Use Cases

Use Case Section	Comment
Use Case Name	Search for events
Scope	Secure Ticketing System
Level	User goal
Primary Actor	User
Stakeholders and Interests	User: Wants to search for events based on user specified criteria
Preconditions	User has logged in, and has entered valid parameters in the search
Success Guarantee	User must have entered valid parameters
Main Success Scenario	User enters parameters and matching events are displayed

Use Case Section	Comment
Use Case Name	Purchase ticket
Scope	Secure Ticketing System
Level	User goal
Primary Actor	User
Stakeholders and Interests	User: Wants to purchase ticket(s) College of Charleston: Wants to sell tickets Bank: Wants to verify the purchase and broker

	monetary exchange
Preconditions	User has logged in, and the requested number of tickets is available.
Success Guarantee	The requested number of tickets is available (specific seat, if we get there). Inventory is updated. Payment is processed. User an account.
Main Success Scenario	User purchases a ticket and the ticket (specific seat, if we get there) is removed from inventory (or marked as unavailable).

Use Case Section	Comment
Use Case Name	Log in
Scope	Secure Ticketing System
Level	User goal
Primary Actor	User
Stakeholders and Interests	User - Access application College of Charleston - Ensure the correct user is accessing the correct information securely
Preconditions	User has an account
Success Guarantee	User has an account and enters information correctly
Main Success Scenario	User logs in and has access to their account

Use Case Section	Comment
Use Case Name	Select seat
Scope	Secure Ticketing System

Level	Subfunction
Primary Actor	User
Stakeholders and Interests	User - Wants to choose a specific seat Event organizer - Wants to ensure attendees have assigned seats and each seat is sold only once
Preconditions	An event is selected and a purchase is in progress
Success Guarantee	The seat in question is still available at the time of finishing seat selection
Main Success Scenario	A seat is attached to the ticket and the user is sent to the payment screen

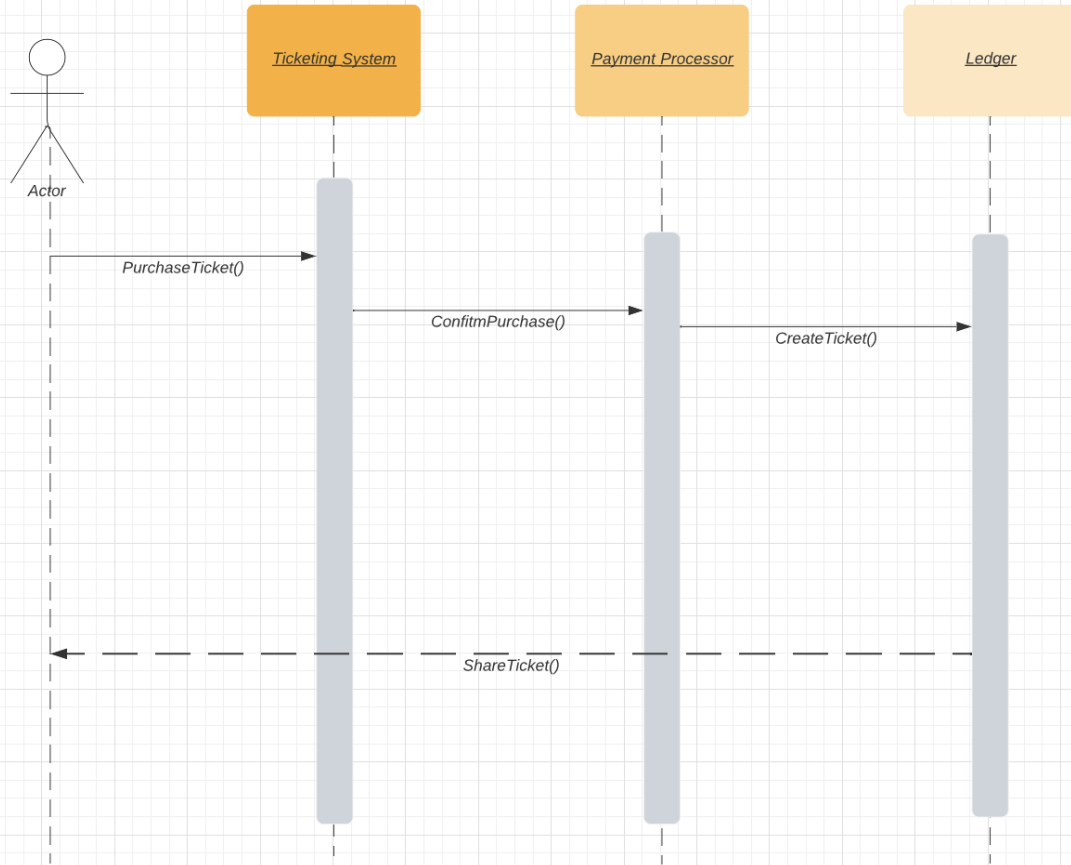| Use Case Section	Comment
Use Case Name	Create account
Scope	Secure Ticketing System
Level	User goal
Primary Actor	User
Stakeholders and Interests	User - Wants to create an account to store information System - store personal info securely and organize users with unique identifiers
Preconditions	User does not have an account User has a CofC ID
Success Guarantee	The information entered is uniquely identifiable (CofC ID).
Main Success Scenario	Account is created and saved

