
**Arab Academy for Science, Technology and Maritime Transport**

**College of Computing and Information Technology**

A Graduation Project Report Submitted to
the AAST's College of Computing and Information Technology
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science in Computer Science.

June, 2019, Alexandria, Egypt.

**Supervisors**

Prof. WALID ABDELMOEZ

Dr. RADWA FATHALLA

**Team Members:**

Hazem Mohamed

Maichael Alber

Mohamed Abd altatif

Mohamed El sayed

Yusuf Abdelmoniem

# TABLE OF CONTENTS

# VISION AND SCOPE DOCUMENT

## 1. BUSINESS REQUIREMENTS

### 1.1 Background

We're introducing an online cinema booking application which allows the user to do a full reservation process to a movie at the cinema he wants to watch the movie at, with the number of tickets he would like to have.

Nowadays, we see this idea is applied in different countries and it's usually solving some cinema issues. Most of these issues are:

People are wasting time waiting in a queue just to get their ticket(s) especially some seasons.
Seats which customers wanted to book weren't available to be reserved.

Customers usually need to know at which cinema they can watch the movie desired.

Moreover, they also must be capable of knowing when will be the movies shown.

So the application facilitate the process by online booking, gives the opportunity to the user to show the movie list, movie description, time schedule, select their preferred seat(s), knowing cinema locations , present an overview of halls available in this cinema and the most important issue the application solves that the customer can pay through different payment methods such as Visa.

### 1.2 Business Opportunity

We care about our goal because the target of our application is to get customers satisfied by reducing the time they may be wasting in booking their ticket(s) and on the other hand, in the future we hope our profit targets increase, and for the application, to spread everywhere.
Cinema will benefit from that application by having their profits increased and getting popular if nobody knows it

### 1.3 Business objectives:

| Financial | Nonfinancial |
|---|---|
| • Advertisements.<br>• Gain a commission on every ticket.<br>• Get sponsors on our product.<br>• Save currently spent on the maintenance for the cinemas' systems.<br>• Reach sales targets.<br>• User will get a discount for buying tickets.<br>• User will get offers from sponsors. | • Increase the popularity of application.<br>• Achieve customers' satisfaction.<br>• Reduce wasted time of ticket booking.<br>• Reduce the loads on cinemas' systems.<br>• Know whether customers are satisfied or not by applying surveys on our application.<br>• The different schedules in cinemas.<br>• Unknown cinemas will be infront of our customers' eyes. |

### 1.4 Success Metrics

**SM-1:** 75% of users who used the cinema at least 2 times per week during Q2 use the OBCS at least once a week within 5 months following initial release

**SM-2:** The average rating on the quarterly Update satisfaction survey increases by 0.45 on a scale of 1 to 6 rating within 3 months following initial release and by 1.0 within 12 months.

### 1.5 Vision Statement

For users who need to book tickets online, the online cinema tickets booking system is reaching them.

System is a mobile application that will ease the process of cinema tickets booking for the user. The application let the user choose the movie, cinema and seats, than spending time in a queue waiting to book their ticket(s).

Our application's system offers a user to choose from different ways to get their payment done.

### 1.6 Business Risks

RI-1: Marketplace will have many issues, such as most of the cinemas may not agree to give us access on their system or may refuse cooperating with us for the application's sake (Probability = 0.6; Impact=5)

Rl-2: Our user interface could be difficult for a user to use or too few people might use the application. For these issues, getting ads or sponsors may not be easy (Probability = 0.4; Impact=2)

Rl-3: Application can take too much time loading therefor a user may get bored (Probability = 0.3; Impact=6)

RI-4: Local cinemas might not have online system, which means that they won't be enjoying many functionalities which we offer. (Probability = 0.7; Impact = 9)

RI-5: If a cinema's rate given from users isn't satisfying enough, other cinemas might fear being in their shoes. (Probability = 0.5; Impact = 6)

### 1.7 Business Assumptions and Dependencies

An executive sponsor might set a business objective that a new application will increase revenue by 6000EGP per month. To establish this revenue target, the sponsor made some assumptions, perhaps that the new application will attract 200 additional unique visitors per day and each ticket booked 2% of its fees will be sent to the cinema. The new application does not attract enough visitors with a target number of tickets to get sold, if the application did not achieve its business objectives. If we learn that certain assumptions are wrong, you might have to change scope, adjust the schedule, or launch other projects to achieve the objectives.

## 2. SCOPE AND LIMITATIONS

### 2.1 Major Features

The main branch of the tree in the middle represents the product being implemented. Each feature has its own line or "branch" coming off that central main branch. The L1 features, the movie. The lines coming off an L1 branch are L2 features: View and choose. The branches off an L2 branch are the L3 features: movie name, category and trailer. Other branch coming off L1 feature: Cinema. The lines coming off an L1 branch are L2 features: select and show. The lines coming off an L2 branch are L3 features: select time, select seat and show location.

FE-1: Select and show movie details from the movie list the user is booking from. viewing movies and its details may not be afforded with the information needed since some cinemas might not be giving us full access.

FE-2: Selecting, viewing and changing cinemas and/or seats for a complete reserving operation.

FE-3: Provide system access through corporate intranet, smartphone, tablet, and outside Internet Access by authorized employees

FE-4: Managing or handle payment method (visa, through cinema) ,movie and cinema reservation such as (place, cancel , change) and Tickets Printing method .

FE-5: Managing Advertising, determining price and duration and the same for sponsors

## 2.2 Scope of Initial and Subsequent Releases

| Feature | Release 1 | Release 2 | Release 3 |
|---|---|---|---|
| FE-1, Movies | Select and show movie details | View movie history | Just view movies and his details from some cinema |
| FE-2,cinema | select, view and change cinema location | reserve, view ,change and cancel seats | Show schedules , halls No and view ticket price |
| FE-3, System access | Intranet and outside Internet access | Android phone and tablet apps | Windows Phone and tablet Apps |
| FE-4, Choose payment | Accept credit and debit card payments | Accept payments through merchant | |
| FE-5, Ticket(s) | View movie name cinema details and ticket price, user information and expiry date | Fully implemented | |

## 2.3 Limitations and Exclusions

LI-1: The OBCS shall be used only for cinemas in Alexandria only.

LI-2: The users who will pay through merchant won't be capable of reserving more than 5 tickets and with expiry date.

LI-3: Promotions will be active for users who will only receive it while other won't be able to use it.

# 3. BUSINESS CONTEXT

## 3.1 Stakeholder Profiles

| Stakeholder | Major value | Attitudes | Major interests | Constraints |
|---|---|---|---|---|
| user | Users mainly choose desired movie, then check the availability of this movie in the cinemas. Afterwards, he starts checking the seats he would like to reserve, and then he chooses the payment method and gets a printed ticket on his screen. | | Saving time , giving feedbacks On cinema , promotions | Availability of seats and schedule time , difference tickets price |
| Inactive user | An indirect user is a user who only opens the application to check the availability of movies in the cinemas around with their descriptions and also to check their schedules. | | The same interests of direct user | Limitation of seats , schedule time and difference tickets price |
| Merchant | The merchant's job regarding our system is to let the customer preview the printed ticket on the screen in case he chose to pay cash, and then checks the amount the customer needs to pay and collect it from him. | Verification reliability | Money and user time saving | Crowded cinema |
| Contact-point | His job is to provide us with the information needed regarding the cinema and the movies that the system needs to be aware of, to provide the users with them. | helpful but lazy | Giving information in time from cinema to app | Get updated information to cinema in exact time |
| Employees | They are usually the ones who connect the cinemas with the application to keep the application updated with the details needed to be provided from the cinemas. | Concern about any defect and permanent ambition for development and improvement | Job preservation | Make every change in time |
| Customer service | That's a team to answer the calls coming from the customers to help them with the issues facing them. | A strong commitment to the user and a | *Satisfy user and respond to feedback quick* | Must stay awake for responding call |

| | | | | |
|---|---|---|---|---|
| | | constant desire for help | | |

### 3.2 Project Priorities

| Dimension | Constraint | Driver | Degree of freedom |
|---|---|---|---|
| Features | All features scheduled for release 1.0 must be fully operational | | |
| Quality | 95% of user acceptance tests must pass; all security tests must pass | | |
| Schedule | | | Release 1 planned to be available by end of Q1 of year, release 2 by end of Q2; overrun of up to 2 Weeks acceptable without sponsor review |
| Cost | | | Budget overrun up to 15% acceptable without sponsor review |
| Staff | | Team size is half-time project manager, half-time BA, 3 developers, and 1 tester; additional developer and half-time tester available if necessary | |

### 3.3 Deployment Considerations

The web server software will need to be upgraded to the latest version. Apps will have to be developed for Android smartphones and tablets as part of the second release, with corresponding apps for Windows Phone and tablets to follow for the third release. Any corresponding infrastructure changes must be in place at the time of the second release. Videos no more than five minutes in length shall be developed to train users in both the Internet-based and app-based versions of OBCS .

