Minor Project - 1

Submitted in partial fulfilment of the
Requirement for the award of the degree of
Masters of Computer Application

“Live Weather App”

Submitted To

Dr. Rajan Gupta

Associate Professor

Submitted By

Yashovardhn

00117704421

(Batch: 2021-2023)

Vivekananda Institute of Professional Studies
(Affiliated to Guru Gobind Singh Indraprastha University)

Qwyk Weather
A Weather App

S. No.	Table Of Contents	Pages from-to
1	**Planning**	**4-6**
	i. **Initial research**	4
	ii. **Project Specification**	5
	iii. **Scope of Project**	5
	iv. **Possible Risk Involved**	6
2	**Navigation and User Interface Design**	**7-8**
	i. **Wireframes**	7
	ii. **UIX Flow Chat**	8
3	**Analysis**	**9**
	i. **iOS User Analysis**	9
	ii. **Scenario Analysis**	9
4	**Design**	**10-11**
	i. **Architecture Design**	10
	ii. **Swift Files Design**	11
5	**Implementation Details**	**12-14**
	i. **Class File Structure**	12-13
	ii. **API Code**	14
	iii. **Xcode Files**	14
6	**Testing Plan/Deployment**	**15-18**
	i. **Planned Testing**	15
	ii. **Functional Testing**	16-17
	iii. **Adjustment/corrections to be conducted**	18
	iv. **Other testing activities to be conducted**	18
7	**Code Structure & Description**	**19**
8	**Future perspective**	**20**
9	**Things I Learned**	**20**
10	**Limitations**	**20**
11	**Summary**	**21**

Planning

Initial research

There are a variety of weather iOS apps in App Store. Those apps have great features and functionalities to satisfy users. However, according to my research, only a few of them have friendly user interface and human centred interactions, which means that a lot of them might be difficult to be navigated even though they provide enough functionalities. It is not convenient for new users.
Therefore, I would like to do improvements on weather iOS apps. The objectives include:

The iOS app allows people to check out the weather in multiple cities worldwide. The weather data is dynamic, which means that users can see the weather anytime.

The iOS app not only show the weather, temperature and humidity, but it also uses various icons to represent the weather accordingly. It will be easy to read and use.

Besides, the iOS app will have friendly user interfaces and human centred interactions. Users can find the information they want in a short time and limited clicks. It is easy to be navigated than other weather iOS apps in the market.

Development environment: Xcode

Open Weather Map API documentation:

https://developer.forecast.io/

Project Specification

- Real time weather forecasting.
- System : MacOS MONTEREY V12.3
- System Specification : 1.4 GHz Quad-Core Intel Core i5
- Graphics : Intel Iris Plus Graphics 645 1536 MB
- Memory : 8 GB 2133 MHz LPDDR3
- Platform: iOS.
- IDE: Xcode.
- Takes user's geolocation as input to provide weather forecast by using apple's CLocation Manager .
- Displays detailed weather information for the current.
- JSON data from the site.
  i) API used to fetch data fromwww.openweathermap.org

Scope of the project

The scope of this iOS app will not be too broad. On the contrary, I will narrow down and only focus on a few functionalities which have high usage frequency. They are basic and also they can totally satisfy users' needs.

The main functionalities include:

- View the temperature.
- View wind data
- View current approx location
- The symbol and its colour will be changed according to the conditions.

Possible risks involve

- API

I got weather information from open weather API, due to I failed bunch of times accessing to the open weather map's API. I have no idea why I failed. I registered and got an API key; however, I still failed.

- Development

I did not have any experience with apple iOS app development, even with swift programming, therefore, making those elements work definitely will be a big challenge for me. I will do research first, or watch some tutorial videos online, practice by myself and then apply them on my app.

- Testing

Different devices have different layouts and screen sizes. The GUI design and development might not fit all kinds of devices.

Meanwhile, different users might have different opinions for this app and I need to consider which feedback I should listen and which I should not.

