**WEATHER ASSIGNMENT**

**WEATHER SPACES APP**

**BY**

**ANGELBERT IMMANUEL SOLA**

**2963154**

**ABDELNOUR MOULAY SLIMAE**

**3020610**

**DEVENDRA DEEPKIRAN ADHIKARI**

**3015840**

BY ABDELNOUR

To tackle this project successfully we as a group decided to create a Gantt chart to understand who is doing what and in what timeframe. I created the template for the chart and amongst us decided what will be designated to each other. Another aspect I took part in was creating our decision tree that will project how our chatbot will function. That was completed in less than a week. To make sure the API was working properly, I took it upon myself to test the API and make sure it worked properly and it did. Along with that I created 2-unit test cases for the chat bot to make sure some aspects of it worked.

BY DEVENDRA

**API SELECTION:**

With team members’ approval, I used Open Weather API for 5 day/3 hour forecast because of the following reasons:

1. It is simple and efficient to get started for every one of us. We just needed the same API key which we could get by just registering at Open Weather webpage.
2. Unlike most of other APIs, some of the parameters like latitude, longitude is optional which makes this API user-friendly. Users just need to input city name to get the weather.
3. This API provides all the description that we needed to make the bot which include weather description, real feel and the location is international.

I wrote python code to connect with this API and display the desired data in Json format on the console. I uploaded the code on GitLab.

**API TESTING:**

I tested the API using Pytest. It includes API connection test, Parameter Test, Date Test and returned Data test.

**Helped Teammates:epl**

I helped teammates on making decision trees, formatting chatbot logic, Gantt chart. I wanted to display only one picture for clothing recommendation according to the weather displayed at the GUI. It did not go as I expected and have to terminate the effort.

