**Requirements Analysis Document**

*Application:* **Theaception**

Date: 24/02/2021

**Index**

[1. Introduction 3](#_heading=h.1fob9te)

[1.1 Purpose of the system 3](#_heading=h.3znysh7)

[1.2 Scope of the system 3](#_heading=h.2et92p0)

[1.3 Objectives and success criteria of the project 3](#_heading=h.tyjcwt)

[1.4 Definitions, Acronyms, and abbreviations](#_heading=h.3dy6vkm) 4

[2. System Description](#_heading=h.1t3h5sf) 6

[2.1 Functional Requirements](#_heading=h.1t3h5sf) 6

[2.2 Non-functional Requirements](#_heading=h.4d34og8) 7

[3. Use Cases](#_heading=h.2s8eyo1) 8

[3.1 Use Case diagram](#_heading=h.17dp8vu) 8

[3.2 Use case descriptions](#_heading=h.3rdcrjn) 9

[3.2.1 Use Case 1: Buy a ticket](#_heading=h.26in1rg) 9

[3.2.2 Use Case 2: Create an event](#_heading=h.26in1rg) 11

[3.2.3 Use Case 3: Arrange the theater](#_heading=h.26in1rg) 12

[4. Mockups](#_heading=h.lnxbz9) 14

**1. Introduction**

**1.1 Purpose of the system**

The system must provide a way to make an easier interaction between a theater and the people, it allows for the users to see the upcoming events that are held in this specific theater as well as purchasing or booking tickets. It will provide the necessary tools for the administrator to manage the events and provide an easier and fluid interaction between the theater and the customers.

**1.2 Scope of the system**

The platform allows the manager to create events, establish prices, set the capacity of the event, create and modify different zones in the theater and its corresponding price. The manager can also check the statistics of each event (earnings, occupancy) and an organized view of this statistics grouped by the type of event (music, dance, theater). The manager has to log in the app with a predefined username and password, this will allow him to access the app with administrator permits and tools such as creating events .

As a customer you can check the events but if you want to buy or book you need to log in or register in the application with a username and a password. The user can book, buy tickets and also can buy cycle or annual passess.

The cycle passes allow the customer to attend one performance of each event in a specific cycle. The annual pass on the other hand allows the customer to assist one performance of all the events in the theater during a whole year. To buy a ticket or pass, the customer has to insert his credit card number and in case of the tickets no refunds are allowed unless the event is cancelled in which case the app will notify the user that he can get a refund by going to the physical offices of the theater.

**1.3 Objectives and success criteria of the project**

* User is able to book, purchase and download a ticket
* Manager is able to create, modify and cancel events
* User may be able to see the upcoming events
* User may be able to create an account and log in
* Application must run on all OS possible
* Manager must be able to rearrange zones and seats
* Manager must be able to see the statistics of past events
* User must be able to purchase a pass
* User must be able to pay either by using its pass or a credit card
* Users must be on a waitlist if there are no tickets left.
* Manager can disable and enable seats
* User can search events by type

**1.4 Definitions, Acronyms, and abbreviations**

|  |  |
| --- | --- |
| Theater | *“A building or outdoor area in which plays and other dramatic performances are given.”* It has different performances of events. |
| Event | *“An* ***event*** *includes the presentation of a performance and the attention of an audience.”* Each event has a type (concert, theater act or dance), duration, title, description, author and director. |
| Cycle | A group of events. It can also contain other cycles inside. |
| Administrator | A user that can manage cycles, events and performances. |
| Occupancy | Event statistic about how full is an event, a performance or a sitting zone. |
| Revenue | Event statistic about how profitable is an event, a performance or a sitting zone. |
| Sitting Zone | Zone of seats of a theater. It can be a standing or a sitting zone, and it has a defined number of seats and a defined number of columns/rows. It can have zones inside. |
| User | Person that uses the application. If it is not logged it would be considered anonymous. When logged, each user can be an administrator or a customer. A logged user has a username and an encrypted password. |
| Ticket | A document showing that the holder is admitted to enter the performance. It has the zone, zone type, seat number, event name and performance |
| TTB | Time To Book. Time needed to be since the booking moment to the performance moment to be able to book. |
| TTP | Time To Pay. Time given to a customer to pay a booked ticket since it was originally booked. |
| Pass | A document that lets the user enter one performance of each event. The pass can be an Annual Pass (one performance of each event of the year) or a Cycle Pass (one performance of each event of the cycle) |
| OS | *“An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs.”* |
| Waitlist | List of users waiting to buy a ticket. |

.