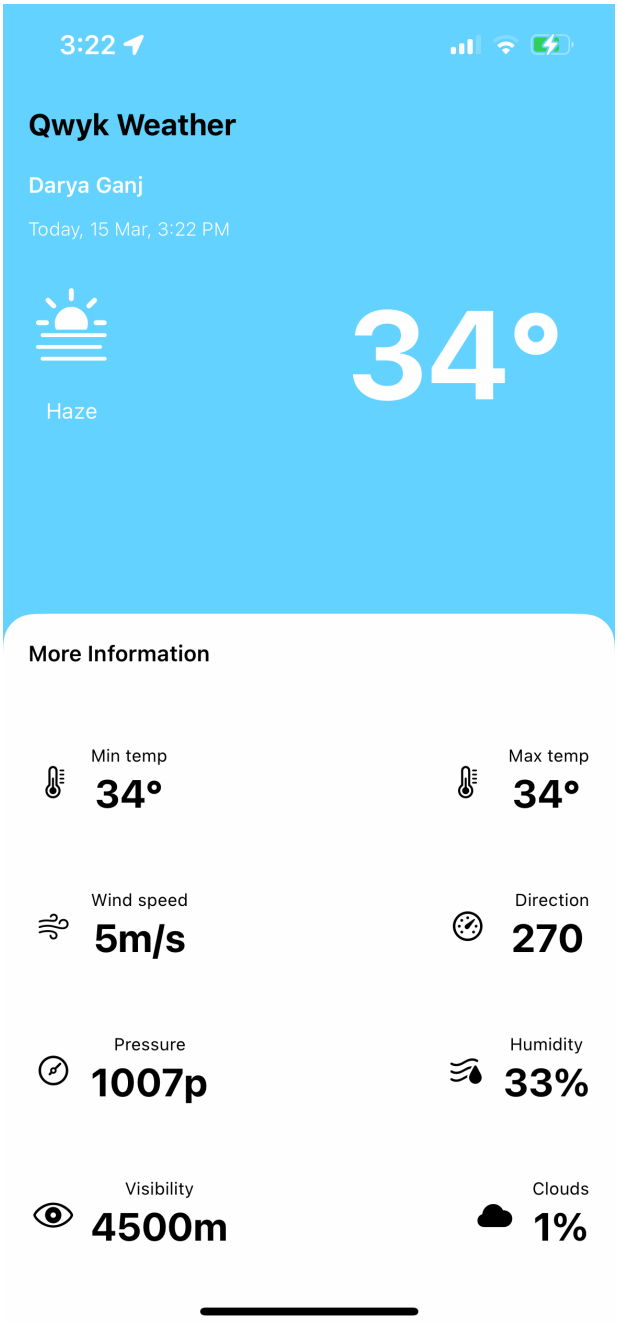
- Debug

It might take a long time to debug and the iOS app might have few unexpected bugs. It takes time to do research and fix. I am not sure how long it will take, but I will try to make them be solved in a certain time and this project is still under control.

Navigation and User Interface Design

Wireframes

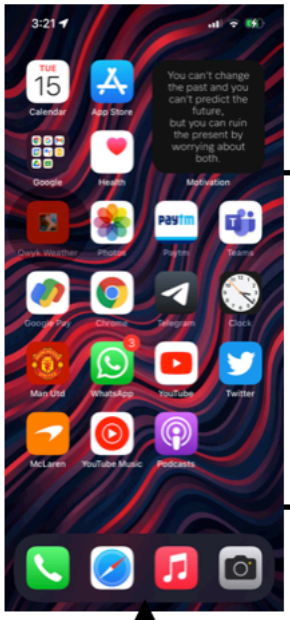
I designed wireframes for this weather iOS app which basically can display the information processes and how it works.



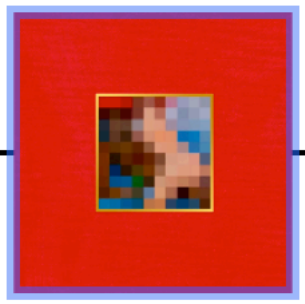
HomeScreen

App Icon

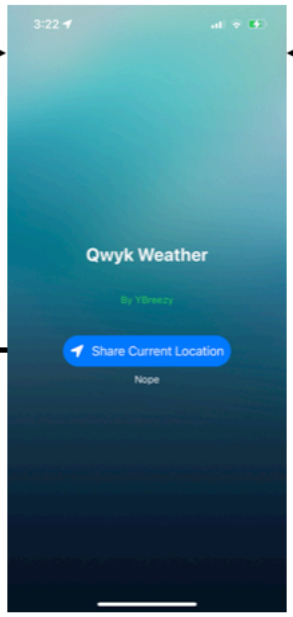
Launch Screen



Select App



Open App

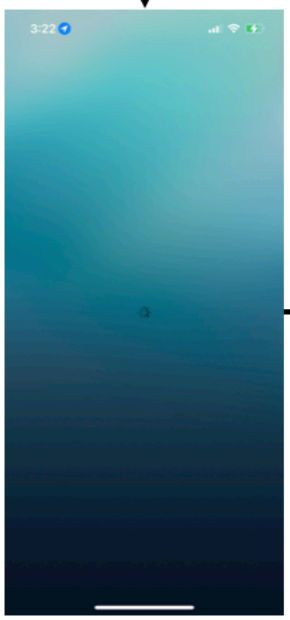


No

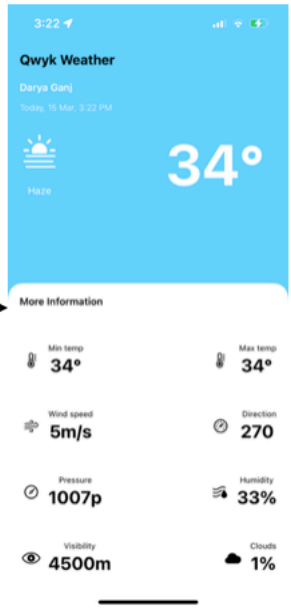
Asking for Location

Yes

Welcome View



Loading



Weather View

Analysis

IOS user analysis

Basically, every iOS users could be the users of my app. However, especially for people who would like to see the weather every day and decide what to wear tomorrow will be the target users for this app.

