SB Works

Freelancing Platform

SDLC Phase 5

TESTING & QUALITY ASSURANCE REPORT

Project Name	SB Works — Online Freelancing Platform
Phase	Phase 5: Testing and Quality Assurance
Prepared By	I. Sai Ganesh — Lead Developer / Tester
Date	February 2024

1. Introduction to Testing

1.1 Why Testing Matters

Testing is Phase 5 of the SDLC. After the code is written in Phase 4, we must verify that everything works correctly — not just the happy path (when everything goes right), but also edge cases and error conditions (when things go wrong). A piece of software that works correctly 90% of the time is not acceptable for a production application.

For SB Works, testing was especially important because the platform involves multiple user roles interacting with shared data. A bug in the proposal system, for example, could cause a freelancer to see another user's private messages, or allow a project to be deleted while a freelancer is actively working on it. These kinds of bugs would destroy user trust.

> ■ Testing Approach: Since this is a solo development project, all testing was done manually using a combination of Postman (for API testing), browser developer tools (for frontend debugging), and end-to-end workflow testing (creating real users and running through complete scenarios from login to project completion).

1.2 Types of Testing Performed

Test Type	What Was Tested	Tool Used
Unit Testing	Individual controller functions and utility functions	Manual + console.log debugging
API Testing	Every endpoint: correct responses, error cases, auth checks	Postman REST Client
Integration Testing	Backend + frontend working together end-to-end	Browser + running server
Authentication Testing	OTP flow, JWT expiry, cookie handling, role guards	Postman + Browser
Real-time Testing	Chat messages, notifications via Socket.io	Two browser windows
Role-based Testing	That each role can only access its permitted routes	Three test accounts
Error Handling Testing	Correct error messages for bad requests	Postman with invalid data
File Upload Testing	Work submission with and without file attachment	Browser file picker
Windows Environment	Full setup from scratch on Windows 10	Clean Windows PC

2. API Test Cases and Results

2.1 Authentication API Tests

Test Case	Input	Expected Result	Actual Result	Pass?
Send OTP with valid phone	+919876543210	200 OK, OTP in console	200 OK, OTP printed	PASS
Send OTP with empty phone	(empty)	400 Bad Request	400 with Joi error	PASS
Verify correct OTP	847291	200 OK, cookies set	200, JWT cookies set	PASS
Verify wrong OTP	000000	400 Invalid OTP	400 "Invalid OTP"	PASS
Verify expired OTP	after 90s	400 OTP expired	400 "OTP has expired"	PASS
Access protected route without token	(no cookie)	401 Unauthorized	401 error returned	PASS
Access owner route as freelancer	freelancer cookie	403 Forbidden	403 Access denied	PASS
Admin login with seeded credentials	+10000000000 / 123456	200 OK, ADMIN role	200, admin dashboard	PASS

2.2 Project API Tests

Test Case	Method	Expected	Result	Pass?
Create project with all fields	POST /project/add	201 Created	201 project object	PASS
Create project with short title	POST /project/add	400 Validation error	400 "min 10 chars"	PASS
List open projects as freelancer	GET /project/list	200 with array	200 projects array	PASS
Get owner's own projects	GET /owner-projects	200 with projects	200 filtered list	PASS
Delete project with freelancer assigned	DELETE /project/:id	400 Cannot delete	400 error returned	PASS
Toggle project status OPEN to CLOSED	PATCH /project/:id	200 CLOSED status	200 status changed	PASS
Get project with populated proposals	GET /project/:id	200 with proposals	200 with nested data	PASS

2.3 Proposal API Tests

Test Case	Scenario	Expected	Result	Pass?

Submit proposal as freelancer	Valid project, valid data	201 Created	201 proposal saved	PASS
Submit duplicate proposal	Same project, same user	400 Duplicate error	400 "Already submitted"	PASS
Submit proposal as owner	Wrong role	403 Forbidden	403 Access denied	PASS
Accept proposal as project owner	status=2	200 Hired, closed project	200 all correct	PASS
Auto-reject other proposals on accept	Accept one of three	Other two status=0	Verified in DB	PASS
Reject proposal	status=0	200 Rejected	200 status updated	PASS

3. End-to-End Workflow Tests

3.1 Complete Project Lifecycle Test

The most important test: running through the entire platform experience as if three real users were using it simultaneously.

