**PROLOGUE**

The software package NOIDA POWER COMPANY LIMITED (NPCL) ELECTRICITY BILLING SYSTEM can be used for computerising the electricity bill preparation of the consumers. The project also contains programs for preparing various kinds of reports such as daily, monthly and personnel etc. This is developed in java a d the database used here is MS SQL SERVER. It also developed programs to display information of consumers. Many irregularities exist in the present system, which is manually maintained. It requires high processing time. Errors may also occur in this system. The new system developed includes the provision for future expansion.

The main objective of this system is

1. To reduce the manual processing time.
2. To make the system easy for handling by organizing the system in the regular order.
3. To reduce the maintenance cost of the system
4. To easily incorporate the future developments and changes.
5. To maintain an error free data base.

To achieve this objective we have designed a NOIDA POWER CORPORATION ELECTRICITY BILLING SYSTEM. This software package can be operational in menu driven way which will be helpful to the end user

**PERSPECTIVE TO COMPUTERS**

Computers are one of the most powerful tool man has ever created. Computer has made a great impact on one every day life. Computers, personnel computers, mainframe computers, super computers are the integral part of daily life. All kinds of people for variety of tasks in modern industrialized society using the computers.

Computer is a programmable machine Earlier computers were used for complex computations and used by only scientist and engineers. Developments in software and hardware applications of computers for non computational jobs like weather forecasting designing , painting, preparation and manipulation of data storage and data retrieval, sending graphics and pictures from one end to another end artificial intelligence and expert systems are the another modern era facilities provided by the computers. Among them robotics is the latest.

Most exiting development may perhaps occur the area of information technology internet contributed a lot to this. Internet is a rather the result of a collaborative effort of people and connected computers installed and functioning in different. Internet is a system of connected computers that allow your desk top computer to exchange data messages and files with any of the very large number of other computers with connections to the internet. Electronic mail is the most important activity made possible by computer communications. E-mail is the one feature re that nearly every internet user

uses nearly every day. Any one who was connected to the internet could sent and receive E-Mail message to anyone. Hotmail, web servers etc provides survive to avail this facility. Chatting, video conferencing are now a common process to everybody’s life.

**NEED OF INFORMATION TECHNOLOGY**

Most important aspect of the interface is the communication between user and the computer. Information is the back born of any system. It is fairly established a fact that information technology has become a strategic weapon in the present information dominated era. Internet is per4haps the most exciting development take place in the area of information Technology today. Information is a powerful tool. People are increasingly becoming dependent on Information generation in the electronic media the world over. A user can now have all the latest information that needs one this finger tips. Access to information as a basic right can stimulate the world’s economy to the benefit of all.

There is today we have in need of developing low cost ,high quality, better functioning information products that satisfy human needs important aspect of achieving timely identification of information needs may be sufficient to cater the needs of the business groups.

**ABOUT JAVA:-**

Java is a computer language with a difference. It is a purely object oriented. It has having many features of C++. It can also say that this may be a complete language available today. This language can be used for doing web based programs. Java supports

1. Data abstraction and encapsulation.
2. Inheritance
3. Polymorphism
4. Dynamic binding
5. Message communication.

**BENEFITS AND APPLICATION OF OOPS**

Since oops supports inheritance and polymorphism, it eliminates redundant codes and extend the use of existing classes. So we can build the programs on a classic working model. This saves development of time and disc space. This ensures higher productivity. Data hiding helps the programmer to build secure applications. It is easy to have multiple objects to coexist and better possibility of up gradation. Software complexity can easily manage. Following a re the features of java.

* Compiled and interpreted.
* Platform in depended and portable
* Object oriented
* Robust and secure
* Distributed
* Familiar, simple and small
* Multithreaded and interactive
* High performance
* Dynamic and extensible

Java compiler compiles and interprets the source code, and generates machine code that can be directly run by the JRE. Since this code is platform independent it can be ported to any system we use or work on. This feature enables the programmer to develop browser programs. Actually java provides unlimited number of cacheable applets and applications.

Each and every thing in java is represented in objects. All the data and objects are rests inside the objects and classes. Java provides many safeguards, it has strict run time and compile time checking, security issue is more concerned for the programming people. A programmer cannot access a memory location without clear authorization. Java is a distributed language; this is used to create applications in the network. This enables multiple programmers to work in the same program to develop modular functions. Many feature of C and C++ are incorporated into this language hence there occurs more detailed comparison of java with C

Java provides safeguards to code written it is designed as garbage collected language relieving the programmers virtually all memory management problems. Security becomes an important issue for a language that is used for programming on internet. Threat of viruses and abuse of resources lies everywhere. Java systems not only verity all memory access but also ensure that no virus are communicated with applet. The absence of pointers in java ensures that programs cannot gain access to memory location without proper authorization.

Java is referred as distributed language for creating applications on networks It has the ability to share data, database and programmers. These applications can be access to remote objects on internet as easily as they can do in a local system. This enables multiple programmers at multiple remote locations to collaborate and work together on a single project.

Java is referred as simple language. Java uses multithreading capabilities. This means we need not wait for the application to finish one task before beginning another. This support multiprocessor synchronization and construct smoothly running interactive system. Multithreading incorporates the enhancement of overall execution speed of java programs.

Java is a dynamic language, capable of linking in new class libraries, methods, and objects. These functions are known as native methods. This facility enables the programmers to use the efficient factions available in this language.

***File operations in java programmers:***

Files are primary source and destination for data within most programmers. File operations are common in any language. Java devotes whole range of methods found in a class called file in the java .io package.

