

CSCE 4925: Project Aero

Test Plan

By: Alyssa Thurston, Breuna Riggins, James Sabetti, Travis Goral

Project Manager: Alyssa Thurston

Instructor: Professor Keathly

Client: Dan Minshew and Kyle Tayle, Denton Techmill

Date: 12 February 2018

Status: Final Draft

Table of Contents

Table of Contents	2
Revisions	3
1 Introduction	4
1.1 Purpose	4
1.2 Scope	4
1.3 List of Definitions and Abbreviations	5
1.4 Applicable Documents	5
1.5 Overview	5
2 Test Plan	6
2.1 Test Items	6
2.2 Features to be Tested	6
2.3 Features not to be Tested	7
2.4 Testing Approach & Process	8
2.5 Item Pass/Fail Criteria	10
2.6 Test Deliverables	10
2.7 Test Tasks	11
3 Execution	12
3.1 Environmental Needs	12
3.2 Responsibilities	12
3.3 Schedule	13
3.4 Risks and Contingencies	13
3.5 Approvals	14

Revisions

Below is a list of any revisions made to this document.

Date	Description of Change Made	Person Making Change
2/12/2018	Document Created	Breuna, Alyssa
2/13/2018	Document Revision	James
2/15/2018	Final Draft Edits	Breuna, Alyssa, Travis
2/16/2018	Final Draft Revision	James

1 Introduction

The air quality in the City of Denton is notoriously bad. It's not a surprising fact given that there are two universities and multiple major highways and roadways that run through the city's limits. Worse still, there is only a single air quality sensor within city limits, and the next closest one is almost 15 miles away from the city.

The goal of this project is to create a network of air quality sensors to measure the air quality in the city of Denton so that citizens and city personnel alike can monitor air quality and take appropriate actions. The data obtained from the sensors is to be made public, so that air quality trends can be tracked all around the city. With these requirements met, the network would provide secure, real-time, and reliable data on the air quality in the city of Denton.

With the aim of creating a robust and open source of air quality data for the city of Denton, the Open Denton group has set out to establish a network of air quality sensors to gather and compile that data. In order to help achieve this goal, our group has been tasked with the following:

- Create a web client to display data in a highly readable manner.
- Create an API that records and displays air quality data in a readable manner
- Implement a scalable database to hold and model recorded data
- Integrate sensors and controllers that will read and relay data to the database over the internet
- Create documentation that allows community members to add sensors to the network

This document will outline the test plan to test this system when it is constructed, and serve as a layout for performing the necessary testing before the system is given to the client.

1.1 Purpose

The purpose of this document is to summarize the action plan and to formulate a plan to test the completed system. This plan will help illustrate client and personnel functionality, and prove that both the sensors and the dashboard web application performs correctly and that it satisfies every requirements of the system.

1.2 Scope

This test plan will test all the features outlined in 2.2, Features to be Tested. Each feature that we test needs to pass with minimal or no errors or bugs encountered. Any unfixable bugs encountered need to be minimal at best, and should not interfere with the integrity of the system.

Each test to be ran will be ran under different environments, using a combination of different operating systems and different browsers. Each operating system will be documented with its name and the last installed update package. For each browser tested, we will document the version of that browser.

We are constrained on the number of Operating Systems and Browsers that can be used because we don't have very much money to work with. Some operating systems and browsers will not be tested because of this. An extensive list of the environments to be used will be outlined in Section 3.1- Environmental Needs.

1.3 List of Definitions and Abbreviations

- D.A.M.N. - Denton Air Monitoring Network
- Node - Air Quality Sensor
- AQI- Air Quality Index

1.4 Applicable Documents

- Detailed Design document

1.5 Overview

This Test Plan covers everything that we will test in Project D.A.M.N. It includes the features to be tested, the features not to be tested, the environments in which the platform was tested, a few risks that were identified in bringing this test plan to fruition. Upon successful execution of each item contained in this document, we should be able to deliver a fully functioning system that contains minimal runtime error and other issues. Overall, this process should take us no more than one month to accomplish and bring everything in fruition.

To accomplish all listed on this plan, this project will go through three testings trials to ensure everything is working correctly. Following the test trials will be a revision period to fix bugs and other issues encountered during the testing process.

2 Test Plan

This chapter will focus on the specifics of the test plan, including:

- Test Items
- Features to be tested
- Features not to be tested
- Testing approach and process
- Item pass/fail criteria
- Test deliverables
- Test tasks

2.1 Test Items

For each feature that is tested, we will be looking for full functionality. More specifically, does the feature work, does it throw errors, and if it does throw errors, how human friendly are these errors? Further, for user input, does the system allow for mistakes to be inputted into the system, or does it catch these errors? Lastly, we will ensure that for any errors encountered, the system can recover quickly and easily from the errors.

