The Local Spot

Project Team: Triforce
Project Members: Eric Nguyen, Luong Luu, Joshua Pellegrino
GitHub Link: https://github.com/Jpelle25/TheLocalSpot490

Table of Contents

1. Project Definition

Why

This is needed to help customers get tickets ahead of time without worrying about being late. It also helps local event coordinators to sell tickets to their potential buyers and customers for small-scale events. In terms of tickets, it helps any facilitator to keep track of event popularity, and ticket management, and prevent fraudulent ticket access.

What

The goal of this project is to provide coordinators and customers with an easy-to-use ticket management service. The service ensures the security and authentication of any registered events.

How

How this will be achieved is with a web application that allows coordinators to register for any event. For customers, it allows a sign-up platform for any interested event. A backend will store coordinator/customer information as well as unique ticket data.

2. Project Requirements

Functionalities

Login screen and account creation, User preferences and tailoring, Event registration and management system

Usability

Web application that customers will use, performance is dependent on the client's connectivity to the web application's server.

System

The hardware varies on the client's device.

A software idea we had in mind would be to use the Vaadin framework for web app development.

The database will carry registered users, events, ticket information, event popularity, and any potential future information needed to be stored.

Security

2-Factor Authentication for logins, encryption for sensitive information in the database

3. Project Specification

Focus / Domain / Area

Social Events, Buying and Selling tickets, Event management

Libraries / Frameworks / Development Environment

Vaadin (Java, CSS), Google OAuth, In-memory database

Platform

