Sprint Planning Document

Team 17: BoilerRide

Nadeem Mahmood

Ming-Da Liu Zhang

Karan Teja

Konstantin Diego Pandl

Srishti Gupta

Natasha Tyagi

Table of Contents

I. Sprint Overview

Overview

In our first sprint, our main focus will be on establishing the basics of the app, the Communication between the app and the backend server and storing data properly in the backend database. At the end of the sprint, the user should be able to sign up, login and logout of the app. Furthermore, the user will be able to request a new password in case she/he forgets it.

Scrum Master: Konstantin Diego Pandl

Scrum Meeting Time: Sunday, Tuesday, Thursday at 1:30 PM.

Risks and/or Challenges: Some team members are not yet familiar with Android programming and no team member is yet familiar with a Firebase backend architecture. Furthermore, some team members are not yet familiar with programming in a team and using a repository. For the first sprint, it will be a challenge that each of us must learn become acceptably proficient in full stack development.

II. Current Sprint Detail

User Story #1

As a user, I would like to download the app and create an account.

#	Task Description	Estimated Time	Owner
1	Setup backend server	6	Nadeem
2	Setup database	5	Nadeem
3	Setup basic Android application, compatible Material Theme and logo	6	Natasha
4	Setup communication to the backend server	8	Nadeem
5	Implement ability to create an account	8	Konstantin
6	Implement ability to login and maintain the session	6	Konstantin

Acceptance Criteria: If the user is implemented successfully, the user would be able to sign in using his pre-existing account credentials.

User Story #2

As a user, I would like to be able to reset my password.

#	Task Description	Estimated Time	Owner
1.	Setup a mail class to send emails to the user.	2	Karan
2.	Implement functionality for password reset on back-end(update to random password and send user	8	Ming-Da

	an email)		
3.	Implement functionality for users to update their password after following reset link.	4	Ming-Da
4.	Implement ‘forgot password?’ button and reset	2	Natasha

Acceptance Criteria:

- Given that the password reset functionality has been correctly implemented, when a user requests a password reset then he or she should receive an email at the email associated with their account that contains a link to reset their password.
- Given that the user has requested a password reset, when the user clicks the link to reset their password then the web page should display a form to enter a new password.
- Given that the user has filled out the password reset form, when the user submits the form to change their password then the database should immediately update to reflect the change

User Story #3

As a user, I would like to choose if I am a new or existing user.

#	Task Description	Estimated Time	Owner
1	Implement ability in the app that allows the user to choose to create an account	3	Konstantin
2	Implement ability in the app that allows the user to choose to log in	3	Konstantin

Acceptance Criteria: When the application opens, the user should be two choices to either login or create an account. Pressing the corresponding buttons, should direct them to correct page.

User Story #4

As a user, I would like to confirm my Purdue email.

#	Task Description	Estimated Time	Owner
1.	Implement a functionality to confirm users Purdue email after she/he logs into the account.	6	Karan
2.	Set up a mailer to send an email with a verification link.	2	Karan
3.	Store the user's information into the database after verification is done.	6	Ming-Da

Acceptance Criteria:

- After the user logs into his account, she/he will be asked to confirm their Purdue email.
- Once they have entered their email, they will be sent a mail on their Purdue account with a verification link.
- After clicking on the the link the user will be redirected to his account and his account details will be stored in the database.

User Story #5

As a user, I would like to choose if I am a driver or passenger.

#	Description	Estimated Time	Owner
1.	Implement ability to choose between being a driver or passenger in the app after login	2	Srishti

Acceptance Criteria: If the user story is implemented successfully, a user should see the correct pages and menus in the app depending on if they are a driver or passenger.

User Story #6

As an administrator, I would like to store and update the information of the drivers.

#	Description	Estimated Time	Owner
1.	Implement functionality for updating user information on backend	3	Natasha
2.	Design a View to allow the user to enter and submit changes to their profile.	6	Srishti
3.	Design a View to display a user's profile information, populated with their publically available information.	6	Srishti

Acceptance Criteria: If the user story is implemented successfully, a user should be able to update their profile information and changes will be saved so that they remain persistent and accessible (for public information) to other users.

User Story #7

As a user, I would like to see my path on a map.

#	Description	Estimated Time	Owner
1.	Integrate google maps in the application	6	Karan
2.	Implement functionality to display the user's path.	8	Ming-Da

3.	Get the user's phone location (GPS and other)	6	Nadeem
4.	Keep updating the user's location and show the path on the integrated Google Map	4	Karan

Acceptance Criteria: If the user story is implemented successfully, a user should be able to show his location and display his path after posting a ride.

User Story #8

As a driver, I would like to post a ride within the app and as a ride seeker I would like to request a ride.

#	Description	Estimated Time	Owner
1.	Create the API function and handle the information sent, and have appropriate responses	8	Natasha
2.	Implement a separate page for posting the rides and include Google Maps to view the route	4	Konstantin
3.	Implement a separate page for requesting the rides and include Google Maps to view the route	4	Natasha

Acceptance Criteria: Once the user story is implemented successfully a user should be able to post for a new ride or accept a ride with all the basic information like date, destination, time etc.

User Story #9

As a user, I would like to see all the posts and have the ability to filter them.

#	Description	Estimated Time	Owner

1.	Pull all the ride posts and requests from the backend database.	6	Srishti
2.	Implement a Listview for viewing the rides	8	Srishti
3.	Implement the basic filtering function and filter between ride offer and request	4	Konstantin
4.	Add more filters based on destination, price, etc.	4	Natasha

Acceptance Criteria: A user would be successfully able to view the list of all the rides offered and would have the ability to filter them according to destination, price, ride offer and request.

Remaining Backlog

#Note: The requirements highlighted in bold are being covered in Sprint-1 planning.

Functional

- **As an administrator, I would like to:**
    1. **store the information of the drivers.**
    2. **send verification emails to the users.**
    3. have the drivers put their profile picture.
    4. suggest a payment method (e.g. Paypal, Venmo, cash).
    5. set the max price drivers can charge.

- **As a user, I would like to:**
    1. **choose if I am a new or existing user.**
    2. a brief tutorial of the application.
    3. **create an account.**
    4. **login into my account.**
    5. **confirm my Purdue email.**
    6. **choose if I am a driver or passenger.**
    7. **have the option of resetting my password.**
    8. see my notifications.
    9. contact other users.
    10. set my privacy settings.
    11. **give my ride share post a title.**
- **As a driver, I would like to:**
    1. know my passengers reviews.
    2. ***offer a ride.***
    3. review the passenger.
    4. set how much I am charging for a ride.
    5. know the passenger's contact information.
    6. change the information about my offered ride.
    7. set the maximum number of passengers with me on the ride.
    8. *see my path on a map.*

- **As a passenger, I would like to:**
    1. ***request a ride.***
    2. look for other passengers for trips with similar destination.
    3. see a list of all the rides offered.
    4. see a list of the rides offered sorted by my filter choice.
    5. see a list of all the rides requested.
    6. see a list of the rides requested sorted by my filter(price, driver rating, time, etc.) choice.
    7. search list of rides by keywords
    8. change the information about my requested ride.
    9. review the driver.
    10. see my drivers reviews.
    11. be redirected to a payment app to fulfill the payment.
    12. set the number of passengers going with me on the ride.
    13. know the estimated time of the ride.
    14. know the estimated fare of the ride.
    15. know the driver's contact information.
    16. **set the pick-up time.**
    17. **set my drop-off location.**
    18. set how much I am willing to pay for a ride.
    19. **see my path on a map.**

Non-Functional

1. The application will run on an android device.
2. ***Must be able to integrate Google maps.***
3. **Must be able to use the GPS.**
4. Must be able to integrate with some backend web server service(e.g. Amazon Web Services or Firebase) (time permitting).
5. The interface needs to be simple and user-friendly.
6. **Must only allow Purdue affiliates to use the app.**
7. As a user, I would like to have fast response times and efficient use of bandwidth.
8. The consequences of security breaches are low, and therefore the only security precautions we will take are data encryption and connection encryption between client and server.
9. The connections between the UI, server and database will be strong