
Software Requirements Specification

for

Catch Movie

Version 1.0 approved

Prepared by SSP 8-Group 4

(Casuarina, Dobrasiewicz Piotr, Godeliva Petrina M, Tan Jian Wei, Zhang Yuhan)

Nanyang Technological University

November 2019

Table of Contents

1. Introduction.....	4
1.1 Purpose.....	4
1.2 Document Conventions.....	4
1.3 Intended Audience and Reading Suggestions.....	4
1.4 Product Scope.....	5
2. Overall Description.....	5
2.1 Product Perspective.....	5
2.2 Product Functions.....	5
2.3 User Classes and Characteristics.....	6
2.4 Operating Environment.....	6
2.5 Design and Implementation Constraints.....	7
2.6 User Documentation.....	7
2.7 Assumptions and Dependencies.....	7
3. External Interface Requirements.....	8
3.1 User Interfaces.....	8
3.2 Hardware Interfaces.....	15
3.3 Software Interfaces.....	15
3.4 Communications Interfaces.....	15
4. System Features.....	15
4.1 User Account Login.....	15
4.2 View Now Showing Info.....	16
4.3 View Upcoming Info.....	17
4.4 Book Movie Ticket.....	18
4.5 Search Movie.....	19
4.6 View Promotions.....	20
4.7 View Account Profile.....	20
4.8 View Support.....	21
4.9 Give App Review.....	22
5. Other Nonfunctional Requirements.....	22
5.1 Performance Requirements.....	22
5.2 Security Requirements.....	23
5.3 Software Quality Attributes.....	23
Appendix A: Data Dictionary.....	24
Appendix B: Functional Requirements.....	24
Appendix C: Use Case Diagram.....	28
Appendix D: Use Case Description.....	29
Appendix E: Class Diagram.....	40
Appendix F: Sequence Diagram.....	41
a. Login.....	41
b. ViewNowShowingInfo.....	42
c. ViewUpcomingInfo.....	43
d. BookMovieTicket.....	44
e. SearchMovie.....	45
f. ViewPromotions.....	46
g. ViewAccountProfile.....	46
h. ViewSupport.....	47
i. GiveAppReview.....	48
Appendix G: System Architecture.....	49
Appendix H: Dialog Map.....	49

Appendix I: Test Cases	50
a. Black Box Testing	50
b. White Box Testing.....	68

1. Introduction

1.1 Purpose

1.1.1 The purpose of this document is to present a detailed description of the Android application, “CatchMovie”.

1.1.2 “CatchMovie” aims to provide a platform for users to view and book movie tickets from various cinemas companies such as Cathay, Golden Village and Shaw. The Android application helps users to eliminate the trouble of having to view the movie schedules across different websites, hence providing convenient way to find showtime and cinema that best suits our needs.

1.2 Document Conventions

1.2.1 “Android Application” or “Application” or “System” refers to the “CatchMovie” application.

1.2.2 “Users” refers to the users of the “CatchMovie” application.

1.3 Intended Audience and Reading Suggestions

1.3.1 This Software Requirements Document is intended for:

- **Developers** to review the project design and implementation to allow for improvement and addition of new features. It sets the guideline for future development.
- **Project Managers** to oversee the whole project and making sure that the direction and scope is clear.
- **Users** to understand how the application works and how each feature is being used.
- **Testers** to use the document as a reference to design their testing strategy and locate bugs. With the help of this document, it becomes much more systematic and organized.
- **Marketing Staff** to reference the document and promote the application based on it’s features and usability.
- **Documentation Writers** to use as a baseline in referencing and creating a concise document for users.

	Introduction	Overall Description	External Interface Requirements	System Features	Other Non-Functional Requirements	Other Requirements
Developers	v	v	v	v	v	v
Project Managers	v	v	v	v	v	v
Users	v	v		v		
Testers	v	v	v	v	v	v
Marketing Staff	v	v		v	v	
Documentation Writers	v	v	v	v	v	v

1.4 Product Scope

1.4.1 “CatchMovie” is an Android application that provides a unified platform for users to get updated movie information across cinemas in Singapore: Cathay, Golden Village and Shaw.

1.4.2 The application uses publicly available data of movie showtimes obtained via web scraping from <https://www.popcorn.app/sg/>

1.4.3 The application’s simplicity and ease of use allow users to quickly search and compare different movie showtimes across cinemas. Subsequently, it allows users to book movie tickets by redirecting users to the corresponding official cinema website. Additionally, it also allows users to get information on upcoming movies and ongoing promotions across the different cinema platforms.

2. Overall Description

2.1 Product Perspective

2.1.1 This application is developed to eliminate the trouble of having to view the movie schedules across different cinema websites. With the help of publicly available movie data, we have designed “CatchMovie” for citizens to utilize in their daily life. The brief scope of the project can be found in section 1.4.

2.2 Product Functions

2.2.1 Users must be able to login with their gmail account to access all features of this application.

2.2.2 Users must be able to query the system for getting movie information of Now Showing movies compiled across different cinemas in Singapore.

2.2.3 Users must be able to query the system for getting movie information of Upcoming movies in Singapore.

2.2.4 Users must be able to do direct search of movie information by providing movie title and movie date as input.

2.2.5 Users must be directed by the system to the official cinema website for booking of movie tickets.

2.2.6 Users must be able to view ongoing promotions across different cinemas in Singapore.

2.2.7 Users must be able to view his/her profile details based on the gmail account used for login

2.2.8 Users must be able to view support/basic FAQ regarding on how to use the application.

2.2.9 Users must be able to submit a rating for the application.

2.3 User Classes and Characteristics

2.3.1 Users

2.3.1.1 The targeted audience of “CatchMovie” are mainly Singaporeans or PRs.

2.3.1.2 The application is to be used throughout the year.

2.3.1.3 The design of the application is meant to be easy and simple for users to understand and use.

2.3.1.4 It is not expected for users to be technically savvy to fully utilize the application.

2.3.2 Developers

2.3.2.1 The developers are responsible for the maintenance of the “CatchMovie” application.

2.3.2.2 The developers are given full access control for the system.

2.3.2.3 The developers are expected to be technically savvy and well-versed in software development to ensure the application is properly maintained.

2.4 Operating Environment

The application “Catch Movie” will operate on mobile phones or tablets with Android operating systems and internet connection. The application is developed using Android Studio. Firebase is used for verification of users’ credentials and storing of users’ data. Additionally, myJSON storage is used to store movie details and showtimes.

2.4.1 IDE

- 1) Android Studio Version 3.5.1

2.4.2 Programming Language

- 1) Java Version 11.0.3

2.4.3 Operating System

- 1) Android OS 9 (Pie)

2.4.4 Web Scraper

- 1) Web Scraper from Chrome Extension Version 0.4.2

2.4.5 Database

- 1) Firebase Realtime Database
- 2) myJSON

2.4.6 User Credentials Authentication

- 1) Google Firebase Authentication

2.5 Design and Implementation Constraints

2.5.1 Due to limitation of the web scraper used, we would only scrape movie information for current day. Therefore, to overcome the constraint we have set up the web scraper to automatically run on a daily basis.

2.5.2 We are only able to retrieve the scraping results as csv files. Hence, we converted the csv files to JSON array and utilized simple JSON storage from *myjson.com*. The application will make HTTP request for fetching the JSON array response body from the url generated by *myjson.com*.

2.5.3 Due to security issues on payment verification, we were not able to implement movie ticket bookings and payments inside “CatchMovie” application itself. Therefore, the application will direct users to official website to proceed with bookings and payments upon choosing the most suitable showtime and cinema.

2.6 User Documentation

2.6.1 The application will be delivered with the UML diagrams, use case descriptions and video demo showing the functionality of the application.

2.7 Assumptions and Dependencies

2.7.1 Assumptions

- 2.7.1.1 The public data is always available for scraping on <https://www.popcorn.app/sg/>
- 2.7.1.2 The data from <https://www.popcorn.app/sg/> is always accurate and reliable.
- 2.7.1.3 Web Scraper is always running and authorized to scrape <https://www.popcorn.app/sg/>
- 2.7.1.4 The Firebase Authentication and Firebase Realtime Database is always up and running.

2.7.2 Dependencies

- 2.7.2.1 The application must rely on the available data from <https://www.popcorn.app/sg/>
- 2.7.2.2 The application is dependent on Web Scraper from Chrome Extension to retrieve movie information data
- 2.7.2.3 The application is dependent on myJSON to store and retrieve movie information data
- 2.7.2.4 The application is dependent on Firebase Authentication to verify users’ login.
- 2.7.2.5 The application is dependent on Firebase Realtime Database to store ratings submitted by user

3. External Interface Requirements

3.1 User Interfaces

3.1.1 The LoginUI interface is set to be the default main page upon launching as user is required to login with their account before accessing features of the application.

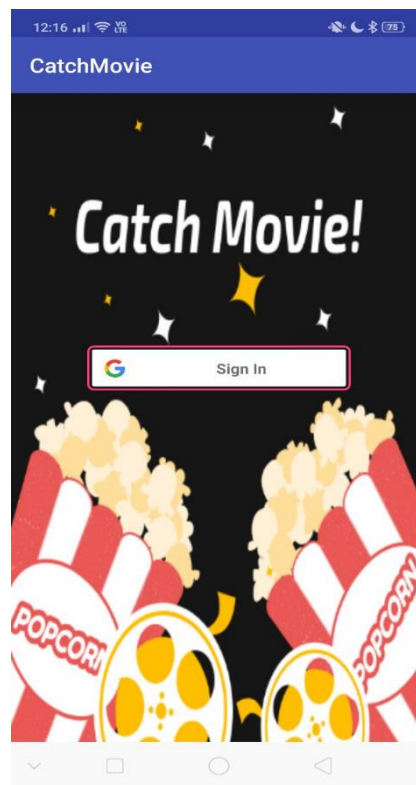
3.1.2 All interfaces aside from LoginUI have 4 tabs displayed at the bottom of the screen for providing navigation to different pages of the application.

3.1.3 Every interface shares standardized color schemes and layouts that does not change regardless of the device's dimension or resolution.

3.1.4 Error messages if any will be displayed on the same interface of the page's input.

3.1.5 LoginUI

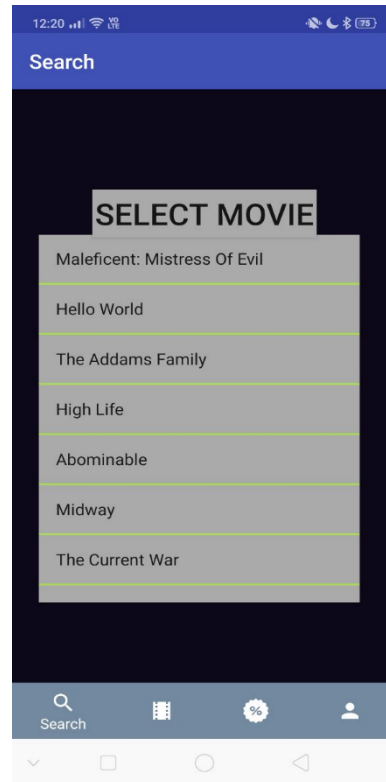
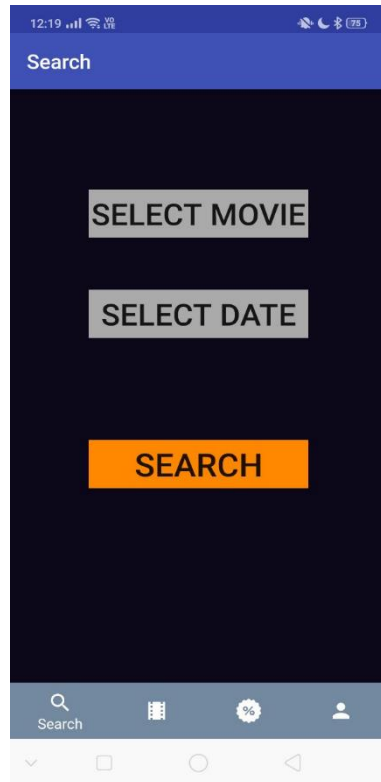
Users may click on the login button to login with their valid gmail account through this interface. Login information must be verified subsequently by Firebase Authentication.



3.1.6 SearchUI

This interface shall allow users to do direct search of movie information by providing movie title and movie date as input.

The system must direct the user to the selected movie interface and show detailed information about that movie.