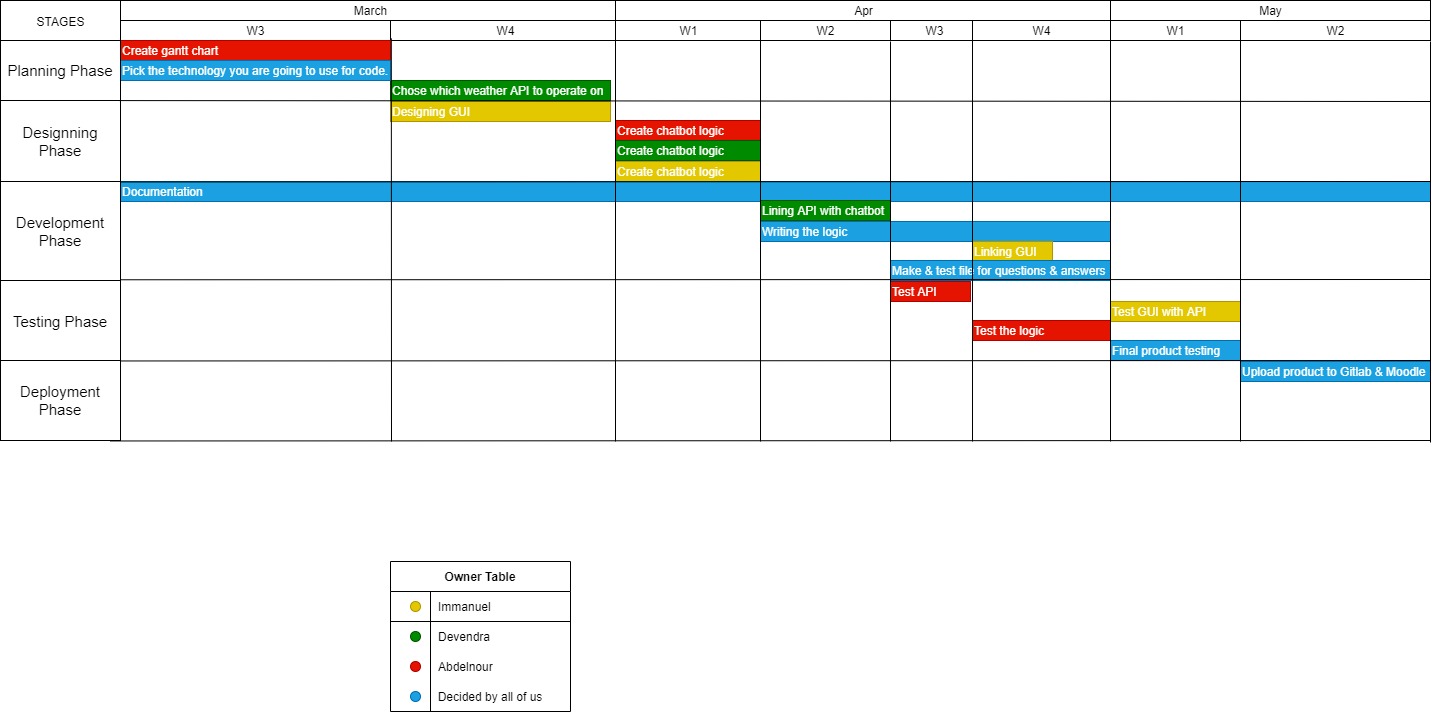
BY ANGELBERT IMMANUEL

**PICKING THE TECHNOLOGY**

For our group project we were tasked with creating a Weather Application called “Weather Spaces” which printed out the weather in any city we requested, it had a chatbot which allowed users to communicate with the bot. Although this was a challenging group endeavour we managed to prevail, we had to give ourselves individual roles and tasks to help with the succession of our project. For the project I was part of the crucial deciding process which was picking which technology to use for the App and writing the chat logic. I was tasked with creating the Gui, the chatbot itself, linking my Gui with the Api my team member created and testing the Gui with the Api for final product testing.

We as a group decided to use Python programming language as the sole program, our thought process is that it was the most popular programming language in the previous years. Python is one of the most accessible programming languages available because it has simplified syntax and not complicated, which gives more emphasis on natural language. Due to its ease of learning and usage, python codes can be easily written and executed much faster than other programming languages. We laid out our project schedule and task list with our Gantt chart that we created with each team members role.

