**Software Requirements Specification Template**

**Acknowledgements:**

This template has been developed based upon the IEEE Guide to Software Requirements Specificatißon (ANSI/IEEE Std. 830-1984). The SRS templates of Karl E. Wiegers, Orest Pilskalns, Jack Hagemeister and others of unknown identity have also been referenced.

ß

**Software Requirement**

**Specification**

For

**Online Ticket Booking System**

Version 1

**Prepared by**

Abdul Mannan Omi

Zhiyun Chen

Yuchao Su

Wenjing Wang

Tanvir Ahmed Akash

Flinders University

21 September 2023

**Table of Contents**

**[1. Introduction](#_Toc981010113)** [3](#_Toc981010113)

[1.1 Purpose 3](#_Toc1605011172)

[1.2 Scope 3](#_Toc880677837)

[1.3 Definitions, Acronyms, and Abbreviations 4](#_Toc1095111335)

[1.4 References 5](#_Toc1601352555)

[1.5 Overview 5](#_Toc1667327681)

**[2. General Description](#_Toc262224864)** [5](#_Toc262224864)

[2.1 Product Perspective 5](#_Toc576845604)

[2.1.1 System Interfaces 6](#_Toc1302883870)

[2.1.2 User Interfaces 6](#_Toc1825938278)

[2.1.3 Hardware Interfaces 6](#_Toc1003322716)

[2.1.4 Software Interfaces 6](#_Toc803291568)

[2.1.5 Communication Interfaces 6](#_Toc1839178334)

[2.1.6 Memory 7](#_Toc190644620)

[2.1.7 Operations 7](#_Toc118527016)

[2.1.8 Site Adaptation Requirements 7](#_Toc1366217143)

[2.2 User Characteristics 7](#_Toc1116368677)

[2.3 General Constraints 8](#_Toc243730500)

[2.4 Assumptions and Dependencies 8](#_Toc1127198671)

**[3. Specific Requirements](#_Toc1874813310)** [8](#_Toc1874813310)

[3.1 External Interface Requirements 8](#_Toc2107232386)

[3.1.1 User Interfaces 8](#_Toc2101888825)

[3.1.2 Hardware Interfaces 9](#_Toc339488625)

[3.1.3 Software Interfaces 9](#_Toc2068753943)

[3.1.4 Communications Interfaces 9](#_Toc1787275071)

[3.2 Functional Requirements 9](#_Toc1878347708)

[3.2.1 Epics/User Stories 9](#_Toc1380317456)

[3.2.2 Use Cases and System Level Diagrams (Use Case Diagrams, Activity Diagrams and Data Flow Diagrams) 11](#_Toc1877128098)

[3.2.3 Expanded Use Case for selected Epics/User Stories (one per student) 17](#_Toc209685009)

[3.3 Non-Functional Requirements 28](#_Toc155281536)

[3.3.1 Performance 28](#_Toc624144447)

[3.3.2 Reliability 29](#_Toc1685588781)

[3.3.3 Availability 29](#_Toc86371043)

[3.3.4 Security 29](#_Toc2086657976)

[3.3.5 Maintainability 29](#_Toc2052647122)

[3.3.6 Portability 29](#_Toc1662874046)

[3.4 Other Requirements 29](#_Toc571909064)

**[4. Review and Reflection](#_Toc2086318323)** [30](#_Toc2086318323)

**[Appendices](#_Toc639066445)** [30](#_Toc639066445)

[A.1 Issues List 30](#_Toc1224022468)

[A.2 Team meeting minutes 30](#_Toc1399765063)

# 1. Introduction

This document provides a comprehensive overview of the Adelaide Fringe Ticket Booking System, outlining its objectives, functionalities, and features. It serves as a valuable resource for both stakeholders and system developers. The system is an integral part of the Adelaide Fringe website, offering a range of services and benefits to both registered members and non-members.

## 1.1 Purpose

This document aims to elucidate the Adelaide Fringe Ticket Booking System's objectives and functionalities. It will provide an overview of the system's features, its intended purpose, the interfaces it utilizes, its core functions, the limitations it must adhere to, and how it responds to external stimuli. This comprehensive document serves as a valuable resource for both stakeholders and system developers.

## 1.2 Scope

On our website, visitors have access to a range of pages and features. They can explore sections such as Home, About Us, Information, Location of Adelaide Fringe, Contact Us, Promotions, Booking, Pricing, Login, and Sign up. Additionally, users can check out the price rates for both non-members and members. For guests who haven't registered, they can still browse the website's content and provide feedback to us.

Once users sign up as members and log in, they gain access to a range of exclusive features and information on our platform. Here's an overview of what our registered members can enjoy:

**Home:** The starting point for all your interactions with us.

**About Us:** Learn more about our organization, values, and mission.

**Information:** Access detailed information about our events, and the experiences we offer.

**Location of Adelaide Fringe:** Explore the exact locations of our place, helping you to plan to move more effectively.

**Contact:** Get in touch with us easily for any queries or assistance you may need.

**Promotion:** Stay updated on our latest promotions, discounts, and special offers.

**Booking:** Exclusively available to our members, this feature allows you to reserve places according to your preferences. Required details include:

* Date and Time
* Number of Seat
* Seat Type
* Payment Information

**Price:** Access pricing information, helping you to book tickets within your budget.

**Payment:** Conveniently make online payments for your bookings through our secure payment gateway.

**Profile Update:** Manage your personal information and preferences to streamline your booking process.

**Prices in Detail:** Gain access to comprehensive pricing details, including any additional charges or fees.

By becoming a member, you unlock the ability to book our event tickets with ease and make secure online payments. We're dedicated to providing our members with a seamless experience from the beginning to confirming their reservations. Your comfort and satisfaction are our top priorities.

To book a ticket as a non-member, you'll need to complete a straightforward registration process to become a member. The registration requirements include providing the following information: Full Name, Username, Address, Phone Number, Email Address, Gender, Date of Birth, Photo ID/Passport Number.

In addition to your personal details, we also require billing information to facilitate payment via bank transfer. The necessary billing information consists of: Reference Number, Name, Bank Name. Once you've registered and provided this information, you'll be able to book a ticket.

## 1.3 Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| SRS | Software Requirement System |
| Administrator | Admin |
| Project | Projects are initiated to generate a distinctive outcome, whether it's a product, service, or result, while adhering to predefined limitations such as time, financial resources, and project scope. |
| Stakeholder | The person who is directly or indirectly associated with and has a vested interest. |

## 1.4 References

1. Aamir Hoda, Tanya Sinha, Amarjeet Kumar (2017). “Movie Ticket Booking System”, February 12, 2017.

2. Shubham Shah, Akshay Yadav, Sudeep Shrivastava (2014). “Online Show Booking System”, Version 1.0, July 07, 2014.

3. Prof. Prithviraj Y J, S Vaishnavi, Swathi R, Vemala Susmitha (2016). “Online Movie Ticket Reservation”, May 09, 2016.

4. Visalatchmi A/P Rajasvaran, Chitran A/L Tamil, Chelvan, Ghanavathi A/L Kumaran, Suthan A/L Karunamuthy (2013). “The Sea Blue Chalet Booking System”, February 2013.

5. Richard Mansfield (2005). “CSS Web Design for Dummies”, March 2005.

## 1.5 Overview

The "General Description" section of this document provides a broad outline of the system's functionality. It presents the informal requirements and serves as a foundation for understanding the technical requirements specification within the document.

The "Requirements Specification" section is designed for the benefit of both stakeholders and developers. It delves into the technical intricacies of the system's functionality, using precise technical language to articulate the system's specific details.

# 2. General Description

This section will demonstrate how the system interacts with external contexts and highlights its fundamental functionality. Additionally, this segment will introduce a straightforward ticket booking management system with integrated promotional features. The Adelaide Fringe Ticket Booking System has been meticulously crafted to streamline ticket sales while also enhancing operational efficiency by minimizing costs. The system enables customers to make bookings and payments, while also providing organizers with the flexibility to interact with customers on an ongoing basis. This interaction includes tasks such as adjusting prices and managing promotions. Besides, the system will allow the organizer to keep track of the viewing number of members who book the tickets and the process of booking. The Adelaide Fringe Ticket Booking System could be able to view booking and payment reports by the organizer. The organizer can attach some promotion updates for customers to view. The new user of the system also can sign up and view the website, then drop feedback.

## 2.1 Product Perspective

The Adelaide Fringe Festival's website plays a vital role in the festival's broader ecosystem. It serves as a crucial interface connecting various interconnected systems, including ticketing, artist management, and administrative tools. This integration is designed to provide users with a seamless experience, streamlining processes such as event ticketing, artist profiles, and administrative functions. By working in synergy with other related products and projects, the website significantly improves the overall functionality and efficiency of the festival.

