CEN 4010

Principles of Software Engineering

Milestone 1 Project Proposal and High-level Description

RideMate

Group: Team 1

Team Lead

Gikai Andrews - gandrews2020@fau.edu

Team members

Daniela Herrera - dherrera2022@fau.edu
Antonio Croissy - acroissy2019@fau.edu
Ameer Khan - akhan2019@fau.edu
Jerusha Stewart - jerushastewa2022@fau.edu

Milestone 1

06/20/2023

Table of Contents

Page

Executive Summary 3
Competitive Analysis 4
Data Definition 7
Overview, Scenario, and Use Cases 9
Scenarios 9
Use Cases 10
High-level Functional Requirements 13
Non-Functional Requirements 14
High-level system architecture 17
Team 18

Executive Summary

Sometimes people take education for granted, overlooking that not everyone has the privilege to experience it. Many students face financial struggles, which can pose challenges when accessing university education, particularly if they lack the means for a personal vehicle or if they live so far that public transportation is not an option. However, these circumstances should not impede their ability to pursue higher education and better themselves.

With RideMate getting to university will be affordable, safe, convenient, and eco-friendly. RideMate is a system designed specifically for students that will eliminate the challenges of commuting alone for hours and connect like-minded students, promoting safety and a sense of community within the campus environment. Some of the system's features are user registration and authentication, ride booking, reviews and ratings, driver matching, and university selection. The main difference between RideMate and other existing ride-hailing apps is that rides will only occur between students. To minimize safety issues, users will be authenticated by school id and email, making sure they are a student at the same university. One of the advantages of RideMate is that it is significantly customizable. It can be personalized by adding profile pictures and reviews, as well as, configured to meet individual needs such as one gender ride or location of pick up. This strengthens the website's identity in the community.

RideMate offers a special level of flexibility for its users. Whether you need a ride from anywhere to your selected university or wish to provide transportation, the platform allows you to post a ride with a specified number of available seats. Alternatively, you can quickly find a ride to school as soon as they become available. This unique flexibility empowers users to obtain or provide transportation at any time. RideMate makes transportation convenient and accessible for all, regardless of their location or schedule. Also, the system is designed to be user-friendly, with intuitive search and navigation tools making it easier for users to find and reserve rides, and receive notifications about availability.

RideMate - university transportation meets the power of community.

Competitive Analysis

Since there are few platforms exclusively dedicated to carpooling among college students, we will compare websites that offer either carpooling services or only students feature. The competitive analysis will compare five websites focusing on their five main features: design, security, usability, feedback, and performance. Using a numerical scale from 1 to 5 where 1 is bad, 2 is poor, 3 is fair, 4 is good, and 5 is excellent.

	RideMate	**Uber**	**Lyft**	**Zimride**	**Waze carpool**	**Scoop**
Design	5.0	4.0	4.0	3.5	3.5	3.5
Security	5.0	3.5	4.0	3.5	4.0	3.5
Usability	4.5	4.0	4.0	3.5	3.5	3.5
Feedback	4.5	3.5	4.0	3.0	3.0	3.0
Performance	4.0	4.0	4.0	3.5	3.5	3.5
Search	5.0	4.5	4.5	3.5	4.0	3.0
Content	4.0	5.0	5.0	3.0	4.0	2.0
Average	**4.6**	**4.1**	**4.2**	**3.3**	**3.6**	**3.1**

RideMate (4.6)

RideMate stands out with its user-friendly and straightforward design, ensuring that features are easily accessible and intuitive for users. The website offers a seamless and uncomplicated booking process with contrasting colors in its design so users can have an intuitive and inclusive interface that simplifies the carpooling experience. RideMate ensures that ride-sharing is limited to students from the same university implementing user authentication through email, school ID verification, and a third-party authentication app. This stringent authentication process minimizes safety concerns by significantly reducing the likelihood of unauthorized individuals gaining access to the platform. This allows students to connect with peers who are likely to share similar schedules, locations, and interests, fostering a sense of camaraderie and convenience

within the university community. The website's content is based on ride availability and ratings. Search and navigation are fast and operate with autocomplete and breadcrumbs features. By focusing on a specific university or a limited number of universities, RideMate can optimize its services by identifying popular routes and pickup/drop-off points.

