SKILL DEVELOPMENT REPORT

ELECTRONICS AND COMMUNICATION ENGINEERING

"Billing Management System (BMS)"

2JR24EC053

DLithe Consultancy Services Pvt. Ltd.

DLithe®
Be Agile..Be in Demand

Skill Development Report

Intern Name: Priya.S.T

Reg. no: 2JR24EC053

Job Assignment: Python skill development project

Project name : Billing Management System(BMS)

Organization: DLithe Consultancy Services Pvt. Ltd.

Supervisor's Name: Ms Sushma

Observations:

Submitted to

Signature of Trainer

Date:

Signature of Student

Date:

DLithe®
Be Agile..Be in Demand

Letter of Transmittal

To,
Program Co-ordinator
DLithe Consultancy Services
Bengaluru

Dear Sir/Madam,

We hereby submit the report on the **Python Skill Development Activity** conducted for the **Electronics and Communication Engineering (ECE) Department**, as a part of our academic skill enhancement program.

The Python Skill Development Activity provided a valuable learning experience and helped us strengthen our programming fundamentals. The program focused on core Python concepts such as syntax, control structures, functions, data structures, and problem-solving techniques. Emphasis was given to hands-on practice, enabling students to apply theoretical concepts through practical exercises and real-world examples.

Through this activity, students developed improved logical thinking abilities and gained confidence in writing and executing Python programs. The training also highlighted the relevance of Python as a versatile programming language widely used in software development, data analysis, automation, and emerging technological areas applicable to ECE domains.

This report presents a detailed account of the training program, including the topics covered, learning objectives, and outcomes achieved. We believe that the knowledge and skills gained through this activity will be beneficial for our academic progress and future professional development.

We respectfully submit this report for your kind review and consideration.

Sincerely,
Name: Priya.S.T.
Reg. No: 2JR24EC053

Table of Contents

INTRODUCTION

A Billing Management System (BMS) is a software application designed to automate and manage the billing process of an organization. It helps in generating invoices, calculating totals, applying taxes or discounts, maintaining customer records, and tracking payments efficiently. This system replaces traditional manual billing methods, reducing human errors and saving time.

The Billing Management System ensures accurate billing, improves data security, and provides quick access to billing information. It is widely used in sectors such as retail stores, supermarkets, hospitals, hotels, and service-based organizations. By maintaining a centralized database, the system allows easy storage, retrieval, and modification of billing data, leading to better financial management and customer satisfaction.

Overall, the Billing Management System enhances operational efficiency, transparency, and reliability in handling billing and payment activities.

PROBLEM STATEMENT

In many organizations, billing is still handled using manual methods or basic record-keeping systems. These traditional approaches are time-consuming, prone to human errors, and difficult to manage as the volume of customers and transactions increases. Manual billing can lead to incorrect calculations, loss of data, duplicate entries, and delays in invoice generation, which negatively affects business efficiency and customer satisfaction.

There is a need for a reliable and automated system that can efficiently manage billing operations such as maintaining customer details, managing product or service information, generating accurate invoices, calculating taxes and discounts, and tracking payments. The absence of a centralized billing system also makes it difficult to retrieve past records, prepare reports, and ensure data security.

The objective of the Billing Management System is to overcome these challenges by providing an integrated software solution that automates the billing process, reduces errors, improves accuracy, and ensures secure storage of billing data. The system aims to enhance operational efficiency, provide quick access to billing information, and support better financial management within an organization.

OBJECTIVES OF THE PROJECT

1) To design and develop an automated billing system that simplifies and speeds up the billing process.

2) To reduce manual work and minimize human errors in bill calculation and invoice generation.

3) To maintain accurate and centralized records of customers, products, and billing transactions.

4) To generate bills and receipts efficiently with correct tax and discount calculations.

5)To provide secure storage and easy retrieval of billing information.

6)To improve overall efficiency and productivity of billing operations.

7)To support report generation for sales, payments, and billing history.

8)To enhance customer satisfaction by providing fast and accurate billing services.

9) To ensure data consistency, reliability, and transparency in billing operations.

10)To create a user-friendly interface that can be easily used by staff or administrators.

SCOPE OF THE PROJECT

The scope of the Billing Management System includes the development of a software application that automates the billing process and manages billing-related activities efficiently. The system is designed to handle customer details, product or service information, invoice generation, tax and discount calculation, and payment records.

The project focuses on providing a centralized database to store and manage billing data securely, enabling quick retrieval and updates. It allows authorized users such as administrators or billing staff to access the system through a user-friendly interface. The system supports the generation of bills and receipts, as well as basic sales and billing reports.

