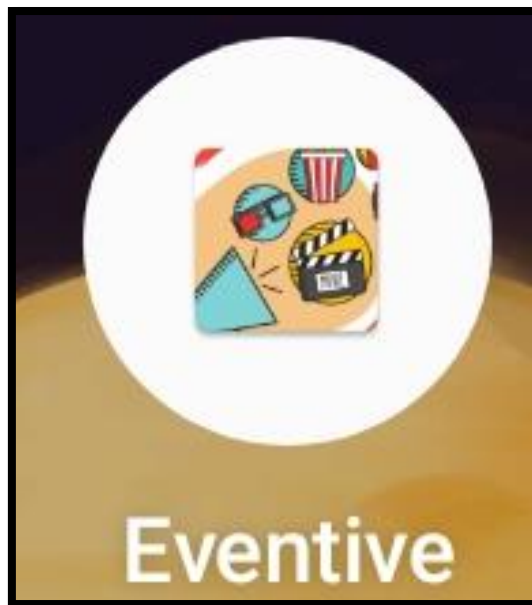




"Only Knowledge can provide salvation"

KESHAV MAHAVIDYALAYA

UNIVERSITY OF DELHI



**A SOFTWARE ENGINEERING PROJECT REPORT ON
“ONLINE EVENT BOOKING MANAGEMENT
SYSTEM”**

Submitted To:

Dr. Sumit Agarwal

Submitted By:

Ishita Rai (5710), Himangi Sharma (5839)

Dhruv Yadav (5708), Harshita Pal (5709)

B.Sc.(H) Computer Science (Section-A)

ACKNOWLEDGEMENT

We would like to express my gratitude towards **Dr. Sumit Aggarwal** for guiding us throughout the project. We also feel thankful and express our kind gratitude towards our Principal **Prof. Madhu Pruthi** for allowing us to work on “**Online Event Booking Management System**” project. Also, we would like to thank our TIC **Mrs. Maulein Pathak** for her contribution and guidance.

We feel thankful to the college staff for giving us such a big opportunity. We believe we will enroll in more such events in the coming future.

Himangi Sharma (20035570038)

Ishita Rai (20035570041)

Dhruv Yadav (20035570033)

Harshita Pal (20035570127)

CERTIFICATE

This is to certify that Software Engineering project report entitled "Hospital Management System" is the work carried out by **Ishita Rai, Himangi Sharma, Dhruv Yadav and Harshita Pal**, students of BSc(H) Computer Science IV Sem, Keshav Mahavidyalaya, University of Delhi under the supervision of **Dr. Sumit Agarwal**.

This report has not been submitted to any other organisation/institution for the award any other degree/diploma.

Supervisor

Dr. Sumit Agarwal

Teacher in Charge

Mrs. Maulein Pathak

Principal

Prof. Madhu Pruthi

LIST OF FIGURES

Name	Figure No.	Page No.
Waterfall Model	1.1	09
DFD Level 0	2.1	13
DFD Level 1	2.2	13
Use Case Diagram	2.3	15
Timeline Chart	2.4	18
Sequence Diagram	2.5	19
Architectural Design	4.1	25
Sign In Page	6.1	30
Sign Up Page	6.2	31
Sign In Using Page	6.3	32
Book Events Page	6.4	33
Event Information Page	6.5	34
Payment Page	6.6	35
Logout Interface	6.7	36
Control Flow Diagram	7.1	39
D-D Graph	7.2	40

LIST OF TABLES

Name	Table No.	Page No.
Data Dictionary	2.1	14
Cost Estimation	3.1	20
Complexity Adjustment Values Table	3.2	23
Function Point Table	3.3	23
Risk Table	5.1	26
Risk Information Sheet	5.2	27
Boundary Value Analysis Testing	7.1	41
Equivalence Testing	7.2	42

TABLE OF CONTENTS

1) Introduction	08
1.1) Aim	08
1.2) Process Model	08
2) Requirement Analysis	10
2.1) Overall Description	10
2.1.1) Product Functions	10
2.1.2) User Characteristics	10
2.1.3) General Constraints	11
2.1.4) Assumptions and Dependencies	11
2.1.5) User Interfaces	11
2.1.6) Data Interfaces	12
2.2) Functional Requirements	12
2.2.1) Booking	12
2.2.2) Event Hosting	12
2.2.3) Performance Requirements	12
2.2.4) Data Requirements	12
2.3) Data Flow Diagram	13
2.3.1) Level 0	13
2.3.2) Level 1	13
2.4) Data Dictionary	14
2.5) Use Case Diagram	15
2.5.1) Log In	15
2.5.2) Book Event	16
2.5.3) Payment	16
2.5.4) Generate Report	16
2.5.5) Selection Of Plans	17
2.5.6) Validate	17
2.5.7) Reset System	17
2.5.8) Log Out Use Case	17
2.6) Timeline/Gantt Chart	18
2.7) Sequence Diagram	19
3) Estimations	20
(i) Cost Estimation	20
(ii) Size Estimation	23
4) Designing	25
(i) Architectural Design	25
5) Risk Management and Analysis	26
5.1) Introduction	26
5.2) Risk Table	26

5.3) Risk Information Sheet	27
6) Interfaces	30
6.1) Sign In Page	30
6.2) Sign Up Page	31
6.3) Sign in using – page	32
6.4) Book Events Page	33
6.5) Events Information Page	34
6.6) Payment Page	35
6.7) Logout Interface	36
7) Testing	37
7.1) Basis Path Testing	37
7.1.1) Code	37
7.1.2) Control Flow Diagram	39
7.1.3) D-D Graph	40
7.1.4) Cyclomatic Complexity	40
7.1.5) Independent Paths	40
7.2) Boundary Value Analysis Equivalence Class Testing	41
7.3) Equivalence Class Testing	42
8) Conclusion	43
9) References	44

Chapter 1

INTRODUCTION

1.1) AIM

The main aim of this project is to develop the software for the process of booking events and movies which should lead to increased efficiency and reduced drawbacks which were present in the previous procedure of online ticket booking and makes it convenient for the customer to book a particular ticket. The software should control redundancy so that no two customers can access the same seat at the same time and transactions should be independent. Moreover, ticket booking has been a hassle for the customer, they are left with just an option i.e. book the tickets by visiting the venue. Technical system is fast, accurate, informative, reliable and user friendly.

The proposed system enables the customer to do things such as search for movies that are available on a specified date for a particular venue. The system displays all the movie details such as price, duration, venue, time, date.

1.2) PROCESS MODEL

Waterfall process model is being used in our project.

This model is generally applied to the projects where the requirements are very clear and the customer keeps on patience.

Different phases of model:-

1. Software Requirements Analysis

In this, software engineer understands the nature of a program to be built, he must understand the information domain for the software as well as required function, behavior, performance and interface. Requirements for both the system and software are documented and reviewed with the customer.

2. Design

It has four distinct attributes of a program: data structure, software architecture, interface representations and procedural details. It is documental and becomes part of software.

3. Code Generation

Design must be translated into a machine-readable form which is done by code generation.

4. Testing

It focuses on the logical internals of the software, ensuring that all the statements have been tested, and on the functional externals, that is conducting test to uncover errors and ensure that defined input will produce actual results.

5. Support

This is a phase when software will undoubtedly undergo change after it is delivered to the customer. Change will occur because errors have been encountered, because the software must be adapted to accommodate changes in its external environment, or because the Customer requires functional or performance enhancements.

The advancement in program does not need to be checked upon by the customer during the process. So, this model does not create problem.

- The requirements are fixed and work can proceed to completion in a linear manner.
- The Waterfall Model provides a structured approach.

Waterfall model is used because of the following reasons:

- It's relatively simple and easier to understand approach as compared to other models.
- The requirements are well stated and understood before in hand.
- In this model we have to complete one stage before proceeding to next. So, we have clearly defined stages and well understood milestones.

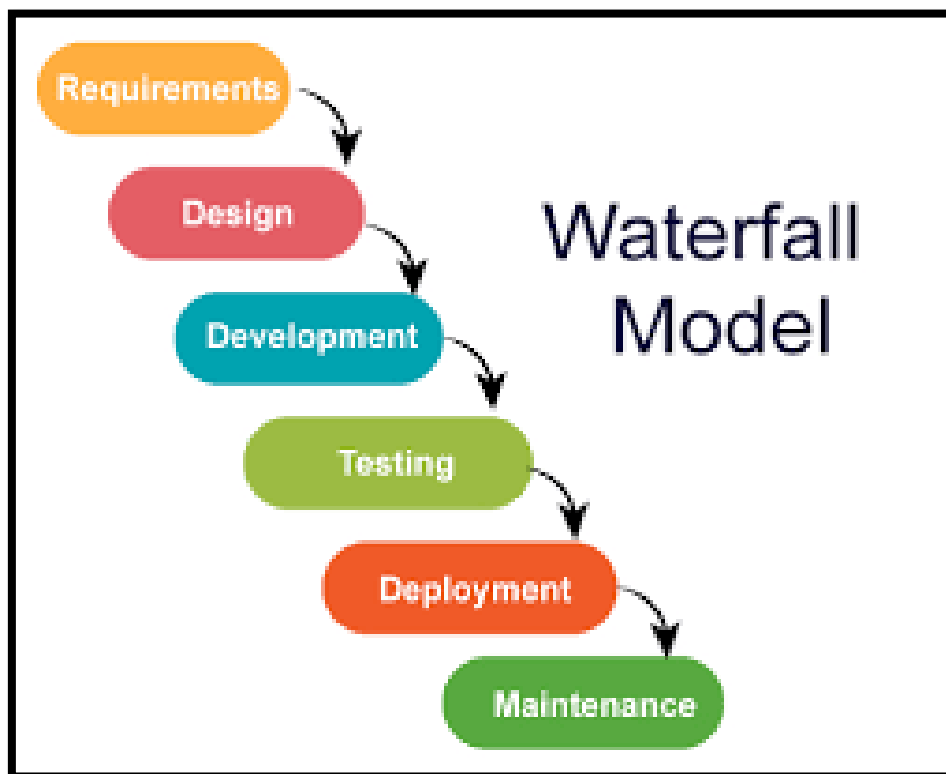


Figure 1.1

Chapter 2

REQUIREMENT ANALYSIS

2.1) Overall Description:

2.1.1) Product Functions:

Our online event management system, application -based system. The customers buy tickets online. Admin can use the system to insert and delete data (e.g. film description, timetable) which will update the webpage (webpage are dynamic page, changing according to the data in database). Also, admin can check the statistic information from the system.