Uber (4.1) https://www.uber.com/

Uber has a clean design and navigation, but sometimes some users find the app overwhelming due to the abundance of features. Despite the implementation of GPS tracking and in-app emergency assistance, incidents involving drivers have been reported, highlighting the importance of enhancing safety measures. Uber's performance and rating system require substantial improvement. The lack of available drivers in specific locations often leads to prolonged waiting times for customers seeking rides. Moreover, the rating system is susceptible to manipulation, as there is a lack of control over unfair ratings that may be influenced by factors unrelated to the ride experience. These issues highlight the need for more effective measures to ensure a reliable and fair system for drivers and passengers.

Lyft (4.0) https://www.lyft.com/

As a direct competitor to Uber, Lyft shares several similarities in its overall performance and usability. Just like Uber, Lyft faces challenges in terms of security measures, objective ratings, and driver availability based on location. In some instances, Lyft may have fewer drivers compared to Uber since the latter is more popular. One characteristic of Lyft is its vibrant design which may be visually appealing to many users. However, it can sometimes result in a less intuitive user experience.

Zimride (3.4) https://www.zimride.com/

Although Zimride initially started as a public ride-hailing app, it has transitioned to mainly offering ride-sharing services to universities and businesses. Despite its shift in market target, it is not a platform exclusively for students like RideMate. Zimride's design lacks the advanced features and visual appeal found in more popular ride-sharing platforms such as Uber or Lyft.

Waze Carpool (3.5) https://www.waze.com/

Even though Waze Carpool connects users within a community, there is a limited additional security measure outside the verified profiles by pictures of commuters and drivers. Despite being fairly known, Waze Carpool's design feels less polished and its rating system may have fewer entries in comparison to Uber or Lyft. Like its competitors, Waze Carpool does not specifically target a particular market. Therefore, college students using this platform for carpooling should take into consideration that it is a public service, and they will not be riding exclusively with fellow students.

Scoop (3.4) https://www.scoopforwork.com/

Scoop's greatest disadvantage lies in its limited offering of rides exclusively for individuals who are meeting at the same time and location. The platform operates by scheduling rides through the use of polls by determining when people want to meet at the office. While this system structure enables efficient coordination among riders, it also restricts the availability of rides, potentially posing challenges for those seeking more flexible transportation options. Scoop's visual appeal may be lacking, characterized by dull colors and an outdated design. Additionally, due to its smaller user base, the number of ratings and reviews on the platform is comparatively lower than on other ride-sharing platforms. Although Scoop implements certain safety measures such as user verification and ratings, it offers fewer additional safety features in comparison to Uber and Lyft.

Planned Advantages:

RideMate concentrates on students' safety and convenience, so some of the future features to be implemented on this site include but are not exclusive to reservations, university specifications, advance schedule availability, special offers, chat, attend events rides, and female-only rides. For now, RideMate is focused on offering a transport platform for students at Florida Atlantic University, but we will soon offer the same services to different college campuses. RideMate can tailor its features to meet the specific needs of university students. For example, it could offer features like in-app communication platforms, making it easier for students to coordinate rides, join clubs, or attend events. RideMate could collaborate with local businesses or university organizations to provide exclusive discounts or rewards for using the app. This can improve the overall value proposition for students and promote the platform within the university community.

