CodeQuest

Report 3

Section: English

Group: Group Number B09

17547, Nasr Ibrahim

Table of Content

1. ***Introduction***

In modern days, technology is evolving very fast, even faster than our brains can comprehend or more importantly keep up. With this evolution comes he increased demand for jobs within this domain. When the internet was first developed, only a handful of people were needed to see what the hype was about. It was not until the true potential was discovered that any company had a full specialist team dedicated solely for this purpose. Moreover, as time moved on, new vulnerabilities were discovered. Same as before, what started as one or two security people turned into a group of cybersecurity specialists.

Also in modern society, what used to take people months to comprehend is now taking children and babies days to understand. Of course, we are talking about electronic devices like smart phones, computers, and TVs. Tech enthusiasts considered the idea behind a touchable screen a miracle. A month-old will take one look and start swiping and going to YouTube. “What are the implications?” is an important question and topic regarding this phenomenon in health and morals. However, it is not asked enough concerning their future careers or even thought process in this domain. How they are being introduced to technology will influence how they perceive it in the long run.

These specialists, whether in web development, front-end, back-end, software, cybersecurity…etc., are in essence students. They finish high school, pick a major, go to university, and after graduating, start working. Due to the increased demand of such jobs, there needs to be an increased supply in the workforce. This workforce did not start development in university. It started when the highschoolers started considering the major because of the opportunities. This is our target audience.

There is an app called “Duo Lingo”. It is specialized in teaching languages to users. What fascinated me when I stared using it is the simplicity of using by laying out the correct path. This helped me to focus on the steps to reach the mastery of Spanish and cut off distractions. This facility was not provided when I started learning coding since when I started HTML, I found something called CSS, and before finishing what I started I started discovering CSS. Then it happened with JavaScript and before I knew it, I was burned out from learning multiple languages at once without a correct base and having false expectations. From this, I got the idea for “CodeQuest”. Its purpose is to slowly introduce the user to a certain language and when they pass a certain test, they go on to the other. This way, potential computer scientists would be setting on the correct path to have a strong base in coding, while being motivated to unlock new heights and skills.

2. ***Main Functionalities***

To start, a user would have a unique profile. This would help in saving progress and in later scalability features. Of course, they would be able to change username, password, or other features.

Then, they would not directly start learning. A user would be introduced to the two most common paths: front-end and back-end. Then, based on their choosing, a selection of languages most related to one of them would appear.