3.1.7 NowShowingUI

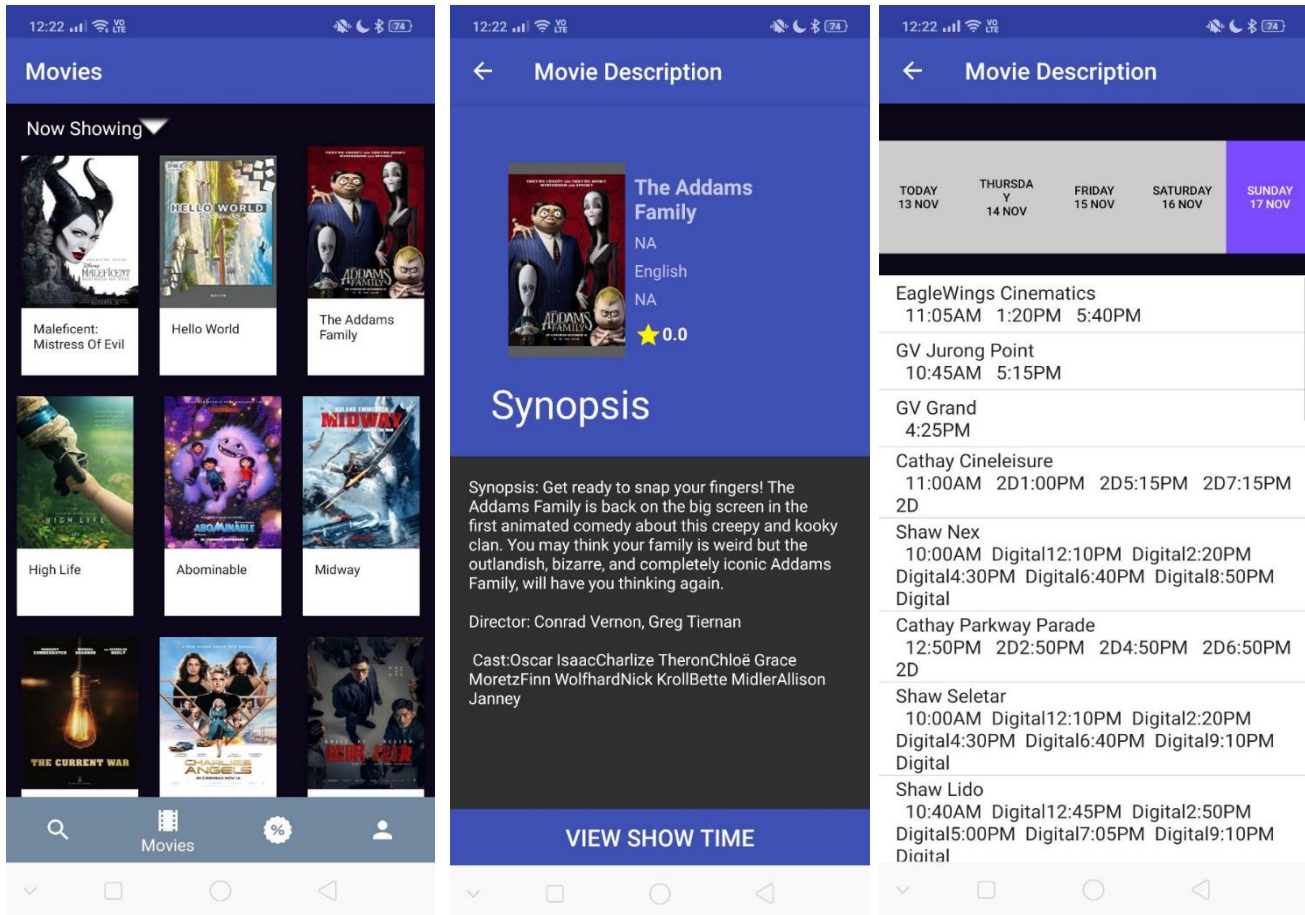
This interface shall display general information of Now Showing movies consisting of movie posters and titles.

Users may click on the movie posters/titles.

The system must direct the user to the selected movie interface and display detailed information about that movie.

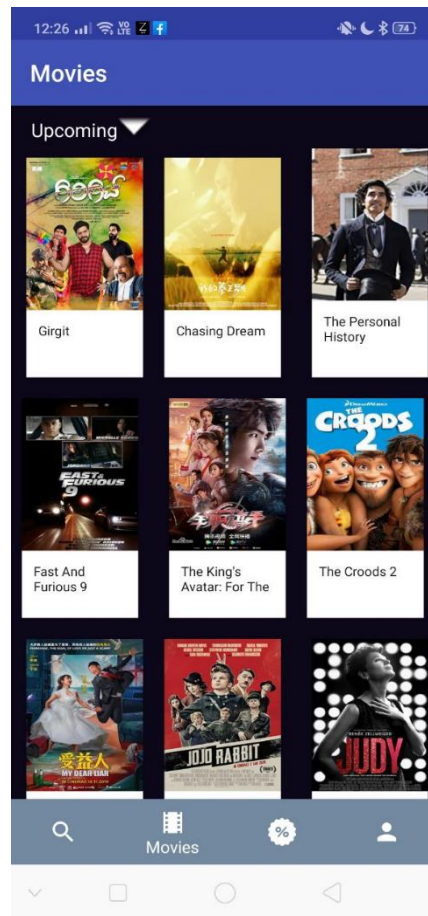
Users may click book movie ticket button for a particular showtime.

The system must direct the user to the official cinema booking page for that particular showtime.



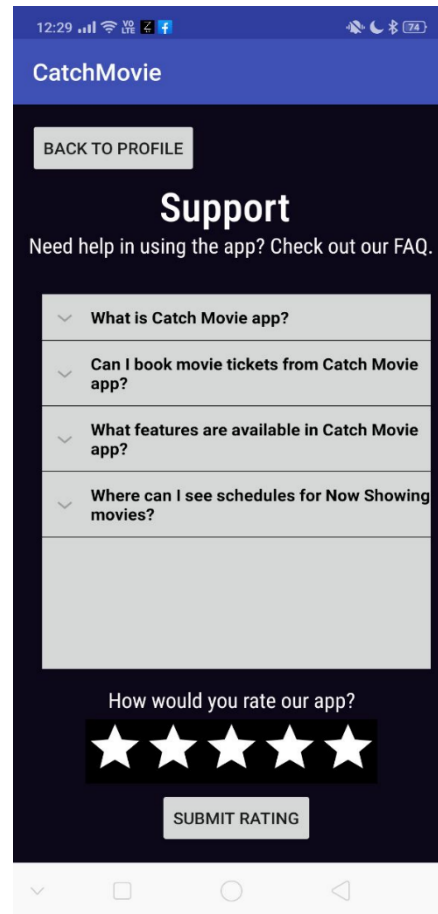
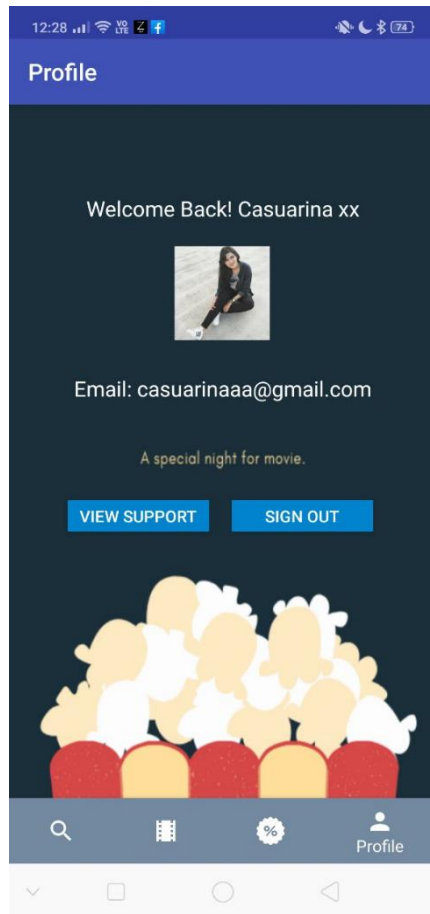
3.1.8 UpcomingUI

This interface shall display general information of Upcoming movies consisting of movie posters and titles.



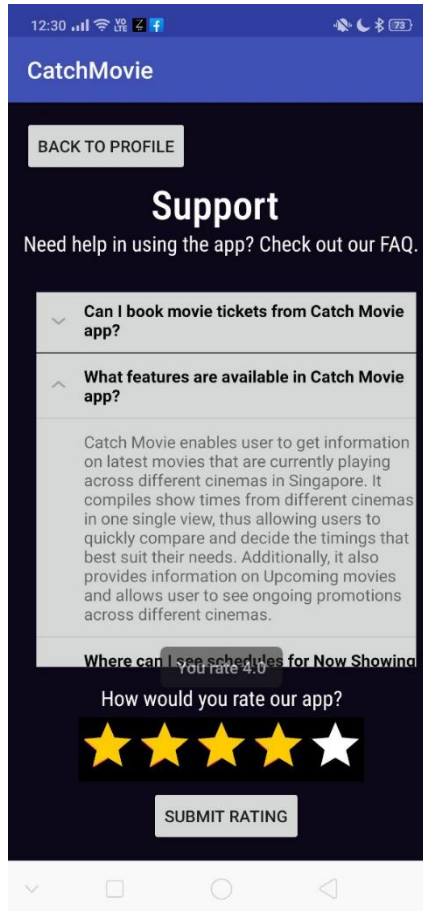
3.1.9 ProfileUI

This interface shall display general information of user's profile consisting of profile picture, first name, last name. User may click View Support button and the system must direct user to the Support page (SupportUI). User may click Sign Out button and the system must direct user to LoginUI



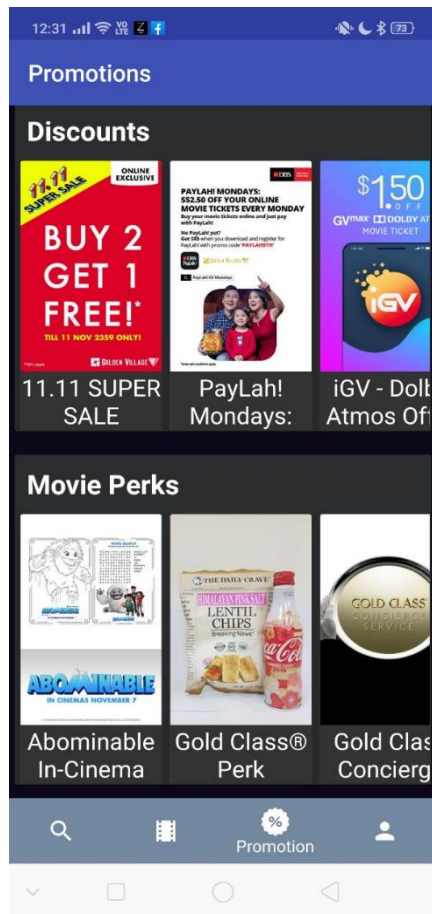
3.1.10 SupportUI

This interface shall display basic FAQ about how to use the application. User may rate the application by clicking the stars and submit button. System shall store rating in Firebase Realtime Database



3.1.11 PromotionsUI

This interface shall display ongoing promotions across different cinemas in Singapore. Users may click on the promotions posters. The system must direct user to the selected promotion page and display detailed information about that promotion.



3.2 Hardware Interfaces

CatchMovie application will be deployed and built on android devices to allow user to locate their movies without any hassle.

3.3 Software Interfaces

3.3.1 The system shall use the Web Scraper from Chrome Extension to pull all relevant information needed by the application.

3.3.2 The system shall use myJSON to store movie information data and retrieve movie information via HTTP request of JSON array.

3.3.3 The system shall use Firebase Authentication to verify user credentials in login phase.

3.3.4 The system shall use Firebase Realtime Database to store ratings given by user.

3.4 Communications Interfaces

3.4.1 The system shall use the HTTP protocol for communication over the internet

4. System Features

4.1 User Account Login

4.1.1 Description and Priority

Users shall be able to login with their gmail account to access features offered by the application. This feature has high priority.

4.1.2 Stimulus/Response Sequences

4.1.2.1 The precondition of this feature will be that user should have a registered gmail account in order to login successfully.

4.1.2.2 The user clicks the Login button at LoginUI

4.1.2.3 The system asks the user to fill his/her gmail address and password

4.1.2.4 The system calls Firebase Authentication to verify user credentials

4.1.2.4.1 If the credential is successfully verified, the system brings user into the main functionalities of the application and show ProfileUI

4.1.2.4.2 If the credential is not successfully verified, the system displays “Failed” message and bring users back to LoginUI

4.1.3 Functional Requirements

REQ-1: The system must allow users to enter the system by displaying a login page upon starting the application.

REQ-2: Users must be able to click the login button in UI and enter their email address and password.

