



Case Study for LL Project

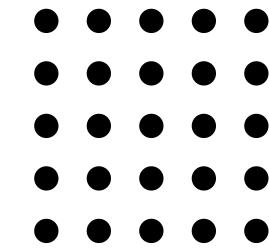
Loving Loyalty (LL) Project

Person:

Prabhat Chandak
Manpreet Bains



www.clanap.com



1. Introduction

Loving Loyalty (LL) is a POS (Point of Sale) system application designed for restaurants to efficiently manage their business operations. It provides features such as menu item management, order processing, multiple payment methods, delivery handling, and reporting tools. The application supports both eat-in and takeaway orders and integrates with various hardware components such as payment terminals and receipt printers. Additionally, LL includes a separate driver application to streamline delivery operations.



www.clanap.com

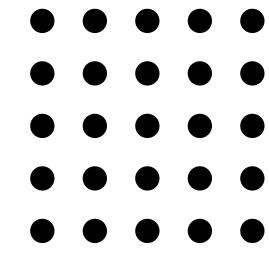


2. Project Overview

The LL project aims to create a robust and scalable POS system that enhances restaurant operations by integrating order management, payment handling, and reporting functionalities. The system supports:

- **Menu management:** Creating and managing product and menu items.
- **Order processing:** Supporting eat-in, takeaway, and delivery orders.
- **Payment handling:** Accepting payments via cash, card, voucher, and pay-later options.
- **Delivery management:** Assigning drivers and tracking deliveries.
- **Reporting & analytics:** Generating daily sales reports and end-of-day summaries.
- **Hardware integration:** Supporting built-in, Wi-Fi, and Bluetooth printers.



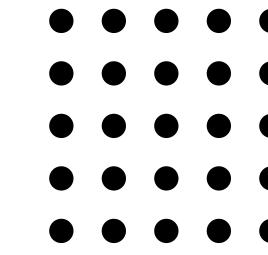


3. Challenges Before QA Involvement

Before a dedicated QA team was introduced to the LL project, several challenges impacted the overall functionality, stability, and usability of the system:

- **Frequent Production Bugs:** The application faced numerous undetected issues that led to order failures, incorrect payment processing, and missing order details.
- **Lack of Structured Testing:** No structured test plan, test cases, or regression testing were in place, leading to repeated issues with new feature releases.
- **Unstable Deployments:** Deployments often resulted in critical blocker issues, causing disruptions for restaurant owners and leading to customer dissatisfaction.
- **Inconsistent User Experience:** Many UI components were misaligned, non responsive, or not properly validated, affecting both desktop and mobile experiences.
- **Hardware Integration Failures:** Payment terminals, receipt printers, and Bluetooth/Wi-Fi device connections frequently failed due to lack of validation





3. Challenges Before QA Involvement

- **Data Inconsistencies:** Order tracking, payment records, and voucher redemptions often displayed incorrect data due to a lack of proper test coverage.
- **No Automated Testing:** Testing was entirely manual, resulting in slower releases and high chances of regression issues.
- **Delayed Issue Resolution:** Bugs were discovered by end-users in production, leading to higher support requests and business impact.
- **Unverified API Responses:** No validation of API responses resulted in incorrect data storage and processing, causing discrepancies in reports.
- **Security Concerns:** There was no proper validation for authentication and authorization flows, making the system vulnerable to security threats. With the introduction of a QA team, these challenges were systematically addressed, significantly improving product stability and reliability



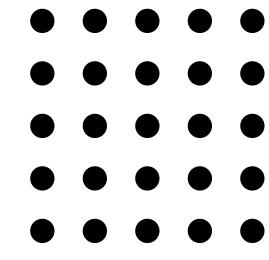


4. Objectives

- Develop a seamless POS system tailored for restaurant operations.
- Automate the order-to-payment workflow for efficiency.
- Provide robust reporting tools for sales and performance tracking.
- Enhance the user experience with an intuitive interface and multi-platform support.
- Establish a strong QA process to prevent major production issues.



www.clanap.com





5. Solution Implemented

Technology Stack

- **Automation & Testing:** Playwright Framework
- **Cloud & DevOps:** GitHub, CI/CD

Key Features

- **Role-based access control:** Supporting Cashier, Customer, and Driver roles.
- **Search & filtering:** Enabling easy discovery of restaurants and menu items.
- **Multi-payment support:** Accepting different payment methods.
- **Voucher system:** Generating and managing customer vouchers.
- **Order tracking:** Assigning orders to drivers and tracking their locations.
- **Table management:** Assigning tables to eat-in orders.
- **Hardware support:** Integrating with payment terminals and printers.
- **End-of-day processing:** Generating daily sales and order reports.





