

Project Documentation

Application name: Weather4U

Group : 4

Team members:

1. R.S.Sneha
2. R.Carli Renita
3. S.K.Padma Parvathi
4. S.Tirumalai Nanthini

Table of Contents :

Description:	2
APP Info :	2
Display Details :	2
Building TOOLS:	2
Ionic Framework :	2
Angular :	3
NodeJS :	4
Visual studio Code :	5
UI Side:	6
App Flow:	6
Welcome Page:	7
Login Page:	8
City- List Page:	9
Weather Details Page:	10
Service Call to external API:	11
Conclusion:	13

Description:

This Weather Forecast app is used to display climate data and inform the users about any natural change in Popular US Cities . This app is built with necessary features which can make the users prepared for any weather-related challenges. This will be useful for any type of user and mostly concentrated on Tourists.

APP Info :

This Mobile Weather Application is designed by using:

1. IONIC 4 framework.
2. Angular CLI
3. Visual Studio Code.
4. NodeJS

Display Details :

This Mobile Weather Application is designed to display on the device “**Galaxy Note II**”

Building TOOLS:

Ionic Framework :

Free and open source, It offers a library of mobile-optimized UI components, gestures, and tools for building fast, highly interactive apps.

In this Internship we have learned how to use the Ionic Platform for building a real time mobile application. It provided the ready-made components , typography and gorgeous base theme that adapts to each platform.

It helped us throughout the internship to build a simple , decorative and high quality UI design for our application. With Ionic's **Adaptive Styling**, our app looks and feels at home on every platform and device.

The most used ionic components in our app are:

1. Ion-item
2. Ion-card
3. Ion-list
4. Ion-button
5. Ion-icon
6. Ion-label
7. Ion-thumbnail
8. Ion-back button

Thus we have created a complete mobile app using this user friendly Mobile app framework.

Angular :

Angular is a platform and framework for building single-page client applications using HTML and TypeScript. Angular is written in TypeScript. It implements core and optional functionality as a set of TypeScript libraries that we import into our apps.

Angular is a platform that makes it easy **to build applications** with the web. **Angular** combines declarative templates, dependency injection, end-to-end tooling, and integrated best practices to solve **development** challenges. **Angular**

empowers developers to **build applications** that live on the web, mobile, or the desktop.

Our Application required the call for an external rest api , so we have created the service called “Restapi” where we have imported the http client from Angular.

Using this Command line:

```
import { HttpClient } from '@angular/common/http';
```

We used **ngOnInit** : A callback method that is invoked immediately after the default change detector has checked the directive's data-bound properties for the first time, and before any of the view or content children have been checked. It is invoked only once when the directive is instantiated.

Here we use this in every .ts file to invoke and display the weather details of a particular city whose api is represented in the restapi service that we have already created

```
ngOnInit() {  
    this.restapi.gettucson().subscribe( (response) =>  
{  
        console.log( response );  
        const data = JSON.stringify(response);  
        const jsdt = JSON.parse(data);  
        const jsdt1 = jsdt['properties'];  
        this.forecast = jsdt1['periods'];  
        console.log(this.forecast);  
    });  
} //This code is used to invoke the restapi service to display the details for  
Tucson city
```

NodeJS :

Node. js (Node) is not a framework nor a language but a runtime open source development platform for executing JavaScript code server-side. It is easy to learn for both front-end and back-end developers.

Visual studio Code :

It is a free source-code editor made by Microsoft for Windows, Linux and macOS.

We can build apps for Android, iOS, and Windows devices by using Visual Studio. As we design our app, used tools in Visual Studio to easily add connected services

Built our app by using HTML and JavaScript. Share code, strings, images, and in some cases even the user interface.

At its heart, Visual Studio Code features a lightning fast source code editor, perfect for day-to-day use. With support for hundreds of languages, VS Code helps you be instantly productive with syntax highlighting, bracket-matching, auto-indentation, box-selection, snippets, and more.