### 2.1.1 System Interfaces

This booking system comprises seven distinct modules, each designed to serve specific functions to achieve the system's objectives. These modules include the User Module, Admin Desk Module, and Member Module.

The system's administrator, referred to as the owner, has exclusive access to the Admin Desk Module, which provides essential functions not available in the other modules.

For customers, the primary interface is the Member Module, which allows them to perform actions such as logging in, booking tickets, and logging out.

Additionally, new users who wish to use our booking system can utilize the User Module to create an account (signup) and subsequently proceed to log in.

### 2.1.2 User Interfaces

The user interface (UI) of this system adapts to individual user roles and permissions, offering a tailored experience. Users begin by logging in, which grants them access to the homepage. The homepage displays essential functions, but their availability depends on the user's authorization level.

This system prioritizes user-friendliness, making it accessible even to newcomers. New users can sign up easily and gain insights from experienced users before fully utilizing the system's features.

### 2.1.3 Hardware Interfaces

Like any conventional system, this setup necessitates a fundamental computing configuration comprising a CPU, monitor, keyboard, and mouse. Alternatively, one can opt for a laptop or smartphone for input and output purposes. If printing is necessary, a printer is also required.

### 2.1.4 Software Interfaces

To develop the Adelaide Fringe Ticket Booking System, you'll need two essential software components: XAMPP and Visual Studio Code. This system streamlines the booking process, facilitating user access within a local network and enabling online reservations.

### 2.1.5 Communication Interfaces

The system's communication interface relies entirely on the server software to ensure the accurate transmission and retrieval of data from the database. Additionally, it employs an online protocol to establish a connection between the GPS and the system.

### 2.1.6 Memory

For optimal performance, the program requires a minimum of 2 GB of RAM and at least 80 GB of hard disk space.

### 2.1.7 Operations

* To enhance security, users must log in to the system before accessing its functions.
* The system operates online 24/7 for continuous availability.
* Daily data backups will be generated by the system.
* The system can retrieve files from these backups when needed.

### 2.1.8 Site Adaptation Requirements

No modifications are necessary for site adaptation of the software. The system operates seamlessly on a standard computer connected to the organization's local area network.

## 2.2 User Characteristics

The online booking system for Adelaide Fringe has been meticulously crafted to ensure user-friendliness. It has been designed in such a way that individuals can easily navigate it without the need for extensive training. A brief introduction by experienced users will suffice to acquaint newcomers with its various features. However, it is essential for all users to possess basic computer skills and a proficiency in English since the system operates exclusively in this language. Users will also appreciate the significant time-saving benefits when making reservations through this streamlined system.

 The Adelaide Fringe Website serves a diverse array of users, each with their unique roles and needs. These users include artists, volunteers, audiences, software developers, managers, and fringe committees.

* Artists utilize the platform to showcase and promote their creative work, making it essential for them to have user-friendly tools for self-promotion and event promotion.
* Volunteers rely on the website's event management tools to efficiently organize and assist in various festival activities, necessitating robust features for event coordination and scheduling.
* Audiences seek a seamless and enjoyable experience while exploring events and purchasing tickets. This demands a user-friendly interface and a smooth ticketing process.
* Software developers play a crucial role in maintaining the website's functionality and ensuring it runs smoothly, necessitating access to developer resources and support.
* Managers are responsible for planning, organizing, and executing various festival aspects. They require efficient tools and information to make informed decisions and coordinate festival logistics.
* Fringe committees are instrumental in the success of different event types, ranging from corporate conferences to community festivals. They rely on the website for communication, planning, and execution.

Users of the website vary in technical proficiency, access the site from various devices, and engage in social interactions. As such, the website must cater to these diverse user types while providing a user-friendly and immersive experience for festival planning, exploration, and interaction.

## 2.3 General Constraints

Ensuring a successful festival-related project entails several critical considerations. First and foremost, adhering to a precise timeline is imperative to meet all festival-related deadlines. Simultaneously, it's crucial to manage budgetary constraints for both the initial development and ongoing maintenance phases.

Moreover, the project must accommodate compatibility with a diverse range of web browsers and mobile devices, catering to a broad audience. Additionally, prioritizing compliance with web accessibility standards is essential to guarantee inclusivity, and robust security measures must be in place to safeguard user data and transactions.

Furthermore, the website's design and functionality should seamlessly integrate with the festival's well-established branding and identity. This ensures a consistent and immersive user experience that aligns with the festival's ethos and enhances overall engagement.

## 2.4 Assumptions and Dependencies

* Users of the system are required to have their own unique ID and password for logging in.
* Adelaide Fringe must have backup capabilities in place to prevent data loss. If the company lacks backup storage, it is assumed that the local server will maintain uptime and prevent data loss.
* It is assumed that all users of this system possess basic computer knowledge and can operate the system proficiently.

# 3. Specific Requirements

## 3.1 External Interface Requirements

### 3.1.1 User Interfaces

The primary user interface of the Adelaide Fringe Booking System shall be accessible through web browsers. It shall be designed to provide an intuitive and user-friendly experience for patrons, event organizers, and administrators. Furthermore, the system shall offer mobile-optimized interfaces for users accessing the system via smartphones and tablets. These interfaces shall provide full functionality and responsiveness.

### 3.1.2 Hardware Interfaces

For physical ticket sales and merchandise purchases at event venues, the system shall interface with various POS systems, including Cash Registers, Card Readers and Receipt Printers. For generating physical tickets, event organizers shall be able to connect a variety of printers to the system, including thermal ticket printers and standard office printers. The system shall provide print templates and support multiple printer types to accommodate various ticket formats. In venues that support barcode ticket scanning, the system shall be compatible with standard barcode scanners.

### 3.1.3 Software Interfaces

|  |  |
| --- | --- |
| Database Interface | The system interacts with the MySQL database to store and retrieve essential data related to events, user profiles, bookings, and ticketing information. It uses SQL queries to access, update, and manage this data. |
| Operating System Interface | * The system is hosted on a Linux server, ensuring stability and security. * It utilizes the operating system's file management for log storage and data backup |
| APIs and Libraries | * The system integrates with various APIs and libraries to enhance functionality, including social media sharing via Facebook, Twitter, and Instagram. * APIs are used to retrieve event data from external providers, ensuring comprehensive event listings. * The system also utilizes standard web development libraries and frameworks (e.g., React, Express.js) for the web-based user interface |

### 3.1.4 Communications Interfaces

* **HTTP/HTTPS:** The system should communicate with browsers by the HTTP/HTTPS protocols to ensure data encryption and secure transmission.
* **Database:** Database connection should use secure protocols to protect the transmission of sensitive data.
* **Email communication:** The system should use SMTP or other email protocols to send and receive the email notifications.

## 3.2 Functional Requirements

### 3.2.1 Epics/User Stories

User Stories for the Online Event Booking System: **Admin Portal**

**Epic 1: Event Management**

As an admin staff member, I want to create and manage events so that customers can browse and book tickets for upcoming events.

**Epic 2: Venue and Location Management**

As an admin staff member, I want to add and maintain venue and location details so that events can be associated with specific places.

**Epic 3: Ticket Type Configuration**

As an admin staff member, I want to define ticket types, their pricing, capacity, and availability so that customers can choose the ticket options that suit them.

**Epic 4: Revenue Tracking**

As an admin staff member, I want to track event revenue, view sales progress, and generate reports so that we can make data-driven decisions.

**Epic 5: User Account Management**

As an admin staff member, I want to manage user accounts, including registration and updating user details, to enhance the customer experience.

**Epic 6: Location and Venue Setup**

As an admin staff member, I want to set up and configure locations and venues, including specifying seating layouts, to ensure accurate event planning.

**Public-Facing Website:**

**Epic 7: Event Search and Discovery**

As a customer, I want to search for events based on various criteria, such as date, location, event type, and price, so that I can find events that interest me.

**Epic 8: Event Details and Seating Plans**

As a customer, I want to view detailed event information, including event descriptions and seating plans, to make informed decisions about attending.

**Epic 9: Ticket Booking**

As a customer, I want to book tickets for events, select specific seats (if available), and receive a confirmation with a QR code for admission so that I can easily attend the event.

**Epic 10: Customer Account Management**

As a customer, I want to create and manage my account to simplify the booking process for future events and keep my information up to date.

**Epic 11: Ticket Access and Printing**

As a customer, I want to access my booked tickets on the website, view them, and print them when necessary to ensure a smooth check-in process at events.

**Epic 12: Event Sharing and Feedback**

As a customer, I want to share event information on social media and provide feedback or reviews for events I attended, enhancing community engagement, and sharing my experiences.

