Question Secure- Final Report

Matas Aleksiejus – C00239630

Contents

Acknowledgements

I would firstly like to thank Christopher Staff for being a fantastic and very patient supervisor who was able to help me every step of the way with any problems I had, whether it was rethinking the core project functions or time management. Even when my personal situation got in the way he was very understanding and was always there to find a way to help. I would also like to thank all the supervising lecturers who have been overseeing all presentations and demos for providing critical insight and advice in regards to this project that I would not have been able to see without their feedback. Lastly I would like to thank all of my lecturers over the last 4 years that have equipped me with the skills and knowledge to carry out such a huge undertaking as the final project.

Achievements

The greatest achievement and something to be proud of is that I managed to finish the project that I set out to complete and have a working and functioning product at this time in April 2022. On a more in depth look on what I actually achieved:

Research Document

1. Carried out research on security questions as a concept in security itself. Firstly, talking about what security questions are and how they came to be used outlining their earliest documented uses.
2. Secondly, explored possible future solutions of using shared secrets for account recovery such as generated stories. Based on Propp's theory that all stories follow the same 7-8 character archetypes.
3. Lastly, I identified the major downfalls of security questions and their use for account security. Explaining how the downfalls work and how they could be prevented.
4. Researched and documented people's solutions to security questions and how they get other applications to work for them in account recovery.
5. Specified possible technologies to be used in the implementation of the solution which I proposed. The solution was a direct answer to the downfalls of security questions.

Functional Document

1. I began by setting out goals for the implementation part of the project, outlining the things I wanted to achieve based on the spec that was provided by my project supervisor. The goals I set out were:
    a. To provide the end user with an easy to use interface for generating and storing security questions and answers.
    b. To ensure end user safety with the ability to generate secure and unique answers that cannot be recreated.
    c. To securely store previously used security questions and answers as well as store newly generated ones.
    d. To provide the end user with a browser plugin that makes it as easy as possible to create secure answers.
2. I then went into more detail about the technologies that I was planning to use carrying out research and outlining specific libraries and dependencies that my project will deploy to achieve the goals I previously set out.
3. I covered how I was going to develop the main application stating that it will be developed in android studio and java, and mentioned the development of the browser extension that was to accompany my application.
4. I wrote about the cryptographical methods that I was going to deploy to allow my implementation to protect my users information. Taking feedback from my lecturers and planning to implement safer authentication.
5. Then I explored how I was going to set up and utilize the database for my application and how it was going to store user data securely, where it was going to reside and what layout it will have.
6. I played around with the design of the application until I received a design that I liked and wanted to follow through the whole project which I documented in the functional document. I mentioned the application I used to create the design as well as included some design screenshots.
7. I used context diagrams to show how the system interacted with its internal functions and its external parts. I used data flow diagrams to detail how the application handles data and what functions process it before its stored or displayed to the user.

Presentations

1. I prepared a primary presentation that explained my application giving a description of the general idea of the project and the functionality of my proposed solution

2. I also wrote about the work I had completed so far such as, the research document and functional document to give the lecturers an idea of the progress I had made.
3. I presented the application in front of the project supervisors and listened to their feedback.
4. Co-ordinated with my project supervisor on how to process the feedback I received and how to implement new ideas into the project.
5. I prepared a second presentation for my project using the design I had came up with for my entire project. I introduced the project again and included a screenshot of the first landing page to show what the application was going to look like.
6. I included changes from the last presentation based on the feedback I received such as implementing biometric authentication and specifying the keystore used that was secure enough for encryption of answers.
7. I also included a short demo of my application that showed some of the design aspects and the database working to demonstrate to the lecturers the progress I had made.

Application

1. I carried out research into android development for the entire lifetime of this project and learnt how to develop an android application on android studio that supported database, cryptographical and external function. This was a massive undertaking.
2. I began by learning how to create views and how to design interfaces using xml. I then learnt how to call aspects of the UI into the java code and retrieve values from input texts or button clicks.
3. After I had learnt this I started developing my android application. The first page was a login page that had enter username and password fields and a button to submit.
4. At this time I also wrote a class that handled all database function such as create, update, read and delete. I tied the two together making a working login page.
5. After this I developed a home landing page and all of its buttons that led to different functions within the application. At this point I has gotten rid of the sign in page and replaced it with biometric authentication.
6. The first functionality page I developed was to add a question. I configured a function to take user input of a question and a website and generate a secure answer. I had to then set up the cryptographical function of the program so I could encrypt and store the generated answer along with the question and website.

7. I built a class to handle all cryptographical function such as retrieving the key from the secure android keystore, encrypting the data, decrypting the data, encoding to base 64 and appending the iv. This took a lot of learning as it was difficult to learn about the Android KeyStore, the trusted execution environment and how to access the key for decryption.
8. I then updated the application database methods to store the encrypted data locally on the device and wrote functions to read that data from the data base for decryption.
9. Next I needed to configure a function to retrieve all entries from the database, so I had to create a new view that would automatically generate dynamic text views that you could interact with. This was really challenging because textviews on their own don't have a lot of functionality so I had to research and figure out how to make them work with a relative layout and set their position to the right place on screen. Each of the text views had to be decrypted when clicked so each generated text view also had to have an on click listener that called to retrieve the entry by website name for questions and answer from the database. Then the data was sent to the decrypt function and the displayed to the screen in a dialog box.
10. The last main function main function of the application was to implement the QR code scanner. This was a lot more taxing than I expected. I needed to first of all allow for use of the camera for the application, this included asking the user for permissions. I also needed to add dependencies for the zXing library which is what I used for the library for the QR code scanner.
11. I then added a way to delete the database that contained the security questions and answers stored in case you were getting rid of the phone or giving it to someone else.
12. Lastly I followed through the application to ensure that styling was consistent with the design I had laid out in the functional document and was the same throughout.

