# CHAPTER 1

# INTRODUCTION

Android is a mobile operating system based on a modified version of the Linux kernel and other open source software, designed primarily for touch screen mobile devices such as smart phones and tablets. It is the most widely used operating systems, it's the highest-selling mobile operating system nowadays, Over 1 billion users already using this operating system mobiles, This is open source and anyone can use to build the applications. Android apps are developed with the Java coding language and the Android Software Development Kit, which runs on Mac, PC or Linux. Android software development kit (SDK) helps us to write original code and assemble software modules to create apps for Android users. Android also provides a marketplace to distribute apps. All together, Android represents an *ecosystem* for mobile apps. Android provides a touchscreen user interface (UI) for interacting with apps. Android's UI is mainly based on direct manipulation. People use touch gestures such as swiping, tapping, and pinching to manipulate on-screen objects. In addition to the keyboard, there's a customizable on-screen keyboard for text input. Android can also support game controllers and full-size physical keyboards connected by Bluetooth or USB.

The Android home screen can contain several panes of *app icons*, which launch their associated apps. Home screen panes can also contain *app widgets*, which display live, auto-updating content such as the weather, the user's email inbox, or a news ticker. Android can also play multimedia content such as music, animation, and video. The figure above shows app icons on the home screen (left), playing music (center), and displaying app widgets (right). Along the top of the screen is a status bar, showing information about the device and its connectivity. The Android home screen may be made up of several panes, and the user swipes back and forth between the panes.

Android is designed to provide immediate response to user input. Besides a dynamic interface that responds immediately to touch, an Android-powered device can vibrate to provide haptic feedback. Many apps take advantage of internal hardware such as accelerometers, gyroscopes, and proximity sensors to respond to additional user actions. These sensors can also detect screen rotation. For example, you could design a racing game where the user rotates the device as if it were a steering wheel.

To develop apps that take advantage of the Android operating system and UI, use the Android software development kit (SDK). The SDK includes software libraries of prewritten code, a debugger, a device emulator, documentation, sample code, and tutorials. Use the SDK to create apps that look great and take advantage of the hardware capabilities available on each Android-powered device.

As we know to develop apps using the SDK, we use the Java programming language to develop the app and Extensible Markup Language (XML) files to describe data resources. By writing the code in Java and creating a single app binary, we can create an app that can run on both phone and tablet form factors. We can declare our UI in lightweight sets of XML resources. At runtime, Android applies the correct resource sets based on the device's screen size, screen density, locale, and so on.

To help us develop our apps efficiently, Google offers an integrated development environment (IDE) called Android Studio. It offers advanced features for developing, debugging, and packaging Android apps. Using Android Studio, you can develop for any Android-powered device, or create virtual devices that emulate any hardware configuration.

While the Android platform provides rich functionality for app development, there are still a number of challenges we need to address, such as:

- Building for a multiscreen world
- Getting performance right
- Keeping your code and your users more secure
- Making sure your app is compatible with older platform versions
- Understanding the market and the user

We need to take precautions to make our code, and the user's experience when they use our app, as secure as possible.

- Using tools such as ProGuard, which is provided in Android Studio. ProGuard detects and removes unused classes, fields, methods, and attributes.
- Encrypt all of your app's code and resources while packaging the app.
- To protect critical user information such as logins and passwords, secure your communication channel to protect data in transit across the internet, as well as data at rest on the device.

In the recent trend as we know mobile-phones have become soulmates for public. They are making use of mobile-phones for every single need. Right from calling, texting, tracking, listening to music and so on they prefer to get data in their mobile-phones. Many new applications and technologies have come into existence these days, which people are using very efficiently. Still many are looking forward for new technologies in mobile-phones as we know android is an open source. The Android platform continues to improve and provide new features. There are many advantages of the android such as it provides, multitasking, easy accessibility to several apps, widget.

This project is followed the concept of the E-commerce thoroughly. The goal of this project is to design an online book store cum book library named Book Store which helps the user to read books of their choice anywhere anytime through this app. User can get to know different kinds of books that they were unaware of by just searching in the system using keywords. This project also helps the user to buy the books through e-cart. The user will be able to search the books they want, check the availability and place the order to buy the book using their credit cards. It’s users choice whether to buy the books or not as the app provides the facility to read the books on online platform itself.

Note: Some of the books may not be available for buying.

By using this project, the user can save his or her time of purchasing the product wasting time by roaming in the market in search of books i.e., this system saves both time and travelling cost of users or customers. From here, they can get most probably all the books they want besides fast-moving Entertainment goods. They have various choices in the one Collection.

# CHAPTER 2

# SYSTEM ANALYSIS

## DEFINITION

Systems analysis is the process of examining a business situation for the purpose of developing a system solution to a problem or devising improvements to such a situation or it also defined as a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components Before the development of any system can begin, a project proposal is prepared by the users of the potential system and/or by systems analysts and submitted to an appropriate managerial structure within the organization..

Analysis specifies **what the system should do**.

## FEASIBILITY STUDY

As the name implies, a feasibility analysis is used to determine the viability of an idea, such as ensuring a project is legally and technically feasible as well as economically justifiable. It tells us whether a project is worth the investment—in some cases, a project may not be doable. There can be many reasons for this, including requiring too many resources, which not only prevents those resources from performing other tasks but also may cost more than an organization would earn back by taking on a project that isn’t profitable.

A well-designed study should offer a historical background of the business or project, such as a description of the product or service, accounting statements, details of operations and management, marketing research and policies, financial data, legal requirements, and tax obligations. Generally, such studies precede technical development and project implementation.

## TYPES OF FEASIBILITY STUDY

A feasibility analysis evaluates the project’s potential for success; therefore, perceived objectivity is an essential factor in the credibility of the study for potential investors and lending institutions. There are five types of feasibility study—separate areas that a feasibility study examines, described below.

## 1. Technical Feasibility

This assessment focuses on the technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. Technical feasibility also involves the evaluation of the hardware, software, and other technical requirements of the proposed system.

## 2. Economic Feasibility

This assessment typically involves a cost/ benefits analysis of the project, helping organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated. It also serves as an independent project assessment and enhances project credibility—helping decision-makers determine the positive economic benefits to the organization that the proposed project will provide.

## 3. Legal Feasibility

This assessment investigates whether any aspect of the proposed project conflicts with legal requirements like zoning laws, data protection acts or social media laws.

## 4. Operational Feasibility

This assessment involves undertaking a study to analyze and determine whether—and how well—the organization’s needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

## 5. Scheduling Feasibility

This assessment is the most important for project success; after all, a project will fail if not completed on time. In scheduling feasibility, an organization estimates how much time the project will take to complete.

# CHAPTER 3

# SYSTEM REQUIREMENT SPECIFICATION

Software Requirement Specification (SRS) is a fundamental document, which forms the foundation of the software development process. SRS not only lists the requirements of a system but also has a description of its major features. These recommendations extend the IEEE standards. The recommendations would form the basis for providing clear visibility of the product to be developed serving as baseline for execution of a contract between client and the developer.

A system requirement is one of the main steps involved in the development process. It follows after a resource analysis phase that is the task to determine what a particular software product does. The focus in this stage is one of the users of the system and not the system solutions. The result of the requirement specification document States the intention of the software, properties and constraints of the desired system.

SRS constitutes the agreement between clients and developers regarding the contents of the software product that is going to be developed. SRS should accurately and completely represent the system requirements as it makes a huge contribution to the overall project plan.

The software being developed may be a part of the overall larger system or may be a complete standalone system in its own right. If the software is a system component, the SRS should state the interfaces between the system and software portion.

A good SRS defines the how Software System will interact with all internal modules, hardware, communication with other programs and human user interactions with wide range of real life scenarios. Using the Software requirements specification (SRS) document on QA lead, managers creates test plan. It is very important that testers must be cleared with every detail specified in this document in order to avoid faults in test cases and its expected results.

## 3.1 EXISTING SYSTEM

The existing system is a simple application with wide collection of books. Using this application the user can search for different types of books and read them. The user has search bar to search for the books of their choice when searched it displays the lists of books related to that topic and also displays the number of pages the book contains. The system had no central database for storing the information related to the user login. It does not have any facility of login. There was no proper data management in this existing system therefore to overcome all this a new system is proposed.

## 3.2 PROPOSED SYSTEM

The main objective of this proposed system was to overcome the drawbacks of the existing system. Proposed system mainly includes the login page which provides the user with login facility. Providing login option for the users makes them feel secure to use the application i.e., it provides user friendly environment. Other than previewing of the books the user also can buy the book through e cart. Once the registration is done the user can directly login and access the facilities provided. After the login process the user can search books of his/her choice and read the book anywhere anytime.

When the Preview option of the book is opened the user can find add to my library button on clicking provides the user with other options  Favorites, Reading now, To read and Have read. The user can mark any of the books as favorites which will be stored in my library. The user can go to my library and see all the related options with respect to the books read or to be read using his account. Clicking on My History gives the entire history of the user i.e., the books which are purchased by the user and the books which are reviewed and also the browsing history is provided.

### Advantages

1. It provides better and efficient services to users.

2. Faster retrieval of information about the desired book.

3. To provide wide collection of book ranges.

4. It saves both time and travelling cost of the user

5. Provides facility for proper monitoring.

6. All details will be available on a click.

## 3.3 REQUIREMENT SPECIFICATION

It’s the process of writing down the user and system requirements into a document. The requirements should be clear, easy to understand, complete and consistent. In practice, this is difficult to achieve as stakeholders interpret the requirements in different ways and there are often inherent conflicts and inconsistencies in the requirements. The user requirements for a system should describe the functional and non-functional requirements so that they are understandable by users who don’t have technical knowledge. The system requirements on the other hand are expanded version of the user requirements that are used by software engineers as the starting point for the system design.

### Functional Requirements:

The System must provide following functionalities

- Keeping records of registration of customers.
- Keeping the records of books.
- Storing the feedback given by the customer.
- Keeping details about the product it is delivered or not etc.
- Storing the items selected by the customer in the temporary storage.

### Non Functional Requirements:

Following Non-functional requirements will be there in the online shopping portal.

- Secure access of confidential data (customer’s details).
- 24 X 7 availability.
- Better component design to get better performance at peak time.

