Software Development Laboratory

B.Tech. IV Semester

Name : KAUSHAL VASHISTH

Roll Number : 18ETCS002147

Department : Computer Science and Engineering

Faculty of Engineering & Technology

M. S. Ramaiah University of Applied Sciences

NAME:- KAUSHA

M. S. Ramaiah University of Applied Sciences

Private University Established in Karnataka State by Act No. 15 of 2013

Faculty	Engineering & Technology
Programme	B. Tech. in Computer Science and Engineering
Year/Semester	2018/4th Semester
Name of the Laboratory	Software Development Laboratory
Laboratory Code	CSL216

List of Experiments

1. Requirements Analysis - I
2. Requirements Analysis - II
3. Data flow modelling with CASE tools – High level Design
4. Data flow modelling with CASE tools – Low level Design
5. UML Modelling: Use Case and Sequence Diagrams
6. UML Modelling: Class Diagrams
7. UML Modelling: State Chart Diagrams
8. Implementation of Software Design
9. Unit Testing with JUnit
10. Integration testing with JUnit

Scenario for all labs:

Various scenarios will be given to students in the lab. Work in groups of 2 and develop the software solution. The Course leader is the customer. Contact the Course leader for any clarifications.

NAME:- KAUSHAL VASHISTH ROLL NO: 18ETCS002147

Index Sheet

No.	Lab Experiment	Viva	Results	Documentation	Total Marks
1	Requirements Analysis - I				
2	Requirements Analysis - II				
3	UML Modelling: Use Case and Sequence Diagrams				
4	UML Modelling: Class Diagrams				
5	UML Modelling: State Chart Diagrams				
6	Data flow modelling with CASE tools – High level Design				
7	Data flow modelling with CASE tools – Low level Design				
8	Implementation of Software Design				
9	Unit Testing with JUnit				
10	Lab Internal Test conducted along the lines of SEE and valued for 50 Marks and reduced for 20 Marks				
	Total Marks				

Component 1 (Lab Internal Marks) =

Signature of the Staff In-charge

Laboratory 1

Title of the Laboratory Exercise: Requirements Analysis - I

1. Introduction and Purpose of Experiment

   Students get familiar with the documentation and scenario specified for all the lab exercises while analysing the requirements of the scenario

2. Aim and Objectives

   Aim

   - To develop formal software requirements in a standard format for a given engineering problem

   Objectives

   At the end of this lab, the student will be able to

   – Identify software requirements from problem statement
   – Identify type of a software requirement
   – Create an unambiguous list of software requirements based on interaction with a client

3. Experimental Procedure

- Work in teams of 2 students
- Each team should read the problem statement and identify requirements as a group
- Each team will then confirm the requirements and document the requirements in an SRS document
- Each individual will then write their lab manual, documenting their observations

4. Calculations/Computations/Algorithms:-

Movie ticket system:-

Functional requirements:-

FR1:-

Registration

If a customer wants to book the ticket then he/she must be registered, an unregistered user can't book the ticket.

FR2:-

Login -

Customer logins to the system by entering valid user id and password for booking the ticket.

FR3:-

Browse Movie:-

The system/software should allow the user to search movies based on movie name, date, time and venue.

FR4:-

Selection and show timings:-

The system/software should have a function to select the movie and display the show timings.

FR5:-

Seat Viewing and booking ticket -

The system/software should have a function to show a 2D image of the available, non-available and user selected seats and book ticket.

FR6:-

Payment:-

The system/software should direct to payment gateway and display payment mode.

FR7:-

Generate ticket -

After booking, the system can generate the E-ticket or then send one copy to the user's Email-address or as an SMS to user's phone number.

FR8:-

Ticket canceling

The user's shall be given an option to cancel ticket.

FR9:-

User support:-

The system/software should contain a function to contact the customer care.

FR10:-

Logout -

The system/software should contain a function to logout.

Non-functional requirements:-

[NFR1]:-

Security: -

The system uses SSL (secured socket layer) in all transactions that include any confidential customer information.

[NFR2]:-

Reliability:-

Reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes.

[NFR3]:-

Availability:-

The service should be available at all times (i.e. 24x7).

[NFR4]:-

Maintainability –

the software design is being done with modularity in mind so that maintainability can be done efficiently.

[NFR5]:-

Portability:-

The system shall run on PC, Laptops etc.

5. Analysis and Discussions:-

The functional requirement is describing the behaviour of the system as it relates to the system's functionality. The non-functional requirement elaborates a performance characteristic of the system.

- An example of a functional requirement would be:

  A system must send an email whenever a certain condition is met (e.g. an order is placed, a customer signs up, etc).

- A related non-functional requirement for the system may be:

  Emails should be sent with a latency of no greater than 12 hours from such an activity.

6. Conclusions:-

All the functional and non-functional requirements have been noted down for better design of the software/system.

Component	Max Marks	Marks Obtained
Viva		
Results		
Documentation		
Total		

Laboratory 2

Title of the Laboratory Exercise: Requirements Analysis - II

7. Introduction and Purpose of Experiment

   Students will formally document the identified requirements in an SRS document for the scenario

8. Aim and Objectives

   Aim

   - To develop formal SRS document in a standard format for a given engineering problem

   Objectives

   At the end of this lab, the student will be able to

   – Identify dependencies of a software requirement
   – Create SRS document in a standard format

9. Experimental Procedure

- Work in teams of 7 students
- Each team should read the problem statement and identify requirements as a group
- Each team will then confirm the requirements and document the requirements in an SRS document
- Each individual will then write their lab manual, documenting their observations

10. Calculations/Computations/Algorithms

REQUIREMENT TAG:	FR1
REQUIREMENT STATEMENT:-	If a customer wants to book the ticket then he/she must be registered, an unregistered user can't book the ticket.
DEPENDENT ON :- (REQUIREMENTS)	Not applicable.
STAKEHOLDER: (OWNER OF REQ.)	Admin and End user
DESCRIPTION:	The application should allow the user to link there e-mail id, phone number and set the username and password. Password must be more than 8 characters.

REQUIREMENT TAG:	FR2
REQUIREMENT STATEMENT:-	Customer logins to the system by entering valid user id and password for booking the ticket.
DEPENDENT ON :- (REQUIREMENTS)	FR1
STAKEHOLDER: (OWNER OF REQ.)	Admin and End user
DESCRIPTION:	The application should allow the user to login using correct username and password. Password must be more than 8 characters

REQUIREMENT TAG:	FR3
REQUIREMENT STATEMENT:-	The system/software should allow the user to search movies based on movie name, date, time and venue.
DEPENDENT ON :- (REQUIREMENTS)	Not applicable.
STAKEHOLDER: (OWNER OF REQ.)	End user
DESCRIPTION:	The application should allow the user to browse movies based on users preference like language, movie name, venue etc.

REQUIREMENT TAG:	FR4
REQUIREMENT STATEMENT:-	The system/software should have a function to select the movie and display the show timings.
DEPENDENT ON :- (REQUIREMENTS)	FR3
STAKEHOLDER: (OWNER OF REQ.)	End user
DESCRIPTION:	The application should allow the user to select movies, and display languages of movie released ,2D,3D,theatres, show timings and .

REQUIREMENT TAG:	FR5
REQUIREMENT STATEMENT:-	The system/software should have a function to show a 2D image of the available, non-available and user selected seats and book ticket.
DEPENDENT ON :- (REQUIREMENTS)	FR4
STAKEHOLDER: (OWNER OF REQ.)	End user
DESCRIPTION:	The application should allow user to view seats in selected theatre and display available, non-available and user selected seats in different colors and book the ticket of chosen seats.

REQUIREMENT TAG:	FR6
REQUIREMENT STATEMENT:-	The system/software should direct to payment gateway and display payment mode.
DEPENDENT ON :- (REQUIREMENTS)	FR5
STAKEHOLDER: (OWNER OF REQ.)	Admin and End user

DESCRIPTION:	Once the booking is done application directs to payment gateway and display the offers and modes of payment can be done via phone pay, PayPal, debit card, COD etc .

REQUIREMENT TAG:	FR7
REQUIREMENT STATEMENT:-	After booking, the system can generate the E-ticket and then send one copy to the user's Email-address or as an SMS to user's phone number.
DEPENDENT ON :- (REQUIREMENTS)	FR5,FR6
STAKEHOLDER: (OWNER OF REQ.)	Admin and End user
DESCRIPTION:	Once the payment was done the application should get generate the the ticket which contains the chosen theatre, show time, seat no etc. And send the ticket to registered e-mail or Phone.no

REQUIREMENT TAG:	FR8
REQUIREMENT STATEMENT:-	The user's shall be given an option to cancel ticket.
DEPENDENT ON :- (REQUIREMENTS)	FR5,FR6
STAKEHOLDER: (OWNER OF REQ.)	Admin and End user
DESCRIPTION:	The application should allow user to cancel the booked ticket and re-transfer money based on terms and conditions.

REQUIREMENT TAG:	FR9
REQUIREMENT STATEMENT:-	The system/software should contain a function to contact the customer care.
DEPENDENT ON :- (REQUIREMENTS)	FR2
STAKEHOLDER: (OWNER OF REQ.)	End user
DESCRIPTION:	The application should allow user to contact the customer care via email or phone number.

REQUIREMENT TAG:	FR10

REQUIREMENT STATEMENT:-	The system/software should contain a function to logout .
DEPENDENT ON :- (REQUIREMENTS)	FR2
STAKEHOLDER: (OWNER OF REQ.)	Admin and End user
DESCRIPTION:	The system/software should allow user to logout of logged in account whenever user wishes.

11. Analysis and Discussions

12. Conclusions :-

    All the functional requirements have been analysed thoroughly.

Component	Max Marks	Marks Obtained
Viva	6	
Results	7	
Documentation	7	
Total	20	

Laboratory 3

Title of the Laboratory Exercise: **Use case diagram**

1. Introduction and Purpose of Experiment

   Students will apply data flow modelling to develop the high level design for given scenario

2. Aim and Objectives

   Aim

   - To develop software architecture for a given requirements specification using Structured analysis and Design Technique

   Objectives

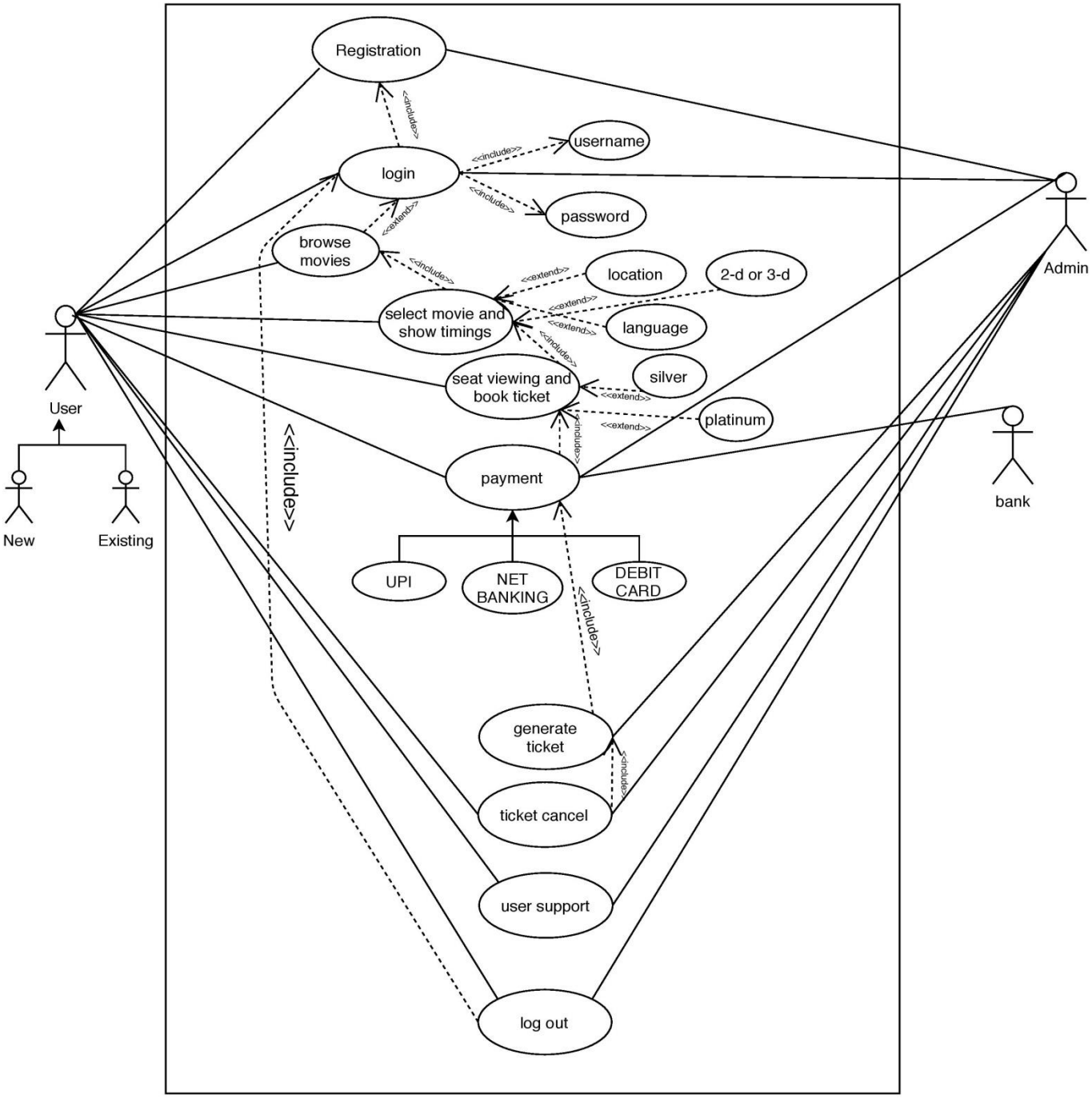
   At the end of this lab, the student will be able to

   – Identify context of the software
   – Identify Inputs, Outputs and Data Stores for a given software
   – Identify modules in a software and their dependencies
   – Create design document for a given SRS

3. Experimental Procedure

- Work in teams of 7 students
- Each team should read the problem statement and identify requirements as a group
- Each team will then confirm the requirements and document the requirements in an high level design document
- Each individual will then write their lab manual, documenting their observations

4. Presentation of Results



USE CASE DIAGRAM FOR MOVIE TICKET BOOKING SYSTEM

NAME:- KAUSHAL VASHISTH

REG:- 18ETCS002147

5. Analysis and Discussions

Purpose of Use Case Diagram :-

Use case diagrams are typically developed in the early stage of development and people often apply use case modelling for the following purposes:

- Specify the context of a system
- Capture the requirements of a system
- Validate a systems architecture
- Drive implementation and generate test cases
- Developed by analysts together with domain experts

6. Conclusions

We have successfully learned about “Use Case Diagrams”.

