

## 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 01*

### Team Members

Number	Name	Email(s)	CSGitLab Username
TM1	Qiyue Zhu	qiyue.zhu@ucdconnect.ie	@Victoria
TM2	Wenyi Liang	wenyi.liang@ucdconnect.ie	@Wenyi
TM3	Yunfei Xu	yunfei.xu@ucdconnect.ie	@Kira210
TM4	Zitong Wang	zitong.wang@ucdconnect.ie	@sukikatte
TM5	Siqi Du	siqi.du@ucdconnect.ie	@Yuki
TM6	Jiayi Cai	jiayi.cai@ucdconnect.ie	@22207285
TM7	Yani Yang	yani.yang@ucdconnect.ie	@hs
TM8	Boran Duan	boran.duan@ucdconnect.ie	@boran

## Analysis

This phase continues the development process by realising the courses of events that we described in the requirements analysis phase. This requires the creation of sequence diagrams for each course of events. Decisions made while constructing these diagrams are also documented in the class diagram.

### Use Case Realisations (Sequence Diagrams)

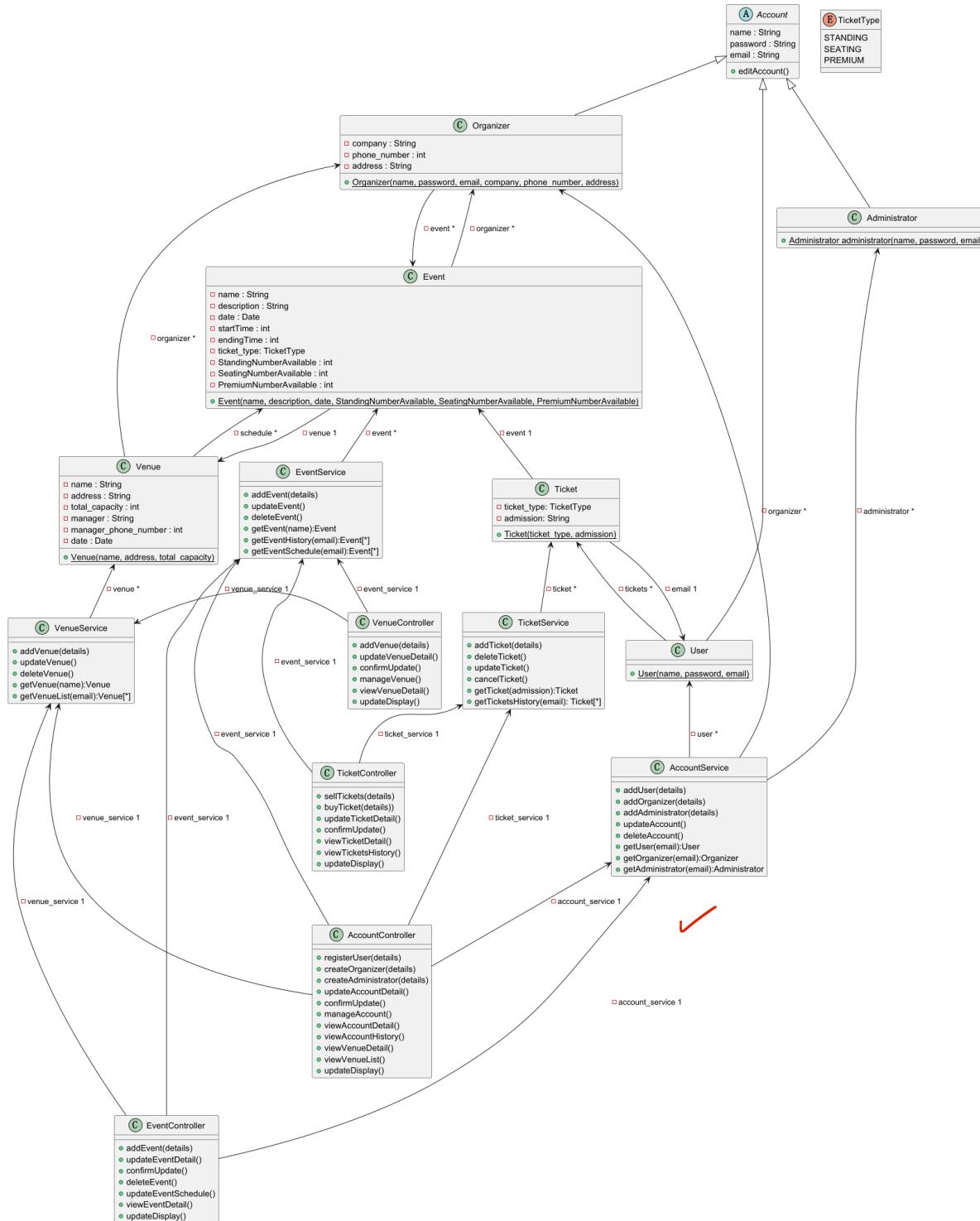
Each use case contains the separate sequence diagrams for each course of events. The diagrams are stored in the `images` folder and referenced in the markdown files.

1. Creat Account
2. List Account
3. View Account
4. Modify Account
5. Register Account
6. List Venues
7. Add Venue
8. View Venue
9. Modify Venue
10. Delete Venue
11. List Events
12. Add Event

- 13. View Event
- 14. Modify Event
- 15. Delete Event
- 16. List My Events
- 17. List Tickets
- 18. Buy Tickets
- 19. View Tickets

## **Class Diagram**

The class diagram represents the information gained about the system by completing the use case realisations.



## Description of Responsibilities

### Account

The **Account** class is an abstract entity class. It represents a user account in the system. The sole responsibility of account objects is to store the account's name, password, and email. To note, the email should be unique for every account.

### Administrator

The **Administrator** class is a concrete subclass of the **Account** class. It represents an administrator user in the system. The sole responsibility of administrator objects is to manage system-wide settings and user accounts.

### Organizer

The **Organizer** class is a concrete subclass of the **Account** class. It represents an event or a venue organizer in the system. The sole responsibility of organizer objects is to store extra information related to the organizer's company, phone number, and address. Organizers should be responsible for managing the events, venues, tickets that within their rights.

### User

The **User** class is a concrete subclass of the **Account** class. It represents a regular user in the system. The sole responsibility of user objects is to manage their tickets for events.

### Venue

The **Venue** class is a basic entity class. It represents a physical location where events are held. The sole responsibility of venue objects is to store details about the venue's name, address, total capacity, manager, and manager's phone number. To note, the name should be unique for every venue.

### Event

The **Event** class is a basic entity class. It represents an event scheduled at a venue. The sole responsibility of event objects is to store the event's name, description, date, start time, ending time, ticket type, and the number of available tickets (standing, seating, and premium). To note, the name should be unique for every event.

### Ticket

The **Ticket** class is a basic entity class. It represents a ticket purchased by a user for an event. The sole responsibility of ticket objects is to store the ticket type and admission details. To note, the admission should be unique for every ticket.

### EventController

The **EventController** class is a controller class. It is responsible for managing event-related operations, including adding, updating, confirming updates, deleting events, updating event schedules, and viewing event details.

### AccountController

The **AccountController** class is a controller class. It is responsible for managing account-related operations, including registering users, creating organizers and administrators, updating accounts, confirming updates, managing accounts, and viewing account details and histories.

## **VenueController**

The **VenueController** class is a controller class. It is responsible for managing venue-related operations, including adding, updating, confirming updates, managing venues, and viewing venue details.

## **TicketController**

The **TicketController** class is a controller class. It is responsible for managing ticket-related operations, including selling and buying tickets, updating ticket details, confirming updates, viewing ticket details, and viewing ticket history.

## **VenueService**

The **VenueService** class is a service class. It is responsible for performing operations related to venues, including adding, updating, deleting venues, retrieving venue details, and getting the schedule of events at a venue.

## **EventService**

The **EventService** class is a service class. It is responsible for performing operations related to events, including adding, updating, deleting events, and retrieving associated venues and organizers.

## **AccountService**

The **AccountService** class is a service class. It is responsible for performing operations related to accounts, including adding users, organizers, and administrators, updating accounts, deleting accounts, and retrieving account details and histories.

## **TicketService**

The **TicketService** class is a service class. It is responsible for performing operations related to tickets, including adding, deleting, updating, canceling tickets, and retrieving ticket details and histories.

## **Milestone 2 Analysis**

### **Distribution of work on this milestone**

#### **Overall Distribution of Work**

Team Member	@Victoria	@Wenyi	@Kira210	@sukikatte	@Yuki	@22207285	@hs	@boran
Percentage	12%	12%	12%	12%	12%	12%	12%	12%

#### **Task Allocation**

Item	Contributor 1	Contributor 2	Reviewer
Description of Responsibilities	@Victoria(50%)	@hs(50%)	Other Members
Class Diagram	@Victoria(50%)	@hs(50%)	Other Members
Use Case 1: Create Account	@Kira210(100%)		Other Members
Use Case 2: List Account	@Kira210(100%)		Other Members
Use Case 3: View Account	@Wenyi(100%)		Other Members

Item	Contributor 1	Contributor 2	Reviewer
Use Case 4: Modify Account	@Wenyi(100%)		Other Members
Use Case 5: Register Account	@Wenyi(100%)		Other Members
Use Case 6: List Venues	@sukikatte(100%)		Other Members
Use Case 7: Add Venue	@boran(100%)		Other Members
Use Case 8: View Venue	@Kira210(100%)		Other Members
Use Case 9: Modify Venue	@boran(100%)		Other Members
Use Case 10: "Delete Venue"	@22207285 (100%)		Other Members
Use Case 11: List Events	@sukikatte(100%)		Other Members
Use Case 12: Add Event	@boran(100%)		Other Members
Use Case 13: "View Event"	@22207285 (100%)		Other Members
Use Case 14: Modify Event	@boran(100%)		Other Members
Use Case 15: "Delete Event"	@22207285 (100%)		Other Members
Use Case 16: List My Event	@sukikatte(100%)		Other Members
Use Case 17: List Tickets	@Yuki(100%)		Other Members
Use Case 18: Buy Tickets	@Yuki(100%)		Other Members
Use Case 19: View Tickets	@Yuki(100%)		Other Members

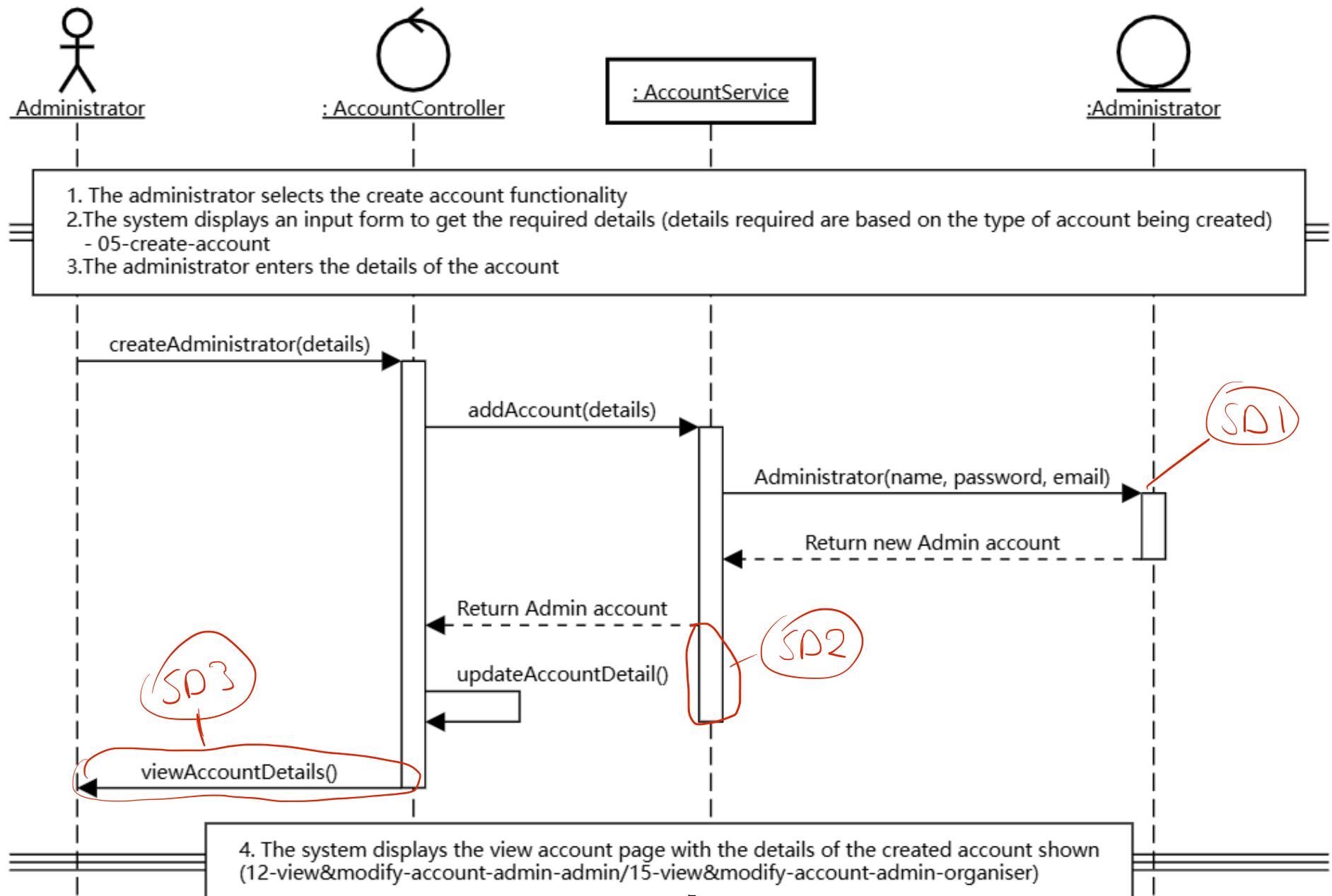
## Reflection Statements

Team	Member	Contribution	Reflection Statement
	@hs		Participated in class diagramming and writing of description of responsibility. Overview other members' usecase.
	@Boran		Drew the sequence diagrams and wrote docs of use case 07, 09, 12 and 14 about its basic event and alternate events.
	@sukikatte		Drew the sequence diagramS and wrote docs of use case 06, 11 and 16 about its basic event and alternate events.
	@Kira210		Drew the sequence diagramS and wrote docs of use case 01, 02 and 08 about its basic event and alternate events.
	@Yuki		Drew the sequence diagramS and wrote docs of use case 17, 18 and 19 about its basic event and alternate events.
	@Wenyi		Drew the sequence diagrams and wrote docs of use case 03, 04 and 05 about its basic event and alternate events.
	@Victoria		Participated in class diagramming, writing and modifying the description of responsibility and modify the domain model diagram accordingly. Reviewing the work of other members and providing suggestions. Following up on the team's progress.
	@22207285		Drew the sequence diagrams and wrote docs of use case 10, 13 and 15 about its basic event and alternate events.

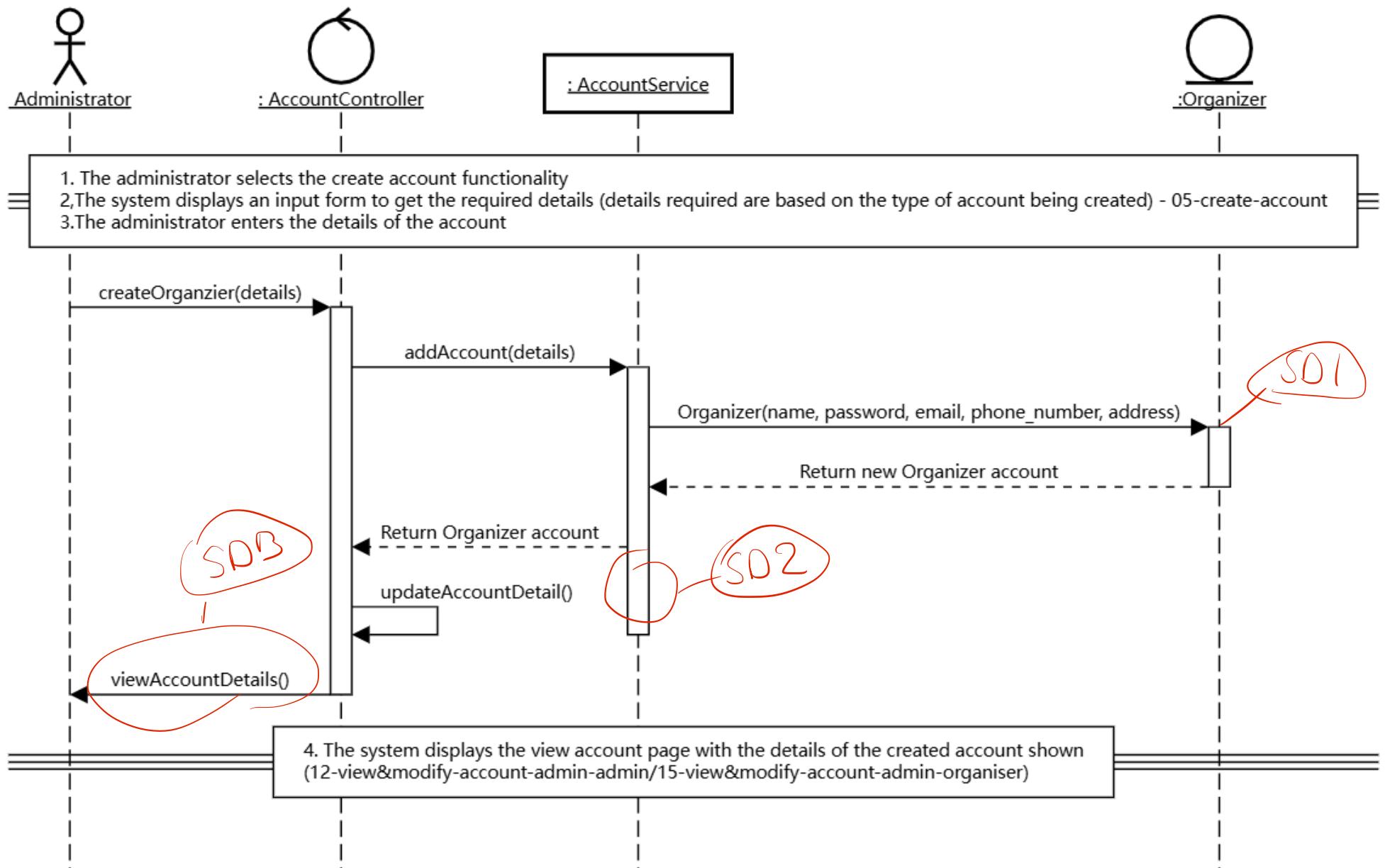
## Use Case 01 - Create Account

### 01 - Basic Course of Events

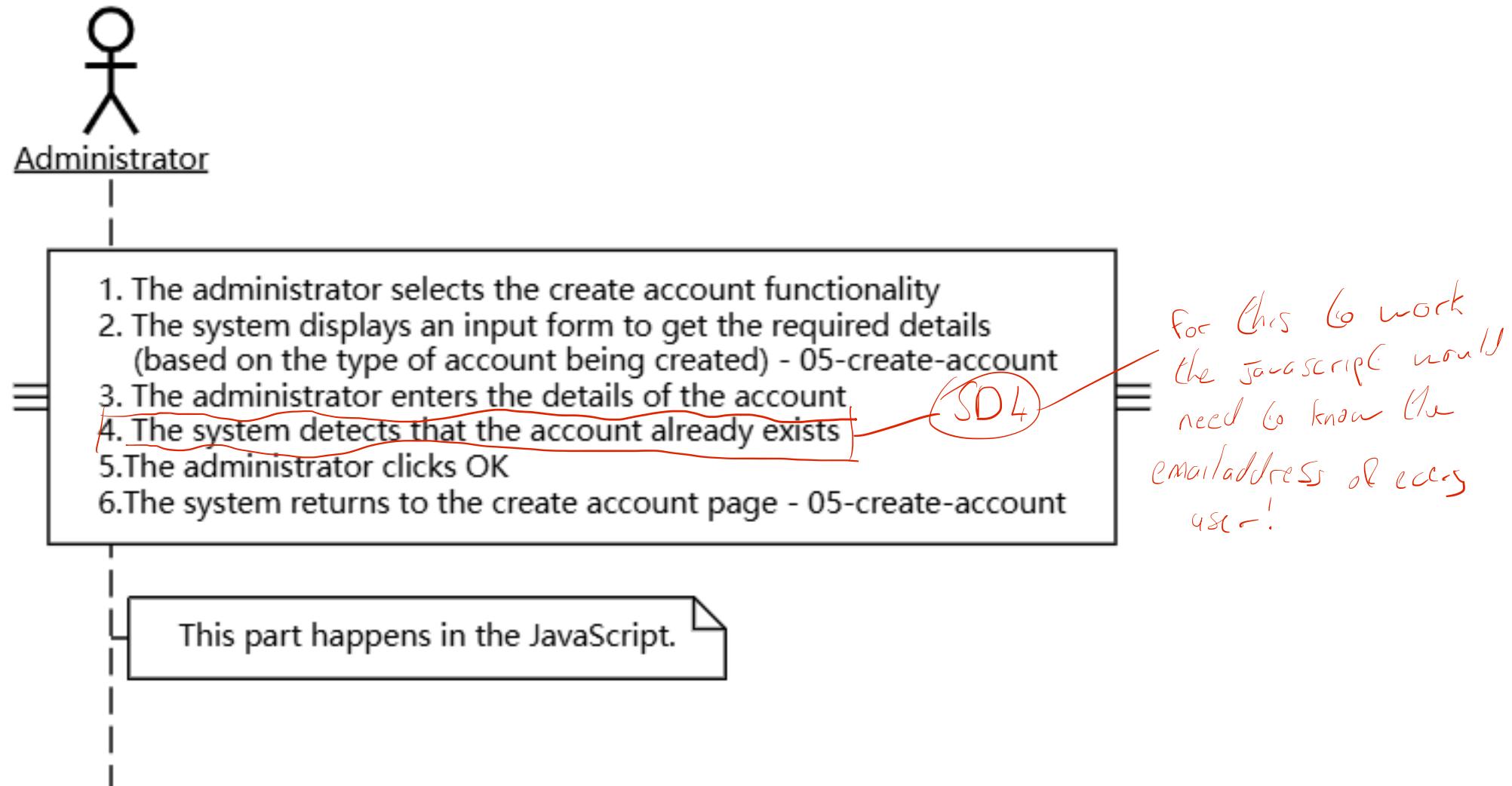
Sequence Diagram:01 Create Account(Creating an Administrator Account)



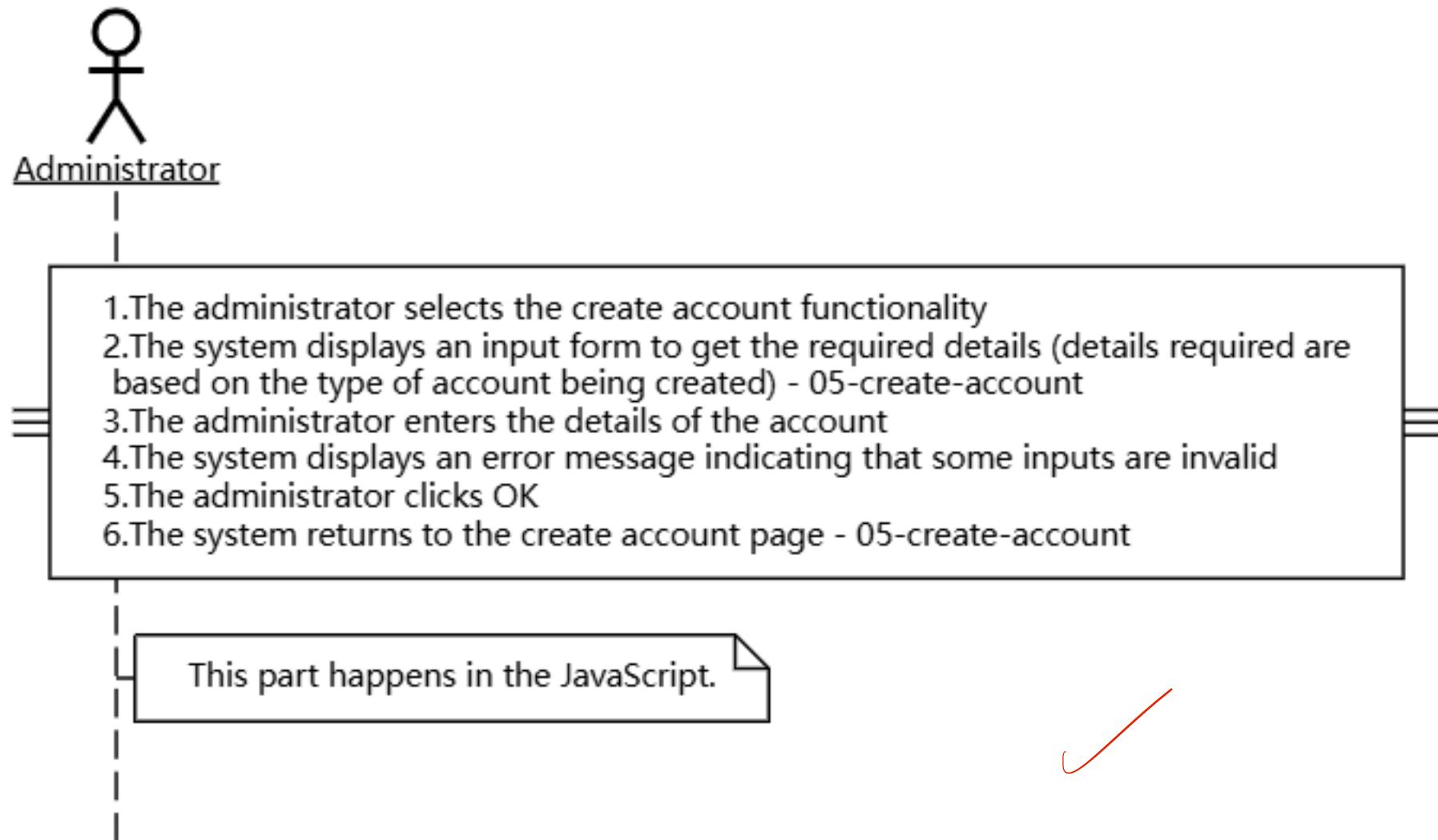
## Sequence Diagram:01 Create Account( Creating an Organizer Account)



## Sequence Diagram:01 Create Account (Alternative Courses of Events - Account Already Exists)



## Sequence Diagram: 01 Create Account (Alternative Courses of Events - Invalid Input)



Related UI Prototypes

[Home](#)[Profile](#)[Venues](#)[Accounts](#)

Identification ▾

input email or user name.....