### 3.2.2 Use Cases and System Level Diagrams (Use Case Diagrams, Activity Diagrams and Data Flow Diagrams)

**High-Level Use Cases for the Online Event Booking System:**

|  |  |
| --- | --- |
| **Search for Events** |  |
| * Actors: Customer (Primary) * Type: Primary * Description: The customer initiates this use case by entering search criteria (e.g., date, location, event type) to discover events that match their preferences. The system then retrieves and displays a list of relevant events. |

|  |  |
| --- | --- |
| **Book Tickets for an Event** |  |
| * Actors: Customer (Primary) * Type: Primary * Description: The customer selects an event of interest, chooses ticket types, specifies seat preferences (if applicable), and proceeds to book tickets. The system verifies availability, processes the booking, and sends a confirmation with a QR code for admission. |

|  |  |
| --- | --- |
| **Manage User Account** |  |
| * Actors: Customer (Primary) * Type: Primary * Description: Customers can create, update, or delete their user accounts. This use case allows customers to maintain their contact information and preferences for a seamless booking experience. |

|  |  |
| --- | --- |
| **Create and Manage Events** |  |
| * Actors: Admin Staff (Primary) * Type: Primary * Description: Admin staff initiated this use case to create and manage events. They input event details, specify venues, configure ticket types, and set pricing, ensuring events are available for customer booking. |

|  |  |
| --- | --- |
| **Track Event Revenue** |  |
| * Actors: Admin Staff (Primary) * Type: Primary * Description: Admin staff monitor event revenue and sales progress. This use case provides insights into the financial performance of events and allows staff to make informed decisions. |

|  |  |
| --- | --- |
| **Generate Reports** |  |
| * Actors: Admin Staff (Primary) * Type: Primary * Description: Admin staff can generate various reports, including attendance reports and revenue trends, to analyze data and improve event planning and marketing strategies. |

|  |  |
| --- | --- |
| **Define Ticket Types and Capacity** |  |
| * Actors: Admin Staff (Primary) * Type: Primary * Description: Admin staff define different ticket types for events, specifying pricing, capacity, and availability. This use case ensures a variety of ticket options are available to customers. |

|  |  |
| --- | --- |
| **Access Seating Plans** |  |
| * Actors: Customer (Primary) * Type: Primary * Description: Customers can access seating/venue plans for events that offer assigned seating. This feature helps customers choose specific seats or understand the venue layout. |

|  |  |
| --- | --- |
| **Share Event Information** |  |
| * Actors: Customer (Primary) * Type: Primary * Description: Customers can share event details, ticket availability, and their booking status on social media or with friends and family, promoting events and increasing engagement |

|  |  |
| --- | --- |
| **Provide Event Feedback** |  |
| * Actors: Customer (Optional) * Type: Optional * Description: Customers have the option to provide feedback and reviews for events they attend. This use case helps gather feedback for event improvement and community engagement. |

|  |  |
| --- | --- |
| **Configure Locations and Venues** |  |
| * Actors: Admin Staff (Primary) * Type: Primary * Description: Admin staff configure locations and venues, including setting up seating layouts. This ensures accurate event planning and proper venue setup. |

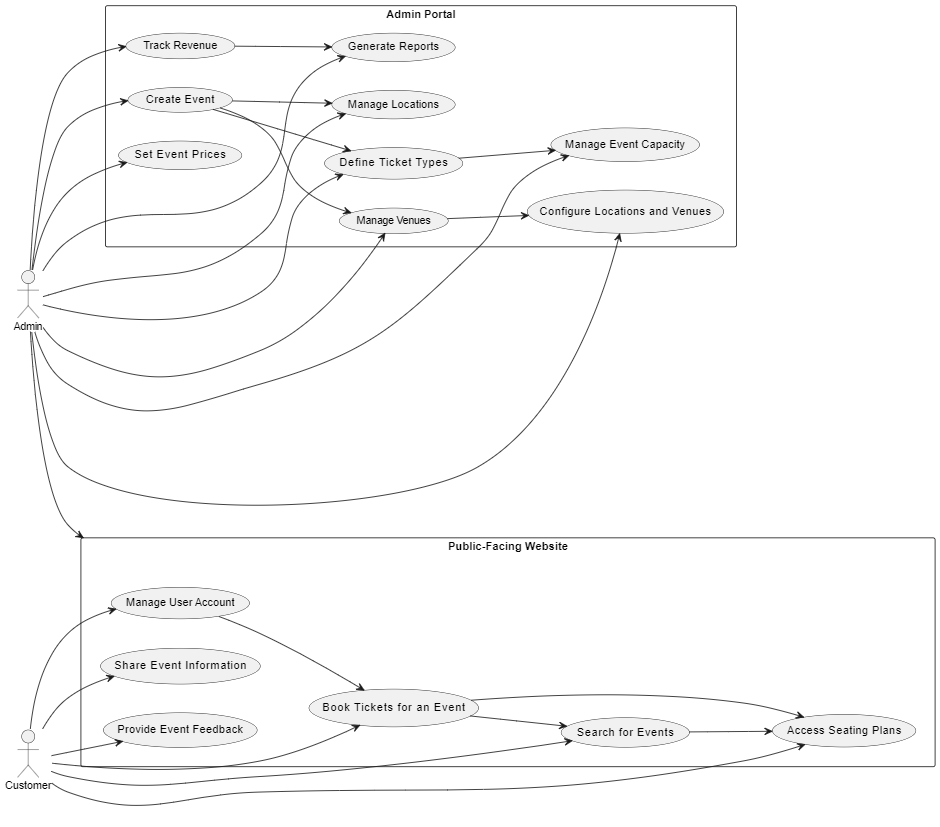
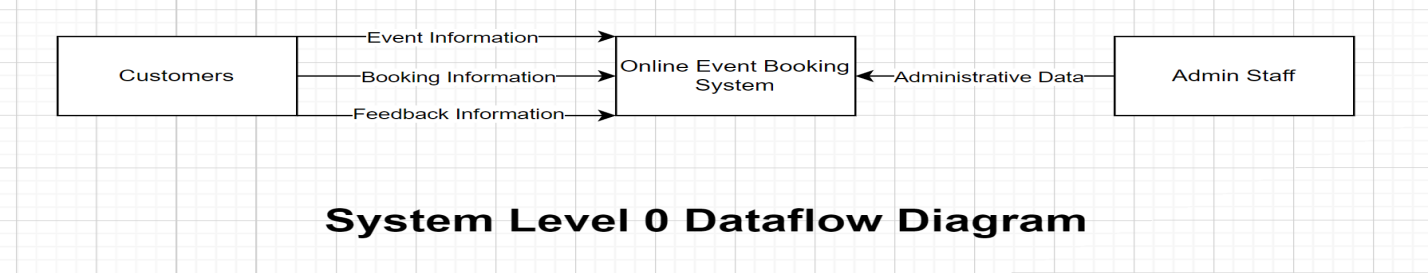