**2. System Description**

**2.1 Functional Requirements**

2.1.1 Anonymous User

  2.1.1.1 . Search events

  2.1.1.2 . Log in

  2.1.1.3 . Register

2.1.2 Registered User

  2.1.2.1 . Book seat

  2.1.2.2 . Purchase seat

  2.1.2.3 . Search events

  2.1.2.4 . Choose seat

  2.1.2.5 . Purchase annual pass

  2.1.2.6 . Purchase cycle pass

  2.1.2.7 . pay reservations

  2.1.2.8 . See tickets

  2.1.2.9 . Apply to a waitlist

cancel performance

2.1.3 Admin User

  2.1.3.1 . Log in

  2.1.3.2 . Create event

Create performance

  2.1.3.3 . Postpone performance

  2.1.3.4 . Cancel event

  2.1.3.5 . Set zones

  2.1.3.5 . Set event prices

  2.1.3.6 . Set event capacity

  2.1.3.11 . Create cycles

2.1.3.12 . Enable and disable seats

  2.1.3.7 . Limit booking time

  2.1.3.8 . Limit payment time after booking

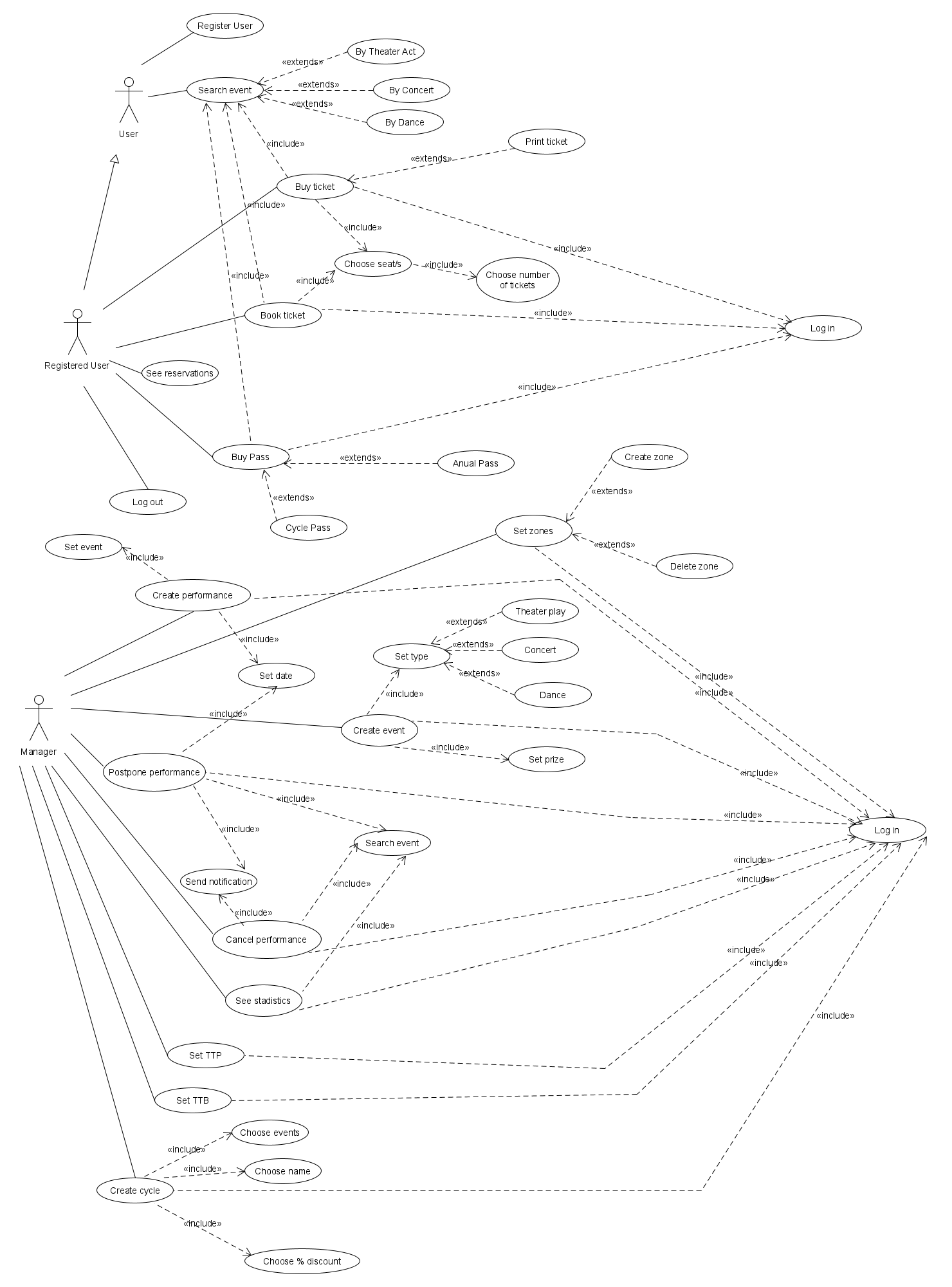
  2.1.3.9 . See occupancy statistics

  2.1.3.10 . See revenue statistics

**2.2 Non-functional Requirements**

* Passwords must be cyfrated (Security)
* App must work on every OS possible (Usability)
* Generated tickets must have an unique code that must be cyfred (Security)
* A database where the users are stored (Operational)
* A database with the events and its details (Operational)
* Visible and easy to use interface (Usability)

**3. Use Cases**

**3.1 Use Case diagram**

**3.2 Use case descriptions**

**3.2.1 Use Case 1: Buy a ticket**

Primary Actor : Registered User.

Stakeholders and Goals:

Registered User: buy a ticket for a chosen play.

Preconditions:

The user has selected a play by searching it after having logged in.

Success guarantee (Post-conditions):

The user buys a ticket for the selected play, and a pdf with all the information and an encrypted and unique code for each ticket. The capacity of the theater gets reduced by the number of tickets bought and the seats selected are now occupied so they cannot be selected for other users.

Main Success Scenario:

1. The user selects the date for the performance.
2. The user selects “purchase ticket”.
3. The user chooses a standing zone for the performance.
4. The user chooses the number of tickets to buy.
5. The user selects to buy the tickets by credit card (by pass)
6. The user enters the credit card number and the CCV.
7. The system connects with the bank account and proceeds with the payment.
8. The system redirects the user to the “manage tickets” window where he can download the pdf.

Extensions (Alternative paths):

3a The user selects a seating zone for the play.

3a.1. The user selects the seats for the play.

3a.2. The user selects to buy the tickets by credit card.

3a.3. The user enters the credit card number and the CCV.

3a.4. The system connects with the bank account and proceeds with the payment.

3a.5. The system redirects the user to the “manage tickets” window where he can download the pdf.

3a.2b The user selects to buy the tickets by pass.

3a.2b.1 The user selects the pass.

3a.3b.2 The system redirects the user to the “manage tickets” window where he can download the pdf.

3b The user selects a composed zone.

3b.a.1. The user selects the standing subzone.

3b.a.2. The user selects the number of tickets to buy.

3b.a.3. The user selects to buy the tickets by credit card.

3b.a.4.The user enters the credit card number and the CCV.

3b.a.5.The system connects with the bank account and proceeds with the payment.

3b.a.6.The system redirects the user to the “manage tickets” window where he can download the pdf.

3b.a.3a The user selects to buy the tickets by pass.

3b.a.3a.1 The user selects the pass.

3b.a.3a.2 The system redirects the user to the “manage tickets” window where he can download the pdf.

3b.b.1.The user selects the seating subzone

3b.b.2.The user selects the seats for the play.

3b.b.3.The user selects to buy the tickets by credit card

3b.b.4.The user enters the credit card number and the CCV.

3b.b.5.The system connects with the bank account and proceeds with the payment.

3b.b.6.The system redirects the user to the “manage tickets” window where he can download the pdf.

3b.b.3a The user selects to buy the tickets by pass.

3b.b.3a.1 The user selects the pass.

3b.b.3a.2 The system redirects the user to the “manage tickets” window where he can download the pdf.

6a The user selects to buy the tickets by pass.

6a.1 The user selects the pass.

6a.2 The system redirects the user to the “manage tickets” window where he can download the pdf.

Special Requirements:

A database to store all the information of the event and the available tickets.

A database to store the information of the users that have bought a ticket.

Technology and Data Variations List:

Graphical interface.

Possibility of downloading a pdf that contains the ticket bought.

Connection with a bank website to process the payments.

Frequency:

For this application we don’t have to take into account the possibility of different users trying to access .

Open Issues:

Can a seating subzone have more subzones inside of it? Should it change the price of the ticket?

Can a standing zone have subzones inside of it? Should it change the price?

**3.2.2 Use Case 2: Create an event**

Primary Actor : Administrator

Stakeholders and Goals:

Administrator to create an event with all the necessary data.

Preconditions:

The user has logged in as administrator and has a new event to add.

Success guarantee (Post-conditions):

The new event is added and ready for the users to find.

Main Success Scenario:

1. The manager selects the “create event” option.
2. The manager sets a director.
3. The manager sets an author.
4. The manager sets a title.
5. The manager sets a description.
6. The manager sets the price of each zone.
7. The manager sets the theater act as the event type.
8. The manager sets the actors of the theater act.
9. The manager finishes the event creation and the system adds the new event to the database.

Extensions (Alternative paths):

7a The manager sets “concert” as the event type.

7a.1. The manager sets the music pieces.

7a.2. The manager sets the orchestra name.

7a.3. The manager finishes the event creation and the system adds the new event to the database.

7b. The manager sets “dance” as the event type.

7b.2.The manager sets a dancer.

7b.3. The manager sets an orchestra.

7b.4. The manager sets a conductor.

7b.5. The manager finishes the event creation and the system adds the new event to the database.

Special Requirements:

A database for all the events.

A database of the users for login.

Technology and Data Variations List:

Graphical interface.

Frequency:

For this application we don’t have to take into account the possibility of different users trying to access because there is only one unique manager.

Open Issues:

The type of events are fixed to the existing ones. There is no way of creating a new event type.

**3.2.3 Use Case: Arrange the theater**

Primary Actor : Manager

Stakeholders and Goals:

Manager: modify all the structure of the theater, create and delete the zones, enable and disable seats, determine the percentage of occupancy of the theater, the times to book and to pay, and the price of the annual pass.

Preconditions:

The manager has logged in and clicked the “manage” button.

(Is the first time that the manager arrange the theater, so it is necessary to set everything from the beginning)

Success guarantee (Post-conditions):

The structure of the theater changes. Zones are created, the times to book and to pay are defined and applied to the next events. The price for the annual pass is setted.

Main Success Scenario:

1. The manager selects a seating zone type.
2. The manager chooses an arbitrary name for the zone.
3. The manager set the number of seats per row.
4. The manager set the number of rows.
5. If the manager agrees with that zone, he presses the add zone button.
6. If the manager wants to create more zones, he has to do the same process from step 1.
7. The manager does not disable any chair for maintenance.The manager sets the a 100% of the occupancy.
8. The manager sets the TTB.
9. The manager sets the TTP.
10. The manager selects a price for the annual pass on each zone that he has created.
11. Once the manager has finished editing the theater, the changes are saved automatically, so he has to press the application logo.
12. The system redirects the manager to the main page.

Extensions (Alternative paths):

7a The manager wants to disable some seats because of the maintenance.

7a.1. The manager selects the zone by name.

7a.2. The manager selects the seat number that he wants to disable.