**GANTT CHART**

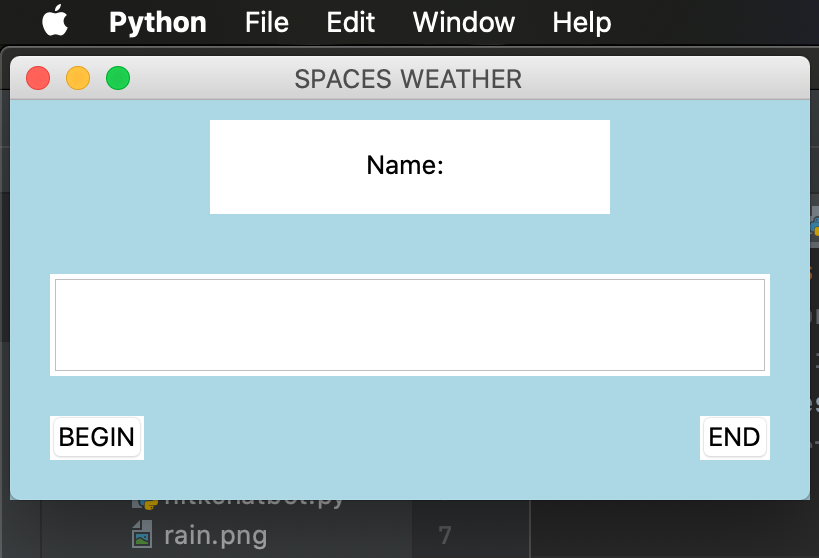


**DESIGNING THE GUI**

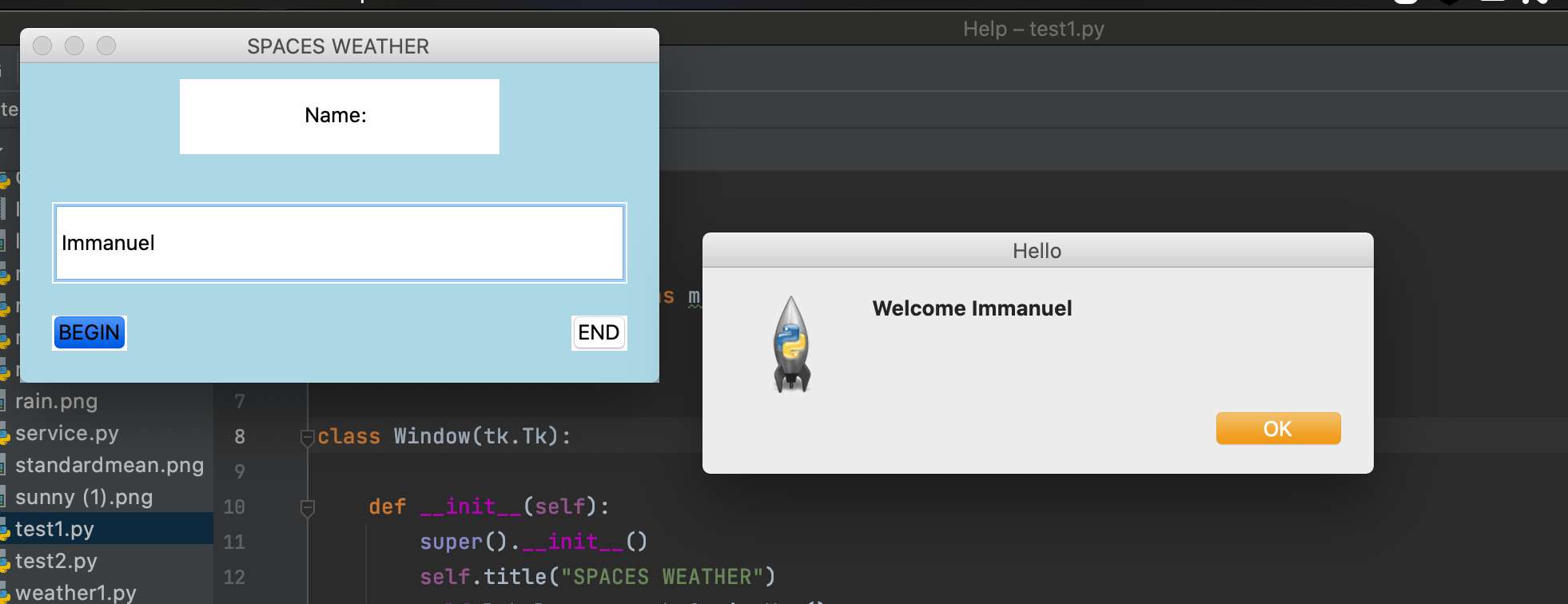
During the process of designing the gui I used Tkinter which is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit, and is Python's de facto standard GUI.I felt it was the best suited Gui to use for this weather application especially with the way we had thought out the project layout in the early stages. Python is a new language for me personally and I learnt a lot from it, I found certain aspects extremely stramineous and challenging due to my ideas not working or simply just not being able to do so resulting in so many errors. I came up with different designs for the Gui each of them was compatible, whilst others didn’t meet the requirements in my opinion.

This the first design I had for my GUi , each image below shows the functionality of my first Gui.

