

## Message From Sean

I have now checked out the contents of the repository for grading. Any further details added after the deadline will be ignored.

## Team Project: *Group 05*

### Team Members

Number	Name	Email(s)	CSGitLab Username
TM1	Wang Shuhan	shuhan.wang@ucdconnect.ie	@WangShuhan
TM2	Gu Ruixin	ruixin.gu@ucdconnect.ie	@RuixinGu
TM3	Rong Yuze	yuze.rong@ucdconnect.ie	@22207274
TM4	An Ran	ran.am@ucdconnect.ie	@RanAnQWQ
TM5	Li Siying	siying.li@ucdconnect.ie	@siying.li
TM6	Shen Jinyan	jinyan.shen@ucdconnect.ie	@Shen_Jinyan
TM7	Wan Zitong	zitong.wan@ucdconnect.ie	@Rostiuete
TM8	Qin Chenle	chenle.qin@ucdconnect.ie	@22207284

### Milestone 1 Requirements Analysis

#### Distribution of work on this milestone

#### Overall Distribution of Work

Team Member	TM1	TM2	TM3	TM4	TM5	TM6	TM7	TM8
Percentage	20%	16%	13%	13%	12%	12%	7%	7%

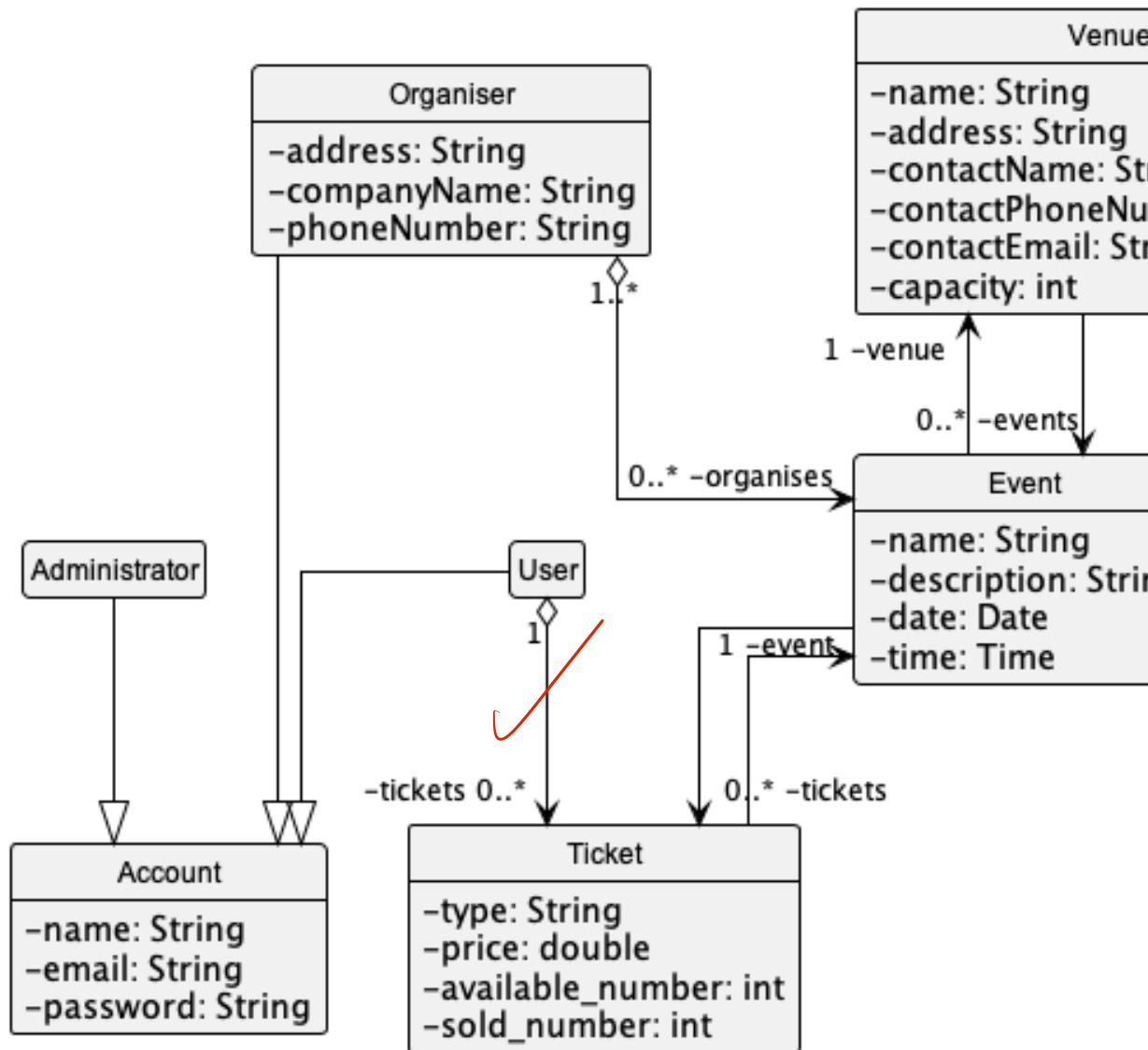
### Reflection Statements

Team Member	Contribution	Reflection Statement
TM1	<Required:	Team Leader. Responsible for organizing team meetings, discussing task distribution, and leading the Use Case design. Completed use case descriptions for the Organizer role and designed UI diagrams for special scenarios. Responsible for final compilation and review of all project documentation.>
TM2	<Required:	Designing the domain model diagrams; Reviewing, and supplementing the use case descriptions and UI diagrams for the Admin and User roles. Compiled the final README file.>
TM3	<Required:	Completing the detailed descriptions of the domain model. Assisted with the diagram design.>
TM4	<Required:	Determine the use case content and name, drew a use case diagram and completed the System Glossary.>
TM5	<Required:	Completed half of the UI diagram and participated in the design of the use case description.>
TM6	<Required:	Completed half of the UI diagram and participated in the design of the use case description.>
TM7	<Required:	The User section in the use case description.>
TM8	<Required:	The Admin section in the use case description.>

### Requirements Analysis

This is the requirements analysis or business modelling for the restaurant application.

## Domain Model



Domain Model Introduction

## Use Case Diagram

The use cases are shown in this diagram



## ## System Glossary

Term	Description
Administrator	A user role responsible for managing the entire system including user accounts and events.
Organiser	A user role that manages events, including creating and editing them and managing venues.
User	A regular user who can purchase tickets and manage their own account details.
Account	Stores personal or organizational information of users, organizers, and administrators, including names, email addresses, passwords, and more.
Venue	A location where events are held, with specific attributes like name, address, and capacity.
Event	An occurrence that has a specific date, time, and place, with ticketing details associated.
Ticket	A reservation for an event that includes details about the seat type and price.

## Use Case Descriptions

This includes the following use cases:

### Administrator Use Cases

1. View All Events
2. View Events History
3. Delete Event
4. View All Venues
5. View Venue Usage History
6. Delete Venue
7. View All Accounts
8. Create User Account

9. Create Organizer Account
10. Create Administrator Account
11. Edit Account Permission
12. Edit User Account
13. Edit Organizer Account
14. Edit Administrator Account
15. Disable Account
16. Delete Account

## Organizer Use Cases

17. View All Events
18. List Events
19. Create Event
20. Edit Event
21. Delete Event
22. View Event Details
23. View Event Registrations
24. Display Date
25. List Venues
26. Create Venue
27. Edit Venue
28. View Venue Details
29. View Venue Bookings
30. View Organizer Account
31. Update Organizer Account



## User Use Cases

32. Create User Account
33. Display Date
34. View All Events
35. View User Account
36. List Upcoming Events
37. View Event Details (User)
38. Search and Filter Events
39. Purchase Ticket
40. View Purchased Tickets
41. Cancel Ticket

## UI Prototypes/Sketches

Page Name	Image																																																																																																																																																																									
	<p>The UI prototype for 'View All Events' features a header with navigation buttons (back, forward, search, history), a date selector (2024/10/09), and links for 'View All Venues' and 'View All Accounts'. Below the header is a grid-based calendar view. The columns represent hours from 2:00 to 22:00. The rows are labeled 'Venue 1' through 'Venue 12'. Two events are shown: 'Event 1' from 8:00 to 20:00 at Venue 1, and 'Event 2' from 14:00 to 22:00 at Venue 2.</p> <table border="1"> <thead> <tr> <th></th> <th>2:00</th> <th>4:00</th> <th>6:00</th> <th>8:00</th> <th>10:00</th> <th>12:00</th> <th>14:00</th> <th>16:00</th> <th>18:00</th> <th>20:00</th> <th>22:00</th> <th>0:</th> </tr> </thead> <tbody> <tr> <td>Venue 1</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td style="background-color: #cccccc;">Event 1</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Venue 2</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td style="background-color: #cccccc;">Event 2</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Venue 3</td> <td></td> </tr> <tr> <td>Venue 4</td> <td></td> </tr> <tr> <td>Venue 5</td> <td></td> </tr> <tr> <td>Venue 6</td> <td></td> </tr> <tr> <td>Venue 7</td> <td></td> </tr> <tr> <td>Venue 8</td> <td></td> </tr> <tr> <td>Venue 9</td> <td></td> </tr> <tr> <td>Venue 10</td> <td></td> </tr> <tr> <td>Venue 11</td> <td></td> </tr> <tr> <td>Venue 12</td> <td></td> </tr> </tbody> </table>		2:00	4:00	6:00	8:00	10:00	12:00	14:00	16:00	18:00	20:00	22:00	0:	Venue 1							Event 1						Venue 2									Event 2				Venue 3													Venue 4													Venue 5													Venue 6													Venue 7													Venue 8													Venue 9													Venue 10													Venue 11													Venue 12												
	2:00	4:00	6:00	8:00	10:00	12:00	14:00	16:00	18:00	20:00	22:00	0:																																																																																																																																																														
Venue 1							Event 1																																																																																																																																																																			
Venue 2									Event 2																																																																																																																																																																	
Venue 3																																																																																																																																																																										
Venue 4																																																																																																																																																																										
Venue 5																																																																																																																																																																										
Venue 6																																																																																																																																																																										
Venue 7																																																																																																																																																																										
Venue 8																																																																																																																																																																										
Venue 9																																																																																																																																																																										
Venue 10																																																																																																																																																																										
Venue 11																																																																																																																																																																										
Venue 12																																																																																																																																																																										

Use Case 1: "View All Events"

A screenshot of a modal window titled "EVENT". It displays two event entries:

Event	Date	Venue	People Reserved	Action
Event 1	01/01/2024	Venue 1	50 people reserved	<a href="#">Delete</a>
Event 2	02/01/2024	Venue 2	100 people reserved	<a href="#">Delete</a>

At the bottom left of the modal is a "Cancel" button.

Use Case 2: "View Events History"

A screenshot of a modal window titled "Confirm". It asks the user if they are sure to delete the event:

Are you sure to delete this event?

At the bottom right of the modal are two buttons: "Yes" and "No".

Use Case 3: "Delete Event"

A screenshot of a page titled "Venues". It lists four venues with their details and actions:

Venue	Action	Action
Venue 1	<a href="#">View History</a>	<a href="#">Delete</a>
Venue 2	<a href="#">View History</a>	<a href="#">Delete</a>
Venue 3	<a href="#">View History</a>	<a href="#">Delete</a>
Venue 4	<a href="#">View History</a>	<a href="#">Delete</a>

Use Case 4: "View All Venues"

## Venue 1

### History

Event 3 03/01/2024

[Delete](#)

Event 2 02/01/2024

[Delete](#)

Event 1 01/01/2024

[Delete](#)

[Cancel](#)

[Submit](#)

Use Case 5: “View Venue Usage History”

## Confirm

Are you sure to delete this venue?

[Yes](#)

[No](#)

Use Case 6: “Delete Venue”

## Account Management

Eszter

User

[Permission](#)

[Edit](#)

[Disable](#)

[Delete](#)

Siiing

Organizer

[Permission](#)

[Edit](#)

[Disable](#)

[Delete](#)

[Cancel](#)

[New Account](#)

Use Case 7: “View All Accounts”

## Account Info

**Name**

**Email**

**Password**

 ▼

**Type**

 ▼

**Cancel** **Submit**

Use Case 8: “Create User Account”

## Account Info

Name

Email

Password

 ▼

Address

Company Name

Phone Number

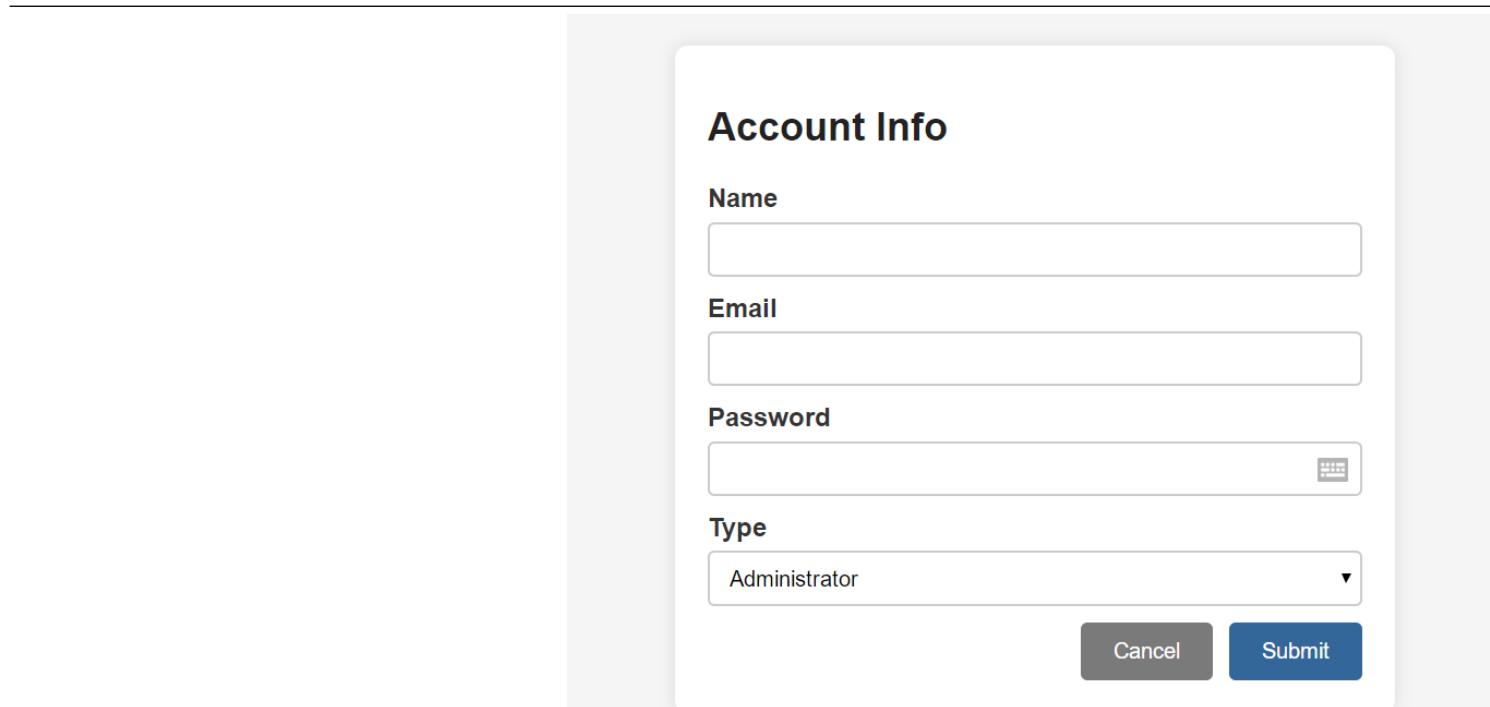
Type

 ▼

Cancel

Submit

Use Case 9: “Create Organizer Account”



A screenshot of a web-based account creation form titled "Account Info". The form includes fields for Name, Email, Password, and Type. The Type field is set to "Administrator". At the bottom are "Cancel" and "Submit" buttons.