Desktop/Mobile with browsing web connectivity

Genre

Web application

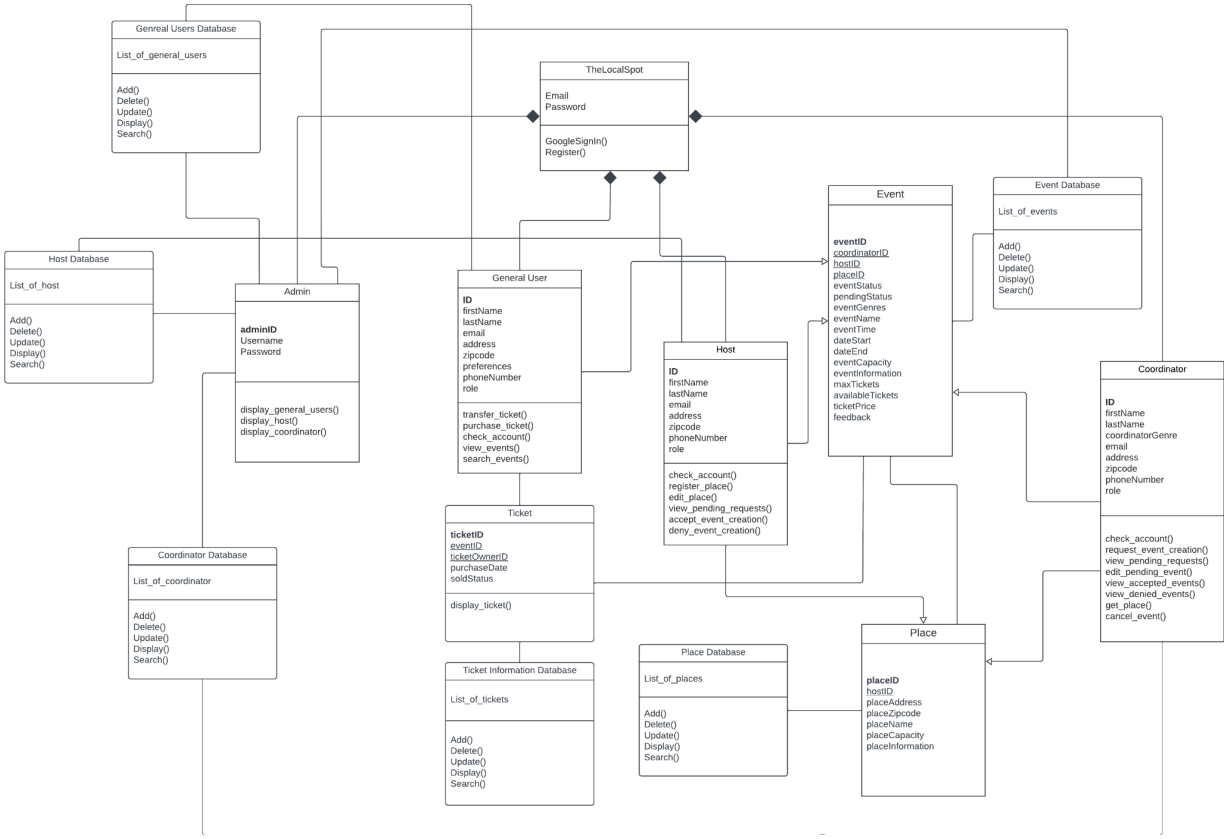
Types

Web application design and development, Database management, Programming

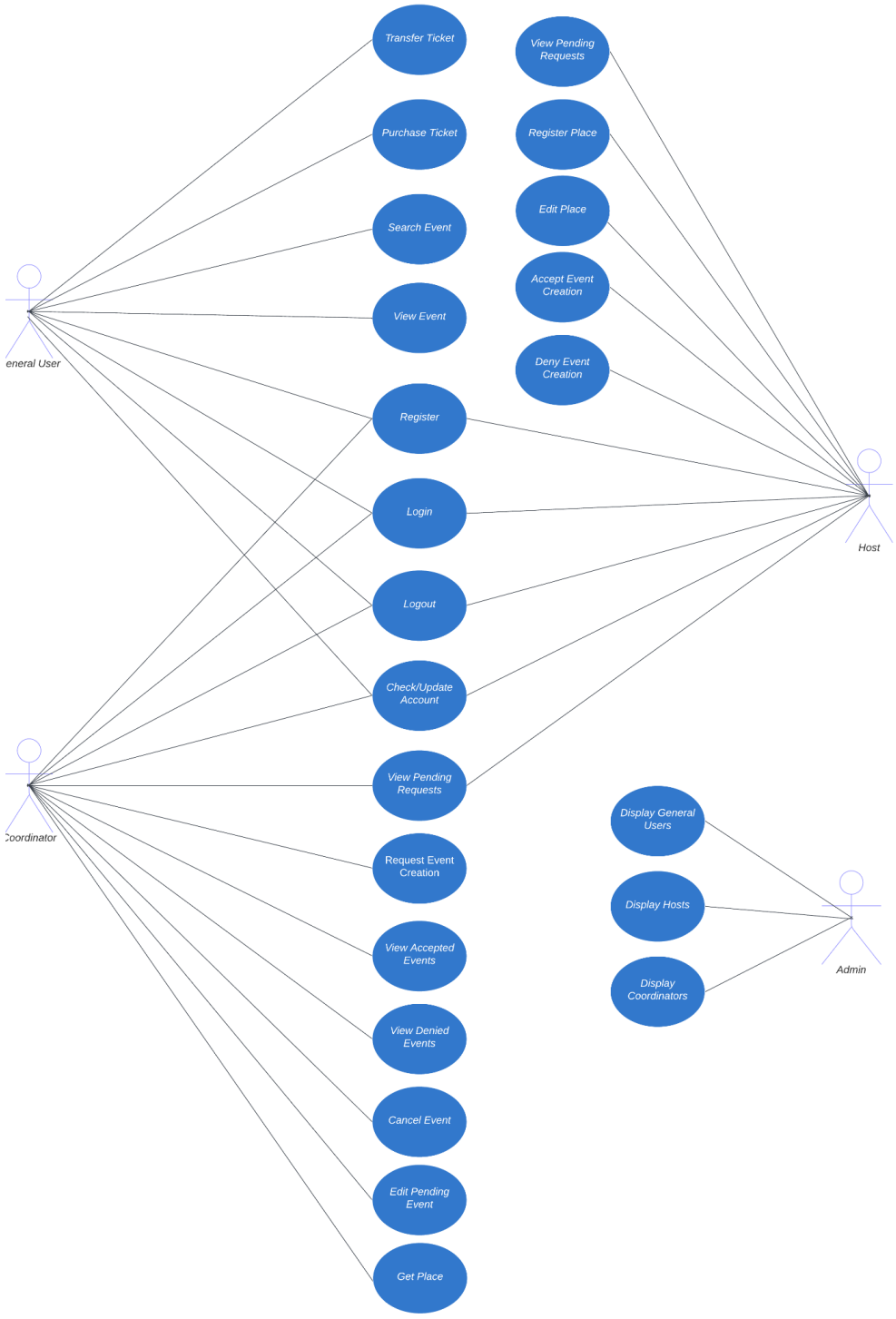
4. System – Design Perspective

- Identify subsystems – design point of view

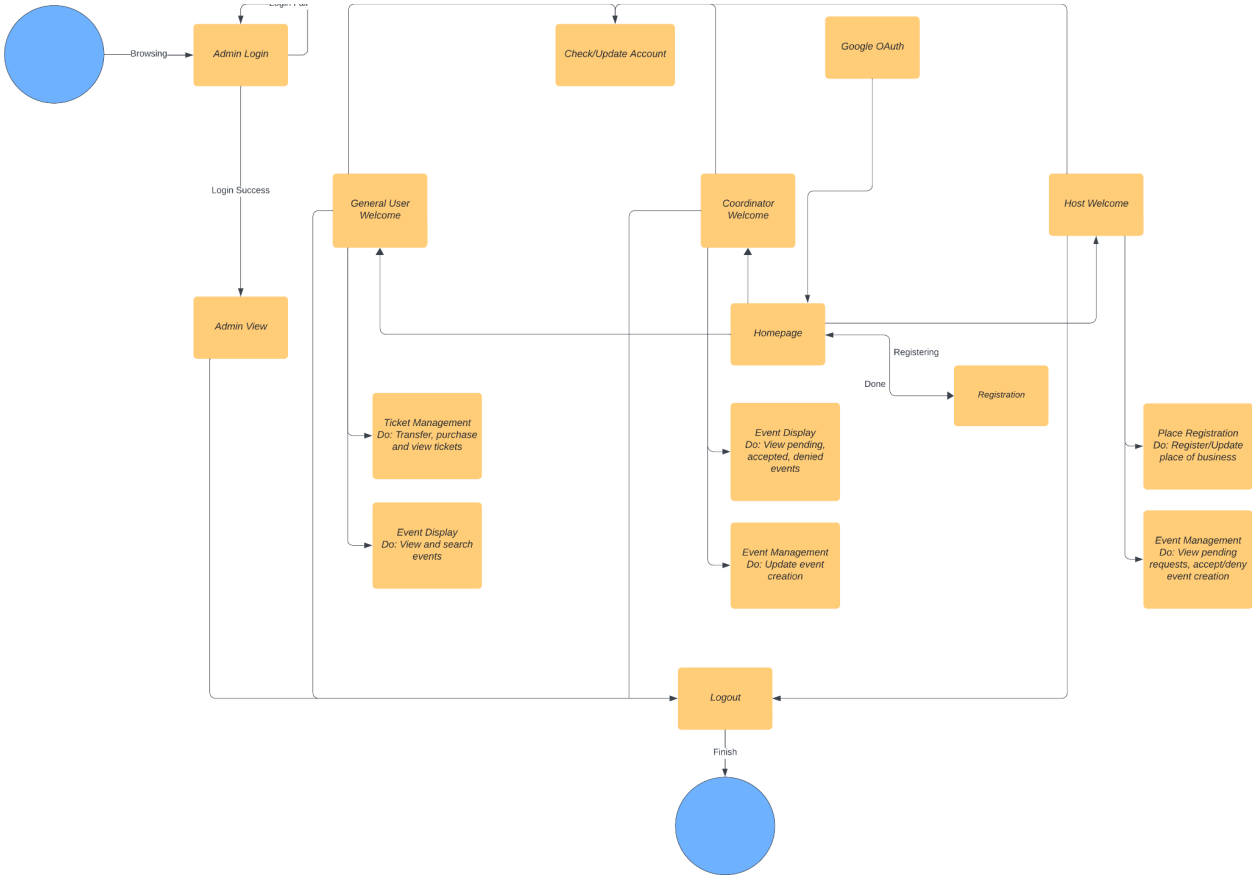
UML Diagram


Genreal Users Database
List_of_general_users
Add()
Delete()
Update()
Display()
Search()
TheLocalSpot
Email
Password
GoogleSignIn()
Register()
Event
eventID
coordinatorID
hostID
placeID
eventStatus
pendingStatus
eventGenres
eventName
eventTime
dateStart
dateEnd
eventCapacity
eventInformation
maxTickets
availableTickets
ticketPrice
feedback
Event Database
List_of_events
Add()
Delete()
Update()
Display()
Search()
Host Database
List_of_host
Add()
Delete()
Update()
Display()
Search()
Admin
adminID
Username
Password
display_general_users()
display_host()
display_coordinator()
General User
ID
firstName
lastName
email
address
zipcode
preferences
phoneNumber
role
transfer_ticket()
purchase_ticket()
check_account()
view_events()
search_events()
Host
ID
firstName
lastName
email
address
zipcode
phoneNumber
role
check_account()
register_place()
edit_place()
view_pending_requests()
accept_event_creation()
deny_event_creation()
Coordinator
ID
firstName
lastName
coordinatorGenre
email
address
zipcode
phoneNumber
role
check_account()
request_event_creation()
view_pending_requests()
edit_pending_event()
view_accepted_events()
view_denied_events()
get_place()
cancel_event()
Ticket
ticketID
eventID
ticketOwnerID
purchaseDate
soldStatus
display_ticket()
Coordinator Database
List_of_coordinator
Add()
Delete()
Update()
Display()
Search()
Ticket Information Database
List_of_tickets
Add()
Delete()
Update()
Display()
Search()
Place Database
List_of_places
Add()
Delete()
Update()
Display()
Search()
Place
placeID
hostID
placeAddress
placeZipcode
placeName
placeCapacity
placeInformation

