

## **EXPERIMENT NO: 09**

**Aim:** Identify the design principle that is being violated in relation to the given scenario. (Give any Scenario)

### **Theory:**

- **Design Principles**

Software design is both a process and a model. The design process is a sequence of steps that enable the designer to describe all aspects of the software to be built. It is important to note, however, that the design process is not simply a cookbook. Creative skill, past experience, a sense of what makes “good” software, and an overall commitment to quality are critical success factors for a competent design.

**There are 10 Design Principles: -**

**1. The design process should not suffer from “tunnel vision.” :-**

A good designer should consider alternative approaches, judging each based on the requirements of the problem, the resources available to do the job.

**2. The design should be traceable to the analysis model: -**

Because a single element of the design model often traces to multiple requirements, it is necessary to have a means for tracking how requirements have been satisfied by the design model.

**3. The design should not reinvent the wheel: -**

Systems are constructed using a set of design patterns, many of which have likely been encountered before. These patterns should always be chosen as an alternative to reinvention. Time is short and resources are limited! Design time should be invested in representing truly new ideas and integrating those patterns that already exist.

**4. The design should “minimize the intellectual distance” between the software and the problem as it exists in the real world: -**

That is, the structure of the software design should (whenever possible) mimic the structure of the problem domain.

**5. The design should exhibit uniformity and integration: -**

A design is uniform if it appears that one person developed the entire thing. Rules of style and format should be defined for a design team before design work begins. A design is integrated if care is taken in defining interfaces between design components.

**6. The design should be structured to accommodate change: -**

The design concepts discussed in the next section enable a design to achieve this principle.

**7. The design should be structured to degrade gently, even when aberrant data, events, or operating conditions are encountered: -**

Well-designed software should never “bomb.” It should be designed to accommodate unusual circumstances, and if it must terminate processing, do so in a graceful manner.

**8. Design is not coding, coding is not design: -**

Even when detailed procedural designs are created for program components, the level of abstraction of the design model is higher than source code. The only design decisions made at the coding level address the small implementation details that enable the procedural design to be coded.

**9. The design should be assessed for quality as it is being created not after the fact: -**

A variety of design concepts and design measures are available to assist the designer in assessing quality.

**10. The design should be reviewed to minimize conceptual (semantic) errors: -**

There is sometimes a tendency to focus on minutiae when the design is reviewed, missing the forest for the trees. A design team should

ensure that major conceptual elements of the design (omissions, ambiguity, inconsistency) have been addressed before worrying about the syntax of the design model.

When these design principles are properly applied, the software engineer creates a design that exhibits both external and internal quality factors. External quality factors are those properties of the software that can be readily observed by users (e.g., speed, reliability, correctness, usability). Internal quality factors are of importance to software engineers. They lead to a high-quality design from the technical perspective. To achieve internal quality factors, the designer must understand basic design concepts.

- **Scenario: Home security system with camera and sensors**

- **Questions:**

**1. Which Principles are being invalid with the given scenario.**

In the given scenario, 'Home Security system with camera and sensors' the principles being violated are:

1. The Design should be structured to accommodate change.
2. The design should be structured to degrade gently, even when aberrant data, events, or operating conditions are encountered.

**2. How it is violated with the given scenario.**

The principle – 'The Design should be structured to accommodate change' is violated in the scenario. As the scenario states – 'Home security system with camera and sensors', it has been specified that the security system must have camera and sensors only nothing less, nothing more. This scenario is too specific and thus it negates the ability to accommodate change, thus violating the principle.

The principle – 'The design should be structured to degrade gently, even when aberrant data, events, or operating conditions are encountered' is violated in the Scenario, as the scenario of 'home security system with camera and sensors' uses hardware equipment such as camera and sensors. These equipments can wear out and as a result the system might not work as intended. These equipments might get affected by the climate, voltage fluctuations, etc. Due to this the system might get corrupted and even has a risk of failing. But the principle states that 'the software must not be a bomb' that is it should be designed to accommodate unusual circumstances and must terminate the processing in refined manner if a problem arises, which is not applicable for the provided scenario. The failures in camera and sensors would not be able to cope to these unusual circumstances and may act like a bomb and just get corrupted or stop working, thus violating the principle.

