Functional requirements

1- Functional User requirements

1- The user can register with all his basic details.\*

2-user can log in if he has an email\*

3-user can search a movies by title,cast and genre\*

4- user can browse movies based on the ( where the cinema , the movie )\*

5- users can check the time of movie in any cinema.\*

6- user could discover all their movie information, with trailers, cast, synopsis, ratings, & more.

7- user can save movies to favorites list and book a ticket to any movie that he browse in website.\*

8-user can choose ticket type(Vip , normal).

9-user can pay by credit card or cash by promocode

10- user can pre-order the food & drinks before they arrive at the cinema

11- user can book his chair before going to the cinema

12- Admin must log in (name – password ) before do any thing.

13- admin able to ( add, edit, delete and categorized movies ).

14- admin mange tickets orders and offers.

15- admin should be able to mange the customer support(set times of cinema and movies).

16- admin have access to user authentication.

17-admin can view feedback from user

2- Functional system requirements

1- The user can register with ((first,last) Name – password – E-mail – Region - date of birth – (male,female))

2-system check if admin or user.

3-System check if user name and password is correct in database.

4- system display a list of available movies including info ( title – cast – rating – poster – synopsis - review – trailers )

5-system Allows users to search for a movie with their favorite actors, cinemas, & more and add it to his favourite list.

6- system allows user to book movies , choosing number of ticket and time,choosing seet and tim.

7-system allow to guest searching about movies based on categories

8-system allow to guest registering to be user if he want

9-system allow to guest browse as a user but he cannot book a tickets

10- system have e-ticket (QR code ) to order and save it in database.

11-Allows users to pre-order their food & drinks before they arrive at the cinema.

12- system allow user to give feedback and rate movies.

Non Functional requirements (first way)

1-Product Requirements :

1- System saves user’s time and help him to search about cinema and film by categories

1. If system down .. it will work again in 20 sec
2. The site can handle more than a thousand users at the same time
3. Organizational requirements

1- user shall enter his credit card to book cinema or film.

2- User shall log in first to book a film in cinema

3- user can log out of website and log in with another email

4- user is allowed to update or modify his details.

5-Website can be running in Linux .. windows 10

3-external requirements

1- Customer data cannot be leaked or stolen

Non Functional requirements (second way)

1- Look-and-Feel REQS : website should be beautiful .. The colors

are consistent and comfortable for eyes..

2- Usability & Humanity REQS : website should be simple ,doesn't need to training

3- Performance requirements : System shall show the product that user choose in 1 second ..

system should allow the user write his feed back ..

web site shall handle up to more than user simultaneously ..

web site shall on average, operate without failure for 5 days ..

4-Operational & Environmental REQS: website shall conserve data of internet ..

5- Maintainability & Support REQS : The admin shall modify the website every season

6- Cultural REQS : The language used shall be English only ..

The language used in the interface should be formal and polite

8- The site shall be immune to attacks and insured against theft of customer or user data..

**Use case description for user, admin**

**Actor: User**

**use case name: login**

**description: we use to check that the user have account in the system by login and the right account with using another use case it’s verify password that check if the password and the email is true and the extend use case that tell the user there’s error in the email or the password and to can treat with system as user actor: user, Database**

**function: login the app**

**input: name and password of user**

**source: database**

**output: successful login**

**destination: user**

**req: user has an account.**

**precondition: display the login buttons.**

**main sequence:**

**1. the user will write the email**

**2. the user will write the password**

**3. then will verify the email and the password by sending to the database**

**alternative sequence :**

1. **if the user write the wrong email or password will send message there’s something wrong**

**post condition : when the login is true it will open the cinema app.**

**use case name: registration**

**description : make account if the user don’t have account and make log in**

**actors : user , admin**

**function: create a new e-mail.**

**Input: all the details of user ((first, last) Name – password – E-mail – Region - date of birth – (male, female)).**

**Output: a new e-mail**

**Destination: user**

**precondition: display button create e-mail then the registration form**

**main sequence:**

**1. when the user doesn’t have account it will make registration**

**2. the user will see the registration form**

**3. the user will enter the ((first ,last) Name – password – E-mail – Region - date of birth – (male ,female)) and submit**

**4. then send the data to the Database**

**alternative sequence :**

1. **if user submit email & password that has been in database before it will show “the email has been create before”.**

**post condition : user can make login.**

**Use Case: Browse**

**Description: viewing a list of available movies to selecting a specific movie to view more details and if he logged In he can see the latest movie.**

**Function: show the movies.**

**Destination: user**

### **Preconditions**

* **The user has internet access.**
* **The Cinema app is opened, and the user is on the home screen.**

**main sequence:**

1. **The user selects a movie to view more details, such as the movie's synopsis, cast, trailer, and showtimes.**

### **Postconditions**

* **The user can choose to book tickets for the selected movie if he logged in.**

**Use case: book  tickets**

**Description: booking movie tickets in a cinema app, the user must be logged in and select the movie before and showtime and seat to purchasing tickets and number of tickets the user must select a payment method (credit card) he can choose the type of ticket(vip – normal) then user can save ticket in the app to show it when he go to the selected cinema, or he can cancel it, user can also pre-order food .**

**Function: buy a ticket for a movie.**

**Input: enter a credit card to pay for tickets.**

**Output: an e-ticket with all details the movie(cinema – movie).**

**Destination: user, database**

**Preconditions:**

**1. User has logged in to the cinema app**

1. **User has selected a movie and showtime, seat, number of tickets, type of ticket they want to attend**
2. **User have a credit card to pay for ticket**