## USE CASSES

| | |
|---|---|
| User | 1. Browse Cinema<br>2. Browse Movie<br>3. Make Reservation<br>4. Cancel Reservation<br>5. Make review<br>6. User Authorizing |
| Staff | 7. Manage Reservation |
| Cinema (s)server | 8. Exchanges Data |
| Bank | 9. Check card details validity |
| Advertising agency | 10. Manage Advertising<br>11. Manage Sponsors |

# I- Browsing movies

| | |
|---|---|
| ID and Name: | UC-1: Browsing movies |
| Created By: Michael | Date Created: December 15, 2019 |
| Primary Actor: user | Secondary Actors: OBCS |
| Description: It's a function to look through the application without doing anything, or to search around the application like searching for many movies without having intentions to book or attend any of them. | |
| Trigger: user browse movies managed by cinema and app | |
| Preconditions: PRE-1. User is logged in or a guest | |
| Post conditions: POST-1: OBCS loading movie list from DB | |
| Normal Flow: 2**.0 Browsing movies**<br>1. User log into app and requested available movie<br>2. OBCS show all available movie in different cinemas<br>3. OBCS recommend movie(s) for user based on previous choices | |
| Alternative Flows: : **None** | |
| Exceptions: 2**.0.E1 App may not be loading properly.**<br>**2.0.E2 App is showing the movies which were already deleted by cinema** | |
| Priority: High | |
| Frequency of Use: Approximately 300 users, average of one usage per day. Browsing movie first | |
| Other Information: None | |
| Assumptions: : Assume that 60 percent of Users will prefer browsing movie first | |

## II- Browsing cinema

| | |
|---|---|
| ID and Name: | UC-2: Browsing cinema |
| Created By: Michael | Date Created: December 15, 2019 |
| Primary Actor: user | Secondary Actors: OBCS |
| Description: is the act of looking through a set of information quickly, without a specific sense of purpose. In the context of the cinemas | |
| Trigger: user browse cinema location managed by app | |
| Preconditions: PRE-1.user must log in to app | |
| Post conditions: POST-1: OBCS show the cinema list to user<br>POST-1:OBCS get the recommended cinemas from DB | |
| Normal Flow: 2**.0 Browsing cinema**<br>1. User log into app and access GPS<br>2. OBCS get all the Cinemas from DB<br>3. If user Opens GPS then OBCS recommend His/her nearest cinema<br>4. Else OBCS get Recommended cinema based on His/her pervious rating | |
| Alternative Flows: **None** | |
| Exceptions: 2**.0. E1 App maybe can't get GPS location** | |
| Priority: High | |
| Frequency of Use: Approximately 400 users, average of one usage per day. Browsing Cinema first | |
| Other Information: OBCS get recommended Cinema from form filled by this user before | |
| Assumption: Assume that 60 percent of Users will prefer browsing movie first | |

## III- Make reservation

| | |
|---|---|
| ID and Name: | UC-3: Make reservation |
| Created By: Yusuf | Date Created: April 4, 2019 |
| Primary Actor: User | Secondary Actors: Application |
| Description: User after browsing movie or cinema, he/she will start the reservation sequentially; movie from the list , date of the movie from cinema schedules, seat(s) and payment. | |
| Trigger: A user select movie, cinema and seats which he wants to book. | |
| Preconditions: PRE-1. User finished browsing<br>PRE-2. User is logged in or a guest | |
| Post conditions : POST-1:user choices are Stored in OBCS<br>POST-2:OBCS store these info in the DB with status "User Recommendation" | |
| Normal Flow: **1.0 Make reservation**<br>1. User asks to view movie list.<br>2. OBCS displays movie list which is available in different cinemas.<br>3. User select only one movie for every ticket from the movie list.<br>4. OBCS display movie details for user to be capable of showing its details.<br>5. Then User will start choosing the cinema which is near or preferred.<br>6. OBCS get all cinemas that have the selected movie.<br>7. Then the user selects the cinema which he will watch the movie at.<br>8. OBCS represent cinema parties' schedule that have the movie(s) selected.<br>9. User selects the party from schedule.<br>10. OBCS get available seat(s) from database and show them for user(s).<br>11. User selects preferred seat(s) from the available seats.<br>12. OBCS displays payment methods.<br>13.0) if User choose visa as a way of payment then OBCS send the user to the visa page that contact directly with the banks.<br>13.1) user confirm the reservation.<br>13.2) OBCS show ticket for user(s).<br>14. else if the User chose to pay at the cinema then confirm and print ticket with an expiry date.<br>15. OBCS notify all users with successful reservation. | |
| Alternative Flows: **3.1 Reserve at the cinema itself**<br>1. User selects a near or preferred cinema.<br>2. Then OBCS show movie(s) list available in this cinema<br>3.Return to step 3 of normal flow<br><br>**3.2 Reserving more than one ticket**<br>1. User selects more than one ticket from the movie list.<br>2. Return to step 4 of normal flow | |
| Exceptions: **3.0.E1 The selected movie is not available for reservation**<br>1. OBCS informs the user that the movie isn't available | |

2. Return to step 1 of normal flow

**3.2E3 Limited number of tickets to pay at the cinema**

1. OBCS informs the user of the maximum number of tickets he can reserve

**3.3E4 After selecting the payment method and he changed his mind**

1a. if user finishes the payment by visa OBCS will Update the Balance

1.b return to step 12

2a. else if user wants to change the payment from paying at cinema to visa

2.return to step 13

Priority: High

Frequency of Use: Approximately 200 users, average of one usage per day. Peak usage load for this use case is Party from 9:00A.M to 11:00 A.M

Other Information:

1. User shall be able to cancel the reservation process at any time prior confirming it.
2. If the user cancelled the reservation before the party starting time, OBCS take 50% of ticket price otherwise take the whole ticket price
3. User shall be able to view all the movies he watched within the previous six months and OBCS recommend some movies in the same category

Assumptions: Assume that 10 percent of Users will choose movie from the recommended list

## IV- Cancel reservation

| | |
|---|---|
| 1. ID and Name: | UC-4: Cancel reservation |
| Created By: Yusuf | Date Created: Nov 10, 2019 |
| Primary Actor: User | Secondary Actors: Application |
| Description : user is able to cancel the reservation at any time based on the rubric of the OBCS except when the movie starts | |
| Trigger: User changes his opinion by cancelling his reservation. | |
| Preconditions: PRE-1. User's reservation already done. | |
| Post conditions: POST-1: when the user cancels his reservation, OBCS update his database.<br>POST-2: with every reservation cancellation, OBCS by default mark (x) on his username. | |
| Normal Flow: **4.0 Cancel reservation**<br>1. user cancels the reservation<br>2. OBCS send to database to delete reservation and put the status "cancel reservation". On user table<br>3. OBCS will apply the rubric depending on the time the user cancelled at.<br>4. OBCS notifies the user with this update. | |
| Alternative Flows: **4.1 cancel reservation (pay through cinema)**<br>1. Start after step 2<br>2. OBCS will check the number of (x)s put on his account.<br>3. If there were more than 3 (x)s, OBCS will block his account.<br>4. Else, notify user with the number of (x)s left.<br>**4.2 cancel reservation (visa)**<br>1. After step 3<br>2. OBCS deduct by default the price of the ticket based on rubric.<br>3. Bank reply to OBCS with the new user balance<br>4. OBCS update the user wallet | |
| Exceptions: 2.0E1: User cancelled at a time which no penalty fees are applied but his money did not get back to his wallet.<br>2. User cancelled at a time which penalty fees should be applied, but the full amount was sent back to his wallet.<br>3. User was blocked before exceeding his 3 (x)s. | |
| Priority: | High |
| Frequency of Use: Approximately 30% of users follow this process. | |
| Other Information: N/A | |
| Assumptions: Assume that 10% of the users per day who cancel reservation are divide to (cancelling through visa or bank). | |