Step	Action	User	Result
1	Register with phone +919000000001	Rajesh (Client)	OTP appeared in terminal PASS
2	Complete profile as OWNER role	Rajesh (Client)	Redirected to owner dashboard PASS
3	Admin approves Rajesh's account	Admin	Status changed to Approved PASS
4	Post project: "React E-Commerce Website"	Rajesh (Client)	Project appears in OPEN status PASS
5	Register with phone +919000000002	Rahul (Freelancer)	OTP in terminal PASS
6	Complete profile as FREELANCER role	Rahul (Freelancer)	Redirected to freelancer dashboard PASS
7	Admin approves Rahul's account	Admin	Status Approved PASS
8	Browse projects, see Rajesh's project	Rahul (Freelancer)	Project visible in browse PASS
9	Submit proposal: $1200, 14 days	Rahul (Freelancer)	Proposal saved, Rajesh notified PASS
10	Rajesh sees new proposal with bell notification	Rajesh (Client)	Real-time notification received PASS
11	Rajesh accepts Rahul's proposal	Rajesh (Client)	Project closed, Rahul hired, notified PASS
12	Rahul sees "Proposal Accepted!" notification	Rahul (Freelancer)	Real-time notification received PASS
13	Rajesh and Rahul exchange 5 chat messages	Both	Messages delivered instantly PASS
14	Rahul submits work with zip file attached	Rahul (Freelancer)	File saved in /uploads/, Rajesh notified PASS
15	Rajesh downloads and reviews the file	Rajesh (Client)	File download works PASS
16	Rajesh requests revision with feedback	Rajesh (Client)	Rahul notified, can resubmit PASS
17	Rahul resubmits improved work	Rahul (Freelancer)	New submission saved PASS
18	Rajesh approves the work	Rajesh (Client)	Project marked complete PASS
19	Rajesh leaves 5-star review	Rajesh (Client)	Rating saved, Rahul profile updated PASS
20	Rahul checks his profile rating	Rahul (Freelancer)	Shows 5.0 average, 1 review PASS

4. Bug Report and Fixes

4.1 Bugs Found During Testing

The following bugs were discovered during the testing phase and subsequently fixed before the software was deployed:

Bug ID	Description	Severity	Fix Applied
BUG-001	Backend crashed on startup if Twilio credentials missing from .env	Critical	Added dev mode that skips Twilio and prints OTP to console
BUG-002	Phone validation regex only accepted US format (+1XXXXXXXXXX)	High	Changed regex to accept 7-20 chars, any international format
BUG-003	Users with status=1 (pending) got 403 Forbidden after login	High	Fixed middleware to allow status 1 and 2; only block status 0 (rejected)
BUG-004	Proposals did not store project reference, breaking submission linking	High	Added project field to ProposalSchema, saved in controller
BUG-005	Frontend pages had hardcoded http://localhost:5000 URLs	Medium	Removed hardcoded URLs, used Vite proxy for all /api calls
BUG-006	window.confirm() called without window prefix in strict mode	Low	Changed all confirm() calls to window.confirm()
BUG-007	FreelancerProjects page used wrong field for project ID in submissions	Medium	Fixed field mapping to use correct _id from proposal.project
BUG-008	File download in OwnerProject used wrong base URL	Medium	Updated download URL to use Vite proxy path
BUG-009	MongoDB Atlas connection string had port number (:27017)	High	Removed port from SRV connection string (not allowed)
BUG-010	Atlas username contained @ character breaking connection URL	High	Encoded @ as %40 in URL, or created new user with simple name

4.2 Testing Summary

Category	Tests Planned	Tests Run	Tests Passed	Pass Rate
Authentication APIs	8	8	8	100%
Project APIs	7	7	7	100%
Proposal APIs	6	6	6	100%
Submission APIs	5	5	5	100%
Chat & Socket.io	4	4	4	100%

Admin APIs	5	5	5	100%
End-to-End Workflow	20 steps	20 steps	20 steps	100%
Role-based Access	8	8	8	100%

■ Phase 5 Outcome: All 10 bugs discovered during testing were fixed before deployment. The complete 20-step end-to-end workflow test passed successfully. All APIs returned correct responses for both valid and invalid inputs. The system was declared ready for Phase 6: Deployment.