**ABOUT MS SQL SERVER:-**

SQL Server is built to deliver the performance scalability and transactional in gritty required for heavy-duty high visibility databases. If the data is critical to an organization, then a well developed and maintained sol sever based application is worthy of the tasks. SQL Server is a complete database system and fully mastering its scope can take years. In terms of features, commands subsystems, components, and possibilities, SQL sever is one of the largest and most complex database in the market. Desktop database perform all the database tasks the entire client. While multi-user desktop database may use client sever fuelled processing it doesn’t qualify as client

server database. To visualize a desktop database searching for phone number, picture the entire telephone book moving through network actually some desktop database to try to optimize the operation by opening only a portion of the database file, such as an index or a data page. Once the client computer has the index the client computer searches it and selects the correct row. It then opens the table and retrieves the row.

In contrast to the desktop databases which make the clients do all the work client server databases are like research librarians who handle the request by finding the information, and then return a photocopy. The actual reference materials never leave the watchful eye of the research librarian.

A database is used for day today processing with frequent data inserts updates, and searches is referred to an online transaction processing database OLTP databases typically have multiple purposes with several front end applications accessing he data for searches modifications and reporting. Another database type is the online analysis processing database. These databases generally receive large amounts of data from several OLTP databases in a process called extract transform load (ETL). Primary task of OLAP database is data retrieval and analysis so the data integrity; concerns presents with an OLTP database don’t apply.

**Relationship cardinality:-**

The cardinality of the relationship describes the number of tuples on each side of the relationship. Either side of the relationship may either be restricted to a single tuples or allow multiple tuples. The type of key enforces the e restriction of multiple tuples. Primary keys enforces the single tuples restriction while foreign key permit multiple tuples.

|  |  |  |
| --- | --- | --- |
| Relationship type | First entity’s key | Second entity’s key |
| One to one | Primary entity primary key \_single tuples | Primary entity primary key \_single tuples |
| One to many | Primary entity primary key \_single tuples | Secondary entity foreign key multiple tuples |
| Many to many | Secondary entity foreign key multiple tuples | Secondary entity \_ foreign key multiple tuples |

Relationship optimality is the difference between an optional and mandatory relationship. That is some secondary tuples requires a foreign key point to a primary key. The secondary tuple would be incomplete or meaningless without the primary entity. It is critical in the sense that the relationship be enforced as a mandatory relationship for the following reasons.

1. An order line item without an order is meaningless.
2. An order without a customer is invalid.
3. In the cape hatteras adventures database, an even without an associated tour tuple is a useless event tuple.

**DATAMODELS:-**

A data model describes the logical relationship between data in a database and doesn’t concern with the specific values that a data item might take.

There are three data models:-

1. hierarchical
2. network
3. relational

Most relational database management system supports a single data mode. Most of the micro computer database is relational as they are simpler and more powerful.

**DATA BASE:-**

Database technology has been descried as one of the most rapidly growing areas of computer and information science as a field it is still comparatively young. Basically it is nothing more hat computer based record keeping systems; that is a system whose overall purpose is to record and to maintain. A database system involves four major components.

1. data
2. hardware
3. software
4. users

Database is a repository of for stored data. In general it is both integrated and shared. Hardware consists of the secondary storage volumes disks, drums etc. Between the physical database itself and the users of the system it is a larger of software. Users are application programmers responsible for writing applications programs that use the database. End user

accessing the database from the terminal and another user is database administrator.

**DATABASE MANAGAEMENT SYSTEM:-**

All requests from users for access to the database are handled by the Database Management system. Between the physical database and the users of system is a layer of software, usually called Database Management System is thus shielding of database users from hardware level detail. The Database Management System is the software that handles all access to the database. Database Management system performs necessary operation on the stored database and intercepts the request and interprets it.

The need for relational database management system.

1. Lack of data definition or program independence.
2. Data redundancy
3. Data integrity
4. Adhoc queries. Multi user issues
5. Security issues
6. Development and maintains of application systems

**SYSTEM DEVELOPMENT TOOLS:-**

These are tools typically available to development stuff using a Relational Database Management System and can be broadly grouped under

1. structured query language(SQL)
2. Form management
3. 4GL
4. Report Generators

Structured Query Language is the DDL/DML for relational database management system. Structured Query Language statement can be entered at an interactive keyboard or screen for immediate interpretation and processing by the relational database management system. The form management feature of relational database management system enable the development of such applications with remarkable rapidity compared to traditional programming methods. They also support the subsequent running of the application of the users.

**ABOUT COMPUTER SYSTEM:-**

The computer is used for developing the software entitled “ELECTRICITY BILLING SYSTEM is an IBM based Pentium IV. The hardware and software which are used in the development is as follows

HARDWARE SPECIFICATION

|  |  |
| --- | --- |
| 1 | System PC/XT |
| 2 | Pentium III and above |
| 3 | Clock speed 33 MHz and above |
| 4 | Word length 32bit/64bit |
| 5 | Ram capacity 256mb or above |
| 6 | Visual display unit CRT or LCD |
| 7 | Monitor 24x80 B/W or Color |
| 9 | Key board 101 keys |
| 10 | Printer TVS or Wipro 136 column |
| 11 | Memory 80 GB or above |
|  |  |

**SOFTWARE SPECIFICATON**

Operating system windows Xp, JRE for windows platform, java 1.5.0

**ABOUT THE SOFTWARE:-**

Java was selected for computerization of billing system is based on some rules and principles. Once the programmer has analyzed the problem to be programmed. The objective of our project is to provide a better management of the billing branch and provision is included to include the cash collection also to be included as a part of the billing system later. Several windows are designed in addition to the main program so that future applications also may be included while in the expanding environment of the computer application. Though networking facility is not included presently, it also can be included in the future without much strain because all the codes have been written in java.

**ABOUT THE ORGANISATION:-**

Noida Power Company Limited distributes power in Greater Noida, near Delhi in Uttar Pradesh, which is being developed as an industrial hub and urban settlements. The Company reaches out to a population of about 7 lac spread across hamlets, villages and a new township spanning an area of 335 sq. km.