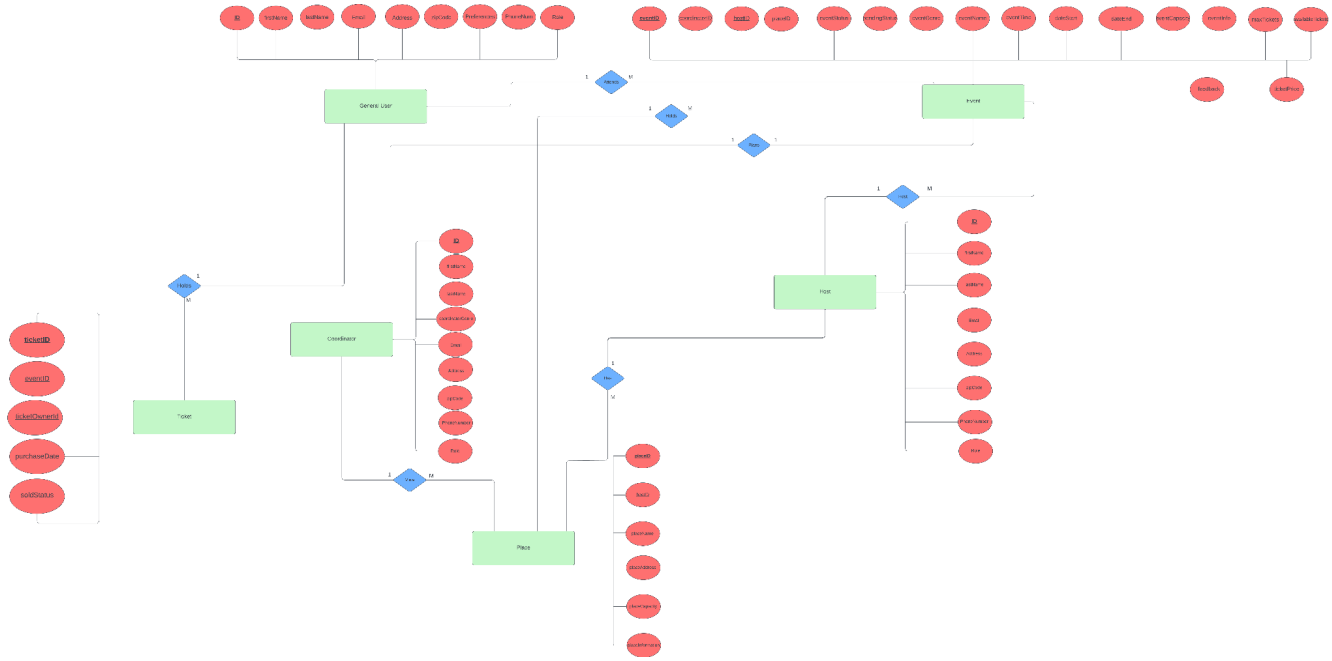
Use Case Diagram



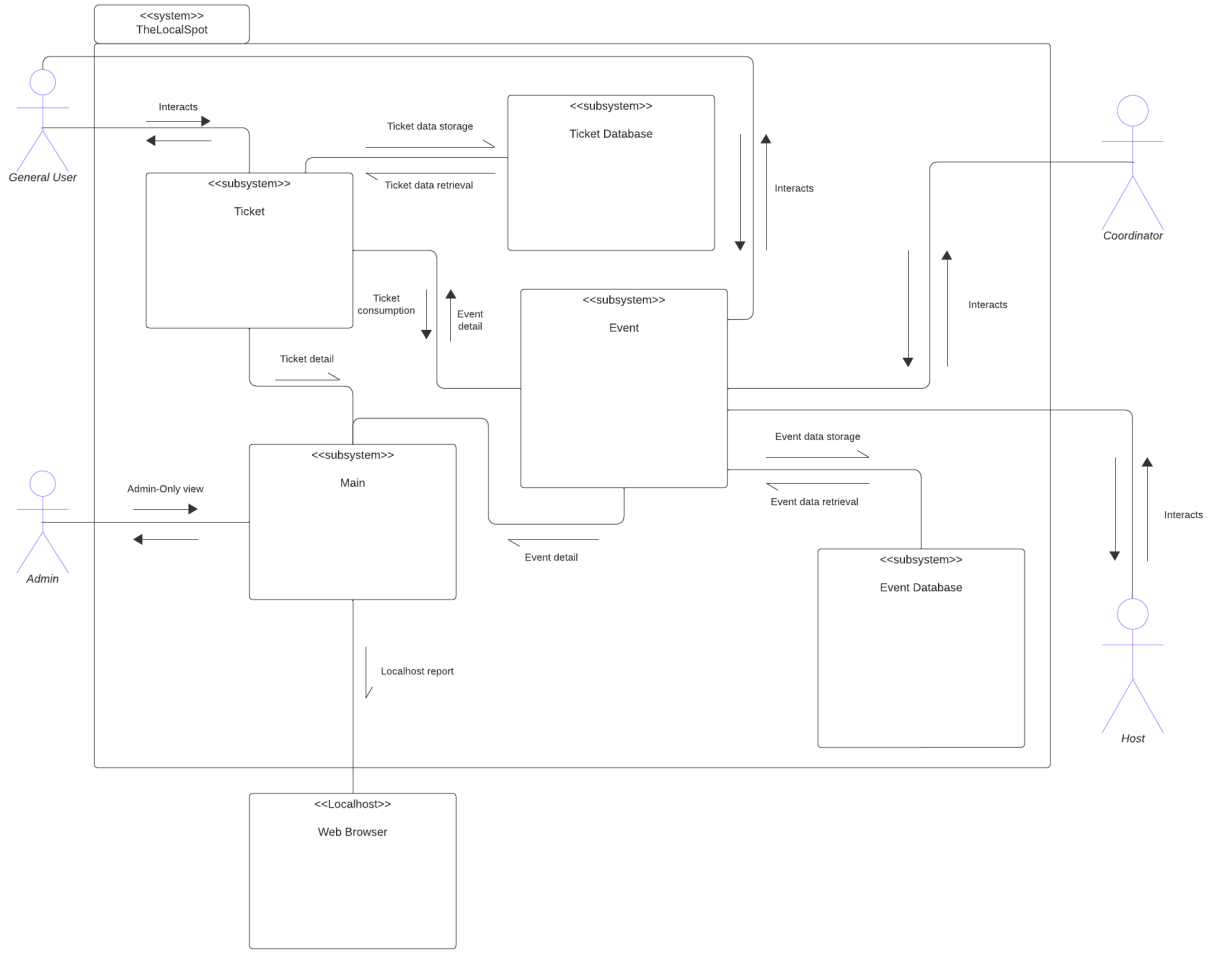
Sub-System Communication



ER Model



System Model



5. System – Analysis Perspective

Identify subsystems – analysis point of view

- For our project, there are four main points of view (clients).
- General User – Default client. Some of the functionalities here include transferring and purchasing tickets. Viewing and searching for events for potential ticket purchasing. The purpose of this client is to provide the standard functionalities for someone who is neither a host nor a coordinator.
- Host – Establishes and registers places of business to host events. Some of the functionalities here include registering places, maintaining events, and accepting or denying event creations. The purpose of the host is to work in tandem with coordinators to allow for event participation by the general users.
- Coordinator – Establishes their professional work and requests to present their background through events. Some of the

functionalities here include requesting events, getting places to host their events, and maintaining and updating event details and parameters. The purpose of the coordinator is to allow people who have a profession easier access to communicate with hosts through a desired event.