### 3.3.1 HARDWARE AND SOFTWARE REQUIREMENTS

### Hardware Requirements

Processor : intel CORE i5

Ram : 8GB

Hard Disk : 1TB

### Software Requirements

Operating System : Windows 7 and above

Coding Language : Java

Database : SQLLITE

Front End : XML

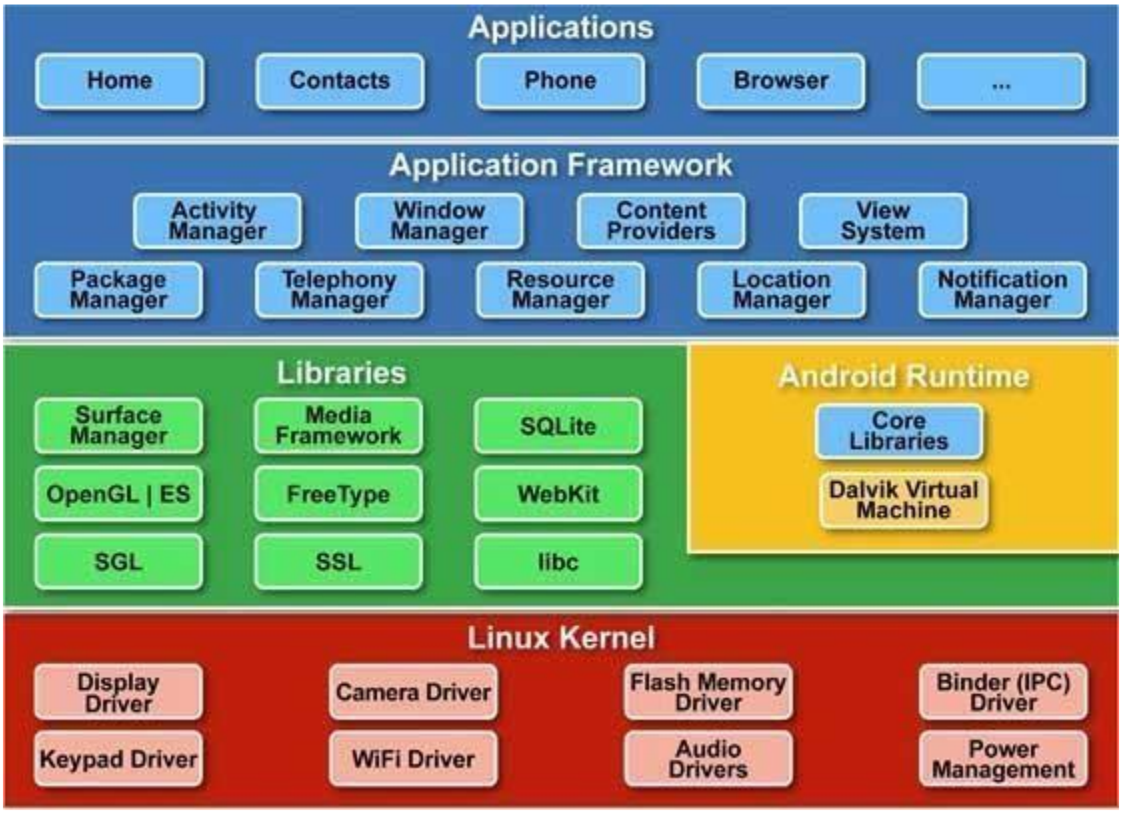
Tool Kit : Android studio 3.2

## 3.3.1 ABOUT THE TECHNOLOGIES USED

### Android

Android is a Linux based operating system and it is designed primarily for touch screen mobile devices such as smart phones and tablet computers. The operating system has developed a lot in the last 15 years starting from the black and white phones to recent smart phones or minim computers. One of the most widely used mobile OS in these days is android. Android is a software that was founded in Palo Alto of California in 2003.

Android is a powerful operating system and it supports large number of applications in smartphones. These applications are more comfortable and advanced for the users. The hardware that supports android software is based on the ARM architecture platform. Android is an open source, which means that its free and any one can use it. Android has got millions of apps available that can help us manage the tasks based in the requirement and it is available low cost in the market for which the android operating system is very popular.



**Android Architecture**

## Features of Android

With Android operating system (OS) KitKat version 4.4 rolling out to mobile devices across the globe, Google is on to something - Android is now more popular than iOS or the Windows phone. The OS tends to garner rave reviews for its ease of use and open source, but many iOS or Windows phone users argue that their preferred operating system can do just as much. In some respects that’s true, but here are 10 features unique to Android’s OS.

### 1) Near Field Communication (NFC)

Most Android devices support NFC, which allows electronic devices to easily interact across short distances. The main aim here is to create a payment option that is simpler than carrying credit cards or cash, and while the market hasn’t exploded as many experts had predicted, there may be an alternative in the works, in the form of Bluetooth Low Energy (BLE).

### 2) Alternate Keyboards

Android supports multiple keyboards and makes them easy to install; the SwiftKey, Skype, and 8pen apps all offer ways to quickly change up your keyboard style. Other mobile operating systems either don't permit extra keyboards at all, or the process to install and use them are tedious and time-consuming.

### 3) No-Touch Control

Using Android apps such as Wave Control, users can control their phones touch-free, using only gestures. Have messy hands but need to turn off your screen or change a song? Simple. This could prove especially useful if you're driving, so you can keep both eyes on the road.

### 4) Wireless App Downloads

Accessing app stores on any mobile device can be frustrating, but iOS makes it a little more difficult - download an app on your computer, and it won't sync to your mobile device until you plug in and access iTunes. Using the Android Market or third-party options like AppBrain, meanwhile, let you download apps on your PC and then automatically sync them your Droid, no plugging required.

### 5) Storage and Battery Swap

Android phones also have unique hardware capabilities. Google's OS makes it possible to remove and upgrade your battery or to replace one that no longer holds a charge. In addition, Android phones come with SD card slots for expandable storage.

## Introduction to Java

Java is a general-purpose computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of computer architecture. As of 2016, Java is one of the most popular programming languages in use, particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by James Gosling at Sun Microsystems (which has since been acquired by Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++ but it has fewer low level facilities than either of them.

The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licences. As of May 2007, in compliance with the specifications of the Java Community Process, Sun relicensed most of its Java technologies under the GNU General Public License. Others have also developed alternative implementations of these Sun technologies, such as the GNU Compiler for Java (bytecode compiler), GNU Class path (standard libraries), and Iced Tea Web (browser plugin for applets).

The latest version is Java 8, which is the only version currently supported for free by Oracle, although earlier versions are supported both by Oracle and other companies on a commercial basis.

**Features of Java Language**

- Simple
- Object-oriented
- Interoperability
- Type safe
- Platform Independence
- Security
- Reliability
- Language Feature

## SQLLITE DATABASE

SQLite is a popular choice as embedded database software for local/client storage in application software such as web browsers. It is arguably the most widely deployed database engine, as it is used today by several widespread browsers, operating systems, and embedded systems (such as mobile phones), among others. SQLite has bindings to many programming languages.

SQLite implements most of the SQL-92 standard for SQL but it lacks some features. For example, it partially provides triggers, and it cannot write to views (however it provides instead of triggers that provide this functionality). While it provides complex queries, it still has limited Alter table function, as it cannot modify or delete columns.

SQLite uses an unusual type system for an SQL-compatible DBMS; instead of assigning a type to a column as in most SQL database systems, types are assigned to individual values; in language terms it is dynamically typed. Moreover, it is weakly typed in some of the same ways that is: one can insert a string into an integer column (although SQLite will try to convert the string to an integer first, if the column's preferred type is integer). This adds flexibility to columns, especially when bound to a dynamically typed scripting language. However, the technique is not portable to other SQL products. A common criticism is that SQLite's type system lacks the data integrity mechanism provided by statically typed columns in other products. The SQLite web site describes a "strict affinity" mode, but this feature has not yet been added. However, it can be implemented with constraints like .

Tables normally include a hidden rowid index column which gives faster access. If a database includes an Integer Primary Key column SQLite will typically optimize it by treating it as an alias for rowid, causing the contents to be stored as a strictly typed 64-bit signed integer and changing its behavior to be somewhat like an auto-incrementing column. Future versions of SQLite may include a command to introspect whether a column has behavior like that of rowid to differentiate these columns from weakly-typed, non-autoincrementing Integer Primary Keys.

```
CHECK(typeof(x)='integer')
```

SQLite with full Unicode function is optional. Several computer processes or threads may access the same database concurrently. Several read accesses can be satisfied in parallel. A write access can only be satisfied if no other accesses are currently being serviced. Otherwise, the write access fails with an error code (or can automatically be retried until a configurable timeout expires). This concurrent access situation would change when dealing with temporary tables. This restriction is relaxed in version 3.7 when write-ahead logging (WAL) is turned on enabling concurrent reads and writes.

## My SQL

In this project, MySQL is used as the database. MySQL is an open source database management system. The features of MySQL are given below:

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- MySQL is a relational database management system. A relational database stores information in different tables, rather than in one giant table. These tables can be referenced to each other, to access and maintain data easily.

- MySQL is open source database system. The database software can be used and modify by anyone according to their needs. It is fast, reliable and easy to use. To improve the performance.

- MySQL is multithreaded database engine. A multithreaded application performs many tasks at the same time as if multiple instances of that application were running simultaneously.

- In being multithreaded MySQL has many advantages. A separate thread handles each incoming connection with an extra thread that is always running to manage the connections. Multiple clients can perform read operations simultaneously, but while writing, only hold up another client that needs access to the data being updated. Even though the threads share the same process space, they execute individually and because of this separation, multiprocessor machines can spread the thread across many CPUs as long as the host operating system supports multiple CPUs. Multithreading is the key feature to support MySQL's performance design goals. It is the core feature around which MySQL is built.

- MySQL database is connected to JSP using an ODBC driver. Open Database Connectivity (ODBC) is a widely accepted application programming interface (API) for database access. The ODBC driver is a library that implements the functions supported by ODBC API. It processes ODBC function calls, submits SQL requests to MySQL server, and returns results back to the application. If necessary, the driver modifies an application's request so that the request conforms to syntax supported by MySQL.

## Android SDK:

Android software development is the process by which new applications are created for devices running the Android operating system. Google states that "Android apps can be written using Kotlin, Java, and C++ languages" using the Android software development kit(SDK), while using other languages is also possible. All non-JVM languages, such as Go, JavaScript, C, C++ or assembly, need the help of JVM language code, that may be supplied by tools, likely with restricted API support. Some languages/programming tools allow cross-platform app support, i.e. for both Android and iOS. Third party tools, development environments and language support have also continued to evolve and expand since the initial SDK was released in 2008. In addition, with major business entities like Walmart, Amazon, Bank of America etc. eyeing to engage and sell through mobiles, mobile application development is witnessing a transformation.

# CHAPTER 4

# SYSTEM DESIGN

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization.

A systemic approach is required for a coherent and well-running system. Bottom-Up or Top Down approach is required to take into account all related variables of the system. A designer uses the modelling languages to express the information and knowledge in a structure of system that is defined by a consistent set of rules and definitions. The designs can be defined in graphical or textual modelling languages.

Some of the examples of graphical modelling languages are:

a) Unified Modelling Language (UML): To describe software both structurally and behaviourally with graphical notation.

b) Flowchart: A schematic or stepwise representation of an algorithm.

c) Business Process Modelling Notation (BPMN): Used for Process Modelling language

d) Systems Modelling Language (SysML): Used for systems engineering.

Some of the examples for designing methods are:

a) Architectural design: To describes the views, models, behaviour, and structure of the system.

b) Logical design: To represent the data flow, inputs and outputs of the system. Example: ER Diagrams (Entity Relationship Diagrams).

c) Physical design: Defined as a) How users add information to the system and how the system represents information back to the user. b) How the data is modelled and stored within the system. c) How data moves through the system, how data is validated, secured and/or transformed as it flows through and out of the system.

## Input Design

Very careful attention had to be given to input design, which is a major part of the overall system design. In order to make the data entry as easy, logical and error free as possible, specific standards had been followed. Validation checks, provided in the system prevented the user in entering incorrect, erroneous data. This made sure that, only valid data had been available for data processing. If valid data was entered, then meaningful error messages had been prompted to enter correct data. The interactive screen formats facilitate the entry of valid data.

The numbers of clear objectives of input design are,

- To produce a cost effective method of input
- To achieve the highest possible level of accuracy
- To ensure that the input is acceptable to and understand by the user

### Output Design

Output, as you probably know, generally refers to the results and information that are generated by the system. For many end-users, output is the main reason for developing the system and the basis on which they will evaluate the usefulness of the application. Most end users will not actually operate the information system or enter data through workstations, but they will use the output from the system.

