VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI-590018

"A MINI PROJECT REPORT"
(Subject: Mobile Application Development)
(Subject Code: 18CSMP68)
ON

"INVENTORY MANAGEMENT SYSTEM"

Submitted in partial fulfillment for the requirements for the Award of Degree of

BACHELOR OF ENGINEERING
IN
INFORMATION SCIENCE AND ENGINEERING
BY

Mounika S	1EP20IS059
Poornima B	1EP20IS070
S P Sindhuja	1EP20IS078
Usha S	1EP20IS096

UNDER THE GUIDANCE OF

Prof. Kavana K	Prof. Vaishali Sontakke
Assistant Professor	Assistant Professor
Dept. of ISE	Dept. of ISE

EAST POINT COLLEGE OF ENGINEERING & TECHNOLOGY
Department of Information Science and Engineering
Bidarahalli, Bengaluru – 560 049

2022-2023

EAST POINT COLLEGE OF ENGINEERING & TECHNOLOGY

(Affiliated to Visvesvaraya Technological University, Belgavi)

Bangalore-560049

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the **Mini Project Work** entitled **"INVENTORY MANAGEMENT SYSTEM"** carried out by **Ms. Mounika S, USN 1EP20IS059, Ms. Poornima B, USN 1EP20IS070, Ms. S P Sindhuja, USN 1EP20IS078, Ms. Usha S, USN 1EP20IS096,** bonafide students of **East Point College of Engineering and Technology** in partial fulfillment for the award of **Bachelor of Engineering in Information Science and Engineering** of **Visvesvaraya Technological University, Belagavi** during the year **2022-23**. It is certified that all the corrections/suggestions indicated for the Internal Assessment have been incorporated in the report deposited in the department library. The project report has been approved as it satisfies the academic requirements in respect of Mobile Application Development (18CSMP68) prescribed for the award of the said degree.

GUIDE	**HOD**	**PRINCIPAL**
Prof. Kavana K	**Prof. N Kemparaju**	**Dr. Yogesh G S**
Prof. Vaishali Sontakke	EPCET	EPCET
Asst. Professor		

Examiners

Name of the Examiners **Signature with date**

1.

2.

ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. We would like to take this opportunity to thank them all.

First and foremost we would like to express our sincere regards and thanks to **Mr. Pramod Gowda** and **Mr. Rajiv Gowda,** CEOs, East Point Group of Institutions, Bangalore, for providing necessary infrastructure and creating good environment.

We express our gratitude to **Dr. Prakash S**, Senior Vice President, EPGI and **Dr. Yogesh G S,** Principal, EPCET who has always been a great source of inspiration.

We express our sincere regards and thanks to **Prof. N Kemparaju,** Professor and Head, Department of Information Science and Engineering, EPCET, Bangalore, for his encouragement and support.

We are grateful to acknowledge the guidance and encouragement given to us by **Prof. Kavana K,** Assistant Professor and **Prof. Vaishali Sontakke**, Assistant Professor, Department of Information Science and Engineering, EPCET, Bangalore, who has rendered a valuable assistance.

We also extend our thanks to the entire faculty of the Department of **Information Science and Engineering, EPCET**, Bangalore, who have encouraged us throughout the course of the Project.

Last, but not the least, we would like to thank our family and friends for their inputs to improve the Project.

MOUNIKA S	1EP20IS096
POORNIMA B	1EP20IS070
S.P.SINDHUJA	1EP20IS078
USHA S	1EP20IS096

ABSTRACT

Inventory management System app is used to track inventory levels, orders, sales and deliveries. It can also be used in the manufacturing industry to create a work order, bill of materials and other productionrelated documents. Companies use inventory management software to avoid product overstockand outages. This application makes it easier to manage large inventories as it helps in tracking it whichreduces the time required to search a particular stock. The core concept is to track the sale of items fromthe cash registers with additional features for interpreting the data. This app can also be used to store thedetails of the inventory, update the inventory based on the sale details, produce receipts for sales, generatesales and inventory reports periodically etc.

Keeping accurate material inventory is critical to ensure smooth operational activity. The academiclaboratory was facing issues with inaccurate inventory record. This study is about designing an electronicinventory control mobile application to manage the inventory and improve the stock record accuracy.The mobile application was developed using the Android system and is called E-Inventory. The mobile E-Inventory application was successfully tested by the academic staffs and engineering assistants. TheE-inventory has proven able to reduce the time to access inventory by 80% and to accurate track inventorycompared to manual

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CONCLUSION

To conclude, Inventory Management System is a simple desktop based application basically suitable for small organization. It has every basic items which are used for the small organization. Our team is successfulin making the application where we can update, insert and delete the item as per the requirement. The EInventory system was successfully developed and tested to perform the intended function for the manufacturing department's fast and accurate inventory management system. Further improvement includesadded new feature to access camera to take and store the material photo and to add chat feature to the systemfor user to leave a message to the store keeper to request material. Further, functions to import and export data for report generation should also be included. Since currently only the Android version is available, future improvement should also allow E-Inventory to be accessed using the iOS application This applicationalso provides a simple report on daily basis to know the daily sales and purchase details. This application matches for small organization where there small limited if godwoms. Through it has some limitations, our team strongly believes that the implementation of this system will surely benefit the organization.

REFERENCES

http://www.final-yearprojects.co.cc

http://www.vbforums.com/showthread.php?p=2686697

http://www.aspalliace.com

http://bro.gov.in

http://www.w3schools.com

Chapter 1

INTRODUCTION

Inventory control is very important to an organization because all the stock must be managed, controlled and tracked at anytime and anywhere. Nowadays, inventory control is even more crucial to successand sustainability. Inaccurate inventory control can lead to lost business profits. Chuang and Oliva [1] claimed that IRI (Inventory record inaccuracy) reduces a company's total profits through a study on a retailstore which has an IRI of 29% causing the company losing 10% profits. Manually managing the stock may be inaccurate, consume time and also prone to human error. Among the factors causing the inventory recordinaccuracy; (i) missing main sources of stock (ii) misplacement of stock and (iii) stock transaction errors [2]To solve these issues, inventory control requires a computerized system to better manage the inventory [3]. With the increased amount of material variations and quantity, manually tracking and counting inventory isnearly impossible. Hence, there are many types of inventory computer software available in the market. However, relying on inventory control and tracking methods solely on a desktop computer is no longer feasible. In 2017, there are almost 20 million smart phone uses in Malaysia [4]. Thus, there is a vast opportunity to develop application software using smart phones for user convenience There are 101 laboratories in the Faculty of Engineering Technology, University Technical Malaysia Melaka. These laboratories require various consumables materials for the laboratory activities. The record keeping and retrieving were done manually using a yellow form and an Excel spreadsheet. The location of the various stores also added to the issue of sharing information between the departments and tracking similar inventories. These consumable goods need proper management in order to ensure that the students' learningprocess runs smoothly at all times but still keeping the costs of keeping the inventory at a minimum. Thus, the aim of this study is to develop a system to accurately manage these consumable goods in storage. The use of the E-Inventory system has helped to improve the efficiency of the laboratory consumables materialsmanagement.

Chapter 2

LITERATURE SURVEY

Products are considered as the business resources for the organization. This includes managing the productwith appropriate way to review any time as per the requirement. Therefore it is important to have a computerbased IMS which has the ability to generate reports, maintain the balance of the stock, details about the purchase and sales in the organization. Before developing this application we came up with 2

Inventory Management System existing in the market, which helps to give the knowledge for the development of our project. These application software are only used by the large organization but so we came up with the application which can be used by the small company for the management of their stock inthe production houses. After analyzing the other inventory management system we decided to include someof common and key features that should be included in every inventory management system. So we decidedto include those things that help the small organization in a way or other.

2.1. Advantages of propsed system:

After analyzing many existing IMS we have now the obvious vision of the project to be developed. Beforewe started to build the application team had many challenges. We defined our problem statement:

- To make desktop based application of IMS for small organization.
- To make the system easily managed and can be secured.
- To cover all the areas of IMS like purchase details, sales details and stock management.

2.2. Scope of the Application:

Inventory Management System (IMS) is targeted to the small or medium organization which doesn't have many godwom or warehouses i.e. only to those organization that has single power of authority. Some of thescope are:

- Only one person is responsible in assigning the details or records .
- It is security driven.
- Godown can be added as per the requirement.

Chapter 3

SYSTEM REQUIREMENT & SPECIFICATION

The hardware and software requirements are very minimal and the software can run on most of the machineeven of the past . Here we have used the system of below specification to develop.

3.1. HARDWARE REQUIREMENTS

- PROCESSOR :Intel CORE i5
- RAM : 4GB
- HARD DISK SPACE : 1TB
- MOUSE AND KEYBOARD

3.2. SOFTWARE REQUIREMENTS

- JDK 1.8 or above.
- Android Studio SDK.
- Language: JAVA, XML

3.3. MOBILE PHONE SPECIFICATION

- Android OS 4.4 or above
- 512 MB RAM
- Screen Size 3.5 inch or about

Chapter 4

SYSTEM ANALYSIS

4.1. Background Research :

We started research by identifying the need of IMS in the organization. Initially we bounded our research to find the general reasons that emerged the needs of Inventory Management System. We useddifferent techniques to collect the data that can clearly give us the overall image of the application. The techniques we used were interview with the developers, visiting online websites that are presented as thetemplates and visiting some organization to see their IMS application. Basically the following factors forced us to develop IMS application:

- Cost and affordability
- Lack of stock management.
- Effective flow of stock transfer and management.
- Difficulty in monitoring the stock management.

4.2. Requirement Analysis:

We collected a number of requirements for project from our primitive research, website visits, and interview to the concerned personnel and their experiences regarding the concepts of its development. We have even visited some organization in Kathmandu valley and analyze its importance and try to develop the project by fulfilling all the weakness that were found in the application. We then decided tobulid same type of application with different logic flow and new language which will be suitable for thesmall organization.

4.3. IMS Requirement:

The goal for the application is to manage the inventory management function of the organization. Onceit is automated all the functions can be effectively managed and the organization can achieve the competitive advantage. Business requirement are discussed in the Scope section, with the following additional details:

- Helps to search the specific product and remaining stock.
- Details information about the product sales and purchase.
- It helps to identify the total presented inventory in the company.

Chapter 5

SYSTEM DESIGN

Process Flow Diagram or Flowchart is a diagram which uses geometric symbols and arrows to definethe relationships. It is a diagrammatic representation of the algorithm. The Process flow Diagram of our application is shown below:

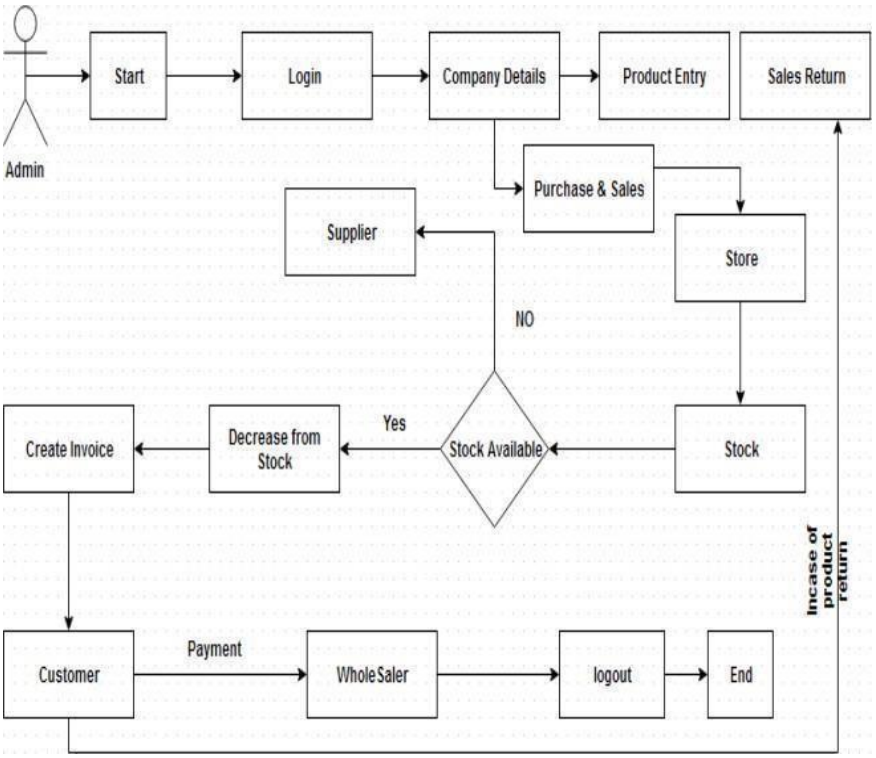

FIGURE 5.1 IMS Process flow diagram

Working in procurement means tackling responsibilities that cover a lot of different, but interrelated, tasks. One of the most important, and often most complicated, of these tasks is supply chain management—and within it, the Inventory management system (IMP). The methods you choose when recording, tracking, and managing your inventory have a huge impact on a wide array of critical business processes. From production essentials like raw materials to indirect but important items such as office supplies and marketing materials, all the items in your physical inventory have a chance to enhance or deplete your bottom line, depending on how they're managed.

5.2. Use Case Diagram

Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors and their goals. The main purpose of a use case diagram is to show what system functions are performed forwhich actors.

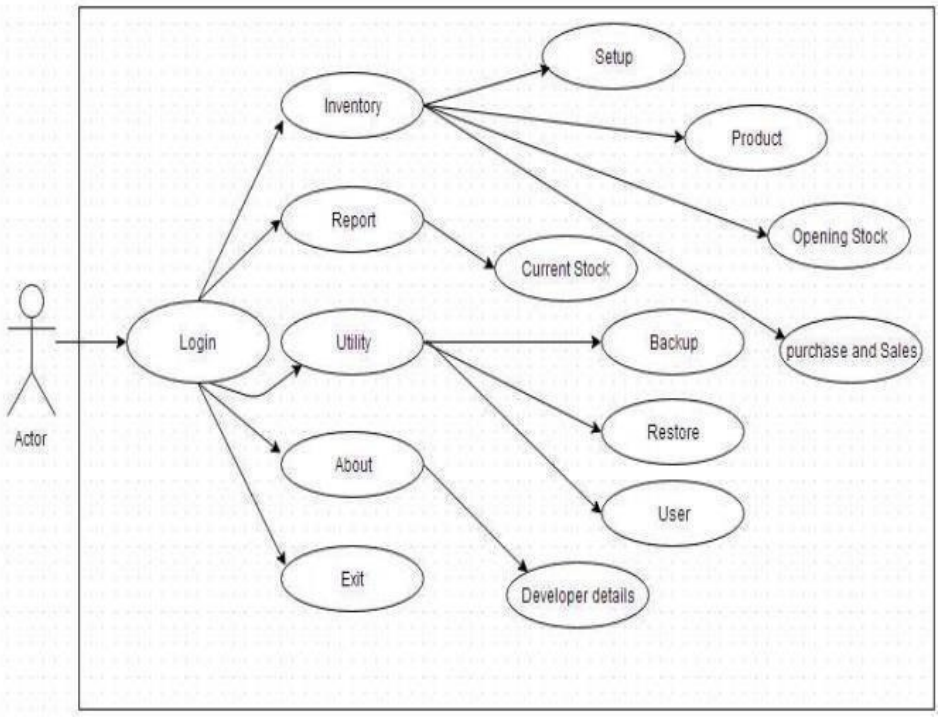

FIGURE 5.2 IMS Use Case Diagram

Inventory Management System Use Case Diagram– shows the UML illustration of use cases in the project. The use cases in the diagram represents the main processes in **Inventory Management System**. After that, it will be broken down into more specific use cases based on the processes that are part of the main use case. So, in this article you will know the Inventory Management System Use Case Diagram withits general use case and its specific use cases.

A use case diagram is a visual representation of how a user might interact with a program. Also, usecase diagram shows how the system is used in a lot of different ways by a lot of different users. Circles or ellipses are used to show the different ways people use them.

Chapter 6

IMPLEMENTATION

App's user interface is everything that the user can see and interact with. Android provides a variety of pre- built UI components such as structured layout objects and UI controls that allow us to build the graphical user interface for the app. The UI design can be done in 2 ways.

- One is using plug and play where in we directly design using the inbuilt controls such as EditText,TextView, Buttons etc..
- Second method is by directly writing XML code.

XML CODE:

```
<?xml version="1.0" encoding="utf-8"?>

<manifestxmlns:android="http://schemas.android.com/apk/res/andr

   oid"package="com.example.admin.augscan">

    <uses-permission android:name="android.permission.CAMERA" />

    <uses-feature android:name="android.hardware.camera" />

   <application android:allowBackup="true"

       android:icon="@mipmap/ic_launcher"

       android:label="WareHouse App"

       android:roundIcon="@mipmap/ic_launcher_ro

       " android:supportsRtl="true" android:theme="@style/AppTheme">

       <activity android:name=".ScanCodeActivitysearch"></activity>

       <activity android:name=".ScanCodeActivity" />

       <activity android:name=".ScanCodeActivitydel" />
```

```
    <activity  android:name=".viewWareHouseActivity"   />
    <activity android:name=".scanItemsActivity" />
     <activity android:name=".deleteItemsActivity" />
    <activity android:name=".additemActivity" />
    <activity android:name=".dashboardActivity" />
    <activity android:name=".RegisterActivity" />
    <activity android:name=".LoginActivity" />
    <activity android:name=".MainActivity" />
    <activityandroid:name=".SplashsActivity"
    android:theme="@style/AppTheme.AppCompat.NoActior">
       <intent-filter>
          <action android:name="android.intent.action.MAIN" />
         <category android:name="android.intent.category.LAUNCHER" />
       </intent-filter>
    </activity>
    <meta-data android:name="preloaded_fonts" android:resource="@array/preloaded_fonts"/>
  </application>
</manifest>
```

JAVA CODE:

```
Package com.example.admin.augscan;

import android.content.Intent;

importandroid.support.v7.app.AppCompatActvit

y;import android.os.Bundle;

import android.view.View;

importcom.google.firebase.auth.FirebaseAu

th;import

com.google.firebase.auth.FirebaseUser;

public class MainActivity extends

AppCompatActivity {private FirebaseAuth auth;

 @Override

    protected void onCreate(Bundle

        savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

    auth = FirebaseAuth.getInstance();

    FirebaseUser user = auth.getCurrentUser(); if(user !=

          null){finish();

          startActivity(new Intent(this, dashboardActivity.class));

        }
```

```
    }
    public void login (View view)
{
startActivity(new Intent(this,LoginActivity.class));

    }
public void register (View view)
    {
       startActivity(new Intent(this,RegisterActivity.class));
    }

  }
 //Register

 packagecom.example.admin.augsc

 an;import android.content.Intent;

 import android.support.annotation.NonNull;

 importandroid.support.v7.app.AppCompatActivi

 ty;import android.os.Bundle;

 importandroid.text.TextUti

 ls;import android.util.Log;

 importandroid.util.Pattern;

 importandroid.view.View;

 impandroid.widget.Button;

 import android.widget.EditText;

 importandroid.widget.ProgressB
```

```
ar;import android.widget.Toast;

importcom.google.android.gms.tasks.OnCompleteListen

er;import com.google.android.gms.tasks.Task;

import com.google.firebase.auth.AuthResult;

import com.google.firebase.auth.FirebaseAuth;

import com.google.firebase.auth.FirebaseUser;

importcom.google.firebase.auth.UserProfileChangeReqes

t;import com.google.firebase.database.FirebaseDatabase;

public class RegisterActivity extends AppCompatActivity{

private EditText editTextName, editTextEmail, editTextPassword,

editTextPhone,editTextcPassword;public Button UserRegisterBtn;

privateProgressBarprogressB

ar;privateFirebaseAutmAuth;

@Override

protected void onCreate(Bundle savedInstanceState{

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_register);

editTextName =findViewById(R.id.departmentName);

editTextEmail = findViewById(R.id.emailRegister);

editTextPassword=findViewById(R.id.passwordRegistr;

editTextcPassword=findViewById(R.id.confirmPasswd)
```

```
  ;UserRegisterBtn= findViewById(R.id.button_register);

  progressBar = findViewById(R.id.progressbar);

  progressBar.setVisibility(View.GONE);

  mAuth = FirebaseAuth.getInstance()

UserRegisterBtn.setOnClickListener(newView.OnClickListen) {

@Override
public void onClick(View v) { registerUser();
   });

       }
@Override
protected void onStart() { super.onStart();if

(mAuth.getCurrentUser() != null) {

}
       }
   private void registerUser() {

          finalStringname=editTextName.getText().toString().trim()

          ;final String email = editTextEmail.getText().toString();

          Stringpassword= editTextPassword.getText().toString().trim();

          Stringcpassword=editTextcPassword.getText().toString().trim

          ();if (email.isEmpty()) { editTextEmail.setError("It's empty");

          editTextEmail.requestFocus();

          return;

          }
```

```
    if(name.isEmpty()){

        editTextName.setError("It's Empty");

        editTextName.requestFocus();return;  }

    if(!Patterns.EMAIL_ADDRESS.matcher(email).matches()

        ) {editTextEmail.setError("Not a valid emailaddress");

        editTextEmail.requestFocus(); return;

    if (password.isEmpty()){

        editTextPassword.setError("Itsempty"

        );    editTextPassword.requestFocus();

        return;

}

    if    (password.length() < 6)    {

        editTextPassword.setError("Less

        length");

        editTextPassword.requestFocus();

        return;

    }

    if(!password.equals(cpassword)){

    editTextcPassword.setError("PasswordDonotMatch");

    editTextcPassword.requestFocus();

     return;
```

```
        }
        progressBar.setVisibility(View.VISIBLE);
        mAuth.createUserWithEmailAndPassword(email, password)
        .addOnCompleteListener(newOnCompleteListener<AuthResult>() {
                @Override
                public onComplete(@NonNull
                Task<AuthResult> task) {
                 if (task.isSuccessful()) {
           User user = new User( name,email);
);
FirebaseUser usernameinfirebase =
mAuth.getCurrentUser();String
UserID=usernameinfirebase.getEmail();
                        String resultemail = UserID.replace(".","");
                        FirebaseDatabase.getInstance().getReference("Users")
                        .child(resultemail).child("UserDetails")
                             .setValue(user).addOnCompleteListener(new
                          OnCompleteListener<Void>() {@Override  public  void
                        Task<Void> task)                                    {
                                    onComplete(@NonNull
                        progressBar.setVisibility(View.GONE);                 if
  (task.isSuccessful()) {
```

```
        Toast.makeText(RegisterActivity.this, "Registration Success",
                                Toast.LENGTH_LONG).show();startActivity(new
                                Intent(RegisterActivity.this,dashboardActivity.class));
                            } else {
                                //display a failure message
                            }
                        }
});
                    } else { progressBar.setVisibility(View.GONE);
                        Toast.makeText(RegisterActivity.this, "Registration
Failed",Toast.LENGTH_LONG).show();
                    }
                }
});
}
 }

 }

// Login package
 com.example.admin.auan;
 importandroid.app.ProgressDialog
 ;import android.content.Intent;
 importandroid.support.annotation.
 NonNull;
```

```
importandroid.support.v7.app.AppCompatActivi
ty;import android.os.Bundle;
import android.view.View;
importandroid.widget.Button;
importandroid.widget.EditText
importandroid.widget.Progresar;
importandroid.widget.TextView
import android.widget.Toast;
importcom.google.android.gms.tasks.OnCompleteListen
er;import com.google.android.gms.tasks.Task;
importcom.google.firebase.auth.AuthResult
importcom.google.firebase.auth.Firebasth;
importcom.google.firebase.auth.Firebaseer;
public class LoginActivity extends AppCompatActivity { private EditText Email; private
EditTextPassword;
private Button Login;
private TextView passwordreset;
private EditText
passwordresetemail;private
ProgressBar progressBar; private
FirebaseAuth auth;
private ProgressDialog processDialog;
```

```
@Override protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_login);
    Email = (EditText) findViewById(R.id.emailSignIn);
    Password = (EditText)
    findViewById(R.id.password); Login = (Button)
    findViewById(R.id.Login); passwordreset =
    findViewById(R.id.forgotpassword);
    passwordresetemail =
    findViewById(R.id.emailSignIn);
    progressBar = (ProgressBar)
    findViewById(R.id.progressbars);
    progressBar.setVisibility(View.GONE);auth =
    FirebaseAuth.getInstance();processDialog = new
    ProgressDialog(this);
    Login.setOnClickListener(new View.OnClickListener() {
        @Override
        publicvoidonClick(View,view){
        validate(Email.getText().toString(),
        Password.getText().toString());
        }
```

```
});
        @Override public
        voidonClick(View
        v) {
        resetpasword();
        }
});
    }
    public void resetpasword(){
        final String resetemail =
        passwordresetemail.getText().toString();if
        (resetemail.isEmpty()) { passwordresetemail.setError("It's
        empty"); passwordresetemail.requestFocus();return;
        }
        progressBar.setVisibility(View.VISIBLE); auth.sendPasswordResetEmail(resetemail)
              .addOnCompleteListener(new
             OnCompleteListener<Void>() {@Override public  void
             onComplete(
               @NonNull Task<Void> task) {  if (task.isSuccessful()) {
                    Toast.makeText(LoginActivity.this, "We have sent you instructions to reset
yourpassword!", Toast.LENGTH_SHORT).show();
                  } else {
                    Toast.makeText(LoginActivity.this,"Failed  to  send  reset  email!",  Toast.
LENGTH_SHORT) .show();
```

```
                }
progressBar.setVisibility(View.GONE);
            }
});
  Public void validate(String userEmail, String userPassword){ process Dialog

        .setMessage(".....Please Wait ");

  processDialog.show();

    auth.signInWithEmailAndPassword(userEmail,

    userPassword).addOnCompleteListener(newOnCompleteListener<AuthResult>() {

      @Override

      public void onComplete(@NonNull Task<AuthResult>

      task) {if(task.isSuccessful()){

            processDialog.dismiss();

            Toast.makeText(LoginActivity.this, "Login Successful", Toast.LENGTH_SHORT).show();

         startActivity(new Intent(LoginActivity.this, dashboardActivity.class)); }

else{
            Toast.makeText(LoginActivity.this,"LoginFailed",Toast.LENGTH_SHORT).show();

            processDialo .dismiss(); }

      }
});
  }

 }
```

Chapter 7

TESTING

The purpose of software testing is to access or evaluate the capabilities or attributes of a software program's ability to adequately meet the applicable standards and application need. Testing does not ensure quality and the purpose of testing is not to find bugs.The primary objective if testing includes:

- To identifying defects in the application.
- The most important role of testing is simply to provide information.
- to check the proper working of the application while inserting updating and deleting.

7.1. Type of Testing

We have used one type of testing to ensure the error free features of our software application:

7.2. Units Test

This type of testing is the testing of individual software components. It is typically done by the programmer and not by the testers. It requires details information and knowledge about the internal program design and code to perform this. During unit testing, we carried out various testing task such asthe reflection of the unit data on database and its interface. Various types of bugs associated with the component were identified and fixed. We use various functional keys to test our software.In our software unit testing is concerned with the stock units, opening stock units and product units validation as well as the validation of product units.

Table 7.2 Login Test Case

Test Case	Login page
Test Name	Test case to enter the Email and Password of Login
Test Description	To test whether the report is generated or not
Input	Press the LOGIN button
Expected Output	The result should be saved and Login to next page
Actual Output	The result is saved and Login to next page
Test Result	Success

Chapter 8

SNAPSHOTS

FIGURE.8.1.REGISTER PAGE

FIGURE.8.2. LOGIN PAGE

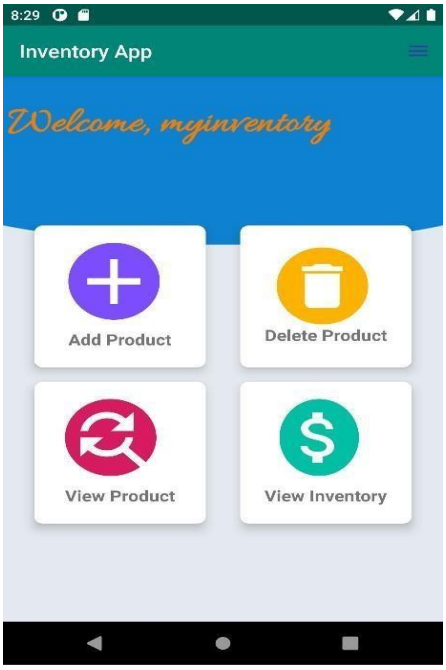

FIGURE.8.3. WELCOME PAGE

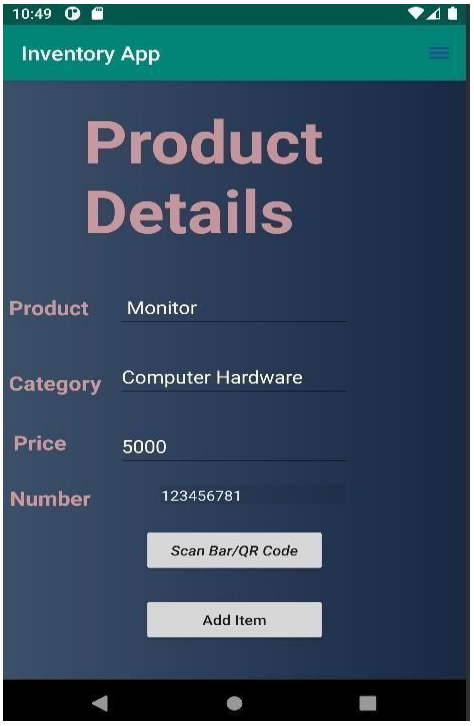

FIGURE.8.4. ADDPRODUCT ITEM

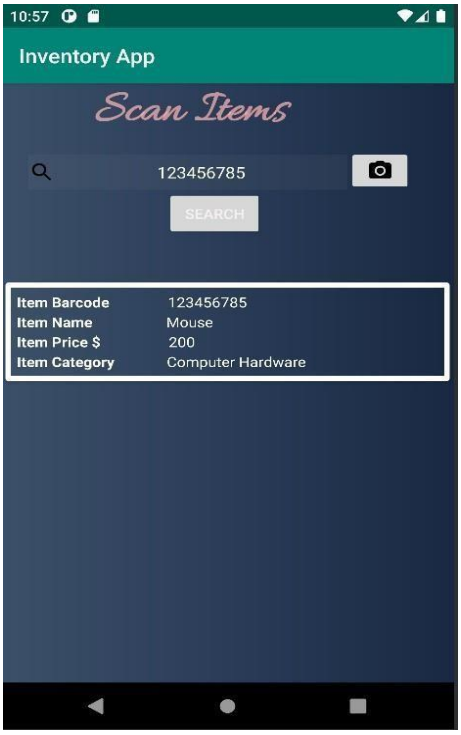

FIGURE.8.5. SEARCH ITEM

FIGURE.8.6. VIEW INVENTORY

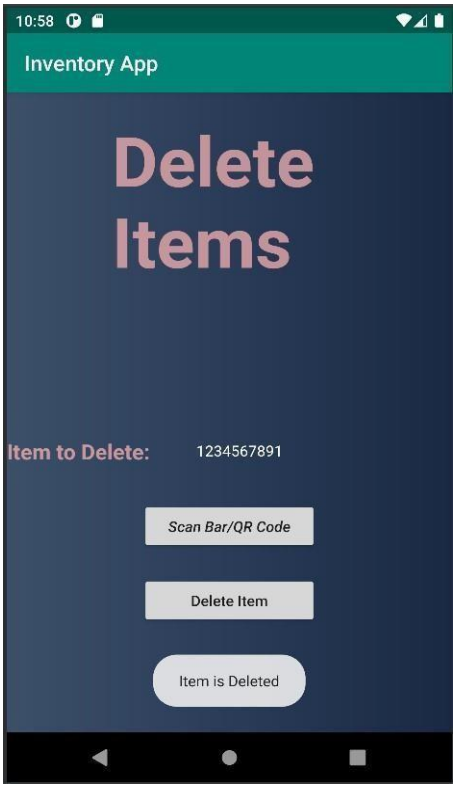

FIGURE.8.7. DELETE ITEM

Chapter 9

PERFORMANCE ANALYSIS

Software performance analysis looks at how a specific program is performing on a daily basis and chronicleswhat slows down performance and causes errors now and what could pose a problem into the future.

Performance issues aren't always built into software in a way that can easily be spotted through the QA process. Instead, it is something that can emerge over time after the project has been deployed. Software performance analysis keeps your team honest: it requires developers to continually test what they are doingand IT teams to monitor the code as version are updated, more code is added, other applications interact withit, or when there are changes in hosting. While many businesses will perform software performance analysisat random times, that isn't enough. Performance is something that needs to be constantly monitored for problems because they aren't always apparent and catching something early (perhaps even before the end user is impacted) can save a lot of time and strife.

- Software Performance Evaluations & Analysis Eliminate Rework When designing new applicationsor even making changes to existing ones, there are bound to be some errors or things that go wrong.This can impact software performance immediately or it can impact it as a slow leak over time. Eitherway, it should be caught as soon as possible. While those immediate problems may seem like the most hazardous, they don't have to be. The slower, smaller problems that compound over time can cause eventual crashes or a lapse In security that can bring down your entire portfolio. When you think of all that could be stopped by a simple mistake - web applications, mobile applications, internaloperations, and so much else, it can be disconcerting. The software performance needs to meet the performance requirements of today's world, regardless of when they were created. Sometimes, tryingto keep up can fall by the wayside and reveal many, many problems. When you let something go toolong, the answer may be losing all of that work and trying to develop something completely new: losing everything that you have done already. To avoid this, software performance analysis needs tobe automated into every stage of the process. The IT teams and developers need to get their work right the first time and eliminate the need for a lot for rework. By adding software performance analysis into your existing culture, you will see changes that result in more productivity and even more employee satisfaction.

- Software Performance Analysis Helps Businesses Achieve Optimization Of course, IT teams are expected to perform at a certain level and take a certain amount of pride in their work. Still, we all make mistakes and when work becomes rote or repetitive as coding can be, mistakes are bound to happen. Everyone must take steps to ensure that they are working toward goals with efficiency: the work needs to be good and it needs to get done quickly. With software performance analysis, they will have some back up in checking their work. It is important to note that this is far more than simplistic testing. Instead, it is complete, comprehensive monitoring of code throughout the lifespanof the application.
- Incorporate Software Performance Analysis into Your System At the end of the day, software performance analysis will help to simulate how a system will perform today, tomorrow, three weeksfrom now, and into the next year. It checks for potential problems that may emerge when you test it against traffic, load conditions, and business requirements. The goal is to reduce technical debt whileat the same time providing a better business value.