## V- Make review

| | |
|---|---|
| ID and Name: | UC-5: Make review |
| Created By: Hazem | Date Created: Nov 11, 2019 |
| Primary Actor: User | Secondary Actors: OBCS |
| Description: After the movie ends, the OBCS send a form to the user to rate the cinema, the movie and the service he received, to put these choices in consideration at the future reservations. | |
| Trigger: User makes a review. | |
| Preconditions: PRE-1. User already finished watching the movie. | |
| Post conditions: POST-1: Form will be saved in the database under the users' opinion table.<br>POST-2: Will take the user's review from the form filled, to view his recommended movies list. | |
| Normal Flow: 5.0Make review<br>1. OBCS notifies the user with this form to fill it after the movie.<br>2. User fills the form, and then it get saved in the OBCS.<br>3. OBCS filters this form, then saves it in the database.<br>4. OBCS thanks the user for filling this form. | |
| Alternative Flows: N/A | |
| Priority: | High |
| Frequency of Use:<br>Approximately 90% of the application users will fill out this form. | |
| Other Information: 1. Cinema rating is sent to the cinema server to provide better service.<br>2. According to this form, the OBCS creates statistics to provide them to the users afterwards. | |
| Assumptions: Assume that 50 users won't fill this form (monthly). | |

## VI- User Authorizing

| | |
|---|---|
| ID and Name: | UC-6: User Authorizing |
| Created By: Mohamed abdelwahed | Date Created: January 15, 2019 |
| Primary Actor: staff | Secondary Actors: User |
| Description: When the user opens the application and enters his verified information, on the other hand, the application must perform this operation to verify the user and then open his/her profile | |
| Trigger: OBCS check User Authorization | |
| Preconditions: PRE-1.user must have an account to make an authorization | |
| Post conditions: POST-1: OBCS save last authorized user in DB<br>POST-2: OBCS notify the user with update | |
| Normal Flow: 6**.0 User Authorizing**<br>5. User Opens the application and login<br>6. OBCS get user name & password and send them to the DB to start checking them<br>7. If the information where identical, OBCS send a verification MSG to confirm and open the account<br>8. Else OBCS ask user to try again | |
| Alternative Flows: **create account**<br>1- user select sign up<br>2- OBCS open the registration form<br>3- user fill the form and save it<br>4- OBCS use it to start saving in the DB<br>5- normal flow step 3 | |
| Exceptions: 6.**0** user insert the same email or user name<br>1-OBCS check email or user name in DB, if they do not exist, confirm<br>2- else ask user to choose other name or email | |

| |
|---|
| Priority: High |
| Frequency of Use: none |
| Other Information: none |
| Assumption: Approximately 90 % of people who use application have account and 10% use it as guest |

## VII- Manage Reservation

| | |
|---|---|
| ID and Name: | UC-7: Manage Reservation |
| Created By: Mohamed abd ellatif | Date Created: December 15, 2019 |
| Primary Actor: staff | Secondary Actors: OBCS |
| Description: This function is used to manage booking and maintain a lot of functions such as reviewing user reservations, view/ change seat(s) at the cinema, add a special request, change reservation and change the payment method | |
| Trigger: OBCS Manage the reservation for the user and the cinema server | |
| Preconditions: PRE-1.user complete reservation<br>PrE-2:user cancel reservation | |
| Post conditions: POST-1: OBCS get user name from the DB to cancel or confirm the ticket<br>POST-1:OBCS delete the id in the DB when the user cancels the reservation | |

**Normal Flow:** **6.1 Review user reservations**

1-user ask to review his last reservation details

2-OBCS get the reservation details from the DB and review tickets(s) details

And return a full description about the reservation

**6.2 view/change seat(s)**

1- User(s) ask to view available seats
2- OBCS get available seats from database
3- user ask to change the seats, then the seats are removed and other ones are chosen
4- OBCS get the user choice and updates seats and the user ticket in the DB
5- OBCS notify user to confirm

**6.3 change reservation**

1- User ask OBCS to change reservation (either the movie or the cinema)
2- OBCS check the availability of the user's requests in the DB
3- If the DB returned true, the OBCS sends a confirmation

to the user and get the DB updated

4- Else OBCS notifies the user that the request cannot be done

**Alternative Flows: 6.4 Add a special service request**

1- User Requested Alternative services such as VIP seats, Drinks and food
2- OBCS get the cinema name from the ticket
3- OBCS sends to the cinema server to check the availability of the user's requests
4- Cinema server replies to the OBCS (accept, reject)
5- OBCS return to user status (done or not available)

**6.4 change payment method**

1-user asks to change the payment method
2- OBCS gets the current payment method from DB
3-if the users want pay by visa then the OBCS move to the visa page
4-else, OBCS notify the user that it is not available to change from visa to pay at the cinema

Exceptions: E1-6.**0** user has a problem with the payment

1. If the user has any problems in his payment, then send to the OBCS to check payment
2. OBCS responded to the user with all the details about his payment

Priority: High

Frequency of Use: Approximately. all users deal use the manage reservation feature

Other Information: this function have some features which need a staff or an administrator to deal with it

Assumption: Assume that 10 percent of the users have a problem in their payments and 70 percent of them want to make changes

## VIII- Exchange data(s)

| | |
|---|---|
| 1. ID and Name: | UC-8: Exchange data(s) |
| Created By: Mohamed abdelwahed | Date Create: April 18, 2019 |
| Primary Actor: Cinema server | Secondary Actors: OBCS |
| Description: cinema(s) server must always connect to the OBCS for many reasons such as checking available seats<br>and getting all the info about the new movie, halls and schedule | |
| Trigger: Cinema server exchanges data with cinema OBCS | |
| Preconditions: PRE-1. OBCS must be connected to the cinema server | |
| Post conditions: POST-1: cinema server updates the DB with the new reservation from the OBCS | |
| Normal Flow: **8.0** Exchange data(s)<br>1. OBCS send the movie name, parties' time table and halls to the cinema server to check available seats<br>2. Cinema server check the availability in the DB using the Info sent from the OBCS<br>3.a if there are any seat(s) available, cinema server return (empty)<br>3.b else return (reserved)<br>4. OBCS Update the DB with the new response and display the new list for the users or confirm the reservation<br>5. OBCS when a ticket is reserved, it sends the ticket ID to the cinema to check the ticket and update his DB | |
| Alternative Flows:<br>1- OBCS send a request to the cinema server with the new movie , halls no. and the last modification for the schedule<br>2- Cinema server check the DB and sends the requested information to OBCS | |
| Priority: | High |
| Frequency of Use: OBCS requests their latest update, to update its own DB. | |
| Exception: E1-0 Cinema server dropped for some reason and we weren't updated or didn't receive any confirmation regarding the seat(s)<br>E2-0 Cinema server updates the DB after OBCS's request is done | |
| Other Information: N/A | |
| Assumptions: None. | |

## IX- Check card details validity

| | |
|---|---|
| ID and Name: | UC-9: Check card details validity |
| Created By: Michael | Date Create: May 5, 2019 |
| Primary Actor: Bank | Secondary Actors: Application |
| Description: When the user enters his card details at the bank's page, the application requests the card validity from the bank's server and waits for the response. | |
| Trigger: OBCS requests the card validity from the bank. | |
| Preconditions: PRE-1. User must choose paying using visa. | |
| Post conditions: POST-1: The bank checks if this transaction can be done or not.<br>POST-2: The bank sends a notification to the OBCS whether the card is valid or not.<br>Post-3: The OBCS informs the user with the response received from the bank. | |
| Normal Flow: 13.0 Check card details validity<br>1. User chooses to pay using a visa.<br>2. User will enter his information to validate his visa and pay.<br>3. If the money is withdrawn, the bank will send a confirmation message to the OBCS.<br>4. If the amount can't be withdrawn, a notification will be sent to the OBCS with the reason.<br>5. The response will be shown to the user. | |
| Alternative Flows: None | |
| Exception: If the user has no enough money in his bank account, the OBCS will offer him paying in another way, such as paying at the cinema. | |
| Priority: High | |
| Frequency of Use:<br>Approximately | None |
| Other Information: None | |
| Assumptions: Assume that 10 percent of the users, their card will be invalid. | |