Data Definition

Name	Meaning	Usage	Comment
User	Actor	Use Case scenarios	The person accessing the website
Review	Service	Site user service	Allows users to rate Student car owners and fellow passengers
Max	Service	Site user/car owner service	Allows users to input the max amount of people they would be comfortable carpooling with. Also allows the owner of the car to input how many people can fit within their car
Search	Service	Site user service	Allows users to search for rides available near them
Login	Service	Site user service	Allows users to have the ability to carpool while sharing their ratings with possible future fellow passengers
Filter	Service	Site user service	Allows users to filter things such as max and distance between them and other cars
RideMate	User Interface	User interface	Website name and front-end display for user interaction
Znumber	Data	Use Case Scenarios	Must have a valid Z-number to use the site's services

System	Platform Hardware	Use Case Scenarios	The database, with all code and front-end and backend hosting
Resource	Data and hardware	Use Case Scenarios / Hardware	Hardware / Data on the website
Invalid User	Actor	Use case scenarios	Definition of a non-university person
Login	Service	Site user Service	Allows user access to the system for use
Server	Platform Hardware	Use Case Scenarios	Where site and data stored
Homepage	User Interface	User Interface	The landing page of the site
Website	User Interface	User Interface	Front end for user interaction
API	Program Interface	Use Case scenarios	The interface between the front and back end
Reserve/Reservation	Use Case scenario	Use Case Scenario	When a ride is requested and booked

Overview, Scenario, and Use Cases

RideMate is a rideshare application that has been developed by a group of college students. The goal of the appliance is to provide transportation services for college students so they can travel all over campus. Also, our goal is to ensure that we have fair and reasonable pricing for the rideshares. We can offer competitive pricing options, including student discounts or discounted rates for frequent users. This can make the app more appealing to cost-conscious college students who are looking for affordable transportation options within their limited budgets. Ridemate will include enhanced safety measures because we do emphasize our commitment to ensuring a safe and secure transportation system/experience for college students. Also, we can do a seamless integration with Campus Life. Our application could establish partnerships/integrations with the college administration or student services. We could also collaborate with campus security for enhanced safety measures, integrating with the college's official app for seamless navigation and scheduling, etc.

Scenarios

- **Morning Commute:** Sarah, a college student, uses a ride-sharing app to request a ride from her off-campus apartment to the university every morning. The app matches her with a nearby driver who drops her off at the main entrance of the campus, saving her time and hassle.

- **Late-Night Study Session:** John is studying late at the library on campus and realizes it's past midnight. Feeling unsafe walking alone at this hour, he opens the ride-sharing app to request a ride back to his dormitory. The app promptly assigns a driver who picks him up and safely drops him off at his dormitory entrance.

- **Group Outing:** A group of friends wants to go to a concert happening downtown. Instead of relying on public transportation, they use a ride-sharing app to request a ride together. They select a meeting point on campus, and the app matches them with a driver who can accommodate their group size. The driver takes them directly to the concert venue and later returns to pick them up for a hassle-free ride back to campus.

- **Off-Campus Errands:** Lisa needs to run errands outside of the campus area. She uses the ride-sharing app to request a ride to the grocery store, bank, and pharmacy. The driver waits for her during each stop and then takes her back to campus. This saves her time and eliminates the need for multiple transportation arrangements.

Use Cases

Use Case - Search

The user comes to the page and wants to use the search field to find rideshare. The user arrives at the main page and uses the search field to enter search criteria. The application displays the search results for the location they want to go and the user picks among the search results.

1. **Description:**

   The use case describes the process of how the User will utilize the search feature of the system.

2. **Actors:**

   2.1 User

   2.2 System

3. **Preconditions:**

   3.1 User has an active Internet connection

   3.2 System is available

4. **Primary Flow of Events:**

   4.1 User arrives on the web page.

   4.2 User enters search criteria of location in the search field

   4.3 Web page displays relevant search results.

   4.4 User selects Location from the search results.

   4.5 Terminate Use Case: Search

5. **Alternate Flows**

   5.1 User Enters Prohibited Characters/Format into Search Bar

   If, in step 2, the user enters prohibited characters or a prohibited format in the search bar. (User enters all numbers or punctuation)

1. Website notifies users that the characters they used are prohibited

2. Return to Step 1

5.2 Users enter Unknown Location

If, in step 2, the user enters a location/activity that doesn't exist in the database.