REQ-3: The system must be able to validate user credentials through Firebase Authentication

REQ-3.1 If authentication via Firebase is successful, the system must directs user to ProfileUI

REQ-3.2 If authentication via Firebase is not successful, the system must display “Failed” message and directs user back to LoginUI

4.2 View Now Showing Info

4.2.1 Description and Priority

Users shall be able to view detailed information of the Now Showing movies. This feature has high priority.

4.2.2 Stimulus/Response Sequences

4.1.2.1 The precondition of this feature will be that user should user must have login to the system.

4.1.2.2 The user clicks the Now Showing Movie button at bottom navigation bar

4.1.2.3 The system directs user to NowShowingUI which displays movie posters and titles of Now Showing movies

4.1.2.4 The user may click on any of the movie posters or titles to view detailed information of that movie

4.1.2.5 The system directs user to the selected movie interface and display detailed information about that movie.

4.2.3 Functional Requirements

REQ-1: The system must allow users to view Now Showing movies anytime by clicking the Now Showing Movie button at bottom navigation bar

REQ-2: The system must be able to direct user to NowShowingUI and display movie information consisting of:

REQ-2.1: Now Showing movies posters

REQ-2.2: Now Showing movies titles

REQ-3: User must be able to click on any of the movie posters or titles to get detailed information of the particular movie

REQ-3.1: The system must be able to direct user to selected movie page containing detailed information:

REQ-3.1.1 Movie title

- REQ-3.1.2 Movie poster
- REQ-3.1.3 Movie rating
- REQ-3.1.4 Language
- REQ-3.1.5 Synopsis
- REQ-3.1.6 Directors and casts
- REQ-3.1.7 Duration
- REQ-3.1.8 The system should display NA if the data is not available online

REQ-3.2: The system must allow user to view showtimes across different cinemas for the selected movie.

REQ-3.2.1 The system must be able to display showtimes for today and next 4 days whenever the data is available.

REQ-3.2.2 The showtime details must include cinema name and movie format whenever the data is available.

4.3 View Upcoming Info

4.3.1 Description and Priority

Users shall be able to view information of the Upcoming movies. This feature has high priority.

4.3.2 Stimulus/Response Sequences

4.3.2.1 The precondition of this feature will be that user should user must have login to the system.

4.3.2.2 The user clicks the Upcoming Movie button at bottom navigation bar

4.3.2.3 The system directs user to UpcomingUI which displays movie posters and titles of Upcoming movies

4.3.2.4 The user may click on any of the movie posters or titles to view detailed information of that movie

4.3.2.5 The system directs user to the selected movie interface and display detailed information about that movie.

4.3.3 Functional Requirements

REQ-1: The system must allow users to view Upcoming movies anytime by clicking the Upcoming Movie button at bottom navigation bar

REQ-2: The system must be able to direct user to UpcomingUI upon user request and display movie information consisting of:

REQ-2.1: Upcoming movies posters

REQ-2.2: Upcoming movies titles

REQ-3: User must be able to click on any of the movie posters or titles to get detailed information of the particular movie

REQ-3: The system must be able to direct user to selected movie page containing detailed information:

REQ-3.1 Movie title

REQ-3.2 Movie poster

REQ-3.3 Movie rating

REQ-3.4 Language

REQ-3.5 Synopsis

REQ-3.6 Directors and casts

REQ-3.7 Duration

REQ-3.8 The system should display NA if the data is not available online

4.4 Book Movie Ticket

4.4.1 Description and Priority

Users shall be able to book movie tickets from official cinema webpage. This feature has high priority.

4.4.2 Stimulus/Response Sequences

4.4.2.1 The precondition of this feature is that this use case extends View Now Showing Info. It is invoked whenever the user decides to book movie tickets after viewing the showtimes for a particular movie.

4.4.2.2 For the selected movie, the system displays detailed showtimes across different cinemas as described in View Now Showing info.

4.4.2.3 The user may click the book ticket button for his corresponding choice of movie showtime and cinema

4.4.2.4 The system directs user to the booking page of the official cinema website

4.4.3 Functional Requirements

REQ-1: The system must allow users to click on a particular showtime and cinema for the selected movie

REQ-2: The system must be able to direct users to the booking page of the official cinema website

REQ-2.1: If the cinema is GV, the system must direct users to <https://www.gv.com.sg/GVBuyTickets#/>

REQ-2.2: If the cinema is Cathay, the system must direct users to <https://www.cathaycinplexes.com.sg/showtimes.aspx>

REQ-2.3: If the cinema is Shaw, the system must direct users to <https://shaw.sg/Showtimes/>

REQ-2.4: If the cinema is Filmgarde, the system must direct users to <https://fgcineplex.com.sg/buyticket>

4.5 Search Movie

4.5.1 Description and Priority

Users shall be able to do direct search for movies by inputting a movie title. This feature has high priority.

4.5.2 Stimulus/Response Sequences

4.5.2.1 The precondition of this feature will be that user should user must have login to the system.

4.5.2.2 The user clicks the Search button at bottom navigation bar

4.5.2.3 The system directs user to SearchUI which displays a dropdown list of available movie titles

4.5.2.4 The user may choose a movie from the dropdown list

4.5.2.5 The system directs user to the selected movie page and displays detailed information for the movie

4.5.3 Functional Requirements

REQ-1: The system must allow users to initiate search options by clicking the Search button at bottom navigation bar

REQ-2: The system must be able to direct users to SearchUI and displays a dropdown list of available movie titles

REQ-3: The user must be able to choose any movie from the dropdown list

REQ-4: The system must be able to display a dropdown list of dates from today to next 4 days

REQ-5: The user must be able to choose any date from the dropdown list

REQ-6: The system must be able to direct user to the selected movie details page or movie showtime page depending on users' input:

REQ-6.1: If the user inputs movie title only, the system must direct user to the movie details page for the selected movie.

REQ-6.2: If the user inputs both movie title and date, the system must

direct user to the movie showtimes page for the selected movie and date,

REQ-6.3: If the user inputs date only, the system should not direct user to any page.

4.6 View Promotions

4.6.1 Description and Priority

Users shall be able to view current promotions across different cinemas. This feature has medium priority.

4.6.2 Stimulus/Response Sequences

4.6.2.1 The precondition of this feature will be that user should user must have login to the system.

4.6.2.2 The user clicks the Promotions button at bottom navigation bar

4.6.2.3 The system directs user to PromotionsUI which displays ongoing promotions across different cinemas

4.6.3 Functional Requirements

REQ-1: The system must allow users to view promotions by clicking the Promotions button at bottom navigation bar

REQ-2: The system must direct users to PromotionsUI which displays ongoing promotions' poster and names

REQ-3: The user may click on any of the promotions posters or titles to view detailed information of that promotion

REQ-4: The system directs user to the selected promotion interface and display detailed information about that promotion consisting of:

REQ-4.1: Promotion name

REQ-4.2: Promotion description

REQ-4.3: Promotion poster

4.7 View Account Profile

4.7.1 Description and Priority

Users shall be able to users to view his/her profile details based on the gmail account used for login. This feature has high priority.

4.7.2 Stimulus/Response Sequences

4.7.2.1 The precondition of this feature will be that user should user must have login to the system.

4.7.2.2 The user clicks the Profile button at bottom navigation bar

4.7.2.3 The system directs user to ProfileUI which displays his/her profile details extracted from gmail account used for login

4.7.3 Functional Requirements

REQ-1: The system must allow users to view his/her profile by clicking the Profile button at bottom navigation bar

REQ-2: The system must be able to direct users to ProfileUI and display profile details consisting of:

REQ-2.1: First name

REQ-2.2: Last name

REQ-2.3: Profile picture

REQ-2.4: Email address

REQ-3: The system must allow users to exit/log out of the system by clicking Sign Out button and direct back to LoginUI

4.8 View Support

4.8.1 Description and Priority

Users shall be to view help/support regarding how to use the app. This feature has medium priority.

4.8.2 Stimulus/Response Sequences

4.8.2.1 The precondition of this feature will be that user should user must have login to the system.

4.8.2.2 The user clicks the View Support button at the ProfileUI

4.8.2.3 The system directs user to SupportUI which displays lists of frequently asked questions about using the application

4.8.2.4 The user may click on any of the questions inside the list

4.8.2.5 The system expands the list and displays answer to the selected question

4.8.3 Functional Requirements

REQ-1: The system must allow users to view support by clicking View Support button from ProfileUI

REQ-2: The system must be able to direct users to SupportUI and displays list of questions

REQ-2.1: The user must be able to click on any of the questions inside the list

REQ-2.2: The system must be able to expand the list and display answer to the selected question

REQ-3: The system must be able to direct user back to ProfileUI anytime by clicking Back to Profile button

4.9 Give App Review

4.9.1 Description and Priority

Users shall be able to give ratings for the application. This feature has low priority.

4.9.2 Stimulus/Response Sequences

4.9.2.1 The precondition of this feature is that it extends View Support. It is invoked whenever the user decides to give rating for the application inside SupportUI.

4.9.2.2 The user gives rating by clicking the stars scale in SupportUI

4.9.2.3 The system displays acknowledgement message indicating the rating that the user has just inputted

4.9.2.4 The user clicks the submit button

4.9.2.5 The system displays confirmation message that rating is submitted and directs user back to ProfileUI

4.9.2.6 The system stores user ratings in Firebase Realtime Database

4.9.3 Functional Requirements

REQ-1: The system must allow user to give rating by clicking a scale from 1-5

REQ-2: The system must be able to display the rating that the user has just inputted

REQ-3: The system must allow user to submit his/her rating by clicking a submit button

REQ-3.1: The system must be able to show confirmation message upon rating submission

REQ-3.2: The system must be able to direct user back to ProfileUI upon rating submission

REQ-3.3: The system must be able to store ratings given by user in Firebase Realtime Database

5. Other Nonfunctional Requirements

5.1 Performance Requirements

5.1.1 The application must be able to completely load each of its internal UI within 5 seconds or less.

5.1.2 The application shall be able to return user input/query within 5 second or less.

5.1.3 The system must be able to redirect to respective cinema's payment page within 5 seconds

5.2 Security Requirements

5.3.1 User credentials for login must be verified

5.3 Software Quality Attributes

5.3.1 Usability

5.4.1.1 The user must be able to get to a page within 5 clicks from any page.

5.4.1.2 Every text of this program (including error text messages) shall be written in simple English language.

5.4.1.3 The application must permit easy reversal actions between pages

5.4.1.4 The application shall indicate to user where they are at in the process

5.4.1.5 The layout of the application must be consistent throughout

5.4.1.6 The features of the application shall be self-explanatory and intuitive

5.4.1.7 The user should not have to remember more than 5 information

5.3.2 Reliability

5.4.2.1 User's account details displayed in ProfileUI must match with actual details in user's gmail account

5.4.2.2 The movie information and promotions data must reflect daily changes

5.3.3 Supportability

5.4.3.1 A maintenance developer shall be able to rectify any bugs within the system in less than 24 hours.

Appendix A: Data Dictionary

Term	Definition
Now Showing	Movies that are currently playing in Singapore
Upcoming	Movies that are going to be released in later date