Figure: System level Use Case Diagram

 Figure: System level 0 Data Flow Diagram

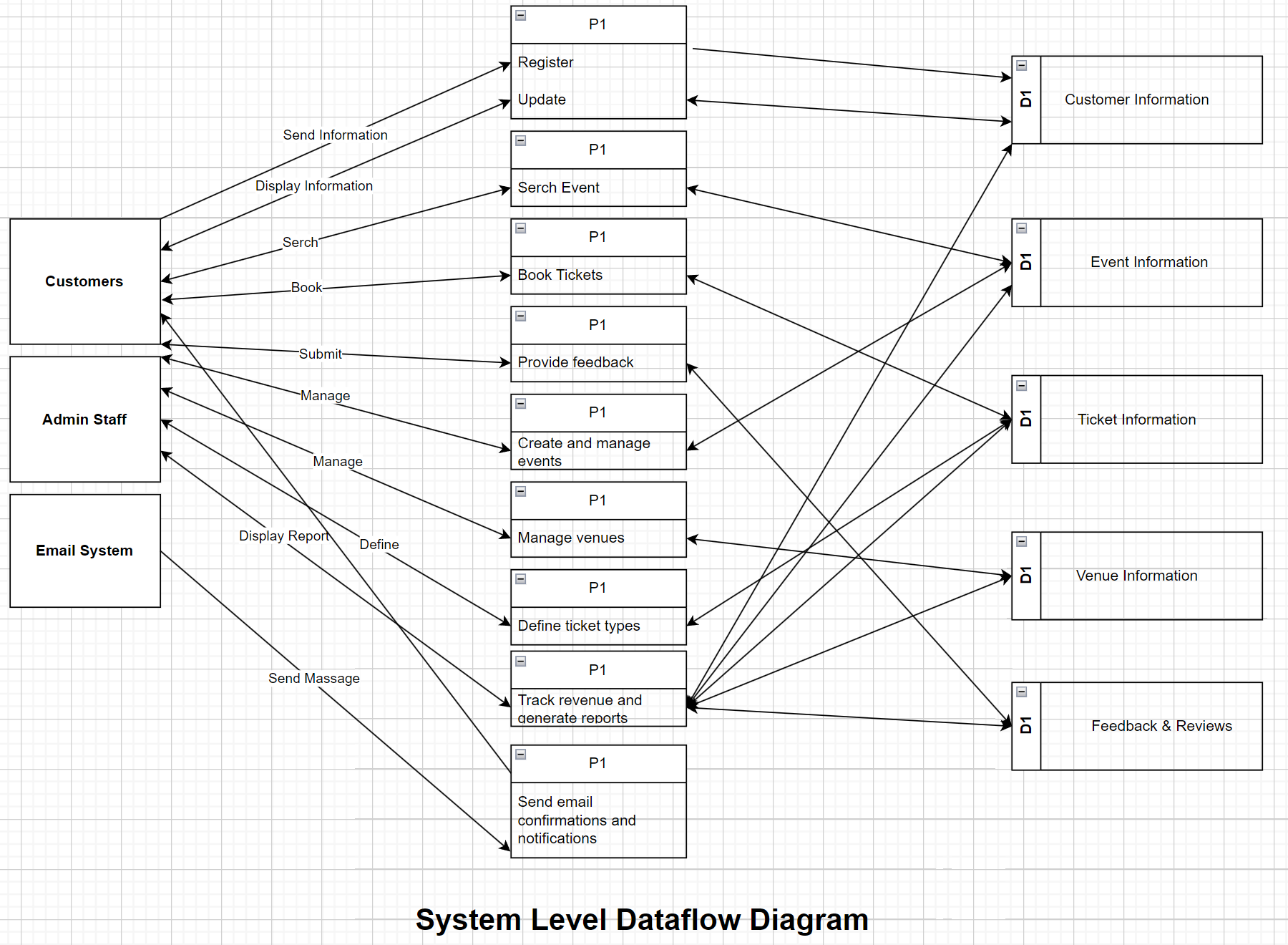


Figure: System level 1 Data Flow Diagram

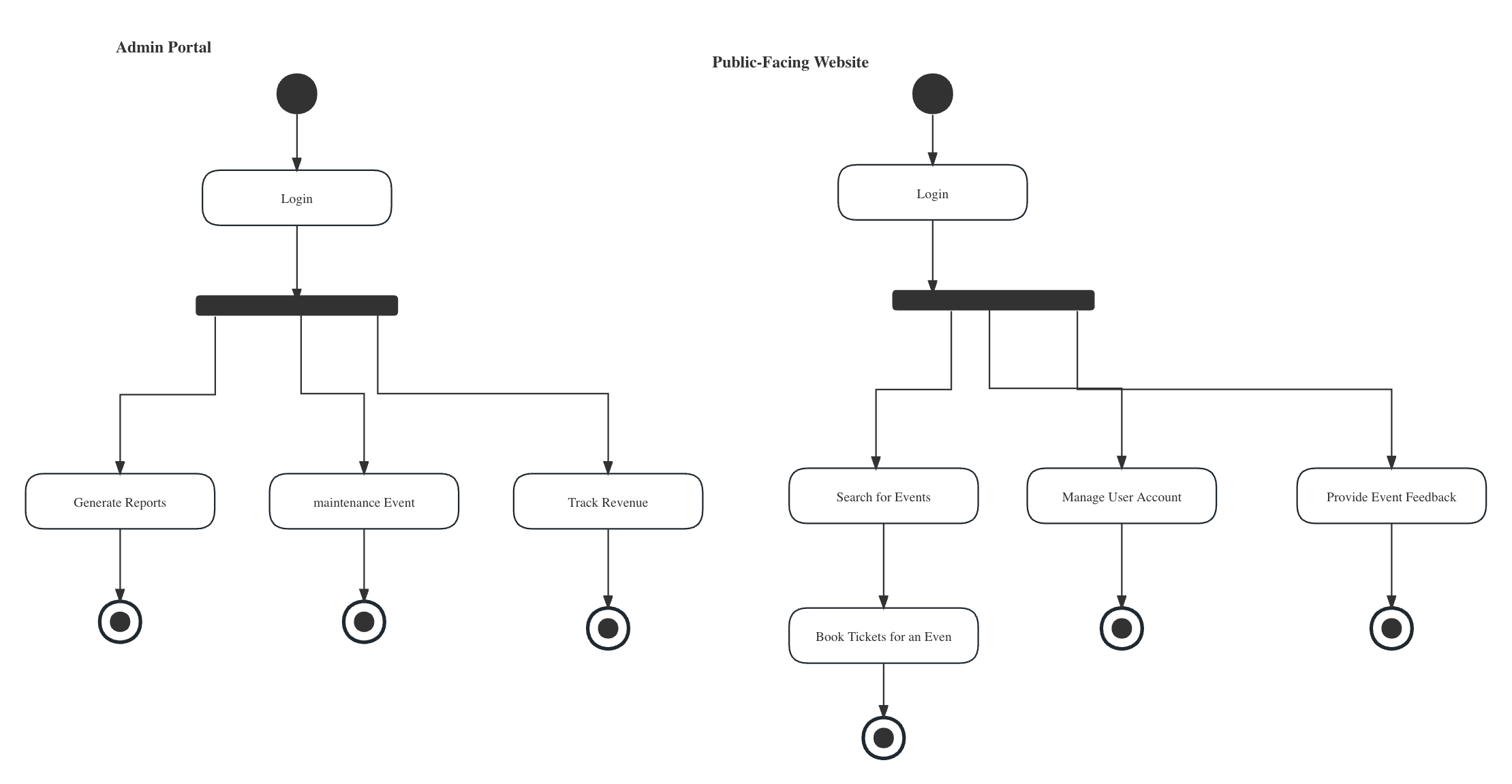


Figure: System level Activity Diagram

### 3.2.3 Expanded Use Case for selected Epics/User Stories (one per student)

**Expanded Use Case 1: Book Tickets for an Event (Yuchao Su)**

Actors: Customer (Primary)

Trigger: The customer intends to reserve tickets for a selected event.

Purpose: To facilitate the booking of tickets for events, allowing customers to attend their chosen events.

Summary: The customer selects an event from the search results, specifies the number of tickets and seating preferences, and proceeds to book the tickets. The system verifies ticket availability and sends a confirmation with a QR code for admission.

Type: Primary and Essential

Cross Reference: This use case is closely related to the "Search for Events" and "Access Seating Plans" use cases.

Assumed Pre-Conditions: The customer has successfully completed the "Search for Events" use case, and there are available tickets for the selected event.

Process:

1. The use case begins when the customer selects an event from the search results.
2. The system displays event details, including event name, date, location, ticket types, and seating plans (if applicable).
3. The customer chooses the desired ticket types and specifies the number of tickets they wish to purchase.
4. If the event offers assigned seating, the customer can select specific seats from the seating plan.
5. The system verifies ticket availability and reserves the selected tickets.
6. The system prompts the customer to review their booking and proceed to payment.
7. The customer provides payment information.
8. The system processes the payment.
9. The system sends a booking confirmation via email, including a QR code for admission.

Variations:

A - Ticket availability exhausted:

* If all tickets for the selected event are sold out during the booking process, the system informs the customer and suggests they choose another event or date.

B - Payment failure:

* If the payment transaction fails, the system provides an error message and prompts the customer to retry the payment process.

