

A
REPORT
ON
GST BILL GENERATOR

SUBMITTED BY
HARSH N. SHAH
UDIT K. PAREKH

GUIDED BY
Prof. BHAVESH K. CHHATRALA
Computer Engineering Department
Darshan Institute of Engineering & Technology
For Diploma Studies,
Rajkot.



2021-22
DEPARTMENT OF COMPUTER ENGINEERING
DARSHAN INSTITUTE OF ENGINEERING & TECHNOLOGY
FOR DIPLOMA STUDIES
RAJKOT-MORBI HIGHWAY, RAJKOT (GUJARAT).

A
REPORT
ON
GST BILL GENERATOR

SUBMITTED TO
GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD.
IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARDS OF
DIPLOMA IN COMPUTER ENGINEERING
SUBMITTED BY

ENROLLMENT NO

- 1) 196620307059
- 2) 196620307043

NAME OF STUDENT

HARSH N. SHAH
UDIT K. PAREKH

GUIDED BY

Prof. BHAVESH K. CHHATRALA
Computer Engineering Department
Darshan Institute of Engineering & Technology
For Diploma Studies,
Rajkot.



2021-22
DEPARTMENT OF COMPUTER ENGINEERING
DARSHAN INSTITUTE OF ENGINEERING & TECHNOLOGY
FOR DIPLOMA STUDIES
RAJKOT-MORBI HIGHWAY, RAJKOT (GUJARAT).



DARSHAN
INSTITUTE OF ENGINEERING & TECHNOLOGY
FOR DIPLOMA STUDIES

Certificate

This is to certify that,

Mr. HARSH N. SHAH having Enrollment No: **196620307043** has completed **Part-II UDP** Project work for **Semester VI** having title **GST BILL GENERATOR**, in a group of **two** persons under the guidance of the Faculty Guide **Prof. BHAVESH K. CHHATRALA**.

Guide

Prof. BHAVESH K. CHHATRALA
Computer Engg. Dept.
Darshan Institute of Engg. & Tech.
For Diploma Studies,
Rajkot.

Head of Department

Prof. CHINTAN N. KANANI
Computer Engg. Dept.
Darshan Institute of Engg. & Tech.
For Diploma Studies,
Rajkot.

2021-22

DEPARTMENT OF COMPUTER ENGINEERING
DARSHAN INSTITUTE OF ENGINEERING & TECHNOLOGY
FOR DIPLOMA STUDIES
RAJKOT-MORBI HIGHWAY, RAJKOT (GUJARAT).



DARSHAN
INSTITUTE OF ENGINEERING & TECHNOLOGY
FOR DIPLOMA STUDIES

Certificate

This is to certify that,

Mr. UDIT K. PAREKH having Enrollment No: **196620307043** has completed **Part-II UDP** Project work for **Semester VI** having title **GST BILL GENERATOR**, in a group of **two** persons under the guidance of the Faculty Guide **Prof. BHAVESH K. CHHATRALA**.

Guide

Prof. BHAVESH K. CHHATRALA
Computer Engg. Dept.
Darshan Institute of Engg. & Tech.
For Diploma Studies,
Rajkot.

Head of Department

Prof. CHINTAN N. KANANI
Computer Engg. Dept.
Darshan Institute of Engg. & Tech.
For Diploma Studies,
Rajkot.

2021-22

DEPARTMENT OF COMPUTER ENGINEERING
DARSHAN INSTITUTE OF ENGINEERING & TECHNOLOGY
FOR DIPLOMA STUDIES
RAJKOT-MORBI HIGHWAY, RAJKOT (GUJARAT).



DARSHAN
INSTITUTE OF ENGINEERING & TECHNOLOGY
FOR DIPLOMA STUDIES

EXAMINER'S CERTIFICATE OF APPROVAL

This is to certify that draft report entitled **GST BILL GENERATOR** submitted by **HARSH N. SHAH (196620307059)** and **UDIT K. PAREKH (196620307043)** in partial fulfillment for the award of the diploma in **Computer Engineering** of the Gujarat Technological University- Ahmedabad is hereby approved.

Internal Examiners

External Examiners

2021-22
DEPARTMENT OF COMPUTER ENGINEERING
DARSHAN INSTITUTE OF ENGINEERING & TECHNOLOGY
FOR DIPLOMA STUDIES
RAJKOT-MORBI HIGHWAY, RAJKOT (GUJARAT).

ACKNOWLEDGEMENT

We would like to express special thanks to our guide Prof. “BHAVESH K. CHHATRALA” and our Head of the Department Prof. “CHINTAN N. KANANI” for their guidance and support to complete our project. They helped us in sorting out errors raised during the project development. They also suggested some changes like adding or removing certain features and making appropriate changes according to the main objective of the project. Last but not least, we would also like to thank our friends who have supported and helped us in our project.

INDEX

Chapter No.	Sub Chapter No.	Content	Page No.
		Abstract	
1		Introduction	1
	1.1	Problem Summary.....	1
2		Planning	2
	2.1	Model Description.....	5
	2.2	Risk Management.....	5
3		Detail Description	7
4		Diagrams	9
	4.1	Class Diagram.....	9
	4.2	Sequence Diagram.....	10
	4.3	Collaboration Diagram.....	11
	4.4	State Diagram.....	12
	4.5	Activity Diagram.....	13
	4.6	Use Case Diagram.....	14
	4.7	Data Flow Diagram (Level 0).....	15
		Data Flow Diagram (Level 1).....	16
5		Data Dictionary	17
	5.1	E-R Diagram.....	19
6		Screen Shots	20
7		Conclusion	29
8		Future Enhancement	30

ABSTRACT

When we want to generate the bill for the purchased products, it takes lots of time to calculate bill amount and there may chances of mistake in calculation. In case of mistake in calculate, it leads to wastage of paper and time. So, we have developed Android Application **“GST Bill Generator”** will generate the bill in PDF format very easily. In application we have to just add customer details first time, add list of products purchased by the customer and it will generate a bill quickly. This application has functionalities to save generated bill for future use, user can view previously purchased product details of the customer etc. So, it saves our precious time and environment.

1. INTRODUCTION

1.1 PROBLEM SUMMARY

1.1.1 Problem Identification

When we want to generate the bill for the purchased products, it takes lots of time to calculate bill amount and there may chances of mistake in calculation. In case of mistake in calculate, it leads to wastage of paper and time. It is also difficult to maintain hard copy of the bills for user/shopkeeper.