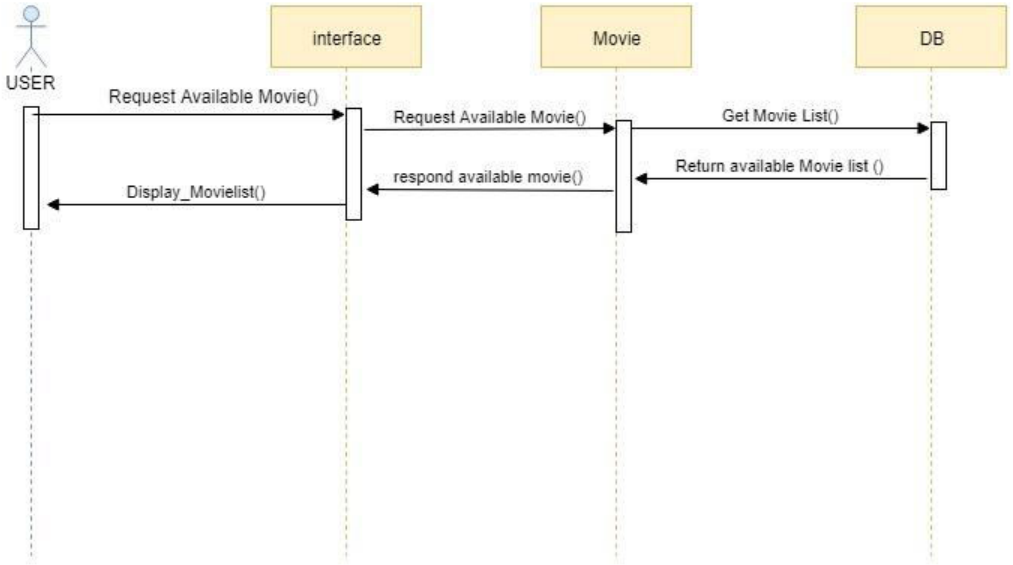
## X- Manage advertisement

| | |
|---|---|
| ID and Name: | UC-10: Manage Advertisement |
| Created By: Hazem | Date Created: December 14, 2019 |
| Primary Actor: OBCS | Secondary Actors: Advertising agency |
| Description: This function will be responsible for setting all regulations between the business and the advertising agency. | |
| Trigger: Setting the deals between the business and the advertising agency. | |
| Preconditions: PRE-1. None. | |
| Post conditions: POST-1: Save advertisement(s) expire in the database. | |
| Normal Flow: **10.0 Advertisement operation**<br>1. The agency will request showing an advertisement on our pages.<br>2. We will respond to the agency whether with an approval or a denial.<br>3. If they received our approval, they will start sending the advertisement content.<br>4. If our business is okay with this content, will set the agreements and show the advertisement when done.<br>5.else we ask for new ads | |
| Alternative Flows: : None | |
| Exceptions :**E10.0 Our business denied the advertisement for some reason (either would not like to post advertisements for the instance, or the content wasn't satisfying from the business' point of view).** | |
| Priority: | High |
| Frequency of Use: Business receives 10-15 advertisement request per month. | |
| Other Information: The advertisement won't get a an area or a bar in the application interface to be shown, but it will appear for the user as a photo or a video which he can stop showing it after about 10 seconds. | |

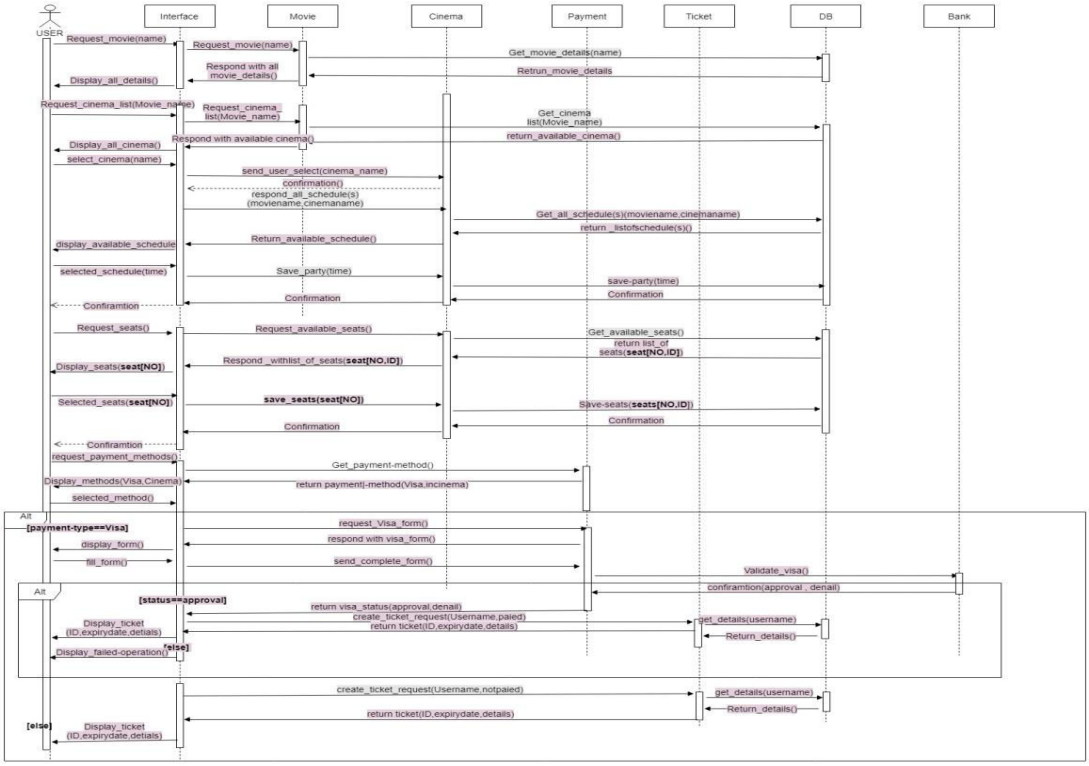
## XI- Manage Sponsor

| | |
|---|---|
| ID and Name: | UC-11: Manage Sponsor |
| Created By: Mohamed abdelatif | Date Created: December 14, 2019 |
| Primary Actor: OBCS | Secondary Actors: Advertising agency |
| Description: the advertising agency will offer some sponsors to choose one to be our official sponsor | |
| Trigger: Setting the deals between the sponsor and the OBCS. | |
| Preconditions: PRE-1. None. | |
| Post conditions: POST-1: upload logos and ads for official sponsor.<br>POST-2: save all the sponsors' contracts in our database. | |
| Normal Flow: **11.0 Manage Sponsor**<br>1. The agency will request showing some offers from the sponsors<br>2. We will respond to the agency whether with an approval or a denial.<br>3.if the offer was accepted then the agency will connect us with the sponsor to sign the contracts<br>4. if the offer was rejected, then the agency will search for alternative sponsors. | |
| Alternative Flows: : None | |
| Exceptions :**N/n** | |
| Priority: | High |
| Frequency of Use: None | |
| Other Information : OBCS allow the official sponsor to show ads and logos. | |
| Assumption: Assume OBCS will get from 1-4 sponsors each year. | |