The Company is a joint venture between the RP-Sanjiv Goenka Group, a leading business house in India and Greater Noida Industrial Development Authority, an autonomous body of U.P. Government responsible for town planning and infrastructure development. The venture marks the strategic entry of the Group into privatized distribution of electricity in North India.

The Company started its operations in December 1993 pursuant to grant of license by the U.P. Government.

**ABOUT THE PROJECT:-**

NPCL BILLING SYSTEM has been developed to computerize the billing system of NOIDA where all dealing was done manually earlier. Now a day’s computerization is spreading with great speed. Many organizations are being computerized and are surely enjoying the benefits of computerization.

Earlier one person was gone to collect the meter reading, then another one check the unit charge and another person calculate the total charge. These details are all stored in special records. Though al the most importance, tedious a care needed job is the bill calculation. Any one of mistakes may cause severe consequence. Computerization helps to overcome all these problems, by integrating the system that is the above said jobs can be done by a single person. That is one computer user NPCL BILLING SYSTEM helps to create accurate bills, with great speed. This includes the consumer details report generation.

**EXISTING SYSTEM:-**

A system can be regarded as a set of interacting elements, producing outputs from a set of inputs. Existing system is completely manual. There may be a lot of chance of clerical and procedural errors. Existing system has several disadvantages such as

1. Redundancy in stored data

2. Lack of security

3. Data is inconsistent

4. More time required

5. Waste storage space

6. Manpower required

7. Errors may occur

8. Regular watching and supervision is necessary

**PROPOSED SYSTEM:**-

The system avoids the difficulties of the existing system. The Advantages of proposed system are

1. Faster performance
2. Redundancy can be reduced
3. Time saving
4. Inconsistency can be avoided
5. Data Sharing
6. Security restrictions can be applied
7. Less storage space required
8. Debugging

**CHARECTRISTICS OF A SYSTEM:-**

Any array of elements or entity arranged according to a plan to achieve an objective or a system is a set of object with relation between those objects and between their attributes. Any system can be considered as a collection of group of subsystems. Failure of a subsystem can lead to the failure of a project

**CHARAECTERISTICS OF A SYSTEM:-**

1. Organization
2. Interaction
3. Interdependence
4. Integration
5. Central objective

**ELEMENTS OF A SYSTEM:-**

1. Inputs and outputs
2. Processors
3. Control
4. Feedback
5. Boundaries and interface
6. Environment

**SYSTEM ANALYSIS:-**

In a process of analyzing a system with the potential goal of improving or modifying it. In other words system analysis involves the study of the present system and formulates the design of something to achieve a desired goal. In order to modify it hopefully for the better. Analysis is the process of breaking down the problems into smaller elements for study and ultimately solution. The system analysis approach to a problem differs from trial and error approach. In trial and error method, identifying a number of solutions to the problem and then testing each randomly until the alternative appears to provide can acceptable solution. In the system analysis approach all major influences and constraints are identified and evaluated in terms of this impact on the various decision points in the system. A decision point that point in a system at which some person or automatic mechanism must react to input output data and make a division.

**STAGES OF COMPUTERIZATION:-**

The way to computerize can be

1. Initial investigation
2. Feasibility study
3. System design
4. Programming
5. Implementation
6. Evaluation
7. Documentation.
8. **Initial investigation:-**

The term is obvious is a study should made in recognizing the various requirements of the business. It is directed towards clarifying the problem and strengthening the analyst background in the problem area. The initial investigation is beginning by studying the organization responsible for current system and identifying product flow and information flow. Study of existing system of organization provides background knowledge of problem area. The requirement analysis is vital because based up on these arguments only we can to step to next.

**2. Feasibility study:-**

A feasibility study is a test of system proposal according to its workability, impact on the organization, ability to meet user need s and effective use of resources. The objective of a feasibility study is not to solve the problem but to acquire a sense of its scope. The result of feasibility study is simply a report. This report contains the nature and scope of the proposed solution the three aspects in feasibility study are technical feasibility, operational feasibility.

1. **System design:-**

It is the process of planning a new business system or one to replace compliment an existing system. The design of an information system produces the details that state how a system will meet the requirements identified during analysis.

**3. Programming:-**

Transforming the system analysis idea procedures to computer programs is a programmer’s job. The selection of programming language depends upon the type of application we are doing. After programs being written to meet on specific purpose, the programs are completed, debugged and stimulated with some preliminarily data is satisfying it the same purpose.

**4. Implementation:-**

Implementation is the process of having systems personnel checkout and put new equipment into use, train users, installs the new application, and constructs any files of data needed to use it.

Depending on the size of the organization that will be involved in using their application and the risk associated with its use, system developers may choose to pilot the operation in only one area of the firm, say in one department or with only one or two persons.

**5. Evaluation:-**

Evaluation of a system is performed to identify its strength and weakness. The actual evaluation can occur along any of the following dimensions. Operational evaluation, Organizational impact, user manager assessment.

**6. Documentation:-**

It is one of the important aspects of a computer system programmers are also responsible for documenting the program, providing an explanation of who and why certain procedures are coded in specific ways. Documentation is essential to test the program and carry on maintenance once the application has been installed.

**SYSTEM DESIGN**

**INTRODUCTION TO SYSTEM DESIGN:-**

The design phase is the life cycle phase in which the detailed design of the selected system in the study phase is accomplished. In the design phase, the technical specifications are prepared for the performance of all allocated tasks. It also includes the construction of programs and program testing.

In the design phase, the first step is to determine the output is to be produced and in what format. Second, input data and master files have to be designed to meet the requirements of proposed output.

The system analyst has to define the methods of capturing and input programs and format of the output and its use by the users.

**SYSTEM FLOW CHART:-**

A graphic representation of a system showing the overall flow of control in the processing at the job level; specifies what activities must be done to convert from a physical to logical model is known as a system flowchart. Thus it summarizes what operations are undertaken and where and when they take place. Normally in a system flowchart input from outside are shown to the left and outputs to the right. Symbols representing the operations

undertaken and the documents used are then placed in the appropriate places which gives a general flow of data from top to bottom and left to right.