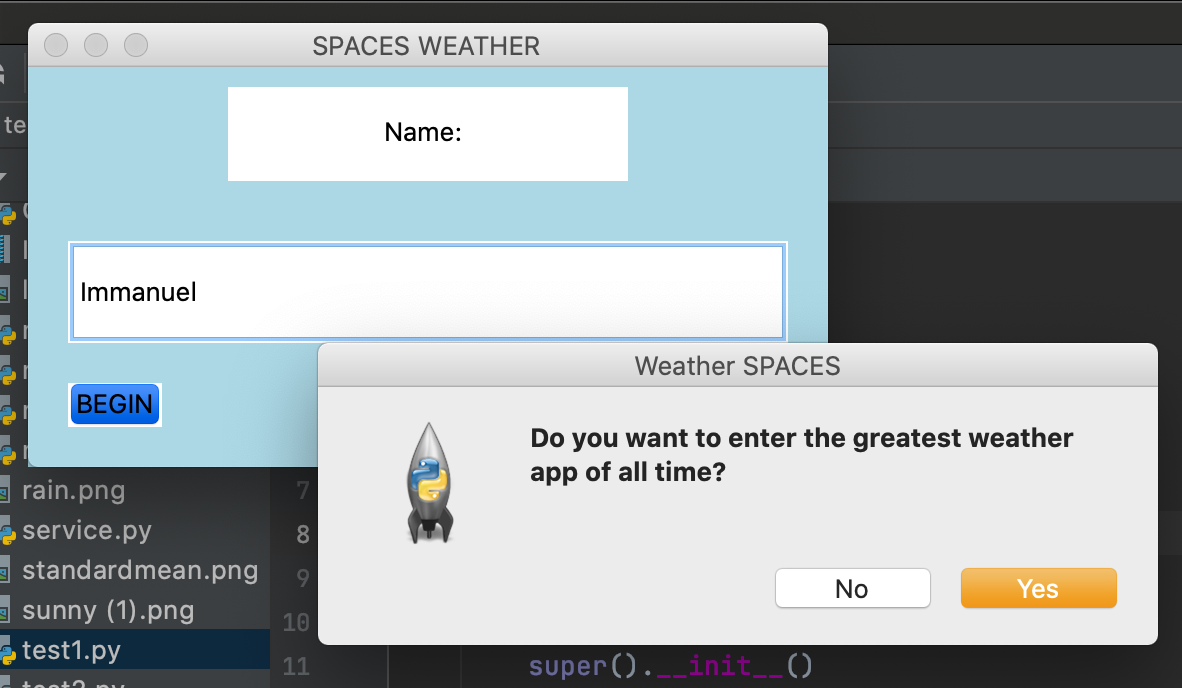
#1



#2 I had gone with message box pop ups which opened up new windows upon pressing the buttons.



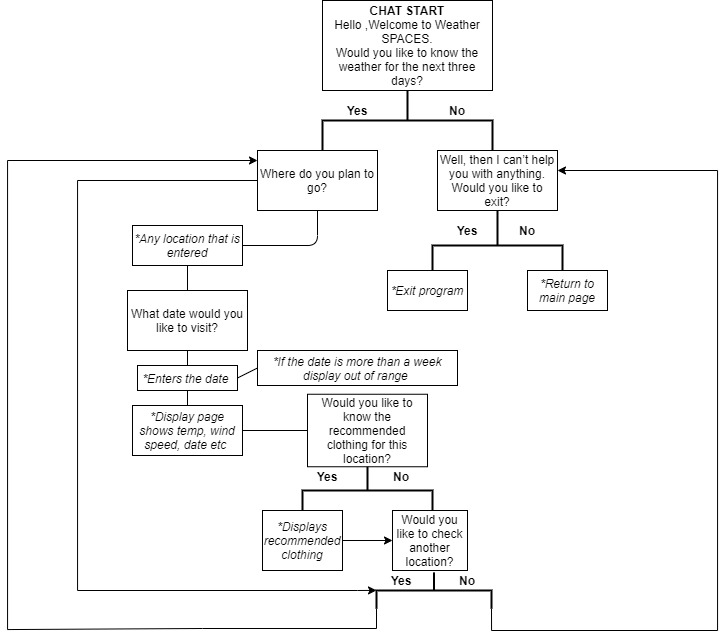
#3 Although I was a big fan of this method, I later realised that it wasn’t meant to be because I tried to link different python files and I just couldn’t integrate them.