1.1.2 Problem Solution

The solution for above mentioned problem, we have developed Android Application “**GST Bill Generator**” will generate the bill in PDF format very easily. In application we have to just add customer details first time, add list of products purchased by the customer and it will generate a bill. This application has functionalities to save generated bill for future use, user can view previously purchased product details of the customer etc. So, it saves our precious time, environment and cost effectively.

2. PLANNING

2.1 MODEL DESCRIPTION

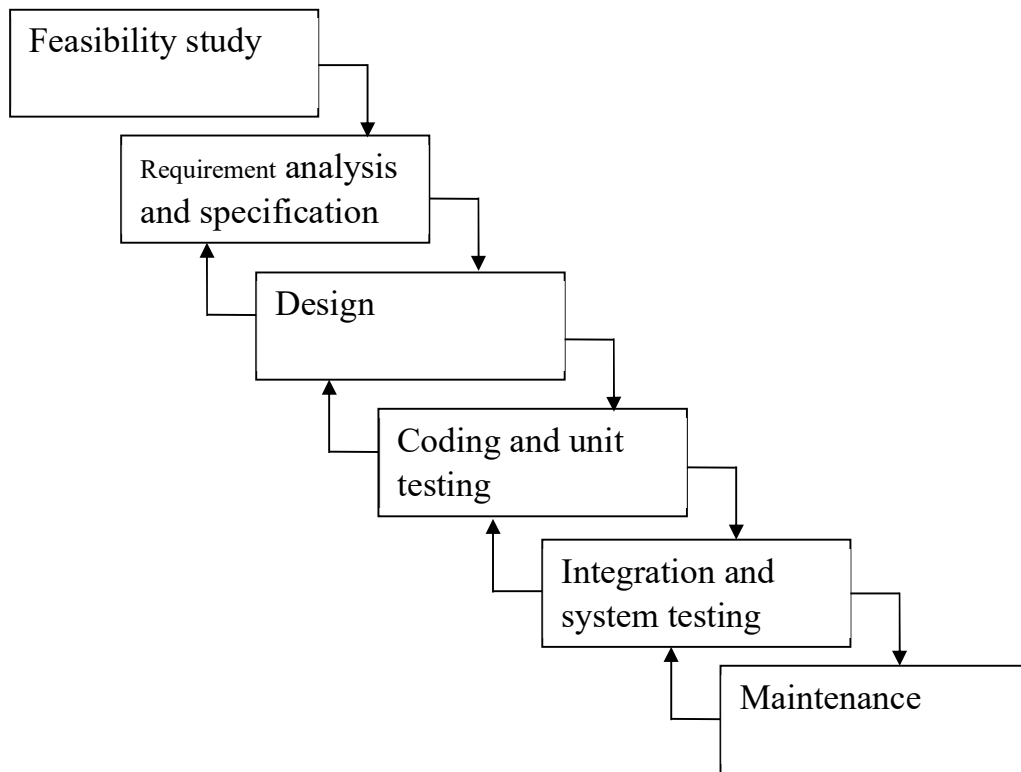


Fig. Iterative Waterfall Model

- In our project we are using iterative waterfall model.
- It is not possible to strictly follow the classical waterfall model.
- Making necessary changes to the classical waterfall model so that it becomes applicable to practical software development projects.
- The main change to the classical waterfall model is in the form of providing feedback paths from every phase to its preceding phases as shown in figure.
- The feedback paths allow for correction of the errors committed during a phase as and when these are detected in a later phases.
- For example if during testing a design error is identified then the feedback path allows the design to be reworked and the changes to be reflected in the design document.
- There is no feedback path to the feasibility stage. This means that the feasibility study errors cannot be corrected.

Requirements analysis and specification

- The aim of the requirements analysis and specification phase is to understand the exact requirements of the customer and to document them properly. This phase consists of two distinct activities, namely
 - Requirements gathering and analysis, and
 - Requirements specification
- The goal of the requirements gathering activity is to collect all relevant information from the customer regarding the product to be developed. This is done to clearly understand the customer requirements so that incompleteness and inconsistencies are removed.
- The requirements analysis activity is begun by collecting all relevant data regarding the product to be developed from the users of the product and from the customer through interviews and discussions.
- During SRS activity, the user requirements are systematically organized into a Software Requirements Specification (SRS) document.

Design

- During the design phase the software architecture is derived from the SRS document. Two distinctly different approaches are available.
- Traditional design consists of two different activities; first a structured analysis of the requirements specification is carried out where the detailed structure of the problem is examined. During structured design, the results of structured analysis are transformed into the software design.

Coding and unit testing (Implementation)

- The purpose of the coding and unit testing phase of software development is to translate the software design into source code. Each component of the design is implemented as a program module. The end-product of this phase is a set of program modules that have been individually tested.
- Each module is unit tested for determine the correct working of all the individual modules.

Integration and system testing

- Integration of different modules is done once they have been coded and unit tested. During the integration and system testing phase, the modules are integrated in a planned manner.
- Finally, when all the modules have been successfully integrated and tested, system testing is carried out. The goal of system testing is to ensure that the developed system conforms to its requirements laid out in the SRS document. System testing usually consists of three different kinds of testing activities.
- α – testing: It is the system testing performed by the development team.
- β – Testing: It is the system testing performed by a friendly set of customers.
- Acceptance testing: It is the system testing performed by the customer himself after the product delivery to determine whether to accept or reject the delivered product.

Maintenance

- Maintenance involves performing any one or more of the following three kinds of activities:
 - Correcting errors that were not discovered during the product development phase. This is called corrective maintenance.
 - Improving the implementation of the system, and enhancing the functionalities of the system according to the customer's requirements. This is called perfective maintenance.
 - Porting the software to work in a new environment. For example, porting may be required to get the software to work on a new computer platform or with a new operating system. This is called adaptive maintenance.

2.2 RISK MANAGEMENT

- The aim of risk management is to reducing the impact of all kind of risks that might affect a project. Risk management consists of three essential activities: risk identification, risk assessment, and risk containment.

Risk Identification

- A software project can be affected by a large variety of risks. In order to be able to systematically identify the important risks which might affect a software project, it is necessary to categorize risks into different classes.
- The project manager can then examine which risks from each class are relevant to the project. There are three main categories of risks which can affect a software project:

Project Risks

- Project risks concern varies forms of budgetary, schedule, personnel, resource, and customer-related problems. An important project risk is schedule. It is very difficult to monitor and control a software project.
- It is very difficult to control something which cannot be seen.
- The invisibility of the product being developed is an important reason for many software projects failure.
- So in our project we are trying to resolve this kind of project risk which is also known as schedule risk.