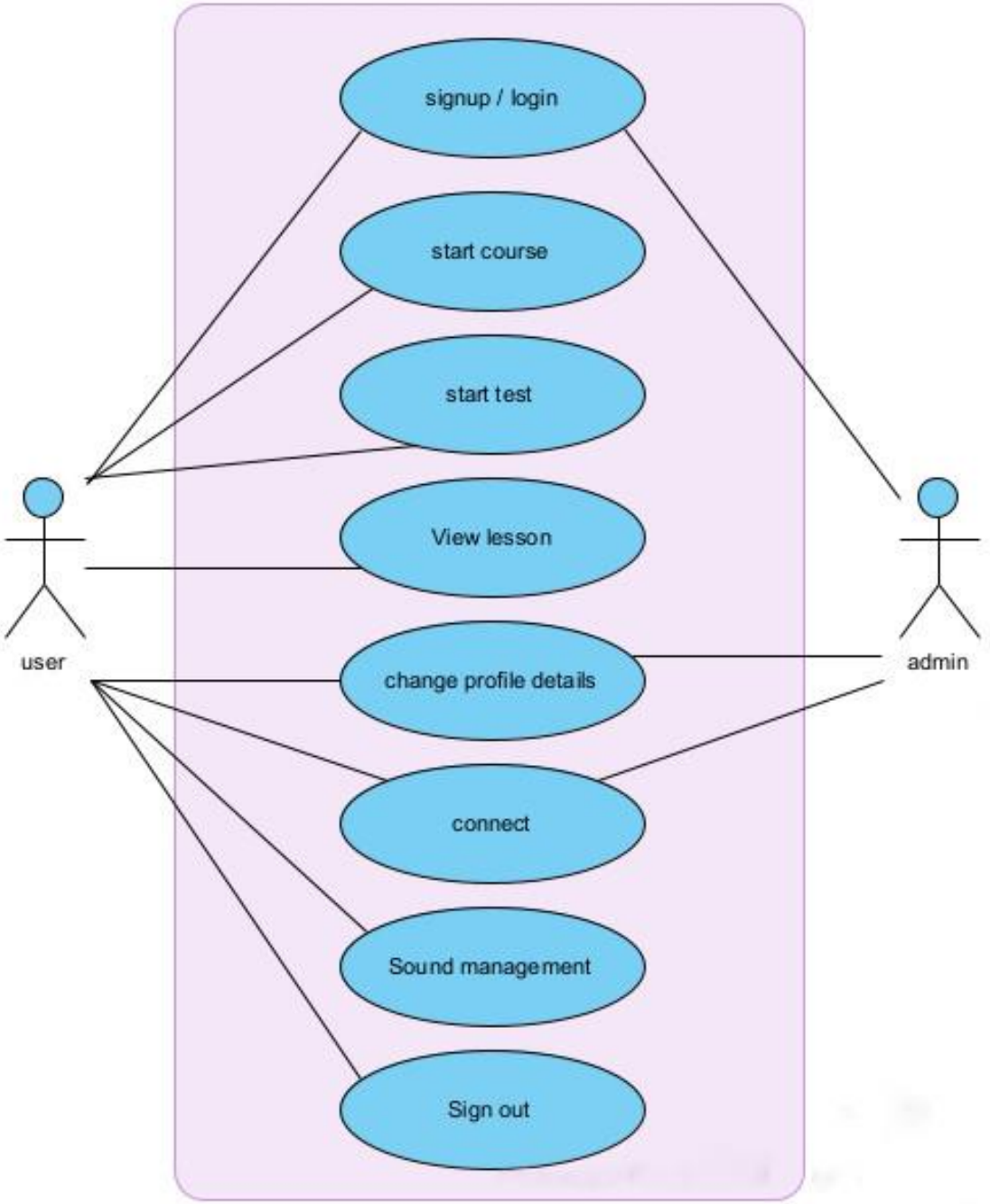
There would be two main ways to learn: theoretical reading and practical applications.

In theoretical, there would be technical documentation, where a user to read about a specific aspect in the language. There would also be a multiple-choice test to test their knowledge.

In practical, a user would be given a short prompt. They would type in the needed code into a prompt and see the output in real-time. This would combine practical application and critical thinking where users would be able to directly see what went wrong and fix accordingly.

There would also be the ability to connect to other users and see each other's progress. That way, users would be motivated by others progress and would help relate them based on their skillsets.

3. ***Use Case***


signup / login
start course
start test
View lesson
change profile details
connect
Sound management
Sign out
user
admin

1. User Signs Up / Logs in

The user initiates the sign-up process by providing necessary details or log in by entering valid credentials. The admin validates the information.

2. User starts course

The user can start any course they want after seeing the purposes and objectives of the coding language.

3. User starts test

The user can start a test to measure progress gained

4. User Views Lesson

The user selects a lesson from the Lesson Menu, and the LessonCourse screen is displayed. The user can navigate through the lesson pages.

5. User Changes Profile Details

The user accesses the profile settings and modifies details such as username, email, or password. The admin validates any changes.

6. User connects to other users

The user can view and connect to others using the application. The admin keeps track of followers list.