7a.3. The manager selects the maintenance status.

7a.4. The manager writes the date when the maintenance is over.

7a.5. The manager writes the reason for the maintenance of the seat.

7a.6. The manager clicks on the submit button to apply changes.

7a.7. If the manager wants to do maintenance in more seats, he has to do this process again.

7a.8.The manager sets the a 100% of the occupancy.

7a.9. The manager sets the TTB.

7a.10. The manager sets the TTP.

7a.11. The manager selects a price for the annual pass on each zone that he has created.

7a.12. Once the manager has finished editing the theater, the changes are saved automatically, so he has to press the application logo.

7a.13.The system redirects the manager to the main page.

8a

8a.1. The manager selects an occupancy percentage different form 100.

8a.2. The system automatically disables the percentage of seats corresponding to the selected value.

8a.3. The manager sets the TTB.

8a.4. The manager sets the TTP.

8a.5. The manager selects a price for the annual pass on each zone that he has created.

8a.6. Once the manager has finished editing the theater, the changes are saved automatically, so he has to press the application logo.

8a.7. The system redirects the manager to the main page.

Special Requirements:

A database to store all the information about the characteristics of the theater.

Technology and Data Variations List:

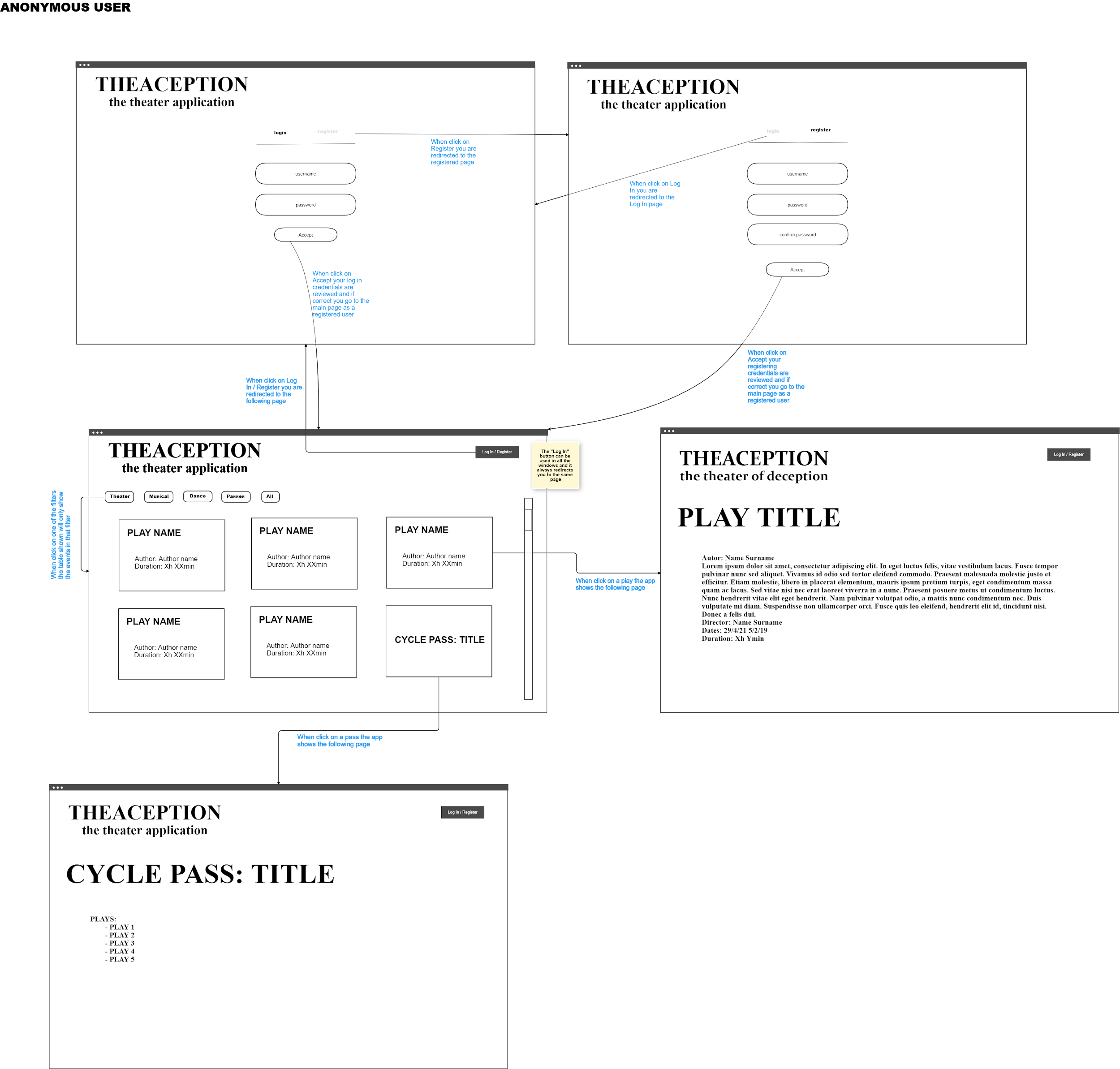
Graphical interface

Frequency:

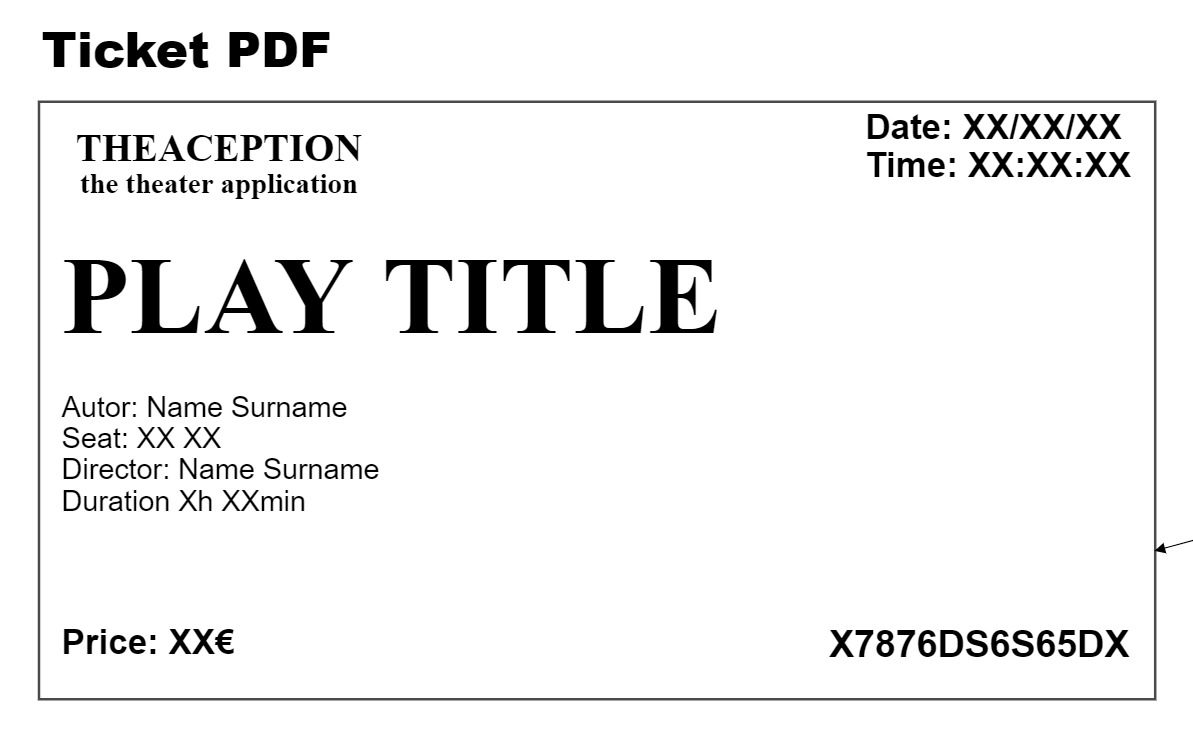
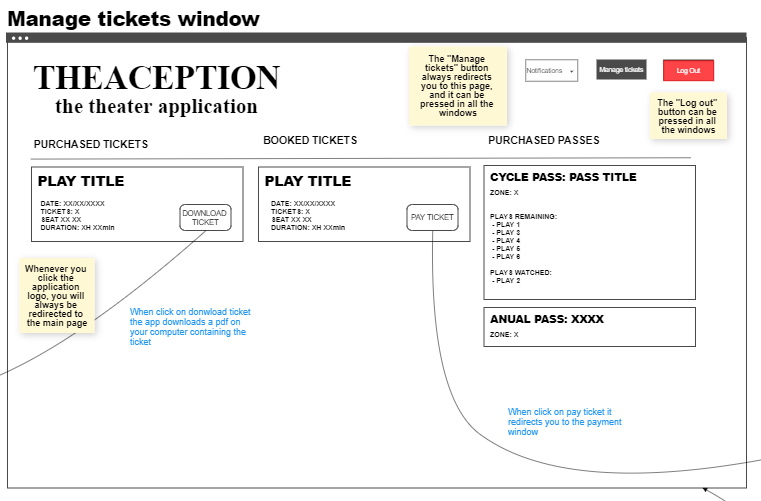
For this application we don’t have to take into account the possibility of different users trying to access because there is only one unique manager.