[Search](#)[New Account](#)

Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com

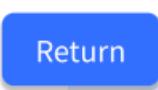


Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com

List Accounts Page

Return

Identification:

Name:

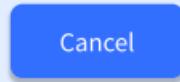
XXXX

Email:

123456789@email.com

Password: \*\*\*\*\*

Verify Password: \*\*\*\*\*

Sign inCancel

Create Account Page

Return



Administrator name: Lily [Change](#)

Email: 12345678@163.com [Change](#)

Password: \*\*\*\*\* [Change](#)

[Change Image](#) [Save](#) [Delete](#)

View Account Page  
(Admin-Admin)

Return



Organizer Name: Lily [Change](#)

Email: 12345678@163.com [Change](#)

Password: \*\*\*\*\* [Change](#)

Company Name: XXXXXX [Change](#)

Telephone: XXXX-XXXXXX [Change](#)

Address: XX Street, XX Road [Change](#)

[Save](#) [Delete](#)

View Account Page  
(Admin-Organiser)



## Duplicate Email

An account with that email address already exists.

Duplicate Email Message



## Missing Details

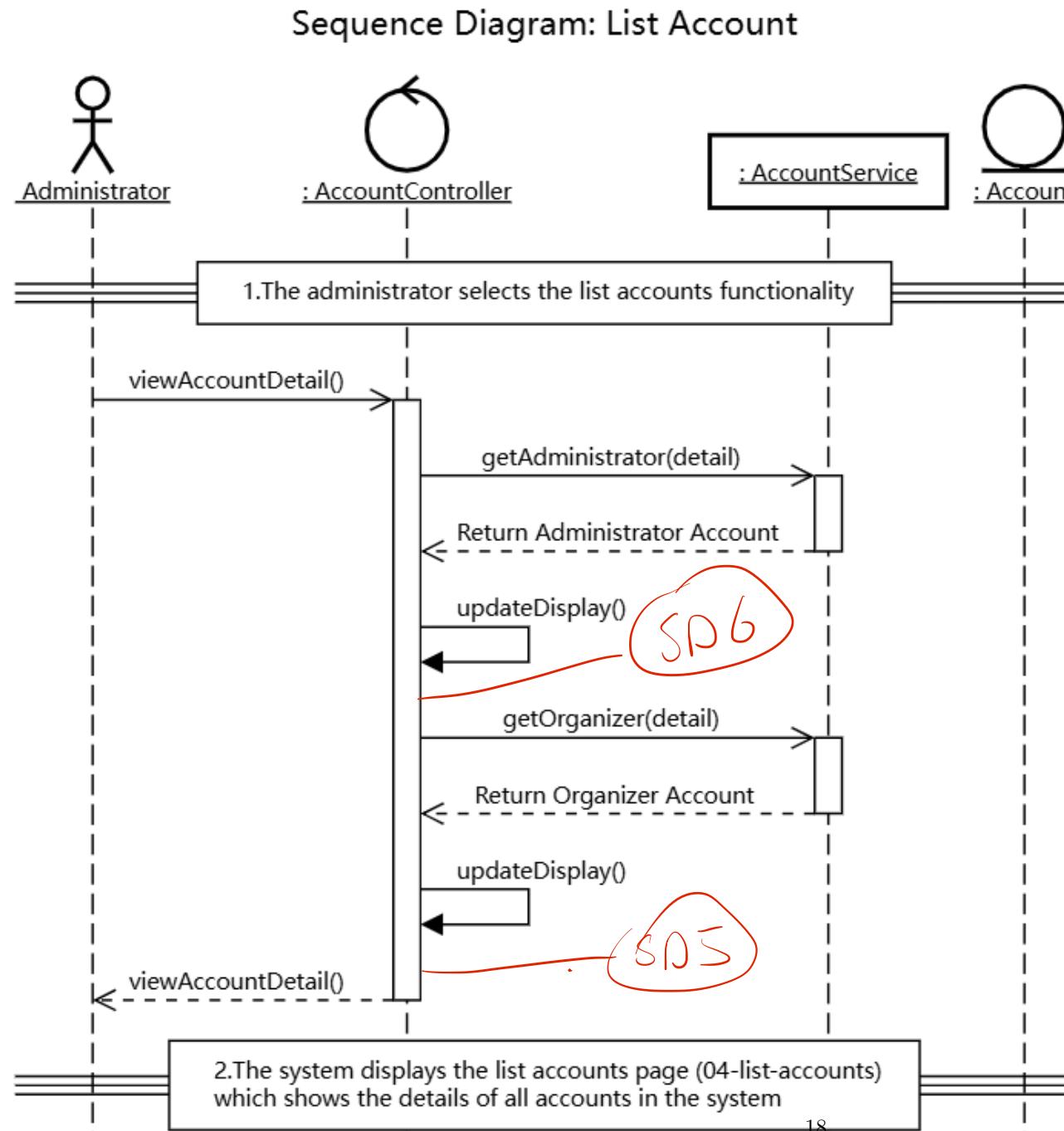
The following items in the create account form are required:

- Account Type
- Name
- Email address
- Password
- Company Name
- Company Address
- Phone Number



## Use Case 02 - List Account

### 01 - Basic Course of Events



The page should list  
all of the accounts

## Related UI Prototypes

Page Name    Image

The screenshot displays a user interface prototype for a platform, likely related to events or social networking. At the top, a navigation bar includes tabs for "Home", "Profile", "Venues", and "Accounts". A search bar with a placeholder and a blue "Search" button are positioned to the right of the tabs. Below the navigation, the main content area is organized into two rows of four light-gray rectangular cards each. Each card contains the text "Event" followed by a number and "Name". The cards are arranged in a grid pattern. The "Home" tab is highlighted with a blue underline, indicating it is the active page. The other tabs are in a standard black font.

Event	Name
Event 1	Name
Event 2	Name
Event 3	Name
Event 4	Name
Event 5	Name
Event 6	Name
Event 7	Name
Event 8	Name

Admin Main  
Page

Home

Profile

Venues

Accounts

---

Identification

input email or user name.....

Search

New Account



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com

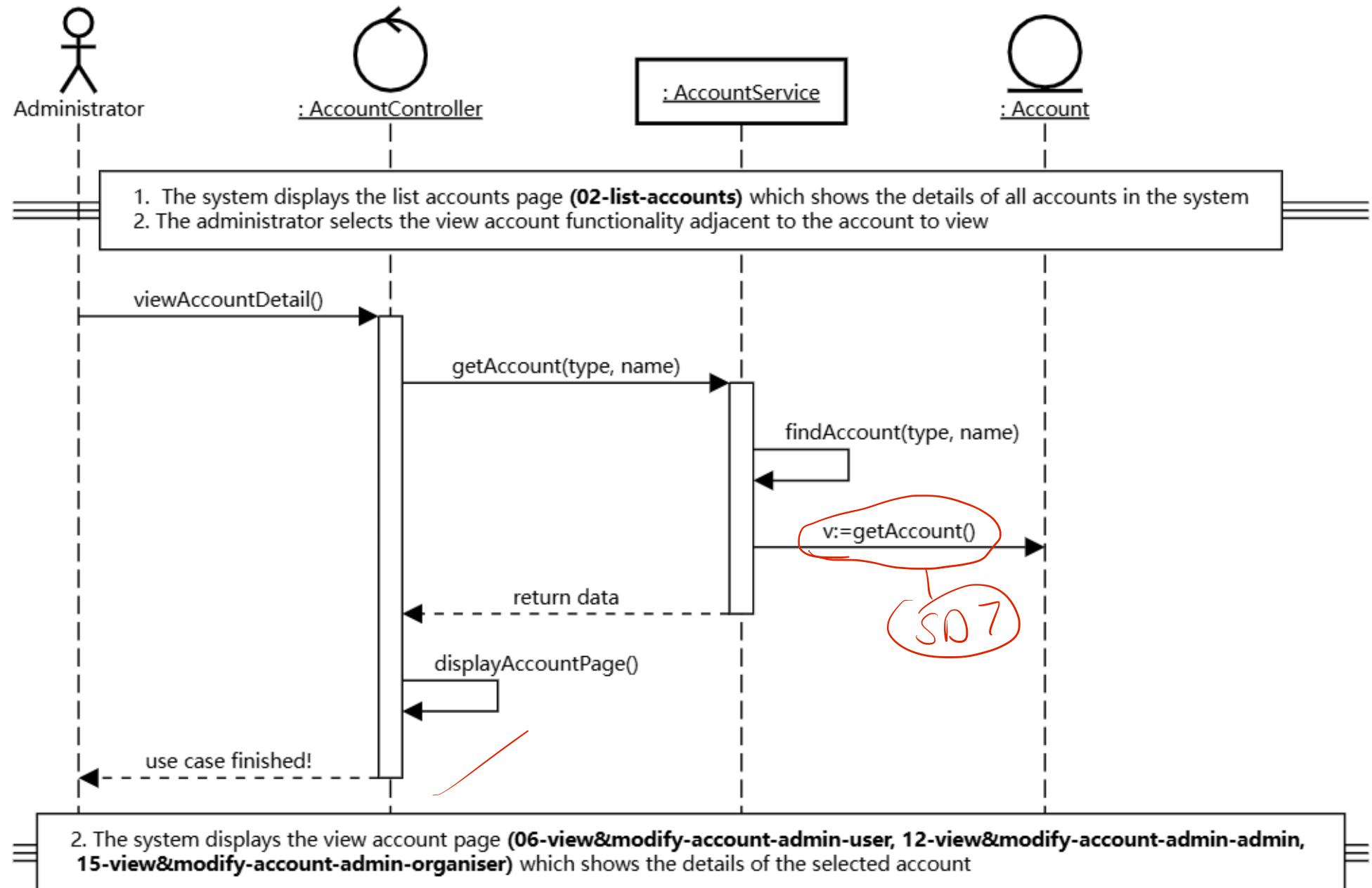


Name: XXXX  
Email: XXXXXXXX@email.com

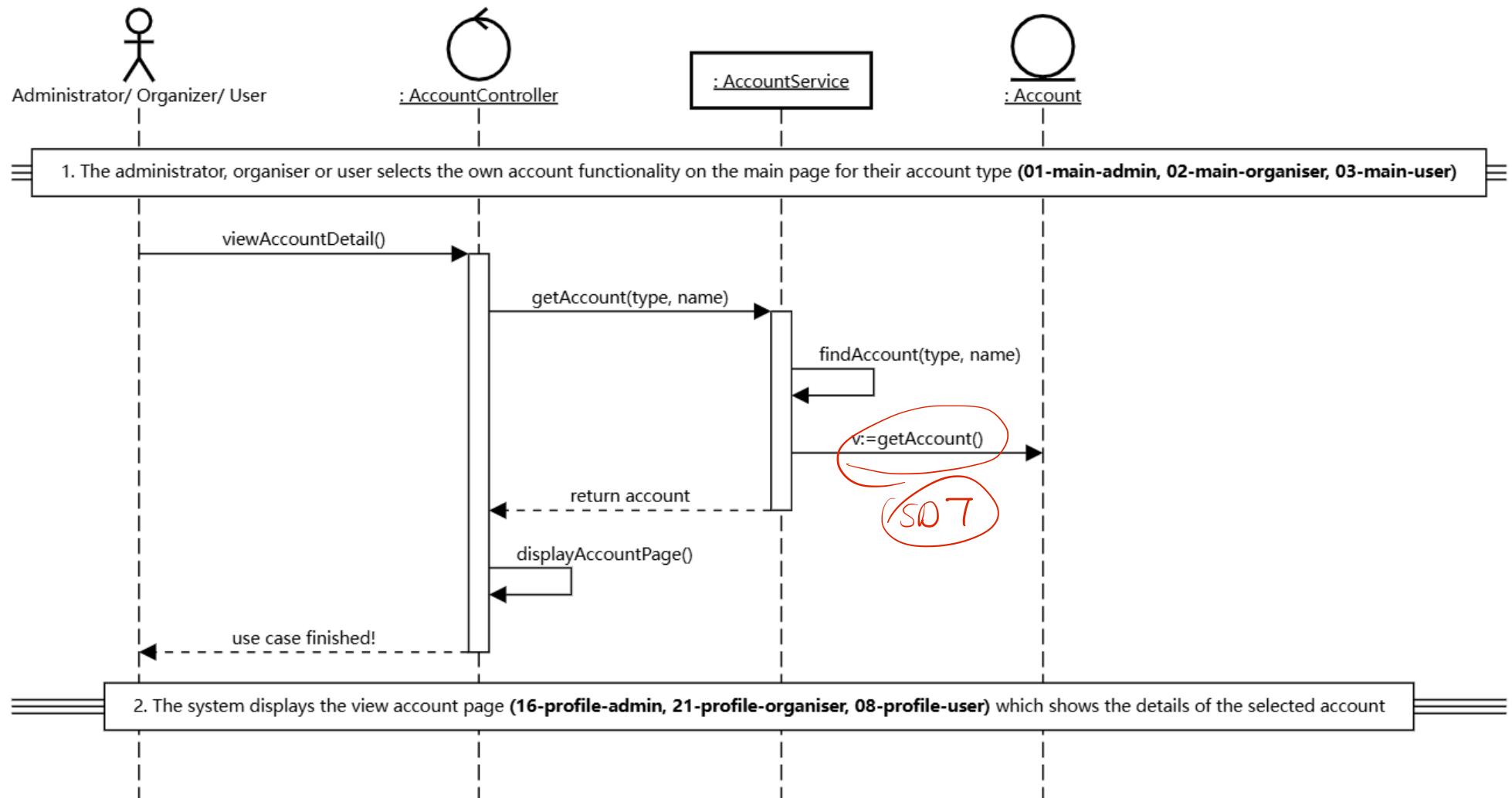
## Use Case 03 - View Account

### 01 - Basic Course of Events

Sequence Diagram: 03 View Account (Administrator Viewing Other Accounts)



## Sequence Diagram: 03 View Account (Administrator, Organiser or User Viewing Their Account)



Relevant UI Sketches

The image shows a wireframe of a web application interface. At the top, there is a navigation bar with four items: "Home" (which is underlined), "Profile", "Venues", and "Accounts". To the right of the navigation bar is a search bar with a placeholder and a blue "Search" button. Below the navigation bar is a grid of eight light-gray rectangular boxes, arranged in two rows of four. Each box contains the text "Event [number] Name", where [number] is a value from 1 to 8. The first row contains "Event 1 Name", "Event 2 Name", "Event 3 Name", and "Event 4 Name". The second row contains "Event 5 Name", "Event 6 Name", "Event 7 Name", and "Event 8 Name".

Home      Profile      Venues      Accounts

Event 1 Name      Event 2 Name      Event 3 Name      Event 4 Name

Event 5 Name      Event 6 Name      Event 7 Name      Event 8 Name

Main Admin Page

The image shows a user interface for an event organiser. At the top, there is a navigation bar with four items: "Home" (which is underlined in blue), "Profile", "My Events", and "Venues". To the right of the navigation bar is a search bar with a placeholder and a blue "Search" button. Below the navigation bar are six rectangular cards, each representing an event. The first two cards are larger and positioned above the others. The third through sixth cards are smaller and arranged in a row below them. Each card contains the text "Event" followed by a number and the word "Name".

Event	Name
1	Event 1 Name
2	Event 2 Name
3	Event 3 Name
4	Event 4 Name
5	Event 5 Name
6	Event 6 Name

Main Organiser Page

Home

Profile

Tickets

Search

Event 1  
Name

Event 2  
Name

Event 3  
Name

Event 4  
Name

Event 5  
Name

Event 6  
Name

Main User Page

[Home](#)[Profile](#)[Venues](#)[Accounts](#)Identification 

input email or user name.....

[Search](#)[New Account](#)

Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com

List Accounts Page

[Return](#)

  
[Change Image](#)

User name:  [Change](#)

Email:  [Change](#)

Password:  [Change](#)

[Save](#) [Delete](#)

View Account Page (Admin-User)

Return



Administrator name: Lily

Email: 12345678@163.com

Password: \*\*\*\*\*

View Account Page (Admin-Admin)

[Return](#)

  
[Change Image](#)

Organizer Name:	Lily	<a href="#">Change</a>
Email:	12345678@163.com	<a href="#">Change</a>
Password:	*****	<a href="#">Change</a>
Company Name:	XXXXXX	<a href="#">Change</a>
Telephone:	XXXX-XXXXXX	<a href="#">Change</a>
Address:	XX Street, XX Road	<a href="#">Change</a>

[Save](#) [Delete](#)

View Account Page (Admin-Organiser)

Page Name

Image

<a href="#">Home</a>	<a href="#">Profile</a>	<a href="#">Venues</a>	<a href="#">Accounts</a>
----------------------	-------------------------	------------------------	--------------------------

  
[Change Image](#)

Administrator Name:  [Change](#)

Email:  [Change](#)

Password:  [Change](#)

[Save](#)

[Sign out](#)

View Account Page (Admin)

[Home](#)[Profile](#)[My Events](#)[Venues](#)

Organizer Name:  [Change](#)

Email:  [Change](#)

Password:  [Change](#)

Company Name:  [Change](#)

Telephone:  [Change](#)

Address:  [Change](#)

[Save](#) [Sign out](#)

View Account Page (Organiser)

Page Name

Image

---

Home      Profile      Tickets



User name:  [Change](#)

Email:  [Change](#)

Password:  [Change](#)

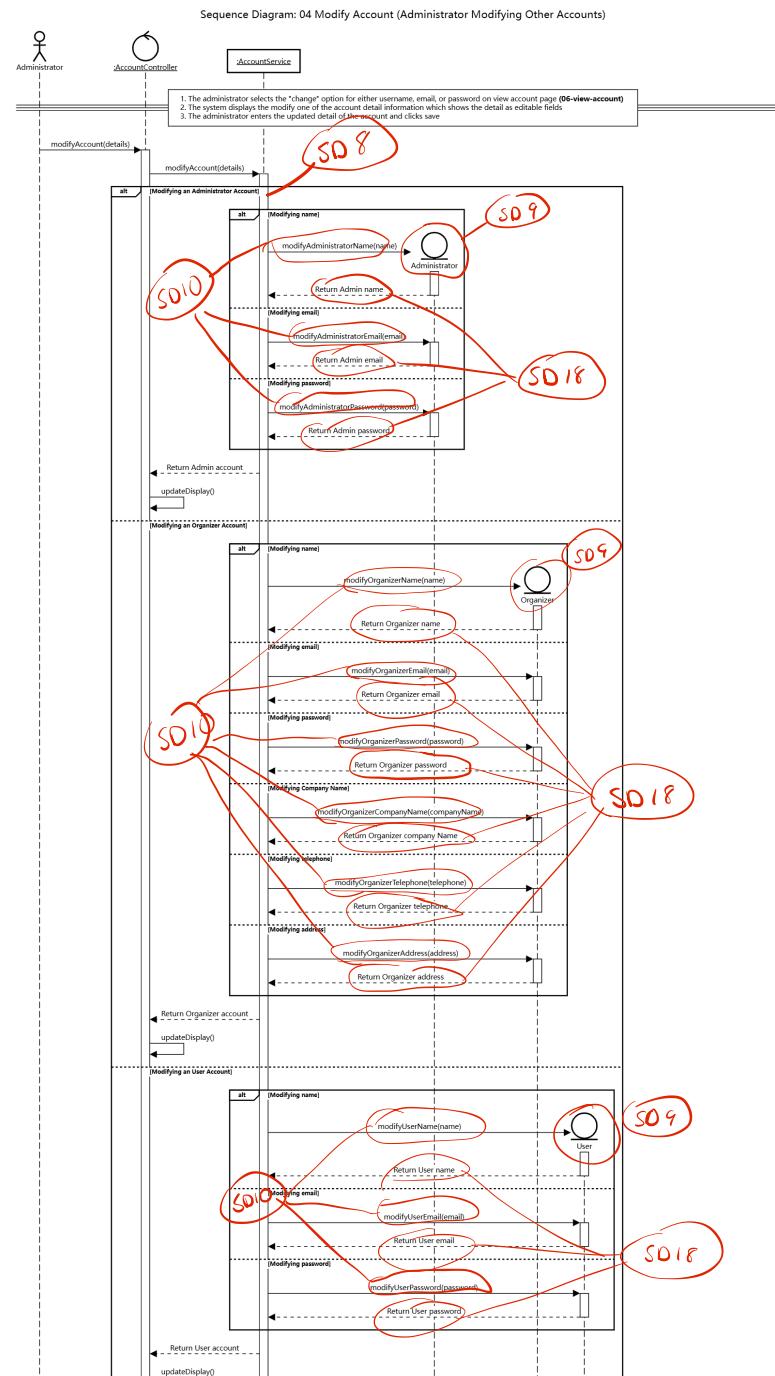
[Change Image](#) [Save](#) [Sign out](#)

View Account Page (User)