The project is being developed keeping in mind the following objectives:

- Reduce the paperwork and storage area.
- Improve the output of operators.
- Improve accuracy in result.
- Manage the man and machine resources efficiently.
- It has user friendly interface having quick authenticated access to documents.
- More security and safety of data.
- Quick query processing.
- Easy access to data.
- Reliable and efficient.
- Increase the processing speed.
- Quick access to appropriate candidate.
- Easy retrieval of information.

The features of this product are as follows:

- Prevention against unauthorized access
- Prevention against spamming.
- Easy navigation.
- Online ticket booking of all movies and events at any time.
- A presentable, fast and easy user interface.
- Minimizes the number of staff at the ticket box.
- Online promotion of the events and movies.
- Obtains statistic information from the booking record.

2.1.2) User Characteristics:

- Customer uses the system to fetch information about available events and movies, their price and majorly to book tickets

- Admin manages the system and keeps it up-to-date. Admin also looks over customers and organizers.
- Organizers use the system to list their events, movies and other related information like availability, price, duration, date and timings.

2.1.3) General Constraints:

- User interface is only in English. No other language option is available.
- An internet connection is required to use the system. A broadband (DSL or cable) or faster connection is recommended.
- The files in which the information regarding account should be secured against malicious deformations.
- Data should not become corrupted in case of system crash or power failure.

2.1.4) Assumptions and Dependencies:

- Admin is created in the system already.
- Roles and tasks are predefined.
- There is no limit for booking for the website. If this were so then the instantaneous communication between the two systems would not exist.
- In general it has been assumed that the user has complete knowledge of the system that means user is not a naïve user. Any data entered by him/her will be valid.
- It depends that the one should follow the international standards for the generating the User ID & should fill the related information in the proper format.
- Central server of the system must be able to handle all the incoming requests simultaneously.
- Backup of the databases in case of hardware failure, disaster, natural calamities.
- No data loss in case of handling of the system by the administrators or the system related personnel.
- Username are valid email addresses of respective user
- Administrator has the authority to reserve/cancel movie tickets.

2.1.5) User Interfaces:

- It is a login window that requires user to enter correct ID and password, so that after matching values stored in the database you are proved as a valid user to enter into the application.
- If the user does not exist, then the user must register in order to access the system functionalities. ID and password will be stored in the database for future login purpose.
- Customer logs in to book event tickets by selecting event, date, time, venue and number of tickets required.
- Organizer logs in to list his/her event on the system or to update his/her existing event.

2.1.6) Database Interfaces:

- All databases for the software will be configured. These databases include events' details database, customers' details database, and organizers' details database.
- The customers' details database includes userID, password and previous and current booking details.
- The events' details database includes information about all listed events, their organizers, dates, timings, venue, price per ticket and available seats.
- The organizers' details database includes organizers' ID, password and previous and current listed events' details.

2.2) Functional Requirements:

a) Booking:

This module basically deals with the booking of events. It accepts order details, retrieve event details, processes the order based on customer requirements and finally processes the payment to generate tickets. After successful transaction, customers' details database is updated with E-tickets and other details, events' details database is updated with the number of seats left after last transaction and other details, organizers' details database is updated with number of tickets booked and left and other details also.

b) Event Hosting:

This module basically deals with the hosting of events. It accepts event details, add event details. After successful hosting, events' details database is updated with order and customer details, events' details database is updated with the number of seats and other event details, organizers' details database is updated with number of tickets booked and left and other details also.

c) Performance Requirements:

- Each component of the system can accept and work with only one account at a time. Internet access is required for filtering events, booking tickets, adding events and making payments.

d) Data Requirements:

- The system can store the following data: text data including name, address, event details, venue and numeric data including number of tickets, price, date, time and much more.
- Assuming a user has correctly authenticated him/her via email ID, system will read back any of the previous data.
- The system will also maintain a backup database.

2.3 Data Flow Diagram:-

DFD – LEVEL 0

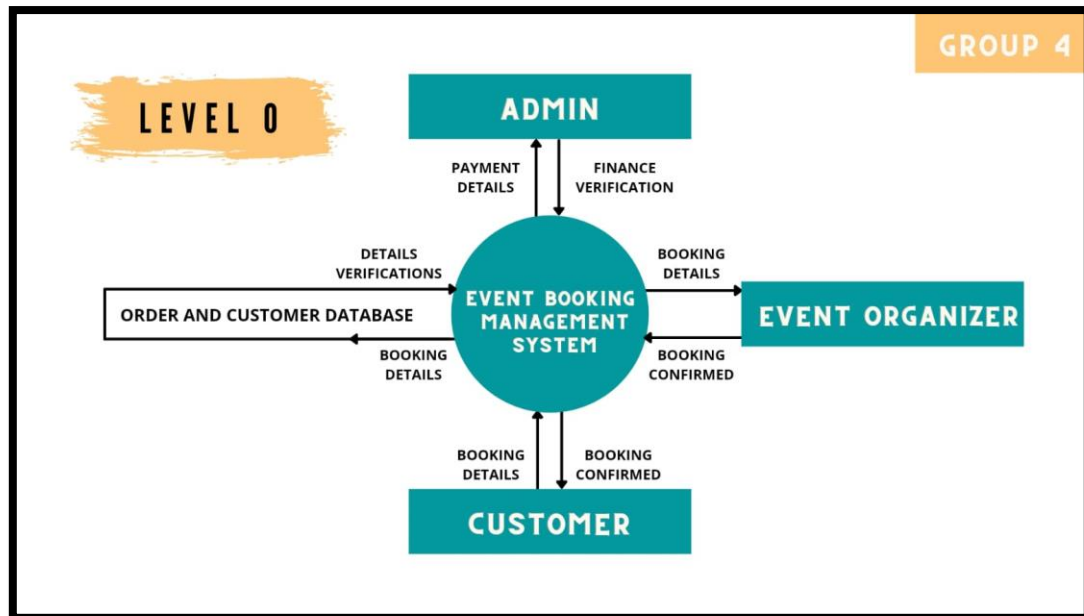


Figure 2.1

DFD – LEVEL 1

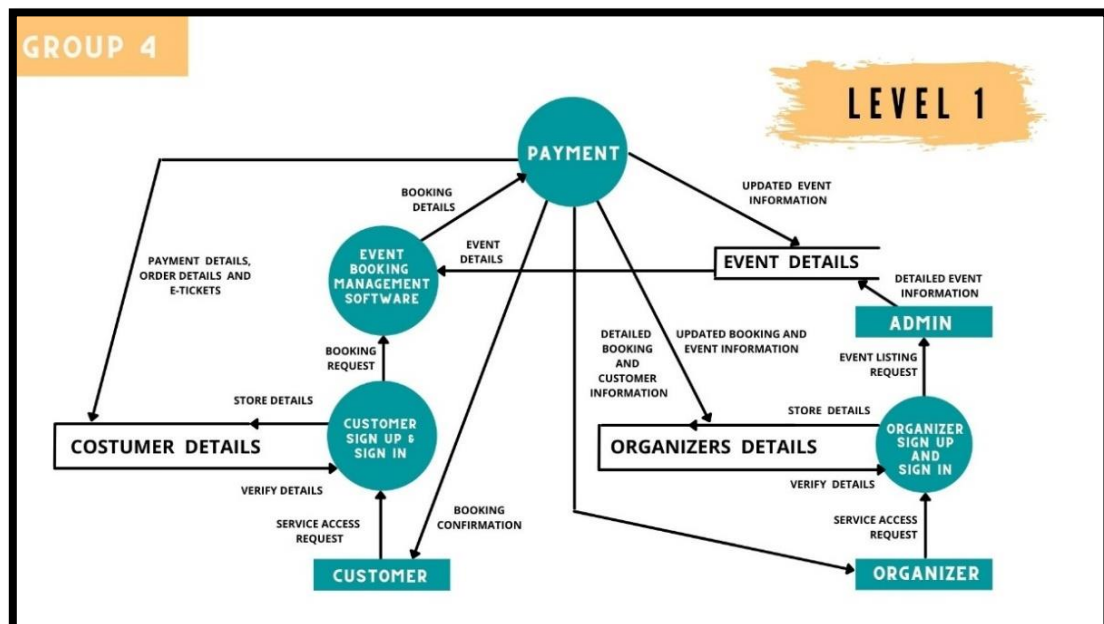


Figure 2.2

2.4) Data Dictionary

Table 2.1

DATA FLOW ITEMS	SUB-FIELDS	DESCRIPTION	DATA TYPE
CUSTOMER	Name	Name of the customer	Char(30)
	Phone	Mobile number of the customer	Int(10)
	Address	Address of the customer	Varchar(50)
	Email	Unique Email address of the customer	Varchar(30)
	Password	Password of the customer	Varchar(30)
	ID	Unique ID of the customer	Varchar(30)
ORGANIZER	Name	Name of the organizer	Char(30)
	Phone	Mobile number of the organizer	Int(10)
	Address	Address of the organizer	Varchar(50)
	Email	Unique Email address of the organizer	Varchar(30)
	Password	Password of the organizer	Varchar(30)
	ID	Unique ID of the organizer	Varchar(30)
ADMIN	Purchase ID	Unique order ID	Int(10)
	Venue	Venue of the event	Varchar(30)
	Timing	Timing of the event	Date(10)
	Event Name	Name of the event	Char(30)
	Number of Tickets	Available tickets of the event	Int(10)
	Price	Price of a ticket of the event	Float(10)
	Event ID	Unique event ID of the event	Int(10)
	Total Cost	Total cost of the order	Float(10)

