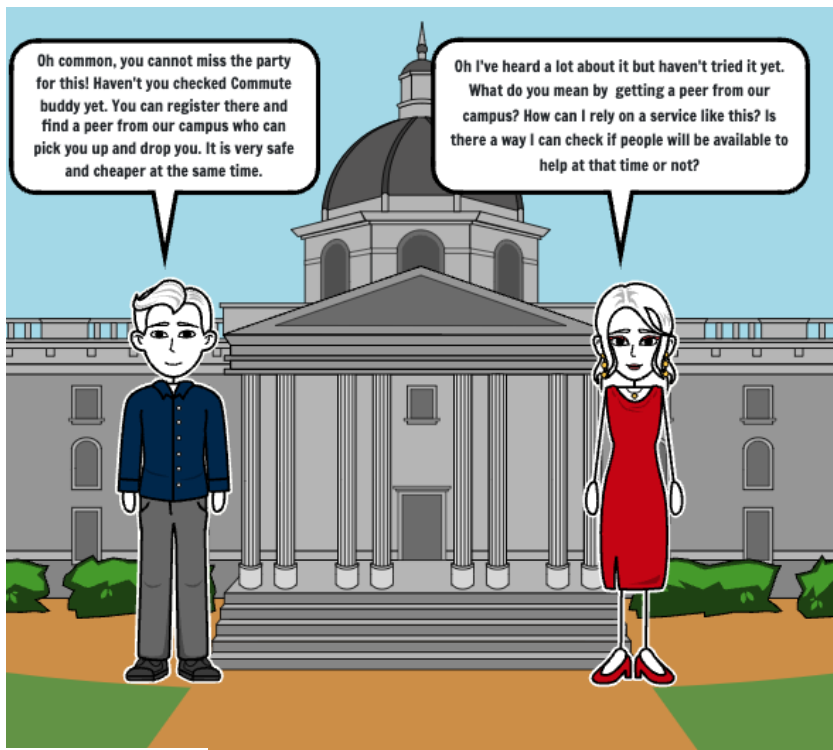
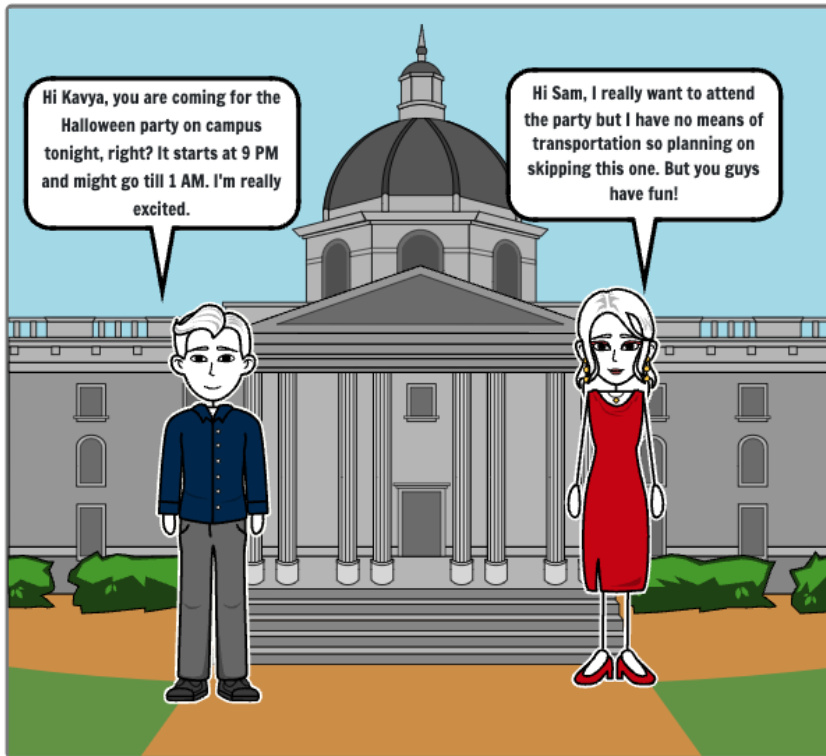
Q: That is a good question- We are implementing the MVP in a way that you would be able to pick the people you want to travel with. Maybe we can later build in an algorithm that will let ride requests be suggested randomly. Thank you for taking your time for this interview on a short notice.

