Milestone 3: System Prototype and Evaluation Plan

CPSC 4140/6140

Byte Force Team 10

Ethan Waugaman, Steven Spivack, Vincent Hung, Delaini Daughenbaugh

November 7, 2025

Usability Specifications

Quantitative Benchmarks:

Due to the time sensitive nature of our problem space, efficiency and accuracy of task performance and completion are of utmost importance. To test this we intend to utilize a few different metrics to measure user experience, task efficiency, success rate, and learnability. By nature, we want our program to be an educational and instructional tool, so making the interventions memorable and the interface easy to navigate is essential. At this stage in our program development, and due to developer licensing restrictions, we are unable to get user data from the general public. However, some of our team members are able to load the application on their phones and have access to groups of individuals who may be considered secondary stakeholders. Our goal is to deploy the program to a group of nursing students to see how they interact with the program as both potential primary and secondary users. For quantitative benchmarks, we can conduct time on task trials to collect data on how long it takes users to complete expected tasks. Alongside this, we can measure the success rate of them completing the outlined tasks such as moving through the slideshows, creating a profile, and orienting themselves with the interface. To do this, we can manually time the users to determine on average the duration of task completion as well as the success rate. Additionally, could potentially integrate code that creates a CSV file that will quantify these statistics for us, as well as adding in Fitts' law metrics to measure click time and click count. Ideally, we would also be able to test how effective the users are at completing the outcome task of using the program, which would be successfully performing the intervention that is being displayed. However, this benchmark may not be achievable due to the sensitive nature of the intended problem space, the access to voluntary participants, and the feasibility of conducting these kinds of benchmarks. Instead, we will focus on the performance of user-interface tasks outlined by the measures stated previously.

Quantitative Benchmarks:

For qualitative benchmarks, we will use focus group feedback on the design and implementation of the program. As we present in class, we will use the feedback from our peers to help gauge the effectiveness of our design and the usability of our program. Ideally, this would be from the perspective of primary users, as the target audience of our program could include but

is not limited to our classmates, as anyone who may want to have helpful information for emergency situations can be considered a primary user and benefit from our program. We can also conduct some open-ended interviews or surveys with our peers to get some feedback on features they like, don't like, or would like to see in future versions. Ideally, we would also like to gather feedback from a group of nursing students by having them interact with the product via one of our team members. Nurses (or nursing students) could be considered either secondary or tertiary stakeholders, as they could potentially interact with the application, or primary use of the program could affect them when the endangered individuals are eventually brought into the hospital. We would also like to conduct some surveys and polling on these users to see what elements they think are both useful and important to properly caring for patients/individuals in emergency situations. Since a team member needs to be present for preliminary user testing, additional qualitative benchmark data that we can collect includes user observation. This can be done for all stakeholders, as we can observe users directly interacting with the app to examine the ease of use and their conceptual models in understanding how to complete tasks within the interface. During observations, we can also gather quantitative data such as timing tasks and overall use or interaction with the UI, while also gathering behavior observation data. These behavioral metrics can consist of patterns or habituations, such as how many times they revisit a specific page, tab, or intervention. Additionally, we can observe body language such as facial expressions to assess the effect of the user, to see if the workflow of the system is intuitive or causes frustration or confusion.

- **A discussion about the implementation challenges you faced. Were there aspects that you wanted to build but were unable to do so? Why was this?**

There were a few aspects of our design that came with some challenges. In our initial brainstorming and design phase, we wanted to have a login screen at the start up of the app. We did not move forward with this feature because it would have necessitated also building a database, which proved to be more difficult and time consuming than we anticipated. To curb this, we are assuming that at this stage of the design, the program is in a logged in state. Another feature we wanted to include was a search bar to allow users to easily search for content or slideshows that they need instead of scrolling to find the information. While we do have several