### 3. Which Principles are valid with the given scenario.

In the given Scenario – ‘Home security system with camera and systems’ the principles which are valid to the scenario are:

1. The design should be traceable to the analysis model.
2. The design should exhibit uniformity and integration.
3. The design process should not suffer from “tunnel vision”.
4. Design is not coding, and coding is not design.

The principle – ‘The design should be traceable to the analysis model’ is valid as the design statement is concrete and clear which makes tracking the other required elements easier.

The principle - ‘The design should exhibit uniformity and integration.’ Is valid as the design statement clearly states that it should be a security system for the home consisting of camera and sensors, thus it makes sure that the design stays uniform.

The principle – ‘The design process should not suffer from “Tunnel vision”.’ Here the approach has been confirmed which can be done by judging the requirements of the problem, and the resources available to do the job.

The principle – ‘Design is not coding, and coding is not design’, is valid in this scenario as the design clearly states only the design components of the system and not the code for the components.

- **Conclusion:**

Hence, by performing this practical I learnt about the about the concept of Design Principles, the 10 Design Principles and also got to understand how to apply them in a scenario by checking if they are valid or invalid with the principles or not. I also identified the principles which were being violated in relation to the given scenario – ‘Home security system with camera and sensors.’

(10)	(20)	(10)	(10)	TOTAL

## **EXPERIMENT NO: 10**

**Aim:** Using all the concepts of software engineering make a mini-project on e-commerce store – Book Store.

**Theory:**

- **Introduction:**

The Book store application is an application software made to help students, teachers, and a majority of book readers for buying books and magazines and get them delivered to you through our android app. This application's major goal is to provide students, teachers and book readers books at their homes with zero-contact delivery due to the Covid-19 Pandemic situation.

- **Software Development Processes:**

1. **Communication:**

The project was given by Prof. L L Bhadekar sir to the team.

The topic of the project was - 'Book Store app'.

The basic features for the app were to have option to buy books using credit card, get books suggestion and to review the books by giving a rating (1-5 stars).

2. **Planning:**

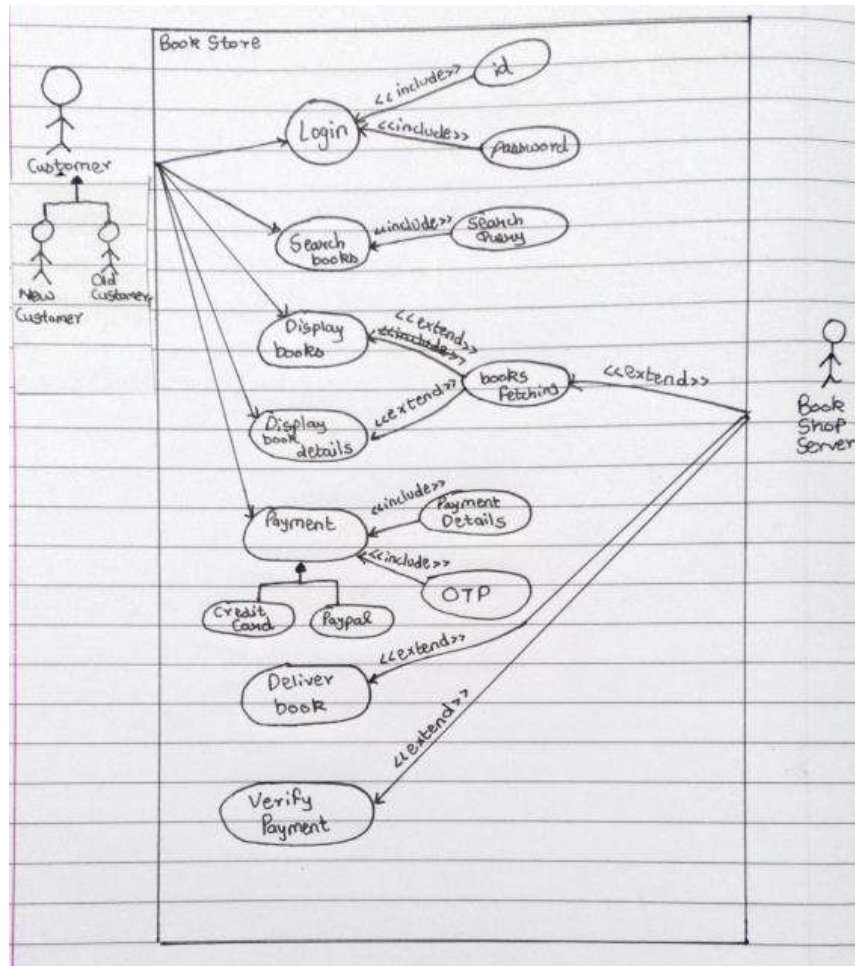
We discussed the plans to be made for the features of the app.

