

# Internship Report

## 1. Internship Details

- **Company Name:** QualiZeal
- **Internship Duration:** 1st September 2024 to 31st January 2025 (5 months)
- **Team Size:** 5 Members  
Bamla Varun Singh (22211A0538)  
Sowmya Gopiseti (22211A05A8)  
P. Ajay Kumar (22211A05P1)  
K. Harshith Sai (22211A05E0)  
T. Sathwik (22211A05U3)
- **Project Title:** Web Application for Testers to Solve Issues and Get Paid
- **Technology Stack:**
  - **Frontend:** React.js, TailwindCSS
  - **Backend:** Node.js, Express.js
  - **Database:** MySQL
  - **Authentication:** JWT
  - **Payment Integration:** PayPal API
  - **Hosting:** Vercel

## 2. Internship Overview

This full-stack web application is built to provide a platform similar to uTest, where testers can apply to test real-world software, solve issues, and get rewarded. The goal was to replicate a real QA workflow in a practical, simplified environment where testers can explore, apply, contribute, and earn for their testing efforts.

The flow begins with the admin posting a new project for testing. Interested testers can apply for the project through a linked Google Form. Once the admin reviews the applications, they can manually assign testers to specific tasks. Upon assignment, testers receive notifications (either via email or in-app). Testers then work on the assigned tasks by reviewing the product, identifying bugs or areas of improvement, and submitting a detailed report.

The work was executed in **four key phases**:

- **Phase 1:** Project Setup and Planning
- **Phase 2:** Frontend Development and Deployment

- **Phase 3:** Backend Development and Workflow Integration
- **Phase 4:** Final Integration, Testing, and Documentation

## Internship Phases

### Phase 1: Project Setup and Planning

**Duration:** September 1 – September 30, 2024

- Conducted requirement gathering sessions and finalized project scope.
- Designed user flow and testing structure inspired by platforms like uTest.
- Created UI/UX wireframes using Figma.
- Initialized the React frontend with basic routing and authentication pages.
- Deployed the initial frontend on Vercel.
- Integrated a Google Form to collect tester applications per project post.

### Phase 2: Frontend Development and Deployment

**Duration:** October 1 – October 31, 2024

- Built dashboard UI for testers to track their application status.
- Implemented responsive components for tasks and issue submissions.
- Created form structure similar to JIRA: Steps to Reproduce, Severity, Bug/Improvement tagging.
- Developed interactive views for task status and bug report history.
- Connected PayPal sandbox for mock testing payout integration.

### Phase 3: Backend Development and Workflow Integration

**Duration:** November 1 – November 30, 2024

- Built Node.js backend with Express to handle authentication and project management.
- Connected MySQL via Sequelize ORM for database storage.
- Developed role-based access layers for testers and future admins.
- Created endpoints for tester assignment, bug submission, and status tracking.
- Enabled email or in-app notifications to inform testers of assignment updates.

### Phase 4: Final Integration, Testing, and Documentation

**Duration:** December 1, 2024 – January 31, 2025

- Integrated PayPal payment logic to initiate payouts on valid bug submissions.
- Simulated full tester journey: application → task assignment → submission → approval → payment.
- Validated backend logic and tested edge cases in PayPal payout integration.
- Created detailed documentation on APIs, database schema, and flow diagrams.
- Conducted internal walkthroughs and gathered feedback.

## **Outcome:**

The following outcomes were achieved:

- A functional and polished frontend, deployed live via Vercel
- User authentication with JWT and registration/login flows
- Admin can post testing projects and collect tester applications via Google Form
- Manual tester assignment with email/in-app notifications
- Testers can submit structured issue reports (with reproduction steps, severity, etc.)
- Admin can review issues and mark them as valid (Bug or Improvement)
- PayPal integration is successfully implemented for issuing payments

## **Tools & Technologies Used**

- Frontend: React.js, TailwindCSS
- Backend: Node.js, Express.js
- Database: MySQL
- Authentication: JWT
- Payment Integration: PayPal API
- Hosting: Vercel