ebilling and Invoice System

A Project Report Submitted

In partial fulfillment of the Requirements

for the Degree of

MASTER OF COMPUTER APPLICATION

by

Divyansh Pandey

University Roll No 1900290140014

Under the Supervision of

Prof. Neelam Rawat

Assistant Professor

KIET Group of Institutions, Delhi -NCR ,Ghaziabad

to the

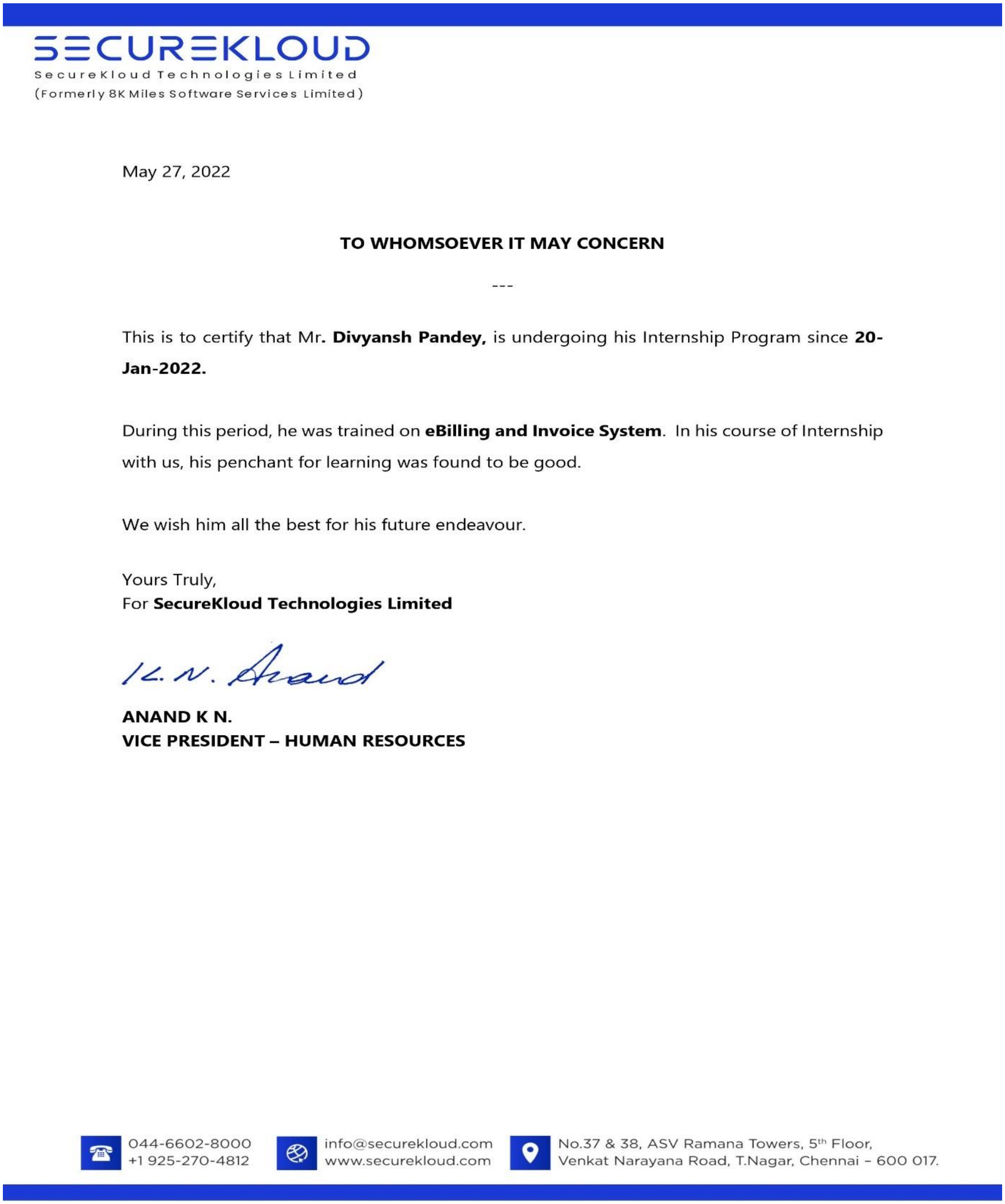
Faculty of MASTER OF COMPUTER APPLICATION

DR. APJ ABDUL KALAM TECHNICAL UNIVERSITY

LUCKNOW

(Formerly Uttar Pradesh Technical University, Lucknow)

June, 2022



SECUREKLOUD
SecureKloud Technologies Limited
(Formerly 8K Miles Software Services Limited)

May 27, 2022

TO WHOMSOEVER IT MAY CONCERN

This is to certify that Mr**. Divyansh Pandey,** is undergoing his Internship Program since **20-Jan-2022.**

During this period, he was trained on **eBilling and Invoice System**. In his course of Internship with us, his penchant for learning was found to be good.

We wish him all the best for his future endeavour.

Yours Truly,
For **SecureKloud Technologies Limited**

K. N. Anand

ANAND K N.
VICE PRESIDENT – HUMAN RESOURCES

☎ 044-6602-8000
+1 925-270-4812

info@securekloud.com
www.securekloud.com

No.37 & 38, ASV Ramana Towers, 5th Floor,
Venkat Narayana Road, T.Nagar, Chennai – 600 017.

Declaration

I hereby declare that The Project entitled **eBilling and Invoice System** is an outcome of my own efforts under the guidance of **Prof. Neelam Rawat**. The project is submitted to the department of MCA. For the partial fulfilment of Master of Computer Application 2019-22.

I also declare that project report is not submitted in any of the university previously.

Date: 26-May-2022

Place: Ghaziabad

CERTIFICATE

Certified that **Divyansh Pandey (University Roll No 1900290140014)** have carried out the project work having “ebilling and invoice System” for Master of Computer Application from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU),Technical University, Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date: **Divyansh Pandey**
(1900290140014)

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

Date: **Prof. Neelam Rawat**
Assistant Professor
Department of Computer Application
KIET Group of Institutions, Ghaziabad

Signature of External Examiner **Signature of Internal Examiner**

Dr. Ajay Kumar Shrivastava
Head, Department of
Computer Application KIET
Group of Institutions, Ghaziabad

ebilling and Invoice system: Divyansh Pandey

Abstract

This project deals with the design of Internet billing system, in which it is possible pay invoices electronically. This approach is implemented via virtual banks, in which the process of money transfer can be implemented. In other hand many applications can be realize such as; deposit e-money, withdrawal e-money and determine account balance. A Gate way translator is used to apply authentication rules, security and privacy. The client uses MS Excel, and maintains their product list, customer list, and prints the invoice, however it is not possible them to share the data from multiple system in multi user environment, there is lot of duplicate work, and chance of mistake. When the product price is changed, they need to update each and every excel file. There is no option to find and print previous saved invoice. There is no security; anybody can access any report and sensitive data, also no reports to find out the sales volume, stock list, and summary report. This E-Billing and invoicing system is used to overcome the entire problem which they are facing currently, and making complete atomization of manual billing and invoicing system.

ACKNOWLEDGEMENTS

Success in life is never attained single handly. My deepest gratitude goes to my team leader at The SecureKloud for his guidance, help and encouragement throughout my work. Their enlightening ideas, comments, and suggestions. Words are not enough to express my gratitude to Dr. Ajay Kumar Shrivastava, Professor and Head, Department of Computer Applications and all the faculties for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Table of Contents

List Of Figures

List Of Tables

Chapter 1

Introduction

1.1 Project Description

This project deals with the design of Internet billing system, in which it is possible pay invoices electronically. This approach is implemented via virtual banks, in which the process of money transfer can be implemented. In other hand many applications can be realize such as; deposit e-money, withdrawal e-money and determine account balance. A Gate way translator is used to apply authentication rules, security and privacy.

The client uses MS Excel, and maintains their product list, customer list, and prints the invoice, however it is not possible them to share the data from multiple system in multi user environment, there is lot of duplicate work, and chance of mistake. When the product price is changed, they need to update each and every excel file.

1.2 Background

As billing system in traditional mode is not sufficient for the growth of economy of the nation so the proposed system mostly focused on increasing the economy of the nation by changing the traditional pen and paper mode of billing in various firms , organizations and shops etc. bringing the digital payment and billing system help in improving the digital payment as well as improving the graphical rise of tax payee. it is a big challenge for me to change the mindset of the people and start the billing system digitally ahead. Developing a mobile app plays an important role in the technology nowadays, as the number of mobile devices are multiple times more than the number of our population. From that huge market, Android devices get 85.9% of all mobile devices, which show us how big potential that an Android application has to affect our society. Android Studio is one of the most useful types of API, which offers a cross platform mobile development. Developing an application for android using Android Studio will give us an opportunity for later, if we want to expand the application to IOS operating system. Java is one of the most popular programming languages at the moment, with approximately 51% of all developers using it regularly. The language creates 17000 jobs each month globally. It is very effective and powerful.

1.3 Motivations

The motivation for this project comes from personal issues of myself, as me and my friends one day visit to the nearby restaurant to celebrate the day of Birthday. After warm welcome we have food and spend our quality time but while we are their to go out from the restaurant then the owner started giving us the pen paper receipt having its own cost which was high enough we were ready to pay the cost only when the owner give the printed receipt of it the reasonable price of the product that we have. But he refused to do that because of some gst tax he have to pay to government then we decided to work on such project which will increase our country economy growth.

Chapter 2

LITERATURE REVIEW

E-payment system is increasingly becoming a daring means of payments in today's business world. This is due to its efficiency, convenience and timeliness. It is a payment system that is continuously being embraced and adopted in the financial system of both developed and developing countries with a view to simplify and ease payments in business transactions.

As a result, many studies were conducted around the globe by scholars on e-payment adoption. It is based on this that this research paper looks at the available past literature on e-payment adoption across the world, with a view to highlight the scope, methodology and Information System (IS) models used by previous researchers so as to identify research gaps and recommend such for future studies.

The study employed an extensive literature search on e-payment adoption with the aid of Google Scholar for those recent studies between the years 2010-2015. To facilitate the understanding of the issue under study, previous studies were analysed based on scope-geographical location of the study, theories/models adopted and methodology used.

Finally, the paper has identified the patterns of previous researches with regards to these three items and further highlights and recommends key areas in which future research should delver on.

A systematic review of the literature consists on identifying, evaluating and interpreting all available information on a research topic or phenomenon of interest by using a reliable, auditable and rigorous methodology . This systematic review consists of three phases:

(i) planning the review,
(ii) conducting the review,
(iii) reporting the review.

Planning the review This phase consists of establishing the steps that will be performed during the review, for which research questions and strategies of search are determined.

On the other hand, exploring the benefits of electronic invoicing versus traditional paper invoicing show the reduction of costs by the reduction of transportation costs and saving time by reducing workflow . Even the environmental impacts of electronic invoicing are much lower than traditional invoicing, mainly due to energy consumption.

At an organizational or business level, electronic invoicing improves control of billing processes by increasing productivity in invoice handling units, reducing processing time, reducing paper consumption, among others .

In any case, as far as it is known, there have not been found secondary studies about e-invoicing systems with notifications. It is important to know how the existing solutions of einvoicing systems, associated technologies, security considerations and notifications delivery mechanisms have been proposed. Therefore, in this paper we present a systematic review of literature, which addresses the state of the art in terms of e-invoicing systems along with the delivery of notifications by digital means. The results obtained with this research show that the relationship between e-invoicing systems and the use of digital notifications has not been addressed yet.

Chapter 3

PROJECT CATEGORY

3.1 Technology Used

Android Studio

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA . On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps.

Software and Applications Used

APPLICATION	:	Android Studio
OPERATING SYSTEM	:	WINDOWS 10
FRONT END	:	JAVA, Xml
BACK END	:	Firebase

Back-end: Firebase

- **Firebase:** It is real-time database developed by firebase and then acquired by Google in 2014 used for developing the high-quality applications act as a storage of information of various data.

Firebase Features

- **Real-time Database** − Firebase supports JSON data and all users connected to it receive live updates after every change.
- **Authentication** − We can use anonymous, password or different social authentications.
- **Hosting** − The applications can be deployed over secured connection to Firebase servers.

3.2 Language Used

This project has been developed XML and Java.

- **XML:** XML stands for extensible markup language. A markup language is a set of codes, or tags, that describes the text in a digital document. The most famous markup language is hypertext markup language (HTML), which is used to format Web pages.

  XML tags identify the data and are used to store and organize the data, rather than specifying how to display it like HTML tags, which are used to display the data. XML is not going to replace HTML in the near future, but it introduces new possibilities by adopting many successful features of HTML.

JAVA: Java is an object-oriented programming language developed by Sun Microsystems, and it was released in 1995.

James Gosling initially developed Java in Sun Microsystems (which was later merged with Oracle Corporation).

Java is a set of features of C and C++. It has obtained its format from C, and OOP features from C++.

Java programs are platform independent which means they can be run on any operating system with any processor as long as the Java interpreter is available on that system.

Java code that runs on one platform does not need to be recompiled to run on another platform; it's called write once, run anywhere (WORA)

Chapter 4

SOFTWARE REQUIREMENT SPECIFICATION

4.1 GENERAL DESCRIPTION

4.1.1 PRODUCT DESCRIPTION:

This project deals with the design of Internet billing system, in which it is possible pay invoices electronically.

This approach is implemented via virtual banks, in which the process of money transfer can be implemented.

In other hand many applications can be realize such as; deposit e-money, withdrawal e-money and determine account balance.

4.1.2 PROBLEM STATEMENT:

The problem occurred before having computerized system includes:
There is no option to find and print previous saved invoice.

There is no security; anybody can access any report and sensitive data, also no reports to find out the sales volume, stock list, and summary report.

This E-Billing and invoicing system is used to overcome the entire problem which they are facing currently, and making complete atomization of manual billing and invoicing system.

4.2 SYSTEM OBJECTIVE

This project is made for one of the big decorator services in Mumbai, they supply decorating item to film industry for movie shooting. Presently they issue their client handwritten invoice and they enter details in manual register. And maintain MS Excel file for product rate. They want computerization of their manual invoice and bill generation process.

The client uses MS Excel, and maintains their product list, customer list, and prints the invoice, however it is not possible them to share the data from multiple system in multi user environment, there is lot of duplicate work, and chance of mistake.

When the product price is changed, they need to update each and every excel file. There is no option to find and print previous saved invoice.

There is no security; anybody can access any report and sensitive data, also no reports to find out the sales volume, stock list, and summary report.

This eBilling and invoicing system is used to overcome the entire problem which they are facing currently, and making complete atomization of manual billing and invoicing system.

4.3 Requirement Specifications:

The software requirement specification is produced at the analysis task. The function and performance allocated to application as part of system engineering are refined by establishing a complete information description, a detailed functional and behavoural description, an indication of performance requirements and design constraints.

4.3.1 Functional Requirements:

Internet Connectivity:

As discussed that Application will work on Online mode so it need regular Internet Connectivity to signup and login.

Email id and Mobile Number

To access the application and to signin or login user must have email id and mobile number to fill the mandatory field in the form.

These requirements include format requirements as well as an outline of the specific information each invoice needs to include.

1. Software must be able to calculate math automatically
2. No reference to TIN (tax identification number) or SSN
3. Style Requirements of All Invoices:
   - Must be in table format
   - Must meet all functional requirements
   - Font size: 9-14 pt
4. Invoice Header Section to include:
   - Interpreter Name
   - Contact Information: Mailing address, Email address, Phone/fax number
5. Contractual Hourly Rates Listed on invoice:
   - Daytime rate
   - Evening rate
   - Weekend/Holiday rate
6. Invoice Number
7. Invoice Date
8. One assignment listed per line
9. Retained time listed as a separate line item
10. Total of Invoice (Total amount invoiced to the UW)
11.

4.3.2 Non-functional Requirements:

Performance Requirements

1. User friendly**:** The system should be user friendly so that it can easily be understand by the user without any difficulty.
2. Ease of maintenance :- System should be easy to maintain and use.
3. Less time consuming: The system should be less time consuming which could be achieved by good programming.
4. Error free: The system should easily handle the user error in any case.
5. Static: Application runs on stand alone machine i.e. Android mobile phone of API level 19 and onward. Support only single user.

4.4 SOFTWARE AND HARDWARE REQUIREMENTS

This section describes the software and hardware requirements of the system .

4.4.1 SOFTWARE REQUIREMENTS

- **Operating system**- Android operating system is required for the android apps with the API Level 16 and onwards.
- **Database**:- Firebase is used as database as it is easy to maintain and retrieve records by simple queries which are in English language which are easy to understand and easy to write.
- **Development tools and Programming language**- XML is used to write the whole designing code and coding is done in java programming language.

4.4.2 HARDWARE REQUIREMENTS

➢ Android Mobile phone of API Level 19 and onward.

➢ Memory Used Total 6.31 MB with 5.98 MB of application space and 340 KB of data

4.5 Requirements Summary

The following preliminary lists are based on initial interviews

4.5.1 Business Requirements

The business goal for the application is to support an increase the productivity and complete automation of existing manual bill and invoice generation process. Business requirements are discussed in the Scope section, with the following additional detail:

- Sales representatives need a method to store and access sales opportunity data, and when a sale is generated, convert some or all of the information into a sales order without re-entering information.
- Each sales representative should receive customer and sales data pertinent only to them.
- The accountant should be able to enter or update product information in one interface only, with all necessary product information being received by sales staff.
- Manager must receive his or her customer and appointment data plus detailed and rollup information for each sales representative on his or her team.
- The application should support the capability to use multi user environment.
- The MIS Executive should able to generate all type of reports as and when required by the management.

The sales staff wants to improve their current ability to analyze their customers. In particular, they want to focus on identifying their best customers and building long-term relationships within that base. To enable them to accomplish this goal, they want to extract meaningful data that easily answers the following questions:

- What are the early warning signs of problems?
- Who are my best customers across product lines?
- With whom do I focus my efforts for building a long-term relationship?
- What are my customers' issues as groups?
- Geographically, where are my best customers?
- What products are my customers buying and at what rate?

4.5.2 User Requirements

User requirements are categorized by user type.

Sales Staff (Representatives and Managers)

- View the data in various ways, for example:
    - Customers who are the top buyers of specific items
    - Best customers based on criteria to be determined
    - Best customers based on geographical analysis
    - Drops in a customer's sales
- Store multilingual and multiregional information in the database rather than relying on the sales staff to translate the information
- Identify which product prices have been modified, especially on current orders in progress
- Use opportunity rules, which are statements that help the sales representative convert an opportunity into a sale
- Add third-party data sources and financial evaluation tools
- Identify where promotions and programs would be the most beneficial
- Apply discounts to customer orders:
    - Sales representatives can offer discounts up to 15 percent, or up to 20 percent with authorization.
    - Sales managers can offer discounts up to 20 percent
- Enable capture, analysis, and sharing of data about a customer across the company
- Use forecasts to establish sales goals
- Allow all staff to view all contacts, but allow each contact to be assigned to a sales representative

Accountant

- Update product details, including price, photo, and description
- Add, delete, and update product specifications
- Generate stock reports and check the availability, order the item as and when needed.