When designing output, systems analysts must accomplish the following.

- Determine what information to present
- Arrange the presentation of information in an acceptable format.
- Decide how to distribute the output to intended recipients.

That alignment of information on a display or printed document is termed as layout. Accomplishing the general activities listed above will require specific decisions, such as whether to use pre printed forms when preparing reports and documents, how many lines to plan on a printed page, or whether to use graphics and color. The output design is specified on layout performs, sheets that describe the location characteristics, and format of the column headings and pagination. As we indicated at the beginning of this discussion, these elements are analogous to an architect's blue print that shows the location of the each component.

## 4.1 PROJECT ARCHITECTURE

A system architecture or systems architecture is the conceptual model that defines the structure, behaviour and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structure of the system which comprises system components, the externally visible properties of those components, the relationships (e.g. the behaviour) between them, and provides a plan from which products can be procured, and systems developed, that will work together to implement the overall system. The language for architecture description is called the architecture description language. This project consist many forms, each form is used to do particular task, below each form is explained in detail.

1. Login form
2. Sign in form/ new user form
3. Home page form
4. Preview form
5. Buy now form

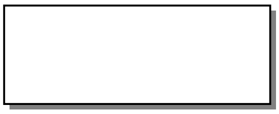
## 4.2 DATAFLOW DIAGRAM

A data flow is a graphical technique that describes information flow and transforms that are applied as data move from input to output. The DFD is also known as data flow graphs or bubble chart. The DFD is used to represent increasing information flow details. Also DFD can be stated as the starting point of the design phase that functionality decomposes the requirement specification down to the lowest level of detail. A level zero is also called the fundamental system model or context level DFD that represent the entire software elements as a single bubble with input and output data indicated by incoming and outgoing arrows respectively. Additional process and information flow parts are represented in the next level, i.e., level1 DFD. Each of the processes represented at level1 will be further represented into sub-functions in the next level i.e., level2.

Data flow diagram is a mean of representing a system at any level of detail with a graphic network of symbols showing data flows, data stores, data processes and data sources. The purpose of data flow diagram is to provide semantic bridge between users and system developers. The diagram is the basis of structured system analysis. A DFD describes what data flows rather than how they are processed, so it does not depend on hardware, software, data structure, or file organization.

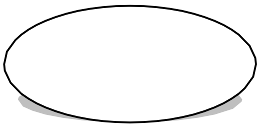
Components of Data Flow Diagram There are four symbols that are used in the drawing of Data Flow Diagrams.

- **ENTITIES**



External entities represent the sources of data that enter the system or the recipients of data that leave the system.

- **PROCESS**



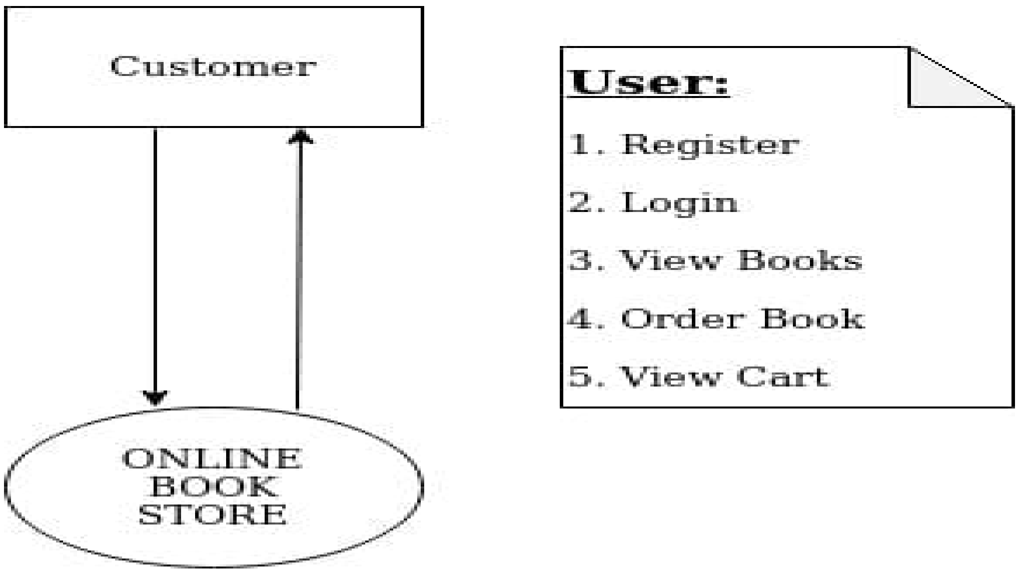
Processes represent data in which data is manipulated by being stored or retrieve or transformed in some way. A circle represents it. The process will show the data information or charge.

- **DATABASE**

Database represents storage of data within the system

- **DATA FLOW**

A data shows a flow of information from its source to its destination. A line represents a dataflow, with arrow heads showing the direction of flow.

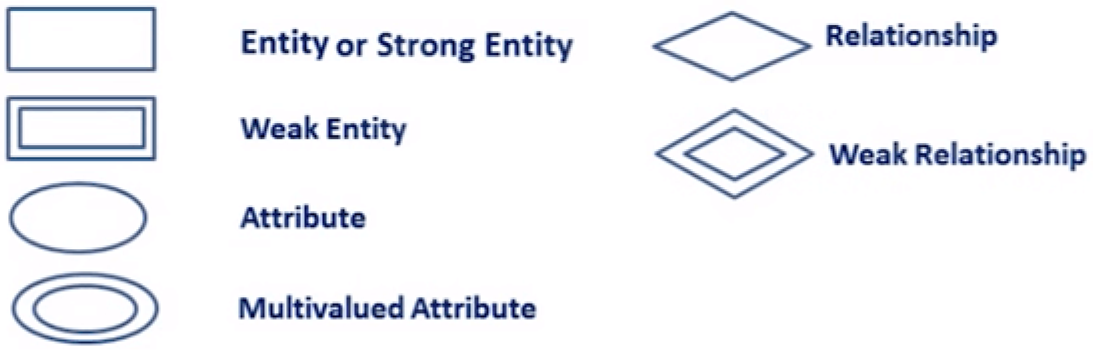


## 4.3 ER DIAGRAM

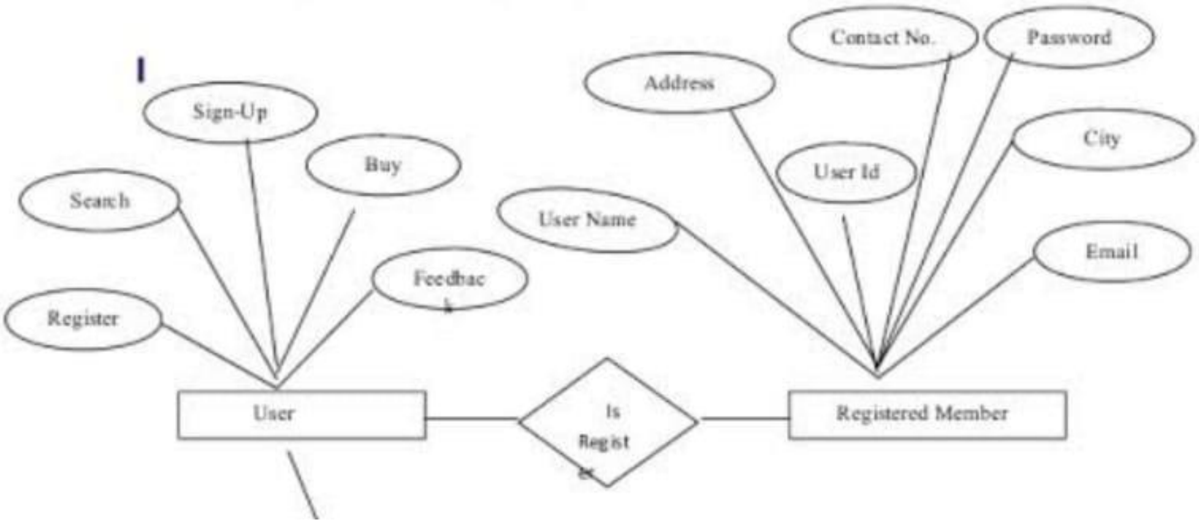
An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs. ER diagrams are related to data structure diagrams (DSDs), which focus on the relationships of elements within entities instead of relationships between entities themselves. ER diagrams also are often used in conjunction with data flow diagrams (DFDs), which map out the flow of information for processes or systems.

**Following are the main components and its symbols in ER Diagrams:**

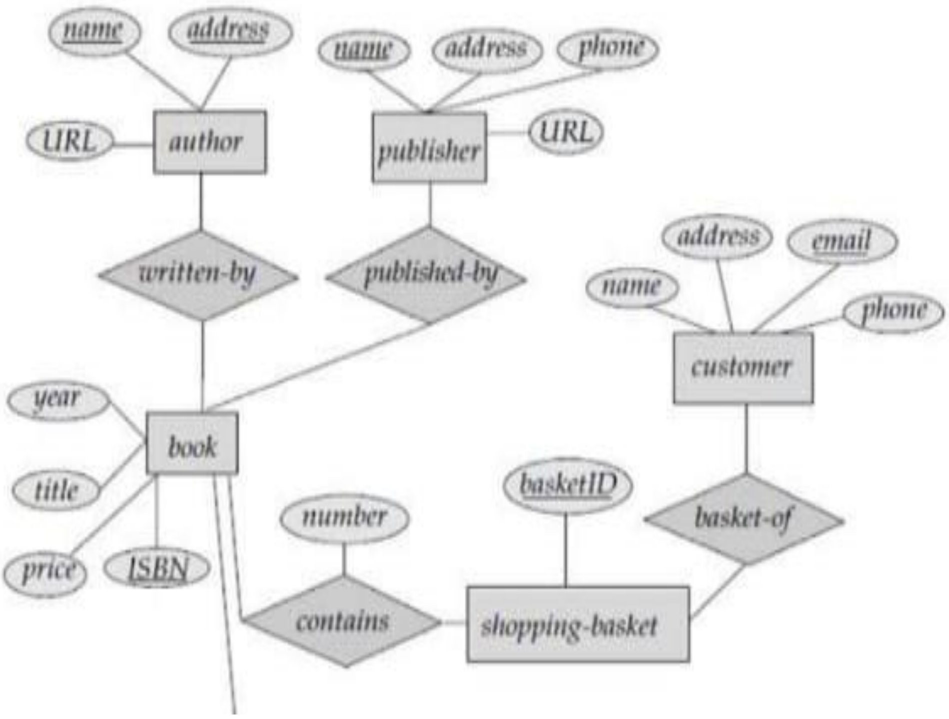
- **Rectangles:** This Entity Relationship Diagram symbol represents entity types
- **Ellipses :** Symbol represent attributes
- **Diamonds:** This symbol represents relationship types
- **Lines:** It links attributes to entity types and entity types with other relationship types
- **Primary key:** attributes are underlined
- **Double Ellipses:** Represent multi-valued attributes



**ER Diagram for User**



**ER Diagram for Online Book Store**



## 4.4 DATABASE DESIGN (TABLES)

A table is a collection of related data held in a table format within a database. It consists of columns and rows. "Table" is another term for "relation".

