Magic Mirror

Project Team Members

Behram Dossabhoy
Jonah Landry
Jason Myles

Project Summary

The Magic Mirror will do three things: display the current date and time, fetch data from Yahoo's Weather API to tell a user what the current weather is and the forecast for the next three days, and display to the user the next three upcoming events, each with respective date and time.

Goals and Objectives

Utilize the Raspberry Pi kit to act as a "mirror" that displays necessary information such as date, time, weather, and the user's daily schedule every morning. It should be able to be fed a calendar and fetch weather data for a user's specified location.

GitHub Repository

This project's GitHub repository is located at: https://github.com/bnd011/MagicMirror

Bill of Materials

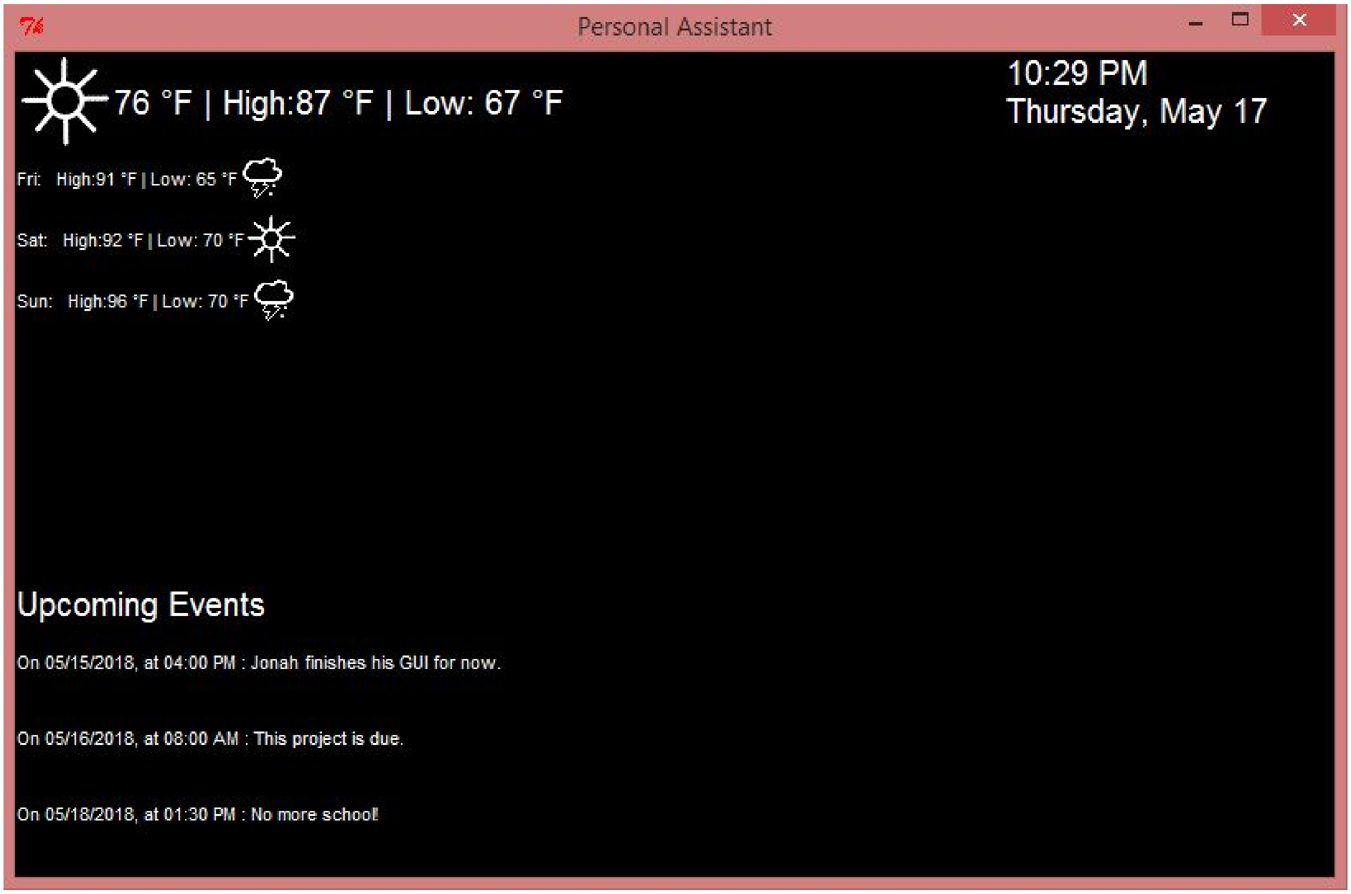
In the end, the only hardware that the project requires is the Raspberry Pi kit provided by Louisiana Tech. All of the code in the project was written using Python 2.7, with data retrieved from Yahoo's Weather API and Google's Calendar API.

