PROJECT REPORT :-

SIMPLE INVENTORY MANAGEMENT

SUBMITTED BY :-

MR. Dharmaraj Sharanappa Wadeyar

PRN :- 2018320325

Year – 2020-2021

BSC-CS (SEMESTER-V)

Examination University of Mumbai

College :-

B.N Bandodkar College of Science, Thane – 400601

ACKNOWLEDGMENT

In the accomplishment of this project successfully it required a lot of guidance from experienced people and also their heart pledged support. I express my sincere thanks to the college project guide, my SIR Mr. Abhishek Vartak, whose valuable guidance has been the ones that helped me work this project and make it full proof of success. His suggestions and his instructions have served as the major contributor towards the completion of the project.

DECLARATION

I Mr.DHARMARAJ SHARANAPPA WADEYAR the student of **B.N Bandodkar College of Science**, TYBSC (Computer Science) hereby declare that, I have completed the project on **"SIMPLE INVENTORY MANAGEMENT SYSTEM".**

The information submitted is true and original to the best of knowledge.

NAME AND SIGNATURE OF STUDENT

SYNOPSIS OF THE PROJECT

TITLE OF THE PROJECT : SIMPLE INVENTORY MANAGEMENT SYSTEM

1.About the problem: Earlier, the shopkeeper has to write on hardcopies. Writing and managing on hardcopies is very time consuming and tedious task.

2.The primary reason to choose this particular topic : Due to pandemic situation and also the shopkeeper uses old management system i.e writing on papers. So making a project on Inventory Management saves a lot of time and it also helps shopkeepers to work digitally rather than writing on papers.

3.The main objective of the project (a clear picture of the project):

Describing the main objective of the project , in earlier days shopkeeper were used to write on hardcopies and it was difficult to store those papers. Also there were loss of hardcopies due to which the situation became more difficult to manage those copies.

4.Working Methodology (the summary of the project must also be incorporated) :

A) The application contains 6 major fields and 1 Add button and a table to display Product name , Price , Quantity and Total.

B) The 6 major fields are classified as follows :-

Product Name – The name of the specified product.

Price – The price of the given product.

Quantity (Qty) – Quantities to be listed in numbers i.e how much we want to buy the product.

Subtotal – The subtotal is the total of the all product prices listed.

Payment - The final payment of the products to be given by customer.

Balance – The balance amount shows the remaining amount to be given by shopkeeper.

C) The ADD Button is basically adding all the products in table with the Product name , Price , Quantity and Total.

D) The Table :-

The table consist of product name , price , quantity and total.

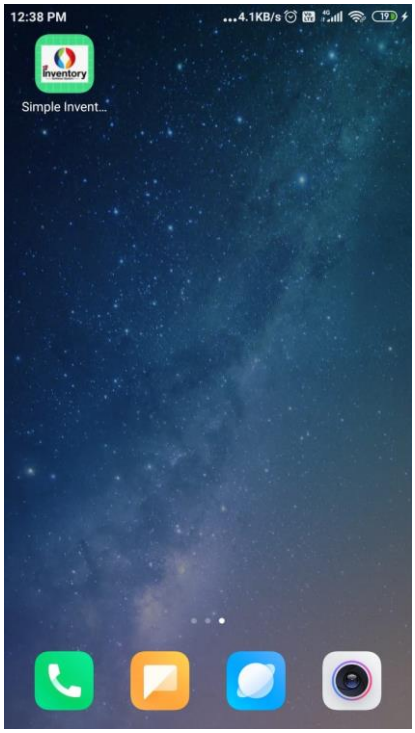
The product name displays the product name.

The price displays the price of the given product.

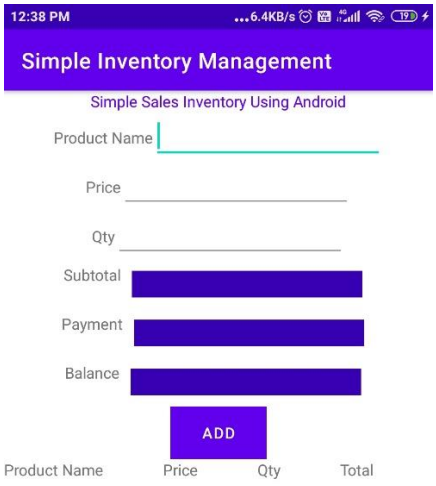
The quantity displays the amount (in numbers) of the given product.