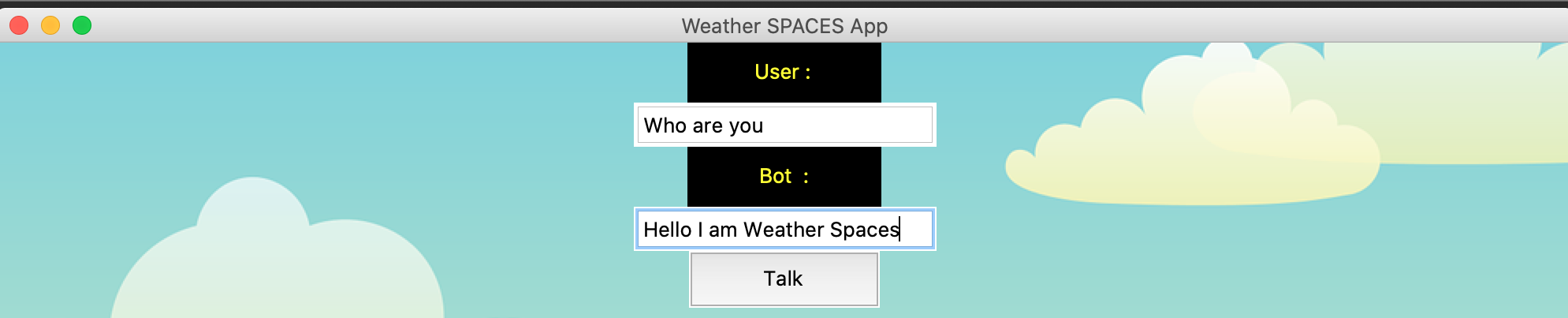
**CREATING CHATBOTLOGIC**

This was conducted by the group where we decided upon the diction and language required for our chatbot. It was made in a manner for so that the user could communicate with bot, we put all our chat bot logic in the decision tree showing our thought process. It later came to my realisation as I was creating the chatbot itself I couldn’t follow the decision tree chat logic exactly as it was written because I was unable to implement some of the functions the chatbot wanted the logic to show. I had to use certain elements of the chatbot logic and curate it so that kit still met both the needs and requirements.