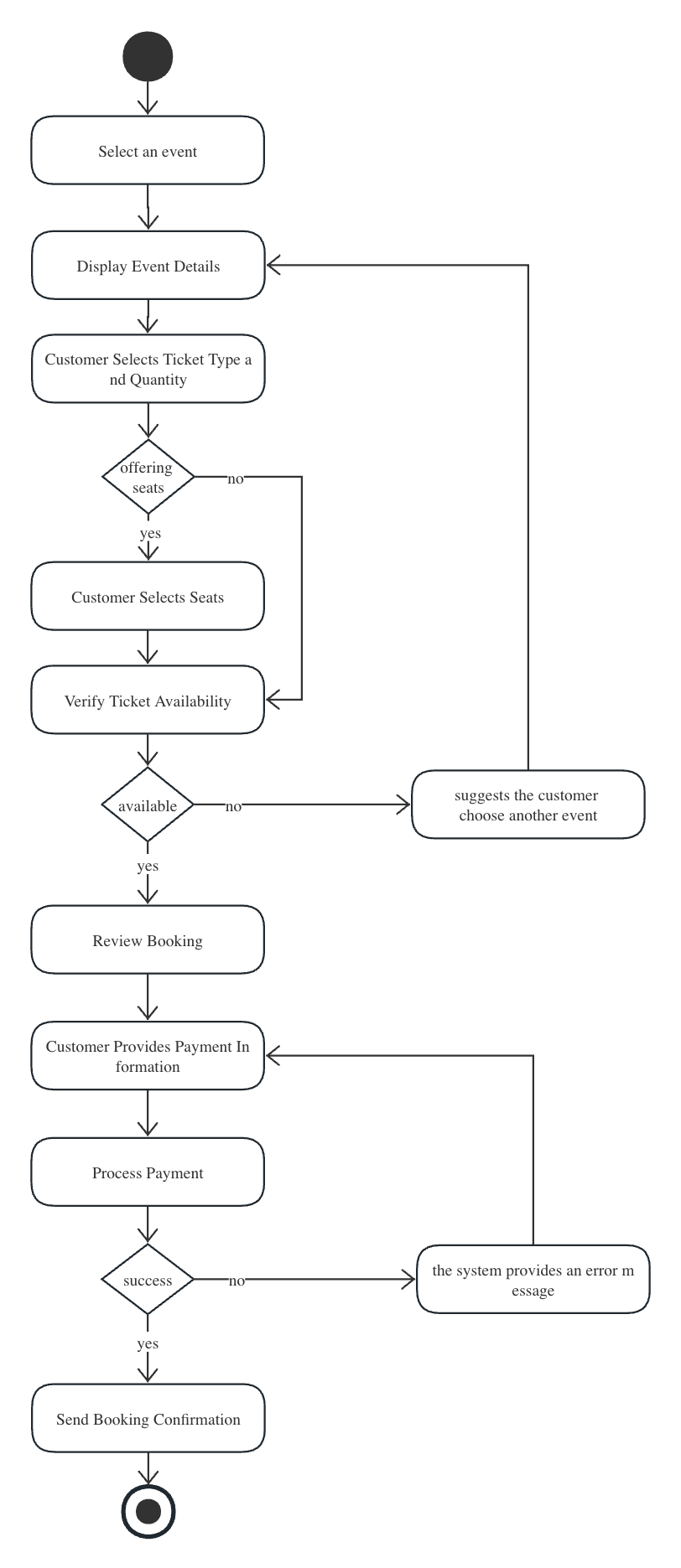


Figure: Activity Diagram

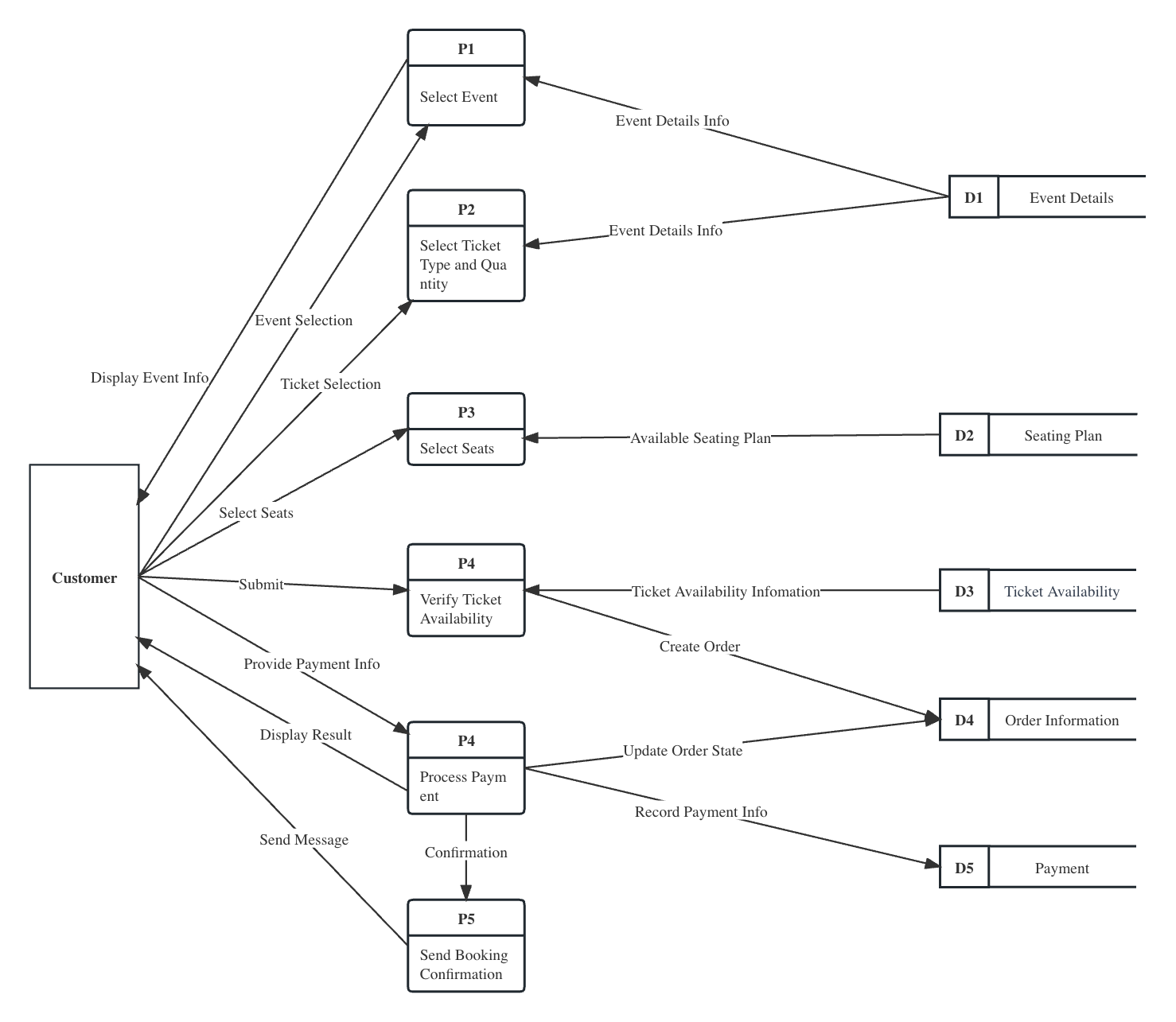


Figure: Data Flow Diagram

**Expanded Use Case 2: Search for Events (Omi)**

Actors: Customer (Primary)

Trigger: The customer intends to find events based on specific criteria.

Purpose: To assist customers in discovering events that match their preferences and interests.

Summary: The customer initiates the use case by entering search criteria, such as date, location, event type, and price range. The system retrieves events that meet the criteria and displays them to the customer.

Type: Primary and Essential

Cross Reference: This use case is closely related to the "Book Tickets for an Event" and "Access Seating Plans" use cases.

Assumed Pre-Conditions: The system is operational, and events have been added to the system.

Process:

1. The use case begins when the customer enters search criteria, such as event date, location, event type, and price range.
2. The system performs a search based on the entered criteria.
3. The system retrieves a list of events that match the customer's criteria.
4. The system displays the search results, including event names, dates, locations, and brief descriptions.
5. The customer reviews the search results and selects an event of interest for further details or booking.
6. The system allows the customer to proceed with booking or view more information about the selected event.

Variations:

A - No events match the entered criteria:

* The system informs the customer that no events match the specified criteria.
* The system provides an option for the customer to modify their search criteria and retry the search.

B - Technical issues:

* If there are technical issues preventing the search, the system displays an error message and suggests the customer try again later or contact support.

