

Enhance Visual Experience and Usability of Ticketing Platform: Ticketing System Enhancement and New Function Implementation

Cheung Ka Yuen
Supervised By Dr. Desmond Yau Chat TSOI



OVERVIEW

In response to the rapid growth of the ticketing industry and the increasing demand for seamless event experiences, POPTICKET, a prominent Hong Kong-based ticketing platform, faces challenges that hinder the overall experience when using the control panel. These obstacles include a non-intuitive control panel interface, system bugs, and a lack of visual representation for critical event statistics.

This project aims to revamp the platform's data visualization, optimize system usability, and refine the user interface to boost customer satisfaction and enhance operational effectiveness.

OBJECTIVES



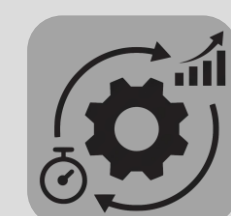
Enhance the user interface to deliver a clear and intuitive user experience



Enhance data visualization for improved analysis and user interaction



Resolve system bugs to enhance performance



Implement new functions to enhance operational efficiency

REVAMPED POPTICKET SYSTEM

Revamped Create and Edit Pages

The revamped pages provide a consistent and responsive user interface. The upgrade enhances user experience with dynamic content rendering and improved performance.



Sales Chart Component

The revamped sales chart component provides various types of data visualization. This allows users to analyze sales information effectively. With an intuitive date and chart selector, users can easily view different charts across various timeframes.

Smart JSON Editor

The revamped JSON Editor allows users to create and modify JSON data in a user-friendly interface, without the need for manual typing, streamlining the editing process. Reduce the rate of syntax error.