7. Comments

1. Limitations of Experiments

Use cases are not well suited to capturing non-interaction based requirements of a system (such as algorithm or mathematical requirements) or non-functional requirements (such as platform, performance, timing, or safety-critical aspects).

Component	Max Marks	Marks Obtained
Viva	7	
Results	6	
Documentation	6	
Total	20	

Laboratory 4

Title of the Laboratory Exercise: **Sequence Diagram**

1. Introduction and Purpose of Experiment

   Students will apply data flow modelling to develop the low level design for given scenario

2. Aim and Objectives

   Aim

   - To develop low level software design for a given requirements specification using Structured analysis and Design Technique

   Objectives

   At the end of this lab, the student will be able to

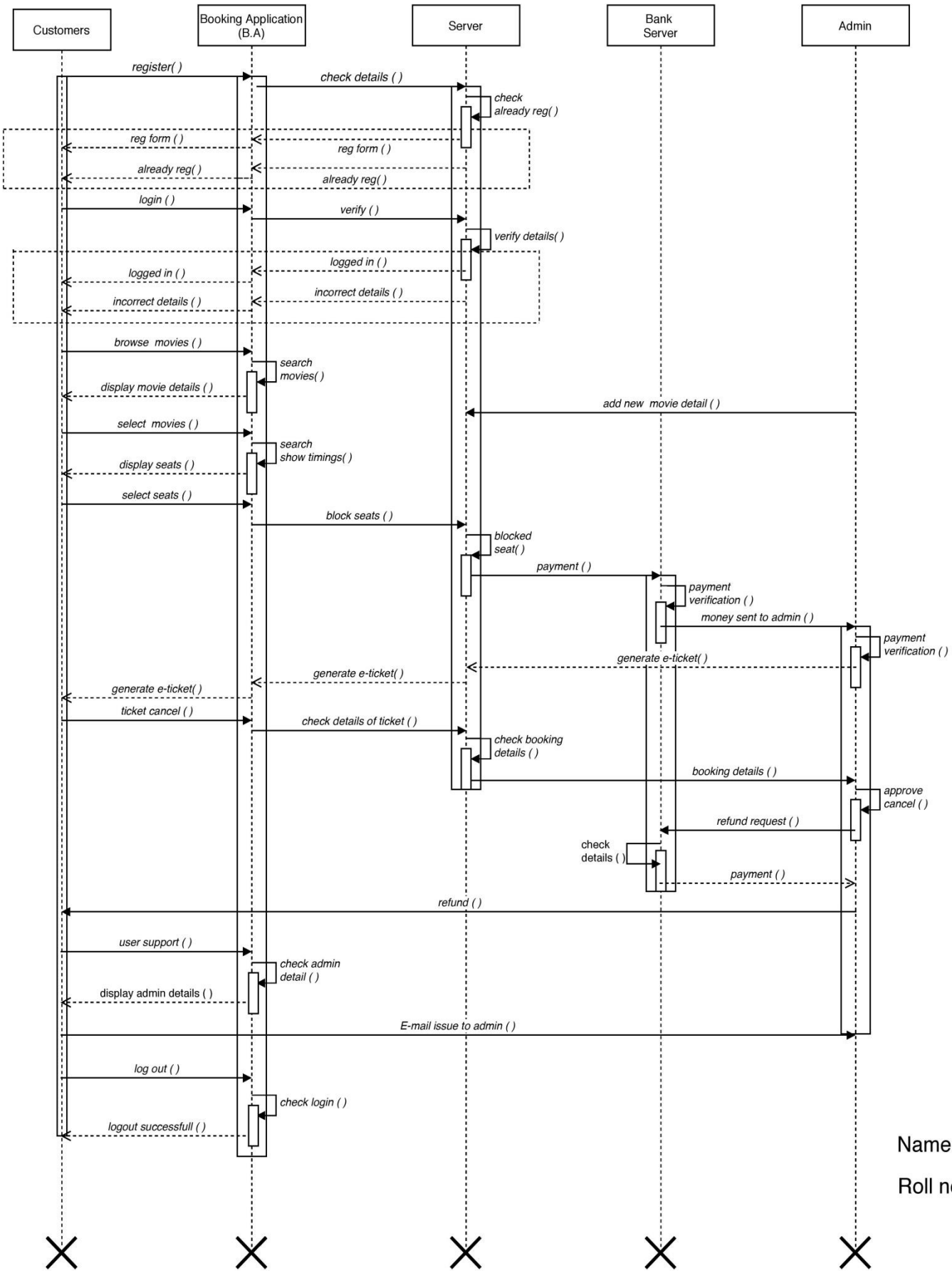
   – Identify functions in modules
   – Identify Inputs, Outputs and Data dependencies for functions
   – Create low level design document for a given SRS

3. Experimental Procedure

- Work in teams of 7 students
- Each team should read the problem statement and identify requirements as a group
- Each team will then confirm the requirements and document the requirements in an low level design document
- Each individual will then write their lab manual, documenting their observations

4. Presentation of Results

NAME:- KAUSHAL VASHISTH ROLL NO: 18ETCS002147



Name:- Kaush

Roll no:- 18ET

5. Analysis and Discussions

Sequence Diagrams captures:

- the interaction that takes place in a collaboration that either realizes a use case or an operation (instance diagrams or generic diagrams)
- high-level interactions between user of the system and the system, between the system and other systems, or between subsystems (sometimes known as system sequence diagrams)

Purpose of Sequence Diagram

- Model high-level interaction between active objects in a system
- Model the interaction between object instances within a collaboration that realizes a use case
- Model the interaction between objects within a collaboration that realizes an operation
- Either model generic interactions (showing all possible paths through the interaction) or specific instances of a interaction (showing just one path through the interaction)

6. Conclusions:

We have successfully learned about UML sequence diagrams.

7. Comments

1. Limitations of Experiments

sequence diagrams may define optional layers, as long as they do not make graphical elements appear or disappear on the diagram when they are selected or de-selected. Layers which contribute new tools in the palette for example are fine.

Component	Max Marks	Marks Obtained
Viva	7	
Results	6	
Documentation	6	
Total	20	

Laboratory 5

Title of the Laboratory Exercise: Class Diagrams

1. Introduction and Purpose of Experiment

   Students will apply object oriented analysis and design for the given scenario for object decomposition

2. Aim and Objectives

   Aim

   - To construct a UML class diagram for a given system and identify the class members and determine their relationships

   Objectives

   At the end of this lab, the student will be able to

   – Identify the main members of the family
   – Identify how they are related to each other
   – Find the characteristics of each family member
   – Determine relations among family members
   – Decide the inheritance of personal traits and characters

3. Experimental Procedure

- Work in teams of 7 students
- Each team should read the problem statement and discuss the requirements as a group
- Each team will then create and confirm the design and document the design in an software design specifications document
- Each individual will then write their lab manual, documenting their observations

4. Calculations/Computations/Algorithms

To draw class diagram there are several things we need to consider:-

Each class is represented by a rectangle having a subdivision of three compartments name, attributes and operation

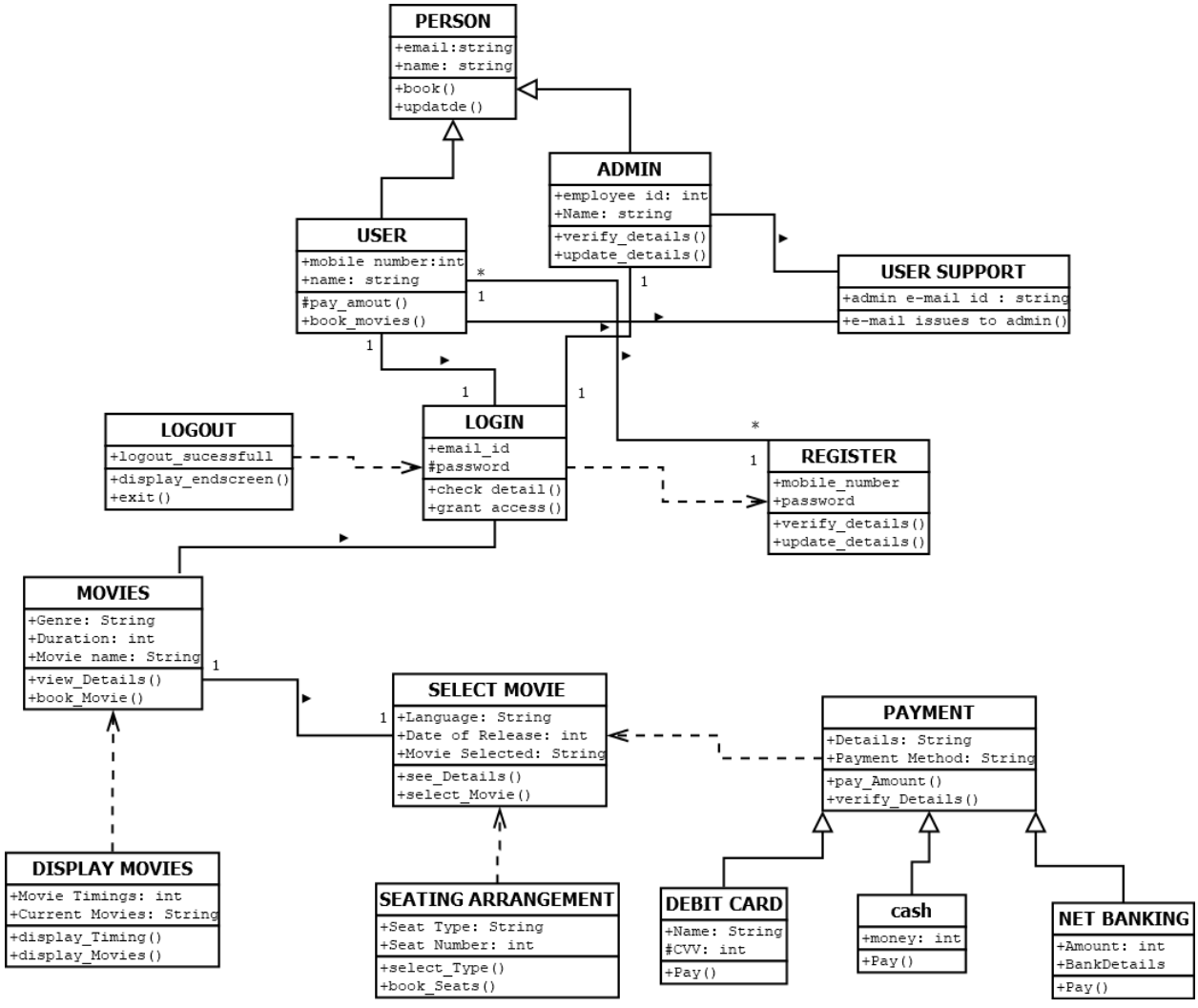
There are three types of modifiers which are used to decide the visibility of attributes and operations.

+ is used for public visibility (for everyone)

is used for protected visibility (for friend and derived)

– is used for private visibility (for only me)

5. Presentation of Results



6. Analysis and Discussions

The Purpose of Class Diagrams :-

- Shows static structure of classifiers in a system
- Diagram provides a basic notation for other structure diagrams prescribed by UML
- Helpful for developers and other team members too
- Business Analysts can use class diagrams to model systems from a business perspective

7. Conclusions :-

It can be concluded that a class diagram is made up of following notations:-

1. Class Name
   - The name of the class appears in the first partition.
2. Class Attributes
   - Attributes are shown in the second partition.
   - The attribute type is shown after the colon.
   - Attributes map onto member variables (data members) in code.
3. Class Operations (Methods)
   - Operations are shown in the third partition. They are services the class provides.
   - The return type of a method is shown after the colon at the end of the method signature.
   - The return type of method parameters is shown after the colon following the parameter name.
   - Operations map onto class methods in code

8. Comments

1. Limitations of Experiments

The class diagrams might often take a longer time manage, and maintain which is sometimes annoying for a developer. It requires time for the synchronization with the software code, to set it up, and maintain. Often developers or small companies find it difficult to synchronize the code as it required an added amount of work.

NAME:- KAUSHAL VASHISTH ROLL NO: 18ETCS002147

Component	Max Marks	Marks Obtained
Viva	6	
Results	7	
Documentation	7	
Total	20	

Laboratory 6

Title of the Laboratory Exercise: **State Chart**

1. Introduction and Purpose of Experiment

   Students will apply object oriented analysis and design for the given scenario for object decomposition

2. Aim and Objectives

   Aim

   - To construct a UML class diagram for a given system and identify the class members and determine their relationships

   Objectives

   At the end of this lab, the student will be able to

   - Identify the main members of the family
   - Identify how they are related to each other
   - Find the characteristics of each family member
   - Determine relations among family members
   - Decide the inheritance of personal traits and characters

3. Experimental Procedure

- Work in teams of 7 students
- Each team should read the problem statement and discuss the requirements as a group
- Each team will then create and confirm the design and document the design in an software design specifications document
- Each individual will then write their lab manual, documenting their observations

4. Presentation of Results

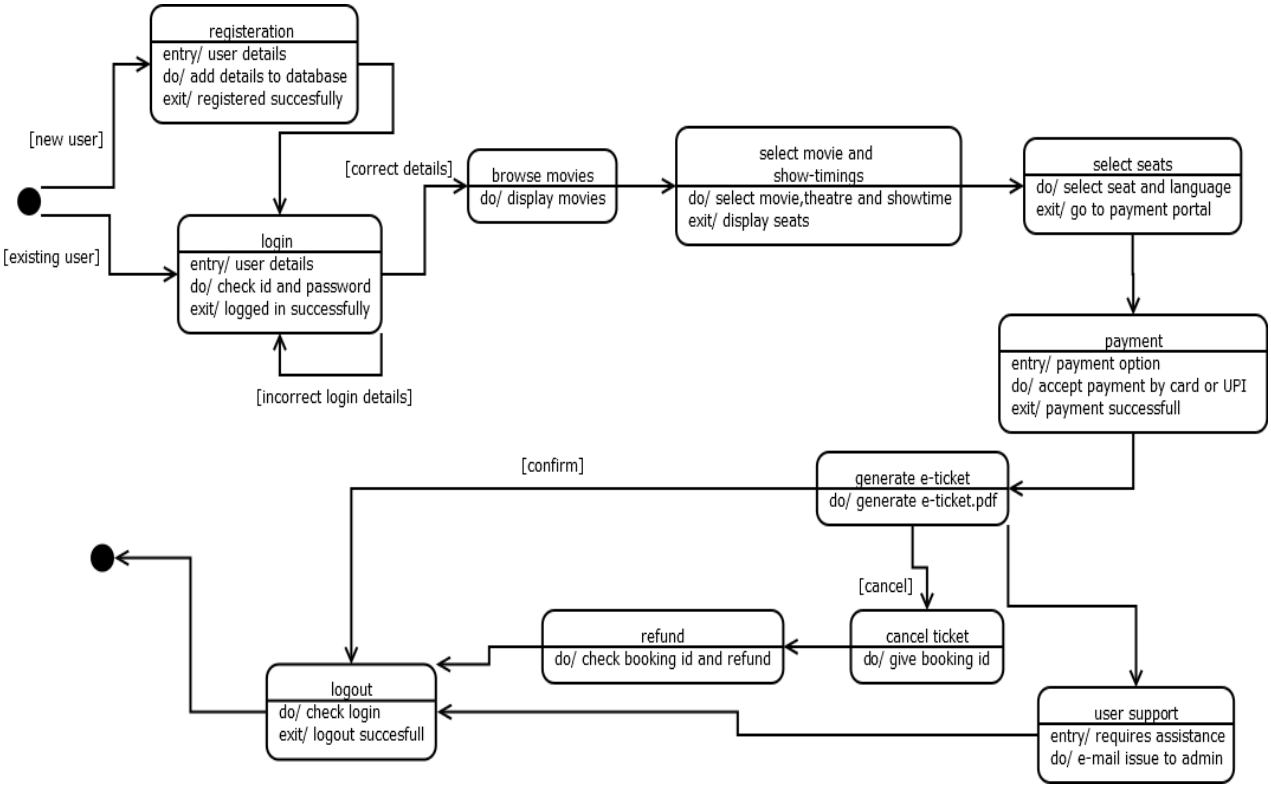


Figure 1 state chart diagram

Analysis and Discussions

This lab deals with designing part of the software. Here, we are using state chart and activity diagram to describe our software.