- Admin - Controls and monitors the web application. Some of the functionalities here include viewing the other client's point of view and adjusting any parameters. The purpose here is to provide easy access to other accounts and provide any support.

System (Tables and Description)

Admin Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	
adminID	Integer	10	ID for Admin account	Yes	No	PK
Username	String	20	Sign-on name	Yes	No	
Password	String	30	Sign-on password	Yes	No	

General User Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	
ID	Integer	10	ID for Gen. User account	Yes	No	PK
Name	String	50	Name of user	Yes	No	
Email	String	50	Email of user	Yes	No	
Address	String	50	Address of user	Yes	Yes	
Preferences	String	20	Tailoring preferences to user	No	Yes	
Phone Number	Char	10	Phone number of user	Yes	No	
Zipcode	Integer	5	Zipcode of user	Yes	No	
Role	String	20	Role (general user)	Yes	No	

Coordinator Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	

ID	Integer	10	ID for Coordinator account	Yes	No	PK
Name	String	50	Name of coordinator	Yes	No	
Email	String	50	Email of coordinator	Yes	No	
Address	String	50	Address of coordinator	Yes	No	
Coor. Genre	String	20	Genre of event that coordinator sets for general user preferences	No	Yes	
Phone Number	Char	10	Phone Number of coordinator	Yes	No	
Zipcode	Integer	5	Zipcode of coordinator	Yes	No	
Role	String	20	Role (coordinator)	Yes	No	