Browser Extension

1. I had to carry out research on the browser extension development. A type of development I had never did before. Browser plugin

development is carried out in HTML mostly with any functionality being developed in JavaScript.

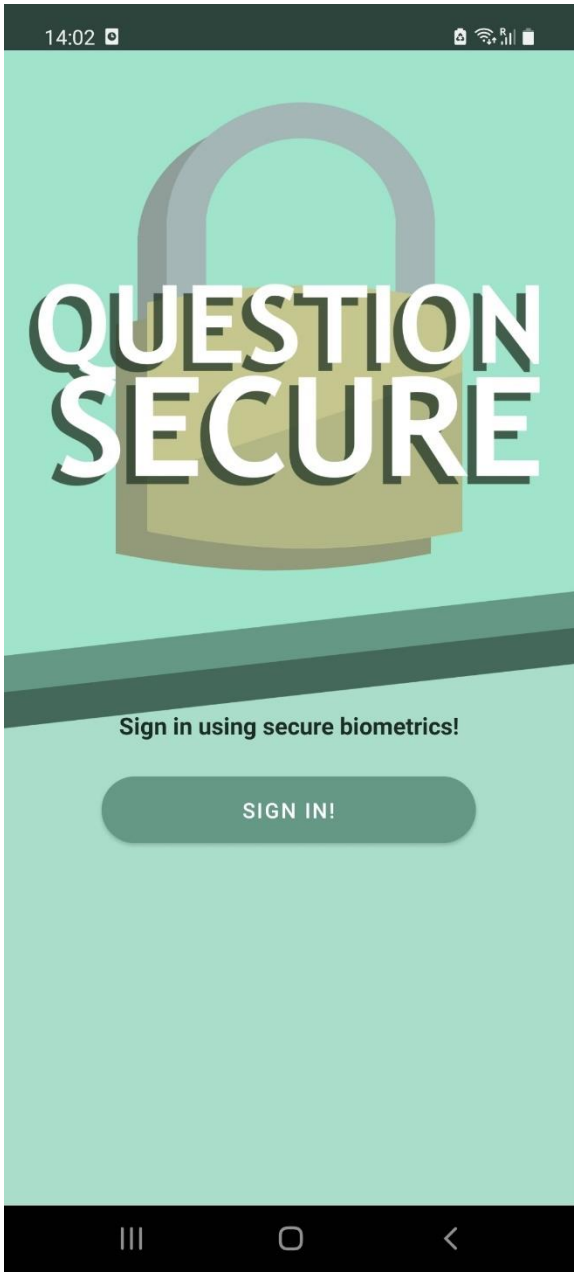
2. I first configured the manifest.json file which specifies the extensions name, version, permissions and actions. You can also add icons to the browser extension that show up on the browser to signify your extension. I used the question secure logo I had designed earlier in the year.
3. After that I configured a html file that would pop up when the extension icon was pressed. This popup allowed the user to enter a security question.
4. I imported the library file QRCode.js to my html file to allow for QR code generation. I also wrote a script that would take the user input and generate a QR code passed on the input and display it in the popup. This was really difficult to achieve as browser extension shouldn't really have inline scripts running and so I got a lot of errors and had to get the hash of all the scripts in the file to add to the manifest.json as an exception.
5. I needed to program a way to retrieve the domain name of the current page to add to the QR so I did research on google JavaScript functions and how to handle URLs and turn them into domain names.
6. After that I had to update the script inside the pop up file to generate a QR code based on the domain name and the security question entered. This meant that every change made to the script had to be then accompanied by updating the hash in the manifest file.
7. Next I had to update the main application to handle the QR code and the data inside it. I this this by splitting the string retrieved from the QR code into two parts, the website and the question. I then used the same function from the add activity to generate a secure answer.
8. Finally I called the same method to encrypt and store the data on the database, thus equipping the application with the ability to read QR codes made by the accompanying web browser to generate secure answers and store them on the device.

Final Design

Below I will include screen shots of all pages from the application and browser extension including any dialogue boxes and actions taken to show the design and functionality of the final product.