Appendix B: Functional Requirements

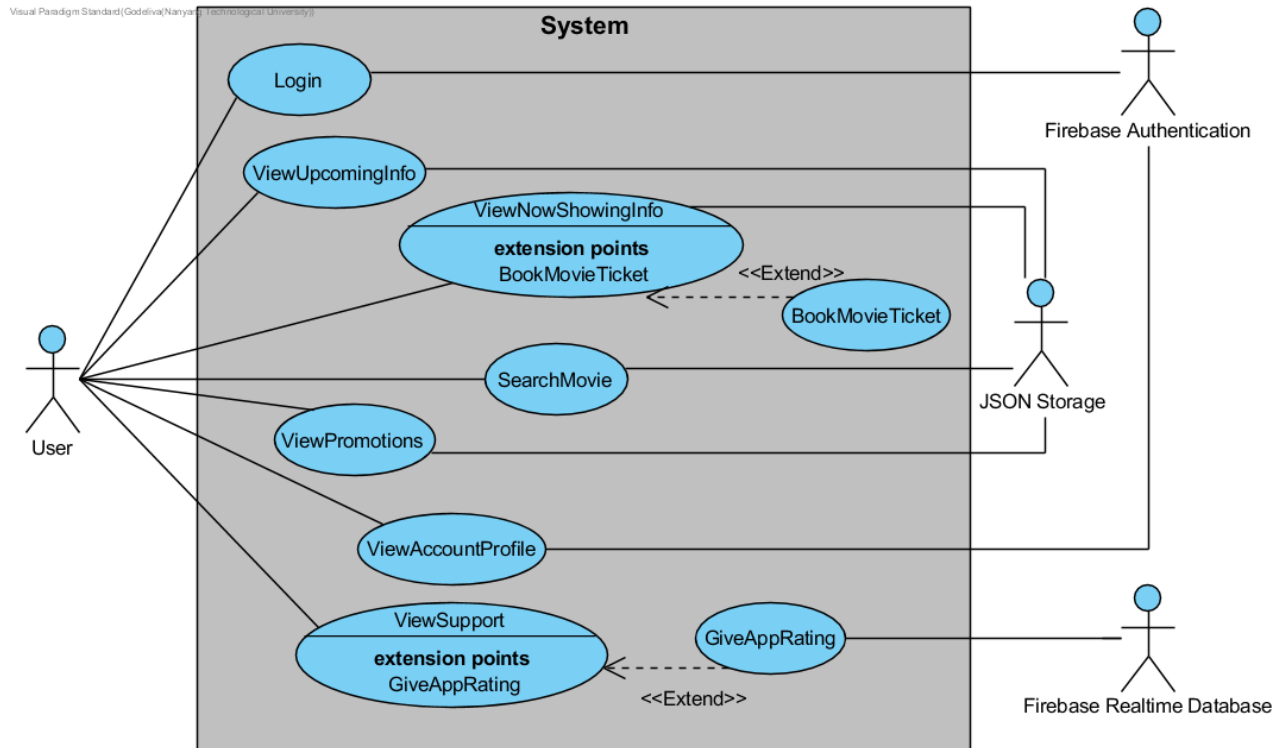
1. The system must allow users to enter the system by displaying a login page upon starting the application.
 - 1.1. Users must be able to click the login button in UI and enter their email address and password.
 - 1.2. The system must be able to validate user credentials through Firebase Authentication
 - 1.2.1. If authentication via Firebase is successful, the system must direct user to ProfileUI
 - 1.2.2. If authentication via Firebase is not successful, the system must display “Failed” message and direct user back to LoginUI
2. The system must allow users to query the system for getting movie information.
 - 2.1. Users must be able to query the system for ‘Now showing’ movies.
 - 2.1.1. The system must allow users to view Now Showing movies by clicking the Now Showing Movie button at bottom navigation bar
 - 2.1.2. The system must be able to direct user to NowShowingUI and display movie information consisting of:
 - 2.1.2.1. Now Showing movies posters
 - 2.1.2.2. Now Showing movies titles
 - 2.1.3. User must be able to click on any of the movie posters or titles to get detailed information of the particular movie
 - 2.1.3.1. The system must be able to direct user to selected movie page containing detailed information:
 - 2.1.3.1.1. Movie title
 - 2.1.3.1.2. Movie poster
 - 2.1.3.1.3. Movie rating
 - 2.1.3.1.4. Language
 - 2.1.3.1.5. Synopsis
 - 2.1.3.1.6. Directors and casts
 - 2.1.3.1.7. Duration

- 2.1.3.1.8. The system must display NA if no information is available online
 - 2.1.3.2. The system must allow user to view showtimes across different cinemas for the selected movie
 - 2.1.3.2.1. The system must be able to display showtimes for today and next 4 days whenever the data is available.
 - 2.1.3.2.2. The showtime details must include cinema name and movie format whenever the data is available.
- 2.2. Users must be able to query the system for upcoming movies.
 - 2.2.1. The system must allow users to view Upcoming movies by clicking the Upcoming Movie button at bottom navigation bar
 - 2.2.2. The system must be able to direct user to UpcomingUI and display movie information consisting of:
 - 2.2.2.1. Upcoming movies posters
 - 2.2.2.2. Upcoming movies titles
 - 2.2.3. User must be able to click on any of the movie posters or titles to get detailed information of the particular movie
 - 2.2.4. The system must be able to direct user to selected movie page containing detailed information:
 - 2.2.4.1. Movie title
 - 2.2.4.2. Movie poster
 - 2.2.4.3. Movie rating
 - 2.2.4.4. Language
 - 2.2.4.5. Synopsis
 - 2.2.4.6. Directors and casts
 - 2.2.4.7. Duration
 - 2.2.4.8. The system must display NA if no information is available online
- 3. The system must allow users to book movie tickets
 - 3.1. The system must allow users to select a particular showtime to proceed with tickets bookings from the official website
 - 3.2. The system must be able to direct users to the official cinema website
 - 3.2.1. If the cinema is GV, the system must direct users to <https://www.gv.com.sg/GVBuyTickets#/>
 - 3.2.2. If the cinema is Cathay, the system must direct users to <https://www.cathaycineplexes.com.sg/showtimes.aspx>
 - 3.2.3. If the cinema is Shaw, the system must direct users to <https://shaw.sg/Showtimes/>

- 3.2.4. If the cinema is Filmgarde, the system must direct users
<https://fgcineplex.com.sg/buyticket>
- 4. The system must allow users to do direct search for movie information
 - 4.1. The system must allow user to initiate search options by clicking the Search button at bottom navigation bar
 - 4.2. The system must be able to direct users to SearchUI and displays a dropdown list of available movie titles
 - 4.3. The user must be able to choose any movie from the dropdown list
 - 4.4. The system must be able to display a dropdown list of dates from today to next 4 days
 - 4.5. The user must be able to choose any date from the dropdown list
 - 4.6. The system must be able to direct user to the selected movie details page or movie showtime page depending on users' input:
 - 4.6.1. If the user inputs movie title only, the system must direct user to the movie details page for the selected movie.
 - 4.6.2. If the user inputs both movie title and date, the system must direct user to the movie showtimes page for the selected movie and date,
 - 4.6.3. If the user inputs date only, the system must not direct user to any page.
- 5. The system must allow users to view ongoing promotions
 - 5.1. The users must be able to view promotions by clicking the Promotions button at bottom navigation bar
 - 5.2. The system must be able to direct users to PromotionsUI which displays ongoing promotions' poster and names
 - 5.3. The user must be able to click on any of the promotions posters or names to view detailed information of that promotion
 - 5.4. The system must direct user to the selected promotion interface and display detailed information about that promotion consisting of:
 - 5.4.1. Promotion name
 - 5.4.2. Promotion description
 - 5.4.3. Promotion poster
- 6. The system must allow users to view his/her profile details based on the gmail account used for login
 - 6.1. The user must be able to view his/her profile by clicking the Profile button at bottom navigation bar
 - 6.2. The system must be able to direct users to ProfileUI and display profile details consisting of:
 - 6.2.1. First name

- 6.2.2. Last name
 - 6.2.3. Profile picture
 - 6.2.4. Email address
- 7. The system must allow users to view support/help regarding how to use the application
 - 7.1. The user must be able to view support by clicking View Support button from ProfileUI
 - 7.2. The system must be able to direct user to SupportUI and displays list of questions
 - 7.2.1. The user must be able to click on any of the questions inside the list
 - 7.2.2. The system must be able to expand the list and display answer to the selected question
 - 7.3. The system must be able to direct user back to ProfileUI anytime by clicking Back to Profile button
 - 8. The system must allow users to give ratings for the application.
 - 8.1. The system must allow user to give rating by clicking a scale from 1-5
 - 8.2. The system must be able to display the rating that the user has just inputted
 - 8.3. The system must allow user to submit his/her rating by clicking a submit button
 - 8.3.1. The system must be able to show confirmation message upon rating submission
 - 8.3.2. The system must be able to direct user back to ProfileUI upon rating submission
 - 8.3.3. The system must be able to store ratings given by user in Firebase Realtime Database
 - 9. The system must allow users to exit/log out of the system by clicking Sign Out button

Appendix C: Use Case Diagram



Appendix D: Use Case Description

Use Case Template

Use Case ID:	1
Use Case Name:	Login

Actor:	User, Firebase Authentication (Participating Actor)
Description:	Existing user login to the system
Preconditions:	User should have a registered gmail account in order to login successfully.
Postconditions:	The system must bring user into main functionalities of the application and displays ProfileUI
Priority:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The user clicks the Login button at LoginUI 2. The system asks the user to fill his/her gmail address and password 3. The system calls Firebase Authentication to verify user credentials 4. The system brings user to ProfileUI
Alternative Flows:	AF-S4:

	<ol style="list-style-type: none"> 1. If the credential is invalid, , the system displays “Failed” message 2. The system returns to Step 1
--	--

Use Case Template

Use Case ID:	2
Use Case Name:	ViewNowShowingInfo

Actor:	User
Description:	User to view detailed information of the now showing movies
Preconditions:	The user must have login to the system using the Login use case
Postconditions:	The system must display detailed information of the now showing movie depending on user's choice
Priority:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The user clicks the Movie button at bottom navigation bar 2. The system directs user to NowShowingUI which displays movie posters and titles of Now Showing movies

	<ol style="list-style-type: none"> 3. The user may click on any of the movie posters or titles to view detailed information of that movie. 4. The system directs user to direct user to the selected movie interface and display detailed information about that movie. 5. The user may click on View Showtimes button to view showtimes for that movie 6. The system directs user to showtimes page and display showtimes across different cinemas in Singapore for today and next 4 days
--	--

Use Case Template

Use Case ID:	3
Use Case Name:	ViewUpcomingInfo

Actor:	User
Description:	User to view information of the upcoming movies
Preconditions:	The user must have login to the system using the Login use case
Postconditions:	The system must display information of the upcoming movies

Priority:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The user clicks the Upcoming Movie button at bottom navigation bar 2. The system directs user to UpcomingUI which displays movie posters and titles of Upcoming movies 3. The user may click on any of the movie posters or titles to view detailed information of that movie. 4. The system directs user to direct user to the selected movie interface and display detailed information about that movie.

Use Case Template

Use Case ID:	4
Use Case Name:	BookMovieTicket

Actor:	User
Description:	This functionality allows user to book movie tickets from official cinema webpage
Preconditions:	This use case extends ViewNowShowingInfo use case. It is invoked whenever the user decides to book movie tickets after viewing the showtimes.

Postconditions:	The system must direct users to the booking page of the official cinema website
Priority:	High
Flow of Events:	<ol style="list-style-type: none"> 1. For the selected movie, the system displays detailed showtimes across different cinemas as described in ViewNowShowingInfo use case. 2. The user may click the book ticket button for his corresponding choice of movie showtime and cinema. 3. The system directs user to the booking page of the official cinema website depending on the chosen cinema name
Alternative Flows:	<p>AF-S3 :</p> <ol style="list-style-type: none"> 1. If the cinema is GV, the system must direct users to https://www.gv.com.sg/GVBuyTickets#/ 2. If the cinema is Cathay, the system must direct users to https://www.cathaycinplexes.com.sg/showtimes.aspx 3. If the cinema is Shaw, the system must direct users to https://shaw.sg/Showtimes/ 4. If the cinema is Filmgarde, the system must direct users to https://fgcineplex.com.sg/buyticket

Use Case Template

Use Case ID:	5
Use Case Name:	SearchMovie

Actor:	User
Description:	This functionality allows user to do direct search for movies by inputting a movie title
Preconditions:	The user must have login to the system using the Login use case
Postconditions:	System displays detailed information for the selected movie depending on user's input
Priority:	High
Flow of Events:	<ol style="list-style-type: none">1. The user clicks the Search button at bottom navigation bar2. The system directs user to SearchUI which displays a dropdown list of available movie titles and a dropdown list of dates from today to next 4 days3. The user may choose any movie from the dropdown list4. The user may choose any date from the dropdown list

	5. The system directs user to the movie showtime page and displays showtimes for the selected movie and selected date
Alternative Flows:	<p>AF-S5 :</p> <ol style="list-style-type: none"> 1. If the user chooses movie title only, the system directs user to the movie details page for the selected movie. The user may proceed to view showtimes by clicking the View Showtimes button. 2. If the user chooses date only, the system should not direct user to any page.

