User Acceptance Testing (UAT) Plan for GoPal Application

1. Introduction

This document outlines the User Acceptance Testing plan for the GoPal application. The purpose of UAT is to verify that the application meets business requirements and functions correctly from the end-user perspective before final deployment.

2. Test Environment

- **Application Environment**: Local development environment or deployed test environment
- **Browser**: Chrome (latest version), Firefox (latest version), Safari (latest version)
- **Devices**: Desktop/laptop computers, mobile devices (optional for responsive testing)
- **Test Database**: Test database with pre-populated test data for users, activities, and achievements

3. Test Users

The UAT will be conducted by:

- Team members (internal testing)
- 3-5 potential users outside the development team (to provide unbiased feedback)
- Users should have varying levels of technical expertise and fitness backgrounds

4. Features to Test

Feature 1: User Registration and Authentication

Test Case 1.1: User Registration

Description: Verify that a new user can successfully register for an account
Test Steps:

1. Navigate to the registration page
2. Enter valid email, username, and password
3. Submit the form
4. Verify successful account creation

Expected Results:

- User is directed to the login page after successful registration
- User data is stored in the database with a hashed password
- System displays appropriate success message

Test Data:

- Email: testuser@example.com

- Username: testuser123
- Password: TestPass123!

Test Case 1.2: Password Validation

Description: Verify that password requirements are enforced during registration
Test Steps:

1. Navigate to the registration page
2. Enter valid email and username
3. Enter a password that doesn't meet requirements (e.g., too short, no uppercase, no special character)
4. Submit the form

Expected Results:

- Registration should fail
- System should display error message explaining password requirements
- No user account should be created in the database

Test Data:

- Email: testuser@example.com
- Username: testuser123
- Password: weak

Test Case 1.3: User Login

Description: Verify that registered users can successfully log in
Test Steps:

1. Navigate to the login page
2. Enter valid username and password for an existing account
3. Submit the form

Expected Results:

- User is successfully logged in
- User is redirected to the home dashboard
- Session is created for the authenticated user

Test Data:

- Username: testuser123
- Password: TestPass123!

Feature 2: Activity Tracking

Test Case 2.1: Activity Addition

Description: Verify that users can add a new activity
Test Steps:

1. Log in with valid credentials
2. Navigate to the Activity section
3. Click on "Add Activity" button
4. Fill in all required fields (activity type, duration, distance, date, time)
5. Submit the form

Expected Results:

- New activity is saved to the database
- Activity appears in the user's activity list
- Activity statistics are updated to include the new activity

Test Data:

- Activity Type: Running
- Duration: 30 minutes
- Distance: 3.5 miles
- Date: Current date
- Time: Current time

Test Case 2.2: Activity Deletion

Description: Verify that users can delete their activities
Test Steps:

1. Log in with valid credentials
2. Navigate to the Activity section
3. Locate an existing activity
4. Click the delete button for that activity
5. Confirm deletion

Expected Results:

- Activity is removed from the database
- Activity no longer appears in the user's activity list
- Activity statistics are updated to reflect the deletion

Test Data:

- Use an existing activity from the test user's account

Test Case 2.3: Activity Statistics

Description: Verify that activity statistics are accurately calculated
Test Steps:

1. Log in with valid credentials
2. Add multiple activities with known values
3. Navigate to the Activity section
4. Review the calculated statistics (total activities, distance, duration, streak)

Expected Results:

- Weekly summary shows correct number of activities
- Total distance matches sum of all activity distances
- Total duration matches sum of all activity durations
- Current streak is calculated correctly based on consecutive days with activities

Test Data:

- Multiple activities with varied dates, distances, and durations

Feature 3: Weather Integration

Test Case 3.1: Weather Display

Description: Verify that current weather information is displayed on the home page
Test Steps:

1. Log in with valid credentials
2. Navigate to the home page
3. Verify that weather information is displayed
4. Allow location access if prompted

Expected Results:

- Weather widget displays on the home page
- Temperature, conditions, wind speed, humidity, and AQI are shown
- City name is correctly identified based on geolocation

Test Data:

- User with geolocation permissions enabled

Test Case 3.2: Weather Data Accuracy

Description: Verify that weather data is accurate and current
Test Steps:

1. Log in with valid credentials
2. Navigate to the home page
3. Compare displayed weather data with a reliable weather source for the same location

Expected Results:

- Weather data should be reasonably accurate and current
- Temperature should be displayed in the correct unit (°F)
- Weather condition description should match current conditions

Test Data:

- Compare with weather.gov or other reliable weather source

Test Case 3.3: Weather API Fallback

Description: Verify system behavior when weather data cannot be retrieved
Test Steps:

1. Simulate a failure in the weather API connection
2. Log in with valid credentials
3. Navigate to the home page

Expected Results:

- System should display a graceful error message or placeholder
- Application should continue to function without weather data
- No application crash should occur

Test Data:

- Simulated API failure (may require testing code modification)

5. Test Schedule

- **Preparation Phase**: Set up test environment and test data (1 day)
- **Execution Phase**: Conduct all test cases (2-3 days)
- **Reporting Phase**: Document results and issues (1 day)
- **Fix & Retest Phase**: Address identified issues and retest (2-3 days)

6. Issue Reporting

All issues found during UAT should be documented with:

- Test case reference
- Detailed description of the issue
- Steps to reproduce
- Expected vs. actual results
- Screenshots or videos where applicable
- Environment details (browser, device, etc.)

7. Exit Criteria

UAT will be considered successful when:

- All critical and high-priority test cases pass
- No critical or high-severity defects remain open
- At least 90% of all test cases pass
- All reported defects have been addressed or scheduled for future releases

8. Observations and Results Documentation

For each test case, document:

- Pass/Fail status
- Actual behavior observed
- Any deviations from expected results
- User feedback and suggestions
- Performance observations