



Bilkent University

Department of Computer Engineering

Senior Design Project

3KET: 3D Event Venue Viewing Application

Project Analysis Report

Ege Kaan Gürkan, Musa Ege Ünalın, Mehmet Çalışkan, Doğa Tansel, Mustafa Göktan GÜDÜKBAY

Supervisor: Asst. Prof. Dr. Eray Tüzün

Jury Members: Asst. Prof. Dr. Shervin Arashloo and Asst. Prof. Dr. Hamdi Dibekliolu

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Contents

1. Introduction	1
2. Current System	1
3. Proposed System	4
3.1. Overview	4
3.2. Functional Requirements	4
3.3. Non-functional Requirements	6
3.4. System Models	8
3.4.1. Scenarios	8
3.4.2. Use Case Model	20
3.4.3. Object and Class Model	21
3.4.4. Dynamic Models	22
3.4.5. User Interface - Navigational Paths and Screen Mock-ups	27
3.4.5.1. Business Side Mockups	27
3.4.5.2. Customer Side Mockups	33
4. Other Analysis Elements	39
4.1. Consideration of Various Factors in Engineering Design	39
4.2. Risks and Alternatives	42
4.3. Project Plan	44
4.4. Ensuring Proper Teamwork	54
4.5. Ethics and Professional Responsibilities	54
4.6. Planning for New Knowledge and Learning Strategies	55
5. Glossary	55
6. References	56

1. Introduction

Nowadays, attending live entertainment events is widespread among people, and purchasing tickets for these events is usually done through online ticket-sale systems. Although ticket-sale systems monopolizing this market (Ticketmaster or Biletix) give detailed information regarding their events and venues, they do not offer a highly interactive approach. Therefore, customers cannot get a taste of the real experience they expect before they participate, and they cannot annotate or help improve any of these experiences.

3KET is a new cutting-edge online entertainment venue viewing system with an interactive 3D venue walkabout and event/venue rating feature. 3KET aims to help any participants of entertainment events find, review and help improve the events they participate in and the venues they visit. The 3D walkabout of an event venue helps customers be more decisive in their purchases and gives them an idea of what to expect during an event in that venue. The rating/reviewing system of 3KET is fun and interactive, with scoreboards and achievements for attending events and comments that will directly go to the venue owners or event organizers as feedback. Participants can share memories related to events or venues, which makes the reviewing and participating process all more personal. Another essential characteristic of 3KET is that businesses will create and delete their events while also customizing them. They will be able to update the 3D walkabout, set up scoreboards or memory sharing contests. All in all, 3KET aims to offer customers and businesses to build a better event-going experience interactively.

2. Current System

The most common way to purchase tickets to an event is through online ticket-sale systems; the most popular ones currently being used are Biletix, Ticketmaster, Peatix, Plandy, and BSO (in our case). In Biletix and Ticketmaster,

customers can view any event in the system, obtain ticket information, see the events that have the most interaction, perform payment, and see a 2D seating view for some events/venues [1]. Additionally, business owners can create events in Ticketmaster, and customers can make event recommendations [2].

Peatix is an online event creation and viewing platform, also offering ticket purchasing [3]. Plandy is an application for viewing and planning events; however, other features such as ticket information are not available on the website. Plandy redirects the user to other websites for further information [4]. There are also minor, local online ticket-purchasing systems such as Biletino, Passo, or organizer/promoter websites such as Milyon Bilet, which do not have any additional features compared to the previously mentioned competitors [5], [6], [7].

Bilkent Symphony Orchestra (BSO) Ticket Sales is a web application where customers can view the concerts given in Bilkent University by either the university orchestra or guest artists. Customers can view the 2D seating plan and buy their tickets from this application [8]. We compared all the essential features of competitor applications and our application in Table 1.

	Features										
Competitors	Creating Events (for Business)	Viewing Events	Displaying Events from Various Websites	3D View of Venue (Perspective of Seats)	Ticket Information	Event Leaderboard	Share Photos, Videos, and Reviews on Participated Events	Event Recommendation (Based on your preferences)	Payment	2D Seating View	Sell your tickets to others (black market)
Biletix	X	✓	X	X	✓	✓ (Öne Çıkanlar, Hot Tickets)	X	X	✓	✓	X
Ticketmaster	✓	✓	X	X	✓	✓ (Trending Events)	X	✓	✓	✓	✓
Peatix	✓	✓	X	X	✓ (Mainly Online Events)	X	X	X	✓	X (Online Events)	X
Plandy	X	✓	✓ (Biletix)	X	X (redirected to other websites)	✓	X	X	X	X	X
BSO	X	✓	X	X	✓	X	X	X	✓	✓	X
3KET	✓	✓	✓	✓	✓	✓	✓	✓	X	✓	X

Table 1 - Competitor Analysis

3. Proposed System

3.1. Overview

3KET will be an online entertainment venue viewing system that allows customers to view events and venues before attending the events they plan to attend. The application will offer many interactive features. Customers and regular users will use features such as 3D venue walkabouts with seat selection and information, event browsing through various filters, posting memories (such as photos and videos) or reviews related to events and venues, and participating in leaderboards. Business clients who are either venue owners or event organizers and sponsors will also have the advantage of many functional features. They will be able to create, edit and delete events, upload 3D models of a venue, match tickets to seats in a 3D model of a venue, add animations and sound effects to a model, and view and respond to reviews.

3.2. Functional Requirements

Venue Owner Functionalities

- Venue owners (event organizers) will have the ability to create, edit and delete events. They will have the option to add descriptions to events and assign websites for redirection during payment.
- Venue owners will be able to upload 3D models for venues that they designed using applications such as but not limited to Blender, Sketchup, Unity. The application should be able to accept the formats OBJ, FBX, glTF.
- Venue owners will match each seat in the 3D venue model with a ticket (assigned to that seat). Furthermore, presented with a birds-eye view, the seat marking process can be automated after some seats are matched by calculating the positions of the yet unmatched seats (such as marking the seats of a row between the beginning and end seats of that row).

- Venue owners will be able to view and respond to the reviews posted about their events or their venues by customers.
- Venue owners will be able to view and respond to the memories posted about their events or their venues by customers.
- Venue owners will be able to add animations and sound effects of events on the stage (animation of players playing soccer or an orchestra playing music) to make the view realistic and attractive.

Customer Functionalities

- Customers will be able to create a profile by entering their information (including name, surname, username, contact details, and profile picture), and selecting their preferences and hobbies.
- Customers will be able to roam inside 3D venue models, viewing the perspective for each seat that is matched to a ticket. Information about the tickets attached to the selected seat will be shown. Customers can be redirected to the preferred website for the ticket-sale of a seat they select.
- Customers will be able to view events in a categorized manner and search particular events according to filters such as location, genre, artist, venue and preferences.
- Customers will be able to see events recommended by the application according to their past events, recent searches, preferences, and leaderboards.
- Customers will be able to share memory posts containing photos and videos, as well as reviews about the events they participate in.
- Customers will be able to look at the event leaderboards, containing events that passed and got the highest interaction in the past day, week, month, and year.

- Customers will be able to look at the rank of events depending on their interaction that will happen at most in a week's time.

3.3. Non-functional Requirements

Usability

- Venue owners should be able to match seats with tickets in the editor only relying on the user manual.
- Customers should be able to look at the perspectives of different seats and proceed to ticket checkout websites by selecting the seat they wish to book only relying on the user manual.
- Customers should be able to view upcoming events within their region/city without specifying their location or time (events that are shown automatically without a search filter).
- Customers should be able to search, view, and favorite/mark events only relying on the user manual.
- Customers should be able to view events, view seats, purchase tickets by being redirected to the actual ticket vendor and view leaderboards without signing up and only relying on the user manual.
- Customers should be able to upload videos, photos, and reviews for events they have attended by only relying on the user manual.

Reliability

- Venue owners should be able to upload their 3D models using the service to the 3KET server without encountering any technical problems. They should also be able to edit or add some features to the models of their venues.

- Customers should be able to view 3D models for the venues and proceed to ticket checkouts on other websites using our browser application without encountering any technical problems.

Extensibility

- Our application can be extended to add new functionalities and interactive features for users and features on different types or genres of events and 3D modeling.

Maintainability

- The application can be modified in order to accept new 3D model formats that can be produced in the future.

Accessibility

- The users should be able to view the 3D models from popular web browsers like Google Chrome, Mozilla Firefox, Opera, Safari.
- Users should be able to reach 3KET by using different search engines such as Google, Bing, Yandex, etc.

Scalability

- The application will be able to automatically scale the resources to support the incoming requests without any additional configuration.

3.4. System Models

3.4.1. Scenarios

Scenario 1	
Use Case Name	Sign Up
Participating Actors	Initiated by guest or venue owner
Main Flow Events	<ol style="list-style-type: none">1. The user clicks at the “Sign Up” button.2. The user selects the desired user type to sign up, which can be “Customer” or “Venue owner”.3. The system returns a form to the user according to the user type choice.4. The user fills the form and clicks at the “Submit” button.5. The system gets the form data and checks the data for validation constraints.6. The system returns an acknowledgement to the user.
Entry Conditions	<ul style="list-style-type: none">• The user is not logged into the system.
Exit Conditions	<ul style="list-style-type: none">• The user received the acknowledgement.• The user goes back to the home page.

Scenario 2	
Use Case Name	Log In
Participating Actors	Initiated by guest or venue owner
Main Flow Events	<ol style="list-style-type: none">1. The user clicks at the “Log In” button.2. The user selects the desired user type to log in.3. The system returns a form to the user according to the user type choice.4. The user fills the form and clicks at the “Log In” button.5. The system gets the form data and checks its validity.6. The system returns an acknowledgement to the user.
Entry Conditions	<ul style="list-style-type: none">• The user is not logged into the system.• The user signed up to the system.
Exit Conditions	<ul style="list-style-type: none">• The user received the acknowledgement

Scenario 3	
Use Case Name	Log Out
Participating Actors	Initiated by registered customer or venue owner
Main Flow Events	<ol style="list-style-type: none"> 1. The user clicks at the “Log Out” button. 2. The web browser sends a request to log out. 3. The system handles the request and returns an acknowledgement to the user.
Entry Conditions	<ul style="list-style-type: none"> • The user is logged into the system.
Exit Conditions	<ul style="list-style-type: none"> • The user received the acknowledgement.

Scenario 4	
Use Case Name	See list of events
Participating Actors	Initiated by guest or registered customer
Main Flow Events	<ol style="list-style-type: none"> 1. The web browser sends a request to the system to get all events. 2. The system receives the request and responds back with the event list. 3. The web browser receives the event list and displays it to the customer.
Entry Conditions	<ul style="list-style-type: none"> • The customer is logged into the system.
Exit Conditions	<ul style="list-style-type: none"> • The customer logs out from the system. • The customer clicks to an event to see its details.

Scenario 5	
Use Case Name	See details of an event
Participating Actors	Initiated by guest or registered customer
Main Flow Events	<ol style="list-style-type: none"> 1. The customer clicks to the “See details” button of an event. 2. The web browser sends a request to get the details of the event to the system. 3. The system receives the request and responds back with the details of the event. 4. The web browser displays the details of the event to the customer.
Entry Conditions	<ul style="list-style-type: none"> • Customer is logged into the system.
Exit Conditions	<ul style="list-style-type: none"> • Customer goes back to the list of events. • Customer clicks at the “Select seat” button.

Scenario 6	
Use Case Name	View 2D layout of a venue
Participating Actors	Initiated by guest or registered customer
Main Flow Events	<ol style="list-style-type: none"> 1. The customer clicks at the “Select seat” button. 2. The web browser sends a request to the system to get the 2D layout of the venue. 3. The system receives the request and responds back with the 2D layout. 4. The web browser displays the 2D layout to the customer.
Entry Conditions	<ul style="list-style-type: none"> • Customer is logged into the system.
Exit Conditions	<ul style="list-style-type: none"> • The customer goes back to the details of the event. • The customer clicks on a seat in the 2D layout.

Scenario 7	
Use Case Name	Select seat from 2D layout
Participating Actors	Initiated by guest or registered customer
Main Flow Events	1. The customer selects a seat from the 2D layout.
Entry Conditions	<ul style="list-style-type: none"> • The customer is logged into the system. • The selected seat is available to buy.
Exit Conditions	<ul style="list-style-type: none"> • The customer goes back to the details of the event.

Scenario 8	
Use Case Name	See seat view from 3D model
Participating Actors	Initiated by guest or registered customer
Main Flow Events	<ol style="list-style-type: none"> 1. The customer selects a seat from the 2D layout. 2. The web browser sends a request to the system to get the 3D view from the specified seat. 3. The system receives the request and responds with the 3D view. 4. The web browser loads the 3D view and displays it to the customer.
Entry Conditions	<ul style="list-style-type: none"> • The customer is logged into the system. • The selected seat is available to buy.
Exit Conditions	<ul style="list-style-type: none"> • The customer goes back to the 2D layout.

Scenario 9	
Use Case Name	Share memory for an event
Participating Actors	Initiated by registered customer
Main Flow Events	<ol style="list-style-type: none"> 1. The customer clicks at the “Share memory” button on the events page. 2. The web browser sends a request to the system to get the “Share memory” form. 3. The system receives the request and responds with the form. 4. The customer fills the form and clicks at the “Submit” button. 5. The system handles the request and responds with an acknowledgement.
Entry Conditions	<ul style="list-style-type: none"> • The customer is logged into the system. • The customer has gone to the event.
Exit Conditions	<ul style="list-style-type: none"> • The customer receives the acknowledgement. • The customer goes back to the home page.

Scenario 10	
Use Case Name	Give feedback to the venue
Participating Actors	Initiated by registered customer
Main Flow Events	<ol style="list-style-type: none"> 1. The customer clicks at the “Give Feedback” button. 2. The web browser sends a request to the system to get the “Give Feedback” form. 3. The system receives the request and responds with form. 4. The web browser gets the form and displays it to the customer. 5. The customer fills the form and clicks at the “Submit” button. 6. The web browser sends the form data to the system. 7. The system gets the form data and returns an acknowledgement.
Entry Conditions	<ul style="list-style-type: none"> • The customer is logged into the system.
Exit Conditions	<ul style="list-style-type: none"> • The customer receives the acknowledgement. • The customer goes back to the home page.