Use Case Template

Use Case ID:	6
Use Case Name:	ViewPromotions

Actor:	User
Description:	This functionality allows user to view the current promotions across different cinemas
Preconditions:	The user must have login to the system using the Login use case
Postconditions:	System displays all available promotions to user

Priority:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. The user clicks the Promotions button at bottom navigation bar 2. The system directs user to PromotionsUI which displays ongoing promotions posters and names 3. The user clicks on any of the promotions posters/names 4. The system directs user to the selected promotion page and display detailed information for that promotion

Use Case Template

Use Case ID:	7
Use Case Name:	ViewAccountProfile

Actor:	User
Description:	This functionality allows user to view his/her profile details based on the gmail account used for login.
Preconditions:	The user must have login to the system using the Login use case
Postconditions:	System displays user profile details extracted from gmail account used for login

Priority:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The user clicks the Profile button at bottom navigation bar 2. The system directs user to ProfileUI which displays his/her profile details extracted from gmail account used for login

Use Case Template

Use Case ID:	8
Use Case Name:	ViewSupport

Actor:	User
Description:	This functionality allows user to view help/support regarding how to use the app
Preconditions:	The user must have login to the system using the Login use case
Postconditions:	System displays help/support information to user
Priority:	Medium

Flow of Events:	<ol style="list-style-type: none"> 1. User clicks the View Support button at the ProfileUI 2. System directs user to SupportUI which displays lists of frequently asked questions about using the application 3. User may click on any of the questions inside the list 4. The system expands the list and displays answer to the selected question
-----------------	---

Use Case Template

Use Case ID:	9
Use Case Name:	GiveAppReview

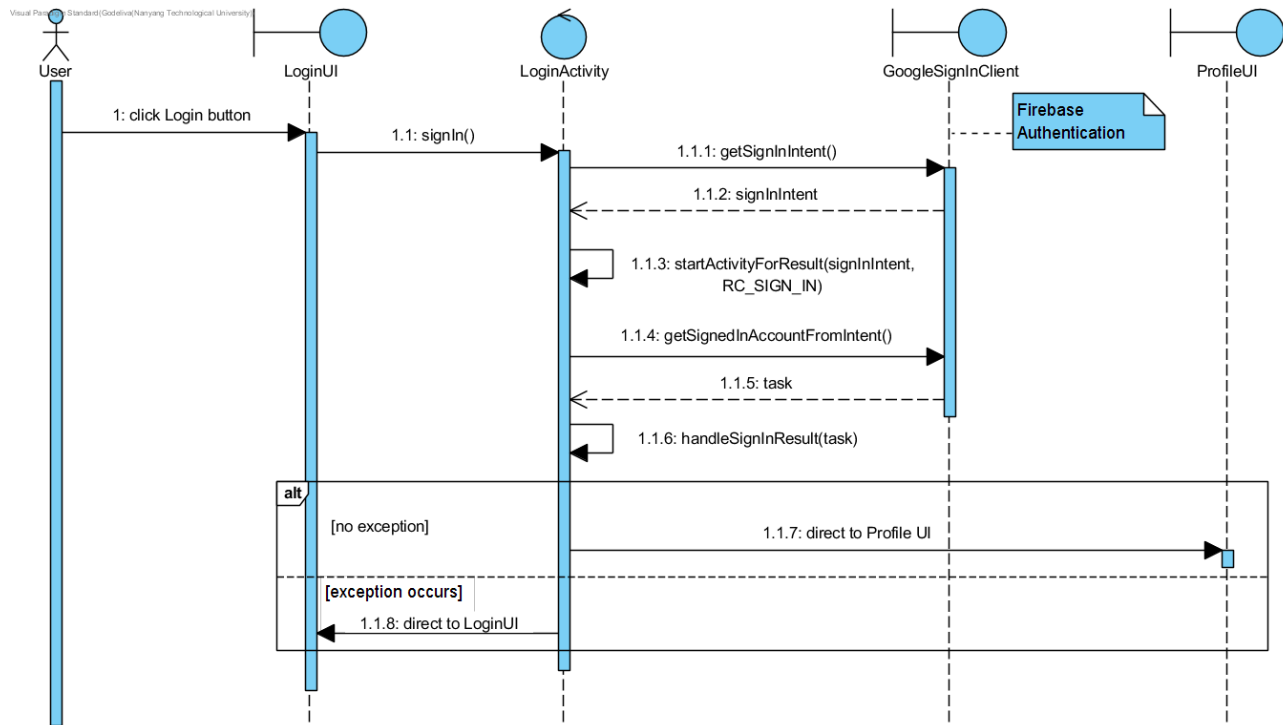
Actor:	User, Firebase Realtime Database (Participating Actor)
Description:	This functionality allows user to give ratings for the application
Preconditions:	This use case extends ViewSupport use case. It is invoked whenever the user decides to give rating for the application inside SupportUI.
Postconditions:	System displays confirmation message that rating is submitted
Priority:	Low

Flow of Events:	<ol style="list-style-type: none">1. User gives rating by clicking the stars scale button (scale 1-5) in SupportUI2. The system displays acknowledgement message indicating the rating that the user has just inputted3. The user clicks the submit button4. The system displays confirmation message that rating is submitted and directs user back to ProfileUI5. The system stores ratings in Firebase Realtime Database
-----------------	---

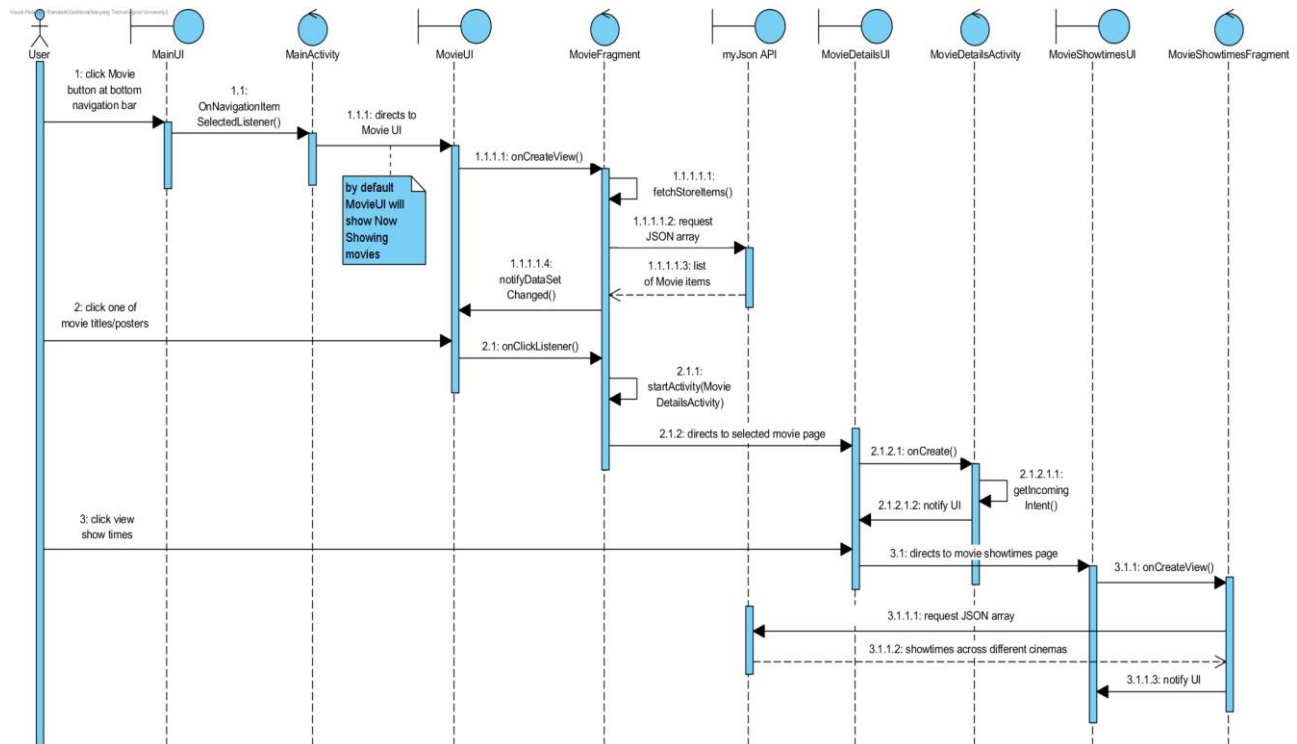
[illegible]

