

WeatherPlus: Your Personalized Weather Companion

Keyur Patel - 16010421073
Ronit Mehta - 16010421056
Batch - A2
Exp-8(Software Test Document)

April 2024

Aim:

To prepare Software Test Document (STD)

Resources:

Internet Explorer, LaTeX Editor

Results: Software Test Document (STD) in given format

1 Introduction

1.1 System Overview The Weather Forecasting App is designed to provide users with accurate and timely weather information for their specified locations. Users can search for weather forecasts, add favorite locations, customize settings, and receive weather alerts. The current version of the software to be tested is v1.0.

1.2 Test Approach The following are the major groups of features and their corresponding testing approaches:

1. User Interface Testing:

- **Approach:** Manual testing of the user interface elements.
- **Activities:** Validate layout, navigation, and responsiveness across devices.
- **Techniques:** Exploratory testing, UI walkthroughs.
- **Tools:** Manual testing, browser dev tools.
- **Constraints:** Limited device availability for physical testing.

2. Weather Data Retrieval and Display:

- **Approach:** Automated and manual testing of data accuracy and display.
- **Activities:** Verify weather data accuracy, forecast display, and updates.
- **Techniques:** Test data against known values, boundary value analysis.
- **Tools:** API testing tools (Postman), manual validation.
- **Constraints:** Reliability of external weather API services.

3. User Authentication and Profile:

- **Approach:** Automated and manual testing of user authentication and profile management.
- **Activities:** Validate login/logout functionality, user profile creation, and settings.
- **Techniques:** Positive and negative testing of credentials, boundary testing.
- **Tools:** Test scripts, manual testing.
- **Constraints:** Security considerations for user data.

4. Favorites Management:

- **Approach:** Automated and manual testing of adding, removing, and listing favorites.
- **Activities:** Verify adding and removing locations from favorites, displaying favorites.
- **Techniques:** Equivalence partitioning, user scenario testing.
- **Tools:** Test scripts, manual testing.
- **Constraints:** Synchronization of favorites across devices.

5. Notifications and Alerts:

- **Approach:** Automated and manual testing of weather alerts.
- **Activities:** Test notification triggers, display of alerts.
- **Techniques:** Test boundary conditions for alerts, real-time testing.
- **Tools:** Test scripts, manual testing.
- **Constraints:** Consistency and timeliness of notifications.

2 Test Plan

2.1 Features to be tested

1. User Interface Testing:

- **Test Case:** Verify layout, navigation, and responsiveness on different devices.

2. Weather Data Retrieval and Display:

- **Test Case:** Validate accuracy of weather data, forecast display, and updates.

3. User Authentication and Profile:

- **Test Case:** Confirm login/logout functionality, user profile creation, and settings.

4. Favorites Management:

- **Test Case:** Ensure adding, removing, and listing favorite locations.

5. Notifications and Alerts:

- **Test Case:** Test notification triggers and display of alerts.

2.2 Features not to be tested

1. The following features will not be tested:

- Third-party weather API service reliability and availability.
- Device-specific issues on unsupported devices.

2.3 Testing Tools and Environment

1. Test Staffing:

- *For individual project:* 3 hours per week for testing.
- For group project: 2 testers, each working 1.5 hours per week.

2. Test Environment Requirements:

- *Space:* Dedicated testing area with workstations.
- *Hardware:* Mobile devices for testing mobile app compatibility.
- *Software:* Latest browsers (Chrome, Firefox, Safari), API testing tools (Postman).
- Special Test Tools: Load testing tools (JMeter), automation testing framework (Selenium).

3. Additional Needs:

- Internet access for external API testing.
- Cloud storage for test data and results.
- Collaboration tools for remote team testing.

3 Test Cases

Test Case ID	Test Case	Purpose	Input	Expected Output	Test Procedure
TC-001	UI Layout	Verify layout and responsiveness	Open app on different devices	App displays properly without layout issues	1. Open on desktop browser. 2. Resize window for tablet. 3. Open on mobile. Verify layout.
TC-002	Weather Data	Test weather data retrieval and accuracy	Search for "New York" weather	Weather data matches expected values	1. Search "New York" weather. 2. Verify temperature, humidity, conditions. Repeat for different locations.
TC-003	User Login/Logout	Validate user login and logout functionality	Enter valid username and password	Redirect to home screen after login, to login screen after logout	1. Enter valid credentials and login. 2. Verify redirection. 3. Click "Logout" and verify redirection to login screen.
TC-004	Favorites Management	Test adding and removing favorite locations	Add "London" to favorites	"London" added and removed from favorites list	1. Add "London" to favorites. 2. Verify in favorites list. 3. Remove "London" from favorites. 4. Verify removal.

4 Test Logs

1. Test Log 1: TL-001

- (a) **Test Case:** TC-001 (UI Layout)

- (b) **Date:** April 20, 2024
- (c) **Tester:** Ronit
- (d) **Environment:** Desktop, Tablet, Mobile
- (e) **Status:** Pass
- (f) **Description:**
 - Opened the app on desktop, tablet, and mobile devices.
 - Verified that the layout was consistent and responsive across all devices.
- (g) **Notes:** No layout issues observed.

2. Test Log 2: TL-002

- (a) Test Case: TC-002 (Weather Data)
- (b) Date: April 21, 2024
- (c) Tester: Keyur
- (d) Environment: App
- (e) Status: Pass
- (f) Description:
 - Searched for "New York" weather.
 - Verified temperature, humidity, and conditions matched expected values.
- (g) Notes: Weather data for "New York" displayed accurately.

3. Test Log 3: TL-003

- (a) **Test Case:** TC-003 (User Login/Logout)
- (b) **Date:** April 22, 2024
- (c) **Tester:** Keyur
- (d) **Environment:** App
- (e) **Status:** Pass
- (f) **Description:**
 - Entered valid credentials and clicked "Login".
 - Verified redirection to home screen.
 - Clicked "Logout" and verified redirection to login screen.
- (g) **Notes:** User authentication and redirection functioning as expected.

4. Test Log 4: TL-004

- (a) **Test Case:** TC-004 (Favorites Management)
- (b) **Date:** April 22, 2024
- (c) **Tester:** Ronit
- (d) **Environment:** App
- (e) **Status:** Pass
- (f) **Description:**
 - Added "London" to favorites.
 - Verified "London" appeared in favorites list.
 - Removed "London" from favorites.
 - Verified "London" was removed from favorites list.
- (g) **Notes:** Favorites management working correctly.

5 Test Results

Test Result 1:

- **Date/Time:** April 20, 2024 - 10:00 AM
- **Test Case:** TC-001 (UI Layout)
- **Status:** Pass
- **Observation:**
 - Layout displayed correctly on desktop, tablet, and mobile devices.
 - No errors or anomalies observed.

Test Result 2:

- **Date/Time:** April 21, 2024 - 11:30 AM
- **Test Case:** TC-002 (Weather Data)
- **Status:** Pass
- **Observation:**
 - * Weather data for "New York" displayed accurately.
- **Temperature:** 65°F, Humidity: 45No errors in data retrieval.

Test Result 3:

- **Date/Time:** April 22, 2024 - 9:00 AM
- **Test Case:** TC-003 (User Login/Logout)
- **Status:** Pass
- **Observation:**
 - Successfully logged in with valid credentials.
 - Redirected to home screen after login.
 - Logged out and redirected to login screen.
 - No errors in authentication process.

Test Result 4:

- **Date/Time:** April 22, 2024 - 2:00 PM
- **Test Case:** TC-004 (Favorites Management)
- **Status:** Pass
- **Observation:**
 - Added "London" to favorites successfully.
 - "London" appeared in favorites list.
 - Removed "London" from favorites and it disappeared from the list.
 - No errors in adding or removing favorites.

Questions:

1. Differentiate between verification and validation.

(a) **Verification:**

- **Definition:** Verification ensures that the software is being developed correctly, confirming that the software is meeting its specified requirements.
- **Focus:** It focuses on the process of development, answering the question, "Are we building the product right?"
- **Activities:** Activities include reviews, walkthroughs, and inspections to check documents, code, and specifications.
- **Goal:** The goal is to identify whether the software conforms to its specification.

(b) **Validation:**

- **Definition:** Validation ensures that the software meets the needs of the customer or end-user, confirming that the right product is being built.
- **Focus:** It focuses on the end product, answering the question, "Are we building the right product?"
- **Activities:** Activities include testing the software against user requirements and user acceptance testing (UAT).
- **Goal:** The goal is to verify whether the software fulfills its intended use in the real world.

2. List down all OO software testing strategies.

- (a) **Unit Testing:** Testing individual units or components of the software in isolation. Ensures that each unit functions correctly.
- (b) **Integration Testing:** Testing the integration of different units or modules together. Checks if the interactions between integrated components work as expected.
- (c) **System Testing:** Testing the entire system as a whole. Verifies that all components work together as intended.
- (d) **Acceptance Testing:** Includes User Acceptance Testing (UAT). Testing conducted to determine whether the software meets the user's requirements and can be accepted for delivery.
- (e) **Regression Testing:** Testing to ensure that new changes or fixes have not adversely affected existing functionalities. Helps in maintaining the integrity of the software over time.
- (f) **Object-Oriented Testing:** Testing that specifically targets object-oriented features like inheritance, polymorphism, encapsulation. Ensures that these features are implemented and behave as expected.

Outcomes

CO4 Demonstrate test case design

Conclusion

After completing this experiment we have successfully created Software Test Document for my Weather app forecasting project.

References

Books:

1. Roger S. Pressman, Software Engineering: A practitioners Approach, 7th Edition, McGraw Hill, 2010.
2. Ian Somerville, Software Engineering, 9th edition, Addison Wesley, 2011.
3. <http://vlabs.iitkgp.ernet.in/se/>