**Account Info**

**Name**

**Email**

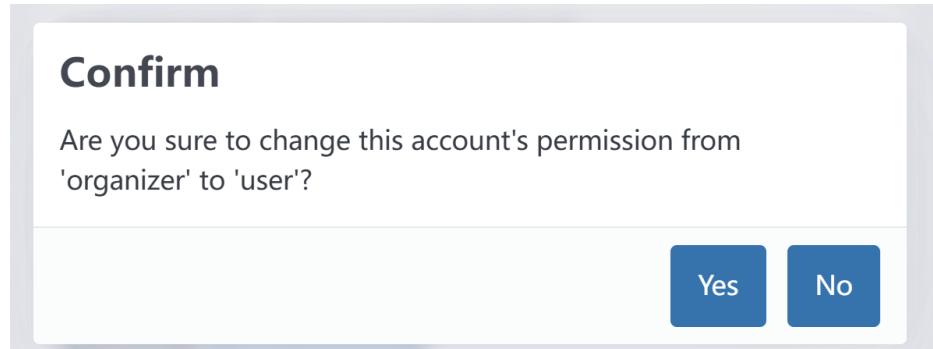
**Password**

**Type**

Administrator

Cancel      Submit

Use Case 10: "Create Administrator Account"



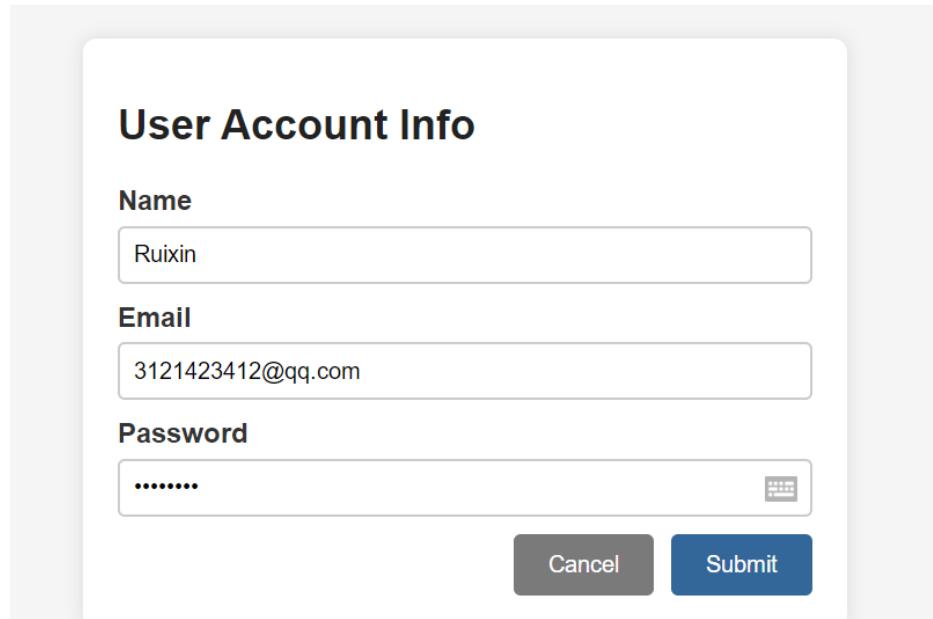
A confirmation dialog box asking if the user is sure to change an account's permission from 'organizer' to 'user'. It has "Yes" and "No" buttons.

**Confirm**

Are you sure to change this account's permission from 'organizer' to 'user'?

Yes      No

Use Case 11: "Edit Account Permission"



A screenshot of a user account editing form titled "User Account Info". It contains fields for Name, Email, and Password. The Name field is filled with "Ruixin", the Email field with "3121423412@qq.com", and the Password field shows a masked password. At the bottom are "Cancel" and "Submit" buttons.

**User Account Info**

**Name**

Ruixin

**Email**

3121423412@qq.com

**Password**

.....

Cancel      Submit

Use Case 12: "Edit User Account"

## Organizer Account Info

Name

Email

Password



Address

Company Name

Phone Number

CancelSubmit

Use Case 13: “Edit Organizer Account”

## Administrator Account Info

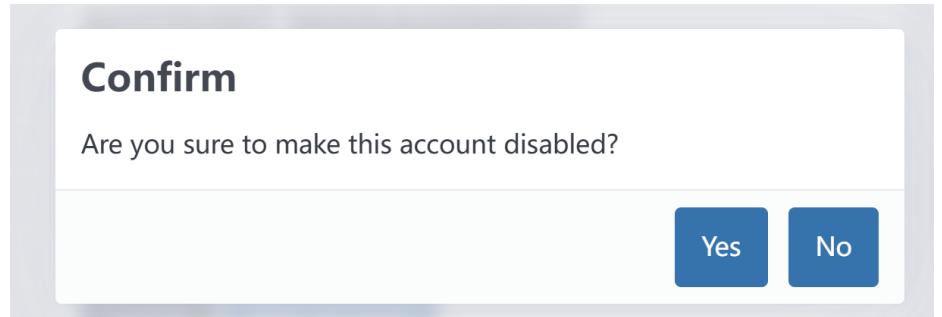
Name

Email

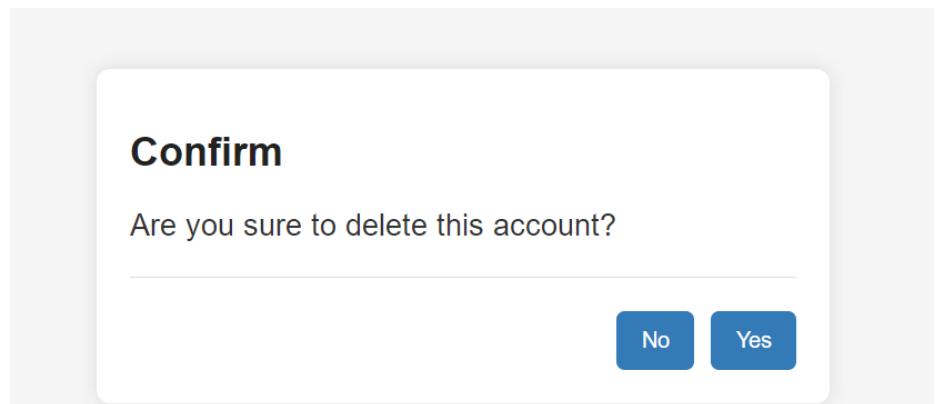
Password

CancelSubmit

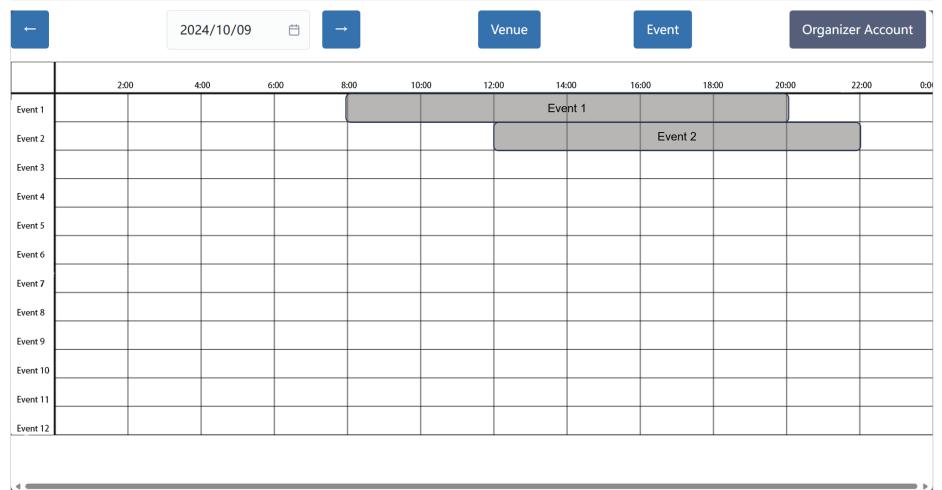
Use Case 14: “Edit Administrator Account”



Use Case 15: "Disable Account"



Use Case 16: "Delete Account"



Use Case 17: "View All Events (Organizer)"

## Available Events

**Event 1** 01/01/2024 123 Main St 50 people reserved [Delete](#) [Edit](#)

**Event 2** 02/01/2024 456 Broadway 100 people reserved [Delete](#) [Edit](#)

**Event 3** 03/01/2024 789 Market St 75 people reserved [Delete](#) [Edit](#)

**Event 4** 04/01/2024 987 Center St 150 people reserved [Delete](#) [Edit](#)

+

Use Case 18: "List Events"

**Add Event**

Event Name	<input type="text" value="Event Name"/>
Event Address	<input type="text" value="Event Address"/>
Date	<input type="text" value="yyyy/mm/日"/> 
Time	<input type="text" value="--:--"/> 
Event Venue	<input type="text" value="Event Venue"/>
<b>Local Contact Details</b>	
Organizer Name	<input type="text" value="Organizer Name"/>
Phone Number	<input type="text" value="Phone Number"/>
Email	<input type="text" value="Email"/>
<b>Tickets Type Available:</b>	
Ticket Types	<input type="text" value="Standing"/> 
Total Capacity	<input type="text"/>
Ticket Price	<input type="text"/>
Number of Tickets Available	<input type="text"/>

**Buttons:**

[Cancel](#) [Submit](#)

Use Case 19: "Create Event"

**Edit Event**

Event Name  
Event 1

Event Address  
123 Main st

Date  
yyyy/mm/dd

Time  
15:44

Event Venue  
Venue 1

**Local Contact Details**

Organizer Name  
Adam

Phone Number  
12345678

Email  
Adam@example.com

**Tickets Type Available:**

Ticket Types  
Standing

Total Capacity  
3

Ticket Price  
3

Number of Tickets Available  
3

**Cancel Changes** **Save Changes**

Use Case 20: “Edit Event”

**Confirm**

Are you sure to delete this event?

**Yes** **No**

Use Case 21: “Delete Event”

## Event Details

**Event Name:** Annual Conference

**Description:** This is a description about the selected event.

**Date:** 03/01/2024

**Time:** 10:00 AM - 4:00 PM

**Total Reservations:** 150 people

**Venue:** Convention Center, 123 Main St

**Organiser:** John Doe, john.doe@example.com

### Ticket Details

<b>Ticket Type:</b> Standing	<b>Price:</b> \$50	<b>Available:</b> 100	<b>Sold:</b> 75
------------------------------	--------------------	-----------------------	-----------------

<b>Ticket Type:</b> Seating	<b>Price:</b> \$80	<b>Available:</b> 50	<b>Sold:</b> 40
-----------------------------	--------------------	----------------------	-----------------

<b>Ticket Type:</b> Premium	<b>Price:</b> \$120	<b>Available:</b> 20	<b>Sold:</b> 15
-----------------------------	---------------------	----------------------	-----------------

Use Case 22: "View Event Details"

## Event Details

**Event Name:** Annual Conference

**Description:** This is a description about the selected event.

**Date:** 03/01/2024

**Time:** 10:00 AM - 4:00 PM

**Total Reservations:** 150 people

**Venue:** Convention Center, 123 Main St

**Organiser:** John Doe, john.doe@example.com

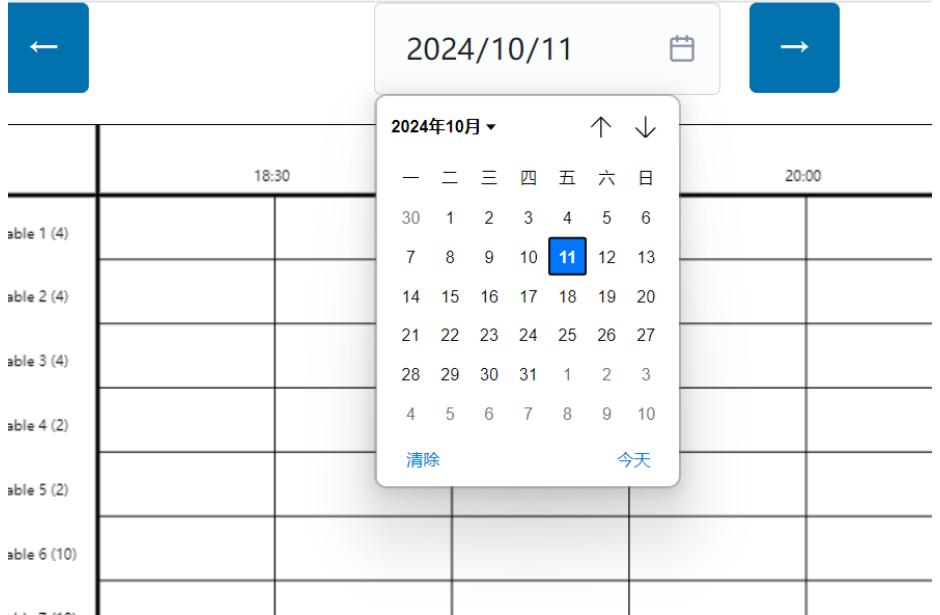
### Ticket Details

<b>Ticket Type:</b> Standing	<b>Price:</b> \$50	<b>Available:</b> 100	<b>Sold:</b> 75
------------------------------	--------------------	-----------------------	-----------------

<b>Ticket Type:</b> Seating	<b>Price:</b> \$80	<b>Available:</b> 50	<b>Sold:</b> 40
-----------------------------	--------------------	----------------------	-----------------

<b>Ticket Type:</b> Premium	<b>Price:</b> \$120	<b>Available:</b> 20	<b>Sold:</b> 15
-----------------------------	---------------------	----------------------	-----------------

Use Case 23: "View Event Registrations"



Use Case 24: “Display Date”

## Available Venues

Venue 1	123 Main St	50 people reserved	<a href="#">Edit</a>
Venue 2	456 Broadway	100 people reserved	<a href="#">Edit</a>
Venue 3	789 Market St	75 people reserved	<a href="#">Edit</a>
Venue 4	987 Center St	150 people reserved	<a href="#">Edit</a>

+

Use Case 25: “List Venues”

## Add Venue

Venue Name

Venue Address

### Local Contact Details

Name

Phone Number

Email

### Tickets Type Available:

Total Capacity

Ticket Types

Number of Tickets Available

Use Case 26: "Create Venue"

## Edit Venue

Venue Name

Venue Address

### Local Contact Details

Name

Phone Number

Email

### Tickets Type Available:

Total Capacity

Ticket Types

▼

Number of Tickets Available

Use Case 27: "Edit Venue"

## Venue Details

**Venue Name:** Grand Convention Center

**Address:** 456 City Road, Business District

### Local Contact Details

**Contact Name:** Jane Smith

**Phone Number:** 123-456-7890

**Email:** jane.smith@example.com

**Total Capacity:** 2000

### Ticket Breakdown

**Ticket Type:** Standing

**Available:** 1000

**Ticket Type:** Seating

**Available:** 700

**Ticket Type:** Premium

**Available:** 300

**Note:** Venues can only be removed from the system by administrators.

Use Case 28: "View Venue Details"

## Venue 1

### History

Event 3 03/01/2024 [Delete](#)

Event 2 02/01/2024 [Delete](#)

Event 1 01/01/2024 [Delete](#)

[Cancel](#)

[Submit](#)

Use Case 29: "View Venue Bookings"

**Organizer Account**

Name

Address

Company Name

Email

Password

Phone Number

**Past Events**

<b>Event 1</b>	01/10/2023	123 Main St	100 people reserved
<b>Event 2</b>	15/09/2023	456 Broadway Ave	150 people reserved

**Future Events**

<b>Event 3</b>	03/01/2024	789 Market St	75 people reserved
<b>Event 4</b>	10/02/2024	101 Elm St	120 people reserved

**Edit**

Use Case 30: "View Organizer Account"

**Organizer Account**

Name  
Adam

Address  
789 st

Company Name  
Company 1

Email  
Admin01

Password  
\*\*\*\*\*

Phone Number  
1234567

**Past Events**

**Event 1** 01/10/2023 123 Main St 100 people reserved [Edit](#)

**Event 2** 15/09/2023 456 Broadway Ave 150 people reserved [Edit](#)

**Future Events**

**Event 3** 03/01/2024 789 Market St 75 people reserved [Edit](#)

**Event 4** 10/02/2024 101 Elm St 120 people reserved [Edit](#)

**Save Changes**

Use Case 31: “Update Organizer Account”

## Account Info

**Name**

**Email**

**Password**

**Type**

User

**Create**

**Cancel**

Use Case 32: “Create User Account (User)”

← 2024/10/11 →

2024年10月 ↑ ↓

30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

清除 今天

18:30 20:00

able 1 (4)

able 2 (4)

able 3 (4)

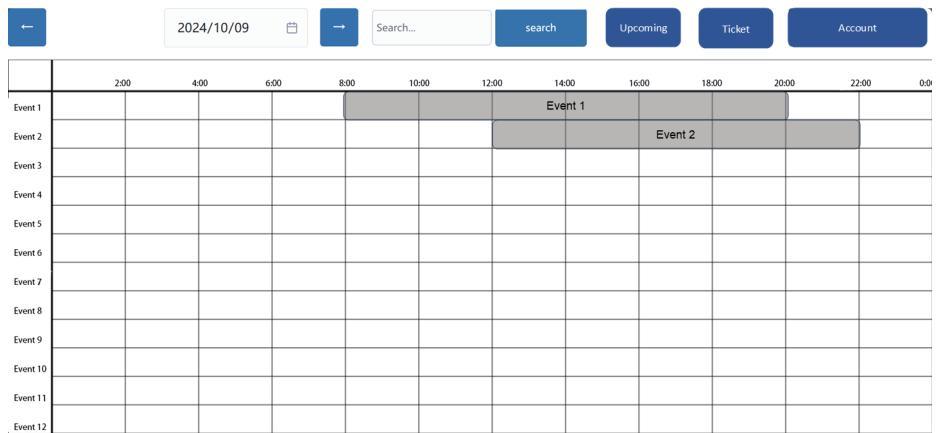
able 4 (2)

able 5 (2)

able 6 (10)

... - . . .