Appendix F: Sequence Diagram

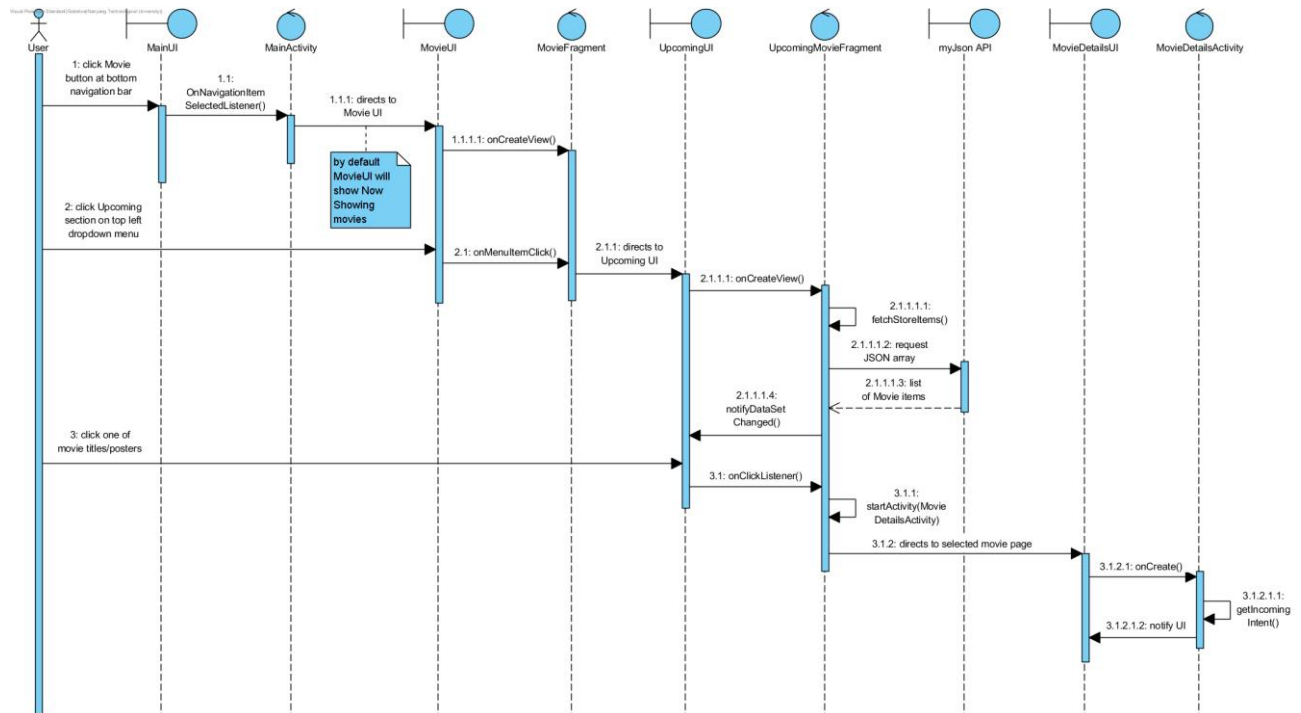
a. Login

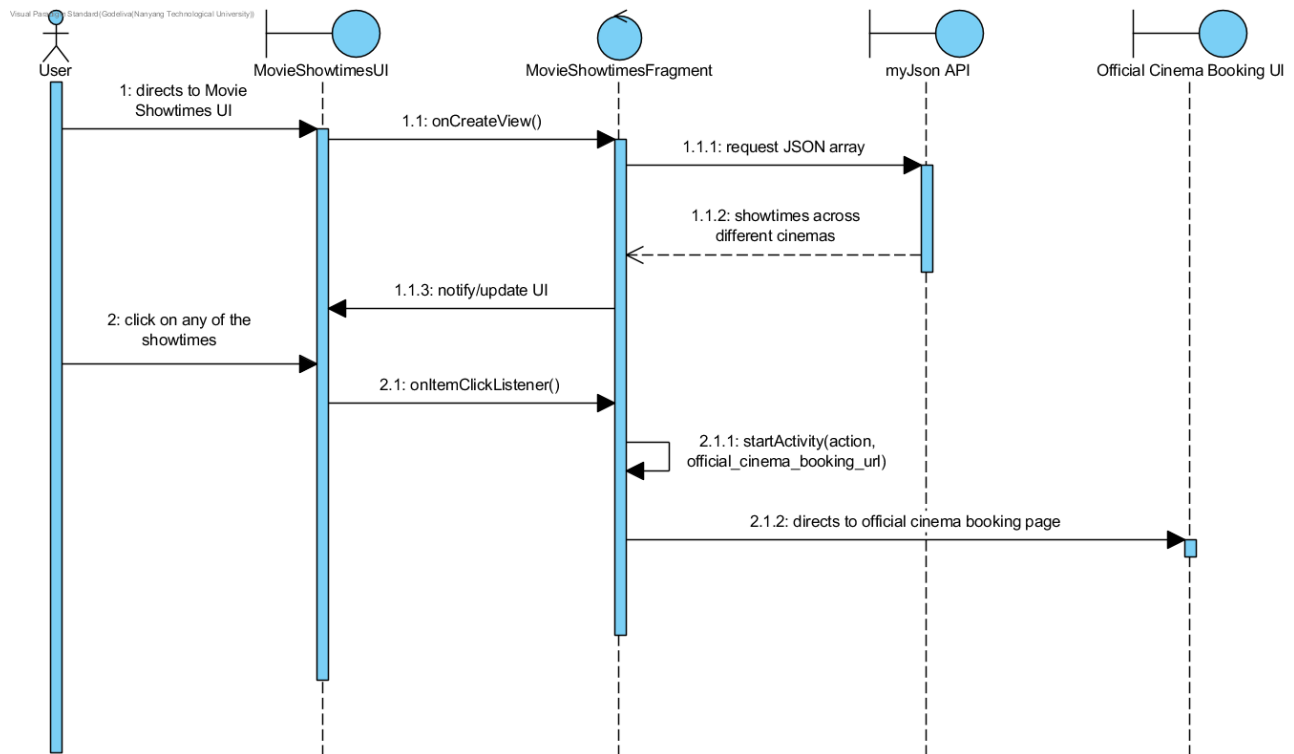


b. ViewNowShowingInfo

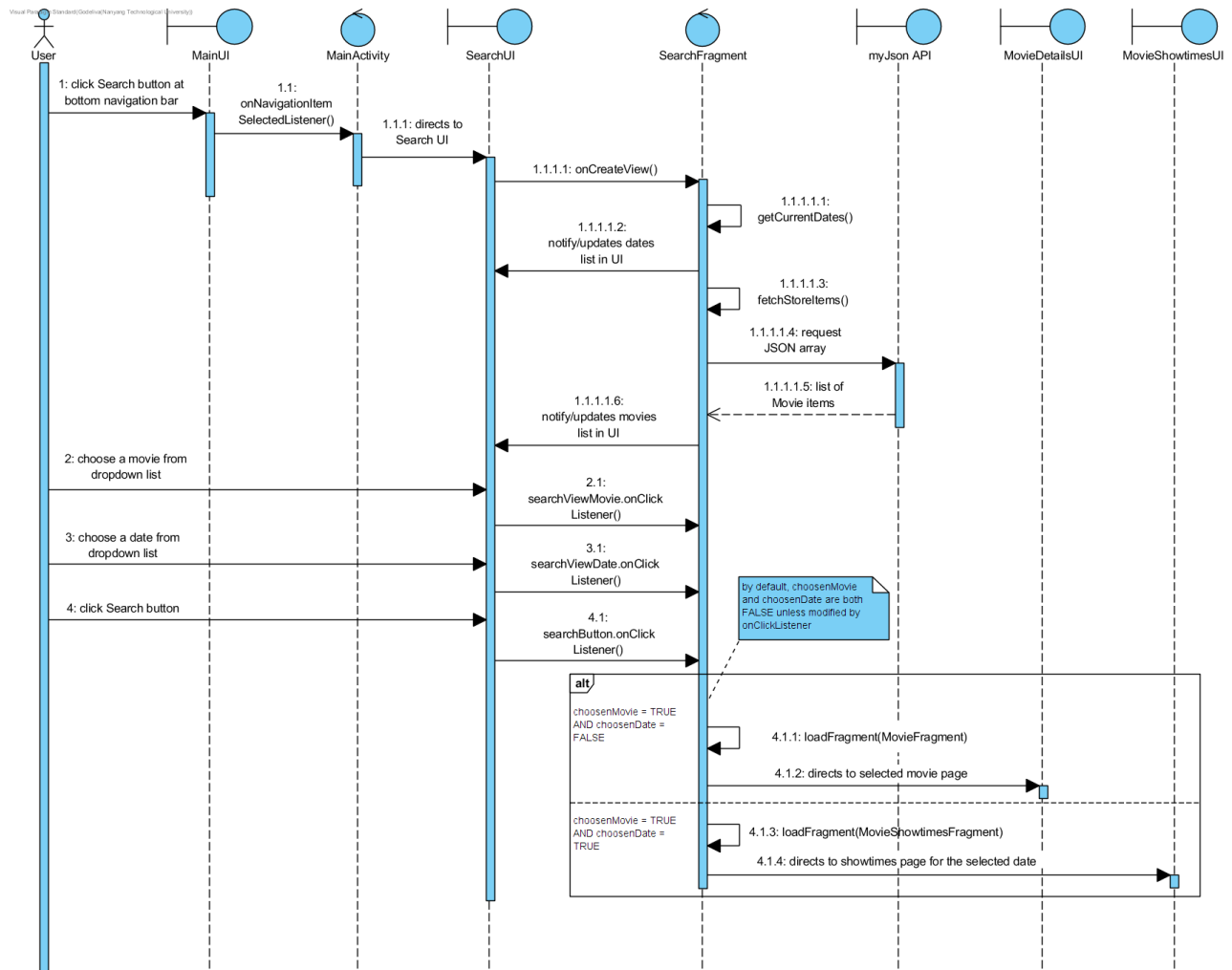


c. ViewUpcomingInfo

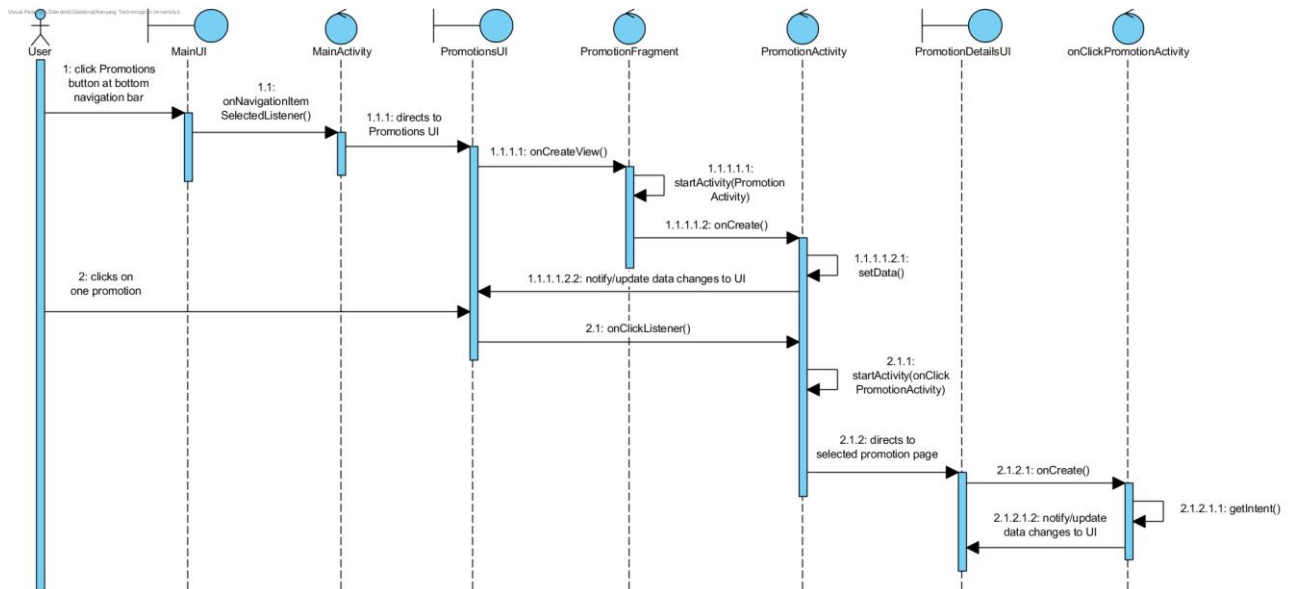


d. BookMovieTicket

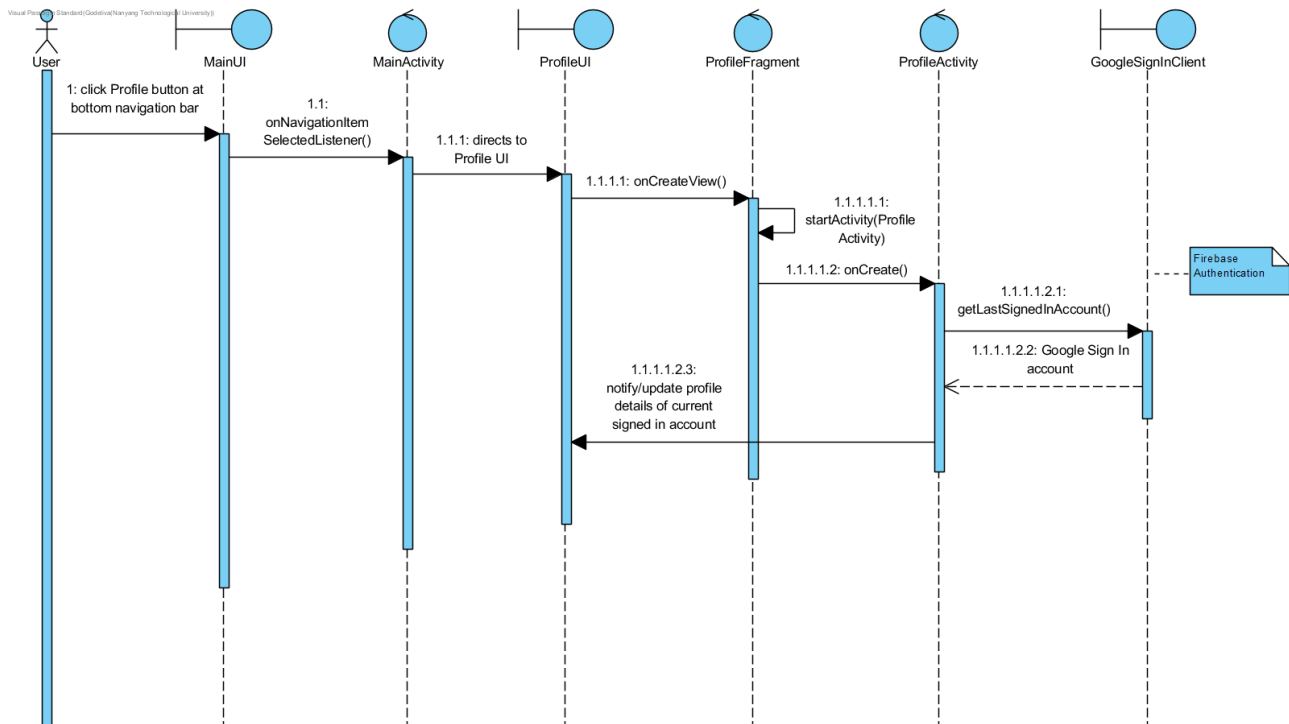
e. SearchMovie



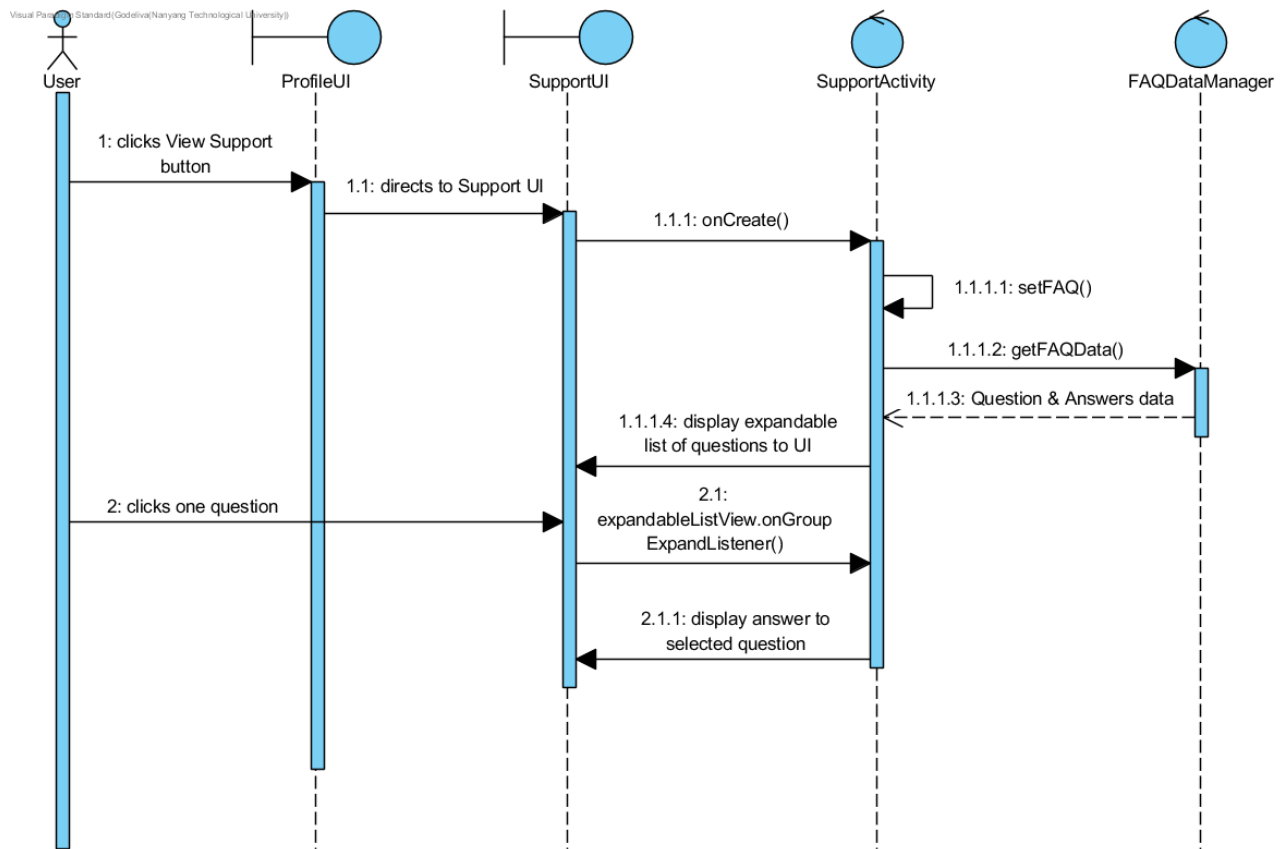
f. ViewPromotions



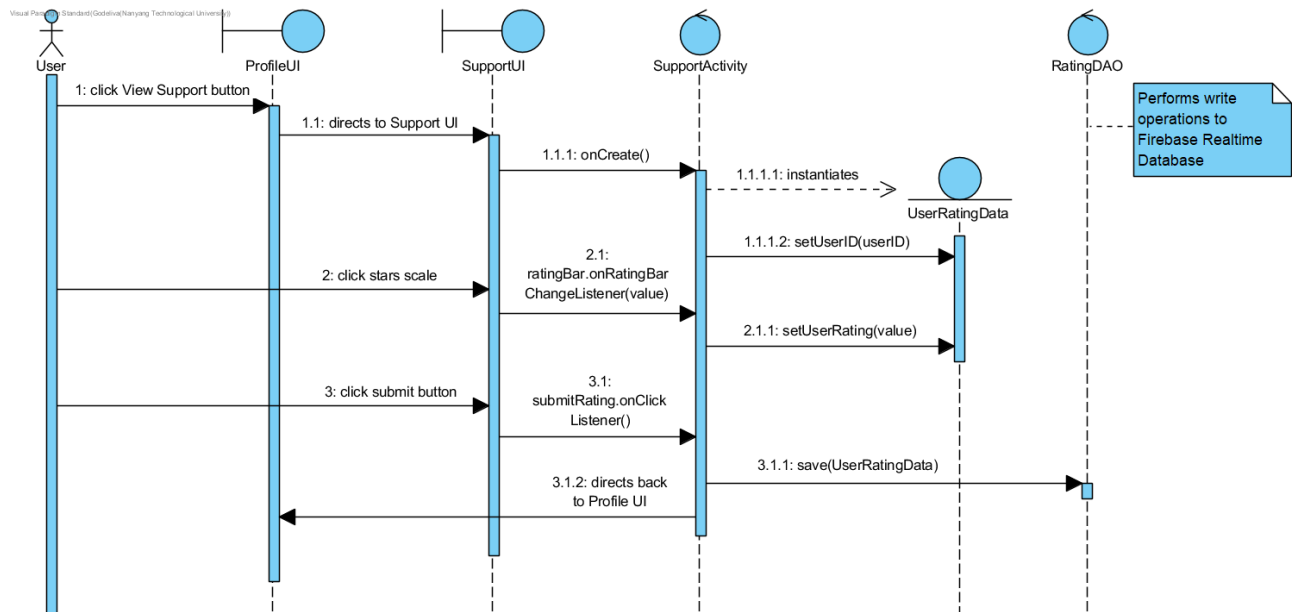
g. ViewAccountProfile



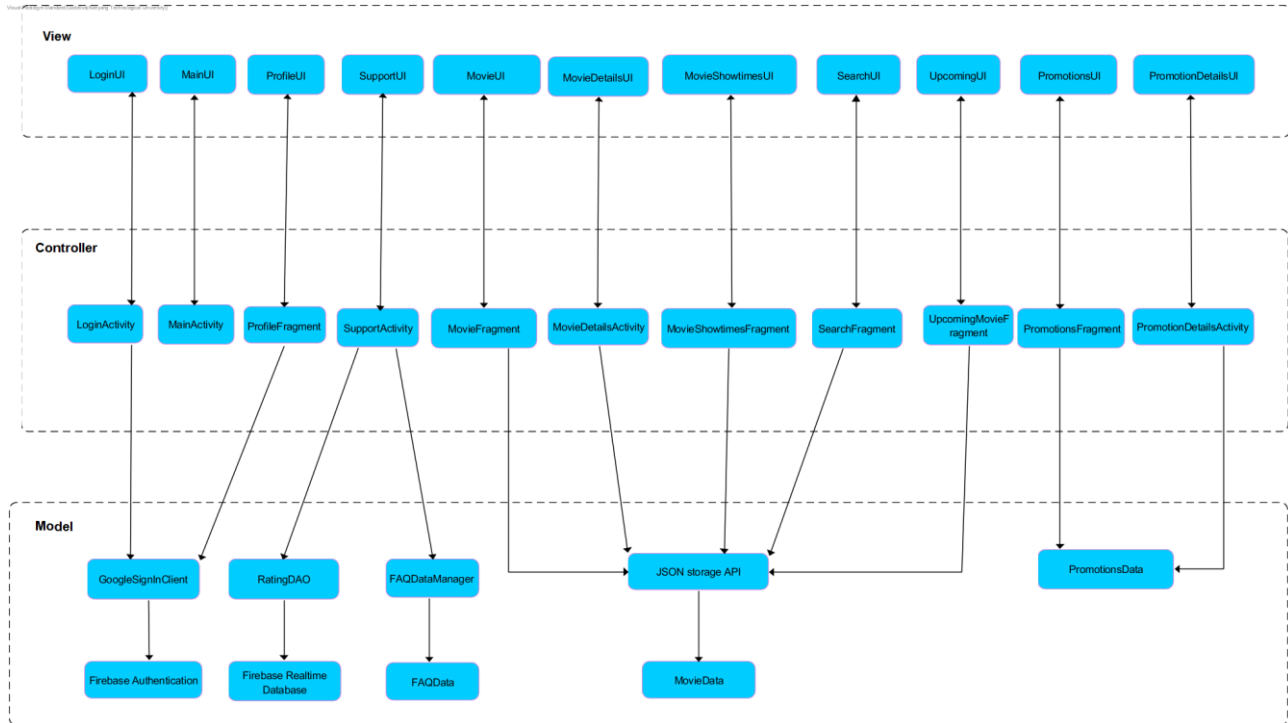
h. ViewSupport



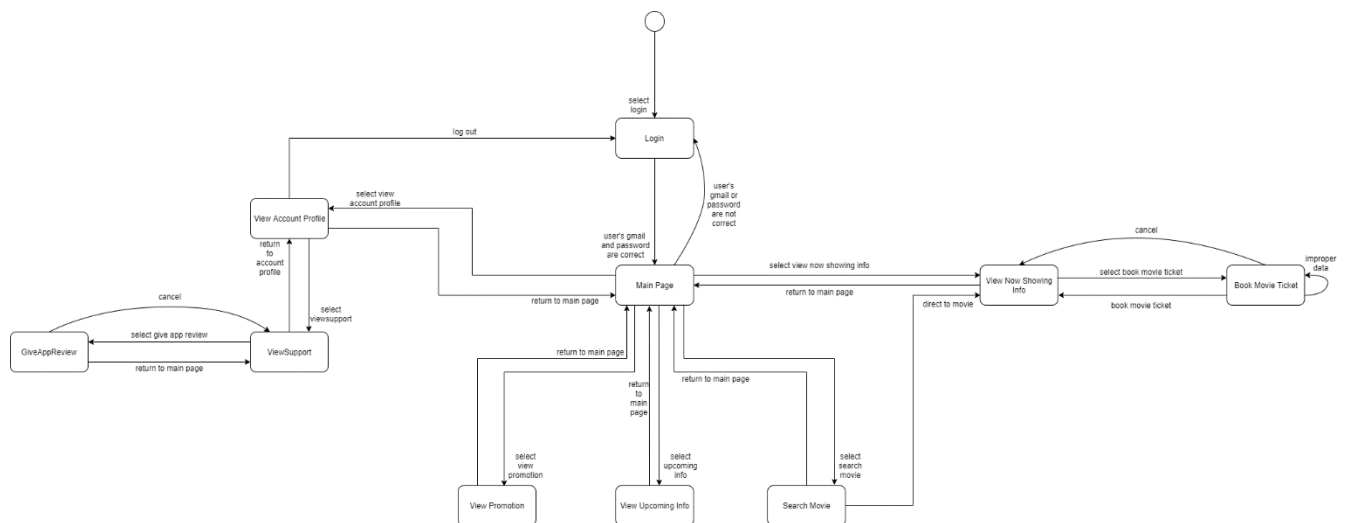
i. GiveAppReview



Appendix G: System Architecture



Appendix H: Dialog Map



Appendix I: Test Cases

a. Black Box Testing

Test Case ID: 001						
Test Priority (Low/Medium/High): High						
Test Title: Login						
Description: To test login authentication						
Pre-Conditions: -						
Step	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User clicks Sign In button at Login UI		Upon Sign In button is clicked, Firebase Authentication is activated and the user is prompted to enter his gmail address and password	Upon Sign In button is clicked, Firebase Authentication is activated and the user is prompted to enter his gmail address and password	Pass	

2a	User input his gmail address and password	Valid gmail address and valid password	Firestore authentication is successful and user is directed to Profile UI inside Catch Movie App	Firestore authentication is successful and user is directed to Profile UI inside Catch Movie App	Pass	
2b	User input his gmail address and password	Invalid gmail address and valid password	Firestore authentication is not successful and displays error message	Firestore authentication is not successful and displays error message	Pass	
2c	User input his gmail address and password	Invalid gmail address and invalid password	Firestore authentication is not successful and displays error message	Firestore authentication is not successful and displays error message	Pass	
2d	User input his gmail address and password	Valid gmail address and invalid password	Firestore authentication is not successful and displays error message	Firestore authentication is not successful and displays error message	Pass	

Post-conditions:

User is directed to Profile UI inside Catch Movie App upon successful login

Test Case ID: 002
Test Priority (Low/Medium/High): High
Test Title: Now Showing Movies
Description: To test listing of now showing movies

Pre-Conditions: User must be login, user must be in the main menu
--