4.5.3 Operational Requirements

The following requirements provide a high-level view of how the system will run:

- Processor usage should not exceed 80 percent during concurrent uses.
- Backups will occur incrementally throughout the day.
- A full weekly backup is required to WORM drives.
- Ensure that information is easy to access either, and meaningful for the sales representative and the company.
- Minimize the technical knowledge that sales and marketing staff need to access the data, generate ad hoc queries, track promotions, and view customer segmentation information.
- Any change to information must be reflected immediately, and the changes must be propagated to the search engine so that employees that perform searches see this new information.
- The application should work with the existing communications and networking infrastructure.
- The application should deploy with a minimum of additional operational processes, manual or otherwise.

4.5.4 System Requirements

These are additional constraints from a system perspective:

- Previous data of customer, product details must be imported in the new system.
- The administrator must be able to monitor everything from the IT department.
- The information must be accessible by everyone in the company as per the rights specify.

4.6 Traditional paper vs. Electronic Billing

∗ Paper is static content

∗Costly with postage or courier services

∗ Resource intensive

∗ Time consuming and slow delivery

∗ Low security-documents easily misplaced and can be opened.

∗ Documents can be copied :

E- billing is interactive and includes raw data files.

: Cost saving : Automated : Same day quick delivery.

: High security sent to specified user and tracking document

: Can not be manipulated

4.7 Features of ebilling and Invoice System:

- **Customized Invoicing**

  Every business needs to issue an invoice according to their requirements. A billing software must allow the customization of the invoice so that a business could add or delete the details as required. There are many details such as logo, signatures, transport details, e-way bill number, etc. to the invoice. These days GST details, shipping address, billing address, party details, bank details, etc. are some of the important things that need to be mentioned on the bills.

- ***GST Related Features***

  Every month a business needs to file returns that become a difficult task without software. If your software has GST returns-related features, you can directly upload the data to the GST portal from the software and there will be fewer chances of errors. Your GST input and output calculations will be done automatically. Moreover, if your work is managed by some expert, you can even send the GST data directly from the software.

- ***SMS Or WhatsApp Integration***

  These days invoices are emailed to the clients or sent through WhatsApp. Your software must have the feature that will send the invoices directly to their email or WhatsApp numbers. It will save them time and orders can be delivered quickly. The software should also show the history of invoices emailed or sent for future reference.

- ***Online Payments***

  Your software should allow online payments by creating encrypted codes for each invoice. It will help in matching and reconciliation and automatic payments can be made. The payment reminders can also be sent to the customers from whom payments need to be received.

4.8 Preliminary investigation:

Fact Finding:

After obtaining the background knowledge, we began to collect data on the existing system.

The tools that are used in information gathering are as follows:

- On-site observation.
- Questionnaire.
- Review of the peoples.

The model we have used is Waterfall Model. In this model, first of all the existing system is observed, then customer requirements are taken in consideration then planning, modelling, construction and finally deployment.

4.9 Approach used

Software Development life cycle (SDLC) is a spiritual model used in project management that defines the stages include in an information system development project, from an initial feasibility study to the maintenance of the completed application.

There are different software development life cycle models specify and design, which are followed during the software development phase. These models are also called "**Software Development Process Models**." Each process model follows a series of phase unique to its type to ensure success in the step of software development.

Here, are some important phases of SDLC life cycle:

Waterfall Model

The waterfall is a universally accepted SDLC model. In this method, the whole process of software development is divided into various phases.

The waterfall model is a continuous software development model in which development is seen as flowing steadily downwards (like a waterfall) through the steps of requirements analysis, design, implementation, testing (validation), integration, and maintenance.

Linear ordering of activities has some significant consequences. First, to identify the end of a phase and the beginning of the next, some certification techniques have to be employed at the end of each step. Some verification and validation usually do this mean that will ensure that the output of the stage is consistent with its input (which is the output of the previous step), and that the output of the stage is consistent with the overall requirements of the system.

RAD Model

RAD or Rapid Application Development process is an adoption of the waterfall model; it targets developing software in a short period.

The RAD model is based on the concept that a better system can be developed in lesser time by using focus groups to gather system requirements.

Spiral Model

The spiral model is a risk-driven process model. This SDLC model helps the group to adopt elements of one or more process models like a waterfall, incremental, waterfall, etc. The spiral technique is a combination of rapid prototyping and concurrency in design and development activities.

Each cycle in the spiral begins with the identification of objectives for that cycle, the different alternatives that are possible for achieving the goals, and the constraints that exist. This is the first quadrant of the cycle (upper-left quadrant).

The next step in the cycle is to evaluate these different alternatives based on the objectives and constraints. The focus of evaluation in this step is based on the risk perception for the project.

The next step is to develop strategies that solve uncertainties and risks. This step may involve activities such as benchmarking, simulation, and prototyping.

V-Model

In this type of SDLC model testing and the development, the step is planned in parallel. So, there are verification phases on the side and the validation phase on the other side. V-Model joins by Coding phase.

Incremental Model

The incremental model is not a separate model. It is necessarily a series of waterfall cycles. The requirements are divided into groups at the start of the project. For each group, the SDLC model is followed to develop software. The SDLC process is repeated, with each release adding more functionality until all requirements are met. In this method, each cycle act as the maintenance phase for the previous software release.

Modification to the incremental model allows development cycles to overlap. After that subsequent cycle may begin before the previous cycle is complete.

Agile Model

Agile methodology is a practice which promotes continues interaction of development and testing during the SDLC process of any project. In the Agile method, the entire project is divided into small incremental builds. All of these builds are provided in iterations, and each iteration lasts from one to three weeks.

Any agile software phase is characterized in a manner that addresses several key assumptions about the bulk of software projects:

1. It is difficult to think in advance which software requirements will persist and which will change. It is equally difficult to predict how user priorities will change as the project proceeds.

2. For many types of software, design and development are interleaved. That is, both activities should be performed in tandem so that design models are proven as they are created. It is difficult to think about how much design is necessary before construction is used to test the configuration.

3. Analysis, design, development, and testing are not as predictable (from a planning point of view) as we might like.

Iterative Model

It is a particular implementation of a software development life cycle that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is **c**omplete. In short, iterative development is a way of breaking down the software development of a large application into smaller pieces.

Like other SDLC models, Iterative and incremental development has some specific applications in the software industry. This model is most often used in the following scenarios −

- Requirements of the complete system are clearly defined and understood.
- Major requirements must be defined; however, some functionalities or requested enhancements may evolve with time.
- There is a time to the market constraint.
- A new technology is being used and is being learnt by the development team while working on the project.
- Resources with needed skill sets are not available and are planned to be used on contract basis for specific iterations.
- There are some high-risk features and goals which may change in the future.

Big bang model

Big bang model is focusing on all types of resources in software development and coding, with no or very little planning. The requirements are understood and implemented when they come.

This model works best for small projects with smaller size development team which are working together. It is also useful for academic software development projects. It is an ideal model where requirements are either unknown or final release date is not given.

Prototype Model

The prototyping model starts with the requirements gathering. The developer and the user meet and define the purpose of the software, identify the needs, etc.

A '**quick design**' is then created. This design focuses on those aspects of the software that will be visible to the user. It then leads to the development of a prototype. The customer then checks the prototype, and any modifications or changes that are needed are made to the prototype. In many instances, the client only has a general view of what is expected from the software product. In such a scenario where there is an absence of detailed information regarding the input to the system, the processing needs, and the output requirement, the prototyping model may be employed.

Agile Approach

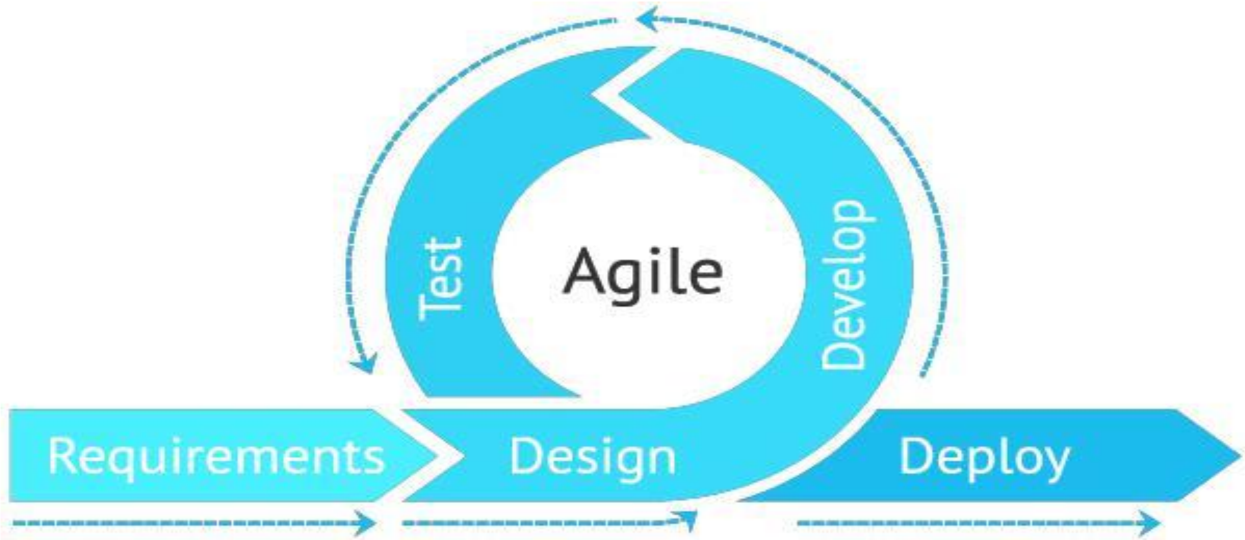


Fig 1 Approach Used in development

Agile is an iterative approach to project management and software development that helps teams deliver value to their customers faster and with fewer headaches. Instead of betting everything on a "big bang" launch, an agile team delivers work in small, but consumable, increments.

This Project goes from various development models i.e. incremental model, evalution model and most important Agile Approach. Every model give a brief description about the project development. Each has its own importance.

4.10 Preliminary Description:

The first step in the system development life cycle is the preliminary investigation to determine the feasibility of the system. The purpose of preliminary investigation is to evaluate project requests. It is not a design study nor does it include the collection of details to describe the system in all respect. Rather, it is the collecting of information that helps committee members to evaluate the merits of project request and make an informed judgement about the feasibility of the proposed project. The project preliminary report **describes your progress so far**. It should form the basis of your final report. The preliminary report should include: Problem: A clear description of the problem you are addressing

Analyst working on the preliminary investigation should accomplish the following objectives:

- Clarify and understand the project request.
- Determine the size of the project.
- Access costs and benefits of alternative approaches.
- Determine the technical and operational feasibility of alternative approaches.
- Report the findings to management with recommendations outlining the acceptance and rejection of the proposal.

Chapter 5

Feasibility study

After studying and analyzing all the existing and requires functionalities of the system, the next task is to do the feasibility study for the project. Feasibility study includes consideration of all the possible ways to provide a solution to a given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

5.1 Economical Feasibility:

It will be freely available on the Google play store without having any cost. This term means the assessment and analysis of a project's potential to support the decision-making process by objectively and rationally identifying its strengths, weaknesses, opportunities and risks associated with it, the resources that will be needed to implement the project, and an assessment of its chances of success. It consists of market **analysis**, **economic analysis**, **technical and** strategic analysis.

5.2 Technical feasibility:

This included the study of function, performance and constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we studied complete functionalities to be provided in the system, as described in the System Requirement Specification (SRS), and checked if everything was possible using different type of front end and back end platform. Technical feasibility is **the process of figuring out how you're going to produce your product or service to determine whether it's possible for your company**. Before launching your offerings, you must plan every part of your operations, from first sourcing your production materials all the way to tracking your sales.

5.3 Operational Feasibility:

No doubt the technically growing Bihar needs more enhancement in technology, this apps is very user friendly and all inputs to be taken all self-explanatory even at layman. Operational feasibility is **the measure of how well a proposed system solves the problems**, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

5.4 Tools for Conducting a Feasibility Study

5.4.1 Suggested Best Practices

Although each project can have unique goals and needs, below are some best practices for conducting a feasibility study:

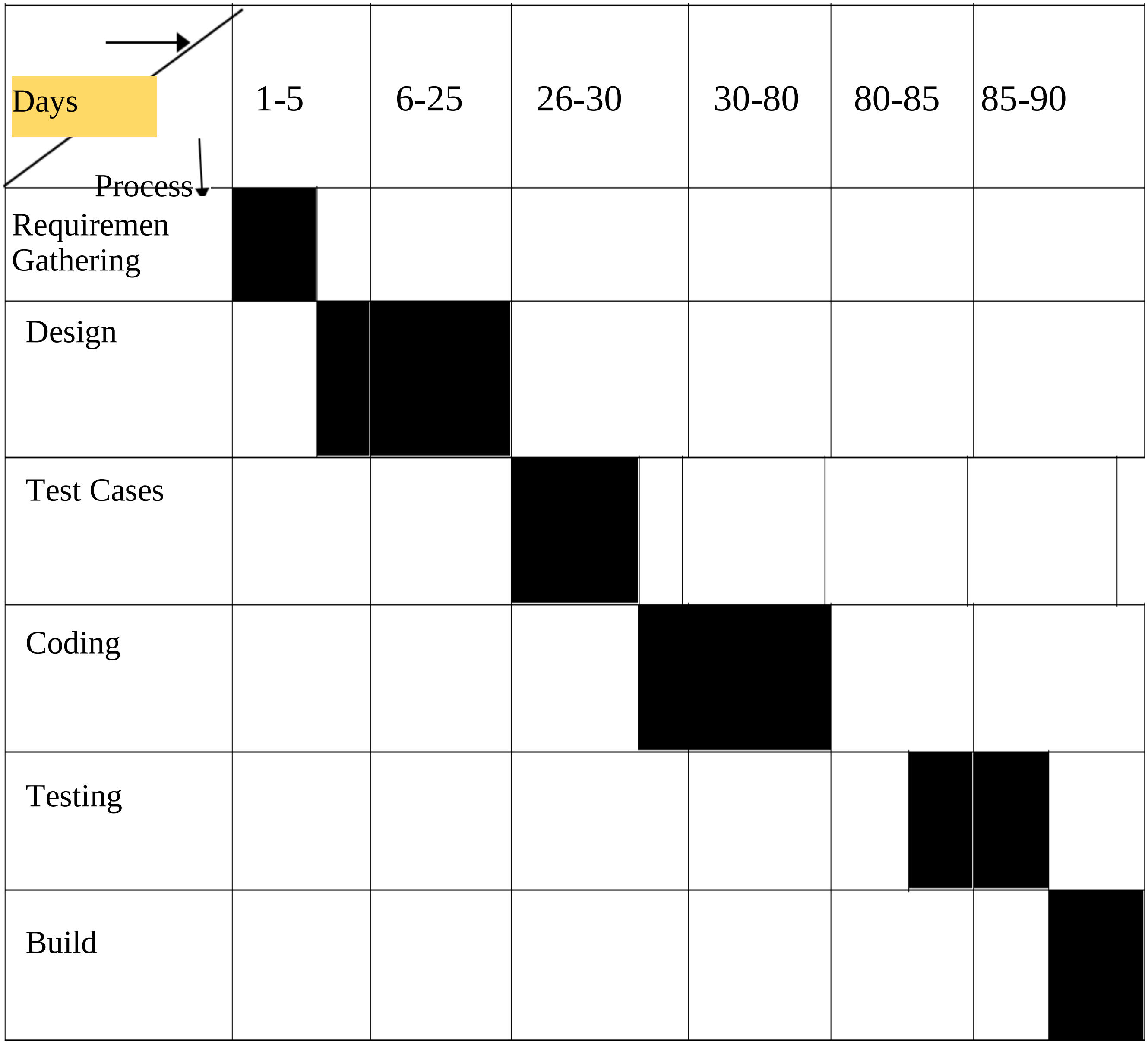
- Conduct a preliminary analysis, which involves getting feedback about the new concept from the appropriate stakeholders; consider other business scenarios and ideas
- Analyze and ask questions about the data obtained in the early phase of the study to make sure that it's solid
- Conduct a market survey or market research to identify the market demand and opportunity for pursuing the project or business
- Write an organizational, operational, or business plan, including identifying the amount of labor needed, at what cost, and for how long
- Prepare a projected income statement, which includes revenue, operating costs, and profit
- Prepare an opening day balance sheet
- Identify obstacles and any potential vulnerabilities, as well as how to deal with them
- Make an initial "go" or "no-go" decision about moving ahead with the plan

Chapter 6

Planning and Scheduling and Flow

6.1 Gantt chart

A Gantt chart can be developed for the entire project or a separate chart can be developed for each function. A tabular form is maintained where rows indicate the task with milestones and columns indicate duration(weeks/months).A Gantt chart shows all of the tasks that need to be done, the amount of time each task is expected to take, the time frames in which individual tasks are to be completed, and the relationship between various task.



Days → / Process ↓	1-5	6-25	26-30	30-80	80-85	85-90
Requiremen Gathering	■					
Design	■	■				
Test Cases			■			
Coding			■	■		
Testing					■	■
Build						■

Fig 2 Gantt Chart

6.1.1 Software Requirements with specifications

Name of Components	Specifications
Operating system	Android Operating system
Language	Java eclipse 2 Runtime Environment
Database	Firebase
Software Development kit	Android Studio
Markup Language Enable	Xml

Table Num 1: Software requirement with specifications

6.1.2 Hardware Requirements with specifications

Name of Components	Specifications
Cell Phone	Android Mobile phone API 17 and onward
Application	13.52MB
Data	304KB

Table Num 2: Hardware requirement with specifications

6.2 DATA FLOW DIAGRAM

DFD is the abbreviation for **Data Flow Diagram**. The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. DFD does not have control flow and no loops or decision rules are present. Specific operations depending on the type of data can be explained by a flowchart. Data Flow Diagram can be represented in several ways. The DFD belongs to structured -analysis modeling tools. Data Flow diagrams are very popular because they help us to visualize the major steps and data involved in software-system processes.

6.2.1 Components of DFD

The Data Flow Diagram has 4 components:

- **Process**
  Input to output transformation in a system takes place because of process function. The symbols of a process are rectangular with rounded corners, oval, rectangle or a circle. The process is named a short sentence, in one word or a phrase to express its essence

- **Data Flow**
  Data flow describes the information transferring between different parts of the systems. The arrow symbol is the symbol of data flow. A relatable name should be given to the flow to determine the information which is being moved. Data flow also represents material along with information that is being moved. Material shifts are modeled in systems that are not merely informative. A given flow should only transfer a single type of information. The direction of flow is represented by the arrow which can also be bi-directional.

- **Warehouse**
  The data is stored in the warehouse for later use. Two horizontal lines represent the symbol of the store. The warehouse is simply not restricted to being a data file rather it can be anything like a folder with documents, an optical disc, a filing cabinet. The data warehouse can be viewed independent of its implementation. When the data flow from the warehouse it is considered as data reading and when data flows to the warehouse it is called data entry or data updation.

- **Terminator**
  The Terminator is an external entity that stands outside of the system and communicates with the system. It can be, for example, organizations like banks, groups of people like customers or different departments of the same organization, which is not a part of the model system and is an external entity. Modeled systems also communicate with terminator.

