

Team Assignment #3: Implementation
Polling Software: RateTheBeach
CECS 443 (Section 01)

Team Members:

Joey Rice
Benjamin Nguyen
Kenneth Su
Daryl Nguyen
Naz Diam
Chelsea Aguilar
James Ta

1. Project Title and Authors

a. Your team name

Rate the Beach

b. A list of all team members:

- [Chelsea Aguilar] (<https://github.com/chelcx2>)
- [Naz Diam] (<https://github.com/yichii>)
- [Benjamin Nguyen] (<https://github.com/nguyenbenjamin3>)
- [Daryl Nguyen] (<https://github.com/nguyendaryl>)
- [Joey Rice] (<https://github.com/Ricearoni26>)
- [Kenneth Su] (<https://github.com/ksu4>)
- [James Ta] (<https://github.com/JamesTa2000>)

2. Preface:

This project proposal provides the necessary information about the contents of the mobile app called RateTheBeach and how it will be deployed as a survey, polling and review system using the latest security standards. The source code for the project will not be provided directly in this document but instead, will be provided in the following github repository down below. (We will notify the customer when we have developed a working prototype for the system as there is currently none at the moment)

The following project proposal is written in response to our client, Daniel Link, who has requested a “Survey App” on February 14, 2023 at 6:30 PM. The client has expressed that there is currently a lack of a reputable surveying app in California State University Long Beach that is why there is such a need for the app in the first place.

The requested app is described to be “...an app for CSULB students that provides public ratings and votes. Using the app, users can rate anything from food and coffee to classes and labs. Other users can see current ratings or votes of any item. Anyone with a valid CSULB ID can post reviews/ratings on all sorts of entities - courses, professors, labs, events, food, cafes, concerts, etc. The system will keep its users anonymous to the public.”

Github repository:

Version 3.0 <https://github.com/nguyenbenjamin3/RateTheBeach>

Link to README file:

<https://github.com/nguyenbenjamin3/RateTheBeach/blob/master/README.md>

3. Introduction:

The need for the app is due to the lack of reputable surveying apps currently in use on campus as previously mentioned. Specifically, the need for an app where reviews and opinions are shared regarding campus related topics has arisen as mentioned by the client. That is the reason for the inception of the idea of making such an app has come about as stated by the client.

The system functions as a platform for verified students at CSULB to participate in the polling/rating of various subjects and categories of topics (i.e. courses, professors/instructors, events, food vendors or options, etc.), as well as participating in the creation of polls and rating posts. Students will have freedom to also interact with the posts, by upvoting and downvoting the posts, commenting on posts, and sharing them across different social platforms via a link.

At the base of this application, the motivation for its creation is to create a community monitored and developed by both campus officials and students and hopefully encourage a hub of ideas and opinions to form that can guide students and provide peer-to-peer information.

4. Architectural Change

a. What architectural changes did you make during implementation?

As of now, we have not made any changes to our architectural design. Our current design follows the MVC pattern, which we chose because it offers a comprehensive framework for organizing the different components of an application and defining their roles and responsibilities. This pattern is versatile enough to be applied to a wide range of applications, regardless of complexity, and it allows us to easily understand the

various components by dividing them into the Model, View, and Controller. Our model includes rating and polls, and our controller and view are linked to a Firebase database. The rationale behind our decision to use the MVC pattern was to ensure flexibility in designing the specific implementation details of each component, making it applicable to a wide range of applications, from simple to complex.

b. What's the rationale behind the decision change?

We did not make any architectural changes during implementation as we believe that the current design is well-suited to our needs. The rationale behind our decision to use the MVC pattern was to ensure flexibility in designing the specific implementation details of each component, making it applicable to a wide range of applications, from simple to complex. The pattern also allows for easy understanding of the various components by dividing them into the Model, View, and Controller. Our app includes a rating feature that allows users to rate different items, and a polling feature that enables users to create and participate in polls on various topics. These features are designed to be user-friendly and engaging, and are geared towards students. The app also includes user authentication, a user profile page, and a search function, all of which are designed to enhance the user experience.

To implement these features, we are using Firebase as our backend database, which allows for real-time synchronization of data across different devices and platforms. This means that users can access the app from their phones, tablets, or computers and see the same data regardless of the device they are using.

Overall, our app is designed to be organized and easy to maintain, even as we continue to add new features and functionality over time. The use of the MVC pattern helps to ensure that the app is well-structured, and allows for flexibility in design to accommodate a variety of use cases.

5. Detailed Design Change

a. What detailed design changes did you make during implementation?

During our implementation of the project, we made some detailed design changes to enhance the overall user experience. One of the changes we made was moving the Reload button from the bottom to the top, as we noticed that the nav bar would cover the button at the bottom. We also decided to opt for a card style view for each post instead of the sliding cards view, as we realized that the sliding cards view would take more time to implement than we had available.