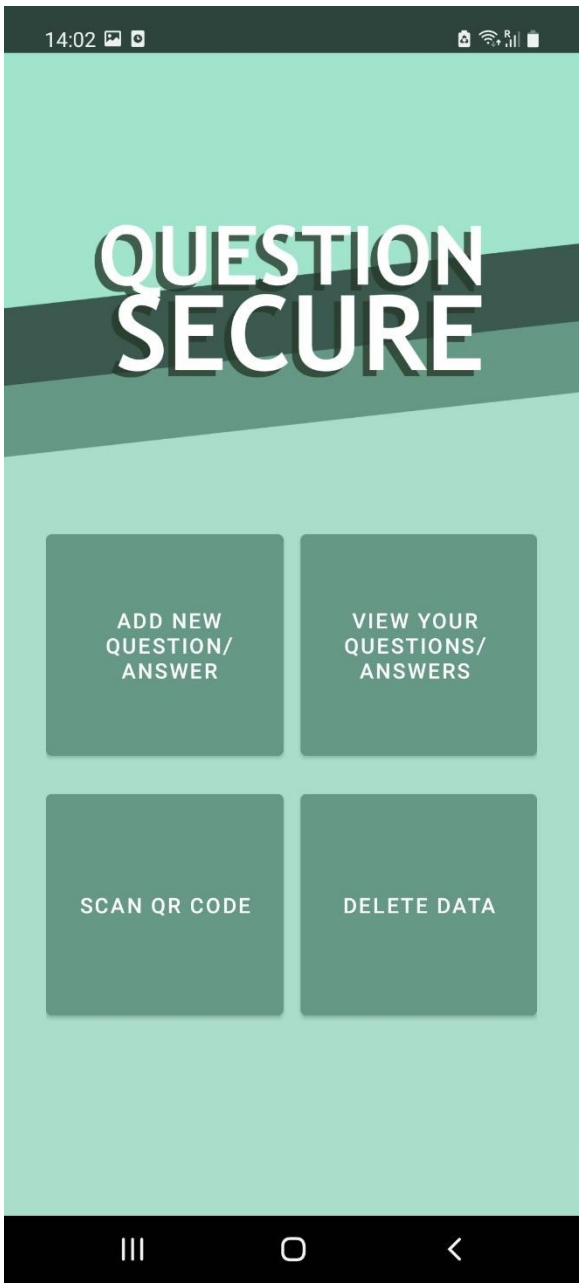
Design

Landing/Sign In page



This is the Landing page of the application. The page you see when you first open the app. Utilizing the logo and the design from the design document, the view created here continues throughout the entire application. Pressing the sign in button will show a biometric prompt. This will give you an option to use either a fingerprint or facial recognition to authenticate. If no biometric authentication is enrolled, the application will display “authentication error”, wrong authentication will display “authentication failed” and successful authentication will display “authentication successful” and lead the user to the home page.

Home page

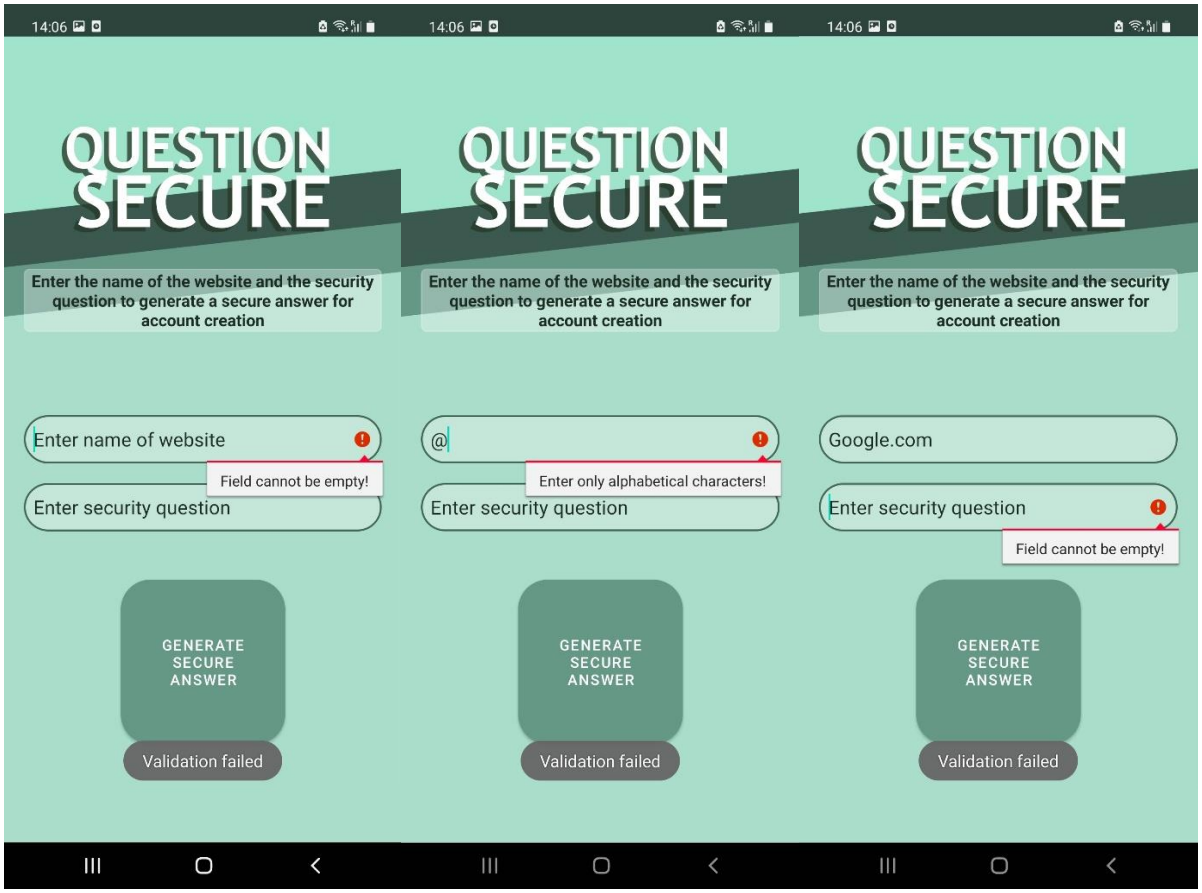


This is the homepage. The user is directed here after they authenticate on the landing page. This page contains the logo without the lock this time and it has 4 buttons. Each button leads to a specific part of the application.