Use Case 33: “Display Date”



Use Case 34: “View All Events”

### User Account Info

**Name**

**Email**

**Password**

Cancel
Submit

Use Case 35: “View User Account”

Available Events		
<b>Event 1</b>	01/01/2024 123 Main St 50 people reserved	<a href="#">Buy it!</a>
<b>Event 2</b>	02/01/2024 456 Broadway 100 people reserved	<a href="#">Buy it!</a>
<b>Event 3</b>	03/01/2024 789 Market St 75 people reserved	<a href="#">Buy it!</a>
<b>Event 4</b>	04/01/2024 987 Center St 150 people reserved	<a href="#">Buy it!</a>
<a href="#">+</a>		

Use Case 36: “List Upcoming Events”

## Event Details

**Event Name:** Annual Conference

**Description:** This is a description about the selected event.

**Date:** 03/01/2024

**Time:** 10:00 AM - 4:00 PM

**Total Reservations:** 150 people

**Venue:** Convention Center, 123 Main St

**Organiser:** John Doe, john.doe@example.com

### Ticket Details

<b>Ticket Type:</b> Standing	<b>Price:</b> \$50	<b>Available:</b> 100	<b>Sold:</b> 75
------------------------------	--------------------	-----------------------	-----------------

<b>Ticket Type:</b> Seating	<b>Price:</b> \$80	<b>Available:</b> 50	<b>Sold:</b> 40
-----------------------------	--------------------	----------------------	-----------------

<b>Ticket Type:</b> Premium	<b>Price:</b> \$120	<b>Available:</b> 20	<b>Sold:</b> 15
-----------------------------	---------------------	----------------------	-----------------

Use Case 37: "View Event Details (User)"

search

Use Case 38: "Search and Filter Events"

## Event Details

**Event Name:** Annual Conference

**Description:** This is a description about the selected event.

**Date:** 03/01/2024

**Time:** 10:00 AM - 4:00 PM

**Total Reservations:** 150 people

**Venue:** Convention Center, 123 Main St

**Organiser:** John Doe, john.doe@example.com

### Ticket Details

<b>Ticket Type:</b>	<b>Standing</b>	<b>Price:</b>	<b>Available:</b>	<b>Sold:</b>
---------------------	-----------------	---------------	-------------------	--------------

<b>Ticket Type:</b>	Seating	<b>Price:</b>	\$80	<b>Available:</b>	50	<b>Sold:</b>	40
---------------------	---------	---------------	------	-------------------	----	--------------	----

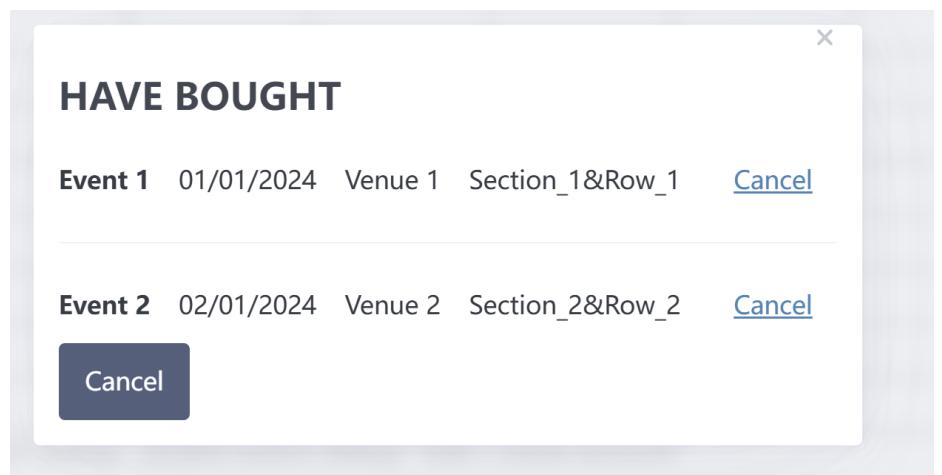
<b>Ticket Type:</b>	Premium	<b>Price:</b>	\$120	<b>Available:</b>	20	<b>Sold:</b>	15
---------------------	---------	---------------	-------	-------------------	----	--------------	----

Ticket Types

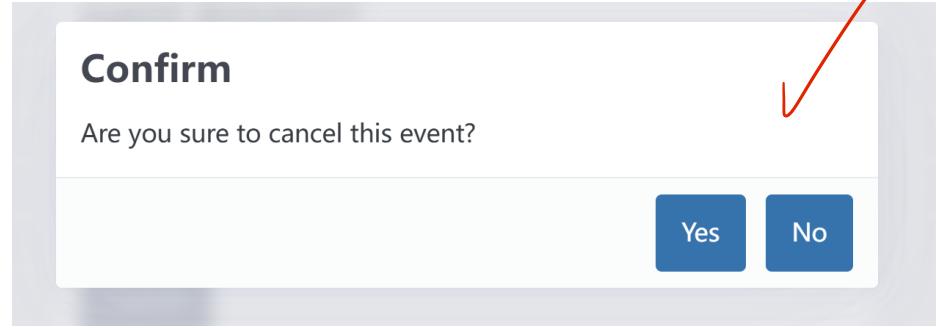
Standing

Buy it!

Use Case 39: "Purchase Ticket"



Use Case 40: "View Purchased Tickets"



Use Case 41: "Cancel Ticket"

## Task Allocation

Item	Primary Author	Contributor	Contributor	Reviewer
Use Case Diagram	TM4			TM2
Domain Model	TM2	TM3	TM1	TM4
System Golssary	TM4			TM2
UI Prototypes	TM5&6	TM2	TM1	TM1
Use Case 1: "View All Events"	TM1	TM2		TM1
Use Case 2: "View Events History"	TM8	TM2		TM1
Use Case 3: "Delete Event"	TM2			TM1
Use Case 4: "View All Venues"	TM8	TM2		TM1
Use Case 5: "View Venue Usage History"	TM8	TM2		TM1
Use Case 6: "Delete Venue"	TM8	TM2		TM1
Use Case 7: "View All Accounts"	TM8	TM2		TM1
Use Case 8: "Create User Account"	TM2			TM1
Use Case 9: "Create Organizer Account"	TM2			TM1
Use Case 10: "Create Administrator Account"	TM2			TM1
Use Case 11: "Edit Account Permission"	TM8	TM2		TM1
Use Case 12: "Edit User Account"	TM8	TM2		TM1
Use Case 13: "Edit Organizer Account"	TM8	TM2		TM1
Use Case 14: "Edit Administrator Account"	TM2			TM1
Use Case 15: "Disable Account"	TM8	TM2		TM1
Use Case 16: "Delete Account"	TM2			TM1
Use Case 17: "View All Events"	TM1			TM2
Use Case 18: "List Events"	TM1			TM2
Use Case 19: "Create Event"	TM1			TM2
Use Case 20: "Edit Event"	TM1			TM2
Use Case 21: "Delete Event"	TM1			TM2
Use Case 22: "View Event Details"	TM1			TM2
Use Case 23: "View Event Registrations"	TM1			TM2
Use Case 24: "Display Date"	TM1			TM2
Use Case 25: "List Venues"	TM1			TM2
Use Case 26: "Create Venue"	TM1			TM2
Use Case 27: "Edit Venue"	TM1			TM2
Use Case 28: "View Venue Details"	TM1			TM2
Use Case 29: "View Venue Bookings"	TM1			TM2
Use Case 30: "View Organizer Account"	TM1			TM2
Use Case 31: "Update Organizer Account"	TM1			TM2
Use Case 32: "Create User Account"	TM7	TM1		TM2
Use Case 33: "Display Date"	TM1			TM2
Use Case 34: "View All Events"	TM1			TM2
Use Case 35: "View User Account"	TM7	TM1		TM2
Use Case 36: "List Upcoming Events"	TM7			TM2
Use Case 37: "View Event Details (User)"	TM7			TM2
Use Case 38: "Search Filter Events"	TM7			TM2
Use Case 39: "Purchase Ticket"	TM7			TM2
Use Case 40: "View Purchased Tickets"	TM7			TM2

Item	Primary Author	Contributor	Contributor	Reviewer
Use Case 41: "Cancel Ticket"	TM7			TM2

## Filename - UseCase/Administrator/01-view\_all\_events.md

### Use Case: View All Accounts

#### Description

- This use case allows the administrator to view a list of all events in the system.

#### Actors

- Administrator

#### Triggers

- The administrator decides to view all existing events in the system.

#### Preconditions

- The administrator must be logged into the system.
- There are events available in the system.

#### Postconditions

- The administrator is able to view the details of all events in the system.

#### Courses of Events

##### 01 - Basic course of events -



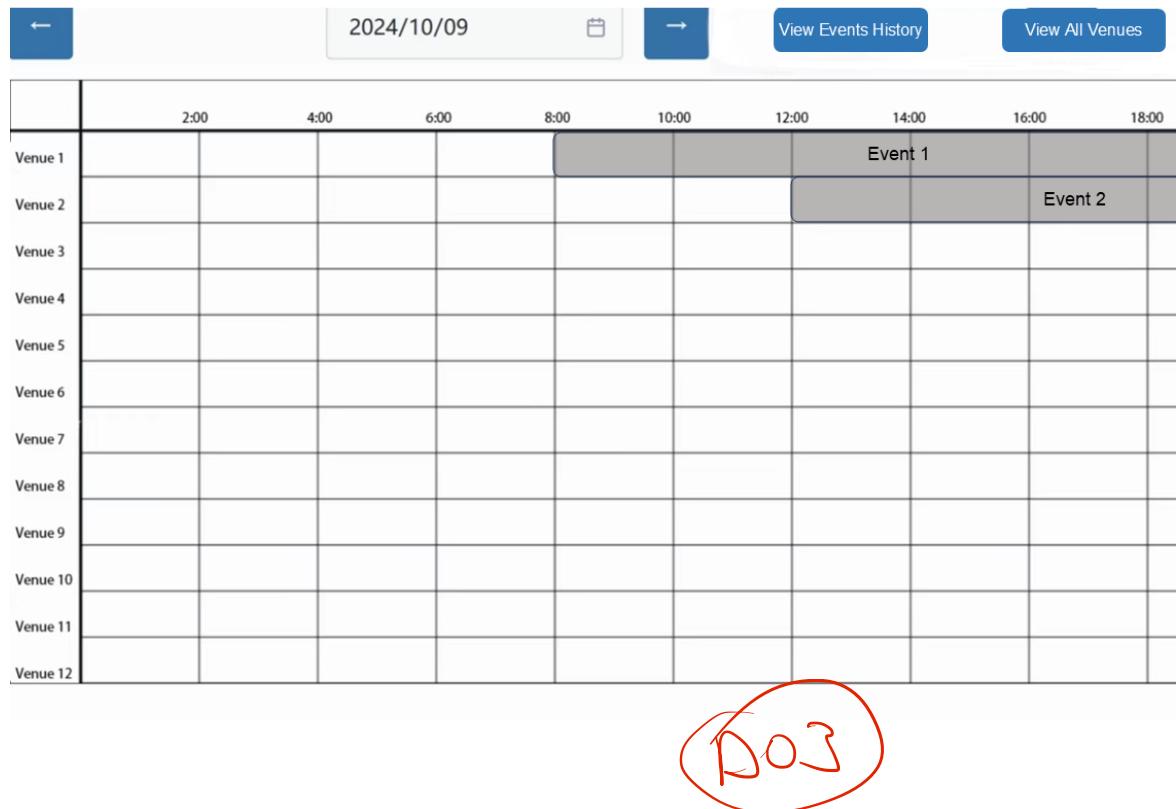
##### Course of Events

1. After logging in, the administrator is presented with the event management menu, displaying a table of all scheduled events for the current date
2. The system displays a table showing all events scheduled for the current date.
3. The administrator can see the following details in the table:
  - Event Name
  - Time Slots (e.g., 2:00, 4:00, 6:00, etc.)
  - Available Slots for Each Event
4. The administrator can use the navigation buttons (left, right) or date picker to change the date and view events scheduled on different dates.
5. If there are no events for the selected date, the system displays an empty table or a "No Events Scheduled" message.

##### 02 - Alternate course of events -

##### No Events Found

- The system shows that no events exist in the system.



Related UI Prototypes

DO3

**Filename - UseCase/Administrator/02-view\_events\_history.md**

## Use Case: View Event History

### Description

- This use case allows the administrator to view the history of events organized in the system.

### Actors

- Administrator

### Triggers

- The administrator decides to view the past events that have been organized.

### Preconditions

- The administrator must be logged into the system.
- Events must exist in the system.

### Postconditions

- The administrator can view a list of past events.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The administrator selects 'View Event History' from the event management menu.
- The system retrieves and displays a list of all events histories.

## 02 - Alternate course of events -

### No Events Found

- The system shows that there are no past events in the system.

### Related UI Prototypes



Filename - UseCase/Administrator/03-delete\_event.md

### Use Case: Delete Event

#### Description

- This use case allows an administrator to delete an existing event from the system.

#### Actors

- Administrator

#### Triggers

- The administrator decides to remove an event from the system.

#### Preconditions

- The administrator must be logged into the system.
- The event must exist.

#### Postconditions

- The event is permanently removed from the system.

### Courses of Events

#### 01 - Basic course of events -

##### Course of Events

- The administrator navigates to the 'View All Events' page, which displays all existing events in the system.
- The system displays a list of existing events.

- The administrator selects the event to delete.
- The system asks the administrator to confirm.
- The administrator selects Yes.
- The event is deleted. CEM3
- A confirmation message is shown to the administrator.

## 02 - Alternate course of events -

### Do Not Delete

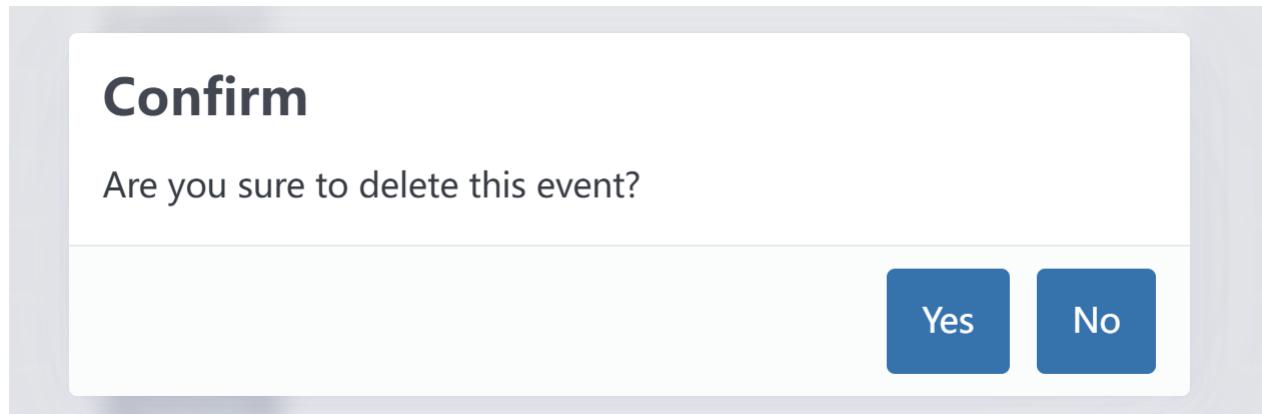
- The administrator selects an event to delete.
- The system asks the administrator to confirm.
- The administrator selects No.
- The system returns to the previous interface, and no event is deleted.



### Event Not Found

- The administrator tries to delete an event that no longer exists.
- The system displays an error message.

### Related UI Prototypes



**Filename - UseCase/Administrator/04-view\_all\_venues.md**

## Use Case: View All Venues

### Description

- This use case allows the administrator to view a list of all venues in the system.

### Actors

- Administrator

### Triggers

- The administrator decides to view all existing venues in the system.

### Preconditions

- The administrator must be logged into the system.
- There are venues available in the system.