- The main target users include:
- Business men/women
- For business men or women, they need to view the weather information before they go to work every day, so that they will know what to wear, such as long sleeve t-shirt or short sleeve t-shirt.
- Mothers
- Every mother cares about their children, so another group of target users could be mothers. They will view weather information every morning to know how to prepare clothes for their children.
- Travellers
- Travellers will view destinations' weather information before they depart. Also, before they go back to home, they need to view weather information again.

Scenario analysis

- Screen and Interaction Analysis

The users will use this iOS app on iOS devices . All the information of this iOS app will be displayed full screen.

- Usage Analysis

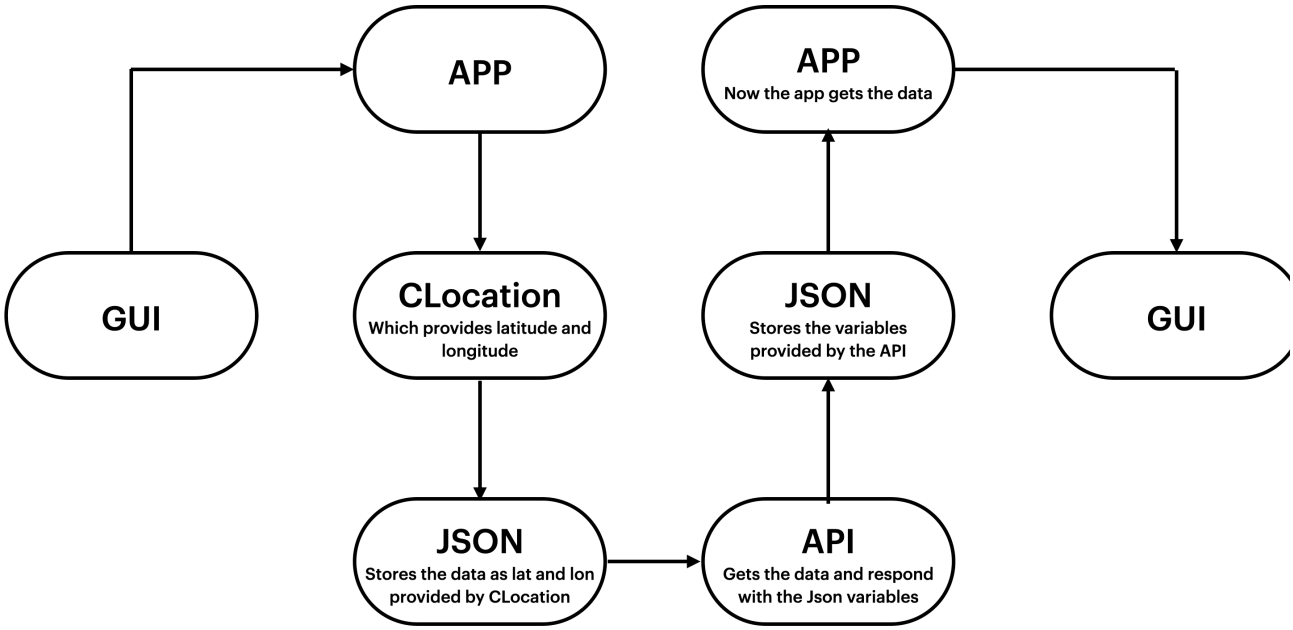
Users can use this iOS app on the morning every day at home, on their way to travel, and other situations as long as they want to know weather information.