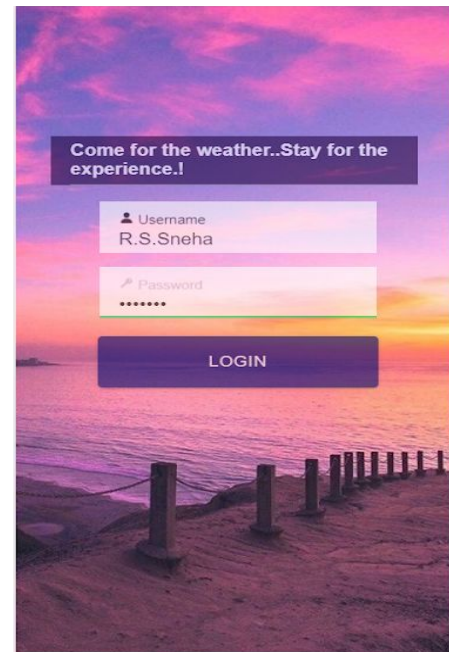
The VSC editor is user-friendly for the app or web developers. So we developed our app with great ease.

UI Side:

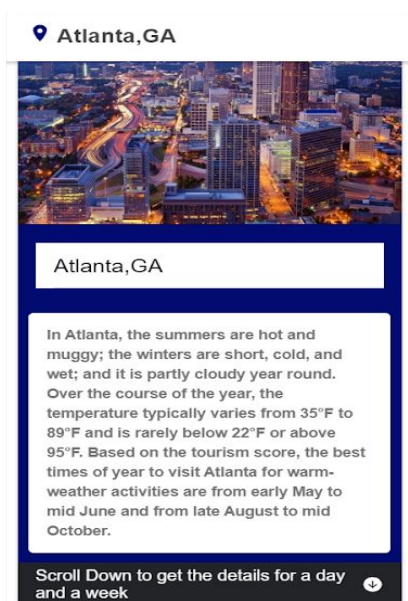
App Flow:



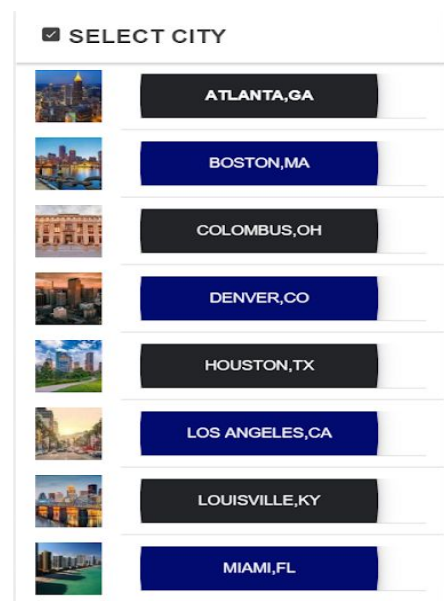
1



2



4



3

Welcome Page:



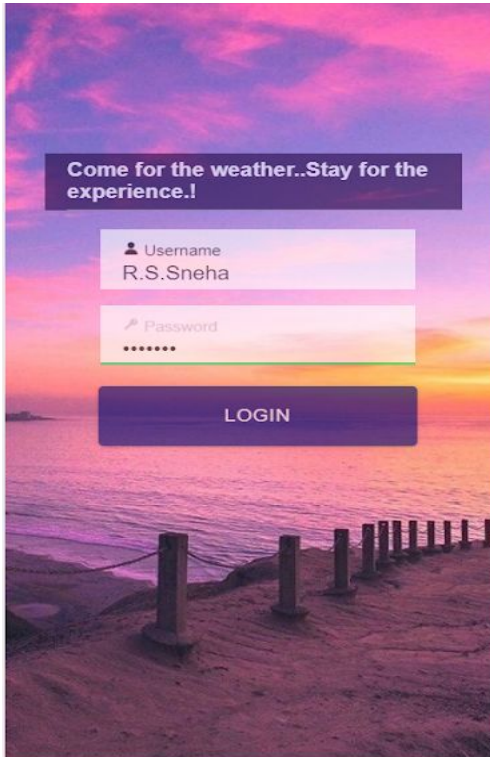
1

“The Exact Welcome Page looks like this”

About Welcome Page:

Background	Set using .scss page of Welcome Page .
Logo	Designed initially using Wix.com then embedded here using .scss page.
Caption	“ Beneath the Weather ” designed using the “ ion-item ” component and .scss page to format it.
Navigation	Both Logo and Caption bar act as the navigation buttons to the next page .