Use Case Section	Comment
Use Case Name	Create ticket
Scope	Secure Ticketing System
Level	Subfunction
Primary Actor	System
Stakeholders and Interests	User - Wants a unique ticket to the event with their specific information Event Organizer - Wants something unique that verifies valid tickets
Preconditions	A payment has been verified and completed The payment is for a specific ticket
Success Guarantee	User has purchased a ticket, payment is successful (processed)
Main Success Scenario	A ticket with information of the ticket holder and event is generated and attached to the payment receipt

Use Case Section	Comment
Use Case Name	Send receipt
Scope	Secure Ticketing System
Level	Subfunction
Primary Actor	System
Stakeholders and Interests	User - want to receive confirmation of the their payment and their ticket System- wants to record the payment on their ledger and assure the user of payment completion

Preconditions	The payment was successful A ticket was generated
Success Guarantee	There is a valid email attached to the account
Main Success Scenario	A receipt with the ticket and payment information is sent to the user and is also recorded on the system ledger.

Create Ticket

Anthony Ruvo | September 20, 2023

Ticketing System

Payment Processor

Ledger

Actor

PurchaseTicket()

ConfitmPurchase()

CreateTicket()

ShareTicket()

Create Account

Anthony Ruvo | September 20, 2023

Actor

Ticketing System

Account Database

CreateAccount()

SendAccountInformation()

StoreAccountInformation()