- Environment Analysis

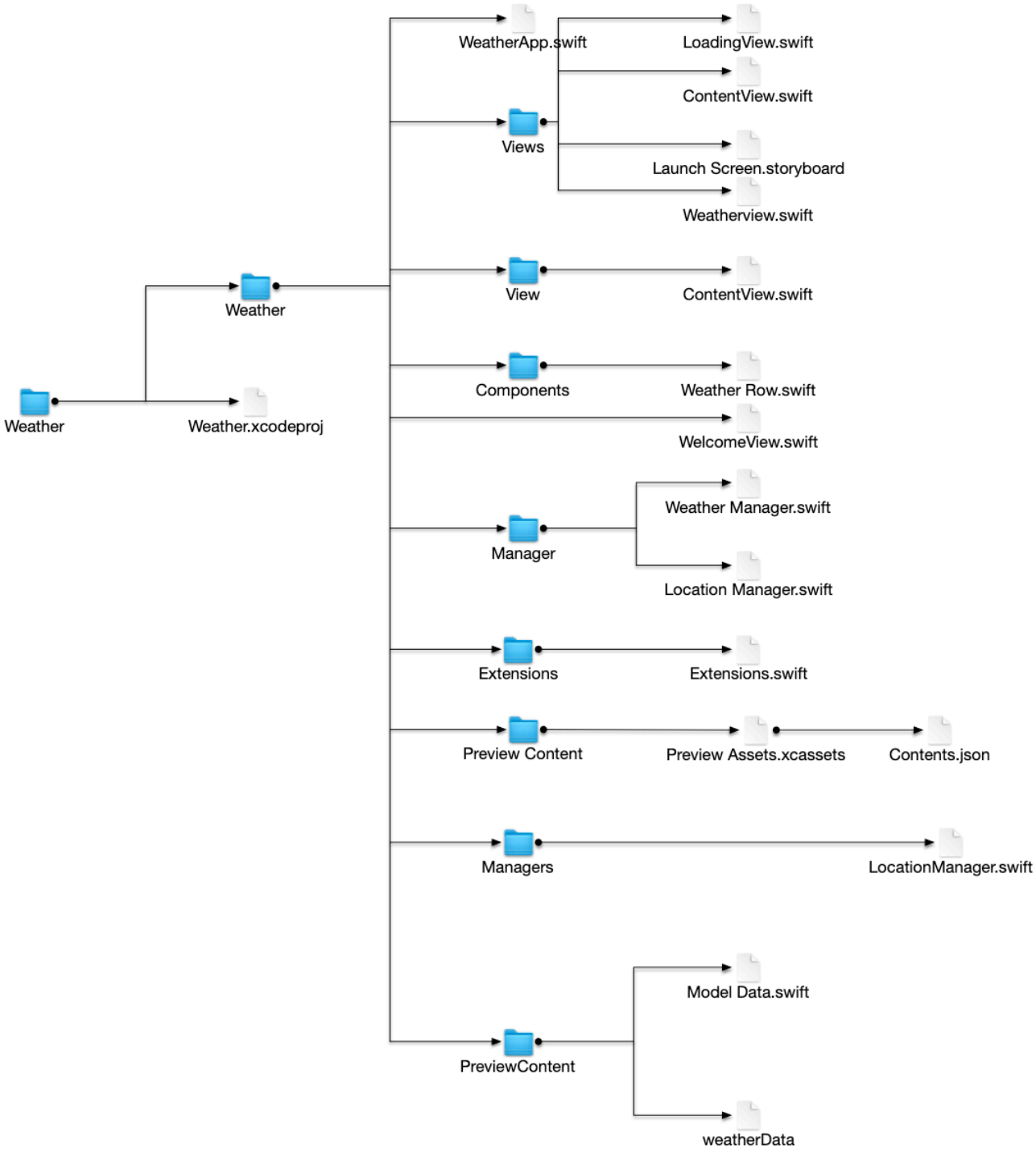
This iOS app only can be used on iPhones , iPads . It will access to the open weather API to get the weather information. It sends requests, and then get responses from the API through the internet.