In relational databases, and flat file databases, a *table* is a set of data elements (values) using a model of vertical columns (identifiable by name) and horizontal rows, the cell being the unit where a row and column intersect. A table has a specified number of columns, but can have any number of rows. Each row is identified by one or more values appearing in a particular column subset. A specific choice of columns which uniquely identify rows is called the primary key.

**1.User Table**

| Column Name | Data Type | Index |
|---|---|---|
| Email | Varchar | |
| Password | Varchar | |

**2.Register Table**

| Column Name | Data Type | Index |
|---|---|---|
| Id | Integer | Primary Key |
| Name | Varchar | |
| Phone number | Varchar | |
| Gmail | Varchar | |
| Password | Varchar | |

# CHAPTER 5

# IMPLEMENTATION

Implementation is the process of converting a or a revised system design into an operational one. The Objective is to put the new or revised system that has been tested into operation While holding costs, risks, and personal irritation to the minimum. A critical aspect or the implementation process is to ensure that there will be no disrupting the functioning or the organization. The best method for gaining control while implanting any new system would be to use well planned test for testing all new programs. Before production files are used to test live data, text files must be created on the old system, copied over to the new system, and used for the initial test of each program

## CODING OF HOME PAGE

### 1. MainActivity.XML

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/new_login_gradients"
    android:orientation="vertical"
    android:paddingBottom="16dp"
    android:paddingLeft="18dp"
    android:paddingRight="18dp"
    android:paddingTop="16dp"
    tools:context=".MainActivity">
    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_gravity="center"
    android:orientation="vertical">



    <EditText
        android:id="@+id/etLogGmail"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="50dp"
        android:background="@android:color/transparent"
        android:drawablePadding="12dp"
        android:ems="10"
        android:inputType="textEmailAddress"
        android:padding="8dp"
        android:hint="Email"
        android:textColorHint="#ffffff"
        android:maxLines="1"
        android:drawableLeft="@drawable/ic_email_white_18dp" />
    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="#f9d7db" />
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:ems="10"
        android:id="@+id/etLoginPassword"
        android:drawablePadding="12dp"
        android:paddingLeft="8dp"
        android:paddingRight="8dp"
        android:paddingTop="8dp"
        android:paddingBottom="8dp"
        android:textColorHint="#ffffff"
        android:hint="Password"
        android:maxLines="1"
        android:background="@android:color/transparent"
        android:layout_marginTop="4dp"
        android:drawableLeft="@drawable/ic_lock_white_24dp" />
    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="#f9d7db" />
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/btnLogin"
        android:background="@drawable/background_login"
        android:text="Login"
        android:textAllCaps="false"
        android:textStyle="normal"
        android:layout_marginTop="30dp"
        style="@style/Base.TextAppearance.AppCompat.Body1"
        android:textSize="18sp"
        android:textColor="#f16f7d"
        android:padding="16dp"
        android:clickable="true"/>

    <TextView
        android:id="@+id/tvRegister"
        android:layout_width="199dp"
        android:layout_height="40dp"
        android:paddingTop="6dp"
```

```
            android:layout_marginEnd="8dp"
            android:layout_marginStart="8dp"
            android:layout_marginTop="18dp"
            android:layout_gravity="center_horizontal"
            android:text="New User? Register"
            android:textSize="22sp" />

        </LinearLayout>
    </ScrollView>

</LinearLayout>
```

## 2.MainActivity.java

```
package com.example.librapp;

import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    private TextView tvRegister;
    private EditText etLoginGmail,etLoginPassword;
    private Button loginButton;

    private SQLiteDatabase db;
    private SQLiteOpenHelper openHelper;
    private Cursor cursor;



    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        openHelper = new MyHelper(this);
        db = openHelper.getReadableDatabase();
        tvRegister = findViewById(R.id.tvRegister);
        etLoginGmail = findViewById(R.id.etLogGmail);
        etLoginPassword = findViewById(R.id.etLoginPassword);
        loginButton = findViewById(R.id.btnLogin);

        loginButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String email = etLoginGmail.getText().toString().trim();
                String password = etLoginPassword.getText().toString().trim();
                if (email.isEmpty() || password.isEmpty()) {
                    Toast.makeText(MainActivity.this, "Enter your Email and Password to login",
Toast.LENGTH_SHORT).show();
                } else {
                    cursor = db.rawQuery("SELECT *FROM " + MyHelper.TABLE_NAME + "
WHERE " + MyHelper.COL_4 + "=? AND " + MyHelper.COL_5 + "=?", new
String[]{email, password});
                    if (cursor != null) {
                        if (cursor.getCount() > 0) {
                            startActivity(new Intent(MainActivity.this, Search.class));
                            Toast.makeText(getApplicationContext(), "Login sucess",
Toast.LENGTH_SHORT).show();
```

```
                        } else {
                            Toast.makeText(getApplicationContext(), "Login error",
Toast.LENGTH_SHORT).show();
                        }
                    }
                }
            }
        });



        tvRegister.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(MainActivity.this,Registeration.class));
                finish();
            }
        });