Technical Risks

- Technical risks concern design, implementation, interfacing, testing, and maintenance problems.
- Technical risks also include ambiguous specification, incomplete specification, changing specification, technical uncertainty. Most technical risks occur due to the team member's insufficient knowledge about the project.
- So in order to prevent this risk, we have done appropriate project analysis before starting our project.

Business Risks

- This type of risks include risks of building an excellent product that no one wants, losing budgetary or personnel commitments, etc.

Risk Assessment

- Risk assessment involves identifying risk, analyzing them and then assigns priority to them on the basis of the analysis.
- The objective of risk assessment is to rank the risks in terms of their damage. For risk assessment, first each risk should be rated in two ways:
- The probability of a risk coming true (denoted as r).
- The result of the problems associated with that risk (denoted as s).
- Based on these two factors, the priority of each risk can be computed:

$$p = r * s$$

Where, p is the priority with which the risk must be handled, r is the probability of the risk becoming true, and so is the result of damage caused due to the risk becoming true. If all identified risks are prioritized, then the most likely and damaging risks can be handled first and reject procedures can be designed for these risks.

Risk Containment

- After all the identified risks of a project are assessed, plans must be made to containment the most damaging and the most likely risks.
- Different risks require different containment procedures. In fact, most risks require expertness on the part of the project manager in handling the risk.
- There are three main strategies to plan for risk containment:
 - **Avoid the risk:** This may take several forms such as discussing with the customer to change the requirements to reduce the scope of the work.
 - **Transfer the risk:** This strategy involves getting the risky component developed by a third party.
 - **Risk reduction:** This involves planning ways to containment the damage due to a risk.
- To choose between the different strategies of handling a risk, the project manager must consider the cost of handling the risk and the corresponding reduction in risk.
- For this we may compute the risk leverage of the different risks. Risk leverage is the difference in risk divided by the cost of reducing the risk.

$$\text{Risk leverage} = (\text{Risk before reducing} - \text{Risk after reducing}) / \text{cost of reducing}$$

3. DETAIL DESCRIPTION

Customer:

The details of customer have been stored in this module. User can insert detail about their customer such as Customer Name, City and Contact Number etc.

- **CustomerID** :ID of customer
- **CustomerName** :Name of customer
- **ContactNumber** :Contact number of customer
- **City** :City of customer
- **CreatedDate** :Date when customer added

Product:

In this module, the details of product have been stored. User can view product detail like Product Name, GST Percent, Price etc.

- **ProductID** :ID of product
- **ProductName** :Name of product
- **ProductGSTPercent** :GST percent of product
- **ProductPrice** :Price of product
- **ProductQuantity** :Quantity of product

Purchase:

In this module, the details of purchased product have been stored. User can insert and delete purchased product. User can insert product details like Product Name, GST Percent, Purchase Quantity, Total Price etc.

- **PurchaseID** :ID of purchase
- **ProductID** :ID of product
- **CustomerID** :ID of customer
- **PurchasedQuantity** :Quantity of purchase
- **TotalPrice** :Total price of purchase

Bill:

In this module, After purchase products by the customer, Bill will be generated and save all details for future use. Bill contains details like Customer Name, Product Name, Product GST Percent, Purchased Quantity, Total Price of Bill etc.

- **BillID** :ID of Bill
- **PurchaseID** :ID of purchase
- **CustomerID** :ID of customer
- **TotalBill** : Total Bill Amount to be pay
- **BillDate** :Date of bill created
- **Ispaid** :Is bill paid or not

History:

In this module, the past purchase details of the customers will be stored. User can view & download previously generated bill. Bill contains details like Customer Name, Product Name, Product GST Percent, Purchased Quantity, Total Price of Bill etc.

- **HistoryID** :ID of History
- **CustomerID** :ID of customer
- **PurchaseID** :ID of purchase
- **BillID** :ID of BillGenerate