The total displays the each total of the individual product.

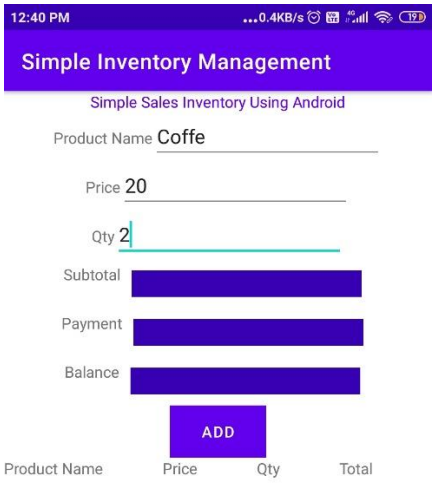
E) THE APP LOGO:-



F) When we open the application the interface appears as follows :-



G) We have to enter the details of the product name , price and quantity :-



H) On clicking add button the following product list gets added to the table :-

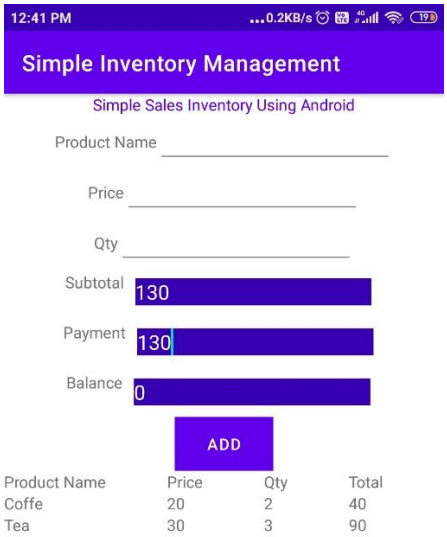
I) Also we can enter details of the other product as follows :-

J) Adding details of the other product :-

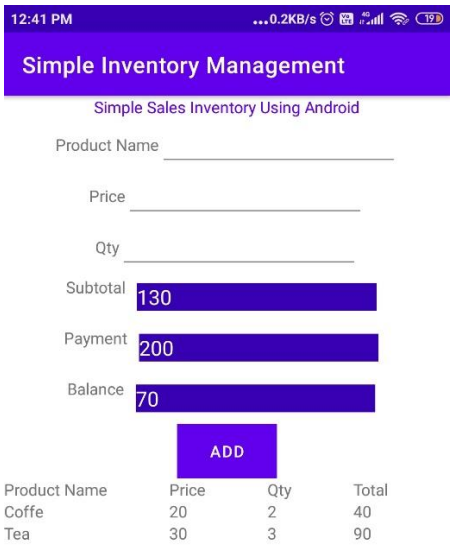
K) Now calculating the subtotal , it appears as follows:-

L) Payment :-

When the user has fixed amount equal to subtotal the balance returns the value as 0.



When the user has greater amount that of subtotal it shows the calculated amount :-



5.Scope of the project :-

This project is helpful for all shopkeepers and by using this application it reduces manual entering of data in hardcopies and there will be change in the existing system.

6.Details about the hardware and software to be used :-

A) Hardware

*** Minimum RAM 1 GB Or above**

*** Storage Maximum 10GB**

B) Softwares

***Android Studio**

*** Java programming language**

7.Listing out the testing technologies:-

Unit testing

8.Limitations of the system proposed :-

No update and delete options.

This app can run only on ice cream sandwhich versions and above.

TABLE OF CONTENTS

1 . INTRODUCTION

The shop is located at NEAR Raghukul society Thane (West) Opposite to Sidd tower road Thane(E)-400603. The owner of the shop is MR.Mahavir Patil. The aim of the shop is used to manage records and transactions manually.

2.OBJECTIVE AND SCOPE OF THE PROJECT

The main objective of the project is that in earlier system shopkeepers used to write on hardcopies and started to maintain records manually and it was tedious to store those records. The old management system consisted of manual managing of records which also returned to loss of hardcopies due to poor management of copies.