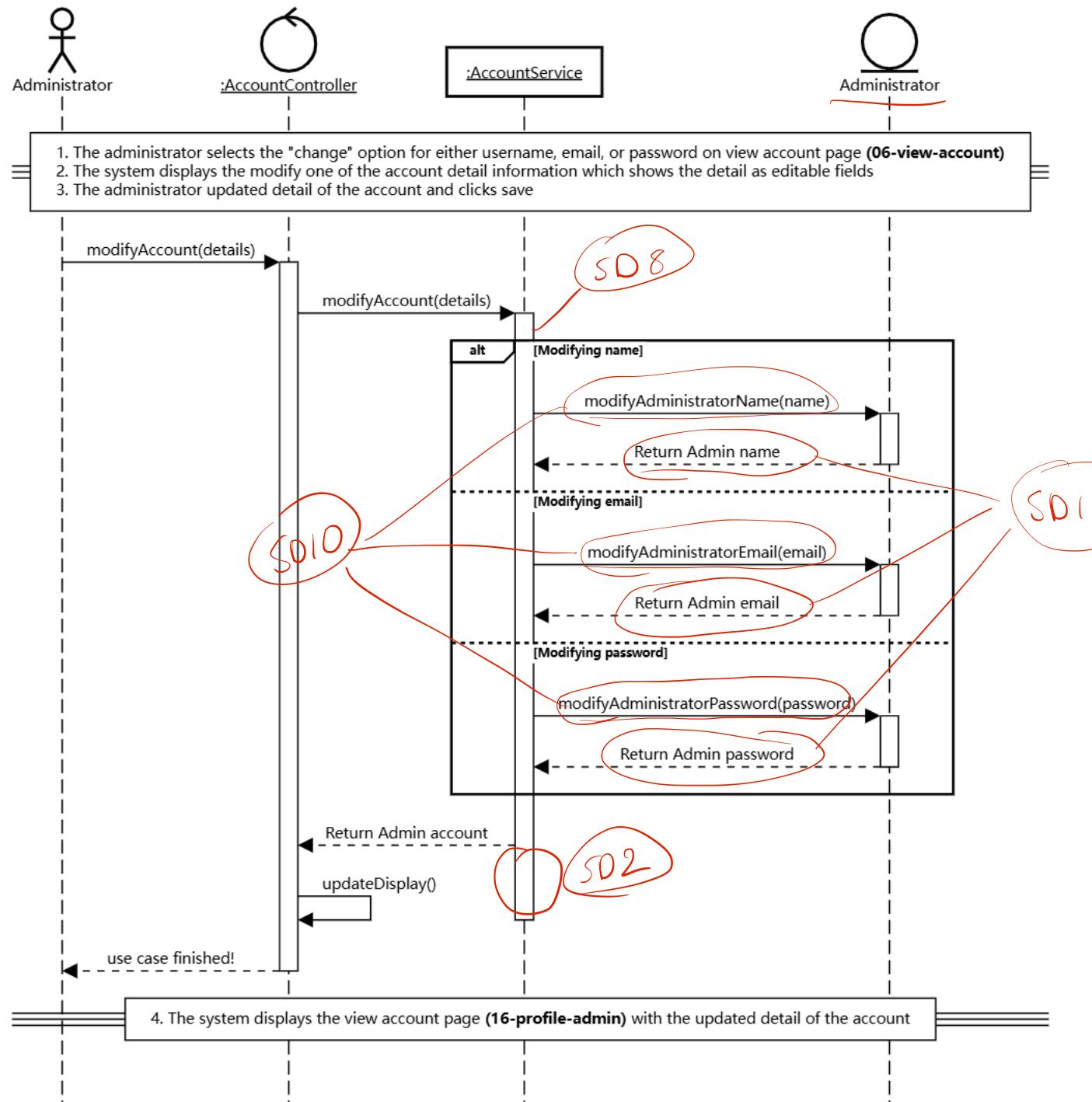
---

## Use Case 04 - Modify Account

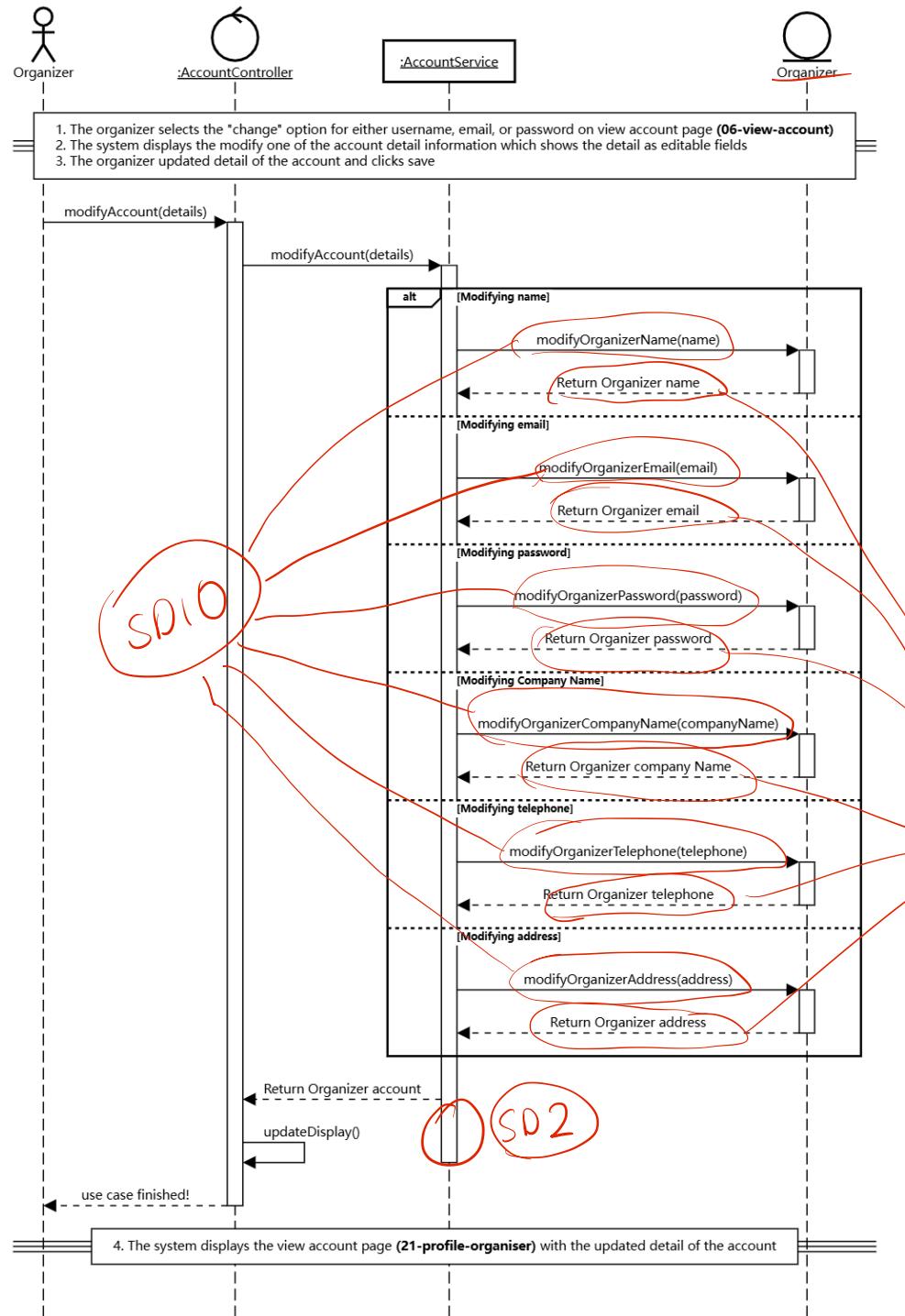
### 01 - Basic Course of Events



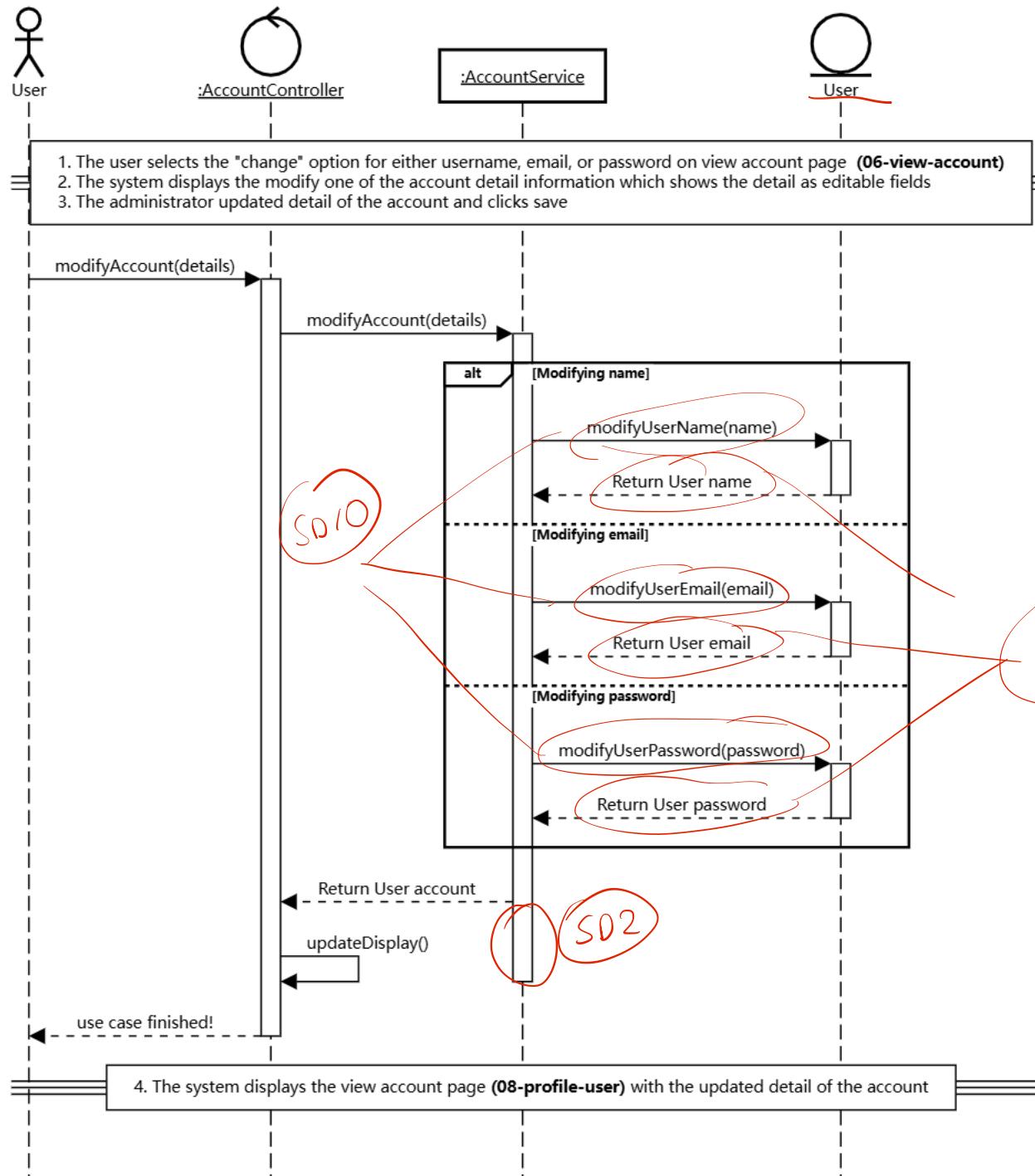
## Sequence Diagram: 04 Modify Account (Administrator Modifying Their Own Account)



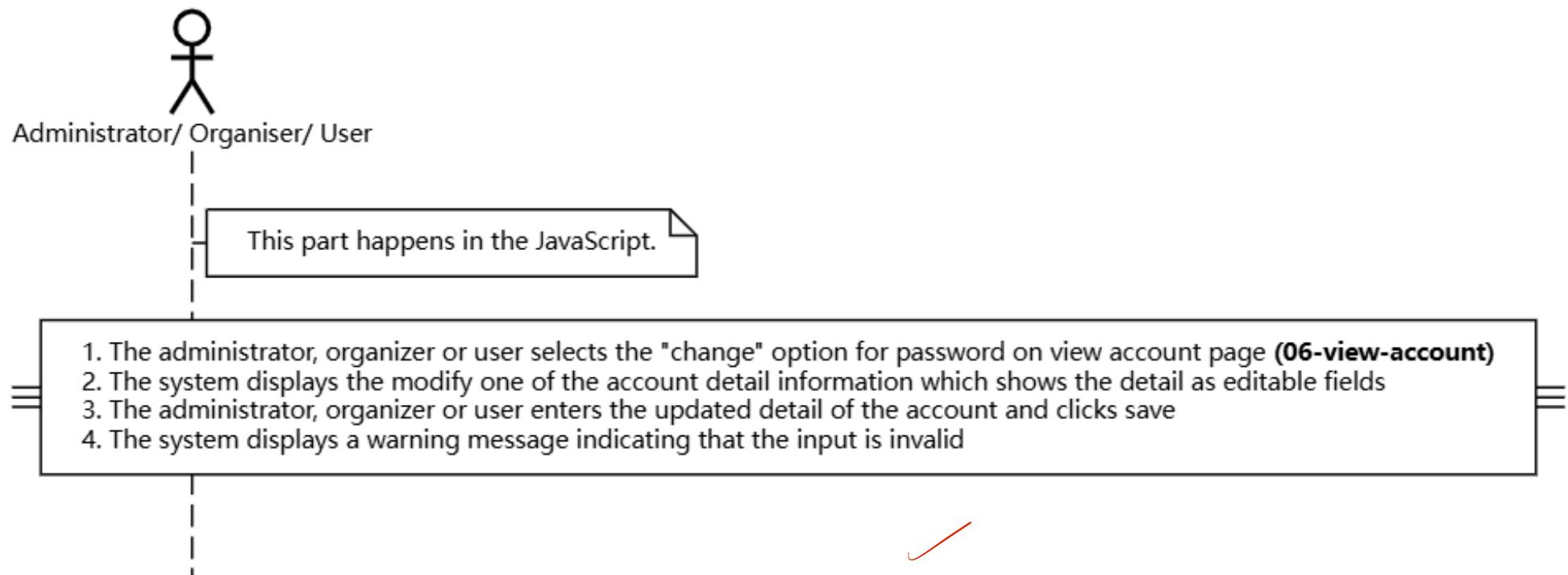
Sequence Diagram: 04 Modify Account (Organizer Modifying Their Own Account)



## Sequence Diagram: 04 Modify Account (User Modifying Their Own Account)



## Sequence Diagram: 04 Modify Account (Alternate Course of Events - Invalid Input)



Relevant UI Sketches

[Home](#)[Profile](#)[Venues](#)[Accounts](#)Identification 

input email or user name.....

[Search](#)[New Account](#)

Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com



Name: XXXX  
Email: XXXXXXXX@email.com

List Accounts Page

Return



User name:  Change

Email:  Change

Password:  Change

Change Image Save Delete

View Account Page (Admin-User)

Return



Administrator name: Lily

Email: 12345678@163.com

Password: \*\*\*\*\*

View Account Page (Admin-Admin)

[Return](#)

  
[Change Image](#)

Organizer Name:	Lily	<a href="#">Change</a>
Email:	12345678@163.com	<a href="#">Change</a>
Password:	*****	<a href="#">Change</a>
Company Name:	XXXXXX	<a href="#">Change</a>
Telephone:	XXXX-XXXXXX	<a href="#">Change</a>
Address:	XX Street, XX Road	<a href="#">Change</a>

[Save](#) [Delete](#)

View Account Page (Admin-Organiser)

Page Name

Image

Home      Profile      Venues      Accounts

---



Administrator Name:  [Change](#)

Email:  [Change](#)

Password:  [Change](#)

[Change Image](#) [Save](#)

[Sign out](#)

View Account Page (Admin)

[Home](#)[Profile](#)[My Events](#)[Venues](#)

Organizer Name:  [Change](#)

Email:  [Change](#)

Password:  [Change](#)

Company Name:  [Change](#)

Telephone:  [Change](#)

Address:  [Change](#)

[Save](#) [Sign out](#)

View Account Page (Organiser)

Page Name

Image

---

Home      Profile      Tickets



User name:  [Change](#)

Email:  [Change](#)

Password:  [Change](#)

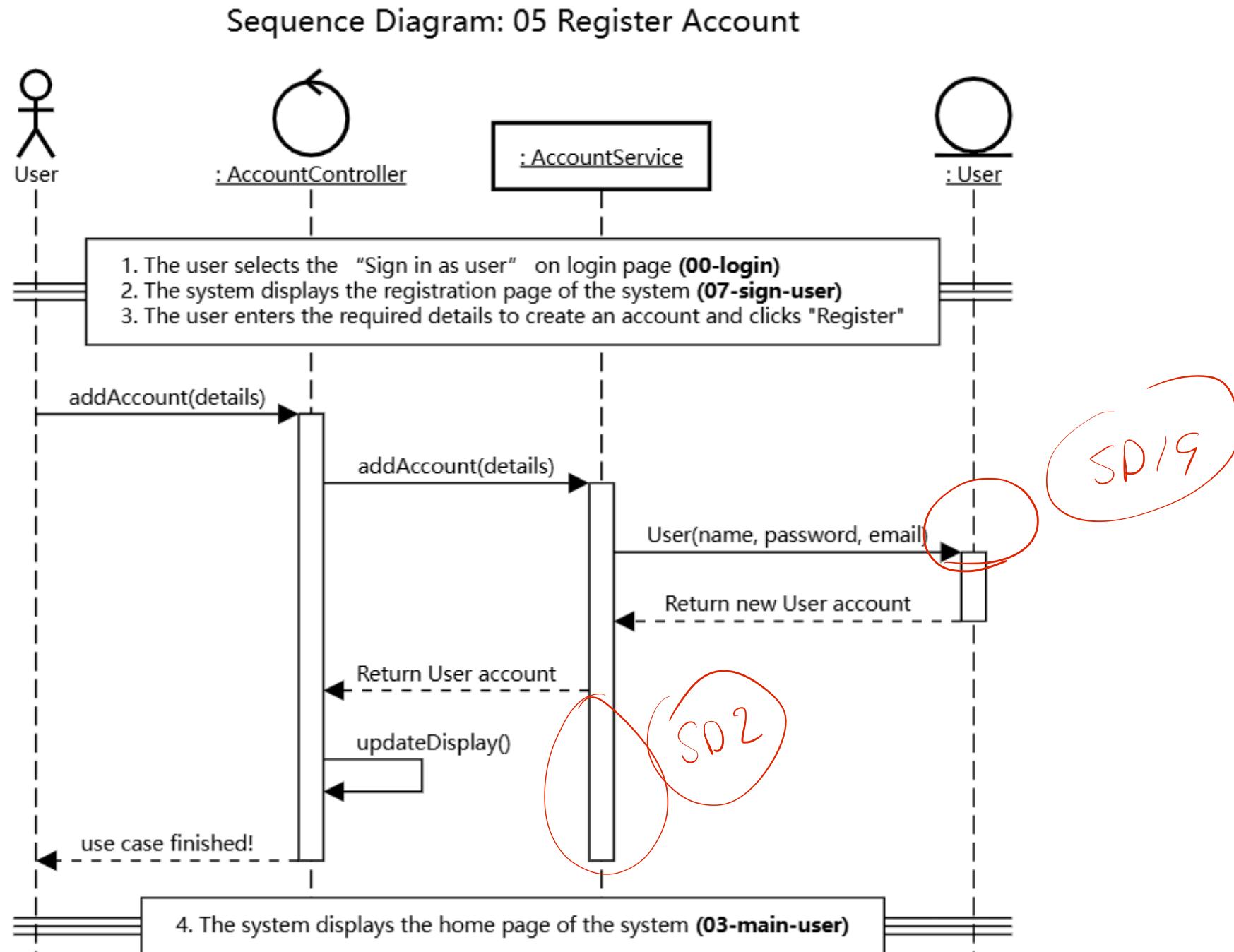
[Change Image](#) [Save](#) [Sign out](#)

View Account Page (User)

---

## Use Case 05 - Register Account

### 01 - Basic Course of Events



## 02 - Alternate Course of Events

### Sequence Diagram: 05 Register Account (Alternate Courses of Events - Invalid Input)



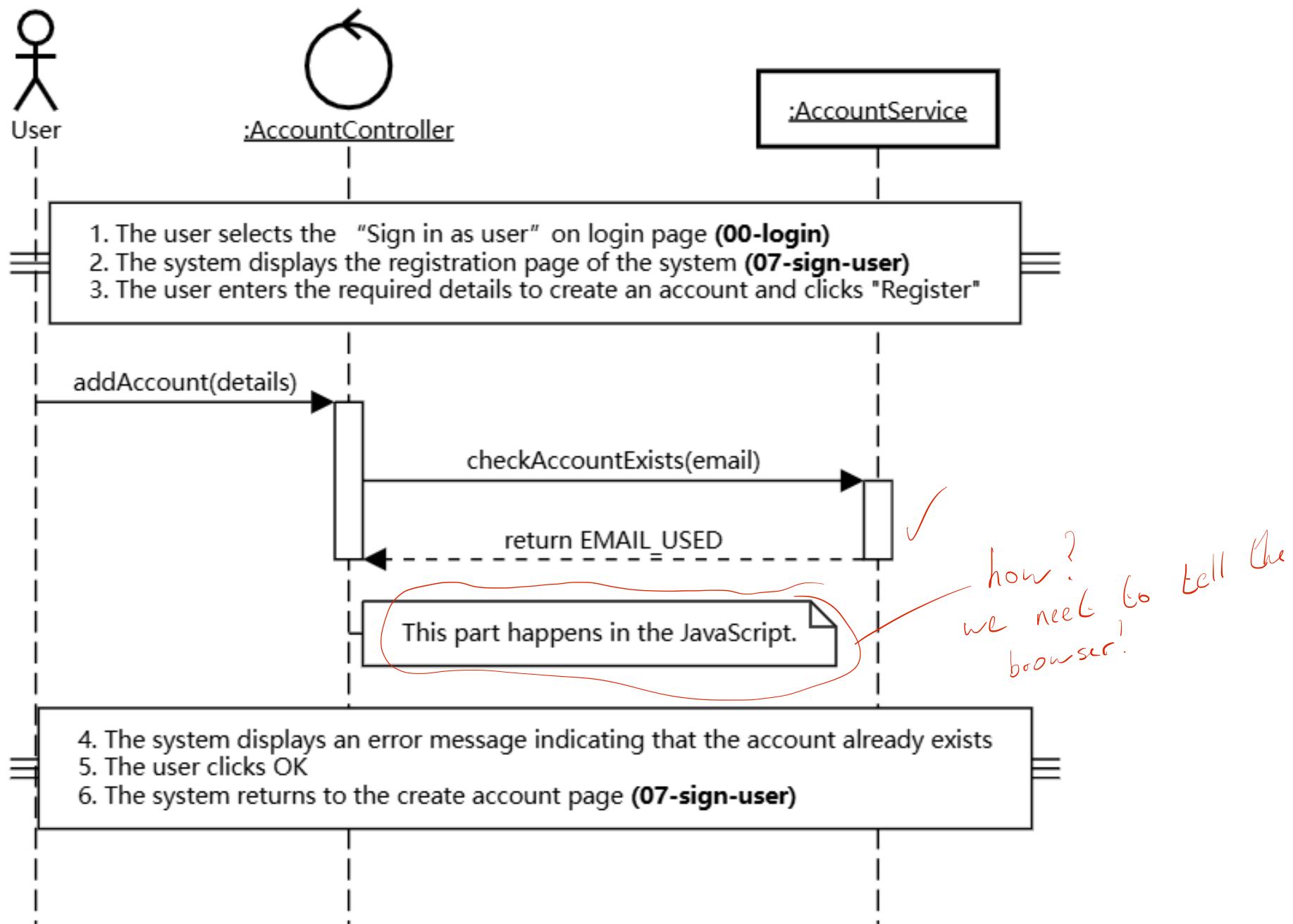
User

This part happens in the JavaScript.