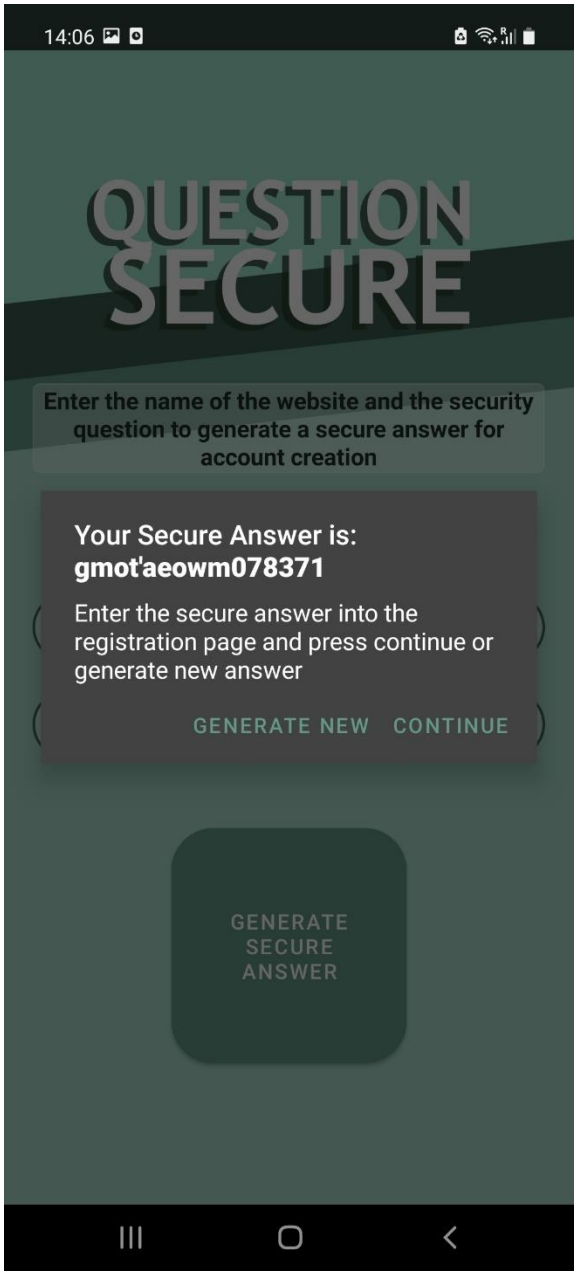
1. To add a new question or answer
2. To view questions and answers
3. To scan a QR code from the accompanying browser extension
4. To delete existing data

All 4 options contain a new page and pressing the back button will take the user back to the sign in page, however the user will still be authenticated.

