Use Cases

for

Kim’s Convenience Modern Cinema

Version 1.1 approved

Prepared by Team Kim’s Convenience

Nanyang Technology University

02 September 2021

Revision History

Name	Date	Reason For Changes	Version
Hussain, Jeremy, Wong Ying, Duc, Desmond	30 Aug 2021	Initial use case description draft	1.0
Wong Ying	1 Sep 2021	Minor edits	1.1

1. Guidance for Use Case Template

Document each use case using the template shown in the Appendix. This section provides a description of each section in the use case template.

2. Use Case Identification

2.1. Use Case ID

Give each use case a unique integer sequence number identifier. Alternatively, use a hierarchical form: X.Y. Related use cases can be grouped in the hierarchy.

2.2. Use Case Name

State a concise, results-oriented name for the use case. These reflect the tasks the user needs to be able to accomplish using the system. Include an action verb and a noun. Some examples:

- View part number information.
- Manually mark hypertext source and establish link to target.
- Place an order for a CD with the updated software version.

2.3. Use Case History

2.3.1 Created By

Supply the name of the person who initially documented this use case.

2.3.2 Date Created

Enter the date on which the use case was initially documented.

2.3.3 Last Updated By

Supply the name of the person who performed the most recent update to the use case description.

2.3.4 Date Last Updated

Enter the date on which the use case was most recently updated.

3. Use Case Definition

3.1. Actors

An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks. Different actors often correspond to different user classes, or roles, identified from the customer community that will use the product. Name the actor that will be initiating this use case and any other actors who will participate in completing the use case.

3.2. Trigger

Identify the event that initiates the use case. This could be an external business event or system event that causes the use case to begin, or it could be the first step in the normal flow.

3.3. Description

Provide a brief description of the reason for and outcome of this use case, or a high-level description of the sequence of actions and the outcome of executing the use case.

3.4. Preconditions

List any activities that must take place, or any conditions that must be true, before the use case can be started. Number each precondition. Examples:

1. User's identity has been authenticated.
2. User's computer has sufficient free memory available to launch task.

3.5. Postconditions

Describe the state of the system at the conclusion of the use case execution. Number each postcondition. Examples:

1. Document contains only valid SGML tags.
2. Price of item in database has been updated with new value.

3.6. Normal Flow

Provide a detailed description of the user actions and system responses that will take place during execution of the use case under normal, expected conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. This description may be written as an answer to the hypothetical question, "How do I <accomplish the task stated in the use case name>?" This is best done as a numbered list of actions performed by the actor, alternating with responses provided by the system. The normal flow is numbered "X.0", where "X" is the Use Case ID.

3.7. Alternative Flows

Document other, legitimate usage scenarios that can take place within this use case separately in this section. State the alternative flow, and describe any differences in the sequence of steps that take place. Number each alternative flow in the form "X.Y", where "X" is the Use Case ID and Y is a sequence number for the alternative flow. For example, "5.3" would indicate the third alternative flow for use case number 5.

3.8. Exceptions

Describe any anticipated error conditions that could occur during execution of the use case, and define how the system is to respond to those conditions. Also, describe how the system is to respond if the use case execution fails for some unanticipated reason. If the use case results in a durable state change in a database or the outside world, state whether the change is rolled back, completed correctly, partially completed with a known state, or left in an undetermined state as a result of the exception. Number each alternative flow in the form "X.Y.E.Z", where "X" is the Use Case ID, Y indicates the normal (0) or alternative (>0) flow during which this exception could take place, "E" indicates an exception, and "Z" is a sequence number for the exceptions. For example "5.0.E.2" would indicate the second exception for the normal flow for use case number 5.

3.9. Includes

List any other use cases that are included ("called") by this use case. Common functionality that appears in multiple use cases can be split out into a separate use case that is included by the ones that need that common functionality.

3.10. Priority

Indicate the relative priority of implementing the functionality required to allow this use case to be executed. The priority scheme used must be the same as that used in the software requirements specification.

3.11. Frequency of Use

Estimate the number of times this use case will be performed by the actors per some appropriate unit of time.

3.12. Business Rules

List any business rules that influence this use case.

3.13. Special Requirements

Identify any additional requirements, such as nonfunctional requirements, for the use case that may need to be addressed during design or implementation. These may include performance requirements or other quality attributes.

3.14. Assumptions

List any assumptions that were made in the analysis that led to accepting this use case into the product description and writing the use case description.

