



# Contact List App – API Testing

Comprehensive Test Plan and Execution

Presented by: Ahmed & Aya & Mesk & Lina

Date: 25 July 2025



# Introduction: Contact List App API Testing

The **Contact List App** enables users to manage contacts with key features like login, signup, and CRUD operations.

## Purpose of Testing

Ensure APIs function correctly, securely, and efficiently to support core features.

## Scope

Focuses on User Authentication, Contact Management, and all related backend API endpoints.

## Objectives

- Verify APIs meet requirements
- Detect bugs early
- Safeguard data integrity and security

## Tools Used

- Postman for API testing
- Excel for test cases
- Trello for bug tracking
- Various reporting tools (Microsoft 365)



# Comprehensive API Test Plan

## ✓ Project Overview

- **App Name:** Contact List App (Web)
- **Testing Duration:** July 21 – July 25, 2025
- **Tools Used:** Postman, Excel, Trello
- **Teams:** Contacts API, Users API

## 🎯 Testing Objectives

- Validate CRUD & Authentication
- Ensure error handling, security, data integrity
- Follow Scrum methodology (5-day sprint)

## 🔍 In-Scope API Endpoints

- **Contacts API:** POST, GET (List & by ID), PUT, PATCH, DELETE
- **Users API:** Register, Login, Logout, Get Profile, Update, Delete

## 🚫 Out-of-Scope

- Mobile Testing
- UI/UX Testing & Load/Stress Testing
- Third-party Integrations

## 📋 Test Types

- Functional Testing
- Security Testing
- Basic Performance Testing
- Automated Assertions (Postman Scripts)

## 📅 Schedule

Phase	Date
Planning & Analysis	Jul 21
Test Case Design	Jul 22
Postman Setup	Jul 23
Execution & Bug Logging	Jul 24
Review & Demo	Jul 25

## ⚠️ Key Risks & Mitigation

- API changes → Daily sync with docs
- Token expiry → Pre-request scripts
- Env. issues → Shared files & review

## 👥 Team Roles

- **QA Lead:** Planning, Review
- **QA Engineers:** Test Design, Execution, Bug Reporting
- **Scrum Master:** Coordination

## 📁 Deliverables

- Test Plan & Excel Test Cases
- Postman Collections & Environment
- Bug Reports & RTM
- Test Summary Report & Final Presentation



# UI to API Mapping in the Contact List App

In the Contact List App, every user action on the frontend (UI) corresponds to a backend API call.



## User APIs

- **Login:** POST /api/login
- **Sign Up:** POST /api/signup
- **View Profile:** GET /api/user/me
- **Edit Profile:** PATCH /api/user/me
- **Delete Account:** DELETE /api/user/me
- **Log Out:** POST /api/users/logout

## Contact APIs

- **Add Contact:** POST /api/contacts
- **View All:** GET /api/contacts
- **View Details:** GET /api/contacts/:id
- **Edit Contact:** PUT/PATCH /api/contacts/:id
- **Delete Contact:** DELETE /api/contacts/:id

This tight UI-API mapping ensures a smooth, dynamic, and testable user experience.



# Excel Test Case Document

## Project Summary

- **Project Name:** Contact List App
- **Release Number:** 1
- **Delivery Date:** 23/07/2025
- **Environment:** Windows 11/10
- **Browser:** Google Chrome
- **API Base URL:** Contact List App
- **Test Type:** Manual + Collection Run (Automation)
- **Tools Used:** Postman, Trello, Excel

## QA Team

Aya, Mesk, Lina, Ahmed

## Test Case Statistics

Metric	Count
✔ Total Test Cases	172
✔ Passed Test Cases	144
✖ Failed Test Cases	28

## Test Case Format (in Excel)

Each test case includes:

- Priority
- Summary
- Preconditions
- Test Data
- Steps to Reproduce
- Expected Result

**Note:** Test cases cover **Contacts API** and **Users API**, including: Functional, Negative, Security, Response Time, Boundary scenarios.





# Postman Collections, Environment Variables, Assertions & Bug Log

We used **Postman** to automate and validate the API functionality for both **Contacts** and **Users** features of the Contact List App.

## Contacts API

### ✓ Postman Collection

- Created a collection covering all CRUD operations.

### 🌐 Environment Variables

- `{{base_url}}`: App base URL
- `{{token}}`: Authorization token
- `{{contact_id}}`: Dynamic chaining

### 📋 Assertions

- Validated status codes (200, 201, 400, 401, 404)
- Response body content (e.g., message, contact\_id)
- Error messages on invalid input

## Bug Log (Contacts)

- **20 bugs found:**  
**Example:** 503 Service Unavailable for PUT/DELETE with invalid IDs or tokens.

## Users API

### ✓ Postman Collection

- Covered user management APIs (Sign Up, Login, Profile, etc.).