Add new questions/answer page

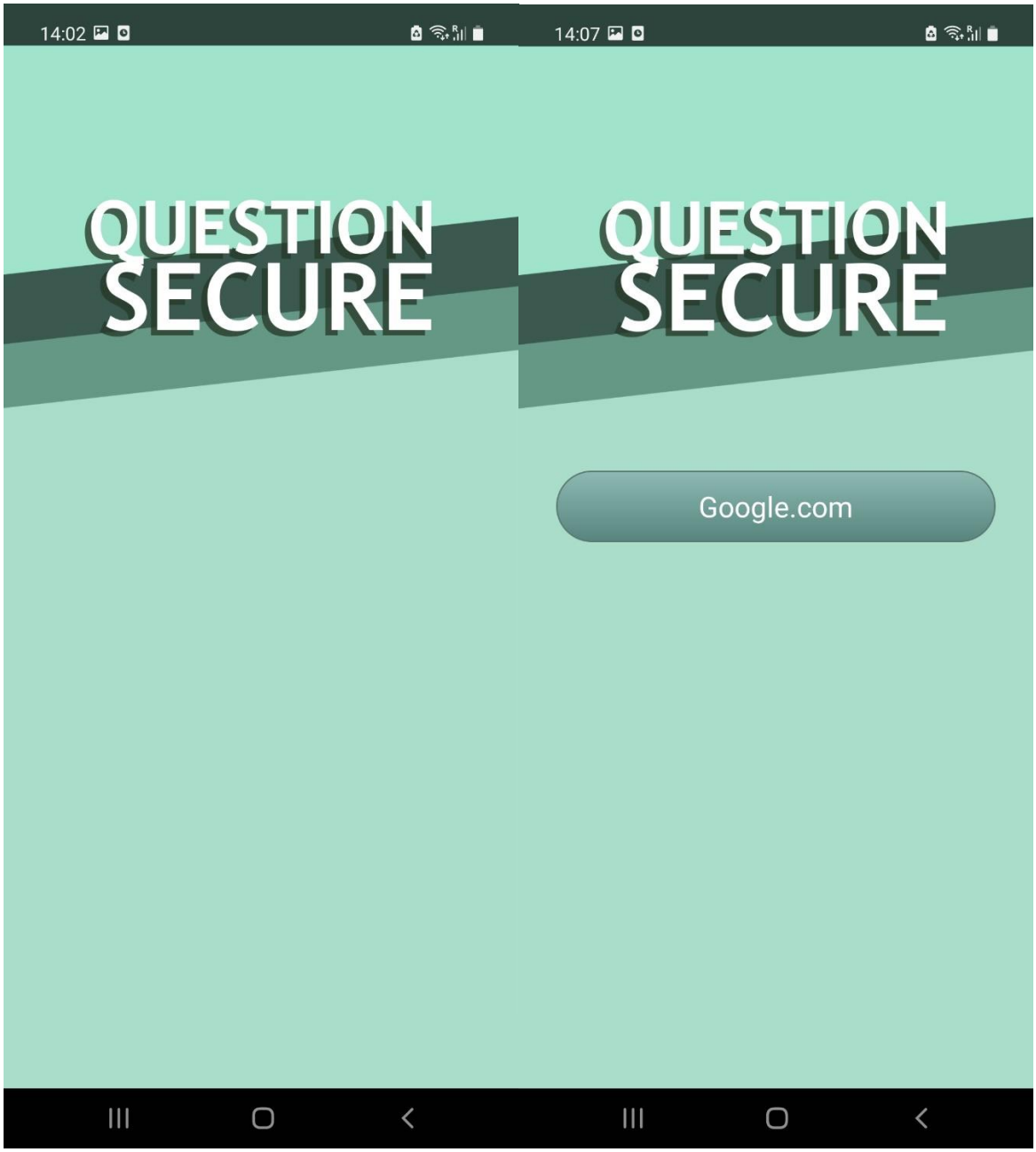


This is the page that the user gets directed to add a new question and generate an answer. Above we can see two text fields that allow for user input. We can see some validation taking place. The red exclamation point is an error text that is set when focus is brought to an individual field. It shows the user that both fields must be entered correctly for validation to succeed and these error messages show up only when user presses "generate secure answer". Once validation is correct and the user has entered a correct format for the website and question will the dialogue appear.

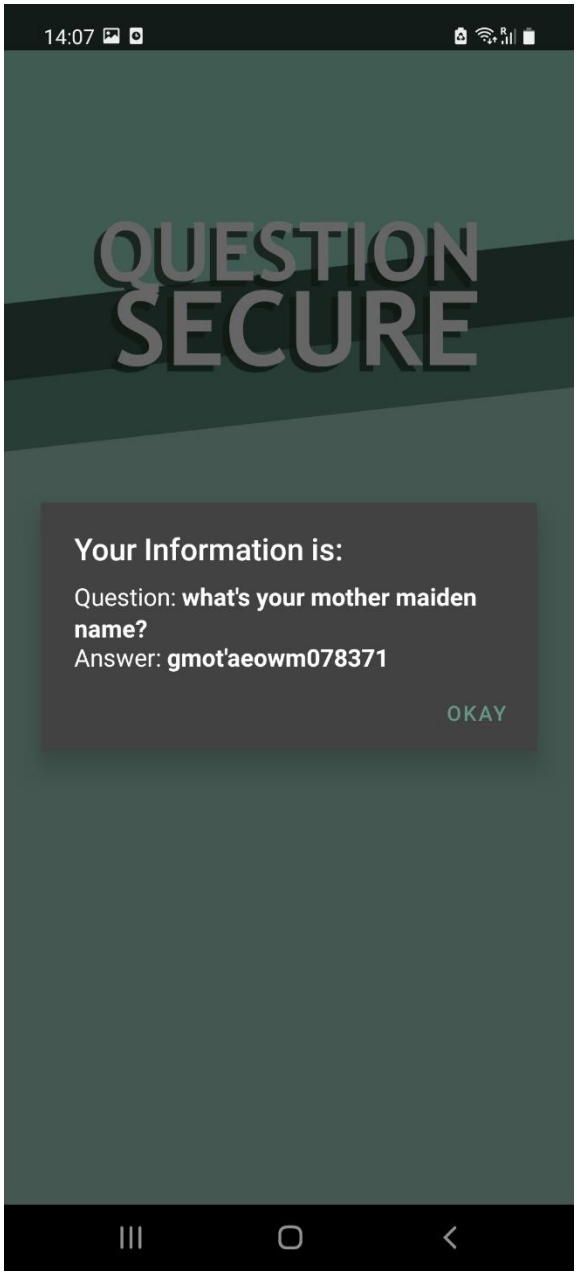


This is what is displayed once validation has succeeded and a questions is generated. The user has two options, to enter the answer they have been given or generate a new one. This is specifically set up this way in case the specific website they are making an account for doesn't allow any special characters, then they can generate another random answer that doesn't contain any. Once the answer has been generated the user will then enter that answer in to the answer field of the website they're registering for and click continue. Clicking continue will ask them to authenticate with biometrics again and a message "stored successfully" will appear on the screen.

