Table of Contents

Testing Strategies:

Smoke Testing:

Smoke testing is a type of software testing that determines whether the most crucial functions of a program are working correctly. It is sometimes called “build verification testing,” and it is usually the first type of testing done after a new software version has been built. Smoke testing is used to quickly identify major issues that would prevent a program from working properly and require more in-depth testing.

White Box Testing:

White box testing is a testing technique that is used within software's internal design and structure to verify the inputs and outputs, the security of the software and the usability of the software. White box testing can verify many things within the software such as:

- security flaws within the software
- coding paths that could be flawed/broken
- the intended output
- the functionality of conditional loops
- the testing of individual statements/objects

One of the main types of white box testing is unit testing which is where you develop a few lines of code, a single function or an object and test it to make sure it works before continuing as this can allow a programmer to spot major bugs early on so they can be fixed before moving on to the next part of the program.

Black Box Testing:

Black box testing is a testing technique that is used to check the external inputs and outputs that are generated by the system. Black box testing is usually done without knowing how the program

functions which can make it easier for the person to identify how quickly the system responds, as well as how the program handles certain user inputs.

There are 3 main types of black box testing which include:

- Functional Testing - focuses on the most important aspects of the software, for example, being able to log into an account with the correct information, it also focuses of the integration between certain key components, or the functional testing can focus on the entire system.
- Non-Functional Testing - checks how the system performs any specific action that the user wants it to, this means that by testing the performance the user is able to see how usable and easy to understand the information is when being presented, how responsive the system is when given heavy work loads, how compatible the system is with multiple devices and it could also expose any, before hidden security vulnerabilities or other common types of security threats.
- Regression Testing - Regression testing refers to a software testing technique that re-runs non-functional and functional tests to ensure that a software application works as intended after any code changes, updates, revisions, improvements, or optimisations.

Test Plan:

All of the tests below will be carried out in the way that they are placed within the tables below as this way will be the easiest and most effective way to test each componenet individually.

Registration Page:

Description of Test	Component that will be tested	Type of test	Data that will be used	Expected Outcome	Prerequisites and dependencies
Testing the registration page with correct account information	Register Page	White box testing	First Name: test Last Name: test Email Address: test@gmail.com Password: test!ng1 Confirm Password: test!ng1	For the expected outcome the program should take all of the inputs and validate them before adding the information to a data base and outputting a message that states that the account has been made	The program would need to have set validation in place for each of the inputs that is reqiured to ensure that the user accounts will be secure when being created The program may also need set error messages if certain inputs are incorrect or are in the wrong format so the user knows

					which part is incorrect
Testing the registration page but leaving the first and last name blank	Register Page	White box testing	First Name: Last Name: Email Address: test@outlook.com Password: test!ng1 Confirm Password: test!ng1	The program should take the inputs and realise that the first and last name have been left blank and should remind the user that the inputs should not be left blank After this it should give the user a chance to re-enter the information before checking again	The program would need to have set validation in place for each of the inputs that is reqiured to ensure that the user accounts will be secure when being created The program may also need set error messages if certain inputs are incorrect or are in the wrong format so the user knows which part is incorrect
Testing the registration page with a blank email address	Register Page	White box testing	First Name: testing Last Name: testing Email Address: Password: test!ng1 Confirm Password: test!ng1	The program should take the inputs and validate the first name and last name and stop at the email address as there is nothing there, so the program should remind the user that the email address is invalid and give the user another chance to re-enter the email address	The program would need to have set validation in place for each of the inputs that is reqiured to ensure that the user accounts will be secure when being created The program may also need set error messages if certain inputs are incorrect or are in the wrong format so the user knows which part is incorrect
Testing the registration page with a password that doesnt contain	Registration Page	Black box testing	First Name: tester Last Name: tester	The program should validate all of the other inputs	The program would need to have set

special characters, numbers or capitals			Email Address: tester@hotmail.com Password: testing Confirm Password: testing	however it should check the password to see that it doesnt meet the security requirements, so every time it doesnt meet one of the requirements it adds to the error message before outputting the error message to the user stating everything that is wrong with the password so that the user knows what to change	validation in place for each of the inputs that is reqiured to ensure that the user accounts will be secure when being created The program may also need set error messages if certain inputs are incorrect or are in the wrong format so the user knows which part is incorrect
Testing the registration page with a password that isnt the same as the confirmation password	Registration Page	Black box testing	First Name: testname Last Name: testname Email address: tesname@gmail.com Password: goodPa$$word Confirm Password: goodpassword	The program should identify that the password and the confirmation password do not match and output a suitable error message stating that the passwords do not match so the account cannot be made	The program would need to have set validation in place for each of the inputs that is reqiured to ensure that the user accounts will be secure when being created The program may also need set error messages if certain inputs are incorrect or are in the wrong format so the user knows which part is incorrect