Host Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	
ID	Integer	10	ID for Host account	Yes	No	PK
Name	String	50	Name of host	Yes	No	
Email	String	50	Email of host	Yes	No	
Address	String	50	Address of host	Yes	No	
Phone Number	Char	10	Phone Number of host	Yes	No	
Zipcode	Integer	5	Zipcode of host	Yes	No	
Role	String	20	Role (host)	Yes	No	

Event Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	
EventID	Integer	10	ID for event	Yes	No	PK
CoordinatorID	Integer	10	Respective ID of the coordinator that is participating	Yes	No	FK

HostID	Integer	10	Respective ID of the host that owns place	Yes	No	FK
PlaceAddress	String	50	Physical address of the place	Yes	No	FK
EventStatus	Boolean	10	Active status of event	Yes	No	
EventName	String	50	Name of event	Yes	No	
EventTime	Time	10	Event start time	Yes	No	
DateStart	Date	10	Event date start	Yes	No	
DateEnd	Date	10	Event date end	Yes	No	
EventCapacity	Integer	10	Capacity of the event	Yes	No	
Information	String	50	Small description of the event	Yes	No	
MaxTickets	Integer	10	Max amount of tickets created for the event	Yes	No	
AvailableTickets	Integer	10	Currently available tickets for the event	Yes	No	
TicketPrice	Decimal	10,2	Price of tickets for the event	Yes	No	
PendingStatus	Boolean	10	Pending status of event	Yes	No	
Feedback	String	100	Feedback on accepted/denied event	No	Yes	

Place Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	
PlaceID	Integer	10	ID for place	Yes	No	PK
HostID	Integer	10	Respective id of the host that owns place	Yes	No	FK
PlaceName	String	50	Name of the place	Yes	No	
PlaceAddress	String	50	Physical address of the place	Yes	No	
PlaceCapacity	String	20	Capacity of the place	Yes	No	
Information	String	50	Small description of the place set by the host	Yes	No	

Ticket Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	
TicketID	Integer	10	ID for Ticket	Yes	No	PK
EventID	Integer	10	Respective ID of the event that the ticket is attached to	Yes	No	FK
TicketOwnerID	Integer	10	Respective ID of the user that owns the ticket(s)	Yes	No	FK
PurchaseDate	Date	10	Purchase date for ticket(s)	Yes	No	
SoldStatus	Boolean	10	Whether or not the ticket(s) is/are sold	Yes	No	

Algorithm Analysis

- Account Creation – O(1) : This will be the time analysis for this subsystem because account creation is going to be done through the usage of Google's sign-on API. For other fields in account details, the entered values of a client are a direct mapping to the assignment of those particular fields.
- Event Searching – O(n) : This will be the time analysis for this subsystem because the general user needs to query the entire event database and potentially apply a filter to limit what is wanting to be searched. This has the potential of being O(1) in the case that a filter maps to only one particular event.
- Ticket Purchasing. - O(1) : This will be the time analysis for this subsystem because a ticket can only be purchased for one particular event. Given the parameter of said event, the ticket amount allotted will decrease by one and be assigned to the given user.
- Time Analysis Pending - Further time analysis is to be evaluated for host and coordinator subsystems when the code and logic are created.

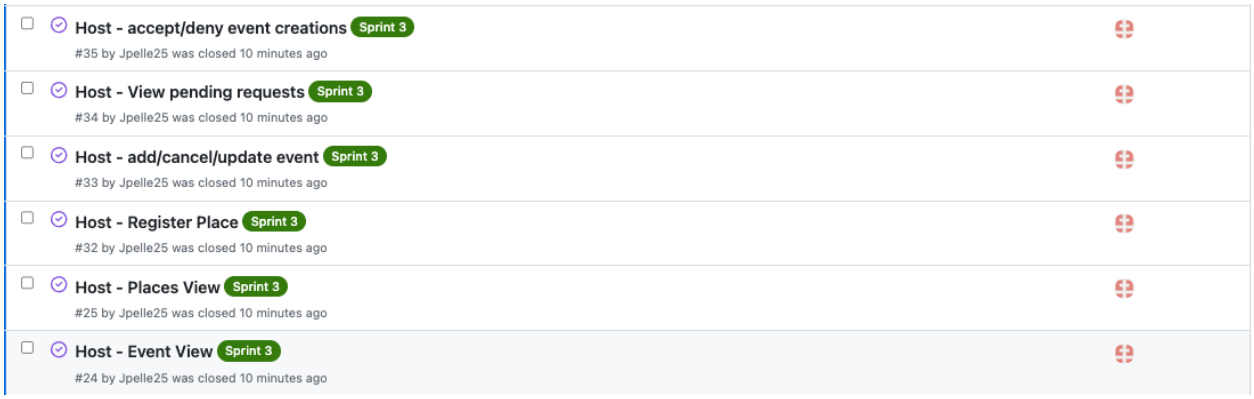
6. Project Scrum Report

Product Backlog - Link to all previous tasks and sprints.

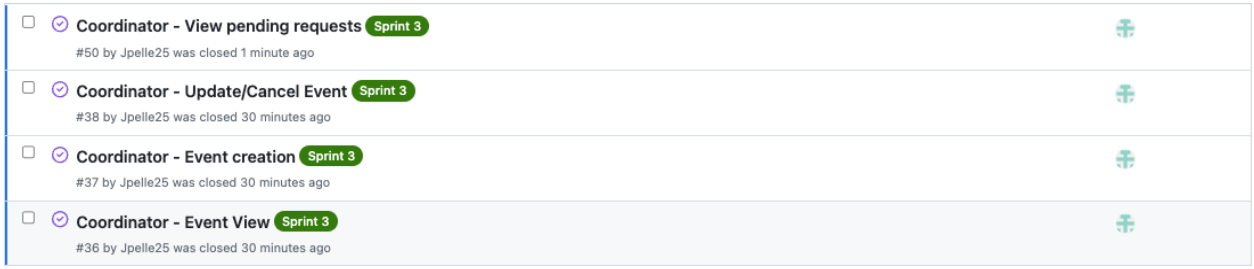
- https://github.com/Jpelle25/TheLocalSpot490/issues?q=is%3Aissue+is%3Aclosed