View questions / answers

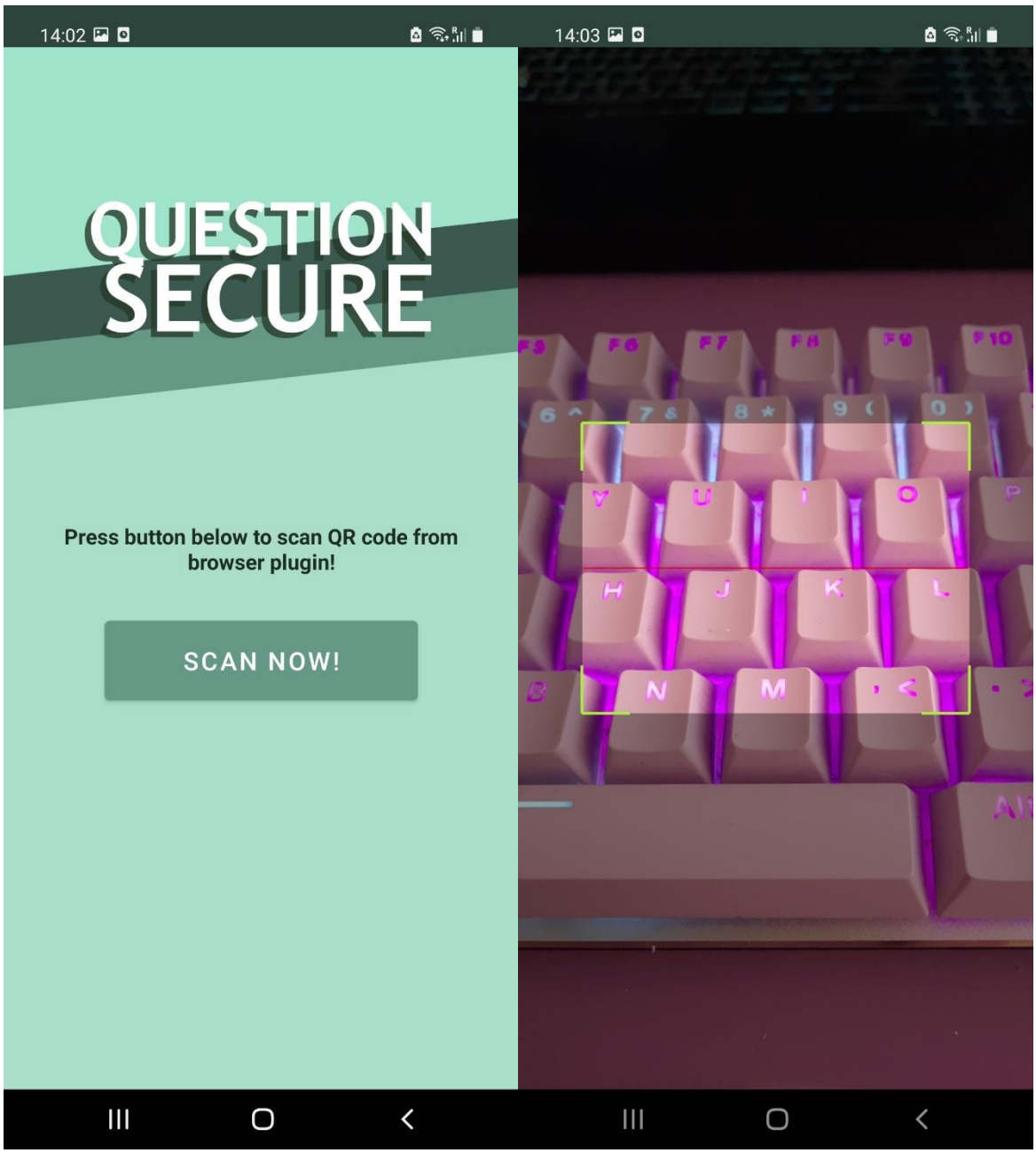


This is the page where the user will view their questions and answers. When opening this page, initially it will be blank. This is because the database is empty. Once you add a question/answer and store it in the database, all entries will appear on this page. These are created dynamically for each entry in the database and each can be clicked to view that specific website security and answer combination.



This is the dialogue the user is greeted by when they click on an entry in the view questions/answers page. They are prompted by a display of the questions and answer they had used for that account creation. This is possible because each of the text views in the view questions/answers page is dynamically generated with an on click listener. When clicked they send a fetch request from the database based on which website is clicked. The data retrieved is then sent to the decrypt function and the data that is returned from there is displayed on the screen for the user. Once the user is finished viewing their answer they can click okay to return to the view your questions/answers page.

