

Use Case View

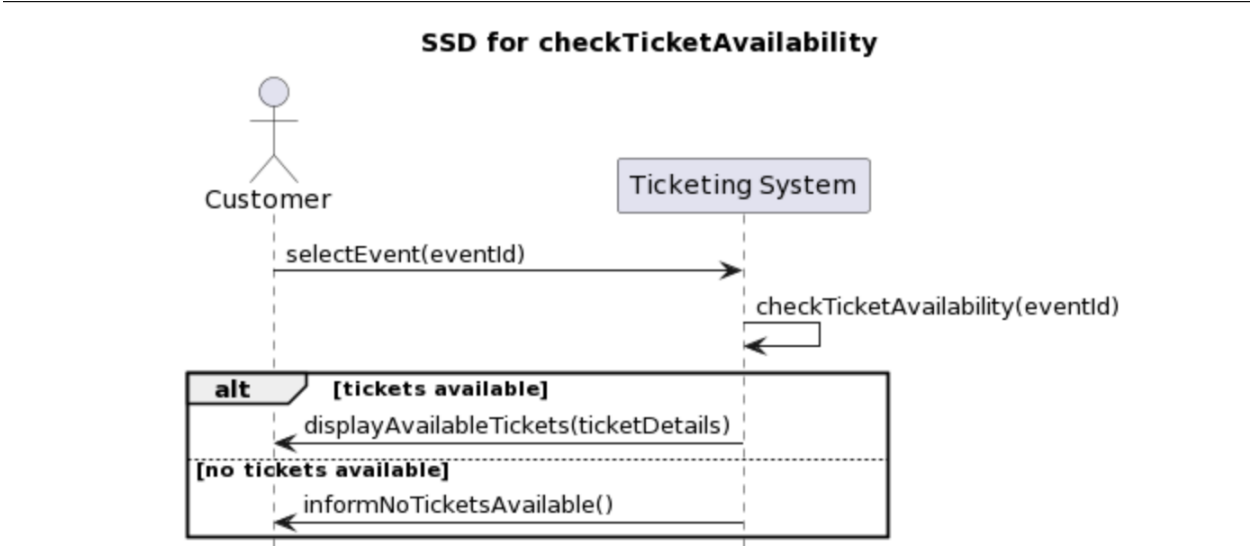
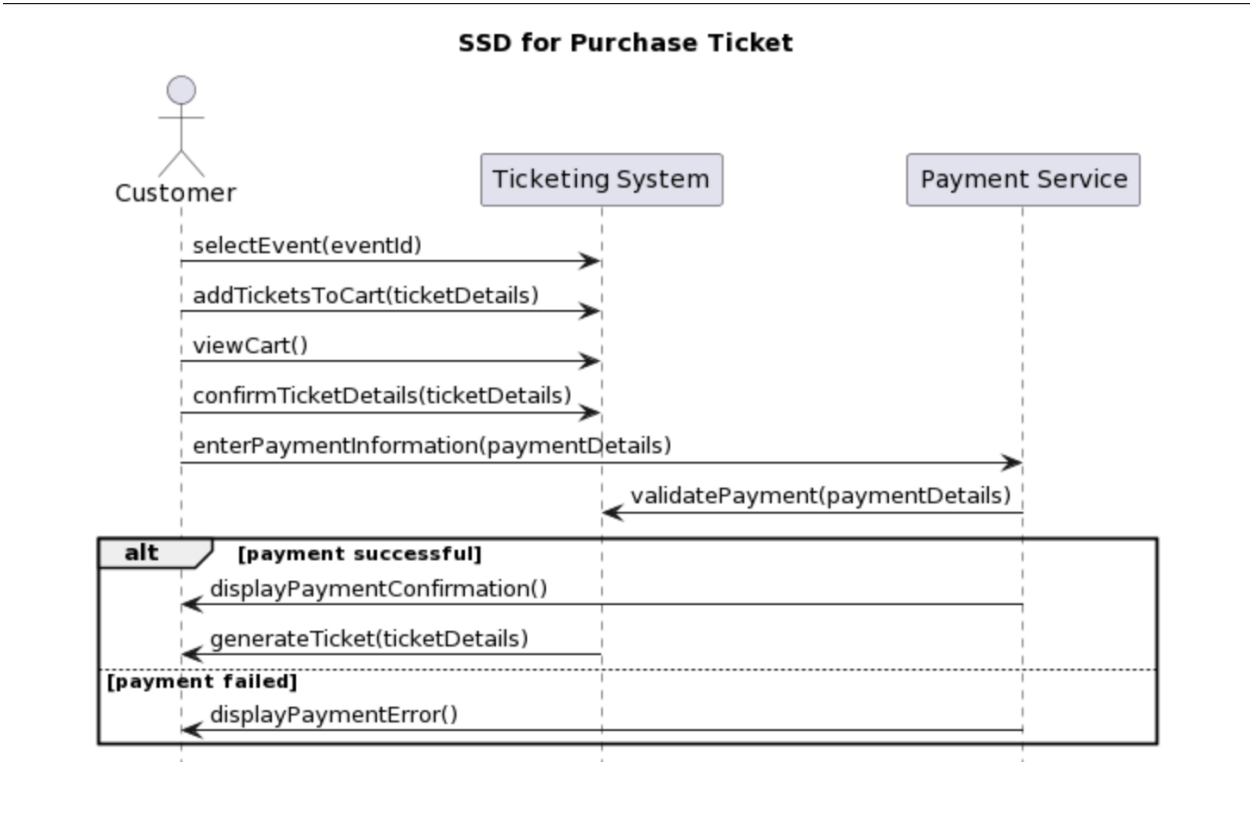
Fully Dressed Use Case #1