Scenario 11	
Use Case Name	See leaderboard of events
Participating Actors	Initiated by registered customer
Main Flow Events	<ol style="list-style-type: none"> 1. The customer clicks the “Explore” button. 2. The web browser sends a request to the system to get the leaderboard of events. 3. The system receives the request and responds with the leaderboard. 4. The web browser displays the leaderboard.
Entry Conditions	<ul style="list-style-type: none"> • The customer is logged into the system.
Exit Conditions	<ul style="list-style-type: none"> • The customer goes back to the home page.

Scenario 12	
Use Case Name	Search events
Participating Actors	Initiated by guest or registered customer
Main Flow Events	<ol style="list-style-type: none"> 1. The customer fills the search bar and clicks the “Search” button. 2. The web browser sends a request to the system to get the corresponding events according to the input. 3. The system handles the request and responds with the list of events. 4. The web browser displays the searched events.
Entry Conditions	<ul style="list-style-type: none"> • The customer is logged into the system.
Exit Conditions	<ul style="list-style-type: none"> • The customer receives the acknowledgement. • The customer goes back to the home page.

Scenario 13	
Use Case Name	See recommended events
Participating Actors	Initiated by registered customer
Main Flow Events	<ol style="list-style-type: none"> 1. The customer clicks at the “Explore” button. 2. The web browser sends a request to the system to get the recommended events. 3. The system handles the request and responds with the list of recommended events. 4. The web browser displays the recommended events.
Entry Conditions	<ul style="list-style-type: none"> • The customer is logged into the system.
Exit Conditions	<ul style="list-style-type: none"> • The customer receives the acknowledgement. • The customer goes back to the home page.

Scenario 14	
Use Case Name	Add venue
Participating Actors	Initiated by venue owner
Main Flow Events	<ol style="list-style-type: none"> 1. The venue owner clicks at the “Add Venue” button. 2. The web browser sends a request to the system to get the “Add Venue” form. 3. The system receives the request and responds with the form. 4. The web browser gets the form and displays it to the venue owner. 5. The venue owner fills the form and clicks at the “Submit” button. 6. The web browser sends the form data to the system. 7. The system gets the form data and returns an acknowledgement.
Entry Conditions	<ul style="list-style-type: none"> • The venue owner is logged into the system.
Exit Conditions	<ul style="list-style-type: none"> • The venue owner receives the acknowledgement. • The venue owner goes back to the home page.

Scenario 15	
Use Case Name	Import 3D model of venue
Participating Actors	Initiated by venue owner
Main Flow Events	<ol style="list-style-type: none"> 1. The venue owner clicks at the “Import 3D model” button. 2. The venue owner selects the 3D model file from the file browser. 3. The web browser uploads the 3D model to the system. 4. The system returns an acknowledgement.
Entry Conditions	<ul style="list-style-type: none"> • The venue owner is logged into the system. • The venue owner has a venue added to the system.
Exit Conditions	<ul style="list-style-type: none"> • The venue owner receives the acknowledgement. • The venue owner goes back to the “Venue details” page.

Scenario 16	
Use Case Name	Match seats in 3D model with tickets
Participating Actors	Initiated by venue owner
Main Flow Events	<ol style="list-style-type: none"> 1. The venue owner clicks at the “Match seats with tickets” button. 2. The web browser gets the 3D model from the system and displays it. 3. The venue owner matches each seat with a ticket. 4. The venue owner clicks at the “Done” button. 5. The web browser sends a request to the system to submit the seat data. 6. The system receives the data, processes it, and returns an acknowledgement to the venue owner.
Entry Conditions	<ul style="list-style-type: none"> • The venue owner is logged into the system. • The venue owner has a venue added to the system. • The venue owner uploaded a 3D model of the venue.
Exit Conditions	<ul style="list-style-type: none"> • The venue owner receives the acknowledgement. • The venue owner goes back to the “Venue details” page.

Scenario 17	
Use Case Name	Create an event
Participating Actors	Initiated by venue owner
Main Flow Events	<ol style="list-style-type: none"> 1. The venue owner clicks at the “Add event” button. 2. The web browser sends a request to the system to get the “Add event” form. 3. The system receives the request and returns the form. 4. The venue owner fills the form and clicks on the “Submit” button. 5. The web browser sends the form data into the system. 6. The system receives the form data, handles it, and returns an acknowledgement to the venue owner.
Entry Conditions	<ul style="list-style-type: none"> • The venue owner is logged into the system. • The venue owner has a venue added to the system.
Exit Conditions	<ul style="list-style-type: none"> • The venue owner receives the acknowledgement. • The venue owner goes back to the home page.

Scenario 18	
Use Case Name	Update an event
Participating Actors	Initiated by venue owner
Main Flow Events	<ol style="list-style-type: none"> 1. The venue owner clicks on the “Update event” button. 2. The web browser sends a request to the system to get the “Update event” form. 3. The system receives the request and returns the form. 4. The venue owner fills the form and clicks on the “Submit” button. 5. The web browser sends the form data into the system. 6. The system receives the form data, handles it and returns an acknowledgement to the venue owner.
Entry Conditions	<ul style="list-style-type: none"> • The venue owner is logged into the system. • The venue owner has a venue added to the system. • The venue owner has an event added to the system.
Exit Conditions	<ul style="list-style-type: none"> • The venue owner receives the acknowledgement. • The venue owner goes back to the home page.

Scenario 19	
Use Case Name	Delete an event
Participating Actors	Initiated by venue owner
Main Flow Events	<ol style="list-style-type: none"> 1. The venue owner clicks at the “Delete event” button. 2. The web browser sends a request to the system to delete the event. 3. The system handles the request and returns an acknowledgement to the venue owner.
Entry Conditions	<ul style="list-style-type: none"> • The venue owner is logged into the system. • The venue owner has a venue added to the system. • The venue owner has an event added to the system.
Exit Conditions	<ul style="list-style-type: none"> • The venue owner receives the acknowledgement. • The venue owner goes back to the home page.

Scenario 20	
Use Case Name	See feedback of the customers
Participating Actors	Initiated by venue owner
Main Flow Events	<ol style="list-style-type: none"> 1. The venue owner clicks at the “Feedbacks” button. 2. The web browser sends a request to the system to get the feedback. 3. The system handles the request and sends the feedback. 4. The web browser displays the feedback.
Entry Conditions	<ul style="list-style-type: none"> • The venue owner is logged into the system. • The venue owner has a venue added to the system. • The venue owner has an event added to the system. • The customers have given feedback to the venue.
Exit Conditions	<ul style="list-style-type: none"> • The venue owner goes back to the main page.

Scenario 21	
Use Case Name	Respond to the feedbacks
Participating Actors	Initiated by venue owner
Main Flow Events	<ol style="list-style-type: none"> 1. The venue owner clicks at the “Respond” button. 2. The web browser sends a request to the system to get the “Respond” form. 3. The system receives the request and returns the form. 4. The web browser displays the form to the venue owner. 5. The venue owner fills the form and clicks the “Send” button. 6. The web browser sends the form data to the system. 7. The system handles the form data, and returns an acknowledgement.
Entry Conditions	<ul style="list-style-type: none"> • The venue owner is logged into the system. • The venue owner has a venue added to the system. • The venue owner has an event added to the system. • The customers have given feedback to the venue.
Exit Conditions	<ul style="list-style-type: none"> • The venue owner receives the acknowledgement. • The venue owner goes back to the main page.

Scenario 22	
Use Case Name	Checkout
Participating Actors	Initiated by the registered customer and the payment gateway
Main Flow Events	<ol style="list-style-type: none"> 1. The customer gets redirected to the payment gateway website by the system. 2. The customer fills the form received from the payment gateway and clicks at the “Submit” button. 3. The payment gateway receives the form data, checks its validity, and returns an acknowledgement.
Entry Conditions	<ul style="list-style-type: none"> • The customer is logged into the system. • The customer has selected a seat from the seat layout. • The selected seat is available to buy.
Exit Conditions	<ul style="list-style-type: none"> • The customer receives the acknowledgement.

Scenario 23	
Use Case Name	Report a memory
Participating Actors	Initiated by the registered customer
Main Flow Events	<ol style="list-style-type: none"> 1. The customer clicks at the “Report” button. 2. The web browser sends a request to the system to get the “Report” form. 3. The system receives the request and returns the form. 4. The web browser displays the form to the customer. 5. The customer fills the form and clicks at the “Send” button. 6. The web browser sends the form data to the system. 7. The system handles the form data, and returns an acknowledgement.
Entry Conditions	<ul style="list-style-type: none"> • The customer is logged into the system. • There are shared memories of an event.
Exit Conditions	<ul style="list-style-type: none"> • The customer receives the acknowledgement.

Scenario 24	
Use Case Name	Delete a reported memory
Participating Actors	Initiated by the venue owner
Main Flow Events	<ol style="list-style-type: none"> 1. The venue owner clicks at the “Delete” button. 2. The web browser sends this delete request to the system. 3. The system handles the request and returns an acknowledgement
Entry Conditions	<ul style="list-style-type: none"> • The venue owner is logged into the system. • The venue owner has a venue added into the system. • The venue owner has an event added into the system. • There are shared memories of an event. • The memory that will be deleted is reported by another customer.
Exit Conditions	<ul style="list-style-type: none"> • The venue owner receives the acknowledgement.

3.4.2. Use Case Model

Figure 1 shows the use case diagram of our application.

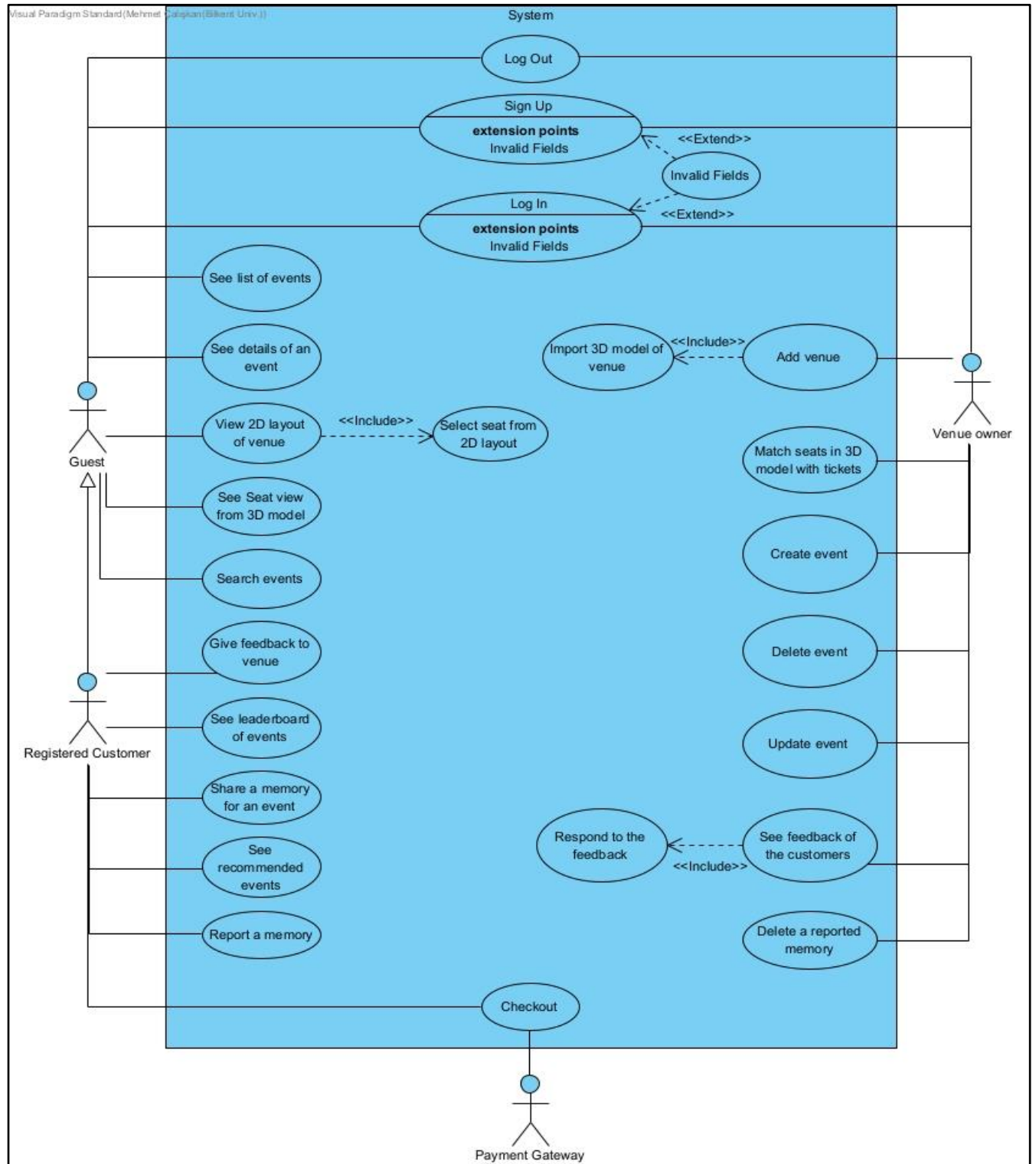


Figure 1 - Use Case Diagram

3.4.3. Object and Class Model

Figure 2 shows the UML Object and Class Diagram of our application.