No worries, Maneesh! I look forward to seeing the prototype.

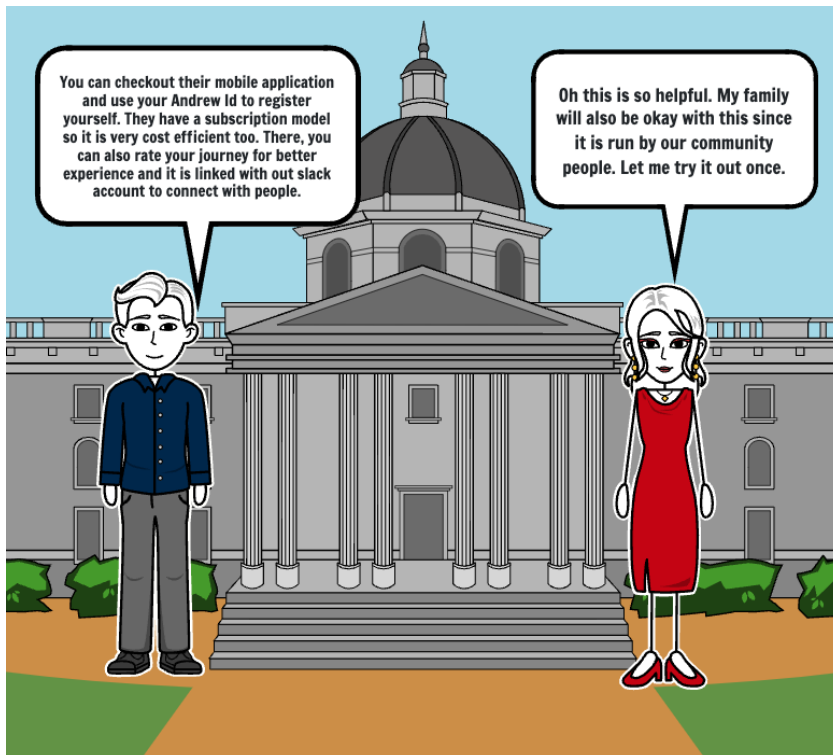
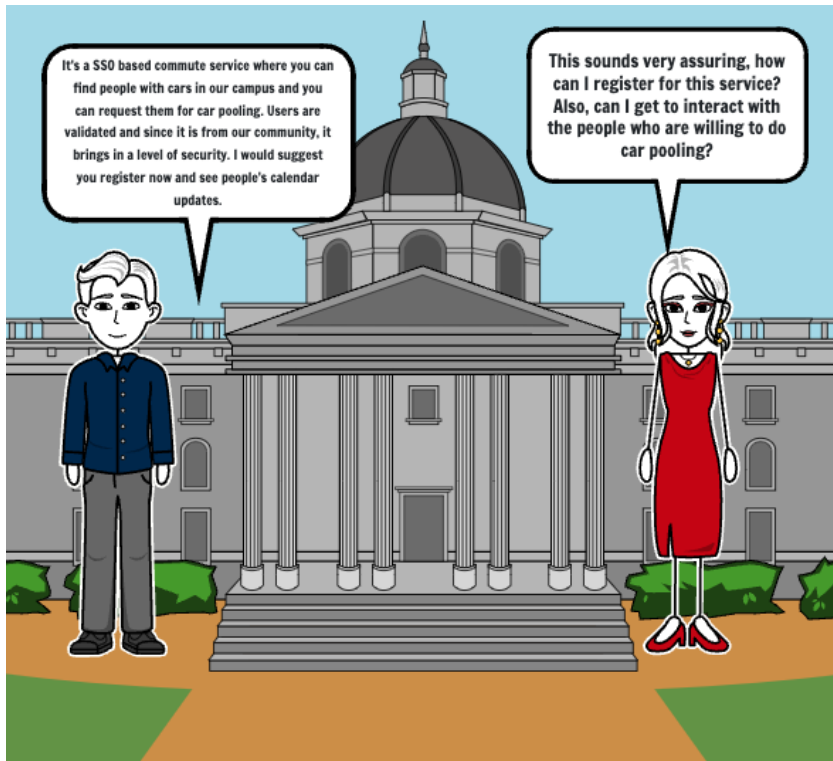
Q: Thank you Eva, See you around.