The software and tools to be used in the process of development of the software are decided and a plan is made for the software.

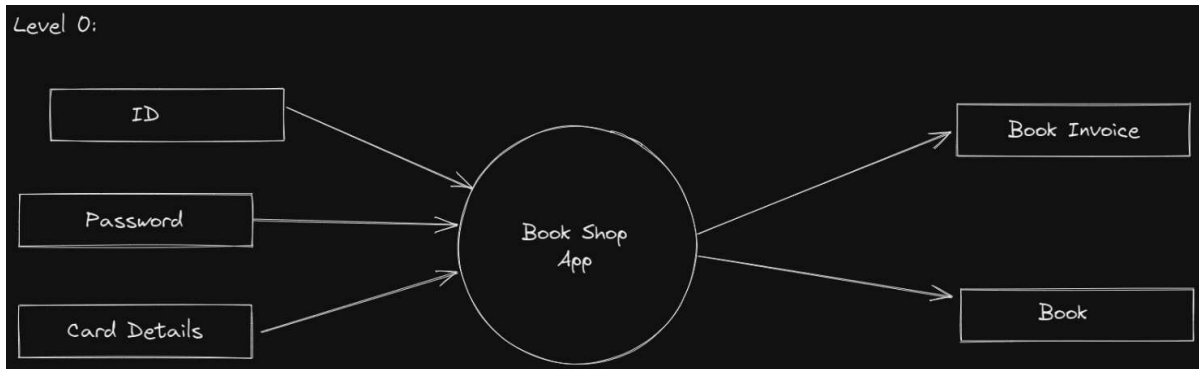
The working for each member is decided and distributed.

### 3. Modelling:

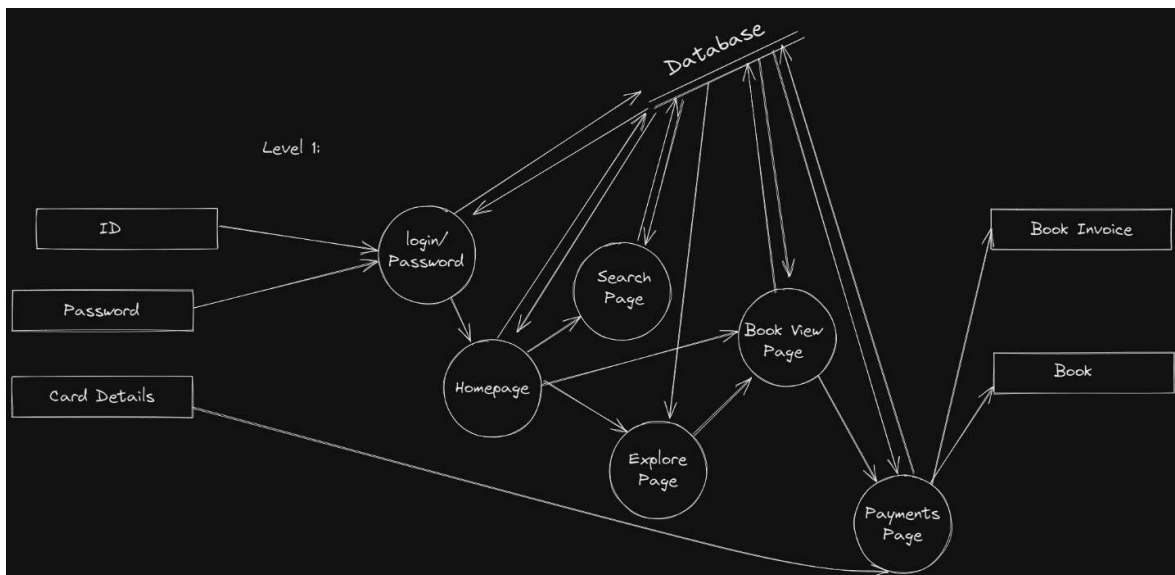
#### Use Case Diagram:



## Data Flow Diagram:

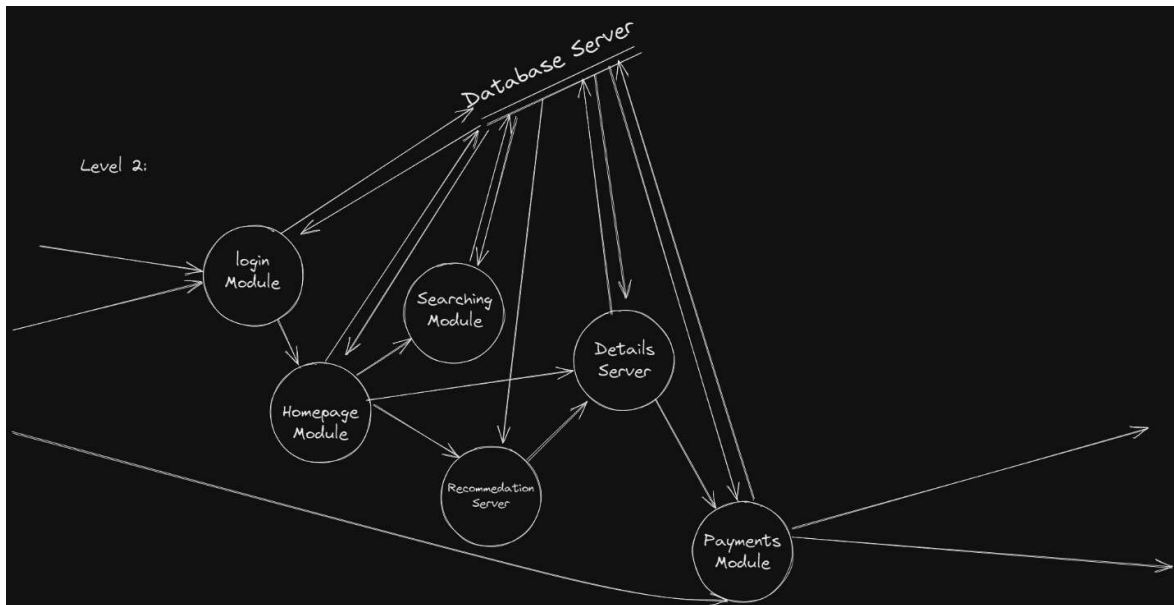


Level 0



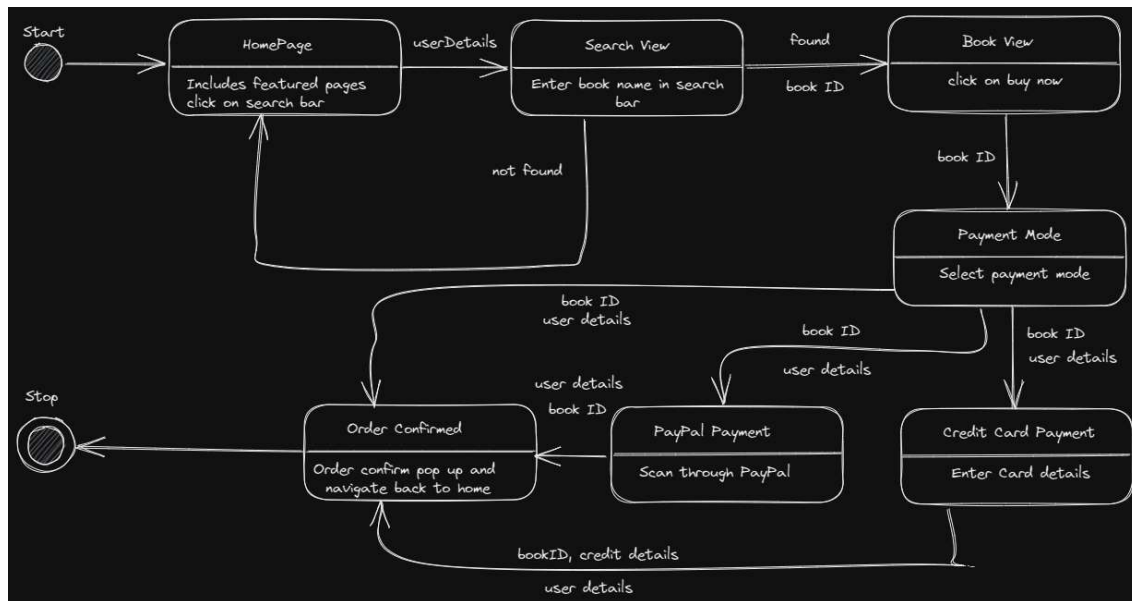
Level 1





Level 2

### State transition Diagram:



### Critical Path Diagram:

Task	Prec. Task	Description	Time (hrs).
A	none	Communication	3
B	A	Decision for the topic	1
C	A	choosing the platform	1
D	B	making UI on figma	6
E	D	Start distributing the models to team members.	3
F	C, D	start coding	9, 9
G	E	merging the coding through github.	3
H	E	Documentation	1

Task	Preced. Task	Description	Time (hrs).
A	none	Communication	3
B	A	Decision for the topic	1
C	A	Choosing the platform	1
D	B	making UI on figma	6
E	D	Start distributing the models to team members.	3
E	C, D	start coding	9,9
G	E	merging the coding through github.	3
H	E	Documentation	1