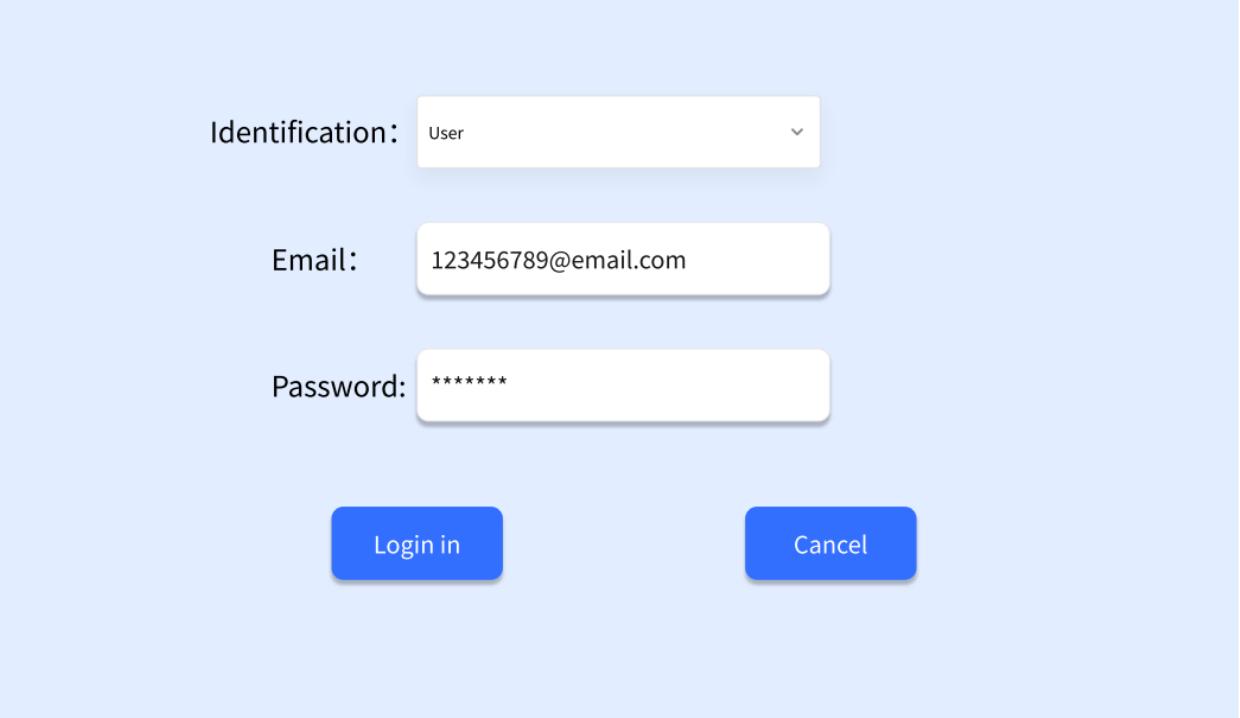
1. The user selects the create account functionality
2. The system displays an input form to get the required details (details required are based on the type of account being created) (02-sign-user)
3. The user enters the details of the account
4. The system displays an error message indicating that some inputs are invalid
5. The user clicks OK
6. The system returns to the create account page (02-sign-user)

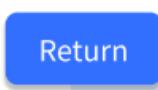


## Sequence Diagram: 05 Register Account (Alternate Courses of Events Account Already Exists)



## Relevant UI Sketches

Page Name	Image
Login Page	 <p>The image shows a user interface sketch for a login page. At the top right is a blue button labeled "Sign in as user". Below it is a light blue rectangular area containing three input fields. The first field is labeled "Identification:" with a dropdown menu showing "User". The second field is labeled "Email:" with the value "123456789@email.com". The third field is labeled "Password:" with the value "*****". At the bottom left is a blue button labeled "Login in", and at the bottom right is a blue button labeled "Cancel".</p>

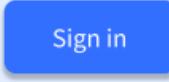
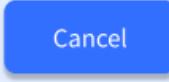
Return

Name: XXXX

Email: 123456789@email.com

Password: \*\*\*\*\*

Verify Password: \*\*\*\*\*

Sign inCancel

Sign Page

Home

Profile

Tickets

Search

Event 1  
Name

Event 2  
Name

Event 3  
Name

Event 4  
Name

Event 5  
Name

Event 6  
Name

Main User Page



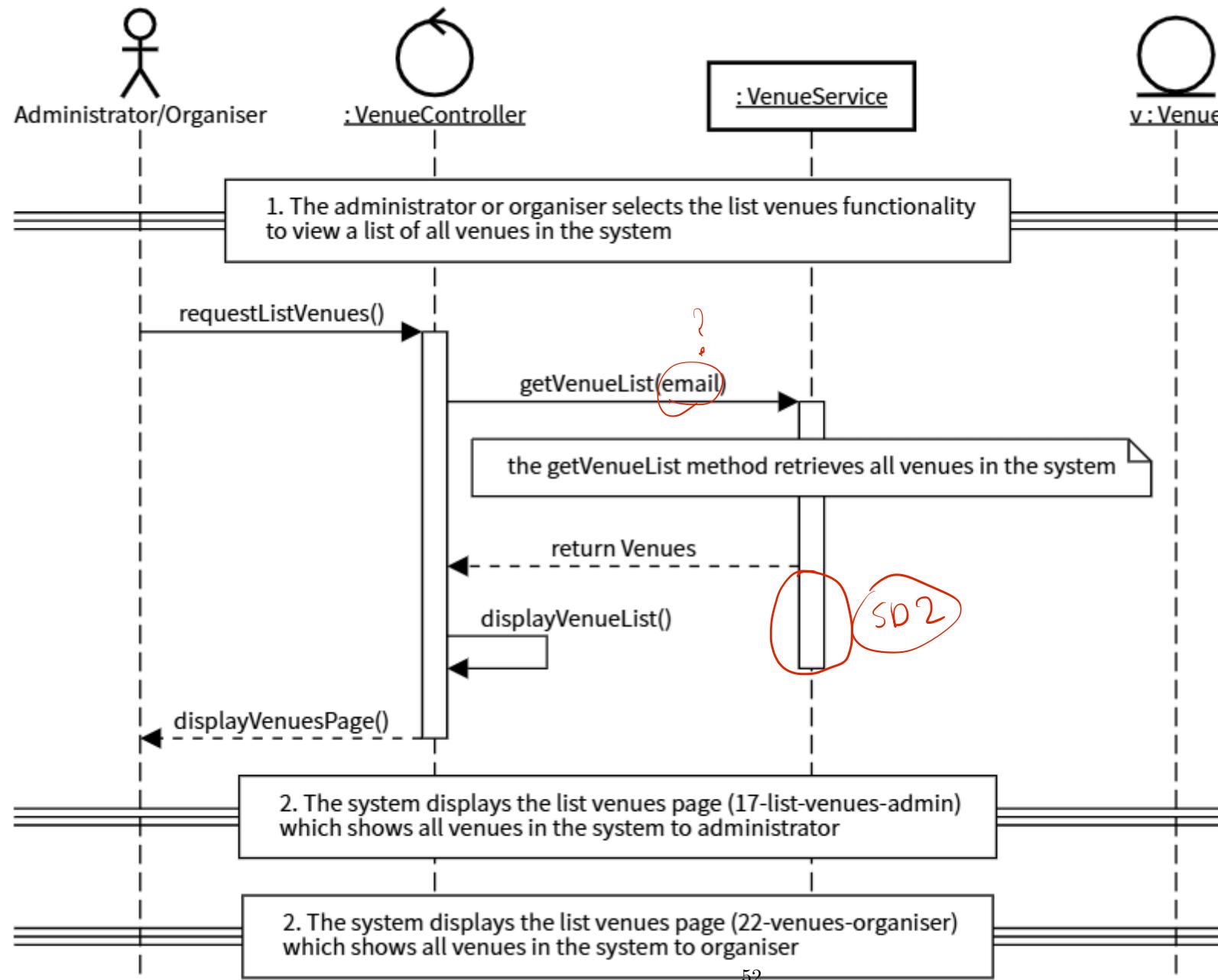
## Duplicate Email

An account with that email address already exists.

## Use Case 06 - List Venues

### 01 - Basic Course of Events

Sequence Diagram: 06 List Venues



## Related UI Prototypes

Page Name

Image

Home

Profile

Venues

Accounts

Search

Venue 1  
Name

Venue 2  
Name

Venue 3  
Name

Venue 4  
Name

Venue 5  
Name

Venue 6  
Name

Venue 7  
Name

Venue 8  
Name

17-list-venues-admin

[Home](#)[Profile](#)[My Events](#)[Venues](#)[Search](#)[New Venue](#)

Venue 1  
Name

Venue 2  
Name

Venue 3  
Name

Venue 4  
Name

Venue 5  
Name

Venue 6  
Name

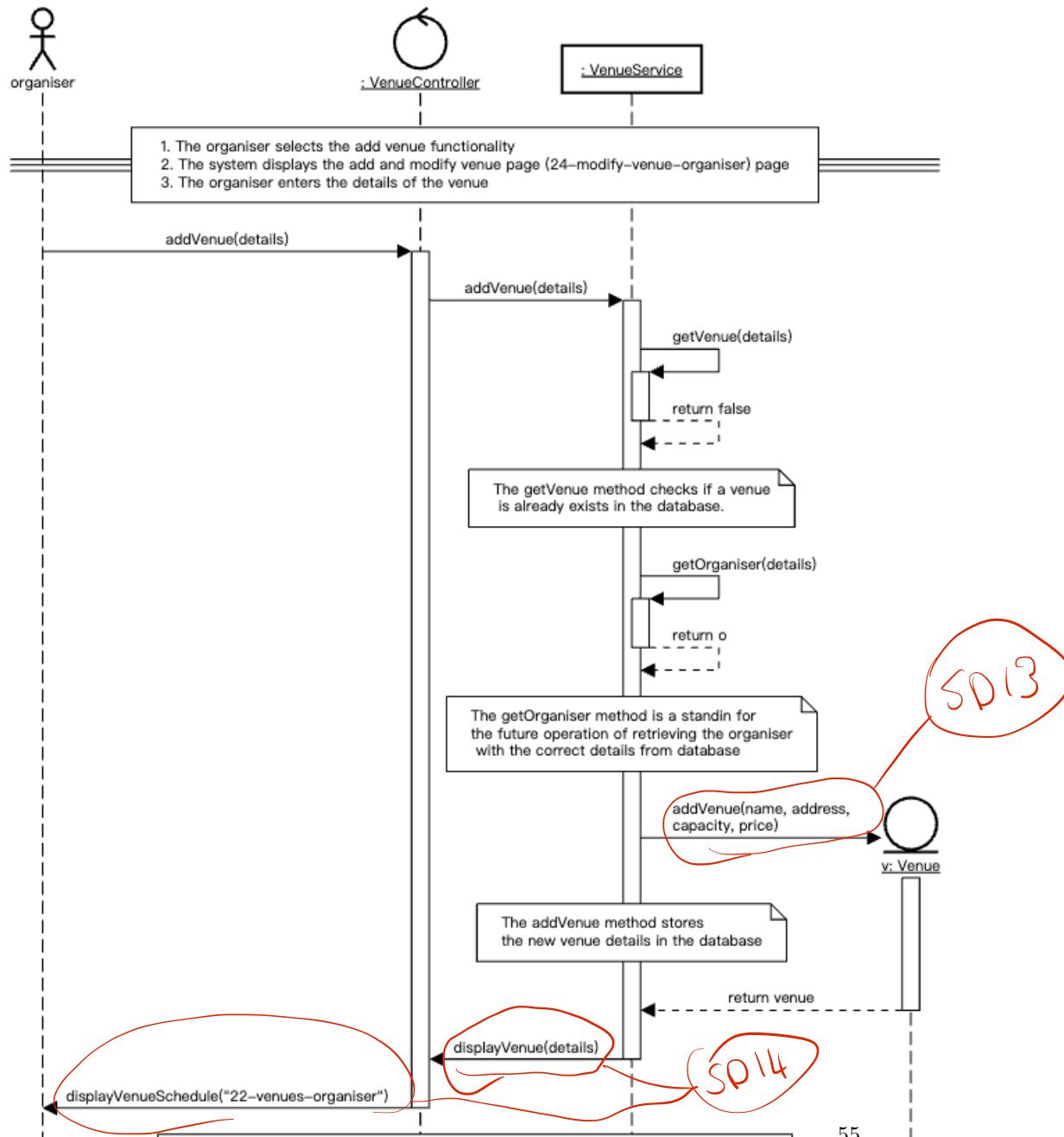
Venue 7  
Name

Venue 8  
Name

## Use Case 07 - Add Venue

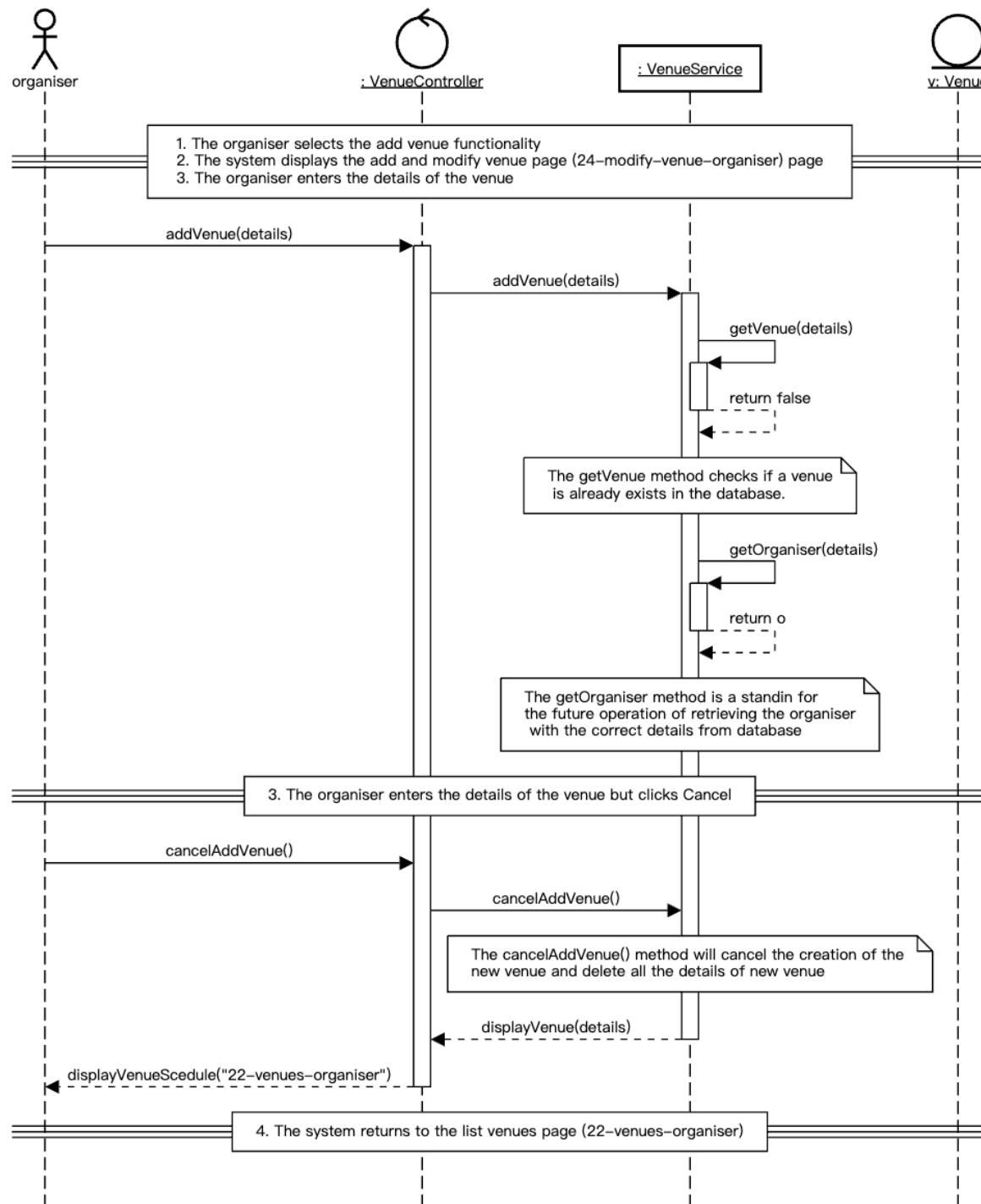
### 01 - Basic Course of Events

Sequence Diagram: 07 Add Venue



## 02 - Alternate Course of Events - The Organiser Not Save

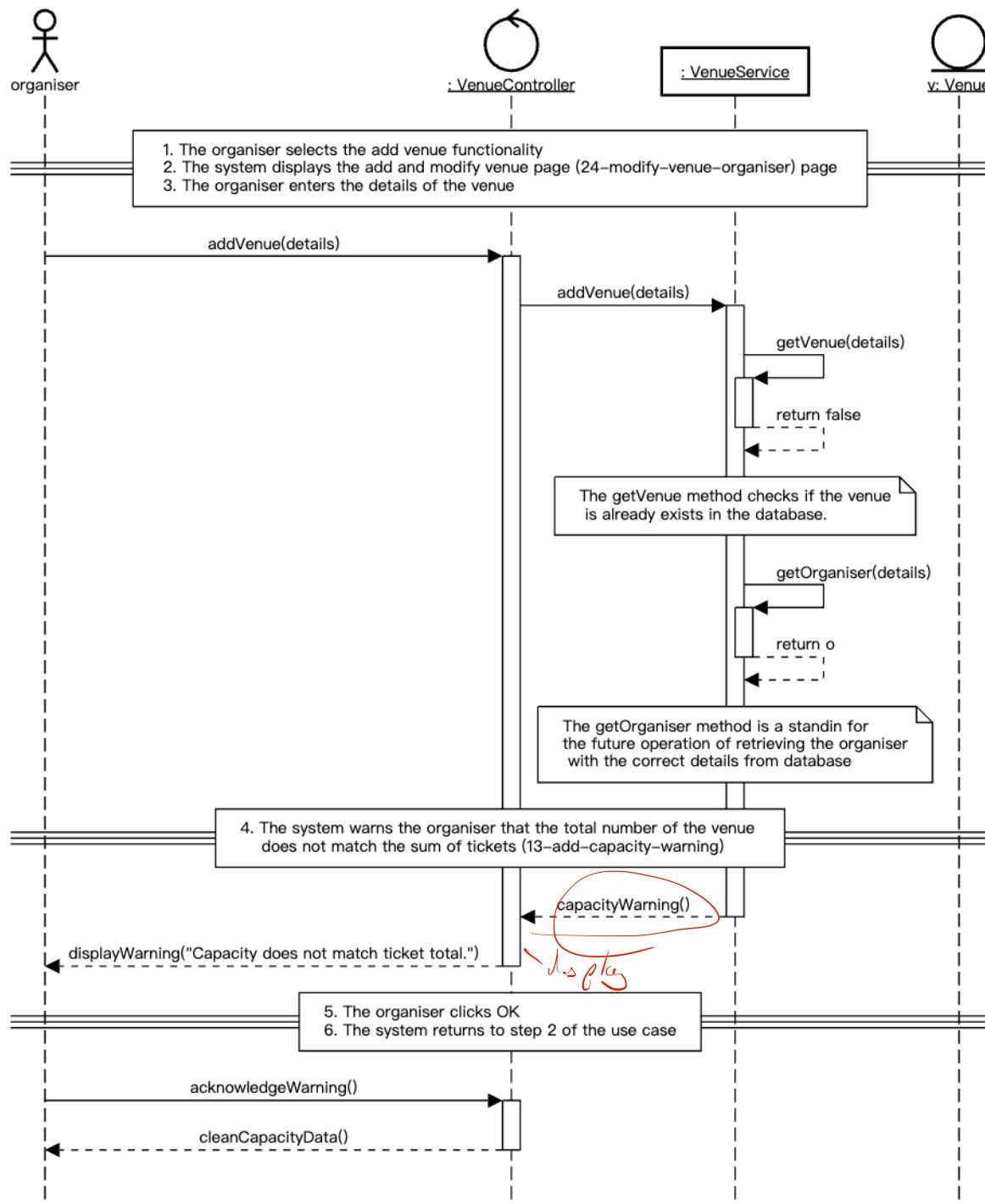
Sequence Diagram: 07 Add Venue (Alternate Course of Events – The Organiser Not Save)



SP16

## 03 - Alternate Course of Events - Incorrect Capacity

Sequence Diagram: 07 Add Venue (Alternate Course of Event – Incorrect Capacity)



## Related UI Prototypes

Page Name	Image
Organiser Main Page	<p>The image shows a user interface prototype for an 'Organiser Main Page'. At the top, there is a navigation bar with five items: 'Home' (which is underlined in blue), 'Profile', 'My Events', 'Venues', and a search bar containing the word 'Search'. Below the navigation bar are six rectangular cards, each representing an event. The first two cards are larger and positioned above the others. The first card contains the text 'Event 1 Name'. The second card contains 'Event 2 Name'. The remaining four cards are smaller and arranged in a row below them. They contain the text 'Event 3 Name', 'Event 4 Name', 'Event 5 Name', and 'Event 6 Name' respectively.</p>

Page Name	Image
	<p>A screenshot of a web application interface. At the top, there is a navigation bar with links: Home, Profile, My Events, Venues (which is underlined, indicating it is the active page), and a search bar. Below the navigation bar, there is a large blue button labeled "New Venue". The main content area displays a grid of eight rectangular cards, each representing a venue. The cards are arranged in two rows of four. Each card contains the text "Venue [number] Name". The first row contains cards for "Venue 1 Name", "Venue 2 Name", "Venue 3 Name", and "Venue 4 Name". The second row contains cards for "Venue 5 Name", "Venue 6 Name", "Venue 7 Name", and "Venue 8 Name".</p>

The List Venues Page(Organiser)

[Return](#)

Venue Name:

XXXX

Change

Address:

XXXX

Change

Total Capacity:

XXXX

Change

Price Classification:

XXXX

Change

Local Contact:

Name:

XXXX

Change

Phone Number:

XXXX

Change

Email:

XXXX

Change

Save

Cancel

The Add and Modify Venue Page |

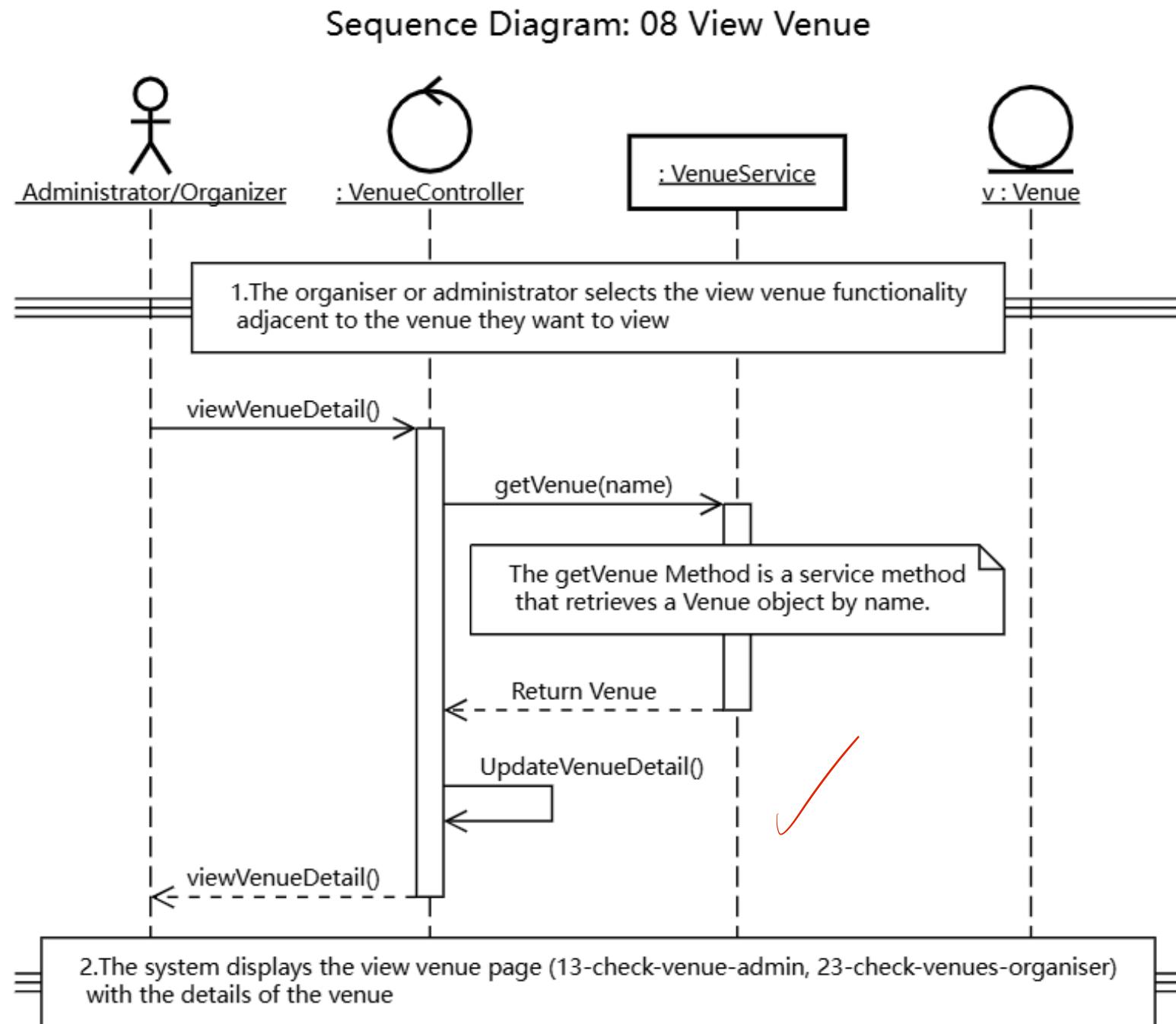
## Total Capacity Error

The total capacity of the venue must be equal to the sum of the capacities of the ticket types.