Design

Architectural design

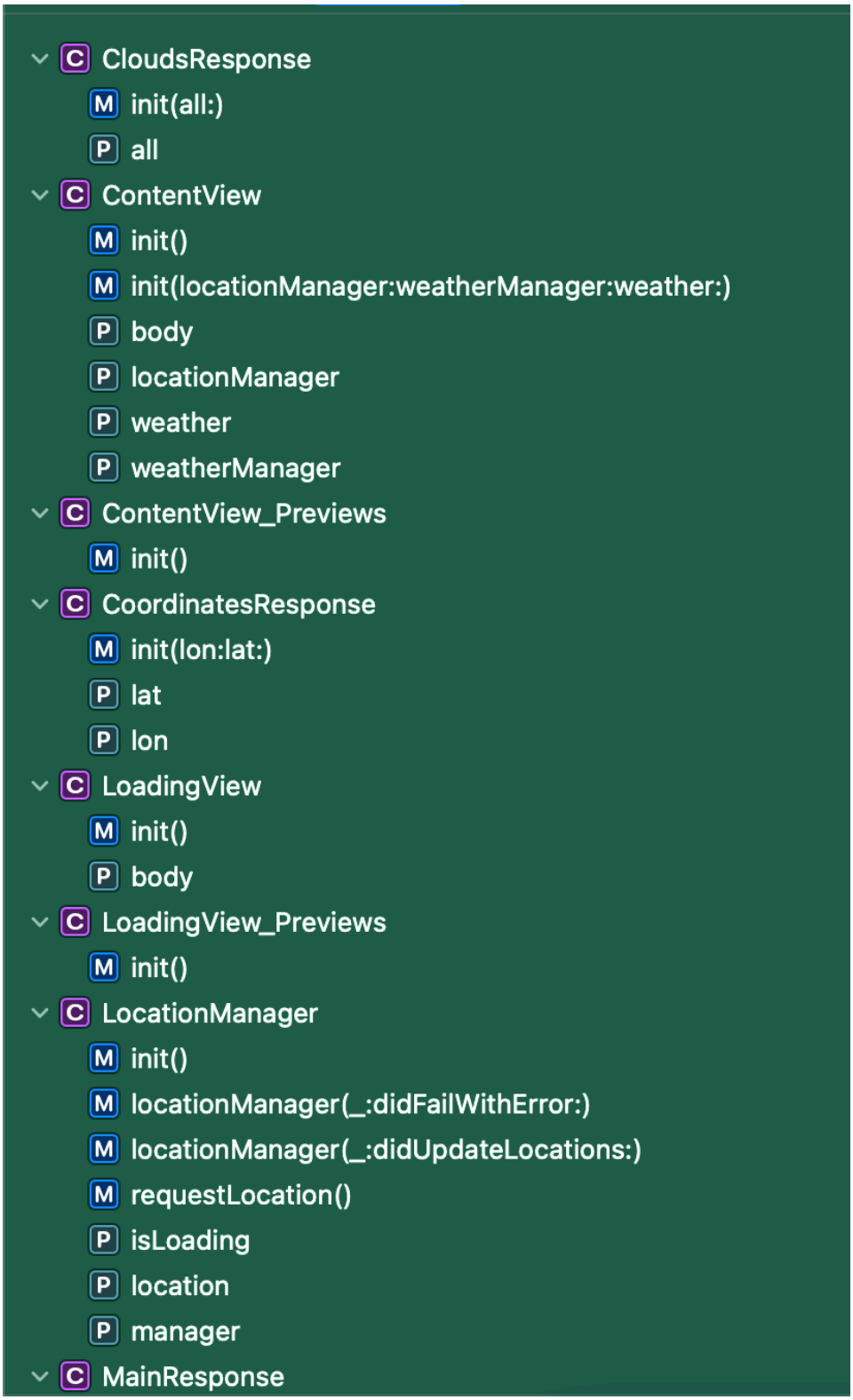


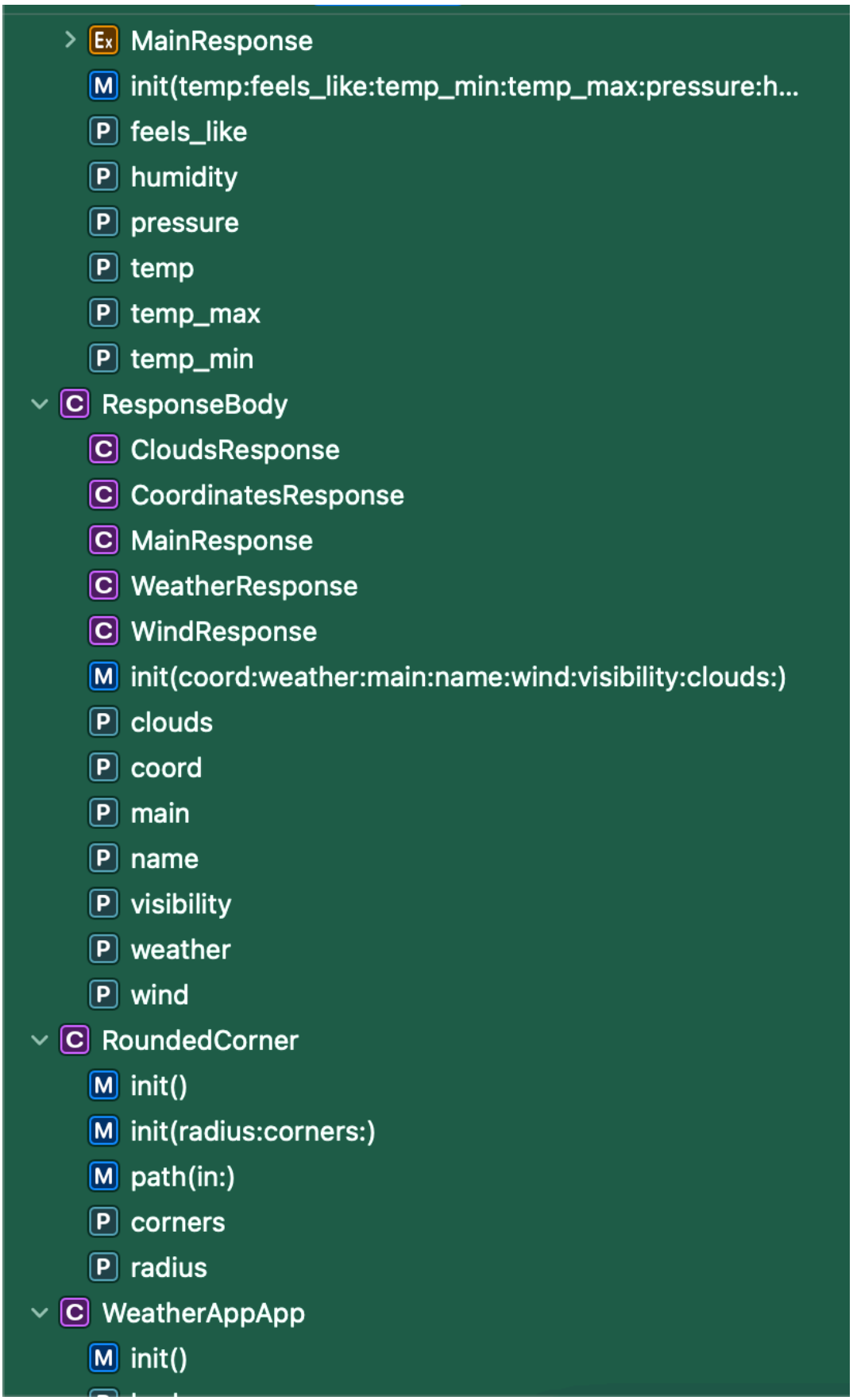
Swift files

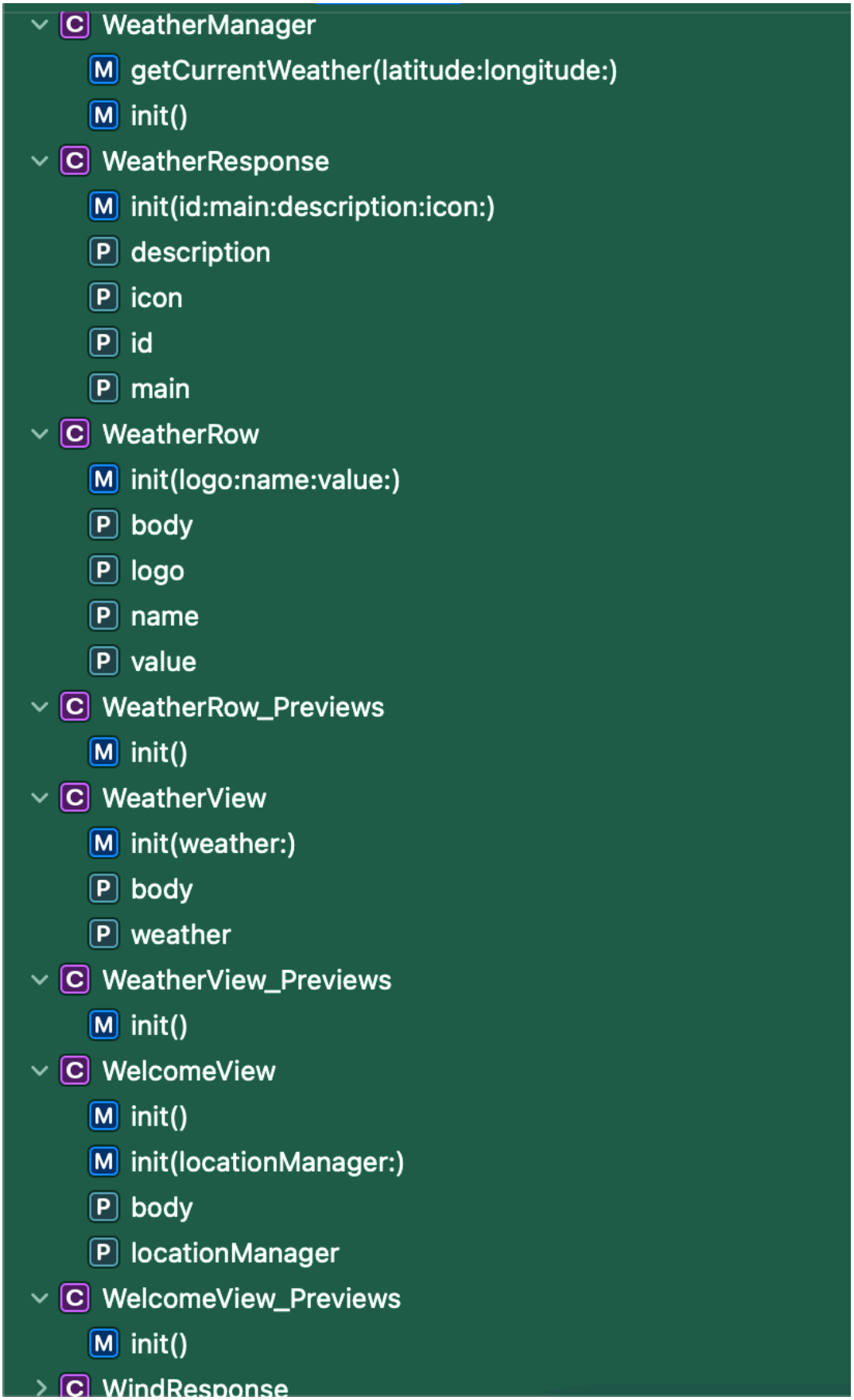

WeatherApp.swift
LoadingView.swift
ContentView.swift
Views
Launch Screen.storyboard
Weatherview.swift
Weather
View
ContentView.swift
Components
Weather Row.swift
Weather
Weather.xcodeproj
WelcomeView.swift
Weather Manager.swift
Manager
Location Manager.swift
Extensions
Extensions.swift
Preview Content
Preview Assets.xcassets
Contents.json
Managers
LocationManager.swift
Model Data.swift
PreviewContent
weatherData

Implementation Details

Class files




MainResponse
init(temp:feels_like:temp_min:temp_max:pressure:h...
feels_like
humidity
pressure
temp
temp_max
temp_min
ResponseBody
CloudsResponse
CoordinatesResponse
MainResponse
WeatherResponse
WindResponse
init(coord:weather:main:name:wind:visibility:clouds:)
clouds
coord
main
name
visibility
weather
wind
RoundedCorner