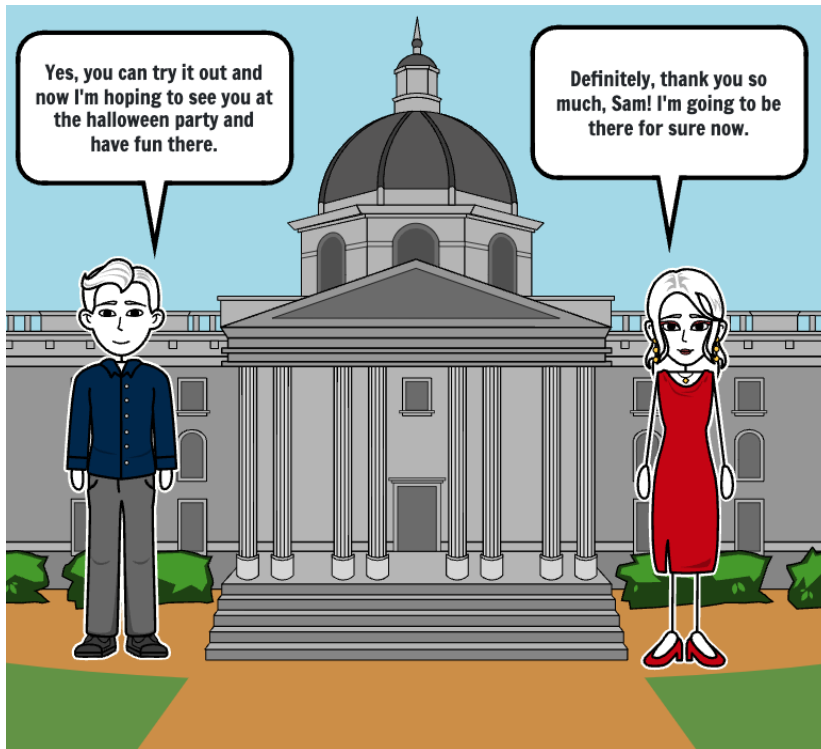
Storyboards:

Storyboard 01:

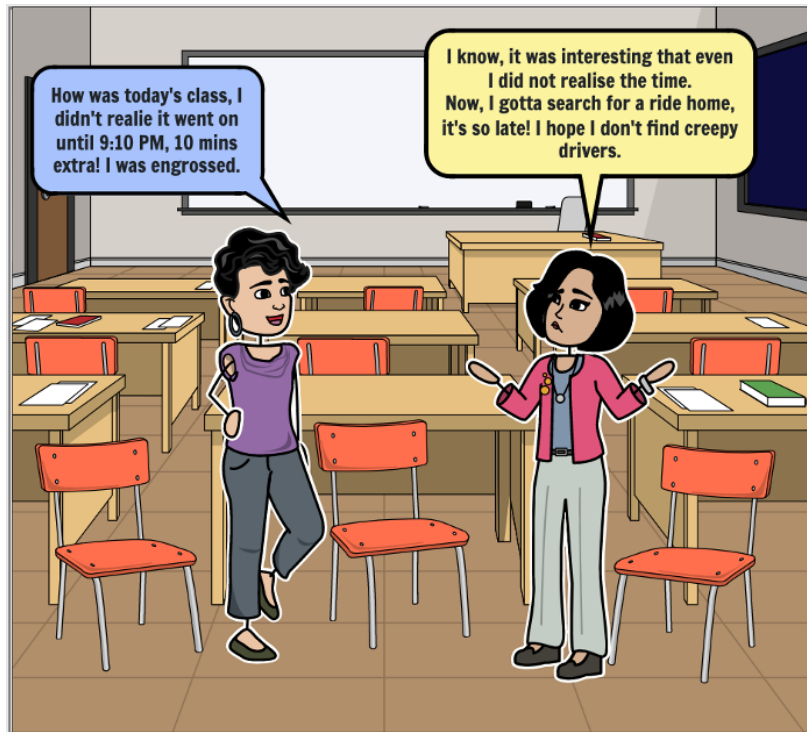


User Requirements Document - Always In Vogue

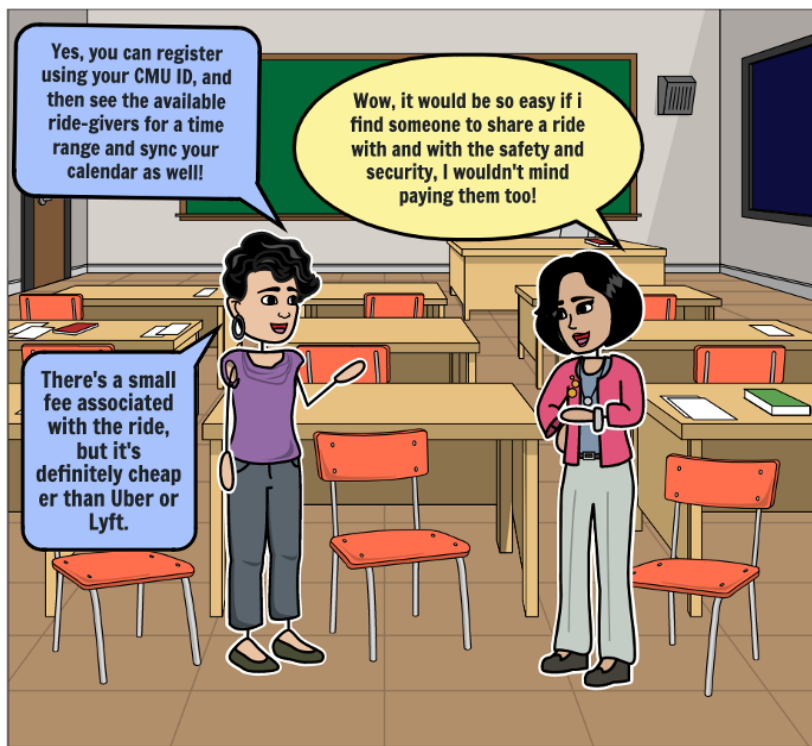
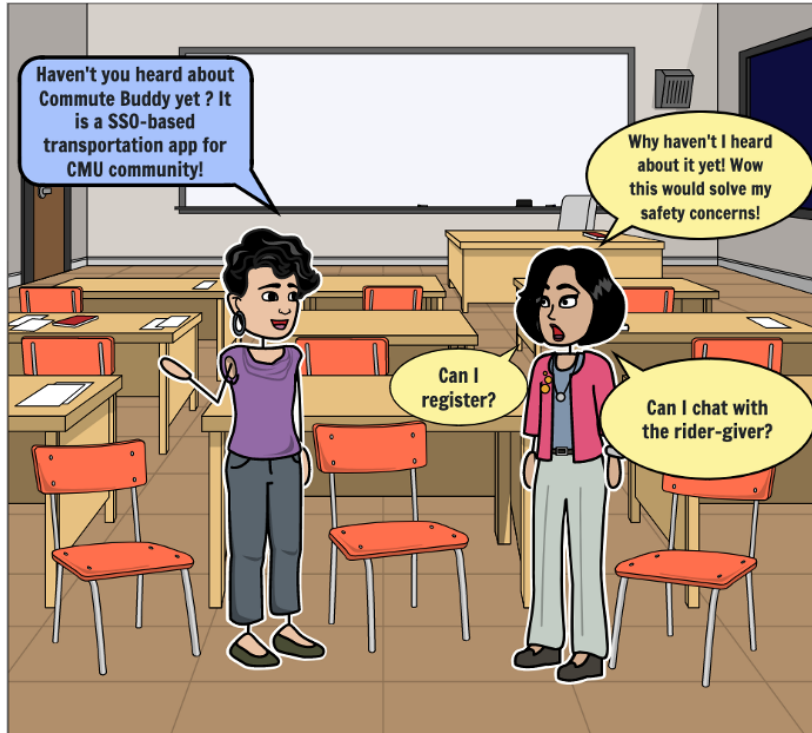