6.2.2 Data flow diagram levels

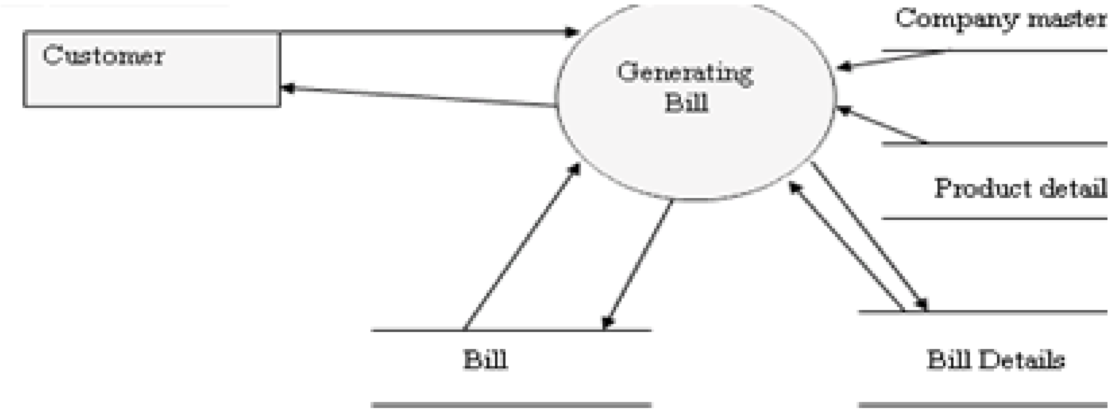
Data flow diagrams are also categorized by level. Starting with the most basic, level 0, DFDs get increasingly complex as the level increases. As you build your own data flow diagram, you will need to decide which level your diagram will be.

Level 0 DFDs, also known as context diagrams, are the most basic data flow diagrams. They provide a broad view that is easily digestible but offers little detail. Level 0 data flow diagrams show a single process node and its connections to external entities.

Level 1 DFDs are still a general overview, but they go into more detail than a context diagram. In a level 1 data flow diagram, the single process node from the context diagram is broken down into subprocesses. As these processes are added, the diagram will need additional data flows and data stores to link them together.

Level 2+ DFDs simply break processes down into more detailed subprocesses. In theory, DFDs could go beyond level 3, but they rarely do. Level 3 data flow diagrams are detailed enough that it doesn’t usually make sense to break them down further.

6.2.3 DFD OF ebilling and Invoice System



First Level DFD

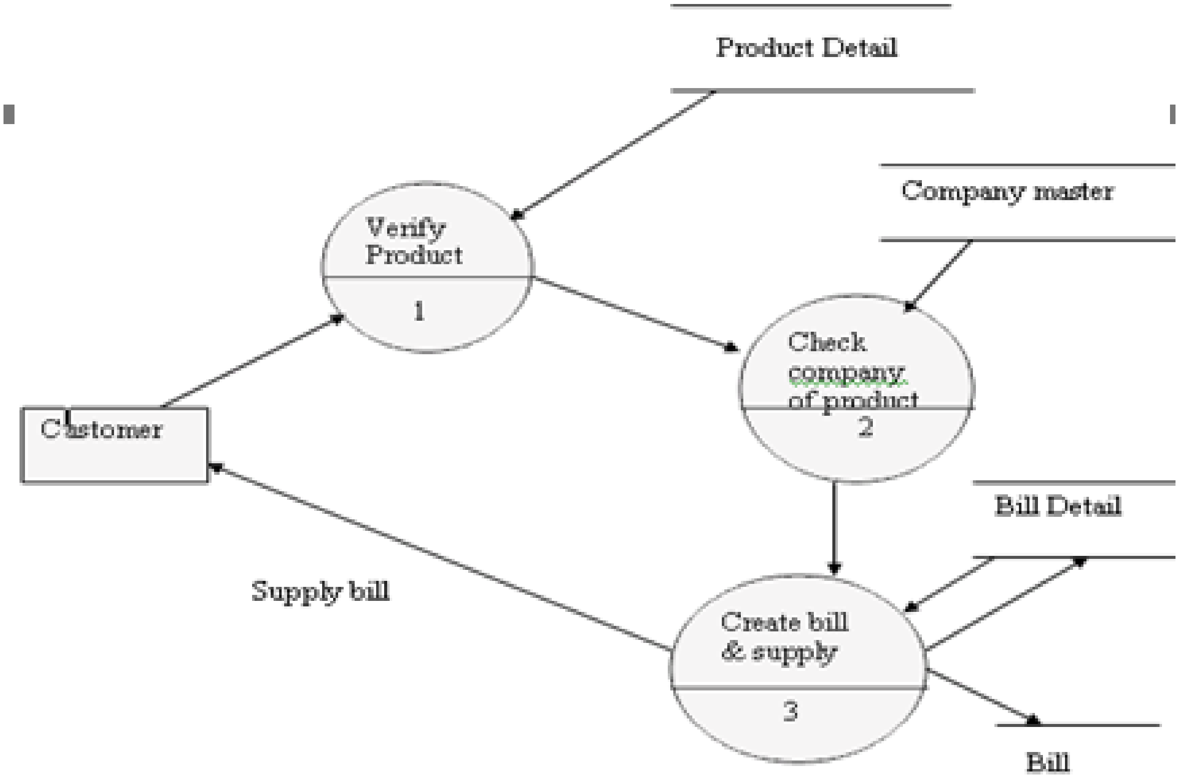


Fig 3 DFD Of ebilling and invoice Systems

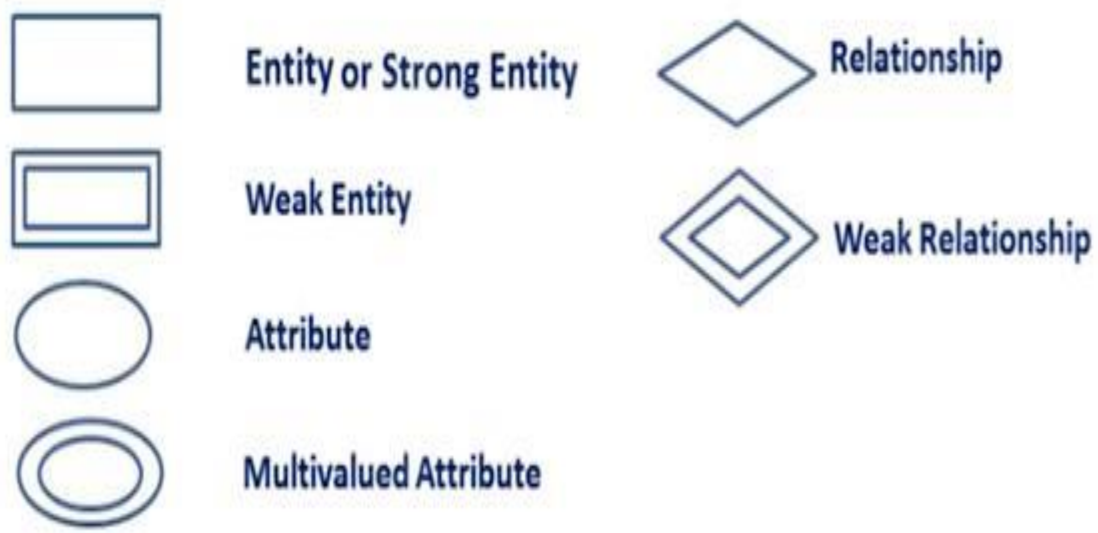
6.3 ENTITY RELATIONSHIP DIAGRAM:

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes and diamond shapes to represent relationships.

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

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



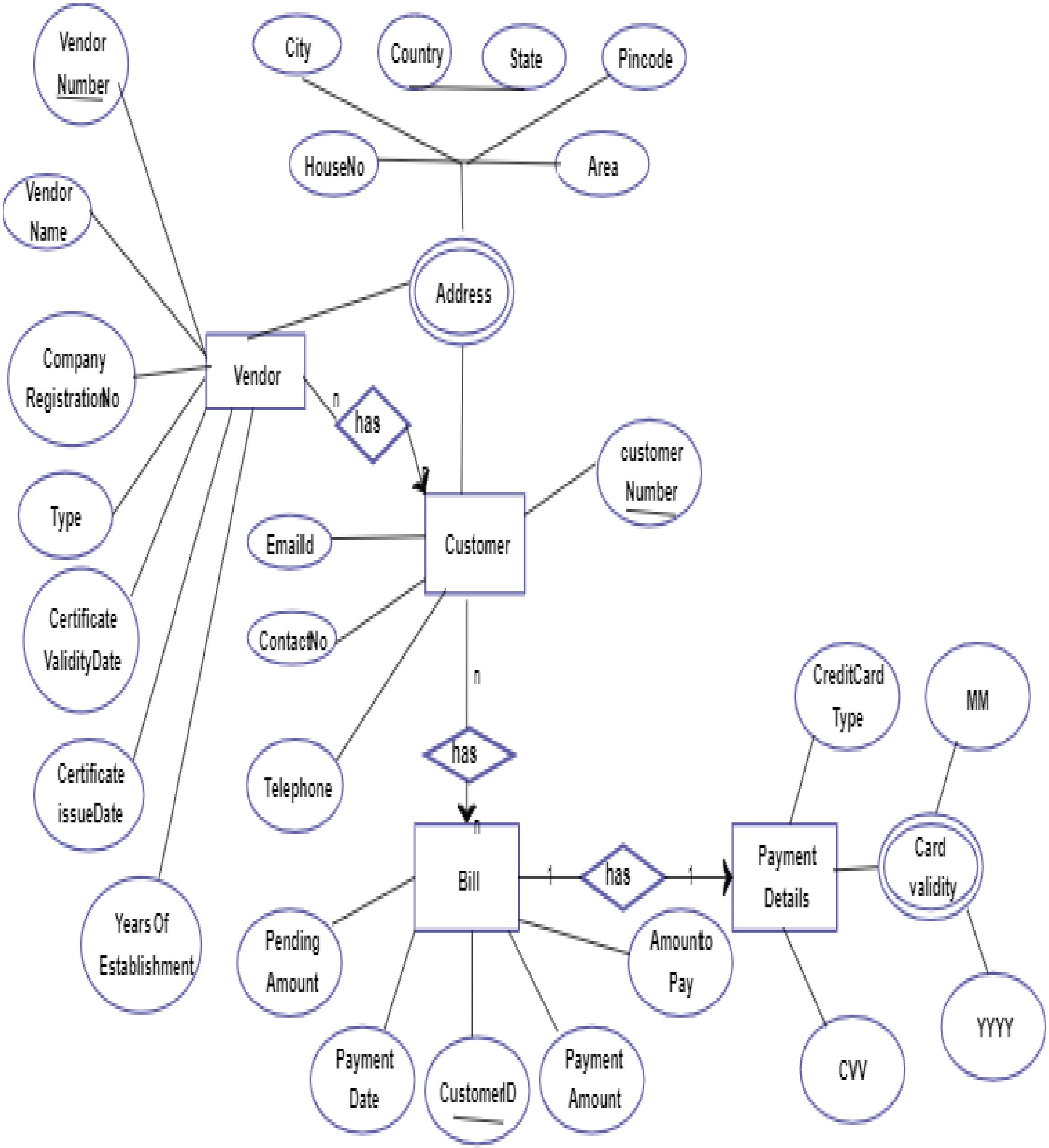


Fig 4 ER Diagram of Ebilling and Invoice System

6.4 Use Case Diagram

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.

Use Case Diagram objects

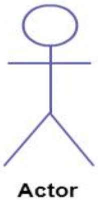
Use case diagrams consist of 4 objects.

- Actor
- Use case
- System
- Package

The objects are further explained below.

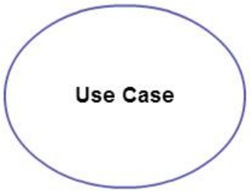
Actor

Actor in a use case diagram is **any entity that performs a role** in one given system. This could be a person, organization or an external system and usually drawn like skeleton shown below**.**



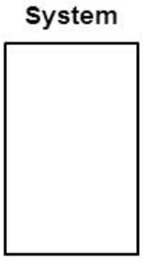
Use Case

A use case **represents a function or an action within the system**. It’s drawn as an oval and named with the function.



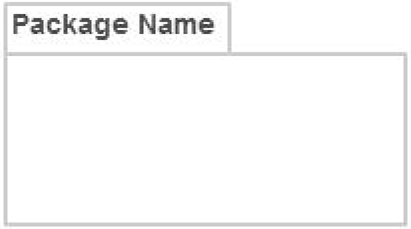
System

The system is used to **define the scope of the use case** and drawn as a rectangle. This an optional element but useful when you’re visualizing large systems. For example, you can create all the use cases and then use the system object to define the scope covered by your project. Or you can even use it to show the different areas covered in different releases.



Package

The package is another optional element that is extremely useful in complex diagrams. Similar to class diagrams, packages are **used to group together use cases**. They are drawn like the image shown below.



Use Case Diagram Guidelines

Although use case diagrams can be used for various purposes there are some common guidelines you need to follow when drawing use cases.

These include naming standards, directions of arrows, the placing of use cases, usage of system boxes and also proper usage of relationships.

We’ve covered these guidelines in detail in a separate blog post. So go ahead and check out use case diagram guidelines.

6.5 Relationships in Use Case Diagrams

There are five types of relationships in a use case diagram. They are

- Association between an actor and a use case
- Generalization of an actor
- Extend relationship between two use cases
- Include relationship between two use cases
- Generalization of a use case

We have covered all these relationships in a separate blog post that has examples with images.

We will not go into detail in this post but you can check out relationships in use case diagrams.

6.6 Usage Summary

eBilling and Invoicing System Version 1.0 will address the following use cases. The complete usage scenarios will be completed during the information-gathering process. Use cases will be created and prioritized.

A project summary is a document that contains a concise, comprehensive overview of an entire project and its key details.

It usually consists of a project's objectives, background information, requirements, problems, analysis, and conclusion.

Selected use cases will be expanded into usage scenarios and features that are derived from both use cases and the usage scenarios, as represented in the following diagram:

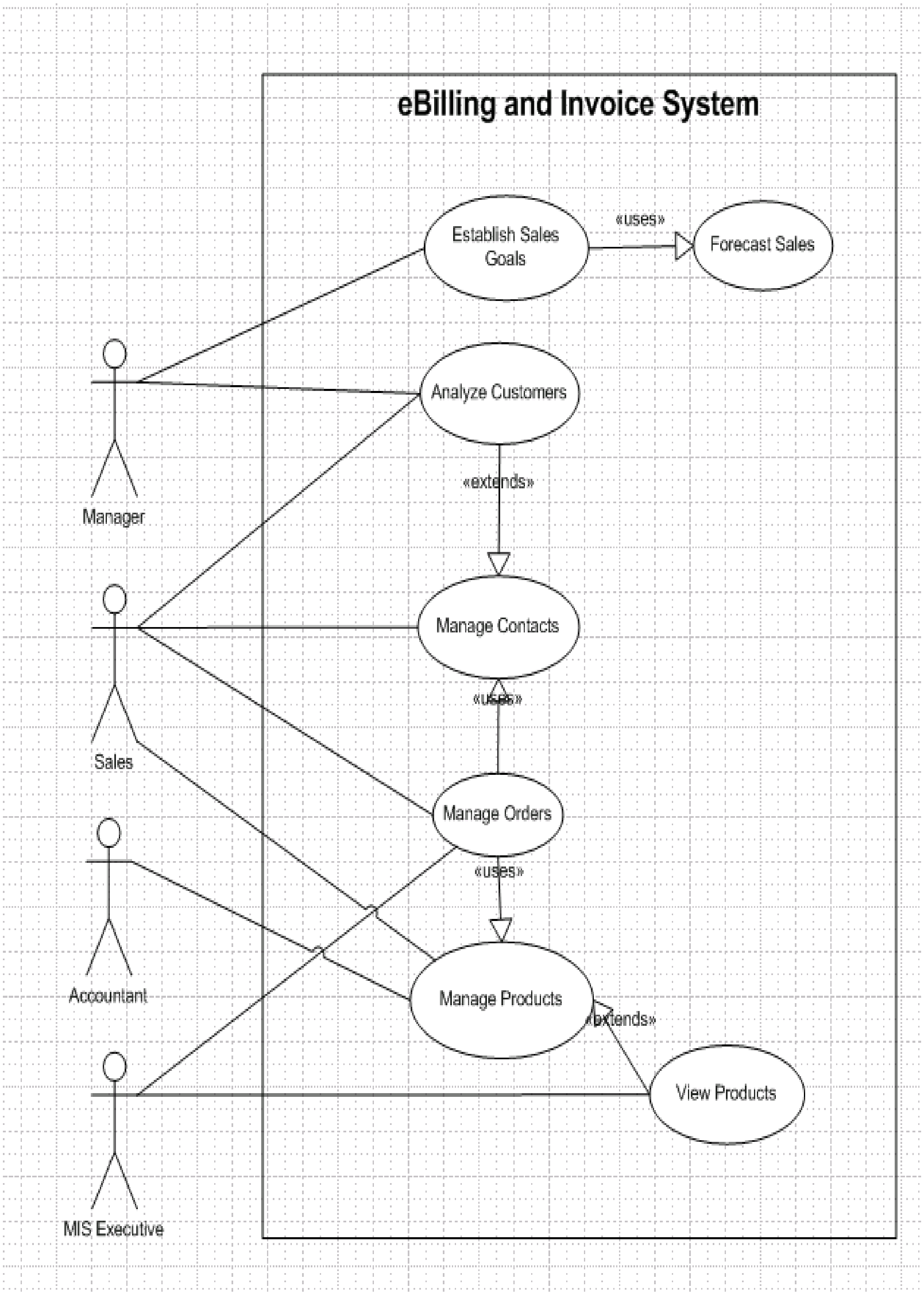


Fig 5 Usecase Diagram

Use Case Model: Retrieve Customer Data

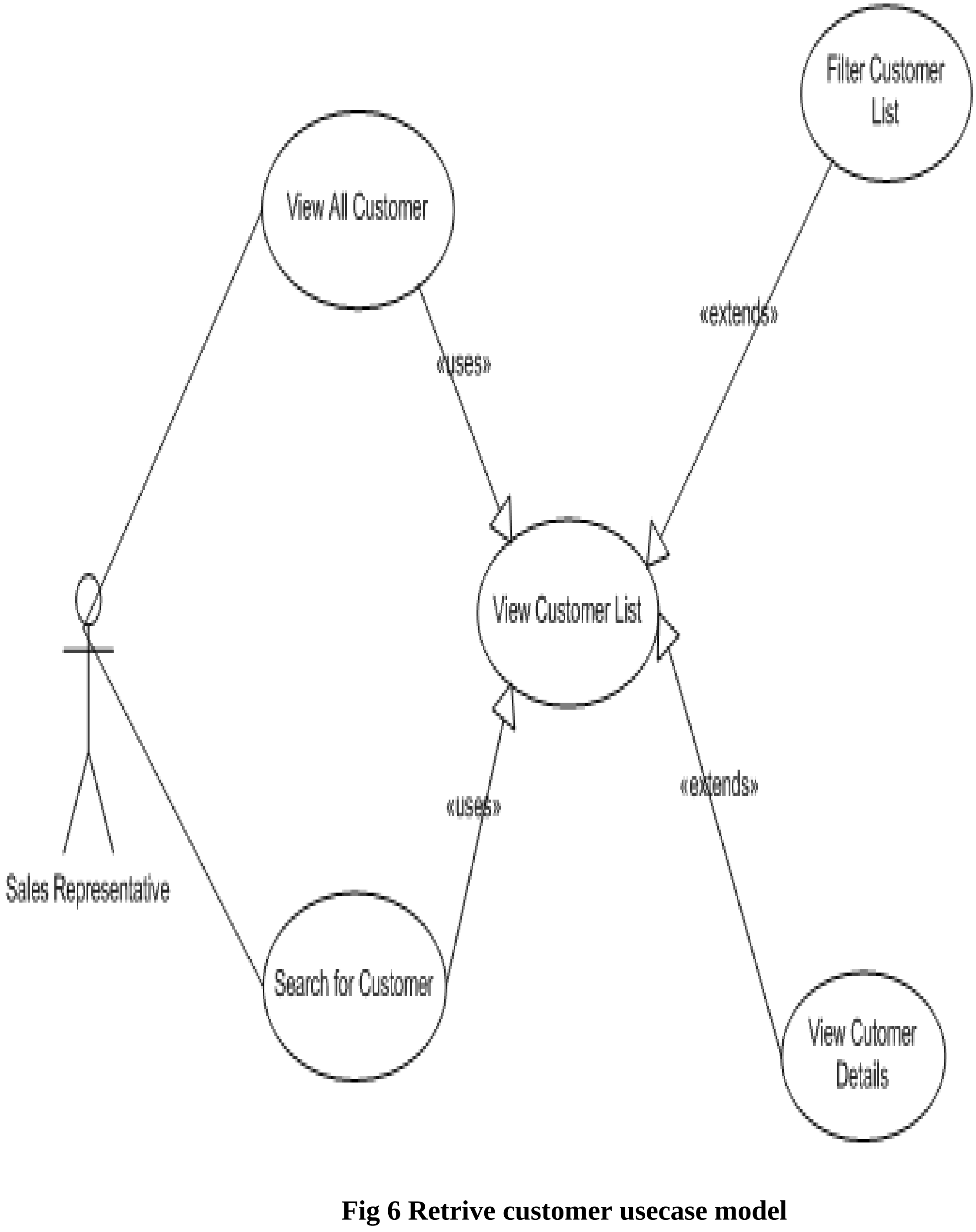


Fig 6 Retrive customer usecase model

Use Case: Manage Orders

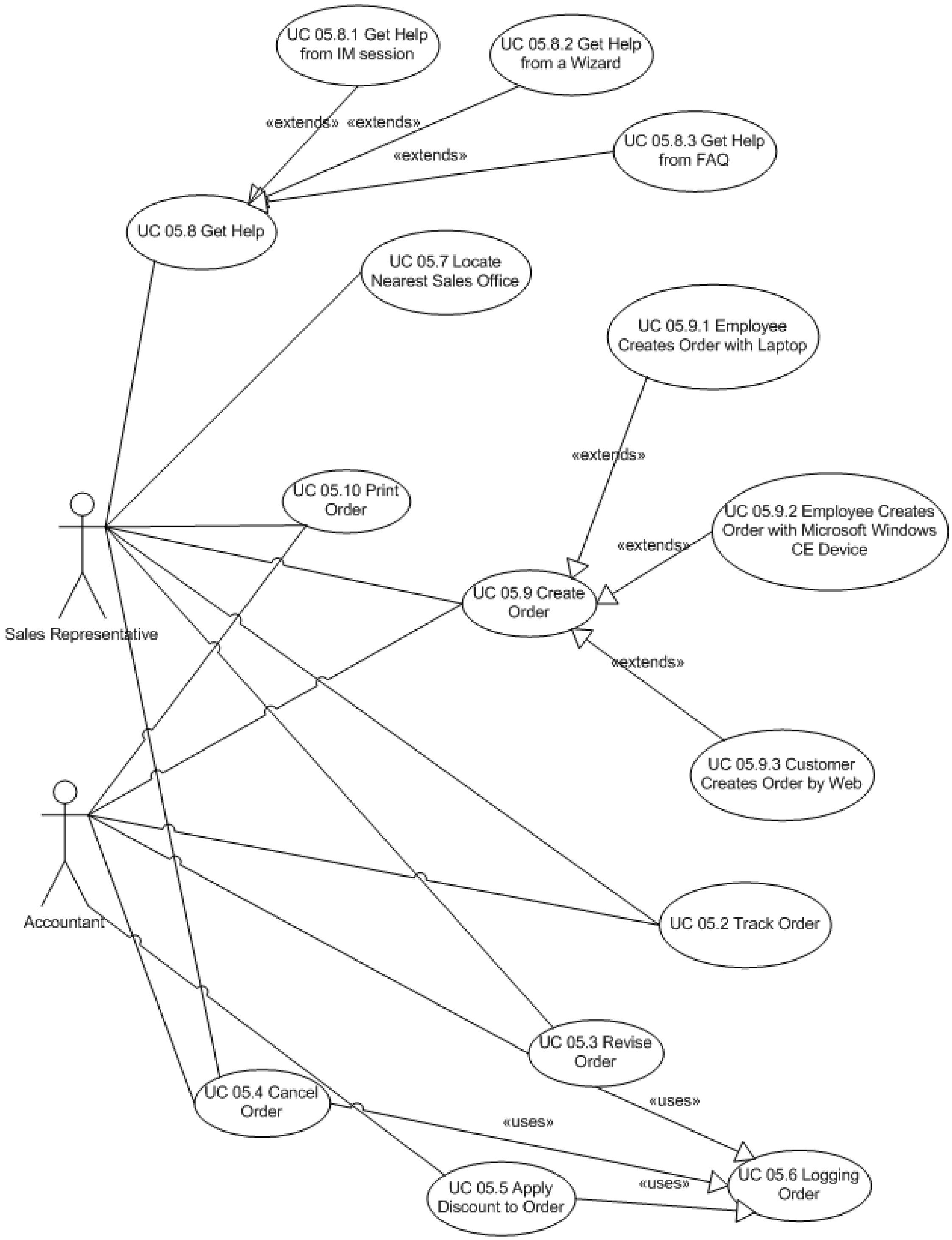

UC 05.8.1 Get Help from IM session
UC 05.8.2 Get Help from a Wizard
UC 05.8.3 Get Help from FAQ
«extends»
«extends»
«extends»
UC 05.8 Get Help
UC 05.7 Locate Nearest Sales Office
UC 05.9.1 Employee Creates Order with Laptop
«extends»
UC 05.10 Print Order
UC 05.9.2 Employee Creates Order with Microsoft Windows CE Device
«extends»
UC 05.9 Create Order
«extends»
UC 05.9.3 Customer Creates Order by Web
Sales Representative
Accountant
UC 05.2 Track Order
UC 05.3 Revise Order
«uses»
UC 05.4 Cancel Order
«uses»
«uses»
UC 05.5 Apply Discount to Order
UC 05.6 Logging Order

Fig 7 Manage Orders

6.7 Usage Scenarios

Usage scenarios that describe how users will use the solution within the context of the business, See following usage scenarios.

Use Case ID: Retrieve Customer Data

Title:	UC Retrieve Customer Data
Abbreviated Title:	Retrieve Customer Data
Requirement ID:	21

Intent

Return the selected information about one or more customers.

Scenario Narrative

A sales representative may want to search for and retrieve information about one or more customers for use in analysis and forecasting activities.

Assumptions/Preconditions

1. Sales representative has access rights to view customer data.

Actors

1. Sales Representative

Basic Course

1. Use case begins when the sales representative decides to retrieve customer information.
2. System prompts for the type of customer information that is required.
3. Sales representative selects the type of customer information to be retrieved.
4. System presents the appropriate customer information to the sales representative.
5. Use case ends when the selected information is passed to a method of delivery.

Alternate Course

1. If no appropriate customer information can be found, system reports this fact.
2. Use case restarts to enable sales representative to update the type of customer information.

Uses/Extends

1. None

User Implementation Requests

1. None

Frequency

Quite frequent

Authority

1. Not applicable

Issues

1. None identified

Decision Points

1. None

Future Requirements

1. The sales representative may want to apply filters to the retrieved data.
2. The sales representative may want to sort the data. For example, the sales representative may want to list the names of customers in chronological order based on the sale date.

Use Case ID: UC 04.7.3 Manage Contacts

Without any form of contact management, important contact data can get lost in many areas of your business. Either the data is lost deep down in an employee’s e-mail inbox which you cannot access, or the data is on a business card that can easily get lost in the office.

Title:	UC 04.7.2 Manage Contacts
Abbreviated Title:	Manage Contacts
Requirement ID:	19

Intent

Enable actor to create, view, modify of the information about customer contacts.

Scenario Narrative

An actor wants to create view, modify, and delete the customer contacts.

Assumptions/Preconditions

- The actor has rights to access to the eBilling and Invoice System, and having rights to view, ads, edit and delete the customer contacts.

Actors

- Sales Representative
- Manager
- Accountant

Basic Course:

- Use case begins when actor decides to search customer.
- Actor searches the customer.
- System displays the customer details.
- Actor view the customer contact details:
- Actor add, modify the contact details.

Uses/Extends

1. None

User Implementation Requests

None

Frequency

Frequent

Authority

- Not applicable
- Issues
- Decision Points
- None
- Legitimacy
- Rationality
- Accountability
- An informal power

Future Requirements

- None

6.8 Application Management and Priority

6.8.1 Manage Orders

Description

This function will enable sales representative to place orders for eBilling and Invoice System products, track submitted orders, and view completed orders and print the invoice.

Business Need

This function will enable sales representative to interact with eBilling and Invoice System catalog directly without the intervention of any other employees.

Priority

High

6.8.2 Manage Products

Description

This function allows user of eBilling and Invoice System to create, maintain, and delete information about products in the product database that can then be viewed and ordered by customers, and accessed and updated by the sales staff.

Business Need

This function will support the sales staff with accurate and relevant information, and will also support the Manage Orders function by ensuring up-to-date product information.

Priority

High

6.8.3 Manage Contacts

Description

This function enables the sales staff to create and manage contacts as well as share contact information with the rest of the company.

Business Need

This function enables user of eBilling and Invoice System to have accurate and up-to-date contact information when working with customers.

Priority

High

6.8.4 Analyze Customers

Description

This function allows management to analyze the customer database and find out information, such as the identities of the best customers, the top buyers, and the most popular products.

Business Need

This function will allow management to determine the most (and least) profitable customers and product lines, enabling better decision making in the running of the business.

Priority

Medium

6.8.5 Forecast Sales

Description

This function allows management to plan and track production costs, track sales results, and plan sales staffing needs.

Business Need

This function will allow management to manage the revenue streams of the organizations and control costs.

Priority

Medium

6.8.6 Establish Sales Goals

Description

This function allows management to view current sales trends and employee performance and to determine sales goals for the sales staff within the organization.

Business Need

This function will allow management to set goals for staff, review the performance of staff members, and track employee performance.

Priority

Medium

6.9 System Architecture

Infrastructure

New servers must be dual processor 1.26 GHz computers, with 1 GB of RAM. The operating system software will be Microsoft® Windows® 2000 with the latest service pack, although use of Windows Server 2003 will be considered when it is available.

These servers will be installed on existing network.

To help provide security of the application, both an internal and external firewall will be installed.

Visual Design

See later on the visual interface design.

Conceptual Design

For information about the conceptual design see the use case of previous chapters of the eBilling and Invoice System,

Business Rule Catalog of eBilling and Invoice System

Add new product to product master sequence diagram

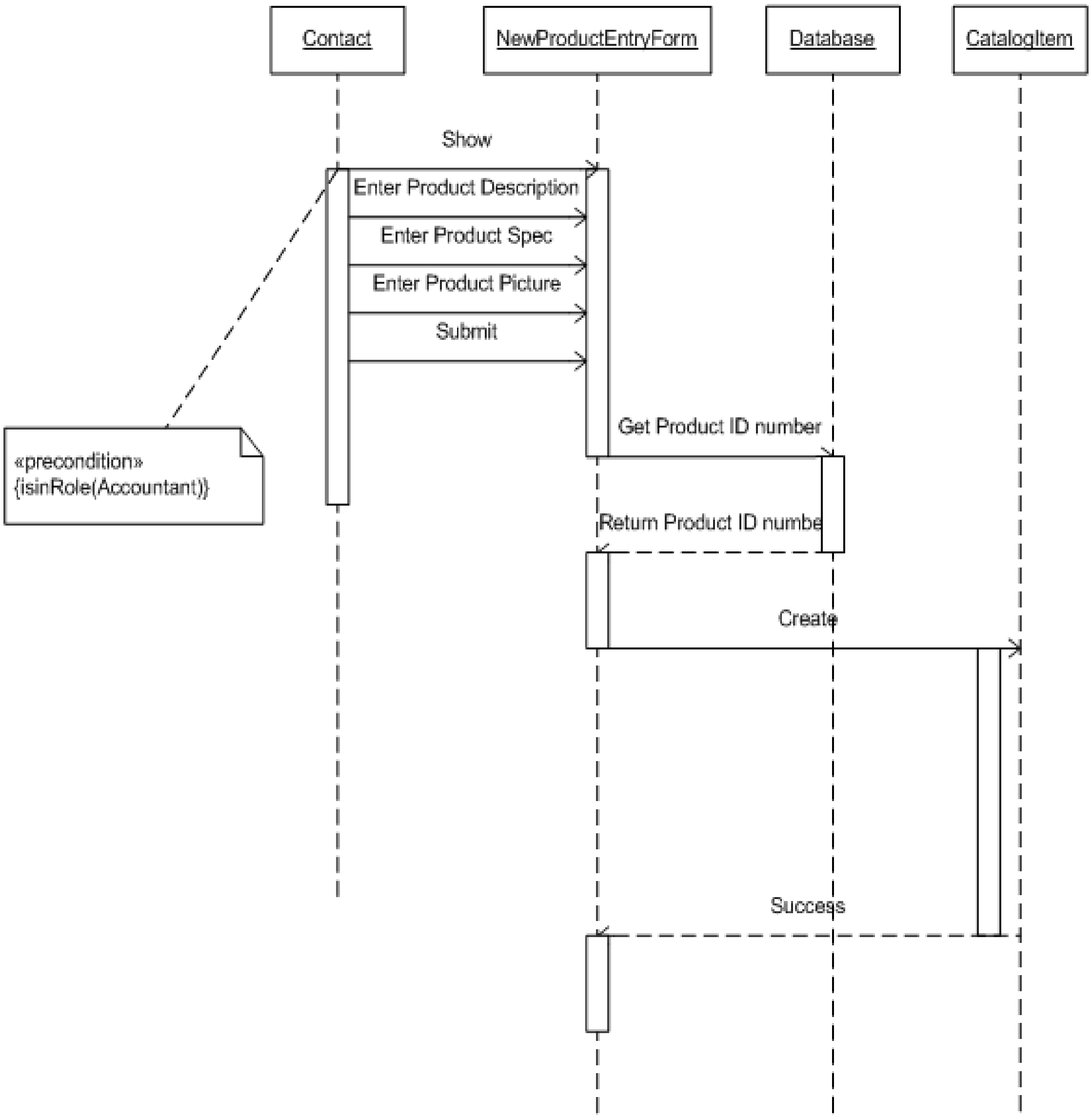


Fig8 Add new product to product master sequence diagram

6.10 Physical Design

eBilling and Invoice System Physical Object Model

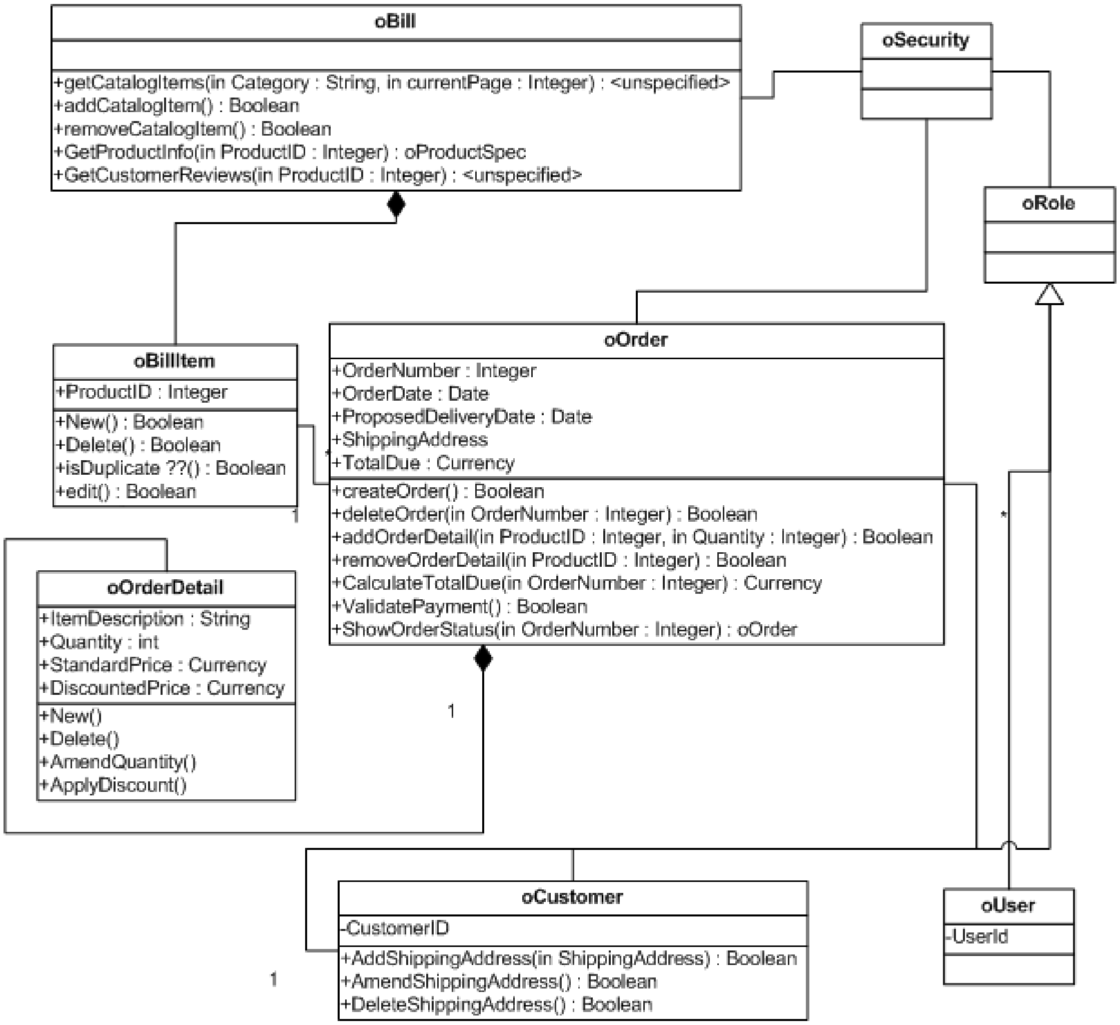

oBill
+getCatalogItems(in Category : String, in currentPage : Integer) : <unspecified>
+addCatalogItem() : Boolean
+removeCatalogItem() : Boolean
+GetProductInfo(in ProductID : Integer) : oProductSpec
+GetCustomerReviews(in ProductID : Integer) : <unspecified>
oSecurity
oRole
oBillItem
+ProductID : Integer
+New() : Boolean
+Delete() : Boolean
+isDuplicate ??() : Boolean
+edit() : Boolean
1
*
oOrder
+OrderNumber : Integer
+OrderDate : Date
+ProposedDeliveryDate : Date
+ShippingAddress
+TotalDue : Currency
+createOrder() : Boolean
+deleteOrder(in OrderNumber : Integer) : Boolean
+addOrderDetail(in ProductID : Integer, in Quantity : Integer) : Boolean
+removeOrderDetail(in ProductID : Integer) : Boolean
+CalculateTotalDue(in OrderNumber : Integer) : Currency
+ValidatePayment() : Boolean
+ShowOrderStatus(in OrderNumber : Integer) : oOrder
*
oOrderDetail
+ItemDescription : String
+Quantity : int
+StandardPrice : Currency
+DiscountedPrice : Currency
+New()
+Delete()
+AmendQuantity()
+ApplyDiscount()
1
oCustomer
-CustomerID
+AddShippingAddress(in ShippingAddress) : Boolean
+AmendShippingAddress() : Boolean
+DeleteShippingAddress() : Boolean
1
oUser
-UserId