Another change we made was to streamline the onboarding process by opting for just one screen instead of multiple screens for login and outlook login. This was because we wanted to prioritize user experience and make it as easy as possible for users to get started with the app. Finally, we decided to display the users' username in the middle of their profile page instead of at the top, as we did not have enough time to properly style the profile page.

b. What's the rationale behind the decision change?

The decision to move the Reload button to the top was based on our desire to ensure that the button was easily accessible to users, regardless of the device they were using. We also wanted to make sure that the button was not obscured by any other elements on the screen, such as the nav bar.

The decision to opt for the card style view instead of the sliding cards view was based on our desire to optimize our use of time and resources. We realized that implementing the sliding cards view would take more time than we had available, and the card style view still allowed us to present the posts in an appealing and user-friendly manner.

The decision to streamline the onboarding process by opting for just one screen was based on our desire to prioritize user experience and make it as easy as possible for users to get started with the app. We wanted to reduce any potential friction points and make it as easy as possible for users to create an account and start using the app.

Finally, the decision to display the users' username in the middle of their profile page was based on our desire to ensure that the profile page was still functional and presentable, even though we did not have enough time to properly style it. This allowed us to present the important information to users in a clear and concise manner.

6. Requirement Change

If you have changed any of the requirements since your assignment 1 and 2, explain the following:

Some requirements that we have changed or removed from assignment 1 and 2 are:

- **Follower/Following Frame**
 - Users are able to click on their follower count, which will redirect them to a page with list of users they follow
 - Users can unfollow others from this page
 - Click on a user's username will redirect them to that user's profile page
 - Users are able to click on their follower count, which will redirect them to a page with list of all users following them
 - Users can follow their followers back from this page
 - Click on a user's username will redirect them to that user's profile page
- **Home Frame**
 - When the Create a Poll Button is pressed, the user is taken to the Create a Poll Page
 - In the recent polls section, the user is able to swipe left or right to view any recent polls that other users have posted

1. Does this change your design?

- **Follower/Following Frame**
 - Yes, we removed the whole follower page entirely which affected how the nav bar looked.
- **Home Frame**

- Yes, we have removed the swiping feature. The screen has been expanded vertically and users can scroll down on the page.

2. If not, why not?

- Some of the requirements that have been removed are the ones that did not change our design because we did not implement anything in the first place. Since we have decided that we would not have enough time to complete it therefore, we did not start on it in the first place.

3. If so, what should be changed and how?

- Both the follower and following frames should be changed. Since we removed the whole follower page entirely which affected how the nav bar looked since instead of 5 icons, it would only contain four icons since we would only have the homepage, saved page, search page, and log out page. We are also changing the home frame as well by removing the swiping feature. Instead, we are expanding the screen vertically, allowing users to scroll down to view more posts.

4. Has this new requirement been implemented? Or will it be implemented in future?

- The new requirements have been implemented inside our app and can be seen on the home screen and the nav bar.

Here are our attached System Requirements for reference:

User Requirements Definition:

Functional Requirements:

1. The Polling Software shall enable the user to generate a poll showing a variety of opinions or rating of a post.
2. The Polling Software allows the user to register with a valid campus ID.
3. The Polling Software provides the user a unique link to share the polling / rating post with others.
4. The Polling Software shall allow users to interact with polling and rating posts (**agree**, **disagree**, commenting, and viewing).

5. The Polling Software shall allow creators to set start and end dates for polling and rating posts, giving reviewers a set duration of time to participate in them.
6. The Polling Software shall allow easy navigation throughout different frames using the **nav bar**.

Nonfunctional Requirements:

1. Upon account creation, the user must verify that they are currently a CSULB student using their campus email.
2. Account verification will be done through the **Outlook Verification** process that students use when accessing campus software.
3. After a certain number of login attempts, the user's account will be temporarily locked to prevent further login attempts. They can unlock it by resetting their password.

4. System Requirements Specifications:

1. Functional Requirements:

a. Login Frame

- i. Upon opening the Rate The Beach app, users are taken to the **Login Page**
- ii. When users are directed to the login page, they are required to put in a valid email address and password in order to login
- iii. When users press the **Login Button**, the system then checks whether the provided email and password matches with any existing account

1. When an account is found, the user is then taken to the **Main Page**
 2. When an account is not found, a text is prompted in the **Login Page** saying that the “email/password is invalid” or something along those lines
 - iv. When the user presses the **Sign Up Button**, they are taken to the sign up page
 - v. When the user presses the **Sign in with CSULB Button**, the user is taken to the **Sign in with CSULB Page**
 - vi. Firebase Authentication will be used to authenticate users
- b. Sign Up Frame**
- i. Users are asked whether they are a student or an alumni(they will not be able to register if they are neither
 - ii. They are then able to view a page which asks them to input a last name, first name, CSULB email address, password, and a reentered password to make that users are inputting the password that they desire
 - iii. When users press on the **Sign Up Button**, the system checks whether they have provided a valid email first after that the system makes sure that the user has inputted a valid password
 - iv. An additional requirement would be to have the password match the second password that they inputted
- c. Rating Frame**
- i. Users can create a rating tile/post.
 1. This includes writing the contents of the poll
 - ii. Users would need to create polls that would be able to be rated from 1-5.
 - iii. The User will be able to set a time for the poll to start and end
 - iv. After the poll is created, the user will be navigated back to the home screen
- d. Polling Frame**
- i. Users can create a polling tile/post.
 1. This includes writing the contents of the poll
 - ii. User can set parameters to a custom boolean value or the default will just be a yes or no

