Jeremy Bryan’s ESRI Weather App

Contents

1. Introduction 1

1.1. Purpose 2

1.2. Intended Audience 2

1.3. Intended Use 2

1.4. Scope 2

1.5. Definitions and Acronyms 2

2. Overall Description 3

2.1. Features 3

2.1.1. Real-time current position 3

2.1.2. Current date 4

2.1.3. Current temperature 4

2.1.4. Current conditions 4

2.1.5. Background 4

2.1.6. 7-day forecast 5

2.1.7. Status indicator 5

2.1.8. Units indicator 5

2.2. Assumptions and Dependencies 5

3. System Features and Requirements 5

3.1.1. Functional Requirements 5

3.1.2. External Interface Requirements 5

3.1.3. System Features .... **Error! Bookmark not defined.**

3.1.4. Nonfunctional Requirements .... **Error! Bookmark not defined.**

1. Introduction

This app has the following general capabilities:

- Display the current weather and 7 day forecast for the current location
- In the event that online data cannot be acquired, use stored data
- Display the device’s current location on a map

1.1. Purpose

This app was developed as an exercise to demonstrate the author’s ability to create functional apps using ESRI AppStudio. Its purpose is to show various features within a weather application that could translate to the day to day activities of an app developer at ESRI. This application was written using AppStudio, Qt, QML, and JavaScript.

1.2. Intended Audience

The audience for this app is the employees and managers at ESRI. It was created so that they may evaluate the author’s talent and work output given a set of requirements combined with a deadline.

1.3. Intended Use

The app is intended for decision makers at ESRI for the purpose of evaluating qualities related to potential job performance.

1.4. Scope

This app was defined by a set of features and requirements designed to highlight the developer’s abilities. It has multiple qualities that accomplish this, including online API usage and display of device’s current location on a Map View.

The only risk with respect to using this application is potentially being unprepared when you go outside if it does not display weather data accurately.

1.5. Definitions and Acronyms

API – Application Programmer Interface – a computing interface which defines interactions between multiple software intermediaries.

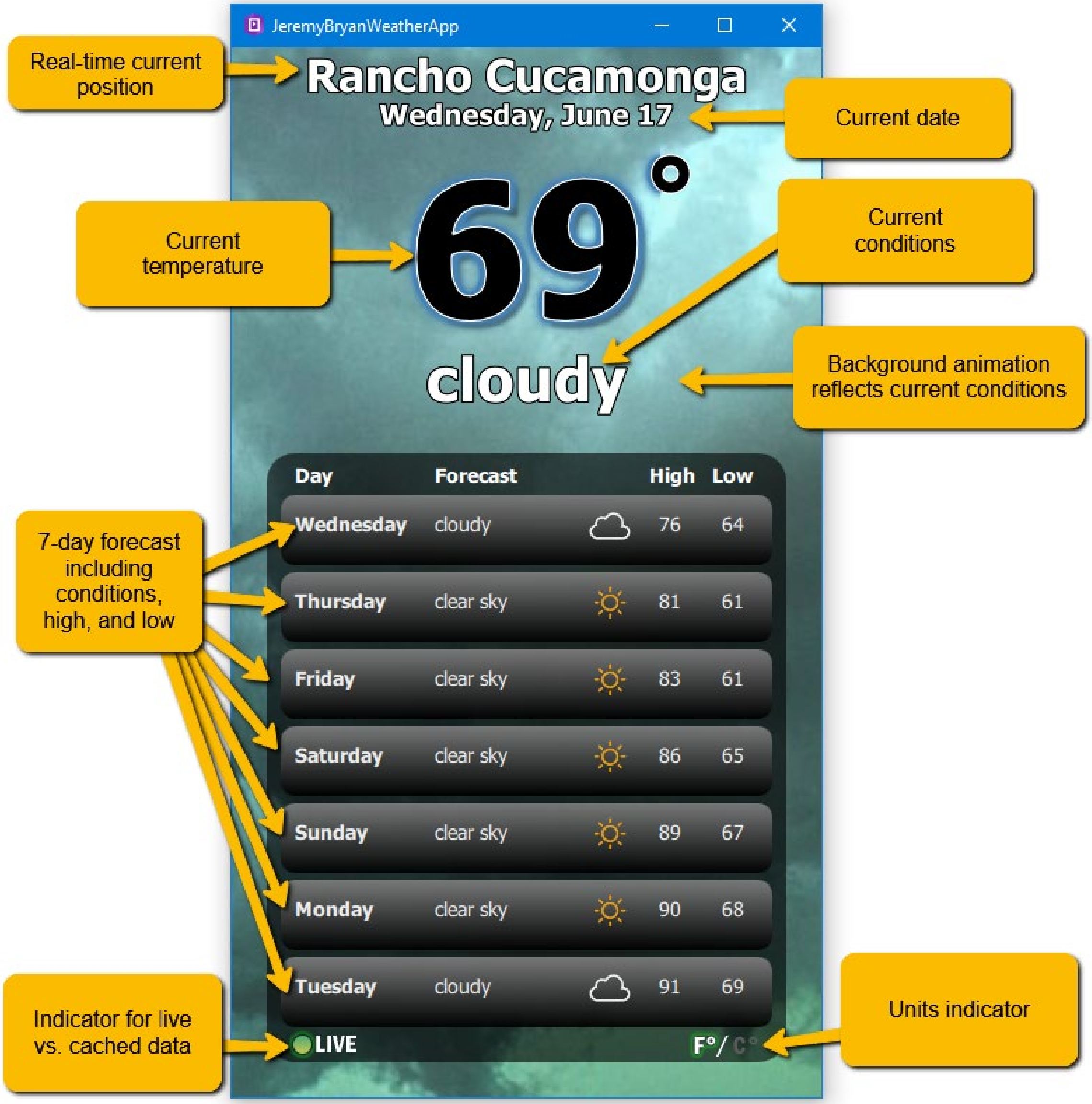
ESRI – Environmental Systems Research Institute – an international supplier of geographic Information system software, web GIS and geodatabase management applications.