Sprint Backlog (Table / Diagram)

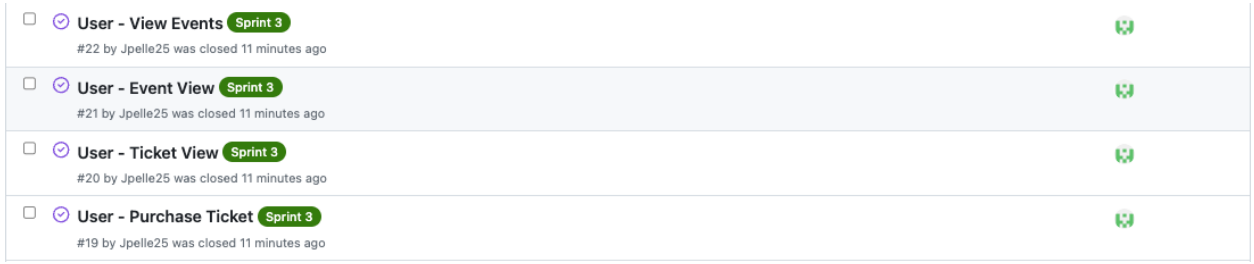
- Host



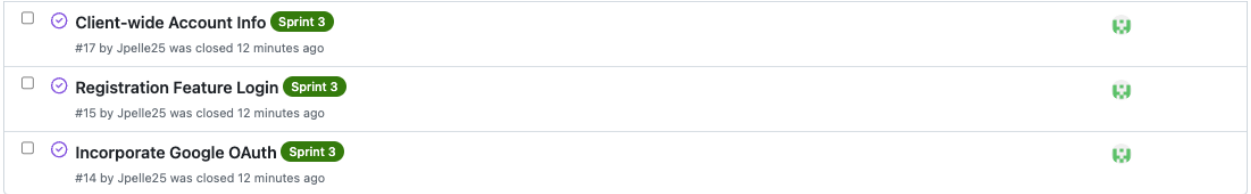
- Coordinator



- General User



- Shared Functionalities

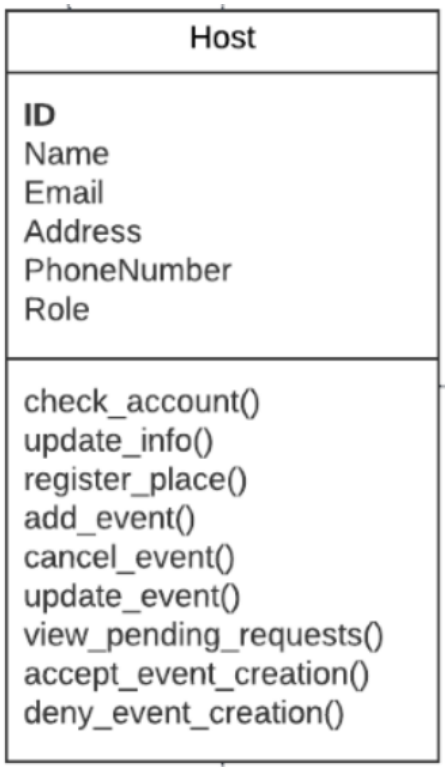


7. Subsystems

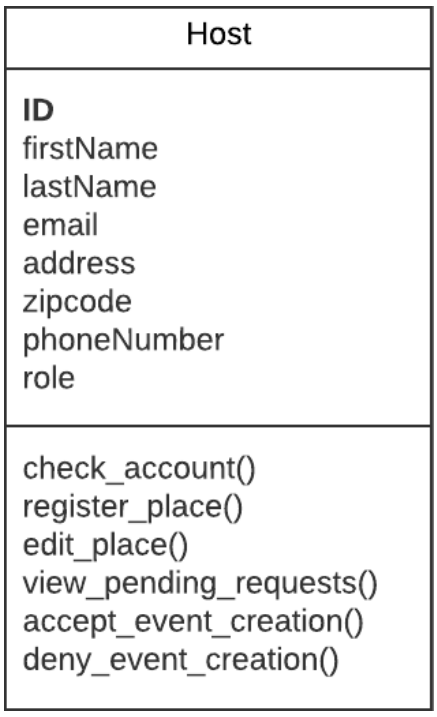
7.1 Subsystem 1 – Eric Nguyen - *Host*

- UML diagram for Host

Old:



New:



- Data dictionary

Host Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	
ID	Integer	10	ID for Host account	Yes	No	PK
Name	String	50	Name of host	Yes	No	
Email	String	50	Email of host	Yes	No	
Address	String	50	Address of host	Yes	No	
Zipcode	Integer	5	Zipcode of host	Yes	No	
Phone Number	Char	10	Phone Number of host	Yes	No	
Role	String	20	Role (host)	Yes	No	

