



Sprint 2 Planning Document

Team 18

Sai Edupulapati, Akshaya Kumar, Shreya Sanapala, Visa Thongdee, Vidya Vuppala

Sprint Overview

During this sprint, we hope to build from the foundational work done in our previous sprint and produce a minimum viable product of custom playlist generation. This includes finalizing our quiz implementation (i.e. retaking the quiz, resetting quiz results, etc), doing research on the best Spotify API, and designing an algorithm for playlist creation. We also hope to begin implementing a few of our basic map features, such as inputting a start location, destination, and mode of transport.

Scrum Master: Vidya Vuppala

Meeting Schedule: Mondays/Wednesdays at 1:30 PM, Fridays at 3:00 PM

Risks and Challenges:

Although we completed the vast majority of our objectives from the previous sprint, there are still a few ways that we can improve upon those tasks in terms of the overall user experience and how each component flows together. We plan on spending a portion of our time this sprint working on those issues which could potentially set back our progress for this sprint's goals. Additionally, the challenge of building an algorithm for playlist creation is new to all of us and comes with a large learning curve. It is a key feature of our project, so it's critical that we are able to finish this user story.

Current Sprint Detail

User Story #1

As a user, I would like to be able to delete my profile picture.

#	Description	Estimated Time	Owner
1	Update UI to 1) add button for deleting picture and 2) display necessary prompts upon success	1 Hr	Akshaya
2	Implement appropriate logic for deletion of profile picture in Firebase and Firestore	1 Hr	Akshaya
3	Debug delete functionality and test with unit testing	2 Hr	Akshaya

Acceptance Criteria:

- Given that the Delete Profile Picture button is implemented correctly, when a user clicks on it, they will be shown a pop-up that asks them to confirm their decision.
- Given that the Firebase backend is linked properly, when a user deletes their profile picture, the picture URL and image is deleted from Firebase and Firestore.
- Given that the UI is linked correctly, when a user deletes their profile picture, the UI is automatically updated to reflect the lack of a profile picture.

User Story #2

As a user, I would like to be able to edit my account information (ex. demographic, music preference, etc.) as needed.

#	Description	Estimated Time	Owner
1	Modify Settings UI to include 'Edit' buttons and textfields to edit account information	2 Hrs	Visa
2	Implement appropriate logic for updating account information	3 Hrs (because she's currently inexperienced in Firebase)	Visa
3	Update UI to reflect edited account	2 Hrs	Visa

	information		
--	-------------	--	--

Acceptance Criteria:

- Given that the UI is created properly, when the user views the page, they see a text field and 'Edit' button in which they can update their demographic information.
- Given that the UI is created properly, when the user views the page, they see a text field and 'Edit' button in which they can update their music preference information.
- Given that the Firebase backend is linked properly, when the user edits their account information, the corresponding user details in the backend is updated.

User Story #3

As a user, I would like to be able to view my listening data (total minutes, genres, artists).

#	Description	Estimated Time	Owner
1	Set up appropriate front end format for displaying the data	4 Hr (because he has no experience with frontend)	Sai
2	Gather the data mentioned above from Spotify Web API	2 Hrs	Sai
3	Set up necessary structure in Firebase backend to store data in an organized way	2 Hrs	Sai
4	Recall data from backend and display onto the appropriate page on the website	1 Hr	Sai
5	Debug and Test (connecting/disconnecting, etc.)	1 Hr	Sai

Acceptance Criteria:

- Given that website navigation is working, when the user navigates to the profile page, they should be able to see a section dedicated for user statistics.
- Given that the Spotify authentication is working, when the user grants permission, an appropriate document/field structure should be created in Firebase to store their info.
- Given that Firebase organization is set up correctly, when the user grants permission, they should be able to see these fields populated with their appropriate data retrieved from Firebase.

User Story #4

As a user, I would like to be able to retake the quiz for different results.

