FlightAdvice

Test Plan



Table of Contents

[1. INTRODUCTION 4](#_Toc458968453)

[Purpose and Scope 4](#_Toc458968454)

[Test Plan Overview 4](#_Toc458968455)

[2. TESTING STRATERGY 5](#_Toc458968456)

[Test Objectives 5](#_Toc458968457)

[Test Principles 5](#_Toc458968458)

[Scope 5](#_Toc458968459)

[Test Approach 5](#_Toc458968460)

[Defect tracking & Reporting 6](#_Toc458968461)

[Configuration Test Suite 7](#_Toc458968462)

[Unit Test Suite 8](#_Toc458968463)

[Integration Test Suite 9](#_Toc458968464)

[System Test Suite 10](#_Toc458968465)

[3. TEST EXECUTING STRATERGY 11](#_Toc458968466)

[4. TEST ENVIRONMENT 12](#_Toc458968467)

[5. TEST CASES 13](#_Toc458968468)

[Configuration Test Suite, Test Cases 13](#_Toc458968469)

[1. CT-01 - Test Arduino board read data from configured sensors 13](#_Toc458968470)

[2. CT-02 - Test Arduino board and SparkFun weather shield and read data from sparkfun inbuilt sensors 14](#_Toc458968471)

[3. CT-03 - Test the weather meter reads the correct wind speed and wind direction. 15](#_Toc458968472)

[4. CT-04 - Test the Arduino Uno it is transmitting correct data format defined in communication protocol. 16](#_Toc458968473)

[Unit Test Suite, Test cases 17](#_Toc458968474)

[1. UT-01 - Test connectivity between Controller Application and Arduino, verify data read, Verify Format. 17](#_Toc458968475)

[2. UT-02 - Test connectivity between Controller Application and Database 18](#_Toc458968476)

[3. UT-03 - Test Database individual table access. 19](#_Toc458968477)

[Integration Test Suite, Test cases 20](#_Toc458968478)

[1. IT-01 Test Database tables has required structure. 20](#_Toc458968479)

[2. IT-02 Test processing of sensor data. 21](#_Toc458968480)

[3. IT-03 - Test saving of sensor data to database table, read saved sensor data, validate data meets expected format 22](#_Toc458968481)

[System Test Suite, Test cases 23](#_Toc458968482)

[1. ST-01 - Test critical weather information web page publishes agreed information 23](#_Toc458968483)

[1. ST-01 - Test general weather information web page publishes agreed information 24](#_Toc458968484)

# INTRODUCTION

## Purpose and Scope

The purpose and scope of this document is to provide an overview of Flight Advice system testing plan. The test plan covers individual component of the system as well as the system out as a whole unit. The scope of this test plan is to capture and report sensor data information to the client via web pages.

This document has been developed by Nilu Herath for project FlightAdvice for the Client, Chalinor Baliuag (Lecturer) at Wellington Institute of Technology.

## Test Plan Overview

Ultimate goal of the test plan is to make sure Flight Advisor system delivers weather information to the Client via a web pages. To achieve such goal test plan will employee different types of testing strategies to suit. Flight Advisor system users various type of hardware components. Test plan will be using multiple types of software tools to test underlying hardware components.

Test plan draws multiple test suites where each suit will have one or more test cases. These test suits will be targeting system components in different level of testing. This Test plan will employee Manual and Automated test execution methods.

The system components that will get tested under this test plan are,

1. **Arduino**: The Hardware Weather Station is built on will be identified as Arduino. This Includes Arduino board, SparkFun Weather station and other hardware components.
2. **Controller Application**: This is a Java Application reading, processing and saving date to the database
3. **Database**: Database used to store sensor data
4. **Web Site**: Web site that will be hosting the web pages to be used by Client.

# 2. TESTING STRATERGY

## Test Objectives

The objective of the test is to verify that the functionality of Weather station system is working according to the specifications. The test will execute test cases identify bugs, fix the bugs and retest the defects.

## Test Principles

Testing will be focused on meeting the business objectives.

Testing processes will be well defined, yet flexible, with the ability to change as needed.

Testing activities will build upon previous stages.

Testing will be divided into different types of testing each with clearly defined objectives and goals.

## Scope

Testing the Flight Advisor System is built by multiple test suits. Each test suite may target one or more components in the system. A given test suite will consist one or more test cases.

Types of testing

1. Configuration Test Suite
2. Unit Test Suite
3. Integration Test Suite
4. System Test Suite

## Test Approach

Each iteration will be tested and test documents will be handed over to the client.