Fig 9 eBilling and Invoice System Physical Object Model

Chapter 7

DataStructure Of eBilling and Invoice System

eBilling and Invoice System Data Dictionary

Table: product master

Owner:	dbo
Destination DB name:	ebill
Number of columns:	5
Number of indexes:	2
Number of foreign keys:	0
Extended attributes:	
On Filegroup	PRIMARY
Clustered PK	Yes

Table 3 ebilling and invoice System

Columns	Data type	Allow NULLs	Value/range
Sno	int	Not allowed	
ProdType	nvarchar(60)	Not allowed	
ProdSubType	nvarchar(60)	Not allowed	
Rate	int	Not allowed	
isactive	nvarchar(30)	Not allowed	

Table 4 Datastructure of Product table

Indexes	Type	Columns
PK_ProductMaster_sno	clustered, unique, primary key located on PRIMARY	sno

Table 5 Datastructure of Product Master Table

Table: bill

Owner:	dbo
Destination DB name:	ebill
Number of columns:	15
Number of indexes:	3
Number of foreign keys:	0
Extended attributes:	
OnFileGroup	PRIMARY
Clustered PK	Yes

Columns	Data type	Allow NULLs	Value/range
sno	int	Not allowed	
invoiceno	nvarchar(60)	Not allowed	
canme	nvarchar(60)	Not allowed	
Invoice_date	datetime	Not allowed	
location	nvarchar(60)	Not allowed	
chalan_no	int	Not allowed	
customer_name	nvarchar(100)	Not allowed	
Customer_address1	nvarchar(100)	Not allowed	
Customer_address2	nvarchar(100)	Allowed	
Total_amt	int	Not allowed	
discount	int	Not allowed	
Amt_words	nvarchar(100)	Not allowed	
Paid_type	nvarchar(60)	Not allowed	
Cheque_no	nvarchar(60)	Allowed	
Entry_date	datetime	Not allowed	

Table 6 Datastructure of Customer Table

Indexes	Type	Columns
PK_Bill_SNO	clustered, unique, primary key located on PRIMARY	sno
IX_bill_cname	nonclustered located on PRIMARY	cname
IX_bill_invoiceno	nonclustered, unique located on PRIMARY	invoiceno

Table: bill details

Owner:	dbo
Destination DB name:	ebill
Number of columns:	6
Number of indexes:	3
Number of foreign keys:	2
Extended attributes:	
OnFileGroup	PRIMARY
Clustered PK	Yes

Table 7 DataStructure of Owner Table

Columns	Data type	Allow NULLs	Value/range
sno	int	Not allowed	
Bill_sno	int	Not allowed	
Prod_sno	nvarchar(60)	Not allowed	
qty	datetime	Not allowed	
rate	nvarchar(60)	Not allowed	
amt	int	Not allowed	

Table 8 Datastructure of Billing table

Indexes	Type	Columns
PK_Bill_details_SNO	clustered, unique, primary key located on PRIMARY	sno
IX_bill_details_bill_sno	nonclustered located on PRIMARY	Bill_sno
IX_bill_deatils_prod_sno	nonclustered, unique located on PRIMARY	Prod_sno

Table 9 Datastructure of Billing index

Table Company Master

Owner:	dbo
Destination DB name:	ebill
Number of columns:	7
Number of indexes:	2

Number of foreign keys: 0

Extended attributes:

OnFileGroup PRIMARY

Clustered PK Yes

Columns	Data type	Allow NULLs	Value/range
Company_name	nvarchar(60)	Not allowed	
Address1	nvarchar(100)	Not allowed	
Address2	nvarchar(100)	Allowed	
city	nvarchar(60)	Not allowed	
pin	int	Not allowed	
telephone	nvarchar(60)	Not allowed	
vatno	nvarchar(60)	Not allowed	

Table 10 DataStructure of Employee Table

Indexes	Type	Columns
PK_comapny_master_company_name	clustered, unique, primary key located on PRIMARY	Comapnay_name
IX_ comapny_master_pin	nonclustered located on PRIMARY	pin

Table:

Owner:	dbo
Destination DB name:	ebill
Number of columns:	3
Number of indexes:	2
Number of foreign keys:	0
Extended attributes:	
OnFileGroup	PRIMARY
Clustered PK	Yes

Table 11 Datastructure of employee Master Table

Columns	Data type	Allow NULLs	Value/range
user_name	nvarchar(60)	Not allowed	
User_password	nvarchar(60)	Not allowed	
User_type	nvarchar(60)	Allowed	

Table 12 User Table

Indexes	Type	Columns
PK_user_master_user_name	clustered, unique, primary key located on PRIMARY	user_name
IX_ user_master_user_type	nonclustered located on PRIMARY	User_type

Table 13 Master User Table

Chapter 8

ScreenShots

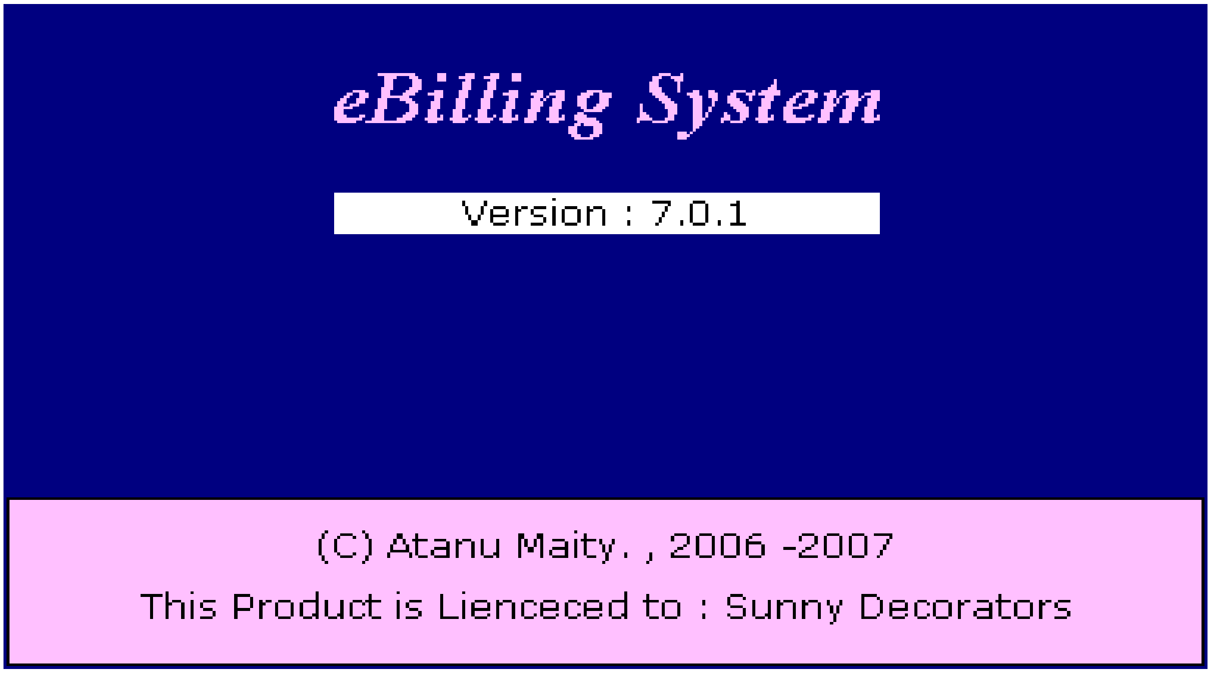
Form Design and Coding

FormSplash

(Module to show startup screen)

Electronic billing or e-billing is **the process by which bills are sent and paid electronically**. This process enables customers to receive bills via email, web portal or even in machine-readable data formats, facilitating more efficient delivery and payment.

This Splash is basically a startup screen which shows only name and version of this copyright software. Which is delivered to various firm,store,organigation and updation of this software is in android applications which will be mob friendly.



eBilling System

Version : 7.0.1

(C) Atanu Maity. , 2006 -2007

This Product is Lienceced to : Sunny Decorators

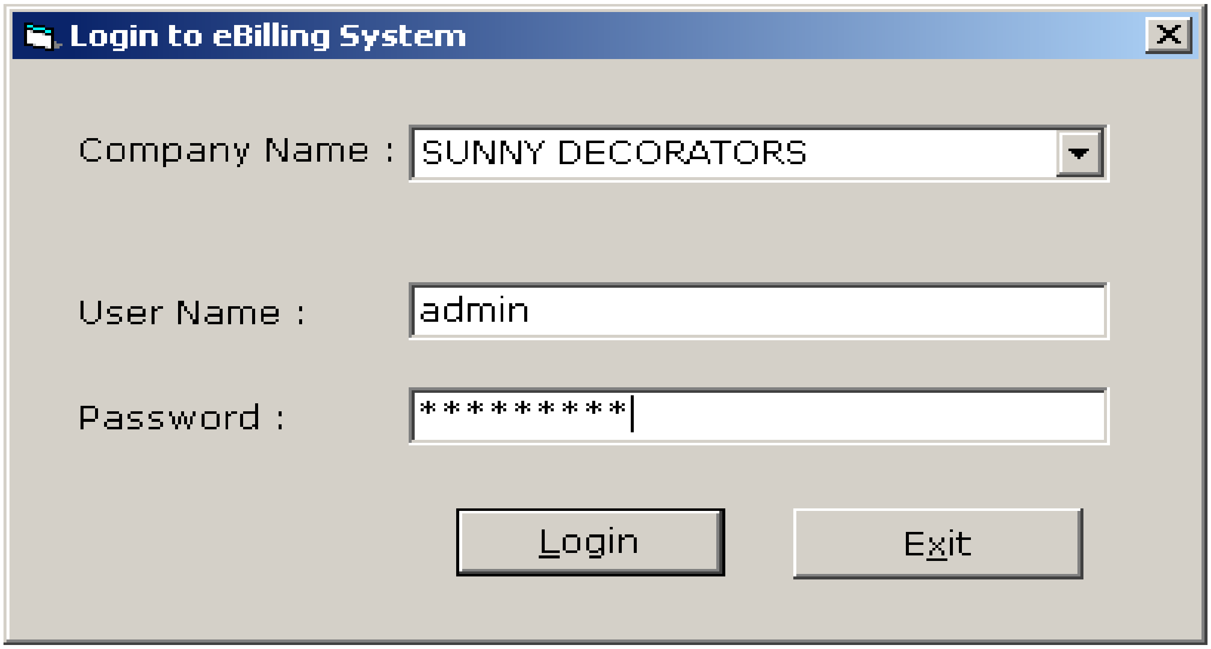
Fig 10 Main Page

FormLogin

(Module to Authenticate User)

A login page is **a web page or an entry page to a website that requires user identification and authentication, regularly performed by entering a username and password combination**.

This Android application is more focused on security privacy integrity of the system.



Login to eBilling System

Company Name : SUNNY DECORATORS

User Name : admin

Password : *********

Fig 11 Login Page

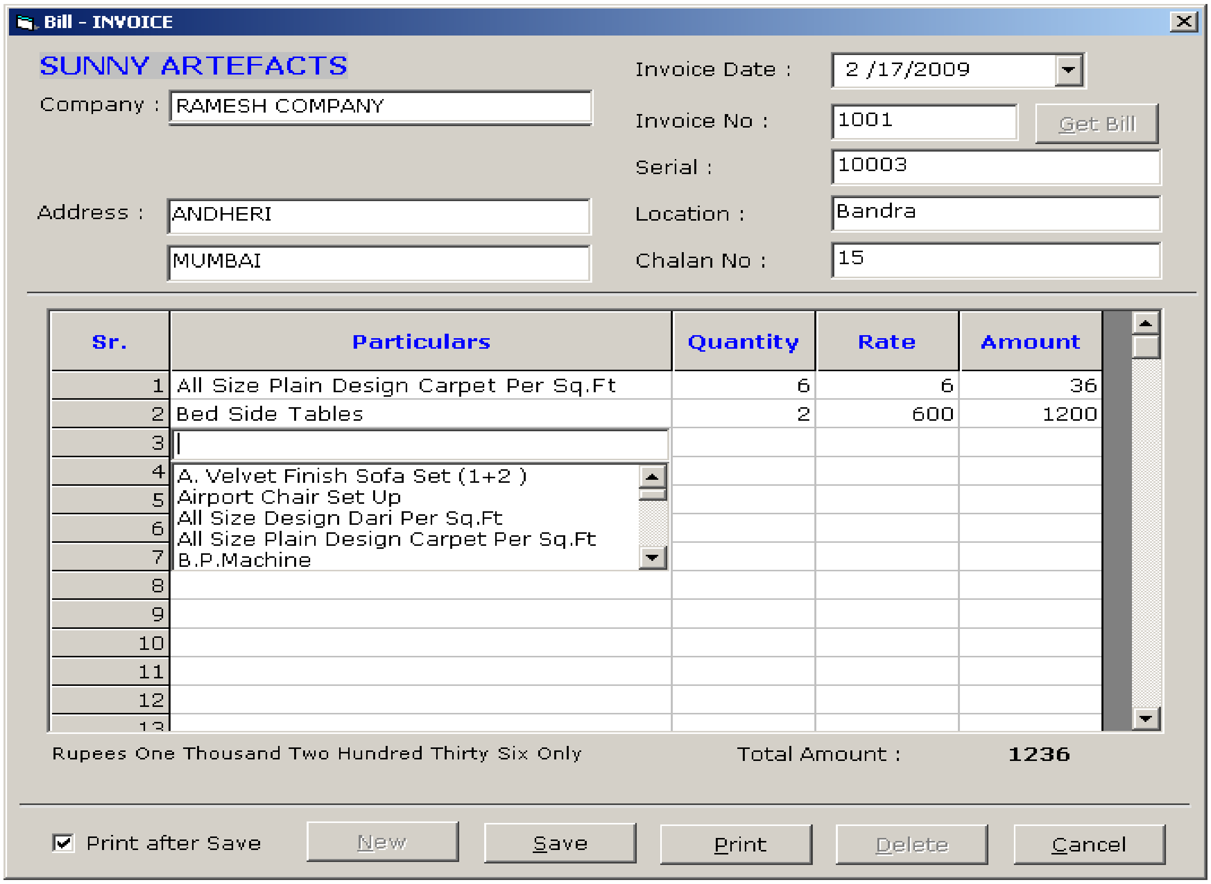
FormProducts

(Maintain Product master)

A product master is a data management solution that compiles, validates, enriches, and curates all your organization's product-related data into a complete, accurate, and easily reportable golden copy.

FormBill

(Create and modify bill and print the saved bill)



Bill - INVOICE

SUNNY ARTEFACTS

Company : RAMESH COMPANY

Address : ANDHERI
MUMBAI

Invoice Date : 2 /17/2009

Invoice No : 1001

Serial : 10003

Location : Bandra

Chalan No : 15

Sr.	Particulars	Quantity	Rate	Amount
1	All Size Plain Design Carpet Per Sq.Ft	6	6	36
2	Bed Side Tables	2	600	1200
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				

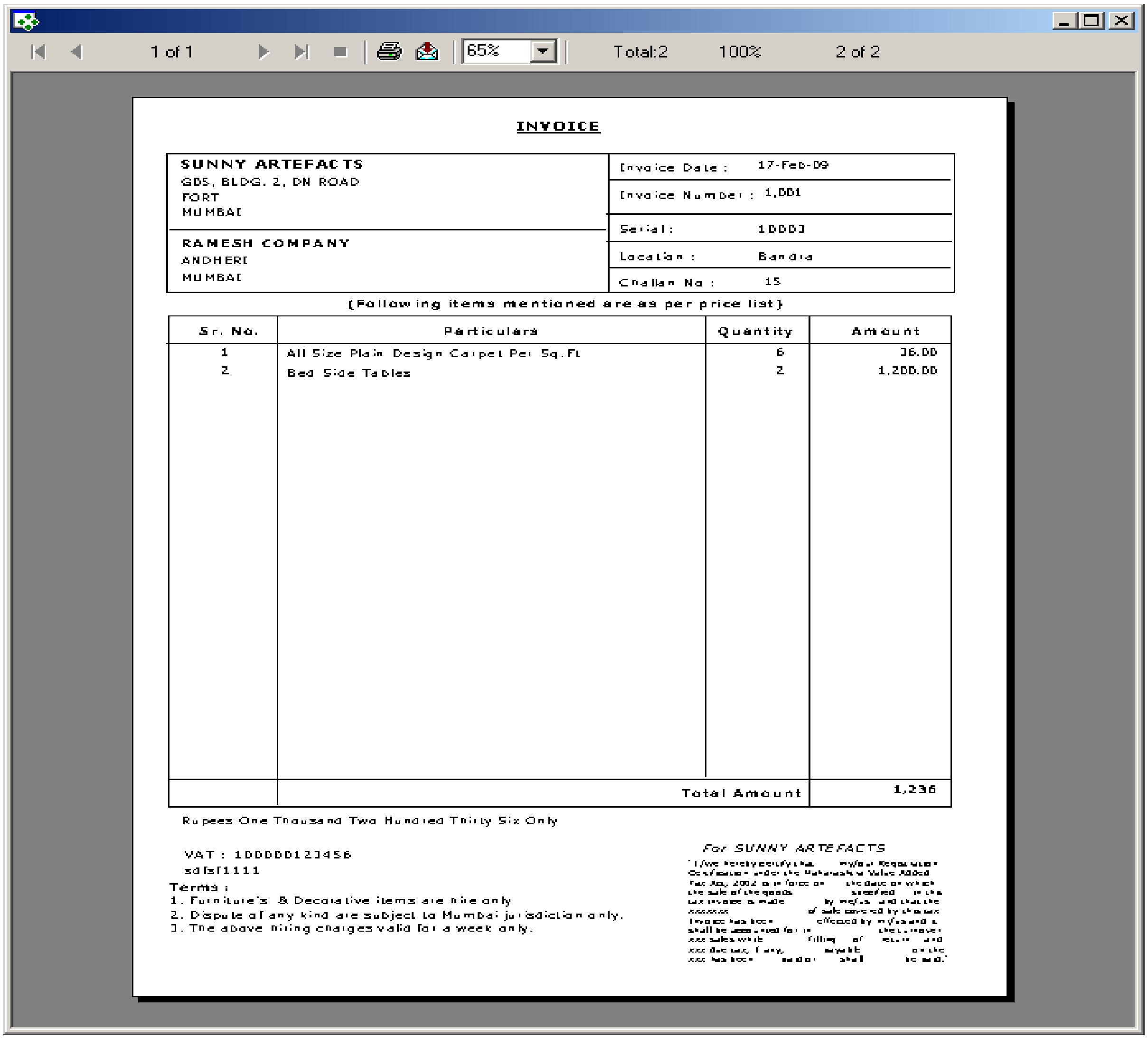
A. Velvet Finish Sofa Set (1+2)
Airport Chair Set Up
All Size Design Dari Per Sq.Ft
All Size Plain Design Carpet Per Sq.Ft
B.P.Machine

Rupees One Thousand Two Hundred Thirty Six Only

Total Amount : 1236

☑ Print after Save

Fig 12 Bill Invoice page



INVOICE

SUNNY ARTEFACTS
GDS, BLDG. 2, DN ROAD
FORT
MUMBAI

RAMESH COMPANY
ANDHERI
MUMBAI

Invoice Date :	17-Feb-09
Invoice Number :	1,001
Serial :	10003
Location :	Bandra
Challan No :	15

(Following items mentioned are as per price list)

Sr. No.	Particulars	Quantity	Amount
1	All Size Plain Design Carpet Per Sq.Ft	6	36.00
2	Bed Side Tables	2	1,200.00
	Total Amount		1,236

Rupees One Thousand Two Hundred Thirty Six Only

VAT : 100000123456
sdfsf1111

Terms :
1. Furniture's & Decorative items are hire only
2. Dispute of any kind are subject to Mumbai jurisdiction only.
3. The above hiring charges valid for a week only.

