QuickHire

"Job Browsing & Application"

FNA Test Plan

Document Author:	Document Version:	Revised Date:	Approved B
QA Engineer	1.0	Feb 28th, 2026	Test Lead

Table of Contents

1. Introduction

The Test Plan has been created to facilitate communication within the team members. This document describes approaches and methodologies that will apply to perform various types of testing on the QuickHire Job Portal. It includes the objectives, scope, test responsibilities, entry and exit criteria, schedule major milestones, approaches, risks, and acronyms. This document has clearly identified what the test deliverables will be, and what is deemed in and out of scope.

Used Framework: React (Front-end), Node.js / Express (Back-end), deployed on Railway cloud platform.

Website URL: https://reliable-rebirth-production-38e7.up.railway.app/

2. Test Items

2.1. Feature to be Tested

Browse Jobs

- Ability to navigate to the Browse Jobs page by clicking the 'Browse Jobs' nav link.
- Ability to view all available job listings with title, company, location, job type, and salary.
- Ability to filter jobs by category (Engineering, Design, Marketing, Product, Data Science, DevOps).
- Job cards should display accurate and complete information.
- Clicking a job card should navigate to the Job Details page.
- Empty state message should appear when no jobs match the selected filter.

Job Application

- Ability to click 'Apply Now' button on the Job Details page to open the application form.
- Application form should contain: Full Name, Email, Resume Link (URL), and Cover Letter fields.
- Validation for empty required fields — error messages should appear.
- Validation for invalid email format (e.g., 'testuser' without @ and domain).
- Validation for invalid URL in Resume Link field.
- Successful form submission should display a success/confirmation message.
- Boundary value testing for Cover Letter field (very long input).

Admin Panel – Post a Job

- Ability to access the Admin Panel via /admin URL.
- Post a Job form should contain: Job Title, Company, Location, Salary, Category (dropdown), and Description fields.
- Category dropdown should contain all 6 categories: Engineering, Design, Marketing, Product, Data Science, DevOps.
- Validation for empty required fields on Post a Job form.
- Validation for invalid data type in Salary field (e.g., alphabets).
- Successfully posted job should appear on the Browse Jobs (/jobs) page.

Homepage

- Verify all Navigation Bar links: Home, Browse Jobs, Admin Panel, Post a Job.

- Verify 'Browse All Jobs' and 'Start Browsing Jobs' CTA buttons navigate to /jobs.
- Verify Browse by Category buttons (6 categories) filter jobs correctly.
- Verify 'How It Works' section shows all 3 steps with correct content.
- Verify statistics counters: 500+ Active Jobs, 120+ Companies, 2,000+ Candidates Hired.
- Verify Footer Quick Links and Categories navigate correctly.

2.2. Feature NOT to be Tested

- Backend API endpoints and direct server responses.
- Database direct access and queries.
- Server-side authentication mechanisms (beyond UI behavior).
- Third-party integrations (payment gateways, email services).
- Cloud infrastructure (Railway.app deployment pipeline).
- Browser developer console internals.

3. Quality Objectives

The objective of this Test Plan is to ensure reliability, system assurance, and optimum performance of the QuickHire Job Portal. This test plan focuses on verifying that application features match end-user expectations. The testing covers:

- Ensuring all functional features work as per requirements (job browsing, applying, posting).
- Verifying user interface consistency, alignment, and responsive behavior across devices.
- Validating form input fields reject invalid data gracefully with meaningful error messages.
- Confirming cross-browser compatibility on Chrome, Firefox, and Edge.
- Identifying usability and security gaps (e.g., unauthenticated admin access).
- Achieving 100% test case execution with documented pass/fail results.

4. Test Methodologies

This project will follow agile methodology with work updated based on iterative cycles. Work is distributed and followed according to requirement-based strategy specified in this test plan. Each feature will be tested separately and in combination after each phase. Reports will be delivered according to the schedule. Test Cases are created during exploratory testing.