SendSuccessMessage()
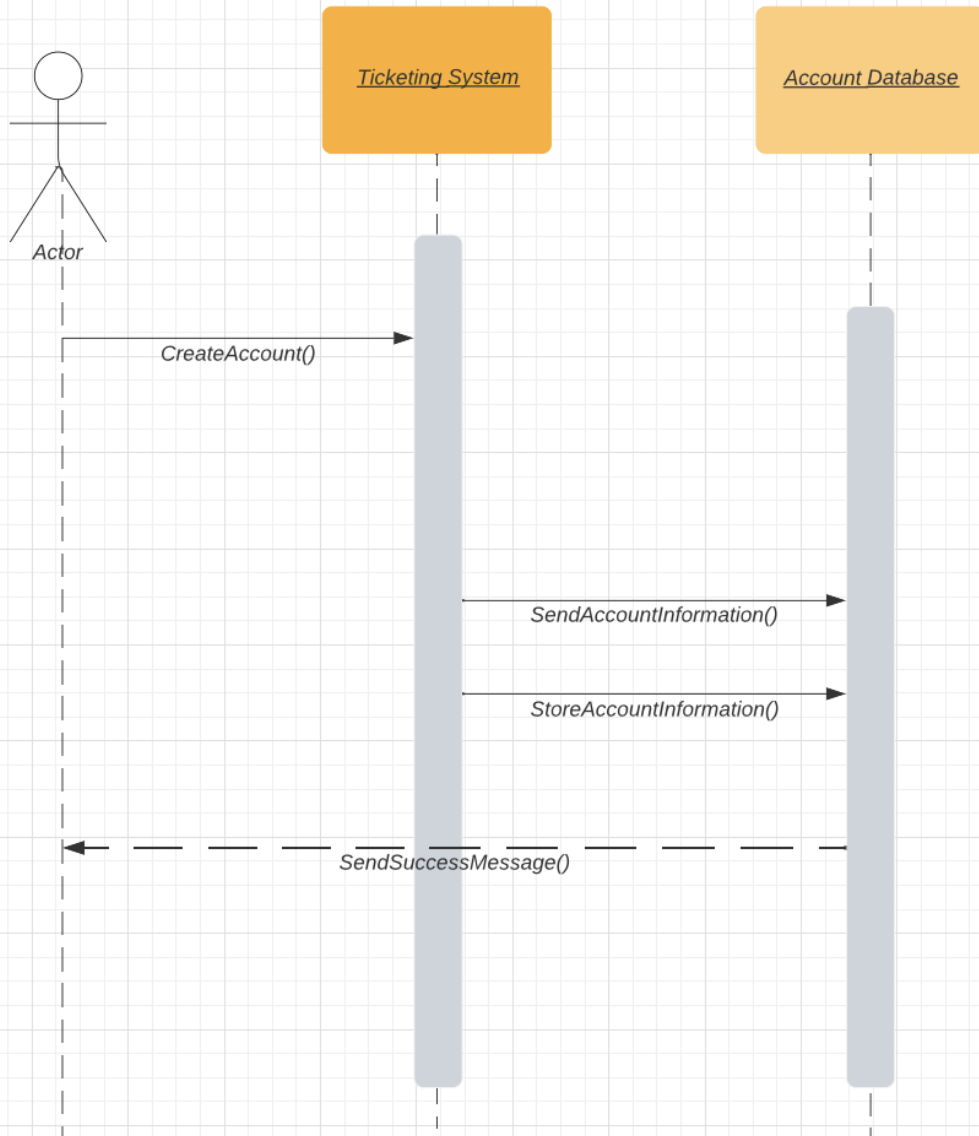
Select Seat

Anthony Ruvo | September 20, 2023

Ticketing System

Seat Database

Actor

ChooseSeat()

ConfirmAvailability()

RemoveFromAvailaible()