State chart diagram:-

Following are the main purposes of using State-chart diagrams −

- To model the dynamic aspect of a system.
- To model the life time of a reactive system.
- To describe different states of an object during its life time.
- Define a state machine to model the states of an object.

The main usage of state chart diagram can be described as −

- To model the object states of a system.
- To model the reactive system. Reactive system consists of reactive objects.
- To identify the events responsible for state changes.
- Forward and reverse engineering.

Conclusions

A state diagram is a type of diagram used in computer science and related fields to describe the behavior of systems. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction.

Comments :-

I. **Limitations of Results and experiment:**

Changing the order of the message results produces incorrect results.

II. **Learning happened:-**

We have learned the designing of state chart diagram and activity diagram.

State chart diagrams enable the depiction of multiple functionalities in many states and are in flow, and they are easy for even non-technical users to follow.

Component	Max Marks	Marks Obtained
Viva	6	
Results	7	
Documentation	7	
Total	20	

`

Laboratory 7

Title of the Laboratory Exercise: **Activity Diagrams**

I. Introduction and Purpose of Experiment

Students will apply object oriented analysis and design for the given scenario for object decomposition

II. Aim and Objectives

Aim

- To construct a UML class diagram for a given system and identify the class members and determine their relationships

Objectives

At the end of this lab, the student will be able to

- Identify the main members of the family
- Identify how they are related to each other
- Find the characteristics of each family member
- Determine relations among family members
- Decide the inheritance of personal traits and characters

III. Experimental Procedure

- Work in teams of 7 students
- Each team should read the problem statement and discuss the requirements as a group
- Each team will then create and confirm the design and document the design in an software design specifications document
- Each individual will then write their lab manual, documenting their observations

IV. Presentation of Results

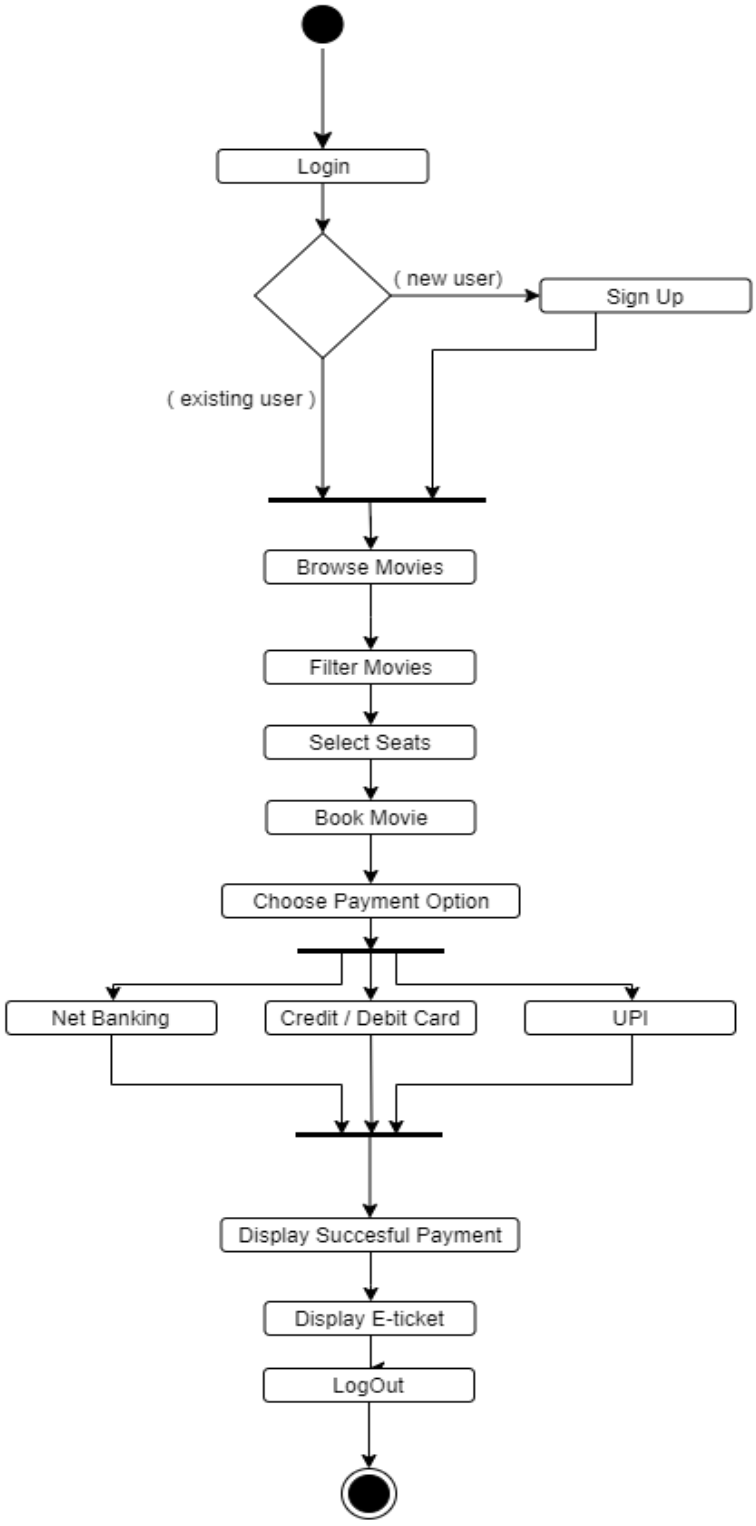


Figure 2 activity diagram

Analysis and Discussions

This lab deals with designing part of the software. Here, we are using activity diagram to describe our software.

Activity diagram:-

The purpose of an activity diagram can be described as −

- Draw the activity flow of a system.
- Describe the sequence from one activity to another.
- Describe the parallel, branched and concurrent flow of the system.

Activity diagram can be used for −

- Modeling work flow by using activities.
- Modeling business requirements.
- High level understanding of the system's functionalities.
- Investigating business requirements at a later stage.

Conclusions

In conclusion, activity diagrams are fairly easy to get the hang of, and will be useful for most projects because they plainly and moderately clearly demonstrate how things work." Unlike many diagramming techniques, these diagrams also enable the depiction of multiple choices and actors within a work flow, and they are easy for even non-technical users to follow.

NAME:- KAUSHAL VASHISTH ROLL NO: 18ETCS002147

Comments:-

1**. Limitations of Experiments**

A limitation of activity diagrams is that they may not be used in lieu of a state diagram or sequence diagram because “activity diagrams do not give detail about how objects behave or how objects collaborate.”

2. Limitations of Results

Changing the order of the message results produces incorrect results.

3. Learning happened:-

We have learned the designing of state chart diagram and activity diagram.

Component	Max Marks	Marks Obtained
Viva	6	
Results	7	
Documentation	7	
Total	20	

Laboratory 8 and 9

Title of the Laboratory Exercise: **ER – Diagram and DFD**

Introduction and Purpose of Experiment

Students will apply object oriented analysis and design for the given scenario for low level design of classes

Aim and Objectives

Aim

To develop low level software design for a given class diagram using state chart diagrams

Objectives

At the end of this lab, the student will be able to

Identify states of each object
Identify triggers and messages for each object
Understand the behavior of a class, given its state chart diagram

Experimental Procedure

Work in teams of 7 students
Each team should read the class diagram and identify objects, interactions and states of objects
Each team will then design state transitions and simulate the same. They will then document the design in an low level design specification document
Each individual will then write their lab manual, documenting their observations

Calculations/Computations/Algorithms

Presentation of Results

ER-Diagram

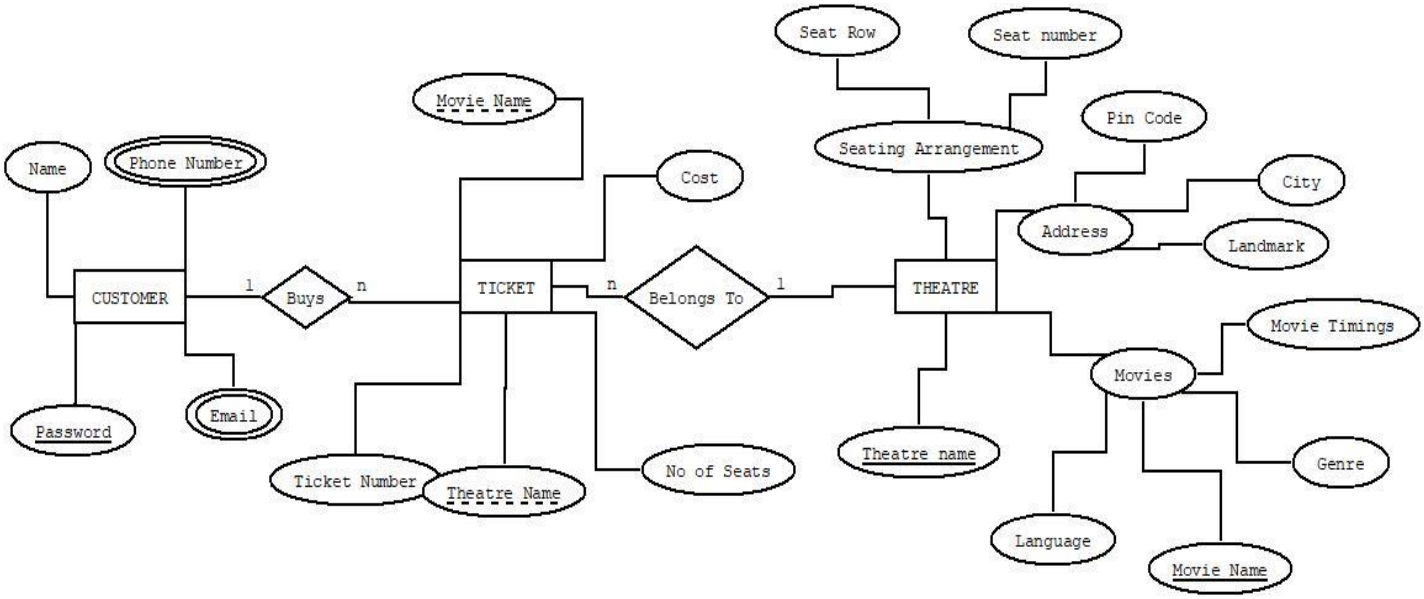


Figure 7.1 Represents the ER - Diagram

DFD – Data Flow Diagram

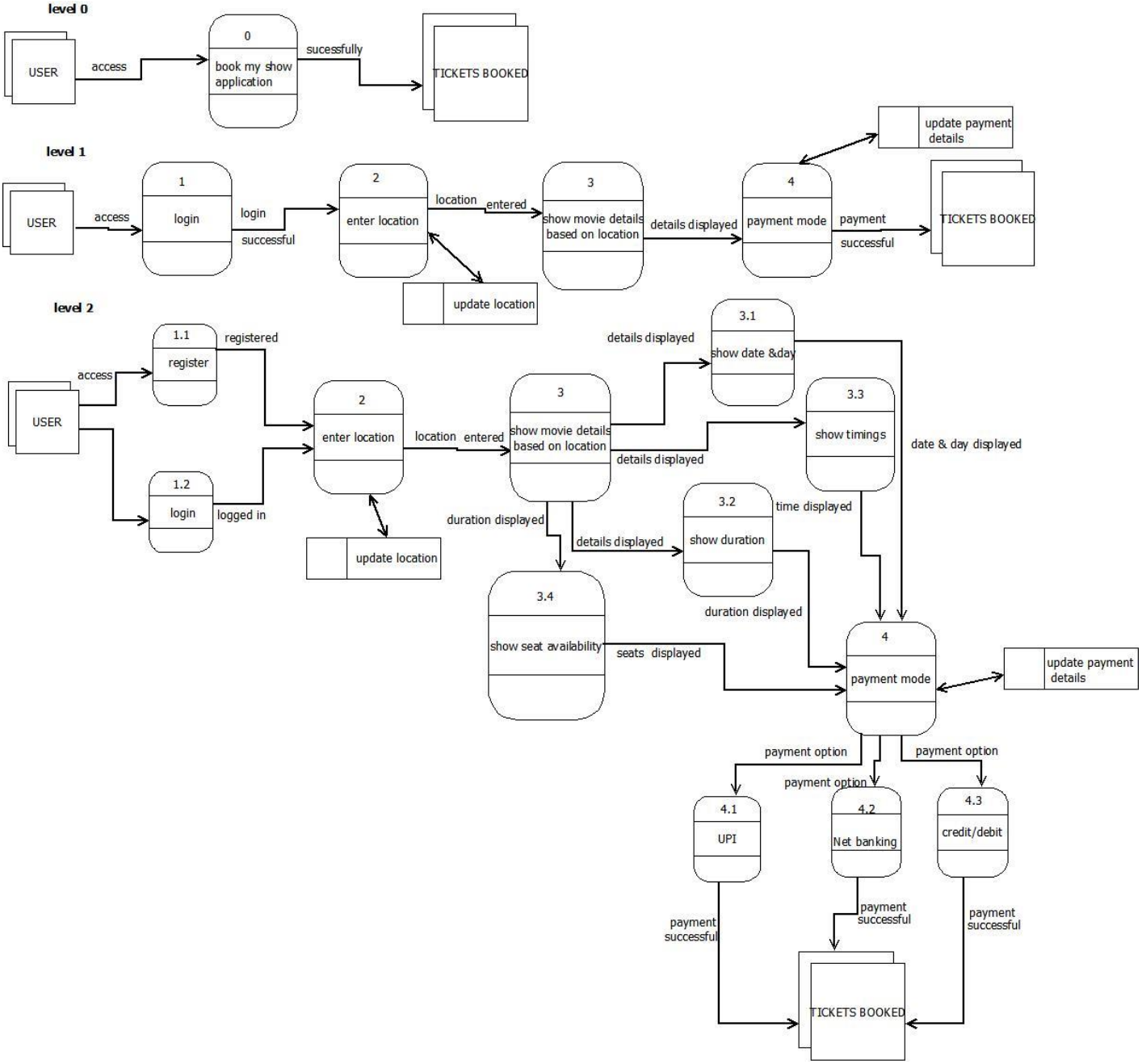


Figure 7.2 Represents the Data Flow Diagram

1.Analysis and Discussions:

In this particular lab we developed an ER – Diagram and DFD (Data Flow Diagram) for online ticket booking application. ER – Diagram has two main components that is users and OBS which further includes elements like Payment , Register , login , Payment Details, Ticket Booking process , Booking Message etc. which includes some more attributes. DFD includes three levels of diagrams . Both the diagram are behavioural diagrams.