### Postconditions

- The administrator is able to view the details of all venues in the system.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The administrator selects 'View All Venues' from the event management menu.
- The system retrieves and displays a list of all existing venues.

### 02 - Alternate course of events -

#### No Venues Found

- The system shows that no venues exist in the system.

#### Related UI Prototypes

Venue	Action	Action
Venue 1	<a href="#">View History</a>	<a href="#">Delete</a>
Venue 2	<a href="#">View History</a>	<a href="#">Delete</a>
Venue 3	<a href="#">View History</a>	<a href="#">Delete</a>
Venue 4	<a href="#">View History</a>	<a href="#">Delete</a>



Filename - UseCase/Administrator/05-view\_venue\_usage\_history.md

## Use Case: View Venue Usage History

### Description

- This use case allows the administrator to view the history of a venue usage in the system.

### Actors

- Administrator

### Triggers

- The administrator decides to view the usage history of a venue.

### Preconditions

- The administrator must be logged into the system.
- Venues must exist in the system and must have hosted events.

### Postconditions

- The administrator can view the history of events hosted at each venue.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The administrator navigates to the 'View All Venues' page, which displays all existing venues in the system.
- In the venue list, the administrator clicks the 'View History' link next to the specific venue.
- The system retrieves and displays a list of events that have been hosted at that venue.

### 02 - Alternate course of events -

#### No Usage Found

- The system shows that there is no history of events hosted at the venues.

#### Related UI Prototypes

Venue 1

History

Event 3 03/01/2024	<a href="#">Delete</a>
Event 2 02/01/2024	<a href="#">Delete</a>
Event 1 01/01/2024	<a href="#">Delete</a>

Cancel    Submit

Filename - UseCase/Administrator/06-delete\_venue.md

## Use Case: Delete Venue

### Description

- This use case involves an administrator deleting a venue from the system.

### Actors

- Administrator

### Triggers

- The administrator decides to remove a venue from the system.

### Preconditions

- The administrator must be logged into the system.
- The venue must exist in the system.

## Postconditions

- The venue is permanently removed from the system.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The administrator navigates to the 'View All Venues' page, which displays all existing venues in the system.
- The system displays a list of existing venues.
- The administrator selects the venue to delete.
- The system asks the administrator to confirm.
- The administrator selects Yes.
- ~~The venue is deleted from the system.~~ ✓
- A confirmation message is shown to the administrator.

### 02 - Alternate course of events -

#### Do Not Delete

- The administrator selects a venue to delete.
- The system asks the administrator to confirm.
- The administrator selects No.
- The system returns to the previous interface, and no venue is deleted.

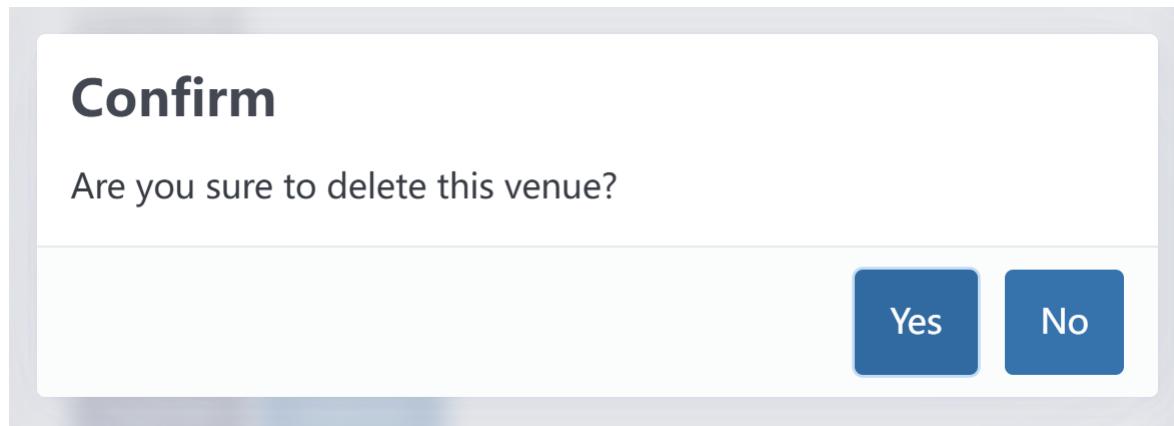
#### Venue Not Found

- The administrator tries to delete a venue that does not exist.
- The system displays an error message.

#### Venue In Use

- The administrator attempts to delete a venue that is still associated with upcoming events.
- The system displays a warning indicating that the venue is still in use and cannot be deleted.

#### Related UI Prototypes



## Filename - UseCase/Administrator/07-view\_all\_accounts.md

### Use Case: View All Accounts

#### Description

- This use case allows the administrator to view a list of all accounts (administrators, organizers, and users).

#### Actors

- Administrator

## Triggers

- The administrator decides to view all existing accounts in the system.

## Preconditions

- The administrator must be logged into the system.

## Postconditions

- The administrator is able to view the details of all accounts in the system.

## Courses of Events

### 01 - Basic course of events -

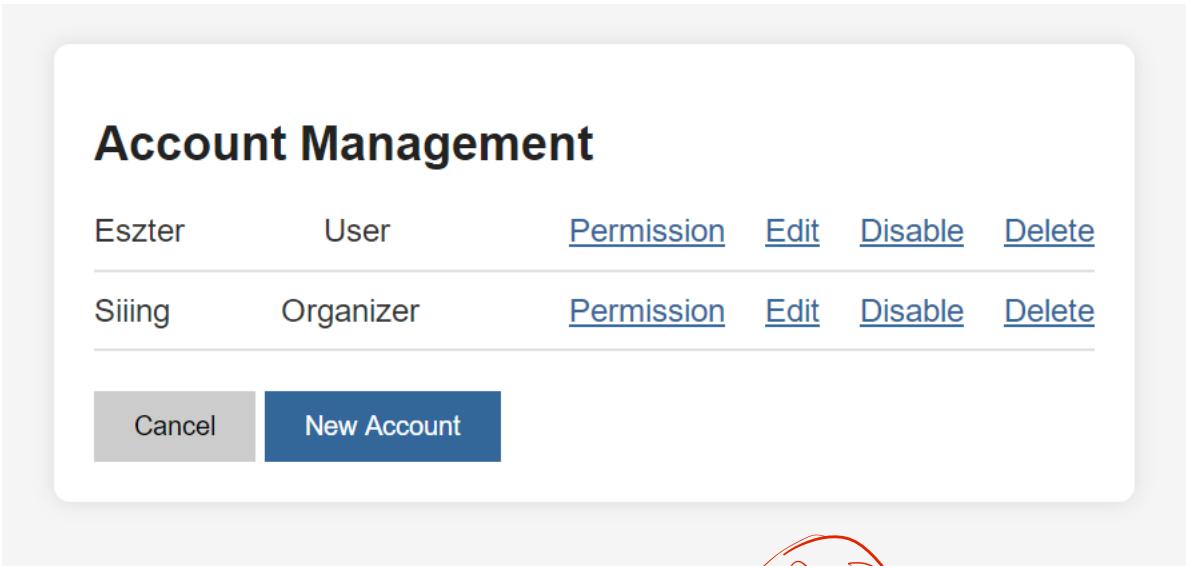
#### Course of Events

- The administrator selects 'View All Accounts' from the event management menu.
- The system retrieves and displays a list of all existing accounts (administrators, organizers, users). 

### 02 - Alternate course of events -

#### No Accounts Found

- The system shows that no accounts exist in the system.



		Permission	Edit	Disable	Delete
Eszter	User	<a href="#">Permission</a>	<a href="#">Edit</a>	<a href="#">Disable</a>	<a href="#">Delete</a>
Siiing	Organizer	<a href="#">Permission</a>	<a href="#">Edit</a>	<a href="#">Disable</a>	<a href="#">Delete</a>

## Related UI Prototypes

Do3

Filename - UseCase/Administrator/08-create\_user\_account.md

## Use Case: Edit Account Permission

### Description

- This use case allows an administrator to create a new user account in the system. The administrator can choose to create one of three types of accounts: User, Organizer, or Administrator. ## Actors
- Administrator

## Triggers

- The administrator decides to create a new user account in the system.

## Preconditions

- The administrator must be logged into the system.

## Postconditions

- A new user account is created in the system.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The administrator navigates to the 'View All Accounts' page, which displays all existing accounts in the system.
- The system displays a list of existing accounts.
- The administrator selects 'New Account'.
- The administrator selects the type of account : User.
- The system presents a form for entering account information (Name, Email, Password).
- The administrator fills out the information.
- The system validates the change and updates this new user account.
- A confirmation message is shown to the administrator.

### 02 - Alternate course of events -

#### Duplicate Email Address

- The administrator submits the form using an email that already exists in the system.
- The system displays an error message indicating that the email is already associated with another account.
- The system prompts the administrator to use a different email or to edit the existing account instead.

#### Invalid Input

- The administrator submits the form with invalid data (e.g., improper email format, password not meeting security requirements).
- The system displays an error message explaining the issue (e.g., "Invalid email format" or "Password must contain at least 8 characters").
- The system prompts the administrator to correct the input and resubmit.

## Account Info

Name

Email

Password

 ▼

Type

 ▼

Cancel Submit

#### Duplicate Email Error

### Error: Duplicate Email

The email address is already associated with another account. Please use a different email or edit the existing account.

Back

#### In-

## Error: Invalid Input

The input provided is invalid. Please ensure that the email format is correct and the password meets security requirements.

Back

valide Input Error

(DOS)

## Filename - UseCase/Administrator/09-create\_organizer\_account.md

### Use Case: Edit Account Permission

#### Description

- This use case allows an administrator to create a new organizer account in the system. The administrator can choose to create one of three types of accounts: User, Organizer, or Administrator. ## Actors
- Administrator

#### Triggers

- The administrator decides to create a new organizer account in the system.

#### Preconditions

- The administrator must be logged into the system.

#### Postconditions

- A new organizer account is created in the system.

#### Courses of Events

##### 01 - Basic course of events -

###### Course of Events

- The administrator navigates to the 'View All Accounts' page, which displays all existing accounts in the system.
- The system displays a list of existing accounts.
- The administrator selects 'New Account'.
- The administrator selects the type of account : Organizer.
- The system presents a form for entering account information (Name, Email, Password, Address, Company Name, Phone Number).
- The administrator fills out the information.
- ~~The system validates the change and updates this new organizer account.~~
- A confirmation message is shown to the administrator.

(CEW3)

#### Related UI Prototypes

- Account permission edit form

## 02 - Alternate course of events -

### Duplicate Email Address

CEW1

- The administrator submits the form using an email that already exists in the system.
- The system displays an error message indicating that the email is already associated with another account.
- The system prompts the administrator to use a different email or to edit the existing account instead. ~~#### Invalid Input~~
- The administrator submits the form with invalid data (e.g., improper email format, password not meeting security requirements).
- The system displays an error message explaining the issue (e.g., "Invalid email format" or "Password must contain at least 8 characters").
- The system prompts the administrator to correct the input and resubmit.

CEW1

CEW4

## Account Info

Name

Email

Password

Address

Company Name

Phone Number

Type

Organizer

 ▾

Cancel

Submit

#### Duplicate Email Error

### Error: Duplicate Email

The email address is already associated with another account. Please use a different email or edit the existing account.

Back

#### In-

## Error: Invalid Input

The input provided is invalid. Please ensure that the email format is correct and the password meets security requirements.

Back

valide Input Error

(Do3)

## Filename - UseCase/Administrator/10-create\_administrator\_account.md

### Use Case: Edit Account Permission

#### Description

- This use case allows an administrator to create a new administrator account in the system. The administrator can choose to create one of three types of accounts: User, Organizer, or Administrator. ## Actors
- Administrator

#### Triggers

- The administrator decides to create a new administrator account in the system.

#### Preconditions

- The administrator must be logged into the system.

#### Postconditions

- A new administrator account is created in the system.

#### Courses of Events

##### 01 - Basic course of events -

###### Course of Events

- The administrator navigates to the 'View All Accounts' page, which displays all existing accounts in the system.
- The system displays a list of existing accounts.
- The administrator selects 'New Account'.
- The administrator selects the type of account : Administrator.
- The system presents a form for entering account information (Name, Email, Password).
- The administrator fills out the information.
- The system validates the change and updates this new administrator account.
- A confirmation message is shown to the administrator.

(CEW1)

(CEW3)

###### Related UI Prototypes

- Account permission edit form

##### 02 - Alternate course of events -

###### Duplicate Email Address

- The administrator submits the form using an email that already exists in the system.
- The system displays an error message indicating that the email is already associated with another account.
- The system prompts the administrator to use a different email or to edit the existing account instead. ##### Invalid Input
- The administrator submits the form with invalid data (e.g., improper email format, password not meeting security requirements).
- The system displays an error message explaining the issue (e.g., "Invalid email format" or "Password must contain at least 8 characters").
- The system prompts the administrator to correct the input and resubmit.

## Related UI Prototypes

**Account Info**

**Name**

**Email**

**Password**

**Type**

Administrator ▾

**Cancel** **Submit**

##### Duplicate Email Error

**Error: Duplicate Email**

The email address is already associated with another account. Please use a different email or edit the existing account.

**Back**

##### In-

## Error: Invalid Input

The input provided is invalid. Please ensure that the email format is correct and the password meets security requirements.

Back

valide Input Error

(D63)

Filename - UseCase/Administrator/11-edit\_account\_permission.md

### Use Case: Edit Account Permission

#### Description

- This use case involves changing the permission level of an existing account by an administrator (e.g., promoting a user to an organizer).

#### Actors

- Administrator

#### Triggers

- The administrator decides to modify the permission level of an account.

#### Preconditions

- The administrator must be logged into the system.
- The account must exist.

#### Postconditions

- The account's permission level is updated in the system.

#### Courses of Events

##### 01 - Basic course of events -

###### Course of Events

- The administrator navigates to the 'View All Accounts' page, which displays all existing accounts in the system.
- The system displays a list of existing accounts.
- The administrator selects the account to modify.
- The system displays the account's current permission level.
- The administrator modifies the permission level (e.g., from user to organizer).
- The system validates the change and updates the account.
- A confirmation message is shown to the administrator.

(CEW3)

##### 02 - Alternate course of events -

###### Invalid Permission Level

- The administrator selects an account and attempts to assign an invalid permission level.

(CEW4)

- The system displays an error message and prompts for a valid selection.

### Account Not Found

- The administrator attempts to select an account that does not exist.
- The system displays an error message indicating the account is not found.

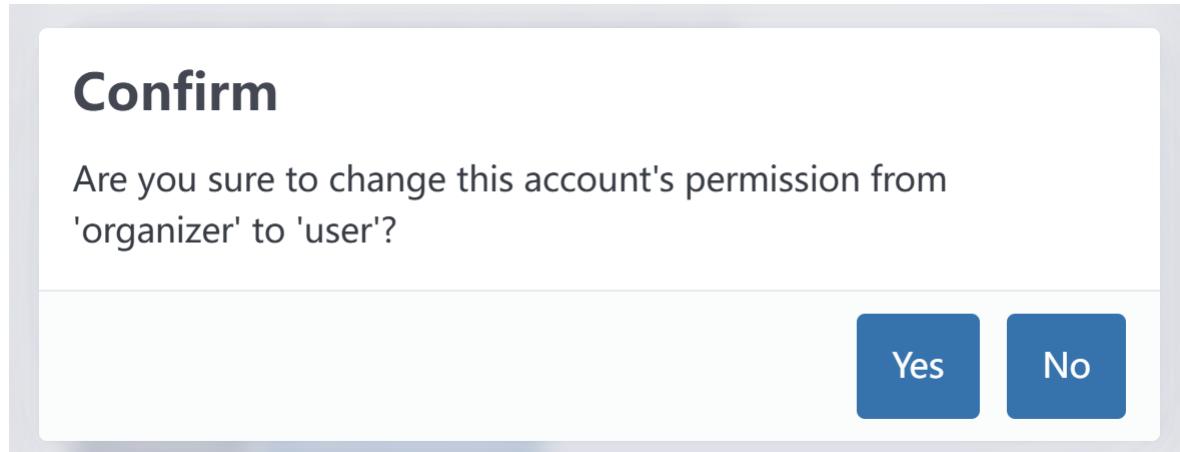
CEW4

### Permission Already Granted

- The administrator attempts to assign a permission level that the account already has.
- The system displays a message indicating no changes are needed.

CEWL

### Related UI Prototypes



**Filename - UseCase/Administrator/12-edit\_user\_account.md**

## Use Case: Edit User Account

### Description

- This use case describes the process of an administrator editing a user account.

### Actors

- Administrator

### Triggers

- The administrator decides to update user account details (e.g., name, email).