2.2 Features to be Tested

The features that will be tested prior to delivery include the following items, which are specified in the requirements specifications document:

Feature ID Number	Test ID Number	Description of Test
1.3.1.1	01	Provide users with a map of all sensors within city limits
1.3.1.2	02	Provide a method of looking up historical data
1.3.1.3	03	Show what sensors are online and offline
1.3.1.4	04	If a sensor is offline, show when it was last online or last “seen”
1.3.1.5	05	Display data for a single sensor
1.3.1.7	06	Allow a user to login to an account
1.3.1.8	07	Allow a user to create an account

1.3.1.9	08	When logged in, show a map of where the user's sensor's are located
1.3.1.10	09	When logged in, show sensor information
1.3.1.11	10	When logged in, show statistics from the user's sensor's
1.3.1.13	11	Verify that users are able to create an account
1.3.1.14	12	The map should use different icons for sensors that are online and offline
1.3.1.15	13	Add new sensors to the network
1.3.1.16	14	Delete sensors from the network
1.3.1.17	15	Store data from the sensor(s)
1.3.1.18	16	Accept user reported data
1.3.1.19	17	The system should accept data from any sensor, regardless of type.
1.3.1.21	18	It should be able to store any air quality value that a user is able to provide with their sensor
1.3.1.22	19	Verify that the data reported from the sensor is not an outlier, and somewhat matched the data reported from other sensors.
1.3.1.24	20	The system should be secure (see section 2.3-Features not to be Tested)
1.3.1.25	21	Raw data should be made available for download
1.3.1.26	22	The system should refresh itself with new data pulled from the sensors every five minutes.
1.3.1.27	23	System should notify a user if the data that is being pulled is an outlier

2.3 Features not to be Tested

The features that will not be tested prior to delivery include the following:

Feature ID Number	Feature Description	Reason for not Testing
1.3.1.0	Display air quality statistics in an easy to read, graphical format	Subjective to feedback from users, not suitable for the data that we aim to gather
1.3.1.6	Provide a help menu for users needing help with the system	There will be a help menu available, but the content within the help menu will not be tested as this would require surveys from humans and the data gathered from this would not provide meaningful test result data.
1.3.1.12	Provide location data of a sensor, either via address or through coordinates	This is something that has to be gathered regardless in order for the map system to work, therefore testing it would be nonsensical.
1.3.1.20	The database should be presented in a time series format	This feature is more a requirement than a feature that needs to be tested.
1.3.1.23	A large volume of users should be able to use this system without issue	We don't have a large pool of testers to ensure that this works. We're also limited to the power of the server that was provided to us.
1.3.1.24	The system should be secure.	We don't have the funds to run a thorough test of this, but we will do basic pen-testing.

2.4 Testing Approach & Process

There is just one rule and that is if at any point during the testing trials the tested item fails in any category, the test is deemed as a failure. We will rerun testing after revision when an issue occurs or when the tested item fails in any category.

In order to test both the dashboard and the sensors, we require the following to run the tests:

- 2.4.7- A computer or virtual machine with Windows 7
- 2.4.8- A computer or virtual machine with Windows 8.1
- 2.4.9- A computer or virtual machine with Windows 10
- 2.4.10- A Mac
- 2.4.11- Minimum of three Android Devices
- 2.4.12- Minimum of one iPhone

2.4.13- Sensors

2.4.43- VirtualBox or similar virtual machine software

Each software listed above should be using the latest version available. The version of each software listed will be made available with the test deliverable. We will run the tests using the following configurations:

- 2.4.14- Internet Explorer, Windows 7
- 2.4.15- Safari, Windows 7
- 2.4.16- Mozilla Firefox, Windows 7
- 2.4.17- Google Chrome, Windows 7
- 2.4.18- Opera, Windows 7
- 2.4.19- Internet Explorer, Windows 8.1
- 2.4.20- Safari, Windows 8.1
- 2.4.21- Mozilla Firefox, Windows 8.1
- 2.4.22- Google Chrome, Windows 8.1
- 2.4.23- Opera, Windows 8.1
- 2.4.24- Internet Explorer, Windows 10
- 2.4.25- Microsoft Edge, Windows 10
- 2.4.26- Safari, Windows 10
- 2.4.27- Mozilla Firefox, Windows 10
- 2.4.28- Google Chrome, Windows 10
- 2.4.29- Opera, Windows 10
- 2.4.30- Mac OS
- 2.4.31- Android Device 1, Default Browser Available
- 2.4.32- Android Device 1, Google Chrome
- 2.4.33- Android Device 2, Default Browser Available
- 2.4.34- Android Device 2, Google Chrome
- 2.4.35- Android Device 3, Default Browser Available
- 2.4.36- Android Device 3, Google Chrome
- 2.4.37- iPhone, Safari
- 2.4.38- iPhone, Google Chrome

The data that we will collect from testing includes:

- 2.4.39- Did the feature work as expected?
- 2.4.40- If any bugs were encountered, how many bugs were encountered, what were they?
- 2.4.41- How much data were we able to input?
- 2.4.42- How many user accounts can we create?