init()
init(radius:corners:)
path(in:)
corners
radius
WeatherAppApp
init()


WeatherManager
getCurrentWeather(latitude:longitude:)
init()
WeatherResponse
init(id:main:description:icon:)
description
icon
id
main
WeatherRow
init(logo:name:value:)
body
logo
name
value
WeatherRow_Previews
init()
WeatherView
init(weather:)
body
weather
WeatherView_Previews
init()
WelcomeView
init()
init(locationManager:)
body
locationManager
WelcomeView_Previews
init()
WindResponse

APIs, code examples or libraries to be used API

String apiKey = "084fa305d2db857db56883843b8c8d48";

String forecastUrl = "https://api.forecast.io/forecast/" + apiKey + "/" + latitude + "," +

longitude;

Code examples

Xcode files

```
import Foundation
import CoreLocation

class LocationManager: NSObject, ObservableObject, CLLocationManagerDelegate {
    let manager = CLLocationManager()

    @Published var location: CLLocationCoordinate2D?
    @Published var isLoading = false

    override init() {
        super.init()
        manager.delegate = self
    }

    func requestLocation() {
        isLoading = true
        manager.requestLocation()
    }

    func locationManager(_ manager: CLLocationManager, didUpdateLocations locations: [CLLocation]) {
        location = locations.last?.coordinate
        isLoading = false
    }

    func locationManager(_ manager: CLLocationManager, didFailWithError error: Error) {
        print("Eror getting location", error)
        isLoading = false
    }
}
```

Testing plan/Deployment

Planned test

I conducted a testing plan to see if the iOS app can connect to the server and get JSON data in while I am programming. Also, I will fix bugs during the development progress as many as possible. Besides, there is another test plans that let me test the entire iOS app.

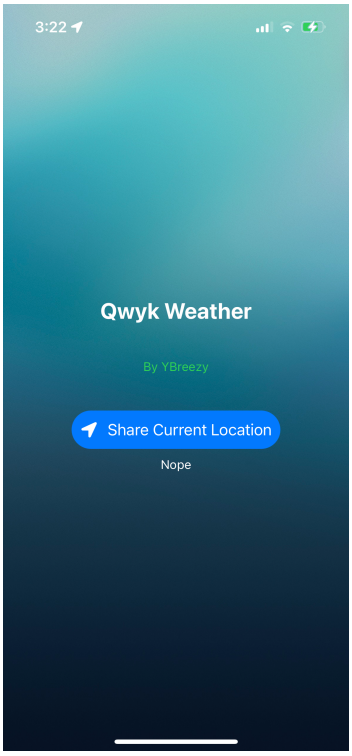
Here is the basic schedule: White Box Testing