- iii. The User will be able to set a time for the poll to start and end
- iv. After the poll is created, the user will be navigated back to the home screen

e. Home Frame

- i. Upon reaching the home frame, users are able to view an introduction message, a profile icon, active polls that they are a part of and as well as recent polls on the platform
- ii. When the **Create a Poll Button** is pressed, the user is taken to the **Create a Poll Page**
- iii. When the **Expand button** on the active polls section is clicked, the user is taken to the **active polls page** where they will see all the active polls that they are part of
- iv. In the recent polls section, the user is able to **swipe** left or right to view any recent polls that other users have posted
- v. The nav bar will also be visible on this frame

f. Navigation Bar

- i. This will contain a home button, likes icon, search icon, and profile icon.
- ii. The home button will take the user to the **home frame**, **liked icon -> liked frame**, **search icon -> explorer frame**, **profile icon -> profile frame**

g. Follower Frame

- i. Users are able to click on their follower count, which will redirect them to a page with list of all users following them
- ii. Users can follow their followers back from this page
- iii. Click on a user's username will redirect them to that user's profile page

h. Following Frame

- i. Users are able to click on their follower count, which will redirect them to a page with list of users they follow
- ii. Users can unfollow others from this page
- iii. Click on a user's username will redirect them to that user's profile page

i. Profile Frame

- i. On the profile frame, users are able to view their own profile

- ii. Information displayed
 - 1. User's profile icon
 - 2. User's display name
 - 3. User's secondary name (i.e. Full first and last name); not required
 - a. User has the ability to change this name
 - 4. Count of followers
 - a. clicking will redirect users to follower frame
 - 5. Count of those the user is following
 - a. clicking with redirect users to following frame
 - 6. User's active posts (rating, polling)
 - a. User can access their own rating and polling posts from their profile page
 - b. They are able to delete their own rating and polling posts from this page

j. Explore Frame

- i. Users will be able to see other Polls tiles and Ratings tiles from other users at the same school
- ii. The different tiles will be displayed in a stack that the user will have the option to swipe left or right (indicating a response of no or yes respectively).
- iii. For the Rating tiles, the user will be able to select a rating value (set by the creator of the poll–Max 5, min 1).. A left swipe on these tiles will indicate a 1 and a right swipe will indicate a 5
- iv. The active user will then be able to like, interact, or share the poll tile or rating tile
- v. The User will have the option to return to the home screen if they are done rating polls.
- vi. If no polls remain, the last item in the stack will be a tile that indicates the user should return later once more tiles (rating tiles or polling tiles) are created.

2. Nonfunctional Requirements:

A. Performance

- a. When entering an email address in the **Email Line**, users are able to type in any character(takes in String)
- b. This applies as well for the **Password Line**
- c. Login process should take no longer than 4 seconds to complete.
- d. There should be reasonable speed and efficiency when the app is being used under load.
- e. Polling and rating posts should appear to the home page in efficient manner, with no extended loading times

B. Scalability

- a. The app should be able to handle increased and decreased usage by optimizing the way storage is handled.

C. Responsiveness

- a. The state of the page should be saved and users should be able to return to it if the user's mobile device is interrupted by a core process such as an incoming phone call, an alarm, or etc.
- b. Depending on the users' viewport, the mobile app should be able to change its dimensions according to it.

D. Security

- a. The users' data should be protected using encryption to ensure that just in case a breach were to happen, decrypting their data would be difficult to accomplish.
- b. When inputting a password in the login page, the password should be hidden using asterisks to increase security.
- c. User passwords must meet the microsoft password guidelines.

E. Availability

- a. A user should be able to access the app at any time as long as they have a reliable internet connection.

3. UI Appearance Requirements

● Interactive Elements

- The UI of the interactive buttons and icons allows the users to respond to the postings whether it is upvoting, replying, downvoting, following, etc.
- The purpose of the UI of Interactive elements is to have an engaging experience when using the app thus connecting the user to other users via interaction through the poll or the following feature.

● Intuitive NavBar

- The UI of the NavBar should allow users to navigate throughout the app and find content based on the navigation bar.

- The requirement of the NavBar is to help users move between different pages on the application without being lost on which page to go to next. The NavBar allows the user to find exactly what they are looking for by effortlessly clicking on an icon.
- **Simplistic Responsive Visual Design**
 - Vibrant colors alongside with clear font and readability enables the User to have a pleasant experience on the app.
 - The screen should be designed to adapt to different screen sizes, resolutions, and orientations.
- **Clarity**
 - The UI should be easy and read to understand for the users
 -