To ensure that both the dashboard and sensors are working and able to gather and display information correctly, the sensor and dashboard will undergo three intensive testing trials. The first trial being that the dashboard works from the dummy data provided and the other two from the live data provided by the sensors.

The process which we will test the system involves identifying the feature to be tested, testing it, and documenting the results of the test, whether the result is expected or not. Testers will input all data in a table similar to the following example:

Feature 1.3.1.1: Provide users with a map of all sensors within city limits			
Test ID Number: 01	Date: 2/13/2018	Test Trial Number: 1	Configuration ID: 2.4.24
Expected Result: A map shown with sensors on a map			
Actual Result: The map showed all sensors in the system on a map			
Test Result: Pass			

We have the appropriate amount of people to perform these tests to the extent that we want to run them, and expect only time to be the only constraint on this procedure. We anticipate that we will have three testing cycles, each cycle lasting for four days.

2.5 Item Pass/Fail Criteria

To pass and complete the test, all of the test requirements should be successfully met. Any outstanding issues remaining at the completion of this testing process should be minimal, and should not compromise the integrity of the system.

In order for an item to pass, it must succeed all usability tests prescribed in this document. If the item fails in a single category, it fails the entire test and must therefore be fixed.

When an item passes the test, it should still be tested in subsequent test trials, in case revisions of the system introduced more bugs into the system.

2.6 Test Deliverables

Here are the items that will be delivered upon completion of this test:

2.6.1- Test plan

- This document

2.6.2- Test Results

- Software versions of all software used
- Keep records of tests that were conducted
- Includes any issues identified during testing
- Summarizes the conclusion of testing

2.7 Test Tasks

For each feature to be tested, there are a set of tasks that must also be completed. Briefly, these include testing the item under all applicable environments, ensuring that all potential user errors are caught and handled appropriately, and ensuring the test is documented appropriately.

Each area of the system that allows for user input and interaction needs to be thoroughly tested for functionality and for error handling. If errors are not handled appropriately, the test will fail.

Lastly, the test needs to be documented, regardless of if it passes or fails. Notes regarding why the test passed or failed also needs to be documented. If the test fails, revisions to the system may be necessary, and the test needs to be reran, this too should be documented.

3 Execution

3.1 Environmental Needs

D.A.M.N should be tested on multiple operating systems and browsers and a combination of the two. Configuration information and identification can be found in Section 2.4, Testing Approach and Process. The platforms and software we will use include the following:

- Internet Explorer
- Microsoft Edge
- Safari
- Mozilla Firefox
- Google Chrome
- Opera
- Windows 7
- Windows 8.1
- Windows 10
- Mac OS
- iOS
- Android

3.2 Responsibilities

Each team member will have a role in the testing process. This is as follows:

- Alyssa: Overseer of the testing process, will assist with other tasks on an as needed basis
- Breuna: Tester- one who will perform the tests and document the results
- Travis: Revisionist- one who will work on identifying what causes a failed test and apply fixes as necessary
- James: Tester- one who will perform the tests and document the results

If there is an area in which the team needs more help, Alyssa will be responsible for filling in that gap. Her primary position, however, will be to ensure that this test plan is executed on schedule, and as specified through this document

3.3 Schedule

Task Name	Duration	Start	Finish
Prototype of System	24 Days	January 30, 2018	March 2, 2018
Prototype of Sensor	24 Days	January 30, 2018	March 2, 2018
Test Round 1	4 Days	March 3, 2018	March 7, 2018
Revisions	8 Days	March 8, 2018	March 17, 2018
Testing Round 2	5 Days	March 18, 2018	March 22, 2018
Final Revisions	6 Days	March 22, 2018	March 29, 2018
Final Testing	4 Days	March 30, 2018	April 3, 2018

3.4 Risks and Contingencies

The risks identified, as they pertain to this document, include:

3.4.1- Delays in the development process

- Where possible, the amount of time provided for testing will be extend out, to allow for proper testing. While time can only be extended so far, a “crunch time” period may have to be enforced.

3.4.2- Changes to the original requirements

- Meetings with the client and the team should help avoid these, though they do come up and should be dealt with as they are identified

3.4.3- Exceptional numbers of test failures

- To be resolved by fixing the systems, or in dire circumstances the requirements should be adjusted.

3.4.4- Lack of experience

- Outside help through other classmates and through resources provided by our client can be consulted.

3.5 Approvals

The following people will be in charge of approving this test plan and other items:

- Test Level Plan - Fantastic Four Group
- Integration Level Plan - Fantastic Four Group
- System Level Plan - Fantastic Four Group, Dan Minshew, and Kyle Taylor
- Master Level Plan - Dan Minshew and Kyle Taylor