1. System performs Use Case: Add a Location

2. Terminate Use Case: Search

<u>Use Case - Availability</u>

The user searches for rideshare on the main page and of course gets the results they want. The user goes to the calendar and has the opportunity to schedule the rideshare now or in the future. The application opens up a calendar with time slots to make the user flexibility.

1. **Description**

   The use case describes the process of how the User will utilize the availability system.

2. **Actors:**

   2.1 User

   2.2 System

3. **Preconditions**

   3.1 User has an active Internet connection.

   3.2 System is available.

4. **Primary Flow of Events**

   1. The user arrives at the main page.

   2. User does confirmation of rideshare.

   3. Web Page pops up with a calendar and time slots of when the user wants to get picked up.

   4. User selects the time and date for rideshare pickup.

<u>Use Case - Location</u>

1. **Description**

   The use case describes the process of how they selected the pickup and drop-off location in the application.

2. **Actors:**

   2.1 User

   2.2 System

3. **Preconditions**

   3.1 User has an active Internet connection

   3.2 System is available

4. **Primary Flow of Events**

   1. The user arrives at the main page

   2. User types in the search field what location to go to

   3. User confirms dropoff location and rideshare

   4. The user chooses the pickup location whether it's exactly where they are or anywhere else.

Use Case - Rating/Tipping

1. **Description**

   The user wants to give a rating and tip to the rideshare driver after completing the rideshare.

2. **Actors:**

   2.1 User

   2.2 System

3. **Preconditions**

   3.1 User has an active Internet connection.

   3.2 System is available.

4. **Primary Flow of Events**

   1. The user arrives at the review page after the rideshare trip has been completed.

   2. The user looks through the rideshare driver, optional tip options, and rating system.

   3. The user picks how much of a rating the user wants to give the rideshare driver.

High-level Functional Requirements

1. **Accounts -** Ridemate will have a system for creating and managing user accounts. Ensuring to properly store account info in a safe manner while preventing the creation of accounts with duplicate usernames and emails, all while allowing for account recovery in case of forgotten passwords.

    a. **Signups -** taking in info from a user to create a username, password, and email for account recovery purposes.

    b. **Logins -** taking in a user's username and password.

    c. **Recovery -** in the case of a forgotten password the system will send an email to the associated account to create a new password.

2. **Getting a Ride -** A user will have the ability to check for available rides in their area for either pick or drop-off from school or a designated location.

    a. **Mapping -** the site will be able to display a map of the local area around the user to check for available rides.

    b. **Tracking -** the systems will be able to keep track of currently available rides for users to call upon.

3. **Scheduling -** The system will allow users to make scheduled pick-ups and drop-offs with other riders.

    a. **Calendar -** A calendar for scheduling rides to work within a user's schedule, and to allow drivers to indicate when they are available for service.

Non-Functional Requirements

1. Performance

The performance requirements of the webpage will count on a number of varying parameters.

1.1 Responsiveness: This is the ability of the website to load on a variety of screen sizes and resolutions as well. The typical screen sizes that should be catered for are 11.6 to 17-inch laptop screens and also caters to 19 to 30-inch desktop monitor screens as well. The resolution should be capable of handling from the lowest 1280 x 1024 to the highest range of 1920 x1200.

1.2. User Concurrency: The system should be efficient to handle multiple users at a time. For this project, we will average 10 people accessing the webpage at a time with the system not slowing down. Any increase in users at a single instance will result in slowing down or lagging of the user interface. We will strive to have a load time of under 2 seconds which is an acceptable industry standard.

1.3. Transaction speed per request: The transaction speed should be under five seconds to avoid users from getting impatient which may lead them to leave the webpage ideally we will be aiming for a time of about 3 seconds.