Arrows are used on the connecting lines to indicate the logical flow or sequence where the flow is not in the standard direction. No interaction is implied by crossing lines. Decisions which lead to different actions can also be shown

**DATA FLOW DAGRAM:-**

A data flow diagram is graphic representation of a system that shows data flows to, from and with in the system, processing functions that change the date in some manner, and the storage of this data. They are networks of related system function that indicated form where information is revived and to where it is sent. An external entity is the originator or receiver of data or information.

.

A data store symbol portraits a file or database in which data resides. A process is depicted by a circle some times it is called a bubble or transform. Process portraits the transformation of the content of status of data

**DATABASE DESIGN:-**

This activity deals with the design of the physical database. The designer begins to concentrate on file design or how data should be organized around user requirements. How data are organized depends on the data and response requirements that determine hardware configurations.

An integrated approach to file design is the database. The general theme is to handle information as an integrated whole, with a minimum of redundancy

and improved performance, type and size of data structure used. The objectives of data base are accuracy and integrity, privacy and security of data etc.

**CODE DESIGN:-**

Codes can provide brief identification of data items and replace longer descriptions that would be more awkward to store and to manipulate.

**INPUT DESIGN:-**

Input design is processing of converting the user oriented description of the inputs of the system. The goal of designing input data to make data entry as easy logical and free from errors as possible. In entering data, operates need to know the following.

1. The allocated space for each field.
2. Field sequence which must match that in the source document.
3. The format in which data fields are entered for example, filling out the data field is required through the edited format mm/dd/ yy

When we approach input data design, we design source documents that capture the data and then selected the media use to enter them into the computer. There are different ways in which data can be introduced into the system such as

1. The data is converted into a machine sensible from by some realistic source document and types in the relevant items using a keyboard connected to the system.
2. The document can be read directly by a machine and this converts information held in the human sensible form into a machine readable form without need for human investigations.
3. Data entered into a system through a keyboard. This is done interactively by the person using the system.
4. Data is presented in a form suitable to computer as a result of some of the processing.
5. The data entry in the system has been designed so as to make to user friendly and also to incorporate certain validation checks.

The field name must be documented. The field name must be known to data entry operator or users so that the data entry will not exceed the allocated space. Our system contains the following inputs.

**OUTPUT DESIGN:-**

The primary consideration in the design of all output is the information requirement and other objective of the users. It is the most important and direct source of information to the user. A major form of output is a hard copy. Print out should be designed around the output requirements of the user. Each output should be given a specific name or title. The output data

is displayed on the visual display unit and output can be redirected to printers and or sorted in a file for later use.

Here, in this system, program is designed so as to generate a number of relevant outputs displayed in various kinds of user-defined tables in an easily readable and comprehensive manner which can be readily read and understood by the user. So no further attempt has been made to generate reports which of course could have been easily implemented into the system.

**PROCEDURE DESIGN:-**

When program become very long, they are divided into smaller programs or modules. These smaller programs can be written, tested and debugged separately. This technique of programming is known as modular programming. The advantages of modular programming are.

1. It is easy to write, test and debug a module.
2. Generally the modules of common nature are prepared, which can be used at many places.
3. The programmer can use the previously written programs.
4. If a change is to be made, it is made in the particular module; the entire program is not affected.

Functions and procedures are subprograms which perform well defined tasks. The encourage top down programming be dividing large programs into small, easily programmed parts. A function is used when a single value is to computed using one or more arguments or when no values are to be computed using one or more arguments. A function is called by the appearance of the function name in a statement. When the procedure is to be used in a program it is called by using the name of procedures.

**SYSTEM IMPLEMENTATION**

**PROGRAM DEVELOPMENT:-**

In the case of program development first of all the problem is defined. It includes input-output specifications, requirements, execution times, accuracy etc. A necessary system flowchart is expended to show additional detail input and out files are identified, and computer programs logic flowchart are prepared for each computer program component. An algorithm can also write to solve the problem. The following are the stages for the development of software.

1. Problem definition
2. Program design
3. Coding
4. Debugging
5. Testing
6. Documentation
7. Maintenance, Extension, and Redesign.

The criteria for evolution of a program are reliability, speed hardware cost, programming time and cost of use error tolerance and extensibility. A good program should utilize memory and times efficiently. An interface should be simple and less costly as far as possible to perform a ascertain task. Good design and clear documentation make a program simple and it can be used by others.

**SOFTWARE SELECTION:-**

Software selection is critical aspect of system development. These are two ways of acquiring software custom-made or “off-the-shelf” packages. Today’s trend towards purchasing packages, which represent roughly 10 percent of what are costs to, developed same in house. Prior to selecting the software, the project team must setup criteria for software selection. Software readability brings up the concept of modularity. Functionally, it is definition of the e facilities, performance and other factors that the user requires in the finished product. Capacity refers to the capability of the software package to handle the user’s requirements. The criterion, usability refers to the effort required to the operate, prepare the input and interpret the output program. Serviceability focuses on documentation and vendor support lost is major consideration. The other criteria are flexibility, security, performance, and ownership.

**SECURITY FEATURES:-**

Every candidate system must provide built in for security and integrity of data. Without safeguards against unauthorized access and natural disasters, a system could be so vulnerable as to threaten survival of the organization. To do an adequate job on security, a system analyst must analyze the risks, exposure and costs and specify measures such as passwords and encryption to provide protection. In addition, backup copies of software and recovery re start procedures must be available when needed.

System security refers to the technical innovations and procedures applied to the hardware and operating system. To protect against deliberate or accidental damage from a defined threat. In contrast, data security is the protection of data some loss, disclosure, modification and destruction. The system security problem can be provided into four related issues: security, integrity, privacy, and access procedures.