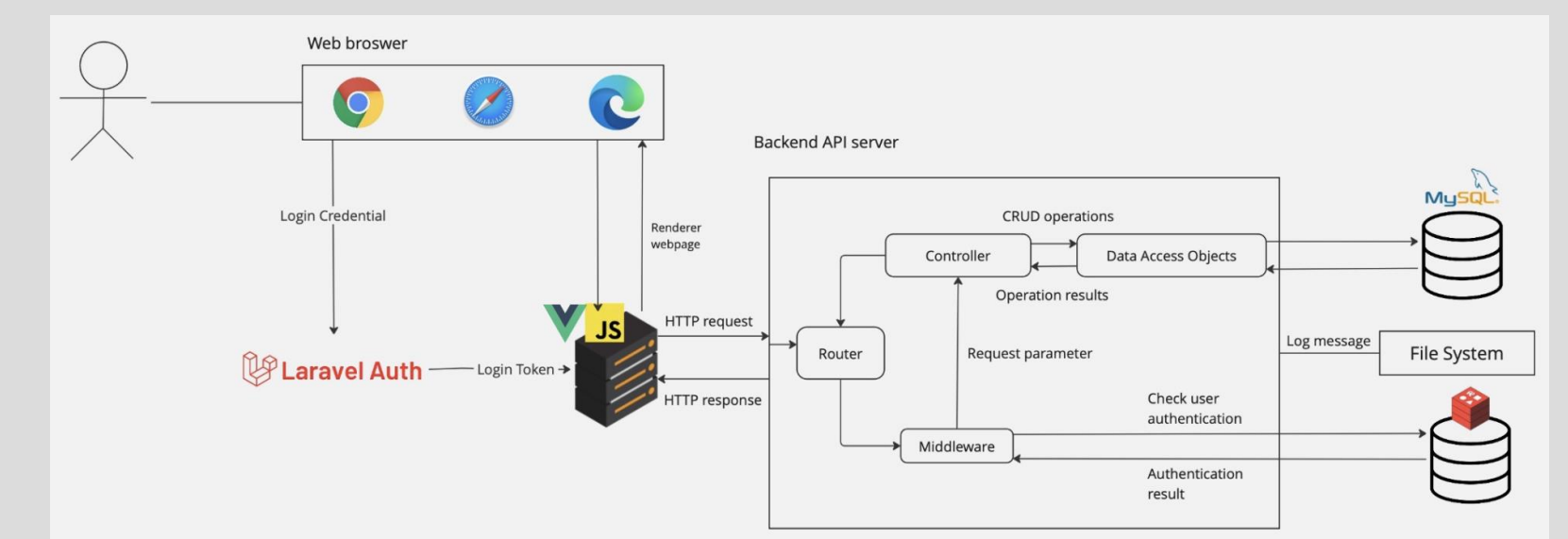
Ticket											
Ticket name		LIKE %	VIP Ticket			Search					
Select	id	NULL	event_id	event_map_id	ticket_name	ticket_category_id	ticket_category	ticket_description	ticket_price	original_price	venue
<input type="checkbox"/>	6813	NULL	1790	16993	[zh-TW]"100T"zh-CN]"100T"	21343	[zh-TW]"100T"zh-CN]"100T"	[zh-TW]"100T"zh-CN]"100T"	100	100	1
<input type="checkbox"/>	6812	NULL	1790	16992	[zh-TW]"Ticket1"zh-CN]"Ticket1"	21342	[zh-TW]"Ticket1"zh-CN]"Ticket1"	[zh-TW]"Ticket1"zh-CN]"Ticket1"	1000	1000	1
<input type="checkbox"/>	6811	NULL	1792	16991	[zh-TW]"Ticket1"zh-CN]"Ticket1"	21341	[zh-TW]"Ticket1"zh-CN]"Ticket1"	[zh-TW]"Ticket1"zh-CN]"Ticket1"	1000	1000	2
<input type="checkbox"/>	6810	NULL	1905	16990	[zh-TW]"245(1.245qm.DJHKT)"	21340	[zh-TW]"DJHKT"zh-CN]"Ticket1"	[zh-TW]"Ticket1"zh-CN]"Ticket1"	100	100	1
<input type="checkbox"/>	6819	NULL	1905	16990	[zh-TW]"245(1.245qm.DJHKT)"	21340	[zh-TW]"DJHKT"zh-CN]"Ticket1"	[zh-TW]"Ticket1"zh-CN]"Ticket1"	100	100	1
<input type="checkbox"/>	6818	NULL	1905	16990	[zh-TW]"245(1.245qm.DJHKT)"	21340	[zh-TW]"DJHKT"zh-CN]"Ticket1"	[zh-TW]"Ticket1"zh-CN]"Ticket1"	140	140	1
<input type="checkbox"/>	6817	NULL	1905	16990	[zh-TW]"245(1.245qm.DJHKT)"	21340	[zh-TW]"DJHKT"zh-CN]"Ticket1"	[zh-TW]"Ticket1"zh-CN]"Ticket1"	170	170	1
<input type="checkbox"/>	6816	NULL	1905	16990	[zh-TW]"245(1.245qm.DJHKT)"	21340	[zh-TW]"DJHKT"zh-CN]"Ticket1"	[zh-TW]"Ticket1"zh-CN]"Ticket1"	180	180	1
<input type="checkbox"/>	6815	NULL	1905	16990	[zh-TW]"245(1.245qm.DJHKT)"	21340	[zh-TW]"DJHKT"zh-CN]"Ticket1"	[zh-TW]"Ticket1"zh-CN]"Ticket1"	200	200	1
<input type="checkbox"/>	6814	NULL	1905	16990	[zh-TW]"245(1.245qm.DJHKT)"	21340	[zh-TW]"DJHKT"zh-CN]"Ticket1"	[zh-TW]"Ticket1"zh-CN]"Ticket1"	0	0	1
<input type="checkbox"/>	6813	NULL	1905	16990	[zh-TW]"245(1.245qm.DJHKT)"	21340	[zh-TW]"DJHKT"zh-CN]"Ticket1"	[zh-TW]"Ticket1"zh-CN]"Ticket1"	140	200	1

Mass Edit Function

The Mass Edit Function empowers users to efficiently modify multiple entries in bulk. This feature streamlines the process of managing and updating information across large datasets, promoting consistency and saving time.

METHODOLOGY

System Design



System Architecture: Implement a client-server model using Vue.js for the front-end application and Laravel for the backend framework.

Frontend Development: Create a dynamic user interface with Vue.js, focusing on responsiveness and interactivity.

Backend Setup: Utilize Laravel to manage data securely, define API routes for front-end interaction, and perform CRUD actions efficiently.

Database Integration: Use MySQL for structured data storage, establishing a robust schema for user data.

User Interaction: Handle HTTP requests from the front-end application to the backend framework to retrieve and manipulate data.

Performance Optimization: Optimize components and database queries for enhanced performance.

Conclusion

- Revamped the platform to enhance usability and performance.
- Improved user experience with intuitive features and dynamic data visualization.
- Enhanced the system's user experience by addressing feedback from the admin team and event organizers.