Place Table

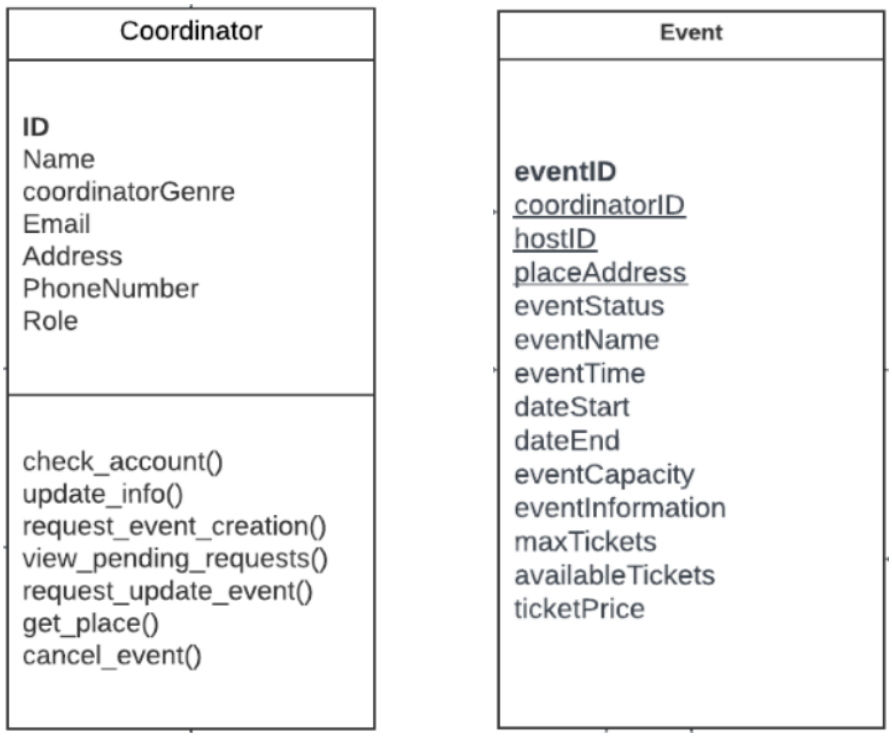
Field Name	Datatype	Field Length	Description	Required	Accept Null	
PlaceID	Integer	10	ID for place	Yes	No	PK
HostID	Integer	10	Respective id of the host that owns place	Yes	No	FK
PlaceName	String	50	Name of the place	Yes	No	
PlaceAddress	String	50	Physical address of the place	Yes	No	
PlaceCapacity	String	20	Capacity of the place	Yes	No	
Information	String	50	Small description of the place set by the host	Yes	No	

- If refined (changed over the course of project)
    - Only changes are the addition of Zip Code and Role for Host.
- Scrum Backlog
    - https://github.com/Jpelle25/TheLocalSpot490/issues?q=is%3Aissue+is%3Aclosed+assignee%3AJpelle25+label%3A%22Sprint+3%22
- Coding
    - Approach: OOP
    - Language: Java
- User training (ReadME)
    - https://github.com/Jpelle25/TheLocalSpot490

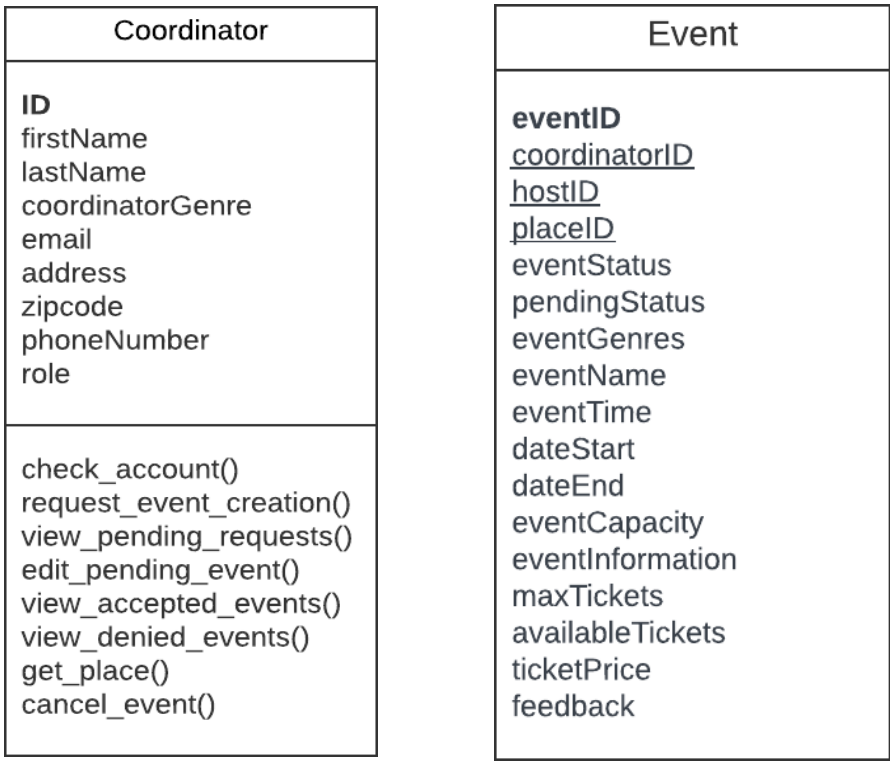
7.2 Subsystem 2 – Luong Luu - *Coordinator*

- UML Diagrams for Coordinator and Event

Old:



New:



- Data dictionary

Coordinator Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	
ID	Integer	10	ID for Coordinator account	Yes	No	PK
Name	String	50	Name of coordinator	Yes	No	
Email	String	50	Email of coordinator	Yes	No	
Address	String	50	Address of coordinator	Yes	No	
Coor. Genre	String	20	Genre of event that coordinator sets for general user preferences	No	Yes	
Phone Number	Char	10	Phone Number of coordinator	Yes	No	
Zipcode	Integer	5	Zipcode of coordinator	Yes	No	
Role	String	20	Role (coordinator)	Yes	No	