### Preconditions

- The administrator must be logged into the system.
- The user account must exist.

### Postconditions

- The user's account details are updated in the system.

### Courses of Events

#### 01 - Basic course of events -

##### Course of Events

- The administrator navigates to the 'View All Accounts' page, which displays all existing accounts in the system.
- The system displays a list of existing accounts.
- The administrator selects the account to edit.

- The system displays the user's current details.
- The administrator modifies the necessary fields.
- The system validates the changes and updates the account.
- A confirmation message is shown to the administrator.

## 02 - Alternate course of events -

### Invalid Data

- The administrator selects a user account to edit.
- The system displays the current details.
- The administrator enters invalid or incomplete information.
- The system prompts for corrections and prevents submission until valid data is provided.

### User Not Found

- The administrator tries to select a user account that does not exist.
- The system displays an error message.

### Conflict in Changes

- The administrator modifies an email or phone number, but it conflicts with another existing account's data.
- The system displays an error message indicating a conflict.
- The administrator is prompted to resolve the conflict or cancel the changes.

### Related UI Prototypes

The image shows a user interface prototype for 'User Account Info'. It features a light gray background with a white central card. The card has a title 'User Account Info' at the top. Below the title are three input fields: 'Name' with the value 'Ruixin', 'Email' with the value '3121423412@qq.com', and 'Password' with a masked value '.....'. To the right of the password field is a small icon representing a keyboard. At the bottom of the card are two buttons: 'Cancel' on the left and 'Submit' on the right. A red oval with the handwritten text 'CEW3' is drawn over the top right corner of the card.

Filename - UseCase/Administrator/13-edit\_organizer\_account.md

### Use Case: Edit Organizer Account

#### Description

- This use case allows an administrator to edit the details of an existing organizer account.

## Actors

- Administrator

## Triggers

- The administrator decides to update an organizer's information.

## Preconditions

- The administrator must be logged into the system.
- The organizer account must already exist in the system.

## Postconditions

- The organizer's account details are updated in the system.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The administrator navigates to the 'View All Accounts' page, which displays all existing accounts in the system.
- The system displays a list of existing accounts.
- The administrator selects the organizer account to edit.
- The system displays the current details of the organizer.
- The administrator modifies the desired fields.
- The system validates the changes and updates the account.
- A confirmation message is shown to the administrator.

-CEW3

### 02 - Alternate course of events -

#### Invalid Data

- The administrator selects an organizer account to edit.
- The system displays the current details of the organizer.
- The administrator enters invalid or incomplete information.
- The system prompts for corrections and prevents submission until valid data is provided.

CEW4

#### Organizer Not Found

- The administrator tries to select an organizer account that does not exist.
- The system displays an error message indicating that the account was not found.

CEW4

#### Conflict in Changes

- The administrator modifies an email or phone number, but it conflicts with another existing account's data.
- The system displays an error message indicating a conflict.
- The administrator is prompted to resolve the conflict or cancel the changes.

CEW4

## Related UI Prototypes

### Organizer Account Info

**Name**  
Yuze

**Email**  
yuzeRong666@163.com

**Password**  
.....

**Address**  
BEIJING

**Company Name**  
WSH company

**Phone Number**  
1231434153

**Buttons**  
Cancel Submit



Filename - UseCase/Administrator/14-edit\_administrator\_account.md

## Use Case: Edit User Account

### Description

- This use case describes the process of an administrator editing an administrator account.

### Actors

- Administrator

### Triggers

- The administrator decides to update administrator account details (e.g., name, email).

### Preconditions

- The administrator must be logged into the system.
- The administrator account must exist.

## Postconditions

- The administrator's account details are updated in the system.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The administrator navigates to the 'View All Accounts' page, which displays all existing accounts in the system.
- The system displays a list of existing accounts.
- The administrator selects the administrator account to edit.
- The system displays the administrator's current details.
- The administrator modifies the necessary fields.
- The system validates the changes and updates the administrator account.
- A confirmation message is shown to the administrator.

CEW3

### 02 - Alternate course of events -

#### Invalid Data

- The administrator selects an administrator account to edit.
- The system displays the current details.
- The administrator enters invalid or incomplete information.
- The system prompts for corrections and prevents submission until valid data is provided.

CEWA

#### Administrator Not Found

- The administrator tries to select an administrator account that does not exist.
- The system displays an error message.

CEWA

#### Conflict in Changes

- The administrator modifies an email or phone number, but it conflicts with another existing account's data.
- The system displays an error message indicating a conflict.
- The administrator is prompted to resolve the conflict or cancel the changes.

CEWL

### Administrator Account Info

**Name**

**Email**

**Password**

 .....

**Cancel** **Submit**

## Related UI Prototypes

## Use Case: Disable Account

### Description

- This use case allows an administrator to disable a user or organizer account, preventing the account from being used to log into the system.

### Actors

- Administrator

### Triggers

- The administrator decides to disable an account due to inactivity or a policy violation.

### Preconditions

- The administrator must be logged into the system.
- The account must exist.

### Postconditions

- The selected account is disabled, and the user cannot log in.

### Courses of Events

#### 01 - Basic course of events -

##### Course of Events

- The administrator navigates to the 'View All Accounts' page, which displays all existing accounts in the system.
- The system displays a list of existing accounts.
- The administrator selects the account to disable.
- The system asks the administrator to confirm.
- The administrator selects Yes.
- The account is disabled and cannot log in. 
- A confirmation message is shown to the administrator.

(CEN3)

#### 02 - Alternate course of events -

##### Do Not Disable

- The administrator selects an account to disable.
- The system asks the administrator to confirm.
- The administrator selects No.
- The system returns to the previous interface, and no action is taken.

(CEW4)

##### Account Already Disabled

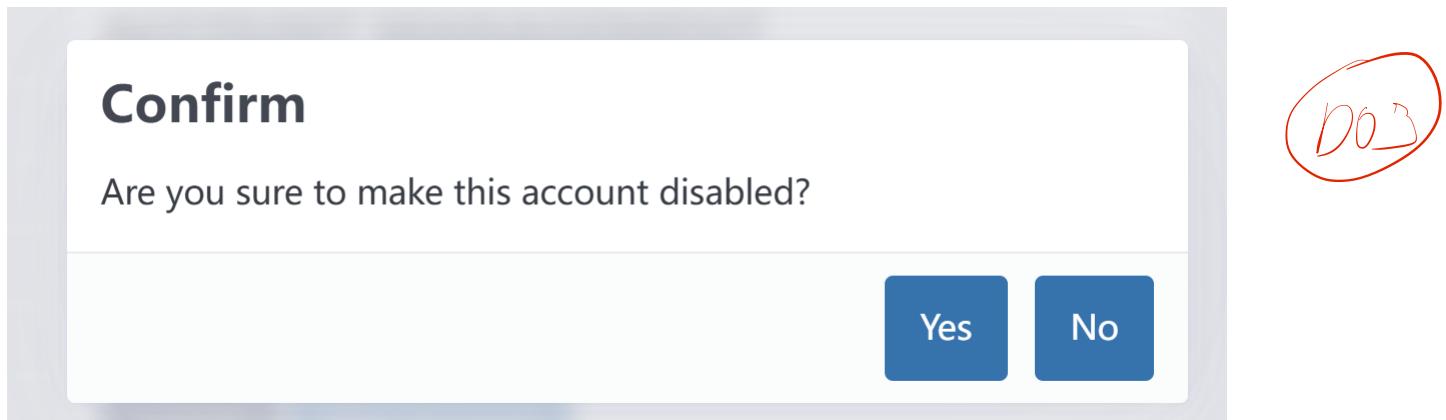
- The administrator selects an account to disable, but the account is already disabled.
- The system displays a message indicating the account is already inactive.

(CEN4)

##### Account Not Found

- The administrator tries to disable an account that does not exist.
- The system displays an error message.

(CEN4)



Filename - UseCase/Administrator/16-delete\_account.md

## Use Case: Delete Event

### Description

- This use case allows an administrator to delete an existing account from the system.

### Actors

- Administrator

### Triggers

- The administrator decides to remove an account (user, organizer or administrator) from the system.

### Preconditions

- The administrator must be logged into the system.
- The account must exist.

### Postconditions

- The account is permanently removed from the system.

### Courses of Events

#### 01 - Basic course of events -

##### Course of Events

- The administrator navigates to the 'View All Accounts' page, which displays all existing accounts in the system.
- The system displays a list of existing accounts.
- The administrator selects an account to delete.
- The system asks the administrator to confirm.
- The administrator selects Yes.
- The account is deleted. CEW3
- A confirmation message is shown to the administrator.

#### 02 - Alternate course of events -

##### Do Not Delete

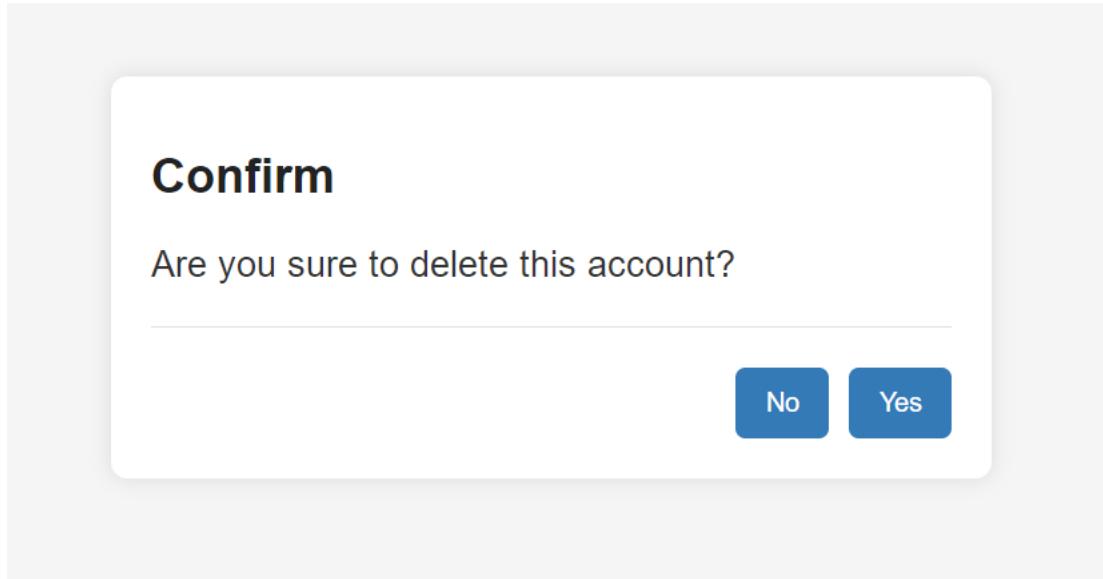
- The administrator selects an event to delete.
- The system asks the administrator to confirm.
- The administrator selects No.
- The system returns to the previous interface, and no event is deleted.

## Event Not Found

- The administrator tries to delete an event that no longer exists.
- The system displays an error message.

(CENL)

## Related UI Prototypes



(DOS)

## Filename - UseCase/Organizer/17-view\_all\_events.md

### View All Events Use Case

#### Description

This use case allows an organizer to view all events scheduled in the event management system. The events are displayed in a tabular format, showing event names and time slots for a selected date. The organizer can use navigation buttons to view events for different dates.

#### Actors

- Organizer

#### Triggers

- The organizer wants to see a complete overview of all events scheduled on a specific date.

#### Preconditions

- The organizer must be logged into the system.
- The organizer has appropriate privileges to view the events.

#### Postconditions

- The system displays all events in a tabular format for the selected date, showing time slots and event names.

#### Courses of Events

##### 01 - Basic course of events

###### Course of Events

1. After logging in, the organizer is presented with the 'View All Events' page, displaying a table of all scheduled events for the current date
2. The system displays a table showing all events scheduled for the current date.
3. The organizer can see the following details in the table:

(CEWS)

- Event Name
  - Time Slots (e.g., 2:00, 4:00, 6:00, etc.)
  - Available Slots for Each Event
4. The organizer can use the navigation buttons (left, right) or date picker to change the date and view events scheduled on different dates.
5. If there are no events for the selected date, the system displays an empty table or a “No Events Scheduled” message.

## 02 - Alternate course of events

### No Events Scheduled

- If there are no events scheduled for the selected date, the system displays an empty table and shows a message such as “No Events Scheduled” to the organizer.

### Invalid Date Selection

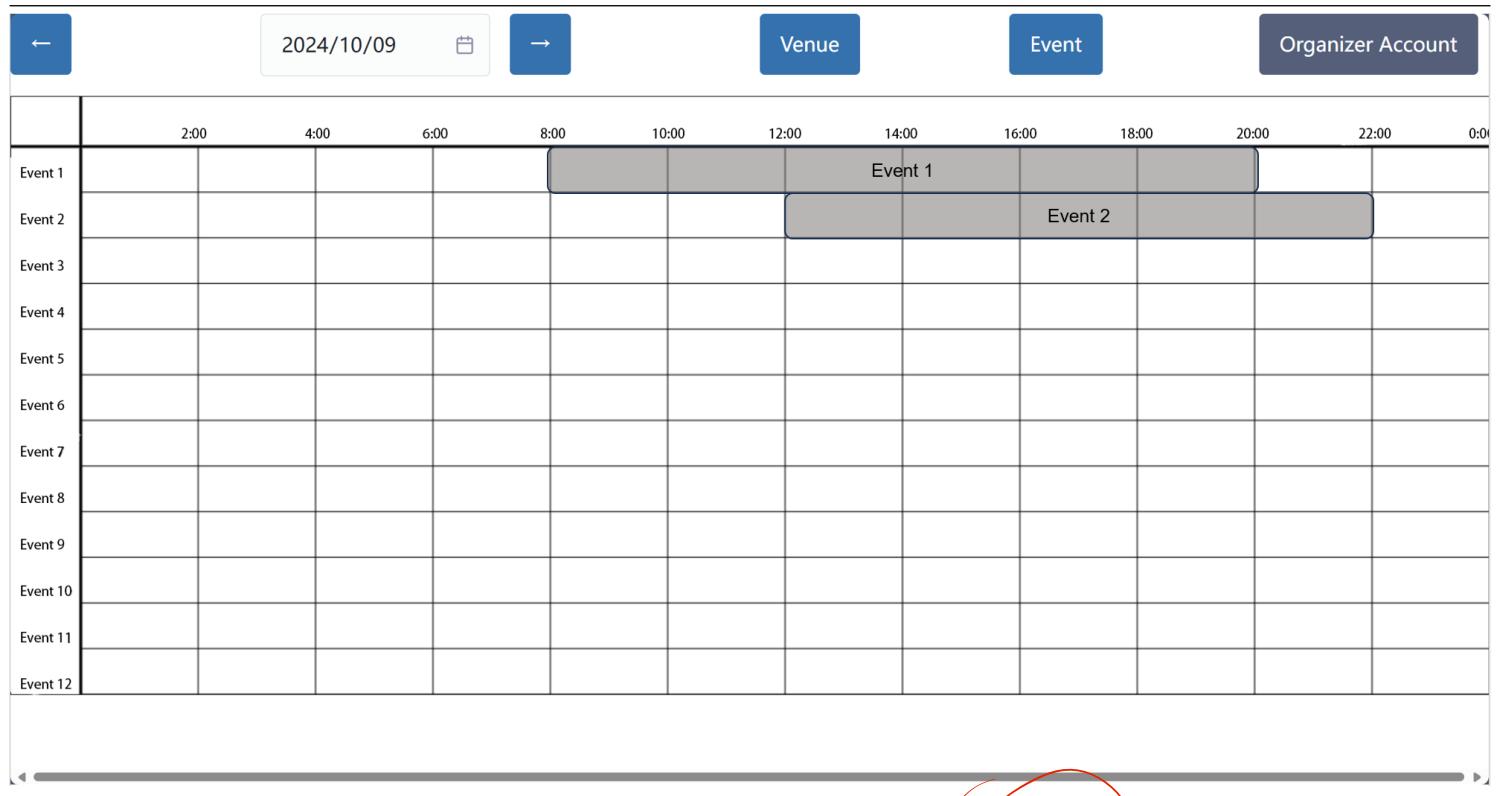
- If the selected date is invalid or out of range (e.g., a past date where the events have been deleted), the system displays an error message and prompts the organizer to select a valid date.

### Exceptions

- **System Error:** If a system error occurs while loading the events, the system displays an error message and prompts the organizer to try again later.

## Related UI Prototypes

### View All Events Page



Filename - UseCase/Organizer/18-list\_events.md

## List Event Use Case

### Description

This use case allows an organizer to view a list of events that they have created in the event management system.