3.EXISTING SYSTEM

The existing system is somewhat like managing of records and doing transactions on hardcopies and storing it manually by writing on papers.

There arise some problems such as storing records and also there was less security in holding records in this type of management system. Sometimes there is a risk of loosing records as its written on papers by manually method.

4.PROPOSED SYSTEM

The proposed system contains the following features:-

Managing of products significantly and adding of products efficiently.

Includes Add button to add products list to table.

Displays Total in which each product total is shown.

Displays Subtotal which is total of whole amount of products i.e whole total.

5.WATERFALL MODEL AND DIAGRAM

This project uses waterfall model.Waterfall model is also known as linear sequential life cycle Model.Waterfall model is used for small projects. In this model each phase is completed successfully before the next phase is started.

Waterfall model is based on following 6 approaches namely Requirement analysis , System Design , Implementation , Testing , Deployment and Maintenance.

Requirement Gathering and analysis's – The requirements are done at this stage and it is documented properly. All requirements are captured in this phase.

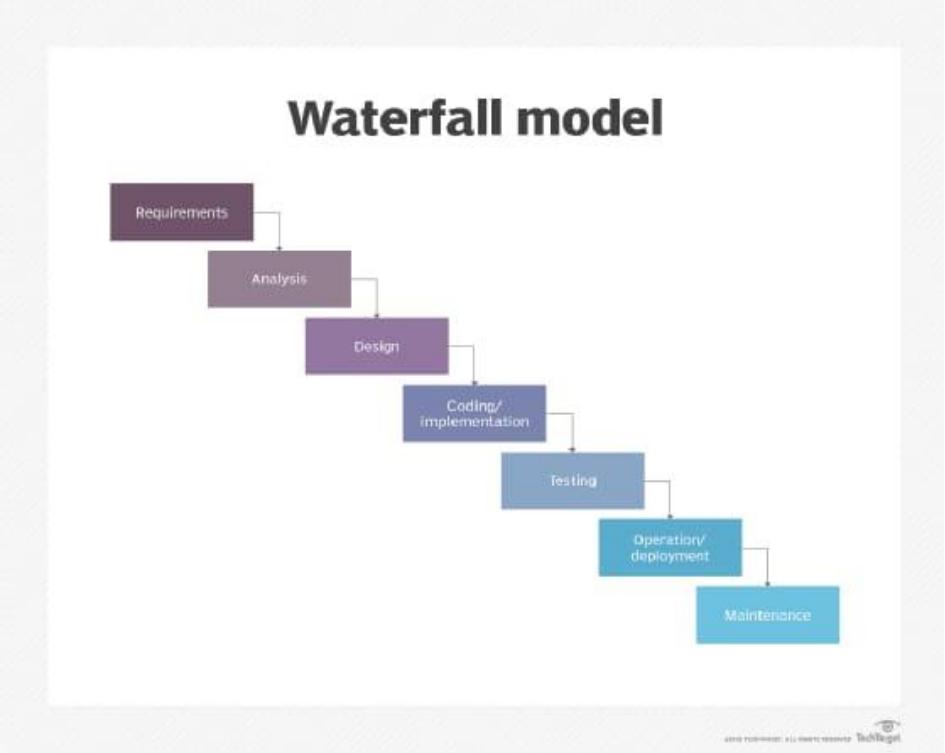
System Design – The system design is prepared after studying Requirement phase and this system design helps in specifying system and hardware components and it also defines overall system architecture.

Implementation – Each unit is developed and tested for its functionality which is refereed as unit testing.

Integration and Testing – All the units developed in the unit testing are integrated here by combining all units together. The unit also refers to small code or program and integration testing refers to combining of all small codes and testibg them. Entire system is also tested to avoid any faults or failures.

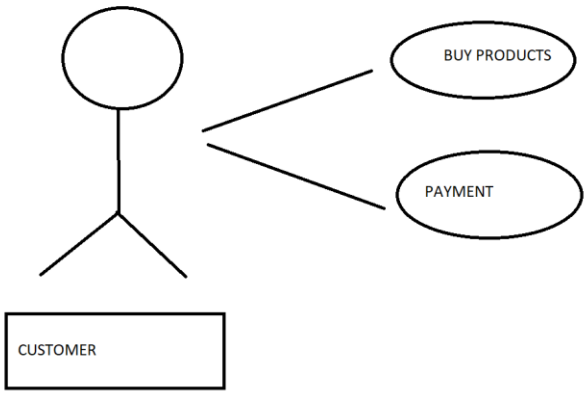
Deployment of system – When all testing are completed the product is finally deployed or released in the market.