Add Capacity Warning Page | ↴

## Use Case 08 - View Venue

### 01 - Basic Course of Events



## Related UI Prototypes

Page Name

Image

Home

Profile

Venues

Accounts

Search

Venue 1  
Name

Venue 2  
Name

Venue 3  
Name

Venue 4  
Name

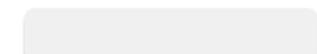
Venue 5  
Name

Venue 6  
Name

Venue 7  
Name

Venue 8  
Name

List Venues Page (Admin)

[Home](#)[Profile](#)[My Events](#)[Venues](#)[Search](#)[New Venue](#)Venue 1  
NameVenue 2  
NameVenue 3  
NameVenue 4  
NameVenue 5  
NameVenue 6  
NameVenue 7  
NameVenue 8  
Name

List Venues Page (Organiser)

[Return](#)

Venue Name:

Address:

Total Capacity:

Price Classification:

Local Contact:

Name:

Phone Number:

Email:

Schedule:

&lt;Event Date&gt; &lt;Event Time&gt;&lt;Event Name&gt;

&lt;Event Date&gt; &lt;Event Time&gt;&lt;Event Name&gt;

&lt;Event Date&gt; &lt;Event Time&gt;&lt;Event Name&gt;

[Delete](#)

View Venue Page (Admin)

[Return](#)**Venue Name:****Schedule:****Address:**

&lt;Event Date&gt; &lt;Event Time&gt;&lt;Event Name&gt;

**Total Capacity:**

&lt;Event Date&gt; &lt;Event Time&gt;&lt;Event Name&gt;

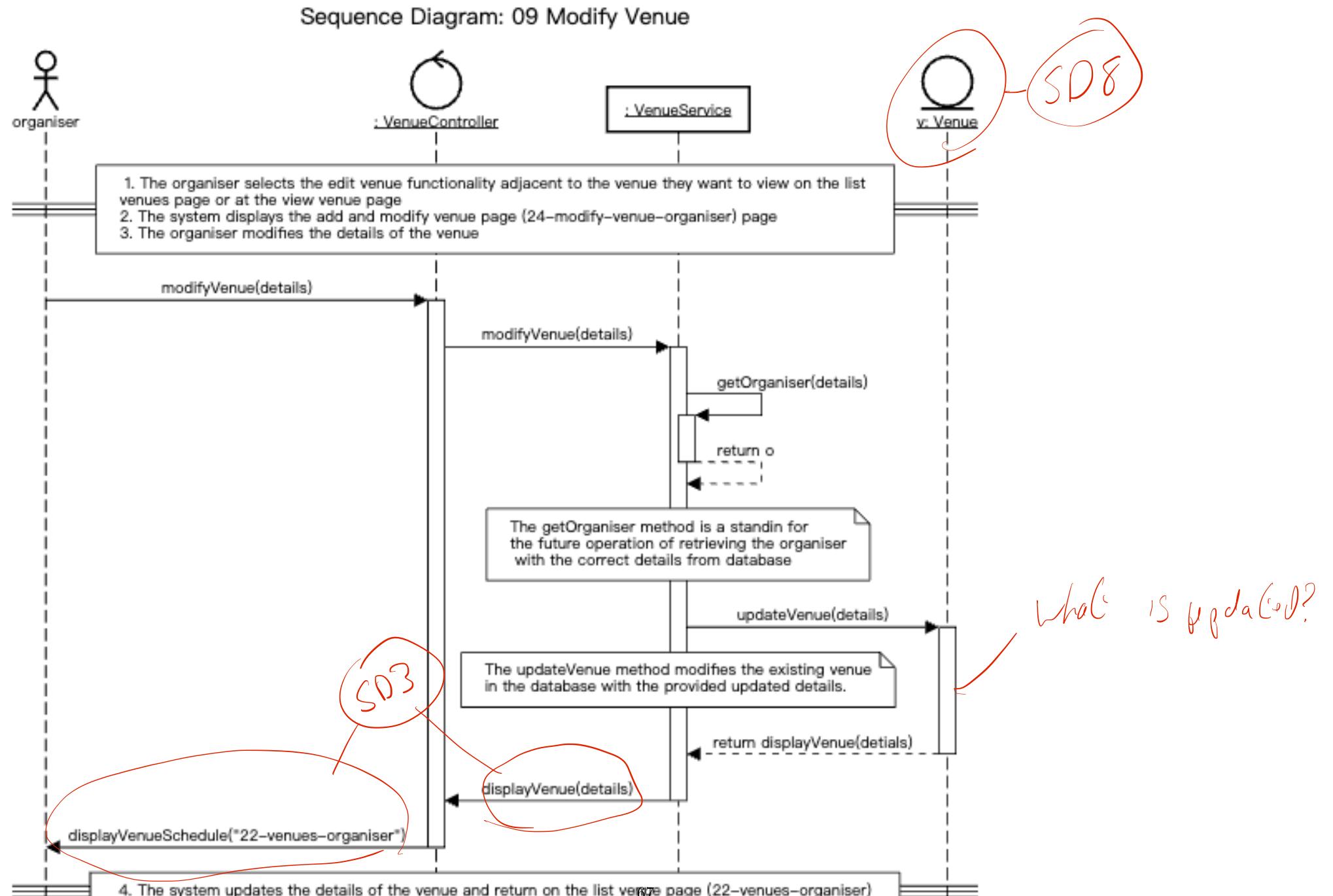
**Price Classification:**

&lt;Event Date&gt; &lt;Event Time&gt;&lt;Event Name&gt;

**Local Contact:****Name:****Phone Number:****Email:**[Edit](#)

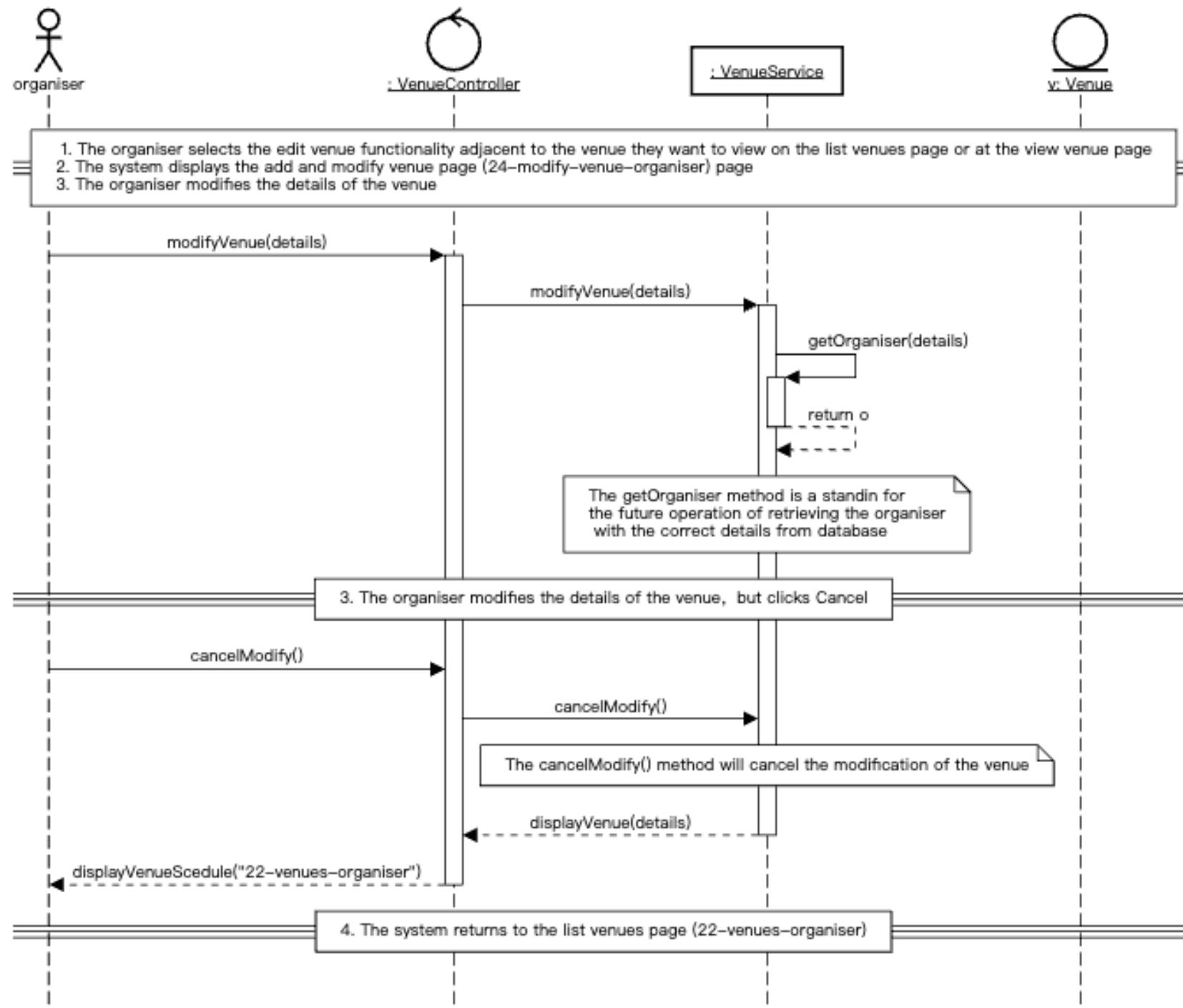
## Use Case 09 - Modify Venue

### 01 - Basic Course of Events



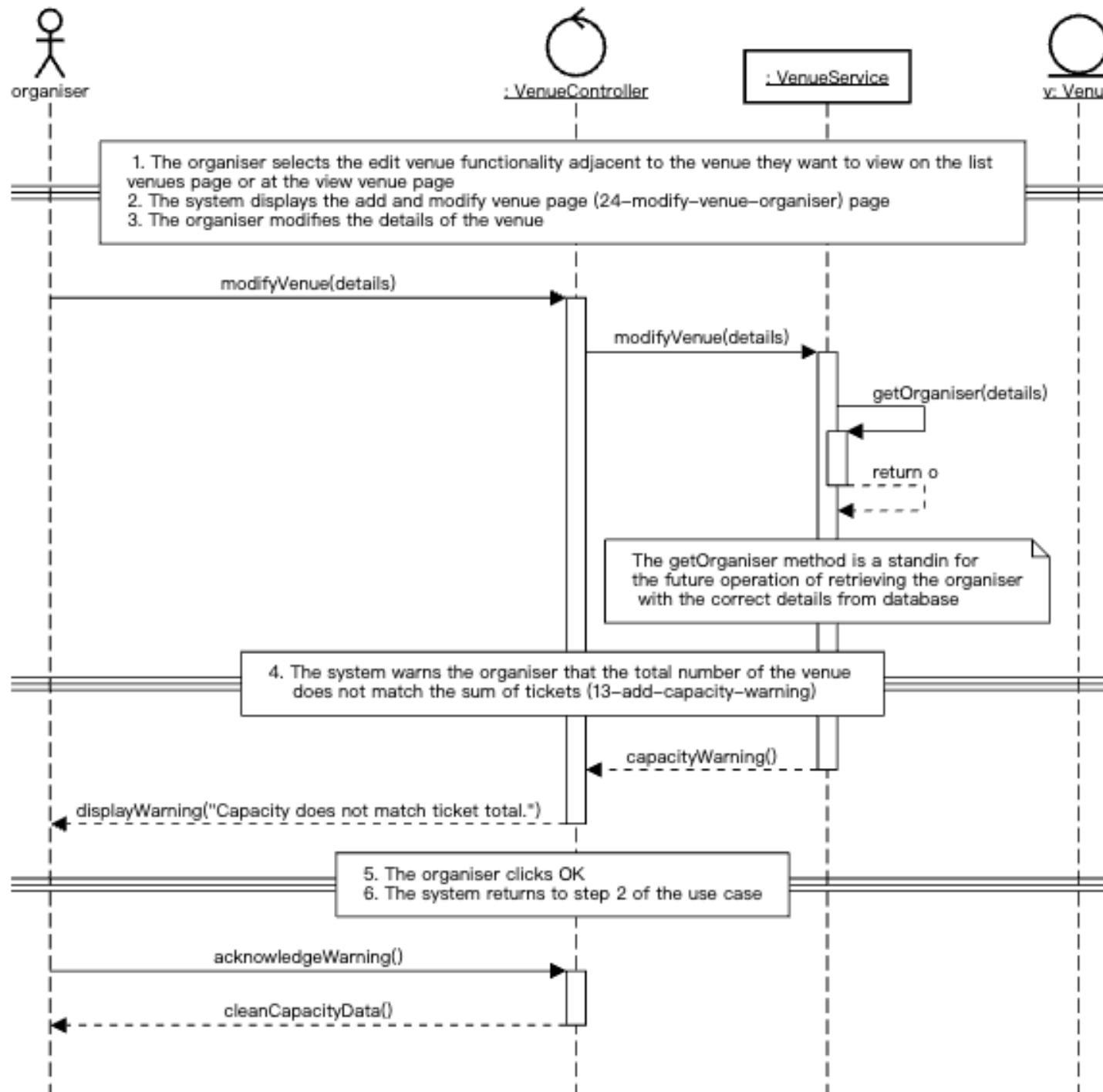
## 02 - Alternate Course of Events - The Organiser Not Save

Sequence Diagram: 09 Modify Venue (Alternate Course of Events – The Organiser Not Save)



## 03 - Alternate Course of Events - Incorrect Capacity

Sequence Diagram: 09 Modify Venue (Alternate Course of Events – Incorrect Capacity)



SD 16

## Related UI Prototypes

Page Name	Image
	<p>The UI prototype displays a header with navigation links: Home, Profile, My Events, and Venues (which is underlined, indicating it is the active page). To the right of the header is a search bar and a blue 'Search' button. Below the header is a 'New Venue' button. The main content area contains eight grey rectangular cards, each labeled 'Venue Name'. The cards are arranged in two rows of four. The first row contains 'Venue 1 Name', 'Venue 2 Name', 'Venue 3 Name', and 'Venue 4 Name'. The second row contains 'Venue 5 Name', 'Venue 6 Name', 'Venue 7 Name', and 'Venue 8 Name'.</p>

The List Venues Page(Organiser)

[Return](#)**Venue Name:****Address:****Total Capacity:****Price Classification:****Local Contact:****Name:****Phone Number:****Email:****Schedule:**

&lt;Event Date&gt; &lt;Event Time&gt;&lt;Event Name&gt;

&lt;Event Date&gt; &lt;Event Time&gt;&lt;Event Name&gt;

&lt;Event Date&gt; &lt;Event Time&gt;&lt;Event Name&gt;

[Edit](#)

The View Venue Page(Organiser)

[Return](#)**Venue Name:**

xxxx

[Change](#)**Address:**

xxxx

[Change](#)**Total Capacity:**

xxxx

[Change](#)**Price Classification:**

xxxx

[Change](#)**Local Contact:****Name:**

xxxx

[Change](#)**Phone Number:**

xxxx

[Change](#)**Email:**

xxxx

[Change](#)[Save](#)[Cancel](#)

The Add and Modify Venue Page

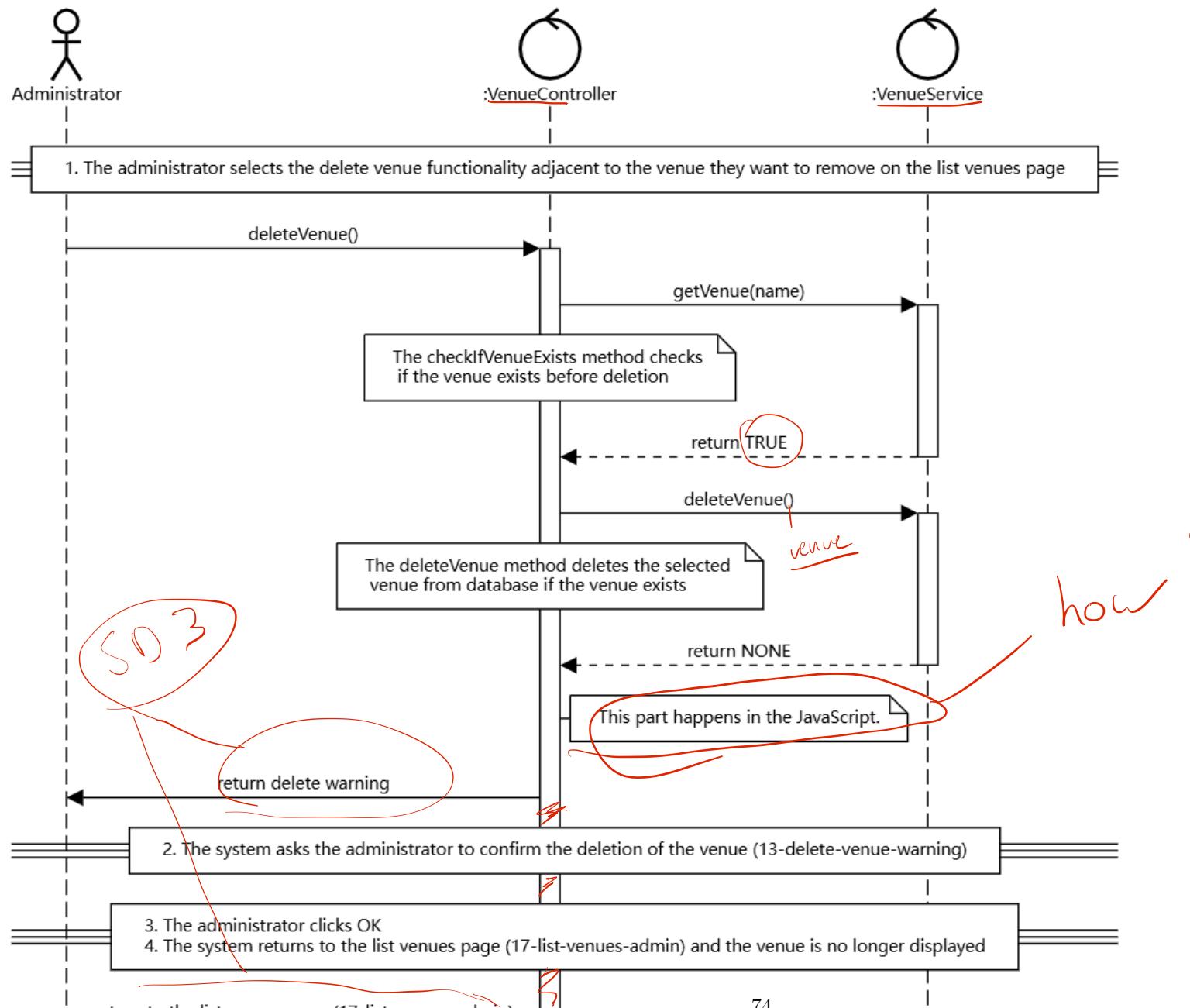
## Total Capacity Error

The total capacity of the venue must be equal to the sum of the capacities of the ticket types.

## Use Case 10 - Delete Venue

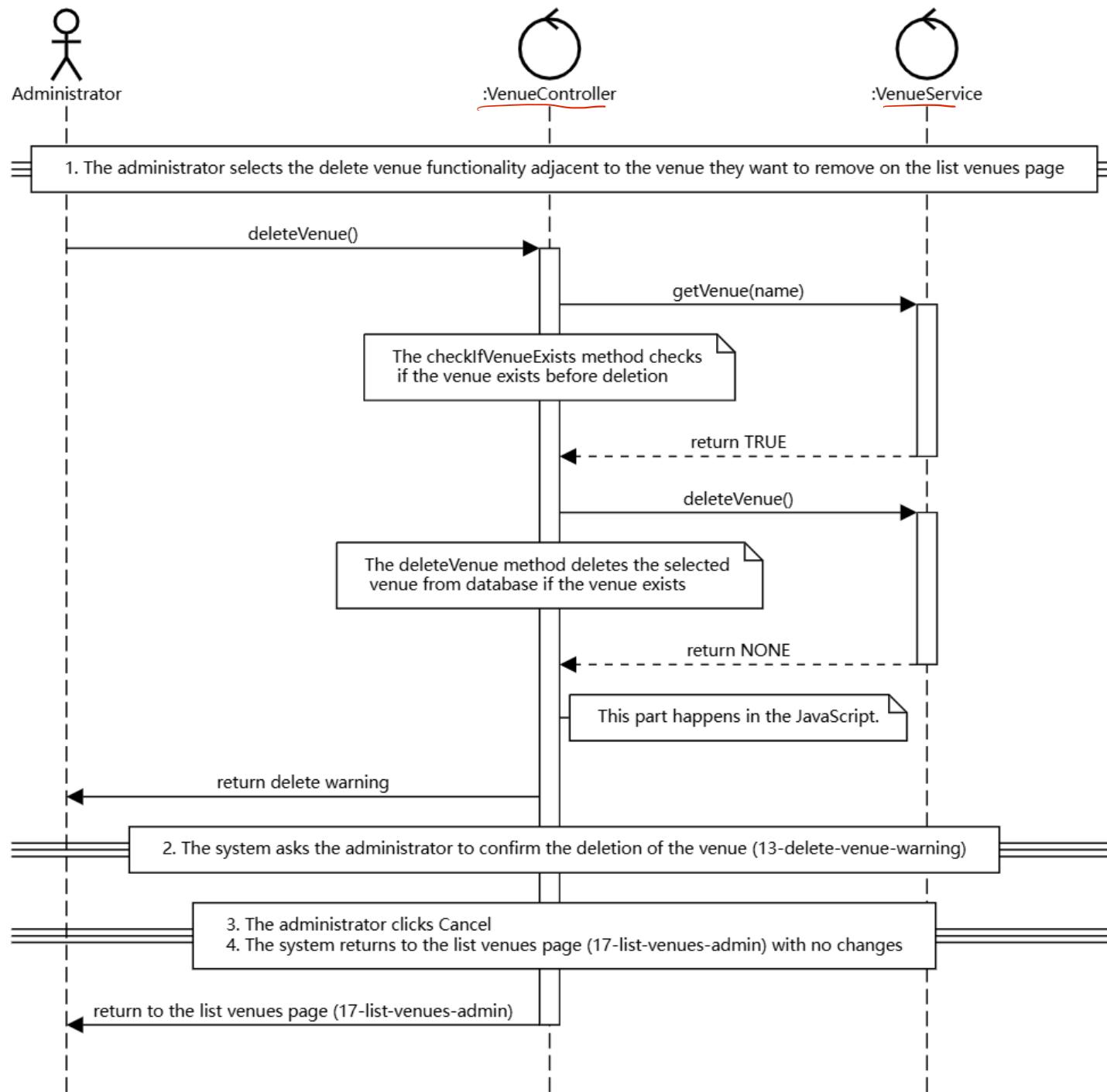
### 01 - Basic Course of Events

Sequence Diagram: 10 Delete Venue (Administrator Delete a Venue)



## 02 - Alternate Course of Events - Not Confirmed

Sequence Diagram: 10 Delete Venue (Alternate Course of Events - Not Confirmed)



SP'16

## Relevant UI Sketches

Page Name

Image

Home

Profile

Venues

Accounts

Search

Venue 1  
Name

Venue 2  
Name

Venue 3  
Name

Venue 4  
Name

Venue 5  
Name

Venue 6  
Name

Venue 7  
Name

Venue 8  
Name

List Venues Page (Admin)

Are you sure?

Are you sure you want to delete this venue? This action cannot be undone.