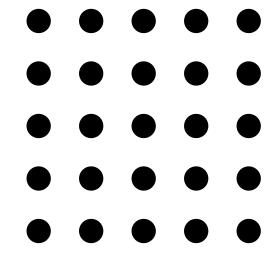
6. QA Engineer Contributions

As a QA Engineer, I actively contribute to ensuring the LL application functions smoothly by performing both manual and automation testing.

Manual Testing Activities

1. Perform **Ad-hoc, Functional, and Black Box Testing** on test and production environments.
2. Identify and log **2-3 bugs daily** and validate fixed bug tickets.
3. Report **over 700 bugs** and suggest **150+ enhancements** to improve the application.
4. Validate **750+ tickets** on GitHub and create **1000+ test cases** for different user roles.
5. Continuously update test cases as new features are developed.
6. Identify **100+ blocker issues** during deployments.
7. Explore and document the **BikeDesk** application to integrate relevant features into LL.



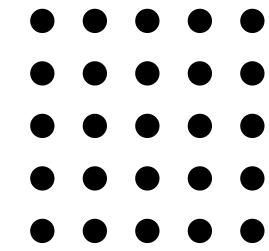


Automation Testing Activities

1. Designed and developed a **Playwright framework** using the **Page Object Model (POM)**.
2. Automated **900+ test scripts** for both desktop and mobile views.
3. Set up **Playwright automation** by cloning the LL repository and installing dependencies.
4. Created a **global function at the fixture level** to dismiss uncaught errors.
5. Handled complex scenarios by implementing functions for capturing **Order IDs** and validating API responses.
6. **Refactored scripts** as needed for bug fixes or new feature implementations under automation.



www.clanap.com

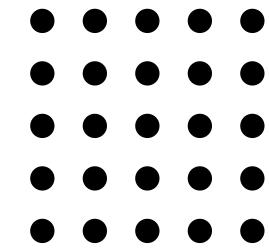


7. Challenges Faced

Handling dynamic UI elements like modals, popups, and dropdowns.

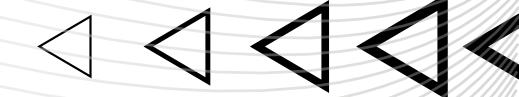
Automating validation of PDFs and QR codes. Managing API responses efficiently for real-time order updates. Ensuring high test coverage for continuously evolving features. Addressing blocker issues during deployments.

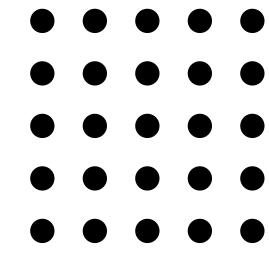




8. Results & Impact

- **30% improvement** in issue detection and resolution time.
- **50% reduction** in manual testing efforts due to automation.
- **Enhanced stability** of releases with early bug detection.
- **Seamless POS experience** for restaurant owners and staff.

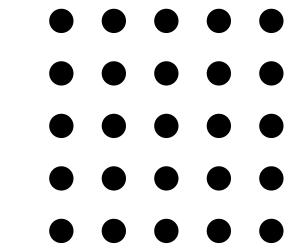




9. Future Scope

- Automate test cases for **Driver and Customer applications** to reduce manual effort and improve efficiency.
- Create comprehensive documentation for **implementing Bike Desk features** in the LL app.
- Break down the **Bike Desk feature implementation** into small, manageable tasks and track progress step by step.
- Continuously create and automate test cases for **new and upcoming features** in the LL app, prioritizing Cashier, Driver, and Customer roles.
- As new features are introduced, ensure thorough **test case creation and automation** to maintain product stability and quality.





10. Conclusion

The Loving Loyalty (LL) project successfully revolutionised POS systems for restaurants by integrating advanced features such as multi-payment support, order tracking, and automation. With continuous improvements, it remains a vital tool for businesses to manage operations efficiently.