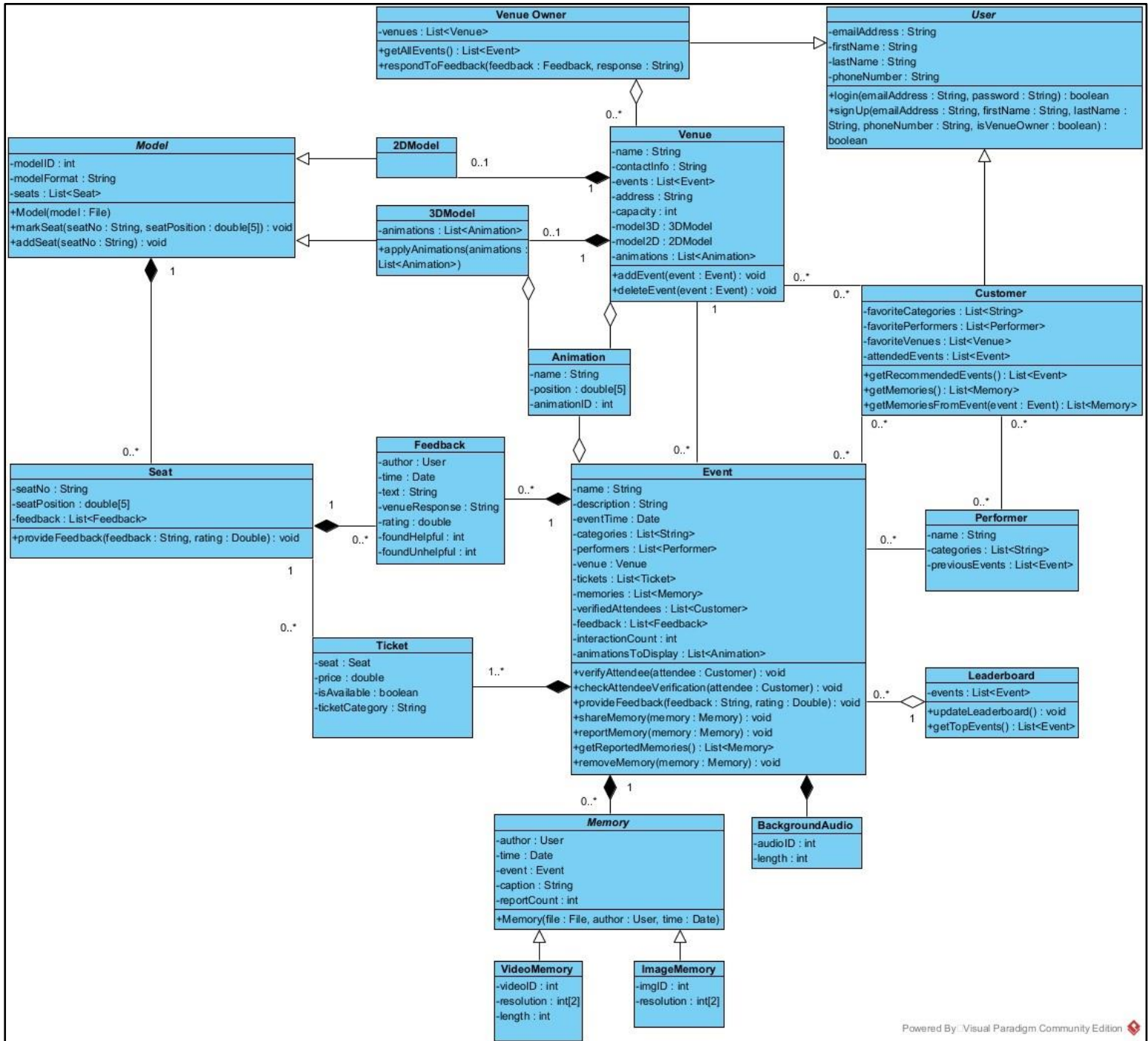


Figure 2 - UML Object and Class Diagram

3.4.4. Dynamic Models

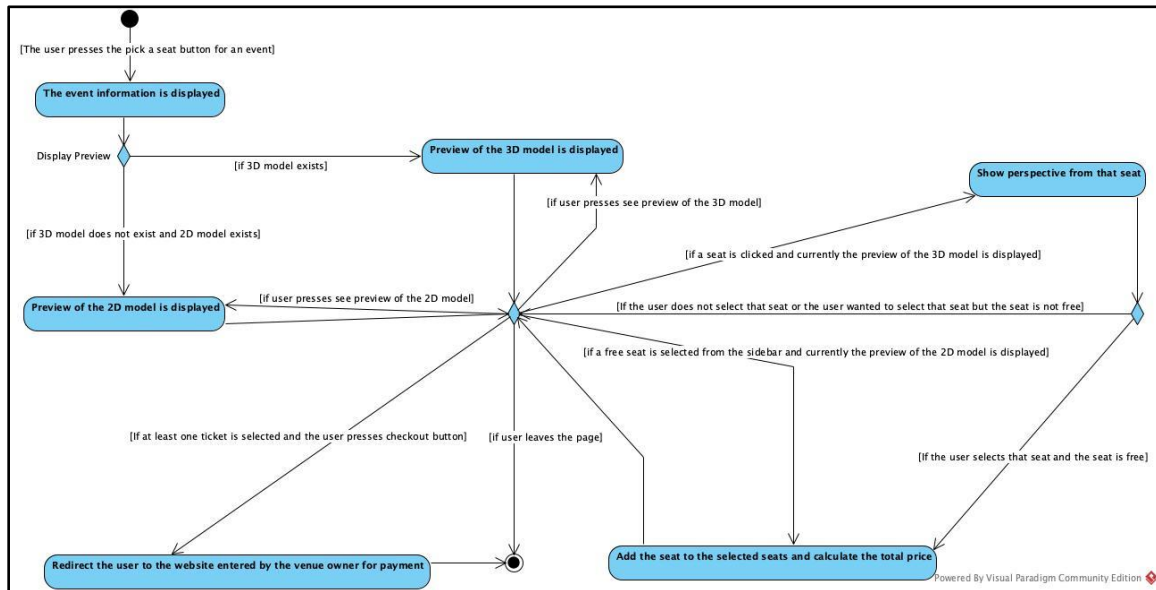


Figure 3 - Activity Diagram for Choosing a Seat for an Event

Figure 3 shows the activity diagram for a customer choosing a seat for an event they wish to attend. The system displays the event information when the user presses the "pick a seat" button for an event. If the 3D model exists, the preview of the 3D model is displayed. If the 3D model does not exist, but the 2D model exists, the preview of the 2D model is displayed. The user can switch the preview if both models exist. If the user clicks on a seat and currently the 3D model is being shown, then that seat's perspective is shown. The user can look from different angles to the seat they want. If the user selects that seat and the seat is free, the seat is added to the selected seats, and the total price is calculated.

Similarly, if the user selects a free seat from the sidebar, the seat is added to the selected seats, and the total price is calculated. In both cases, if the seat is not free, or the user does not select that seat, the system does not take any action. If the user selects one ticket and presses the checkout button, the user is directed to checkout on the website specified by the venue owner.

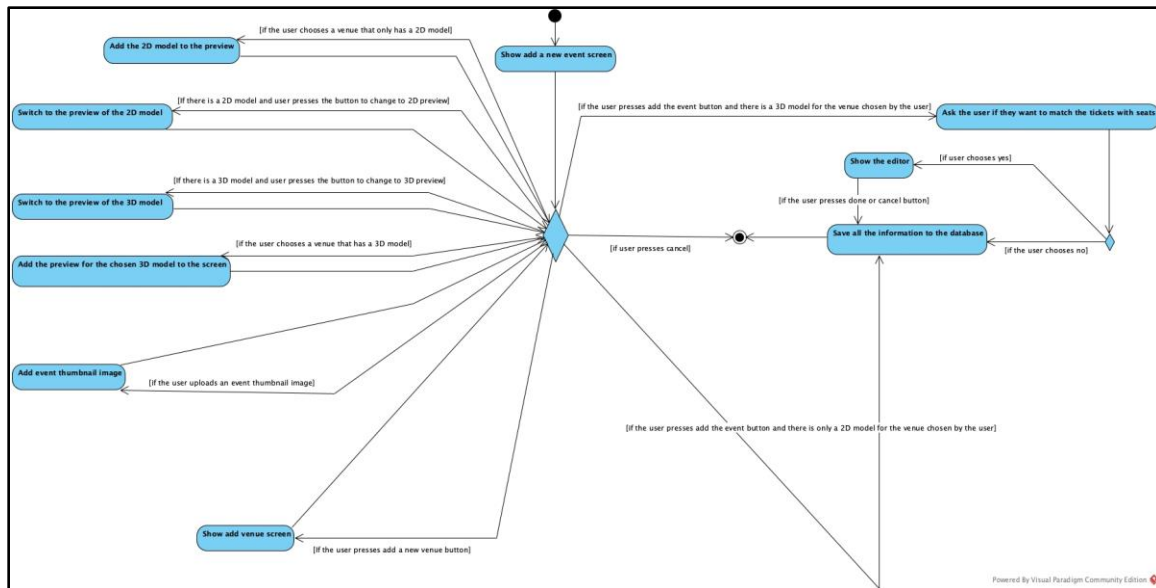


Figure 4 - Activity Diagram for Creating a New Event

Figure 4 shows the activity diagram for creating a new event. When a venue owner wants to create a new event, adding a new event interface is displayed. The user can add an event thumbnail image. The user can choose a venue from the ones s/he uploaded before or upload a new venue. If the user wants to add a new venue, the add venue screen is displayed. When the user chooses a venue, they upload the preview of the model in the past. If the 3D model exists for that venue, the 3D model is displayed in the preview. Otherwise, the 2D preview is shown in the preview. If both models exist, users can switch the previews. If the user presses the add button and there is a 3D model, they are asked if they want to match the tickets with the seats in the 3D model. If the user wants to match the tickets with the seats, the editor is shown. When the user completes the process, the information is uploaded to the database. If the venue owner provides only the 2D model and presses the add event button, the system saves the data.

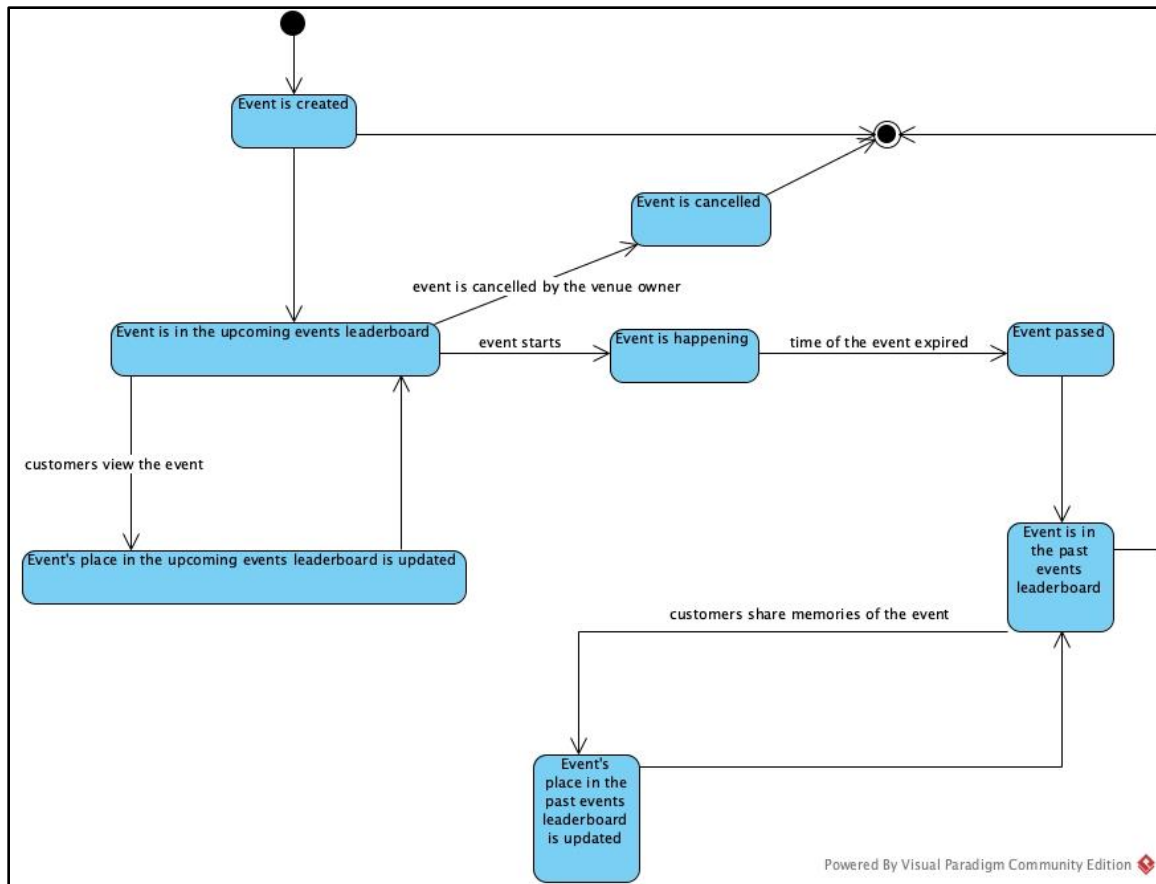


Figure 5 - State Diagram for Event

Figure 5 shows the state diagram of an event. After a venue owner creates an event, it takes its place in the upcoming events leaderboard. When customers view the event, the event's place in the leaderboard changes. When an event passes, it is deleted from the upcoming events leaderboard and added to the past events leaderboard. If customers share memories of the event, the event's place in the past events leaderboard changes.