The Billing Management System is suitable for small to medium-sized organizations such as retail stores, supermarkets, clinics, hotels, and service-based businesses. The project does not include advanced features such as online payment gateways, mobile applications, or integration with external accounting software, but it can be extended in the future to support these functionalities.

Overall, the project aims to improve billing accuracy, reduce manual effort, and enhance operational efficiency.

METHODOLOGY

1. Requirement Analysis :

In this phase, the existing billing process is studied to understand its limitations. User requirements such as invoice generation, customer management, product management, tax calculation, and report generation are identified and documented.

2. System Design :

Based on the gathered requirements, the overall system architecture is designed. This includes designing the database structure, user interface, and system modules such as billing, customer management, and product management.

3. Development :

The system is developed using appropriate programming languages and tools. Individual modules are implemented, including customer records, product or service details, billing calculations, and invoice generation.

4. Testing :

The developed system is tested to identify and fix errors. Different testing techniques such as unit testing and system testing are performed to ensure accuracy, reliability, and proper functioning of the system.

5. Implementation :

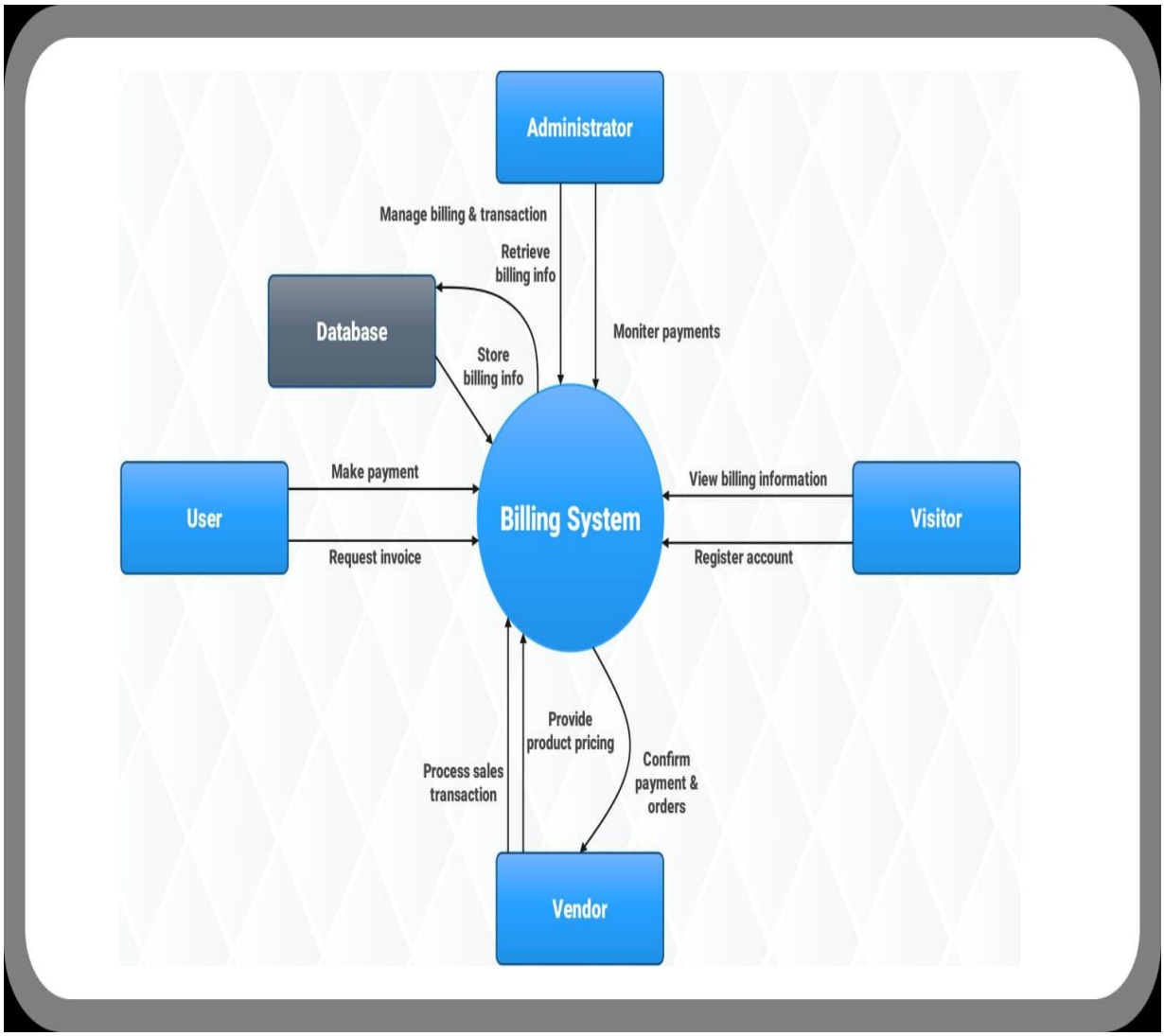
After successful testing, the system is deployed for use. Users are provided with access to the system, and sample data is used to verify real-time billing operations.

