**About the Project:**

*[Slide: About the Project (title), State Goal, and Give app Chosen with icon]*

One of the projects for this month was a Custom App project. In the first week I chose an app to analyze and reverse engineer in order to how it was made and when, where, and how any databases were being used in it.

The goal of the project was to see and understand how databases are used within an app.

The app I chose to research, and reverse engineer was the Apple Notes app.

**Dissection Topic:**

*[Slide: Dissection Topic (title), list basic app info, list of shortened improvements]*

The Notes app is, like the name indicates, a note taking app. I decided to pick a note taking app over any other simple app because of the various types of simple apps I have on my phone I use note taking apps the most.

I chose the Notes app because I could see a few things I thought the app could improve upon, I use it the most and it has the most focused functionality of the different note taking apps I have.

I researched the way users would be able to access and edit the database using the app.

**App Dissection**

*[Slides have Notes Flowchart on Left Side, and Info on Right]*

Notes is a Productivity, note taking app, developed by Apple. *[Info: said here and app icon]*

The pros of Notes are that it is highly focused with most of the features available being directly related to the process of note taking, it was created by a trusted developer, it comes already loaded on apple devices, it allows for the sharing of notes, and it allows users to back up their notes. *[Info: Shortened pros list]*

The cons of Notes are that it has some extra functionality that can either improve user experience or negatively affect it, users need to have an iCloud account to create shared notes, and there is no undo functionality. *[Info: Shortened cons list]*

I created this flowchart based on the core functionality of the Notes app and where potential databases could have been accessed and edited through the app. *[Flowchart moves to middle]*

**Custom App Information**

*[Slides have Custom App Flowchart on Right Side and Info on Left]*

The custom app I created based on the Notes app is called Penny for Your Thoughts. It is also a Productivity note taking app. *[Info: Given info and creator]*

It simplifies the note taking process by allowing the user to use only unformatted text in their notes. This is different from the Notes app which gives the user an unnecessary amount of extra functionality when it comes to creating and editing notes, such as allowing for attachments, bold and italic text formatting, and scribbling on the notes page. Penny for Your Thoughts also differs from the Notes app in that it requires users to create an account or login before the user can view their profile and create and edit thoughts. *[Info: Notes vs PfYT lists]*

Penny for Your Thoughts is better than the Notes app because it minimizes the possibility of an unfixable mistake due to an accidental selection or click and it’s more secure because it ensures that users can only view their thoughts once they’ve logged in rather than allowing anyone with access to the device and the program to view and edit thoughts. *[Info: List of reasons its better]*

The flowchart depicts the possible actions the user could take and the results the user would receive. *[Flowchart in middle]*

**Penny for Your Thoughts Walkthrough**

*[Slide: Title and video of walkthrough with some commentary]*

Show Database:

The users table keeps track of the users created and is used to allow the user to login.

The thoughts table keeps track of which user created the thought, what the thought contains, a preview of the thoughts content, and the date and time the thought was last updated.

Show Program:

The first screen the user encounters is the login screen. It gives the user the option to exit the program by pressing 0, create an account, or to login by providing their username and password.

After validating that both the username and password match, or creating a new account, the user is taken to their profile page. This page displays their username, name, and thought count. The user can then choose to logout or view their thoughts.

Logging out would take the user back to the login screen. Choosing to view their thoughts takes the user to the thoughts page. Here the user has the option to search their thoughts, select or create a thought, or view their profile.

Choosing to view profile would take the user back to the profile page. Choosing to search thoughts would allow the user to enter a search term and then return any thoughts containing the search term.

Choosing to select a thought would display all the thoughts the user currently has, if the user doesn’t have any thoughts yet then they will be asked to create a thought first. Once a user has selected a thought they will be asked if they want to edit or delete the thought or go back.

Editing the thought will show the user the current thought and allow the user to type in whatever they want the updated thought to contain.

Deleting the thought will delete the thought completely. This cannot be undone so the user is asked twice.

Choosing to create a thought will ask the user to enter whatever text they wish which will be the contents of the thought.

The app interacts with the database in nearly every choice the user makes. Can you identify where the app contacts the database based only on the user input and returned results?

**What was Learned:**

*[Slide: Title and list of learned points]*

Working on this project throughout the month has taught me many things.

Now, whenever I use an app or a program, I give at least a little thought to how the program works and what goes on behind the scenes that I as a user don’t see.

I’ve also started to dream in terms of data and databases, trying to organize random data I’ve picked up throughout the day into normalized data tables.

I understand the thought process behind creating a functionality-based flowchart and how to recreate certain basic functions such as a search function.

The thing that struck me the most working on this project was how much effort and thought goes into even the most seemingly simple parts of a program.

I have a newfound appreciation for the intricacies of even the simplest apps.

**Thank you**

*[Thank you screen with author credits]*

Thank you.