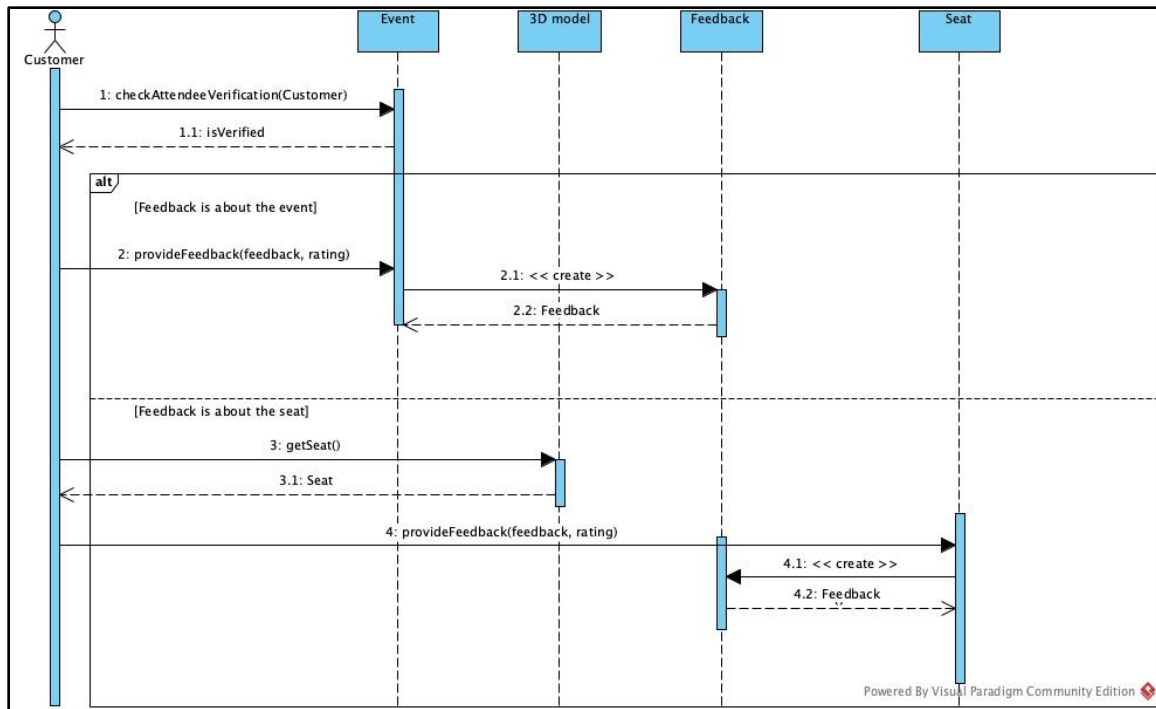


Figure 6 - Sequence Diagram for Giving Feedback

Figure 6 explains the scenario for giving feedback on an event or a seat on a 3D model. The customer must be first verified for that event. If the customer is verified, they can give feedback on the event or the seat in the venue.

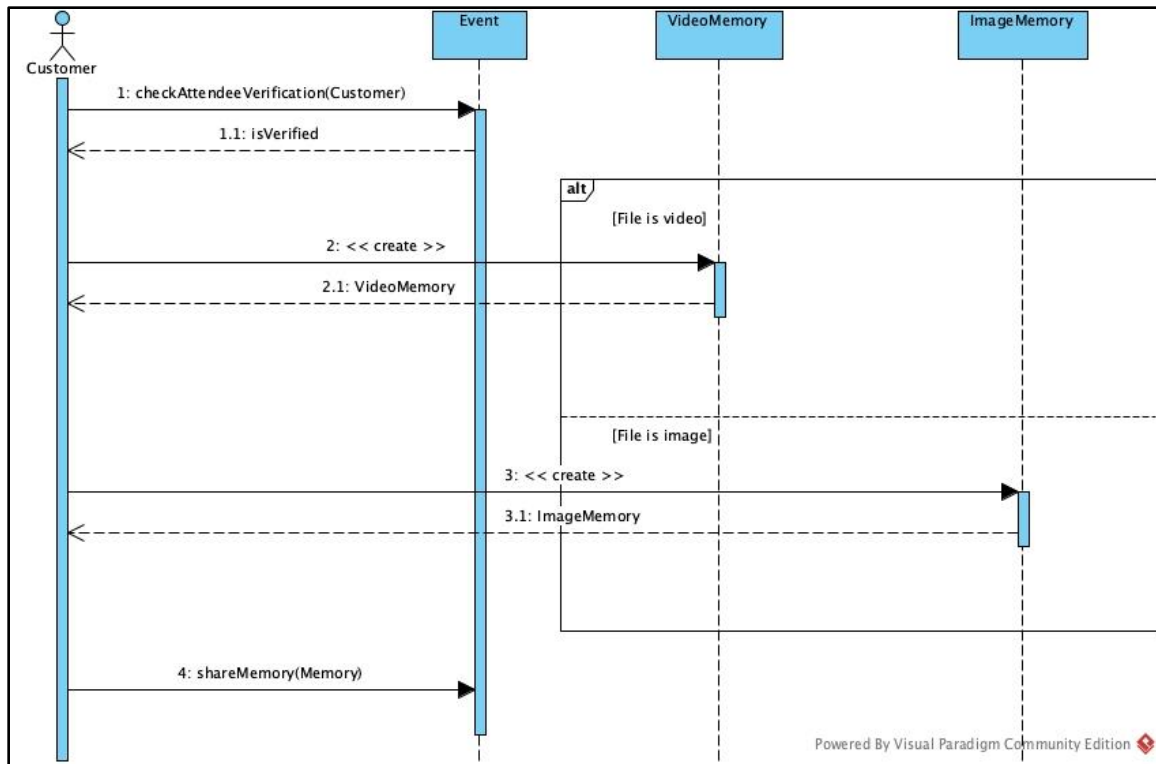


Figure 7 - Sequence Diagram for Sharing Memory

Figure 7 explains the scenario for sharing memory on an event. The customer must be verified for the particular event to share their memory. If the customer is verified, they can upload a video or an image. The created memory is then uploaded to the system to be shared for the event.