2.Conclusions :

We developed a low level software design for a given class diagram using state chart diagrams.

3.Comments

a. Limitations of Experiments:

DFD (Data Flow Diagram) is complex to construct and it is not user friendly. In state chart diagram, the complexity increases as the number of possible states increases.

Component	Max Marks	Marks Obtained
Viva	6	
Results	7	
Documentation	7	
Total	**20**	

Lab10 : IMPLEMENTATION:-

For proper documentation, codes and live images(gif) of implementation :-

Please Do visit:-

https://github.com/Kaushalvashisth/EasyBook

(Loading images may take little time as they are large gif files)

TESTING:- (Also refer to figures below the table:)

TEST CASE ID	FUNCTIONAL REQUIREMENT NUMBER	TEST CASE DESCRIPTION	TEST DATA	EXPECTED OUTPUT	ACTUAL OUTPUT	TEST RESULT
TC 1	FR 1 (registeration)	If a customer wants to book the ticket then he/she must be registered, an unregistered user can't book the ticket. **Total steps:-** **Step1:** open the EasyBook site, landing page and click on the "Let's Go" button. **Step2:** Now, the user is on the 'Sign in Page' click on the "New User?" button and go to register Page. **Step3:** Now, the user is on Register Page and can register by	**Valid data:** ++++++++++ Username:- (6 characters or more) Eg:- blackpanther E-mail:- (an email consisting '@' sign) Eg:- bp@gmail.com	**Message on valid data:-** "Successful Registration" **Message on invalid data:-** Data 1) "please lengthen this text to 6 characters or	**Message on valid data:-** **Fig 1** **Message on invalid data:-** Data 1) **Fig 2**	PASS PASS

		entering valid data.	Password:- (between 8 to 12 characters) Eg:- "Kv123123" **Invalid Data:-** ++++++++++ **Data 1)** Username: XYZ E-mail: john Password: "cena1234" **Data 2)** Username: XYZ123 E-mail: john Password: "cena1234" **Data 3)** Username: XYZ123 E-mail: john@gmail.	more" Data 2) "please include '@' in the e-mail address ." Data 3) "password should be 8-12 characters"	Data 2) **Fig 3** Data 3) **Fig 4**	 PASS PASS

			com Password: "cena12"			
TC 2	FR 2 (login)	Customer logins to the system by entering valid user id and password for booking the ticket. **Total steps:-** **Step 1:-** After registration Click on "Go to Login" button to login into EasyBook. **Step2:-** Now, the user is on login page. **Step3:-** Enter correct credentials to login into EasyBook.	**Valid data:** ++++++++++ Username:- xyz123 Password: "12341234" **Invalid data:** ++++++++++ Username:- Anything that is not "xyz123" Password: Anything that is not "12341234"	**Message on valid data:-** "Login Success" **Message on invalid data:-** "INCORRECT CREDENTIALS"	**Message on valid data:-** "fig 5 " **Message on invalid data:-** "fig 6"	PASS PASS
TC 3	FR 3 (browse movies)	The system/software should allow the user to search movies	**Note:-** these test cases do not	**Action on click :-**		

		based on movie name, date, time and venue. **Total Steps**:- **Step 1**:- After successful login User will enter "Now Showing" section **Step 2**:- User can browse movies by clicking on "browse movies " Button in the navbar. **Step 3**:- Enter movie name and press "enter" and select desired movies.	input any data(just clicks) **Valid click:** Click on browse movies and then enter movie name and press "enter" **Invalid click:** Other buttons	Movies are Displayed on search	** fig 7**	**PASS**
TC 4	FR 4 (Selection and show timings)	The system/software should have a function to select the movie and display the show timings. **Total steps:-** **Step1:-** After successful search	**Note:-** these test cases do not input any data(just clicks) **Valid click**	**Action on click:-** On clicking "movie details ":- display "movie details" page	**fig 8**	**PASS**

		Select any desired movie. **Step 2:-** Click on the “movie details” button. **Step 3:-** user will see movie details page to select any of “Book Tickets” , “IMDB” & “Trailer”	Click on “movie details ” Then:- --- Click on “book tickets” Or Click on “ VIEW IMDB” Or Click on “Trailer” **Invalid click:-** Other buttons.	On clicking “book tickets”:- Display cinemas --- On clicking “VIEW IMDB”:- Display IMDB Page.. --- On clicking “Trailer”:- View trailer on youtube.	**fig 9** **fig 11** **fig 10**	**PASS** **PASS** **PASS**
TC 5	FR 5 (Seat Viewing and booking	The system/software should have a function to show a 2D image of the	**Note:-** these test cases do not input any	**Action on click:-**		

	ticket)	available, non-available and user selected seats and book ticket **Total Steps:-** **Step 1:-** Now, the user is on cinema select page (with the movie named on the top), Select any cinema of your choice. And click on "show timing" button. **Step 2:-** Now , the user is on **"**select date**"** page Select any date of your choice. **Step 3:-** Now, the user is on "select show timing" Select any show timing of your choice . **Step 4:-** Now, the user will be on 2-D seat viewing system.	data(just clicks) **Valid click:** Click on "show timings " Then:- Click on "any date given" Then:- Click on " any show timing " Then:- Click on "seats u want to select " Then:- Click on "proceed to	On clicking "show timings":- display "select date " page On clicking "any date given":- Display "select show timings page" On clicking " any show timing ":- Display2-D seat Selection system On clicking "seats":- Seat buttons	** fig 12** **fig 13 ** ** fig 14**	PASS PASS PASS

		Step 5:- Now, user will be able to select multiple seats of their choice and after selection click on "proceed to payment" button.	payment " **Invalid click:-** Other buttons.	get muted On clicking "proceed to payment" button go to Payment portal.	** fig 15** ** fig 16**	PASS PASS
TC 6	FR 6 (Payment)	The system/software should direct to payment gateway and display payment mode. **Total steps:-** **Step1:-** Now, the user is on payment portal. Fill the details accordingly **Step 2:-** Enter data and click on "Pay rupees X " **Step 3:-** User will see payment successful page.	**Valid data:** ++++++++++ Name on card: - (any string) Eg:- Kaushal Vashisth Cardnumber: (12 digit card num.) Eg:- 123412341234 Expiration:- (valid 2 digit month, 2 digit year and non expiring	**Message on valid data:-** "payment successful" page is displayed with Qrcode, Cancel ticket and download ticket options **Message on invalid data:-** Invalid card number:- "card number should contain exactly 12	**Output on valid data:-** **fig 17** **Output on invalid data:-** Invalid card number:- **fig 18** Invalid month:- **fig 19** Invalid year:-	PASS PASS

			date) Eg:- Month=12 Year=23 <u>CVV:-</u> (3-digit number) Eg:- "123" **Invalid data:-** <u>1) invalid card number:-</u> Eg:- 1234-1234-123 Eg:- 1234 2) <u>Invalid month:-</u> Eg:- 30 Eg:-0 Eg:- 13 3) <u>invalid year:-</u>	digits" Invalid month:- "month should be within range (0-12)" Invalid year:- "year should contain exactly 2-digits" Invalid date:- "Your card is expired " Invalid CVV "CVV should contain exactly 3-digits"	**fig 20** Expired date:- **fig 21** Invalid CVV:- **fig 22**	PASS PASS PASS PASS

			Eg:- 234 Eg:-123456 4) Invalid Date or expired Date:- Eg:- 05/20 is expired 5)Invalid CVV Eg:- 12 Eg:- 5678			
TC 7	FR 7 (Generate ticket)	After booking, the system can generate the E-ticket or then send one copy to the user's Email-address or as an SMS to user's phone number. **Total steps:-** **Step1:-** Now, the user is on Payment successful page. **Step2:-**Click on "download ticket" button. Hence, the ticket is downloaded.	**Note:-** these test cases do not input any data(just clicks) **Valid click:** Click on "Download Ticket " **Invalid click:** other buttons	**Action on click:-** Ticket is downloaded to local host and is having all the data of booking.	**Output on click:-** "E-ticket in form pdf is generated" Refer:- **fig 23** **fig 24**	PASS

TC 8	FR 8 (Ticket cancel)	The user shall be given an option to cancel ticket. **Total steps:-** **Step1:-** Now, the user is on payment successful page. **Step2**:- Click on ticket cancel button. **Step 3:-** Fill up the form and enter valid ticket number to cancel the ticket.	**Valid data:-** Ticket no:- Eg:- 18/1101628 **Invalid Data:-** Eg:- 12345	**Message on valid output:-** "Ticket cancelled successfully" **Message on invalid output:-** " invalid ticket number "	**Message on valid output:-** **fig 25** **Message on invalid output:-** **fig 26**	PASS PASS
TC 9	FR 9 (User Support)	The system/software should contain a function to contact the customer care. **Total steps:-** **Step1:-** Click on the ContactUs button in the navbar section. User will be directed to user support.	**Note:-** these test cases do not input any data(just clicks) **Valid click:** Click on "ContactUs" Button.	**Action on click:-** "CONTACT US " Page is displayed.	**Output:-** **fig 27**	PASS

			Invalid click: other buttons			
TC 10	FR 10 (Logout)	The system/software should contain a function to logout. **Total steps:-** **Step1:-** Click on the Logout button in the navbar section. User will be directed to landing page.	**Note:-** these test cases do not input any data(just clicks) **Valid click:** Click on "Logout" Button. **Invalid click:** other buttons	**Action on click:-** "landing page Is displayed"	**Output:-** ****fig 28**** ****fig 29****	PASS