#	Description	Estimated Time	Owner
1	Change structure of database to implement time based entries and/or quiz number to allow for quiz history	4 Hrs	Shreya
2	Change quiz UI structure so that there are Next and Previous button with quiz selections rather than automatically going to the next question upon answer selection	4 Hrs	Shreya
3	Change UI for quiz so that user's past quiz selections are populated in the quiz when they click the previous/start over buttons	4 Hrs	Shreya

Acceptance Criteria:

- Given that the UI implementation for the quiz structure is correct, when the user takes the quiz, they are able to navigate through the quiz with Next and Previous buttons.
- Given that the UI implementation for the quiz structure is correct, when the user navigates to a question that they have already answered, they will be able to see their past quiz selection.
- Given that the connection to the backend is correctly implemented, when the user changes their quiz selections and submits their quiz, a new entry is made in Firebase corresponding to their selections.

User Story #5

As a user, I would like to be able to see my past music preferences (i.e. past answers to quizzes, past results).

#	Description	Estimated Time	Owner
1	Modify profile UI to include past quiz history button	1 Hr	Vidya, Visa

2	Create a new page and skeleton UI for past quiz history page	1 Hr	Vidya, Visa
3	Connect quiz answers to newly structured database (Refer to User Story #4, Task 1)	1 Hr	Shreya
4	Connect quiz answers database to profile page UI	3 Hrs	Shreya

Acceptance Criteria:

- Given that the database is restructured properly, when a user submits their quiz data, the database will be able to add the most recent quiz data while also maintaining the user's old quiz data.
- Given that the UI for the past quiz data is implemented correctly, when the user views their profile page, they will be able to click a button to travel to the past quiz history page.
- Given that the backend for the quiz history is implemented correctly, when the user views their quiz history, all of their past quizzes are populated.
- Given that the backend for the quiz history is implemented correctly, when the user views their quiz history, the past quizzes are in chronological order.

User Story #6

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

#	Description	Estimated Time	Owner
1	Design an algorithm that locates all of the quiz entries that are made by a certain user and deletes them	4 Hrs	Shreya
2	Create aesthetic button on past quiz history page to delete past quiz history	0.5 Hrs	Vidya
3	Connect the delete past quiz history button to the backend	1 Hr	Shreya
4	Update the UI of the past quiz history page to reflect the deleted quizzes	2 Hrs	Shreya

Acceptance Criteria:

- Given that the quiz history deletion algorithm is implemented correctly, when the user clicks on the delete quiz history button, all of their past quiz entries will be deleted from the database.
- Given that the quiz history deletion algorithm is implemented correctly, when the user clicks on the delete quiz history button, none of their past quizzes will appear on the past quiz history page.
- Given that the button for deleting past quiz history is implemented correctly, when the user tries to delete an already empty list of quizzes, the delete past quiz history button will be inactive/grayed out.

User Story #7

As a user, I would like to be able to toggle on preferences for a playlist (ex. seasonal, genre, mood, location).

#	Description	Estimated Time	Owner
1	Create UI for playlist for toggles	1.5 Hrs	Vidya
2	Connect backend to store user preference	2.5 Hrs (because she's currently inexperienced in Firebase)	Vidya
3	Edit and format settings page to accommodate toggle	2 Hr	Vidya
4	Design UI for toggle and updated setting page	1 Hr	Vidya

Acceptance Criteria:

- Given the UI is designed and set up properly, when the user wants to access the toggle buttons, it should be viewable from the settings page.
- Given the UI is set up properly, when the user toggles on a preference, the UI button switches between options for each preference.
- Given the UI is connected to the backend, when the user chooses a preference, the choice is stored in the backend.

User Story #8

As a user, I would like to be able to enter a starting location.