4.1. Test Levels

In Scope:

- Functional Testing (Black-box testing geared to functional requirements of the application).
- Integration Testing (to verify combined functionality after integration of all modules).
- UI/UX Testing (to verify visual elements, layout, alignment, and user experience).
- Cross-browser Testing (to check whether the system runs correctly in every major browser).
- Negative Testing (to test system behavior with invalid and unexpected inputs).
- Boundary Value Testing (testing at the edge of allowed input ranges).
- Responsive/Mobile Testing (to verify layout adapts correctly on mobile and tablet viewports).
- UAT (from a tester's perspective).
- Regression Testing (for Homepage, Browse Jobs, Apply flow).

Out of Scope:

- Database Testing.
- API / Backend Testing.
- Performance / Load Testing (covered in future sprint).
- Security penetration testing.

4.2. Test Strategy

Test Case Preparation & Execution:

- QA will prepare test cases covering all scenarios for requirements.
- Peer review will be conducted periodically by QA Lead.

• Author will incorporate feedback/improvements and resubmit for approval.

• QA will execute test cases on the test site based on designed scenarios, test cases, and test data.

• Test results (Actual vs Expected → Pass/Fail) will be updated in the test case document.

Defect Logging and Reporting:

• QA will log defects/bugs in the project management tool (Jira).

• QA will notify respective developer immediately after logging the defect.

Retesting:

• QA will retest fixed bugs after resolution by the developer.

• Bug/defect status will be updated accordingly.

• Regression testing will be performed if required after fixes.

Deployment / Delivery:

• Once all logged bugs are fixed and no further bugs are found, the report will be deployed to the test site by the PM.

• If required, the report will be delivered along with sample output via email to respective team members.

• QA will hand over hard copy of delivery documents to the respective developer.

5. Roles and Responsibilities

Project Manager	pm@quickhire.com	1. Primary contact for dev and QA team. 2. Participate in Test Scenarios review. 3. Responsible for project schedule and overall success. 4. Verify sign-off on Requirements, Test Scenarios, Test Cases.
QA Lead	qalead@quickhire.com	1. Participate in project plan creation/update. 2. Plan and organize test process for each release. 3. Coordinate with QA engineers on issues. 4. Report progress to Project Manager. 5. Monitor quality metrics throughout testing phases.
QA Engineer	qa@quickhire.com	1. Understand feature requirements. 2. Write and execute test cases. 3. Prepare RTM (Requirements Traceability Matrix). 4. Review test cases with PM and QA Lead. 5. Prepare test data. 6. Log defects and track status. 7. Perform retesting and regression testing. 8. Attend bug review meetings. 9. Coordinate with QA Lead on issues.

6. Entry and Exit Criteria

6.1. Entry Criteria

- QA resources have completely understood the feature requirements.
- QA resources have sound knowledge of the QuickHire application functionality.
- Test Scenarios and Test Cases are approved.
- All necessary documentation, design, and requirements information is available.
- Application Smoke Test completed successfully.
- Test environment is set up and accessible.

6.2. Exit Criteria

- All planned test cases have been executed.
- A defined level of requirements coverage has been achieved (100%).
- All Severity 1 and Severity 2 defects are resolved.
- No high-priority defects remain open.
- UAT Test Evidence collected.
- Test Closure Memo completed and signed off by QA Lead and PM.

7. Suspension Criteria and Resumption Requirement

If the start of testing is delayed due to design or development tasks, testing cannot be extended beyond the UAT scheduled start date.	Buffer time has been added to the schedule for contingencies.
Server issues or application not accessible.	Server should be checked and renewed/restarted from time to time. Railway.app environment to be monitored.
Significant number of high and critical issues still open at or near End-to-End testing phase.	Analysis of open issues with dev and product support to determine which can be deferred.
New feature or functionality added during the testing phase.	Extend the testing cycle or remove test requirements/expectations accordingly.