Step	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User clicks on Movie logo on bottom navigation		Upon clicking on the Movie tab, a list of now showing movies will be shown. The list contains an image of the movie and the movie title.	Upon clicking on the Movie tab, a list of now showing movies will be shown. The list contains an image of the movie and the movie title.	Pass	

Post-conditions:

-

Test Case ID: 003
Test Priority (Low/Medium/High): High
Test Title: Upcoming Movies
Description: To test listing of upcoming movies

Pre-Conditions: User must be login, user must be in the main menu
--

Step	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User clicks on Movie logo on bottom navigation		Upon clicking on the Movie tab, a list of now showing movies will be shown. The list contains an image of the movie and the movie title.	Upon clicking on the Movie tab, a list of now showing movies will be shown. The list contains an image of the movie and the movie title.	Pass	

2	User clicks on the drop down menu at the top of the page and selects upcoming movies		Upon selecting upcoming movies, the list will change from Now Showing movies to Up coming movies. The list will contains an image of the movie and the movie title.	Upon selecting upcoming movies, the list will change from Now Showing movies to Up coming movies. The list will contains an image of the movie and the movie title.	Pass	
---	--	--	---	---	------	--

Post-conditions:

-

Test Case ID: 004
Test Priority (Low/Medium/High): High
Test Title: Movie Details
Description: To test movie details of a specific movie

Pre-Conditions: User must be login, user must be in the main menu

Step	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User clicks on Movie logo on bottom navigation		Upon clicking on the Movie tab, a list of now showing movies will be shown. The list contains an image of the movie and the movie title.	Upon clicking on the Movie tab, a list of now showing movies will be shown. The list contains an image of the movie and the movie title.	Pass	
2	User clicks on a movie from the list, for example, "Maleficent"		Upon clicking on the movie, more movie details of the movie will be shown such as language, synopsis, director and cast. There will be a "VIEW SHOW TIMES" button at the bottom of the page for user to click.	Upon clicking on the movie, more movie details of the movie will be shown such as language, synopsis, director and cast. There will be a "VIEW SHOW TIMES" button at the bottom of the page for user to click.	Pass	