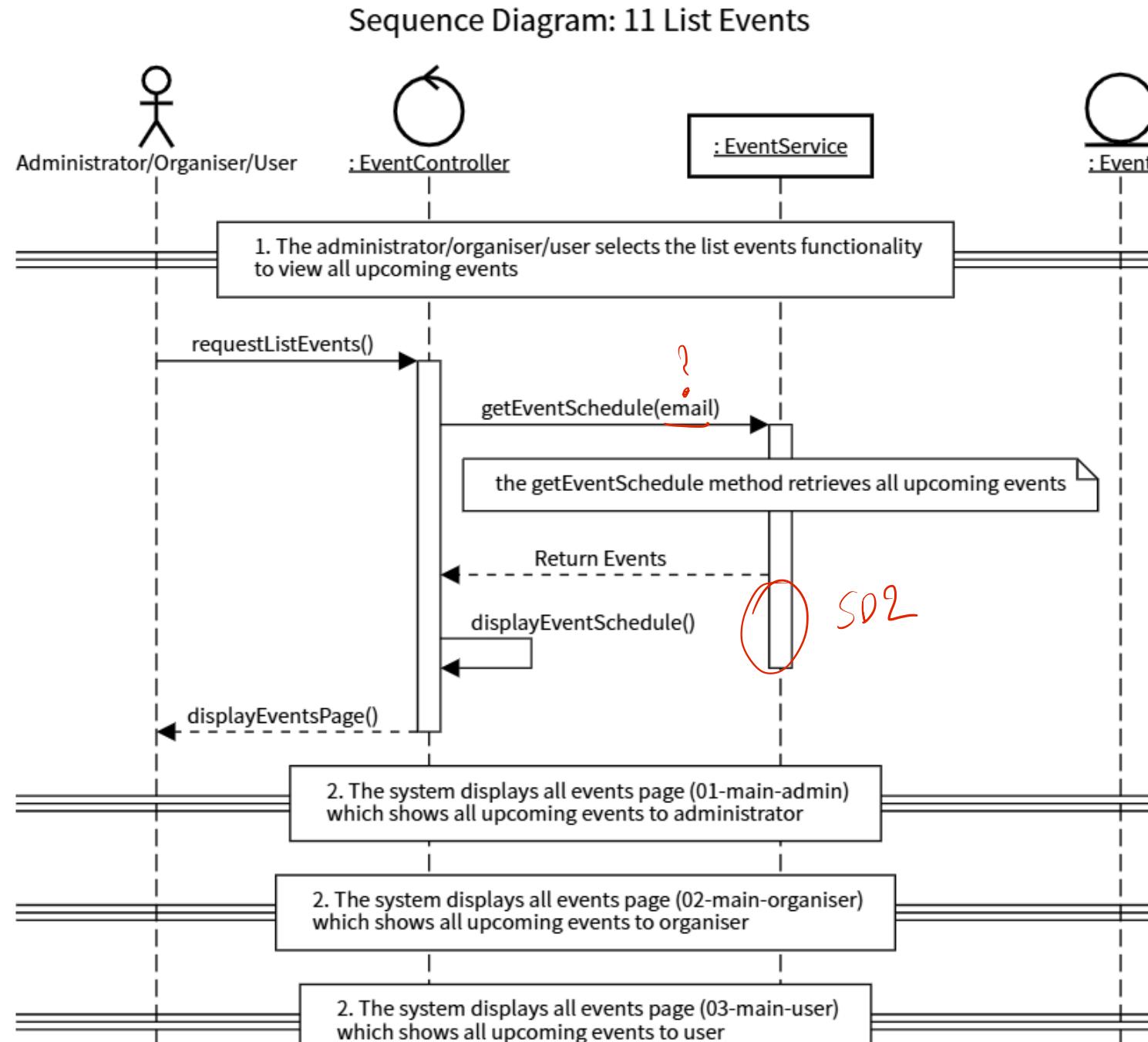
Cancel

Delete

Delete Venue Warning

## Use Case 11 - List Events

### 01 - Basic Course of Events



## Related UI Prototypes

Page Name	Image								
	<p>This image shows a user interface prototype for a home page. At the top, there is a navigation bar with five items: 'Home' (highlighted in blue), 'Profile', 'Venues', 'Accounts', a search input field, and a 'Search' button. Below the navigation, there are two rows of four rectangular cards each. Each card contains the text 'Event N Name' where N is a number from 1 to 8. The cards are arranged in a 2x4 grid.</p> <table border="1"><tbody><tr><td>Event 1 Name</td><td>Event 2 Name</td><td>Event 3 Name</td><td>Event 4 Name</td></tr><tr><td>Event 5 Name</td><td>Event 6 Name</td><td>Event 7 Name</td><td>Event 8 Name</td></tr></tbody></table>	Event 1 Name	Event 2 Name	Event 3 Name	Event 4 Name	Event 5 Name	Event 6 Name	Event 7 Name	Event 8 Name
Event 1 Name	Event 2 Name	Event 3 Name	Event 4 Name						
Event 5 Name	Event 6 Name	Event 7 Name	Event 8 Name						

01-main-admin

The image shows a wireframe of a web application interface. At the top, there is a navigation bar with five items: "Home" (which is underlined), "Profile", "My Events", "Venues", and a search bar followed by a "Search" button. Below the navigation bar, there are six gray rectangular boxes arranged in two rows of three. The top row contains two boxes labeled "Event 1 Name" and "Event 2 Name". The bottom row contains three boxes labeled "Event 3 Name", "Event 4 Name", and "Event 5 Name". To the right of the third box in the bottom row, there is another box labeled "Event 6 Name".

02-main-organiser

Home      Profile      Tickets

Search

Event 1  
Name

Event 2  
Name

Event 3  
Name

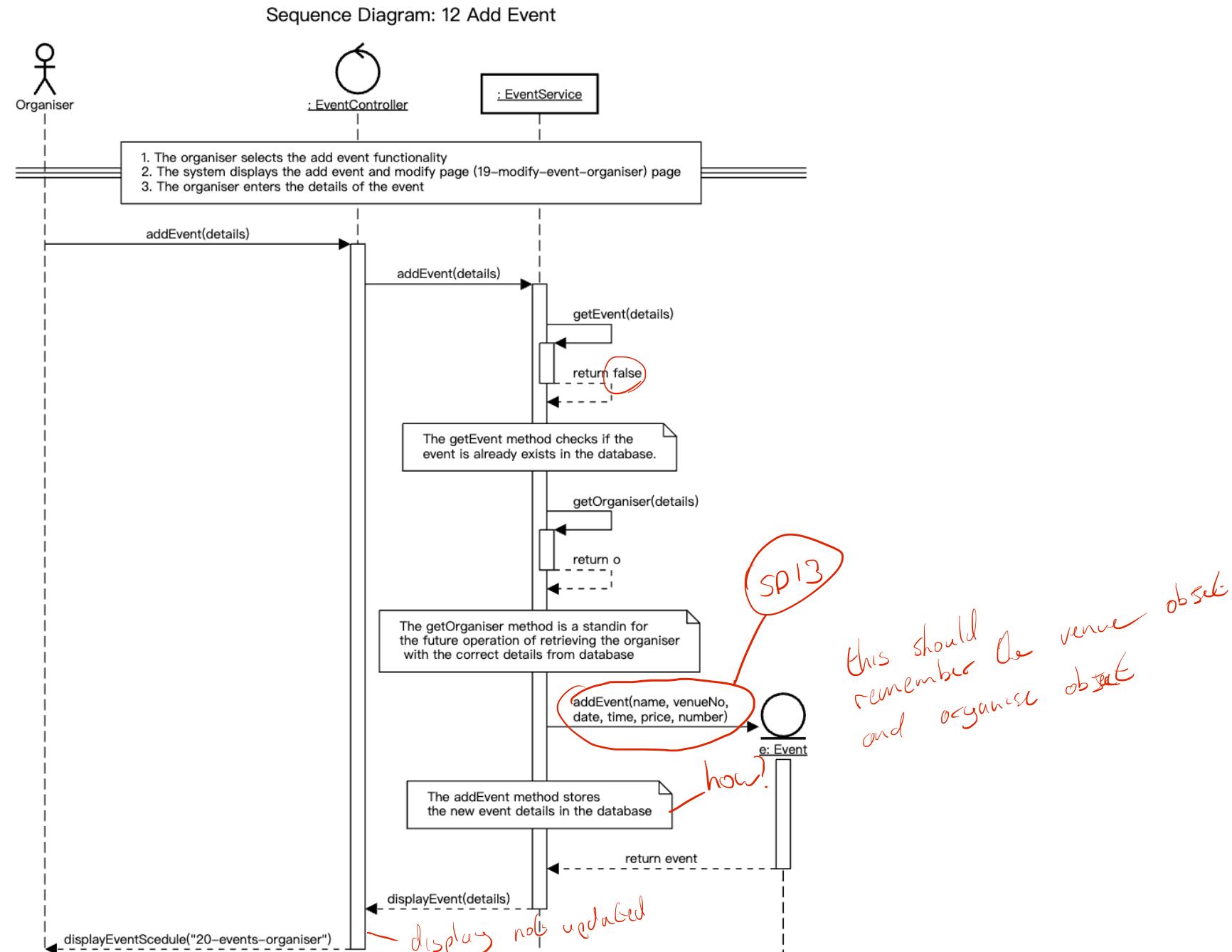
Event 4  
Name

Event 5  
Name

Event 6  
Name

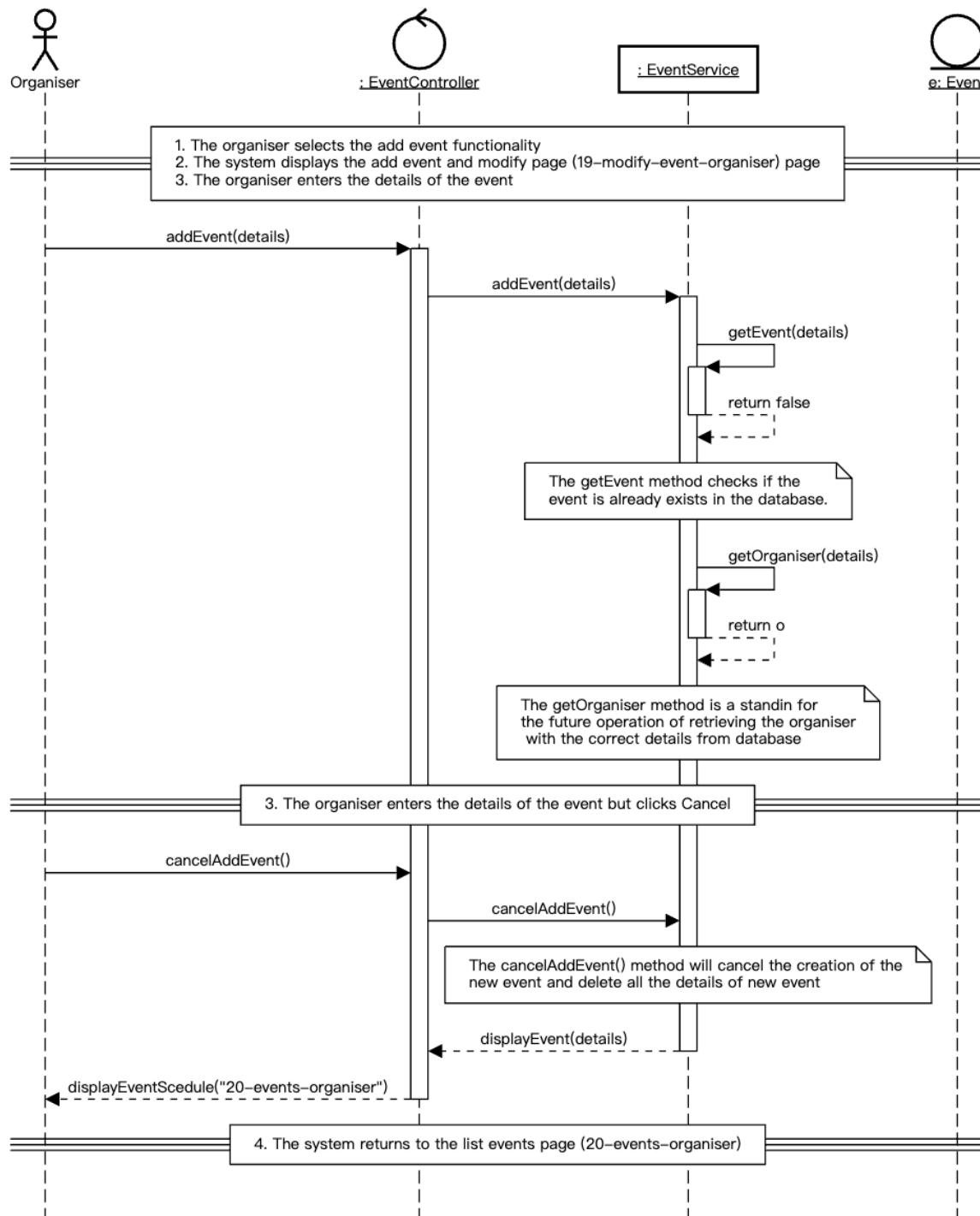
## Use Case 12 - Add Event

### 01 - Basic Course of Events



## 02 - Alternate Course of Events - The Organiser Not Save

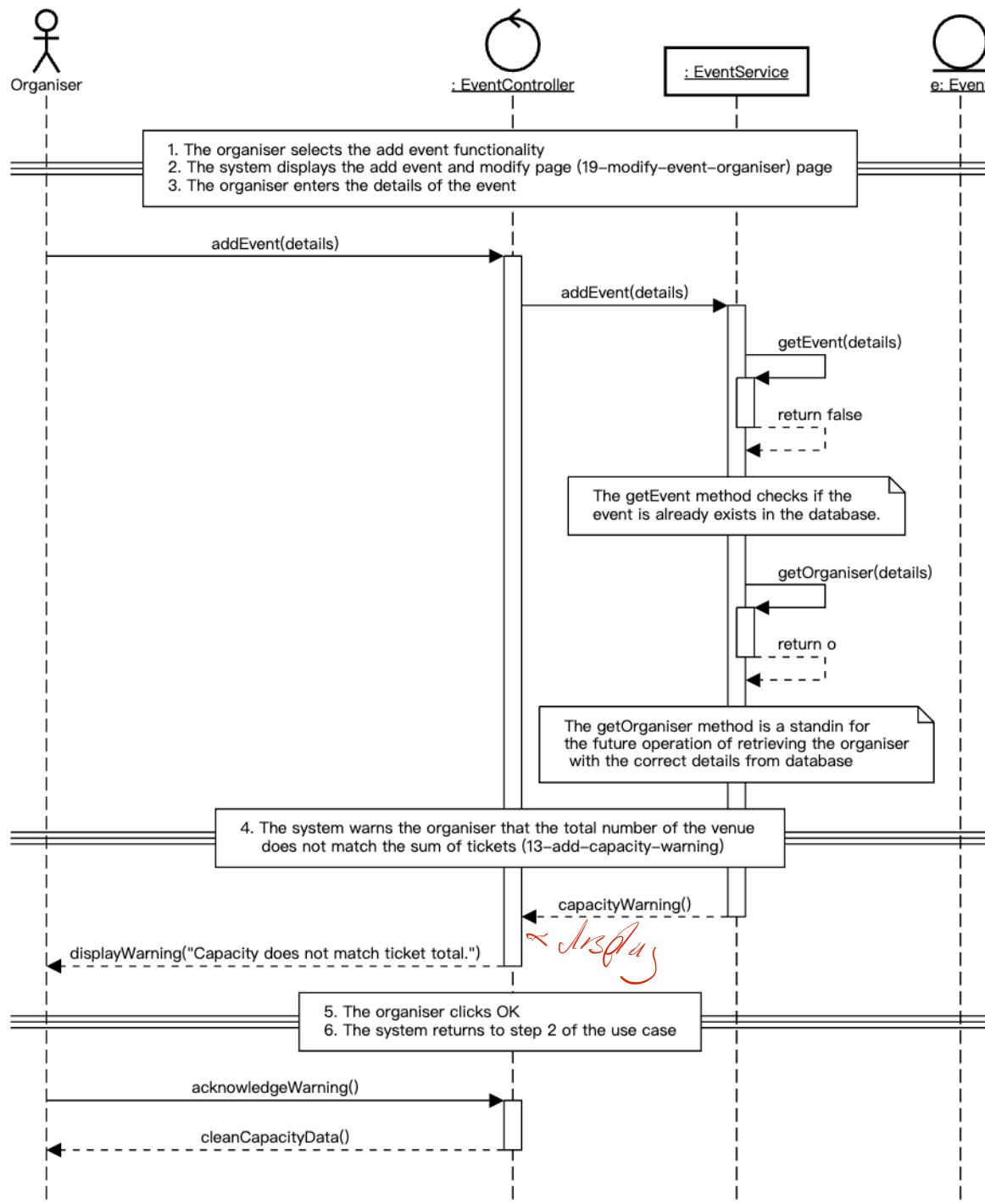
Sequence Diagram: 12 Add Event (Alternate Course of Events – The Organiser Not Save)



SD16

## 03 - Alternate Course of Events - Incorrect Capacity

Sequence Diagram: 12 Add Event (Alternate Course of Events – Incorrect Capacity)



## Related UI Prototypes

Page Name	Image
Organiser-Main-Page	<p>The image shows a user interface prototype for an 'Organiser-Main-Page'. At the top, there is a navigation bar with five items: 'Home' (underlined in blue), 'Profile', 'My Events', 'Venues', and a search bar with a 'Search' button. Below the navigation bar, there is a grid of six rectangular cards, each representing an event. The cards are arranged in two rows of three. The first row contains two cards: 'Event 1 Name' and 'Event 2 Name'. The second row contains three cards: 'Event 3 Name', 'Event 4 Name', 'Event 5 Name', and 'Event 6 Name'. All event cards have a light gray background and black text.</p>

[Return](#)

&lt;Description&gt;

xxxx

[Change](#)**Event Name:**

xxxx

[Change](#)**Venue No.**

Venue1

**Date:**

xxxx

[Change](#)**Time:**

xxxx

[Change](#)**Price:**[Change](#)**Total Number:**

xxxx

[Change](#)[Save](#)[Cancel](#)

The Add and Modify Event Page

Page Name	Image								
	<p>The navigation bar includes a search input field and a blue 'Search' button. A blue 'New Event' button is located below the navigation bar. The 'My Events' tab is currently active, indicated by a blue underline.</p> <p>Home      Profile      <a href="#">My Events</a>      Venues      <input type="text"/> <a href="#">Search</a></p> <p>New Event</p> <table border="1"><tbody><tr><td>Event 1 Name</td><td>Event 2 Name</td><td>Event 3 Name</td><td>Event 4 Name</td></tr><tr><td>Event 5 Name</td><td>Event 6 Name</td><td>Event 7 Name</td><td>Event 8 Name</td></tr></tbody></table>	Event 1 Name	Event 2 Name	Event 3 Name	Event 4 Name	Event 5 Name	Event 6 Name	Event 7 Name	Event 8 Name
Event 1 Name	Event 2 Name	Event 3 Name	Event 4 Name						
Event 5 Name	Event 6 Name	Event 7 Name	Event 8 Name						

The List Events Page(Organiser)

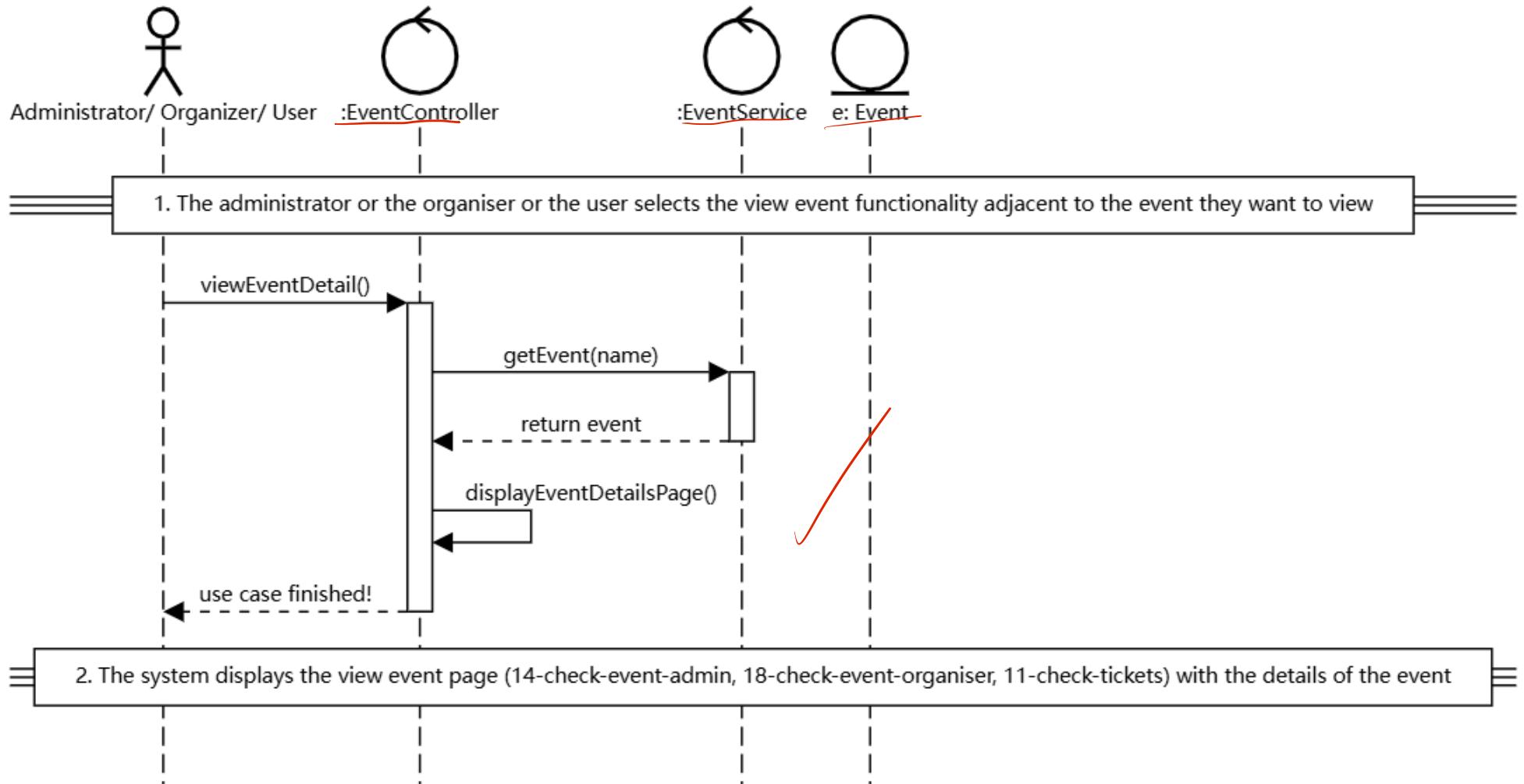
## Total Capacity Error

The total capacity of the venue must be equal to the sum of the capacities of the ticket types.

## Use Case 13 - View Event

### 01 - Basic Course of Events

Sequence Diagram: 13 View Event (Administrator, Organiser or User Viewing an Upcoming Event Details)



Relevant UI Sketches

The screenshot displays a user interface for a web application. At the top, there is a horizontal navigation bar with four items: "Home", "Profile", "Venues", and "Accounts". The "Home" item is underlined, indicating it is the active page. To the right of the navigation bar is a search bar with a placeholder and a blue "Search" button. Below the navigation bar, there is a grid of eight light gray rectangular cards, arranged in two rows of four. Each card contains the text "Event" followed by a number and the word "Name". The first row contains "Event 1 Name", "Event 2 Name", "Event 3 Name", and "Event 4 Name". The second row contains "Event 5 Name", "Event 6 Name", "Event 7 Name", and "Event 8 Name".

Admin Main Page

The wireframe illustrates the 'Organiser Main Page' layout. At the top, there is a horizontal navigation bar with five items: 'Home' (underlined in blue), 'Profile', 'My Events', 'Venues', and a search bar. Below the navigation bar, there are six rectangular boxes arranged in two rows of three. Each box contains the text 'Event' followed by a number and 'Name'. The first row contains 'Event 1 Name' and 'Event 2 Name'. The second row contains 'Event 3 Name', 'Event 4 Name', 'Event 5 Name', and 'Event 6 Name'.