8. Resource and Environment Needs

8.1. Resources

Description	Tool
Web server accessible via Railway.app cloud deployment	Railway.app Server
Stable internet connection, minimum 5 Mb/s	Network
Windows 10 / macOS, RAM 8GB, modern CPU	Computer

8.2. Testing Tools

Process	Tool
Test Case Creation	Microsoft Excel
Test Case Tracking	Microsoft Excel
Test Case Execution	Manual Testing
Test Case Management	Microsoft Excel
Defect Management	Jira
Test Reporting	Microsoft Excel & Jira

Mind Mapping	XMind / draw.io

8.3. Configuration Management

Version control and configuration management will be handled via **GitHub**.

8.4. Test Environment

- • Windows 10: Chrome (latest), Firefox (latest), Edge (latest), Brave.
- • macOS: Chrome (latest), Firefox (latest), Safari (latest).
- • Mobile (DevTools): iOS Safari simulation, Android Chrome simulation.
- • Application URL: https://reliable-rebirth-production-38e7.up.railway.app/

9. Test Schedule

Test Planning	28/02	28/02	QA Engineer	3 hours
Review Requirements & Explore Application	28/02	28/02	QA Engineer	1 hour
Create Test Basis (Mind Map, Scenarios)	28/02	28/02	QA Engineer	1 hour
First Deploy to QA Test Environment	28/02	28/02	QA Engineer	30 min
Functional Testing – Iteration 1	28/02	01/03	QA Engineer	3 hours
UI/UX & Browser Compatibility Testing	01/03	01/03	QA Engineer	2 hours
Negative & Boundary Value Testing	01/03	01/03	QA Engineer	1 hour
System Testing	02/03	02/03	QA Engineer	2 hours
Regression Testing	02/03	02/03	QA Engineer	2 hours
UAT	02/03	03/03	QA Engineer	2 hours
Resolution of Final Defects & Final Build Testing	03/03	03/03	Dev Team	3 hours
Deploy to Staging Environment	04/03	04/03	PM	Future Work
Performance Testing	04/03	04/03	QA Lead	Future Sprint
Release to Production	05/03	05/03	PM	Future Work

10. Deliveries

Before Testing Phase
- Test Plan document.
- Test Cases document.
- Test Scenarios document.
- Mind Map.

During Testing
- Test Data.
- Test Traceability Matrix (RTM).
- Error logs and execution logs.
- Defect reports (Jira).

After Testing Cycle

- Test Results / Summary Report.

- Defect Report.

- Test Metrics Report.

- Recommendations document.

- Release notes.

11. Risk and Assumptions

Risks:

• **Risk 1:** Admin Panel (/admin) is publicly accessible without authentication — risk of unauthorized job postings.

• **Risk 2:** Application hosted on Railway.app free tier may have cold start delays affecting test timing.

• **Risk 3:** In older or non-updated browsers, the React application may fail to load correctly if JavaScript is disabled.

• **Risk 4:** Network instability may cause intermittent failures during form submission testing.

• **Risk 5:** No email service integration — application submission confirmation cannot be fully tested.

Assumptions:

• All test environments are available and stable for the duration of testing.

• The application will not undergo major feature changes mid-testing cycle.

• QA team has full access to the application URL and Admin Panel.

• Test data can be freely created (job postings, applications) without affecting production.

• Browser DevTools are available for responsive design testing.

• QA resources have adequate skill level to execute the defined test cases.

12. Terms / Acronyms

Term / Acronym	Definition
API	Application Programming Interface
PM	Project Manager
QA	Quality Assurance
RTM	Requirements Traceability Matrix
UAT	User Acceptance Testing
FNA	Functional & Non-functional Analysis
CTA	Call To Action (button/link prompting user action)
UI/UX	User Interface / User Experience
TC	Test Case
BVT	Boundary Value Testing

13. Approvals

Project Manager	Product Manager	28-Feb-2026	None
QA Lead	QA Lead	28-Feb-2026	None

— End of Test Plan —