Maintenance – When some issues are arised maintenance is done so that the product is free of failures and defects. These changes are efficiently done in customer environment.

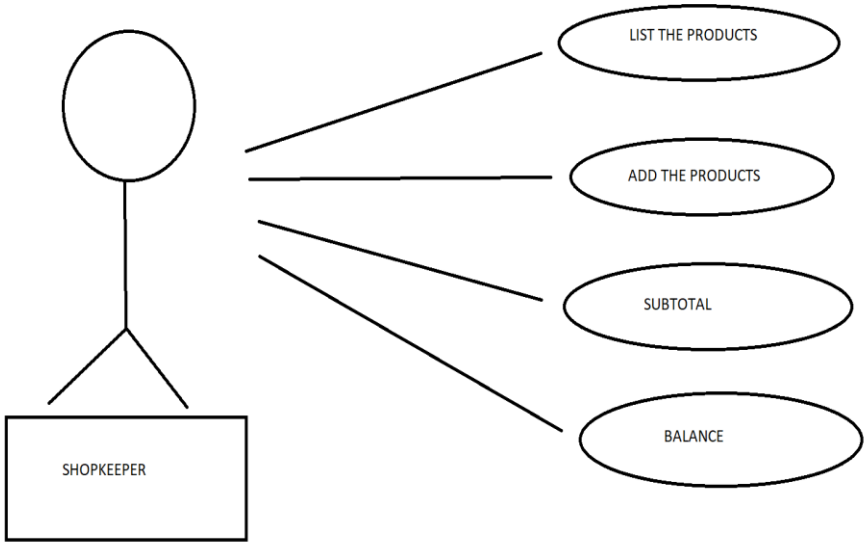

Waterfall model
Requirements
Analysis
Design
Coding/ implementation
Testing
Operation/ deployment
Maintenance

6.USE CASE DIAGRAM

A) ER DIAGRAM FOR CUSTOMER



B) ER DIAGRAM FOR SHOPKEEPER



7. SOURCE CODES

A) *SOURCE CODE FOR ACTIVITY_MAIN.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:orientation="vertical"
   android:gravity="center"
   tools:context=".MainActivity">


  <LinearLayout
     android:layout_width="wrap_content"
     android:layout_height="wrap_content"
     android:orientation="horizontal"
     >

     <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Simple Sales Inventory Using Android"
        android:textColor="@color/design_default_color_primary"
        android:gravity="center" />

  </LinearLayout>


   <LinearLayout
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:orientation="horizontal">
      <TextView
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Product Name" />
    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/ed1"
        android:ems="10" />
</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Price" />
    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/ed2"
        android:ems="10" />
</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Qty" />
    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/ed3"
```

```
        android:ems="10" />
    </LinearLayout>


    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Subtotal" />
        <EditText
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_margin="10sp"

android:background="@color/design_default_color_primary_dark"
            android:id="@+id/txtsub"
            android:textColor="#fff"
            android:enabled="false"
            android:ems="10" />
    </LinearLayout>


    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Payment" />
        <EditText
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_margin="10sp"
```

```
android:background="@color/design_default_color_primary_dark"
        android:id="@+id/txtpay"
        android:textColor="#fff"

        android:ems="10" />
    </LinearLayout>



    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Balance" />
        <EditText
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_margin="10sp"

android:background="@color/design_default_color_primary_dark"
            android:id="@+id/txtbal"
            android:textColor="#fff"
            android:enabled="false"
            android:ems="10" />
    </LinearLayout>



    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <Button
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/btn1"
        android:text="Add"
        android:background="@color/design_default_color_primary"

        />
</LinearLayout>



<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical">

    <TableLayout
        android:layout_width="match_parent"
        android:layout_height="360dp"
        android:id="@+id/tb1"
        android:stretchColumns="*">