GIS – geographic information system – a conceptualized framework that provides the ability to capture and analyze spatial and geographic data.

JSON – JavaScript Object Notation – a open standard data interchange format that uses plaintext to store and transmit data objects.

2. Overall Description

This app provides the current weather conditions as well as a 7-day look ahead forecast.



Day	Forecast	High	Low
Wednesday	cloudy	76	64
Thursday	clear sky	81	61
Friday	clear sky	83	61
Saturday	clear sky	86	65
Sunday	clear sky	89	67
Monday	clear sky	90	68
Tuesday	cloudy	91	69

Real-time current position

Current date

Current temperature

Current conditions

Background animation reflects current conditions

7-day forecast including conditions, high, and low

Indicator for live vs. cached data

Units indicator

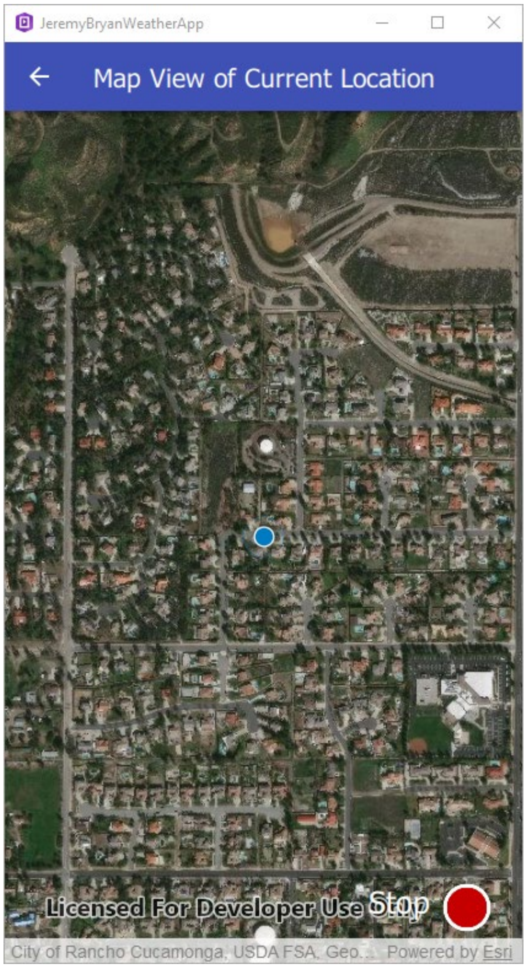
2.1. Features

2.1.1. Real-time current position

After determining the latitude and longitude coordinates from the host device's interface, the app uses the MapQuest Geocoding API to determine and display the current location. The API returns data in JSON format, which must be parsed before being presented. For more information on the API, refer to:

https://developer.mapquest.com/documentation/geocoding-api/

Clicking on the current location text will take you to another screen, showing your current position on a MapView:


JeremyBryanWeatherApp
Map View of Current Location
Licensed For Developer Use
Stop
City of Rancho Cucamonga, USDA FSA, Geo...
Powered by Esri

Clicking on the arrow in the title banner will take you back to the main page.

2.1.2. Current date

The current day of the week, date, and month are displayed. These are bound to the text object using text: *Qt*.formatDateTime(new *Date*(), "dddd, MMMM dd")Current temperature

2.1.3. Current conditions

After determining the latitude and longitude coordinates from the host device's interface, weather data is requested using the OpenWeatherMap API. Again, this API returns data in the JSON format which must be parsed before presentation. For more information on the weather API used, please refer to:

https://openweathermap.org/api/one-call-api

2.1.4. Background

Once the app has determined the current weather conditions, four different backgrounds are animated to help acclimatize the user to the conditions they may expect when venturing outside.

2.1.5. 7-day forecast

When pulling the weather data from the API, the current conditions and a future forecast is included in the data. The number of forecast days is specified in the API request, with a maximum of 7 days. The look-ahead forecast includes:

- The day of the week
- The conditions expected for that day
- An icon representing the expected conditions
- The overall high for that day
- The overall low for that day

2.1.6. Status indicator

Near the bottom of the app screen, there is a status indicator that signifies whether the data displayed is from a live (online) source, or cached data. If the dot it green and the “LIVE” text is white, that means the API call was successful and we are looking at current data. If the dot and text are greyed out, then we see data from the past that has been cached for offline use.

2.1.7. Units indicator

Also, at the bottom of the screen, on the right-hand side of the forecast there is a units indicator. The highlighted unit signifies which unit, imperial or metric, is being displayed on the app. Click on these units will cause the app to send another API request with the toggled unit value.

2.2. Assumptions and Dependencies

The application assumes that the device it is running on has the capability to access network resources. Contingencies are in place if that assumption is not met. Additionally, the assumption that the device’s current location can be obtained.

3. System Features and Requirements

3.1.1. Functional Requirements

This app requires an install of ESRI AppStudio, and a device capable of determining position.

3.1.2. External Interface Requirements

To obtain full functionality, the app required a working network interface through which it can send API requests.