functions included in this preliminary prototype, we do not have enough information in the program to include a search bar, so we decided to revisit that feature to reconsider implementing it in future versions of the program. Another key function we wanted to include in our original design was a text to speech feature to help streamline the search process as well. While we were able to get a text to speech feature to be functional, it does not have voice recognition capabilities at this point in development. Additionally, we wanted to add a bookmark feature to allow users to save their frequently used resources. Unfortunately we faced similar challenges as the login page, where full functionality of the bookmark feature would require a database to save user information. At this stage of the prototype, we have a placeholder for the bookmark feature. We are interested in exploring adding a database for future iterations of the program to include fully functional login and bookmark elements. Finally, though we did not face any challenges with adding pictures and slideshows, we decided to make one tab fully fleshed out as an example, while having placeholders for the other intervention tabs. We decided that for the sake of time and simplicity of early prototype design, we would add more information to the other tabs in the future. This will also allow us to make any necessary changes to the slideshows easier, rather than needing to completely redo the content of the program.

- **A justification of why you choose the design that you implemented. What's special about this particular design with respect to your problem? What did you learn from Studio 2?**

Feedback from our peers from Studio 2 was overwhelmingly positive, where the majority of our peers agreed that the mobile app would be the most effective way to implement a solution to our problem. Many of them liked the simplicity of the UI design from our storyboards, and agreed that it would enable quick usage that will be helpful for using the app as intended in emergency situations. Others liked the website and stressed the importance of education prior to emergency situations. We agree that this is also a key piece of the puzzle, and we predict that the mobile app will double as both a pre-intervention education tool as well as being used during emergency situations. Finally, there were a few of our peers that appreciated the innovation of the medical bracelet and mobile app integration, but they were also quick to identify potential flaws in the design and implementation. Some of those flaws that were addressed included the ethics of mobile app integration with confidential medical information, as well as the added steps of verification that would be required by the users (medical professionals) in order to protect this

information. This added step could slow the user and the process of effectively completing tasks within the UI, which is critical in emergency situations. Comparisons to current technology were also highlighted, with some of our peers drawing on similarities to existing systems such as Apple and other smart watches health/medical tracking. After considering all of these suggestions, we decided that the mobile app would be the best direction to continue forward.

We chose to implement the mobile application design, because we concluded that it was the most practical way to maximize accessibility and usability of the product for our special problem. Since the problem we are trying to solve is largely focused on bystander education, but more specifically intervention, we needed to implement a version of our solutions that is easy to access anywhere. Since most people have access to a mobile device, we decided that mobile app compatibility was the most accessible and practical way to reach a large audience with diverse users. We wanted to make sure the app could be used in any situation, and decided the most effective way to do that was to focus more on the quick slideshow demonstrations that can be easily accessed on a phone to be used during an emergency situation. In designing the app, the goal was to make functionalities as easy to navigate and obvious as possible, to limit the amount of cognitive load the users would need to employ to use the app effectively. This is especially important given the external pressures we considered in our use cases, with the most likely user to be someone who is intervening in an emergency situation. Because of this, it was important to streamline tasks in the app to minimize interaction time with the interface, so users can feel more comfortable aiding others in these emergency situations where emergency service personnel may not be readily available. In accordance with Fitts' law, we wanted to make the buttons for the intervention tabs large enough that they were easy to see and click on, without making them so large that the information was lost to the user. All of the important information is displayed clearly and concisely on the main page with no scrolling needed. The slideshow intervention tabs are color coded, but they also include corresponding iconography as well as clear labels to help with visual accessibility considerations. As stated above, feedback from Studio 2 reflected these ideas as well, as our peers agreed the most viable implementation solution was the mobile app rather than an extensive website that was meant to educate users prior to being involved in an emergency situation. While the web-based solution was a viable option, given the time sensitive context of our problem space, we ultimately concurred that the quick service mobile app would be the best and most widely applicable in real world scenarios. Our alternative idea involving the

medical bracelet and app integration was also received well by our peers, but its practicality and implementation was not as realistic for this project space since it would have been necessary to also build the medical bracelet that was to interact with the interface.

Justification of why the technology you chose to implement your prototype was the best choice?

We chose to use React Native as the coding framework and Expo as the platform that supports our mobile application. First, React Native's component-based architecture supports modular, maintainable UI design, making it ideal for building interactive interfaces like our emergency procedure cards. React Native is similar to React and adaptable for either TypeScript or JavaScript which our team has experience with and able to speed up the development progress. In the early stage of the development, we discussed which we should focus on making implementation, Android or iOS devices. However, React Native provides cross-platform compatibility, which means the app is also fully functional on both Android and iOS systems.