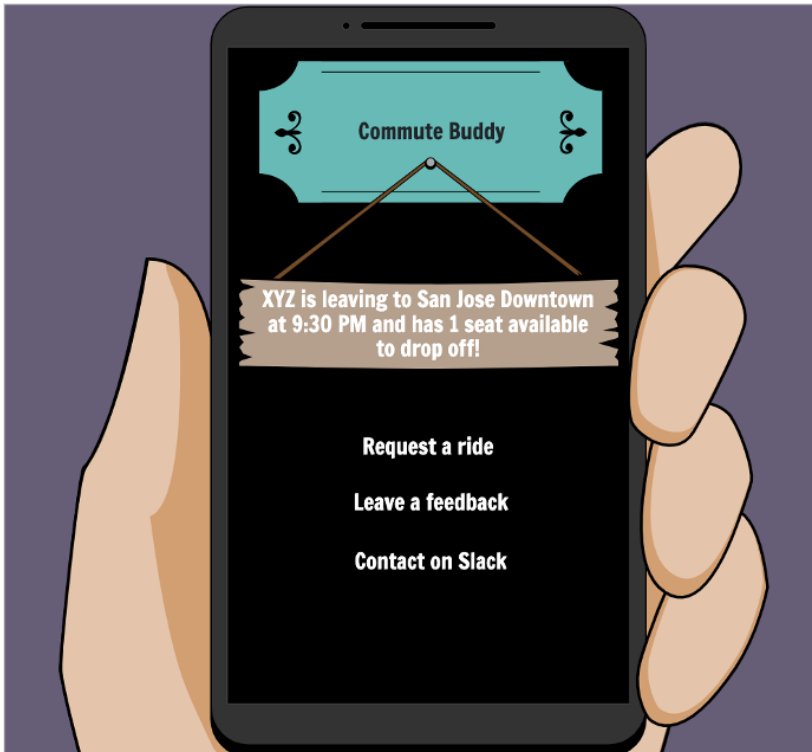
Storyboard 02:



User Requirements Document - Always In Vogue



User Requirements Document - Always In Vogue



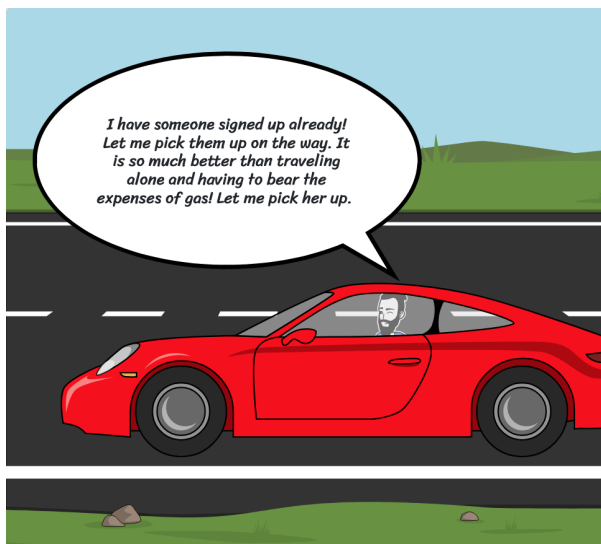
User Requirements Document - Always In Vogue



Storyboard 03:



User Requirements Document - Always In Vogue



User Requirements Document - Always In Vogue