#### 4. Construction:

For making the application, dart programming language and Flutter framework was used.

We used android studio and Visual Studio Code for coding the project.

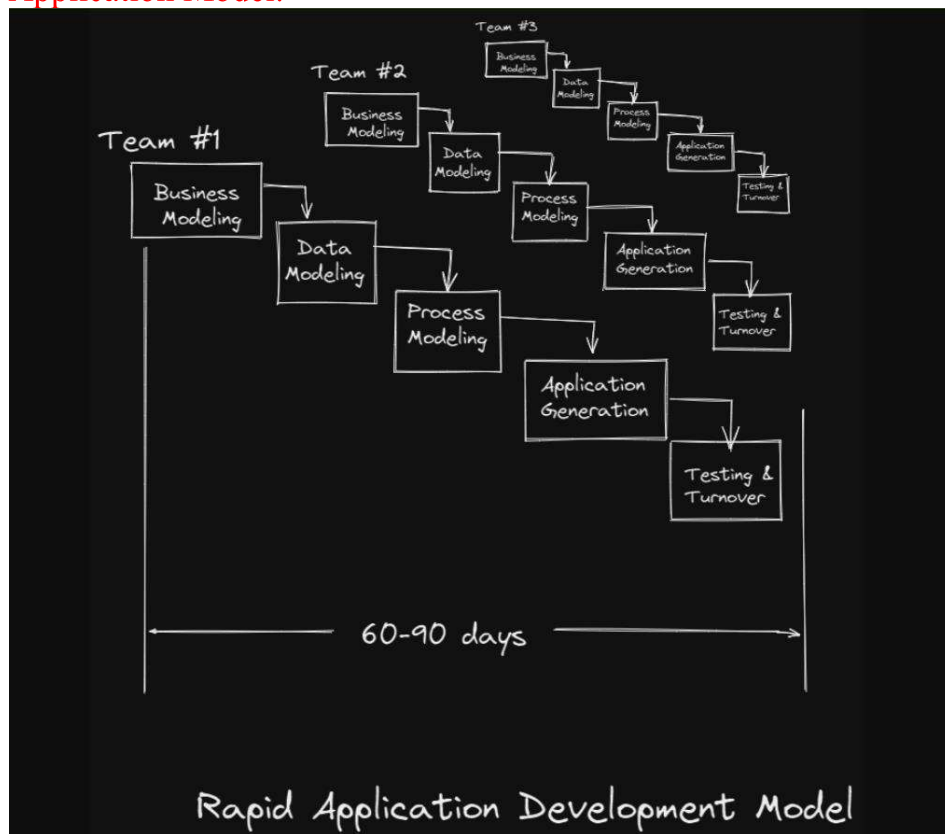
The testing was done by the team members itself for finding the errors and bugs.

#### 5. Deployment:

The application is to be submitted on 18<sup>th</sup> December 2021 on the Schoology platform in pdf format.

### Process Models

#### Rapid Application Model:



Our project has to be submitted in a short amount of time, so for this purpose we decided to use Rapid Application Model which focuses on dividing the functions into modules and carrying out the processes on the modules and then integrating the modules together.