4. DIAGRAMS

4.1 CLASS DIAGRAM

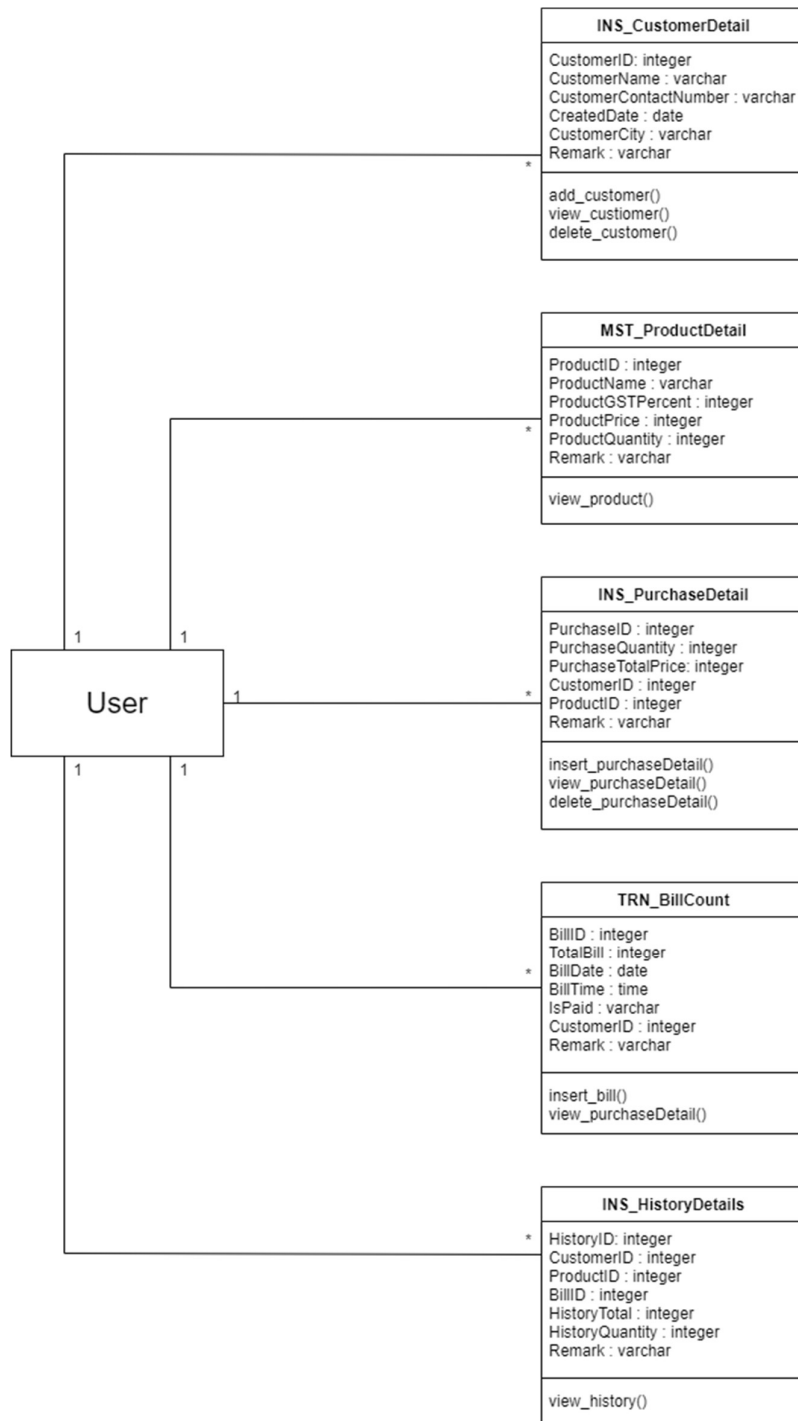


Fig. 4.1 Class Diagram

4.2 SEQUENCE DIAGRAM

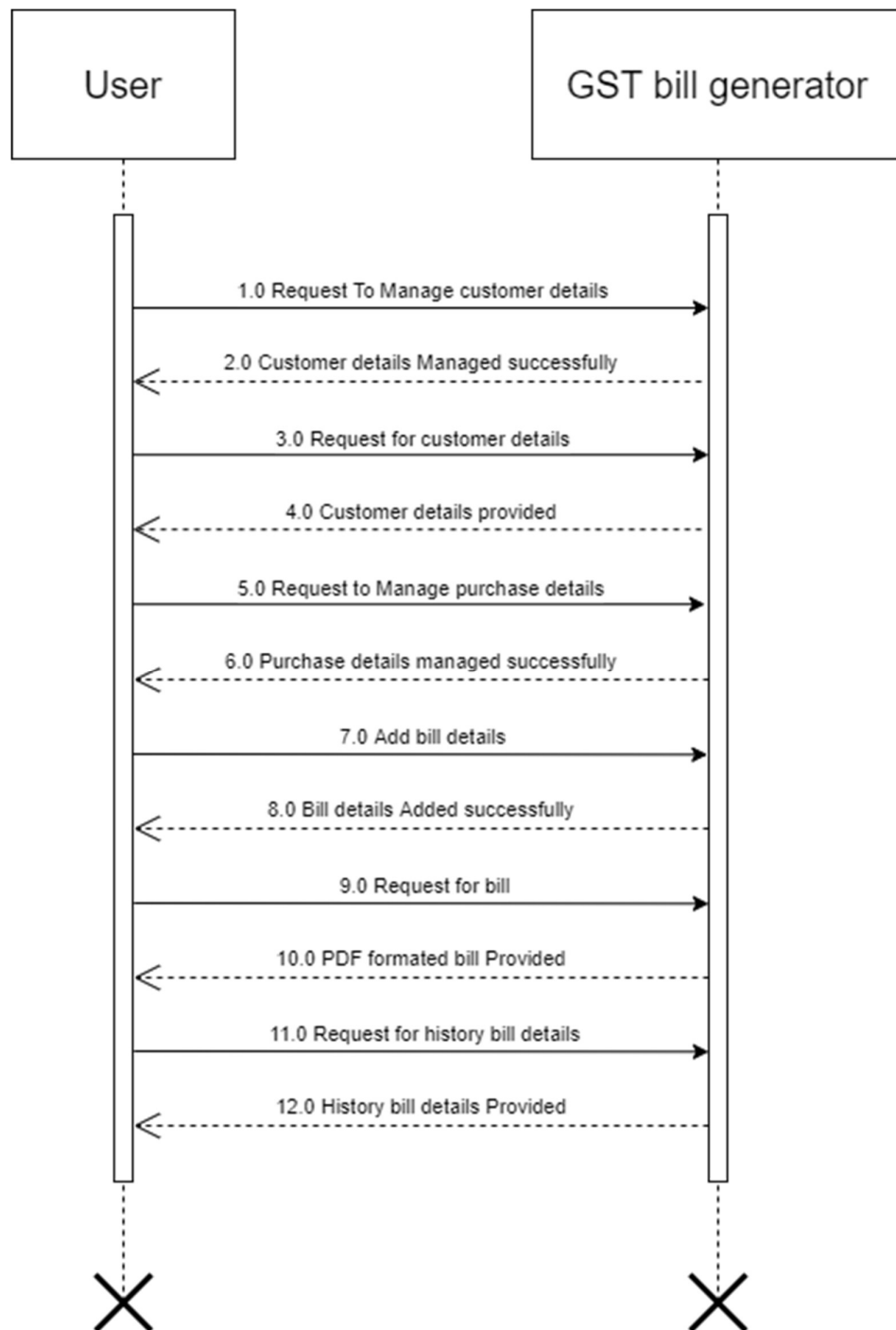


Fig. 4.2 Sequence Diagram

4.3 COLLABORATION DIAGRAM

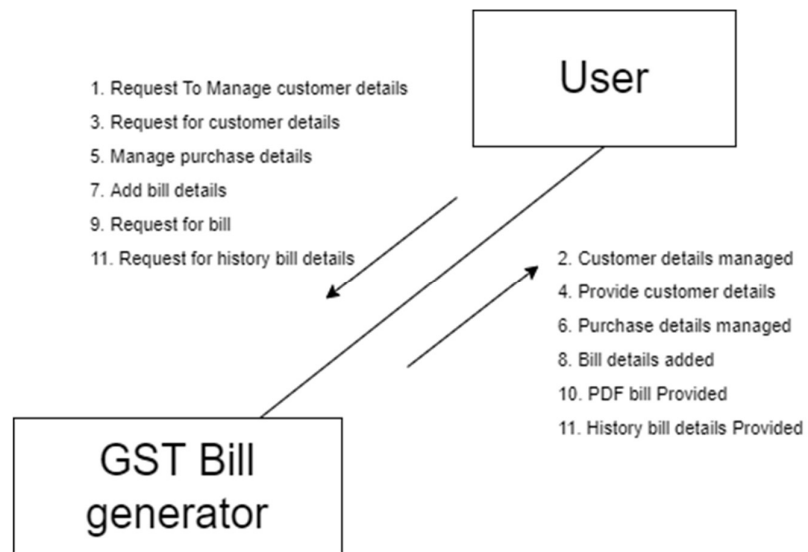


Fig. 4.3 Collaboration Diagram

4.4 STATE DIAGRAM

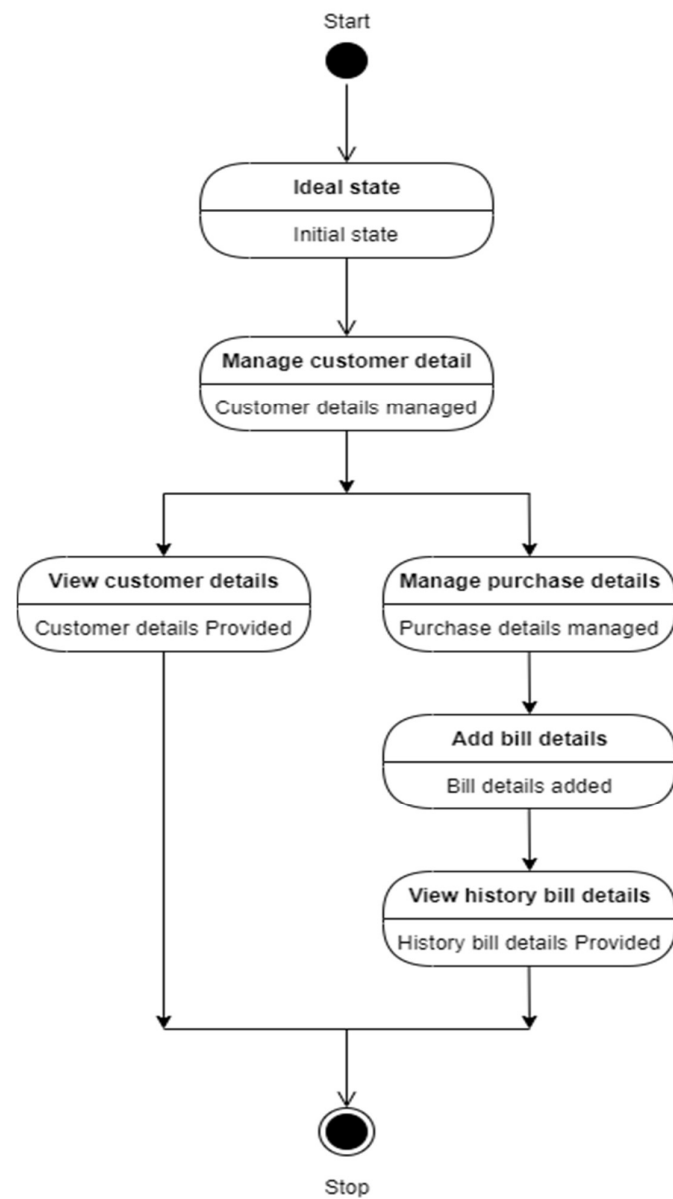


Fig. 4.4 State Diagram

4.5 ACTIVITY DIAGRAM

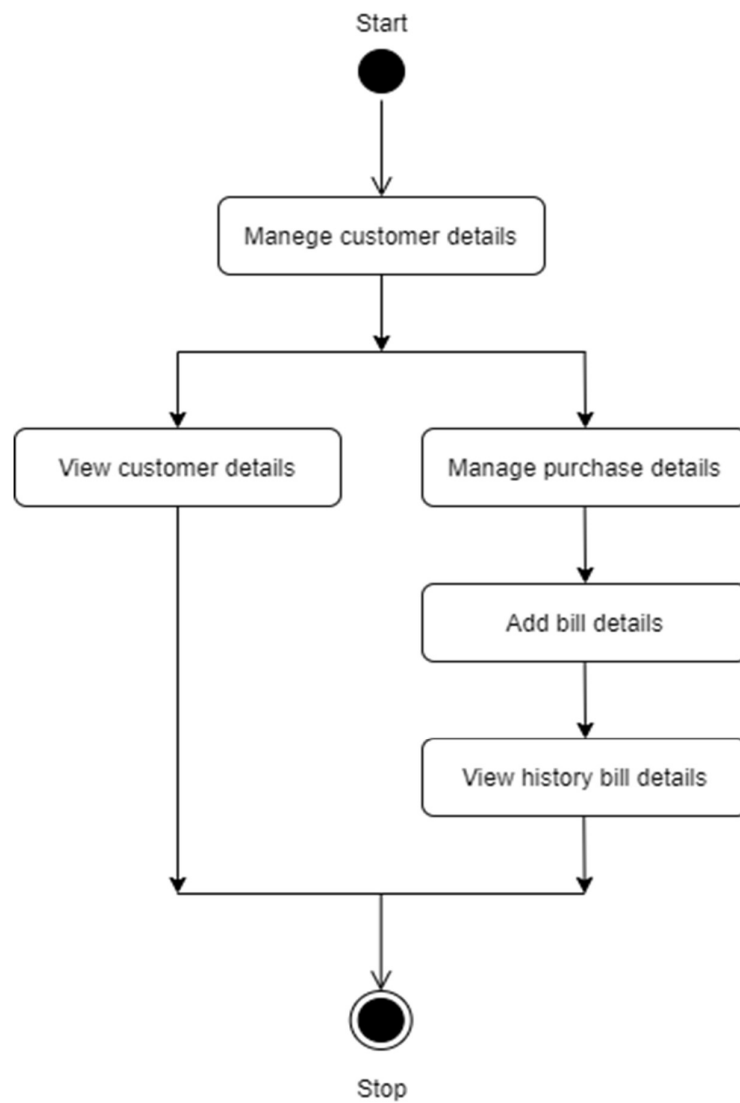