## Actors

- Organizer

## Triggers

- The organizer wants to review the events they have scheduled.

## Preconditions

- The organizer must be logged into the system.
- The organizer has created at least one event.

## Postconditions

- The system displays a list of events created by the organizer.

## Courses of Events

### 01 - Basic course of events

#### Course of Events

1. The organizer selects the “List Events” option from the event management menu.
  2. The system retrieves all events created by the organizer.
- The system displays the list of events, including details such as event name, date, time, and venue.

### 02 - Alternate course of events

#### No Events Found

- If the organizer has not created any events, the system displays a message indicating that no events are available.

## Exceptions

- **System Error:** If a system error occurs while retrieving the events, the system displays an error message and prompts the organizer to try again later.

## Related UI Prototypes

## Available Events

**Event 1** 01/01/2024 123 Main St 50 people reserved [Delete](#) [Edit](#)

**Event 2** 02/01/2024 456 Broadway 100 people reserved [Delete](#) [Edit](#)

**Event 3** 03/01/2024 789 Market St 75 people reserved [Delete](#) [Edit](#)

**Event 4** 04/01/2024 987 Center St 150 people reserved [Delete](#) [Edit](#)

+

D63

Filename - UseCase/Organizer/19-create\_event.md

### Create Event Use Case

#### Description

This use case allows an organizer to create a new event in the event management system by providing event details such as name, date, time, and venue.

#### Actors

- Organizer

#### Triggers

- The organizer wants to create a new event to be hosted at a specific venue.

#### Preconditions

- The organizer must be logged into the system.
- The organizer has appropriate privileges to create an event.
- At least one venue is available in the system.

#### Postconditions

- A new event is added to the system and is available for users to view and purchase tickets.

## Courses of Events

### 01 - Basic course of events

#### Course of Events

1. The organizer selects the “Create Event” option from the event management menu.
2. The system displays a form to enter event details, including:
  - Event Name
  - Date and Time
  - Venue
  - Description
  - Ticket Information (types, prices, quantity)
3. The organizer fills in the required details and clicks “Save”.
4. The system validates the input and checks for any missing required fields.
5. If validation passes, the system stores the new event and displays a confirmation message.
6. The system updates the list of upcoming events.

~~CEN1~~

### 02 - Alternate course of events

#### Missing Required Fields

~~CEN3~~

~~CEN4~~

~~CEN4~~

#### Venue Unavailable

- If the selected venue is not available for the chosen date and time, the system displays an error message and prompts the organizer to choose a different venue or date.

#### Exceptions

- **System Error:** If a system error occurs while saving the event, the system displays an error message and prompts the organizer to try again later.

#### Related UI Prototypes

**Add Event**

Event Name

Event Address

Date  
 

Time  
 

Event Venue

**Local Contact Details**

Organizer Name

Phone Number

Email

**Tickets Type Available:**

Ticket Types  


Total Capacity

Ticket Price

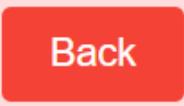
Number of Tickets Available

Error: Venue Unavailable

## Error: Venue Unavailable

The selected venue is not available for the chosen date and time.  
Please choose a different venue or adjust the event date and time.

 Back

PO3

# Filename - UseCase/Organizer/20-edit\_event.md

## Edit Event Use Case

### Description

This use case allows an organizer to edit the details of an existing event in the event management system, such as the event name, date, time, venue, or ticket information.

### Actors

- Organizer

### Triggers

- The organizer wants to update information about an event due to changes in schedule or other reasons.

### Preconditions

- The organizer must be logged into the system.
- The organizer has appropriate privileges to edit the event.
- The event must already exist in the system.

### Postconditions

- The event details are updated in the system and the changes are reflected for users viewing the event.

### Courses of Events

#### 01 - Basic course of events

##### Course of Events

1. The organizer selects the “Edit Event” option from the event management menu.
2. The system displays a list of events created by the organizer.
3. The organizer selects the event they wish to edit.
4. The system displays a form with the current event details.
5. The organizer updates the necessary fields (e.g., name, date, time, venue, ticket information).
6. The organizer clicks “Save” to confirm the changes. — (CEW3)
7. The system validates the updated input and checks for any missing required fields.
8. If validation passes, the system updates the event details and displays a confirmation message.

#### 02 - Alternate course of events

##### Missing Required Fields

- If the organizer submits the form with missing required fields (e.g., event name or date), the system highlights the missing fields and prompts the organizer to provide the necessary information. (CEW4)

##### Venue Unavailable

- If the selected venue is not available for the updated date and time, the system displays an error message and prompts the organizer to choose a different venue or date. (CEW4)

##### Exceptions

- **System Error:** If a system error occurs while saving the updated event, the system displays an error message and prompts the organizer to try again later. (CEW4)

##### Related UI Prototypes

**Edit Event**

Event Name  
Event 1

Event Address  
123 Main st

Date  
yyyy/mm/dd  
15:44

Time  
15:44

Event Venue  
Venue 1

**Local Contact Details**

Organizer Name  
Adam

Phone Number  
12345678

Email  
Adam@example.com

**Tickets Type Available:**

Ticket Types  
Standing

Total Capacity  
3

Ticket Price  
3

Number of Tickets Available  
3

**Cancel Changes** **Save Changes**

Error: Venue Unavailable

## Error: Venue Unavailable

The selected venue is not available for the updated date and time.  
Please choose a different venue



# Filename - UseCase/Organizer/21-delete\_event.md

## Delete Event Use Case

### Description

This use case allows an organizer to cancel an event that has been scheduled in the event management system.

### Actors

- Organizer

### Triggers

- The organizer decides to cancel a scheduled event, either due to unforeseen circumstances or other reasons.

### Preconditions

- The organizer can only cancel events they have created and cannot modify events created by other organizers.
- The organizer must be logged into the system.
- The organizer must have appropriate privileges to cancel the event.
- The event must exist in the system.

### Postconditions

- The event is marked as canceled and is no longer available for ticket booking.
- Notifications are sent to users who have purchased tickets for the event.

## Courses of Events

### 01 - Basic course of events

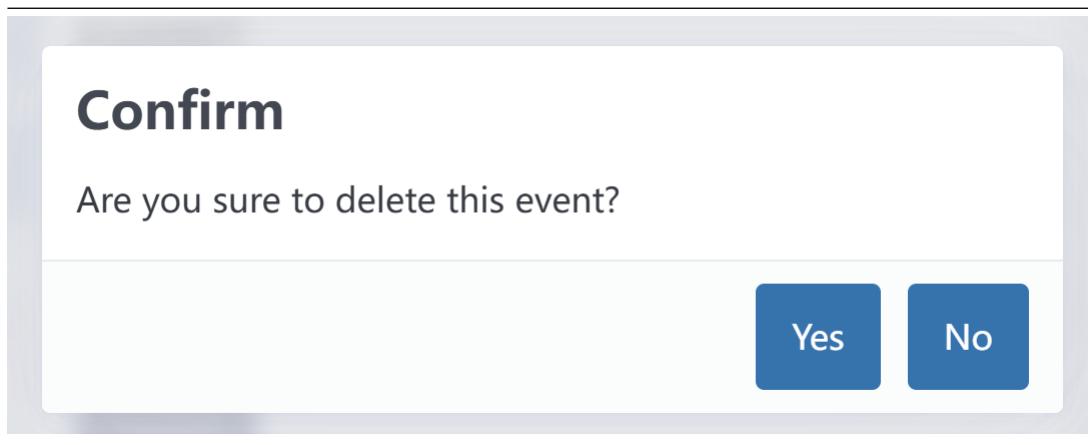
#### Course of Events

1. The organizer selects the “Delete Event” option from the event management menu.
2. The system displays a list of events that can be canceled.
3. The organizer selects the event to be canceled and confirms the cancellation.
4. The system marks the event as canceled and displays a confirmation message.
5. The system sends notifications to users who have purchased tickets for the canceled event.

#### Exceptions

- **No Permission to Cancel:** If the organizer attempts to cancel an event created by another organizer, the system displays an error message and prevents the cancellation.
- **System Error:** If a system error occurs while canceling the event, the system displays an error message and prompts the organizer to try again later.

#### Related UI Prototypes



## Filename - UseCase/Organizer/22-view\_event\_details.md

### View Event Details Use Case

#### Description

This use case allows an organizer to view detailed information about a specific event in the event management system, such as the event name, date, time, venue, description, and ticket sales status.

#### Actors

- Organizer

#### Triggers

- The organizer wants to see more details about a specific event they have created, including ticket sales information.

#### Preconditions

- The organizer must be logged into the system.
- The event must exist in the system.

#### Postconditions

- The system displays the event details for the selected event.

#### Courses of Events

##### 01 - Basic course of events

###### Course of Events

1. The organizer selects the “View Event Details” option from the list of events.
  2. The system retrieves the details of the selected event.
- The system displays the event details, including:
- Event Name
  - Date and Time
  - Venue
  - Description
  - Ticket Information (availability, types, prices, number of tickets sold, remaining tickets)
- A large red checkmark is drawn to the right of the list of details.

##### 02 - Alternate course of events

###### Event Not Found

- If the selected event does not exist, the system displays an error message and prompts the organizer to select a valid event.

CEWL

## Exceptions

- **System Error:** If a system error occurs while retrieving the event details, the system displays an error message and prompts the organizer to try again later.

## Related UI Prototypes

View Event Details Page

### Event Details

**Event Name:** Annual Conference

**Description:** This is a description about the selected event.

**Date:** 03/01/2024

**Time:** 10:00 AM - 4:00 PM

**Total Reservations:** 150 people

**Venue:** Convention Center, 123 Main St

**Organiser:** John Doe, john.doe@example.com

### Ticket Details

<b>Ticket Type:</b> Standing	<b>Price:</b> \$50	<b>Available:</b> 100	<b>Sold:</b> 75
<b>Ticket Type:</b> Seating	<b>Price:</b> \$80	<b>Available:</b> 50	<b>Sold:</b> 40
<b>Ticket Type:</b> Premium	<b>Price:</b> \$120	<b>Available:</b> 20	<b>Sold:</b> 15

DO3

Filename - UseCase/Organizer/23-view\_event\_registrations.md

## View Event Registration Use Case

### Description

This use case allows an organizer to view the registration details of a specific event, focusing on the information about users who have registered for the event, including their contact details and ticket sales.

### Actors

- Organizer

### Triggers

- The organizer wants to see the registration details for a specific event they have created, including participant information and ticket sales.

### Preconditions

- The organizer must be logged into the system.

CEW4

- The event must exist in the system.

## Postconditions

- The system displays the registration details for the selected event, including the list of registered users.

## Courses of Events

### 01 - Basic course of events

#### Course of Events

1. The organizer selects the “View Event Registration” option from the list of events.

2. The system retrieves the registration details of the selected event.

3. The system displays the registration details, including:

- Event Name
- Date and Time
- Venue
- Registered Participants (names, contact information)
- Ticket Information (availability, types, prices, number of tickets sold, remaining tickets)

### 02 - Alternate course of events

#### Event Not Found

- If the selected event does not exist, the system displays an error message and prompts the organizer to select a valid event.

#### Exceptions

- **System Error:** If a system error occurs while retrieving the registration details, the system displays an error message and prompts the organizer to try again later.

#### Related UI Prototypes

# Event Details

**Event Name:** Annual Conference

**Description:** This is a description about the selected event.

**Date:** 03/01/2024

**Time:** 10:00 AM - 4:00 PM

**Total Reservations:** 150 people

**Venue:** Convention Center, 123 Main St

**Organiser:** John Doe, john.doe@example.com

## Ticket Details

**Ticket Type:** Standing

**Price:** \$50

**Available:** 100

**Sold:** 75

**Ticket Type:** Seating

**Price:** \$80

**Available:** 50

**Sold:** 40

**Ticket Type:** Premium

**Price:** \$120

**Available:** 20

**Sold:** 15

Filename - UseCase/Organizer/24-display\_date.md

## Display Date Use Case

### Description

This use case allows an organizer to navigate to a specific date to view the current events scheduled for that date or add new events.

### Actors

- Organizer

### Triggers

- The organizer wants to:
  - View the events scheduled for a particular date
  - Create a new event for a particular date

- Modify existing events scheduled for that date

## Preconditions

- The organizer must be logged into the system.

## Postconditions

- The selected date and all events scheduled for that date will be displayed on the screen.

## Courses of Events

### 01 - Basic course of events

#### Course of Events

1. The organizer enters the date they want to view.
2. The system displays the events scheduled for that date, including details such as event name, time, and venue.

#### Related UI Prototypes

#### Inclusions

None

#### Data Outcomes

**READ** - The system retrieves the details of the existing events for the specified date.

#### Related UI Prototypes

Display Events for Date Page

	18:30
Table 1 (4)	
Table 2 (4)	
Table 3 (4)	
Table 4 (2)	
Table 5 (2)	
Table 6 (10)	

←
2024/10/11
→

↑
2024年10月
↓

清除
今天

03

## List Venues Use Case

### Description

This use case allows an organizer to view a list of venues available in the event management system for a specific date.

### Actors

- Organizer

### Triggers

- The organizer wants to review the available venues to select one for an event on a specific date.

### Preconditions

- The organizer must be logged into the system.
- There are venues available in the system.

### Postconditions

- The system displays a list of available venues for the selected date.

## Courses of Events

### 01 - Basic course of events

#### Course of Events

1. The organizer selects the “List Venues” option from the venue management menu.
2. The system prompts the organizer to enter the desired date.
3. The system retrieves all available venues for the specified date.
4. The system displays the list of venues, including details such as venue name, address, capacity, and contact information.

### 02 - Alternate course of events

#### No Venues Found

- If there are no venues available for the selected date, the system displays a message indicating that no venues are available.

#### Exceptions

- **System Error:** If a system error occurs while retrieving the venues, the system displays an error message and prompts the organizer to try again later.

### Related UI Prototypes

## Available Venues

Venue 1	123 Main St	50 people reserved	<a href="#">Edit</a>
Venue 2	456 Broadway	100 people reserved	<a href="#">Edit</a>
Venue 3	789 Market St	75 people reserved	<a href="#">Edit</a>
Venue 4	987 Center St	150 people reserved	<a href="#">Edit</a>

+



Filename - UseCase/Organizer/26-create\_venue.md

### Create Venue Use Case

#### Description

This use case allows an organizer to add a new venue to the event management system. Organizers can provide details like the name, address, and capacity of the venue.

#### Actors

- Organizer

#### Triggers

- The organizer wants to add a new venue to the system to use for future events.

#### Preconditions

- The organizer must be logged into the system.
- The organizer has appropriate privileges to add a new venue.

#### Postconditions

None

#### Courses of Events

##### 01 - Basic course of events

###### Course of Events

1. The organizer selects the “Add Venue” option from the venue management menu.

2. The system displays a form to enter venue details, including:
  - Name
  - Address
  - Capacity
  - Contact information (name, phone, email)
3. The organizer fills in the required details and clicks “Save”.
4. The system validates the input and checks for any missing required fields.
5. If validation passes, the system stores the new venue and displays a confirmation message.
6. The system updates the list of available venues.

## 02 - Alternate course of events

### Missing Required Fields

- The organizer submits the form with missing required fields (e.g., venue name or capacity).
- The system highlights the missing fields and prompts the organizer to provide the necessary information.

### Duplicate Venue

- The organizer submits the form for a venue that already exists with the same name and address.
- The system displays an error message and prevents saving the duplicate entry.

### Exceptions

- **System Error:** If a system error occurs while saving the venue, the system displays an error message and prompts the organizer to try again later.

### Related UI Prototypes

**Add Venue**

Venue Name

Venue Address

**Local Contact Details**

Name

Phone Number

Email

**Tickets Type Available:**

Total Capacity

Ticket Types

Number of Tickets Available

**Cancel** **Submit**

Error: Duplicate Venue

663

## Error: Duplicate Venue

The venue with the same name and address already exists in the system. Please use a different name or address.

**Back**

## Filename - UseCase/Organizer/27-edit\_venue.md

### Edit Venue Use Case

#### Description

This use case allows an organizer to edit the details of an existing venue in the event management system, such as the venue name, address, contact information, capacity, and ticket type availability.

#### Actors

- Organizer

#### Triggers

- The organizer wants to update information about a venue due to changes in capacity, contact information, or other reasons.

#### Preconditions

- The organizer must be logged into the system.
- The organizer has appropriate privileges to edit the venue.
- The venue must already exist in the system.

#### Postconditions

- The venue details are updated in the system, and the changes are reflected in all events using this venue.

