Project Proposal
Quick Scan

Project Overview:

Quick Scan is a mobile application for Android devices with the purpose of finding pricing and availability information of a desired product. Barcodes can be scanned from your mobile device's camera and you will be able to see product analytics related to it. These analytics will include stores within your local radius that sell the product, as well as websites that sell the product and their corresponding prices. Quick Scan will also be able to show reviews and pricing information of a product so that you have sufficient information to make an educated purchase.

Sprint 1

Sprint 1 Objective:

Enable the user to sign-up/login to the application and and be able to scan a barcode to display the product and information regarding that product.

Sprint 1 User Stories:

1. I, as a user, want to log into the application to make use of it's capabilities:
    a. Create the login UI through Android Studio widgets
    b. Deploy locally on Android Studio emulator
    c. Authenticate users using Firebase API integrated with Google accounts
    d. Local unit tests to ensure that the UI and login authentication is valid
    e. Set up a database to hold users' data using Firebase

2. I, as a user, want to scan a barcode from a product:
    a. Create UI for barcode scanning through Android Studio widgets
    b. Integrate Google barcode API to enable barcode scanning
    c. Local unit tests to ensure the device's camera functionality and barcode scanning capabilities.
    d. Deploy locally on Android Studio emulator
3. I, as a user, want to receive information about the product:
    a. Send a request to Google's barcode API
    b. Use the barcode API to process the scanned barcode
    c. Query product's information with our searching algorithm
    d. Retrieve information based on defined function requests
    e. Test the parsed information accuracy, as well as the search algorithm results

    f. Deploy locally on Android Studio emulator

4. I, as a user, want to display the information about the product:
    a. Retrieve information from Google's barcode API
    b. Create UI through Android Studio widgets to display the information received on the screen.
    c. Test UI design and display capabilities
    d. Deploy Locally

Sprint 2

Sprint 2 Objective:

Enable the user to locate the item within a region around them and list the prices of the items at different locations.

Sprint 2 User Stories:

1. I, as a user, want to scan a product to see availability and pricing in my local area:
    a. Retrieve proper information from barcode API
    b. Utilize Google Maps API to retrieve local geographic information
    c. Display geographic information to user through Android Studio UI
    d. Test geographic information to ensure accurate representation through field testing
2. I, as a user, want to scan a product to see availability and pricing online:
    a. Obtain information through barcode API
    b. Use information to parse online availability and pricing
    c. Display availability and pricing through UI
3. I, as a user, want to view my scan history, to see product details from previous scans:
    a. Make an API request to see a user's past scans within Firebase DB
    b. Perform a query utilizing NoSQL to display requested information
4. I, as a user, want to view my scan history, to delete products from previous scans:
    a. Make an API request to information from the Firebase DB
    b. Perform a query utilizing NoSQL to display requested information
    c. Display the query to the user and prompt the user to delete the requested information
    d. Send a notification to the user that the requested information has been deleted
    e. Send a request to the Firebase DB to delete the scan history within a user's account

Sprint 3

Sprint 3 Objective:
Allow the user to view ratings and reviews of the scanned product and be able to show nearby locations to purchase products

Sprint 3 User Stories:

1. I, as a user, want to scan a product to view reviews on that product.
    a. Make request to the barcode API
    b. Retrieve product information to perform algorithm search
    c. Display the information to the user through Android Studio UI
    d. Test the UI and search capabilities
    e. Deploy Application through Google Store.
2. I, as a user, want to share low prices that I find with my friends and contacts.
    a. Access the user's contacts through Android permissions.
    b. Run a request to find contacts that also have the application installed.
    c. Deploy a messaging system through the backend network java code base.
    d. Allow user to message contacts through Android Studio UI.
    e. Test peer to peer capabilities within our app, as well as the UI.
    f. Deploy Application through Google Store.
3. I, as a user, want directions to one of the provided store locations from the scan.
    a. Access the user's location via device's gps through backend code base.
    b. Use Google Maps API find and get locations to the selected store.
    c. Display directions through Google Maps API to the Android Studio UI.
    d. Test Google Map API capabilities and UI display.
    e. Deploy through Google Store.

Sprint 4

Sprint 4 Objective:
Allow user to compare prices and be able to locate where the item is at the lowest price.

Sprint 4 User Stories:

1. I, as a user, want to find the cheapest price for a product.
    a. Enhance web searching algorithm to compare prices listed online and in stores around the user.
    b. Display the lowest priced item to the user through the Android Studio UI display.
    c. Test searching algorithm, and UI display.
    d. Deploy through Google Store
2. I, as a user, want to see similar products to items I have scanned.

a. Access the Firebase Database for products with the same scan code.
b. Implement product comparison algorithm.
c. Display similar products from the query result.
d. Test Query and product comparison algorithm for accuracy. Test UI display.
e. Deploy through Google Store

3. I, as a user, want to see popular scanned items
   a. Access the Firebase DB to see which items have been scanned most frequently
   b. Obtain the most frequently scanned items using a NoSQL query
   c. Create a display for popular products using Android Studio widgets.
   d. Display items to the user.