### 🌐 Environment Variables

- `{{token}}`: After login
- `{{user_email}}`, `{{password}}`: For chaining login/signup

### 📋 Assertions

- Successful login returns a token
- Invalid login returns 401 Unauthorized
- Duplicate email on signup returns 400 Bad Request
- Password strength validation errors

## Bug Log (Users)

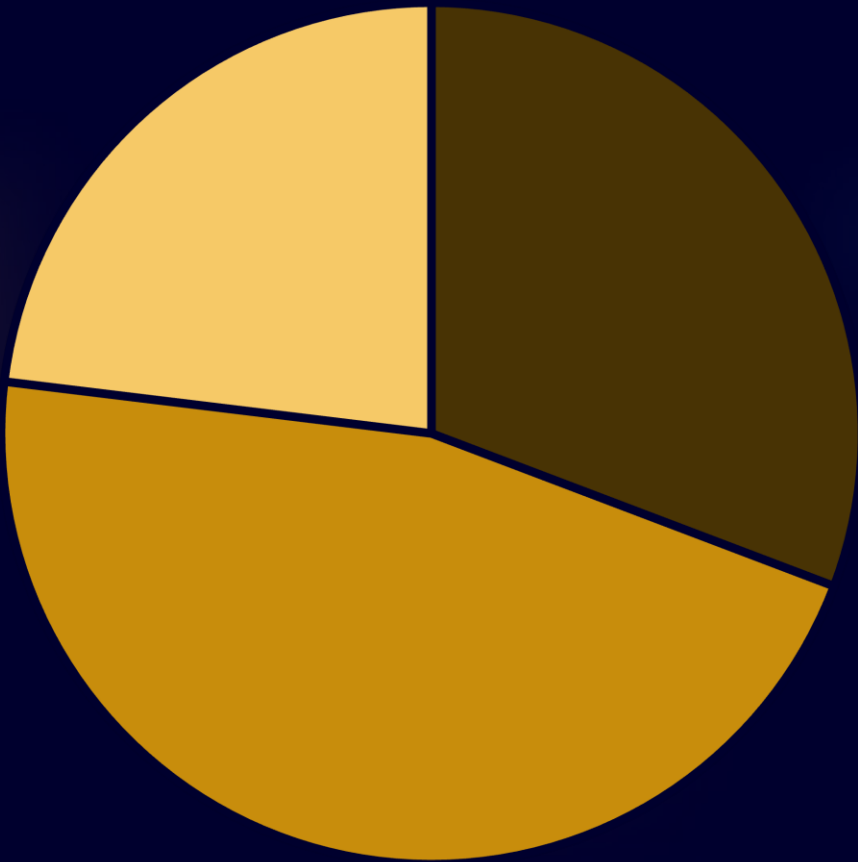
- **8 bugs found:**  
**Example:** User update API returns 200 even when no fields are changed (expected 204).

Our Postman collections automated end-to-end API testing, ensured robust validation, and identified real bugs in both Contact and User flows.



# Bug Report: Key Findings

Total Bugs Reported: 28



 High

 Medium

 Low

## Top Critical Bugs

### SQL Injection (Bug #022)

**Risk:** High security vulnerability.

- **Actual:** 500 Internal Server Error
- **Expected:** 400 Bad Request

### XSS Attack (Bug #023)

**Risk:** Executable script via contact ID.

- **Actual:** 500 Error
- **Expected:** 400 Bad Request

### Duplicate Email (Bug #001, #012)

**Risk:** Data integrity failure.

- **Actual:** 201 Created / 200 OK
- **Expected:** 409 Conflict

## Tools & Process

- **Tools Used:** Postman, Trello, Screenshot for evidence
- **Workflow:** Bug logged in Trello → Assigned to developers → Retested after fixes



# Requirements Traceability Matrix (RTM)

## What is RTM?

The **Requirements Traceability Matrix** links each requirement to its corresponding test cases, ensuring full coverage.

## Purpose of RTM

- Ensure every requirement is tested
- Track progress and identify gaps
- Facilitate impact analysis for changes
- Improve communication among stakeholders

## RTM Structure Example

Requirement ID	Requirement Name	Requested By:	Scope Deliverable or Feature	Scope ID	User Story or Work Package	Assigned To	Test Case	Current Status	Bug ID
Req - 02 / Contacts - 2	GET / Get Contact List	Product Owner	Contact Retrieval	Scope - 1.2	As a user, I want to retrieve all contacts to view my complete contact list	Aya Abu Ali	TC - 032 to TC - 039	Failed	Bug - 007
Req - 03 / Contacts - 3	GET / Get Contact	Product Owner	Contact Details	Scope - 1.3	As a user, I want to fetch a specific contact's details by ID for focused viewing	Aya Abu Ali	TC - 040 to TC - 044	Failed	Bug - 008
Req - 09 / Users - 3	PATCH / Update User	Product Owner	Profile Update	Scope - 2.3	As a user, I want to update my profile to keep information accurate	Mesk Alrayyes	TC - 136 to TC - 145	Passed	-