QR Code Scanner

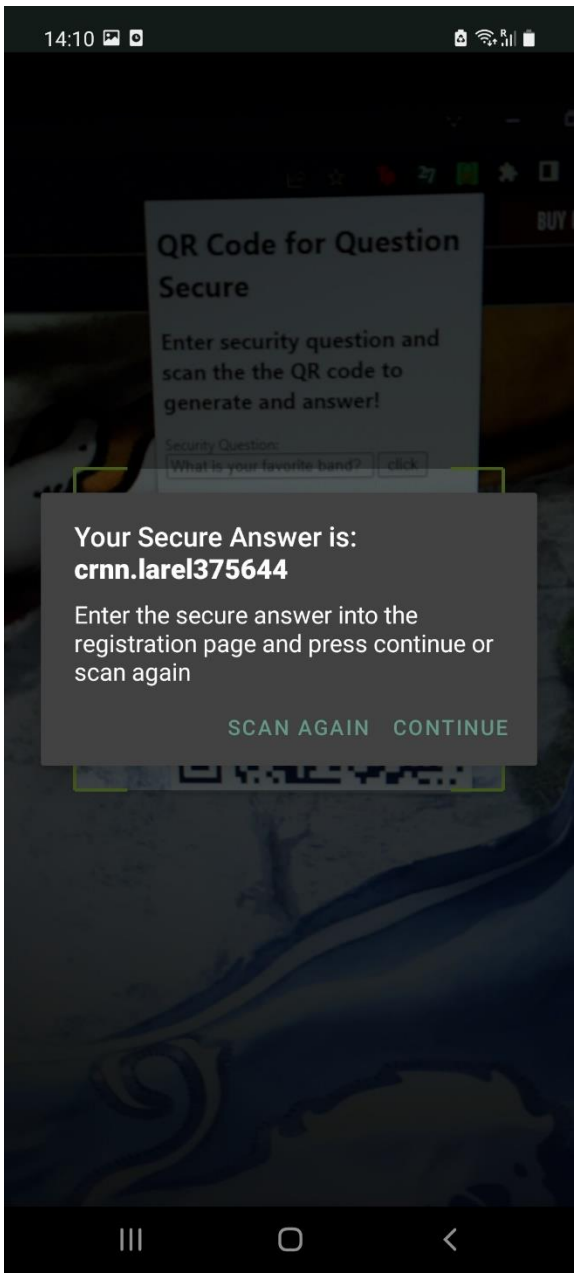


This is the QR code Scanner page. When the user is directed here from the home page they will be prompted for permission and that's why it's necessary to have the landing pages that says "Scan Now!". This also allows them to control the scanner by only activating it while focused on the scanner screen and having it inactive while anywhere else on the application.

Browser Extension

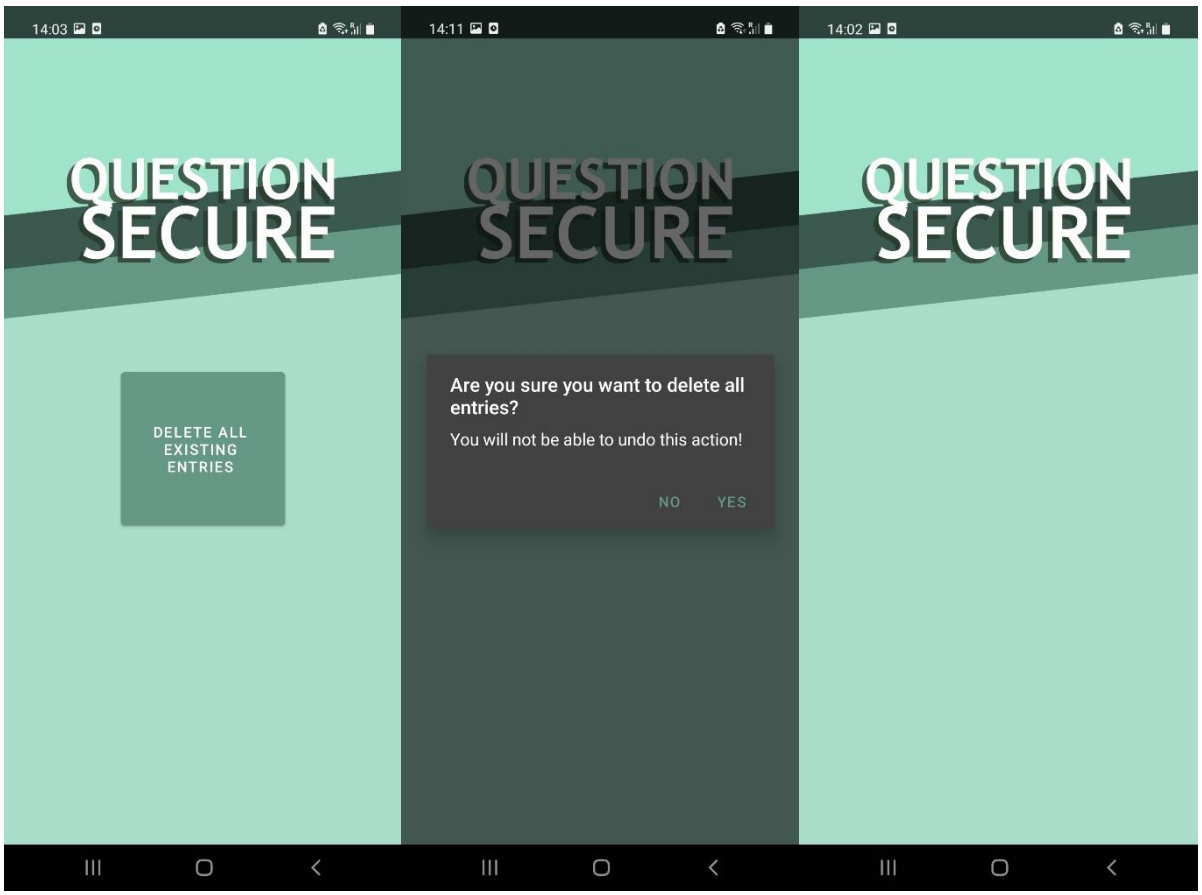
These are the views of the browser extension that accompanies the main application. The view on the left is what the user is greeted by when they click on the browser extension initially. Once they enter the questions they will use for account creation and click the button they will see a generated QR code. This QR code contains the string of question and website in the format of "accounts.elderscrollsonline.com , what is your favorite band?". This QR code is then scanned by the QR code scanner on the main app to be converted to a secure answer and stored.