## Defect tracking & Reporting

Following flowchart depicts Defect Tracking Process:

## Configuration Test Suite

**System Components:**

* Arduino.

**Purpose:**

* Test configuration of the Arduino Uno board only.
* Test the Sparkfun weather shield with Arduino Uno if it reads sensor data from inbuilt sensors.
* Test the weather meter if reads the correct wind speed and wind direction.
* Test the Arduino Uno to check if it is transmitting correct data format for the communication protocol defined.

**Test Tool:**

* Arduino IDE with Serial Monitor

**Test Method and resources:**

* Manual test to be conducted by a developer.

**Test Deliverable**:

* Test cases.

**Test Execution Phase and Reporting:**

* Development Phase by the develop under UC -01 ‘TODO: use case name’.

## Unit Test Suite

**System Components:**

* Arduino
* Controller Application
* Database

**Purpose:**

* Test connectivity between Controller Application and Arduino
* Test connectivity between Controller Application and Database
* Test reading of sensor data
* Test sensor data format is according to agreed format
* Test database individual table access

**Test Tool:**

* Junit Framework

**Test Method and resources:**

* Automated test by a developer.

**Test Deliverable**:

* Test cases.

**Test Execution Phase and Reporting:**

* Development Phase by the develop under UC-02, UC-03, UC-04 and UC-05.

## Integration Test Suite

**System Components:**

* Arduino
* Controller Application
* Database

**Purpose:**

* Test Database has correct table structure
* Test process of sensor data
* Test saving of sensor data to database table
* Test reading sensor data from database table
* Test sensor data saved to database is valid and according to expected format.

**Test Tool:**

* Junit Framework, MySQL Workbench

**Test Method and resources:**

* Automated test by a developer.

**Test Deliverable**:

* Test cases.

**Test Execution Phase and Reporting:**

* Development Phase by the develop under UC-04 and UC-05.

## System Test Suite

**System Components:**

* Database
* Web Site

**Purpose:**

* Test critical weather information web page publishes agreed information
* Test general weather information web page publishes agreed information

**Test Tool:**

* Web Browser

**Test Method and resources:**

* Manual Test by Tester.

**Test Deliverable**:

* Test cases.

**Test Execution Phase and Reporting:**

* Testing phase by Tester covering UC-06, UC-07 deliverables.

# 3. TEST EXECUTING STRATERGY

# 4. TEST ENVIRONMENT

# 5. TEST CASES

Test cases section covers test cases assign for each test suite

## Configuration Test Suite, Test Cases

### 1. CT-01 - Test Arduino board read data from configured sensors

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**: CT-01 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Test Arduino Uno | | | | **Test Executed by:** | | | |
| **Test Title:** Test Arduino Uno if it is reading correct sensor data | | | | **Test Execution date:** | | | |
| **Description:** This test is carried out to make sure that Arduino board has been configured correctly | | | | **Test Tool:** Arduino IDE | | | |
| **Pre -Conditions:** one or two sensors that are giving correct sensor readings | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Install Arduino IDE on the laptop. |  | Read correct temperature and humidity reading | |  | **Pass** | **Done** |
| **2** | Write the code sketch to programme the Arduino board to read sensor data |  | **ID=1,T=22.00C,H=40.00%** | | **ID=1,T=22.00C,H=40.00%** | **Pass** | **Done** |
| **3** | Set up sensors correctly on a bread board and connect them to the Arduino board correctly. |  |  | |  | **Pass** | **Done** |
| **4** |  |  |  | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

### 2. CT-02 - Test Arduino board and SparkFun weather shield and read data from sparkfun inbuilt sensors

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**: CT-02 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Test Sparkfun weather shield | | | | **Test Executed by:** | | | |
| **Test Title:** Test Sparkfun weather shield if it is reading correct sensor data. | | | | **Test Execution date:** | | | |
| **Description:** This test is carried out to make sure that Sparkfun weather shield is reading correct sensor data. | | | | **Test Tool:** Arduino IDE | | | |
| **Pre -Conditions:** Arduino Uno that reads sensor data correct sensor data | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Connect Sparkfun weather shield with Arduino Uno board. |  | Read correct wind speed, wind direction, temperature, humidity, rainfall, barometric pressure and light readings. | |  |  |  |
| **2** |  |  |  | |  |  |  |
| **3** |  |  |  | |  |  |  |
| **4** |  |  |  | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

