

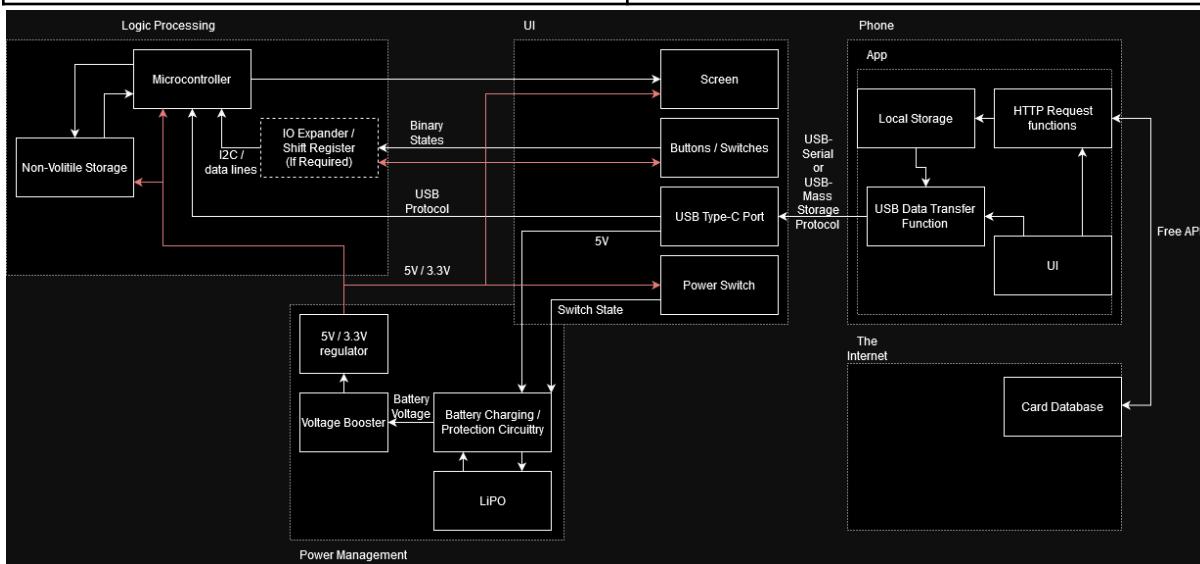
Nathan Worklog

| | |
|--|----------|
| 9/17/2024 - Discussion with TA..... | 2 |
| 9/23/2024 - Design Considerations..... | 3 |
| 10/8/2024 - Initial Setup of the App..... | 3 |
| 10/12/2024 - API querying with App..... | 4 |
| 10/14/2024 - Screen display..... | 5 |
| 10/23/2024 - Organization / Preparing for Export..... | 6 |
| 11/16/2024 - Adding Basic USB functionality..... | 7 |
| 11/23/2024 - Debugging USB..... | 7 |
| 11/26/2024 - Completed USB functionality..... | 8 |
| 12/02/2024 - Final Testing / Adjustments..... | 9 |

9/17/2024 - Discussion with TA

Discussed high level requirements of the project with the TA and explained all the subsystems and what functionalities they each entail. Created a block diagram and a visual aid showing what the design would look like (may be subject to change). High level requirements are shown in the table below.

| Category | Requirements |
|----------------------|---|
| Physical Constraints | <ul style="list-style-type: none"> • 4.2 Inch Screen • Text/Image legibility from at least 60 cm away • Dimensions of 100 x 70 x 10 mm |
| Hardware Features | <ul style="list-style-type: none"> • Response time of less than 1 second from inputs • 4 hours of use after one full charge • Enough storage for 10 unique tokens • Each token type storing 16 unique variations, each having to be able to manage 255 identical copies of each variation |
| Data Transferability | <ul style="list-style-type: none"> • Upload 10 unique tokens within 2 minutes • Search for and select tokens within 30 seconds |

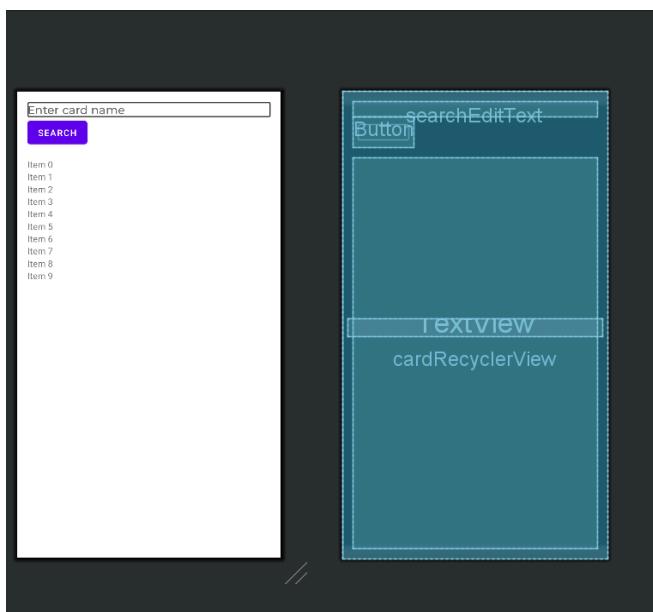


9/23/2024 - Design Considerations

Started work on the design document and considered different approaches to the creation of The Tokenizer. Thought about using iOS vs Android for the companion app, but ended up opting for Android due to compatibility reasons and since our group had more familiarity with app development in Android Studio. Also had to consider using the memory embedded within the microcontroller vs. external storage such as a SD card. Unsure of what to do here but for now sticking to internal memory but this is subject to change.