2.5) Use Case Diagram

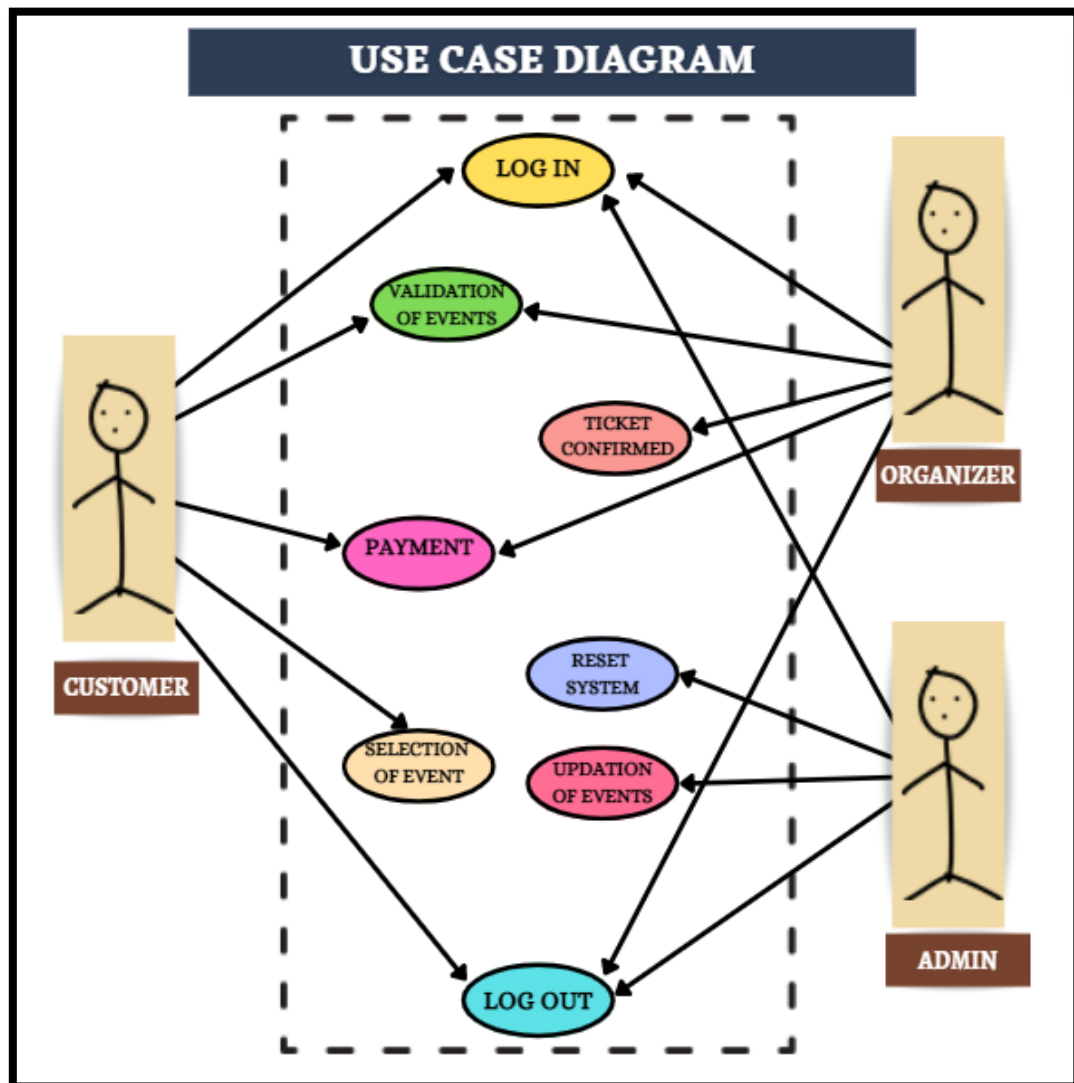


Figure 2.3

2.5.1) LOG IN:

This use case elucidates about how user and admin get signed up/registered into the system.

ACTOR-> User and Admin

Precondition: The user must have a phone number. Admin must have valid id and log in

Postcondition: If the user and admin has entered all the required details, the actors get registered into the system.

Basic Flow: This use case starts when an actor wishes to log in with the system.

- (i) System requests the actor to choose the event.
- (ii) The actor chooses the apt event.

Alternative Flow: In the basic flow, if the user does not enter the required fields then the system displays an error message.

2.5.2) BOOK EVENTS:

This use case describes how an actor can modify recharge plans

ACTOR-> Admin

Precondition: The Admin must have logged in to system.

Postcondition: The event plans are modified.

Basic Flow -> This use case starts when the actor wants to modify plans with different operator.

- (i) The admin goes to plans section.
- (ii) The user enters the feedback.
- (iii) The feedback gets added into the feedback database.

2.5.3) PAYMENT:

This use starts when the actor wants to pay.

ACTOR -> USER And ADMIN

Pre-Condition: The user and admin must be logged in.

Post-Condition: The actors receive Payment confirmation.

Basic Flow: This use case starts when a user wants to pay after selecting the recharge plan:

- (i) System requests the actor to select mode of payment
 - 1. Net Banking
 - 2. UPI
 - 3. Credit card/Debit card
- (ii) System requests the actor to enter the amount to be paid.
- (iii) System displays the admin the payment details of user.

Alternate Flow: In the basic flow, if the user fails to pay an error message get displays .

2.5.4) GENERATE REPORT:

- (i) This use case describes how the admin can generate report for the users
- (ii) ACTOR->ADMIN
- (iii) Pre-condition: The admin must be logged into the system.
- (iv) Postcondition: The reports of all the users of the system are generated
- (v) Basic Flow: This use case starts when the admin wants to generate information for all the users for future purposes.

2.5.5) SELECTION OF PLANS:

- (i) This use case describes how the user select any recharge plans of different operators
- (ii) ACTOR->USER
- (iii) Pre-condition: The user must have entered his/her mobile number into the system.
- (iv) Postcondition: The Plan gets selected.
- (v) Basic Flow: This use case starts when the user selects any plan he/she requires.

2.5.6) VALIDATE:

- (i) This use case describes how the user can generate validate message after recharge
- (ii) ACTOR->USER AND ADMIN
- (iii) Pre-condition: The reports must be generated
- (iv) Postcondition: No post condition.
- (v) Basic Flow: This use case starts when the user has already made the recharge , a message is shown and admin also receives the validate message.

2.5.7) RESET SYSTEM:

This use case describes how the administrator can reset the system

ACTOR->ADMIN

Pre-condition: The admin must log into the system.

Postcondition: The database gets deleted.

Basic Flow:

- i. This use case starts when the administrator wants to reset all the database of system
- ii. The admin has to again create all the databases in the system and enter data.

2.5.8) LOG OUT USE CASE:

This use case elucidates about how a user gets logged into the system.

ACTOR-> USER AND ADMIN

Preconditions: No Precondition

Post-conditions: The actor gets log out of the System.

Basic Flow: This use case initiates when the user wishes to log out the recommended system.

