
High Level Design Document

For



EVENTHIVE

Version 1.1

21/5/2023

Table of Contents

1. Introduction.....	1
1.1 Purpose	1
1.2 Definitions	1
2. Design Overview.....	2
2.1 Technologies Used	2
2.2 System Architecture	2
2.3 System Operations & Workflows.....	3
2.3.1 User-Admin Authentication overview	3
2.3.2 Event Creation, Reporting and Management.....	3
2.3.3 Ticket Sales and Purchasing	3
2.3.4 Ticket Validation:	4
3. Requirements [SRS].....	4
3.1 Functional Requirements.....	4
3.2 Non-Functional Requirements.....	5

Revision History

Name	Date	Reason For Changes	Version
Ahmed Essam	20/5/2023	//	V-1.0
Ahmed Essam	21/5/2023	Add SRS Stuff	V-1.1

1. Introduction

1.1 Purpose

This document outlines the key features, architectural components, and user interactions of the event management and ticketing system. It aims to assist the development team in creating a robust and scalable solution while addressing the specific needs of event organizers, attendees, and potential sponsors.

1.2 Definitions

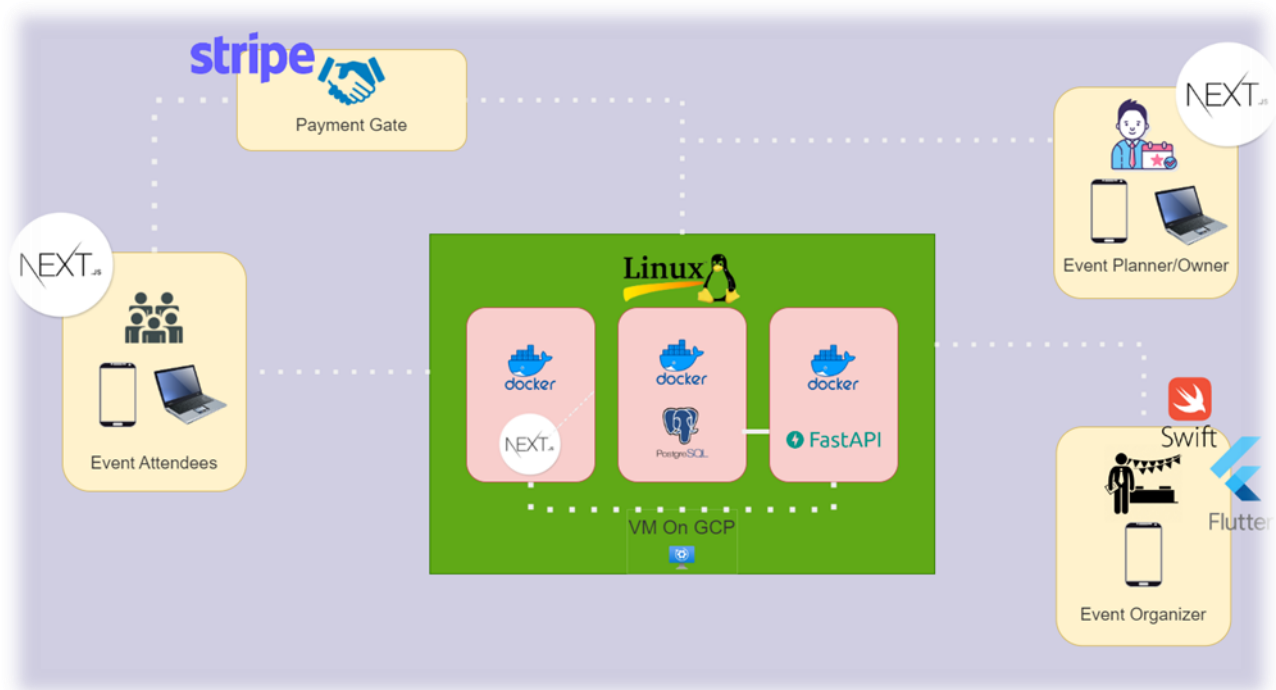
1. **Event Management:** The process of planning, organizing, and executing events, including tasks such as scheduling, venue selection, logistics management, promotion, and reporting.
 2. **Ticketing System:** A software solution that facilitates the creation, sale, and distribution of tickets for events, allowing organizers to manage ticket inventory and attendees to purchase and access tickets.
 3. **Front-End:** The user-facing part of a website or application, which includes the graphical interface and user interactions.
 4. **Back-End:** The server-side of a website or application, responsible for processing requests, managing data, and interacting with external services. It may also include reporting capabilities to provide insights on ticket sales, attendance, and revenue.
 5. **Reporting:** The process of generating and analyzing data related to event management and ticketing, including ticket sales, attendance, revenue, marketing effectiveness, and other relevant metrics. It enables stakeholders to make data-driven decisions and optimize event planning and performance.
6. **User Roles:**
- **Event Planner/Owner:** is responsible for the planning, coordination, and execution of events. They utilize the event management and ticketing system to track ticket sales, manage ticket inventory, and oversee event logistics. Their role includes ensuring a smooth on-ground experience by coordinating with vendors, managing event schedules.
 - **Event Organizer:** stationed at the event entrance or gates who manage ticket validation and entry. They use the event management and ticketing system to scan or verify tickets presented by event attendees, ensuring a smooth and secure entry process.
 - **Event Attendee:** individual who is interested in attending events. They use the event management and ticketing system to discover and purchase tickets for various events. The system provides them with the ability to explore event details, access event schedules, and receive updates or notifications related to the event.