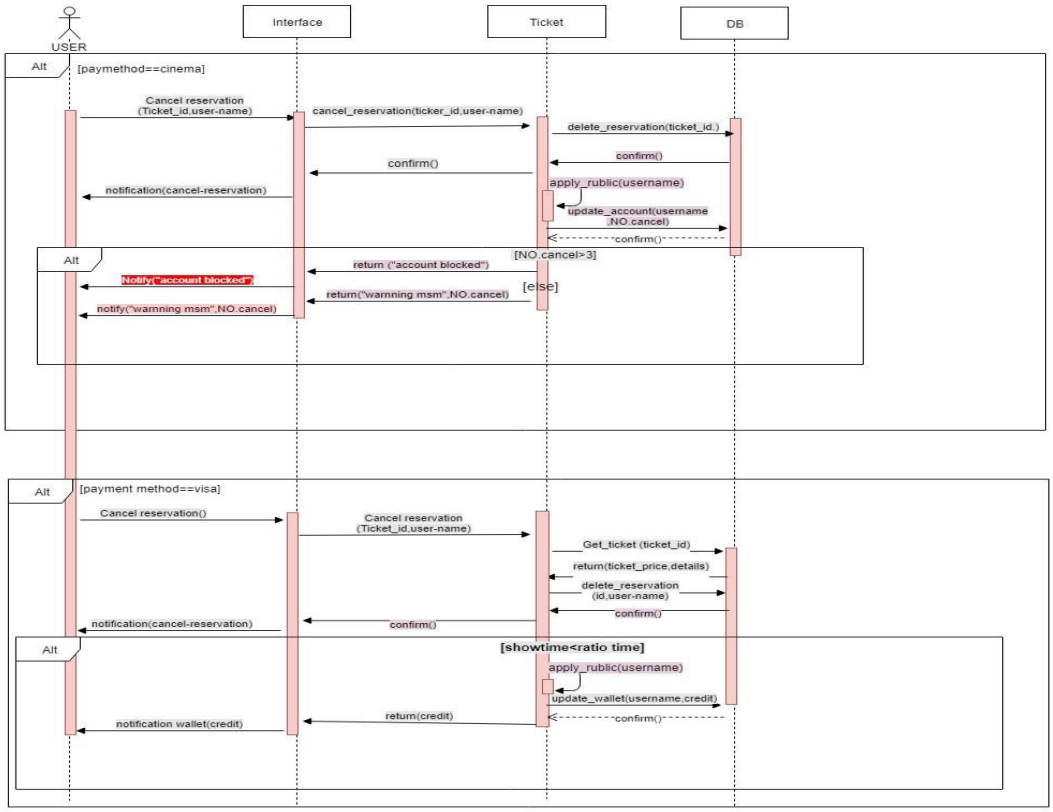
## Browse Movie



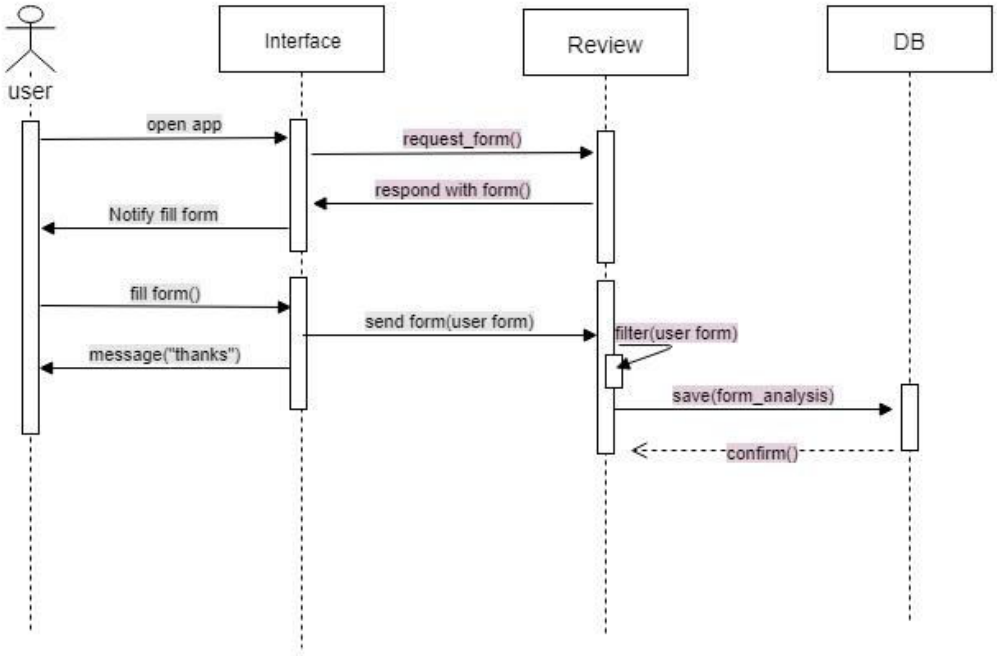
## Make Reservation



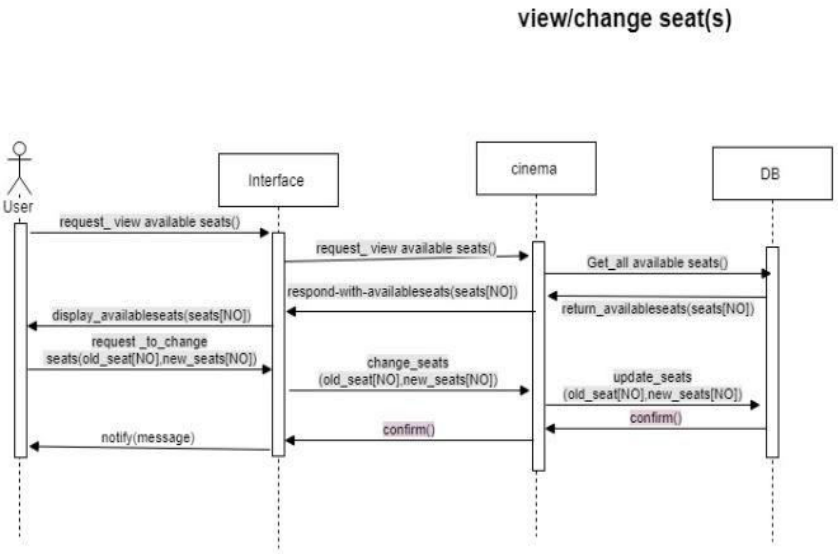
## Cancel Reservation



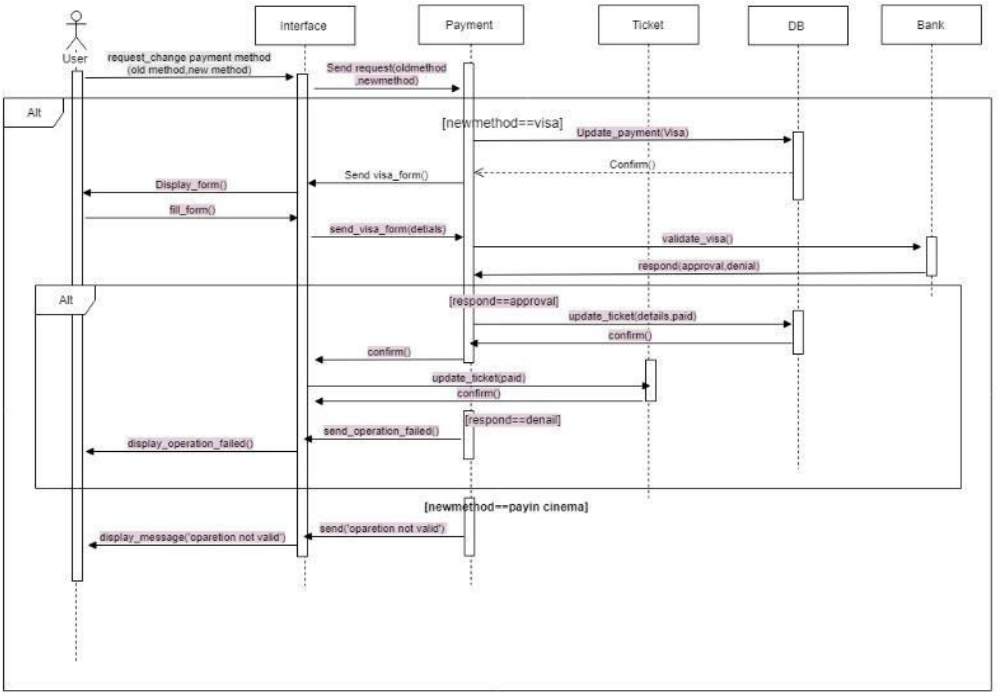
## Make review



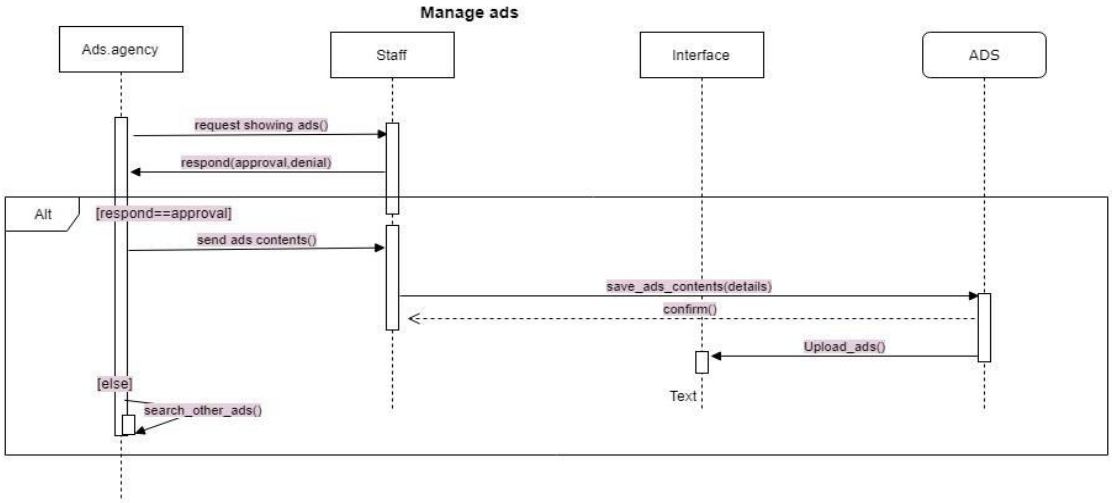

## Manage Reservation

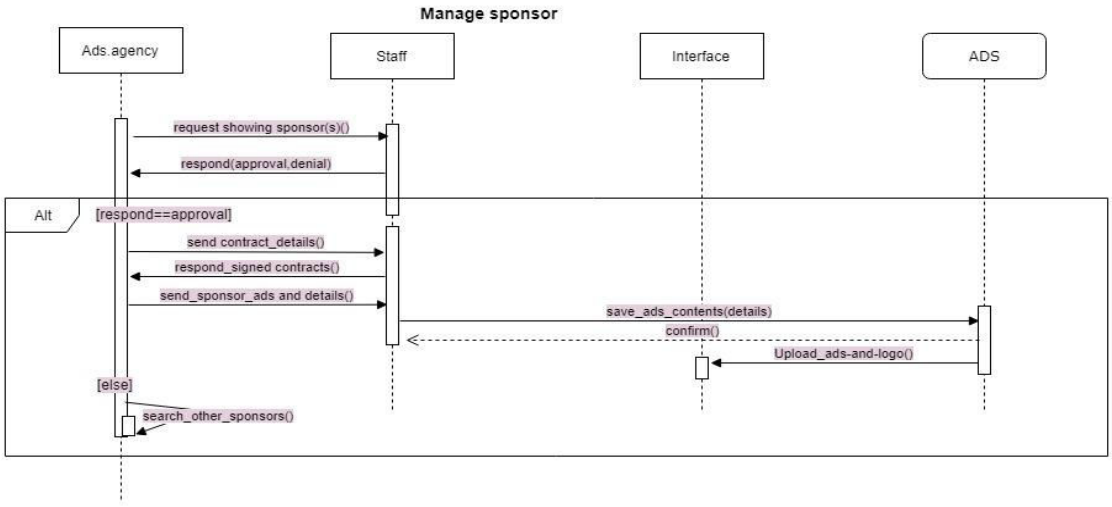


## Manage Reservation
## (Change payment method)

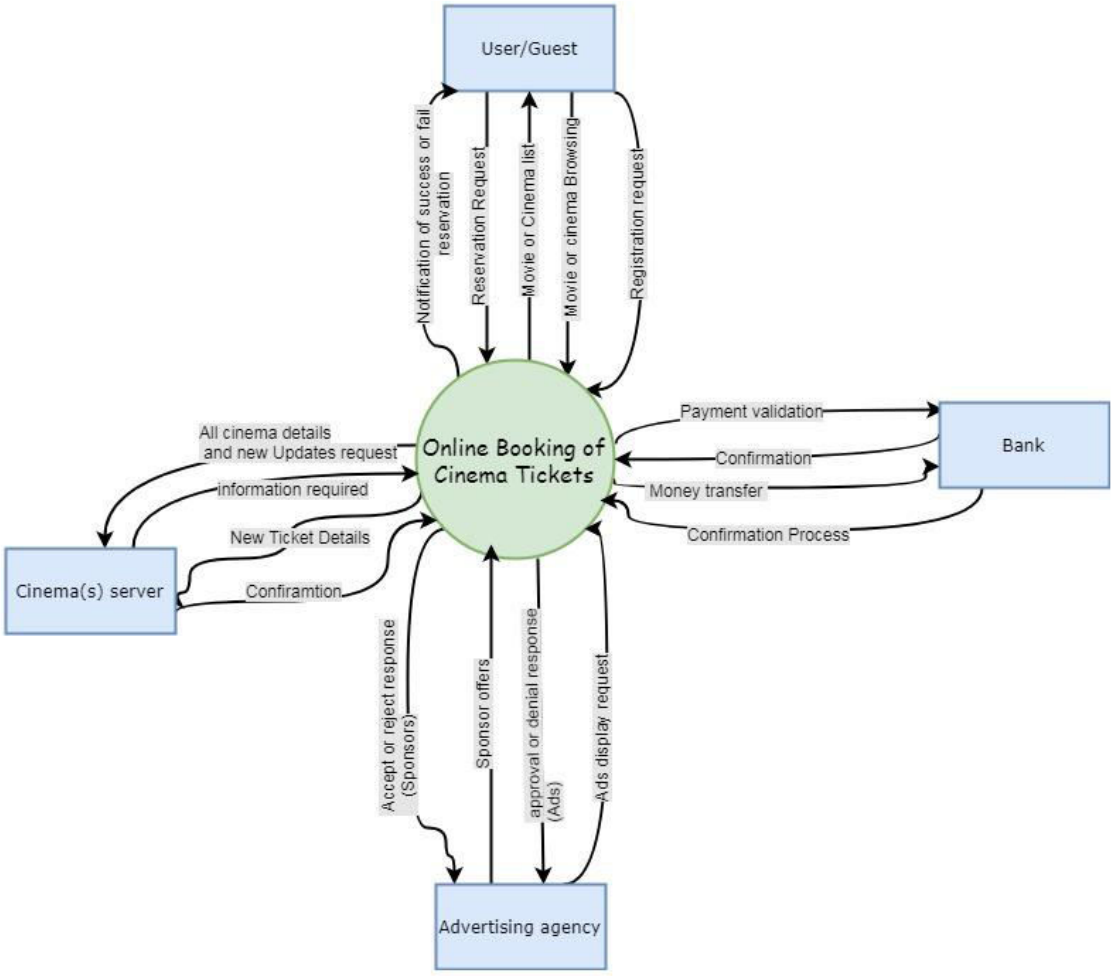


## Manage Advertising

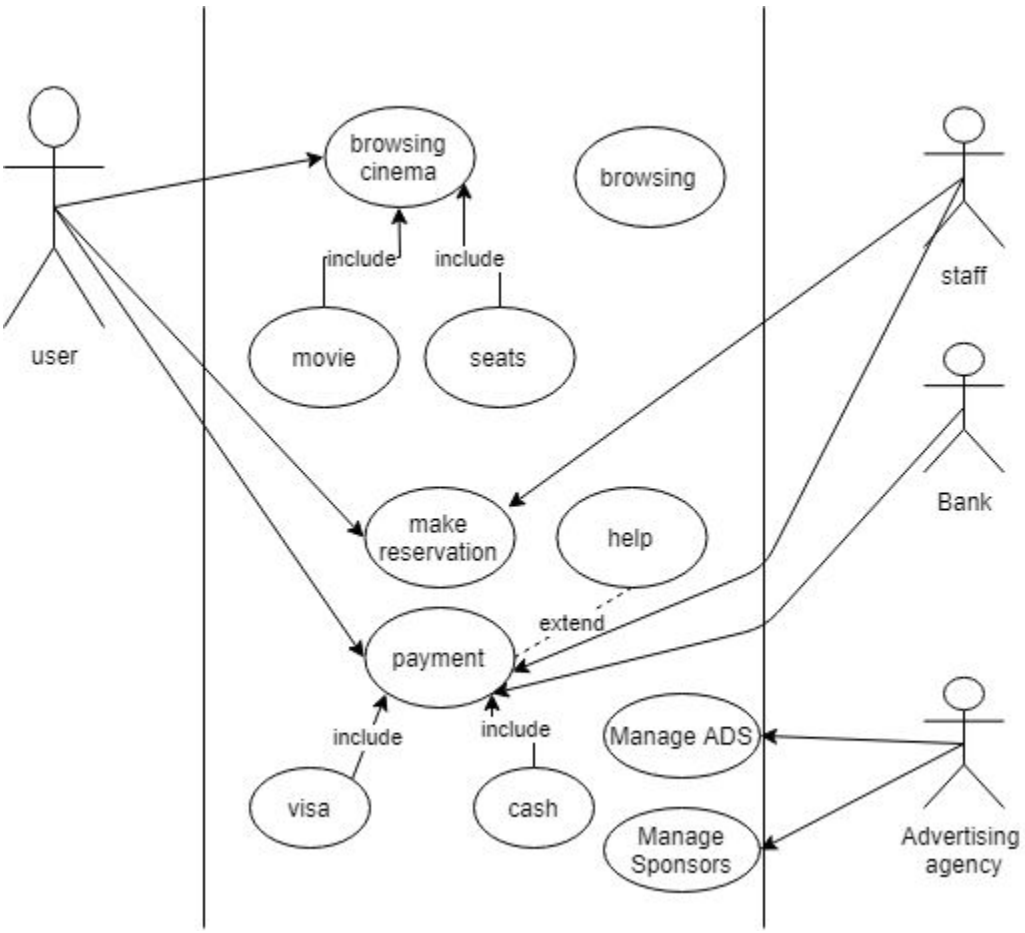