Using Expo and Expo Go to display our mobile app and chose this platform also due to its cross-platform support. Expo also provides the ability to track user statistics and insights such as platform distribution, version data, and bandwidth usage — valuable metrics for testing and benchmarking. Additionally, Expo offers simplified setup, testing, and deployment by managing native dependencies and providing built-in tools for icons, navigation, and device APIs.

Alternative approaches such as native Android (Kotlin/Java) or iOS (Swift) development were considered in the research process but dismissed due to their higher complexity, limited cross-platform support, and longer development time that caused by the unfamiliarity our team have to these mobile specific languages. Likewise, web-based solutions like React.js alone would not have provided the native functionality or offline performance we needed for a reliable emergency procedure app. React Native with Expo therefore offered the best balance of performance, accessibility, and development efficiency for our project goals.

Response to Studio 3 Feedback

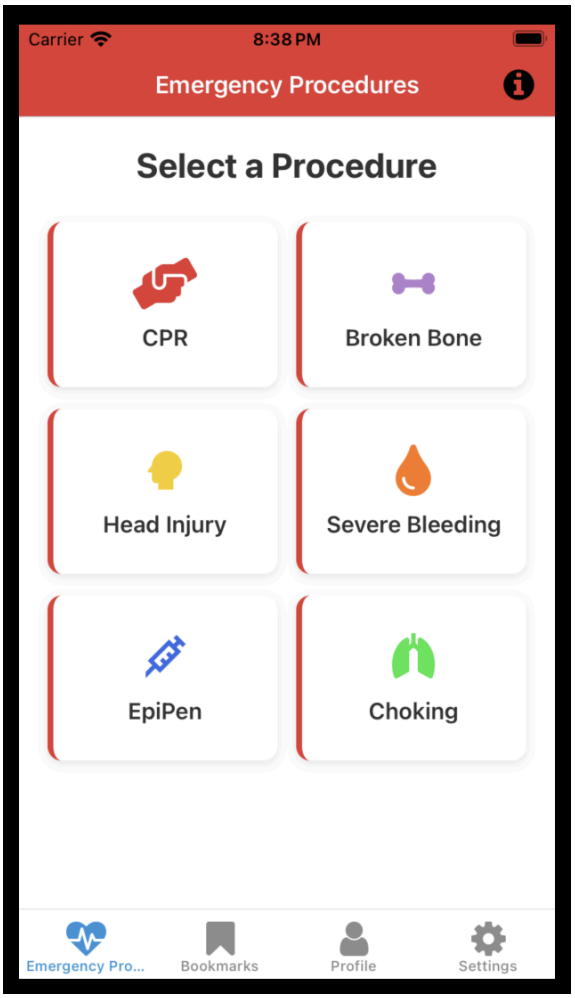
Upon completion of studio 3, our team came away with one implementation change we plan to address immediately. It was brought to our attention that the bookmark feature of our application did not fit the Emergent nature of our application. It was said that we were trying to tackle multiple problem spaces in one application through the use of this feature. We are currently in discussions on how to retain the bookmark functionality for more scholarly functionality (eg, keeping AHA guidelines, and similar publications), while also separating it from our core design goal. We have identified at least two ways in which we can accomplish this goal.

