# Test Case Document and RMMM Plan for Sehat Pal

Syed Ukkashah (Backend Developer)
Ibrahim Johar (Frontend Developer)

7th May, 2025

## 1 Introduction

This document outlines the test cases for Sehat Pal, a web-based symptom checker developed as a university project. The test cases are designed to validate the functionality, performance, and security of the system, ensuring it meets the requirements specified in the Software Requirements Specification (SRS). The test cases cover user management, symptom checking, medical records management, emergency access, and information pages.

## 2 Test Cases

### 2.1 Test Case 1: User Signup and Email Verification

**Test Case ID:** TC-01

**Description:** Verify that a new user can sign up and receive an email verification link.

**Preconditions:** Firebase Authentication service is operational.

**Test Steps:**

1. Navigate to the signup page.
2. Enter a valid email and password (≥6 characters).
3. Submit the form.
4. Check the email inbox for a verification link.
5. Click the verification link.

**Expected Result:** User account is created, and email is verified.

**Actual Result:** To be filled during execution

**Status:** Pass/Fail

## 2.2 Test Case 2: User Login

**Test Case ID:** TC-02

**Description:** Verify that a registered user can log in using their credentials.

**Preconditions:** User account exists and email is verified.

**Test Steps:**
1. Navigate to the login page.
2. Enter the registered email and password.
3. Submit the form.

**Expected Result:** User is logged in and redirected to the homepage.

**Actual Result:** To be filled during execution

**Status:** Pass/Fail

## 2.3 Test Case 3: Symptom Checker Functionality

**Test Case ID:** TC-03

**Description:** Verify that the symptom checker provides a diagnosis based on user inputs.

**Preconditions:** User is logged in; Infermedica API has remaining trial calls.

**Test Steps:**
1. Navigate to the Symptom Checker page.
2. Enter symptoms (e.g., ”fever, cough”).
3. Answer follow-up questions (if any).
4. Submit the form.

**Expected Result:** A preliminary diagnosis is displayed with a disclaimer.

**Actual Result:** To be filled during execution

**Status:** Pass/Fail

## 2.4 Test Case 4: Medical Record Upload

**Test Case ID:** TC-04

**Description:** Verify that a user can upload a medical record (e.g., PDF, image).

**Preconditions:** User is logged in; Firebase Firestore has available storage.

**Test Steps:**
1. Navigate to the Medical Records page.
2. Select a file ($\leq$10 MB).
3. Submit the form.

**Expected Result:** File is uploaded and displayed in the user’s records.

**Actual Result:** To be filled during execution

**Status:** Pass/Fail

### 2.5 Test Case 5: Emergency Helpline Redirect

**Test Case ID:** TC-05

**Description:** Verify that the emergency helpline redirect works.

**Preconditions:** External helpline website is operational.

**Test Steps:**

1. Navigate to the Call an Ambulance page.
2. Click the redirect button.

**Expected Result:** Browser redirects to the external helpline website.

**Actual Result:** To be filled during execution

**Status:** Pass/Fail

### 2.6 Test Case 6: Information Pages Accessibility

**Test Case ID:** TC-06

**Description:** Verify that static pages (About Us, Contact Us, Help) are accessible.

**Preconditions:** Firebase Hosting is operational.

**Test Steps:**

1. Navigate to each static page via the menu.

**Expected Result:** Pages load with correct content.

**Actual Result:** To be filled during execution

**Status:** Pass/Fail

## 3 Test Log Report

| **Test Case ID** | **Description** | **Execution Date** | **Tester** | **Status** | **Notes** |
|---|---|---|---|---|---|
| TC-01 | User Signup | DD/MM/YYYY | Tester | Pass | Email verified. |
| TC-02 | User Login | DD/MM/YYYY | Tester | Pass | Login successful. |
| TC-03 | Symptom Checker | DD/MM/YYYY | Tester | Fail | API limit exceede |
| TC-04 | Medical Record Upload | DD/MM/YYYY | Tester | Pass | File uploaded. |
| TC-05 | Emergency Redirect | DD/MM/YYYY | Tester | Pass | Redirect successfu |
| TC-06 | Static Pages | DD/MM/YYYY | Tester | Pass | Pages loaded. |

# 4 Risk Management, Monitoring, and Mitigation (RMMM) Plan

## 4.1 Identified Risks and Mitigation Strategies

### 4.1.1 Risk 1: Infermedica API Limit Exceeded

**Description:** The trial version of the Infermedica API is limited to 2,000 calls or 60 days, whichever comes first.

**Impact:** Users cannot access the symptom checker if the limit is reached.

**Mitigation:**

- Monitor API usage regularly.
- Display a warning message when nearing the limit.
- Consider alternative APIs or extending the trial if possible.

**Monitoring:** Track API calls via Infermedica dashboard.

**Contingency:** Use mock data for demonstration if the API limit is reached.

### 4.1.2 Risk 2: Firebase Free-Tier Limitations

**Description:** Firebase free-tier imposes limits on storage, bandwidth, and database operations.

**Impact:** System performance may degrade or become unavailable if limits are exceeded.

**Mitigation:**

- Optimize database queries and storage usage.
- Monitor Firebase usage metrics.
- Upgrade to a paid plan if necessary (for future scalability).

**Monitoring:** Use Firebase console to track usage.

### 4.1.3 Risk 3: Network Connectivity Issues

**Description:** Users in remote areas of Pakistan may face intermittent internet connectivity.

**Impact:** System accessibility and performance may be affected.

**Mitigation:**

- Design a lightweight interface to reduce data usage.
- Provide offline functionality for critical features (if feasible).

**Monitoring:** Collect user feedback on accessibility issues.

### 4.1.4 Risk 4: Data Security Breach

**Description:** Unauthorized access to medical records or user credentials.

**Impact:** Compromised user privacy and trust.

**Mitigation:**

- Enforce Firebase security rules for data access.
- Use HTTPS for all communications.
- Educate users on strong password practices.

**Monitoring:** Regularly review security logs and user access patterns.

### 4.1.5 Risk 5: Misdiagnosis by Infermedica API

**Description:** The API may provide inaccurate preliminary diagnoses.

**Impact:** Users may make uninformed health decisions.

**Mitigation:**

- Include clear disclaimers about the non-professional nature of diagnoses.
- Encourage users to consult doctors for confirmation.

**Monitoring:** Collect user feedback on diagnosis accuracy.

### 4.1.6 Risk 6: Academic Deadline Constraints

**Description:** Limited time to complete the project for university evaluation.

**Impact:** Incomplete or rushed implementation.

**Mitigation:**

- Prioritize core features (e.g., symptom checker, user authentication).
- Use agile development practices for iterative progress.

**Monitoring:** Track project milestones and adjust scope as needed.

## 5 Conclusion

This Test Case Document and RMMM Plan ensure that Sehat Pal is thoroughly tested and risks are proactively managed. The test cases validate core functionalities, while the RMMM plan addresses potential risks to ensure the system's reliability and security within the project's constraints.