Mohammed Alam
ISTE 252
Professor Steven Cady
5/2/2022

For this project,t the hybrid app there were a few ideas in my head. I kind of accidentally got a headstart on the project making a wireframe with all the features I was letting my creative mindset go. We were working on Project 2, but I kind of did Project 3's prototype. Then I shared the design with you, and you kind of said it was a bit overkill. I was using at least 4 APIs, and now that I completed the project, I would more than have to agree as well.

So I settled on only one component of this grandiose project and decided to call the new app the Rochester Weather Map App. I wanted to essentially make a project that someone could potentially use. Something that will need a few clicks, and typing wouldn't be necessary. So what I did was premake a list of cities, and display it in a simple list view with the style, of course, so you can just choose a city and run the app.

Now for the color scheme of the app, I decided to make the initial color scheme something that would best suit the eyes when waking up in the morning. You've just woken up, you're in a dark room just opening your eyes, and blam a face full of light. I decided to match the tone using a dark purple/indigo and blue theme. To match the nighttime.

For the page layout, I got the idea from project 2. I decided to make one page that was an intro, with the name, a logo, and a button to start. Also, this would help with loading times, opening a simple display, and letting a user look at it for the rest of the pages to load. The list view came from my JSON file and lab 3. In the lab, I had to make a listview from a string list, and it looked similar to project 2, I thought, hey this looks nice, I can use it and tone it again to suit various cities.

Then for the actual page, I decided not to go the project two route. I decided to be creative, and focus on user experience. I didn't want to block the user with a wall of text so I decided to add an image, and the OpenWeatherAPI comes with an icon easily available through flutter. So next, I decided to include various fonts & styling for the eyes. I centered the description below the image, such as cloud rainy, etc. Then I decided to increase the font on the temperature, putting it in both F and C, to make it universal. Miscellaneous details such as max, min, and feel like temp is in a list view below. Sunset and sunrise are in a right-aligned text list view and given colors matching sunrise and sunset.

On to the actual problems I encountered, one of them was the actual first page, albeit being simple, the margins and spacing were a bit awkward, and I had to research ways of spacing efficiently. The Button is kind of a bunch of nested components but it worked out so well. One huge error I encountered was the use of Futures. I didn't know how to use any of them or what they were, so I spent days reviewing the JSON example you provided, there were some outdated parts there, so I needed help from that and to do lab 3 making that part more

strenuous. It took some trial and error, but I learned it in the end. Also, there was an actual OpenWeather dart library for classes and types, so I thought using that would be more efficient instead of using a JSON request. So I had to replace JSON and API implementation from lab 3 with all that. I go the back end working, and it was smooth sailing from there.

One bonus error, probably the hardest was the fact that flutter doesn't install correctly on mac, one of the requirements was an ios screenshot. I don't have an iPhone, but I do have a MacBook, which I regularly use and the ios simulator available on the Mac version of the android studio was a huge headache wasting hours. I had to update the flutter and put it in the path permanently because it doesn't do that. Use a sudo command to update flutter, and I'm just happy it got resolved.

The final part which I had really fun doing was the designing for the last page. I had API data in a weather class, and just playing around with the class, I realized I could get individual components like max temperature, in Fahrenheit, rounded to 2 decimal spaces. Using trial and error, it worked out in the end, and I essentially perfected it with typography elements and some color. I decided to make a go back button at the bottom so an ios user won't have to go press the back button on the top left of the phone, not very convenient.

As for the future, I believe I can implement different higher quality jpegs and use functions to put those as the weather image instead of the open weather icons which kind of have small pixel density. Another feature I wanted to do was make it so that based on sunrise/set times, the color of the app will follow those conditions. The dark theme for nighttime and a lighter, blue, white, and yellow theme for daytime. Another thing that I spent hours doing after submitting the dart files, was upgrading the dependencies. I removed all non-used dependencies and used weather 2.0.1, which removes all previous bugs, sometimes open weather will have null values for cities, so upgrading the weather version and optimizing code for that worked out.