Home

Profile

My Events

Venues

Search

Event 1  
Name

Event 2  
Name

Event 3  
Name

Event 4  
Name

Event 5  
Name

Event 6  
Name

Organiser Main Page

[Home](#)[Profile](#)[Tickets](#)

Event 1  
Name

Event 2  
Name

Event 3  
Name

Event 4  
Name

Event 5  
Name

Event 6  
Name

User Main Page

[Return](#)

<Event Name>

<Venues No.>

Date:

Time:

Price:

<Description>

Total Number:

Sold Number:

[Delete](#)

[View Event Page \(Admin\)](#)

Return



<Event Name>

<Venues No.>

Date:

Time:

Price:

<Description>

Total Number:

Sold Number:

[View Event Page \(Organiser\)](#)

Return



<Event Name>

<Venues No.>

<Organizer>

Date:

Time:

<Description>

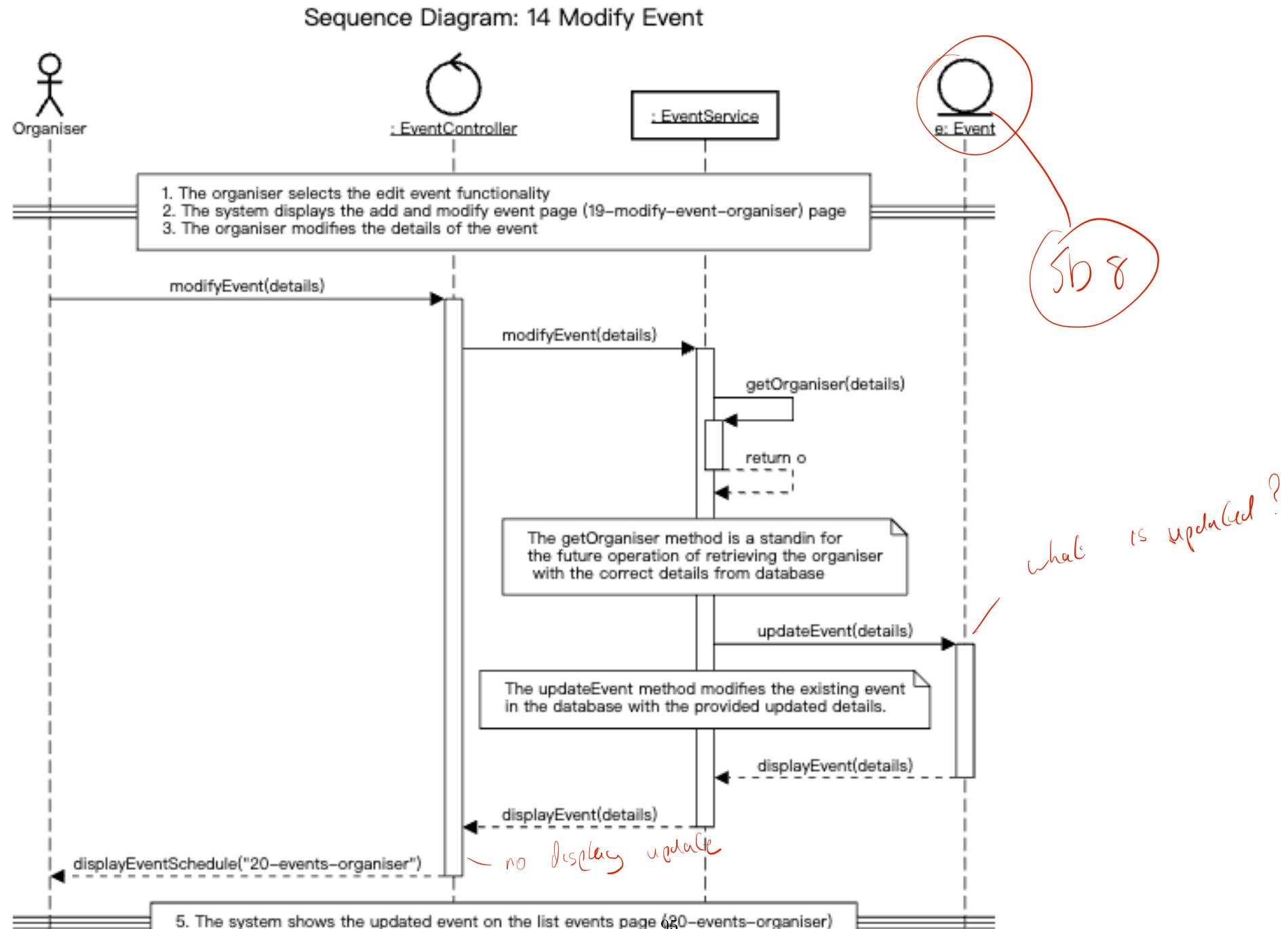
Price:

Number:

Seat:

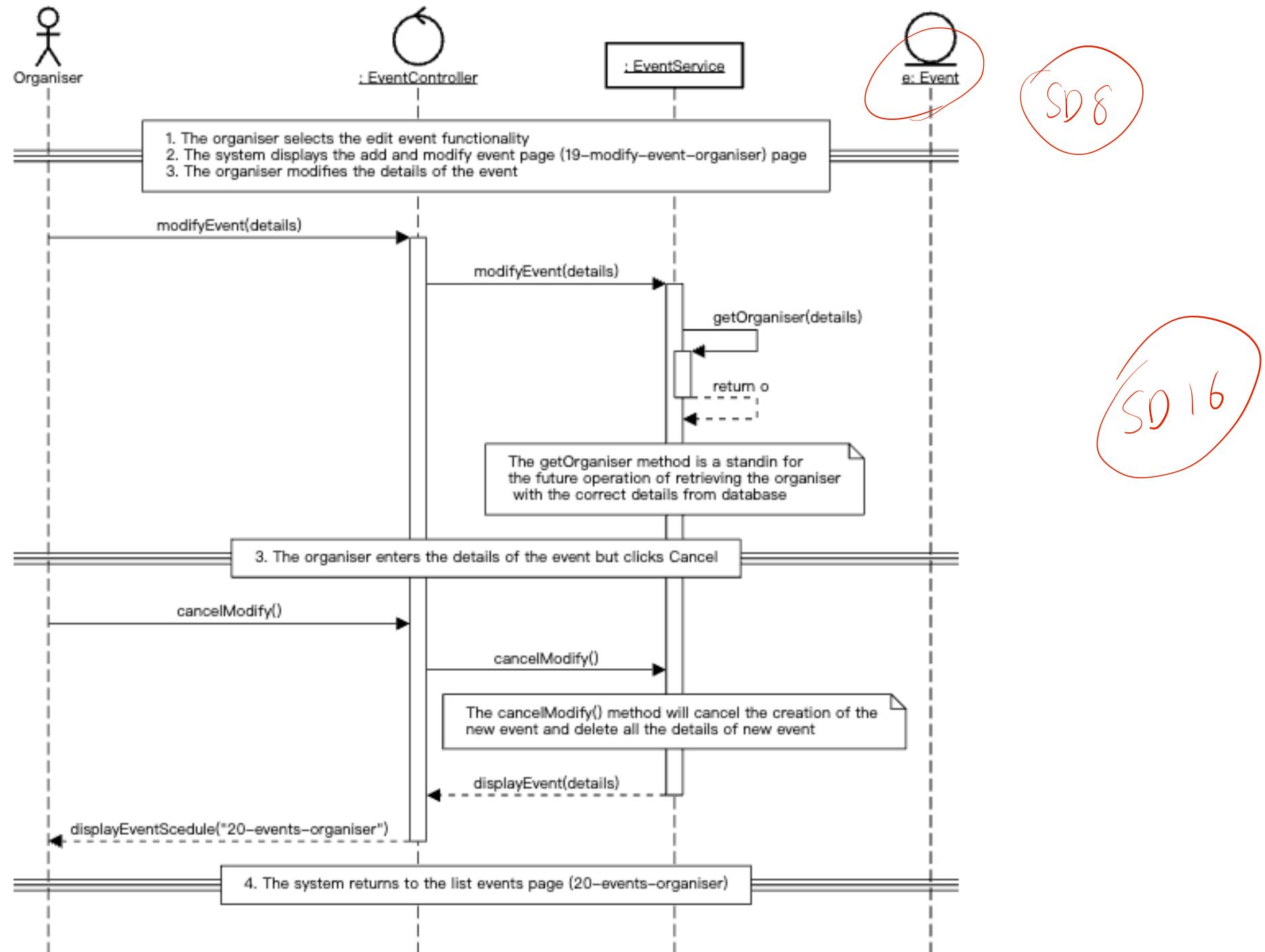
## Use Case 14 - Modify Event

### 01 - Basic Course of Events



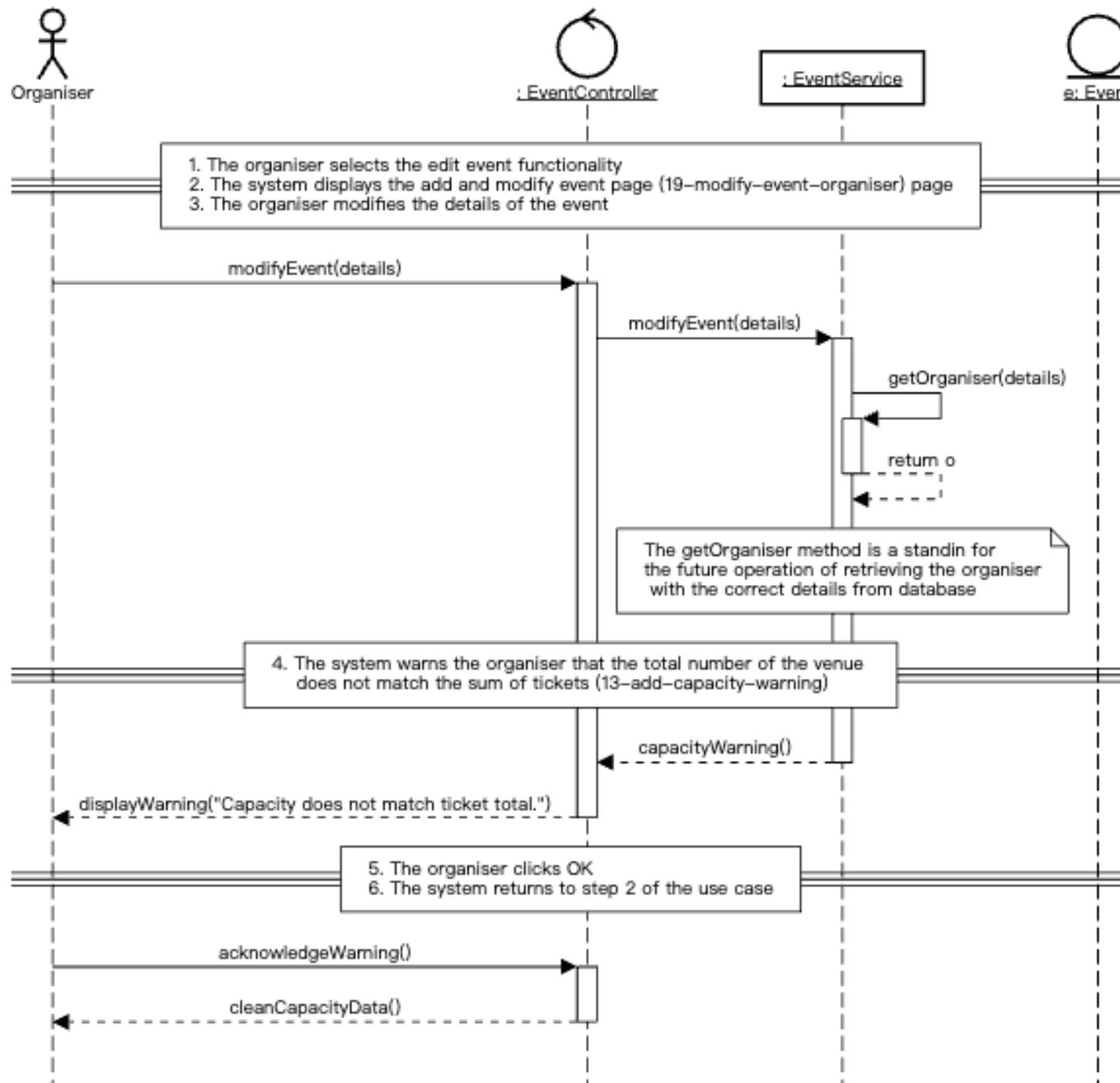
## 02 - Alternate Course of Events - The Organiser Not Save

Sequence Diagram: 14 Modify Event (Alternate Course of Events – The Organiser Not Save)

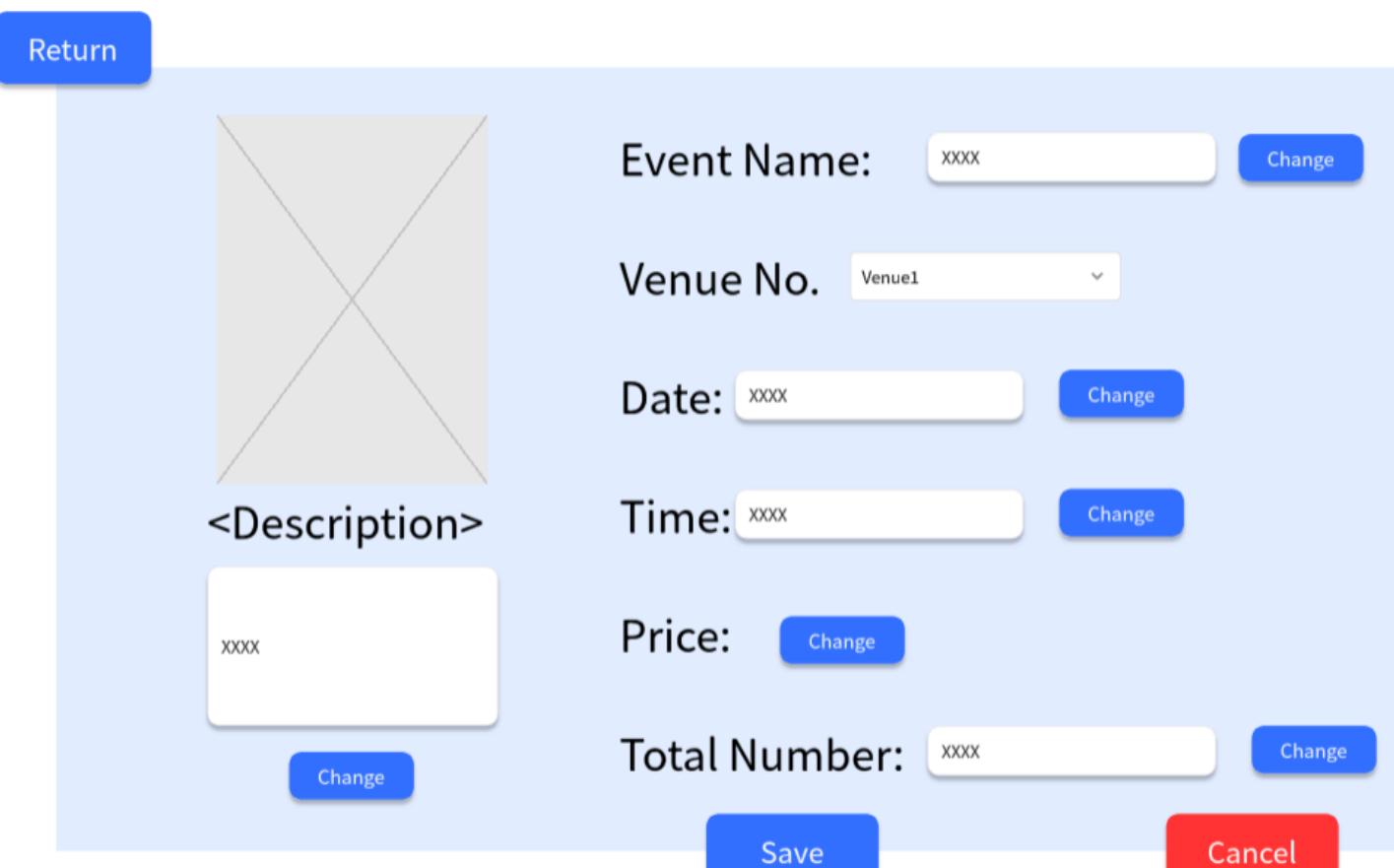


## 03 - Alternate Course of Events - Incorrect Capacity

Sequence Diagram: 14 Modify Event (Alternate Course of Events – Incorrect Capacity)



## Related UI Prototypes

Page Name	Image
	 <p>The UI prototype for the 'Add and Modify Event' page features a light blue background. In the top-left corner is a blue rounded rectangle button labeled 'Return'. To its right is a large rectangular area containing a placeholder image with a large 'X' drawn through it. Below this image is a placeholder text field labeled '&lt;Description&gt;' with the value 'xxxx' and a 'Change' button. On the right side of the screen, there are five input fields with 'xxxx' placeholder values, each accompanied by a 'Change' button: 'Event Name:', 'Venue No.', 'Date:', 'Time:', and 'Total Number:'. At the bottom of the page are two large buttons: a blue 'Save' button and a red 'Cancel' button.</p>

The Add and Modify Event Page

Page Name	Image
	<p>The header features a navigation bar with the following items: Home, Profile, My Events (highlighted with a blue underline and a blue horizontal line below it), Venues, a large empty search input field, and a blue "Search" button. To the right of the search bar is a blue rounded rectangle containing the text "New Event".</p> <p>Below the header are two rows of four light gray rectangular boxes each, representing event cards. The first row contains "Event 1 Name", "Event 2 Name", "Event 3 Name", and "Event 4 Name". The second row contains "Event 5 Name", "Event 6 Name", "Event 7 Name", and "Event 8 Name".</p>

The List Events Page(Organiser)

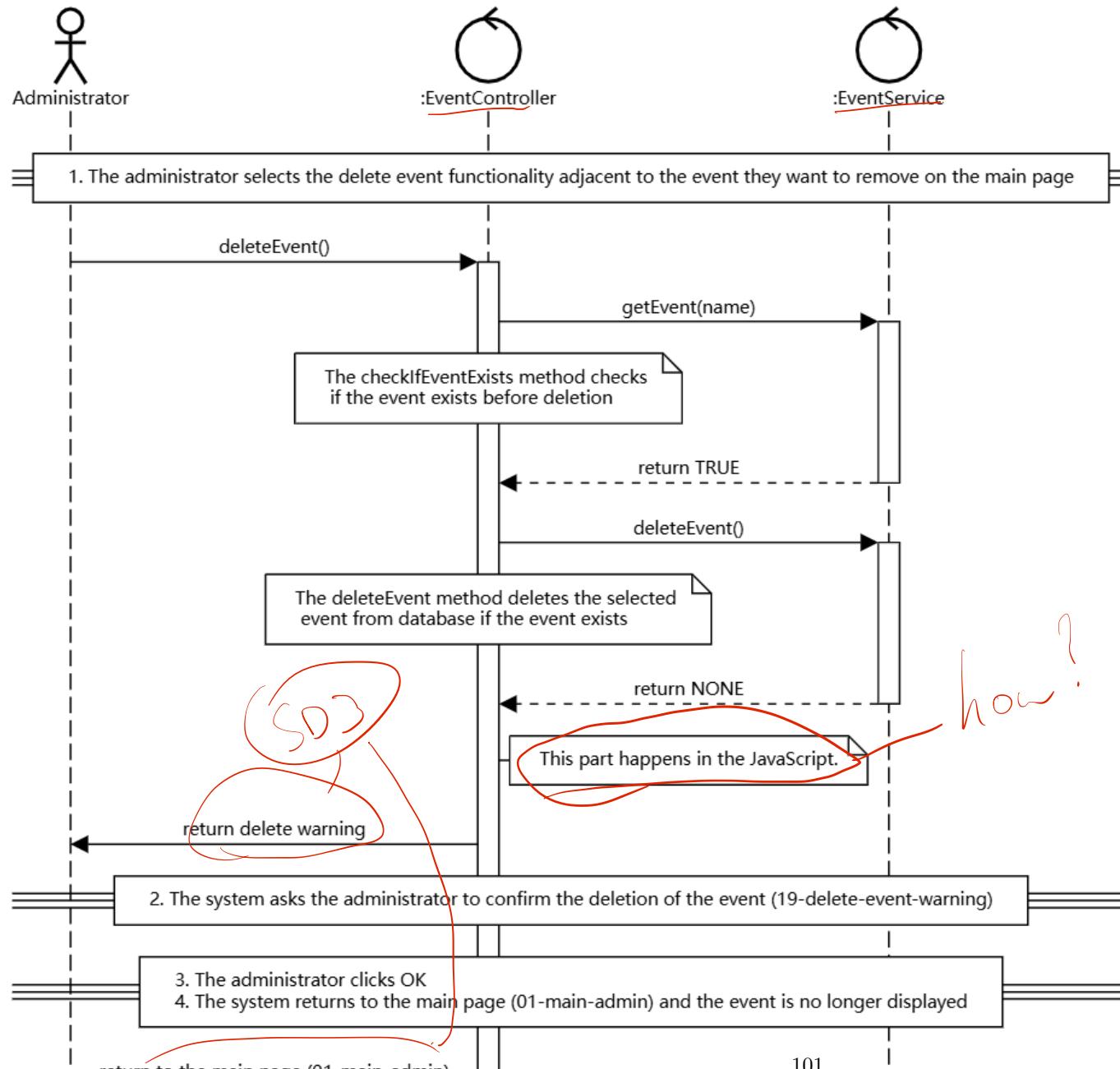
## Total Capacity Error

The total capacity of the venue must be equal to the sum of the capacities of the ticket types.

## Use Case 15 - Delete Event

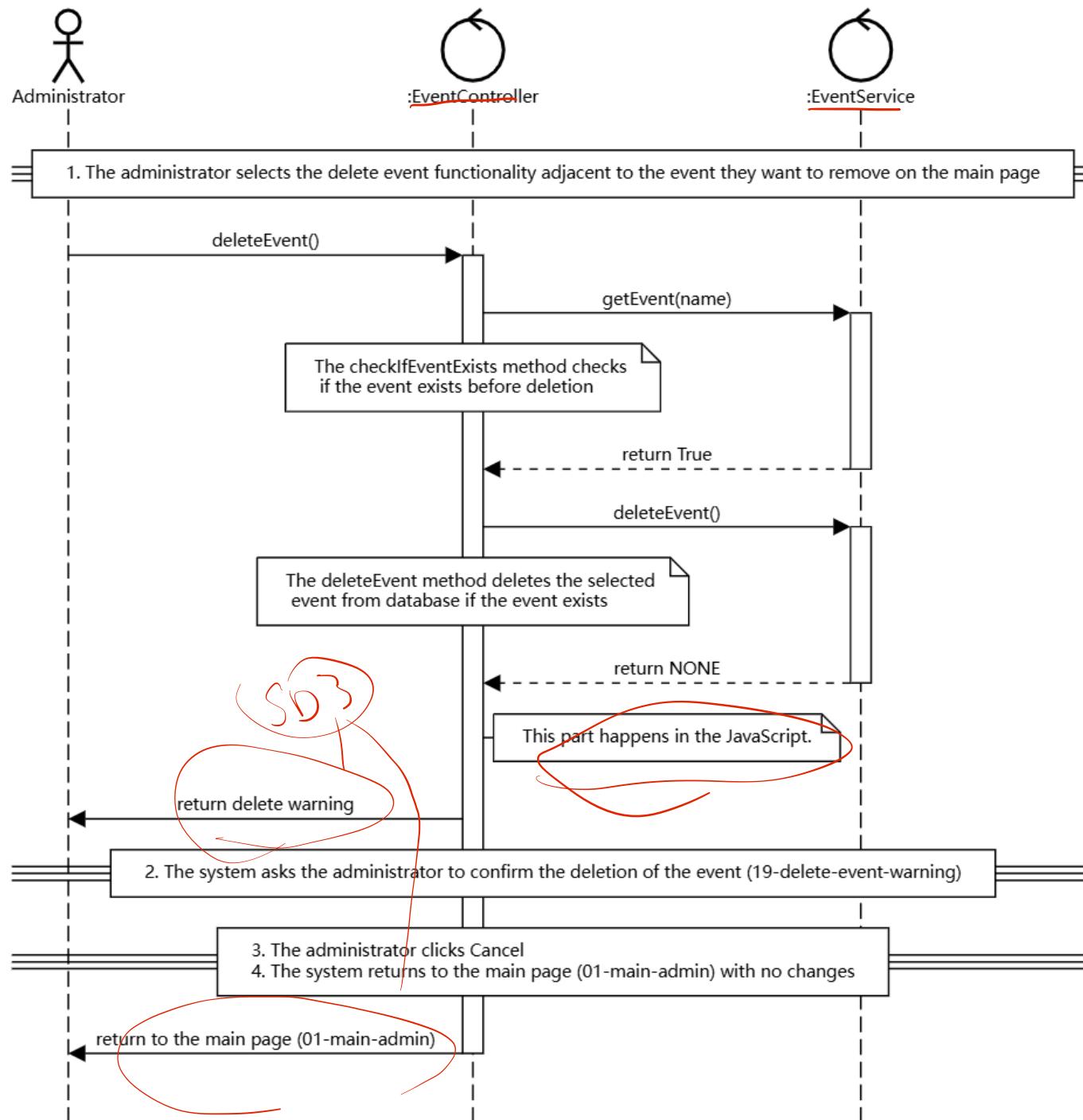
### 01 - Basic Course of Events

Sequence Diagram: 15 Delete Event (Administrator Delete an Event)



## 02 - Alternate Course of Events - Not Confirmed

Sequence Diagram: 15 Delete Event (Alternate Course of Events - Not Confirmed)