3.4.5. User Interface - Navigational Paths and Screen Mock-ups

3.4.5.1. Business Side Mockups

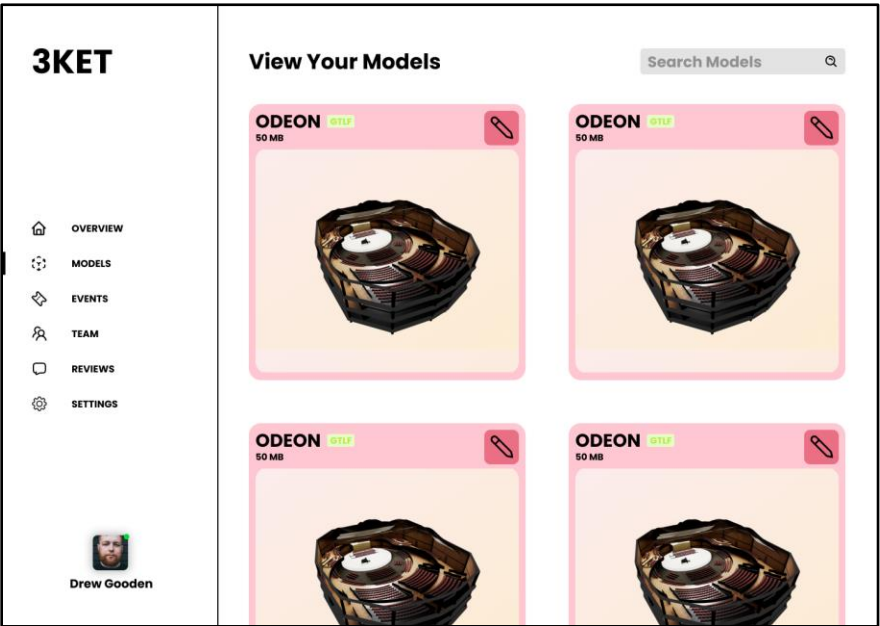


Figure 8 - Business Dashboard, “Overview” Screen

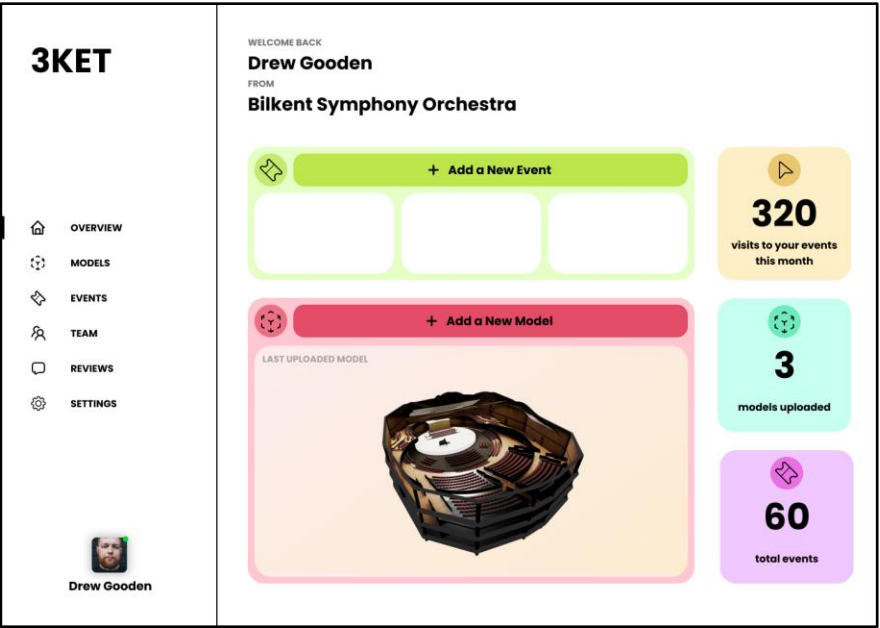


Figure 9 - Business Dashboard, “Models” Screen

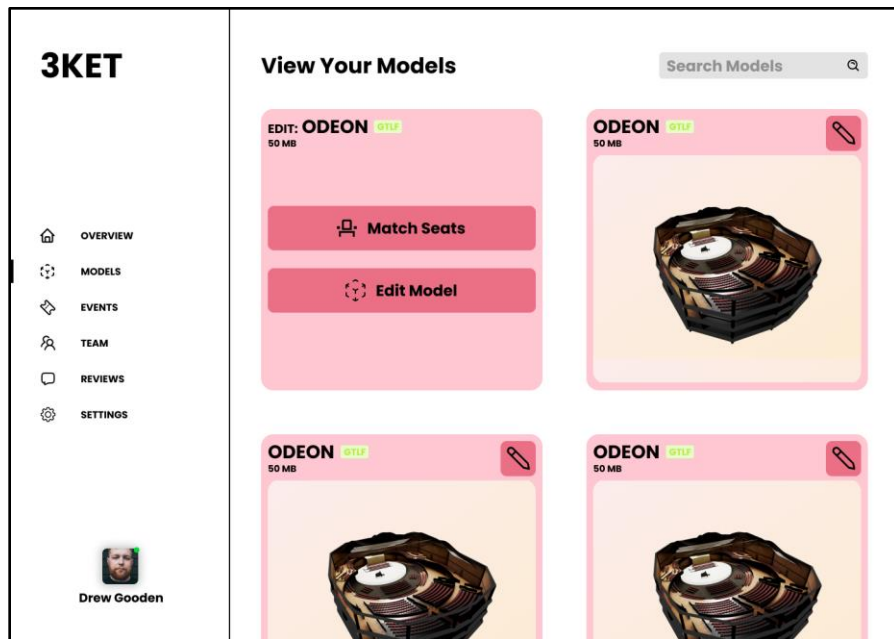


Figure 10 - Business Dashboard, “Edit Model” View



Figure 11 - Business Dashboard, “Match Seats - Editor” Screen



Figure 12 - Business Dashboard, Aerial View of Matched Seats

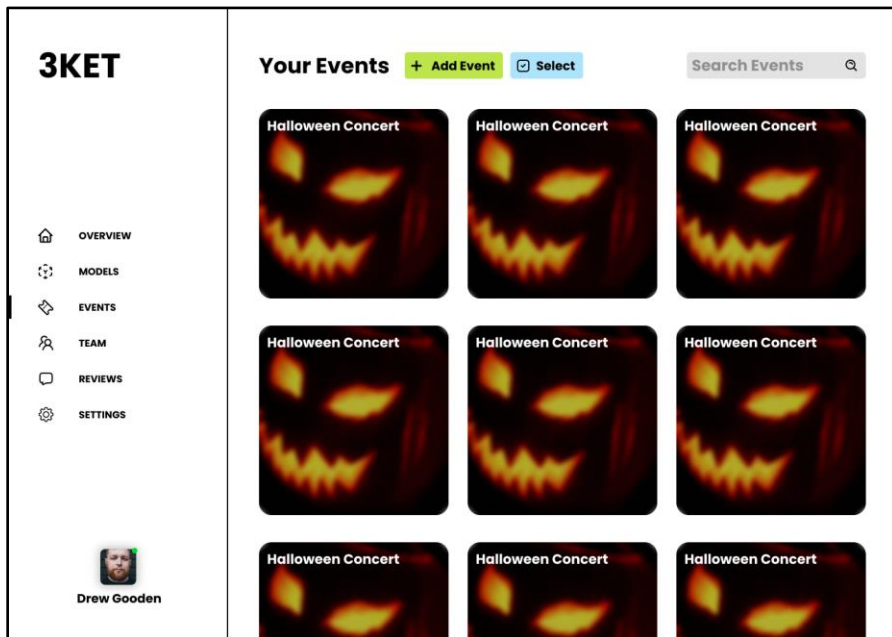


Figure 13 - Business Dashboard, "Events" Screen

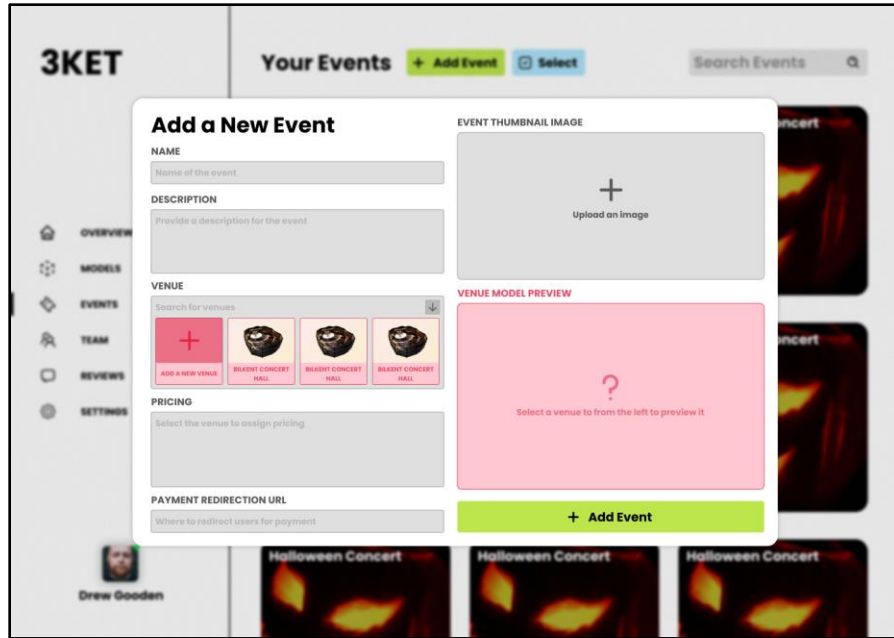


Figure 14 - Business Dashboard, “Add Event” Overlay, unfilled

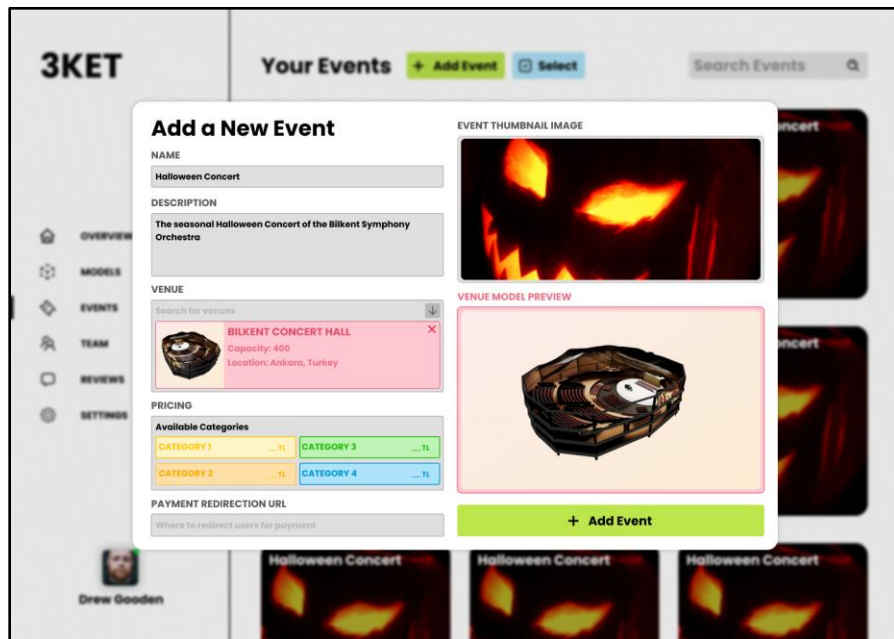


Figure 15 - Business Dashboard, “Add Event” Overlay, filled

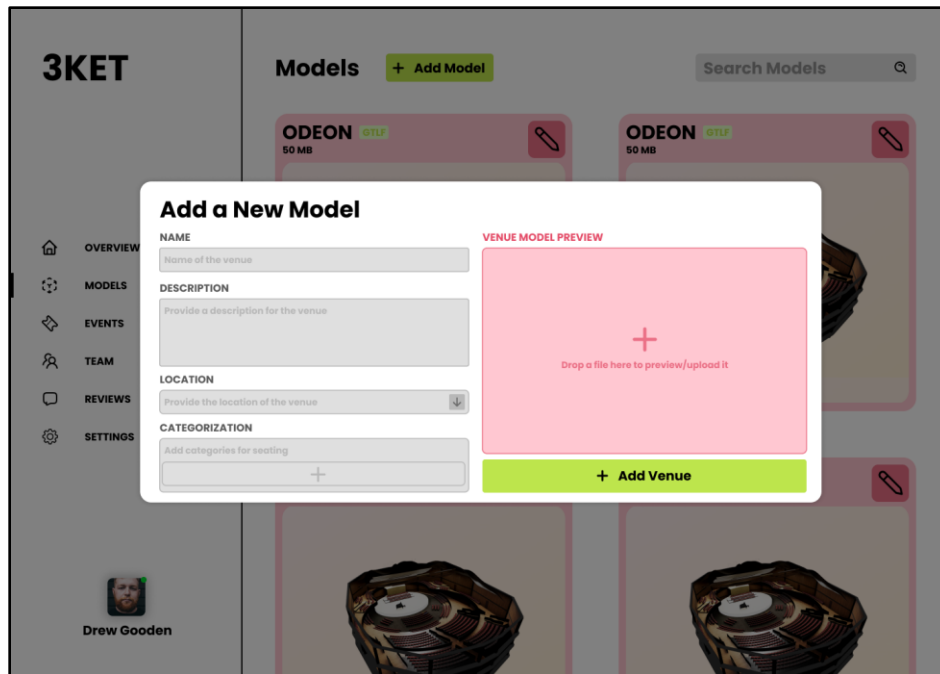


Figure 16 - Business Dashboard “Add Model” Popup, unfilled

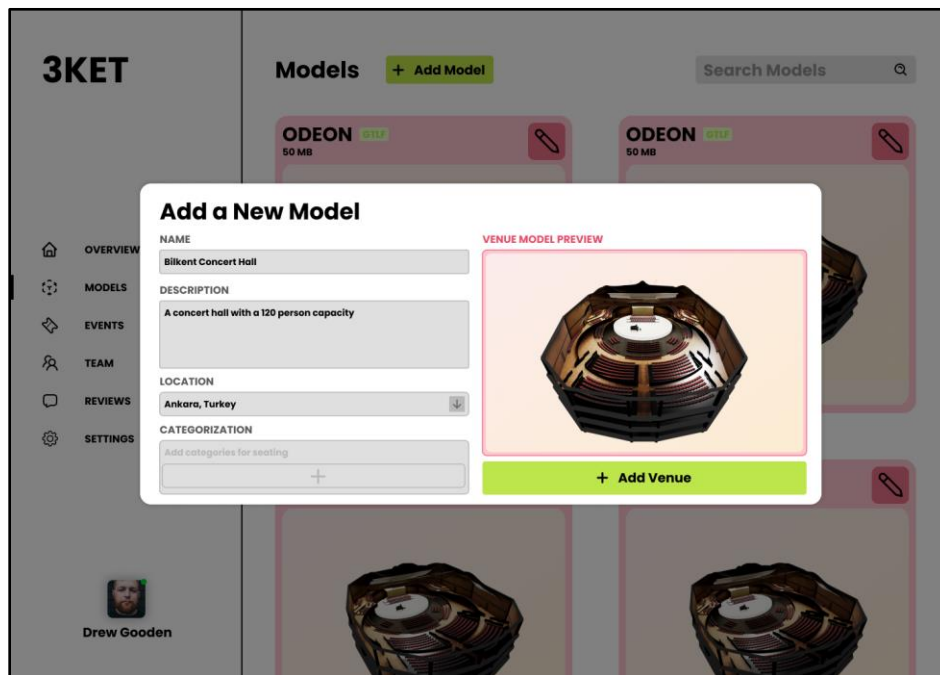


Figure 17 - Business Dashboard “Add Model” Popup, filled

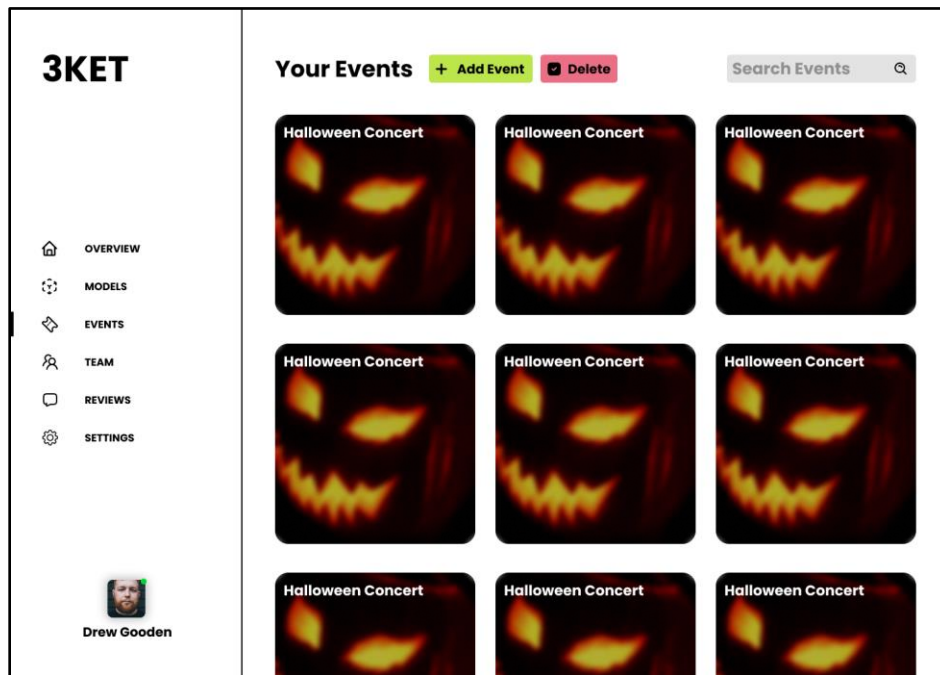


Figure 18 - Business Dashboard "Delete Event" Screen

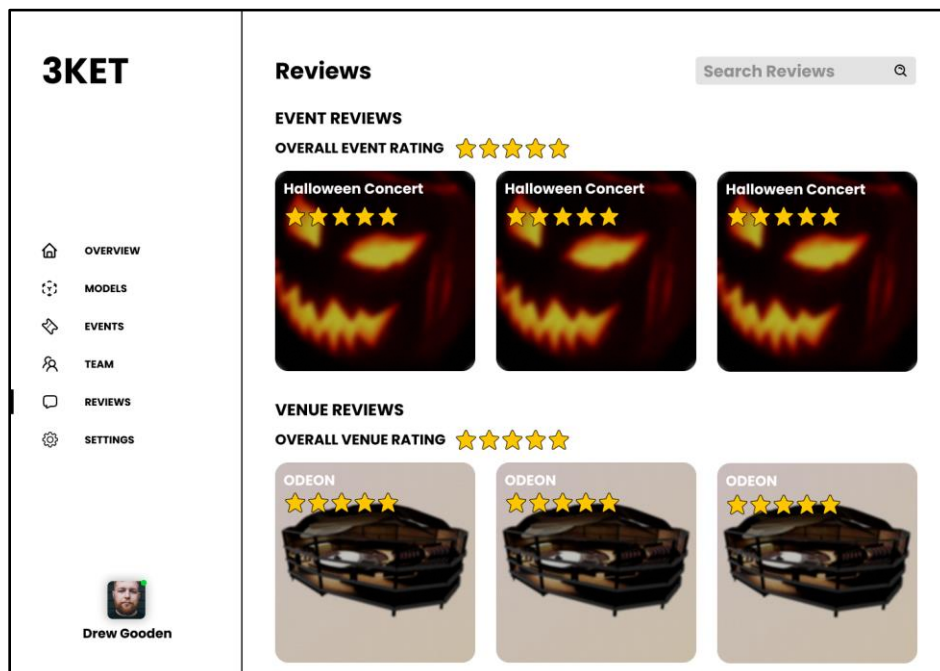


Figure 19 - Business Dashboard "Reviews" Screen

3.4.5.2. Customer Side Mockups

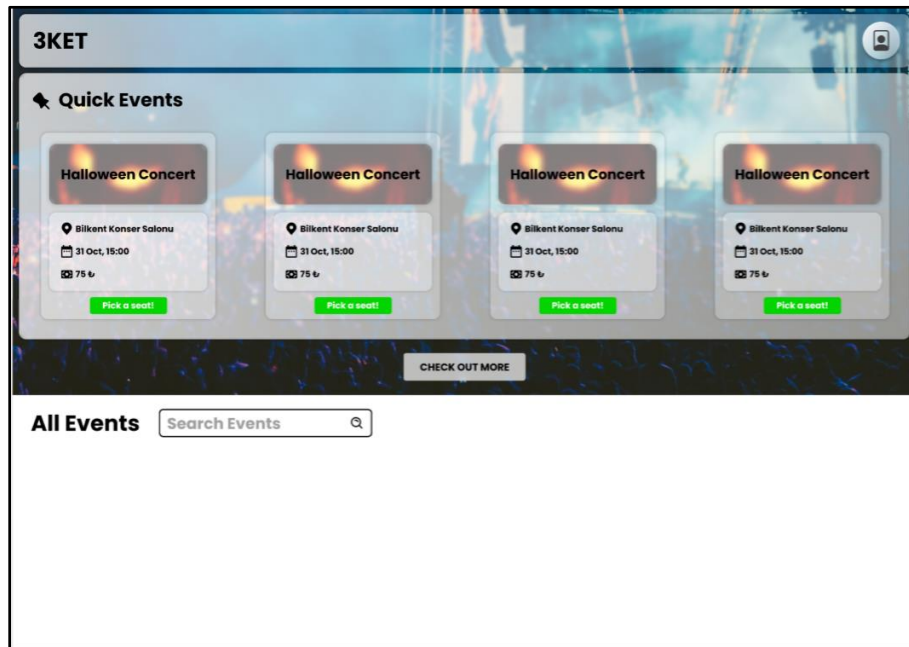


Figure 20 - Customer Landing Page (Not Signed In)