Fig. 4.5 Activity Diagram

4.6 USE CASE DIAGRAM

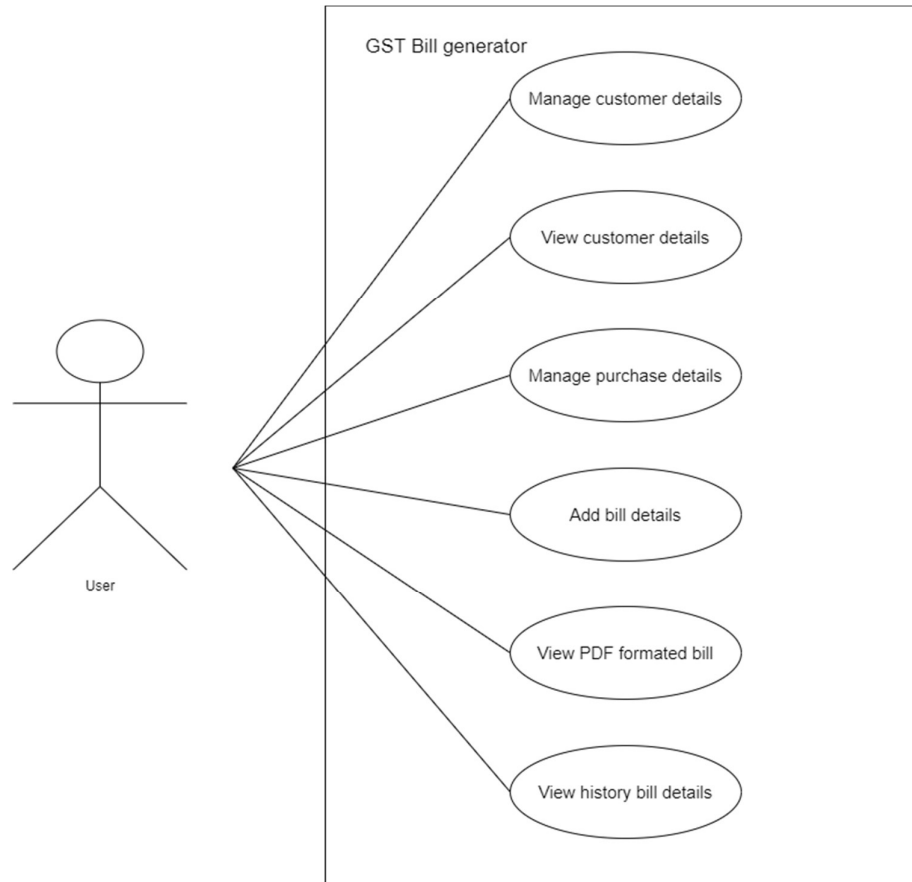


Fig. 4.6 Use Case Diagram

4.7 DATA FLOW DIAGRAM

Data Flow Diagram Level 0

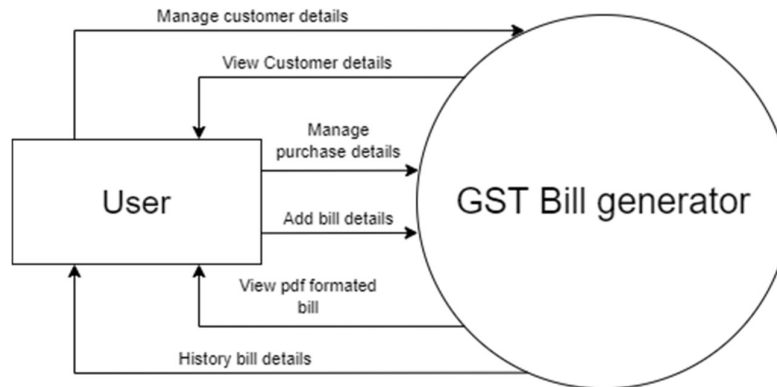


Fig. 4.7.1 Data Flow Diagram (Level 0)

Data Flow Diagram Level 1

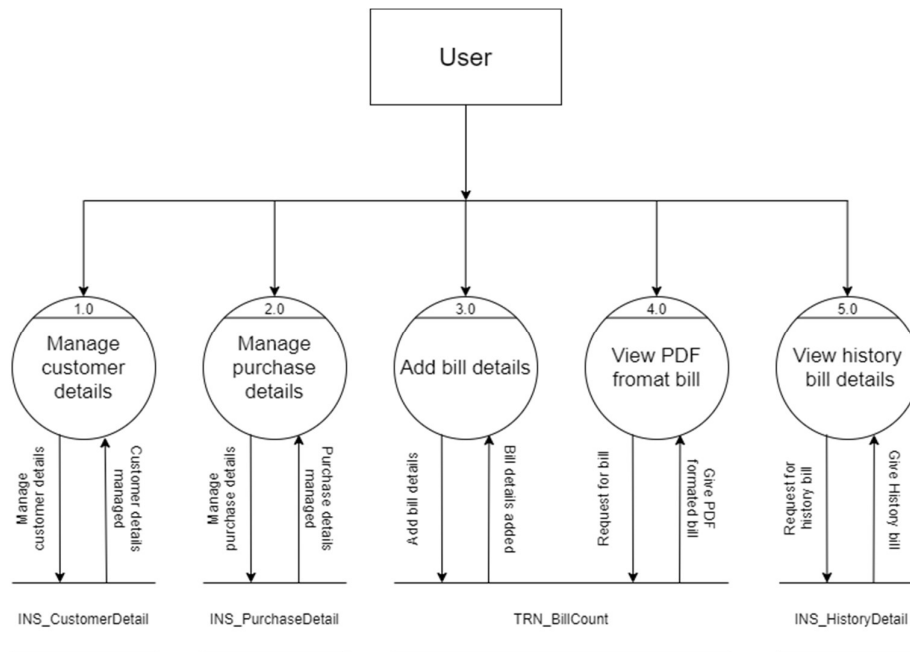


Fig. 4.7.2 Data Flow Diagram (Level 1)

5. DATA DICTIONARY

DATABASE TABLES

INS_CustomerDetail				
Field Name	Data Type	Size	Constraint	Reference
CustomerID	integer	10	PK	-
CustomerName	varchar	50	NOT NULL	-
ContactNumber	varchar	20	NOT NULL	-
CreatedDate	datetime	50	NOT NULL	-
City	varchar	30	NOT NULL	-
Remark	varchar	100	-	-

MST_ProductDetail				
Field Name	Data Type	Size	Constraint	Reference
ProductID	integer	10	PK	-
ProductName	varchar	50	NOT NULL	-
ProductGSTPercent	integer	10	NOT NULL	-
ProductPrice	integer	10	NOT NULL	-
ProductQuantity	integer	10	NOT NULL	-
Remark	varchar	100	-	-