1.4 Reliability: The webpage must be reliable and should be glitching or failing on a continuous basis. The website should be kept current and up to date to eliminate failures and crashes at any point in time. The most acceptable for this new app may be one crash or failure per month.

1.5 Storage Utilization: The website should run and utilize the allocated memory it was given, it should not allow the accumulation of extra files and temporary files will grow the memory in a significant way that all the allocated memory will be used up causing a crash of the site.

1.6 Recovery/Robustness: The website should be easily recovered in case of any failure and should be able to be put back in service within 30 minutes of any failure since this is a small website catered to a limited range of people.

1.7 Backup/ Survivability: An entire backup file of all website data should be kept to easily be loaded back into the system to allow for continuity of service.

2. Usability

2.1 Ease of use: The website should be easy for anyone with basic computer skills should be able to train or adapt to using the website easily. Ease of use will be a key feature of the website.

2.2 Learnability: The website should be easy to learn and memorize ideally a fifteen-minute period the website may be memorized.

3. Accessibility

3.1 The website should be accessible over a range of internet bandwidths. The range should cater for speeds 25 Mbps and upwards.

3.2 The website should be able to load from a variety of internet browsers. These should include but not be limited to Firefox, Chrome, Microsoft Edge, and Safari.

4. Expected Load

The expected load is an average of ten users at a time. We anticipate this as this website and service will be used by the students of Florida Atlantic University only. The initial version may cater to the Boca Raton campus and then expand to the Davie and Harbor branches eventually.

5. Security Requirements

5.1 Login/Password System: The security requirements will require an FAU email to log in along with a password.

5.2 The Duo app or a similar type of authentication will be used to ensure that only FAU-associated students are allowed to use the service.

6. Portability

The website will be available for use on desktops, laptops, and tablets via a web browser eventually there are plans to extend this to an Android and an iOS app.

7. Supportability

The coding to be used will be HTML and CSS. The Figma app will be used for the testing page.

8. Availability

8.1 The system must be reliable ad available 24 hours a day 365 days a year.

8.2 Tech support will be available via an email system.

8.3 All attempts will be made to ensure continuous if there is a need for any downtime communication will be sent out to all users.

9. Fault Tolerance

All attempts will be made to handle any sort of exception handling that may be needed for this app. All attempts will be made to handle this within the allocated resources.

High-level system architecture

1. **Programming Languages: Javascript, HTML, CSS** - These languages will be used in the development of the website's web page and front end to serve as an interface for the use.

2. **Frameworks: React, Node** - Javascript frameworks that will be used for allowing interactions between the website and its users. React will make the basis of the front-end interaction like user login and registration. Whilst node will be the framework that will allow for management of the backend.

    a. **Google Login: https://www.npmjs.com/package/react-google-login** - Used to communicate Google login data between both front and backends.

    b. **Additional Library: https://chakra-ui.com/** - libraries to aid in the development and management of primarily front-end javascript frameworks.

3. **Development Tools: Visual Studio, GitHub** - Tools used to allow for a steady development flow as GitHub will host the project and allow for project members to work on their portions individually, and visual studio as a code editor for html, javascript, and CSS.

4. **Core APIs: Uber API** - The Uber API will be used to allow for the tracking, managing, and scheduling of rides at designated or user-specified locations.

5. **Supported Browser: Chromium browsers** - As the front end of our service is a web page, the website will be developed to support modern browsers and especially browsers based on Chromium such as google chrome, opera, and Microsoft Edge.

6. **Bootstrap Templates: https://getbootstrap.com/** - A bootstrap template will help to bring a friendly, interactive, and intuitive design to our site.

Team

Scrum Master

Gikai Andrews

Product Owner

Daniela Herrera

Front-End Developers

Jerusha Stewart

Antonio Croissy

Daniela Herrera

Back-End Developers

Ameer Khan

Gikai Andrews