6. Maintenance :

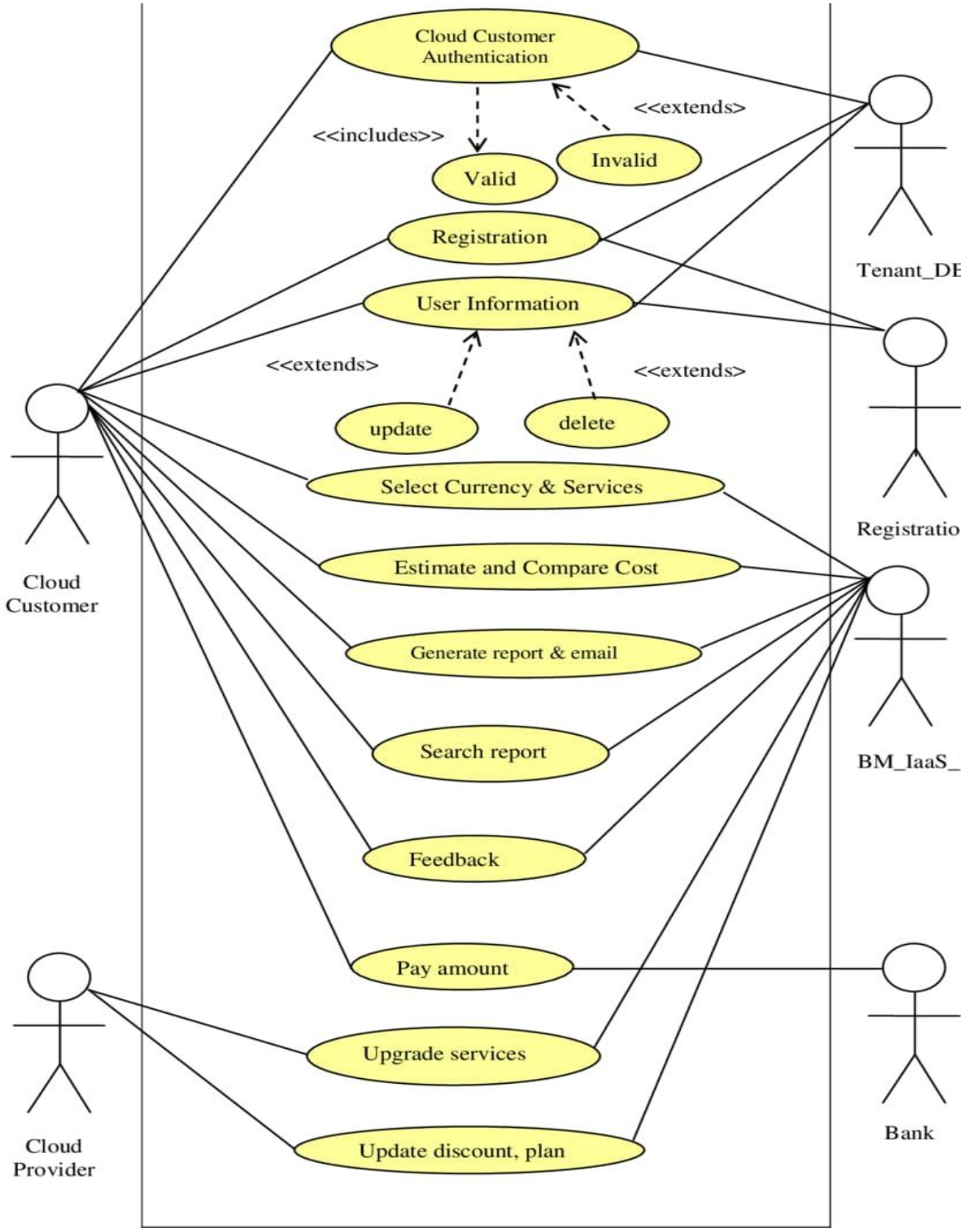
Regular updates and maintenance are performed to improve system performance, fix bugs, and add enhancements based on user feedback.