The rapid application model is a high-speed adaptation of the linear model. In this model the incremental software

development process is emphasising on extremely short development cycle. In this model rapid development is achieved in very short-interval of time by using component based construction.

The RAD is further divided into 5 steps:

1. Business Modelling:

The information flow among business functions is modelled in a way that answers the following questions:

- What information drives the information process?
- What information is generated?
- Who generates it?
- Where does it go?
- Who processes it?

2. Data Modelling:

Set of data objects that are needed to support the business. The attributes of each object are identified and the relationships between these objects are defined.

3. Process Modelling:

The processing descriptions are created for adding, modifying, deleting or retrieving a data object.

4. Application generation:

Application development uses the fourth generation, techniques of RAD process. IT works to reuse existing program components. Automated tools are used to facilitate construction of the software.

5. Testing and turnover:

Since, RAD process emphasises on reuse, many program components have already been tested. This reduces overall testing time.

**People:**

The team leader Pratyay Dhond managed and defined the tasks for all the members of the team.

The modelling and constructing the various models and all other documents was done by the whole team, i.e. Pratyay Dhond, Priyanshu Lapkale, Aditi Khare, Harsh Naidu, Manthan Ghonge, Varad Nimbarte.

The designing of the UI was done by Priyanshu Lapkale.

The coding work for the application was done by Pratyay Dhond, Aditi Khare, Priyanshu Lapkale and Varad Nimbarte.

The testing of the software was done by Harsh Naidu and Manthan Ghonge.

The customer for our product were our faculty, i.e. Prof. L L Bhadekar sir.

The end users of the software product will be the students, teachers and book readers who want to get their books delivered to them.

**Product:**

**Software Scope:**

Book store app is an application software with the motive of providing access to books by getting them delivered to the user's homes once they have purchased the books from us.

**Functions:**

- Through our search for book functionality, the users can search and check the availability, summary of the book and price of the book easily.
- Through the rating functionality the customers can rate the book from a range of 1-5 stars.
- Through the payment functionality the users can pay for the books using credit cards.

**Process:**

- Rapid application model was used for the process.
- For process decomposition, the customer had given us very few details for the software, so we had to break down the components and needed to clarify certain points such as the filter by option and other details.

**Project:**

- Good decisions were made by defining proper jobs to the team members.
- Our major focus was on keeping the application minimalistic, easy to understand, and simple to use.
- We used Git as our version control system, and github for managing the software remotely.

**W5HH Principle:**

- i. Why is the system being developed?

This system is being developed in order to make the books available online and for making the purchase procedure simpler. The system will also be providing samples of the book in epub format for the end users to view the book's sample. This app will also have details of the customer so that the delivery can be done to their house/address.

- ii. What will be done, by when?

The time for each and every task was defined in the Gantt chart, for the key tasks: communication, planning, modeling(Use case diagram,

data flow diagram, state transition diagram, pert chart), construction, deployment. The time to be taken was defined as follows:

Tasks	Duration (hours)
Communication	1.5
Planning	2
Modelling	9
Construction	50
Deployment	1

- iii. Who is responsible for a function?
  - The tasks for the project were divided into team members according to their capabilities.
  - The team leader, Pratyay Dhond defined the tasks for the team members. The coding work was done by Pratyay Dhond, Priyanshu Lapkale and Aditi Khare.
  - The documentation work, i.e. the SRS was made by Varad Nimbarte and Harsh Naidu.
  - The testing of the software was done by Manthan Ghonge and Harsh Naidu.
- iv. Where are they organizationally located?
  - The client/customer for our project is Prof. L L Bhadekar sir.
- v. How will the job be done technically and managerially?
  - On the technical level, software development tools, such as android studio and Visual Studio code IDEs were used for the development of the software.
  - For the programming Flutter framework and dart language was used
  - Managerially the team was managed by the team leader, i.e. Pratyay Dhond and tasks were distributed by him to all the team members
- vi. How much of each resource is needed?
  - Human Resources i.e. the team members and customer will be needed for making the software and for making successful communication and deployment.
  - Hardware resources include an android device, with minimum 512MB RAM and an android version of Android 8.0 or higher.