Post-conditions:

-

Test Case ID: 005
Test Priority (Low/Medium/High): High
Test Title: Movie Show Time
Description: To test movie show time of a specific movie

Pre-Conditions: User must be login, user must be in the main menu
--

S t e p	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User clicks on Movie logo on bottom navigation		Upon clicking on the Movie tab, a list of now showing movies will be shown. The list contains an image of the movie and the movie title.	Upon clicking on the Movie tab, a list of now showing movies will be shown. The list contains an image of the movie and the movie title.	Pass	

2	User clicks on a movie from the list, for example, "Maleficent"		Upon clicking on the movie, more movie details of the movie will be shown such as language, synopsis, director and cast. There will be a "view show time" button at the bottom of the page for user to click.	Upon clicking on the movie, more movie details of the movie will be shown such as language, synopsis, director and cast. There will be a "VIEW SHOW TIMES" button at the bottom of the page for user to click.	Pass	
3	User clicks on view show time button at the bottom of the page		Upon clicking on the button, a list of showtime will displayed with the cinema name it will be playing at. There will be buttons on the top indicating the different dates the movie is playing.	Upon clicking on the button, a list of showtime will displayed. With buttons on the top indicating the different dates the movie is playing.	Pass	User will be able to click on the buttons to view different showtime for respective dates
4	User clicks on a particular showtime for example "GV Suntec 4:20 PM"		Upon clicking, user will be redirected to the respective cinema booking page. In this case, GV booking page, https://www.gv.com.sg/GVBuyTickets#	Upon clicking, user will be redirected to the respective cinema booking page. In this case, GV booking page, https://www.gv.com.sg/GVBuyTickets#	Pass	

Post-conditions:

-

Test Case ID: 006
Test Priority (Low/Medium/High): Medium
Test Title: Search by Search Movie
Description: To test search function when user will select just the movie

Pre-Conditions: User must be login, user must be in the main menu
--

Step	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User clicks on Search logo on bottom navigation		Upon clicking on the Search tab, three buttons display on the screen: Search Movie, Search Date and Search.	Upon clicking on the Search tab, three buttons display on the screen: Search Movie, Search Date and Search..	Pass	

2	User clicks on Search Movie Button		Upon clicking on the Search Movie Button, application displays list of movies which are currently showing.	Upon clicking on the Search Movie Button, application displays list of movies which are currently showing.	Pass	
3	User chooses the movie from the list, for example "Joker"		Upon clicking on element of the list, the Search Movie button changes name to the movie name of chosen movie and hide the list .	Upon clicking on element of the list, the Search Movie button changes name to the movie name of chosen movie and hide the list.	Pass	
4	User clicks on Search Button		Upon clicking on the Search Button, application redirects to movie description of the movie.	Upon clicking on the Search Button, application redirects to movie description of the movie	Pass	

Post-conditions:

User is directed to movie description of chosen movie.

Test Case ID: 007
Test Priority (Low/Medium/High): Medium
Test Title: Search by Search Movie and Search Date
Description: To test search function when user will select both the movie and the date

Pre-Conditions: User must be login, user must be in the main menu						
Step	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User clicks on Search logo on bottom navigation		Upon clicking on the Search tab, three buttons display on the screen: Search Movie, Search Date and Search.	Upon clicking on the Search tab, three buttons display on the screen: Search Movie, Search Date and Search..	Pass	
2	User clicks on Search Movie Button		Upon clicking on the Search Movie Button, application displays list of movies which are currently showing.	Upon clicking on the Search Movie Button, application displays list of movies which are currently showing.	Pass	
3	User chooses the movie from the list, for example "Joker"		Upon clicking on element of the list, the Search Movie button changes name to the movie name of chosen movie and hide the list .	Upon clicking on element of the list, the Search Movie button changes name to the movie name of chosen	Pass	

				movie and hide the list.		
4	User clicks on Search Date Button		Upon clicking on the Search Date Button, application displays list of dates from current day till four days.	Upon clicking on the Search Date Button, application displays list of dates from current day till four days.	Pass	
5	User chooses the date from the list, for example "Thu 14 Nov"		Upon clicking on element of the list, the Search Date button changes name to the date's name of chosen date and hide the list.	Upon clicking on element of the list, the Search Date button changes name to the date's name of chosen date and hide the list.	Pass	
6	User clicks on Search Button		Upon clicking on the Search Button, application redirects to movie showtime on chosen date of chosen movie.	Upon clicking on the Search Button, application redirects to movie showtime on chosen date of chosen movie.	Pass	

Post-conditions:

User is directed to Movie Showtime of chosen date of chosen movie.

Test Case ID: 008
Test Priority (Low/Medium/High): High
Test Title: Give Rating
Description: To test rating submission in Firebase Realtime Database

Pre-Conditions: User must be login, user must be in the Support UI

Steps	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User clicks the star scale at Support UI		Upon clicking star scale, the app displays a message indicating the rating value that the user just inputted	Upon clicking star scale, the app displays a message indicating the rating value that the user just inputted	Pass	
2	User clicks the Submit Rating button		Rating is stored at Firebase Realtime Database. A new entry is created if the user had never submitted any rating before. The entry is updated if the user	Rating is stored at Firebase Realtime Database. A new entry is created if this was the first rating submission. The entry is updated if the	Pass	

			had submitted rating previously.	user had submitted rating previously.		
--	--	--	----------------------------------	---------------------------------------	--	--

Post-conditions:

User is directed to Profile UI after rating submission

Test Case ID: 009
Test Priority (Low/Medium/High): High
Test Title: View Support
Description: To test viewing FAQ

Pre-Conditions: User must be login, user must be in the Profile UI

Steps	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User clicks the View Support button		The app directs user to Support UI and displays lists of questions	The app directs user to Support UI and display lists of questions	Pass	
2	User clicks on the question		The app displays answers to the selected question	The app displays answers to the selected question	Pass	

Post-conditions:

-

Test Case ID: 010
Test Priority (Low/Medium/High): High
Test Title: Promotions
Description: To test listing of now movie promotions

Pre-Conditions: User must be logged in; user must be in the main menu page

Step	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User clicks on Promotion logo on bottom navigation		Upon Promotion logo is clicked, the promotion page loads, showing lists of promotions within different categories of promotions.	Upon Promotion logo is clicked, the promotion page loads, showing lists of promotions within different categories of promotions.	Pass	

Post-conditions:

-

Test Case ID: 011
Test Priority (Low/Medium/High): High
Test Title: On Click Promotions
Description: To test the display if promotions on click

Pre-Conditions: User must be logged in; user must be in the promotion page						
Step	Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)	Notes
1	User scrolls vertically on the page		Upon scrolling vertically, different categories of promotions are shown to the user as the user scrolls down in the order of "All Current", "Discounts", "Movie Perks", "Food and Beverages", and "Partnerships".	Upon scrolling vertically, different categories of promotions are shown to the user as the user scrolls down in the order of "All Current", "Discounts", "Movie Perks", "Food and Beverages", and "Partnerships".	Pass	

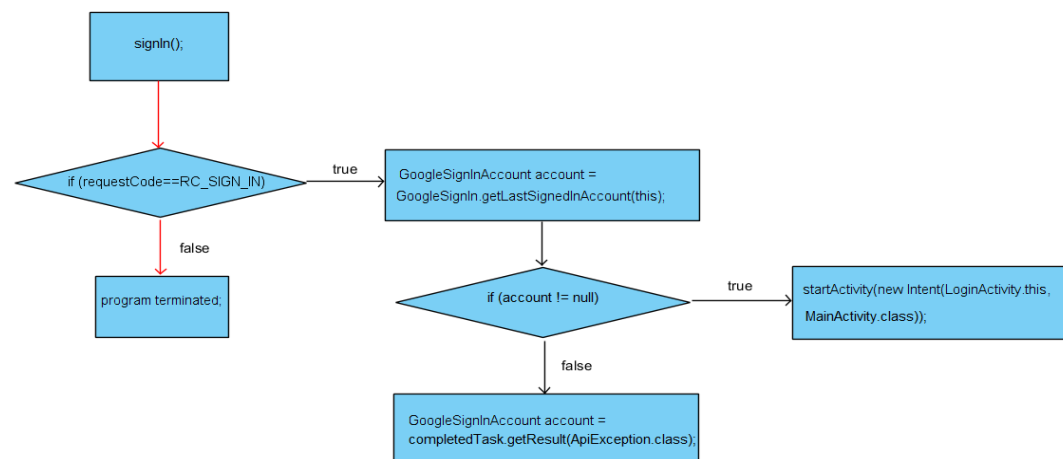
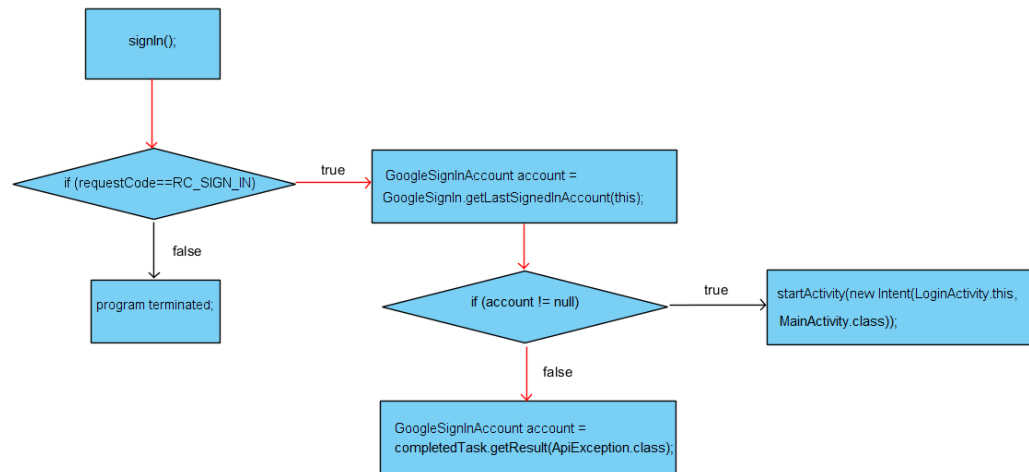
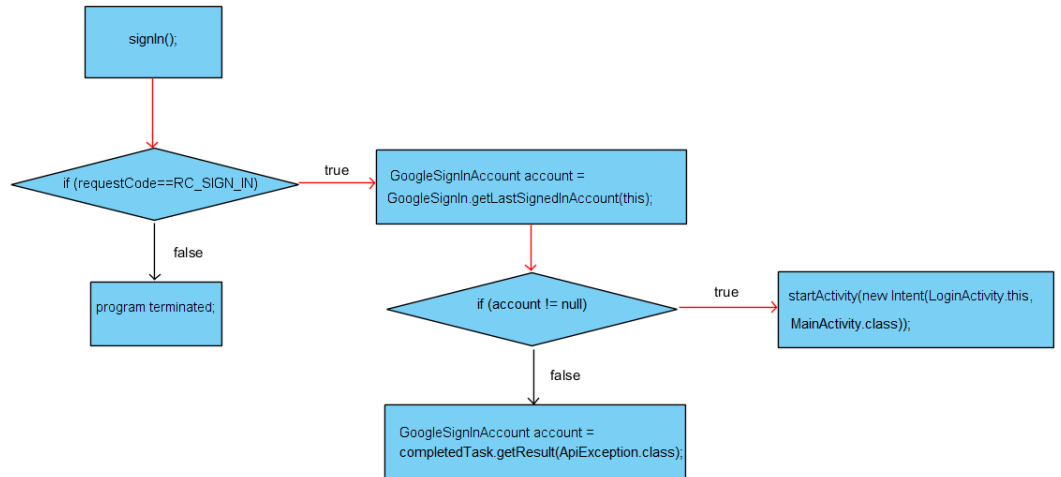
2	User scrolls horizontally along a category of promotions		Upon scrolling horizontally, thumbnails and names of different promotions in each category is shown to the user	Upon scrolling horizontally, thumbnails and names of different promotions in each category is shown to the user	Pass	
3	User clicks on promotion in promotion list		Upon clicking on promotion, a page pops up and displays the chosen promotion's image, name, and description.	Upon clicking on promotion, a page pops up and displays the chosen promotion's image, name, and description.	Pass	

Post-conditions:

-

b. White Box Testing

Login



Search Movie