       <TableRow android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:id="@+id/tbrow1">

          <TextView
             android:id="@+id/t1"
             android:layout_width="wrap_content"
             android:layout_height="wrap_content"
             android:text="Product Name"
             />

          <TextView
             android:id="@+id/t2"
             android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="Price"
            />

        <TextView
            android:id="@+id/t3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Qty"
            />

        <TextView
            android:id="@+id/t4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Total"
            />
    </TableRow>
  </TableLayout>
</LinearLayout>
```

B) ***SOURCE CODE FOR MAIN ACTIVITY.JAVA***

```
package com.example.simpleinventory;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.text.Editable;
import android.text.TextWatcher;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TableLayout;
import android.widget.TableRow;
import android.widget.TextView;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {


   private ArrayList<String> data = new ArrayList<String>();
   private ArrayList<String> data1 = new ArrayList<String>();
   private ArrayList<String> data2 = new ArrayList<String>();
   private ArrayList<String> data3 = new ArrayList<String>();

   private TableLayout table;

   EditText ed1,ed2,ed3,ed4,ed5,ed6;
   Button b1;

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

      ed1 = findViewById(R.id.ed1);
```

```
        ed2 = findViewById(R.id.ed2);
        ed3 = findViewById(R.id.ed3);

        ed4 = findViewById(R.id.txtsub);
        ed5 = findViewById(R.id.txtpay);
        ed6 = findViewById(R.id.txtbal);

        b1 = findViewById(R.id.btn1);




        ed5.addTextChangedListener(new TextWatcher() {
            @Override
            public void beforeTextChanged(CharSequence s, int start, int
count, int after) {

            }

            @Override
            public void onTextChanged(CharSequence s, int start, int
before, int count) {

            }

            @Override
            public void afterTextChanged(Editable s) {

                int subtotal = Integer.parseInt(ed4.getText().toString());
                int pay = Integer.parseInt(ed5.getText().toString());
                int bal = pay - subtotal;
                ed6.setText(String.valueOf(bal));

            }
        });

        b1.setOnClickListener(new View.OnClickListener() {
```

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



}


public void add() {
    int tot;

    String prodname = ed1.getText().toString();
    int price = Integer.parseInt(ed2.getText().toString());
    int qty = Integer.parseInt(ed3.getText().toString());
    tot = price * qty;

    data.add(prodname);
    data1.add(String.valueOf(price));
    data2.add(String.valueOf(qty));
    data3.add(String.valueOf(tot));

    TableLayout table = (TableLayout) findViewById(R.id.tb1);

    TableRow row = new TableRow(this);
    TextView t1 = new TextView(this);
    TextView t2 = new TextView(this);
    TextView t3 = new TextView(this);
    TextView t4 = new TextView(this);

    String total;

    int sum = 0;

    for (int i = 0; i < data.size(); i++)
```

```
        {
           String pname = data.get(i);
           String prc = data1.get(i);
           String qtyy = data2.get(i);
           total = data3.get(i);

           t1.setText(pname);
           t2.setText(prc);
           t3.setText(qtyy);
           t4.setText(total);

           sum = sum + Integer.parseInt(data3.get(i).toString());
        }

        row.addView(t1);
        row.addView(t2);
        row.addView(t3);
        row.addView(t4);
        table.addView(row);


ed4.setText(String.valueOf(sum));
ed1.setText("");
ed2.setText("");
ed3.setText("");
ed1.requestFocus();
    }
}
```

8.TOOLS

A) ANDROID STUDIO

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development.

It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020. It is a replacement for the Eclipse Android Development Tools as the primary IDE for native Android application development.

- IT SUPPORTS FOLLOWING FEATURES :-

1. Instant App Run

2. Visual Layout Editor

3. Fast Emulator

4. Intelligence Code Editor

5. Help to Connect with Firebase

6. Support KOTLIN

B) JAVA PROGRAMMING LANGUAGE

Java is a high-level programming language developed by Sun Microsystems. It was originally designed for developing programs for set-top boxes and handheld devices, but later became a popular choice for creating web applications. The Java syntax is similar to C++, but is strictly an object-oriented programming language.

9.REFERENCES

1.https://www.google.com

2. https://www.youtube.com

3. https://stackoverflow.com