Snapshots of actual results:-

FR1) Registration:-

Valid Data:-

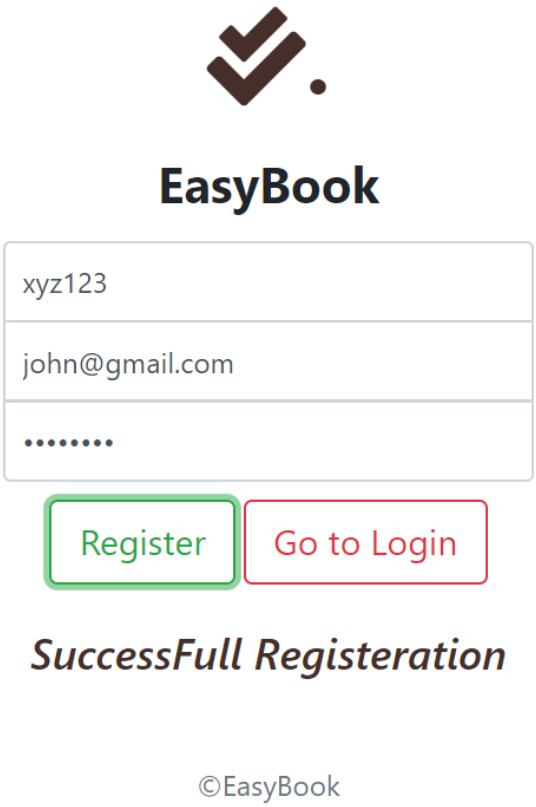


Figure 3:- (FR1) Valid data

Invalid data:-

Figure 4:- Data 1

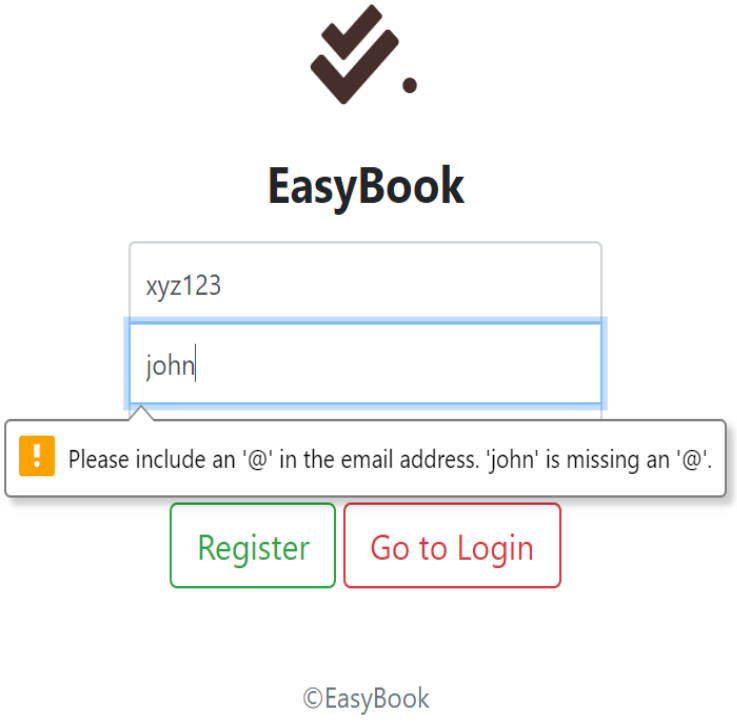


Figure 5: Data 2

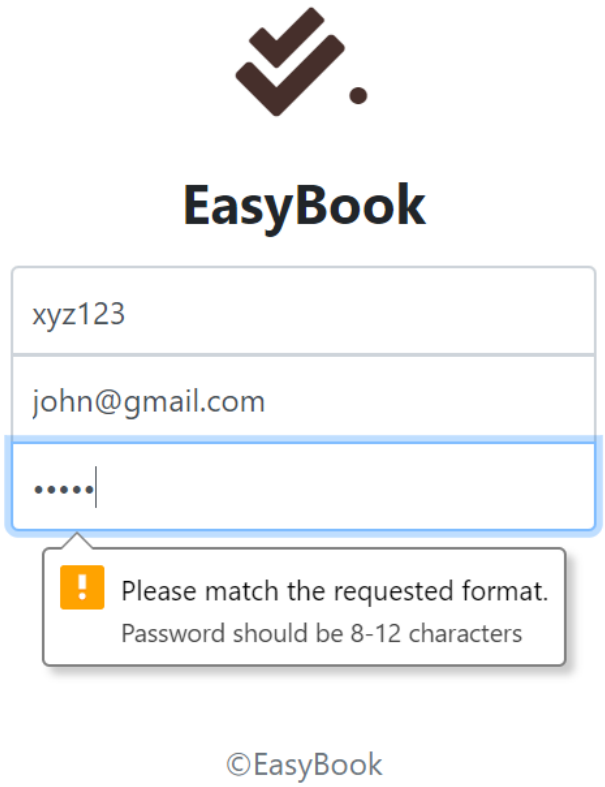


Figure 6:- Data 3

FR2) Login :-

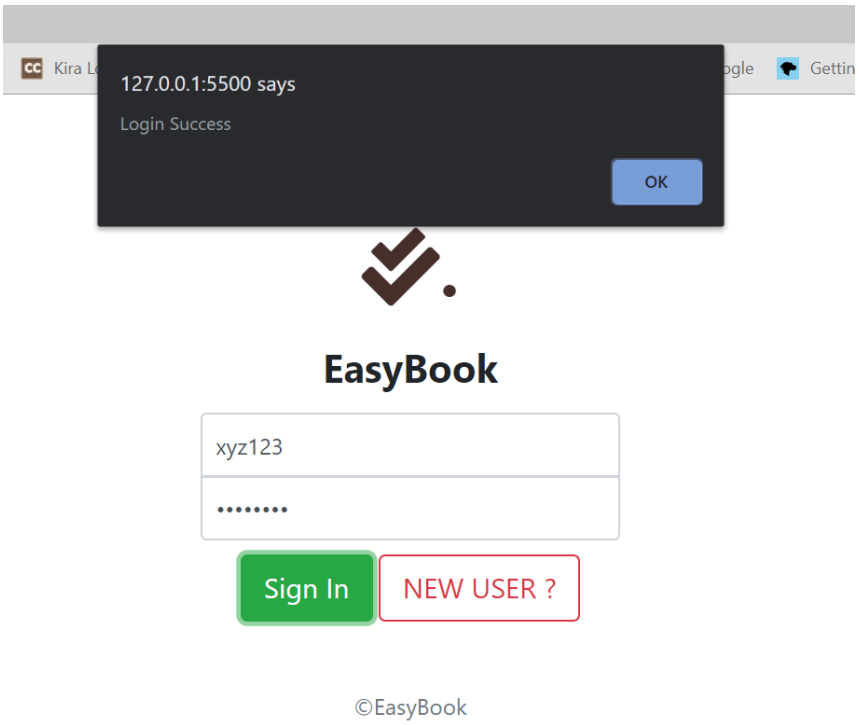


Figure 7: login success

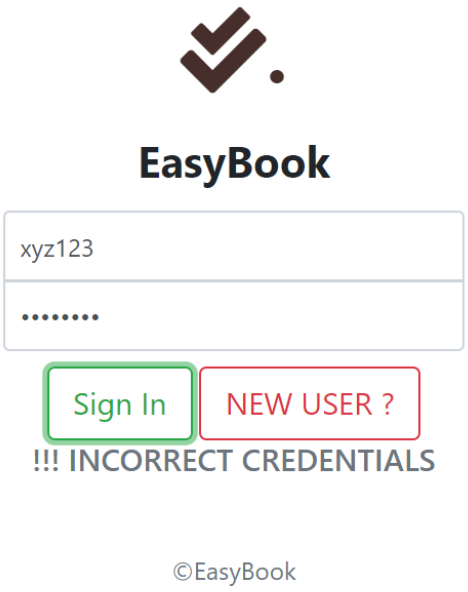
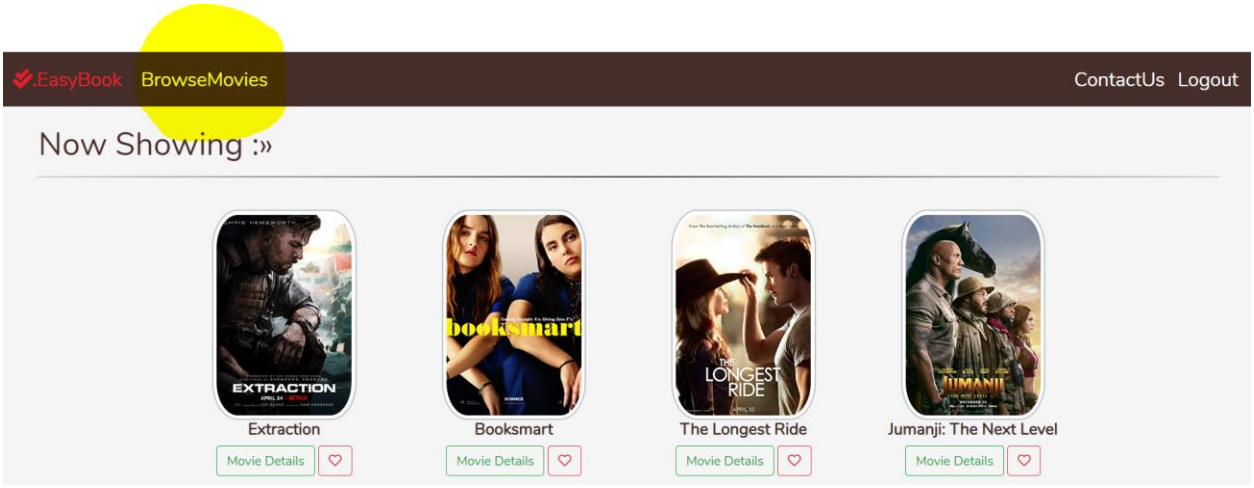


Figure 8: Login Fail

FR3) Browse Movies :-

Click on BrowseMovies in navbar(to search any movie):-

Also, user can browse several movies in “Now Showing” section.



Search for any movie:-

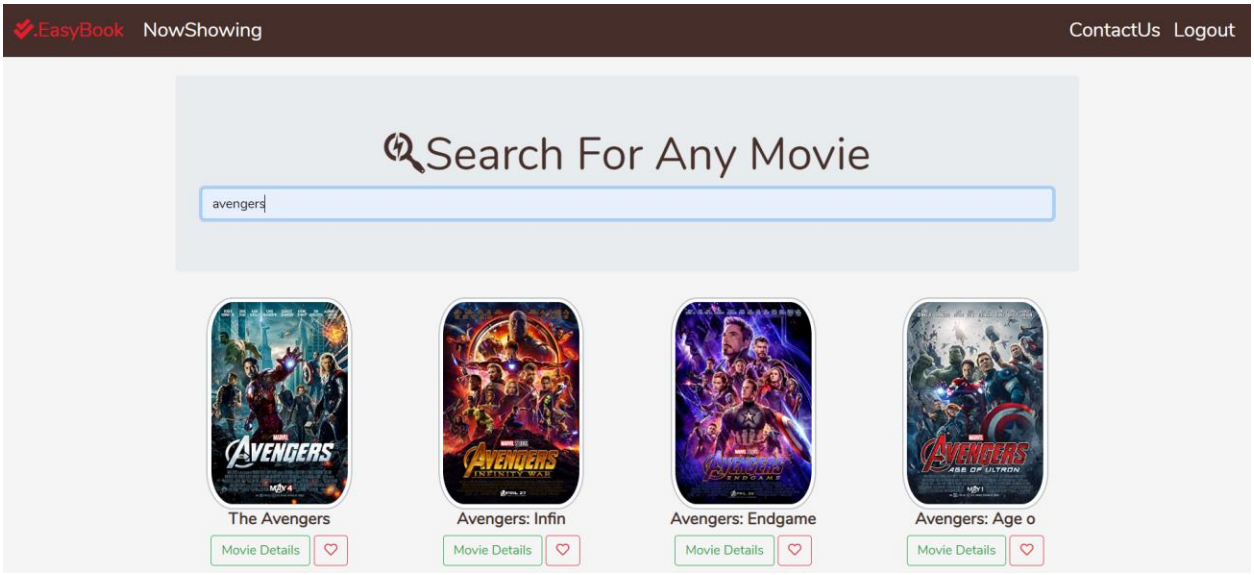
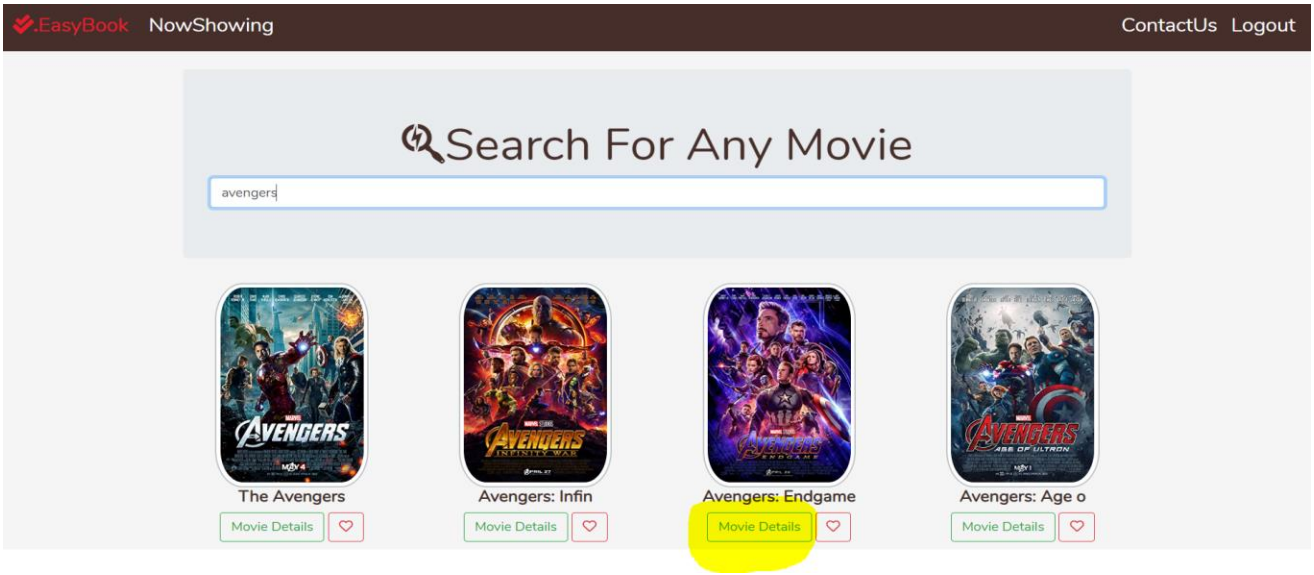