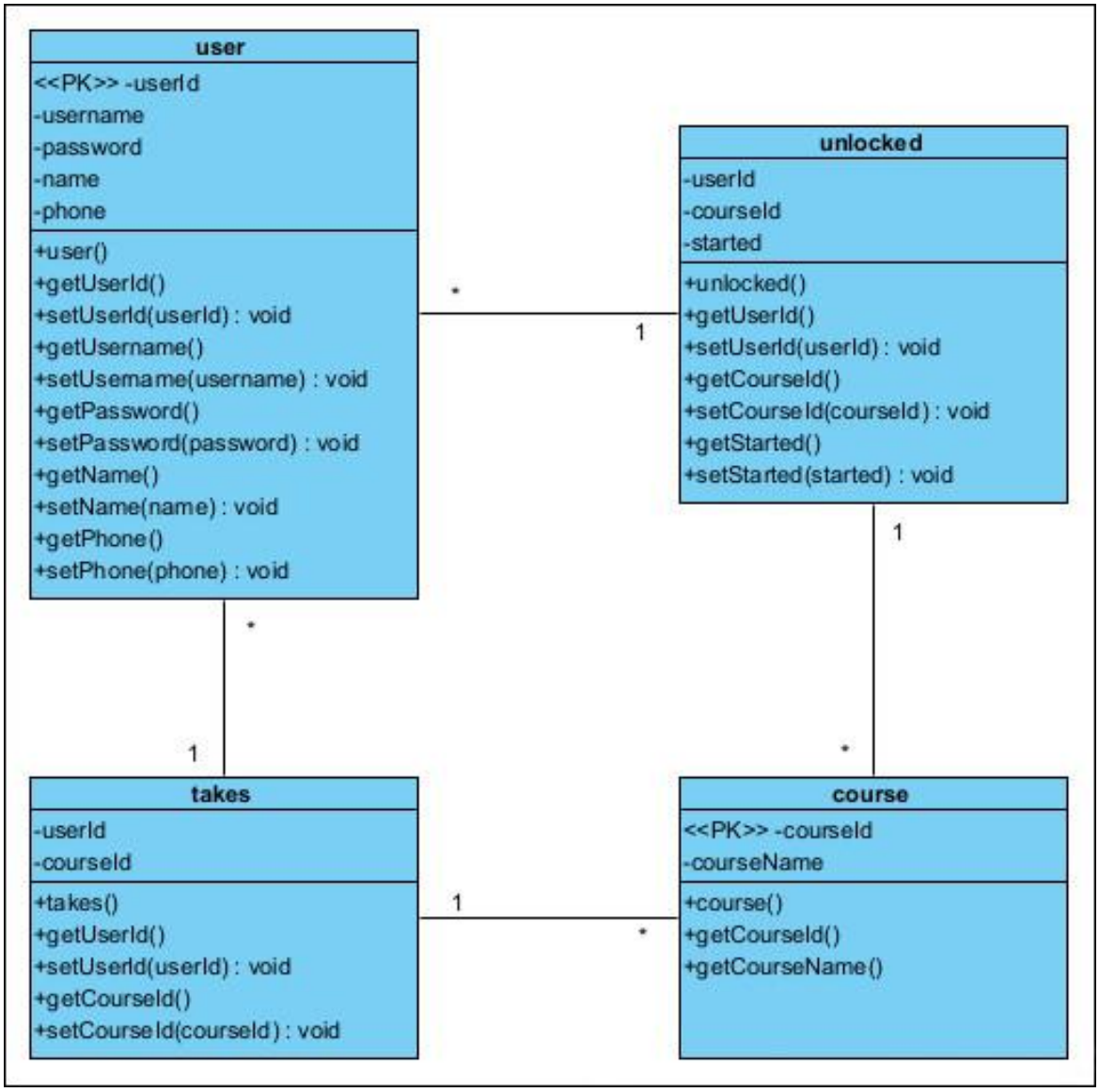
7. User Adjusts Sound Settings

The user adjusts sound-related settings by toggling the sound switch to turn on or off sound effects.

8. User Signs Out

The user initiates the sign-out process by clicking the "Sign Out" button. This action clears user data and returns the user to the login screen.

4. ***Class Diagram***



For the classes, we have as primary classes: user and course.

user:

Has as attributes unique userId (PK), username, password, name, and phone number. All these have setters and getters to facilitate their changing.

course:

Has as attributes a unique courseId (PK) and courseName. Only getters since each course will already be set with these attributes.

In addition to these two we have “takes” and “unlocked”.

takes:

It means “user” takes “course”. It has as attributes userId and courseId both FKs.

The relation between user and course is many-to-many since multiple users can have multiple courses. The “takes” class then has one-to-many with each since there exists one record of each user taking each course.

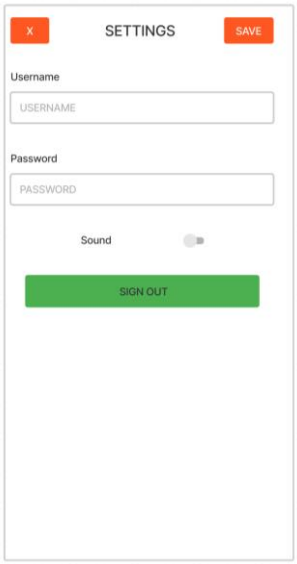
unlocked:

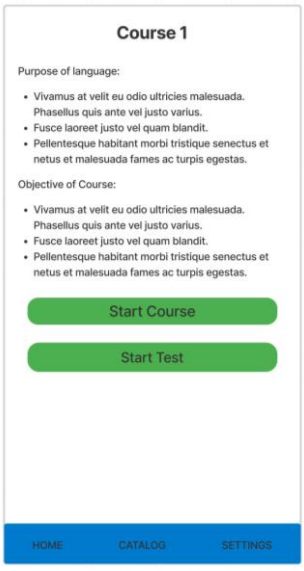
It means “user” unlocked “course”. It has as attributes userId and courseId both FKs and boolean started to show if the user started the course or not.