2.6) TIMELINE CHART

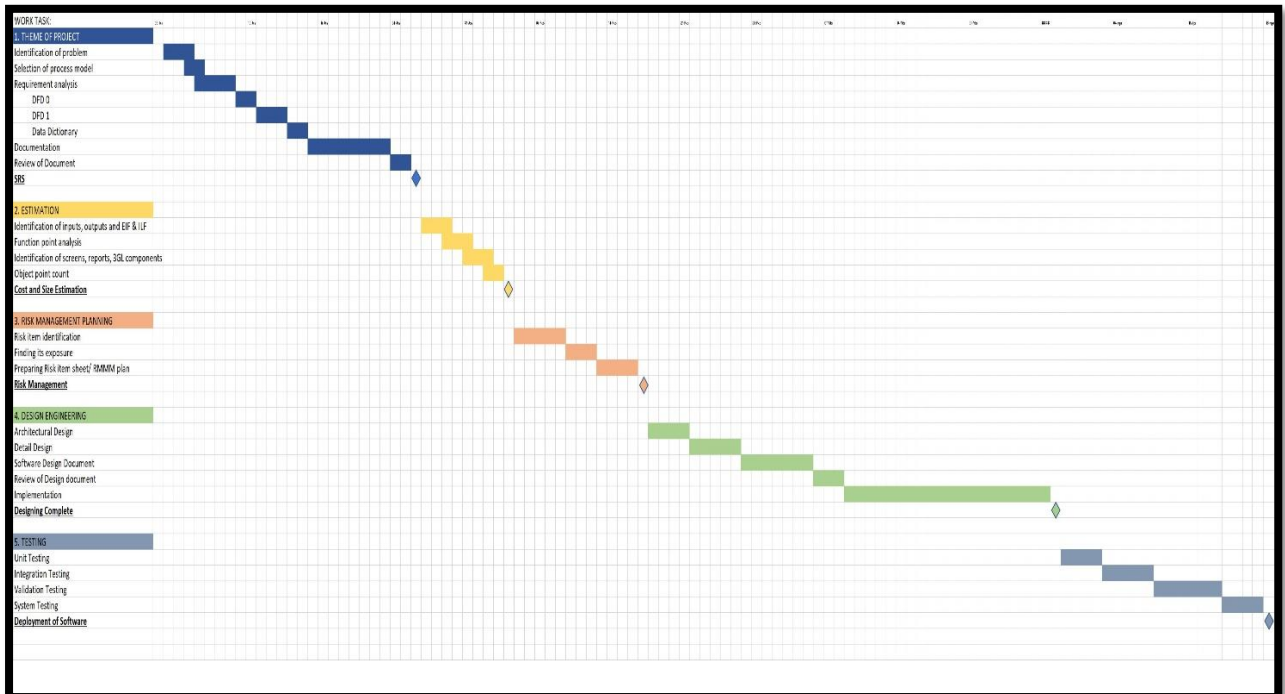


Fig 2.4

2.7) SEQUENCE DIAGRAM

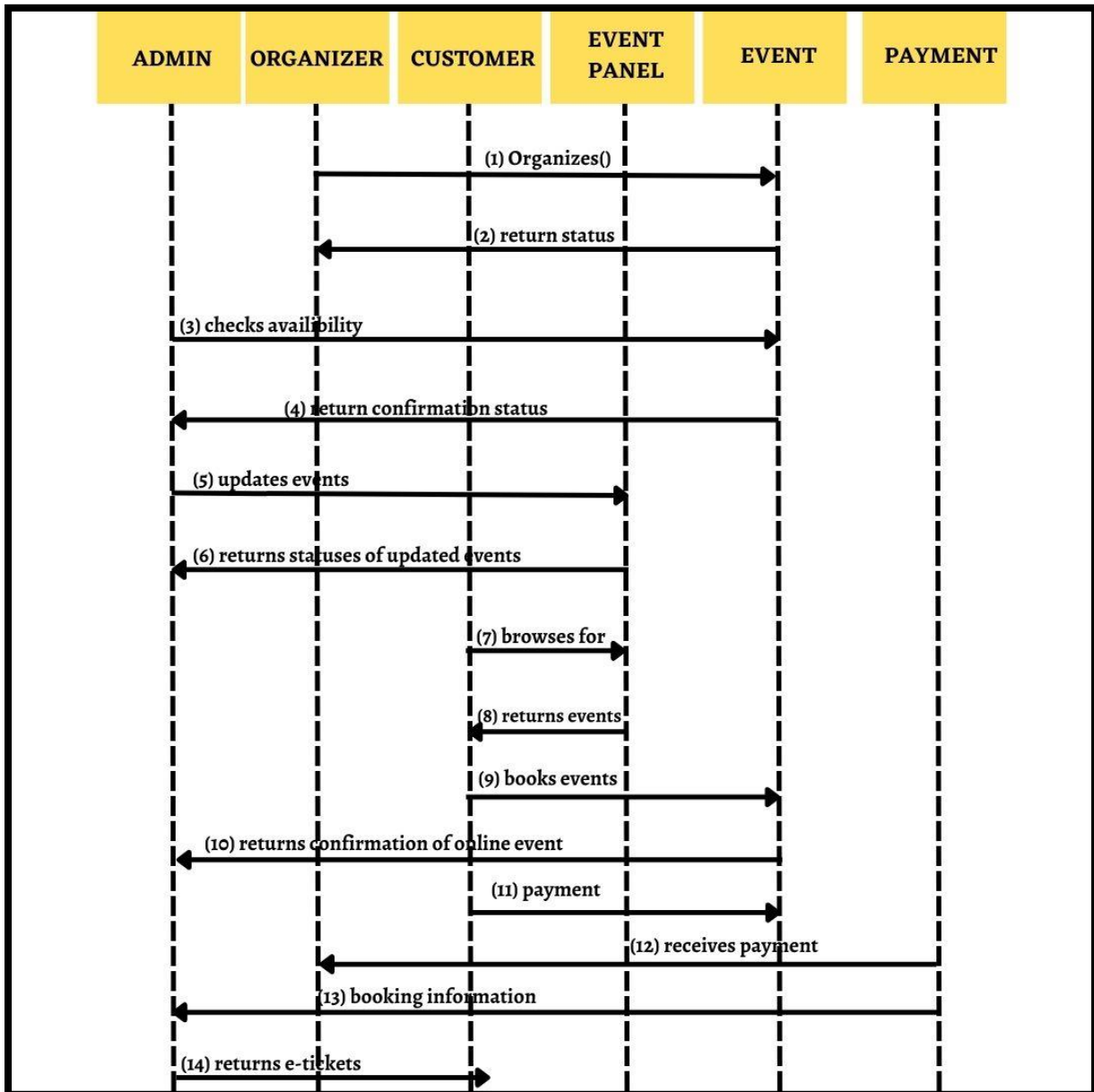


Figure 2.5

Chapter 3: Estimations

3.1) Cost Estimation

Table 3.1

S. No.	Data File	Complexity
1.	Customers' Details Database	Low
2.	Organizers' Details Database	Low
3.	Events' Details Database	Low
4.	Current Events Database	Low
5.	Venues' Details Database	Low
6.	Login	Low
7.	Register	Low
8.	Add Event	Low
9.	Forget Password	Low
10.	Edit Customers'Details	Low
11.	Edit Organizers'Details	Low

12.	Book Tickets	Low
13.	E-Ticket	High
14.	Order Details	Low
15.	Payment Details	High
16.	Event Details	Low
17.	View Events	Low
18.	View Orders	Low

Step-1:

Number of screens = 13

Number of records = 5

Step-2:

For screens,

Number of views = 21

Number of data tables = 7

Number of servers = 3

Number of clients = 7

by using information using standard table,

Complexity level for each screen = difficult

For reports,

Number of sections = 6

Number of data tables = 7

Number of servers = 2

By using information using standard table,

Complexity level for each report = difficult

Step-3:

By using complexity weight table we can assign complexity weight to each object instance

depending upon their complexity level.
Complexity weight for each screen = 3
Complexity weight for each report = 8

Step-4:

Object point count

$$= \text{sigma (Number of object instances)} * (\text{its Complexity weight})$$
$$= 13 * 3 + 5 * 8 = 79$$

Step-5:

% Reuse of object points = 30%

$$\text{NOP} = [\text{object points} * (100 - \% \text{reuse})] / 100$$
$$= [79 * (100 - 30)] / 100 = 55.3$$

Step-6:

Developer's experience and capability is nominal

Using information given about developer and productivity rate table

Productivity rate (PROD) of given project = 13

Step-7:

$$\text{Effort} = \text{NOP} / \text{PROD}$$

$$= 55.3 / 13$$

$$= 4.25 \text{ person-month}$$

Therefore, effort to develop the given project = 4.25 person-month.

3.2) Size Estimation

Table 3.2

Complexity Adjustment Values Table

S. No.	Questions	VAFs
1	Does the system require reliable backup and recovery?	4
2	Are specialized data communications required to transfer information? to or from the application?	2
3	Are there distributed processing functions?	2
4	Is performance critical?	3
5	Will the system run in an existing, heavily utilized operational environment?	2
6	Does the system require online data entry?	4
7	Does the online data enquiry require the input transaction to be built over multiple screens or operations?	2
8	Are the ILFs updated online?	4
9	Are the inputs, outputs, files or inquiries complex?	2
10	Is the internal processing complex?	4
11	Is the code designed to be reusable?	4
12	Are conversion and installation included in the design?	3
13	Is the system designed for multiple installations in different organizations?	4
14	Is the application designed to facilitate change and ease of use by the user?	4
	ΣF(i)	44

Table 3.3

Function Point Table

Information Domain Value		WEIGHING FACTOR			CALCULATION	RESULT
	COUNT	SIMPLE	AVG	COMPLEX		
External Inputs	7	3	4	6	$(7*3)+(0*4)+(0*6)$	21
External Outputs	4	4	5	7	$(2*4)+(0*5)+(2*7)$	22
External Inquiries	2	3	4	6	$(2*3)+(0*4)+(0*6)$	6
Internal Logical Files	3	7	10	15	$(3*7)+(0*10)+(0*15)$	21
External Interface Files	2	5	7	10	$(2*5)+(0*7)+(0*10)$	10
Count Total						80