Figure 9: Browse Movies success

FR4) Selection and show timings:-

Click on movie details Page:-



Display of movie details page:-

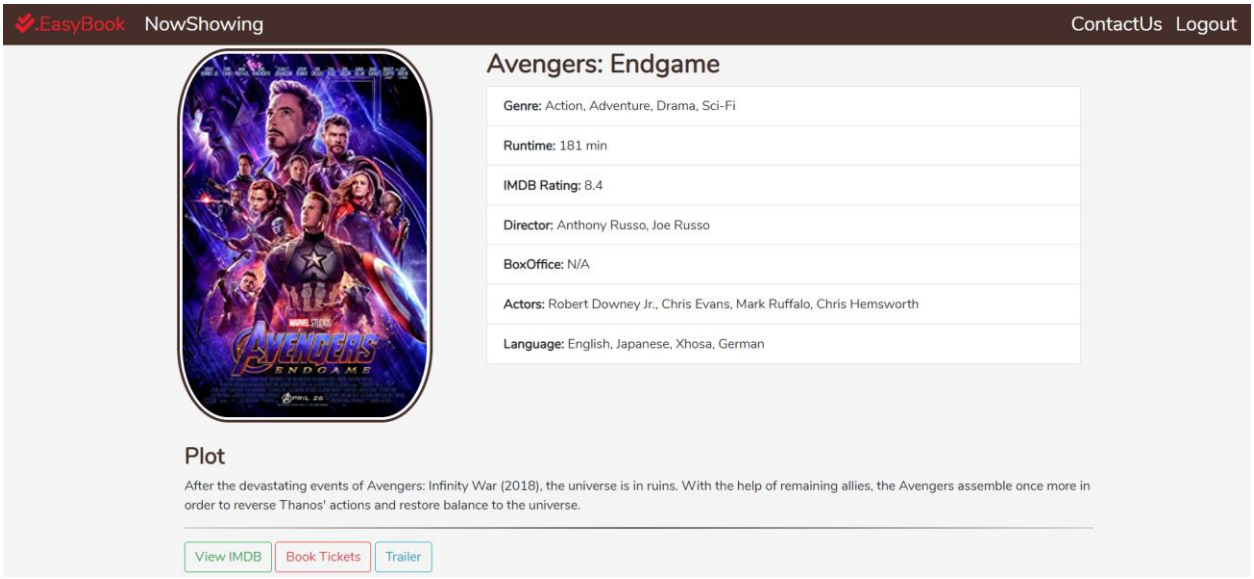


Figure 10: movie details page

After click on “Book Tickets”:-

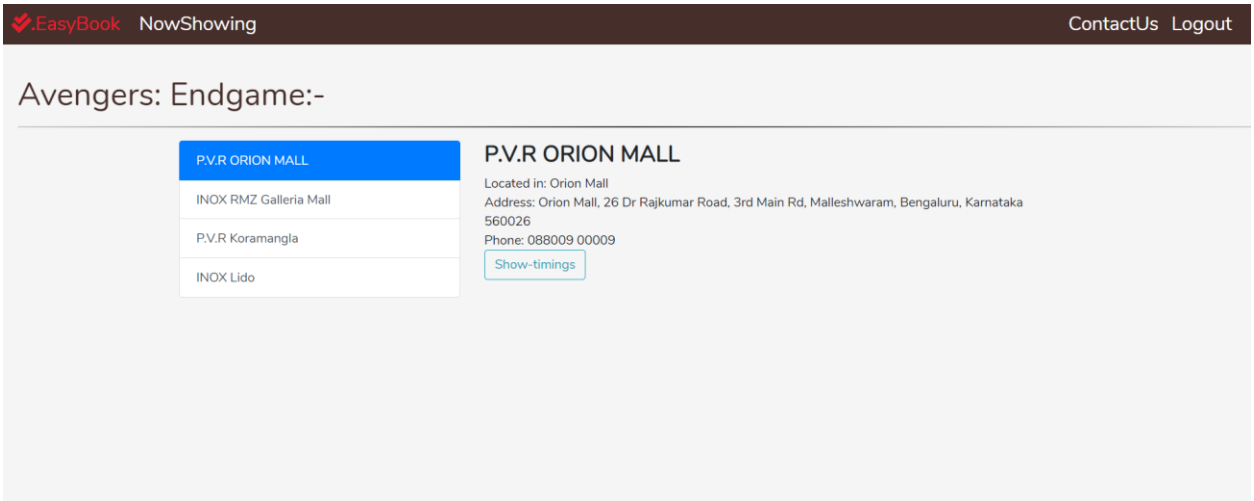


Figure 11:- on click Book Tickets

After click on “Trailer”:-

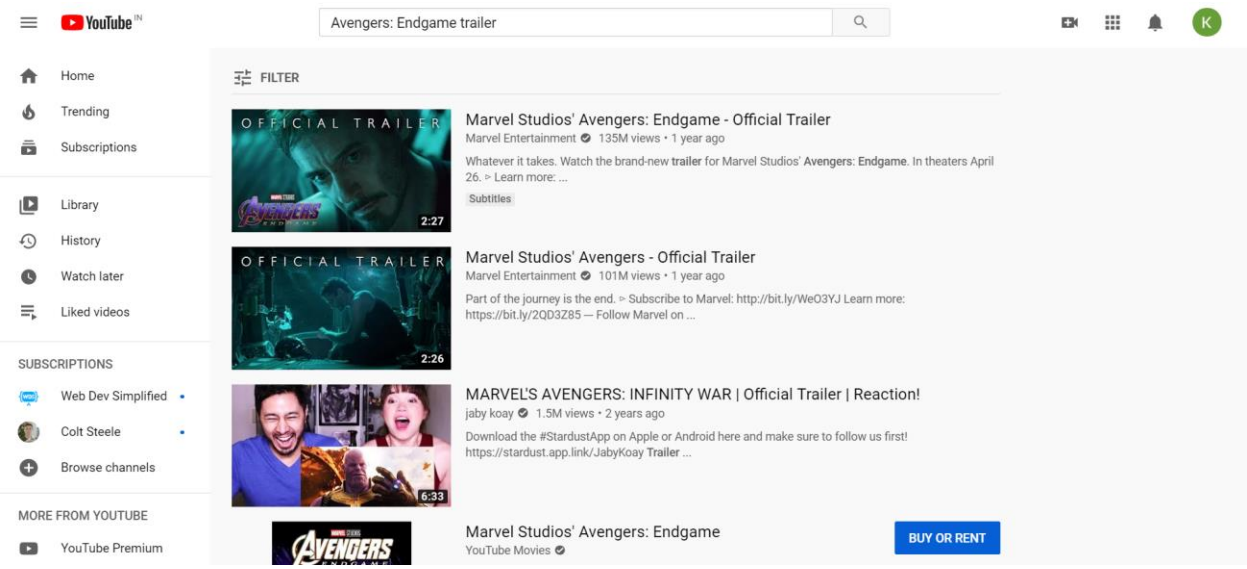


Figure 12:- on click Trailer

After click on " View IMDB" :-

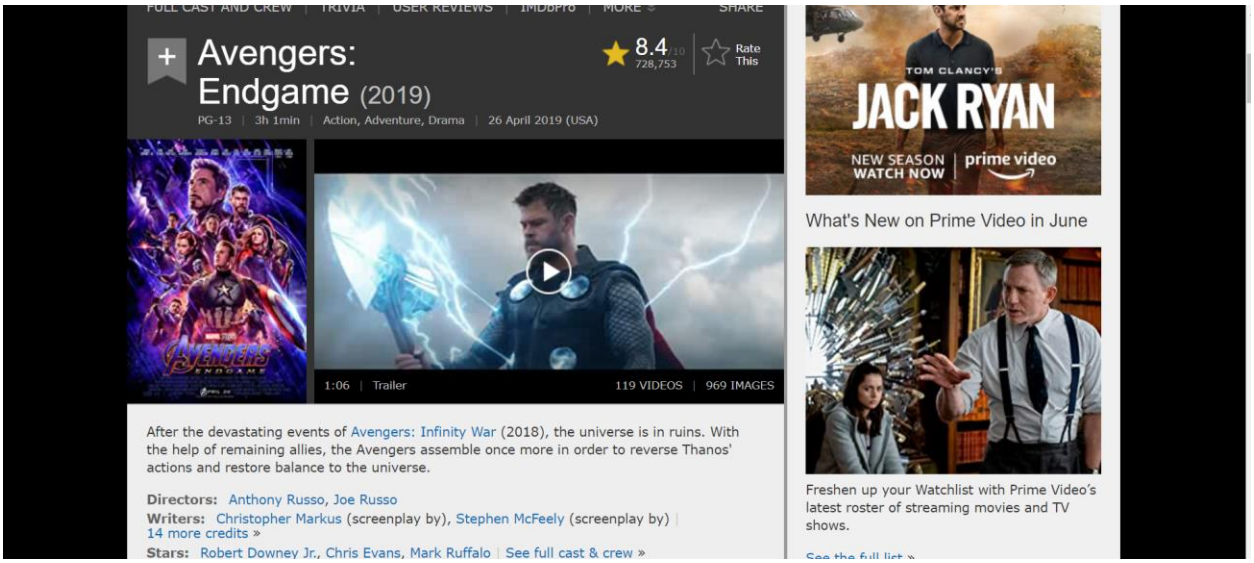


Figure 13:- on click "view IMDB"

FR 5:-(Seat Viewing and booking ticket)

On clicking "show timings" button:- (date selection appears)

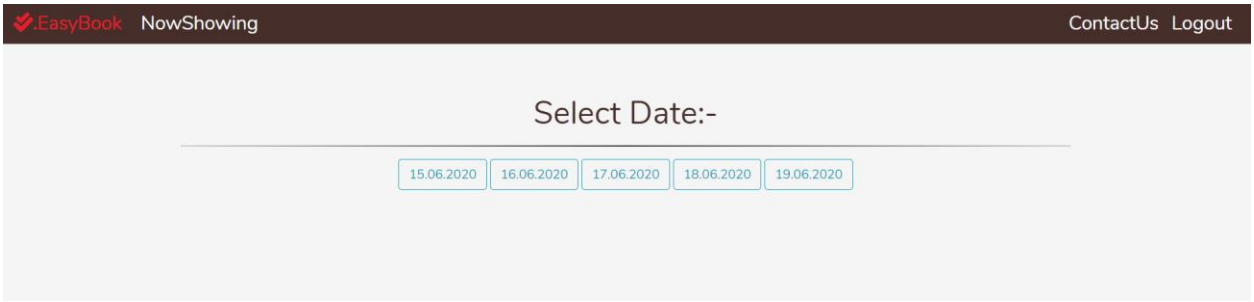


Figure 14:- select date

On clicking “any date given”:- (show timing selection appears):