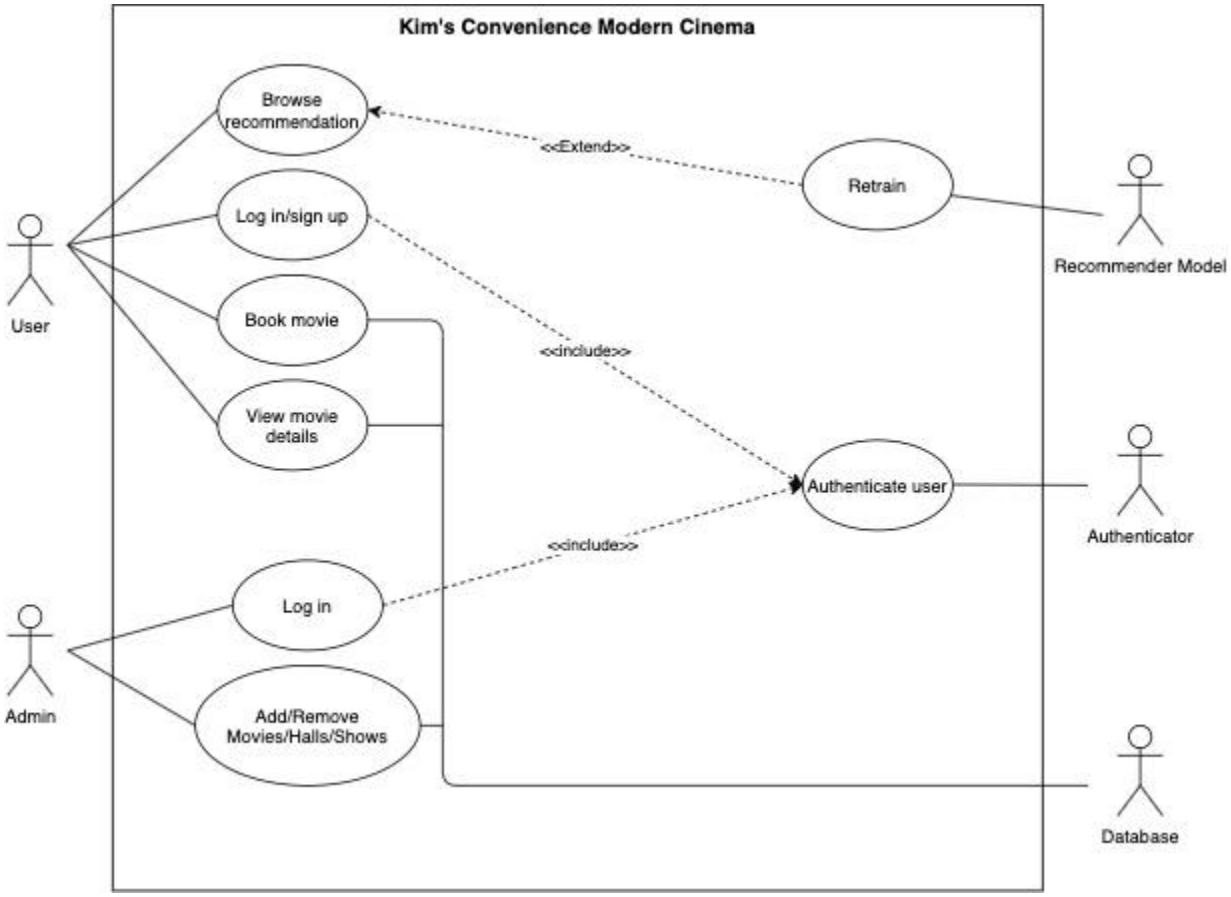
3.15. Notes and Issues

List any additional comments about this use case or any remaining open issues or TBDs (To Be Determineds) that must be resolved. Identify who will resolve each issue, the due date, and what the resolution ultimately is.

Use Case List

Primary Actor	***Use Cases***
User (Cinema Goer)	Log in/Sign up Book a Movie Browse Recommendation View Movie Details
Admin	Log in Add/Remove Movies/Halls/Shows
Model Admin	Nightly model retrain

Use Case Diagram



Use Case Template

Use Case ID:	1		
Use Case Name:	Log in/Sign up		
Created By:	Wong Ying	Last Updated By:	Wong Ying
Date Created:	29 Aug 2021	Date Last Updated:	30 Aug 2021

Actors:	User
Description:	User logs into Kim's Convenience Modern Cinema system.
Trigger:	User clicks on Login
Preconditions:	1. User has launched the KCMC website. 2. User has registered an account
Postconditions:	1. User has logged in.
Normal Flow:	1.0.1 User enter credentials (eg. username/email, password) 1.0.2 User click on "Login" 1.0.3 User successfully logged in and website shows "Welcome, User"
Alternative Flows:	AF-S2: If user has yet to sign up 1.1.1 User will be brought to Sign up Page 1.1.2 User enter credentials (eg. username/email, password) 1.1.3 User receive verification email/SMS 1.1.4 User is redirected to 1.0.3
Exceptions:	1.0.E.1 User forgotten password 1. User will be guided to follow steps to reset password 2. User will reset password 3. User is redirected to 1.0.1
Includes:	Authenticate User
Priority:	High
Frequency of Use:	Once per day
Business Rules:	N.A.
Special Requirements:	N.A.
Assumptions:	User has an email.
Notes and Issues:	N.A.