**DECISION TREE**



**WORKING CHATBOT WITH GUI.**

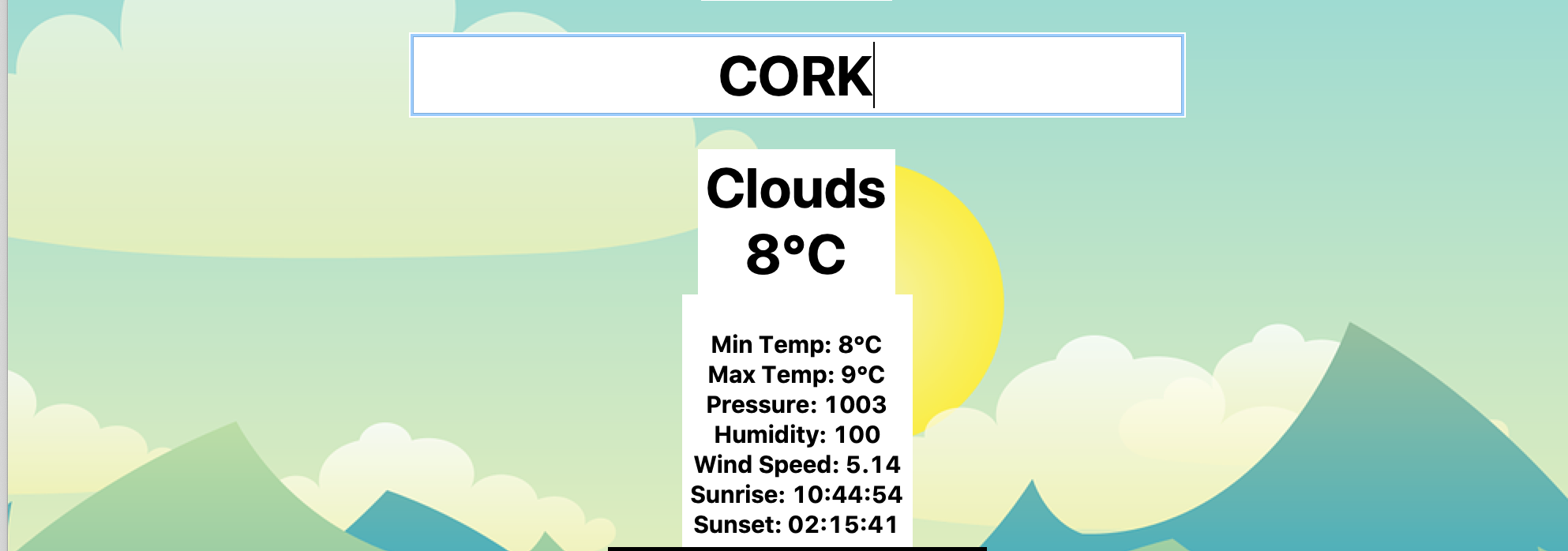


**TEST GUI WITH API**

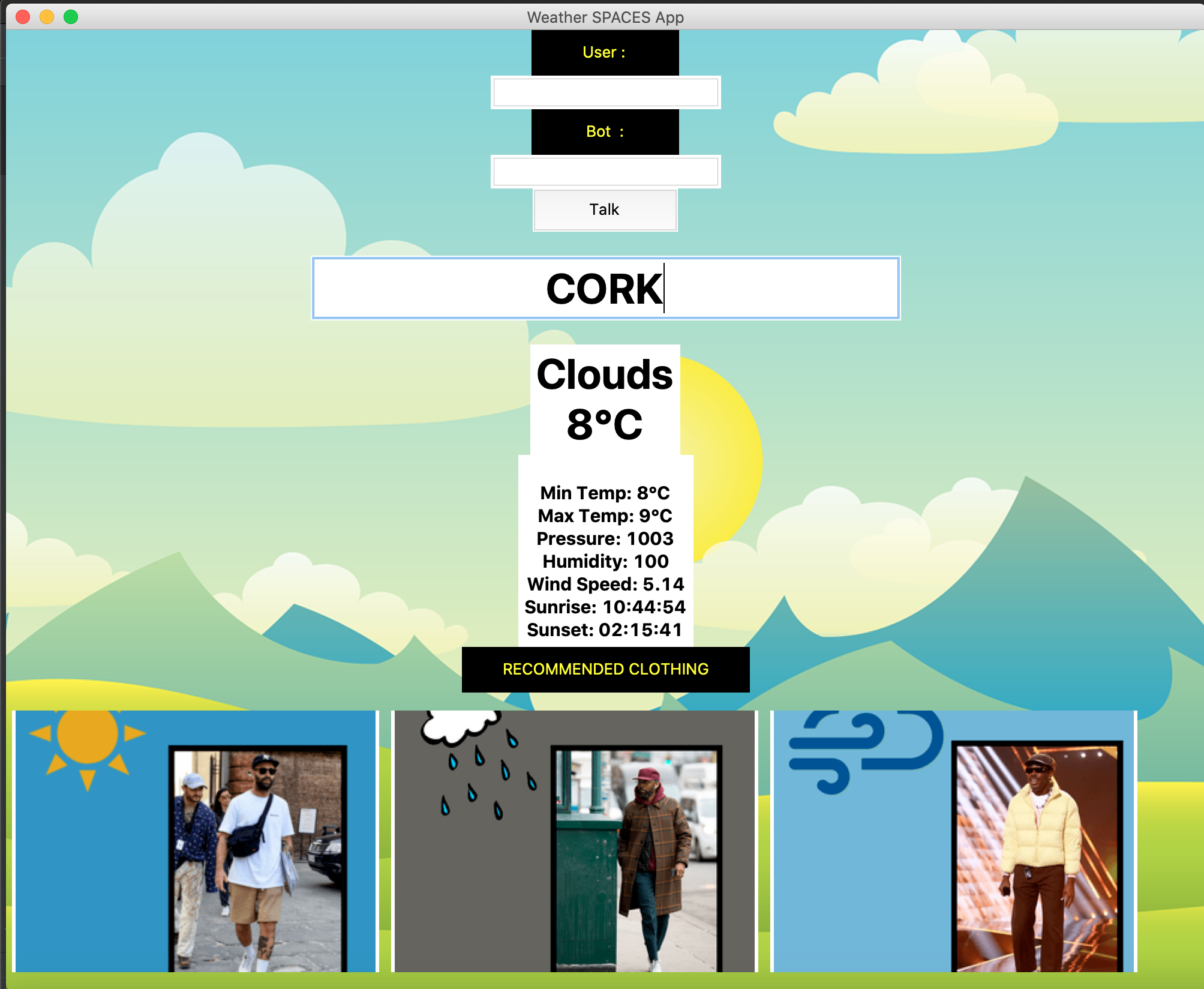
I created a working Gui using Tkinter, and I was satisfied with the outcome of the Gui window. My next step was to link the API to the GUI at first, I struggled to link the two together and it would either just be the APi not printing out the data I wanted or there was an error in the written python code. I eventually managed to sort out everything by doing extra research and also with the help with my Lecturer.