FLOW CHART

KEY FLOW :



USE CASE DIAGRAM



TOOLS AND TECHNOLOGIES USED

1. Programming Language :

Java / Python / C# / PHP Used to develop the core functionality of the Billing Management System, including billing logic, calculations, and data handling.

2. Front-End Technologies :

HTML, CSS, JavaScript Used to design a user-friendly interface for billing operations and data entry.

3. Database :

MySQL / SQLite / PostgreSQL Used to store customer details, product information, billing records, and payment history in a structured and secure manner.

4. Development Environment :

Visual Studio Code / Eclipse / Net-beans Used for writing, testing, and debugging the source code.

5. Server / Runtime Environment :

Apache / XAMPP / Local-host Used to run and test the application locally.

6. Operating System :

Windows / Linux Platform used for development and execution of the system.

7. Design & Modeling Tools :

Draw.io / Lucid-chart Used to design system architecture diagrams, ER diagrams, and UML diagrams.

RESULTS/SNAPSHOTS :

Code :

```
products = {
   1: {"name": "Pen", "price": 10},
   2: {"name": "Notebook", "price": 50},
   3: {"name": "Pencil", "price": 5},
   4: {"name": "Eraser", "price": 3} }
cart = []
def show_products():
   print("\nAvailable Products:")
   print("ID\t Product \t\t Price")
   for pid, details in products.items():

print(f"{pid}\t{details['name']}\t\t{details['price']
}")
def add_to_cart():
   pid = int(input("Enter Product ID: "))
   if pid in products:
      qty = int(input("Enter Quantity: "))
      cart.append({
         "name": products[pid]["name"],
         "price": products[pid]["price"],
         "quantity": qty }  )
      print("Product added to cart.")
   else:
      print("Invalid Product ID.")
def generate_bill():
   print("\n----- BILL RECEIPT -----")
   total = 0
  print("Product\tQty\tPrice\tAmount")
   print("Product\tQty\tPrice\tAmount")
   for item in cart:
      amount = item["price"] * item["quantity"]
      total += amount

print(f"{item['name']}\t{item['quantity']}\t{item
['price']}\t{amount}")
   tax = total * 0.05  # 5% tax
   grand_total = total + tax
   print("\n Sub-total:", total)
   print("Tax (5%):", tax)
   print("Grand Total:", grand_total)
 def main():
   while True:
      print("\nBilling Management System")
      print("1. Show Products")
      print("2. Add to Cart")
      print("3. Generate Bill")
      print("4. Exit")
      choice = input("Enter your choice: ")
      if choice == "1":
         show_products()
      elif choice == "2":
         add_to_cart()
      elif choice == "3":
         if cart:
            generate_bill()          else:
```

```
    .
                print("Cart is empty.")
        elif choice == "4":
            print("Thank you for using the system.")
            Break
else:
            print("Invalid choice.")
main()
```

Snapshots (Sample Output Screens) :

Snapshot 1: Main Menu Screen

Billing Management System

1. Show Products

2. Add to Cart

3. Generate Bill

4. Exit

Enter your choice :

The results show that the system :

-Accurately calculates the bill amount based on product price and quantity

-Applies tax correctly.

-Generates a clear and readable bill receipt

-Reduces manual billing effort and errors

-Provides a user-friendly, menu-driven interface

-The system works as expected for different inputs and produces correct billing outputs.

APPLICATIONS

Retail Stores and Shops :

Used to generate bills quickly for customers and manage daily sales.

Supermarkets and Shopping Malls :

Helps in handling large volumes of products, calculating totals, and generating accurate invoices.

Hotels and Restaurants :

Helps in billing for food, room services, and additional facilities.

Pharmacies and Medical Stores :

Used to generate medicine bills and maintain sales records.

CONCLUSION

The Billing Management System project was successfully designed and implemented using Python. The system effectively automates the billing process by allowing users to manage products, generate bills, calculate taxes, and produce accurate receipts. This project helps reduce manual effort, minimizes calculation errors, and improves overall efficiency in billing operations.

The developed system provides a simple, user-friendly, menu-driven interface that can be easily used by staff with basic computer knowledge. It demonstrates the practical application of Python programming concepts such as dictionaries, lists, functions, loops, and conditional statements in solving real-world problems.

Overall, the Billing Management System meets its objectives and serves as a reliable solution for small-scale billing requirements. With further enhancements such as database integration, graphical user interface, and online payment support, the system can be expanded to meet the needs of larger organizations.

GITHUB REPOSITORY LINK

GIT REPO LINK :-

https://github.com/priyast-tech/project-DLithe

MINDMAP