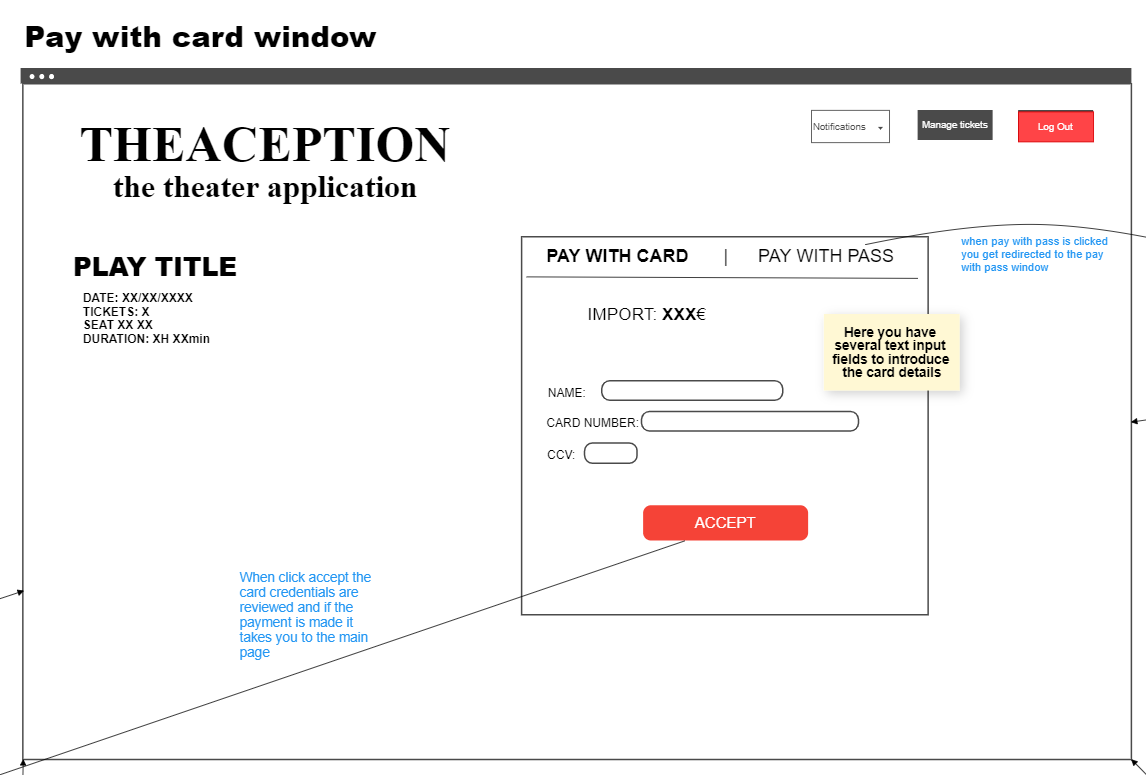
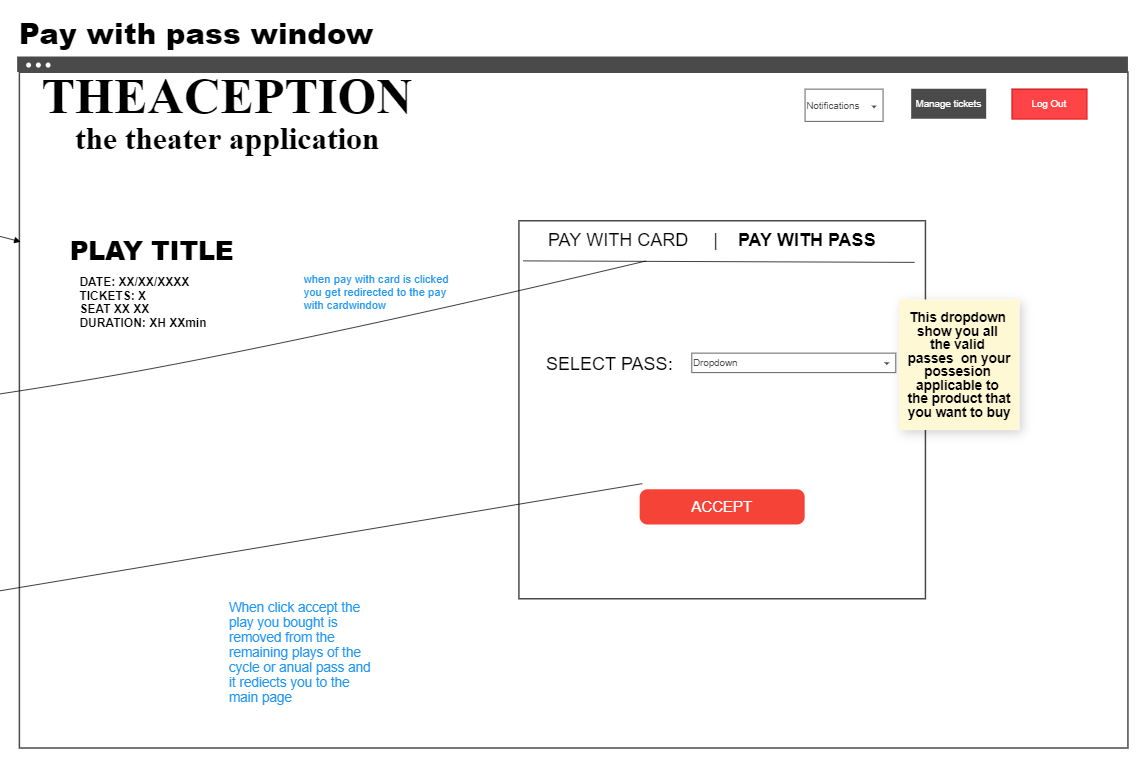
Open Issues:

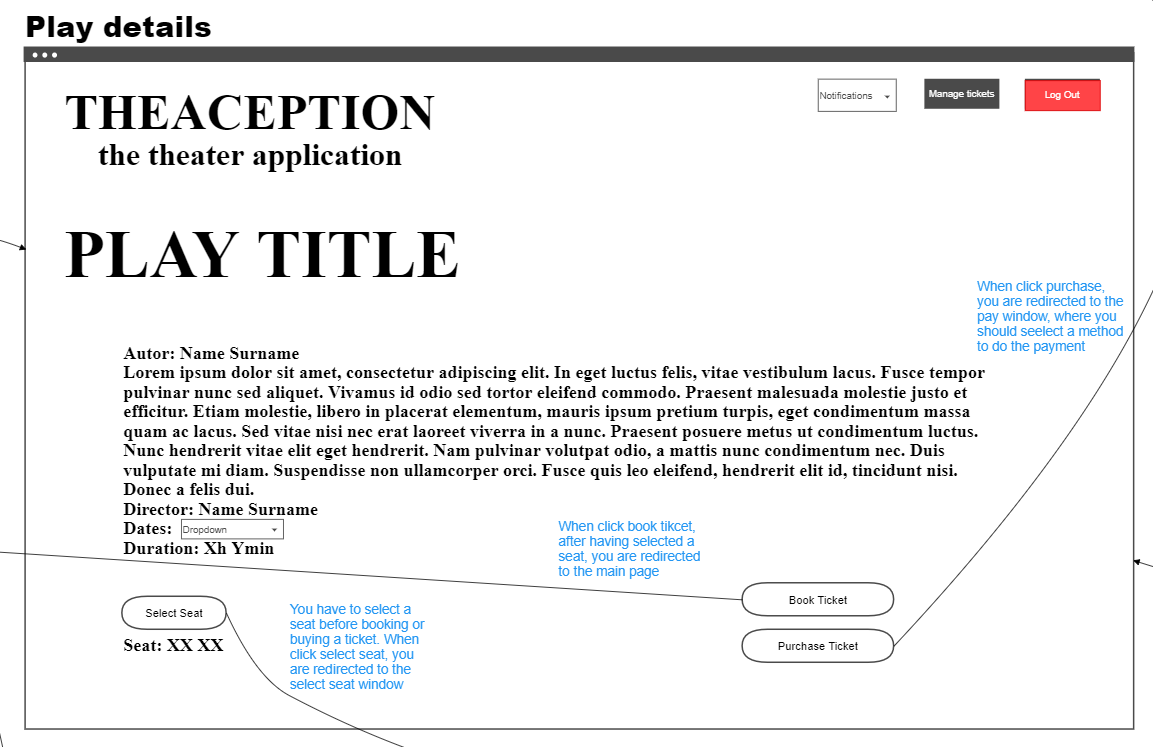
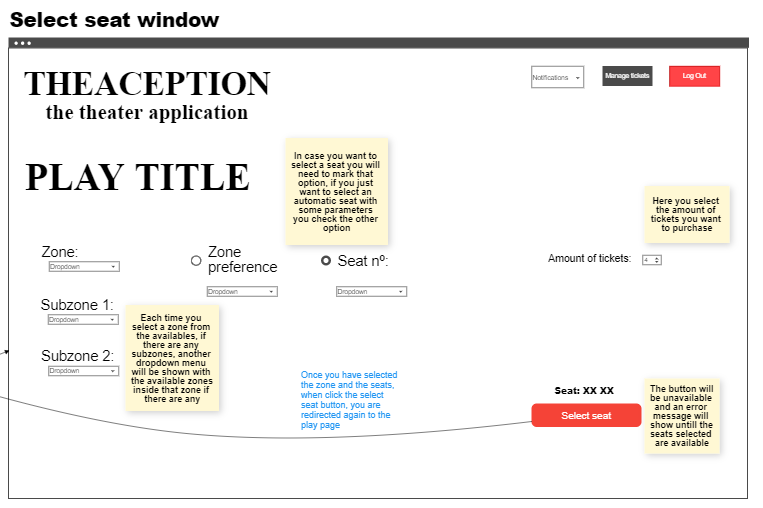
How many subzones inside subzones can be created?

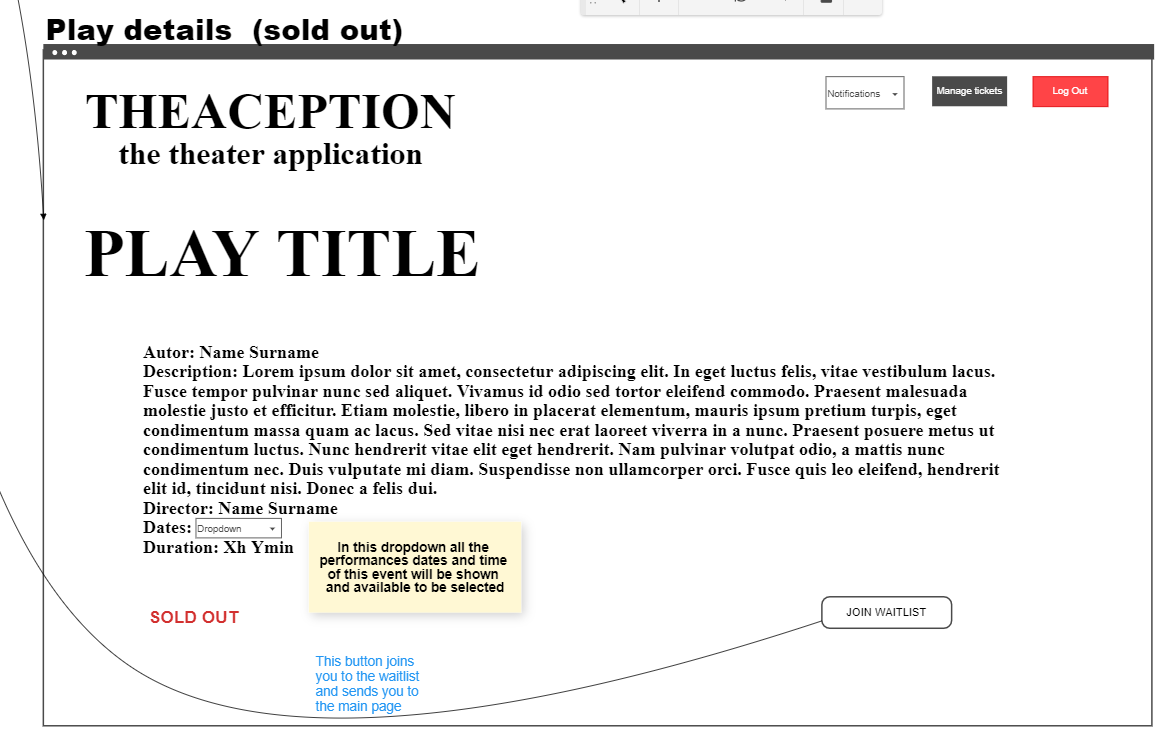
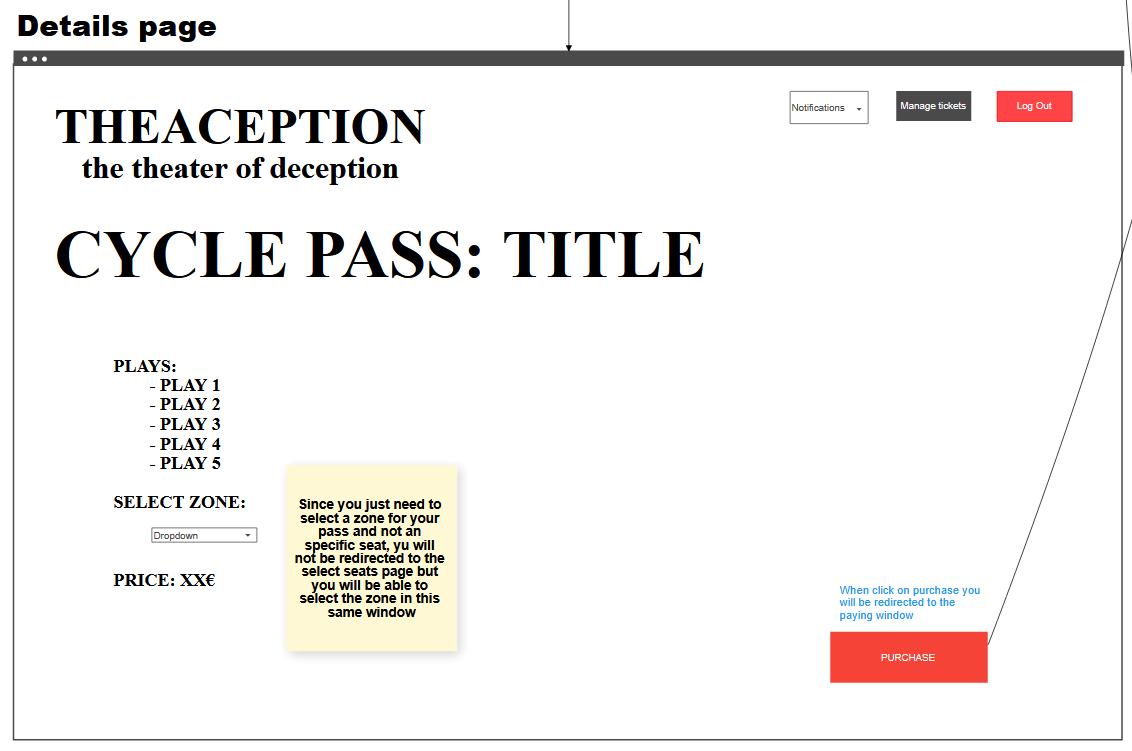
**4. Mockups**