2. Design Overview

2.1 Technologies Used

1. Frontend: Next.js
2. Backend: Fast-API
3. Mobile App: Swift (iOS) and Flutter (Android)
4. Database: PostgreSQL
5. Deployment: GCP

2.2 System Architecture



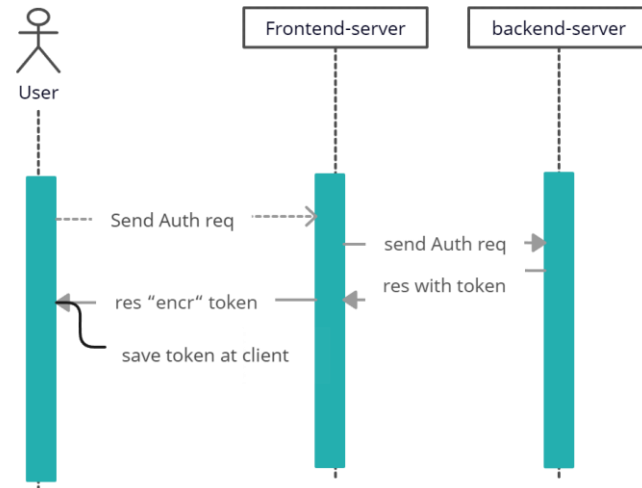
The system architecture diagram illustrates the integration of Next.js, FastAPI, Swift, Flutter, PostgreSQL, and Google Cloud VM in our event management and ticketing system. It showcases the efficient data flow, secure information storage, and scalability of the system.

2.3 System Operations & Workflows

High-level overview of the main functionalities in the event management and ticketing system. While these workflows cover the essential processes of event creation, reporting, ticket sales, and purchasing, it's important to remember that there are additional features and capabilities within the system. These workflows serve as a starting point to understand the core functionalities, but the system may offer more extensive features and workflows to cater to specific event management needs.

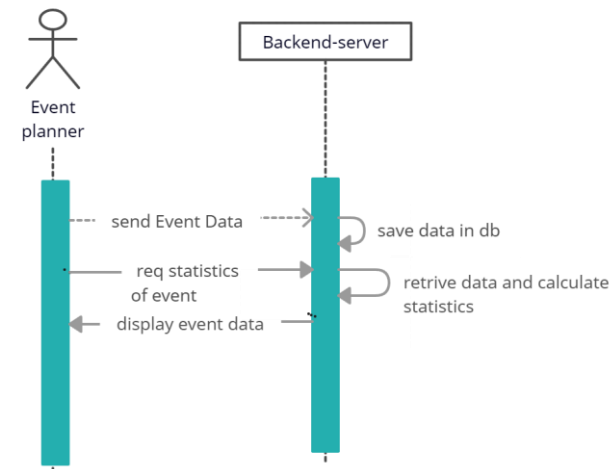
2.3.1 User-Admin Authentication overview

Authentication is implemented through a token-based approach. The client interacts with the frontend server, which handles communication with the backend server. The frontend server sends the user's credentials to Backend, generates an authentication token, and sends it back to the client. The client includes the token in subsequent requests which verifies it and communicates securely with the backend server on behalf of the client. This ensures secure access to the system's resources while maintaining a separation between the client and the backend server.



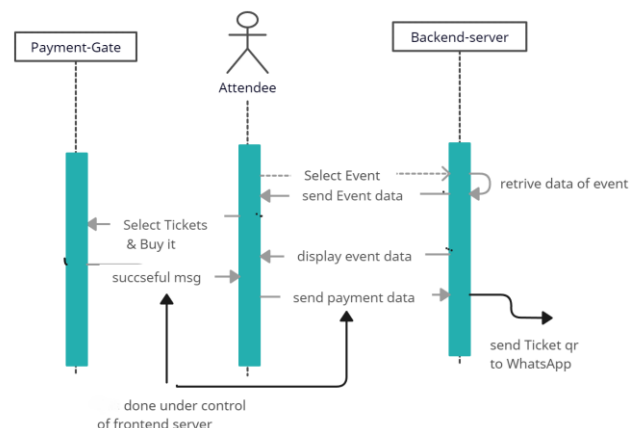
2.3.2 Event Creation, Reporting and Management

System offers a seamless experience for event organizers by providing efficient event creation, reporting, and management functionalities. Organizers can easily create and manage events by inputting event details, setting up ticket types and pricing, and defining event-specific rules. The system generates insightful reports on ticket sales, attendee demographics, revenue, and event performance, enabling organizers to assess event success and make informed decisions.