Login Page:



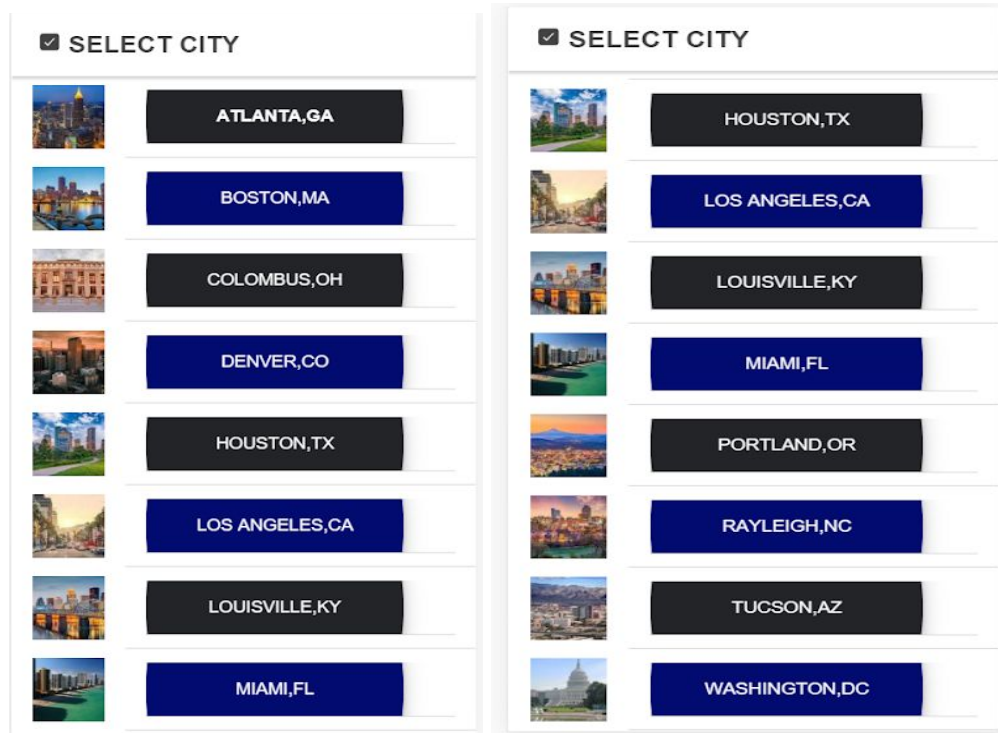
2

“The Exact Login Page looks like this”

About Login Page:

Background	Set using .scss page of Login Page .
Caption	“ Come for the weather Stay for the experience ” designed using the “ ion-item ” component and .scss page to format it.
Username & Password	Designed by using the “ ion-input ” component.
Navigation	The Label “ Login ” is used as the Navigation button

City- List Page:



3.a

3.b

“The Exact City-List Page looks like this”

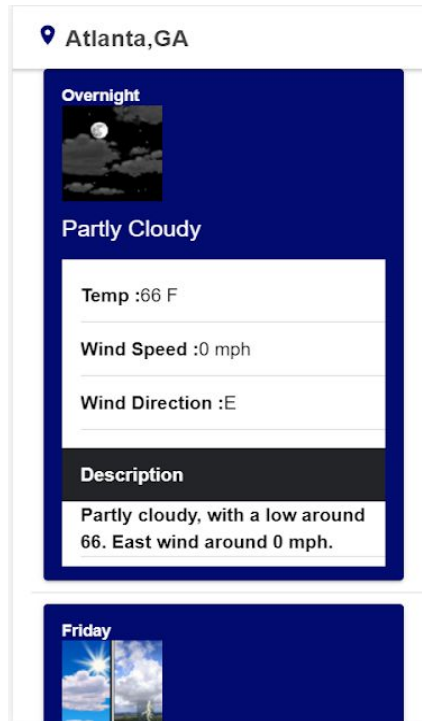
About City-List Page:

Thumbnail (Refer Fig. 3.a and 3.b)	Set for all the cities using Gravatar.com (https://en.gravatar.com/gravatars/new/computer) and .html page to format according to the corresponding item displaying the name of the city.
Cities List (Refer Fig. 3.a and 3.b)	Lists popular 12 US cities by using the ion-item component.
Navigation	Every city name item act as a navigation button designed using “ion-fab-button”

Weather Details Page:



4.a



4.b

“The Exact Weather details Page looks like this” (referred page : Atlanta,GA)

About Weather Details Page:

General Information on weather of particular city	<ul style="list-style-type: none"> ● Picture of the city. ● General Weather Details for the Tourist (refer 4.a) ● Used “ion-card” and “ion-list” components for designing this (4.a).
Current Weather Details for Day , Night and a week	Used “ ion-card ” and “ ion-list ” components for designing this (4.b). This displays the weather captions like:Sunny,Partly Cloudy,Rainy . etc..