# **Software Requirement Specification:**

## **Title: Online Book Store**

### **Table of Contents**

#### **1)Introduction**

##### **1.1 Purpose**

##### **1.2 Scope**

##### **1.3 Definition Acronyms and Abbreviations**

##### **1.4 Reference**

##### **1.5 Overview**

#### **2)Overall Description**

##### **2.1 Product Perspective**

##### **2.2 Product Functions**

##### **2.3 User Characteristics**

##### **2.4 Constraints**

##### **2.5 Assumptions and Dependencies**

#### **3)Specific Requirement**

##### **3.1 External Interfaces**

##### **3.2 Functions**

##### **3.3 Performance Requirements**

#### **4) Appendices**

#### **5) Index**



# 1) Introduction

## 1.1 Purpose

This Software Requirement Specification is meant for an Online Bookstore App. The Online Bookstore App is meant as a way for customers to browse books on the App and buy them from home without the need to travel to a book shop. Defining the functions and specifications of the Online Bookstore App is the primary purpose of this SRS. The SRS illustrates in clear terms, the system's primary uses.

## 1.2 Scope

- **Search the book**
- **Check the availability of the book**
- **Addition of book to Wishlist**
- **Removal of book from Wishlist**
- **Updating of book price and availability**
- **Book order management**
- **Ordering the book by customer**
- **Getting more information about the author**
- **The customer can register himself/herself on the app**
- **Taking the feedback of the customer**

## 1. 3)Definitions and Abbreviations

## Definitions:

Term	Description
Book	A single book belonging to an author.
Wishlist	Consists of items the user wishes to buy at a point of time
Cart	Consists of items the user wishes to buy
Customer	Store patron that orders/pays for ordering the book
Server	Operators of the Database that stores and manages the Book list
Genre	Filter to sort the booklist according to Genre of the book.
Author	Filter to sort the booklist according to Author of the book.
Year	Filter to sort the booklist according to year of book released.

## Acronyms:

Short form	Description
SRS	Software Requirement Specification
BD	Books Data
DBMS	Database Management System
LAN	Local Area Network
IP	Internet Protocol
TCP	Transmission Control Protocol
RAM	Random Access Management
IEEE 802.11	Wireless Local Area Network Standard
WPA2-PSK	Wi-Fi Protected Access 2 with Pre-Shared Key

## 1.4 References:

- **Google play books:**

[https://play.google.com/store/books?utm\\_source=apac\\_med&utm\\_medium=hasem&utm\\_content=Oct0121&utm\\_campaign=Evergreen&pcampaignid=MKT-EDRapac-in-1003227-med-hasem-bk-Evergreen-Oct0121-Text\\_Search\\_BKWSBKWS%7cONSEM\\_kwid\\_43700065205026403\\_creativeid\\_535350509885\\_device\\_c&gclid=Cj0KCQiAkZKNBhDiARIsAPsk0Wg3Lnsgk98J4WHVn1HFq2jo2RyVg11N5OGDFyR\\_8L2H4MY8U5tfdwaAIZNEALw\\_wcB&gclsrc=aw.ds](https://play.google.com/store/books?utm_source=apac_med&utm_medium=hasem&utm_content=Oct0121&utm_campaign=Evergreen&pcampaignid=MKT-EDRapac-in-1003227-med-hasem-bk-Evergreen-Oct0121-Text_Search_BKWSBKWS%7cONSEM_kwid_43700065205026403_creativeid_535350509885_device_c&gclid=Cj0KCQiAkZKNBhDiARIsAPsk0Wg3Lnsgk98J4WHVn1HFq2jo2RyVg11N5OGDFyR_8L2H4MY8U5tfdwaAIZNEALw_wcB&gclsrc=aw.ds)

- **Good reads:**

<https://www.goodreads.com/>

## 1.5 **Overview:**

By using this application, the customer of the app will be able to purchase books online via paying either by card or cash. The customer will be able to check the book details and give the rating in 1-5 star range. The customer can search for the books available on the application. The books can be updated according to the availability of a particular book. The application will generate a receipt on the server end of every book order. The application will reduce the redundancy in the data. By using this application, the book store can increase their efficiency by many folds. The customer can register himself/herself on the application.

## 2) Overall Description

The following section presents an overall description of the subject Book Application. In particular, the product has been put into perspective through a detailed assessment of the system, user, hardware, software, and communication interfaces, memory considerations, operational modes, and app adaptation requirements. Further, the characteristics of the system's end-users are discussed along with the identified system constraints and assumptions.