#### Courses of Events

##### 01 - Basic course of events

###### Course of Events

1. The organizer selects the “Edit Venue” option from the venue management menu.
2. The system displays a list of venues created by the organizer.
3. The organizer selects the venue they wish to edit.
4. The system displays a form with the current venue details, including:
  - Venue Name
  - Venue Address
  - Local Contact Details (name, phone number, email)
  - Total Capacity
  - Ticket Types Available (e.g., Standing, Seated)
5. The organizer updates the necessary fields.
6. The organizer clicks “Save Changes” to confirm the modifications.
7. The system validates the updated input and checks for any missing required fields.
8. If validation passes, the system updates the venue details and displays a confirmation message.

##### 02 - Alternate course of events

###### Invalid Ticket Capacity

- If the total capacity is set to a value lower than the total tickets already sold for events at this venue, the system displays an error message and prompts the organizer to adjust the capacity or cancel certain tickets.

###### Missing Required Fields

- If the organizer submits the form with missing required fields (e.g., venue name or capacity), the system highlights the missing fields and prompts the organizer to provide the necessary information.

CEW4

## Exceptions

- **System Error:** If a system error occurs while saving the updated venue details, the system displays an error message and prompts the organizer to try again later.

## Related UI Prototypes

### Edit Venue Page

**Edit Venue**

Venue Name  
Venue 1

Venue Address  
123 Main st

**Local Contact Details**

Name  
Adam

Phone Number  
12345678

Email  
Adam@example.com

**Tickets Type Available:**

Total Capacity  
500

Ticket Types  
Standing

Number of Tickets Available  
100

**Cancel Changes** **Save Changes**

Error: Invalid Ticket Capacity

## Error: Invalid Ticket Capacity

The total capacity of the venue cannot be less than the tickets already sold for events at this venue. Please increase the capacity or manage the existing tickets before making this change.

[Back](#)

Error: Missing Required Fields

## Error: Missing Required Fields

Some required fields are missing in the venue details. Please fill in all the mandatory fields, such as venue name, address, and capacity.

[Back](#)

Filename - UseCase/Organizer/28-view\_venue\_details.md

### View Venue Details Use Case

#### Description

This use case allows an organizer to view the detailed information of a specific venue, such as the venue name, address, capacity, and contact information.

#### Actors

- Organizer

#### Triggers

- The organizer wants to see the details of a specific venue to assess its suitability for an event.

#### Preconditions

- The organizer must be logged into the system.
- The venue must exist in the system.

## Postconditions

- The system displays the detailed information of the selected venue.

## Courses of Events

### 01 - Basic course of events

#### Course of Events

1. The organizer selects the “View Venue Details” option from the venue management menu.
2. The system retrieves the details of the selected venue.  
The system displays the venue details, including:
  - Venue Name
  - Address
  - Capacity
  - Contact Information (name, phone, email)

### 02 - Alternate course of events

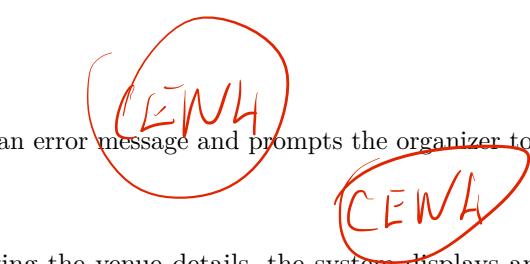
#### Venue Not Found

- If the selected venue does not exist, the system displays an error message and prompts the organizer to select a valid venue.

#### Exceptions

- **System Error:** If a system error occurs while retrieving the venue details, the system displays an error message and prompts the organizer to try again later.

#### Related UI Prototypes



## Venue Details

**Venue Name:** Grand Convention Center

**Address:** 456 City Road, Business District

### Local Contact Details

**Contact Name:** Jane Smith

**Phone Number:** 123-456-7890

**Email:** jane.smith@example.com

**Total Capacity:** 2000

### Ticket Breakdown

**Ticket Type:** Standing

**Available:** 1000

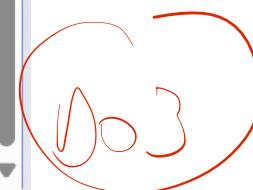
**Ticket Type:** Seating

**Available:** 700

**Ticket Type:** Premium

**Available:** 300

**Note:** Venues can only be removed from the system by administrators.



Filename - UseCase/Organizer/29-view\_venue\_bookings.md

## View Venue Bookings Use Case

### Description

This use case allows an organizer to view the booking details for a specific venue, focusing on the events scheduled at that venue, including dates, times, and event names.

### Actors

- Organizer

### Triggers

- The organizer wants to see the booking details for a specific venue to check its availability or manage events.

### Preconditions

- The organizer must be logged into the system.
- The venue must exist in the system.

### Postconditions

- The system displays the booking details for the selected venue, including all events scheduled at that venue.

## Courses of Events

### 01 - Basic course of events

#### Course of Events

1. The organizer selects the “View Venue Bookings” option from the venue management menu.
2. The system retrieves the booking details of the selected venue.
3. The system displays the booking details, including:
  - Venue Name
  - List of Events (event names, dates, and times)

### 02 - Alternate course of events

#### Venue Not Found

- If the selected venue does not exist, the system displays an error message and prompts the organizer to select a valid venue.

#### Exceptions

- **System Error:** If a system error occurs while retrieving the booking details, the system displays an error message and prompts the organizer to try again later.

#### Related UI Prototypes

---

View Venue Bookings Page

---

The UI prototype shows a modal window titled "Venue 1". Inside, there's a "History" section listing three events with "Delete" links. At the bottom are "Cancel" and "Submit" buttons.

Event	Date	Action
Event 3	03/01/2024	<a href="#">Delete</a>
Event 2	02/01/2024	<a href="#">Delete</a>
Event 1	01/01/2024	<a href="#">Delete</a>

**Cancel**   **Submit**

---

Filename - UseCase/Organizer/30-view\_organizer\_account.md

## View Organizer Account Use Case

### Description

This use case allows an organizer to view their account information, such as name, address, company name, email, and phone number. The organizer can also view a summary of their past and future events in the event management system.

### Actors

- Organizer

## Triggers

- The organizer wants to view their account information and review the list of past and future events.

## Preconditions

- The organizer must be logged into the system.
- The organizer has appropriate privileges to view their account information.

## Postconditions

- The organizer's account details and associated events are displayed in a read-only format.

## Courses of Events

### 01 - Basic course of events

#### Course of Events

1. The organizer selects the "View Account" option from the account management menu.
2. The system displays the organizer's account details, including:
  - Name
  - Address
  - Company Name
  - Email (displayed as read-only)
  - Password (displayed as masked input, e.g., "\*\*\*\*\*")
  - Phone Number

Below the account details, the system displays a summary of the organizer's past events, with the following information:

- Event Name
- Event Date
- Event Address
- Number of People Reserved

The system also displays a summary of future events in a similar format to past events.

5. The organizer can click the "Edit" button to modify their account information (handled in the "Edit Organizer Account" use case).

### 02 - Alternate course of events

#### No Events Available

- If there are no past or future events associated with the organizer, the system displays an empty state message, such as "No Events Available".

#### Exceptions

- **System Error:** If a system error occurs while retrieving the account details or event information, the system displays an error message and prompts the organizer to try again later.

## Related UI Prototypes

**Organizer Account**

Name  
Your Name

Address  
Your Address

Company Name  
Company Name

Email  
Admin01

Password  
.....

Phone Number  
Your Phone Number

**Past Events**

Event 1	01/10/2023	123 Main St	100 people reserved
Event 2	15/09/2023	456 Broadway Ave	150 people reserved

**Future Events**

Event 3	03/01/2024	789 Market St	75 people reserved
Event 4	10/02/2024	101 Elm St	120 people reserved

**Edit**

Do3

Filename - UseCase/Organizer/31-update\_organizer\_account.md

## Update Organizer Account Use Case

### Description

This use case allows an organizer to update their account information in the event management system, such as name, address, company name, email, password, and phone number. The organizer can also view a summary of their past and future events.

### Actors

- Organizer

### Triggers

- The organizer wants to update their account details or review their event history.

## Preconditions

- The organizer must be logged into the system.
- The organizer has appropriate privileges to modify their account information.

## Postconditions

- The organizer's account details are updated in the system and the changes are reflected in their profile.

## Courses of Events

### 01 - Basic course of events

#### Course of Events

1. The organizer selects the "Update Account" option from their account management menu.
2. The system displays a form with the current organizer account details, including:
  - Name
  - Address
  - Company Name
  - Email
  - Password
  - Phone Number
  - Summary of Past Events (event name, date, address, and number of people reserved)
  - Summary of Future Events (event name, date, address, and number of people reserved)
3. The organizer updates the necessary fields.
4. The organizer clicks "Save Changes" to confirm the modifications.
5. The system validates the updated input and checks for any missing required fields or invalid values.
6. If validation passes, the system updates the account details and displays a confirmation message.

### 02 - Alternate course of events

#### Missing Required Fields

- If the organizer submits the form with missing required fields (e.g., name, email), the system highlights the missing fields and prompts the organizer to provide the necessary information.

#### Invalid Email or Password

- If the email format is invalid or the password does not meet security requirements, the system displays an error message and prompts the organizer to enter a valid email or password.

#### Exceptions

- **System Error:** If a system error occurs while saving the updated account details, the system displays an error message and prompts the organizer to try again later.

## Related UI Prototypes

## Organizer Account

Name  
Adam

Address  
789 st

Company Name  
Company 1

Email  
Admin01

Password  
.....

Phone Number  
1234567

**Past Events**

**Event 1** 01/10/2023 123 Main St 100 people reserved [Edit](#)

**Event 2** 15/09/2023 456 Broadway Ave 150 people reserved [Edit](#)

**Future Events**

**Event 3** 03/01/2024 789 Market St 75 people reserved [Edit](#)

**Event 4** 10/02/2024 101 Elm St 120 people reserved [Edit](#)

**Save Changes**

Missing Required Fields

## Error: Missing Required Fields

Some required fields are missing in your account details. Please fill in all the mandatory fields, such as name, email, and phone number.

**Back**

Invalid Email or Password

## Error: Invalid Email or Password

The email format you entered is invalid, or the password does not meet the required security standards. Please enter a valid email or password.

Back

DoB

Filename - UseCase/User/32-create\_user\_account.md

### Create User Account

#### Description

- This use case allows a user to create user account in the account management.

#### Actors

- User

#### Triggers

- The user wants to create an account.

#### Preconditions

- The user's name and email cannot be duplicated.

#### Postconditions:

- The user is created into the account system.
- This user can perform normal operations such as editing information, buying tickets, etc.

#### Courses of Events

##### 01 - Basic course of events -

###### Course of Events

- CEM**
- A new user select 'Create User Account' in the account management.
  - The user fill in necessary information to create a user, including:

1. name
2. email
3. password

- CE**
- The user selects the 'Create'.
  - The system asks the user to confirm.
  - The user select Create.

- The account is created and the information will be stored in the account management.

## 02 - Alternate course of events -

### Do not Create

- (CEW1)*
- A new user select 'Create User Account' in the account management.
  - The user fill in necessary information to create a user, including:

1. name
2. email
3. password

- (CEW2)* The user selects the 'Create'.

- The system asks the user to confirm.
- The user select **Cancel**.
- No account will be created and the system returns to the previous interface.

*(CEW3)*

### Exceptions

- System Error** : If there occur some unknown errors while creating account, the system will prompt user to try again.

### Related UI Prototypes

---

Display Events for Date Page

---

### Account Info

**Name**

**Email**

**Password**

**Type**

User
 ▼

Cancel
Create

---

*(CEW4)*

*(Do3)*

Filename - UseCase/User/33-display\_date.md

## Display Date Use Case

### Description

This use case allows a user to navigate to a specific date to view the current events scheduled for that date or add new events.

## Actors

- User

## Triggers

- The user wants to:
  - View the events scheduled for a particular date
  - Create a new event for a particular date
  - Modify existing events scheduled for that date

## Preconditions

- The user must be logged into the system.

## Postconditions

- The selected date and all events scheduled for that date will be displayed on the screen.

## Courses of Events

### 01 - Basic course of events

#### Course of Events

1. The user enters the date they want to view.
2. The system displays the events scheduled for that date, including details such as event name, time, and venue.

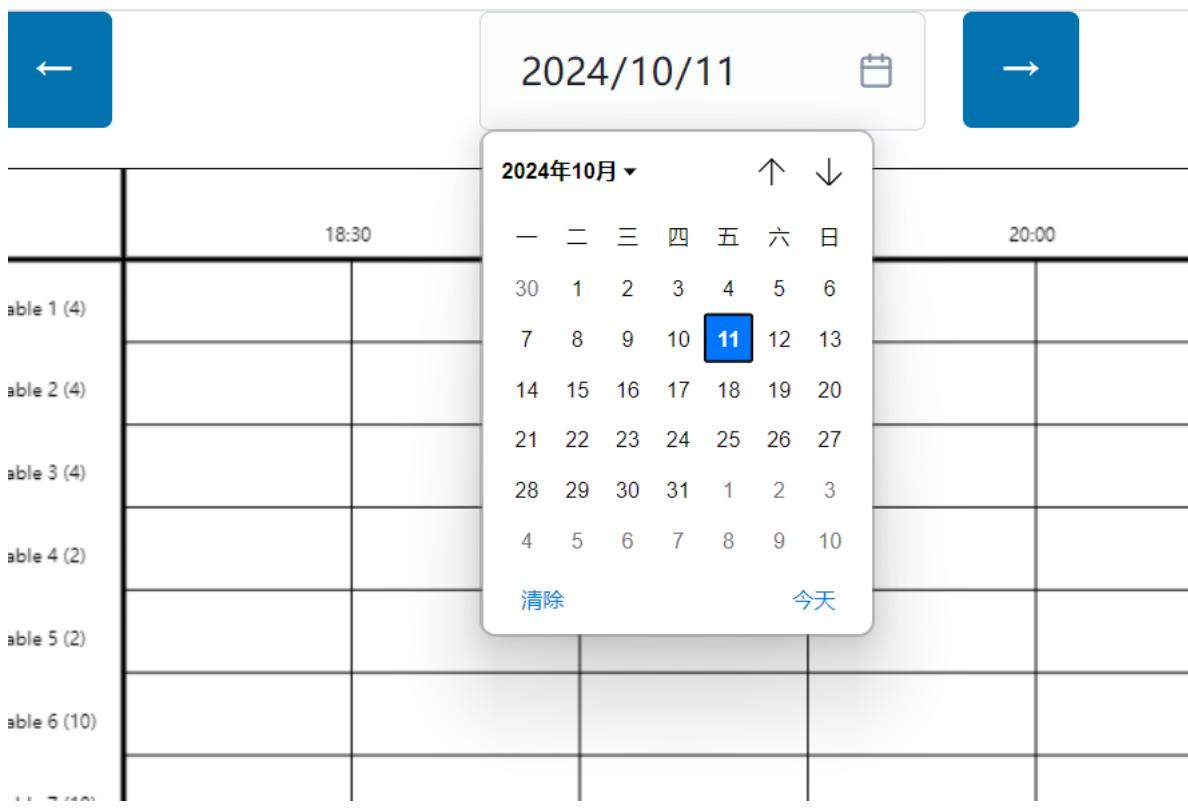
#### Inclusions

None

#### Data Outcomes

**READ** - The system retrieves the details of the existing events for the specified date.

#### Related UI Prototypes



## Filename - UseCase/User/34-view\_all\_events.md

### View All Events Use Case

#### Description

This use case allows a user to view all events scheduled in the event management system. The events are displayed in a tabular format, showing event names, time slots, and available seats for a selected date. The user can use navigation buttons to view events for different dates and filter the events based on criteria such as event type or venue.

#### Actors

- User

#### Triggers

- The user wants to see a complete overview of all events scheduled on a specific date.

#### Preconditions

- The user must be logged into the system.

#### Postconditions

- The system displays all events in a tabular format for the selected date, showing time slots, event names, and available seats.

#### Courses of Events

##### 01 - Basic course of events

###### Course of Events

1. After logging in, the user is presented with the 'View All Events' page, displaying a table of all scheduled events for the current date.

2. The system displays a table showing all events scheduled for the current date.
3. The user can see the following details in the table:
- Event Name
  - Time Slots (e.g., 2:00, 4:00, 6:00, etc.)
  - Available Seats for Each Event
  - Event Type (e.g., Concert, Sports, Theater)
  - Venue Name
4. The user can use the navigation buttons (left, right) or date picker to change the date and view events scheduled on different dates.
5. The user can also apply filters to narrow down events based on event type or venue.
6. If there are no events for the selected date, the system displays an empty table or a “No Events Scheduled” message.