$$FP = \text{count total} * [0.65 + 0.01 * \sum f_i]$$

$$FP = 80 * [0.65 + 0.01 * 44]$$

$$FP = 80 * [1.09]$$

$$FP = 87.2$$

Chapter 4: Designing

4.1) Architectural Design

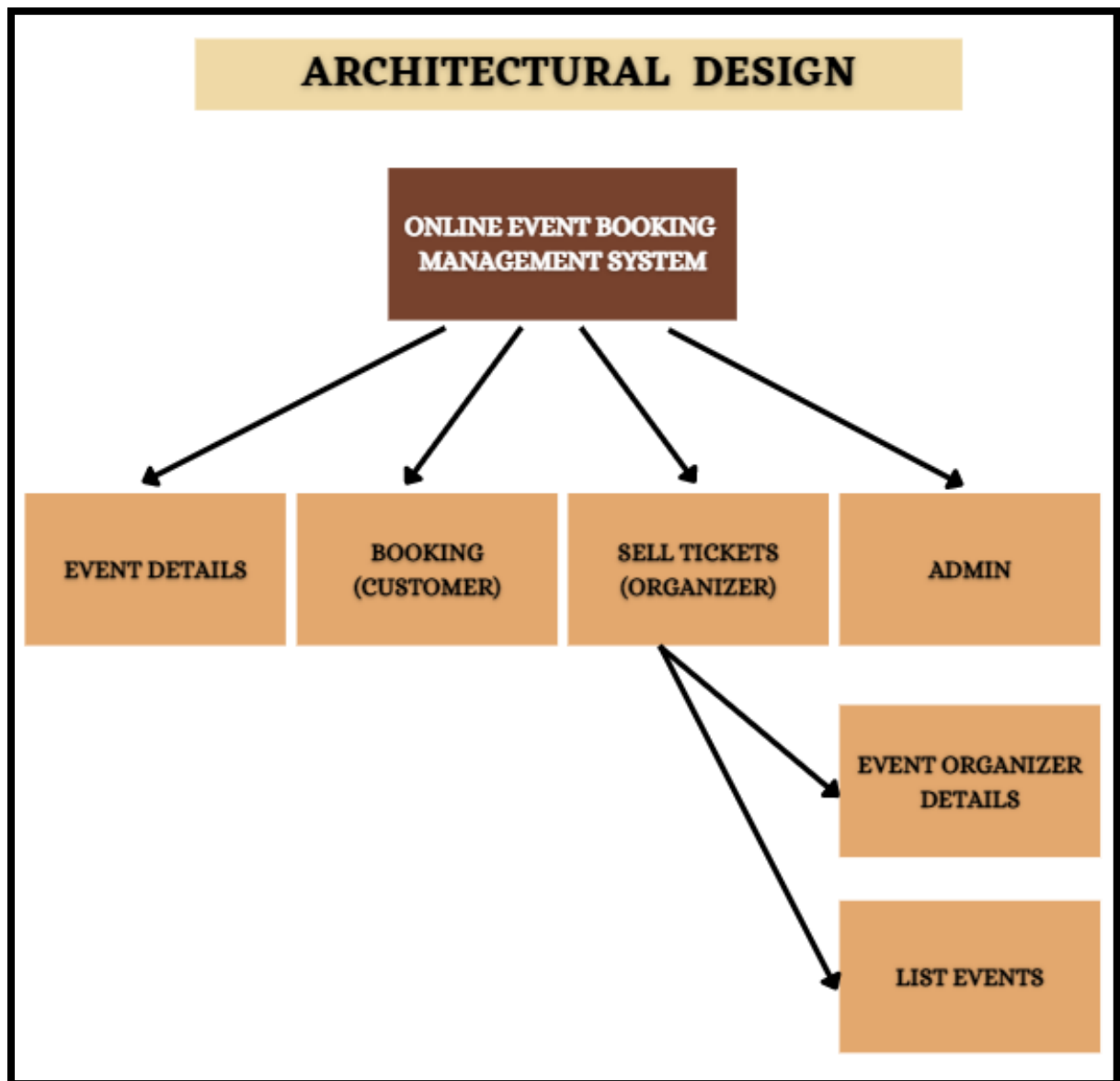


Fig 4.1

Chapter 5

Risk Management

5.1) Introduction

Risk management is the process of identifying, assessing and controlling threats to an organization's capital and earnings. These risks stem from a variety of sources including financial uncertainties, legal liabilities, technology issues, strategic management errors, accidents and natural disasters. A successful risk management program helps an organization consider the full range of risks it faces. Risk management also examines the relationship between risks and the cascading impact they could have on an organization's strategic goals.

5.2) Risk Table

Risk Identification	Category	Probability	Impact	Risk Exposure	Rank
Security Risk.	CU	60%	4	$0.60 * 1 = 0.60$	1
Requirement not properly documented and understood.	CU	40%	4	$0.40 * 1 = 0.40$	2
Technical Risk.	DE	50%	3	$0.50 * 0.75 = 0.375$	3
Occupational and health risk.	ST	60%	2	$0.60 * 0.5 = 0.300$	4
Lack of Communication.	ST	40%	3	$0.40 * .75 = 0.300$	5
Lack of Skills.	ST	35%	2	$0.35 * 0.5 = 0.175$	6
Incorrect Size and cost estimation.	PS	30%	2	$0.30 * 0.5 = 0.150$	7

Table 5.1

Impact of Risk

4:- Catastrophic

3:- Critical

2:- Marginal

1:- Negligible

Category

PD :- Process Definition

ST :- Staff-Size And Experience

DE:- Development Environment

CU :- Customer Characteristic

BU :- Business Impact

5.3) Risk Information Sheet

Table 5.2

Security Risk

Risk Information Sheet			
ID: 1	Probability:60%	Impact:4	Exposure:0.60
Description			
This type of risk can lead to system crash, leaking of customer personal and important information which can cause them mental and financial problem.			
Mitigation			
The cost associated with a System Crash resulting in a loss of data is crucial a System crash itself is not crucial, but rather the loss of data or Important/Personal Information Of customer is at Risk. A loss of data will result in not being able to deliver the3product to the customer.			
The organization can take Backup of Data on Cloud or on secure place. Firewall protection, virus protection can be setup on organization site and on customer system.			
Monitoring			
When working on the product or documentation, the staff member should always be aware of the stability of the computing environment they're working in. Any changes in the stability of the environment should be recognized and taken seriously.			
Management			
The lack of a stable-computing environment is extremely hazardous to a software development team. In the event that the computing environment is found unstable, the development team should cease work on that system until the environment is made stable again, or should move to a system that is stable and continue working there.			

Requirement Not Properly understand

Risk Information Sheet			
ID: 2	Probability: 40%	Impact:4	Exposure:0.40
Description			
When the requirement not properly under documented and understood, it can lead to the wrong product and size estimation. Which can lead to customer dissatisfaction and wrong company image in the market.			
Mitigation			
In order to prevent this from happening, members who are in charge of developing the documentation will keep in contact with each developer on the team. Meetings will be held routinely to offer documentation suggestions and topics. Any topic deemed missing by a particular developer will be discussed and			

it will be decided whether or not to add that particular topic to the documentation. In addition, beta testers will be questioned about their opinion of the documentation.

Monitoring

Throughout development or normal in and out of house testing, the development team and or beta testers will need to keep their eyes open for any possible documentation topics that have not been included.

Management

Should this occur, the organization would call a meeting and discuss the addition of new topics, or removal of unnecessary topics into the documentation

Technical Risk

Risk Information Sheet			
ID: 3	Probability:50%	Impact: 3	Exposure: 0.375
Description: This risk may cause due to interface related problem, Design related problem and maintenance problem.			
Mitigation			
In order to prevent this from happening, meetings (formal and informal) will be held with the customer on a routine business. This ensure that the product we are producing, and the specifications of the customer are equivalent.			
Monitoring			
The meetings with the customer should ensure that the customer and our organization understand each other and the requirements for the product.			
Management			
Should the development team come to the realization that their idea of the product specifications differ from those of the customer, the customer should be immediately notified and whatever steps necessary to rectify this problem should be done. Preferably a meeting should be held between the development team and the customer to discuss at length this issue.			

Health risk

Risk Information Sheet			
ID: 4	Probability:60%	Impact:2	Exposure:0.300
Description			
Health related risk can cause reduction in staff size which can cause delay in project delivery. It also impacts the cost of the project.			

Mitigation

The team members can check and monitor each other works at each step of project milestones. That are setup.

We can hire some members to learn the works with the particular member who is currently doing some work on an important component of the project.

Monitoring

We can setup regular meetings with the team member to take regular project report of the work they are doing currently.

Management

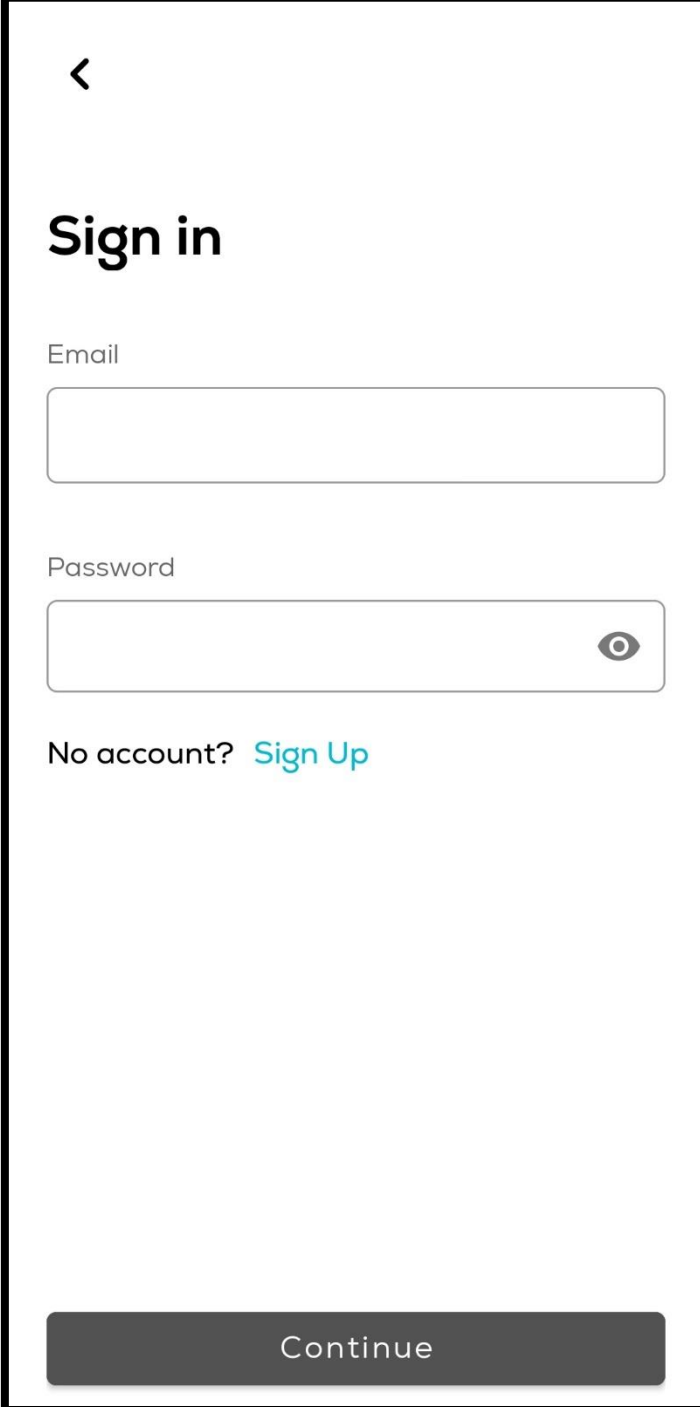
We can buy them health insurance for the team members.