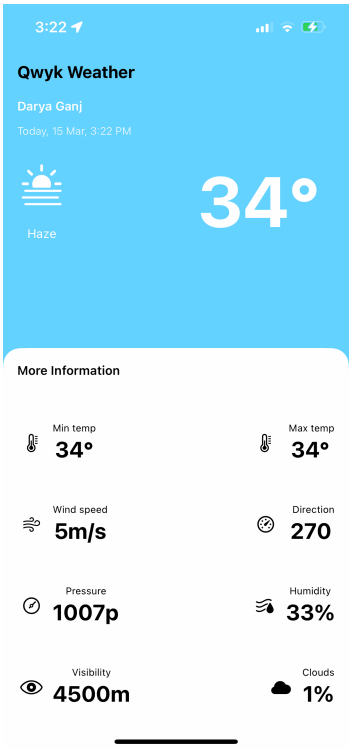
Testing Case	Testing Result
Launching the App	Working
View the weather condition (if it is right or wrong)	Working
Splash/Launch Screen	Working
Location CL Api	Working
Different Background for different weather condition	Not working
JSON data for different variables shown on the app	Working
Exit app if location called or user denies the location	Working
Different icons for different conditions	Working
Weather Row to differentiate the more info tab from the rest	Working
Add 3 days data	Not working
On simulator of different sizes	Some working

Functional testing

As I said above, few functionalities work well.

- The iOS app can connect to the internet/server
- The iOS app can send request to the API server
- The iOS app can get JSON data from the API server
- The iOS app can parse JSON data and display all the data on the screen
- Users can view their current location weather information
- Users can jump from welcome page to exit app thus denying location
- The data is displayed accurately without any difficulties or hiccups
- Working and integrating splash screen using storyboard





User 1	Result	Feedback
Location Access Request	Working	Not very convenient
Exit function	Working	No content
Detail weather information	Working	This is good

User 2	Result	Feedback
Location Access Request	Working	I would like to add more cities if possible
Exit function	No……	
Detail weather information	Working	The colors are too light

User 3	Result	Feedback
Location Access Request	Not Working	Ipad doesn't support this app as of now
Exit function	No……	
Detail weather information	Not Working	

Adjustment/corrections to be conducted

- UI design:

The UI design does not look good due to the size of the icon. I will modify those icon or reset the size of them to make it look appropriately according to the size of the screen.
Besides, the xml file is using relative layout, so some elements will look differently after modifications. I will do more tests to ensure this iOS app has friendly user interfaces.

Other testing activities to be conducted

- User experience

The user experience might be different when I test the app on a real device, due the resolution and the screen size. I will definitely do adjustments based on the data that I get as stated above with the iPad and maybe a phone not of the aspect ratio of 18:9.

- Functionalities

Few functionalities might be changed due to technical reasons. For an instance, I planned to display the basic weather information in a small widget in the home screen of the device. If this functionality cannot be implemented eventually, I probably will cancel it.

Code Structure & Description

Weather Row : to Create the more information tab in the app.

Extensions : contains rounddouble function and the Share Current Location button config

LocationManager : to use cloactionmanager to retrieve the location coordinates.

WeatherManager : to retrieve the info elements from json api and pass the location

ModelData : to relieve response body to retrieve json info to var and string and all

Weather Data : sample json code

Launch Screen : splash screen and uses the swiftui file.

ContentView : used for sending dummy location data the first time.

WelcomeView : code for the first intractable screen.

LoadingView : code for the loading symbol while the system in interacting with the API

WeatherView : code for all the displayed weather information.

Assets : contains images and app icon and gradient colour used in the app.

Future Perspectives

Few functionalities can be improved or accomplished

- Make the multi languages functionality works
- Users are able to select more cities from the search bar
- Users are able to add more cities in the list
- A Weather Widget
- A iPad and Apple Watch app
- Users will be able to see Weather forecast

Things I Learned

- Xcode UI and its working
- Using API on a project or app
- Swift and Swiftui
- Storyboard and its working
- Apple API
- SF Symbolsk

Limitations

- Unpaid APIs provide incomplete services.Many details cannot be fetched
- Often tuples of upcoming days remain empty once again due to free APIs
- The GMS API (Google Manual Search) is actually keyword based that might only provide data of few discrete locations.The data might not be precise and continuous.
- Language diversity could have been implemented.Multilingual apps make it easy for users worldwide

Summary

After taking this Project, I got much more confidence in iOS development. I am so proud that I have learned a lot from this project.

I used to self-learning iOS development and did not make any apps during that time. However, in this project, I developed a functional iOS app, even though it is not perfect.

Working on assignments, I learned more about a variety of elements of iOS and how to make them work. I learned Xcode. This project was a good opportunity to practice this skill and now I also know how to continue self-learning after this course.

Working on the project, I practiced the rapid development process, doing testing and enhancement. Meanwhile, I developed the second version, which has many improvements. The value of designing and developing this iOS app are to clarify the design process, created those files by myself, do research to fix bugs and test it with target users.