GPIO

For now, the system will be self-contained in the RPi, but in the future, we will utilize a bigger, reflective screen for the user such as an LCD touchscreen monitor via HDMI. While you can use a one-way mirror, old monitors are reflective enough to serve the purpose of a mirror.

GUI

The GUI will have no interaction with the user in its current state, as the mirror is able to fully function without user input. The GUI will be minimal, designed with a fair amount of deadspace in order to more accurately represent the RPi touchscreen like a mirror. This is accomplished by GUI elements being located at the mirror's corners.



76 °F | High:87 °F | Low: 67 °F

10:29 PM
Thursday, May 17

Fri: High:91 °F | Low: 65 °F

Sat: High:92 °F | Low: 70 °F

Sun: High:96 °F | Low: 70 °F

Upcoming Events

On 05/15/2018, at 04:00 PM : Jonah finishes his GUI for now.

On 05/16/2018, at 08:00 AM : This project is due.

On 05/18/2018, at 01:30 PM : No more school!

The "past sample events" were forced into this mockup image of the GUI in order to more completely show what it would normally maximally display.

Gantt Chart

Magic Mirror	Week 1					Week 2					Week 3					Week 4					Week 5					Week 6				
	Apr 9-13					Apr 16-20					Apr 23-27					Apr 30-May 4					May 7-11					May 14-18				
Gather website and phone data																									🟦	🟦	🟦			
-Calendar data																											🟦			
-Weather data																					🟦	🟦	🟦	🟦	🟦	🟦				
Parsing the data																					🟦	🟦	🟦	🟦	🟦	🟦	🟦			
Build the GUI															🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦						
Practice presentations																											🟦			
In-class Project Days															🟩					🟩					🟩					
Final project presentation																												🟥		

Originally, there were more things to be included in the project spread out over the time given. As time went on, some items proved to be too difficult or tedious to implement given the provided time, and were subsequently removed. In addition, there was inevitable procrastination along with the fact that finals were coming up, and group members' schedules became harder to line up.

Future Development Plans

If we were to continue working on the project, we would implement a more interactive interface on a larger screen. If this were to happen, we could include options such as a small calendar in which a user could tap on and add or remove events. We could also include buttons such as "View More" or "View

Extended Forecast" next to the weather elements to provide more in-depth weather information such as the current atmosphere status, current wind information, or even an extended forecast with up to ten days! In addition, we could find a way to display an email/message system using a respective API.

Lessons Learned

The main lesson we learned is that there's less emphasis on understanding what code does as long as you follow good programming practices. Since we each took a part of the project to code by ourselves, each of us had to do some research to understand how to structure and get the code working. This related to the curriculum the most by utilizing iterative software development to get each addition to work with the GUI. As for problem solving in general, we learned that we need to keep up with the entire groups' schedule and try to plan things much further in advance in order to talk about it more. Regarding future courses in the curriculum, the project helped us learn how to work through new situations, and for specifically computer science, researching code via internet resources.