SendSuccessMessage()
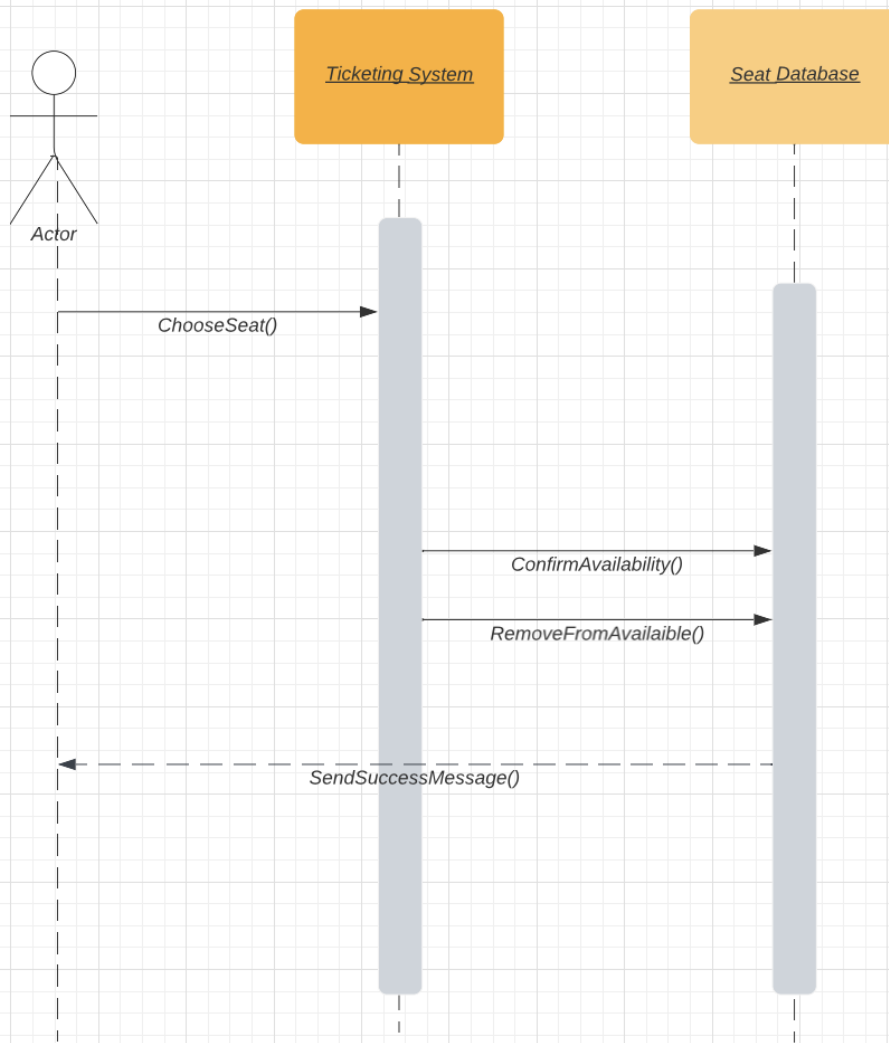
Purchase Ticket
Scenario

```
      O
     /|\          :System        Event Database          :Bank
     / \
    :user
      |    getPrice(event, seats[])    |              |              |
      |------------------------------->|              |              |
      |              |  pullTicketPrice(event, seat)  |              |
      |              |------------------------------->|              |
      |              |              |                 |              |
      |    priceWithTax    |     totalPrice           |              |
      |<-------------------|<------------------------- |             |
      | finishPayment(paymentInfo,                     |             |
      |    priceWithTax)                               |             |
      |------------------------------------------------------------->|
      |              |                                               |
      |              |        isComplete()                           |
      |              |---------------------------------------------->|
      |              |                                               |
      |         verificationMessage                                  |
      |<------------------------------------------------------------ |
```