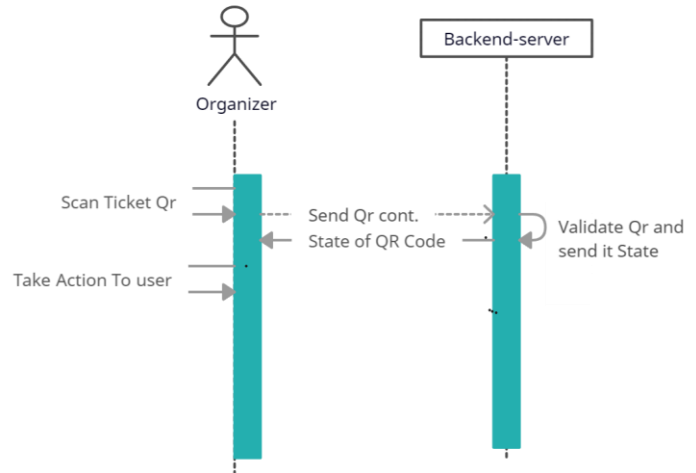
2.3.3 Ticket Sales and Purchasing

seamless ticket sales and purchasing experience. Attendees can browse and search for events, select tickets and quantities, and securely checkout using payment options. Upon successful payment, attendees receive a ticket confirmation with event details and a QR code on WhatsApp. At the event, they present their digital ticket for scanning or verification at the entrance. The system also supports ticket transfers and refunds, providing flexibility for attendees.



2.3.4 Ticket Validation:

Attendees' tickets are scanned or verified using QR codes or barcodes at the event gate. The system instantly verifies the ticket's authenticity and validity, ensuring authorized entry. This streamlined process minimizes the risk of fraudulent or duplicate tickets, enhancing event security and providing a smooth entry experience for attendees.



3. Requirements [SRS]

3.1 Functional Requirements

ID	Requirement
FR-1	Enable event organizers to post event details, including descriptions, prices, and ticket availability.
FR-2	Provide a user-friendly interface for event organizers to easily add and manage events.
FR-3	Provide event organizers with the ability to manage ticket sales, set pricing tiers, and limit ticket availability.
FR-4	Enable event organizers to design their event halls or venues, specifying seating arrangements.
FR-5	Provide a user-friendly hall design interface for event organizers to customize the layout and seating configurations.
FR-6	Allow event organizers to create different types of tickets for their events, such as general admission, VIP, student discounts, etc.
FR-7	Implement a ticket management system that enables event organizers to set ticket prices, quantity limits, and availability for each ticket type.
FR-8	Provide event organizers with the ability to track ticket sales and manage ticket inventory for each ticket type.
FR-9	Allow users to browse upcoming events, search for events based on specific criteria, and view event details and images.
FR-10	Implement search and filtering functionality to help users discover events happening on campus.
FR-11	Enable users to share events on social media platforms
FR-12	Develop a ticket reservation system that allows users to reserve tickets for events online

ID	Requirement
FR-13	Implement QR codes for event check-in and ticket verification to streamline the check-in process.
FR-14	Enable users to review and rate events, and provide a system for sharing feedback with event organizers.
FR-15	Encourage users to share events on social media and facilitate connections around common interests.
FR-16	Enable users to select and book tickets for their desired event, choosing from the available ticket types and quantities.
FR-17	Incorporate a seating selection feature for events with assigned seating, allowing users to choose specific seats or sections during the ticket booking process.
FR-18	Provide event organizers with the ability to export attendee lists or registration data in Excel format for further analysis.

3.2 Non-Functional Requirements

ID	Requirement
NFR-1	Fast, easy, and secure payment process.
NFR-2	Responsive design that works seamlessly on various devices.
NFR-3	High level of security measures to protect user payment information.
NFR-4	Reliable and scalable infrastructure to handle increased user traffic.
NFR-5	Efficient search functionality for quick event discovery.
NFR-6	Intuitive and streamlined ticket reservation process for users.
NFR-7	Support for multiple payment methods, providing flexibility to users.
NFR-8	Quick and accurate QR code generation and scanning for event check-in.
NFR-9	Easy maintainability and future enhancements for scalability and updates.
NFR-10	Comprehensive analytics and reporting features for event organizers to monitor performance.
NFR-11	Thorough documentation and user manuals for easy understanding and guidance.