Requirement ID 3.6: Navigation Bar with Pages

Ticket Details
Description: With this ticket, create a responsive navigation bar that allows users to easily navigate between different pages of the app. The navigation bar should be accessible from most pages and start from the home page by default. Each button or link in the navigation bar should direct the user to the correct corresponding page.
AC: Once the bar is created, it should be at the bottom of every page except for sign-in and sign-up.

Ticket Logistics
Assignee: Nikka Vuong
Estimated Points: 5
Actual Points: 3
Branch: https://github.com/SAJacob7/Capstone_Team_Generated_Group30_Project/tree/nikka-navigationbar
Extra Notes: This is Requirement 3.6.

Requirement ID 4.1: Use Tensorflow to Select the Recommendation Engine for App

Ticket Details
Description: Do the initial research into how to do a recommendation model in a mobile app. Figure out how to store data into Firebase, query it, then bring up the data to train the engine, and integrate into the app.
AC: Create a very detailed plan to follow for implementation of this recommendation engine.

Ticket Logistics
Assignee: Sophia Jacob, Kusuma Murthy, Nimra Syed
Estimated Points: 8
Actual Points: 5
Branch: Plan in Project_Documentation Folder.
Extra Notes: This is Requirement 4.1.

Requirement ID 4.2: Find Relevant Dataset on Kaggle

Ticket Details

Description: To train the model, a dataset is needed that will correlate as user input so the model can learn from it. There are specific parameters needed for a dataset, such as columns that can correlate to the questions we asked the users about their preferences.
AC: The dataset should be finalized, downloaded, and put into the GitHub.

Ticket Logistics
Assignee: Sophia Jacob, Nimra Syed, and Kusuma Murthy
Estimated Points: 5
Actual Points: 5
Branch: We decided to not use Kaggle and prepared our own data which fit our needs better than the Kaggle dataset.
https://github.com/SAJacob7/Capstone_Team_Generated_Group30_Project
Extra Notes: This is Requirement 4.2.

Requirement ID 4.3: Build the Model Architecture

Ticket Details
Description: Create the parameters and imports needed for TensorFlow, initializing it (if not already done), and setting up the basics for it to start training on the Kaggle dataset.
AC: TensorFlow is properly initialized and the necessary parameters and training setup is done.

Ticket Logistics
Assignee: Sophia Jacob, Anna Lin, Kusuma Murthy, Nimra Syed, and Nikka Vuong
Estimated Points: 5
Actual Points: 5
Branch: https://github.com/SAJacob7/Capstone_Team_Generated_Group30_Project
Extra Notes: This is Requirement 4.3.

Requirement ID 4.4: Train the Engine Based on Necessary Datasets

Ticket Details
Description: Based on the acquired datasets, train the model to provide recommendations to users based on their profiles.
AC: The acceptance criteria are to train a model and ensure model accuracy through some testing.

Ticket Logistics
Assignee: Sophia Jacob
Estimated Points: 8
Actual Points: 8
Branch: https://github.com/SAJacob7/Capstone_Team_Generated_Group30_Project

Extra Notes: This is Requirement 4.4.

Requirement ID 4.5: Integrate Recommendation Engine using TensorFlow Lite

Ticket Details
Description: Integrate the well-trained AI Recommendation Engine into the mobile app using TensorFlow Lite.
AC: The acceptance criteria are to fully integrate the two components and ensure functionality.

Ticket Logistics
Assignee: Sophia Jacob
Estimated Points: 8
Actual Points: 8
Branch: https://github.com/SAJacob7/Capstone_Team_Generated_Group30_Project
Extra Notes: This is Requirement 4.5.

Extra Requirement 1: 30: Update Styles for Profile Setup Page

Ticket Details
Description: Update the styles for the Profile Setup page to match the Figma.
AC: The acceptance criteria is to make the UI clean.

Ticket Logistics
Assignee: Anna Lin
Estimated Points: 1
Actual Points: 1
Branch:
https://github.com/SAJacob7/Capstone_Team_Generated_Group30_Project/tree/style_updates
Extra Notes: This is an extra requirement we completed in the Sprint.

Extra Requirement 2: 36: Begin Backend

Ticket Details
Description: To be able to run the model created on Expo Go, we needed to create a hosted backend that will load the compressed model, take in a vectored input, run the model, and get a JSON version of the output.
AC: For this ticket to be completed, it should be able to take in a vectored input and if we are running it locally, it should take in a curl command like this: curl -X POST
http://127.0.0.1:5000/recommend
-H "Content-Type: application/json"

-d '{"query_vector":[3,1,2,0,4,2,1,0,1,0]}'
and query the model to give a JSON output of recommendations.

Ticket Logistics
Assignee: Sophia Jacob
Estimated Points: 5
Actual Points: 5
Branch: https://github.com/SAJacob7/Capstone_Team_Generated_Group30_Project
Extra Notes: This is an extra requirement we completed in the Sprint.

Extra Requirement 3: 37: Host Backend through Render

Ticket Details
Description: Since the backend needs to be hosted to integrate into the mobile app, we need to host it on a free platform. We chose Render and we need to ensure that GitHub is linked to Render and that it is running live with no errors.
AC: Once Render has hosted the backend, it should be able to generate recommendations on the app.

Ticket Logistics
Assignee: Sophia Jacob
Estimated Points: 3
Actual Points: 5
Branch: https://github.com/SAJacob7/Capstone_Team_Generated_Group30_Project
Extra Notes: This is an additional requirement.