Advanced Software Engineering - CP3407

Agile Software Development

Jake Regattieri

Bradley Pemmelaar

|  |  |
| --- | --- |
| Jake Regattieri | * User Stories * Gantt Chart * Architectural Design |
| Bradley Pemmelaar | * Project planning & scope * Project Description and research * User Stories * Database Design * Interface Design |
| GitHub: <https://github.com/pemmilicious/CP3407Assignment> | |

Alpha-release-iteration-1 report:

Quite a number of things changed between the first iteration and the actual build of the alpha release 1. Mostly due to limitations within the chosen framework, being node.js using express. As well as being the first time using node.js and JavaScript. Firstly, we found it difficult to follow our original user stories, I found that certain user stories would fall first when building the project. Secondly, the dashboard did not come too much later in the build as we found it hard to make an interface when we didn’t know how the data was going to be retrieved and stored. Our team ended up having to build the backend of the web app prior to progressing anywhere with the front end.

User story 1: View Dashboard (User should be able to view the latest weather information).

User story 2: Database (User should be able to access the data from the database)

User story 3: Refresh (User is able to refresh the data to the latest information).

User story 4: Select reading (User can select a reading type and view previous readings stored in database).

User story 5: Date and Time stamps (User should be able to see the date and time that the data was retrieved).

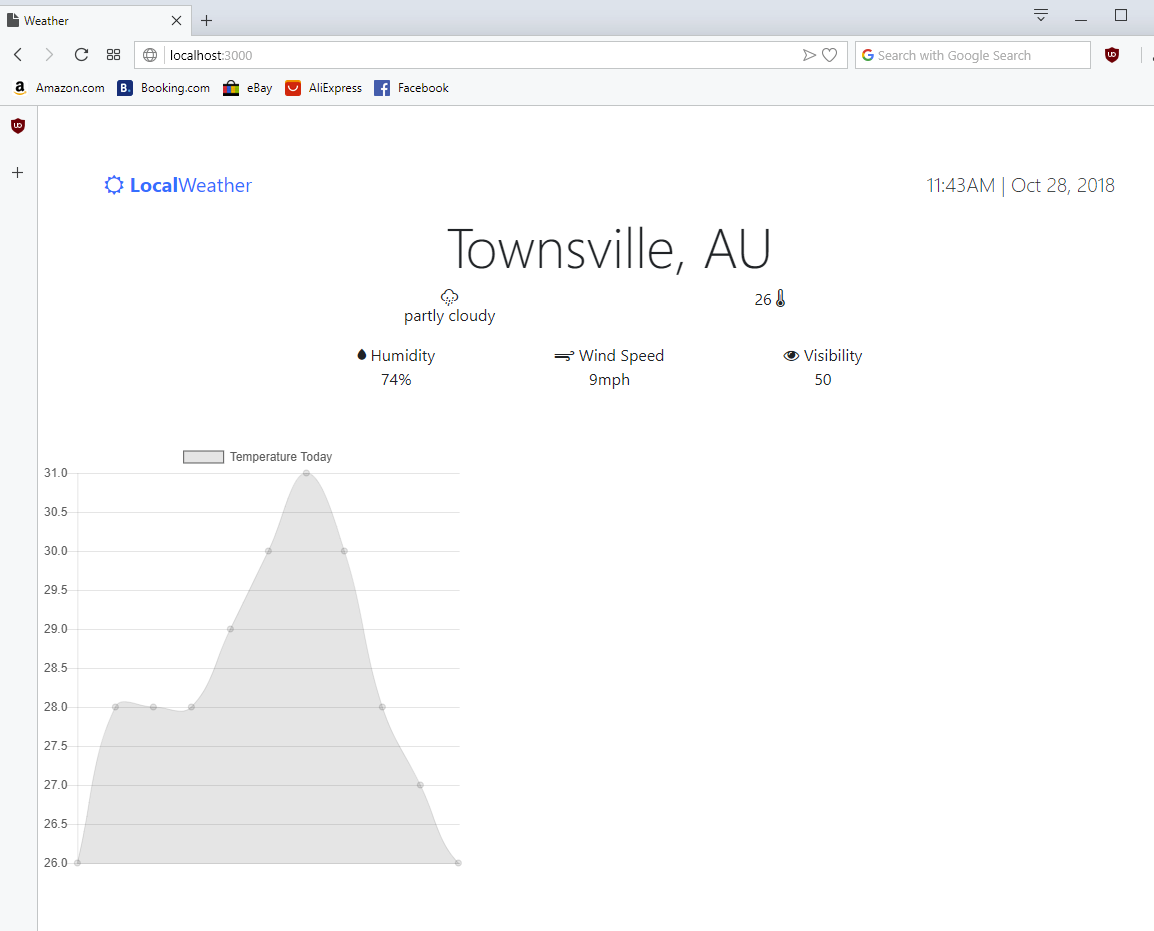
User story 6: Graphing the data (Users should be able to display the information in a graph so that they can look for trends).

User story 7: Settings (Users should be able to change the settings on what they are seeing).

User story 8: View graphs( user should be able to view a selected graph)

User story 9: Alarms (User should be able to view alarm information regarding the sensors).