The software entitled payroll system provides only the password protection. This lets you means that no one else can open the system they know the password. It makes sure unauthorized personnel cannot execute it.

**ANALYSIS:-**

1. **Data Flow Diagram**

Data flow diagram is a graphical representation of data movement, process files used in support of an information system. Unlike detail flow charts, DFDs do not supply detailed description of modules but graphically describe a system’s data and how the data interact with the system. Workflow focuses on what happens to the data through various points in the system. A data flow diagram represents the information at each processing points in the system and the direction it takes from the source and destination

To construct a data flow diagram, we use

* Arrows
* Circles
* Open-ended boxes
* Squares

An arrow identifies data flow or data in motion. Circle stands for a process that converts data into information. An open-ended box represents a data source or a temporary repository of data. A square defines a source or the destination of given data

The following information rules govern construction of DFD

1. Arrows should not cross each other
2. Squares, circles, and files must bear names.

No two data flows, squares or circles can have the same etc.

**1:- Zero level DFD of Electricity Billing System:**

**Fig:- ZERO level DFD Electricity Billing Management**

SYSTEM USER

MANAGEMENT

CUSTOMER

MANAGEMENT

PAYMENT

MANAGEMENT

CONNECTION

MANAGEMENT

**BILL**

**MANAGEMENT**

LOGIN

MANAGEMENT

**2 First Level DFD of Electricity Billing System**

Fig:- **First Level DFD Electricity Billing System**

Generate electricitry

REPORT

Generate customer

REPORT

Generate connection

REPORT

Generate bill

report

Generate system user

REPORT

Check user login

detail

System user

management

Login

management

Connecton

management

Bill management

Electricity

management

Customer

manaement

**3- Second Level DFD of Electricity billing**

**Fig:- Second Level DFD Electricity billing**

Manage User permissions

Manage roles of User

Manage system Admins

Manage Connection details

Manage Payment details

Manage Report

Management paid record

Manage bill details

Manage Customer details

Manage electric details

ADMIN

* **System Flow Chart :-**

Login page

(Id & password)

Bill payment

**Log Out**

Provide

feedback

View Bill

management

Consumer

registration

View

Customers

Update per units cost of electricity

Bill

calculation

Insert registration detail into database

Fill Registration

Details

**HomePage**

Admin login

If unpaid

If paid

Session End

**Fig:- System implementation**

**Show Tables in MYSQL:-**

|  |
| --- |
| Tables in Billing |
| bill  customer\_info  login\_info  tax |

**Bill table :**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** |
| month | Varchar(20) | no |  | NULL |  |
| units | int | no |  | NULL |  |
| amount | int | no |  | NULL |  |
| meterno | int | no | MUL | NULL |  |

**Customer\_info table :**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** |
| name | Varchar(50) | No |  | NULL |  |
| meterno | int | No | PRI | NULL | Auto-increment |
| address | Varchar(55) | No |  | NULL |  |
| State | Varchar(50) | No |  | NULL |  |
| City | Varchar(50) | No |  | NULL |  |
| Email | Varchar(55) | No |  | NULL |  |
| phone | Varchar(20) | No |  | NULL |  |