    }
}
```

# CHAPTER 6

# TESTING

Testing is a process, which reveals errors in the program. It is the major quality measure employed during software development. During software development, during testing the program is executed with a set of test cases and the output of the program for the cases is evaluated to determine if the program is performing as it is expected to perform.

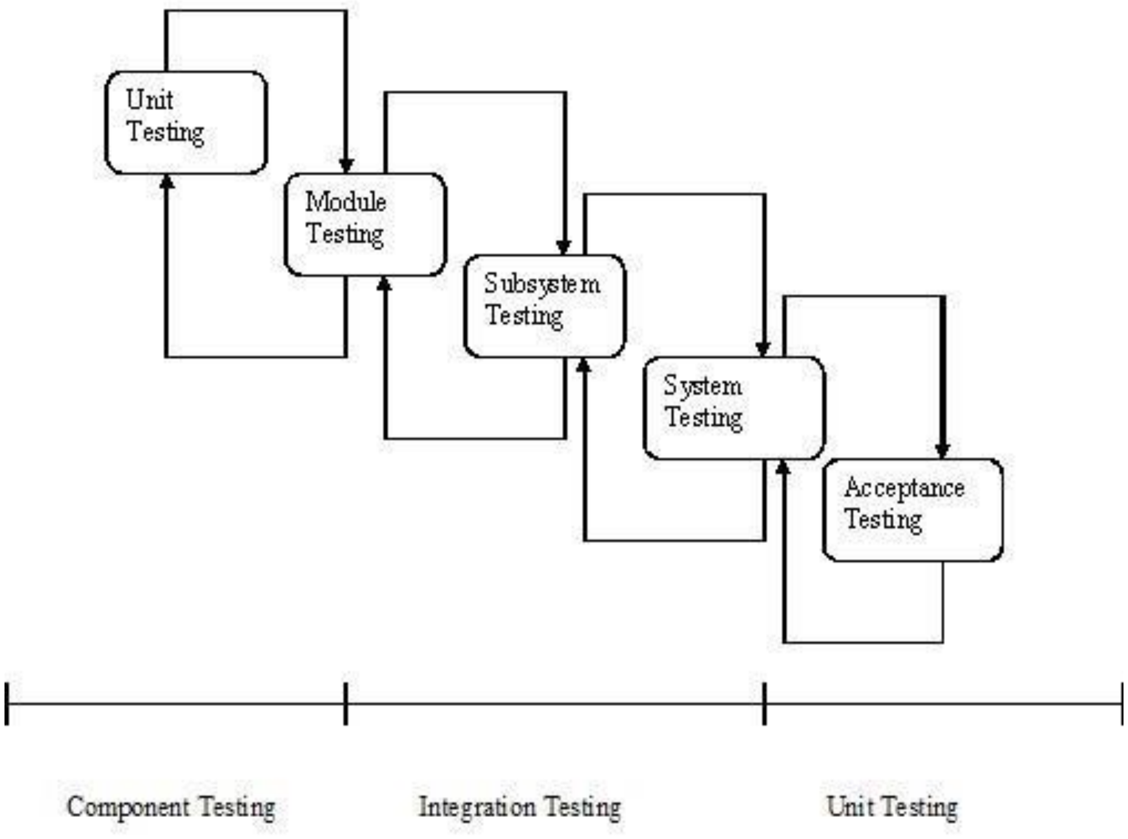
## 6.1 ABOUT TESTING

Testing is the major process involved in software quality assurance (QA). is iterative Here test data is prepared and is used to test the modules individually. System testing sure that all components or the system function properly as a unit by actually forcing the system to fail.

The test causes should be planned before testing begins. Then as the testing progresses, testing shifts focus in an attempt to find errors in integrated clusters of modules. The philosophy behind testing is to find errors. Actually, testing is the state of implementation that is aimed at ensuring that the system works actually and efficiently before implementation.

Testing is done for each module. After testing all the modules, the modules are integrated and testing of the final system is done with the test data, specially designed to show that the system Will operate successfully in all its aspects conditions. The procedure level testing is made first. By giving improper inputs, the errors occurred are noted and eliminated. Thus, the system testing is a confirmation that all is correct and an opportunity to show the user that the system works. Ille final step involves Validation testing, which determines whether the software function as the user expected. The end-user rather than the system developer conduct this test most software developers as a process called "Alpha and Beta test" to uncover that only the end user seems able to find.

This is the final Step in system life cycle. Here we implement the tested error-free system into real-life environment and make necessary changes, which runs in an online fashion. Here system maintenance is done every months or year based on company policies, and is checked for errors like runtime errors, long run errors and other maintenances like table verification and During the requirement analysis and design, the output is a document that is usually textual and non-executable. After the coding phase, computer programs are available that can be executed for testing purpose. This implies that testing not only has to uncover errors introduced during coding, but also errors introduced during the previous phases.



**The Testing Process**

## 6.2 TYPES OF TESTING

The various types of testing done on the system are:

Unit Testing

Integration Testing

Validation Testing

System Testing

Acceptance Testing

### 1. Unit Testing

Unit testing is a process verification effort on the smallest unit of software i.e., the module. Using the detailed design and the process specification testing is done to uncover errors within the boundary of the module. All modules must be successful in the unit test before the start of the integration testing begins. In this project each service can be thought of a module.

Giving different sets of inputs has tested each module. When developing the module as well as finishing the development so that each module works without any error. The inputs are validated when accepting from the user. In this application developer tests the programs up as system. Software units in a system are the modules and routines that are assembled and ignored to form a specific function. Unit testing is first done on modules, independent of one another to locate errors. This enables to detect errors. Through this error resulting from interaction between modules initially avoided.

