

CS 408 Team 10

Kenny Tam

Milia Enane

Michael Rollberg

Jake Losin

Brandon Nice

Product Backlog

Problem Statement: I would like to know how my local community's opinions align with my own but have no means of acquiring such specific and undocumented information.

Background Information: Individuals are often curious about the opinions of others, but there does not exist a program that would allow the user to find this information easily. This is an issue because public opinion can be used in marketing decisions, advertising campaigns, and is generally interesting. As a team, we will create an Android application that will ask the user questions and store the results in a database. After answering a question, the user will be able to view the collective answers of the public.

Environment: Our application will be an Android application coded in Java. We will be using a database, which will be selected later, to store user inputs. We will also implement Google Maps for its location based services and use Facebook API to keep track of user login IDs and secure login.

Functional Requirements:

Backlog ID	Functional Requirements	Hours
1	As a user, I would like to sign in using Facebook.	2
2	As a user, I would like to answer questions immediately whenever I open the app.	10
3	As a user, I would like to avoid seeing the same questions that I have already answered.	3
4	As a user, I would like to primarily see the questions that have the most popularity based on how many people have seen and answered them.	10
5	As a user, I would like to see the answers of other people around me based on my location presented in the form of a chart or graph.	20
6	As a user, I would also like to see the results of other cities and states as well as the entire country.	10
7	As a user, I would like to select question topics based on categories such as sports and science.	8
8	As a user, I would like to create questions that are capable of having either two answer choices or more.	8
9	As a user, I would like to be able to create and submit my own questions that can be seen and judged by other users.	13
10	As a user, I would like to be able to choose between user created and moderator created questions.	10
11	As a user, I would like to go through a list of all the questions I've answered before and re-examine results.	5
12	As a user, I would like to have a settings page where I can logout, and control what type of questions I see.	12
13	As a programmer, I would like to allow users to flag user made questions to decide whether or not they are removed.	5

14	As a programmer, I would like to implement a simple UI that will easily allow users to answer questions quickly.	15
15	As a programmer, I would like to keep a database to store the user's information regarding question.	9
16	As a programmer, I would like to provide an optimal search for the user to filter and search questions by topic or key phrases.	10
	Total:	150

Non-Functional Requirements:

- Our users' login information will be secure with our implementation of Facebook login.
- Our application will require access to the internet, so when connection is lost, we will display a dialog so that the user is aware that they need to reestablish connection.
- User inputs should be able to be stored at the same time so our database has no conflicts with concurrent access.
- Results for searches should load quickly and display relevant results according to user input
- Allow a large amount of users to use the app at the same time while still keeping accurate profile data for each one.
- Prevent pages from crashing or possibly even corrupting user data.

Use Cases:

Case: Facebook Login

1. Open the app
3. Enter login credentials

System Responses

2. Facebook login dialog displays
4. User is logged in and dialog disappears

Case: Answer questions immediately

1. Log in using Facebook credentials

2. System instantly displays a question

Case: Answer popular questions

1. User opens app
3. User starts answering questions

2. System sorts questions based on popularity
4. Questions with the most views and votes appear

Case: Answer location based questions

1. User opens app
3. User logs in

2. System obtains user's location
4. System displays popular question based on user's location

Case: Filter out question topics

1. User taps on drawer in top corner
3. User taps the "filter" setting categories
5. User selects the categories to exclude

2. System presents user with various settings
4. System displays a list of
6. Excluded categories don't show up

anymore

Case: See results from other geographic areas

1. User answers a question
2. System displays option to proceed to next question or to view world, state, or country results
3. User selects "World results"
4. World results are displayed

Case: Vote on questions to be removed

1. User starts to answer a question
2. Like/dislike button will be displayed at the bottom of the screen
3. User selects "dislike"
4. System will record the dislike and will be made visible to us for further investigation of the question

Case: Question templates

1. User selects "Create Question" from the menu
2. System asks user how many possible answers they would like
3. User picks, for example, 3 answers
4. User is provided with a 3 answer template

Case: No duplicate questions

1. User answers a question
2. System will record that question and add it to a list so that the user does not see it again

Case: Select more than one answer choice

1. User creates a question
2. User is given the option to allow others to choose more than one answer to a question
3. User elects to allow for multiple answers
4. People who answer the question can pick multiple answers

Case: Create your own questions

1. User presses the settings button
2. The settings are revealed
3. User selects "create question" button
4. Create question screen is shown
5. User finalizes their question and presses "submit"
6. The question is added to the question pool

Case: User and moderator created questions

1. User presses the settings button
2. The settings are revealed
3. User presses "user/moderator"
4. The app will only show questions created by the user or a moderator, based on the option selected

Case: View previously answered questions

1. User presses drawer in the top corner
2. Settings, Profile, etc. are revealed
3. User selects "profile"
4. System gives the user the option to view previously answered questions

Case: Graphs and plots

1. User answers a question
2. Results page is displayed, containing information about collective results like a pie graph and numerical statistics