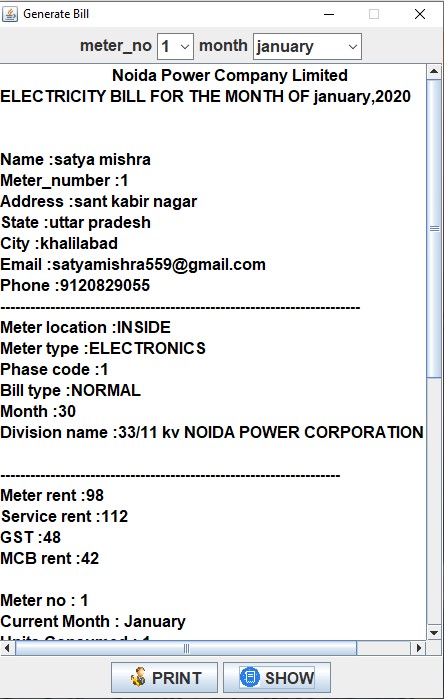
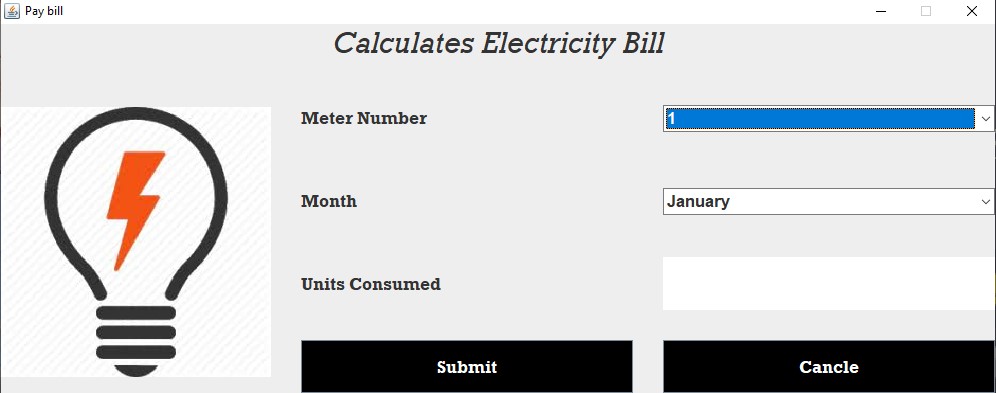
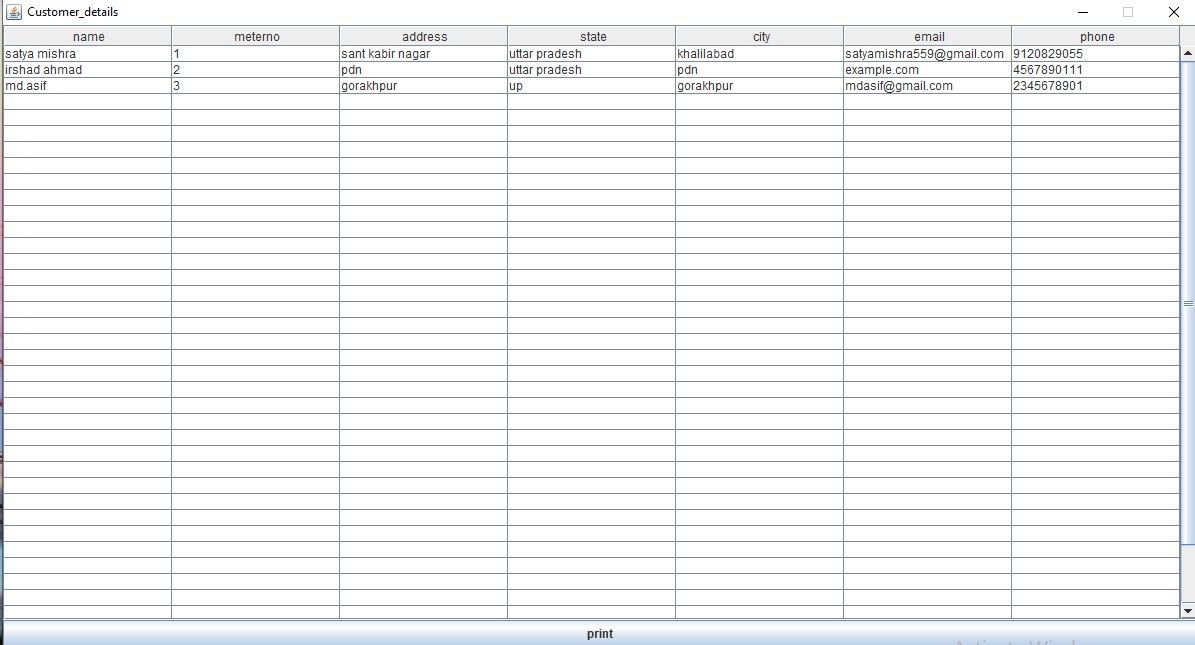
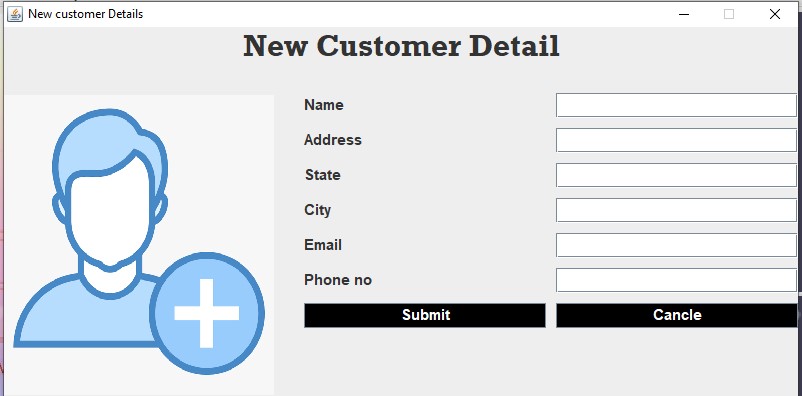
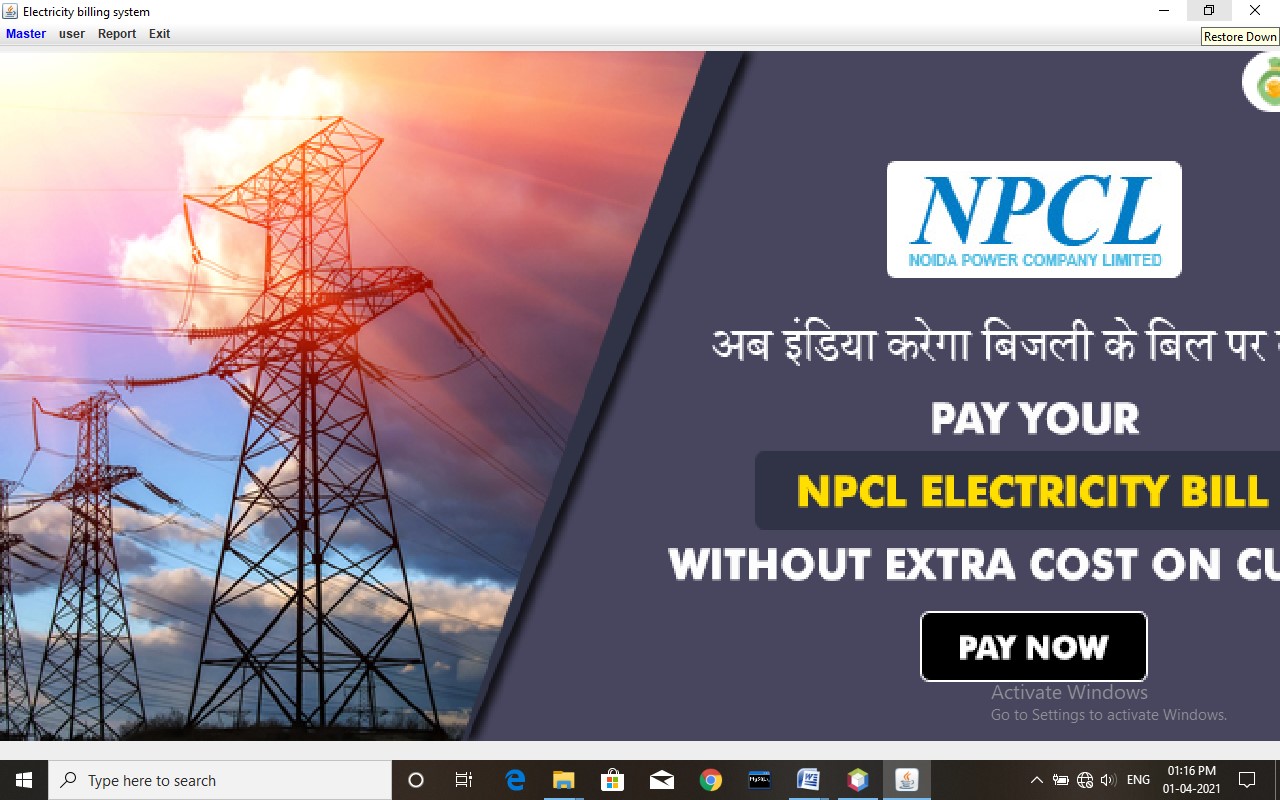
**Login\_info table :**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** |
| usr | Varchar(50) | No | PRI | NULL |  |
| name | Varchar(50) | Yes |  | NULL |  |
| gender | Varchar(10) | Yes |  | NULL |  |
| email | Varchar(50) | Yes |  | NULL |  |
| address | Varchar(100) | Yes |  | NULL |  |
| mobile | Varchar(14) | Yes |  | NULL |  |
| passwd | Varchar(50) | Yes |  | NULL |  |