Use Case ID:	2		
Use Case Name:	Book a movie		
Created By:	Hussain	Last Updated By:	Hussain
Date Created:	29 Aug 2021	Date Last Updated:	30 Aug 2021

Actors:	User
Description:	User logs into Kim's Convenience Modern Cinema system.
Trigger:	User clicks on book ticket
Preconditions:	1. User has launched the KCMC website. 2. User has registered an account 3. User is logged in

Postconditions:	1. User has logged in.
Normal Flow:	1.0.1. User clicks on book ticket 1.0.2. List of movies is shown 1.0.3. User chooses movie they wish to book 1.0.4. List to shows is shown for their movie 1.0.5. User chooses a show 1.0.6. User chooses seats 1.0.7. Confirmation page is shown 1.0.8. User books a movie
Alternative Flows:	AF-S1: User clicks on book ticket below a movie on the home page 1.1.1. User is redirected to step 4 in the normal flow
Exceptions:	N.A
Includes:	Use case 1: Log in/Sign up
Priority:	High
Frequency of Use:	Once per day
Business Rules:	N.A
Special Requirements:	N.A
Assumptions:	User knows which movie they want to book
Notes and Issues:	N.A

Use Case ID:	3		
Use Case Name:	Browse recommendation		
Created By:	Jeremy	Last Updated By:	Jeremy
Date Created:	29 Aug 2021	Date Last Updated:	29 Aug 2021

Actors:	Recommender model
Description:	KCMC website will display recommended movies for each user
Trigger:	N.A
Preconditions:	1. User has launched the KCMC website. 2. User has registered an account 3. User has logged in. 4. User has at least one booking
Postconditions:	1. User will see a list of movies recommended for them
Normal Flow:	1.0.1 User log into KCMC website 1.0.2 If user has booking histories, they will see a list of movies recommended for them
Alternative Flows:	AF-S2: If user does not have booking histories: 1.1.1 User will not see any recommended movies until they have at least one booking
Exceptions:	N.A
Includes:	Use case 1: Log in/Sign up
Priority:	High
Frequency of Use:	Everytime website loads
Business Rules:	N.A
Special Requirements:	N.A
Assumptions:	Users with similar booking history will have similar movie preference in the future

Notes and Issues:	N.A

Use Case ID:	4		
Use Case Name:	View movie details		
Created By:	Ta Anh Duc	Last Updated By:	Ta Anh Duc
Date Created:	29 Aug 2021	Date Last Updated:	1 Sep 2021

Actors:	User
Description:	User can view details of movie like titles, genres, synopsis, current showing cinemas and showtime
Trigger:	User clicks on a cover or title of a movie in the list view
Preconditions:	1. User has launched the KCMC website. 2. User is at the home page
Postconditions:	1. User will see the details of the selected movie
Normal Flow:	4.0.1 User clicks on a movie cover or title 4.0.2 User can view the movie details
Alternative Flows:	AF-S1: If user is not at the home page 4.1.1 User clicks on Home tabs or KCMC logo 4.1.2 User will be brought to the home page 4.1.3 User clicks on a movie cover or title 4.1.4 User can view the movie details
Exceptions:	N.A
Includes:	N.A
Priority:	High
Frequency of Use:	Multiple times a day
Business Rules:	N.A
Special Requirements:	N.A
Assumptions:	Movies and their informations are already in the database
Notes and Issues:	N.A

Use Case ID:	5		
Use Case Name:	Admin login		
Created By:	Hussain	Last Updated By:	Hussain
Date Created:	29 Aug 2021	Date Last Updated:	30 Aug 2021

Actors:	Admin
Description:	Admin logs into Kim's Convenience Modern Cinema system.
Trigger:	N.A
Preconditions:	1. Admin has launched the KCMC website.
Postconditions:	1. Admin has logged in.
Normal Flow:	1.0.1. Admin clicks on admin login 1.0.2. Admin is redirected to admin login page 1.0.3. Admin enters login details 1.0.4. Admin is logged in

Alternative Flows:	N.A
Exceptions:	N.A
Includes:	N.A
Priority:	N.A
Frequency of Use:	Multiple times a week
Business Rules:	N.A
Special Requirements:	N.A
Assumptions:	Admin login details already exists created on the backend
Notes and Issues:	N.A