CEW5

## 02 - Alternate course of events

### No Events Scheduled

- If there are no events scheduled for the selected date, the system displays an empty table and shows a message such as “No Events Scheduled” to the user.

### Invalid Date Selection

- If the selected date is invalid or out of range (e.g., a past date where the events have been deleted), the system displays an error message and prompts the user to select a valid date.

### Exceptions

- **System Error:** If a system error occurs while loading the events, the system displays an error message and prompts the user to try again later.

## Related UI Prototypes

### View All Events Page

	2:00	4:00	6:00	8:00	10:00	12:00	14:00	16:00	18:00	20:00	22:00	0:00
Event 1												
Event 2												
Event 3												
Event 4												
Event 5												
Event 6												
Event 7												
Event 8												
Event 9												
Event 10												
Event 11												
Event 12												

Do3

## View User Account

### Description

- This use case allows a user to view his/her account in the account management.

### Actors

- User

### Triggers

- The user wants to view his/her own account.

### Preconditions

- The user must be logged into the system.

### Postconditions:

- The user can view his/her own account, including: Name, password and email.

### Courses of Events

#### 01 - Basic course of events -

##### Course of Events

- The user click the 'Account' option from the account management.
- The system displays the details about his/her account.

#### 02 - Alternate course of events -

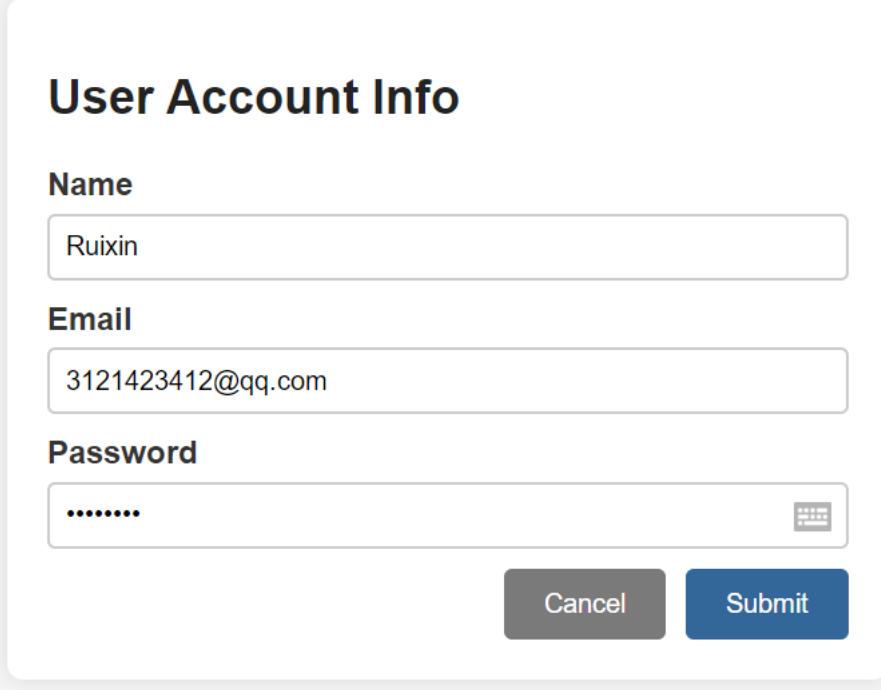
##### Edit Account

- The user click the 'Account' option from the account management.
  - The system displays the details about his/her account.
  - The user can change at most two things:
    1. Name.
    2. Password.
    3. Email.
- The user click the 'Submit' option to save the change.

##### Exceptions

- System Error : If there occur some unknown errors while viewing account, the system will prompt user to try again.

##### Related UI Prototypes



The image shows a user account information form titled "User Account Info". It contains three input fields: "Name" (Ruixin), "Email" (3121423412@qq.com), and "Password" (represented by a series of dots). Below the password field is a small icon of a keyboard. At the bottom right are two buttons: "Cancel" (gray) and "Submit" (blue).

Filename - UseCase/User/36-list\_upcoming\_events.md



## List Upcoming Events

### Description

- This use case allows a user to view upcoming events.

### Actors

- User

### Triggers

- The user wants to see the upcoming events.

### Preconditions

- The user must be logged into the system.
- The system has at least one upcoming event.

### Postconditions:

- The system displays the user's upcoming events.
- The user can select the events they want to see.
- The user can select 'Buy it' button to buy the ticket.

### Courses of Events

#### 01 - Basic course of events -

##### Course of Events

- The user select 'Upcoming' option from the user management menu.
- The system displays a list of upcoming events.

## 02 - Alternate course of events -

### Do Not Have Upcoming Events

- The user select 'Upcoming' option from the user management menu.
- The system prompts the user that there is no upcoming events.

### Exceptions

- **System Error** : If there occur some unknown errors while listing upcoming events, the system will prompt user to try again.

### Related UI Prototypes

List Upcoming Events

## Available Events

<b>Event 1</b>	01/01/2024	123 Main St	50 people reserved	<a href="#">Buy it!</a>
<b>Event 2</b>	02/01/2024	456 Broadway	100 people reserved	<a href="#">Buy it!</a>
<b>Event 3</b>	03/01/2024	789 Market St	75 people reserved	<a href="#">Buy it!</a>
<b>Event 4</b>	04/01/2024	987 Center St	150 people reserved	<a href="#">Buy it!</a>

+

**Do3**

Filename - UseCase/User/37-view\_event\_details\_user.md

## View Event Details User

### Description

- This use case allows a user to

### Actors

- User

## Triggers

- The user wants to view the details about an event.

## Preconditions

- The user must be logged into the system.
- The user needs to select one event to see its details.

## Postconditions:

- The user can see the following information:
  1. Event name, description, date, time, venue and organiser.
  2. Details about ticket:
    1. Type.
    2. Price.
    3. Number available.
    4. Number sold for each type of ticket.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The user select 'Events' option from the event management.
- The system displays a list of all events.
- The user selects an event from the list of events.
- The system displays the details about the target event.

### 02 - Alternate course of events -

#### No Events

- The user select 'Events' option from the event management.
- The system prompts that there are no events.

#### Exceptions

- **System Error :** If there occur some unknown errors while viewing event details, the system will prompt user to try again.

#### Related UI Prototypes

(CEW4)

# Event Details

**Event Name:** Annual Conference

**Description:** This is a description about the selected event.

**Date:** 03/01/2024

**Time:** 10:00 AM - 4:00 PM

**Total Reservations:** 150 people

**Venue:** Convention Center, 123 Main St

**Organiser:** John Doe, john.doe@example.com

## Ticket Details

<b>Ticket Type:</b> Standing	<b>Price:</b> \$50	<b>Available:</b> 100	<b>Sold:</b> 75
<b>Ticket Type:</b> Seating	<b>Price:</b> \$80	<b>Available:</b> 50	<b>Sold:</b> 40
<b>Ticket Type:</b> Premium	<b>Price:</b> \$120	<b>Available:</b> 20	<b>Sold:</b> 15



Filename - UseCase/User/38-search\_filter\_events.md

## Search Filter events

### Description

- This use case allows a user to search target events based on **Name**.

### Actors

- User

### Triggers

- The user wants to see events with specific names.

## Preconditions

- The user must be logged into the system.
- The user has participated in at least one event.

## Postconditions:

- The system displays events that meet the criteria for the user to view.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The user inputs the name of the event they want to search for in the filter box.
- The user click the 'Search' button.
- The system displays the search results.
- ~~The user can view the displayed event information.~~ ✓

### 02 - Alternate course of events -

#### No Matching Events

- The user inputs the name of the event they want to search for in the filter box.
- The user click the 'Search' button.
- The system prompts that no event information that meets the criteria was found.

#### Exceptions

- **System Error** : If there occur some unknown errors while searching events, the system will prompt user to try again.

#### Related UI Prototypes

---

CEW14

Do3

## Filename - UseCase/User/39-purchase\_ticket.md

## Purchase Ticket

### Description

- This use case allows a user to purchase tickets.

### Actors

- User

### Triggers

- A user wants to participant in some venues and needs to purchase the target ticket.

## Preconditions

- The user must be logged into the system.
- The user select one event to purchase its ticket.
- The user can only buy tickets for events that have not yet started.

## Postconditions:

- The number of the remaining tickets for the corresponding event minus one.
- The user can see the purchased ticket and its details.
- The user successfully purchased a ticket, which includes the following information:
  1. Event details.
  2. Venue details.
  3. The price.
  4. The type of seat.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The user select one event from the list of events.
- The user select 'Purchase' from the button of the target event interface.
- The system asks the user to confirm.
- The user select Yes.
- The user successfully purchased the ticket and the remaining number of the event minus one.
- The user can see the ticket details in the ticket system.

### 02 - Alternate course of events -

#### Insufficient Tickets.

- The user select one event from the list of events.
- The user select 'Purchase' from the button of the target event interface.
- The system asks the user to confirm.
- The user select Yes.
- But there is no tickets for sale.  
~~The system will prompt that the tickets have already been sold out. (Synchronization issue of processes)~~ ##### Purchase Timeout
- The user select one event from the list of events.
- The user select 'Purchase' from the button of the target event interface.
- The system asks the user to confirm.
- The user select No.
- ~~No tickets will be purchased~~ and the system returns to the previous interface.

## Exceptions

- System Error : If there occur some unknown errors while purchasing tickets, the system will prompt user to try again.

## Related UI Prototypes



---

Purchase Ticket

---

Purchase Ticket

---

## Event Details

**Event Name:** Annual Conference

**Description:** This is a description about the selected event.

**Date:** 03/01/2024

**Time:** 10:00 AM - 4:00 PM

**Total Reservations:** 150 people

**Venue:** Convention Center, 123 Main St

**Organiser:** John Doe, john.doe@example.com

## Ticket Details

**Ticket Type:** Standing

**Price:** \$50

**Available:** 100

**Sold:** 75

**Ticket Type:** Seating

**Price:** \$80

**Available:** 50

**Sold:** 40

**Ticket Type:** Premium

**Price:** \$120

**Available:** 20

**Sold:** 15

### Ticket Types

Standing

Buy it!

---

Purchase Ticket

Insufficient Tickets

## Error: Insufficient Tickets

The selected ticket type is sold out. Please choose a different type or try again later.

[Back](#)

Purchase Timeout

## Error: Purchase Timeout

Your payment took too long, and the event has already started. Please try again with another event.

[Back](#)



---

Filename - UseCase/User/40-view\_purchased\_tickets.md

## View Purchased Tickets

### Description

- This use case allows a user to view purchased tickets in the ticket management.

### Actors

- User

### Triggers

- The user wants to view his/her purchased tickets.

### Preconditions

- The user must be logged into the system.
- The user must have purchased tickets before.

## Postconditions:

- The system displays a list of the tickets purchased by the user.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The user click 'Purchased' in the account management.
- The system displays all the tickets purchased by the user.
- ~~The user can select a ticket from the list of purchased tickets to view.~~
- ~~The user can also view the following information:~~
  - 1. Event details.
  - 2. Venue details.
  - 3. The price.
  - 4. The type of seat.

### 02 - Alternate course of events -

#### No Purchased Tickets

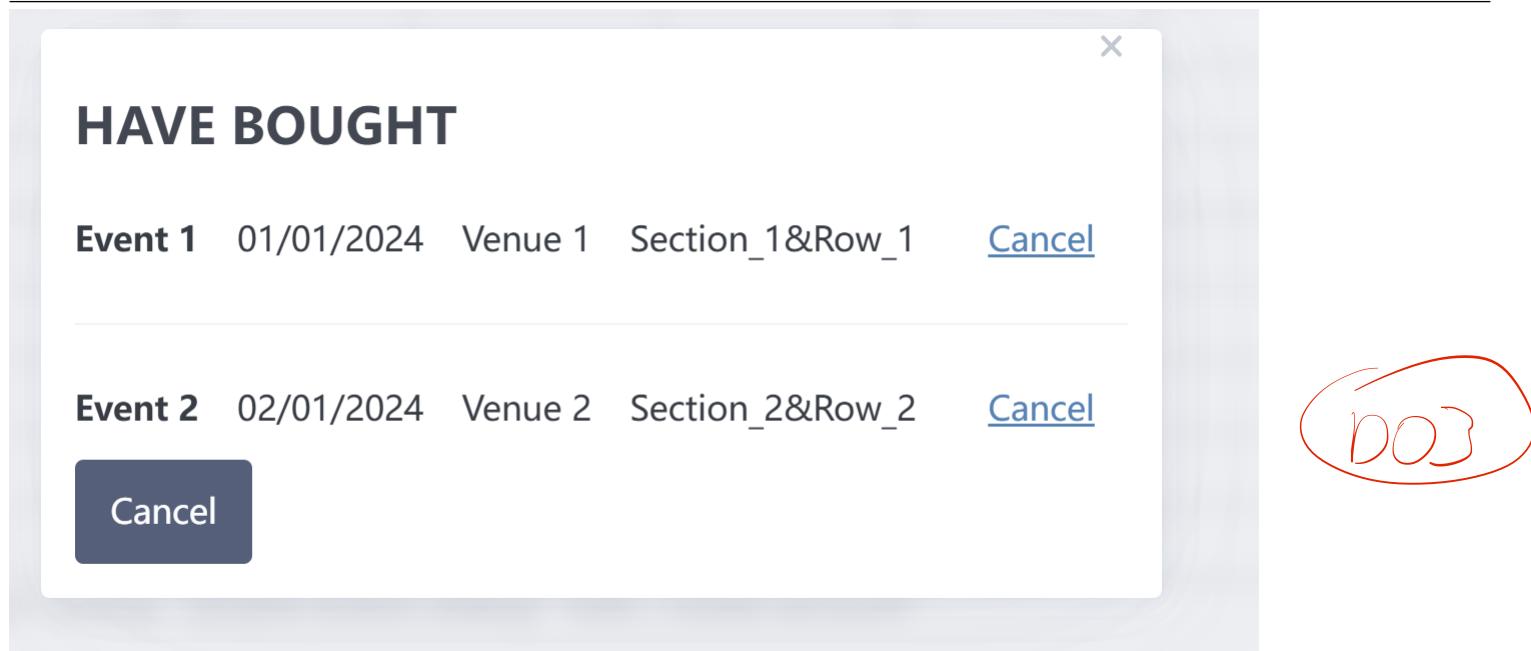
- The user click 'Purchased' in the account management.
- The system prompts that there are no purchased tickets.

## Exceptions

- **System Error :** If there occur some unknown errors while viewing purchased tickets, the system will prompt user to try again.

## Related UI Prototypes

### View Purchased Tickets



Filename - UseCase/User/41-cancel\_ticket.md

## Cancel Ticket

### Description

- This use case allows a user to cancel a ticket that has been scheduled in the ticket management.

## Actors

- User

## Triggers

- The user decides to cancel a ticket because of some reasons.

## Preconditions

- The user must be logged into the system.
- The user must cancel before the deadline of the ticket.
- The user has at least one unused ticket.

## Postconditions:

- The ticket is cancelled from the user's account.
- The remaining number of tickets for the corresponding venue will plus one.

## Courses of Events

### 01 - Basic course of events -

#### Course of Events

- The user selects 'Cancel Ticket' option from the ticket management menu.
- The system displays a collections of tickets that can be cancelled.
- The user selects a ticket to cancel.
- The system asks the user to confirm.
- The user select Yes.
- The ticket is removed from the user's account.
- The remaining number of tickets for the corresponding venue will plus one.

### 02 - Alternate course of events -

#### Do not Cancel

- The user selects 'Cancel Ticket' option from the ticket management menu.
- The system displays a collections of tickets that can be cancelled.
- The user selects a ticket to cancel.
- The system asks the user to confirm.
- The user select No.
- No tickets will be cancelled and the system returns to the previous interface.

#### Do not Have Tickets to be Cancelled.

- The system shows that there are no tickets that can be cancelled.

#### Ticket Exceeds Cancellation Deadline

- The user selects 'Cancel Ticket' option from the ticket management menu.
- The system displays a collection of tickets that can be cancelled.
- The user selects a ticket to cancel.
- The system checks the ticket's eligibility for cancellation and determines that the cancellation deadline has passed.
- The system displays a message indicating that the selected ticket cannot be cancelled because it has exceeded the cancellation deadline.
- The system returns to the previous interface without cancelling any tickets.

#### Exceptions

- System Error : If there occur some unknown errors while cancelling tickets, the system will prompt user to try again.

## Related UI Prototypes

## Confirm

Are you sure to cancel this event?

Yes

No

Ticket Exceeds Cancellation Deadline

### Error: Deletion Timeout

The selected ticket cannot be deleted as the cancellation period has expired. Please contact support if you need further assistance.

Back

DQ3