For SUNNY ARTEFACTS

"I/we hereby certify that my/our Registration Certificate under the Maharashtra Value Added Tax Act, 2002 is in force on the date on which the sale of the goods specified in this tax invoice is made by me/us and that the xxxxxx of sale covered by this tax invoice has been effected by me/us and it shall be accounted for in the turnover xxx sales while filling of return and xxx due tax, if any, payable on the xxx has been paid or shall be paid."

Fig 13 Print Preview Of Bill

FormReports

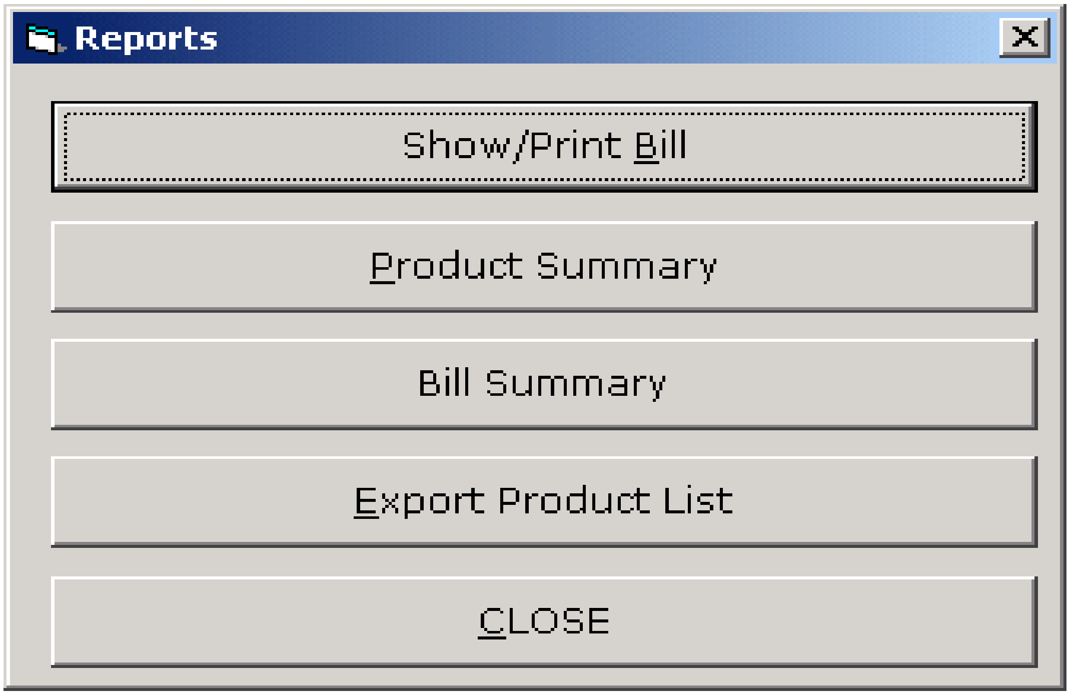
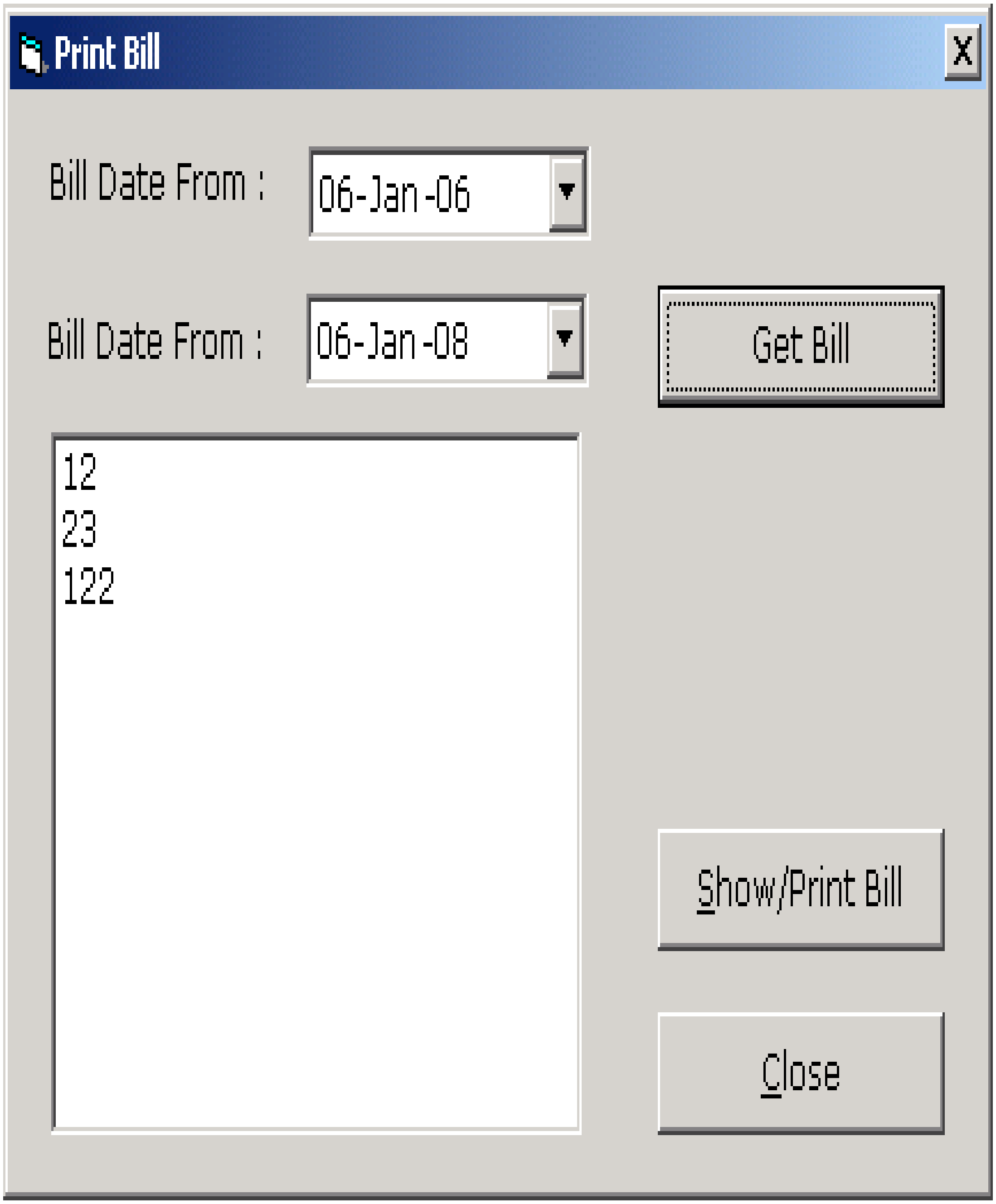


Fig 14 Different Reports

FormPrintBill

(Print already saved bill)



Bill Date From : 06-Jan-06

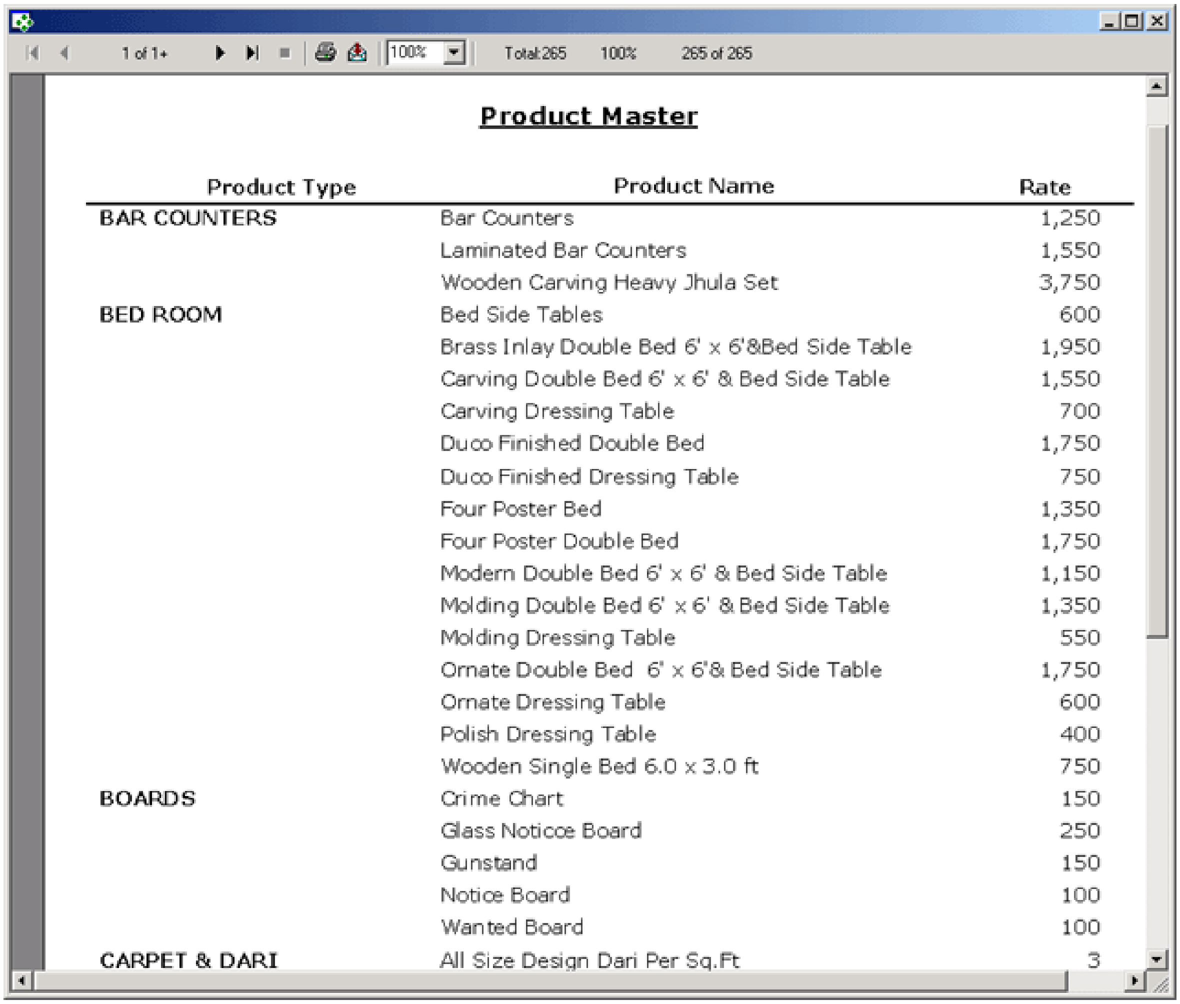
Bill Date From : 06-Jan-08

12
23
122

Fig 15 Print Bill Summary

Product Summary Report

(Show Product Summary report in crystal report)



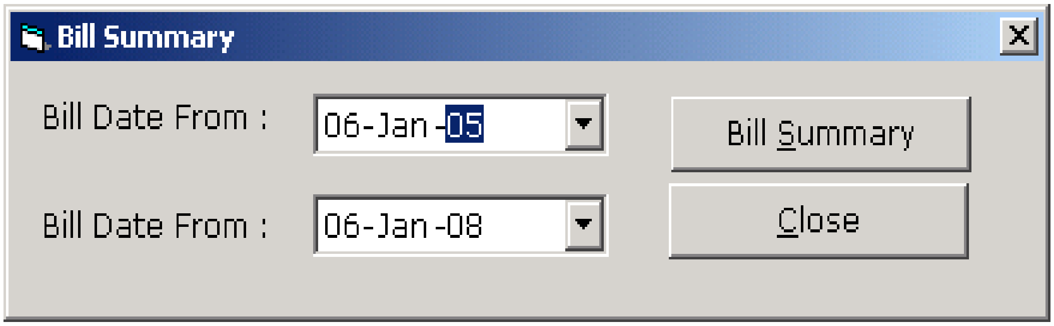
Product Master

Product Type	Product Name	Rate
BAR COUNTERS	Bar Counters	1,250
	Laminated Bar Counters	1,550
	Wooden Carving Heavy Jhula Set	3,750
BED ROOM	Bed Side Tables	600
	Brass Inlay Double Bed 6' x 6'&Bed Side Table	1,950
	Carving Double Bed 6' x 6' & Bed Side Table	1,550
	Carving Dressing Table	700
	Duco Finished Double Bed	1,750
	Duco Finished Dressing Table	750
	Four Poster Bed	1,350
	Four Poster Double Bed	1,750
	Modern Double Bed 6' x 6' & Bed Side Table	1,150
	Molding Double Bed 6' x 6' & Bed Side Table	1,350
	Molding Dressing Table	550
	Ornate Double Bed 6' x 6'& Bed Side Table	1,750
	Ornate Dressing Table	600
	Polish Dressing Table	400
	Wooden Single Bed 6.0 x 3.0 ft	750
BOARDS	Crime Chart	150
	Glass Noticce Board	250
	Gunstand	150
	Notice Board	100
	Wanted Board	100
CARPET & DARI	All Size Design Dari Per Sq.Ft	3

Fig 16 Product Summary

FormBillSummary

(Show bill summary for particular date range)



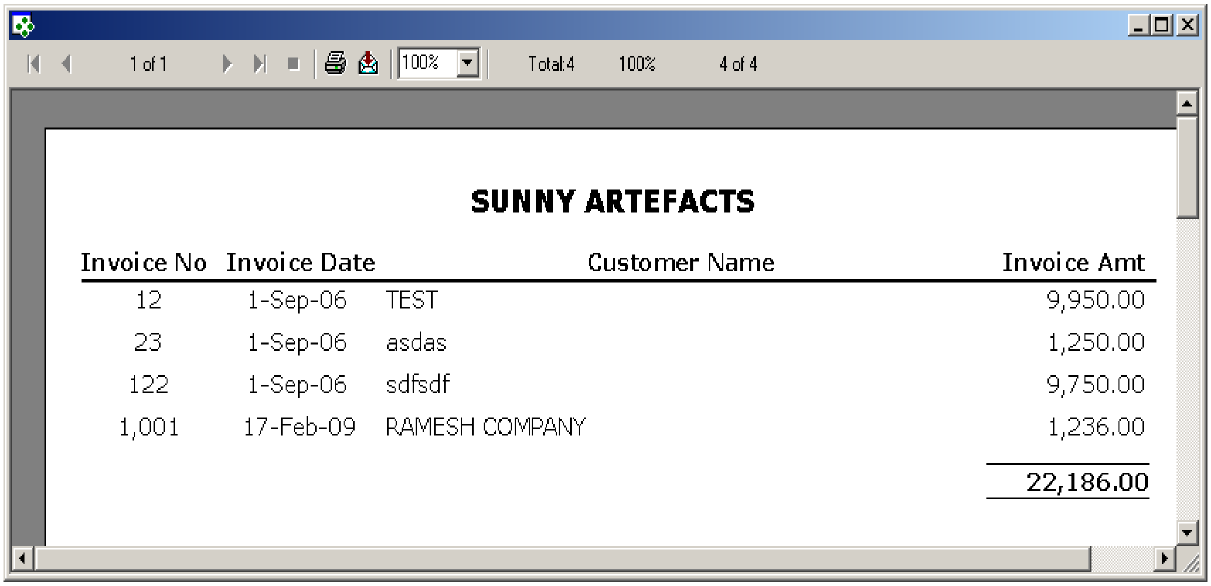
Bill Summary

Bill Date From : 06-Jan-05

Bill Date From : 06-Jan-08

Fig 17 Summary of Bill

Bill Summary Report



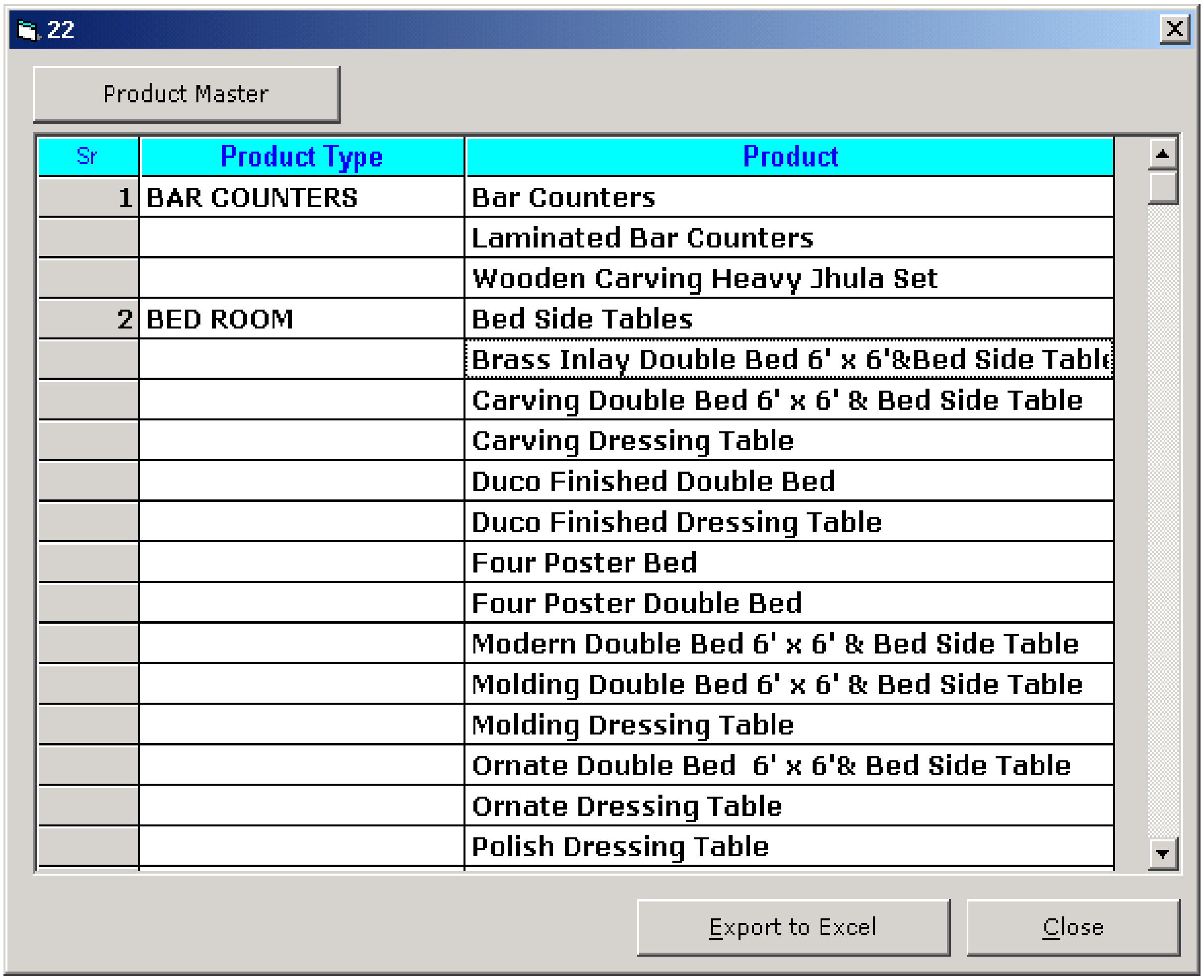
SUNNY ARTEFACTS

Invoice No	Invoice Date	Customer Name	Invoice Amt
12	1-Sep-06	TEST	9,950.00
23	1-Sep-06	asdas	1,250.00
122	1-Sep-06	sdfsdf	9,750.00
1,001	17-Feb-09	RAMESH COMPANY	1,236.00
			22,186.00