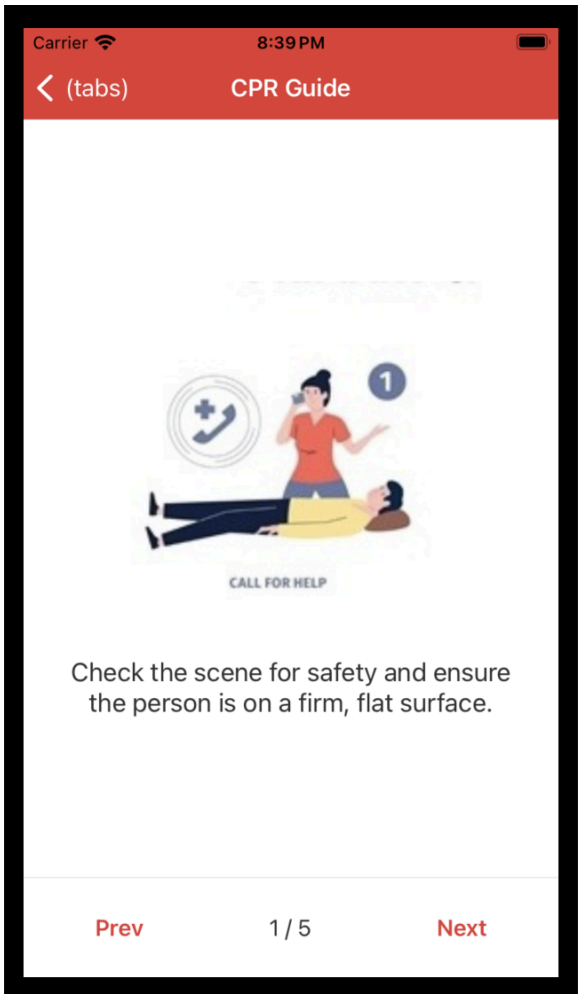
The first approach would be to have two separate modes, “Emergent” and “Research” modes where bookmarks would not render, nor lengthy articles would be shown when the application is running in the “Emergent” mode. The application would have a toggle switch, when activated navigation tabs would change to “Lessons”, “Resources” and “Bookmarks” while losing the “Procedures” tab. We believe if we are to implement this the correct way, the application should default to the “Emergent” mode, which would maintain the spirit of the current design.

The other method we are considering is to add another tab introducing a “Resources” feature that would operate similarly to how it would with the two mode approach. This way, the user would not have to switch between two separate modes, but we would also make UI navigation buttons smaller, and easier to have missclicks into unwanted sections of the application. If we move forward with this approach, we would situate the new navigation button to the right of the renamed “Emergency” button, to encourage users to use the more prominent, or first button first.

Whichever approach we move forward with, we want to keep the implementation consistent with our current goals. We will strive to ensure ease of use, and quick access to vital information to the user, in a manner that’s easy to understand and reliable.