I had to do a run a series of tests which included Testcases. They are a set of actions executed to verify a particular feature or functionality of your software application.

#1 This is an example of the working Weather App, with a Responsive Gui using Tkinter and a working API as seen below:



#2

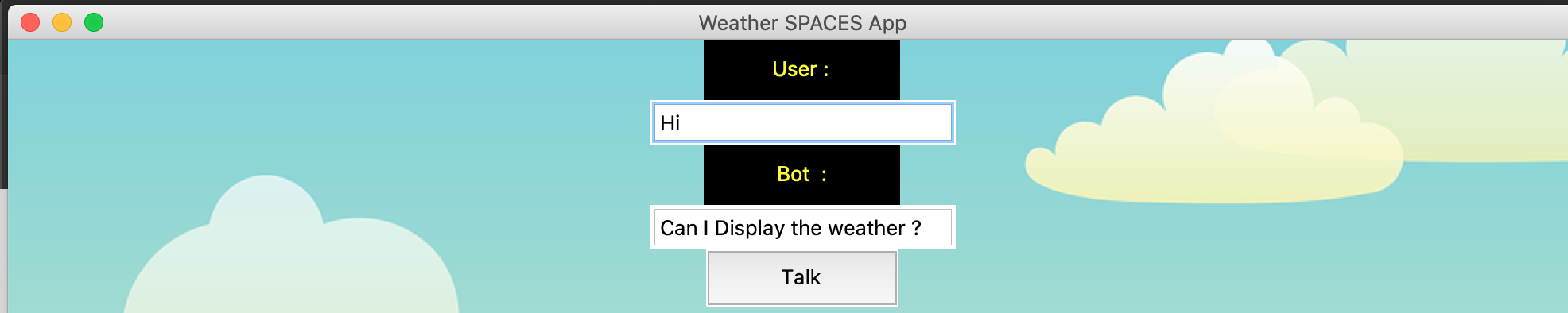


**FINAL PRODUCT TESTING**

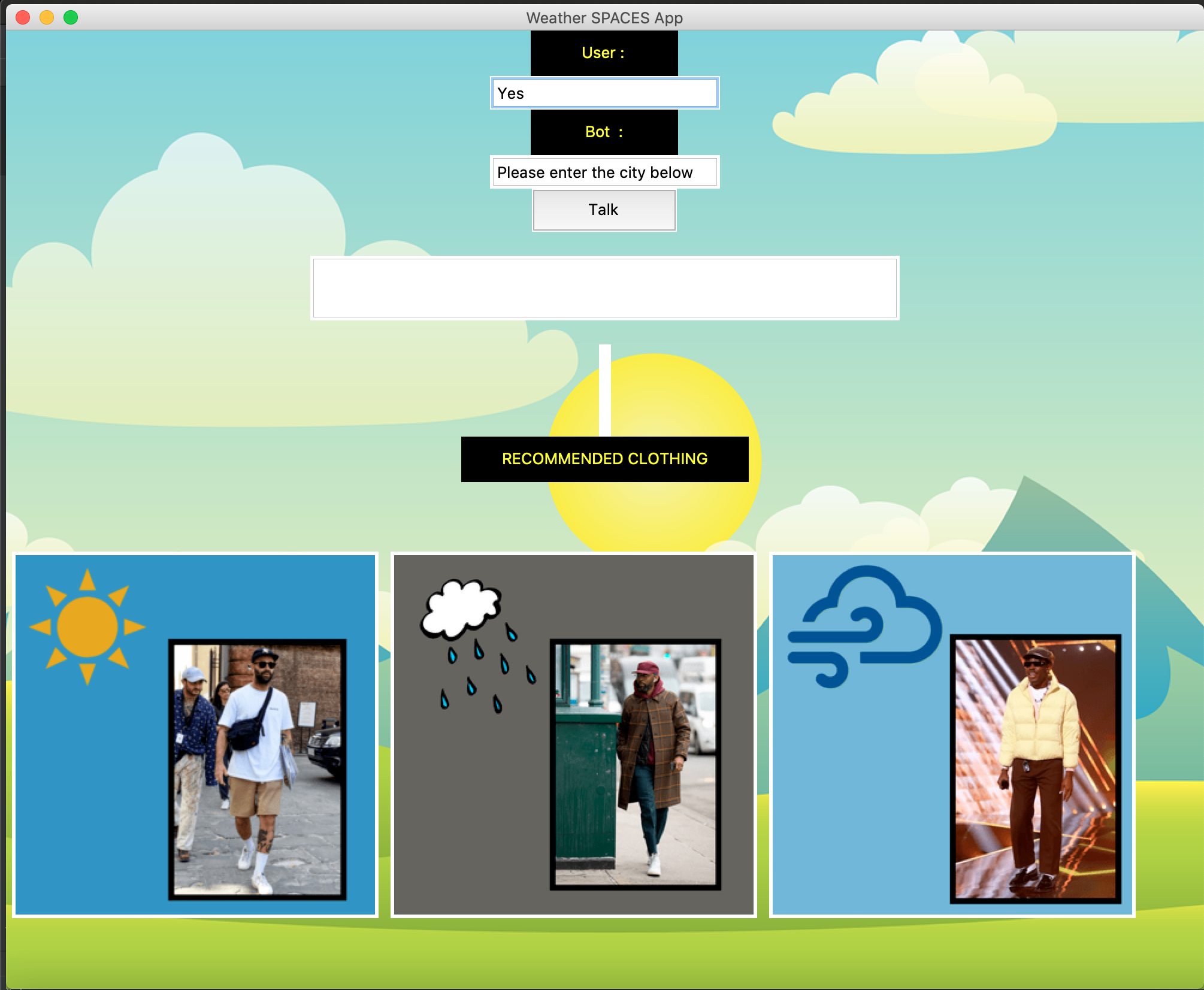
Upon satisfied completion of the “Weather Spaces App” we applied the final touches on the code and the application its self to ensure it was ready for the user. During the final product testing we conducted a set of user tests to check if it was easily accessible by the intend user and if it fulfilled the asked requirements:

* We ran the python code first and checked if it would compile successfully without any error
* Test to see if the chatbot is responsive to user input and also if it would give a suitable output.
* We Checked to see if the Api was linked to the Gui and it printed out the weather for the specific City.
* We also tested it to see if provided the recommended clothing options at the bottom of the GUI.

**#1 Working code**



**´#2**



**#3**