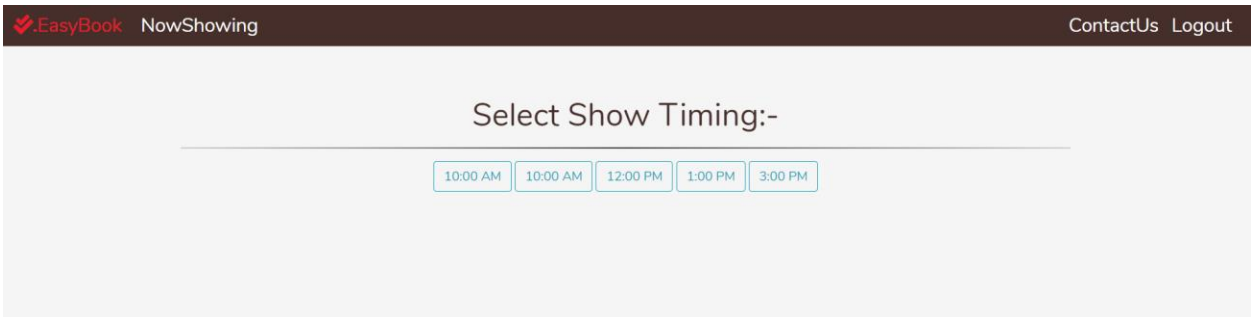


Figure 15:- select show timing

On clicking “any show timing”:-

Before:-

Figure 16:- 2-D seat selection

On selecting seats:-

Figure 17:- selection of seats

On clicking “proceed to payment” button (payment portal appears):-

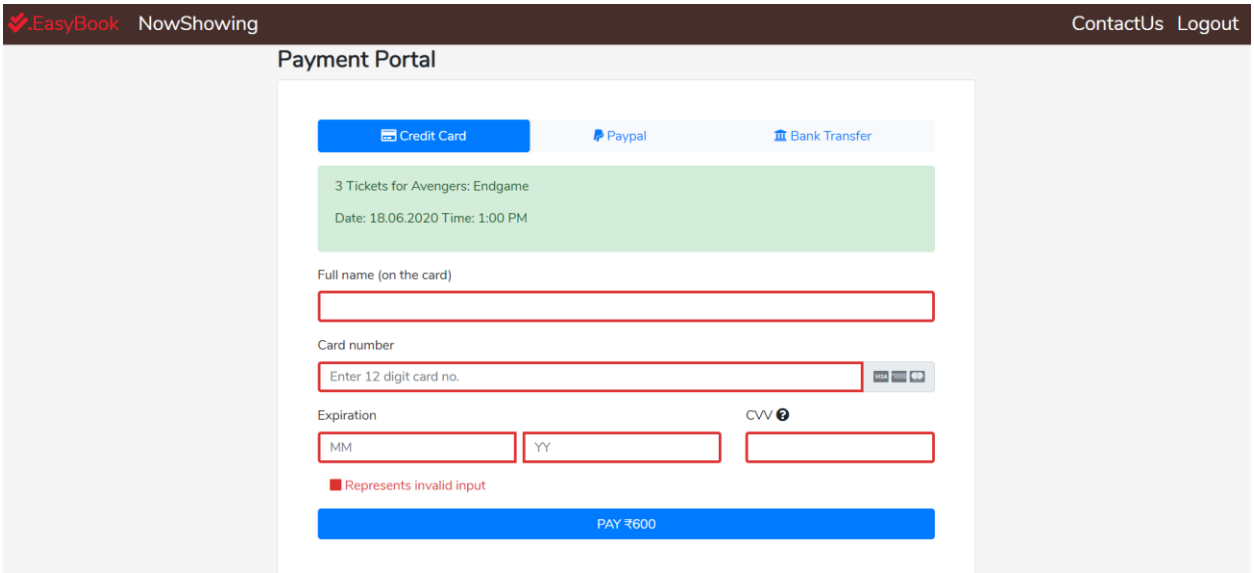


Figure 18:- payment portal

FR 6 (Payment):-

Valid Data:-

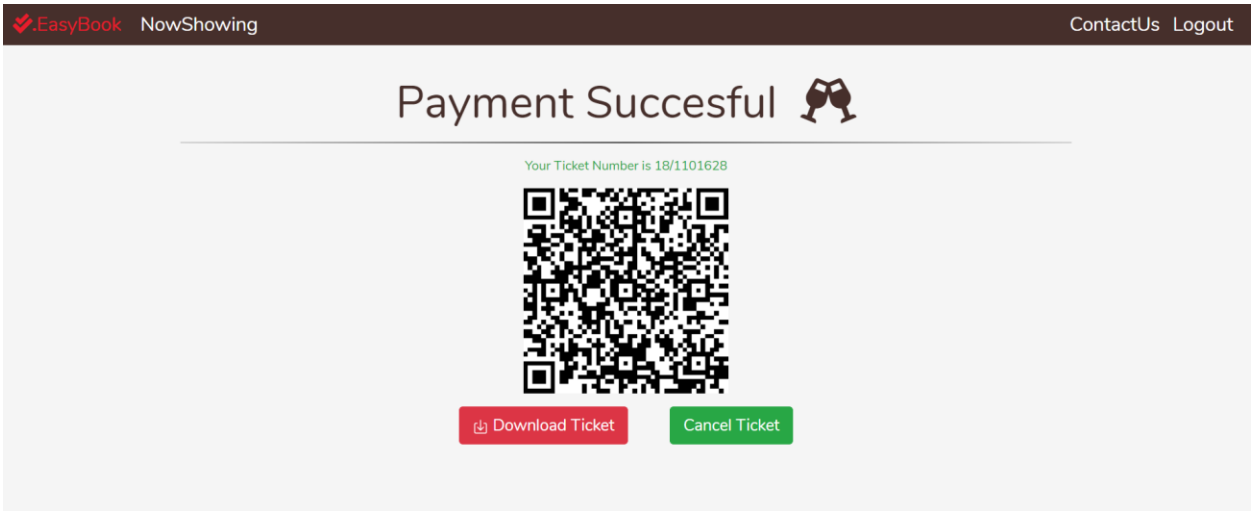


Figure 19: go to payment successful

Invalid card number:-

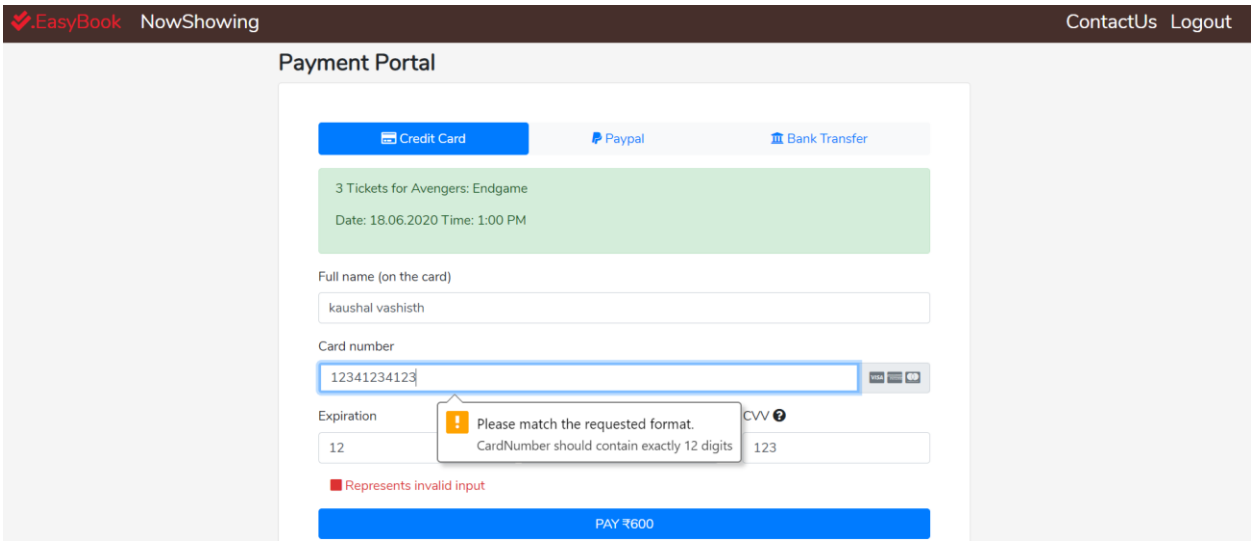


Figure 20: invalid card number

Invalid Month:-

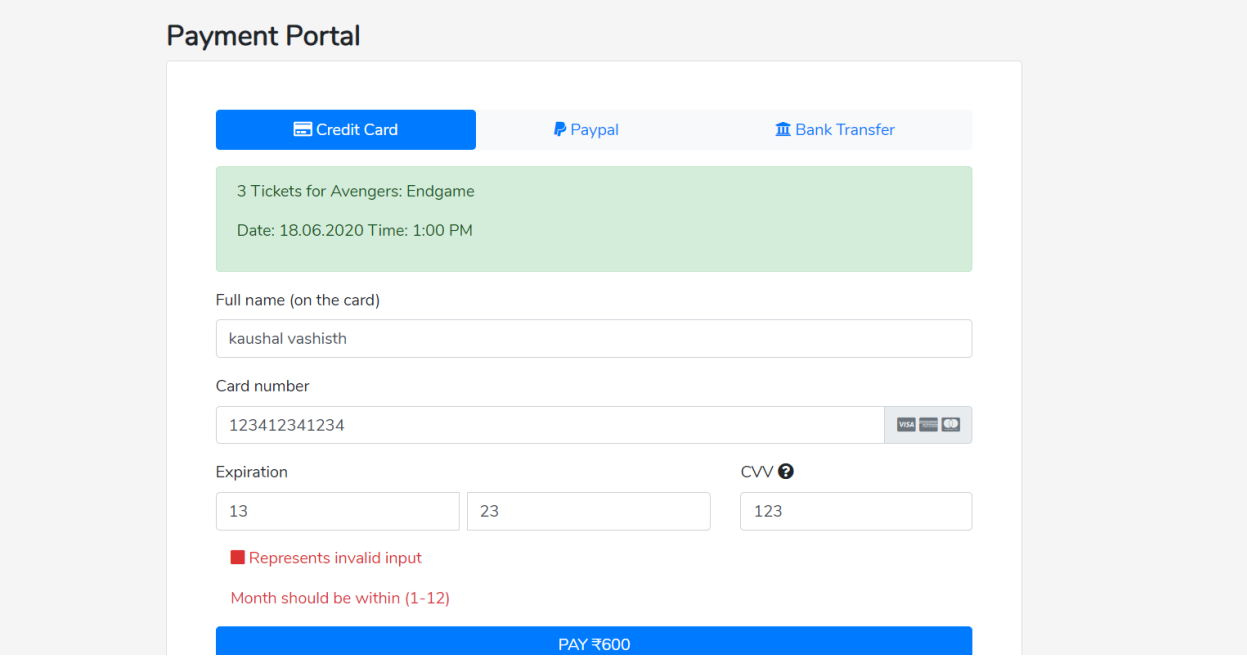


Figure 21: invalid month

Invalid year:-

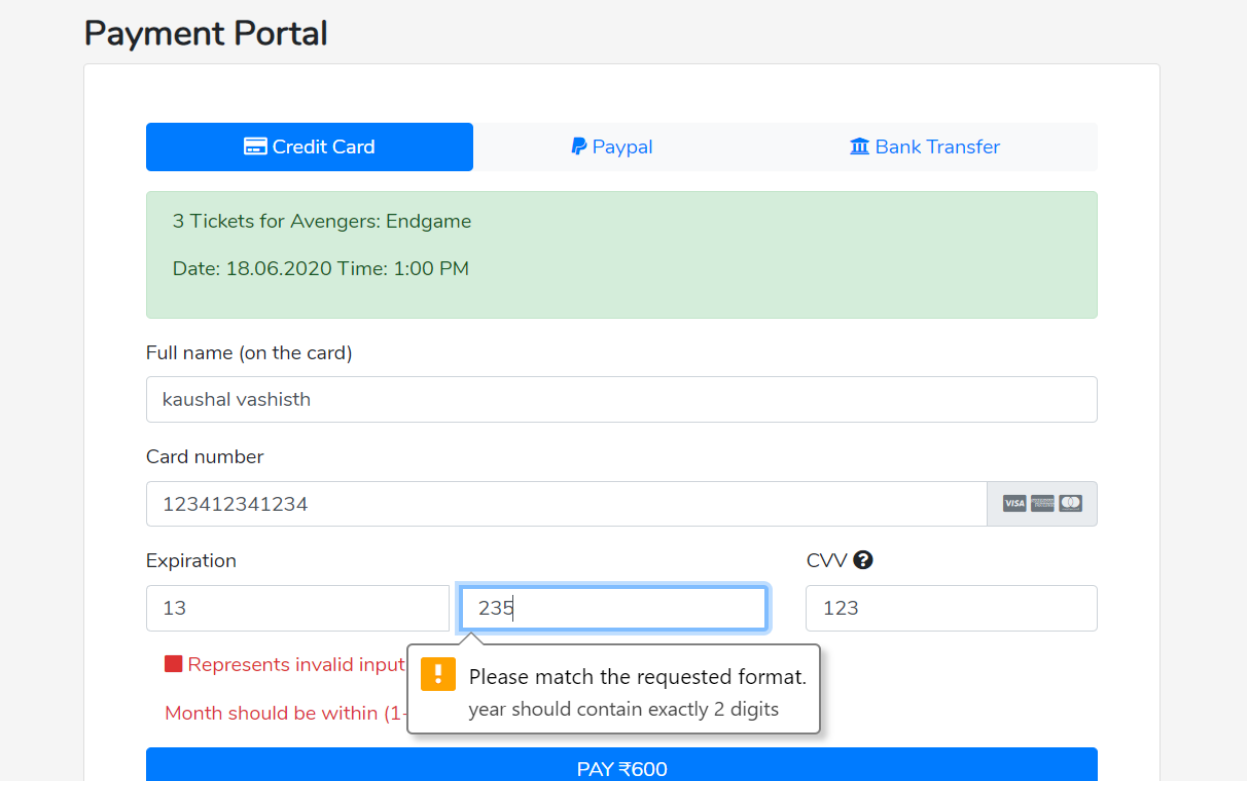


Figure 22:- invalid year

Expired card:-

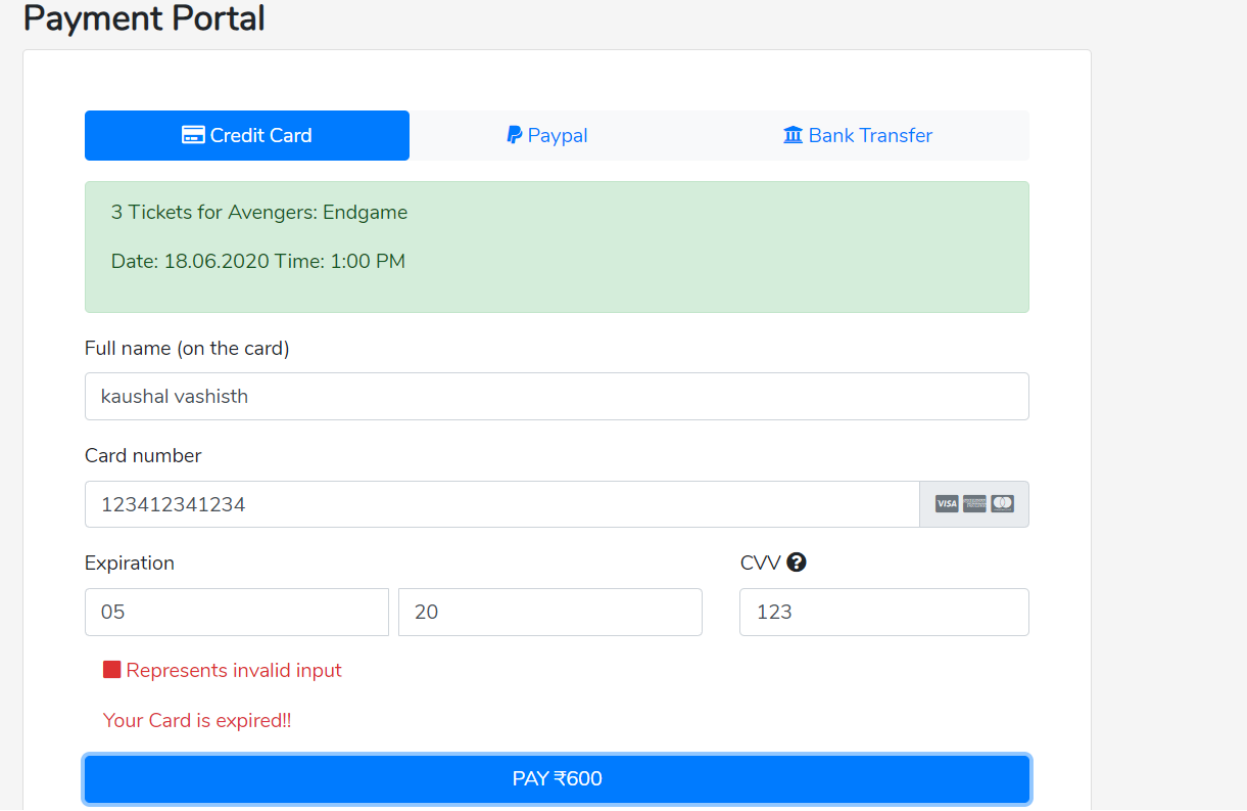


Figure 23: Card expired

Invalid CVV:-

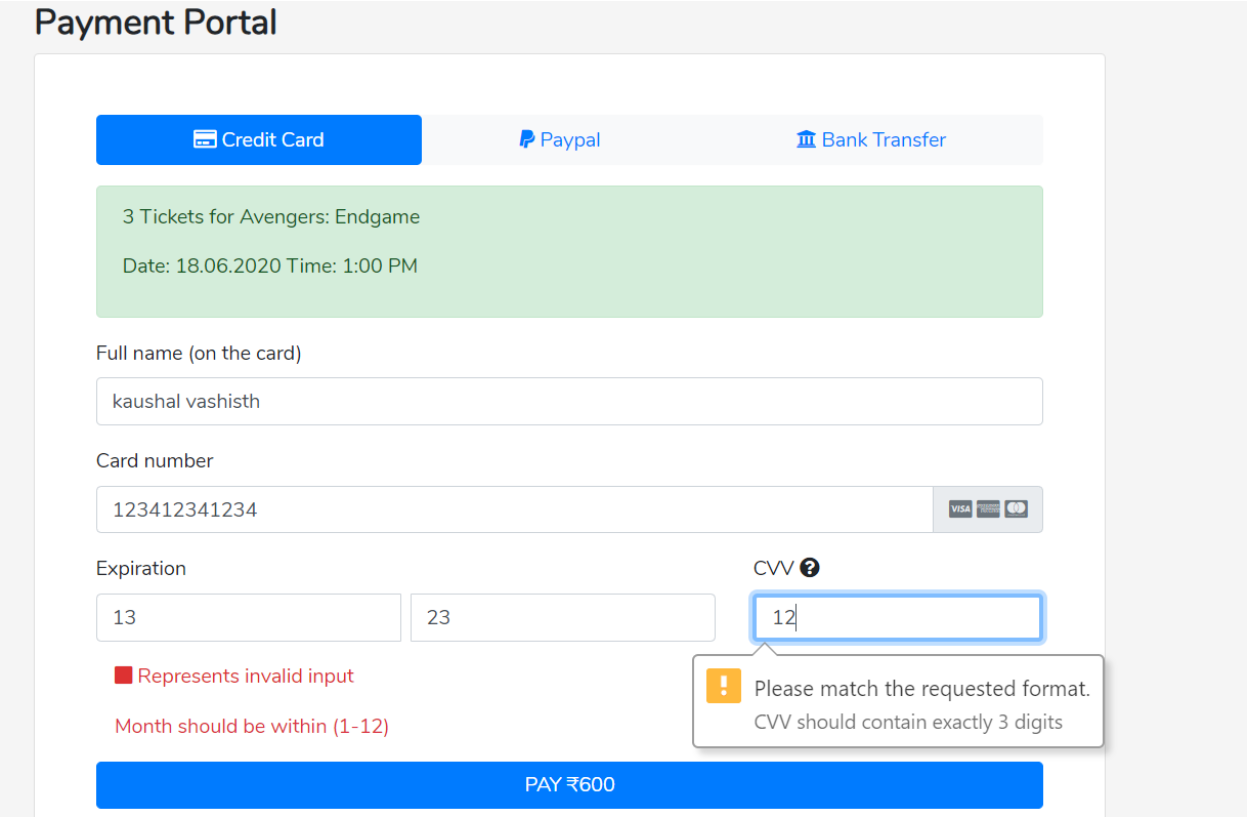


Figure 24: invalid CVV

FR 7 :