Prototype Description

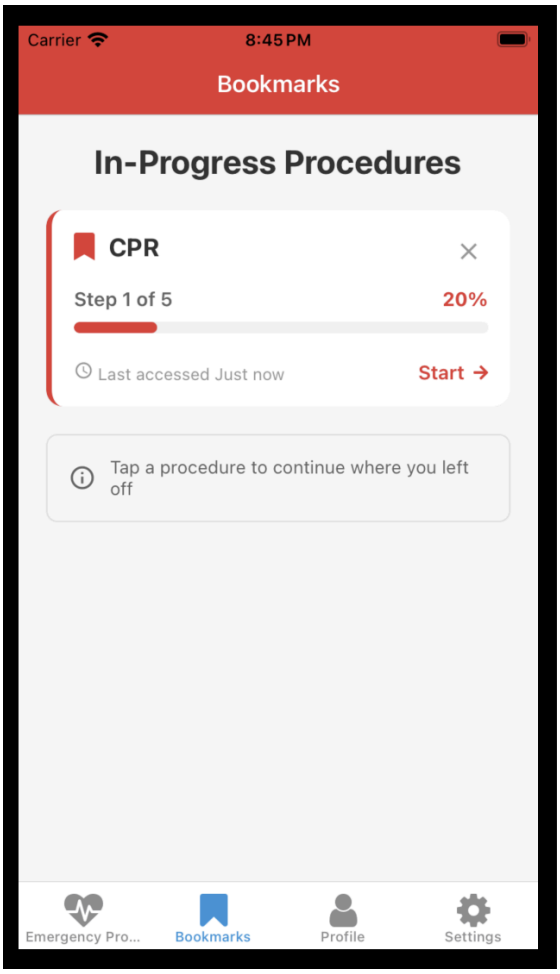




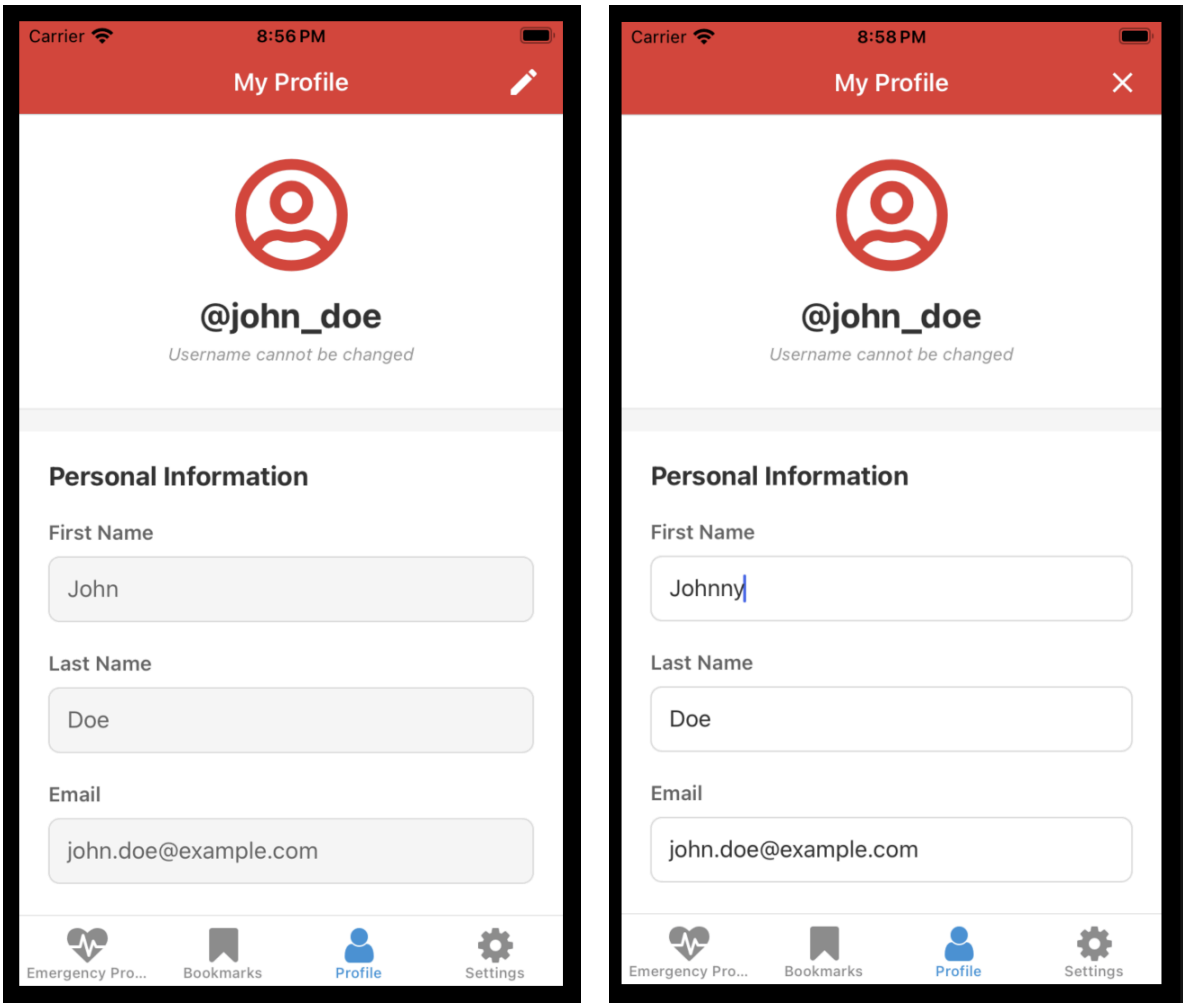
When opening the app, the user is immediately greeted by a simple and clean home screen. Because the design is focused on quick use in stressful situations, large, clearly labeled buttons sit in the center of the screen. To access any one of the guides on the screen, the user simply needs to tap on the desired guide. At the bottom of the screen is a page navigation bar, which allows users to go between the different pages of the app. Similarly to the buttons for the guides, the user simply needs to tap on the icon of the page they wish to navigate to.