Check the workload management of the members.

Organize health care camp to reduce mental fatigue and medical health related issues.

Chapter 6 INTERFACES

6.1) SIGN IN PAGE



A mobile app sign-in screen. At the top left is a back arrow. Below it is the title "Sign in". There are two input fields: "Email" and "Password". The "Password" field has an eye icon on the right. Below the fields is the text "No account? Sign Up" where "Sign Up" is a teal link. At the bottom is a dark grey button labeled "Continue".

<

Sign in

Email

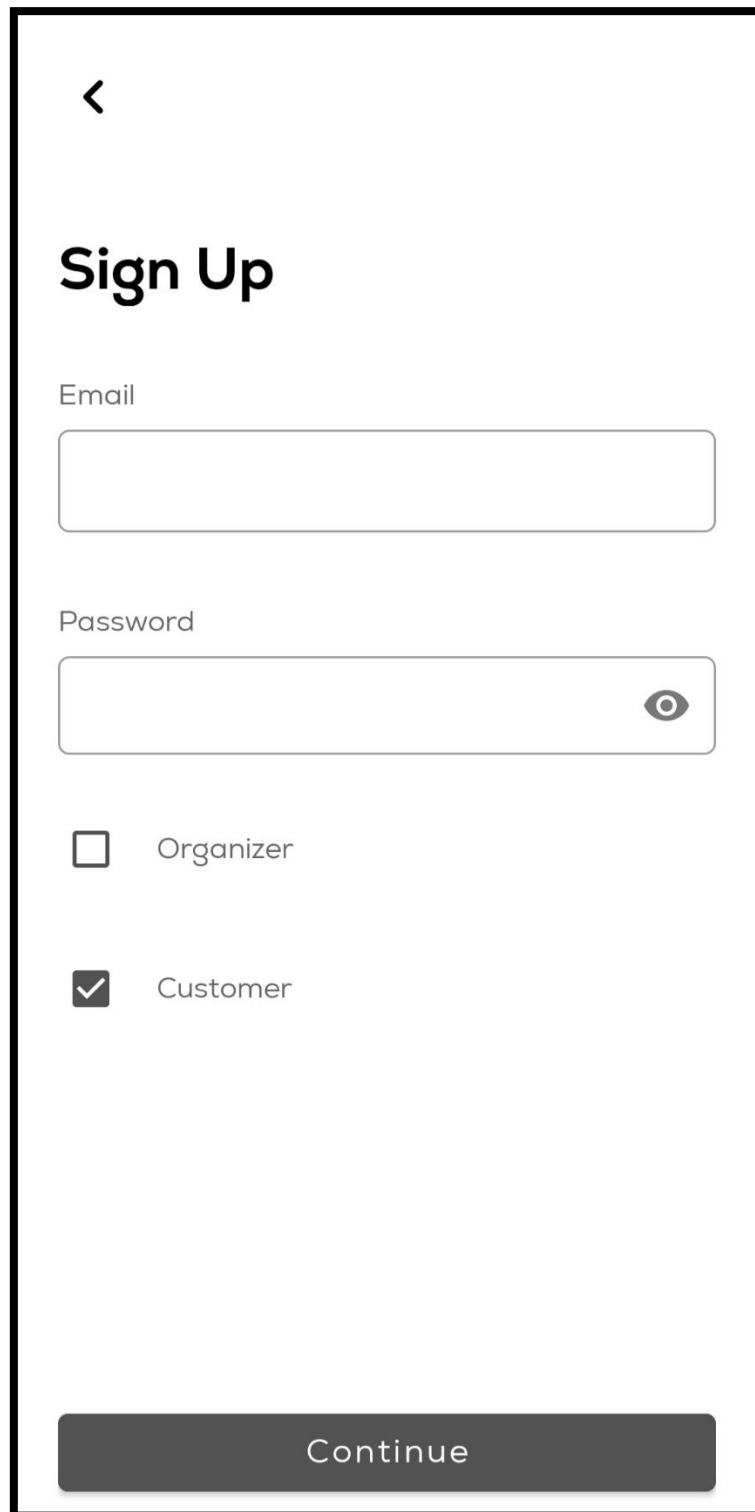
Password

No account? [Sign Up](#)

Continue

Fig 6.1

6.2) SIGN UP PAGE




A mobile app sign-up page mockup. At the top left is a back arrow icon. Below it is the title "Sign Up" in a large, bold font. There are two input fields: "Email" and "Password". The "Password" field has an eye icon on the right side. Below the input fields are two radio button options: "Organizer" (unchecked) and "Customer" (checked). At the bottom is a dark grey button labeled "Continue".

<

Sign Up

Email

Password



☐ Organizer


☒ Customer

Continue

Fig 6.2


6.3) SIGN IN USING- PAGE


SKIP




Enjoy faster show booking through our recommendations tailored for you.

● ● ●

Continue with Google

Continue with Email

OR

+91

Continue with mobile number

I agree to the [Terms | Conditions](#) and [Privacy Policy](#)

Fig 6.3

6.4) BOOK EVENTS PAGE

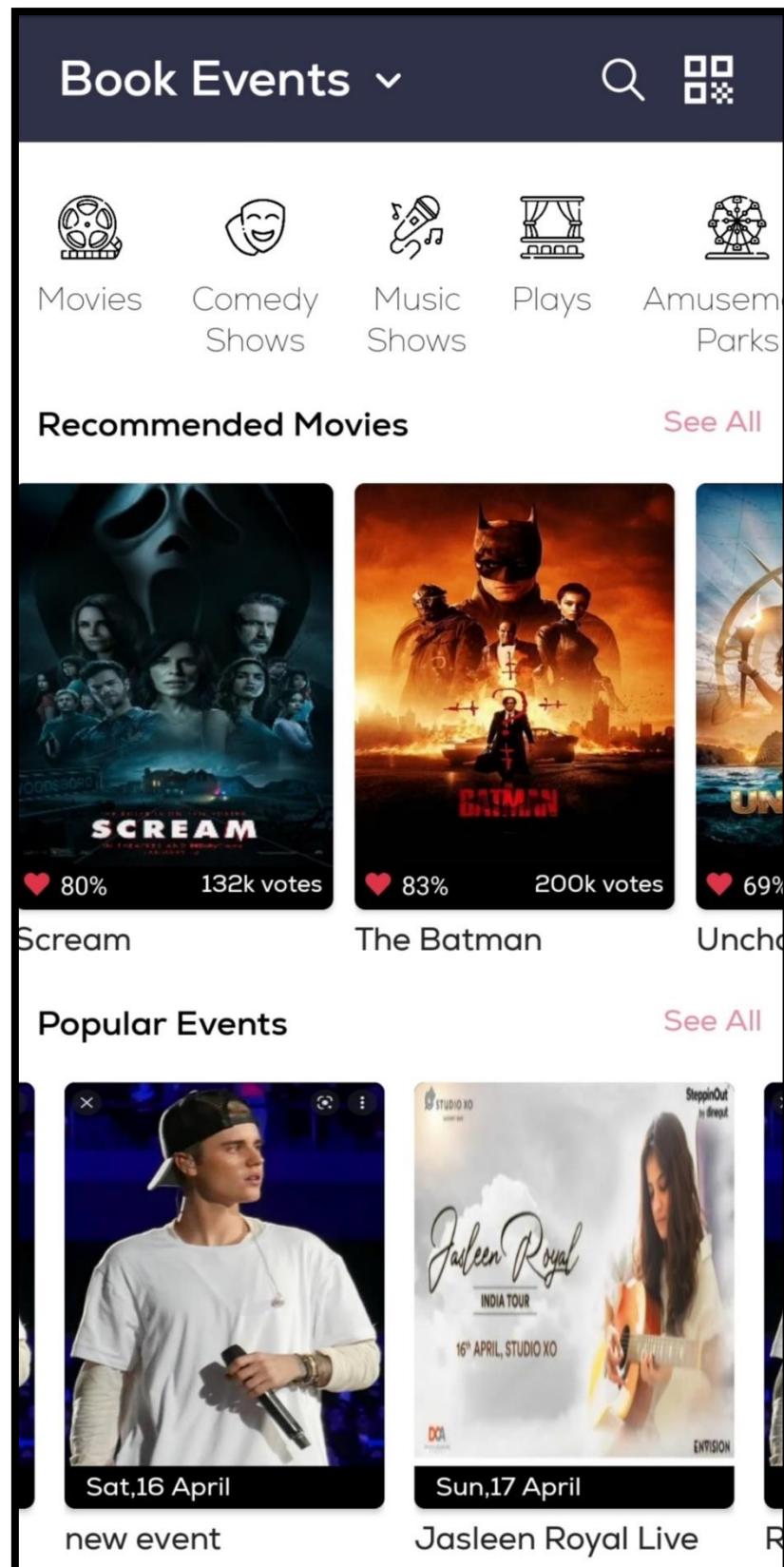


Fig 6.4

6.5) EVENT INFORMATION PAGE



Jasleen Royal Liv...