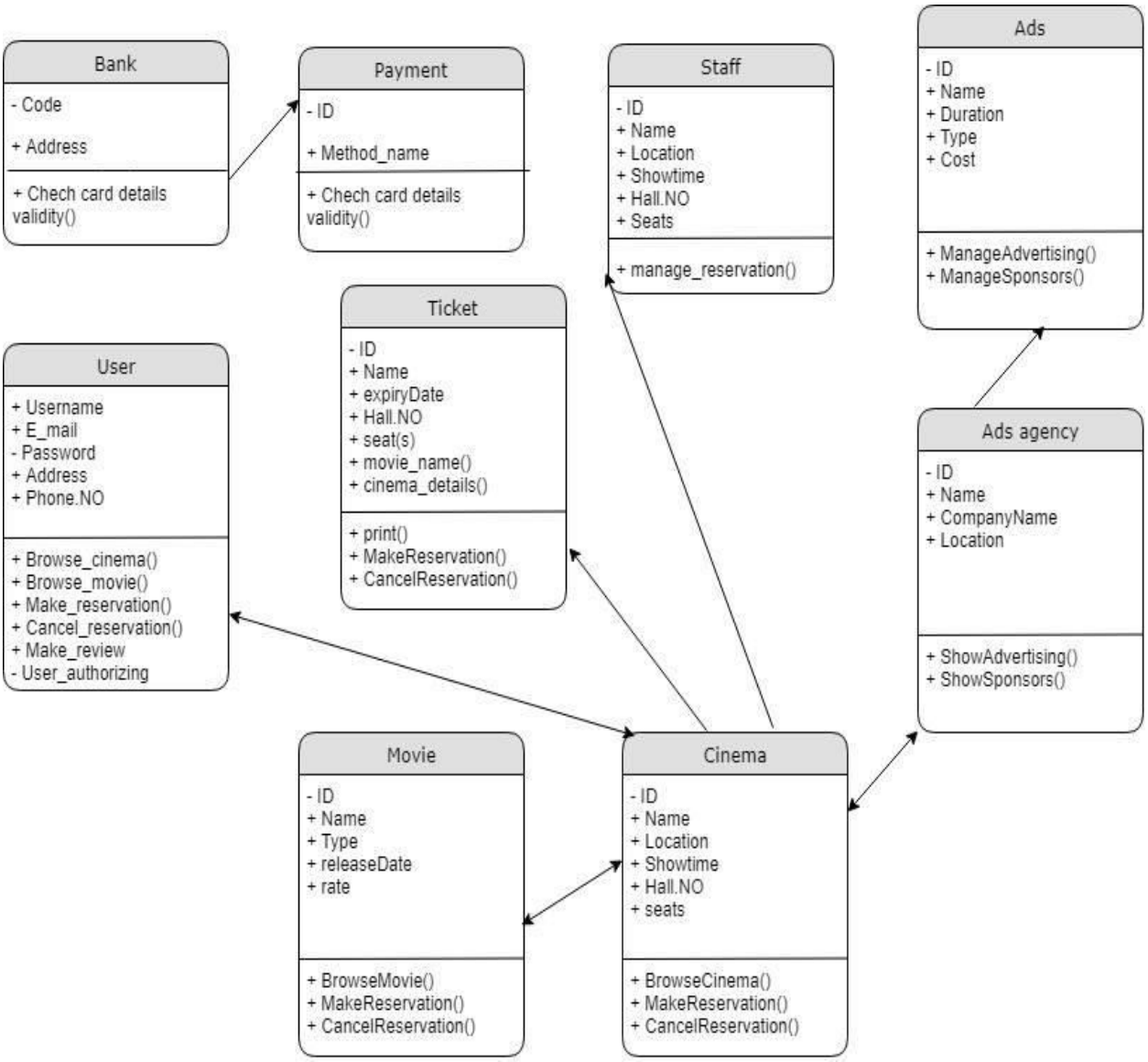
## CONTEXT DIAGRAM



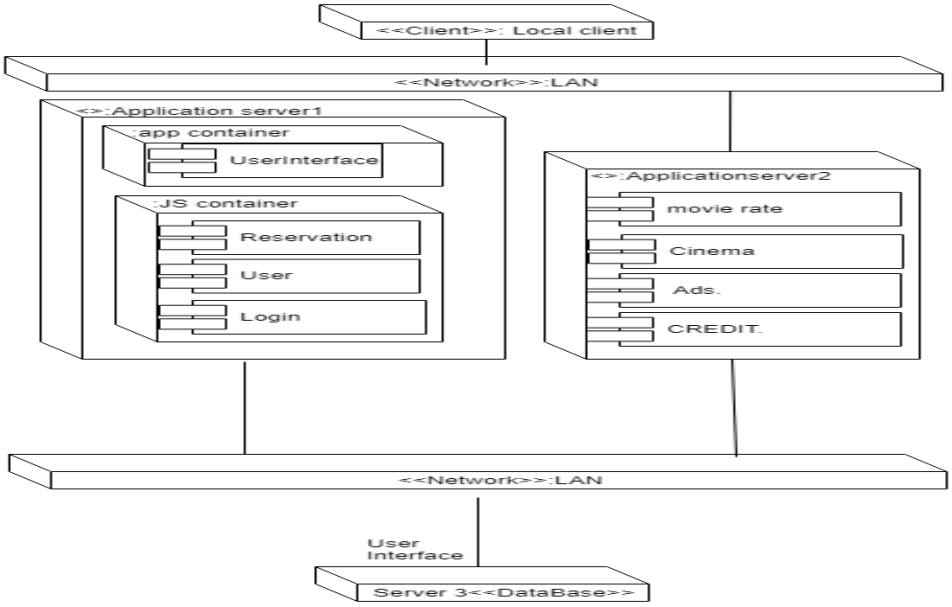
## Use case


browsing cinema
browsing
include
include
user
movie
seats
staff
Bank
make reservation
help
extend
payment
include
include
Manage ADS
visa
cash
Manage Sponsors
Advertising agency

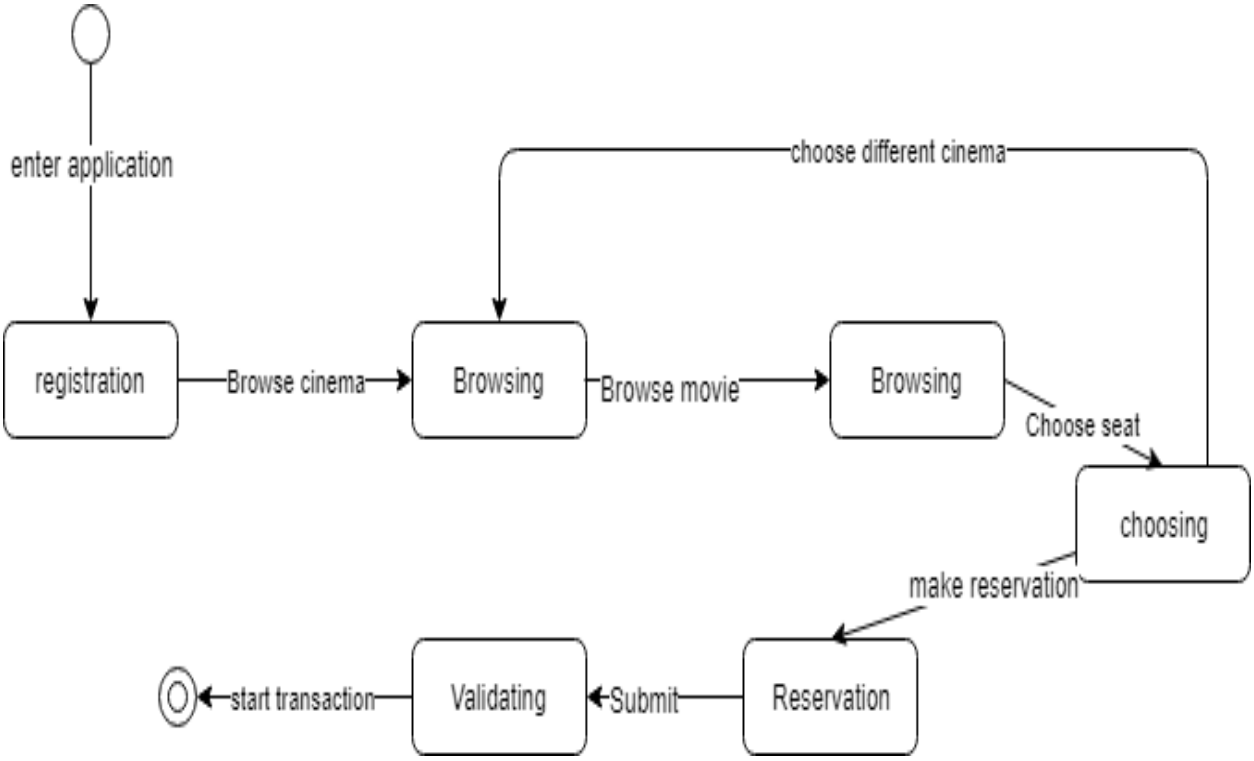
## CLASS DIAGRAM



## DEPLOYMENT DIAGRAM



## STATE CHART


enter application
registration
Browse cinema
Browsing
Browse movie
Browsing
Choose seat
choosing
choose different cinema
make reservation
Reservation
Submit
Validating
start transaction

# Software Tools

1- Draw.io
2- ERD Plus
3- Microsoft Word
4- Microsoft Power Point
5- Visual Paradigm

# Implantation Issues

1- Firebase VS Sequel DB
2- SQL's disadvantages
3- IMDB and Arabic Movie
4- How synchronize with Theatres

## Fire base VS Sequel DB

1) Easiest DB ionic support (Ionic is flexible when using FB)

2) It is working with API which means that my work will be online from the very beginning and accessing the DB on the internet.