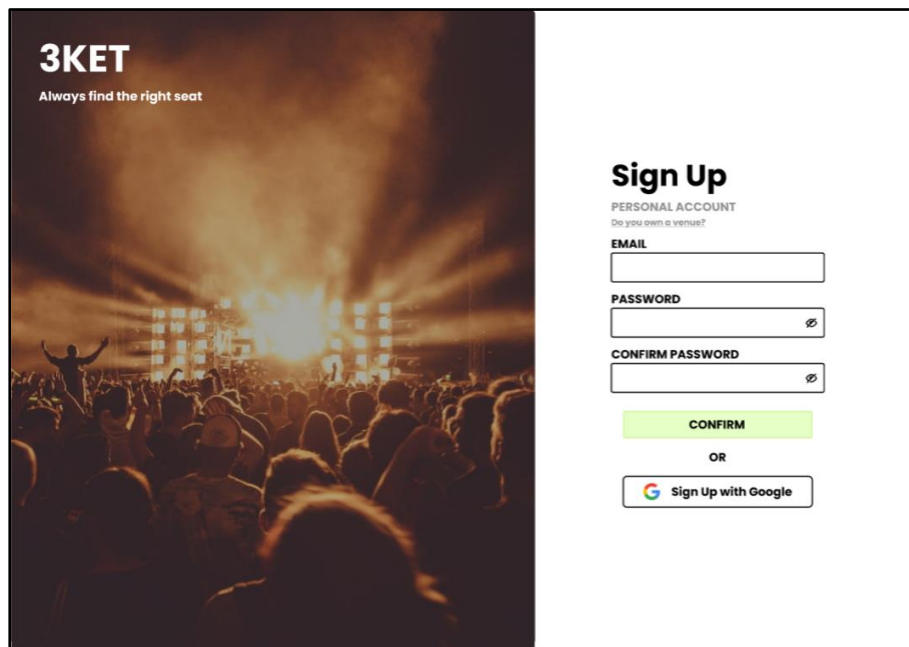


Figure 21 - Customer Sign Up/Sign In

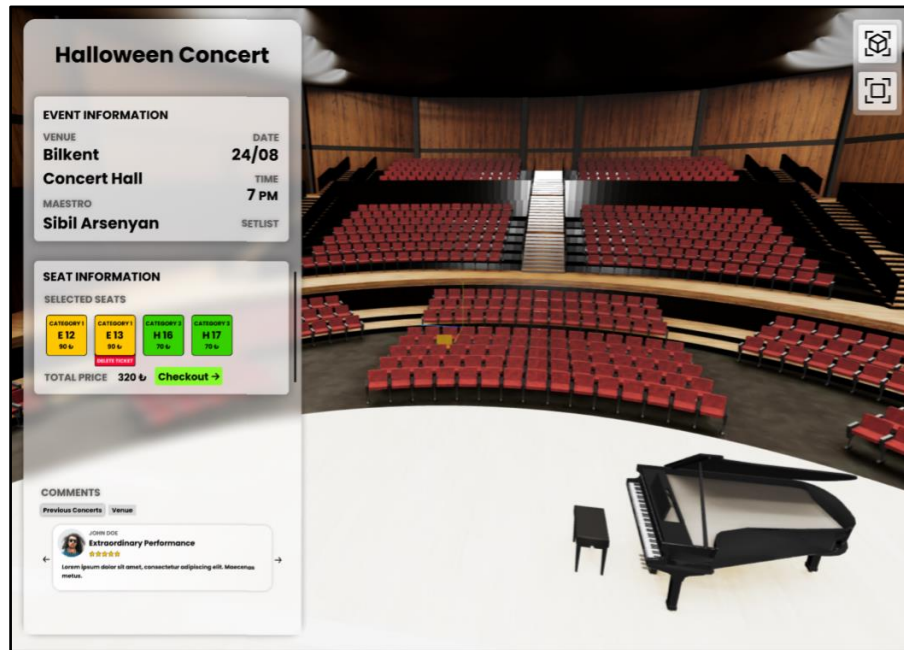


Figure 22 - Customer 3D Venue Walkabout



Figure 23 - Customer 3D Venue Walkabout (Hidden UI)