MUSIC SHOWS

Jasleen Royal Live by SteppinOut Delhi NCR

Sun 17 April 2022

08 : 00 PM

Studio XO Bar

1st Floor, Trillum Avenue, near Hudacity Metro station, Sector 29, Gurugram, Haryana

[View on map](#)

More information

Event information may be out of date due to the coronavirus (COVID-19). Confirm details with event organisers.

SteppinOut by Dineout presents Jasleen Royal India Tour. Jasleen Royal made her mark through her signature melodies and a very soulful and distinct voice.

She can collectively play the guitar, mouth organ, jute, and tambourine while singing. In a short span, the gorgeous and extremely talented Singer-Composer Jasleen Royal became a household name, delivering other hits including Kho Gaye Hum Kahan, Din Shagna Da, Love You Zindagi, the recent chartbuster Ranjha and many more.

She has proved herself scoring for Hindi films as well as her independent outings like Nit Nit and Sang Rahiya.

Opening act: DJ at 8 PM

Jasleen Royal Live: 10:30 PM

event curators in the country. With more than 20

For more details follow us on our social media

About SteppinOut by Dineout SteppinOut is one of the largest experiential events hosted in a month, ranging from night markets, live music shows, concerts, comedy nights, and much more! In August 2020, SteppinOut was acquired by India's largest dining out platform - Dineout. The acquisition was a step forward in creating memorable experiences while expanding Dineout's overall value proposition to consumers across India. This enabled in creating a first-of-its-kind experience for our users with Dineout hosting thousands of restaurant partners, and SteppinOut hosting new and innovatively curated events. Rooted in diversity all while maintaining quality, all SteppinOut's experiences are crafted with intimate attention, impressive detail, and an immense amount of love.

handles:

Instagram: @steppinout.in

Facebook: @steppinout.in

LinkedIn: SteppinOut by Dineout Youtube: SteppinOut by Dineout

Twitter: @SteppinOutIndia

Tickets

Available 476

476

Required 1

2

₹ 200.0 onwards

[Book](#)

Filling fast

Fig 6.5

6.6) PAYMENT PAGE

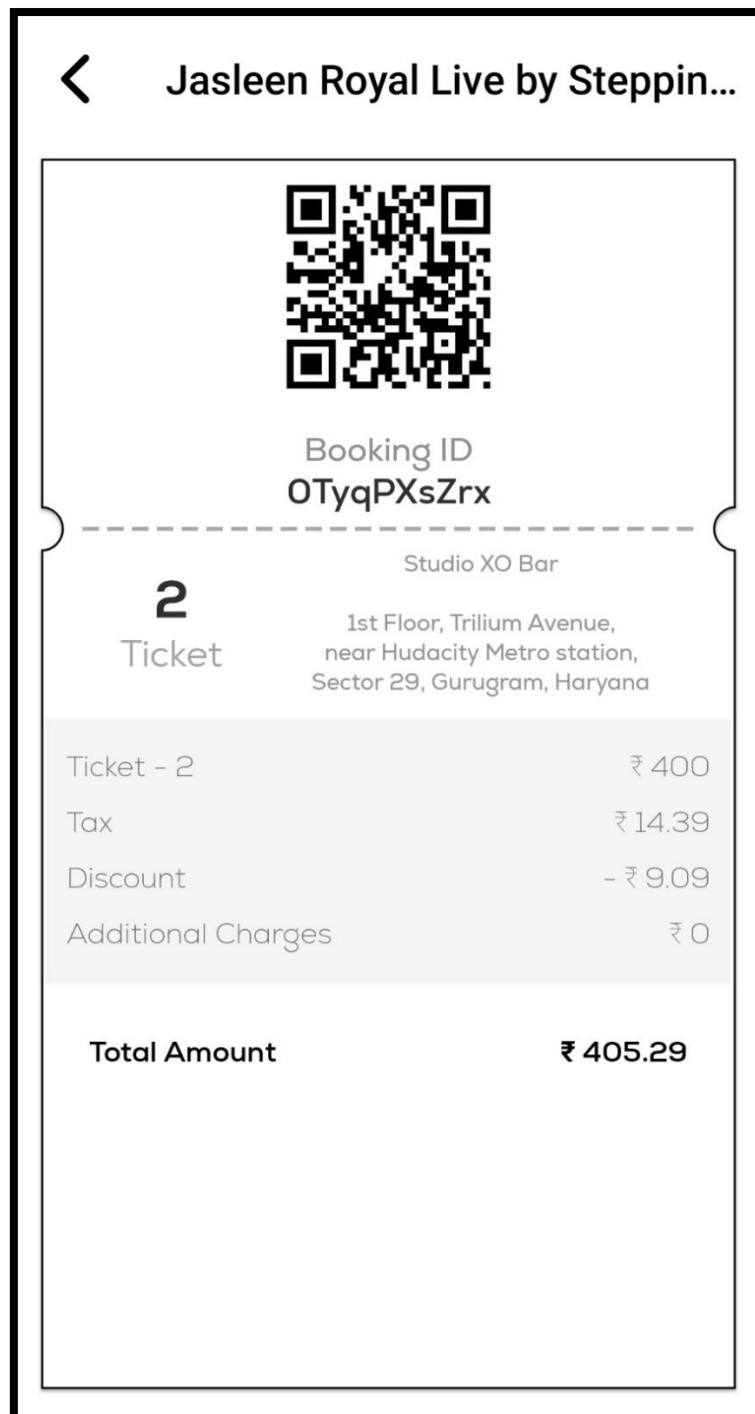
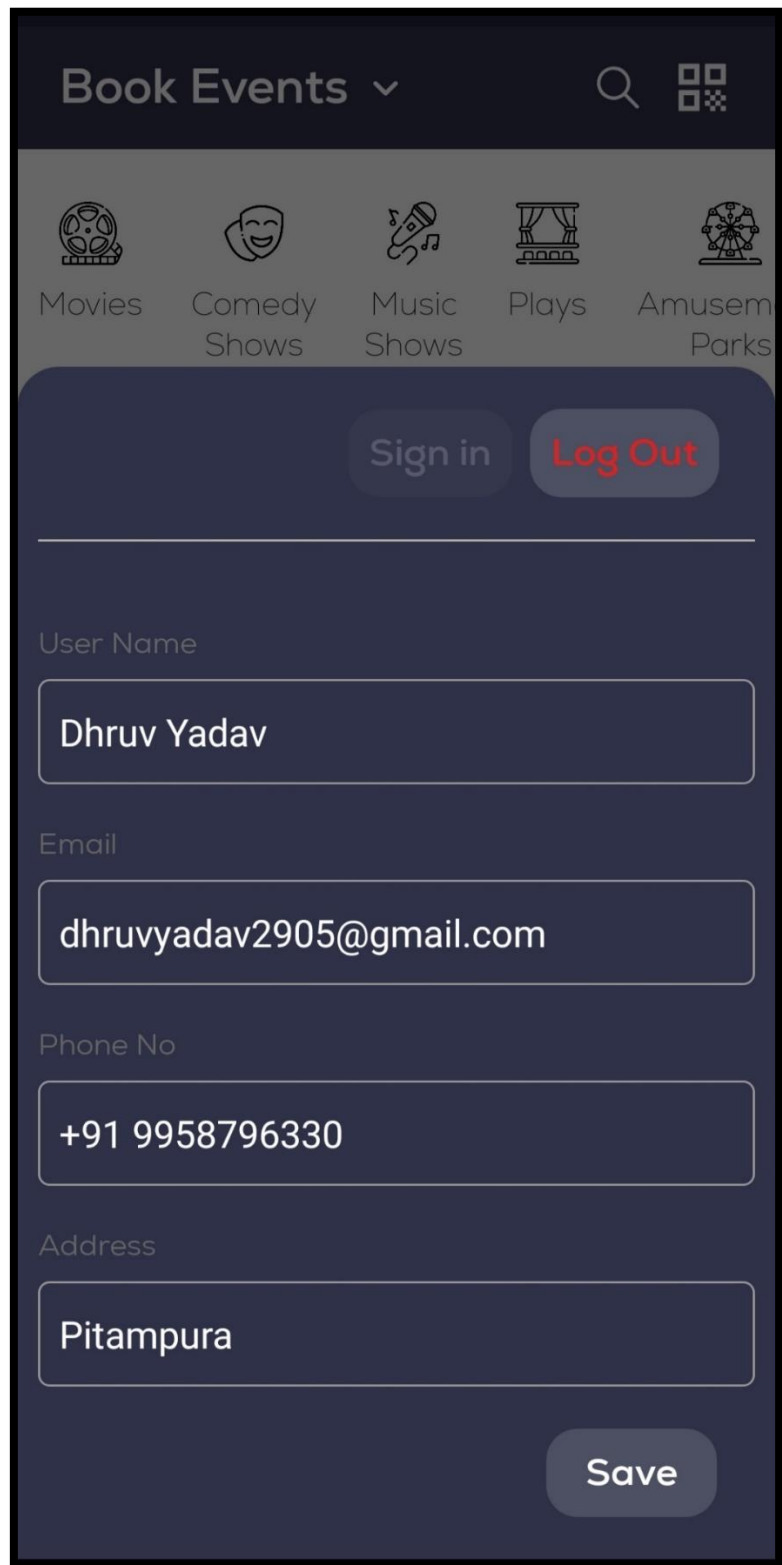


Fig 6.6






6.6) LOGOUT INTERFACE



The image shows a mobile application interface for booking events. At the top, there is a header with the text "Book Events" and a dropdown arrow, a search icon, and a grid icon. Below the header is a navigation bar with five categories: Movies, Comedy Shows, Music Shows, Plays, and Amusement Parks, each with a corresponding icon. Below the navigation bar is a dark blue section containing two buttons: "Sign in" and "Log Out". Below this section is a form with four input fields: "User Name" (containing "Dhruv Yadav"), "Email" (containing "dhruvyadav2905@gmail.com"), "Phone No" (containing "+91 9958796330"), and "Address" (containing "Pitampura"). At the bottom right of the form is a "Save" button.