### 3. CT-03 - Test the weather meter reads the correct wind speed and wind direction.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**: CT-03 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Test Weather meter | | | | **Test Executed by:** | | | |
| **Test Title:** Test weather meter if it is reading correct sensor data. | | | | **Test Execution date:** | | | |
| **Description:** This test is carried out to make sure that weather meter is reading correct sensor data. | | | | **Test Tool:** Arduino IDE | | | |
| **Pre -Conditions:** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Connect the weather meter with the simulator. |  | Read correct wind speed, wind direction. | |  |  |  |
| **2** | Check the readings with a digital wind speed and wind direction meter. |  |  | |  |  |  |
| **3** |  |  |  | |  |  |  |
| **4** |  |  |  | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

### 4. CT-04 - Test the Arduino Uno it is transmitting correct data format defined in communication protocol.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**: CT-04 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Test Arduino IDE reading | | | | **Test Executed by:** | | | |
| **Test Title:** Test Arduino IDE if it is reading correct sensor data. | | | | **Test Execution date:** | | | |
| **Description:** This test is carried out to make sure that that Arduino programme to read sensor data is reading correct sensor data. | | | | **Test Tool:** Arduino IDE | | | |
| **Pre -Conditions:** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Connect the weather meter with the simulator. |  | Read sensor data line. The data line read should comply to the format defined in communication protocol section | |  |  |  |
| **2** |  |  |  | |  |  |  |
| **3** |  |  |  | |  |  |  |
| **4** |  |  |  | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

## Unit Test Suite, Test cases

### 1. UT-01 - Test connectivity between Controller Application and Arduino, verify data read, Verify Format.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**: UT-01 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Test Controller application | | | | **Test Executed by:** | | | |
| **Test Title:** Test Controller application if it is reading correct sensor data. | | | | **Test Execution date:** | | | |
| **Description:** This test carried to make sure that all the functions in the controller application are working correct. | | | | **Test Tool:** JUnit | | | |
| **Pre -Conditions:** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Run Controller application after connecting USB cable to Weather Station. Execute USB data reader. |  | USB Data reader to read a data lines return by Arduino Board | |  |  |  |
| **2** | Verify data lines read is according to agreed communication format |  | The data line read matches agreed communication protocol format. | |  |  |  |
| **3** |  |  |  | |  |  |  |
| **4** |  |  |  | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

### 2. UT-02 - Test connectivity between Controller Application and Database

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**: UT-02 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Test Controller application | | | | **Test Executed by:** | | | |
| **Test Title:** Test Controller application is establishing connection to database. | | | | **Test Execution date:** | | | |
| **Description:** This test carried to make sure that controller application is capable of establishing connection with the database. | | | | **Test Tool:** JUnit | | | |
| **Pre -Conditions:** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Run Controller application and activate connection to database. |  | To establish successful connection with the database. Verify establish connection message against console output | |  |  |  |
| **2** |  |  |  | |  |  |  |
| **3** |  |  |  | |  |  |  |
| **4** |  |  |  | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

### 3. UT-03 - Test Database individual table access.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**: UT-03 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Test Controller application | | | | **Test Executed by:** | | | |
| **Test Title:** Test Controller application is establishing connection to database tables defined. | | | | **Test Execution date:** | | | |
| **Description:** This test carried to make sure that controller application is capable of accessing database tables. | | | | **Test Tool:** JUnit | | | |
| **Pre -Conditions:** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Run Controller application and activate connection to database. |  | To establish successful connection with the database. Verify establish connection message against console output | |  |  |  |
| **2** | Trigger Access of Weather Station Table |  | Successful table access message in console output | |  |  |  |
| **3** | Trigger Access of Sensor Table |  | Successful table access message in console output | |  |  |  |
| **4** | Trigger Access of Sensor Data Table |  | Successful table access message in console output | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

## Integration Test Suite, Test cases

* Test process of sensor data
* Test saving of sensor data to database table
* Test reading sensor data from database table
* Test sensor data saved to database is valid and according to expected format.