Fig 18 Bill and Report Summary

FormExportData

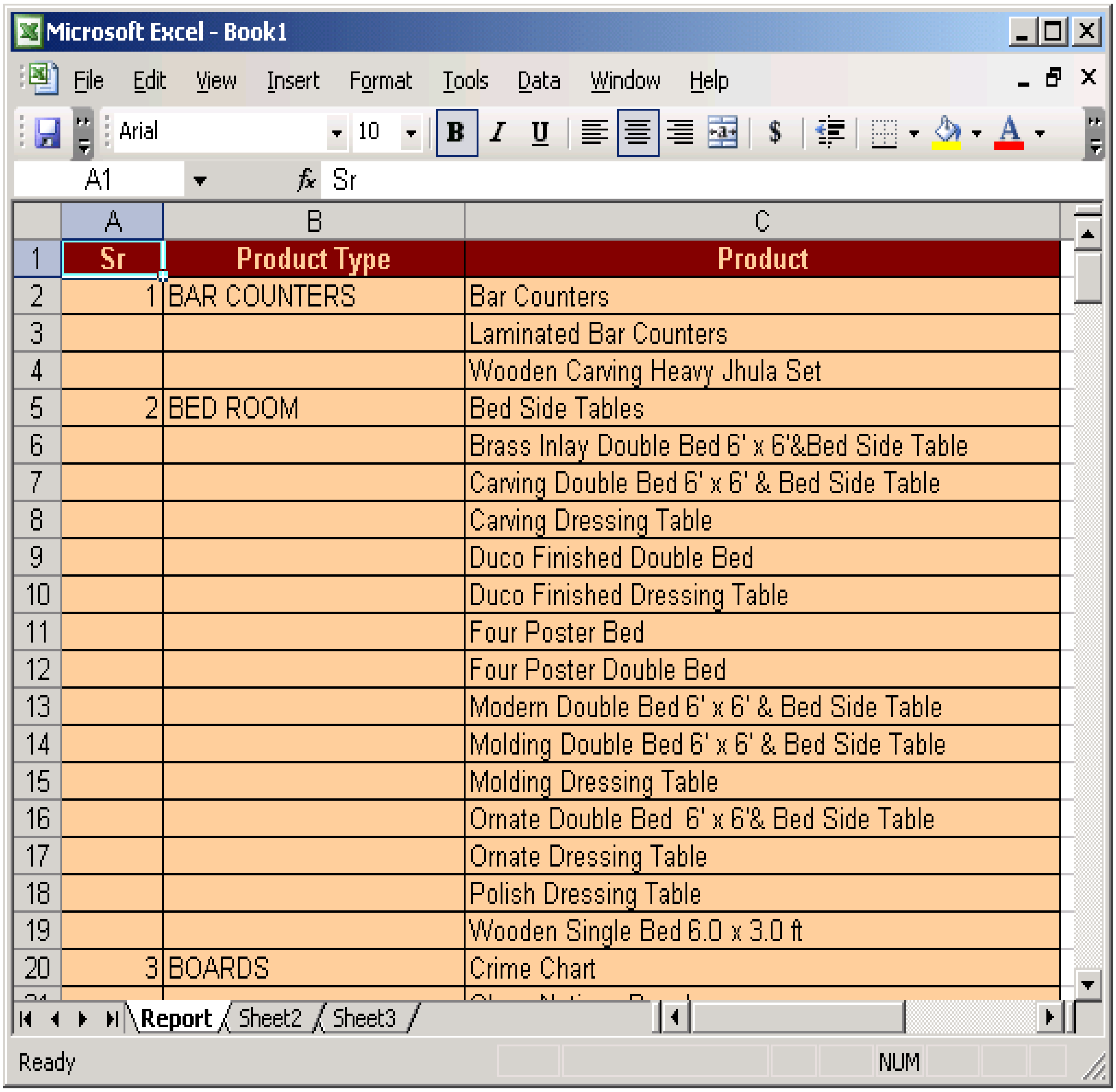
(Export product Data in FlexGrid and MS-Excel with formatting)



Sr	Product Type	Product
1	BAR COUNTERS	Bar Counters
		Laminated Bar Counters
		Wooden Carving Heavy Jhula Set
2	BED ROOM	Bed Side Tables
		Brass Inlay Double Bed 6' x 6'&Bed Side Tabl
		Carving Double Bed 6' x 6' & Bed Side Table
		Carving Dressing Table
		Duco Finished Double Bed
		Duco Finished Dressing Table
		Four Poster Bed
		Four Poster Double Bed
		Modern Double Bed 6' x 6' & Bed Side Table
		Molding Double Bed 6' x 6' & Bed Side Table
		Molding Dressing Table
		Ornate Double Bed 6' x 6'& Bed Side Table
		Ornate Dressing Table
		Polish Dressing Table

Fig 19 Export Data

Export product Data MS-Excel with formatting



	A	B	C
1	Sr	Product Type	Product
2	1	BAR COUNTERS	Bar Counters
3			Laminated Bar Counters
4			Wooden Carving Heavy Jhula Set
5	2	BED ROOM	Bed Side Tables
6			Brass Inlay Double Bed 6' x 6'&Bed Side Table
7			Carving Double Bed 6' x 6' & Bed Side Table
8			Carving Dressing Table
9			Duco Finished Double Bed
10			Duco Finished Dressing Table
11			Four Poster Bed
12			Four Poster Double Bed
13			Modern Double Bed 6' x 6' & Bed Side Table
14			Molding Double Bed 6' x 6' & Bed Side Table
15			Molding Dressing Table
16			Ornate Double Bed 6' x 6'& Bed Side Table
17			Ornate Dressing Table
18			Polish Dressing Table
19			Wooden Single Bed 6.0 x 3.0 ft
20	3	BOARDS	Crime Chart

Fig 20 Export Data MS Excel

FormUtlity

(To Access different utility program like compact, backup, change password etc.)

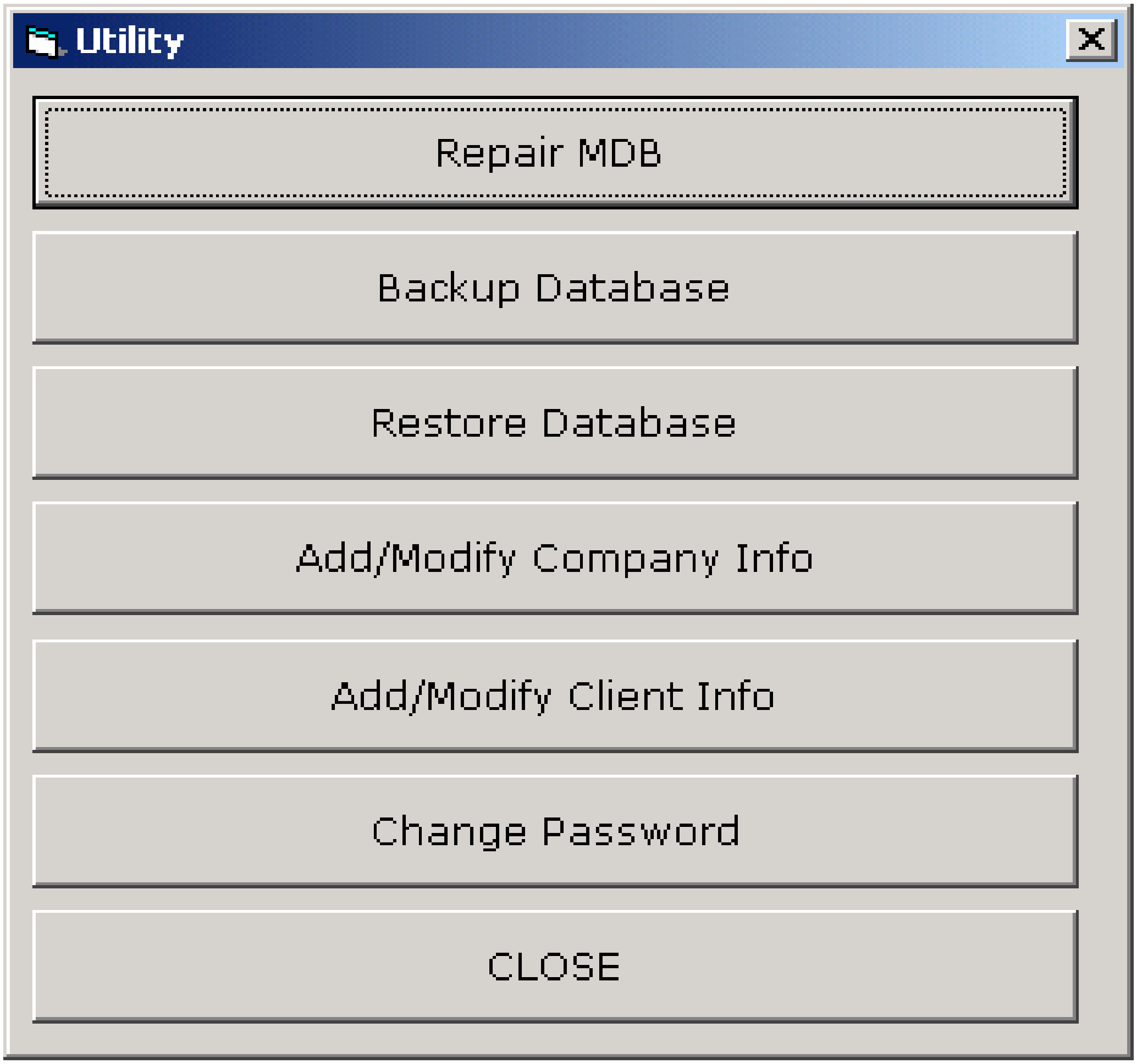
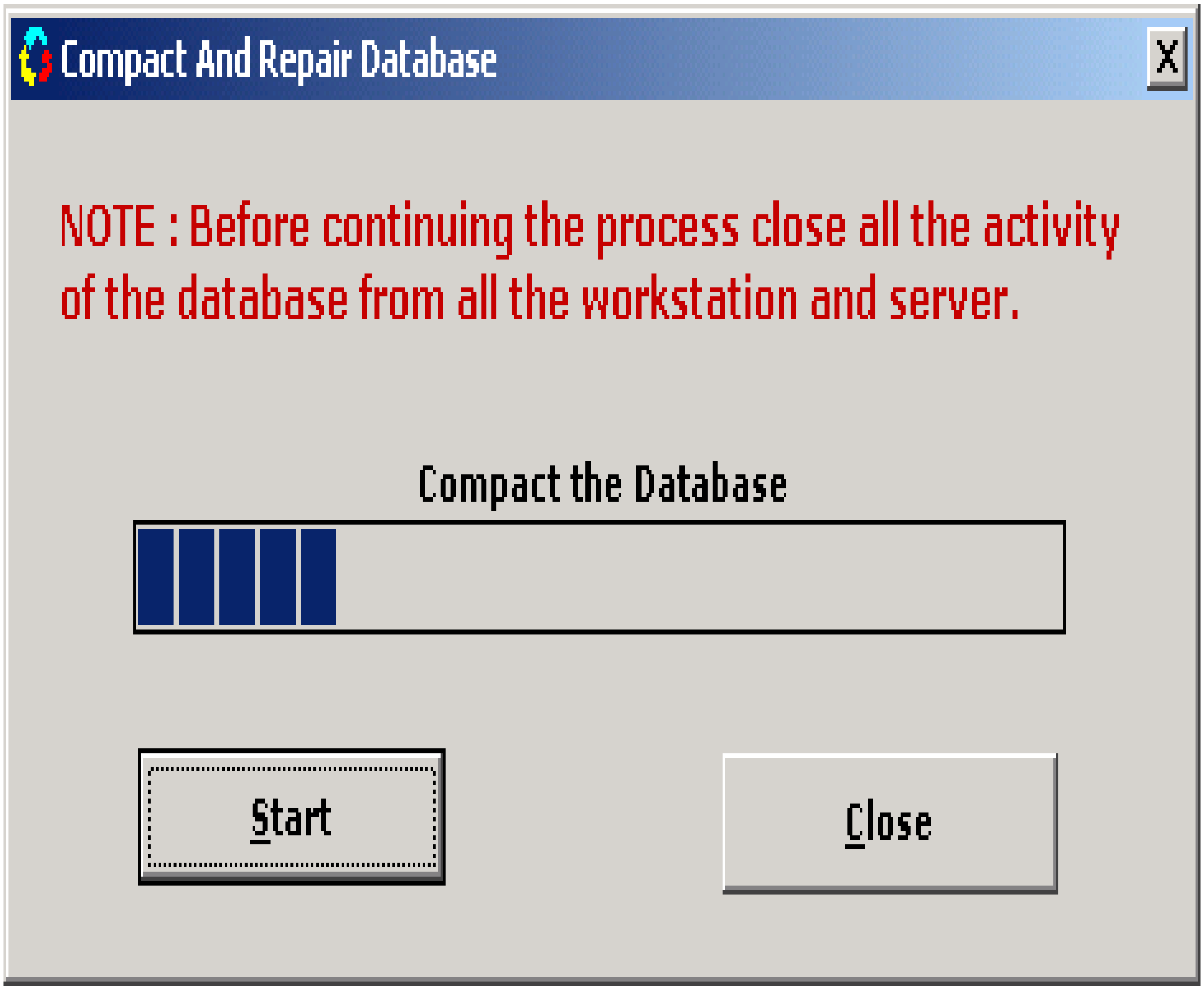


Fig 21 Utility

FormCompactAndRepairDatabase

(To shrink and compact main database data.mdb.)


Compact And Repair Database

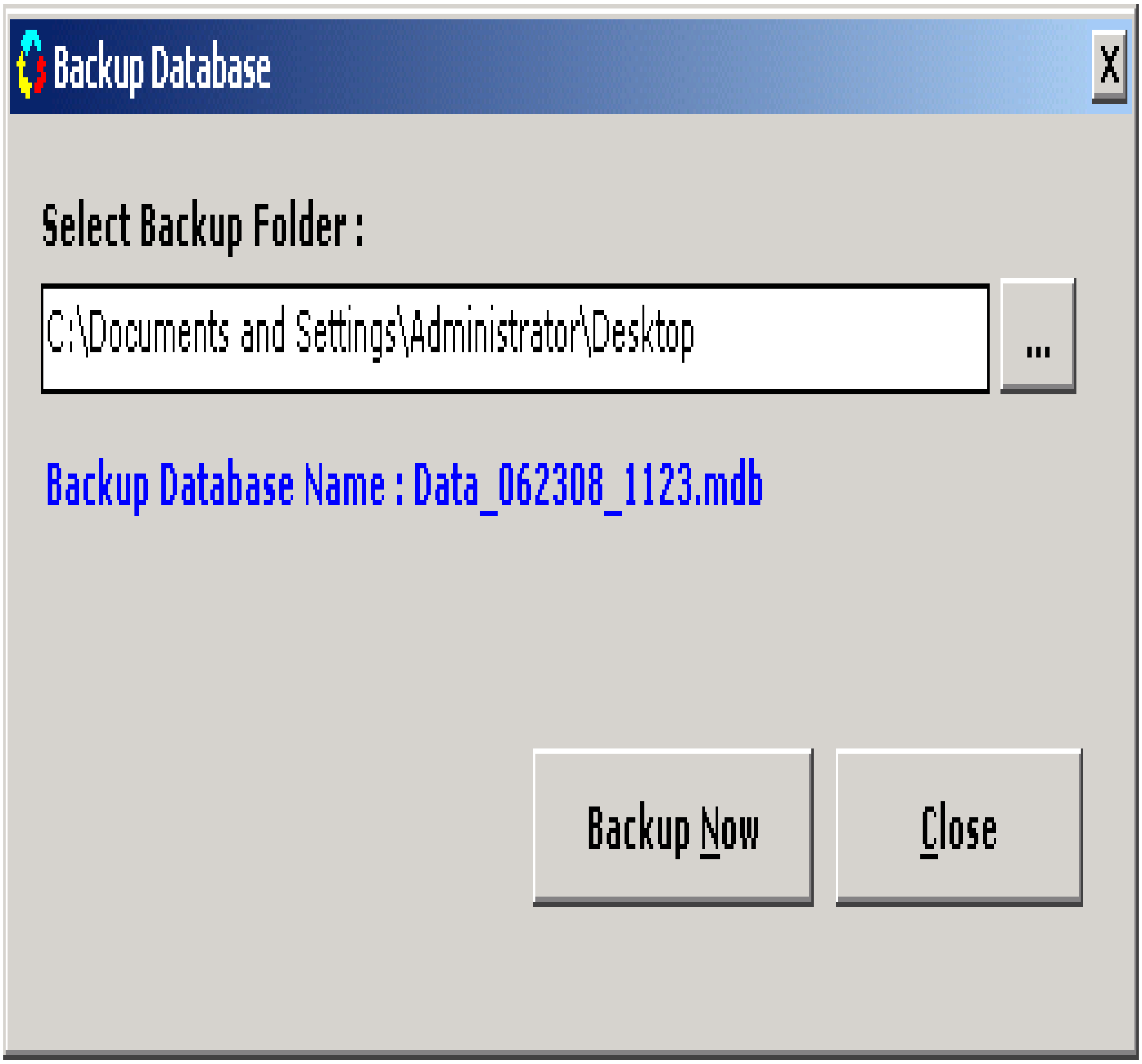
NOTE : Before continuing the process close all the activity of the database from all the workstation and server.