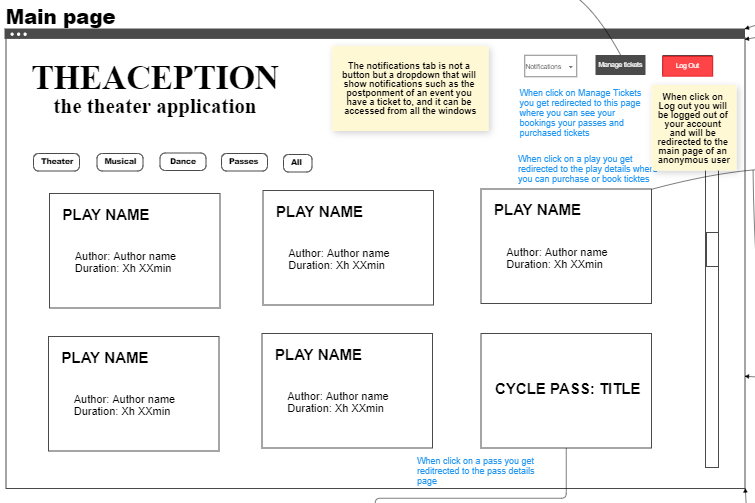
##### Registered User:



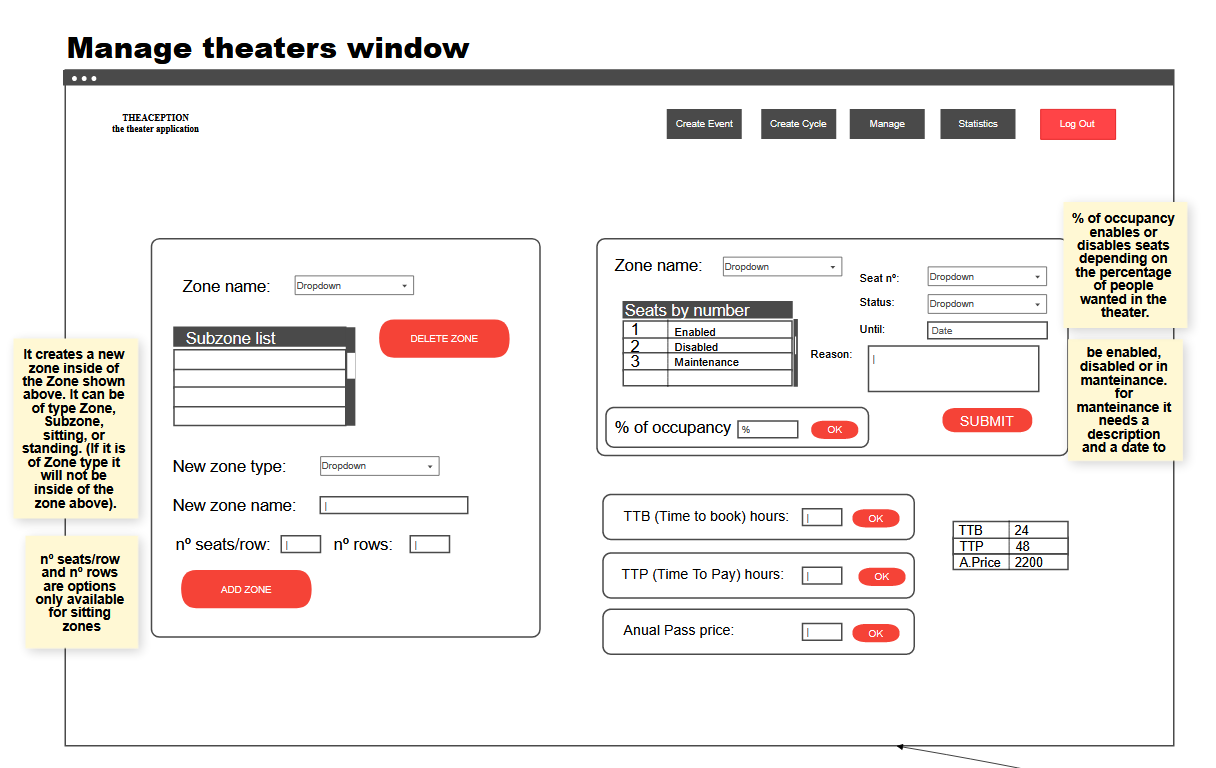


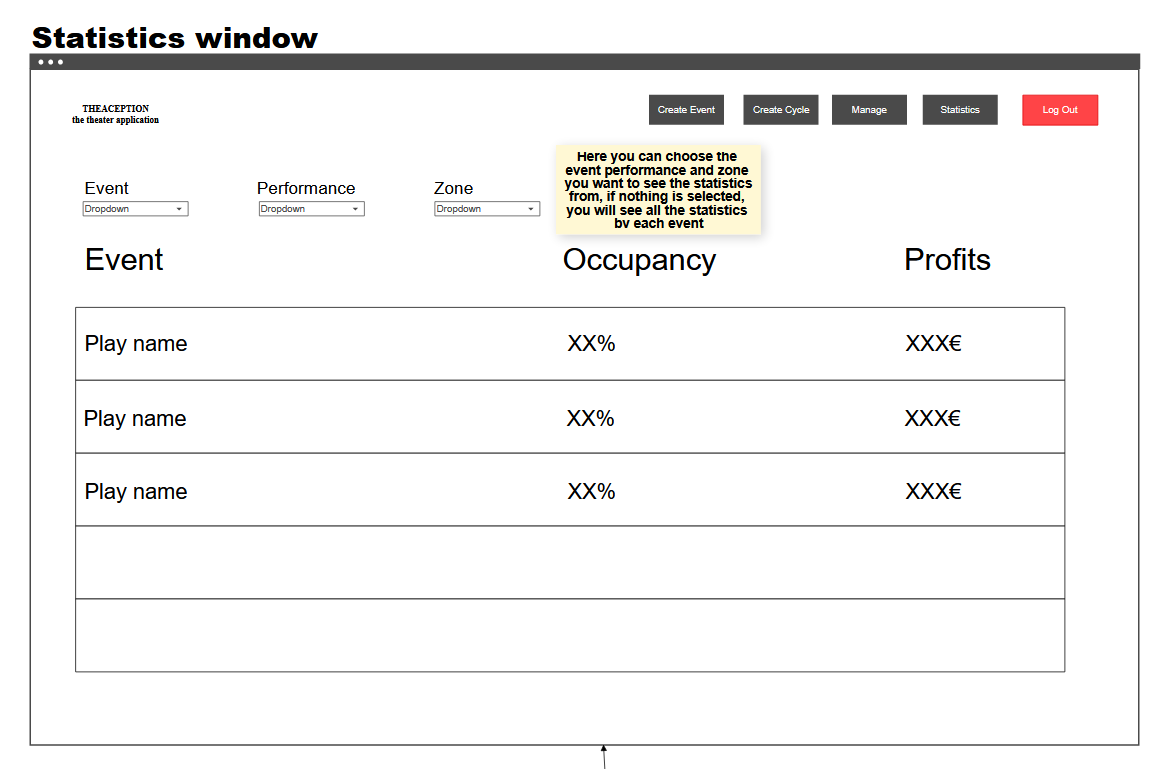