Seat Selection Scenaio

```
:User          :System          Event Database
  |               |                    |
  | pullSeats(currentEvent)            |
  |-------------->|                    |
  |               | findEvent(currentEvent)
  |               |------------------->|
  |               |                    |
  |               |     eventSeats[]   |
  |               |<- - - - - - - - - -|
  | displaySeats(eventSeats[])         |
  |<- - - - - - - -|                   |
  |               |                    |
  | seatSelection(seats[])             |
  |-------------->| updateSeats(seats[])
  |               |------------------->|
  |               |                    |
```
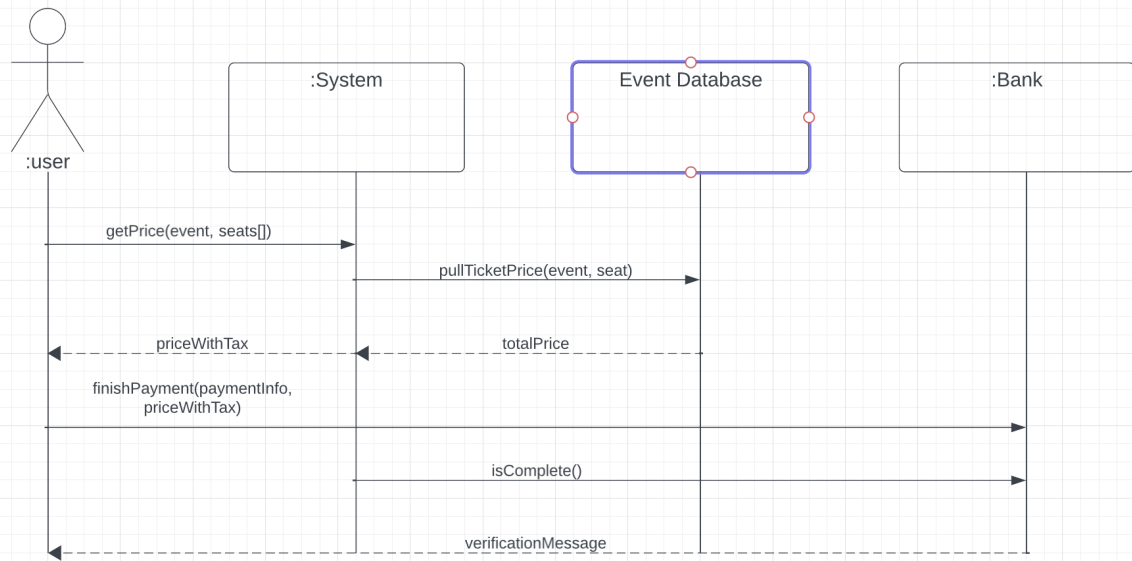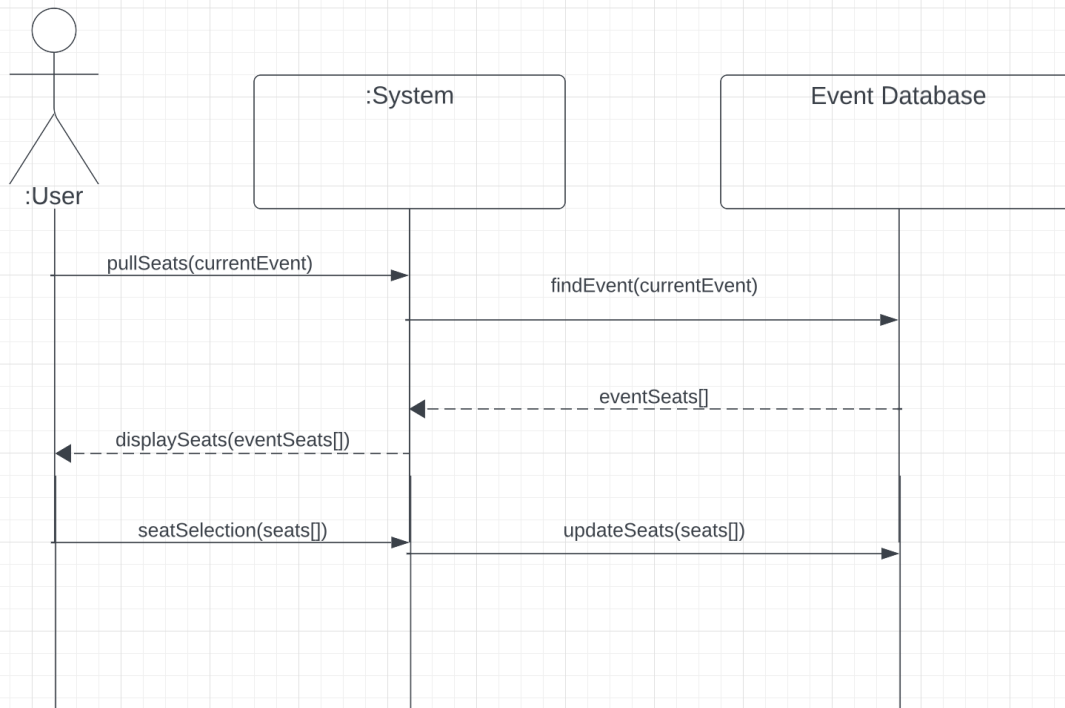
Search for Event
Scenario

:User

:System

EventDatabase

userSearch(searchTerm,
searchFilter)

Loop

withinSearchBounds(
searchTerm, searchFilter)

filteredEvent

displayEventList()