Service Call to external API:

- Here we have used the API Web service called “National Weather service”
<https://www.weather.gov/documentation/services-web-api>
- We generated service called “restapi” to call the web api in http client.
- To the web api we have provided the required Latitude and Longitude which then displays as the figure 5.a

(refer this link for more:

<https://api.weather.gov/points/39.7456,-97.0892>)

- Then navigating into the highlighted link where the grid points are mentioned which refer to the location of a NWS station. (refer 5.a)
- There we got the required details like
 1. Temperature (in F)
 2. Wind Speed
 3. Wind Direction
 4. Short and Detailed Description
 5. Icon.

(Refer the figure 5.b)

(Scroll down to reach the pictures mentioned)

```
← → ↻ 🔒 api.weather.gov/points/39.7456,-97.0892

    },
    "county": {
      "@type": "@id"
    }
  },
  "id": "https://api.weather.gov/points/39.7456,-97.0892",
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [
      -97.08920000000005,
      39.745600000000003
    ]
  },
  "properties": {
    "@id": "https://api.weather.gov/points/39.7456,-97.0892",
    "@type": "wx:Point",
    "cwa": "TOP",
    "forecastOffice": "https://api.weather.gov/offices/TOP",
    "gridId": "TOP",
    "gridX": 31,
    "gridY": 80,
    "forecast": "https://api.weather.gov/gridpoints/TOP/31,80/forecast",
    "forecastHourly": "https://api.weather.gov/gridpoints/TOP/31,80/forecast/hourly",
    "forecastGridData": "https://api.weather.gov/gridpoints/TOP/31,80",
    "observationStations": "https://api.weather.gov/gridpoints/TOP/31,80/stations",
    "relativeLocation": {
      "type": "Feature",
      "geometry": {
        "type": "Point",
        "coordinates": [
          -97.086661000000007,
          39.679375999999998
        ]
      },
      "properties": {
        "city": "Linn",
        "state": "KS",
        "distance": {
          "value": "7366.9851976444",
          "unitCode": "unit:m"
        },
        "bearing": {
          "value": "358",
          "unitCode": "unit:degrees_true"
        }
      }
    }
  },
  "forecastZone": "https://api.weather.gov/zones/forecast/KSZ009",
```

5.a

```
api.weather.gov/gridpoints/TOP/31,80/forecast
{
  "elevation": {
    "value": "441.96",
    "unitCode": "unit:m"
  },
  "periods": [
    {
      "number": 1,
      "name": "Today",
      "startTime": "2020-06-19T09:00:00-05:00",
      "endTime": "2020-06-19T18:00:00-05:00",
      "isDaytime": true,
      "temperature": "74",
      "temperatureUnit": "F",
      "temperatureTrend": null,
      "windSpeed": "5 to 10 mph",
      "windDirection": "NE",
      "icon": "https://api.weather.gov/icons/land/day/tsra,50?size=medium",
      "shortForecast": "Chance Showers And Thunderstorms",
      "detailedForecast": "A chance of showers and thunderstorms. Cloudy, with a high near 74. Northeast wind 5 to 10 mph. Chance of precipitation is 50%."
    },
    {
      "number": 2,
      "name": "Tonight",
      "startTime": "2020-06-19T18:00:00-05:00",
      "endTime": "2020-06-20T06:00:00-05:00",
      "isDaytime": false,
      "temperature": "65",
      "temperatureUnit": "F",
      "temperatureTrend": null,
      "windSpeed": "5 mph",
      "windDirection": "E",
      "icon": "https://api.weather.gov/icons/land/night/tsra,50/tsra,80?size=medium",
      "shortForecast": "Showers And Thunderstorms",
      "detailedForecast": "A chance of showers and thunderstorms. Cloudy, with a low near 65. East wind 5 mph. Chance of precipitation is 50%."
    }
  ]
}
```

5.b

Conclusion:

Thus we have created the real life Weather application named “Weather4U” by using all the App developing technologies which were introduced by this Internship.

We learned pretty new Technologies and functions that we never heard before. We understood the cohesive steps in creating a real time application. We have got impressed by our own work and triggered to create many like this.