### 2. Integration Testing

Integration Testing is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. The purpose of this level of testing is to expose defects in the interaction between these software modules when they are integrated

Integration Testing focuses on checking data communication amongst these modules. Hence it is also termed as ‘I & T’ (Integration and Testing), ‘String Testing’ and sometimes ‘Thread Testing’**.**

## 3. Validation Testing

Validation testing in software engineering is in place to determine if the existing system complies with the system requirements and performs the dedicated functions for which it is designed along with meeting the goals and needs of the organisation.

## 4. System Testing

Once individual module testing completed, modules are assembled to perform as a system. Then the top down testing, which begins from upper level to lower level module testing, to done to check whether the entire system is performing satisfactorily. After unit and integration testing are over then the system as whole is tested. There are two general strategies for system testing.

They are:

Code Testing

Specification Testing

- **Code Testing**

This strategy examines the logic of the program. A path is a specific combination of conditions handled by the program. Using this strategy, every path through the program is tested.

- **Specification Testing**

This strategy examines the specifications stating what the program should do and how it should perform under various conditions. The test cases are developed for each condition of developed System and processed. It is found that the system developed perform according to its specified requirements. The system is used experimentally to ensure that the software will run according to tits specification and in the way user expects. Specification Testing is done successfully by entering various types of end data. It is checked for both valid and invalid data and found System is working properly as per requirement.

### 5.Acceptance Testing

When the system has no measure problem with its accuracy, the system passes through a final acceptance test. This test confirms that the system needs the original goal, Objective and requirements established during analysis. If the system fulfils all the requirements, it is finally acceptable and ready for operation.

We hope that after the acceptance testing the system will perform the best result for the organization. When modification will be made, we will use regression testing during the maintenance of the system. The Software System delivered to the customer may undergo changes. Changes may be due to addition of new functional modules or performance enhancement .For this purpose proper maintenance of the system is must.

## 6.3 TEST CASES

A **TEST CASE** is a set of actions executed to verify a particular feature or functionality of your software application. A Test Case contains test steps, test data, precondition, postcondition developed for specific test scenario to verify any requirement. The test case includes specific variables or conditions, using which a testing engineer can compare expected and actual results to determine whether a software product is functioning as per the requirements of the customer.

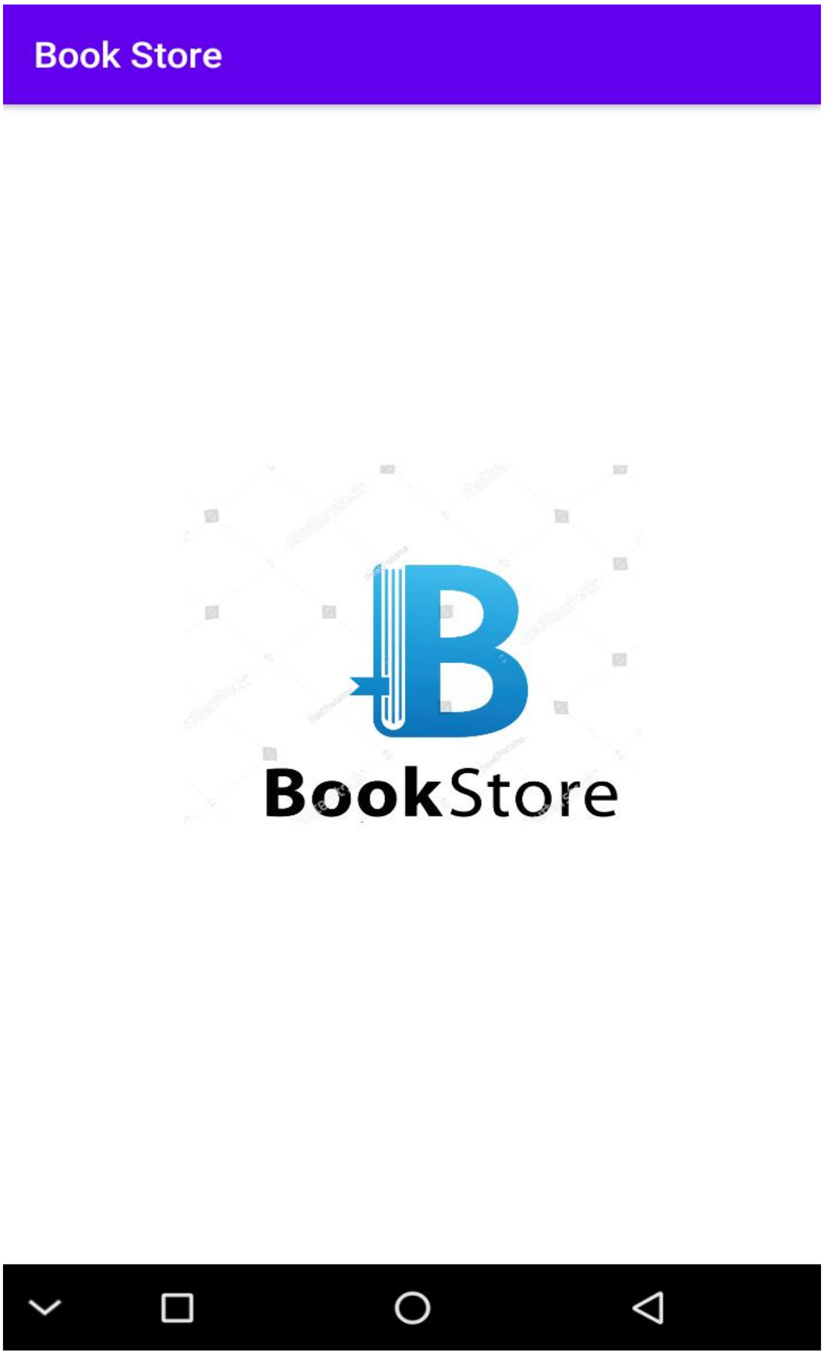
Login Form and Registration Page

| SL NO | TEST CASE | TEST INPUT | EXPECTED OUTPUT | ACTUAL OUTPUT | STATUS (PASS/FAIL) |
|---|---|---|---|---|---|
| 1 | User Login | Email<br>Password | Login success | Login success | Pass |
| 2 | User Login | Email<br>Password | Login success | Account do not exists | Fail |
| 3 | User Registration | Name<br>Phone Number<br>Email<br>Password | Account Created Successfully | Account Created Successfully | Pass |

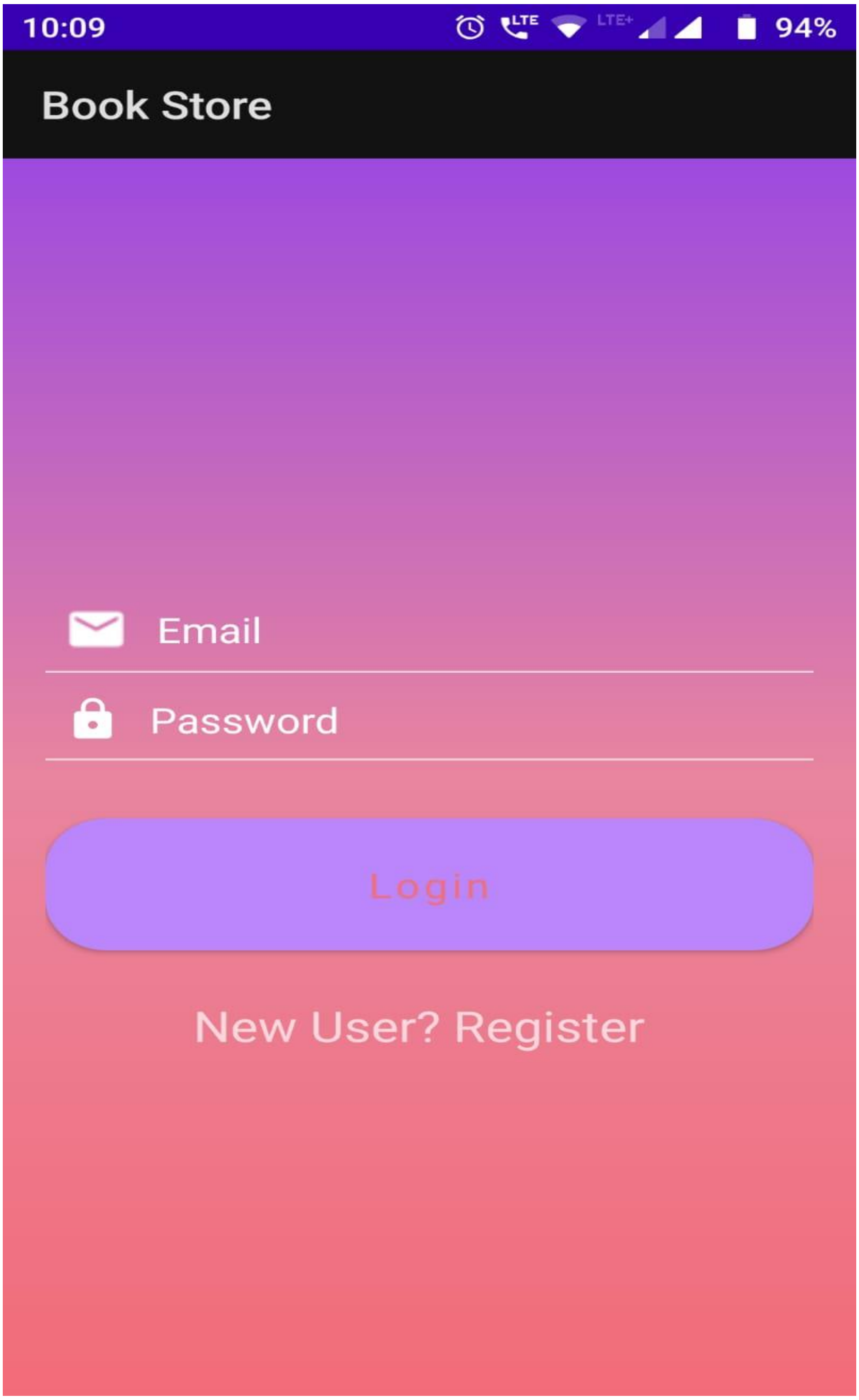
# CHAPTER 7

# SCREENSHOTS

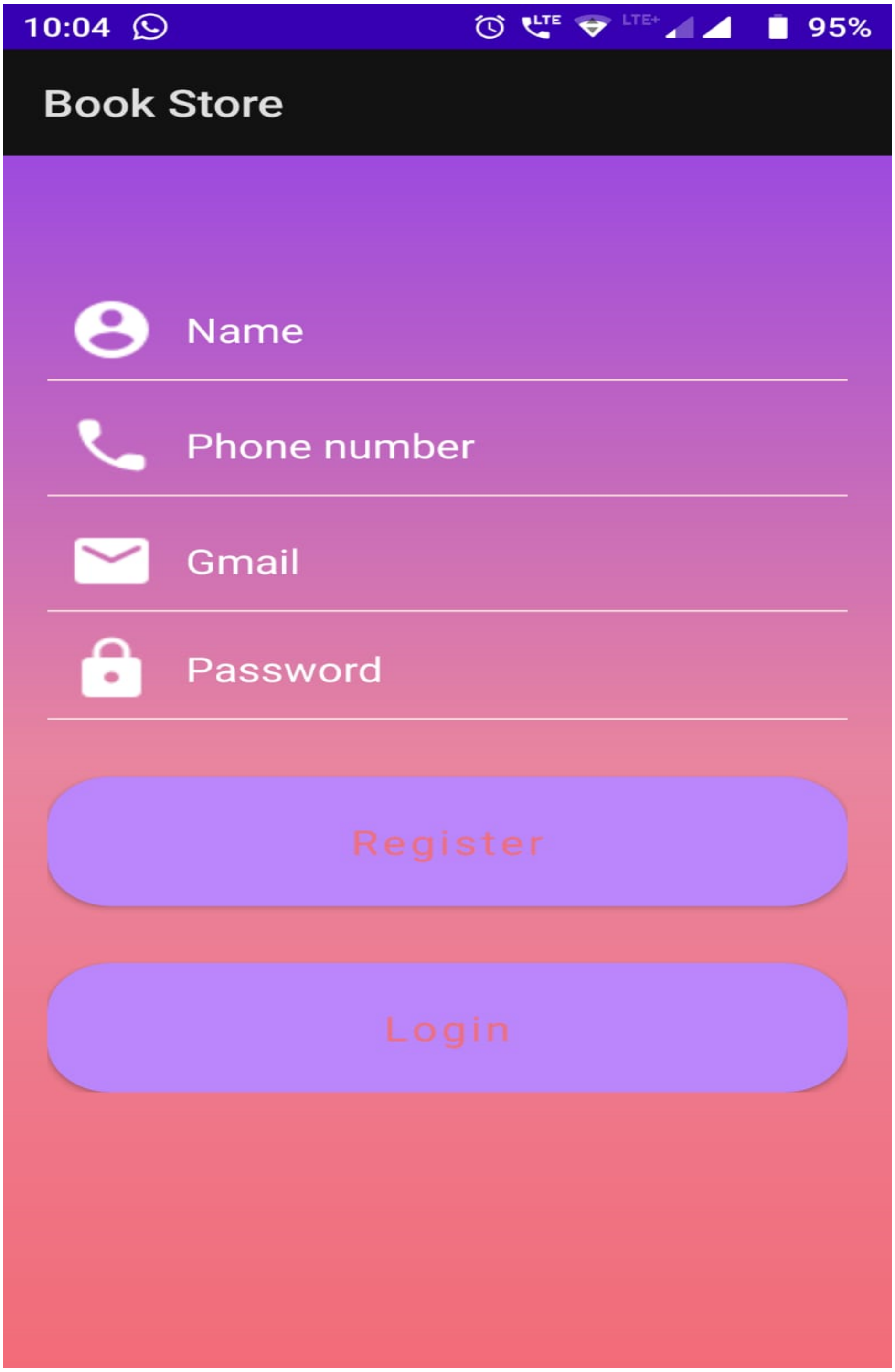
## 1. Logo



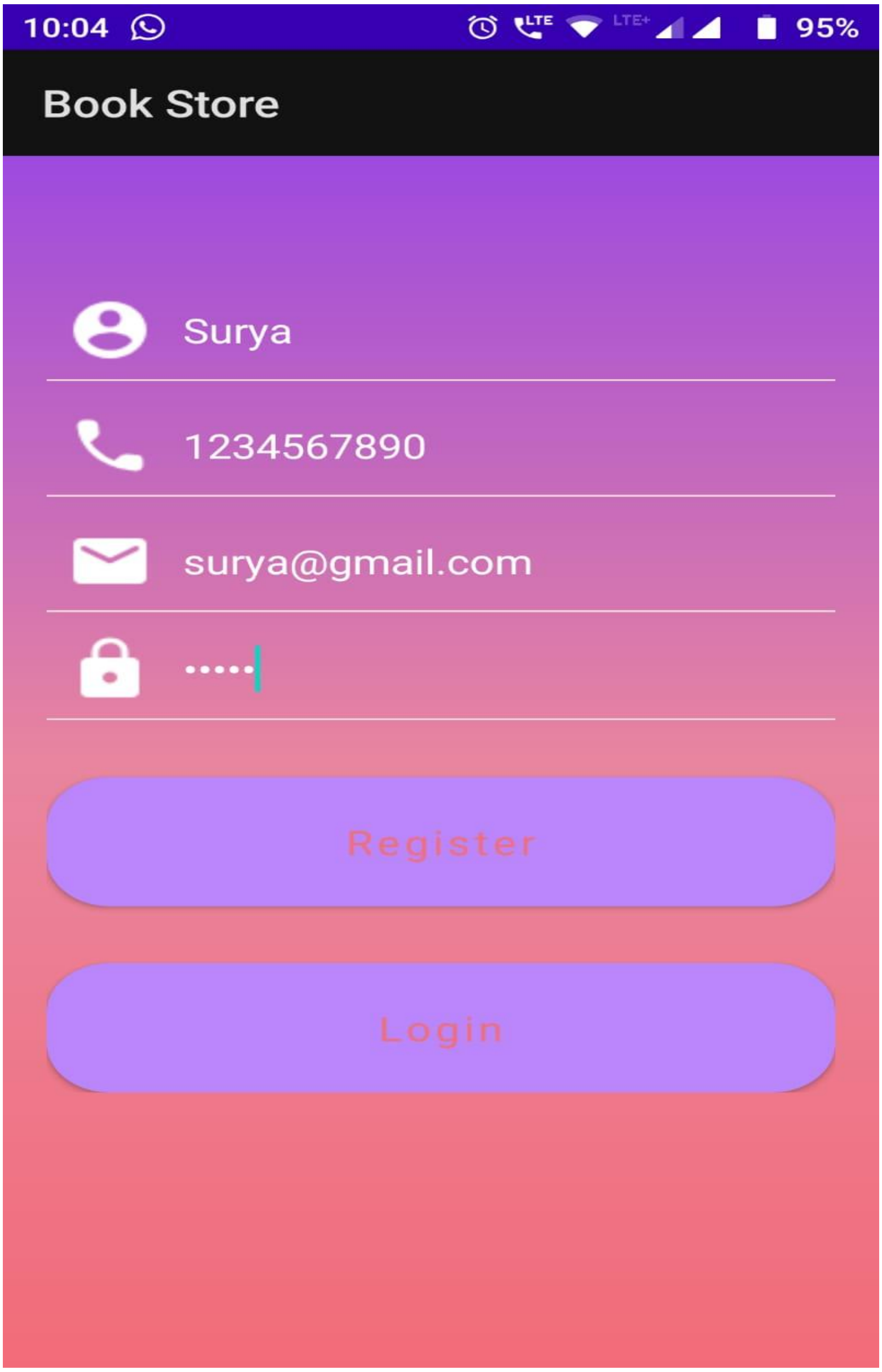
## 2.Login Page



## 3.Register Page



## 4.Registration Process



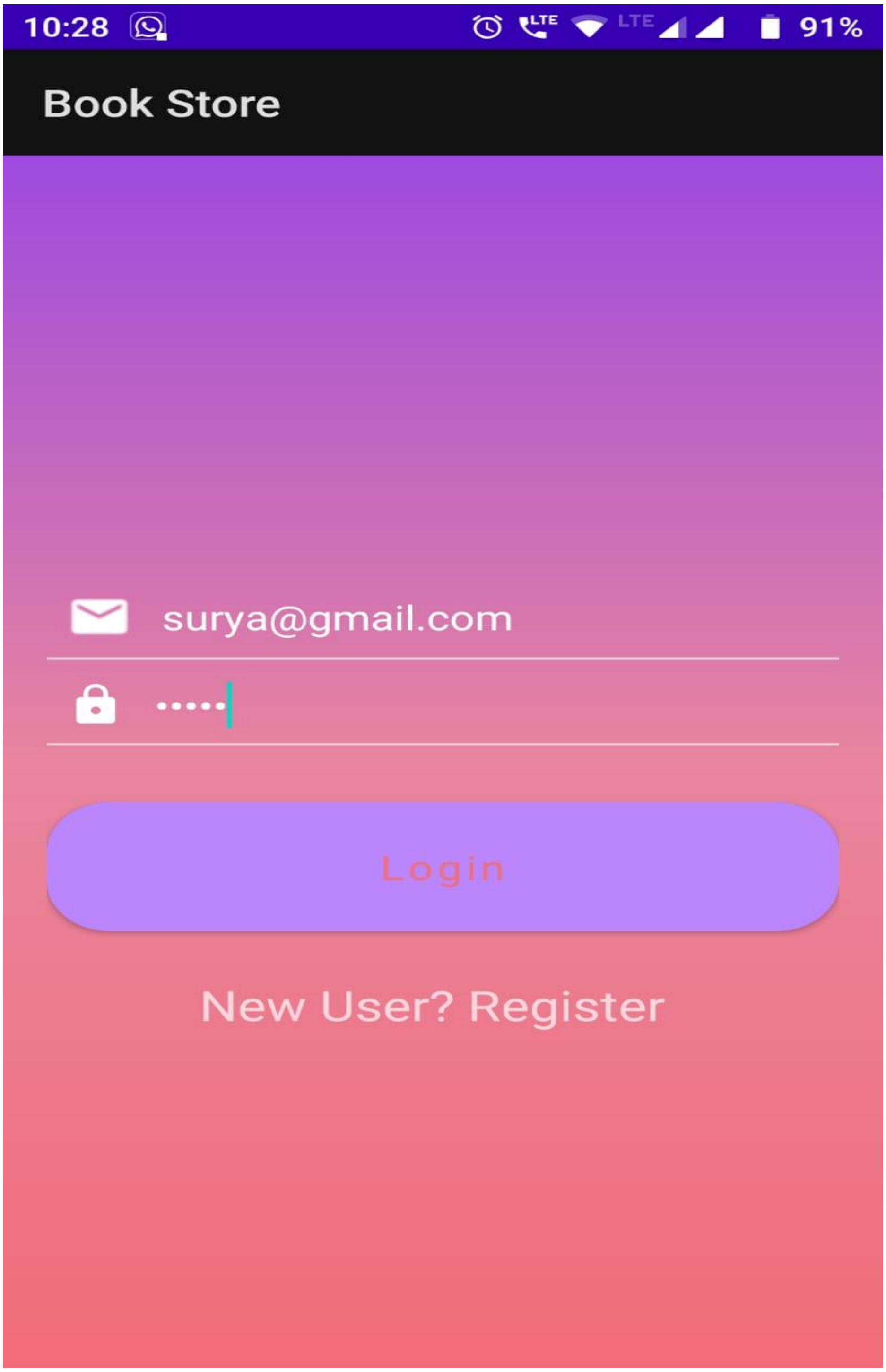
Surya

1234567890

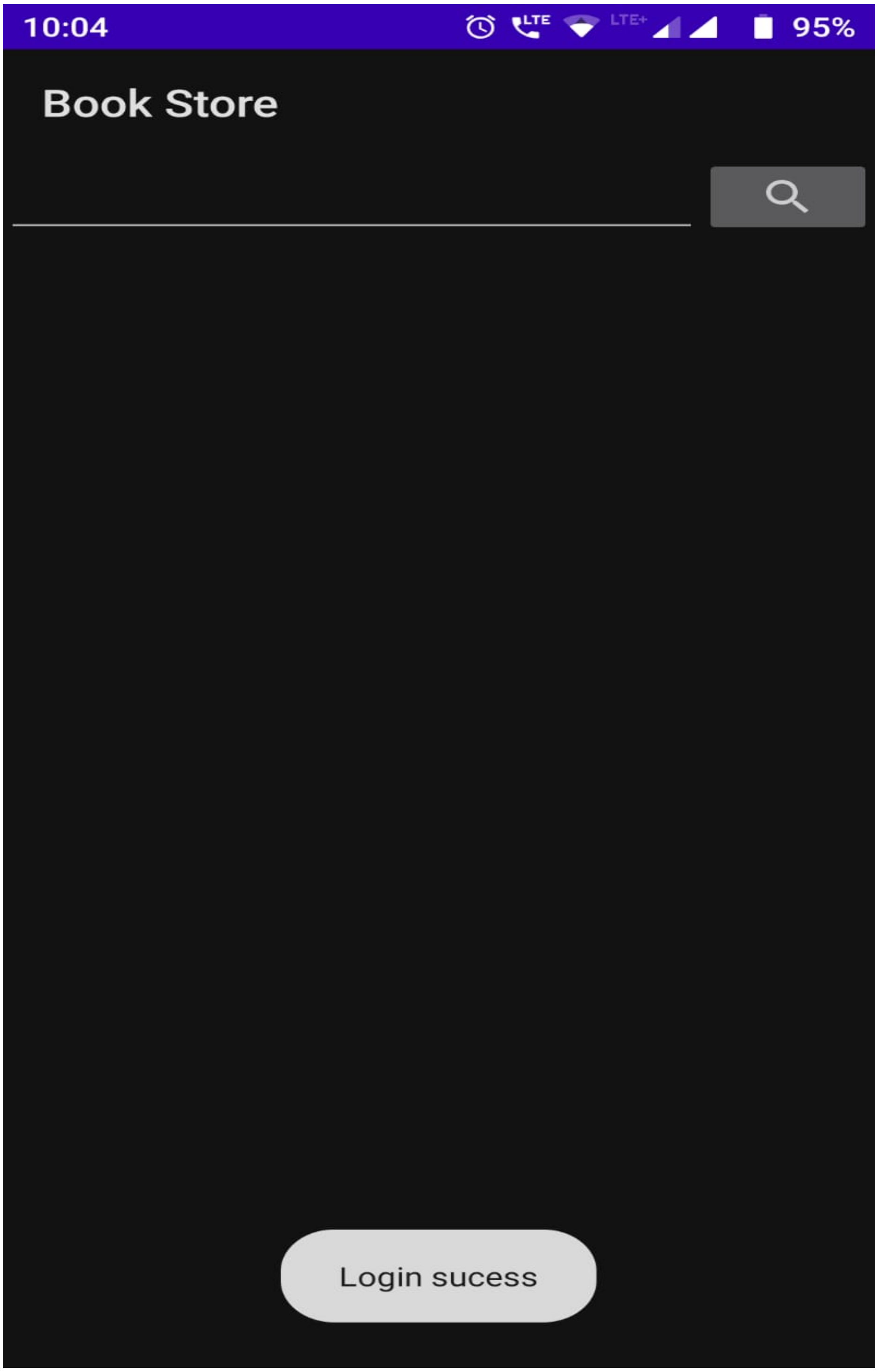
surya@gmail.com

•••••

## 5.User Login

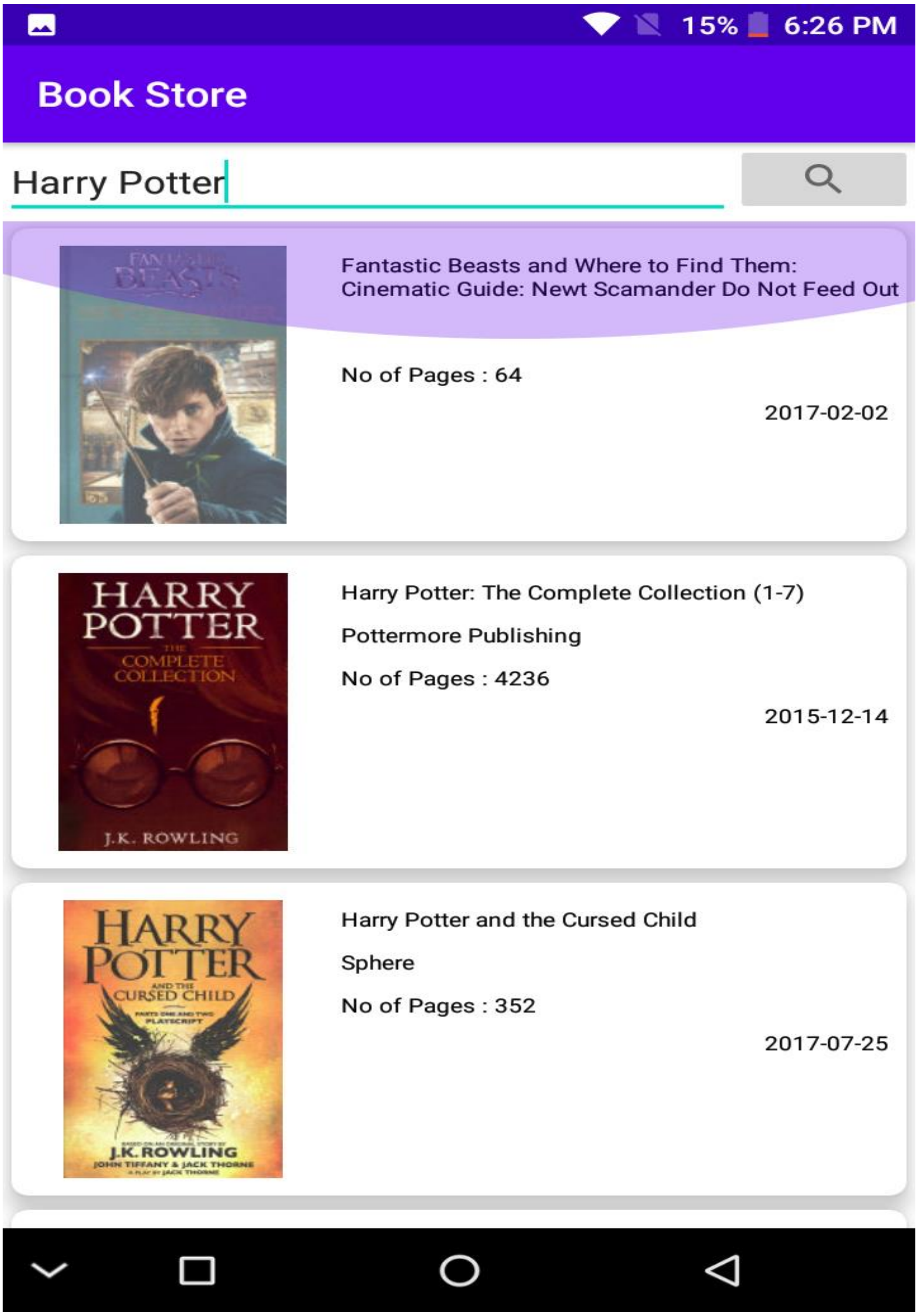


## 6.Login Status



Login sucess

## 7.Searching of Books



Book Store

Harry Potter

Fantastic Beasts and Where to Find Them: Cinematic Guide: Newt Scamander Do Not Feed Out

No of Pages : 64

2017-02-02

Harry Potter: The Complete Collection (1-7)

Pottermore Publishing

No of Pages : 4236

2015-12-14

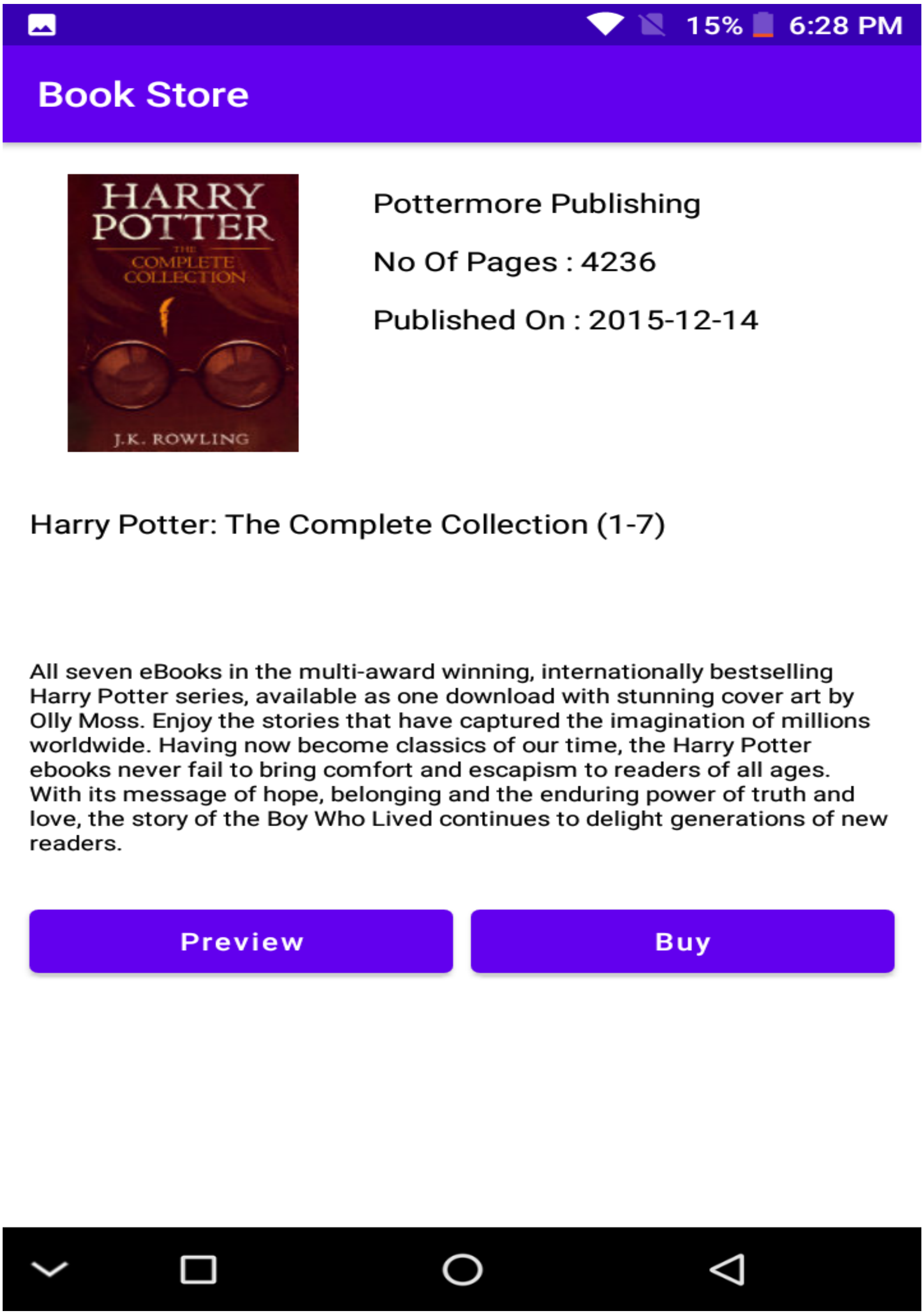
Harry Potter and the Cursed Child

Sphere

No of Pages : 352

2017-07-25

## 8.Preview Page



Book Store

Pottermore Publishing

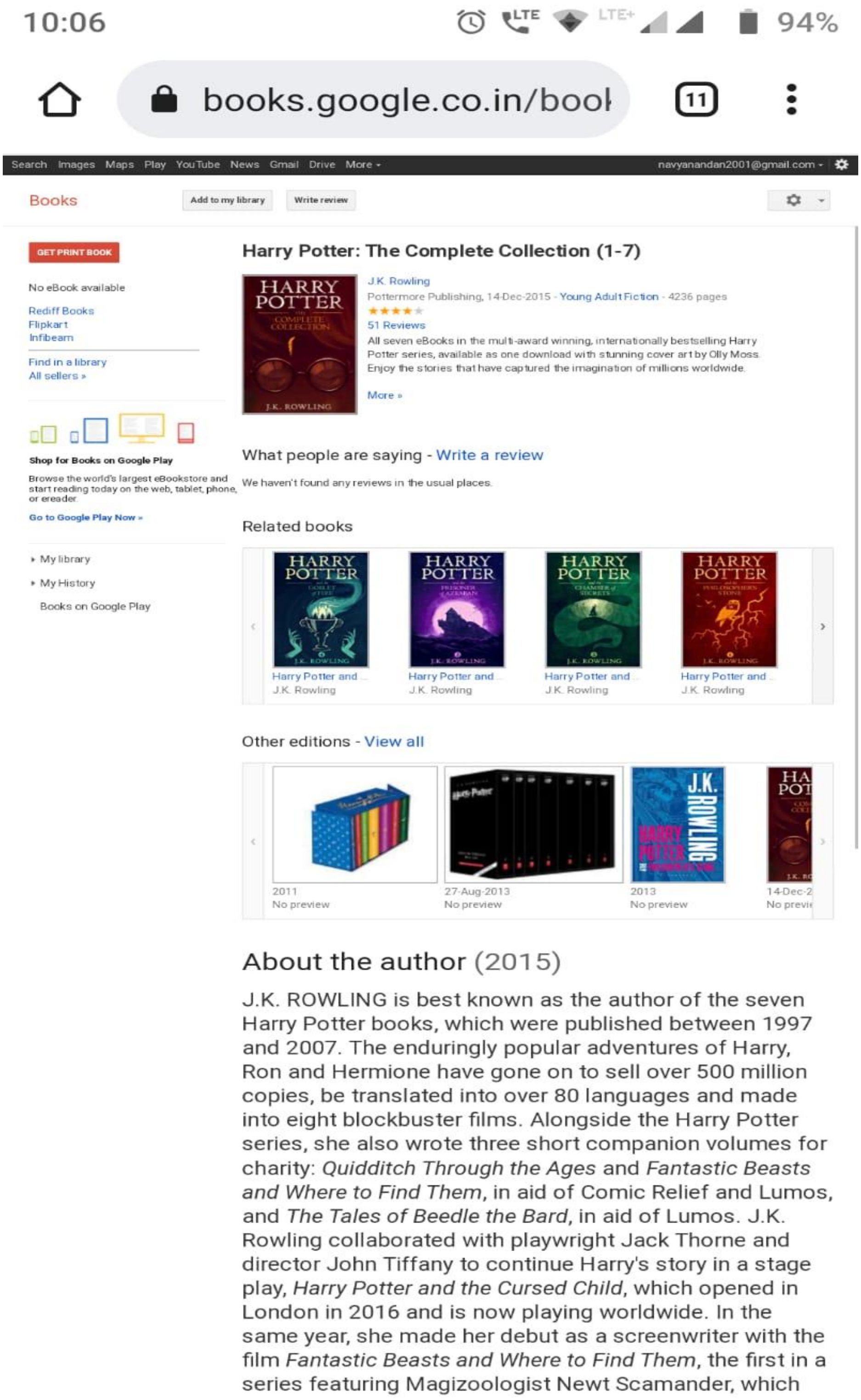
No Of Pages : 4236

Published On : 2015-12-14

Harry Potter: The Complete Collection (1-7)

All seven eBooks in the multi-award winning, internationally bestselling Harry Potter series, available as one download with stunning cover art by Olly Moss. Enjoy the stories that have captured the imagination of millions worldwide. Having now become classics of our time, the Harry Potter ebooks never fail to bring comfort and escapism to readers of all ages. With its message of hope, belonging and the enduring power of truth and love, the story of the Boy Who Lived continues to delight generations of new readers.

## 9.Required Result Page



No eBook available

Rediff Books
Flipkart
Infibeam

Find in a library
All sellers »

**Shop for Books on Google Play**

Browse the world's largest eBookstore and start reading today on the web, tablet, phone, or ereader.

Go to Google Play Now »

- My library
- My History

Books on Google Play

Harry Potter: The Complete Collection (1-7)

J.K. Rowling
Pottermore Publishing, 14-Dec-2015 - Young Adult Fiction - 4236 pages
★★★★☆
51 Reviews

All seven eBooks in the multi-award winning, internationally bestselling Harry Potter series, available as one download with stunning cover art by Olly Moss. Enjoy the stories that have captured the imagination of millions worldwide.

More »

What people are saying - Write a review

We haven't found any reviews in the usual places.

Related books

| | | | |
|---|---|---|---|
| Harry Potter and ...<br>J.K. Rowling | Harry Potter and ...<br>J.K. Rowling | Harry Potter and ...<br>J.K. Rowling | Harry Potter and ...<br>J.K. Rowling |

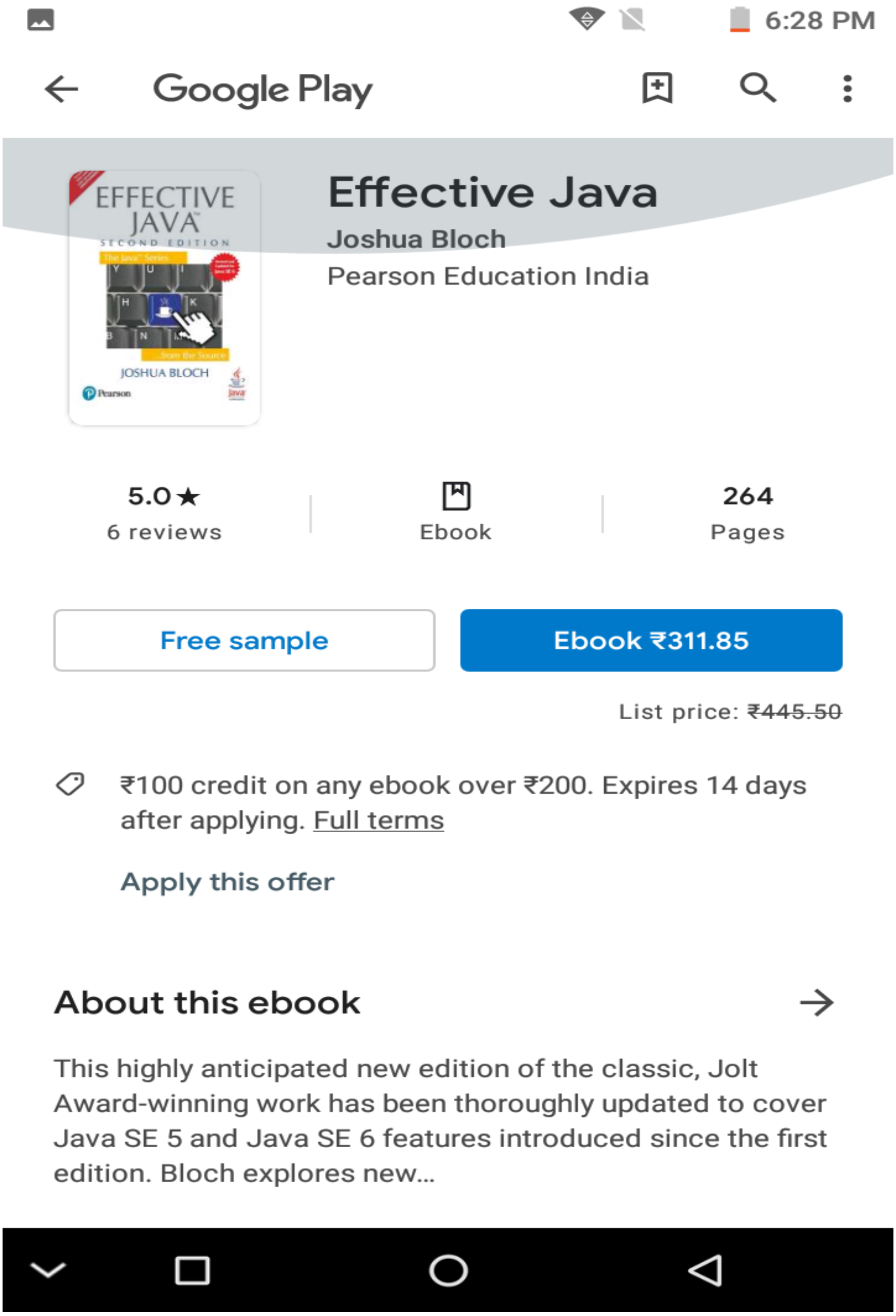
Other editions - View all

| | | | |
|---|---|---|---|
| 2011<br>No preview | 27-Aug-2013<br>No preview | 2013<br>No preview | 14-Dec-2<br>No previe |

About the author (2015)

J.K. ROWLING is best known as the author of the seven Harry Potter books, which were published between 1997 and 2007. The enduringly popular adventures of Harry, Ron and Hermione have gone on to sell over 500 million copies, be translated into over 80 languages and made into eight blockbuster films. Alongside the Harry Potter series, she also wrote three short companion volumes for charity: *Quidditch Through the Ages* and *Fantastic Beasts and Where to Find Them*, in aid of Comic Relief and Lumos, and *The Tales of Beedle the Bard*, in aid of Lumos. J.K. Rowling collaborated with playwright Jack Thorne and director John Tiffany to continue Harry's story in a stage play, *Harry Potter and the Cursed Child*, which opened in London in 2016 and is now playing worldwide. In the same year, she made her debut as a screenwriter with the film *Fantastic Beasts and Where to Find Them*, the first in a series featuring Magizoologist Newt Scamander, which

## 10.Option to buy books



6:28 PM

Google Play

Effective Java

Joshua Bloch

Pearson Education India

5.0★
6 reviews

Ebook

264
Pages

Free sample

Ebook ₹311.85

List price: ~~₹445.50~~

₹100 credit on any ebook over ₹200. Expires 14 days after applying. Full terms

Apply this offer

About this ebook

This highly anticipated new edition of the classic, Jolt Award-winning work has been thoroughly updated to cover Java SE 5 and Java SE 6 features introduced since the first edition. Bloch explores new...

# CHAPTER 8

# FUTURE ENHANCEMENT

Software development is never ending process and continues the life of the software as per the changing needs of the user from time to time. The project is no doubt has been developed keeping in mind easy modification and enhancement that may be required from time to time. The administrator of the website can be given more functionalities like looking at a specific customers profile the books that have to be recorded and also providing download options etc.

Future enhancements include uploading of book to this app where the user can access all the required books when ever need and he doesn’t have to borrow books from library. Another future enhancement where we can show stats of the books that has been borrowed or read by the user in a particular time period this would help to a great extent in customizing application. Future enhancement would also include a portal where students can request the library to get books which currently not available in the library. The mobile application should be developed for other mobile platforms to allow users who do not have Android to access the mobile application’s functionality.

# CHAPTER 9

# CONCLUSION

From a proper analysis of positive points and constraints on the component, it can be safely concluded that the product isa highly efficient GUI based component. This application is working properly and meeting to all user requirements. This component can be easily plugged in many other systems.

The user can login and access the books of his/her desire anywhere anytime. This application also helps students in better way. Students can study using this application as all the textbooks are available in this book store. They no need to carry books everywhere or go in search of books in the market which saves their time also. The users have the facility to read and also to buy the books of their choice. This application is secure and user friendly.

It is hoped that this project will help the future developers to modify and implement the system. After modifying some techniques of the programs, it will give us the best performance as our requirements. The project will be very useful for the users.

# CHAPTER 10

# BIBLIOGRAPHY

We are able to complete our project successfully only because we could find references which were helpful and precise-to-context. We used reference material available on the internet for help during the project implementation. Below is the list of references we would like to quote:

Websites:

www.geeksforgeeks.com

A free online learning platform for learning programming languages.

www.wikipedia.com

A free online platform to acquire information about any things in the world.

Code with harry

A free online video platform which helps us to easily debug the error.