*Note: RTM links all functional, security, and performance requirements with test cases.*

## Benefits

- Guarantees comprehensive test coverage
- Identifies untested requirements early
- Supports regression testing and impact analysis
- Provides a clear, auditable document

## RTM in Our Project

- Created based on project specs and user stories
- Mapped test cases from Excel to requirements
- Regularly updated during test execution
- Used to generate summary reports and highlight progress





# Test Summary Report

## Overall Test Execution

- Total Test Cases Executed: 172
- Passed: 144 (84%)
- Failed: 28 (16%)
- Pending/Blocked: 0

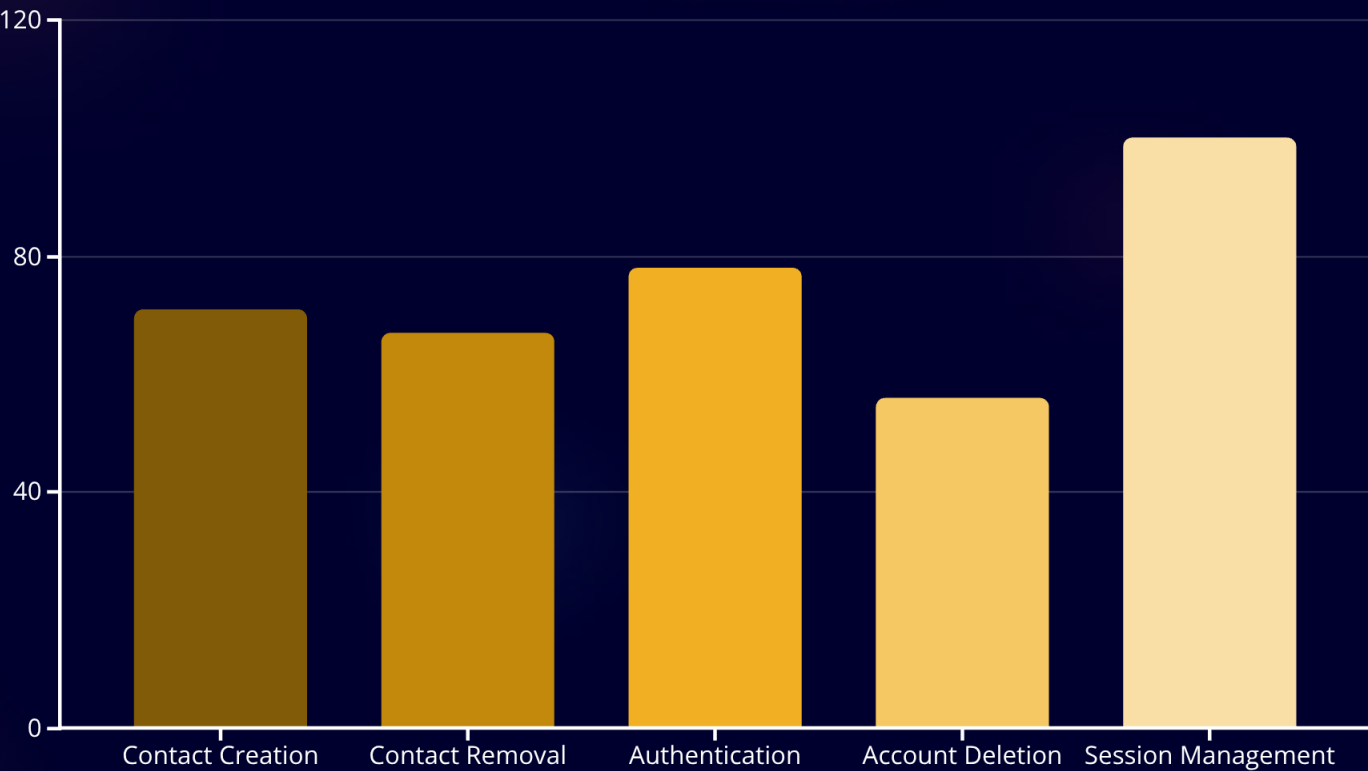
## Key Functional Areas & Defects

Functionality	Pass Rate	Defects	Priority
Contact Creation	81%	6	High
Contact Removal	67%	5	High
Authentication	89%	1	Critical
Account Deletion	56%	4	Medium
Session Management	100%	0	High

## Critical Findings

- Contact Creation:** 6 defects (duplicate emails, invalid inputs). Pass rate below 81%, indicating high risk.
- Authentication:** Incorrect error codes returned (401 vs. 400). Critical priority due to security impact.

## Visualizations



# Closing & Acknowledgments

Thank You for Your Attention!

On behalf of our entire team, I extend our sincere gratitude for your time and engagement throughout this presentation.

We would also like to acknowledge and thank the following team members for their outstanding contributions:

- Ahmed
- Lina
- Aya
- Mesk

Their dedication and hard work were essential to the success of this API testing project.

We now invite your **questions, comments, or feedback** to further enhance our work.