Use Case ID:	6		
Use Case Name:	Add Movies/Halls/Shows		
Created By:	Desmond	Last Updated By:	Desmond
Date Created:	31 Aug 2021	Date Last Updated:	31 Aug 2021

Actors:	Admin, Database
Description:	Admin users will be able to add new movies, halls and shows into the database to be displayed in the website
Trigger:	N.A
Preconditions:	1. Admin launched admin page of KCMC website 2. Admin's login credentials has been authenticated
Postconditions:	1. Database updated with the newly added movies, halls or shows
Normal Flow:	6.0.1. Admin clicks on add new movies 6.0.2. Admin inputs movie name 6.0.3. Admin uploads movie poster image 6.0.4. Admin clicks complete 6.0.5. Database is updated with new movie
Alternative Flows:	AF-S1: Admin clicks on add new halls 6.1.1. Admin inputs hall's name 6.1.2 Admin inputs venue of hall 6.1.3. Admin clicks complete 6.1.4. Database is updated with with new hall AF-S1: Admin clicks on add new shows 6.2.1. Admin inputs the timing 6.2.2. Admin clicks complete 6.2.3. Database is updated with new show
Exceptions:	6.0.E.1 Admin inputs a blank movie name 1. System will inform the admin that the movie name field is empty 2. System will prompt the admin to reenter the movie's name 3. Return to Step 2 of normal flow 6.0.E.2: Admin uploads an incorrect image format 1. System will inform admin that an incorrect image format has been uploaded 2. System will prompt the admin to reupload the movie poster image

	3. Return to Step 3 of normal flow 6.1.E.1: Admin inputs a blank hall name 1. System will inform the admin that the hall name field is empty 2. System will prompt the admin to reenter the hall's name 3. Return to Step 6.1 of alternative flow 1 6.1.E.2: Admin inputs a blank show timing 4. System will inform the admin that the show timing field is empty 5. System will prompt the admin to reenter the timing 6. Return to Step 6.1 of alternative flow 2
Includes:	Use Case 5: Admin login
Priority:	High
Frequency of Use:	Multiple times a week
Business Rules:	N.A
Special Requirements:	N.A
Assumptions:	Admin login account already exists in the database
Notes and Issues:	-

Use Case ID:	7		
Use Case Name:	Remove Movies/Halls/Shows		
Created By:	Desmond Yap	Last Updated By:	Desmond Yap
Date Created:	29 Aug 2021	Date Last Updated:	29 Aug 2021

Actors:	Admin, Database
Description:	Admin will be able to select existing movies, halls or shows in the database to be removed
Trigger:	N.A
Preconditions:	1. Admin launched admin page of KCMC website 2. Admin's login credentials has been authenticated
Postconditions:	1. Database updated with the selected movies, halls or shows removed
Normal Flow:	6.0.1. Admin clicks on remove new movies 6.0.2. Admin selects a movie from the list of movies currently in the database 6.0.3. Admin clicks complete 6.0.4. Database is updated with deleted movie
Alternative Flows:	AF-S1: Admin clicks on remove halls 6.1.1. Admin selects a hall from the list of halls currently in the database 6.1.2. Admin clicks complete 6.1.3. Database is updated with deleted hall AF-S1: Admin clicks on remove shows 6.2.1. Admin selects a show from the list of shows currently in the database 6.2.2. Admin clicks complete

	6.2.3. Database is updated with deleted show
Exceptions:	N.A
Includes:	Use Case 5: Admin login
Priority:	High
Frequency of Use:	Multiple times a week
Business Rules:	N.A
Special Requirements:	N.A
Assumptions:	1. Admin login account already exists in the database 2. Movies/Halls/Shows to be removed already exists in the database
Notes and Issues:	-

Use Case ID:	8		
Use Case Name:	Nightly model retrain		
Created By:	Jeremy	Last Updated By:	Jeremy
Date Created:	29 Aug 2021	Date Last Updated:	29 Aug 2021

Actors:	Recommender Model
Description:	Model will retrain every night to include new users data
Trigger:	Cinema closing time (2am)
Preconditions:	1. New users must have at least one booking
Postconditions:	1. New users will be included in the recommender model
Normal Flow:	1.0.1 New users sign up an account 1.0.2 New accounts added to database 1.0.3 Upon cinema closing time, recommender model retrain with new user data
Alternative Flows:	N.A
Exceptions:	N.A
Includes:	Use case 1: Log in/Sign up
Priority:	High
Frequency of Use:	Every night
Business Rules:	N.A
Special Requirements:	N.A
Assumptions:	N.A
Notes and Issues:	N.A