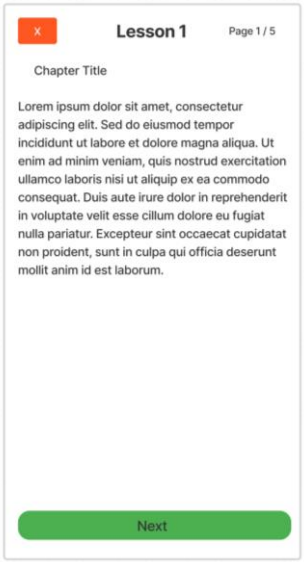
Same as before, the relation between user and course is many-to-many. Then “unlocked” class has one-to-many with each.

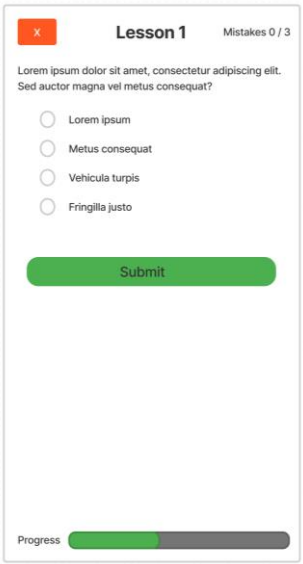
5. ***Screens description***

The above is the complete tree plan of my screen sequence in Figma.
The following are each screen shown separately.









1. Log In:

The Log In screen will be simple. The user will enter their username and password according to the spaces where there will be hints to help. There will also be a "Keep me signed in" button that a user can check to reduce the number of times they log in. In the above image placeholder, we will place the logo.

2. Home:

On entry, the user will be greeted by name in the above text. The lessons they already started can be accessed via buttons in the body where they will be taken to the Lesson Menu screen. In the below navigation bar, they can switch between the Catalog and Settings screens.

3. Forgot Password:

The Forgot Password screen is still being studied. Most probably, the user will enter their username and phone number to change it.

4. Catalog:

The Catalog screen will show lessons that can be started. Of course, the navigation bar below will still be visible.

5. Settings:

In the Settings screen, a user can change their username or password by typing the new ones in place of the old and clicking the Save button. They can also turn the sound on or off by clicking the switch. By clicking the sign out button, they will be redirected to the Log In screen. The navigation bar will not be visible here, instead when the close button is clicked, users will return to the Home screen.

6. Lesson Menu:

In the Lesson Menu screen, the user will first see the Course Title. Under it, they will see the purpose of the language and then the objectives as bullet points. Under these are two buttons, the first will redirect to the Reading Course and the second to the Test. The navigation bar will be visible here.

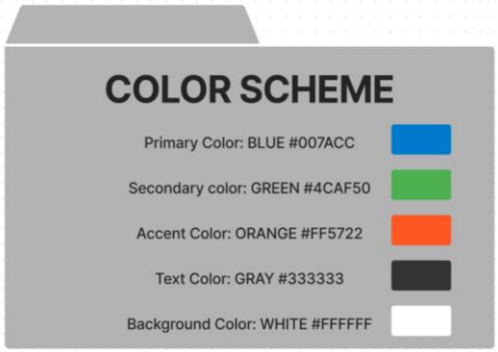
7. Lesson Course:

When a user enters the course, they will see the title, followed by the chapter title and then the text. Under these will be the Next button to switch pages when done reading. In the top-right corner, there will be a text showing the current page out of the total. The navigation bar will not visible and the user will have to confirm exiting when clicking the close button in the top-left corner.