Event Table

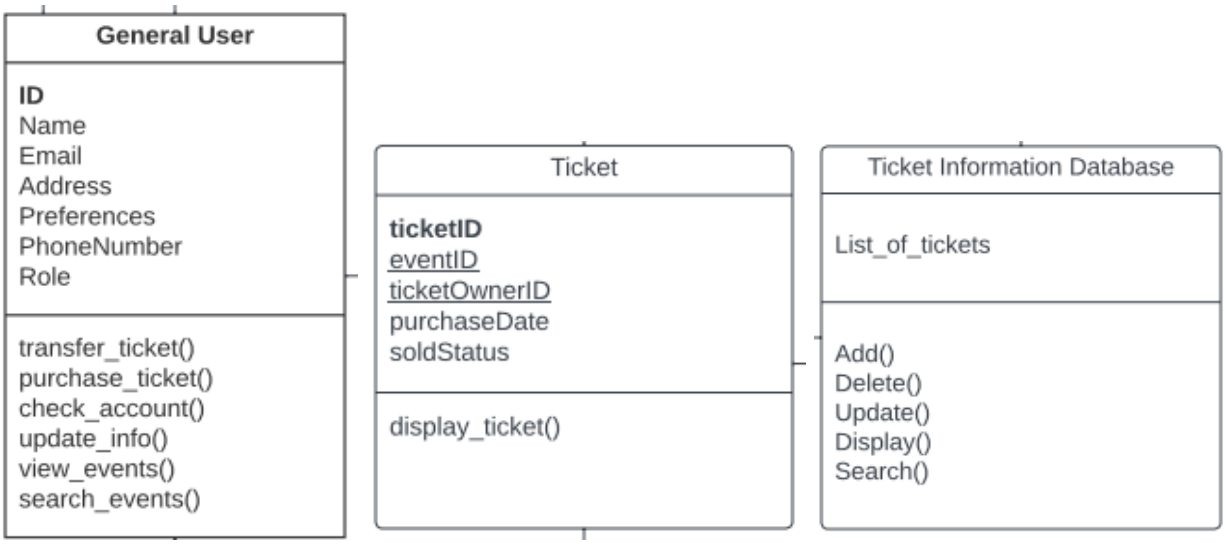
Field Name	Datatype	Field Length	Description	Required	Accept Null	
EventID	Integer	10	ID for event	Yes	No	PK
CoordinatorID	Integer	10	Respective ID of the coordinator that is participating	Yes	No	FK
HostID	Integer	10	Respective ID of the host that owns place	Yes	No	FK
PlaceAddress	String	50	Physical address of the place	Yes	No	FK
EventStatus	Boolean	10	Active status of event	Yes	No	
EventName	String	50	Name of event	Yes	No	
EventTime	Time	10	Event start time	Yes	No	
DateStart	Date	10	Event date start	Yes	No	
DateEnd	Date	10	Event date end	Yes	No	
EventCapacity	Integer	10	Capacity of the event	Yes	No	

Information	String	50	Small description of the event	Yes	No	
MaxTickets	Integer	10	Max amount of tickets created for the event	Yes	No	
AvailableTickets	Integer	10	Currently available tickets for the event	Yes	No	
TicketPrice	Decimal	10,2	Price of tickets for the event	Yes	No	
PendingStatus	Boolean	10	Pending status of event	Yes	No	
Feedback	String	100	Feedback on accepted/denied event	No	Yes	

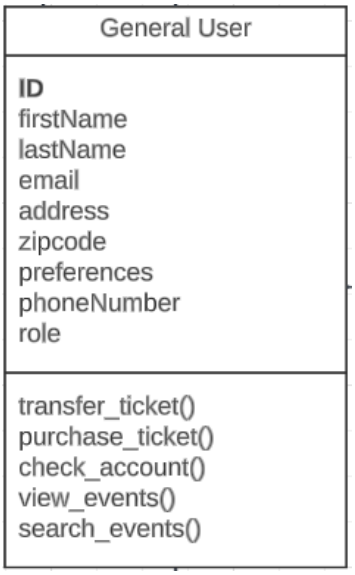
- If refined (changed over the course of project)
    - Add zipcode and role to the coordinator
    - Add pendingStatus and Feedback to the event
- Scrum Backlog
    - https://github.com/Jpelle25/TheLocalSpot490/issues?q=is%3Aissue+is%3Aclosed+assignee%3AJpelle25+label%3A%22Sprint+3%22
- Coding
    - Approach: OOP
    - Language: Java
- User training (ReadME)
    - https://github.com/Jpelle25/TheLocalSpot490

7.3 Subsystem 3 – Joshua Pellegrino - *General User*

- Old Uml



- New Uml



- Data dictionary

General User Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	
ID	Integer	10	ID for Gen. User account	Yes	No	PK
Name	String	50	Name of user	Yes	No	
Email	String	50	Email of user	Yes	No	

Address	String	50	Address of user	Yes	Yes	
Preferences	String	20	Tailoring preferences to user	No	Yes	
Phone Number	Char	10	Phone number of user	Yes	No	
Role	String	20	Role (general user)	Yes	No	
Zip Code	Integer	5	Zipcode of user	Yes	No	

Ticket Table

Field Name	Datatype	Field Length	Description	Required	Accept Null	
TicketID	Integer	10	ID for Ticket	Yes	No	PK
EventID	Integer	10	Respective ID of the event that the ticket is attached to	Yes	No	FK
TicketOwnerID	Integer	10	Respective ID of the user that owns the ticket(s)	Yes	No	FK
PurchaseDate	Date	10	Purchase date for ticket(s)	Yes	No	
SoldStatus	Boolean	10	Whether or not the ticket(s) is/are sold	Yes	No	

- Scrum Backlog
    - https://github.com/Jpelle25/TheLocalSpot490/issues?q=is%3Aissue+is%3Aclosed+assignee%3AJpelle25+label%3A%22Sprint+3%22
- Coding
    - Approach: OOP
    - Language: Java
- User training (ReadME)
    - https://github.com/Jpelle25/TheLocalSpot490

8. Complete System

- Final software/hardware product
    - Clone the Repo
    - Install Dependencies

    - Run Application
- Github Link: https://github.com/Jpelle25/TheLocalSpot490
- User Manual: Please refer to the ReadME in GitHub
- Evaluation by client and instructor
- Team Member Descriptions
    - Eric Nguyen: Host + Place
    - Luong Luu: Coordinator + Event
    - Joshua Pellegrino: General User + Ticket
    - Everyone: Login + UI