**Tax table :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** |
| Meter\_location | Varchar(20) | Yes |  | NULL |
| Meter\_type | Varchar(20) | Yes |  | NULL |
| Phase\_code | int | Yes |  | NULL |
| Bill\_type | Varchar(15) | Yes |  | NULL |
| Month | int | No |  | NULL |
| Meter\_rent | int | Yes |  | NULL |
| Service\_rent | int | Yes |  | NULL |
| Gst | int | Yes |  | NULL |
| Mcb\_rent | int | Yes |  | NULL |
| Division\_name | int | Yes |  | NULL |

**SCREENSHOT**



**PROGRAMME CODES**

**//connection\_class.java**

package Electricity\_billing;

import java.sql.\*;

public class connectionclass {

Connection cn;

Statement stm;

public connectionclass()

{

try

{

Class.forName("com.mysql.cj.jdbc.Driver");

cn=DriverManager.getConnection("jdbc:mysql://localhost:3306/billing","root","");

stm=cn.createStatement();

// if(cn.isClosed())

// {

// System.out.println("closed");

// }

// else

// {

// System.out.println("open......");

// }

}

catch(Exception e)

{

e.printStackTrace();

}

}

public static void main(String s[])

{

new connectionclass();

}

}

//main class.java

package Electricity\_billing;

import java.awt.EventQueue;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JFrame;

import javax.swing.JLabel;

import java.awt.BorderLayout;

import javax.swing.ImageIcon;

import java.awt.Font;

import javax.swing.SwingConstants;

import javax.swing.JButton;

public class mainPage extends JFrame implements ActionListener{

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

mainPage window = new mainPage();

window.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

public mainPage() {

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

private JButton login,registration;

private void initialize() {

setBounds(90, 30, 1200, 700);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

getContentPane().setLayout(null);

JLabel lblNewLabel\_1 = new JLabel("Electricity Billing System");

lblNewLabel\_1.setBounds(0, 63, 1184, 48);

lblNewLabel\_1.setHorizontalAlignment(SwingConstants.CENTER);

lblNewLabel\_1.setFont(new Font("Cambria Math", Font.BOLD, 40));

getContentPane().add(lblNewLabel\_1);

login = new JButton("Login");

login.setFont(new Font("Tahoma", Font.BOLD, 11));

login.setBounds(209, 552, 326, 30);

getContentPane().add(login);

login.addActionListener(this);

registration = new JButton("Registration");

registration.setFont(new Font("Tahoma", Font.BOLD, 11));

registration.setBounds(648, 552, 326, 30);

getContentPane().add(registration);

registration.addActionListener(this);

JLabel lblNewLabel = new JLabel("");

lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);

lblNewLabel.setBounds(0, 0, 1184, 661);

lblNewLabel.setIcon(new ImageIcon(mainPage.class.getResource("/Electricity\_billing/image/screen3.png")));

getContentPane().add(lblNewLabel);

}

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

if(e.getSource().equals(login)) {

this.setVisible(false);

new loginpage().setVisible(true);

}else if(e.getSource().equals(registration)) {

this.setVisible(false);

new Registration().setVisible(true);

}

}

}

//Registration.java

package Electricity\_billing;

import java.awt.EventQueue;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.ResultSet;

import javax.swing.ButtonGroup;

import javax.swing.JButton;

import javax.swing.JComboBox;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextField;

import javax.swing.JRadioButton;

import java.awt.Font;

import javax.swing.ImageIcon;

import java.awt.Color;

public class Registration extends JFrame implements ActionListener{

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

Registration window = new Registration();

window.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

public Registration() {

initialize();

}

private JTextField userNameField,nameField,emailField,passField,confirmPassField;

private JButton button;

private JTextField mobileField;

private JTextField addressField;

private JRadioButton male,female;

private JButton cancel;

private void initialize() {

setTitle("Registration Form");

setBounds(100, 100, 450, 350);

setResizable(false);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

JLabel usrNameLabel = new JLabel("User Name ");

JLabel nameLabel = new JLabel("Name ");

JLabel emailLabel = new JLabel("Email ");

JLabel addressLabel = new JLabel("Address");

JLabel lblMobileNo = new JLabel("Mobile No. ");

JLabel passLabel = new JLabel("Password ");