INS_HistoryDetail				
Field Name	Data Type	Size	Constraint	Reference
HistoryID	integer	10	PK	-
CustomerID	integer	10	FK	INS_CustomerDetail
PurchaseID	integer	10	FK	INS_PurchaseDetail
BillID	integer	10	FK	INS_BillGenerate
Remark	varchar	100	-	-

INS_PurchaseDetail				
Field Name	Data Type	Size	Constraint	Reference
PurchaseID	integer	10	PK	-
CustomerID	integer	10	FK	INS_CustomerDetail
ProductID	integer	10	FK	MST_ProductDetail
PurchasedQuantity	integer	10	NOT NULL	-
TotalPrice	integer	10	NOT NULL	-
Remarks	varchar	100	-	-

TRN_BillGenerate				
Field Name	Data Type	Size	Constraint	Reference
BillID	integer	10	PK	-
PurchaseID	integer	10	FK	INS_PurchaseDetail
CustomerID	integer	10	FK	INS_CustomerDetail
TotalBill	integer	10	NOT NULL	-
BillDate	datetime	50	NOT NULL	-
IsPaid	integer	10	NOT NULL	-
Remarks	varchar	30	-	-

5.1 E-R DIAGRAM

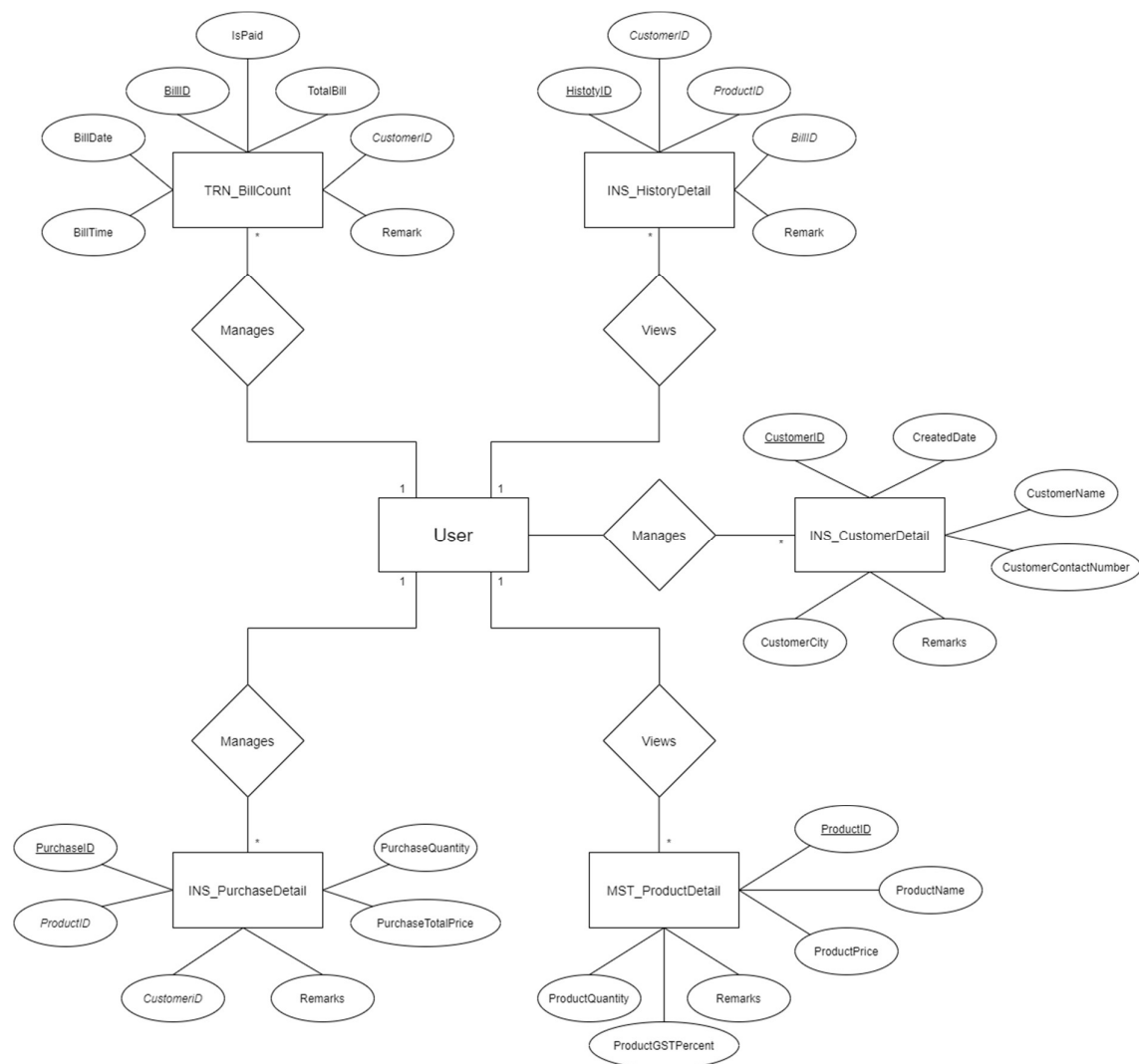


Fig. 5.1 E-R Diagram

6. SCREEN SHOTS

6.1 DASHBOARD

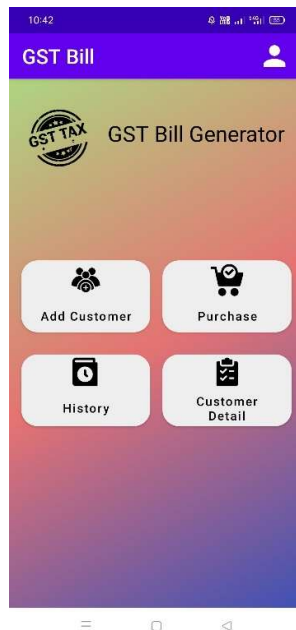


Fig. 6.1 Dashboard

- Dashboard of Application is displayed as above.
- User can add customer, purchase details etc. through dashboard.

6.2 DEVELOPER



Fig. 6.2 Developer

- Details of developers are displayed as above in Application.

