User Acceptance Testing (UAT) & Quality Assurance Plan

Project Name: [Insert Project Title]

Developer/Tester: [Your Name]

Date of Testing: [YYYY-MM-DD]

Project Version/Release: [e.g., v1.0 - Production Candidate]

---

1. Testing Environment Checklist

Before beginning, confirm the environments where testing is being conducted to ensure consistency.

- **Operating Systems Tested:** [e.g., macOS, Windows 11, iOS, Android]
- **Browsers Tested:** [e.g., Chrome, Safari, Firefox]
- **Device Viewports:** * Desktop (1920x1080)
    - Tablet (768x1024)
    - Mobile (390x844)

---

2. Testing Protocol & Evidence

- **Goal:** Verify that all Functional and Technical requirements outlined in the PRD and TDD have been met.
- **Screenshot Protocol:** Every core functionality test **must** be accompanied by a screenshot proving the expected result. In this Markdown document, embed screenshots directly in the "Proof" column using the syntax: ![Description](./path-to-image.png).
- **Status Codes:** * **PASS:** Feature works exactly as expected.
    - **FAIL:** Feature is broken or produces an error.
    - **FLAG:** Feature works, but UI/UX feels clunky or needs refinement.

---

3. Test Cases

Phase 1: Authentication & Security

ID	Feature	Steps to Execute	Expected Result	Pass/Fail	Screenshot Proof	Notes
AUTH-01	User Registration	Submit signup form with valid email, name, and strong password.	DB creates user, JWT generated, redirected to Dashboard.	[ ]	[Insert Image]	
AUTH-02	Invalid Login	Submit login form with an unregistered email or wrong password.	Access denied. UI displays "Invalid credentials" error toast.	[ ]	[Insert Image]	
AUTH-03	Protected Routes	Manually type a protected URL (e.g., /dashboard) without being logged in.	App intercepts request and redirects user back to /login.	[ ]	[Insert Image]	

Export to Sheets

Phase 2: Core CRUD Functionality (API & Database)

ID	Feature	Steps to Execute	Expected Result	Pass/Fail	Screenshot Proof	Notes
CRUD-01	Create Record	Fill out the main submission form and click 'Save'.	UI shows loading state, success toast appears, form resets.	[ ]	[Insert Image]	
CRUD-02	Read/Display	Navigate to the main data table or dashboard.	Newly created record appears correctly formatted in the list.	[ ]	[Insert Image]	
CRUD-03	Update Record	Click 'Edit' on an existing record, change a value, and save.	Value updates immediately in the UI and persists on page refresh.	[ ]	[Insert Image]	

ID	Feature	Steps to Execute	Expected Result	Pass/Fail	Screenshot Proof	Notes
CRUD-04	Delete Record	Click 'Delete' on a record and confirm the modal prompt.	Record is removed from the UI and deleted from MongoDB.	[ ]	[Insert Image]	

Export to Sheets

Phase 3: Form Validation & Error Handling

ID	Feature	Steps to Execute	Expected Result	Pass/Fail	Screenshot Proof	Notes
ERR-01	Missing Fields	Attempt to submit a form with required fields left blank.	Frontend validation blocks submission; missing fields highlighted red.	[ ]	[Insert Image]	
ERR-02	API Failure	Disconnect internet or force a 500 error from the server.	UI handles gracefully (e.g., "Something went wrong, try again").	[ ]	[Insert Image]	
ERR-03	404 Page	Navigate to a random, non-existent route (e.g., /random-url).	Custom 404 "Page Not Found" component renders.	[ ]	[Insert Image]	

Export to Sheets

Phase 4: Third-Party Integrations

ID	Feature	Steps to Execute	Expected Result	Pass/Fail	Screenshot Proof	Notes
INT-01	Google Maps	Begin typing an address into the designated location field.	Autocomplete dropdown appears and populates city/zip correctly.	[ ]	[Insert Image]	

ID	Feature	Steps to Execute	Expected Result	Pass/Fail	Screenshot Proof	Notes
INT-02	reCAPTCHA	Submit a public-facing form (like a contact or lead form).	Submission succeeds only if reCAPTCHA token is generated and verified.	[ ]	[Insert Image]	

Export to Sheets

Phase 5: UI/UX & Responsiveness (Tailwind)

ID	Feature	Steps to Execute	Expected Result	Pass/Fail	Screenshot Proof	Notes
UI-01	Mobile Nav	Open app on a mobile viewport (390px) and click the menu icon.	Hamburger menu expands smoothly, all links are clickable.	[ ]	[Insert Image]	
UI-02	Responsive Data	View a large data table on a mobile viewport.	Table implements horizontal scroll or stacks data cleanly into cards.	[ ]	[Insert Image]	

Export to Sheets

---

4. Final Sign-Off

By signing below, the developer and client agree that all functional requirements have been tested, proven via screenshot evidence, and are approved for production deployment.

Developer Signature: ___________________________ **Date:** ____________

Client Signature: ______________________________ **Date:** ____________