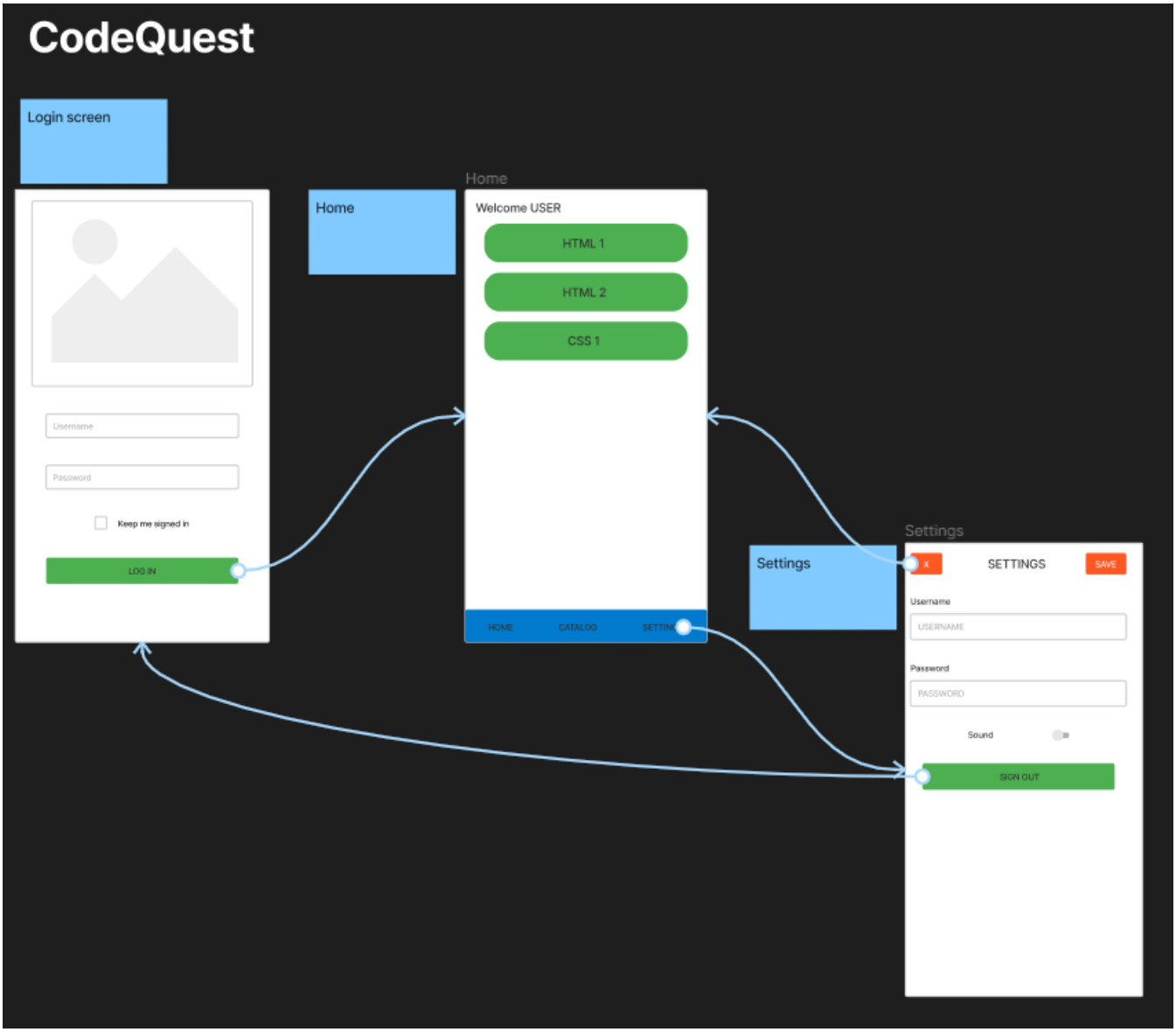
8. Lesson Test:

The test is multiple-choice questions. As usual, above will be the lesson title and the close button. To the right will be a mistake counter, showing the total mistakes out of three (the maximum). In the body, we will start with the question followed by four options, then the submit button. Below will be the progress bar that will fill a bit at each correct answer.

In addition, to the right is the color scheme used.