### 2.1 Product Perspective

#### **Business case:**

- The books will change the data of availability on the server side
- The books purchased will generate a report and get all the receipts from the server-side

#### **Customer case:**

- The user will open the applications
- The user will register himself/herself on the application
- the registering user has to enter their details on the registration of the app
- The user will search the book which they need
- Then the user will fill the delivery form and will click on order now.

- The user can fill the feedback form for the book shop

#### **Business Interface:**

- Server Admin Login
- Generate report
- Get all the receipts
- Real-time orders **Customer Interface:**

- User Register
- Feedback module
- More information module
- Books Module

#### **External Interfaces required for running Book Application:**

- A android system that can run a software
- Minimum 512MB of ram
- Internet connection with minimum 2mbps

## 2.2 Product Functions

Functions of Online Book Store are as follow:

**User Register:** By User register, the user can register themselves on the application.

- **Searching book:** The user can search the various book on the application.
- **Ordering an item:** The user can order the book by filling the delivery form.
- **Giving Feedback:** By this users can give feedback on the book items, on delivery services.

## 2.3 User Characteristics

#### **Customer:**

- The user should have a base knowledge of how to use an android application
- The user should know how to connect to the internet

## 2.4 Constraints

- This app will not run under the android version 7.0 or below platforms of android.
- The user using app should have a minimum memory of 512MB in the device.
- The user using login should have a minimum network speed of 2MBPS

## 2.5 Assumptions and Dependencies

#### **Assumptions**

- Users who use the app should use a valid username and password for registration.
- The application should have access to the internet.
- The user has a compatible android device with android version 8.0 or greater.

### Dependencies

- Hardware and software which we used will be run properly.
- End-users should have a proper understanding of the application.

## 3) Specific Requirement

### 3.1 Externals Interfaces

#### *User Interface*

- Android
- iOS

#### *Hardware Interfaces • Server-side*

- Operating system: Android 8.0 or above
- RAM: 512MB

### SOFTWARE INTERFACES

- Database
  - MYSQL

- *Programming Languages*
  - Flutter
  - Dart

- *COMMUNICATION INTERFACES*

- Dial-Up Modem
- Broadband Internet
- Cellular Data

### 3.2 Functions

#### **Functions accessible by Customers**

- **Create Account/Register**  
Customers can make their account and access various features like booking, etc Users have to mention details like Username, password, first name, last name, address, phone no, and email id
- **Customer Login**  
Let the user access their account and see the details stored in their account. To let the user login the user has to fill in a valid email address and correct password and the user should have created an account.
- **View Book Details**  
By using this the user checks the details of the available books.
- **Search book Items**  
By using this function users can search for books and in searching, users can apply filters for increasing their convenience and accuracy. Users need to enter the book genre to search for the required book.
- **Order**  
Customers can order the selected book items as per his/her choice .The order will be delivered by the delivery person to the customers location.

#### **Functions accessible for Book Shop**

- **Login/Register**
- **Homepage Module**
- **Feedback Module**
- **More-Info Module**

- **Pages Module**
- **Delivery Module**
- **Order Module**

### 3.3 Performances Requirements

- The application can accommodate 50 users during the peak usage time window of 8:00 am to 10:00 am local time, with an estimated average session duration of 45 minutes.
- All Page states generated by the application shall be fully loaded in no more than 10 seconds over a 40KBps modem connection.
- Responses to filters shall take no longer than 10 seconds to load onto the screen after the user submits the filters.
- The system shall display Order confirmation messages to users within 4 seconds after the user confirms the order to the application.

## 4) Appendices

- Flutter : A software development framework for cross platform application development
- Dart : A programming language for developing flutter apps.
- MYSQL : a Database programming language.
- RAM : Random Access Memory

## 5) Index

- |                             |         |
|-----------------------------|---------|
| 1. Functions                | 2.2,3.2 |
| 2. Overview                 | 1.5,2.4 |
| 3. Performance Requirements | 3.1,3.3 |

- **Conclusion:**

Hence, in this experiment, we have made an application – ‘Online book store app’ using all the concepts of the software engineering. We have applied various concepts such as the process model, W5HH principles, the 4P’s of project management, prepared a SRS document, managed risks, and made the product along with all the required diagrams of the product.

(10)	(20)	(10)	(10)	TOTAL