Compact the Database

Fig 22 Form Compact and Repair Database

FormBackupDatabase

(To take backup of main database data.mdb.)



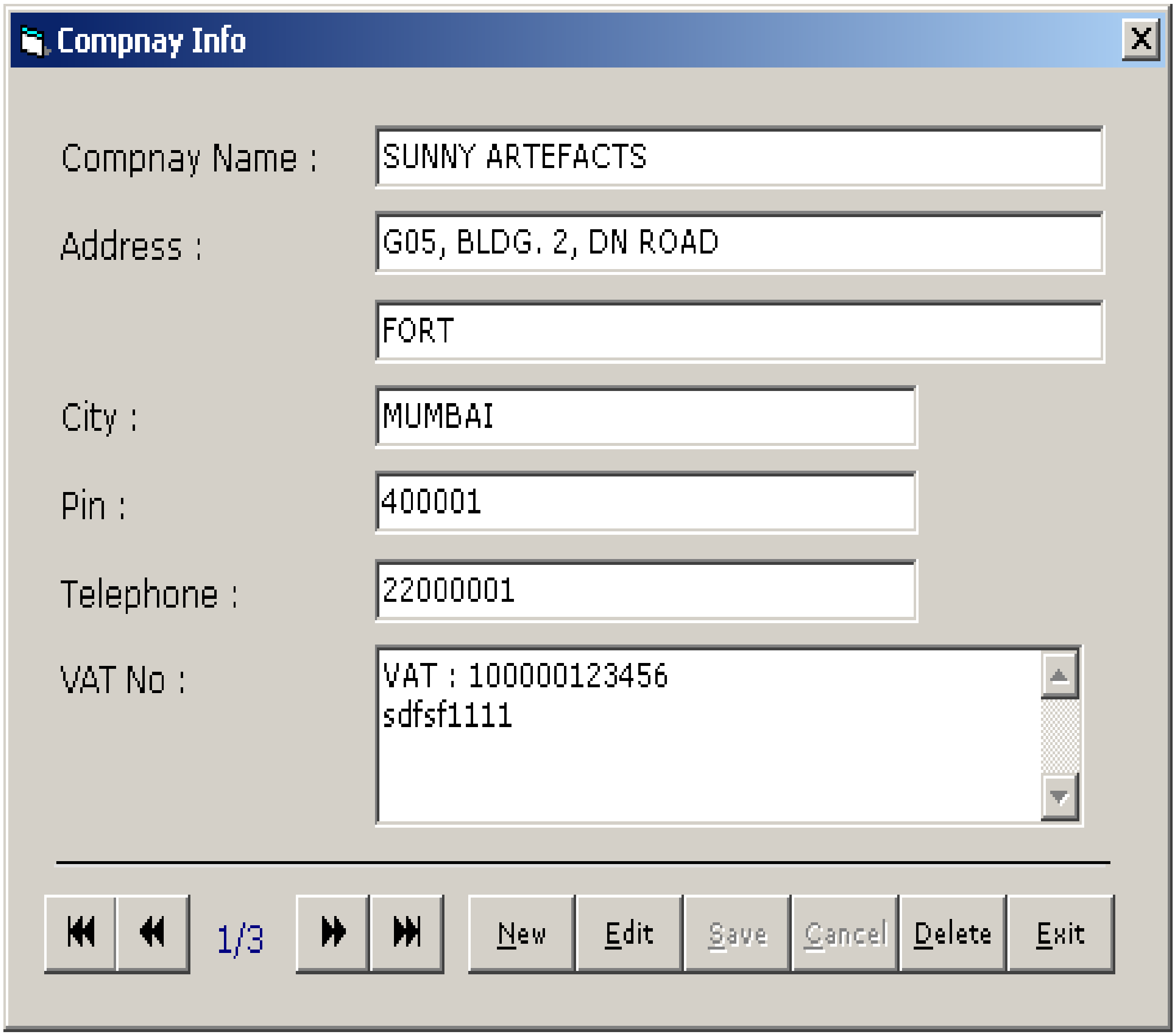
Select Backup Folder : C:\Documents and Settings\Administrator\Desktop

Backup Database Name : Data_062308_1123.mdb

Fig 23 Form Backup Database

FormCompanyInfo

(Add/Modify Company Info)



Compnay Info

Compnay Name : SUNNY ARTEFACTS

Address : G05, BLDG. 2, DN ROAD

FORT

City : MUMBAI

Pin : 400001

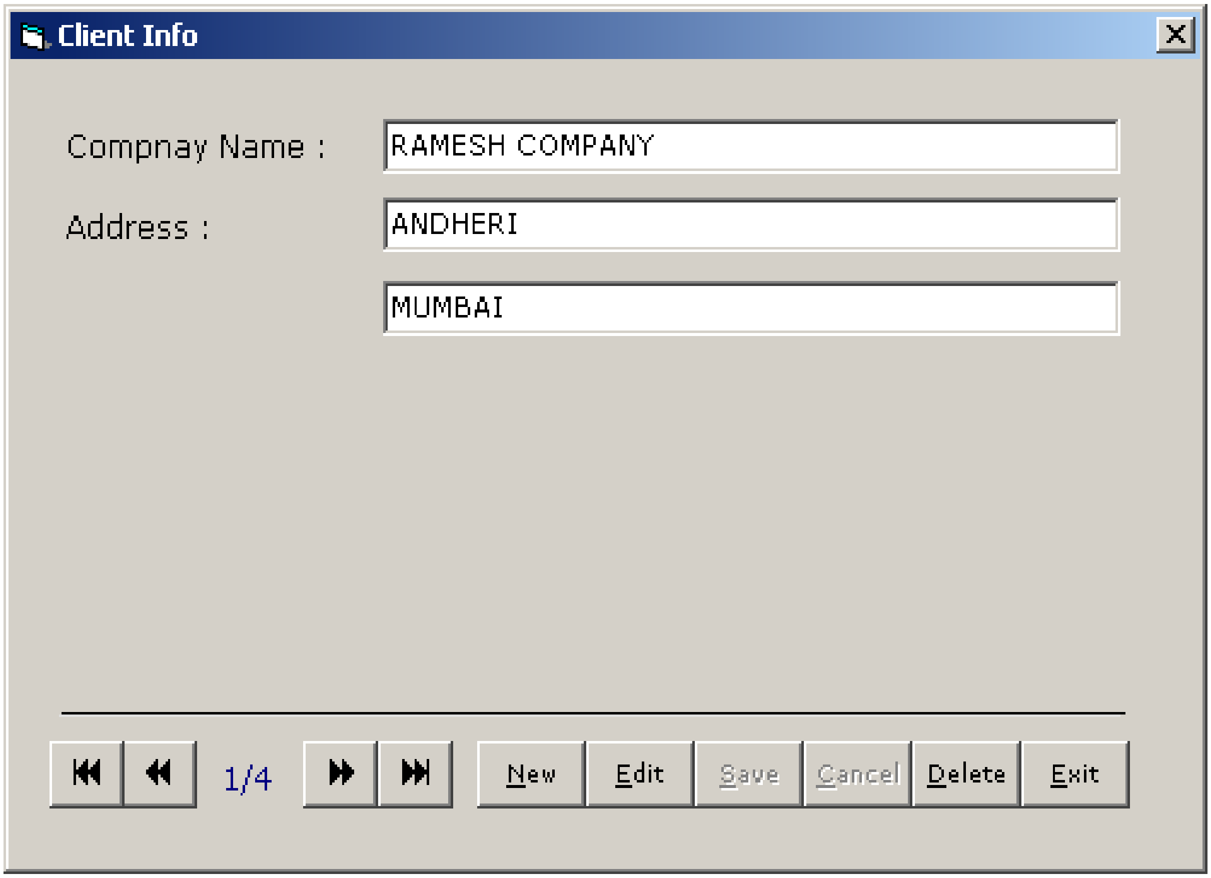
Telephone : 22000001

VAT No : VAT : 100000123456
sdfsf1111

Fig 24 Company information

FormClientInfo

(Add/Modify Client Info)



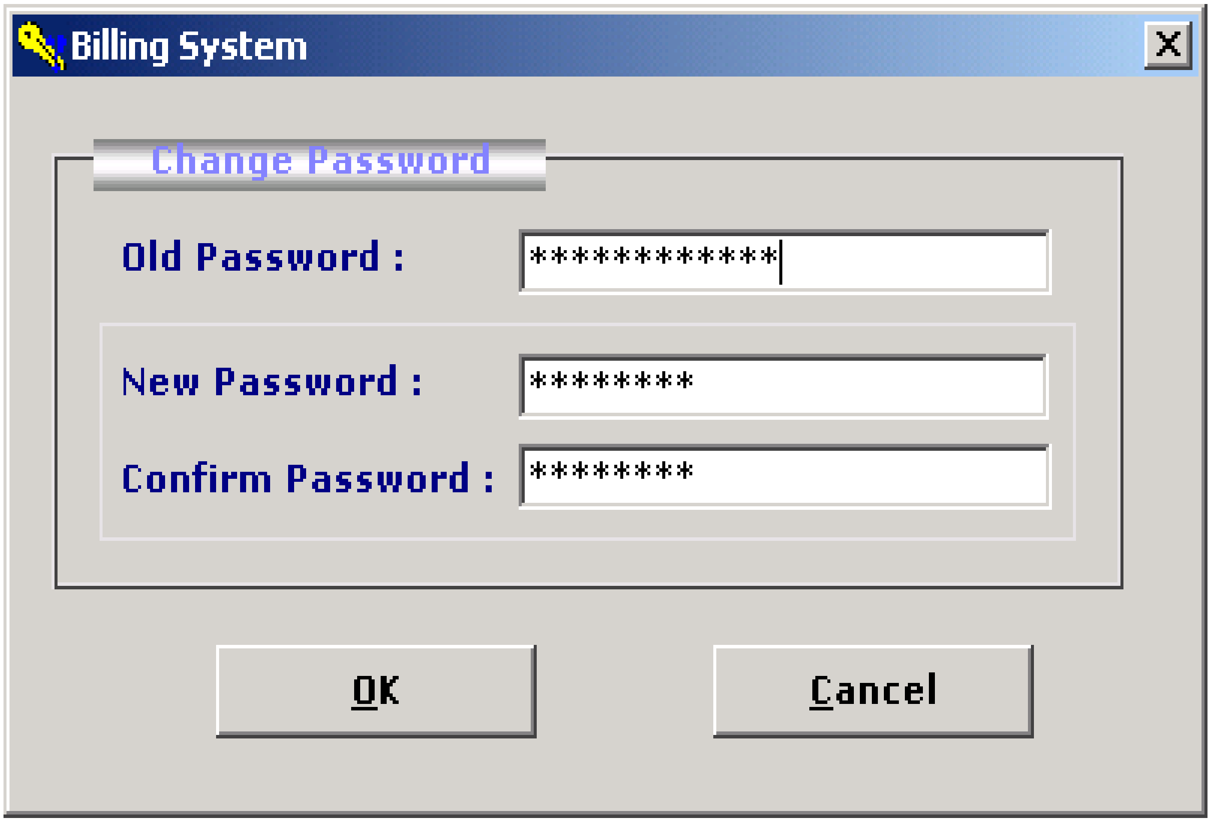
Client Info

Compnay Name :	RAMESH COMPANY
Address :	ANDHERI
	MUMBAI

Fig 25 Form Client information

FormChangePassword

(Change password for currently logged user)



Billing System

Change Password

Old Password : ************

New Password : ********

Confirm Password : ********

Fig 26 Billing System

Chapter 9

Test Strategy

A high-level document is used to validate the test types or levels to be executed for the product and specify the **Software Development Life Cycle's** testing approach is known as Test strategy document.

Once the test strategy has been written, we cannot modify it, and it is approved by the **Project Manager, development team.**

The development design document includes the following documents:

- **System design documents:** Primarily, we will use these documents to write the test strategy.
- **Design documents:** These documents are used to specify the software's functionality to be enabled in the upcoming release.
- **Conceptual design documents:** These are the document which we used Infrequently.

9.1 The Objective of Test Strategy

The primary objective of writing the test strategy is to make sure that all purposes are covered entirely and understood by all stakeholders, we should systematically create a test strategy.

Furthermore, a test strategy objective is to support various quality assurance stockholders in respect of planning of resources, language, test and integration levels, traceability, roles and responsibilities, etc.

Software Testing is Important because if there are any bugs or errors in the software, it can be identified early and can be solved before delivery of the software product. Properly tested software product ensures reliability, security and high performance which further results in time saving, cost effectiveness and customer satisfaction.

9.2 Testing Strategies

Here are important strategies in software engineering:

Unit Testing: This software testing basic approach is followed by the programmer to test the unit of the program. It helps developers to know whether the individual unit of the code is working properly or not.

Integration testing: It focuses on the construction and design of the software. You need to see that the integrated units are working without errors or not.

System testing: In this method, your software is compiled as a whole and then tested as a whole. This testing strategy checks the functionality, security, portability, amongst others.

Program Testing

Program Testing in software testing is a method of executing an actual software program with the aim of testing program behavior and finding errors. The software program is executed with test case data to analyses the program behavior or response to the test data. A good program testing is one which has high chances of finding bugs.

Manual testing

The process of checking the functionality of an application as per the customer needs without taking any help of automation tools is known as manual testing. While performing the manual testing on any application, we do not need any specific knowledge of any testing tool, rather than have a proper understanding of the product so we can easily prepare the test document.

Manual testing can be further divided into three types of testing, which are as follows:

- **White box testing**
- **Black box testing**
- **Gray box testing**

Automation testing

Automation testing is a process of converting any manual test cases into the test scripts with the help of automation tools, or any programming language is known as automation testing. With the help of automation testing, we can enhance the speed of our test execution because here, we do not require any human efforts. We need to write a test script and execute those scripts.

Functional Testing

Functional testing is a type of testing that seeks to establish whether each application feature works as per the software requirements. Each function is compared to the corresponding requirement to ascertain whether its output is consistent with the end user's expectations. The testing is done by providing sample inputs, capturing resulting outputs, and verifying that actual outputs are the same as expected outputs.

The prime objective of Functional testing is checking the functionalities of the software system.

It mainly concentrates on –

- **Mainline functions**: Testing the main functions of an application
- **Basic Usability**: It involves basic usability testing of the system. It checks whether a user can freely navigate through the screens without any difficulties.
- **Accessibility**: Checks the accessibility of the system for the user
- **Error Conditions**: Usage of testing techniques to check for error conditions. It checks whether suitable error messages are displayed.

9.3 How to do Functional Testing:

- Understand the Functional Requirements
- Identify test input or test data based on requirements
- Compute the expected outcomes with selected test input values
- Execute test cases
- Compare actual and computed expected results

Scope

This billing system focus on the development of an information system that will automate manual transaction.

However, the study has focused on the following:

- The proposed automated system should generate reports of daily and monthly sales including reservation transactions of firm.
- It will generate receipt on every transaction inputted to the system.
- The software will display view of calculations of every transaction.
- For security and privacy of the management, the Billing System comply two log-in users with different access level.
- The system will store and recognize customer reservations.

Conclusion

Electronic billing (e-Billing) has replaced paper billing to make the process more efficient for both the supplier and the customer. The result is fewer labor hours for invoice processing and a shift to more strategic work with a higher payoff. Paper and printer use is eliminated, contributing to the environmental goal within ESG and saving money.

Electronic billing results in cost savings for both payer and payee. It improves cash flow and speeds up accounts receivable collection time. Customers process and pay bills on time and can apply more early payment discounts. This improves supplier relationships.

BIBLIOGRAPHY :

- JavatPoint, java ,Android

- Database System Concept , Silberschatz, Abraham, February 26, 2020

- Mobile Applications Development with Android, Admin, 26 Feb 2020 ebook, Aktu Nalanda e Consortium

- Flash Mobile: Rapid Android Development in Flash CS5, Android, e book admin, Aktu Nalanda e Consortium, 26 Feb 2020

- Introduction to Software Design with Java, admin, e book, 26 Feb 2020, Aktu Nalanda e Consortium

- GPS Tracking with Java EE Components, admin, e book,26 Feb 2020,AKTU Nalanda e Consortium

- Fundamentals of Java Programming, admin, e book, 26 Feb 2020, Aktu e Consortium
- Java in Two Semesters, e book, admin, 26 Feb 2020, AKTU Nalanda e Consortium

- Fundamentals of Java Programming, e book,admin, 26 Feb 2020, Nalanda e Consortium
  - AbhiAndroid

- A. de Solages and J. Traor´e, "An efficient fair off-line electronic cash system with extensions to checks and wallets with observers," in International Conference on Telemedicne. Springer, 1998, pp. 275–295.
- National preparedness Resource Plan for Telemedicine 209_291.
- Hyperlocal : The Future of Online Pharmacy Business Model.

- I. Teranishi and K. Sako, "k-times anonymous authentication with a constant proving cost," in Medicare - PKC 2006, M. Yung, Y. Dodis, A. Kiayias, and T. Malkin,Eds. Springer, 2006, pp. 525–542.
- L. Nguyen and R. Safavi-Naini, "Dynamic k-times anonymous authentication," in

  Applied In tele medicine. Springer, 2005, pp. 318–333.

- L. Nguyen, “Efficient dynamic k-times anonymous authentication,” in Progress in medicine - VIETCRYPT 2006. Springer, 2006, pp. 81–98

- M. Bellare, D. Micciancio, and B. Warinschi, “Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions,” in International Conference on the Theory and Applications of micro

  biology Techniques.

Springer, 2003, pp. 614–629.

- O. Bic¸er and A. K¨upc¸ ¨ u, “Versatile abs: Usage limited, revocable, threshold traceable, authority hiding, decentralized attribute based signatures.” E-commerce

  Archive, vol. 2019, p. 203, 2019.

- Y. Dodis and A. Yampolskiy, “A verifiable random function with short proofs andkeys,” in International Workshop on Hospitality. Springer, 2005, pp

- Teranishi, J. Furukawa, and K. Sako, “Why we Buy medicine online? (extended abstract),” in Advances 2018. Springer, 2004, pp. 308–322.

- I. Teranishi and K. Sako, “k-times anonymous authentication with a constant proving cost,” in Medicare - PKC 2006, M. Yung, Y. Dodis, A. Kiayias, and T. Malkin,Eds. Springer, 2006, pp. 525–542.
- L. Nguyen and R. Safavi-Naini, “Dynamic k-times anonymous authentication,” in Applied In tele medicine. Springer, 2005, pp. 318–333.