(Ticket is Downloaded):-

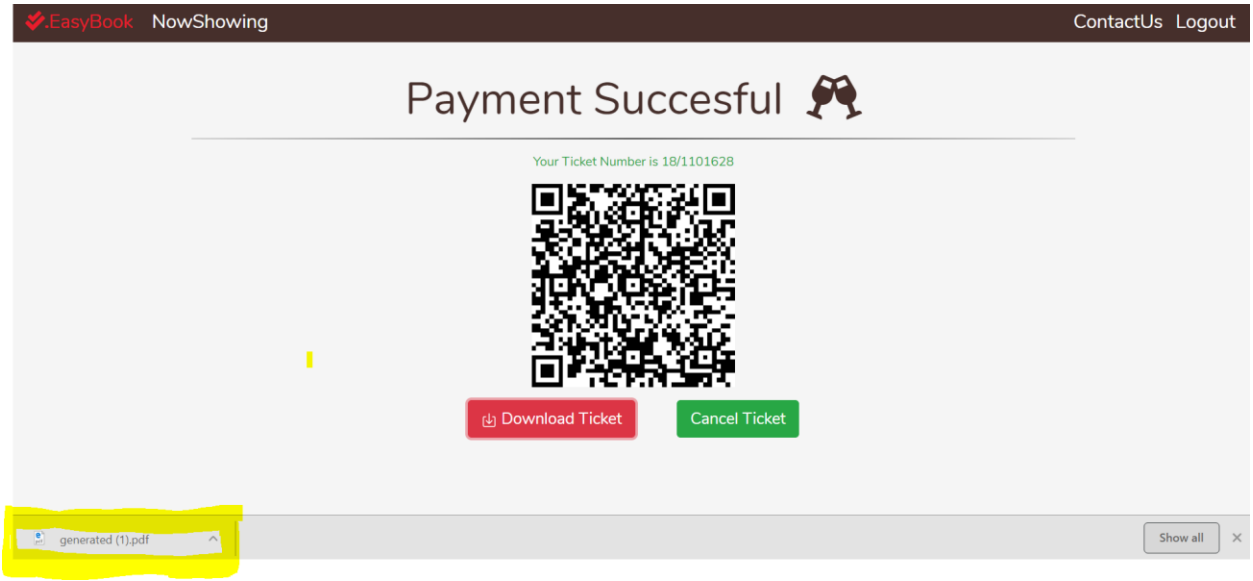


Figure 25:- ticket downloaded

(Generated Ticket):-

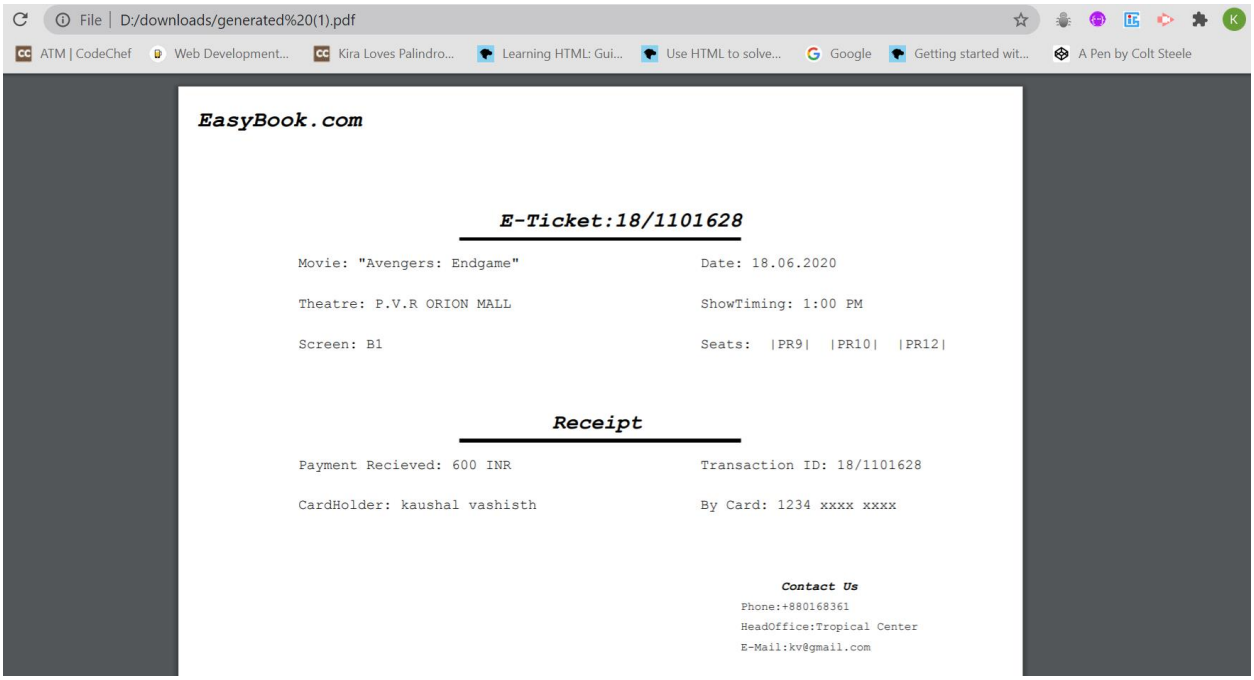


Figure 26: E-ticket in PDF form

FR 8 (Ticket cancel):

Valid ticket number

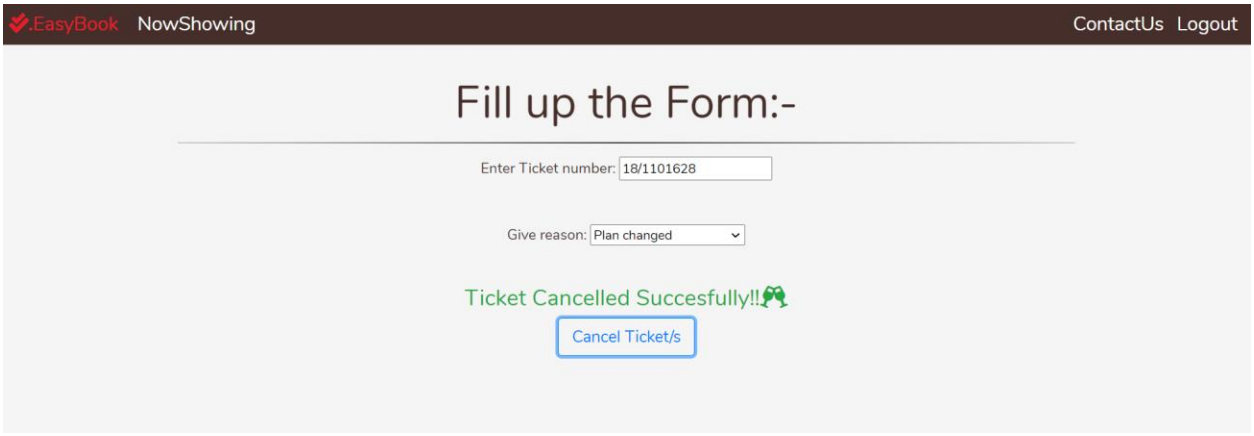


Figure 27: Valid ticket number

Invalid ticket number:-

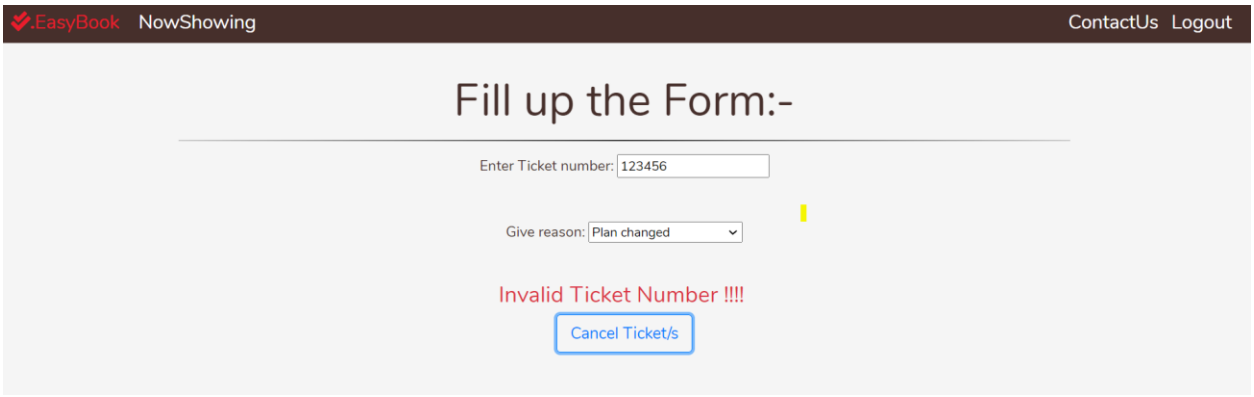


Figure 28: invalid ticket number

FR 9 (User Support or Contact Us):-

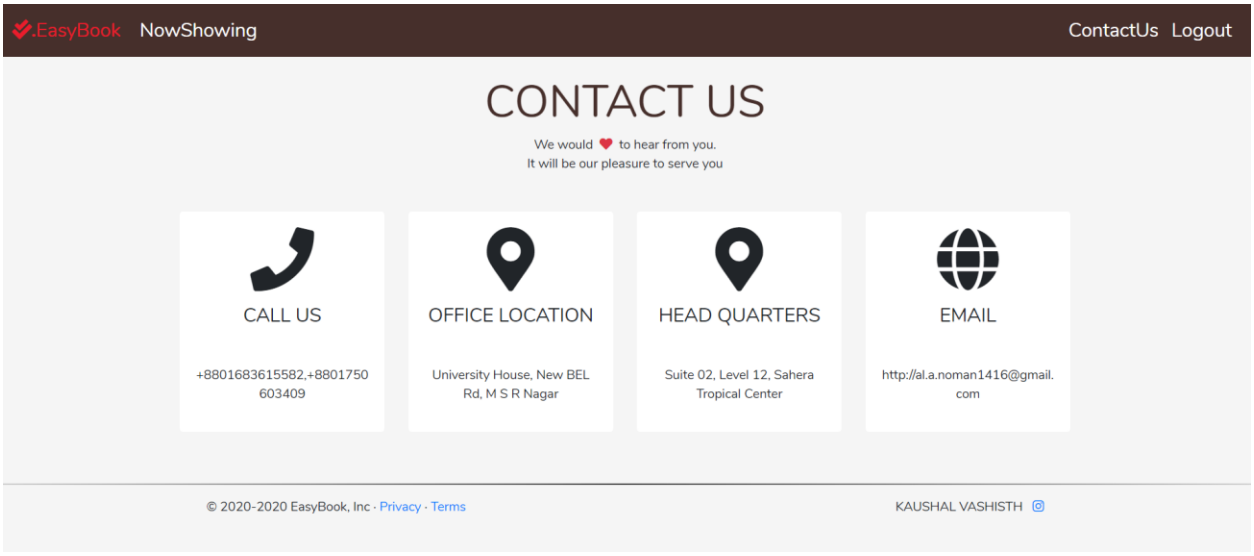


Figure 29: contact us page

FR 9 (Logout):-

Click on logout:-

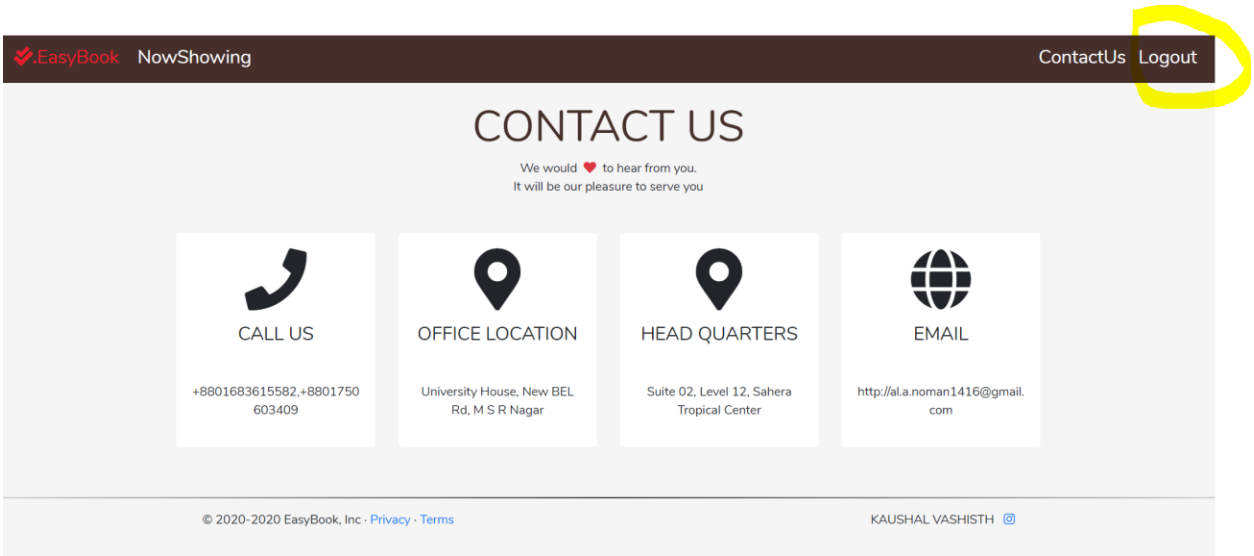


Figure 30: click on Logout

Redirect to Landing Page:-

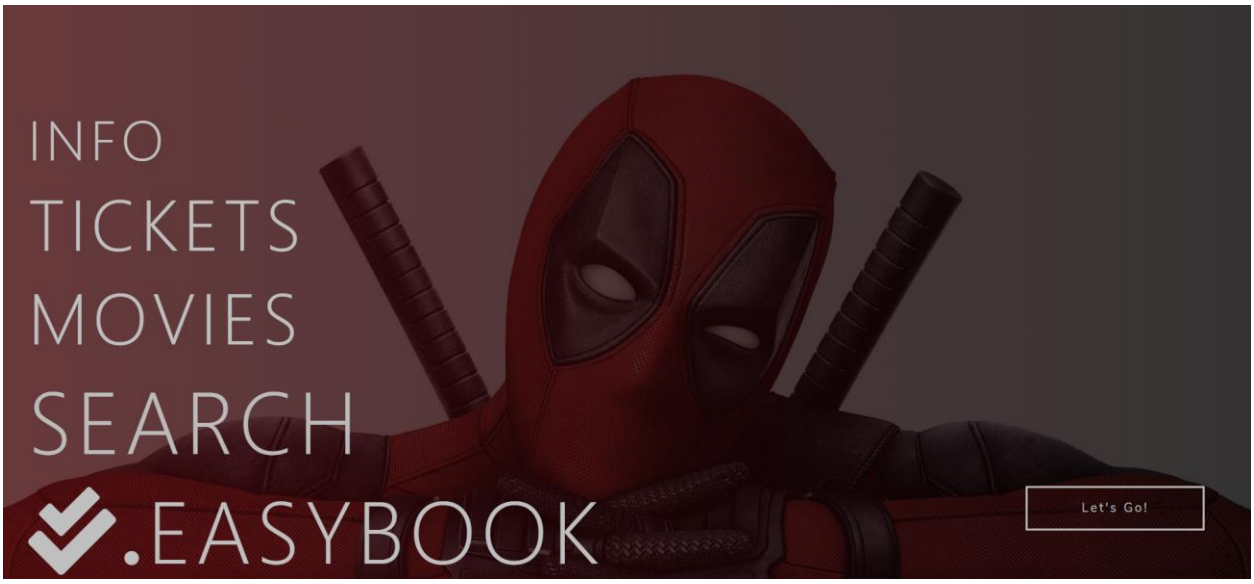


Figure 31: redirect to landing page