Software Implementation and Testing Document

For

Group 14

Version 1.0

Authors:
Davion M
Douglass S
Ryan R
Conner F
James H

1. Programming Languages

List the programming languages use in your project, where you use them (what components of your project) and your reason for choosing them (whatever that may be).

a. Javascript
b. HTML5
c. CSS3

2. Platforms, APIs, Databases, and other technologies used

List all the platforms, APIs, Databases, and any other technologies you use in your project and where you use them (in what components of your project).

a. React
b. SQL/MongoDB
c. AccuWeather API
d. NOAA API
e. Bootstrap
f. Leaflet.js
g. Express.js
h. Node.js MySql

3. Execution-based Functional Testing

*Describe how/if you performed functional testing for your project (i.e., tested for the **functional requirements** listed in your RD).*

Functional testing for GustBuddy was conducted by setting up each feature we outlined in our functional requirements. Each feature was preliminarily developed, and actual functionality gradually implemented. We began with developing the main page of the web app, which currently displays hurricane information including a monetary damage graph, as well as links to information of past hurricanes. Each team member took responsibility for multiple features to develop them in the same functionality and uniform style of the others. The team began with developing the main page and then divided up the other tabs (pages) to be developed.

1. Day/Week/2 Week weather forecast (HIGH)
2. On-this day past weather report (HIGH)
3. Hurricane Probability (HIGH)
4. Chat-based Weather Assistant (HIGH)
5. User Reporting (HIGH)
6. Astrology Report (MEDIUM)
7. Florida Focus (MEDIUM)
8. Social media sharing implementation (LOW)
9. User log in (LOW)
10. Planet Forecasts (LOW)

4. Execution-based Non-Functional Testing

*Describe how/if you performed non-functional testing for your project (i.e., tested for the **non-functional requirements** listed in your RD).*

We did not perform any rigorous non-functional testing of our project and its components as our non-functional elements to be tested are not yet implemented in the project. We focused primarily on developing the first functional version.

5. Non-Execution-based Testing

Describe how/if you performed non-execution-based testing (such as code reviews/inspections/walkthroughs).

Initially Davion created a functioning Dashboard and then everyone else was brought up to speed on how to develop with React, JavaScript, and all the packages needed. From there, the code was reviewed by each team member and inspected before the changes were merged.

Conner worked on researching and implementing initial functionality of the react-router-dom package for the Forecast, Weather Tracker, Reports, and Astrology Features (to be implemented with all features as they are developed and finished).

Ryan and James worked on researching and reviewing how API's could be implemented into the existing code as well as which databases could be utilized most functionally in conjunction with the Javascript code.

DJ helped in organization of the team and aided in the development of the web application design and template developed in Excel.