Interface Design:



This is what we ended up having in terms of an interface, this ran in any web browser. The original goal was to have the top section display the data, and upon clicking a specific datatype for example humidity, it would display the graph of humidity for the last few hours. This information was retrieved through openweathermap.org which is an open API for use in the IT industry. With weather information on 90% of cities and towns around the world.

We used a few different stylesheets referenced from the internet which created the symbols for the weather description, temp humidity etc. as well as chart.js, which is a library for building interactive charts with node.js. The local Weather object at the top left corner ended up being a refresh button, so that the server file could retrieve new information from openweathermap and store it in the MySQL database.

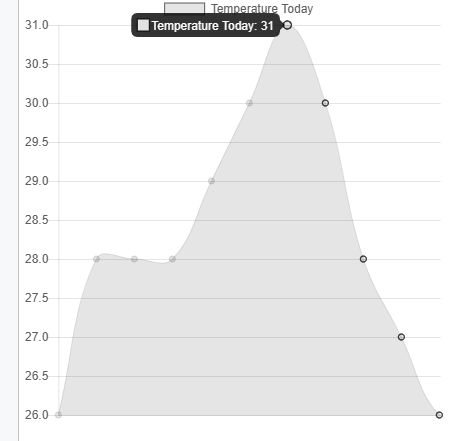
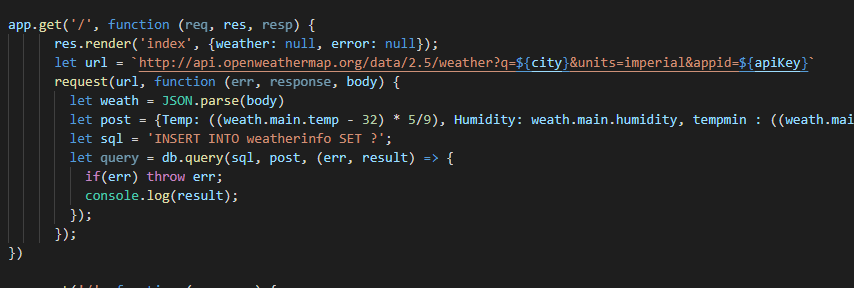


Chart.js allows for a large amount of different options, one of which is a tooltip on the data points when you hover over. The original goal was to have this tooltip display the time of day that this entry was retrieved. The database did store this, however I found it difficult to retrieve from the server.js file and post the information to the index.ejs file.

I was able to make these charts using the information which was updated, passed and storage in the MySQL database using the following function:



This function ran as soon as the user opened the page, it would store data coming from the openweathermap API directly into the MySQL database by using the request extension as a part of the NodeJS framework. Every time the user re-loaded the page, new data would flow into the database. As you can see, we stored the information from openweathermap into a JSON format variable, then selected the necessary data from it and parsed into into the database using the SQL query INSERT INTO weatherinfo SET ?.

In order for us to be able to access the MySQL database, we had to first establish a connection:



*External software libraries:*

To complete this project, we used several libraries built around Node.js which is an open-source, cross-platform JavaScript run-time environment that executes JavaScript code outside of a browser. With help from the extension of express which is a free platform used for developing web applications and building API’s for use in front and back end development.

Node.js is a modern way to develop websites and web apps as its built to stand the test of time, as the packages are constantly evolving thanks to the open source format, many people from around the world are using node.js and are getting better at using its layouts and features. Furthermore, there is a tremendous amount of support for using databases with node.js, as they are easily installed through the means of npm. Node.js can access databases with SQL in real time and render the information in an instant.

The database tool we used for this project was MySQL, I choose to use this as I had prior knowledge with using its backend, being phpMyAdmin. Although MySQL is not the most favored database to use with NodeJS it still worked out for the most part, as I was able to store the necessary information from the weather website into the database using JavaScript.

The extension for our website to be able to create charts live with new data was the ChartJS extension which could easily be installed straight onto the server files and into the node files.

ChartJS’s framework and documentation proved to be a highly valuable asset to the assignment as it was very easy to understand and powerful with rending charts quickly.

Project Development:

development environment:

Our team have mostly working from home when designing the interface and software for a brand-new weather monitoring program but in order to get the most out of the Scrum project management method our team will be conducting weekly stand ups using skype.

source code repositories:

The code repositories our team will be using will be Github. Github link: https://github.com/pemmilicious/CP3407Assignment.git

project collaboration tools:

Our Team will be using project collaboration tools such as trello to help with the implementation of Scrum project management. Trello will be used in the creation of our team's scrum board and organization of the team's backlog. Another project collaboration tool our team used was skype for times where our team members were unavailable to meet in person, this allowed our team to not fall behind in development due to always being able to reliably contact one another throughout the projects development time period.

programming languages Consist of:

* Html
* CSS
* JavaScript
* SQL

Development tools:

* Mysql will be used by our development team to store store the data in a database so that it can be retrieved to make comparisons and graphs.
* Node.js was used by our team as an open-source, cross-platform JavaScript run-time environment that executes JavaScript code outside of a browser. With help from the extension of express.js
* Chart.js is a package that is downloadable with the console into the node.js files and comes with an array of functionalities for creating and deploying responsive and interactive charts.

<https://www.techpowerup.com/162522/stealth-com-introduces-a-new-21-5-marine-all-weather-sunlight-readable-hd-monitor>

<https://www.acurite.com/learn/weather-stations/what-is-a-weather-station>

<https://www.popsci.com.au/?src=redirect>