Login Page:

Description of Test	Component that will be tested	Type of test	Data that will be used	Expected Outcome	Prerequisites and dependencies
Testing the login page with valid details to find out if the system allows the user to log in	Login Page	Black box testing	Email Address: test@gmail.com Password: test!ng1	The program should allow the user to log in as the details are valid	The program would have to check the database in order to find out whether or not the email and password exist, once the database has been check the program make another decision based on this outcome
Testing the login page with an invalid password but a correct email address	Login Page	White box testing	Email Address: test@gmail.com Password: testing	The program should check the database to see if the email address exists, then it should check if the passwords match, however because they dont an error message should be displayed to the user explaining that the password is incorrect	The program would need to be connected to the database in order for the program to gain access to check if the information matches
Testing the login page with an email address that doesnt exist within the database	Login Page	White box testing	Email Address: thisisatest@yahoo.com Password: test!ng1	The program shouldnt crash because the email address doesnt exist, instead it should output a suitable error message stating that the email address that the user has entered doesnt exist.	The program would need to be connected to the database in order for the program to gain access to check if the information matches before going back to the program to make a decision based off of the result given by the database

Home Page:

Description of Test	Component that will be tested	Type of test	Data that will be used	Expected Outcome	Prerequisites and dependencies
Testing the home page to ensure that if the user enters a certain location, it will change the weather forecast to the location that the user has chosen	Home Page forecast	Black Box testing	Location: Leeds Swapped from London	The program should be able to update in real time once the user has changed the location to the one that they would like to view	The program would need to have webscraping functions in order to find the weather forecast for the selected location, as well as being able to input the users location choice to the website
Testing the home page to ensure that the health and weather warnings are being updated in real time	Home Page weather/health warnings	Black box testing	Location: Leeds Swapped from London	The program should update the health and weather warnings/concerns every time that the location is updated to ensure that the users are getting the most up to date and accurate information possible	The program would need to have webscraping functions in order to find the health and weather warnings/concerns for the selected location, as well as being able to input the users location choice to the website

Web Scraping:

Description of Test	Component that will be tested	Type of test	Data that will be used	Expected Outcome	Prerequisites and dependencies
Testing to see if the web scraping data for the forecast for today is correct	Web scraping data	Black box testing	Data that has been web scraped	The data should be in a format that should be easy to read format so that it can be displayed to the user in a meaningful way without being too overwhelming for the user	The program would need to be able to convert the html code that has been scraped from the website in order to make it easy for the user to understand as well as being able to be processed by the program

Testing to see if the web scraped data will be in an easy to read format for the 5 day forecast	Web scraping data	Black box testing	Data that has been web scraped for the 5 day forecast	The data should be in a format that should be easy to read format so that it can be displayed to the user in a meaningful way without being too overwhelming for the user	The program would need to be able to convert the html code that has been scraped from the website in order to make it easy for the user to understand as well as being able to be processed by the program
Testing to see if the web scraped data for the health/weather concerns will be displayed in an easy to read format	Web scraping data	Black box testing	Data that has been web scraped from the website for the health/weather concerns and warnings	The data should be in a format that should be easy to read format so that it can be displayed to the user in a meaningful way without being too overwhelming for the user	The program would need to be able to convert the html code that has been scraped from the website in order to make it easy for the user to understand as well as being able to be processed by the program
Testing to see if the webscraped data for the air quality index will be displayed in an easy to read format	Web scraping data	Black box testing	Data that has been web scraped from the website about the current air quality index as well as what some suitable activities would be due to the air quality index	The data should be in a format that should be easy to read format so that it can be displayed to the user in a meaningful way without being too overwhelming for the user	The program would need to be able to convert the html code that has been scraped from the website in order to make it easy for the user to understand as well as being able to be processed by the program