**main sequence :**

1. **User opens the cinema app and selects the movie they want to watch.**
2. **User selects the showtime seat, type of ticket they want to attend from the available options.**
3. **User selects the number of tickets they want to purchase.**
4. **User confirms their selection and proceeds to the payment page.**
5. **The user presents the ticket at the cinema for entry.**

**Alternate Flow:**

* **If the user experiences an issue during the process, they can contact customer support for assistance.**

**Postconditions:**

* **The user has received the ticket details and is able to attend the movie.**

**Use case: User searches for movies in the app.**

**Description: from entering search criteria to viewing the search results and selecting a specific movie to view more details you can make search by (title – cast – director - genre)**

**Function: search for a movie.**

**Input: (title – cast – director - genre) of movie**

**Output: all similar movies with same name**

**Destination: user, database**

**Preconditions:**

* **User has access to the cinema app.**
* **The app has a search database of movies with information such as title, director, genre.**

**Main sequence:**

1. **User opens the cinema app and go to the search feature.**
2. **User enters the search criteria, such as movie title, director, or genre.**
3. **User submits the search button.**
4. **The app displays a list of movies that match the search criteria.**
5. **User selects a movie from the list to view more details.**
6. **The app displays the movie details, including title, rating, genre, synopsis, and trailer.**
7. **User selects the showtime they want to attend from the available options.**
8. **User selects the number of tickets they want to purchase.**
9. **User confirms their selection and proceeds to the payment page.**
10. **The app confirms the purchase and sends the ticket details.**
11. **The user presents the ticket at the cinema for entry.**

**Alternate Flow:**

* **If the search does not return any results, user can search again with criteria movies and submit again.**
* **If the user experiences an issue during the process, they can contact customer support.**

**Postconditions:**

* **User has successfully searched for movies.**
* **The app has displayed a list of movies matching the search criteria.**
* **User has successfully purchased movie tickets for the selected movie and showtime.**
* **The cinema app has updated the ticket inventory and payment records.**
* **The user has received the ticket details and is able to attend the movie.**

**Use case: Feedback**

**Description**: **user can add feedbacks of his experience or if he had a problem he can back to customer support he also can reporting a problem or issue with the app**

**Function: do feedbacks.**

**Input: user type his suggestion**

**Output: admin can read the feedbacks**

**Destination: user, admin**

**Preconditions:**

* **User has logged in the cinema app**
* **User has recently used a feature or purchased tickets in the app**
* **The app has a feedback feature or form for users to provide feedback**

**Main Flow:**

1. **User opens the cinema app and navigates to the feedback feature or form.**
2. **User selects the feature of the app they want to provide feedback on, such as the rate movie or booking process or movie selection.**
3. **User enters their feedback in the form, providing specific details and suggestions for improvement.**
4. **User submits the feedback to the app.**
5. **The app confirm the feedback and send it to admin then provides a confirmation message.**

**Alternate Flow:**

* **If the user experiences an issue or problem with the app, they can select a "report a problem" option instead of providing general feedback.**

**Postconditions:**

* **User has successfully provided feedback to the app.**
* **The app receives and send it to admin.**

## Use Case: pre-order food

## Description: selecting items from the menu and go to payment method to confirming the order and picking up the food at the cinema.

## Function: order the food for a movie

## Input: user select his food and have a payment method to pay

## Output:

## Destination: user

**Preconditions:**

* **User has access to the cinema app**
* **User is logged in to the app or has access to the pre-order feature**
* **The cinema offers pre-ordering of food and drinks through the app**

**Main sequence:**

1. **User opens the cinema app and logged in & navigates to the pre-order food feature or page.**
2. **User selects the items they want to pre-order from the menu, and select the quantity.**
3. **User adds the items to their order and reviews the order details, such as the total cost and any cinema will receive the order.**
4. **User select payment method and enters the necessary details, such as credit card information or a coupon code.**
5. **User confirms the order and receives a confirmation message with the order details.**
6. **User picks up the pre-ordered food at the designated location in the cinema.**

**Alternate Flow:**

* **If the user wants to modify their pre-order, they can select the order from their order history and make changes as needed before the pickup time.**

**Postconditions:**

* **User has successfully pre-ordered food in the app.**
* **The cinema receives the order and prepares the food for pickup.**
* **The user receives a confirmation message with the order details.**

**Actor: system (admin)**

## Use Case: Manage cinema app is browse

## Description: admin can do a lot of function ( add – edit – delete

## Function: mange the shown movies

## Input: admin must login with email and password

## Output: add, delete, edit the movies.

## Destination: admin.

### **Preconditions**

* **The admin is logged in to the Cinema app's admin panel (database).**

### **Main sequence:**

1. **The admin can add new movies by entering movie details, such as title, genre, release date, synopsis, cast, and trailer link.**
2. **The admin can edit existing movie details.**
3. **The admin can remove movies that are no longer showing in cinema.**

## Use Case: Manage Showtimes

## Function: mange the showtime of movies.

## Input: admin must login the database.

## Output: add, delete, edit the movies time.

### **Preconditions**

* **The admin is logged in to the Cinema app's admin panel(database).**

### **Main sequence:**

1. **The admin can add showtimes for a movie by selecting a movie, date, and time.**
2. **The admin can edit existing showtimes.**
3. **The admin can remove showtimes that are no longer available**

**Use Case: Manage Offer**

**Function: manage discounts**

## Input: admin must login the database.

## Output: add, delete, edit offers.

### **Preconditions**

* **The admin is logged in to the Cinema app's admin panel(database).**

### **Main sequence:**

1. **The admin can add offer for a movie to customer.**
2. **The admin can edit offer for a movie to customer.**
3. **The admin can remove offer for a movie to customer that are no longer available.**