3) More secure.

## SQL's disadvantages:

1) Ionic doesn't support PHP so MySQL and such alternatives won't work

2) Ionic supports Cordova Apache which only supports MySQL Lite (low security).

3) Everything in SQL Lite must be written manually, some files can only be used offline and updates must be done manually.

## How does the cinema gets informed that a seat was already booked through the application to prevent booking it for anyone?

There are 3 solutions; first one is having an API to get access on the cinema's database and update it on spot. But unfortunately won't work because after we searched, we found that there are no cinema's that are using an online accessible database, even VOX doesn't support API.

Second way was booking a number of seats from the cinema manually for each month and pay for them, but we found out that this solution wasn't really efficient since we can never be sure that all of the seats we booked for the application will be taken, which means that we might lose money.

Third solution and the one we really used is giving a cinema agent the access on the booking page by some mean, that agent will be capable of updating both the cinema and the application of the seats taken.

## IMDB

- We are getting movies' details using an API from Amazon IMDB, some of this site's advantages is that it saved us some DB storage, trustful.
- Amazon IMDB and Tomato give better rates and details and give the API for one year.
- The disadvantage is that it doesn't support Arabic movies or unknown ones. It supports huge movies like (الفيل الازرق).

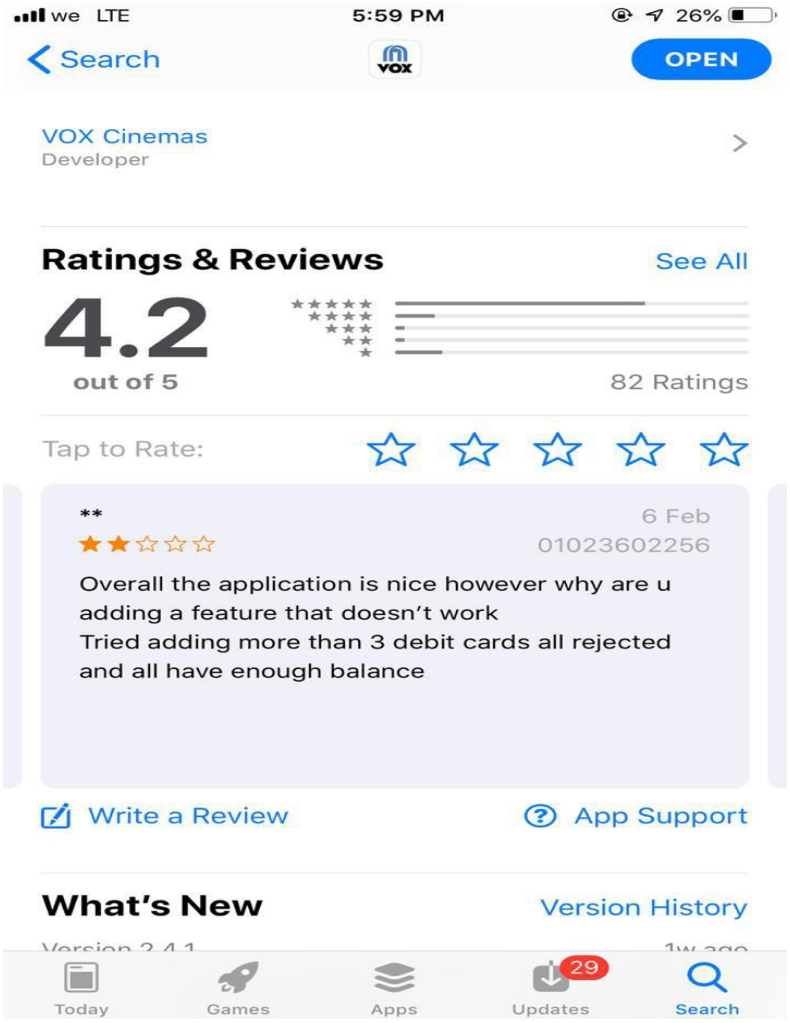
## Future work:

We will create a recommended cinema and a recommended movie depending on the cinemas he visited or rated and the movie genres he entered.

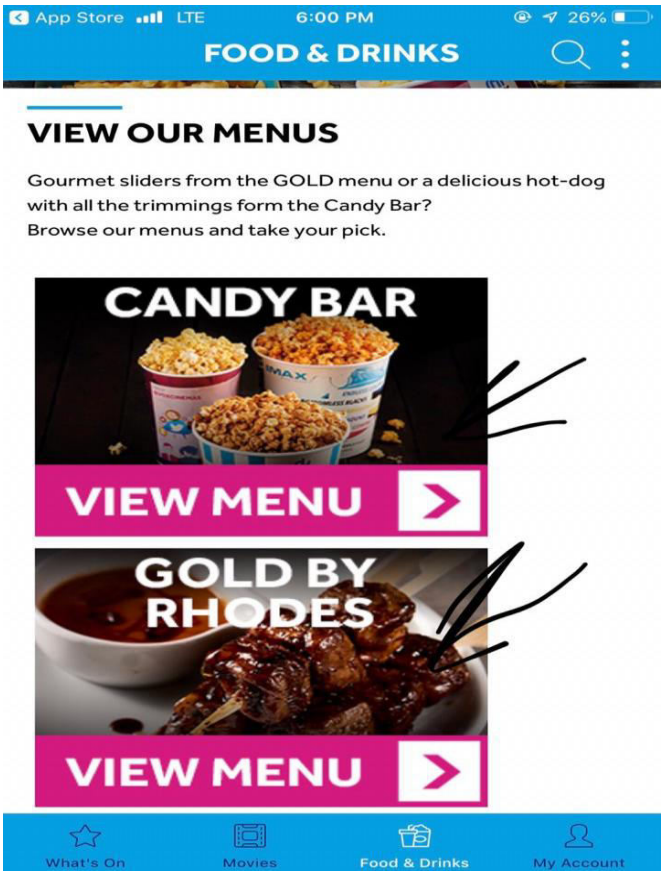
## Bonus feature:

- After a user watches a movie, a page will appear for him to rate the movie and the cinema.
- He can add a cinema to a favourite list of his to show him the movies shown by it.
- Continuous offers for the users as a discount if booked a certain number of tickets.

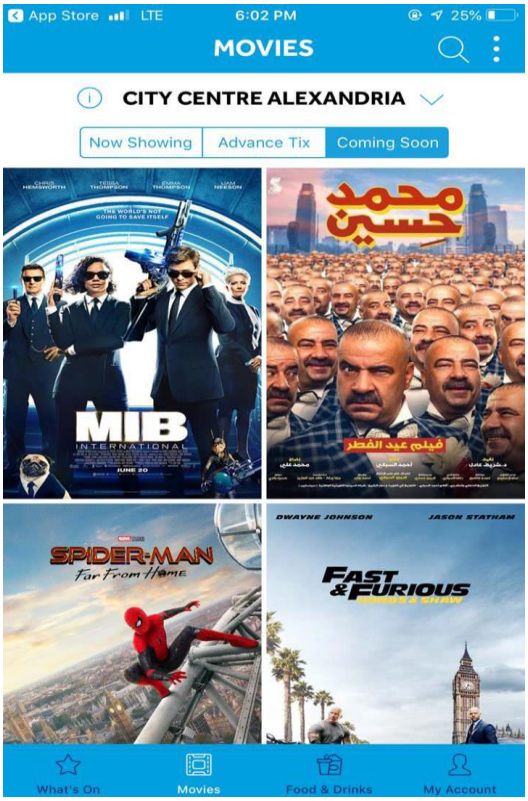
**VOX Cinemas** is the Middle East's most innovative and customer-focused exhibitor.




A negative point at VOX's application is that it sometimes rejects some cards for no reason as stated in the review.




Some of the links which are put on the page are non-functional.




- No synchronization as some movies are viewed on the “Coming Soon” tab while they are already being shown and should be in the “Now Showing” tab.

# Implantation Tools

- Ionic V4
- Cordova Apache
- Node JS
- Angular JS
- JQuery
- Firebase, Firestore
- JavaScript
- Photoshop & Illustrator

## Conclusion

- In this application we afforded multiple ways for the user to go through the application and reach his goal depending on which way he prefers.

- We used secure ways to finish our application to make sure that the user's privacy is safe and sound (Ex: FireBase, Ionic, validation techniques).

- The application saves the users time waiting in long queues and can save him money through the offers a cinema can do to get more seats occupied.