10/8/2024 - Initial Setup of the App

Began to create the layout of the app, starting with the general interface. Have not worked on querying the API database yet. Version control via a [github](#) repository will be used for collaboration of developing the app. Shown below is the layout of the “search tab”.



10/12/2024 - API querying with App

Connection with the [scryfall](#) database has been established, users can now search up a token and see any related results after clicking the “search” button. The time it takes for results to show up after searching the query is almost instantaneous, which means that the 30 second requirement is

easily met.

The screenshot shows an Android Studio interface. On the left, the code editor displays `MainActivity.java` with several red circular markers indicating potential issues or warnings. On the right, a search results window titled "cyclops" is open, showing a list of card names found in the code. The list includes:

- Bloodrock Cyclops
- Bloodshot Cyclops
- Chainwhip Cyclops
- Cyclops Electromancer
- Cyclops Gladiator
- Cyclops of Eternal Fury
- Cyclops of One-Eyed Pass
- Cyclops Superconductor
- Cyclops Tyrant
- Ekundu Cyclops
- Erratic Cyclops
- Flummoxed Cyclops
- Fortress Cyclops

The bottom of the screen shows the Android Device Monitor and the Android Logcat output:

```
PI 29'.
W cat=[android.intent.category.LAUNCHER] cmp=com.example.tokenizerv2/.MainActivity }

52345', transport: 'socket'
host:52345', transport: 'socket'
```

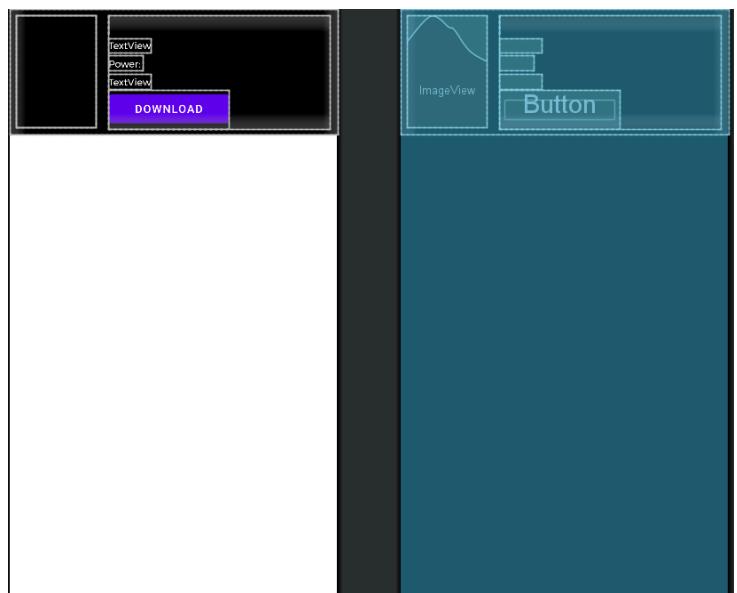
10/14/2024 - Screen display

The display screen has been received and is confirmed to be working. Currently, it is only shown to display changing colors and not tokens as of yet due to the fact that USB transfer within the app has not been implemented yet. The screen size meets the high level requirements of having a 4.2 inch LCD display. The other requirement of being seen from 60 cm away cannot be tested yet as the token display is not functional.



10/23/2024 - Organization / Preparing for Export

Adding a more comprehensive view of searched cards in the search tab, now showing the card image and its related attributes. Also added a download button within the search tab, that downloads the image as a byte array and the associated data such as power, toughness, name, text summary, etc. Once downloaded, it appears on the downloads tab. An “export” button exports the data to the microcontroller. As of now, the export button does not have any functionality, it just exists.



11/16/2024 - Adding Basic USB functionality

Created a rough sketch for USB communication between the app and the device. Since our PCB design is not fully developed yet, there is no way to test this functionality. USB functionality cannot be tested on the emulated device native to Android Studio. Design flow includes sending all of the data through a bulkTransfer function and obtaining permissions via an imported Java library function.

11/23/2024 - Debugging USB

After running an actual test on the USB code, it was found that it caused the app to crash. Mainly was an issue of how the app is handling USB permissions because it would not even make it to the bulkTransfer function. Tried a different approach to USB by using a different library and using write() and read() functions to send data when the microcontroller was ready to receive data.

11/26/2024 - Completed USB functionality

Able to finally complete USB functionality. Required a different approach to handling USB. Since the buffer of the microcontroller cannot receive the data all at once due to the file's large size, it is sent in multiple smaller chunks. After some debugging it was found that there were a few issues, including setting a permission flag and other issues on the Android Studio end, device can now reliably receive data from the app. Conversion of image file format byte array to bmp was also necessary for image cropping so that the display on the device looks neater.

12/02/2024 - Final Testing / Adjustments

Design is now essentially complete. The device has been tested by a few people and was found to be user friendly and created a neater view of the game state. Some extra things were added to improve the ease of access such as labels for the button that were drawn on with sharpie.