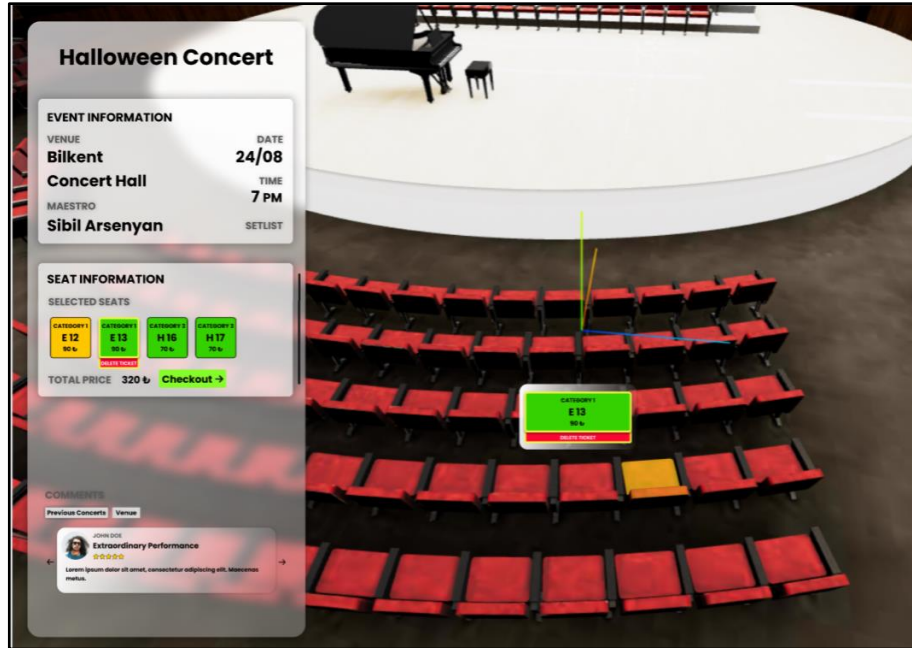


Figure 24 - Customer Seat Selection during 3D Venue Walkabout

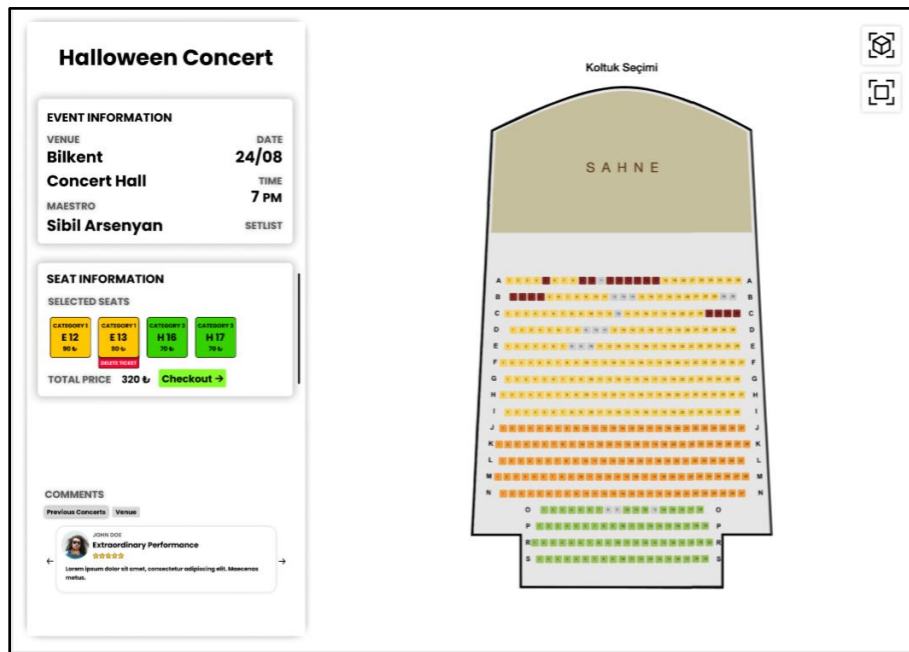




Figure 26 - Customer Perspective 3D View

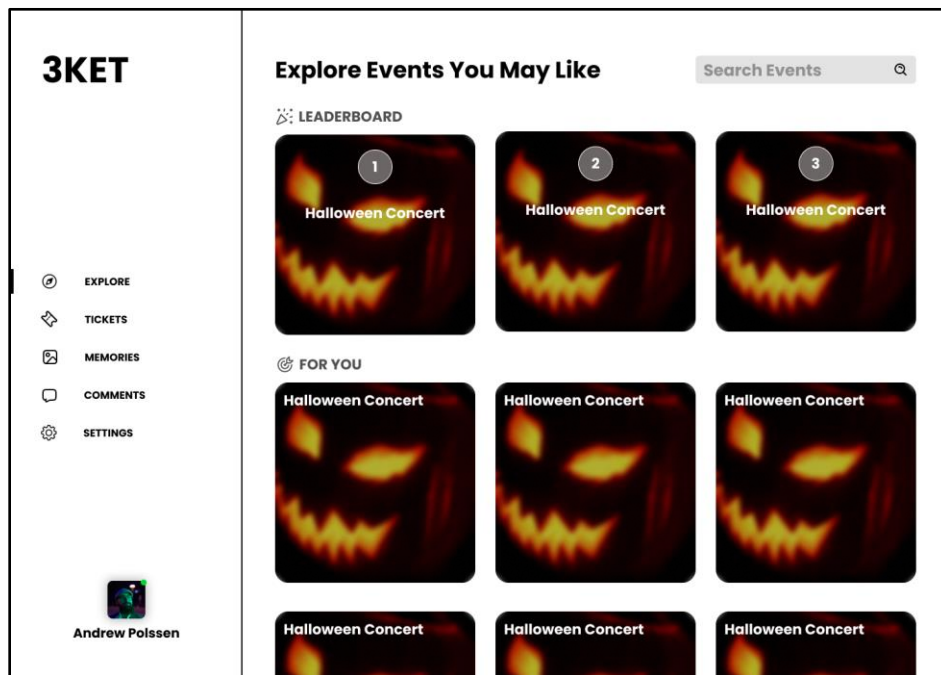


Figure 27 - Customer Explore Page with Recommended Events and Leaderboard

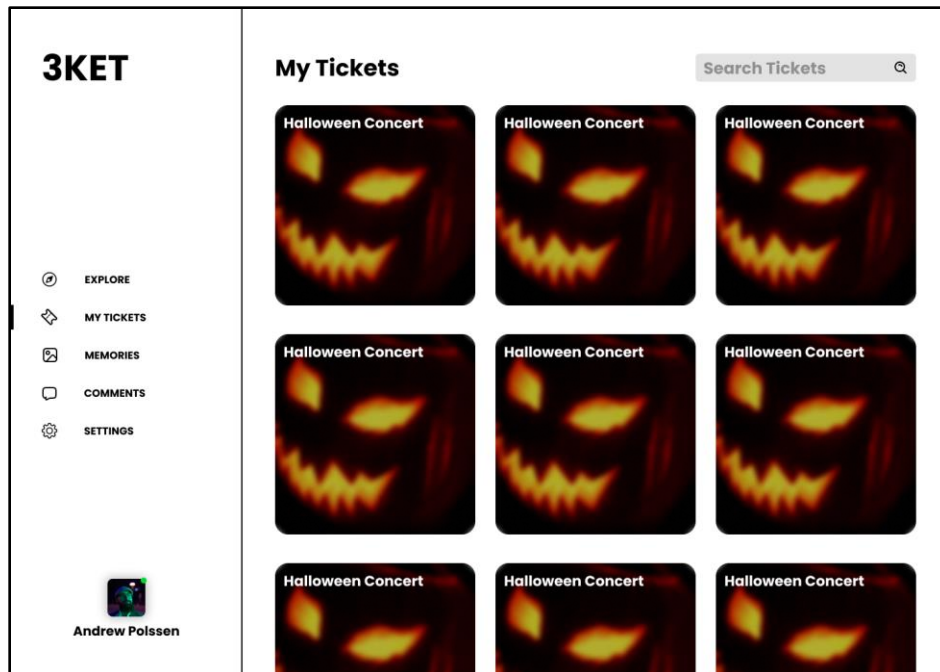


Figure 28 - Customer “My Tickets” Page

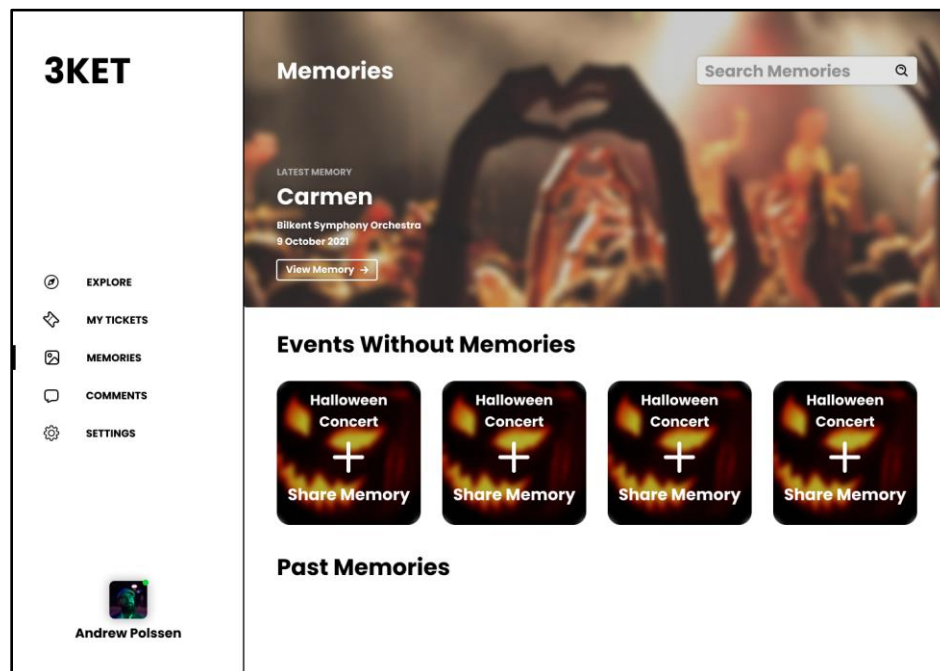


Figure 29 - Customer “Memories” Page

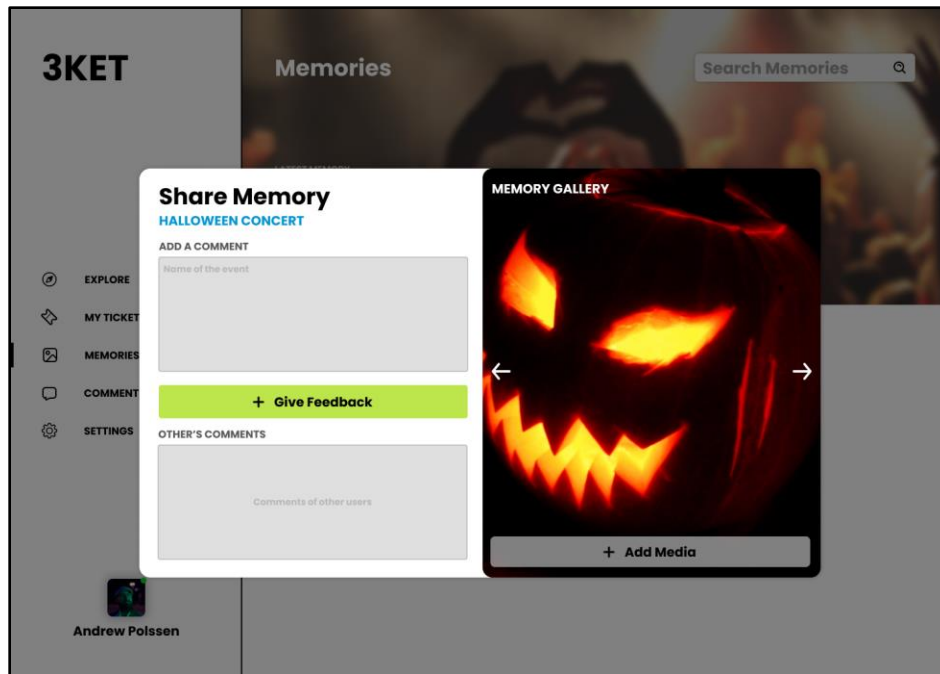


Figure 30 - Customer “Share Memory” Popup

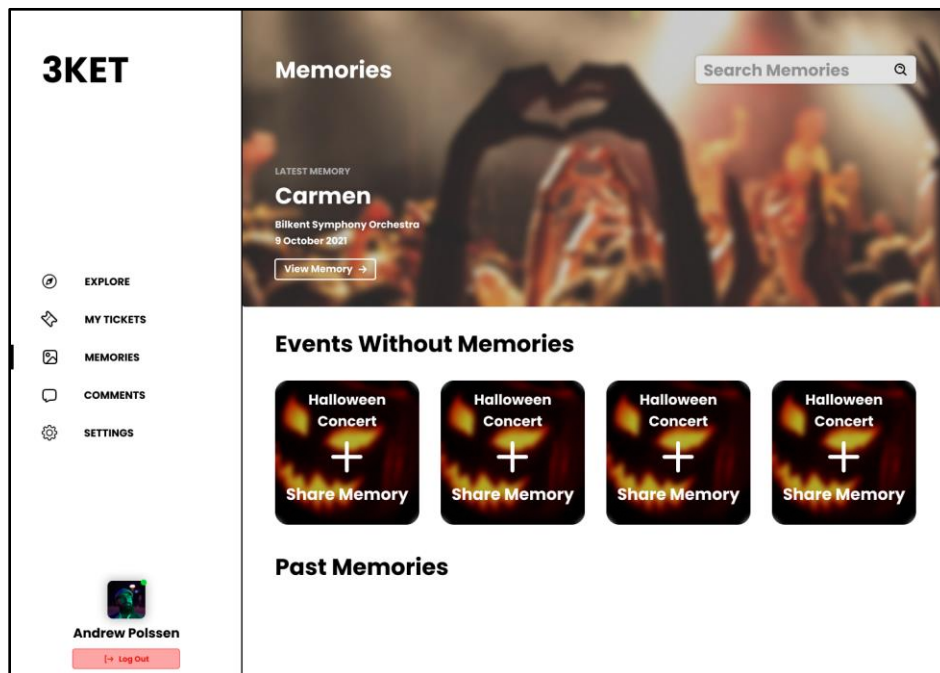


Figure 31 - Customer “Log Out” Button Displayed

4. Other Analysis Elements

4.1. Consideration of Various Factors in Engineering Design

Table 2 summarizes various factors affecting the software analysis and design of our application. We provide the details of these factors in the sequel.

Public Health

- The presented application can direct people to participate in events that can entertain them. This can affect their mental health positively.
- The application will also check if the attendance of events, memories, and seat-selection are following general COVID-19 guidelines. Warnings may be given to users if guidelines are not followed.

Public Safety

- Personal information on users such as email, name, city, or address will be held in our database. This information will not be shared with anyone else by the system.
- Second, users will be able to share photos and videos of the events. There will not be any restriction on the content. The content of some photos and videos may contain unrelated and disturbing content. Additionally, the content may spread on the internet without the notice of the person who shared it. If the shared content is reported, it will first be reviewed by the venue owner, and if found inappropriate, that content will be removed from the application.

Factor	Effect Level	Effect
Public Health	Medium	<ul style="list-style-type: none"> • Participating in events can entertain people, increase their mental health • COVID-19 guidelines will be encouraged
Public Safety	High	<ul style="list-style-type: none"> • User information will be kept secure and private. • Shared contents will be removed if it is reported.
Economic Factors	High	<ul style="list-style-type: none"> • Attract more people to events, generates revenue for business clients • Creating 3D models can cost venue owners • Application is free • Many ticket options
Global Factors	Low	<ul style="list-style-type: none"> • People can travel to participate in events
Cultural Factors	Low	<ul style="list-style-type: none"> • Interaction and attendance of local/underground will help boost local cultures of towns/cities
Environmental Factors	Low	<ul style="list-style-type: none"> • Venue owners can make changes in their venues according to customer feedback
Social Factors	High	<ul style="list-style-type: none"> • Participation in events

Table 2 - Factors Affecting Software Analysis and Design

Economic Factors

- Our application can help venue owners to attract more customers to their events. This will help increase their general revenue. Other business clients like sponsors or organizers will also be able to generate more revenue if their events show an increase in attendance and interaction. The application will allow businesses to interact with participants much more and can help them increase their ratings and advertisements.
- Creating 3D models can be costly for business clients (time, money).
- Our application is free and open to everyone. There are no economic limitations in our application.
- Seat selection will (according to venue availability) offer different prices for different categories of seats, which will be beneficial for users. These seats can also be purchased through different ticket-sale systems, which will be beneficial for users.

Global Factors

- Our application can be used all around the world. People can participate in any event in any location they wish. The user experience can be improved by the number of events and the interactiveness (animation, sound) of the 3D models.

Cultural Factors

- The application can help users discover cultural events close to their location. People can also participate in more underground and local events, which may help build local communities for certain locations and contribute to the culture of that town/city.

Environmental Factors

- Customers and users will be able to comment on the environmental effects of a venue by posting reviews, which could help the venue owners use any feedback to keep their venues environmentally-friendly.

Social Factors

- Our application can help people socialize by participating in events and sharing memories from them. The application will help build communities for events or venues with the amount of interaction it offers to users.

4.2. Risks and Alternatives

The 3D view feature in our application is not being widely used currently in similar applications. As a result of this feature, there are four possible risks in our application (see Table 3).

First, finding 3D models to test our application may be difficult. Although we plan to use 3D models we find on the internet, such as 3D warehouse [15] and Unity Asset Store [16], they can cost money. Although the risk is low, not finding enough 3D models can make the testing of the editor challenging. Also, presenting our application to users will be hard. A possible alternative to this problem would be creating more 3D models or using 2D models to make our application enjoyable.

Second, creating 3D models of venues such as Bilkent Odeon or Bilkent Üniversitesi Müzik ve Sahne Sanatları Fakültesi (MSSF) to test our application and get feedback from people may be difficult. Bilkent Odeon and MSSF have sophisticated designs from an architectural point of view.

Third, designing an editor for matching seats in the 3D model with the tickets can be challenging. We plan to use Unity to develop the editor and then export it to WebGL; however, there can be issues such as a degradation in performance.

Lastly, inserting animations that describe the stage and sound effects in 3D models can be challenging. Animations can decrease the performance when a user is roaming in the 3D model. Also, designing attractive and interesting animations may be difficult.

Risk	Likelihood	Possible Effects	Alternative
Unable to find 3D models	Low	It will be hard to test the editor and present our application	Create 3D models or use 2D models
Unable to create 3D models (Odeon or MSSF)	High	Getting feedback from testers will not be possible	Find 3D models or use 2D models
Unable to design an editor for matching seats with tickets	Low	Showing perspective from each seat will not be possible	Showing 3D perspective of seats will be removed. Users can roam in the 3D model to look at the perspectives themselves.
Unable to insert animations into 3D models	Low	The application will be less attractive	Users can look at memories shared by customers

Table 3 - Risk Analysis

4.3. Project Plan

Table 4 provides the workpackages for the software development life cycle of our application.

WP#	Work Package Title	Leader	Members Involved
WP1	Creating Project Website	Ege Kaan Gürkan	Mehmet Çalışkan
WP2	Specifications Report	Mustafa Göktan Güdükbay	Everyone else
WP3	Analysis Report	Doğa Tansel	Everyone else
WP4	High Level Design Report	Ege Kaan Gürkan	Everyone else
WP5	Low Level Design Report	Mehmet Çalışkan	Everyone else
WP6	Final Report	Doğa Tansel	Everyone else
WP7	Presentations & Demonstration	Doğa Tansel	Everyone else
WP8	Back-end Development	Ege Kaan Gürkan	Mehmet Çalışkan, Musa Ege Ünalın, Mustafa Göktan Güdükbay
WP9	Front-end Development	Mehmet Çalışkan	Musa Ege Ünalın, Ege Kaan Gürkan Doğa Tansel
WP10	Editor for Matching Seats	Mustafa Göktan Güdükbay	Musa Ege Ünalın
WP11	Finding and Creating 3D Models	Musa Ege Ünalın	Ege Kaan Gürkan, Mustafa Göktan Güdükbay, Doğa Tansel
WP12	Testing	Musa Ege Ünalın	Mehmet Çalışkan, Doğa Tansel

Table 4 - Work Packages

WP 1: Creating Project Website			
Start date: 4 October 2021 End date: 11 October 2021			
Leader:	Ege Kaan Gürkan	Members involved:	Mehmet Çalışkan
Objectives: An introductory website is created for the project to host the reports.			
Tasks: Task 1.1. Designing the website Task 1.2. Hosting the website			
Deliverables: D1.1: The project website			

WP 2: Project Specifications Report			
Start date: 1 October 2021 End date: 11 October 2021			
Leader:	Mustafa Göktan Güdükbay	Members involved:	Everyone else
Objectives: Initial requirements of the project will be defined.			
Tasks: Task 2.1. Introduction Task 2.1.1 Description Task 2.1.2 Constraints Task 2.1.3 Professional and Ethical Issues Task 2.2. Requirements Task 2.3. References			
Deliverables: D2.1: The Project Specifications Report			

WP 3: Project Analysis Report			
Start date: 20 October 2021 End date: 15 November 2021			
Leader:	Doğa Tansel	Members involved:	Everyone else
Objectives: An extensive analysis of the project will be done.			
Tasks: Task 3.1. Introduction Task 3.2. Current System (if any) Task 3.3. Proposed System Task 3.3.1 Overview Task 3.3.2 Functional Requirements Task 3.3.3 Nonfunctional Requirements Task 3.3.4 Pseudo Requirements Task 3.3.5. System Models Task 3.3.5.1 Scenarios Task 3.3.5.2 Use Case Model Task 3.3.5.3 Object and Class Model Task 3.3.5.4 Dynamic Models Task 3.3.5.5 User Interface - Navigational Paths and Screen Mock-ups Task 3.4. Other Analysis Elements Task 3.4.1 Consideration of Various Factors in Engineering Design Task 3.4.2 Risks and Alternatives Task 3.4.3 Project Plan Task 3.4.4 Ensuring Proper Teamwork Task 3.4.5 Ethics and Professional Responsibilities Task 3.4.6 Planning for New Knowledge and Learning Strategies Task 3.5. Glossary Task 3.6. References			
Deliverables: D3.1: The Project Analysis Report			

WP 4: High-Level Design Report			
Start date: 1 December 2021 End date: 24 December 2021			
Leader:	Ege Kaan Gürkan	Members involved:	Everyone else
Objectives: Analysis model will be transported into a system design model.			
Tasks: Task 4.1. Introduction Task 4.1.1 Purpose of the system Task 4.1.2 Design goals Task 4.1.3 Definitions, acronyms, and abbreviations Task 4.1.4 Overview Task 4.2. Current software architecture (if any) Task 4.3. Proposed software architecture Task 4.3.1 Overview Task 4.3.2 Subsystem decomposition Task 4.3.3 Hardware/software mapping Task 4.3.4 Persistent data management Task 4.3.5 Access control and security Task 4.3.6 Global software control Task 4.3.7 Boundary conditions Task 4.4. Subsystem services Task 4.5. Consideration of Various Factors in Engineering Design Task 4.6. Teamwork Details Task 4.6.1 Contributing and functioning effectively on the team Task 4.6.2 Helping creating a collaborative and inclusive environment Task 4.6.3 Taking lead role and sharing leadership on the team Task 4.7. Glossary Task 4.8. References			
Deliverables: D4.1: The Project High-Level Design Report			

WP 5: Low-Level Design Report			
Start date: 27 December 2021 End date: 20 February 2022			
Leader:	Mehmet Çalışkan	Members involved:	Everyone else
Objectives: High-level design report will be refined into a detailed low-level design document.			
Tasks: Task 5.1. Introduction Task 5.1.1 Object design trade-offs Task 5.1.2 Interface documentation guidelines Task 5.1.3 Engineering standards (e.g., UML and IEEE) Task 5.1.4 Definitions, acronyms, and abbreviations Task 5.2. Packages Task 5.3. Class Interfaces Task 5.4. Glossary Task 5.5. References			
Deliverables: D5.1: The Project Low-Level Design Report			