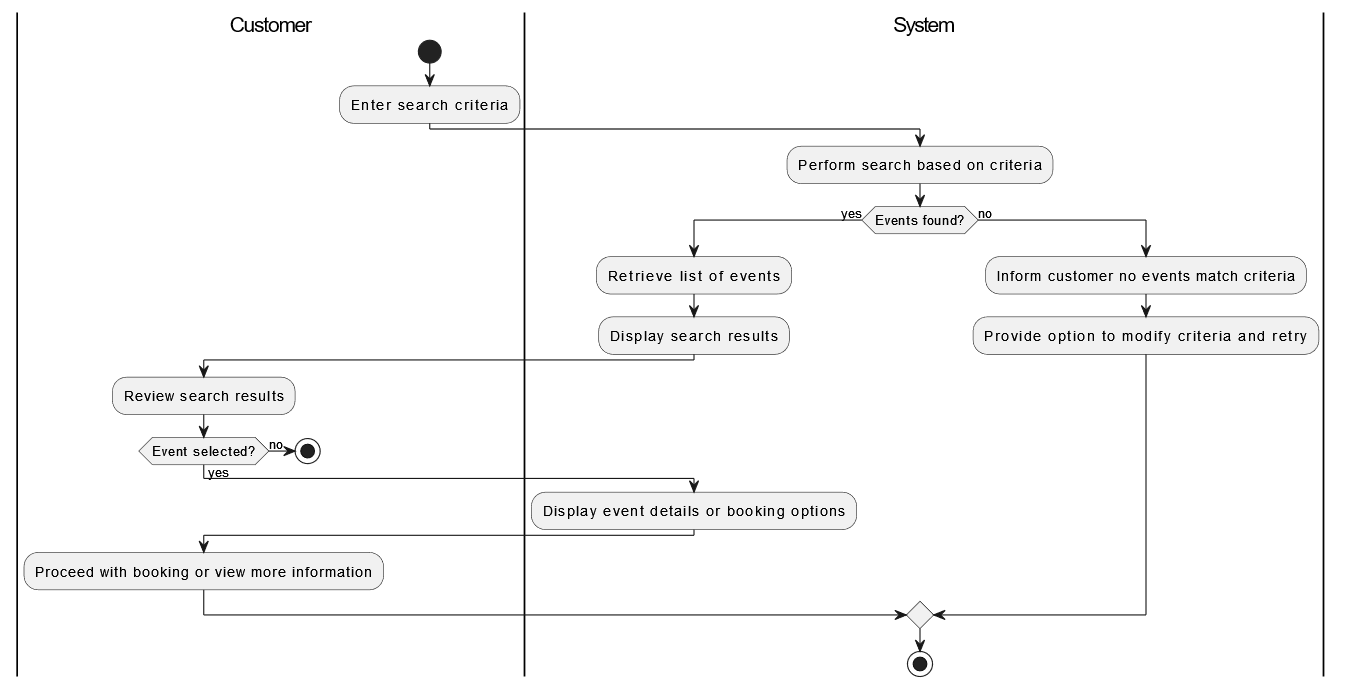


Figure: Activity Diagram

**Expanded Use Case 3: Manage User Account (Wenjing Wang)**

Actors: Customer (Primary)

Trigger: The customer intends to create, update, or delete their user account.

Purpose: To provide customers with the ability to manage their personal information and preferences within the system.

Summary: Customers can initiate this use case to create a new user account, update existing account details, or delete their account if needed.

Type: Primary and Essential

Cross Reference: This use case is essential for maintaining accurate customer information and enhancing the booking experience.

Assumed Pre-Conditions: The customer is accessing the system with or without an existing user account.

Process:

1. The use case begins when the customer chooses to manage their user account.
2. If the customer has an existing account, the system prompts them to log in with their credentials. If not, the customer can proceed to create a new account.
3. For new account creation, the customer provides necessary information such as name, email address, and password.
4. For account updates, the customer can modify their contact information, password, or preferences as needed.
5. If the customer chooses to delete their account, the system prompts them to confirm their decision.
6. The system processes the requested action, whether it's creating, updating, or deleting the user account.
7. The system provides appropriate notifications to confirm the successful completion of the chosen action.

Variations:

A - Incorrect login credentials:

* If the customer provides incorrect login credentials during account updates, the system informs them of the error and prompts them to retry.

B - Account deletion confirmation:

* If a customer chooses to delete their account, the system asks for confirmation to prevent accidental deletion.

