TEAM: Shining Sky

Overview: The main purpose of the "Technical Demos" is to very clearly communicate the extent to which the team has identified key challenges in the project, and has proven solutions to those challenges. Grading is based on how complete/accurate the list of challenges is, and how convincingly and completely the given demos cover the given challenges.

This template is fleshed out by the team, approved by the CS mentor, and brought to demo as a grading sheet.

Risky technical challenges

Based on our requirements acquisition work and current understanding of the problem and envisioned solution, the following are the key technical challenges that we will need to overcome in implementing our solution:

<Concise 1-3 sentence description of what exactly might be challenging here. Then *state exactly what, in principle, you'd have to show/demo to prove a good solution*.>

C1: Framework. The ARORA mentorship app will need to be built using an app development framework, which is an existing library of tools and functions which eliminate the need for creating the app from scratch. The React Native framework has been selected for this app. To prove the functionality of the framework, a simple application could be created, with multiple pages between which the user can navigate.

C2: Framework Platform. The chosen platform for the framework needs to utilize its own elements to add support for functionality along with the capabilities of the platform itself. The platform should support a multi-platform exportation system, support an infinite scrolling list, a chat system, and the ability to access data through server integration.

C3: Database. The ARORA mentorship app needs to be able to communicate with the ARORA server to transfer important information. Setting up an account, logging in, messaging, answering questions, and viewing mood reports, all rely on a sturdy backend database. To prove the database is functional, a mock profile would be created and shown as existing on the ARORA server. Then the profile would be manipulated (username change) and the changes would be shown again on the server.

C4: System Integration. An important part of the ARORA mentorship app is the API component for the server, as well as the complementary need that the app must interact properly with the server. As the server is something created and maintained by the client, developers are not familiar with it and will need to learn how to use it to properly implement and test the new addition to the ARORA system. Since server code and some setup instructions have been provided by the client, an adequate demonstration of the ability to use the server would be starting it up and performing a few interactions with it.

Challenges covered by demos:

In this section, we outline the demonstrations we have prepared, and exactly which of the challenge(s) each one of them proves a solution to.

Demonstration 1: Navigating and Using the App's Utilities

Challenges addressed: Framework, Framework Platform,

Flight Plan: Step by step overview of the demo

1. Open to login screen
2. Log in to app with mock account information of username and password
3. Open to home screen displaying:
    a. Clickable mentee names
    b. With each respective mentee name, an indicator in terms of how in need the mentee is
4. Click a mock mentee
5. Open to a mock mentee page with their respective mock information including:
    a. A few mock mood reports done by the mock mentee
    b. Button to flag the mentee for potentially being in need
    c. Button to start a chat with mentee
    d. Button to view calendar/ appointments with mentee
6. Click chat button with mock mentee, open to chat page
7. Return back to mentee page
8. Click calendar button with mock mentee, open to calendar page
9. Navigate to question list system using toolbar
10. Open to question page with
    a. A search area
    b. A scrollable list of questions
        i. For each question, a button for flagging the question
        ii. An area to leave a comment or answer the question
11. Scroll through the list system showing mock questions
12. Leave a mock response on a question through a comment section underneath the question
13. Navigate back to home screen using toolbar

Evaluation:

- ✔ Convincingly demoed each of the listed challenges?

- ✔ Other evaluative comments:

Demonstration 2: Testing Server

Challenges addressed: System Integration, Database

Flight Plan: Step by step overview of the demo

1. Pre-Install server before demo.
2. Start server via command line.
3. Show that the server has started correctly and then explain any error messages.
4. Open the corresponding server web interface.
5. Show users table as an example.
6. Add an additional user to the database.
7. Shut the server down.

Evaluation:

- ✔ Convincingly demoed each of the listed challenges?
- ✔ Other evaluative comments:

Other challenges recognized by not addressed by demo:

If there were challenges you listed earlier that were *not* covered by a demo, list them here. This will hopefully be a shortlist…but better to be clear about where you are. If you have items here, you could list (if applicable) any pending plans to reduce these risks.

- Implementing a small and simple web application for client-side sending of anonymous questions.
- Showing the exportation of a single implementation into iOS, Android, and a Web App at once.