WP 6: Final Report			
Start date: 1 April 2022 End date: 10 May 2022			
Leader:	Doğa Tansel	Members involved:	Everyone else
Objectives: Final architecture and status of the project will be presented in a final report.			
Tasks: Task 6.1. Introduction Task 6.2. Requirements Details Task 6.3. Final Architecture and Design Details Task 6.4. Development/Implementation Details Task 6.5. Testing Details Task 6.6. Maintenance Plan and Details Task 6.7. Other Project Elements Task 6.7.1.Consideration of Various Factors in Engineering Design Task 6.7.2.Ethics and Professional Responsibilities Task 6.7.3.Judgements and Impacts to Various Contexts Task 6.7.4 Teamwork Details Task 6.7.4.1) Contributing and functioning effectively on the team Task 6.7.4.2) Helping creating a collaborative and inclusive environment Task 6.7.4.3) Taking lead role and sharing leadership on the team Task 6.7.4.4) Meeting objectives Task 6.7.5 New Knowledge Acquired and Applied Task 6.8. Conclusion and Future Work Task 6.9. Glossary Task 6.10. References			
Deliverables: D6.1: The Project Final Report			

WP 7: Presentation & Demonstration			
Start date: 5 May 2022 End date: 10 May 2022			
Leader:	<i>Doğa Tansel</i>	Members involved:	<i>Everyone else</i>
Objectives: <i>The final product will be presented and demonstrated.</i>			
Tasks: Task 7.1. Presentation Task 7.2. Demonstration			
Deliverables: D7.1: Presentation & Demonstration			

WP 8: Back-end Development			
Start date: 20 November 2021 End date: 5 January 2021			
Leader:	<i>Ege Kaan Gürkan</i>	Members involved:	<i>Mehmet Çalışkan, Musa Ege Ünalın, Mustafa Göktan Güdükbay</i>
Objectives: <i>Developing a server for handling client requests and database queries.</i>			
Tasks: Task 8.1. Amazon Web Service (AWS) connection Task 8.2. Database Connection and Methods For Handling Basic Queries Task 8.2. Access and Storage of 3D models in AWS Task 8.3. Handling Client Requests and Sending Responses			
Deliverables: D8.1: Back-end Service			

WP 9: Front-end Development			
Start date: 10 December 2021 End date: 25 January 2022			
Leader:	<i>Mehmet Çalışkan</i>	Members involved:	<i>Musa Ege Ünalan, Ege Kaan Gürkan, Doğa Tansel</i>
Objectives: <i>Developing the user-interface and web connections of the application.</i>			
Tasks: <i>Task 9.1. Creating the customer front-end</i> <i>Task 9.2. Creating the screen for showing 3D models to customers</i> <i>Task 9.3. Creating the venue owner front-end</i> <i>Task 9.4. Creating the screen for showing the 3D editor to venue owners</i>			
Deliverables: <i>D9.1: Application Website</i>			

WP 10: Editor for Matching Seats			
Start date: 10 December 2021 End date: 10 January 2022			
Leader:	<i>Mustafa Göktan Güdükbay</i>	Members involved:	<i>Musa Ege Ünalan</i>
Objectives: <i>Developing a 3D editor for matching the seats with tickets.</i>			
Tasks: <i>Task 10.1. Loading 3D models at runtime</i> <i>Task 10.2. Adding collider objects to locate seats</i> <i>Task 10.3. Storing the location of each seat and the matched ticket</i> <i>Task 10.4. Exporting to WebGL</i>			
Deliverables: <i>D10.1: Editor for matching the seats in the 3D model with tickets</i>			

WP 11: Finding and Creating 3D Models			
Start date: 3 December 2022 End date: 29 February 2022			
Leader:	<i>Musa Ege Ünalan</i>	Members involved:	<i>Ege Kaan Gürkan, Mustafa Göktan Güdükbay, Doğa Tansel</i>
Objectives: <i>Finding 3D models to test our application. Creating the 3D model of ODEON.</i>			
Tasks: Task 11.1. Finding 3D models from various sources in different formats Task 11.2. Creating the 3D model of ODEON using Blender			
Deliverables: D11.1: 3D models of venues D11.2: 3D model of ODEON			

WP 12: Testing			
Start date: 15 February 2022 End date: 30 March 2022			
Leader:	<i>Musa Ege Ünalan</i>	Members involved:	<i>Mehmet Çalışkan, Doğa Tansel</i>
Objectives: <i>Testing the application, finding errors, and fixing them</i>			
Tasks: Task 12.1. Testing the back-end Task 12.2. Testing the front-end Task 12.3. Testing the 3D editor Task 12.4. Creating events and getting real user feedback			
Deliverables: D12.1: <i>Feedbacks of the users</i>			

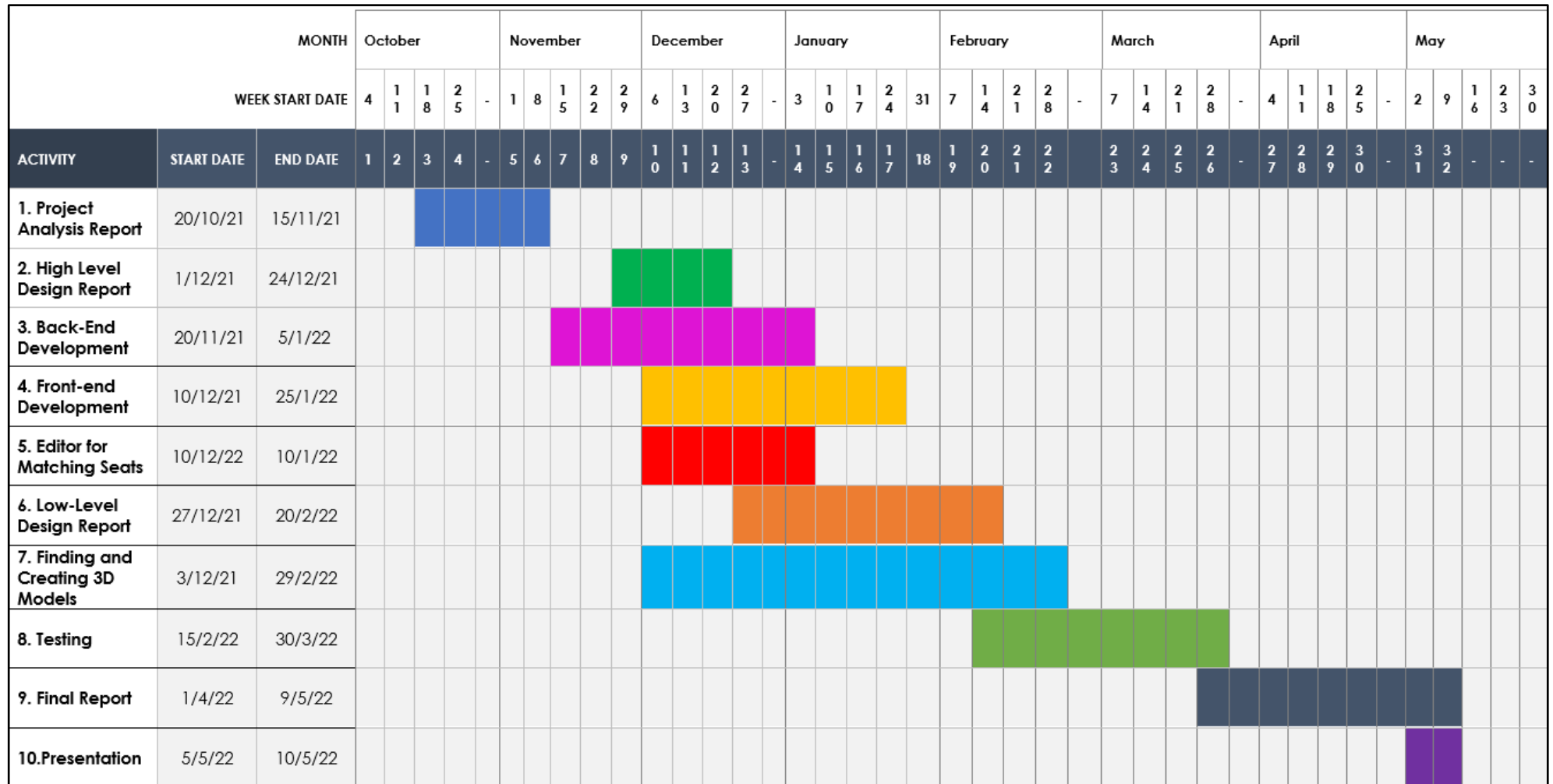


Figure 32 - Gantt Chart for Project Plan

4.4. Ensuring Proper Teamwork

Teamwork is an essential component of a reliable software product. To ensure that every team member properly contributes to the project, we will use sprints and GitHub issues. We plan a sprint to have a duration of 10 days. In a sprint, the following will happen:

- At the beginning of each sprint, we will determine the sub-components of each component and assign these issues to each team member. Besides, story points will be given to these issues so that every team member will have the same workload in each sprint.
- Every two days, we will conduct a meeting to discuss our progress and ask questions about the issues.
- At the end of the sprint, the pull requests will be opened for each issue, and these pull requests will be reviewed by another team member. If the reviewer approves the pull request, it will be merged into the master branch. Else, the pull request will be declined, the reviewer will list the problems in the pull request, and the author of the pull request will fix the pull request.

We also plan to use Slack as our main communication platform for the project. All the questions, screenshots and comments will be delivered through Slack. By doing these, we plan to have good teamwork in our project.

4.5. Ethics and Professional Responsibilities

We also plan to use Slack as our primary communication platform for the project. We will deliver all the questions, screenshots, and comments through Slack. By doing these, we plan to have good teamwork in our project

We will store the user passwords encrypted for ethical responsibilities and make our system as safe as possible. Also, we will not share our users' data with

anyone without their permission. We will also try to comply with the General Data Protection Regulation (GDPR).

4.6. Planning for New Knowledge and Learning Strategies

For the development of the project, we will need to know the following technologies and concepts:

- JavaScript
- React
- Unity
- Python
- Django
- PostgreSQL
- Blender
- Computer Networks
- Computer Graphics

The main learning strategy of our team will be Online Learning. We plan to watch tutorials and courses from platforms like YouTube, Udemy, and Coursera.

5. Glossary

Unity: It is a cross-platform game engine [9].

WebGL: It is a cross-platform, web standard for a 3D graphics API based on OpenGL ES (Embedded Systems) used with HTML5 Canvas element supported by browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge, Safari, Opera [10].

Django: Django is a high-level Python web framework that can be used for both backend and frontend development [17].

React: It is a Javascript library for producing user interfaces on the web [18].

PostgreSQL is an open source object-relational database system [19].

6. References

- [1] “Etkinlik Biletleri | Konser, Spor, Sanat, Tiyatro, Müze | Resmi Biletix sitesi ,” *Biletix*. [Online]. Available: <https://www.biletix.com/anasayfa/TURKIYE/tr>. [Accessed: 13-Nov-2021].
- [2] “Ticketmaster: Buy verified tickets for concerts, sports ...,” *Ticketmaster*. [Online]. Available: <https://www.ticketmaster.com/>. [Accessed: 13-Nov-2021].
- [3] “Tools for communities and events,” *Peatix*. [Online]. Available: <https://peatix.com/>. [Accessed: 13-Nov-2021].
- [4] Plandy, “Senin Etkinliklerin!,” *Plandy*. [Online]. Available: <https://www.plandy.me/>. [Accessed: 13-Nov-2021].
- [5] “Biletino · Etkinlik KEŞFET veya etkinlik oluşturup bilet sat,” *biletino*. [Online]. Available: <https://biletino.com/tr/>. [Accessed: 13-Nov-2021].
- [6] “Maç biletleri, Konser biletleri, Tiyatro Biletleri,” *Passo*. [Online]. Available: <https://www.passo.com.tr/tr>. [Accessed: 13-Nov-2021].
- [7] *Milyon Bilet*. [Online]. Available: <https://milyonbilet.com/>. [Accessed: 13-Nov-2021].
- [8] “Bilkent Senfoni Orkestrası,” *BSO*. [Online]. Available: <http://bso.bilkent.edu.tr/tr/>. [Accessed: 13-Nov-2021].
- [9] Unity Technologies, *Unity*. [Online]. Available: <https://unity.com/>. [Accessed: 13-Nov-2021].
- [10] “WebGL,” *The Khronos Group*, 19-Jul-2011. [Online]. Available: <https://www.khronos.org/webgl/>. [Accessed: 13-Nov-2021].
- [11] “How to publish for WebGL,” *Unity Learn*. [Online]. Available: <https://learn.unity.com/tutorial/how-to-publish-for-webgl>. [Accessed: 13-Nov-2021].

- [12] “Where the world builds software,” *GitHub*. [Online]. Available: <https://github.com/>. [Accessed: 13-Nov-2021].
- [13] *Git*. [Online]. Available: <https://git-scm.com/>. [Accessed: 13-Nov-2021].
- [14] “Trello helps teams move work forward.,” *Trello*. [Online]. Available: <https://trello.com/>. [Accessed: 13-Nov-2021].
- [15] *3D Warehouse*. [Online]. Available: <https://3dwarehouse.sketchup.com/>. [Accessed: 13-Nov-2021].
- [16] “The best assets for game making,” *Unity Asset Store*. [Online]. Available: <https://assetstore.unity.com/>. [Accessed: 13-Nov-2021].
- [17] *Django*. [Online]. Available: <https://www.djangoproject.com/>. [Accessed: 13-Nov-2021].
- [18] “React – a JavaScript library for building user interfaces,” [Online]. Available: <https://reactjs.org/>. [Accessed: 13-Nov-2021].
- [19] The PostgreSQL Global Development Group, *PostgreSQL*, 13-Nov-2021. [Online]. Available: <https://www.postgresql.org/>. [Accessed: 13-Nov-2021].