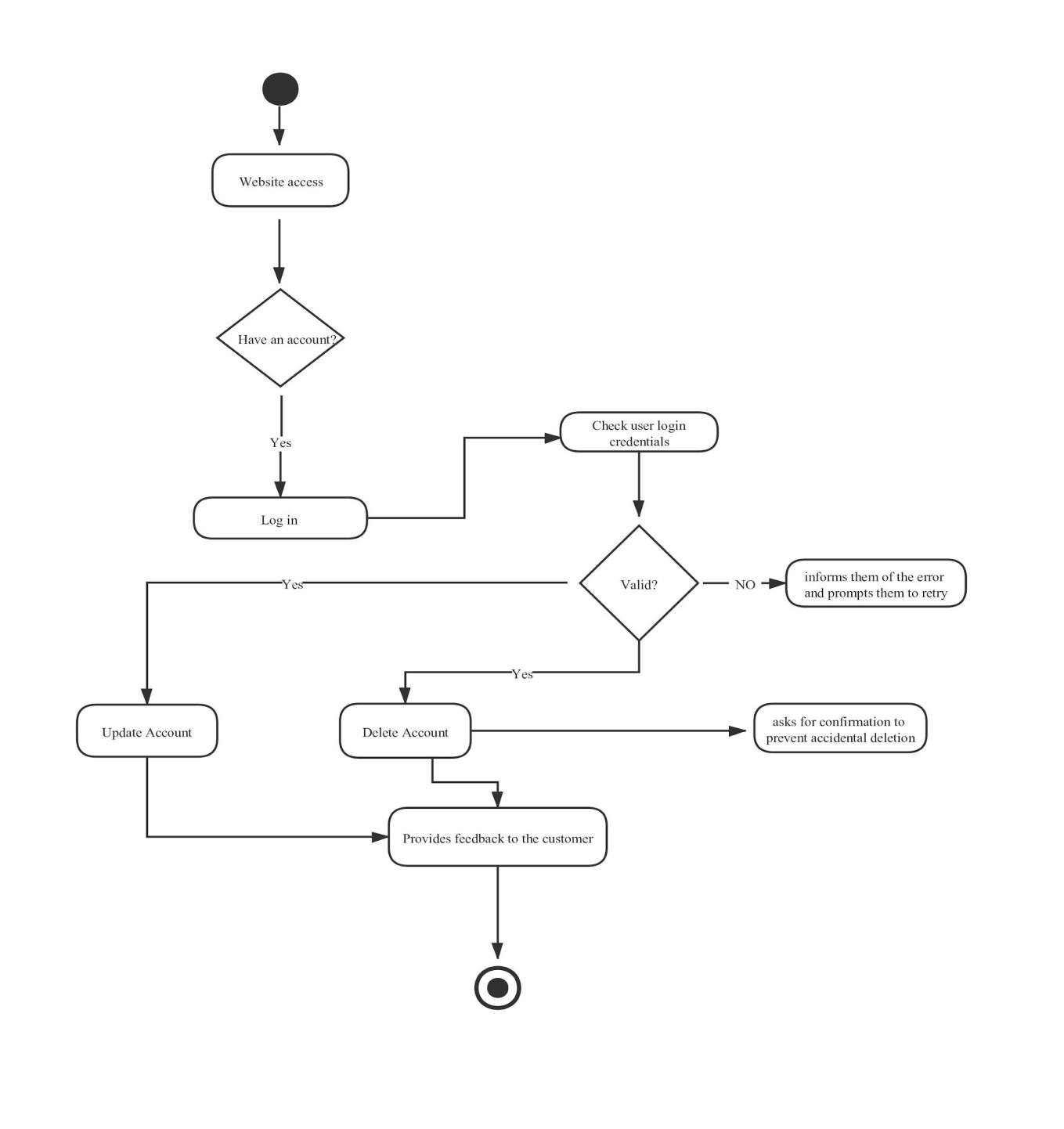


Figure: Activity Diagram

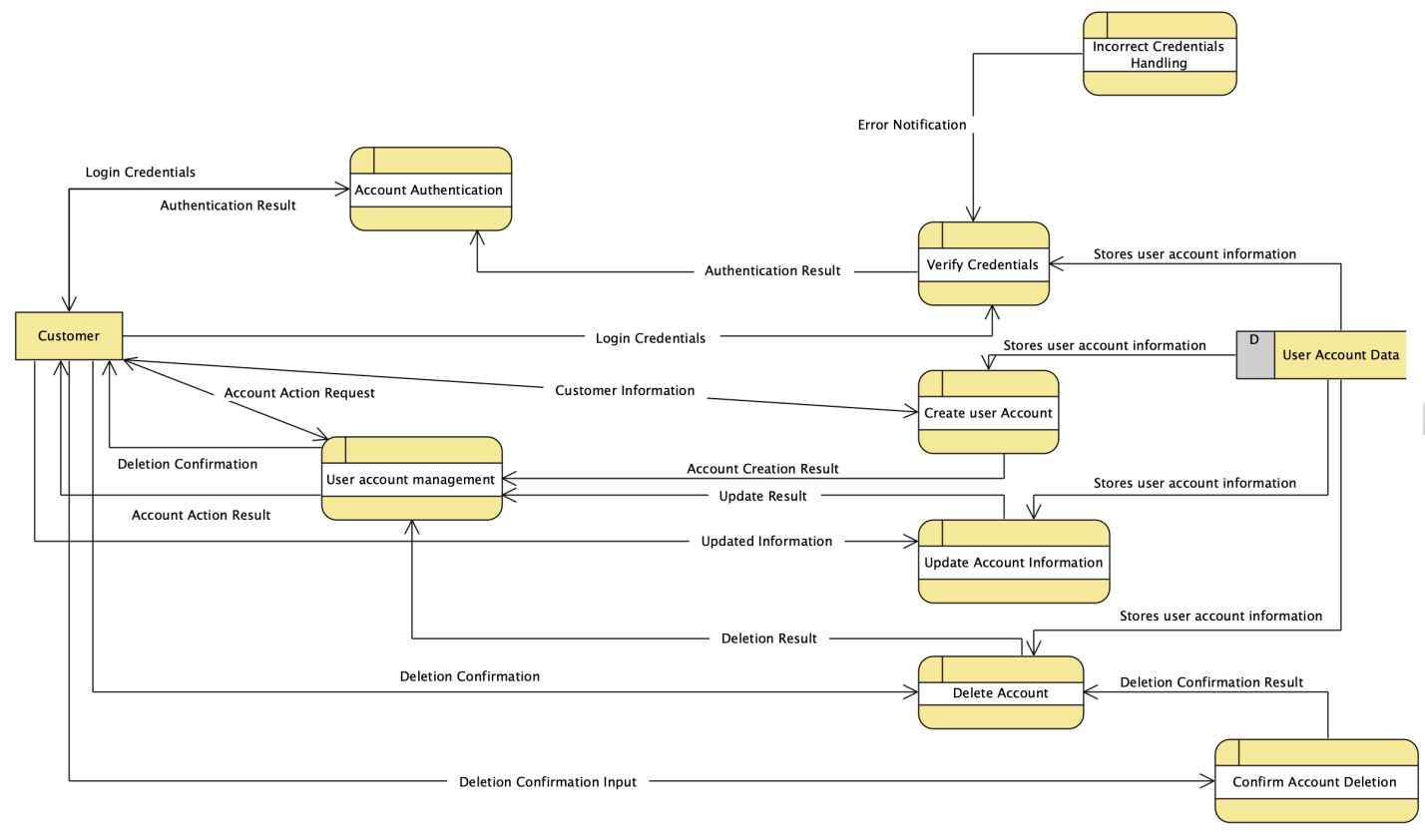


Figure: Data Flow Diagram

**Expanded Use Case 4: Define Ticket Types and Capacity (Zhiyun Chen)**

Actors: Admin Staff (Primary)

Trigger: Admin staff intend to specify ticket types, pricing, capacity, and availability for events.

Purpose: To provide admin staff with the tools to configure ticket options and manage event capacity effectively.

Summary: Admin staff initiated this use case to define various ticket types for events, set pricing, specify the maximum capacity for each type, and determine availability.

Type: Primary and Essential

Cross Reference: This use case is essential for ensuring a variety of ticket options are available to customers.

Assumed Pre-Conditions: Admin staff are logged into the system and have selected an event to configure.

Process:

1. The use case begins when admin staff select an event they wish to configure.
2. The system displays event details and options for defining ticket types.
3. Admin staff input the name of the ticket type, pricing details, and specify the maximum capacity for each type.
4. Admin staff set availability dates and times for each ticket type, if applicable.
5. The system validates the entered information and stores it for the selected event.
6. Admin staff can review and modify the configured ticket types before finalizing the setup.

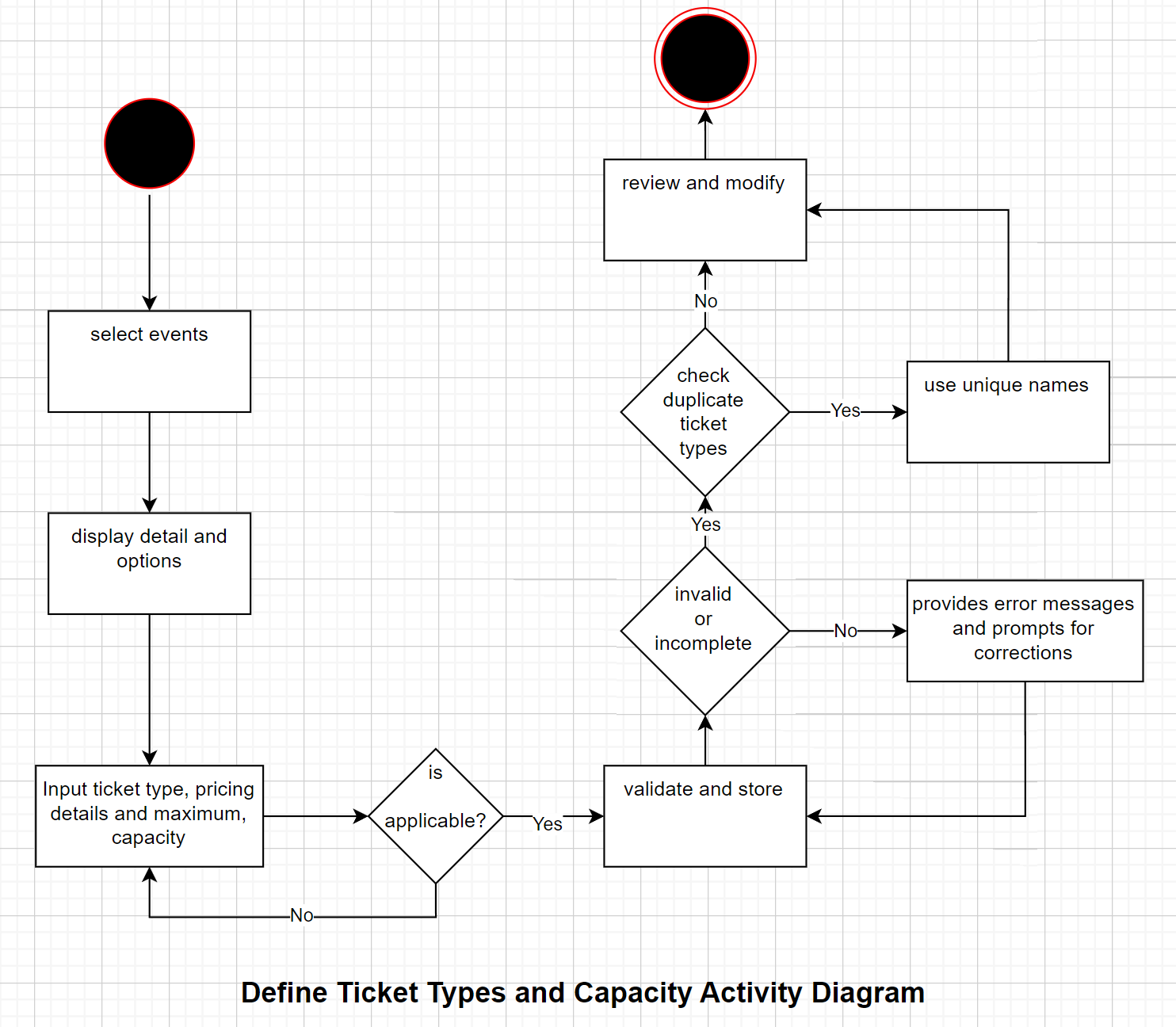
Variations:

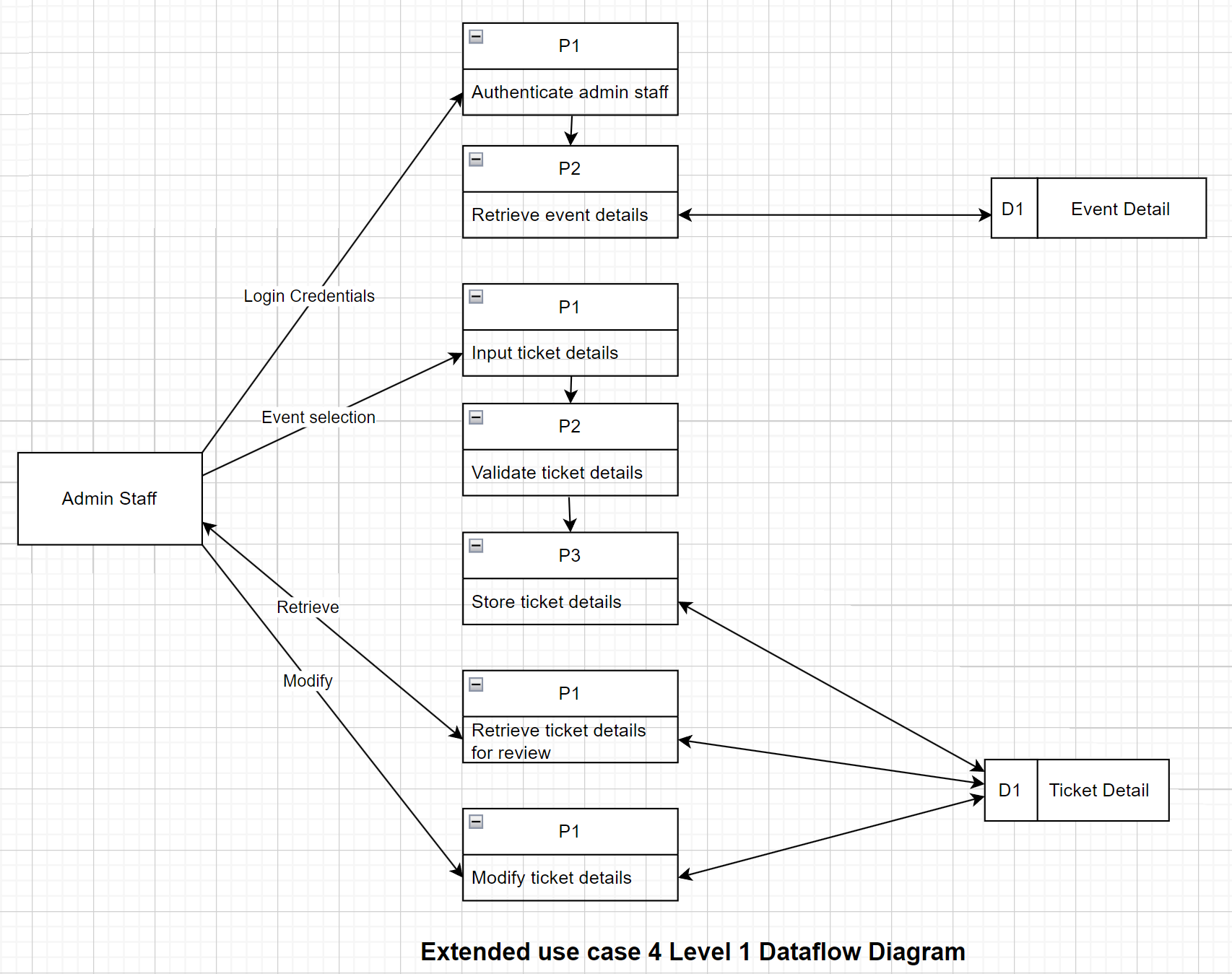
A - Invalid input:

* If admin staff enter invalid or incomplete information, the system provides error messages and prompts for corrections.

B - Duplicate ticket types:

* If admin staff attempt to define duplicate ticket types for the same event, the system prevents duplication and informs them to use unique names.





**Expanded Use Case 5: Access Seating Plans (Tanvir Ahmed Akash**)

Actors: Customer (Primary)

Trigger: The customer intends to view seating/venue plans for events that offer assigned seating.

Purpose: To enable customers to make informed decisions about seat selection when booking tickets.

Summary: Customers can access seating/venue plans for events with assigned seating. They can choose specific seats or understand the venue layout before booking.

Type: Primary and Essential

Cross Reference: This use case complements the "Book Tickets for an Event" use case by providing customers with seating information.

Assumed Pre-Conditions: The customer is viewing an event that offers assigned seating.

Process:

1. The use case begins when the customer selects an event from the search results.
2. If the event offers assigned seating, the system displays the seating/venue plan for the event.
3. The customer can zoom in, plan, and explore the seating layout to view available seats.
4. The system highlights available seats in different colors and provides details about seat categories.
5. The customer selects specific seats or sections and proceeds with ticket booking.

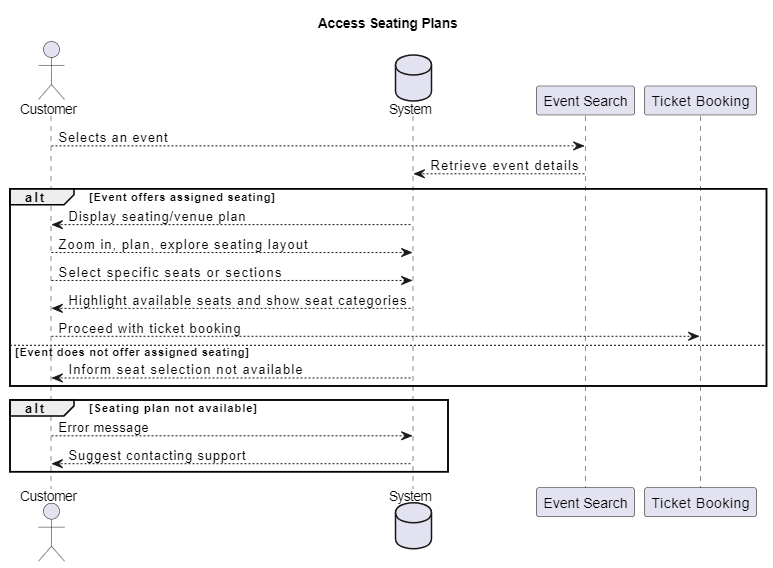
Variations:

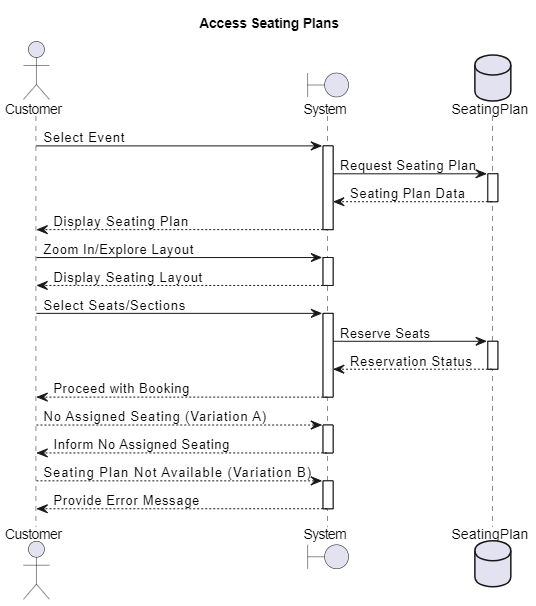
A - No assigned seating:

* If the selected event does not offer assigned seating, the system informs the customer that seat selection is not available for this event.

B - Seating plan not available:

* In rare cases where the seating plan is not available or cannot be retrieved, the system provides an error message and suggests contacting support.





## 3.3 Non-Functional Requirements

Non- Functional Requirements define the quality attributes and limitations that the booking system should meet. These requirements are related to the performance, reliability, availability, security, maintainability, and portability of the system usually.

### 3.3.1 Performance

The maximum acceptable response time for the order confirmation should be specified as within a second. The number of transactions should be defined as 10000 concurrent users that this booking system should handle per unit of time. At the same time, the response time can be maintained within 3 seconds under high load conditions.

### 3.3.2 Reliability

The minimum uptime of this booking system should be 99.9% over a rolling 12-month period, excluding scheduled maintenance windows. The integrity of booking data should be ensured by the booking system. No more than 0.01% of bookings shall result in data inconsistencies or errors. In addition, the system shall be able to automatically handle some of the common errors and provide logging and alarm functions to detect and solve potential problems in a timely manner.

### 3.3.3 Availability

The system should have a minimum uptime of 99.9% during the entire duration of the Adelaide Fringe festival to ensure that customers can book tickets at any time without disruption.

### 3.3.4 Security

The system must comply with industry-standard security practices, including data encryption, secure payment processing, and protection against common web vulnerabilities such as SQL injection and cross-site scripting (XSS). The data protection will be the point. This includes encryption mechanisms, data masking, secure communication protocols, and access controls to prevent unauthorized access to sensitive information.

### 3.3.5 Maintainability

Consideration for backward compatibility and smooth upgrade paths for new versions of the software can help in maintaining user satisfaction and reducing disruptions. Implementing a robust testing strategy with unit tests, integration tests, and regression tests ensures that changes can be verified without introducing new defects. Designing the software in a modular and component-based manner allows for changes to be made to specific parts of the system without affecting other parts. This promotes reusability and reduces the risk of unintended consequences.

### 3.3.6 Portability

The system's adaptability and accessibility shall be ensured across various platforms and environments. It addresses compatibility with operating systems and web browsers, platform independence for mobile devices, support for multiple database management systems, integration with external services, data import and export capabilities, and configurability without code modifications. This section aims to guarantee that the booking system can seamlessly function on diverse systems and provide flexibility to both users and administrators.

## 3.4 Other Requirements

**3.4.1 Legal and Compliance Requirements**

The booking system should comply with all relevant data privacy and protection regulations, including but not limited to GDPR (General Data Protection Regulation) and local data protection laws in SA. The collection and management of user data must comply with relevant legal regulations. Ensure the security of user data.

# 4. Review and Reflection

In this project, considering the complexity of the system and the urgency of time, we brainstormed and made decisions using agile development. For quality evaluation, we mainly focused on three points: function, user experience, and system performance. In teamwork, weekly team meetings are highly effective. We communicate in time to solve the problem encountered, sync progress, and make plans. We have some teamwork rules, and it is helpful. At the same time, we recognized areas for improvement in document updating. So finally, we utilized confluence for document management.

# Appendices

## A.1 Issues List

1. Page interaction logic for the disabled

Pending decisions:

1. Decision pending on whether to support multiple languages, and if so, which specific languages.

## A.2 Team meeting minutes

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Location | Attendees | Meeting notes |
| 2023.08.10 | Tonsley | All team members | Identify mainly functional and non-functional requirements |
| 2023.08.17 | Tonsley | All team members | Discussing for the concept of user story and epic.  Identify mainly user story and epic |
| 2023.08.24 | Zoom | All team members | Introduce completed use cases |
| 2023.08.30 | Zoom | All team members | Introduce completed use cases and activity diagram |