### 1. IT-01 Test Database tables has required structure.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**:IT-01 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Test Controller Application, Database | | | | **Test Executed by:** | | | |
| **Test Title:** Test database to test if it has the correct structure. | | | | **Test Execution date:** | | | |
| **Description:** This test carried to make sure that database tables has setup with required structure | | | | **Test Tool:** Junit, MySQL Workbench | | | |
| **Description:** | | | |  | | | |
| **Pre -Conditions:** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Run Controller application and activate connection to database. |  | To establish successful connection with the database. Verify establish connection message against console output. | |  |  |  |
| **2** | Trigger Access of Weather Station Table |  | Successful table access message in console output | |  |  |  |
| **3** | Trigger insert sample record to Weather Station Table |  | Record to be insert into weather station table | |  |  |  |
| **4** | Trigger Access of Sensor Table |  | Successful table access message in console output | |  |  |  |
| **5** | Trigger insert sample record to Sensor Table |  | Record to be insert into sensor table | |  |  |  |
| **6** | Trigger Access of Sensor Data Table |  | Successful table access message in console output | |  |  |  |
| **7** | Trigger insert sample record to sensor data Table |  | Record to be insert into sensor data table | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

### 2. IT-02 Test processing of sensor data.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**:IT-02 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Test Controller Application | | | | **Test Executed by:** | | | |
| **Test Title:** Test controller application processors incoming sensor data | | | | **Test Execution date:** | | | |
| **Description:** This test carried to make sure that Controller Application processes sensor data with introduction of ReadDateTime value. | | | | **Test Tool:** Junit | | | |
| **Description:** | | | |  | | | |
| **Pre -Conditions:** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Run Controller Application Data Reader and read incoming sensor data |  | Controller Application to introduce ReadDateTime to sensordata read line. | |  |  |  |
| **2** |  |  |  | |  |  |  |
| **3** |  |  |  | |  |  |  |
| **4** |  |  |  | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

### 3. IT-03 - Test saving of sensor data to database table, read saved sensor data, validate data meets expected format

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**: IT-02 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Test Controller Application | | | | **Test Executed by:** | | | |
| **Test Title:** Test controller application processors incoming sensor data, process, saving data and meets format requirements. | | | | **Test Execution date:** | | | |
| **Description:** This test carried to make sure that Controller Application processes and saved the data according to correct format. | | | | **Test Tool:** Junit, MySQL Workbench | | | |
| **Description:** | | | |  | | | |
| **Pre -Conditions:** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Run Controller Application Data Reader and read incoming sensor data |  | Controller Application to introduce ReadDateTime to sensor data read line. | |  |  |  |
| **2** | Trigger Process and save tasks to save processed sensor data to the table |  | Sensor data to be saved successfully to the table with all the required columns populated | |  |  |  |
| **3** | Trigger process to read sensor data from the table |  | Process to read all the records save to the table. | |  |  |  |
| **4** |  |  |  | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

## System Test Suite, Test cases

* Test critical weather information web page publishes agreed information
* Test general weather information web page publishes agreed information

### 1. ST-01 - Test critical weather information web page publishes agreed information

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**:ST-01 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Web site | | | | **Test Executed by:** | | | |
| **Test Title:** Test Web Site for publishing of Critical Weather information. | | | | **Test Execution date:** | | | |
| **Description:** Web site to publish Critical Weather information specified by Customer Requirements. | | | | **Test Tool:** Web Browser | | | |
| **Pre -Conditions:** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Load Web site by the given URL and load Critical Weather information page |  | Web page to publish critical weather information as for customer’s specification. | |  |  |  |
| **2** |  |  |  | |  |  |  |
| **3** |  |  |  | |  |  |  |
| **4** |  |  |  | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |

### 1. ST-01 - Test general weather information web page publishes agreed information

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID**:ST-01 | | | | **Test Designed by:** | | | |
| **Test Priority (Low/Medium/High):** | | | | **Test Designed date:** | | | |
| **Module Name:** Web site | | | | **Test Executed by:** | | | |
| **Test Title:** Test Web Site for publishing of general Weather information. | | | | **Test Execution date:** | | | |
| **Description:** Web site to publish General Weather information specified by Customer Requirements. | | | | **Test Tool:** Web Browser | | | |
| **Pre -Conditions:** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Steps** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1** | Load Web site by the given URL and load General Weather information page |  | Web page to publish general weather information as for customer’s specification. | |  |  |  |
| **2** |  |  |  | |  |  |  |
| **3** |  |  |  | |  |  |  |
| **4** |  |  |  | |  |  |  |
| **5** |  |  |  | |  |  |  |
| **6** |  |  |  | |  |  |  |
| **7** |  |  |  | |  |  |  |
| **8** |  |  |  | |  |  |  |
| **9** |  |  |  | |  |  |  |
| **10** |  |  |  | |  |  |  |
| **Post-Conditions:** | | | | | | | |