Login Scenario

:User

:System

Account database

logIn(cofcID, password)

verifyAccount(cofcID, password)

loginSuccess
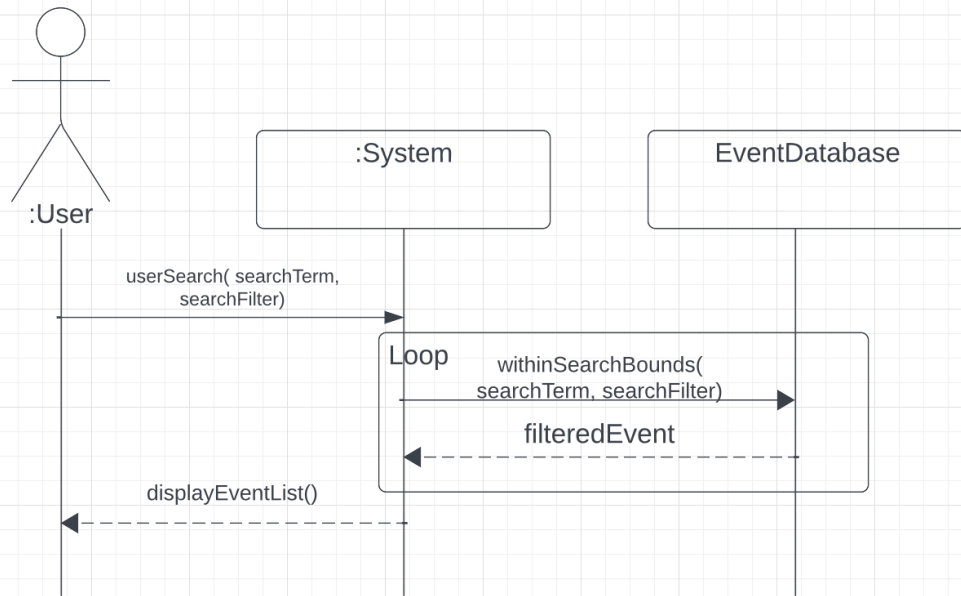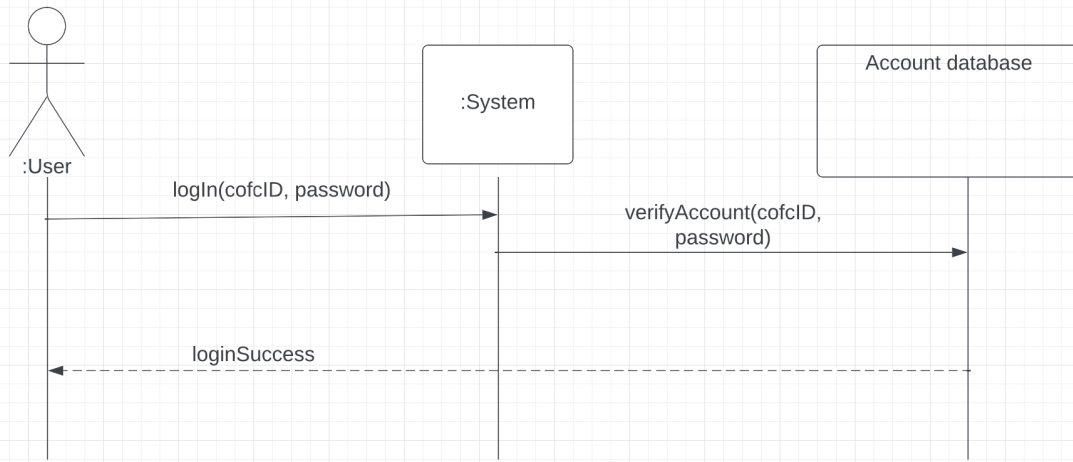
Send Receipt

Anthony Ruvo | September 20, 2023

Ticketing System

Payment Processor

Ledger

Actor

PurchaseTicket()

ProcessPayment()

CreateTicket()

CreateReceipt()

SendReceipt()