QR Code Scanner ext.



This is what the user sees on the QR code scanner view when they scan a QR code generated by the accompanying browser. It is the same dialogue that is shown by the user when they enter a question manually. They are again prompted by two choices to "continue" or "scan again". Pressing "scan again" will bring the user back to the scan view where they can scan the QR code again to generate a different answer. If the user presses "continue" their answer will be stored in the database. This is possible because when the QR code scanner reads the code it gets the string "accounts.elderscrollsonline.com , what is your favorite band?" . The string is then split into two parts and passed to the same function that generates the answer.

Delete data



These are the views for the delete data option. The most left view is what the user sees when they first land here after the home page. The middle view is what the user is greeted by after they press the "Delete all existing entries" button on the left view. This dialogue prompts the user with a question of whether they really want to delete all their data. This is to ensure that their data doesn't get deleted by accident. The dialogue also informs the user that they will not be able to undo this action. If the user presses no they will be returned to the home screen. If the user presses yes they will send a call to the class that manages the database to delete the database file and will be returner to the home screen. The right most view is the view questions/answers view after the database has been deleted. It is empty as before since there is no database entries.

Problems encountered

Throughout the project, I encountered many problems from personal to technological and everything in between. The first and biggest problem I encountered from the start that stayed with me through the whole project was time.

Time and time management was the biggest issue in regards to this project. This was down to a couple of factors. The first was down to a personal issue that still hasn't been resolved and it made the entire process very difficult. It does make me more proud that I finished the project at all considering the personal circumstances that I would rather not share, however, I can't help but think how much easier it would've been if this wasn't the case. Another reason time was a huge problem in the lifetime of this project is the lack of it that you have in 4th year of college. While every lecturer was understanding of the situation that all of the students were in, they also had to assess us in their own subjects. This meant that studying for tests or working on assignments always took away from project time in some way or another.

Time was a factor in a technological sense as well. Since I have never attempted android development before nor have we learned about it in the last 4 years, learning it on my own from the start was really time consuming. Stopping and starting all the time, getting to grips with the IDE and emulation systems, android file directory, learning how to configure views. All of this took an enormous amount of time to learn and the only way I was able to manage it is because I chose to do it in java, a language I was already familiar with.

Technologies were also quite problematic. As I just mentioned before in this project I hadn't attempted android development before. Because of this, every issue I had with development was a lot worse than it should've been. Every error was new and I had a lot of trouble understanding how android development worked. My entire project had depended on the encryption key being locked away behind biometric authentication and when it came to implementing that part in April, I found out the hard way that my mobile device, the one I modelled the application, on didn't support that key algorithm so the full functionality worked on the virtual device but not on my physical device.

Things I've learnt

I have learnt a lot of different skills throughout the development of this application and the implementation of the project as a whole. The biggest skill in all of this would be android application development. Carrying out this project has made me want to pursue a certification in android development after graduation. I put so much time and effort into developing this skill I definitely want to pursue it further. I have already looked into a certification by google for android development that's based on Kotlin and I'm very excited to expand my knowledge into a different language.

Another takeaway from this project is how crucial planning and preparation actually is. It's the part that I imagine most people tend to skip the most while in reality it should be the most important. I know this now because I skipped it and I don't even have a project plan for this project because I didn't plan for it at all but I really wish I did. Even breaking the project into smaller steps makes it so much easier to tackle than the whole big project all at once. Knowing the structure of my application and key functions would have been crucial prior to starting development and would have saved me so much time in the grand scheme of things.

Things I would do differently

There are a few things I would do differently were I to start this exact project again:

1. I would look into developing the application for a couple of different platforms. This consideration was given to me by one of the project supervisors and it came a little too late in the development cycle.
2. If I was focusing on android development again I would definitely look into some courses or learning materials for android studio with the project supervisor. Learning it all by yourself from YouTube videos, random webpages and the android documentation is a massive undertaking and knowing how to use the IDE and all of its systems at the start would have been invaluable and would've helped me get a lot further a lot faster.
3. I would spend a lot more time in the planning an preparation stages to help with the development later on. Building a program that has an interface with a bad plan at best was really tedious and it made me backstep and even start from scratch at the very start.

Conclusion

This project that lasted from October to almost May was the longest and hardest project I've had to do and I feel very proud of what I've accomplished. It wouldn't have been possible without all the help I received from my project supervisor or the rest of the lecturers. I have achieved so much in the last 6 months from research, to design, to application on a topic I studied in so much depth. I have achieved all the goals I set out to achieve in this project and I believe my final project design and its functionality reflects that. There were a lot of problems I encountered and lots of things I have learnt along the way, and so many things I would do differently if I could start again. Despite all that I am satisfied with my work and my project and I am excited to expand all the things I have learned in the future.