Within the guides themselves, the user is taken through a step-by-step tutorial focused on quick comprehension. Each slide presents each necessary step clearly through text with an associated image that helps the user understand what is needed more quickly. Each slide also plays a text-to-speech recording of the information on the slides, allowing the user to follow hands free if necessary. To go between each of the slides, the user can simply swipe left or right just like the

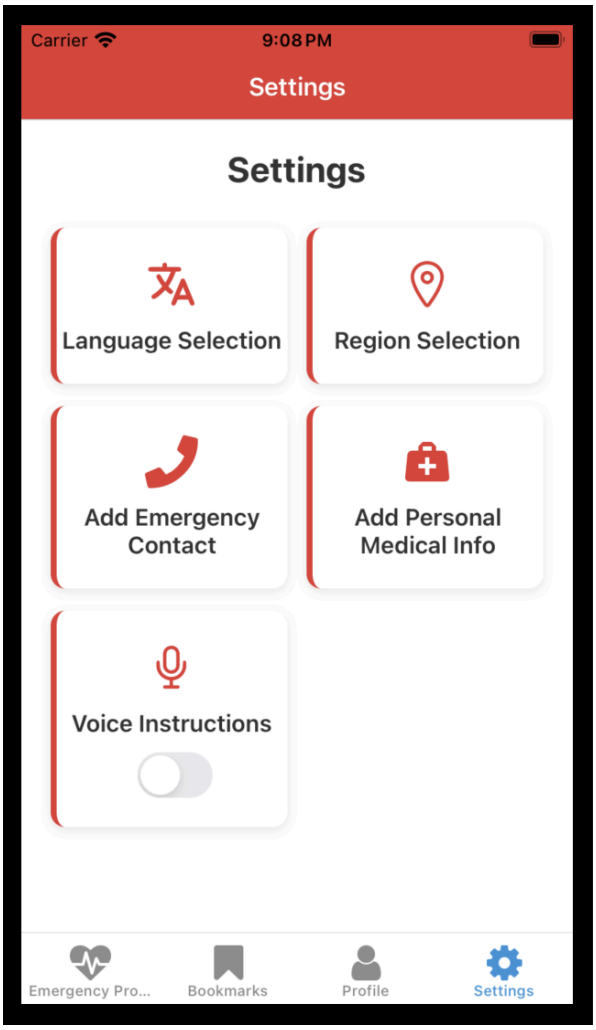
home page of a smartphone, or click the “next” button found at the bottom of the screen. To go back to the home screen of the app, the user can simply click the arrow in the top right corner of the screen,



Bookmarks store the user’s progress through different procedures. When a user leaves a procedure, the system stores the slide number. When the user comes back into the bookmarks tab, they will see the progress they’ve made in every procedure that they’ve opened. Functionally, a bookmark is not rendered unless the user has opened the procedure at least once. For the prototype, the bookmarks are stored in local storage as a proof of concept. In the future, bookmarks will be stored via a database to be consistent through login and logout functionality. We are currently working on revising this system to only apply to more educational portions of the system, and not linked into the emergency procedures system.



The user page is used to change certain information stored for the user in the system. Our user system is to be used to maintain persistence in the bookmark feature, and will be a requirement for progress to be stored. A user can change certain information by depressing the pencil icon in the top right corner, fields which can be edited are then opened to adjust information as desired. When the user is done editing their information, they can scroll to the bottom of the page and depress the save changes button to confirm their changes, or cancel to exit the edit mode. An important note on user sign in, because of the emergent nature of our system, we will not require an account to access the emergency procedures section of the application. Upon opening the application, the user will immediately be able to access this feature without any prompt for a login, or sign in. If the user wishes to enable bookmarks, they will be able to see a login / create account button in place of the profile button.



The settings screen is where users can change how the application behaves. The most important feature within this page is the voice instructions toggle. The voice instructions toggle should allow users to receive input from the application to allow their focus to remain on the patient as much as possible. For this reason, we will have this set as the default position upon full implementation. Along with this, we will have language selection, region selection, emergency contact and personal medical info implemented in the full release. Of these features, we want users to have the ability to select a language other than english, an emergency contact section along with the medical info section could aid responders in the treatment of the user if they became a patient themselves.