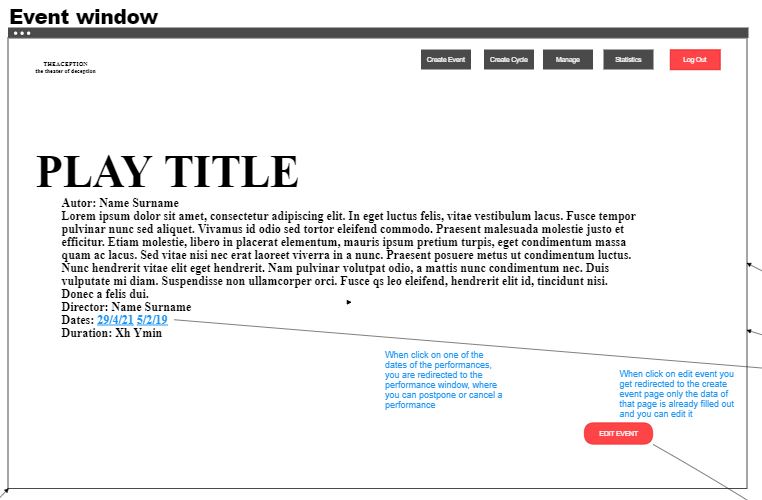


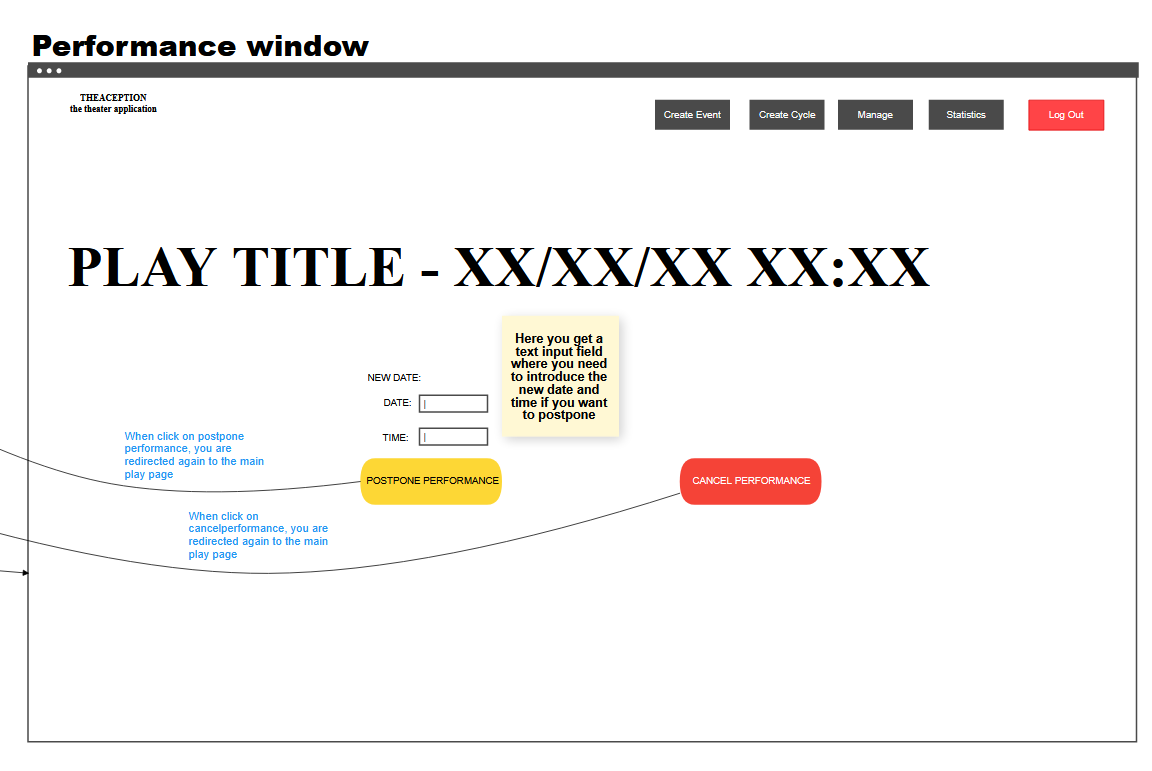


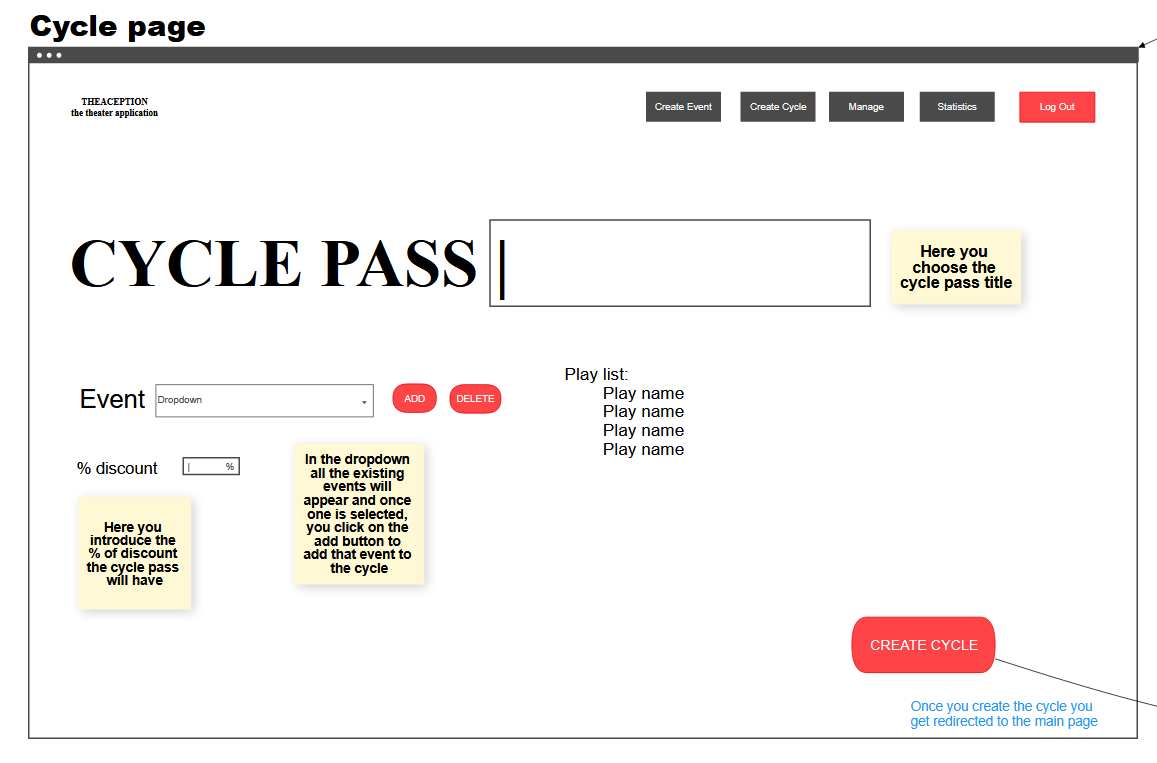
##### Manager:

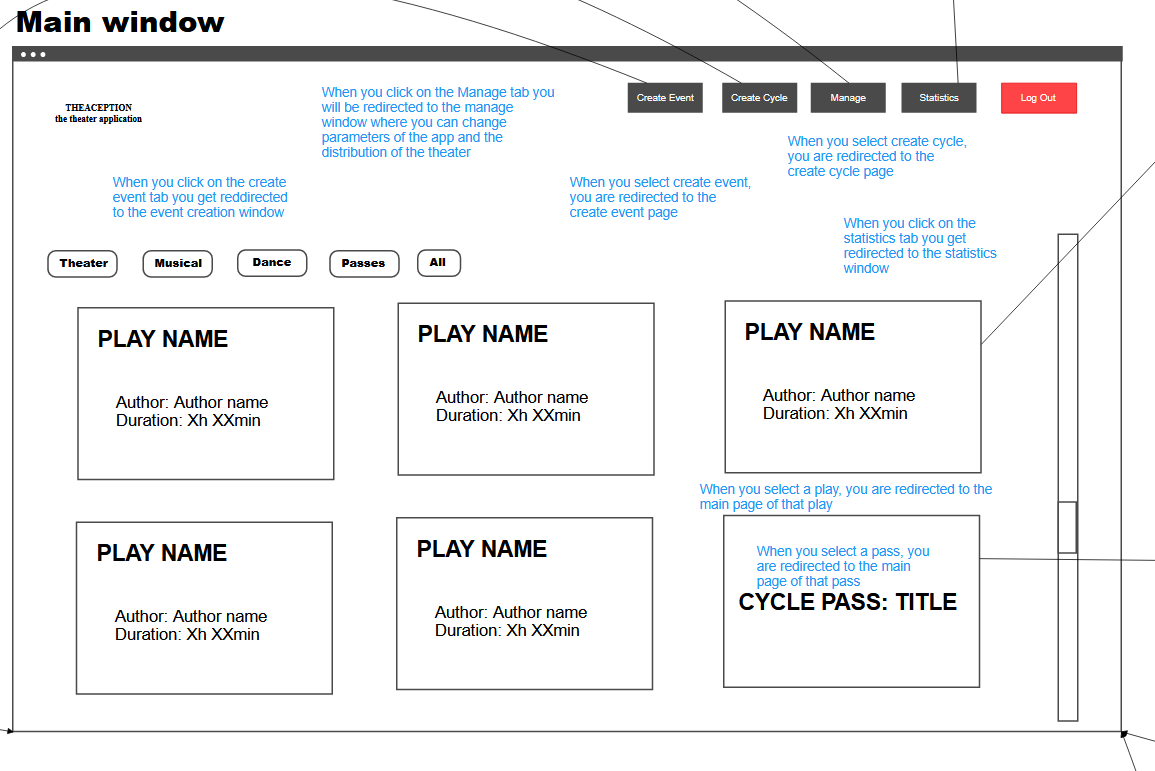


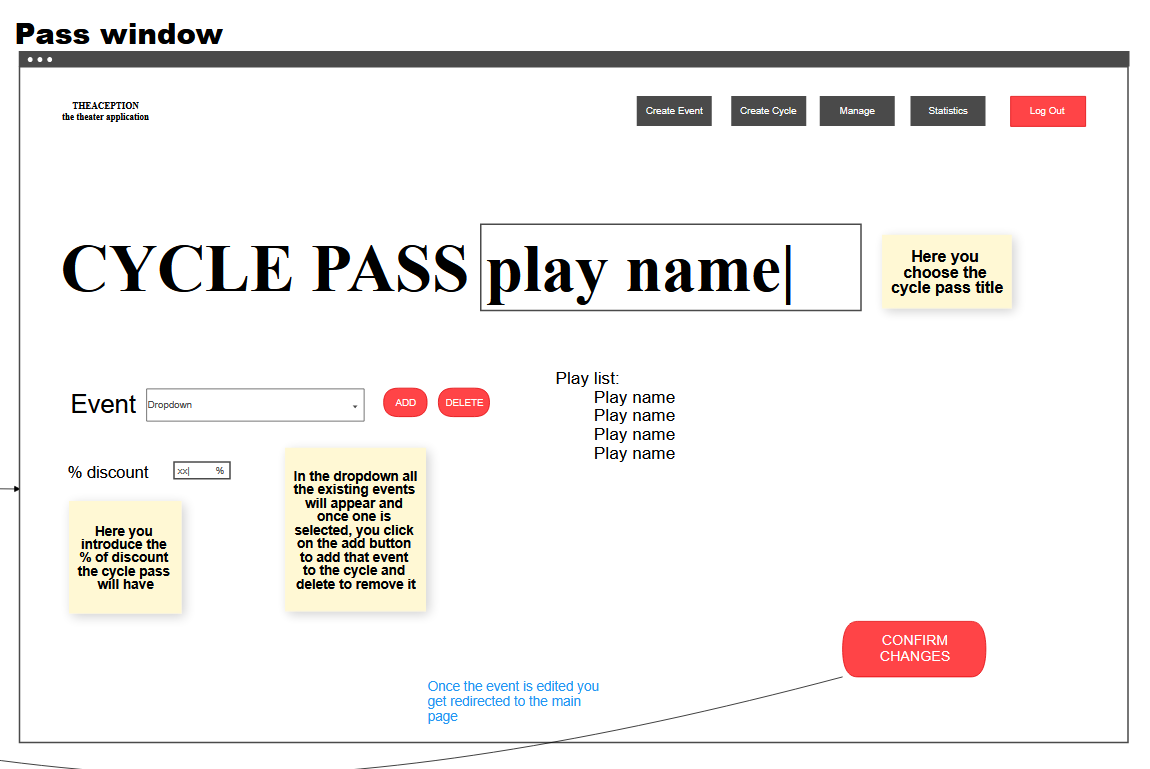


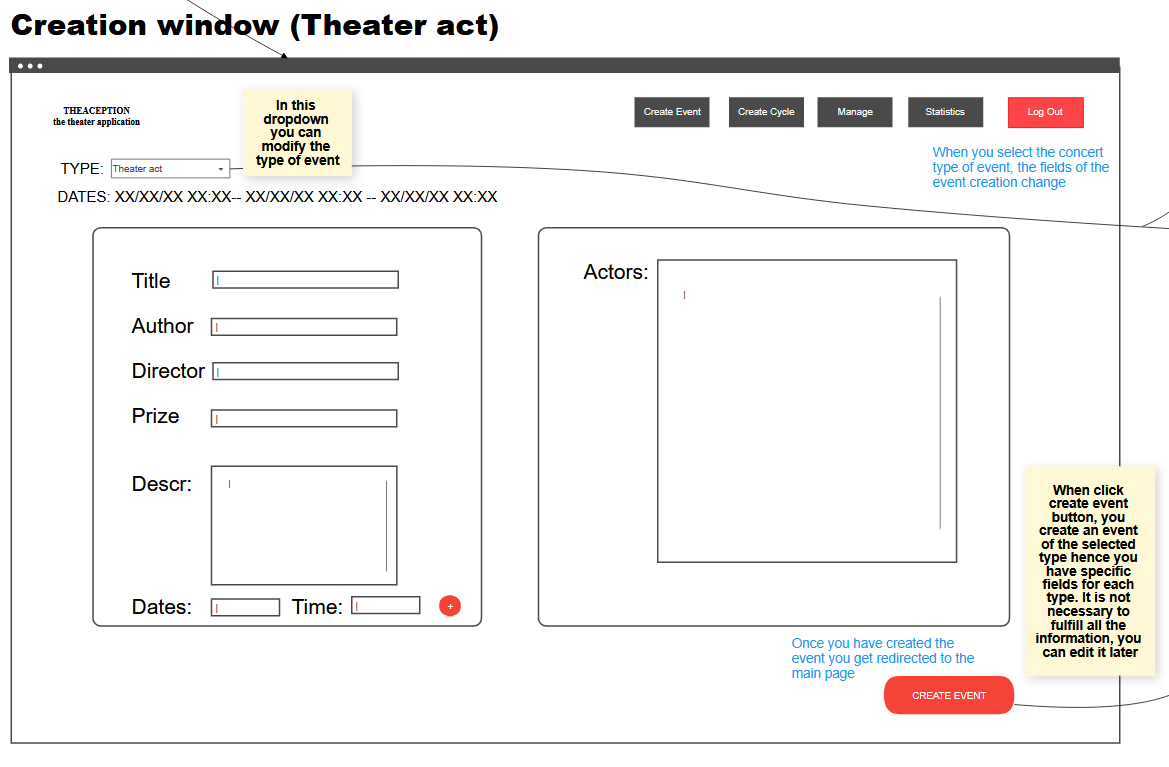












You can see the mockup’s full version in the following link: <https://cacoo.com/diagrams/otME6y9qi3koyv7j/2046B>