Use Case Name:	Checking Availability
Scope:	Ticketing System
Level:	“User - Goal” or subfunction
Stakeholders and Interest:	Attendee: wants to quickly find out if tickets are available for an event. Event Organizer: needs to monitor ticket sales and availability
Primary Actor:	Any person trying to get tickets for a CofC event
Preconditions:	There must be tickets available for people to buy for the event, and the ticketing system is operational
Success Guarantee:	If they have bought a ticket, then the counter of how many available tickets must go down
Main Success Guarantee:	<ol style="list-style-type: none">1. Attendee selects an event to view2. System displays the number of remaining tickets3. If ticket are available, the system enables the option to purchase
Extensions:	A failure would involve when people are buying the ticket, the availability counter is not updating No tickets are available
Special Requirements:	Adding a timer for sales, showing how many tickets are left without clicking on the event, and pictures for the background of the event so that it has more appeal to the users.
Technology and Data Variations List:	Use PayPal to buy the tickets, Using Apple Pay instead of manually typing the numbers on the user’s card
Frequency of Occurrence:	Could be nearly continuous
Miscellaneous:	Open Issues (None at the moment, will update per project)

Fully Dressed Use Case #2

Use Case Name:	Purchase ticket
Scope:	Ticketing System
Level:	“User - Goal” or subfunction
Stakeholders and Interest:	Attendee: Wants to secure a ticket for an event at CofC. Payment Processor: Processes payment transactions securely. System Administrator: Ensures that the ticket purchasing process is smooth and handles any issues with ticket inventory or event information.
Primary Actor:	Any person trying to purchase tickets for a CofC event
Preconditions:	Attendee is registered and has selected a ticket to purchase.
Postconditions:	Attendee receives a confirmation of the purchase.
Success Guarantee:	If they have bought the ticket then the screen will display a receipt
Main Success Guarantee:	Attendee chooses to purchase a ticket and proceeds to payment. Attendee enters payment details. System validates payment and confirms the transaction. Attendee receives a ticket receipt.
Extensions:	1. Payment validation fails. 2. System notifies the attendee of the failure and suggests corrective action. 3. Attendees are prompted to re-enter payment details or choose a different payment method.
Special Requirements:	Adding a timer for sales, having the user select the number of tickets they are purchasing, and having the user enter their email and phone number while purchasing
Technology and Data Variations List:	Use PayPal to buy the tickets, Using Apple Pay instead of manually typing the numbers on the user’s card, and using CofC ID to get access to free tickets

Frequency of Occurrence:	Could be nearly continuous (As often as tickets are purchased.)
Miscellaneous:	Open Issues (None at the moment, will update per project)



Discussion:
Discussion:

This part of our app explains two key features: how to check if tickets are there for an event and how to buy them. We've made sure to describe all the steps and choices you might need to make along the way.

Motivation:

We want to make sure that our app always shows the right number of tickets available, keeps your payment information safe, and lets us easily update event information. Our goal is to make sure that both the people putting on events and the ones going to them have a great time without any hassles.