6.3 ADD CUSTOMER

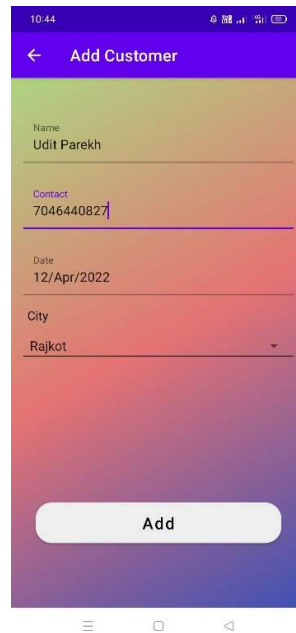


Fig. 6.3 Add Customer

- As above user can add customers.
- Here, user just needs to enter customer name, contact number and city name.

6.4 CUSTOMER DETAILS

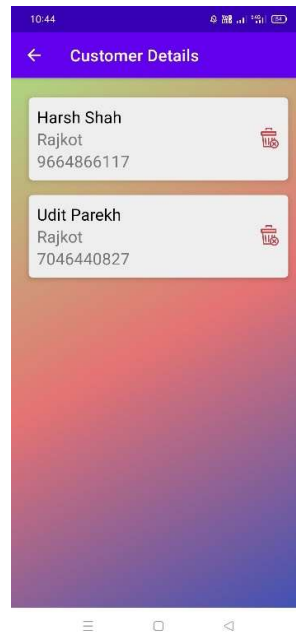


Fig. 6.4 Customer Details

- Customer details are displayed as above.
- Here, displayed details such as Customer name, city name and contact number.

6.5 PURCHASE

The screenshot shows a mobile application interface for a purchase form. The title bar is purple with a back arrow and the text 'Purchase'. The form has a gradient background. Fields include: 'Customer Name' with the value 'Udit Parekh', 'Product Name' with the value 'Graphic Card', 'Product Quantity' with the value '2', 'Product Price' with the value '8999', and 'Product GST' with the value '12'. A summary box at the bottom left shows 'Graphic Card', 'Product Qty: 2', 'Price: 20157.8', and 'GST: 12%'. At the bottom are two buttons: 'Add Items' and 'Finish'.

Field	Value
Customer Name	Udit Parekh
Product Name	Graphic Card
Product Quantity	2
Product Price	8999
Product GST	12

Summary	
Graphic Card	
Product Qty:	2
Price:	20157.8
GST:	12%

Buttons: Add Items, Finish

Fig. 6.5 Purchase

- User enters purchase details as above.
- Details are such as customer name, product name and quantity.
- Here, Product GST Percent and Product Price will generate automatically.
- User can add one or more than product here.

6.6 DISPLAY BILL

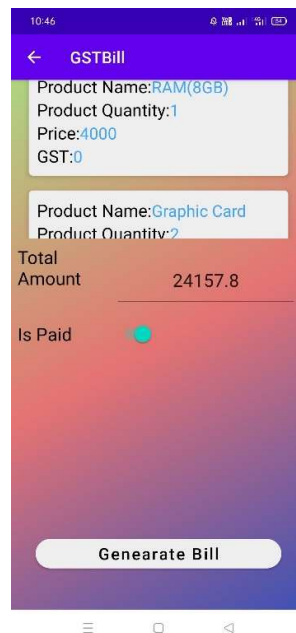


Fig. 6.6 Display Bill

- Purchased items are displayed as above.
- Total amount after calculating all price with GST displayed here.

6.7 BILL GENERATE

The screenshot shows a mobile application interface for generating a GST bill. At the top, there is a purple header bar with a back arrow and the text 'GSTBill'. Below this, the title 'GST' is displayed in large, bold letters, followed by 'Bill Generator' in a smaller font. The user information section includes 'Name: Udit Parekh', 'Contact no.: 7046440827', and 'Date: April 12, 2022 10:46:22'. A table lists the items being billed: 'Graphic Card' with a GST of 12, Price of 8999, Quantity of 2, and Total Price of 20157.8; and 'RAM(8G B)' with a GST of 0, Price of 4000, Quantity of 1, and Total Price of 4000. The total bill amount is 24157.8. A 'Close' button is located below the table. At the bottom, a dark grey box contains the text: 'GSTBill_Udit Parekh_April 12, 2022 _10-46-22.pdf has saved at Android/data/com.appl.gstbill/files/'. The Android navigation bar is visible at the very bottom.

Item	GST	Price	Quantity	Total Price
Graphic Card	12	8999	2	20157.8
RAM(8G B)	0	4000	1	4000
Total:				24157.8

Fig. 6.7 Bill Generate

- Bill is displayed as above in Application.

6.8 HISTORY

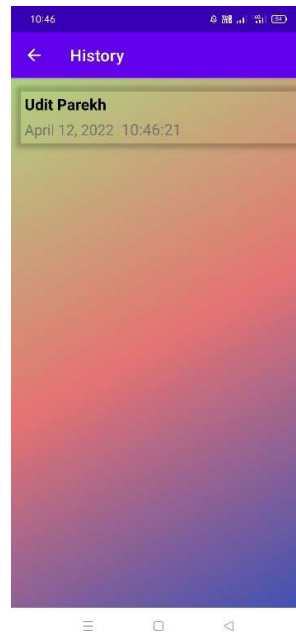


Fig. 6.8 History

- The customers list is displayed here as above.
- Here, are customers whose bill is generated.

6.9 SAVED BILL AS PDF

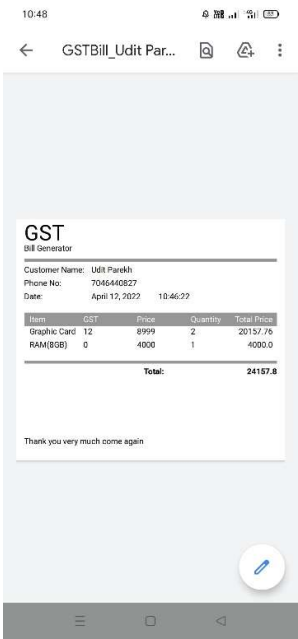


Fig. 6.9 Saved bill as PDF format

- Saved bill in PDF format is displayed as above.

7. CONCLUSION

Users will be able to generate the bill easily by using “GST Bill Generator” Android Application. In application, we have to just add customer details first time, add list of products purchased by the customer and it will generate a bill quickly. This application has functionalities to save generated bill for future use, user can view previously purchased product details of the customer etc. So, it saves our precious time and environment.

8. FUTURE ENHANCEMENT

In future, we will add functionality to print bills using the application. In which users can connect their phone to the printer and print bills. We will also add functionality to share bills, user can add or remove default products, etc. Also, we would expand our application to other platforms i.e. iOS.