## Relevant UI Sketches

Page Name	Image
Admin Main Page	<p>The UI sketch depicts a top navigation bar with four items: 'Home' (highlighted in blue), 'Profile', 'Venues', and 'Accounts'. To the right of the navigation is a search bar containing the placeholder text 'Search'. Below the navigation, there are two rows of four rectangular cards each. Each card contains the text 'Event [number] Name' where [number] corresponds to the card's position. The first row contains 'Event 1 Name', 'Event 2 Name', 'Event 3 Name', and 'Event 4 Name'. The second row contains 'Event 5 Name', 'Event 6 Name', 'Event 7 Name', and 'Event 8 Name'.</p>

**Are you sure?**

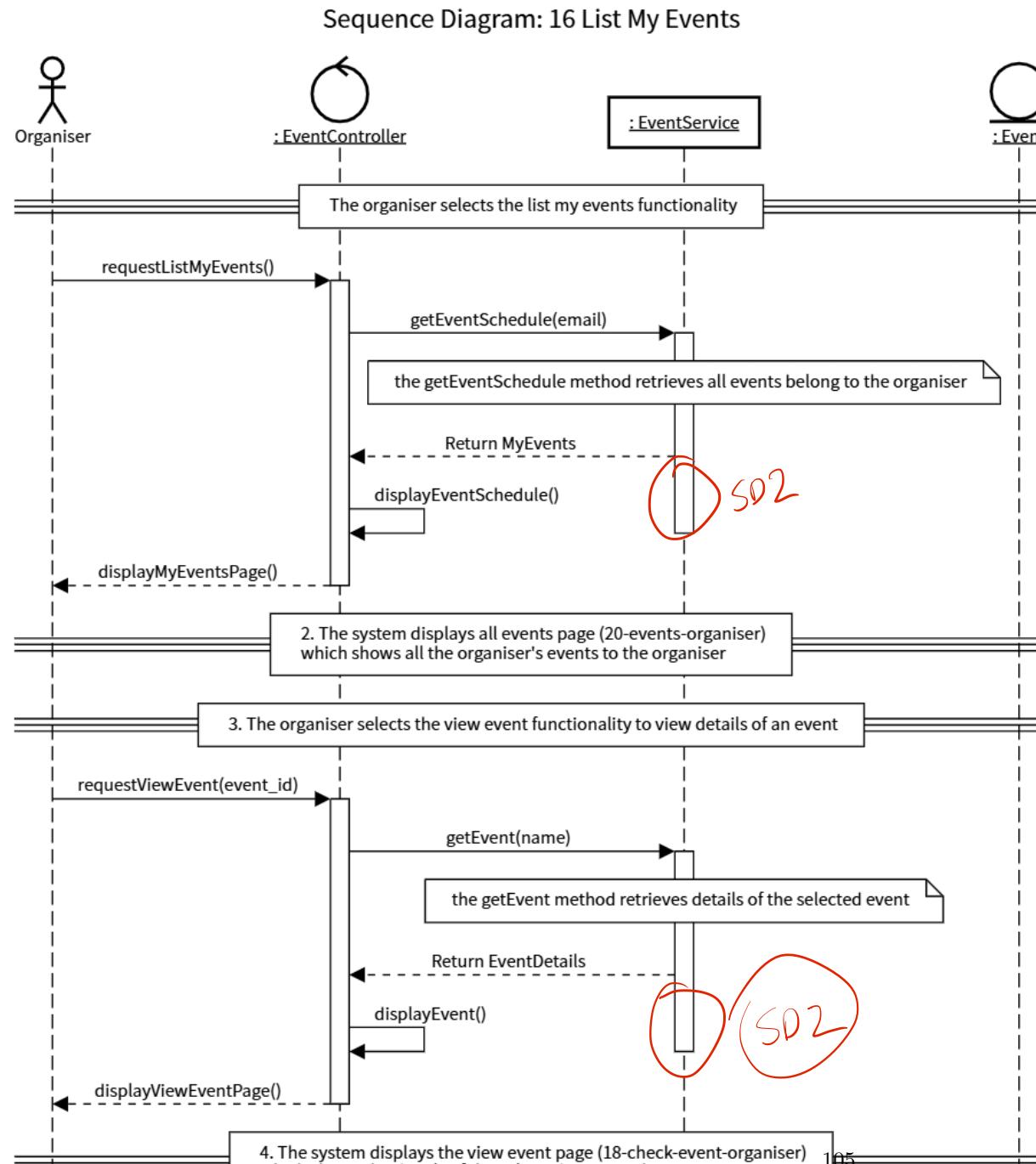
Are you sure you want to delete this event? This action cannot be undone.

**Cancel**

**Delete**

## Use Case 16 - List My Events

### 01 - Basic Course of Events



## Related UI Prototypes

Page Name

Image

---

Return



<Event Name>

<Venues No.>

Date:

Time:

Price:

<Description>

Total Number:

Sold Number:

18-check-event-organiser

[Home](#)[Profile](#)[My Events](#)[Venues](#)[Search](#)[New Event](#)

Event 1  
Name

Event 2  
Name

Event 3  
Name

Event 4  
Name

Event 5  
Name

Event 6  
Name

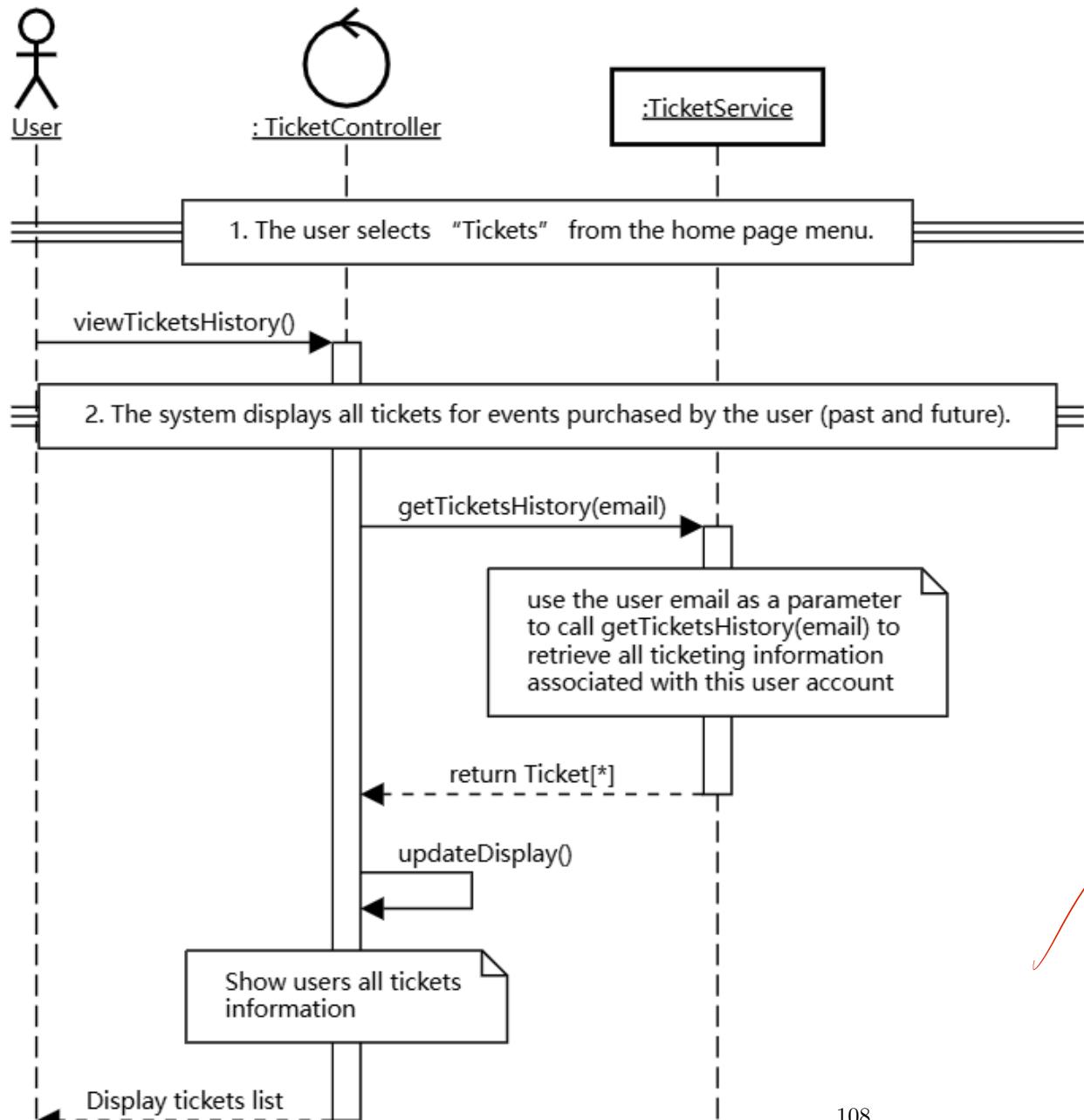
Event 7  
Name

Event 8  
Name

## Use Case 17 - List Tickets

### 01 - Basic Course of Events

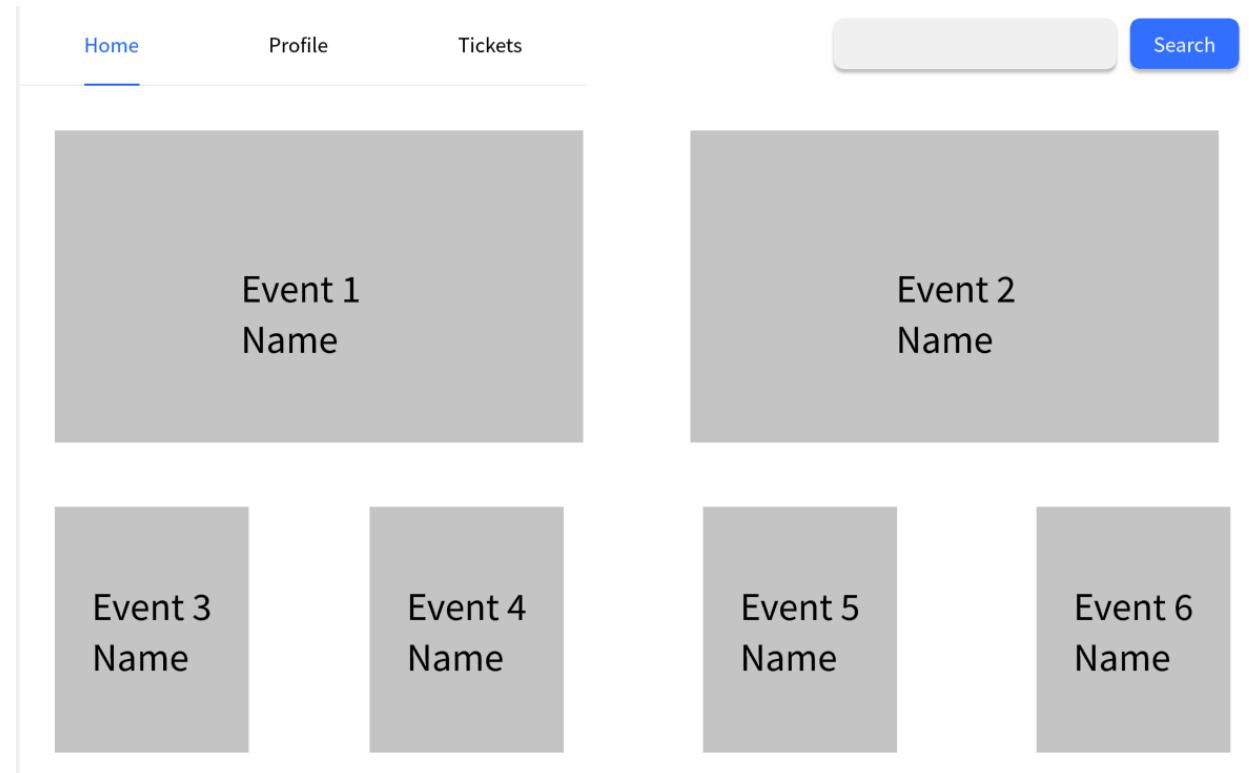
Sequence Diagram: 17 List Tickets



## Related UI Prototypes

Page Name

Image



Main User Page

Home

Profile

Tickets

---

Search

Event 1  
Name

Event 2  
Name

Event 3  
Name

Event 4  
Name

Event 5  
Name

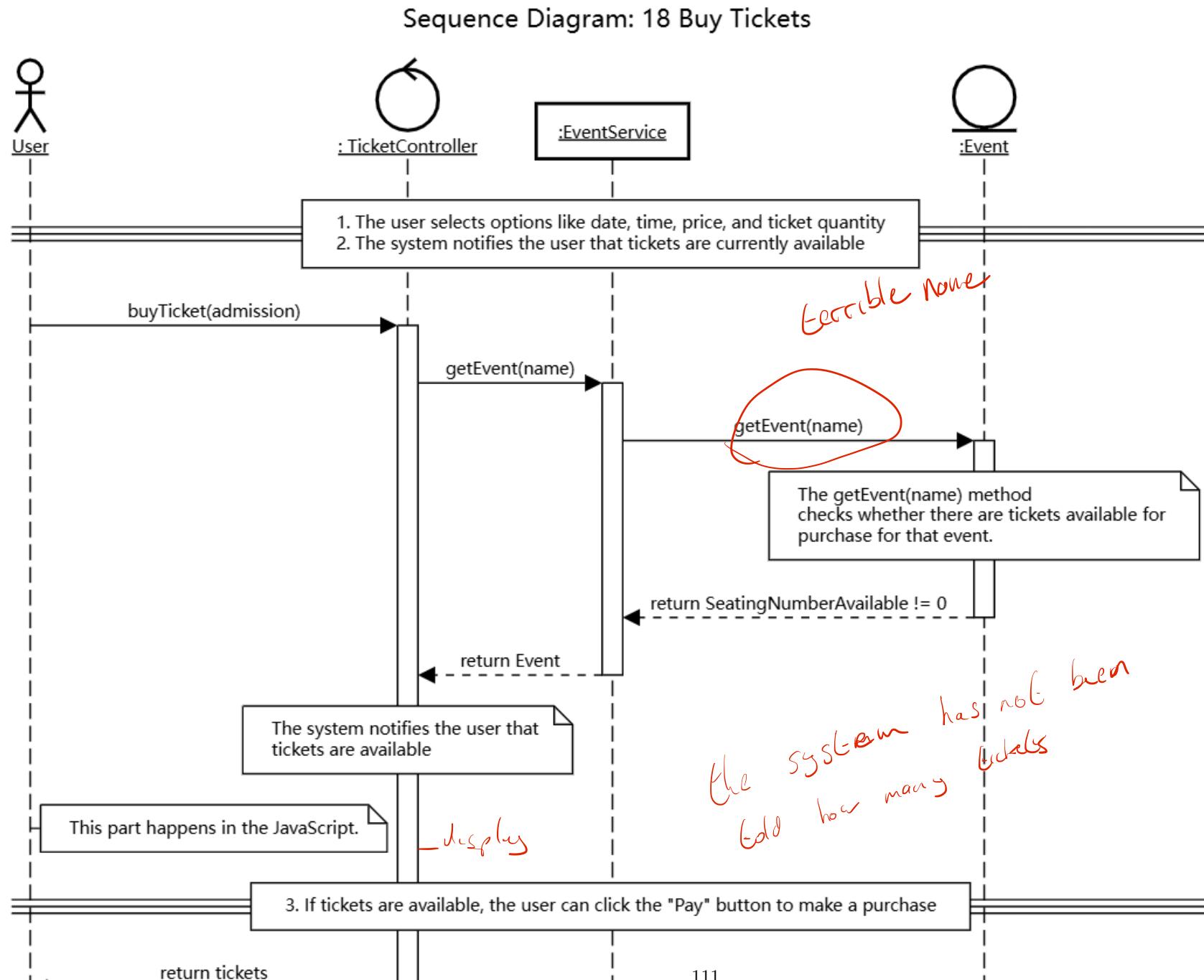
Event 6  
Name

Event 7  
Name

Event 8  
Name

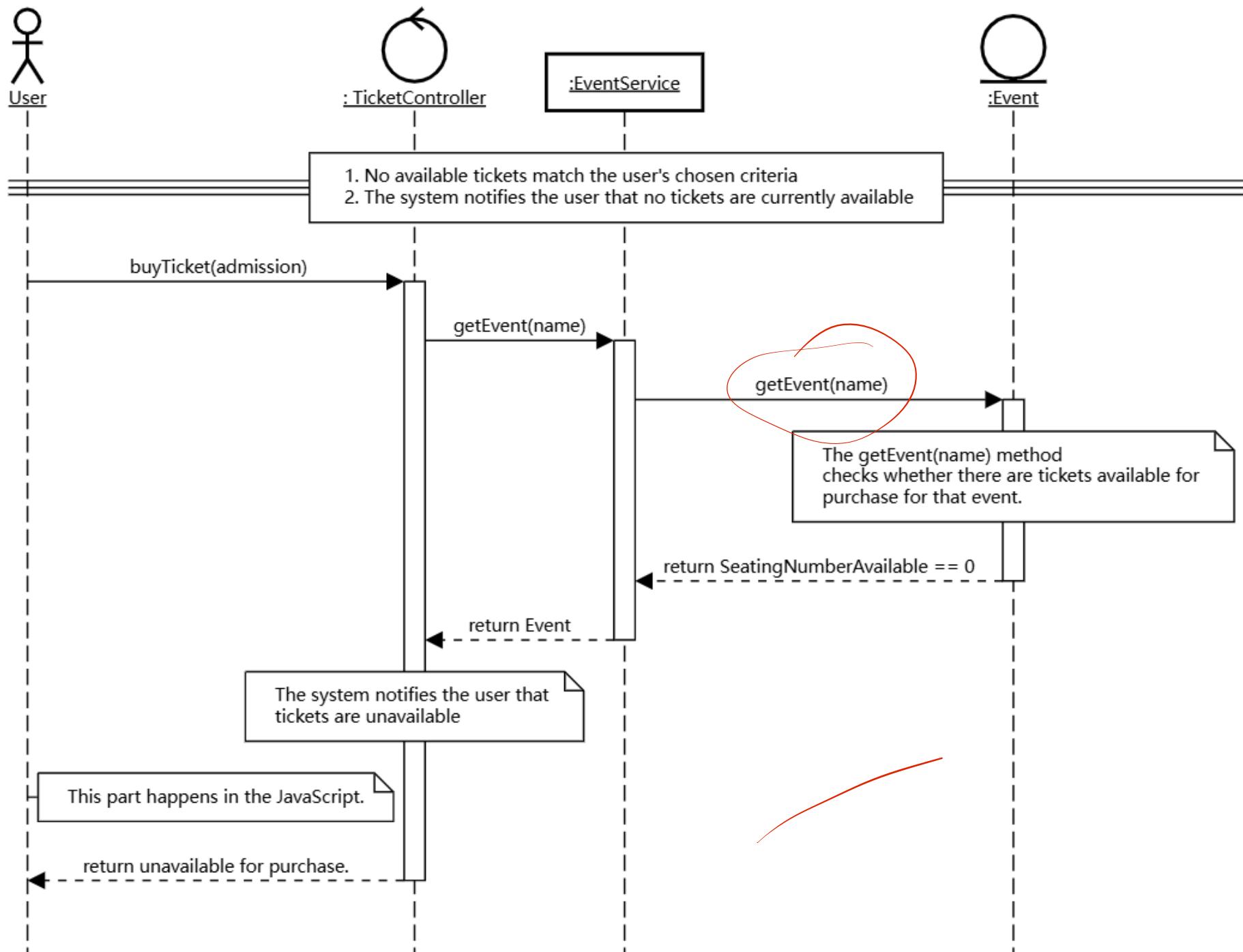
## Use Case 18 - Buy Tickets

### 01 - Basic Course of Events



## 02 - Alternative Courses of Events - Tickets Not Available

### Sequence Diagram: 18 Buy Tickets(Alternative Courses of Events - Tickets Not Available)



## Related UI Prototypes

Page Name

Image

Return



<Event Name>

<Venues No.>

<Organizer>

Date: Date1

Time: Time1

Price: Price1

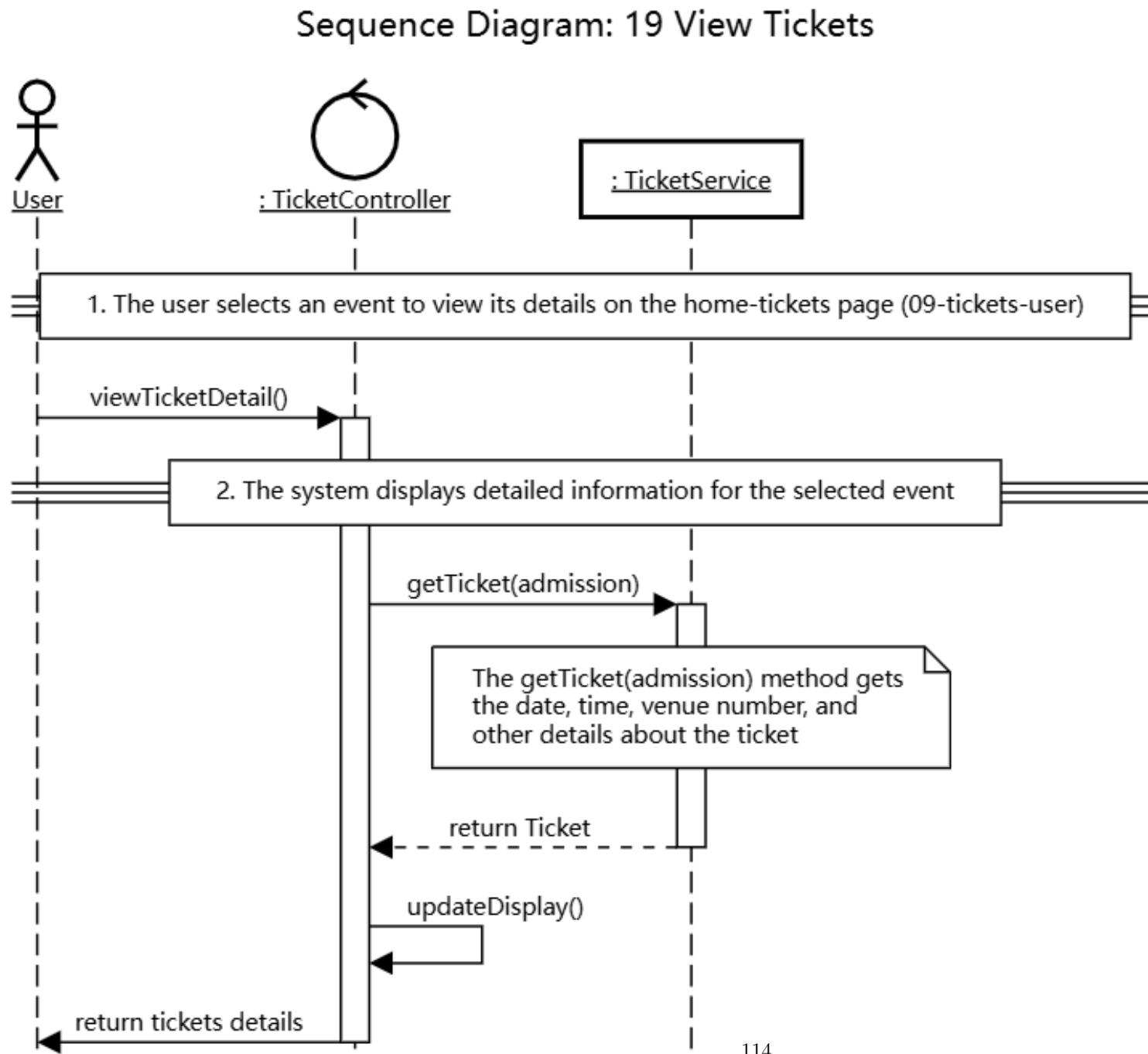
Number: - 1 +

Pay

Buy Tickets Page

## Use Case 19 - View Tickets

### 01 - Basic Course of Events



## Related UI Prototypes

Page Name

Image

Home

Profile

Tickets

Search

Event 1  
Name

Event 2  
Name

Event 3  
Name

Event 4  
Name

Event 5  
Name

Event 6  
Name

Event 7  
Name

Event 8  
Name

Tickets User Page

Return



<Description>

<Event Name>

<Venues No.>

<Organizer>

Date:

Time:

Price:

Number:

Seat: