

Requirements

Requirements		Type of test	Pass or Fail	Contributor
		(I: Integration S: System , O: Operational, , UN: Unit, , US: Usability, A: Acceptance)	(P: Pass, F: Fail)	(SS: Syed Saad, AN: Ashish Nayak , TN: Tawana Ndlovu, JS: Jan Suratos)
Functional Requirements				
1- User				
1.1	Verify that a local user view weather on app opened when connected to Wi-Fi or cellular network with data enabled.	S, A		
1.2	Users must be able to view a five day forecast	S, A		
1.3	Users must be able search a location by name	S, A		
1.4	User must be able to create an account if doesn't have one already	O, A		
1.5	User must login to access the following features	S, A		
1.6	User must be able to login to access and edit their dashboard information	S, A		
1.7	User must be able to save up to five locations	S, A		
1.8	User must be able to share their dashboard information	S, A		
1.9	User must receive error message if location is not sent	S, A		
1.10	User must be able to customize dashboard from given options			
2 - Weather Data Retrieval				
2.1	The system must connect to the external WeatherApp API and retrieve data for the current weather.	I, S		
2.2	The system must connect to the external WeatherApp API and retrieve 5-day forecast data.	I, S		
2.3	The system must update weather data on the dashboard at regular intervals.	UN		
2.4	The system must handle API errors and display appropriate messages to the user.	O, A		
2.5	The system must format and display temperature, weather conditions, and weather icons.	A		
3 - Dashboard Customization				
3.1	The system must allow users to customize the dashboard by adding or removing locations	US, A		
3.2	The system must allow users to change the color of the dashboard.	US, A		
3.3	The system must allow users to set a preferred format for displaying weather data.	US, A		
3.4	The system must save dashboard preferences upon user action.	S, A		
3.5	The system must load saved preferences when the user logs in.	S, A		
4 - Firebase Authentication				
4.1	The system must use the Firebase API to authenticate users during login.	UN, S		
4.2	The system must securely store user credentials via Firebase.	UN, S		
4.3	The system must restrict access to dashboard and settings for unauthenticated users.	UN, S		
4.4	The system must log out users and end sessions upon request.	UN, S		
5 - CI/CD Pipeline				
5.1	The system must automatically build and test on every new commit.	S		
5.2	The system must deploy succesfully without errors.	S		
5.3	The system must send notification for failed builds or tests.	S		

6 - Automated Testing

- | | |
|--|-------|
| 6.1 The system must run unit tests on critical components like API requests and data formatting. | UN, S |
| 6.2 The system must run integration tests for end-to-end user flows. | UN, S |
| 6.3 The system must run User Interface tests to ensure a responsive and accessible interface. | UN, S |

Non Functional Requirements

1 - Security

- | | |
|---|-------|
| 1.1 The system must prevent unauthorized access to user accounts , data and personal information. | O |
| 1.2 The system must use secure connections (HTTPS) for API requests. | O, UN |

2 - Usability

- | | |
|--|----|
| 2.1 The system must have a user-friendly interface that is intuitive for all users | US |
| 2.2 The system must allow users to recover from all errors and exceptions without crashing | US |
| 2.3 The system must provide help messages for key user actions. | US |
| 2.4 The application must be able to run on various browsers | US |
| 2.5 The application must be able to run on various devices | US |

3 - Reliability

- | | |
|---|-------|
| 3.1 The system must use recover from API errors and inform users. | O, US |
| 3.2 The system must maintain session data for the duration of user interaction. | O, A |

4 - Performance

- | | |
|---|----------|
| 4.1 The system must load weather data within 5 seconds of request. | O, A, US |
| 4.2 The system must function optimally on moderate internet connection. | O, A, US |

5 - Maintainability

- | | |
|---|-------|
| 5.1 The system must be modular to support easy updates and bug fixes. | UN, S |
| 5.2 The codebase must follow a consistent style guide to improve readability. | S, I |
| 5.3 Every section of code must be documented to support easy updates and bug fixes. | S, I |

6 - Availability

- | | |
|---|----------|
| 6.1 The system must be available 100% of the time during peak usage hours. | US, A, O |
| 6.2 The system must display maintenance message if temporarily unavailable. | US |