#	Description	Estimated Time	Owner
1	Create UI to make user trip	3 Hr	Vidya
2	Add textfield to enter starting location	0.5 Hrs	Vidya
3	Connect UI to backend to store user input	1 Hr	Akshaya
4	Complete in-depth research on how to incorporate autocomplete tool to give user suggestions and implement tool	6 Hr	Akshaya

Acceptance Criteria:

- Given the UI is set up properly, when the user wants to enter the starting location, they should find that the text field is encapsulated in the UI for the user's trip.
- Given that the UI and backend are connected, when the user enters the location, the user's input is stored in the database.
- Given the autocomplete feature is set up properly, when the user starts entering their location, the user should get suggestions based on their initial input.

User Story #9

As a user, I would like to be able to edit a starting location.

#	Description	Estimated Time	Owner
1	Create UI to be able to edit journey with editable text fields	3 Hrs	Visa
2	Add a text field for editing the starting location	0.5 Hrs	Visa
3	Connect UI to backend to update starting location	1 Hr	Akshaya
4	Incorporate autocomplete to suggest locations	1 Hr	Akshaya
5	Add button to be able to navigate back to trip view	0.5 Hrs	Visa

Acceptance Criteria:

- Given the UI is set up properly, when the user wants to change their location, they should be able to change their input by typing in the text fields.
- Given the autocomplete feature is set up properly, when the user starts entering their location, the user should get suggestions based on their initial input.
- Given the backend and front end are connected, when the user enters the new location, the user should be able to navigate back to the trip view and see the updated location.

User Story #10

As a user, I would like to be able to enter my destination.

#	Description	Estimated Time	Owner
1	Create button for new trip in the homepage	1 Hr	Vidya
2	Create a quiz UI page to enter destination location	1 Hr	Vidya
3	Connect text field to an autocomplete API with maps location	1 Hr	Akshaya
4	Save user input/location in their account's Firebase backend	1 Hr	Akshaya

Acceptance Criteria:

- Given the UI is set up properly, when the user wants to create a new trip, the user can click on a button that will navigate them to another page.
- Given the UI is set up properly, when the user clicks the create trip button, the user will be prompted to enter their destination.
- Given the autocomplete feature is set up properly, when the user starts typing their destination, the user will get a location suggestions/autocomplete dropdown based on their input.
- Given that the UI and backend are connected properly, when the user inputs their destination, the location will be stored in the database.

User Story #11

As a user, I would like to be able to edit my destination.

#	Description	Estimated Time	Owner
1	Create UI to be able to edit destination location with editable text fields	3 Hrs	Visa
2	Connect UI to backend to update destination location	1 Hr	Akshaya
3	Incorporate autocomplete to suggest locations	1 Hr	Akshaya
4	Add button to be able to navigate back to trip view	0.5 Hrs	Visa

Acceptance Criteria:

- Given the UI is set up properly, when the user wants to change their destination, they should be able to change their location by typing in the text fields.
- Given the autocomplete feature is set up properly, when the user starts typing their destination, the user will get a location suggestions/autocomplete dropdown based on their input.
- Given that the UI and backend are connected properly, when the user updates their destination, the location will be changed in the database.
- Given the backend and front end is set up properly, when the user enters the new location, the user should be able to navigate back to the trip view and see the updated location.

User Story #12

As a user, I would like to be able to set and edit my travel preference (ex. car, plane) for each trip.

#	Description	Estimated Time	Owner
1	Create a UI with text field to set travel preferences	1 Hr	Vidya
2	Create a separate UI with editable text field to allow changes to travel preferences	1 Hr	Visa
3	Store travel preferences in Firebase backend	1 Hr	Akshaya

4	Update changes in travel preferences in Firebase backend	1 Hr	Akshaya
---	--	------	---------

Acceptance Criteria:

- Given that the UI works, when the user clicks the new trip button, they are directed to choose their preferred mode of travel for the trip
- Given that the user previously entered their travel preferences, when the user chooses to edit their preferences, they will be able to change their mode of transport
- Given that the front end logic is set up properly, when the user clicks submit to finalize their choices, the information will be accurately stored in Firebase.
- Given that the front end is set up properly, when the user edits their travel preferences, their changes will be reflected in the Firebase backend.

User Story #13

As a user, I would like to be able to navigate through the website and access every appropriate page.

#	Description	Estimated Time	Owner
1	Create page for <i>every</i> feature with base features, color scheme, and corresponding aesthetic	7 Hrs	Vidya, Visa
2	Add the appropriate links/buttons to <i>every</i> page for navigation	2 Hrs (each) = 6 hours	Vidya, Visa, Akshaya
3	Add routing for every page	4 Hrs	Akshaya
4	Debug and test routing with unit tests	3 Hrs	Akshaya

Acceptance Criteria:

- Given that the UIs are created properly, when a user views a page, they are shown any relevant button that links to another page.
- Given that the navigation is set up correctly, when a user clicks a link/button on a page, they are redirected to the appropriate page.

- Given that the navigation is set up correctly, when a user uses our website, they are able to reach every UI (e.g. home page, quiz page for trip locations, confirm quiz answers page, loading page, expanded trip view page, edit source/destination page, etc.).

User Story #14

As a user, I would like to be able to see my past curated playlists.

#	Description	Estimated Time	Owner
1	Create skeleton UI to display past playlists	2 Hrs	Vidya, Visa
2	Set up Firebase backend to handle storage of playlists by user	2 Hrs	Sai
3	Create buttons to delete and clear history	0.5 Hrs	Vidya
4	Set up Firebase logic to remove associated user history	2 Hrs	Sai
5	Set up Firebase to retrieve history based on time intervals (12M ago, 6M ago, 3M ago)	3.5 Hrs	Sai

Acceptance Criteria:

- Given that website navigation is set up, when the user toggles the Past Playlists page, they will be able to view all the playlists curated for them from their various trips.
- Given that frontend handling is set up, when the user selects an appropriate time frame, they should be able to see playlists created only during that time.
- Given that the front end can communicate with the backend, when the user indicates they want to clear their listening history, all associated data stored in Firebase with that user account should be deleted.

Sprint 1 – Incomplete User Stories, Tasks, & Acceptance Criteria

User Story #12

As a user, I would like to get a music playlist for my journey.

#	Description	Estimated Time	Owner
1	Complete in-depth research on how to develop and optimize algorithm that creates customized playlist	7 Hrs	Shreya, Sai, Vidya
2	Design algorithm to implement custom playlist generation based on user preferences	10 Hrs	Shreya, Sai, Vidya
3	Optimize algorithm to implement custom playlist generation based on external criteria (e.g. source location, destination, travel time,)	10 Hrs	Shreya, Sai, Vidya
3	Store current playlist (i.e. name, songs, artists, length, etc) in the database	2 Hrs	Sai
4	Get custom link (includes research, debug, and redesign) and embed custom playlist on website UI	4.5 Hrs	Vidya, Sai
5	Debug algorithm with unit testing	5 Hrs	Shreya

Acceptance Criteria:

- Given that the playlist generation algorithm works properly, when the user provides their information (e.g quiz answers, Spotify data), they will be provided with a custom playlist.
- Given that the UI is set up properly, when the user finishes taking the quiz and submits, they are redirected to the page with an embedded playlist.
- Given that the backend is connected properly, when the user receives a playlist, the playlist is stored in the backend.

Remaining Backlog

Sprint 2 Backlog

1. User Account/Profile

As a user,

- ~~a. I would like to be able to log in with my account.~~
- ~~b. I would like to be able to delete my account.~~
- ~~c. I would like to be able to create a profile with a username.~~
- ~~d. I would like to be able to modify my username.~~
- ~~e. I would like to be able to add a profile picture to my account.~~
- ~~f. I would like to be able to modify my profile picture.~~
- g. I would like to be able to delete my profile picture.
- ~~h. I would like to be able to view a profile page with my profile information.~~
- ~~i. I would like to be able to change my password if needed.~~
- ~~j. I would like to be able to reset my password if needed.~~
- k. I would like to be able to edit my account information (ex. demographic, music preference, etc.) as needed.
- l. I would like to be able to view my listening data (total minutes, genres, artists).

2. Music Preferences

As a user,

- ~~a. I would like to be able to link my Spotify account to my Wander account.~~
- b. I would like to be able to change which Spotify account is linked to Wander.
- ~~c. I would like to be able to remove my Spotify account.~~
- ~~d. I would like to be able to take a quiz to determine my music preferences.~~
- e. I would like to be able to retake the quiz for different results.
- f. I would like to be able to see my past music preferences (i.e. past answers to quizzes, past results).
- g. I would like to be able to reset my quiz history.
- h. I would like to be able to toggle on preferences for a playlist (ex. seasonal, genre, mood, location).
- ~~i. I would like to be able to see my current favorite music preferences.~~

3. Playlist Creation

As a user,

- a. I would like to be able to get a music playlist for my journey.
- ~~b. I would like to be able to see my playlist embedded on the website.~~
- c. I would like to be able to edit my playlist.
- d. I would like to be able to export my playlist.
- e. I would like to be able to see my past curated playlists.

4. Itinerary Creation

As a user,

- a. I would like to be able to enter a starting location.
- b. I would like to be able to edit a starting location.
- c. I would like to be able to enter my destination.
- d. I would like to be able to edit my destination.
- e. I would like to be able to set my travel preference (ex. car, plane) for each trip.
- f. I would like to be able to edit my travel preference (ex. car, plane).
- g. I would like to be able to get a travel itinerary for my destination.
- h. I would like to be able to edit my itinerary.
- i. I would like to be able to see the overall map of my travels (i.e. show the map from starting point to destination).
- j. I would like to be able to see a smaller map of my itinerary midpoints (i.e. show a map from location A to location B to location C).
- k. I would like to be able to have a real-time map to see my current location.
- l. I would like to be able to export my itinerary.
- m. I would like to be able to see my past travel history.

5. Website Customization/Appearance

As a user,

- a. I would like to be able to access Wander on all devices as a website.
- b. I would like to be able to personalize the appearance of the website (light vs. dark theme, etc.)
- c. I would like to be able to hear sound effects from the website.

- d. I would like to be able to see animations within the website.
 - e. I would like to be able to navigate through the website and access every appropriate page.
6. Sharing
- As a user,
- a. I would like to be able to share my non-collaborative playlist within the website with one other user of this website (if time allows).
 - b. I would like to be able to share my non-collaborative itinerary within the website with one other user of this website (if time allows).
 - c. I would like to be able to share my non-collaborative playlist within the website with multiple other users of Wander (if time allows).
 - d. I would like to be able to share my non-collaborative itinerary within the website with multiple other users of Wander (if time allows).
 - e. I would like to be able to collaborate with a friend to create a playlist (if time allows).
 - f. I would like to be able to collaborate with a friend to create an itinerary (if time allows).

Non-Functional Requirements

1. Performance
- As a developer,
- a. I would like the website to run smoothly without crashing.
 - b. I would like the website to be responsive to all user requests.
 - c. I would like the application to be launched in 5 seconds.
 - d. I would like the server to be able to handle concurrent users.
 - e. I want both the client and server to elegantly handle errors and the failure of other components that they interact with.
2. Server
- As a developer,
- a. I would like the server to be able to store user playlist data in a Firebase database.
 - b. I would like the server to be able to store itinerary data in a Firebase database.

- c. I may want the server to be able to save user state and preferences.
- 3. Appearance
 - As a developer,
 - a. I would like to have an aesthetically pleasing website interface.
- 4. Security
 - As a developer,
 - a. I would like to allow the users to choose whether to share data location or not.
 - b. I would like to protect the user's password and other personally identifiable information stored in the database.
 - c. I would like the user data collected (e.g. photos, names, email addresses) to be anonymized or properly secured.
- 5. Usability
 - As a developer,
 - a. I would like the website to be accessible on all platforms and devices.
 - b. I would like to have an easy-to-use website interface.