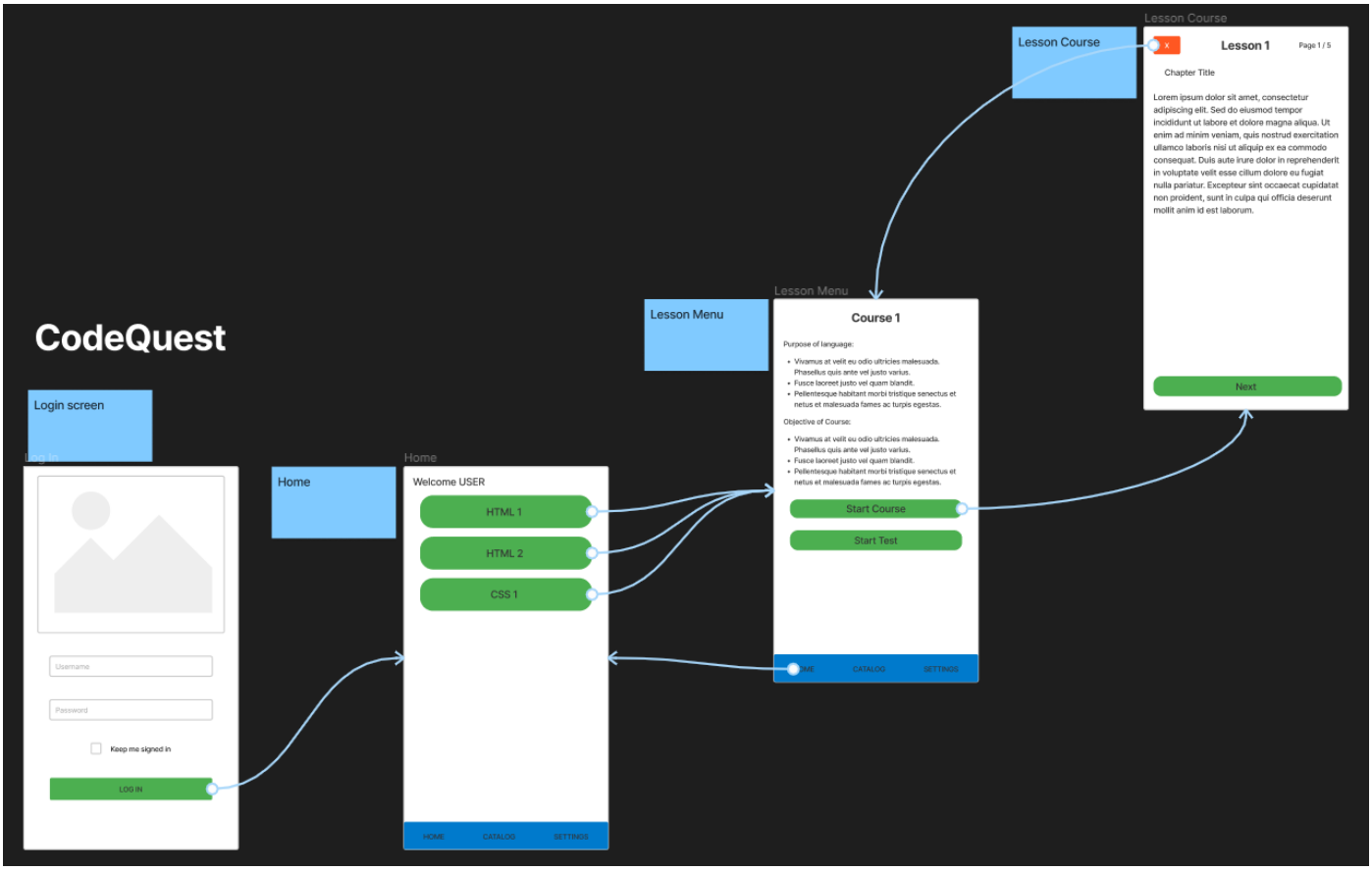
6. ***Sequence Diagram***



As we can see in the above flow diagram, a user would start at the Log In screen. After entering the correct username and password, they would click the Log In button which would take them to the Home screen.

There in the navigation bar, if they click Settings, it would take them to the Settings Screen.

In the Settings screen, there are two options. By clicking the “X” button, they would go back to the Home screen, and the Sign Out button would take them back to the Log In screen.



In this flow diagram, the user would also start at the Log In screen. As before, the Log In button takes them to the Home screen.

Now in the Home screen, presssing any of the buttons in the body would take the user to a Lesson Menu screen with unique text according to the Java code.

In the Lesson Menu screen, a user can go back to the Home screen by clicking Home. Or they could continue to a Lesson Course screen by clicking the Start Course button.

In the Lesson Course, they could continue the course by clicking Next or exit by clicking “X” button. Either way, they will go back to the Lesson Menu Screen.