Book Events ▾

🔍 🗪

 Movies  Comedy Shows  Music Shows  Plays  Amusement Parks

Sign in Log Out

User Name

Dhruv Yadav

Email

dhruvyadav2905@gmail.com

Phone No

+91 9958796330

Address

Pitampura

Save

Fig 6.7

Chapter 7

TESTING

7.1) BASIS PATH TESTING

7.1.1) Code:-

```
1. public class BillActivity extends AppCompatActivity {
2.     private ActivityBillBinding binding;
3.     private EventModel;
4.     private BookingModel;
5.     private TextView totalCostWithoutTax_DiscountTxt, discountTxt, taxTxt,
       totalAmountTxt;

6.     @Override
7.     protected void onCreate(Bundle savedInstanceState) {
8.         super.onCreate(savedInstanceState);
9.         binding = DataBindingUtil.setContentView(this, R.layout.activity_bill);
10.        init();
11.        set();
12.    }

13.    private void set() {
14.        String pattern = "##,##,###.##";
15.        DecimalFormat = new DecimalFormat(pattern);
16.        String str = "\u20B9 " +
            decimalFormat.format(Double.parseDouble(eventModel.getPrice()) *
            Integer.parseInt(bookingModel.getNoTickets()));
17.        totalCostWithoutTax_DiscountTxt.setText(str);
18.        String discountStr = "- \u20B9 ";
19.        String taxStr = "\u20B9 ";
20.        String totalAmountStr = "\u20B9 " +
            decimalFormat.format(Double.parseDouble(bookingModel.getAmountPaid()));

21.        if (bookingModel.getTax() != null)
22.            taxStr += decimalFormat.format(Double.parseDouble(bookingModel.getTax()));
23.        else
24.            taxStr += "0";
```

```
25. if (bookingModel.getDiscount() != null)
26. discountStr +=
    decimalFormat.format(Double.parseDouble(bookingModel.getDiscount()));
27. else
28. discountStr += "0";

29. discountTxt.setText(discountStr);
30. taxTxt.setText(taxStr);

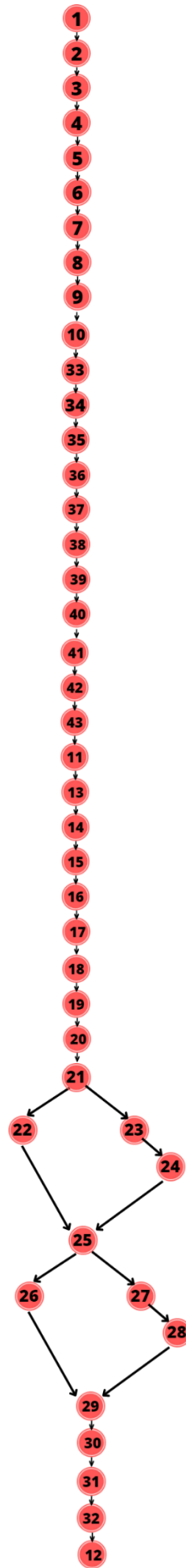
31. totalAmountTxt.setText(totalAmountStr);
32. }

33. private void init() {
34. eventModel = (EventModel) getIntent().getSerializableExtra("event");
35. bookingModel = (BookingModel) getIntent().getSerializableExtra("booking");
36. bind(BR.eventModel, eventModel);
37. bind(BR.bookingModel, bookingModel);
38. setSupportActionBar(binding.toolBar);
39. totalCostWithoutTax_DiscountTxt = binding.totalCostAllTickets;
40. discountTxt = binding.discountTxt;
41. taxTxt = binding.taxTxt;
42. totalAmountTxt = binding.totalAmountTxt;

43. }
```

7.1.2) Control Flow Diagram

Fig 7.1



7.1.3) D-D Path Graph

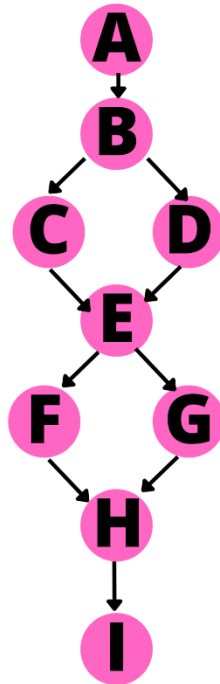


Fig 7.2

7.1.4) Cyclomatic Complexity:

$$\begin{aligned} V(G) &= e - n + 2P \\ &= 44 - 43 + 2 \\ &= 1 + 2 \\ &= 3 \end{aligned}$$

Or

$$\begin{aligned} V(G) &= \text{number of decision nodes} + 1 \\ &= 2 + 1 \\ &= 3 \end{aligned}$$

7.1.5) Independent Paths:

1. A -> B -> C -> E -> F -> H -> I
2. A -> B -> D -> E -> G -> H -> I
3. A -> B -> C -> E -> G -> H -> I
4. A -> B -> D -> E -> F -> H -> I

7.2) BOUNDARY VALUE ANALYSIS TESTING

$n=1$ (password)

Test Case : $4n+1 = 5$

$n[10^7 - 10^{15}]$

Boundary values : 10000000, 10000001, $10^{15} - 1$, 10^{15}

INPUT	EXPECTED OUTPUT	OBSERVED OUTPUT	MATCH
10000000	Valid password	Valid password	Yes
10000001	Valid password	Valid password	Yes
10^{11}	Valid password	Valid password	Yes
$10^{15} - 1$	Valid password	Valid password	Yes
10^{15}	Valid password	Valid password	Yes

Table 7.1

7.3) EQUIVALENCE CLASS TESTING

$n[10^7 - 10^{15}]$

Input Equivalence Class

$I_1 = \{ \text{" Password must be 8 characters long "}, 0 < n \}$

$I_2 = \{ \text{" Valid Password"}, 10^7 < n < 10^{15} \}$

$I_3 = \{ \text{"Invalid Password"}, n > 10^{15} \}$

Output Equivalence Class

$O1 = \{ \langle n \rangle : \text{" Password must be 8 characters long " if } 0 < n < 10^7 \}$

$O2 = \{ \langle n \rangle : \text{"Invalid Password" if } n > 10^{15} \}$

TEST CASE	N	EXPECTED OUTPUT
1	0	Password must be 8 characters long
2	12345678	Valid password
3	2345678100000000000	Invalid password

Table 7.2

CONCLUSION

The aim of our project was to make the cumbersome process of event management easy for customers and completely online. What we did for the same was ideation, then requirement analysis, implementation and then we got an outcome finally.

The ideation of our project involved developing a software for the process of booking events and movies which should lead to increased efficiency and reduced drawbacks which were present in the previous procedure of online ticket booking and makes it convenient for the customer to book a particular ticket. The software should control redundancy so that no two customers can access the same seat at the same time and transactions should be independent. Moreover, ticket booking has been a hassle for the customer, they are left with just an option i.e. book the tickets by visiting the venue.

Talking about the process models, our project followed the waterfall model. We started with requirement gathering and analysis and then designed our system. The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture. With inputs from the system design, the system was first developed in small programs called units, which were integrated in the next phase. Each unit was developed and tested for its functionality, referred to as Unit Testing.

Next step that came was Integration and Testing – All the units developed in the implementation phase were integrated into a system after testing of each unit. Post integration the entire system was tested for any faults and failures. We performed three testing techniques which included Basis Path Testing, Boundary Value Analysis and Equivalence Path Testing. Basis path testing is a technique of selecting the paths in the control flow graph, that provide a basis set of execution paths through the program or module. Boundary value analysis is one of the widely used case design technique for black box testing. It is used to test boundary values because the input values near the boundary have higher chances of error. Whenever we do the testing by boundary value analysis, the tester focuses on, while entering boundary value whether the software is producing correct output or not.

Equivalence Partitioning or Equivalence Class Partitioning is type of black box testing technique which can be applied to all levels of software testing like unit, integration, system, etc. In this technique, input data units are divided into equivalent partitions that can be used to derive test cases which reduces time required for testing because of small number of test cases. All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model".

After following all the steps of this model, and preparing various diagrams and analysis of data, we prepared our eventive App, an online event booking management system app, that provides you all the functionalities of booking tickets for various events. It shows the number of available tickets and updates it every time a user books it. Everything has been done completely online and in a user friendly manner helping audience to book tickets of their favourite movies, concert shows and other events from the comfort of their home.

REFERENCES:-

- 1) Software Engineering: A practitioner's approach by
Roger S. Pressman
Bruce R. Maxim
- 2) <https://www.javatpoint.com/software-engineering-tutorial>
- 3) https://www.tutorialspoint.com/software_engineering/index.htm
- 4) <https://economictimes.indiatimes.com/definition/software-engineering>