JLabel confirmPassLabel = new JLabel("Re-type Password ");

userNameField = new JTextField();

nameField = new JTextField();

emailField = new JTextField();

passField = new JTextField();

confirmPassField = new JTextField();

usrNameLabel.setBounds(63, 40, 72, 20);

userNameField.setBounds(184, 40, 233, 20);

nameLabel.setBounds(63, 80, 77, 20);

nameField.setBounds(183, 80, 234, 20);

emailLabel.setBounds(63, 159, 64, 20);

emailField.setBounds(184, 159, 235, 20);

passLabel.setBounds(62, 290, 66, 20);

passField.setBounds(184, 290, 233, 20);

confirmPassLabel.setBounds(63, 330, 108, 20);

confirmPassField.setBounds(184, 330, 236, 20);

lblMobileNo.setBounds(63, 248, 65, 20);

getContentPane().add(lblMobileNo);

mobileField = new JTextField();

mobileField.setBounds(184, 248, 233, 20);

getContentPane().add(mobileField);

addressLabel.setBounds(63, 203, 65, 20);

getContentPane().add(addressLabel);

setBounds(450, 150, 500, 450);

addressField = new JTextField();

addressField.setBounds(184, 203, 233, 20);

getContentPane().add(addressField);

JLabel genderLabel = new JLabel("Gender");

genderLabel.setBounds(63, 121, 72, 20);

getContentPane().add(genderLabel);

ButtonGroup bg=new ButtonGroup();

male = new JRadioButton("Male");

male.setBounds(184, 120, 109, 23);

getContentPane().add(male);

female = new JRadioButton("Female");

female.setBounds(295, 120, 109, 23);

getContentPane().add(female);

bg.add(male);bg.add(female);

button = new JButton("Ragister");

button.setForeground(Color.BLUE);

button.setFont(new Font("Tahoma", Font.BOLD, 11));

button.setBounds(118, 376, 127 , 23);

button.addActionListener(this);

getContentPane().add(usrNameLabel);getContentPane().add(userNameField);

getContentPane().add(nameLabel);getContentPane().add(nameField);

getContentPane().add(emailLabel);getContentPane().add(emailField);

getContentPane().add(passLabel);getContentPane().add(passField);

getContentPane().add(confirmPassLabel);getContentPane().add(confirmPassField);

getContentPane().add(button);

getContentPane().setLayout(null);

cancel = new JButton("Cancel");

cancel.setForeground(Color.RED);

cancel.setFont(new Font("Tahoma", Font.BOLD, 11));

cancel.setBounds(256, 376, 127, 23);

getContentPane().add(cancel);

cancel.addActionListener(this);

}

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

if(e.getSource().equals(button)) {

String usrName = userNameField.getText();

String name = nameField.getText();

String email = emailField.getText();

String address = addressField.getText();

String mobile= mobileField.getText();

String pass = passField.getText();

String confirmPass = confirmPassField.getText();

String gender="";

if(usrName.isEmpty() || name.isEmpty() || email.isEmpty() || address.isEmpty() || mobile.isEmpty() || pass.isEmpty() || confirmPass.isEmpty()) {

JOptionPane.showMessageDialog(null, "All Field required", "Warrning ", JOptionPane.WARNING\_MESSAGE);

}else{

if(male.isSelected() || female.isSelected()) {

if(!pass.equals(confirmPass)) {

JOptionPane.showMessageDialog(null, "password should be same", "Warrning ", JOptionPane.WARNING\_MESSAGE);

}else {

connectionclass conn=new connectionclass();

String sql="select `usr` from `login\_info` where `usr`='"+usrName+"'";

try {

ResultSet data=conn.stm.executeQuery(sql);

if(data.next()) {

JOptionPane.showMessageDialog(null,"Opps ! this username already exits", "Warrning ", JOptionPane.WARNING\_MESSAGE);

}else {

if(male.isSelected()) {

gender="male";

}else {

gender="female";

}

String sql2="insert into `login\_info` values ('"+usrName+"','"+name+"','"+gender+"','"+email+"','"+address+"','"+mobile+"','"+pass+"')";

conn.stm.executeUpdate(sql2);

JOptionPane.showMessageDialog(null, "Congrats ! you are register successfull..", "Registerd", JOptionPane.PLAIN\_MESSAGE);

this.setVisible(false);

new mainPage().setVisible(true);

}

}catch(Exception ex) {

ex.printStackTrace();

}

}

}else {

JOptionPane.showMessageDialog(null, "please Select your gender", "Warrning ", JOptionPane.WARNING\_MESSAGE);

}

}

}else if(e.getSource().equals(cancel)) {

this.setVisible(false);

new mainPage().setVisible(true);

}

}

}

//login.java

package Electricity\_billing;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

import java.util.\*;

import java.io.\*;

public class loginpage extends JFrame implements ActionListener

{

JLabel l1,l2,l3;

JTextField t1;

JButton b1,b2;

JPanel p1,p2,p3,p4;

JPasswordField t2;

loginpage()

{

super("Admin Login");

setSize(400,250);

setResizable(false);

setLocation(500,250);

/\* ImageIcon img1=new ImageIcon("Electricity\_billing/image/screen2.jpg");

setIconImage(img1.getImage());\*/

l1=new JLabel("username");

l2=new JLabel("password");

ImageIcon i1=new ImageIcon(ClassLoader.getSystemResource("Electricity\_billing/image/login.png"));

Image img=i1.getImage().getScaledInstance(170, 170, Image.SCALE\_DEFAULT);

ImageIcon i2=new ImageIcon(img);

l3=new JLabel(i2);

t1=new JTextField(14);

t2=new JPasswordField(14);

ImageIcon i3=new ImageIcon(ClassLoader.getSystemResource("Electricity\_billing/image/logo7.png"));

Image img2=i3.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

b1=new JButton("Login",new ImageIcon(img2));

ImageIcon i4=new ImageIcon(ClassLoader.getSystemResource("Electricity\_billing/image/logo8.png"));

Image img3=i4.getImage().getScaledInstance(16, 16, Image.SCALE\_DEFAULT);

b2=new JButton("Cancle",new ImageIcon(img3));

Font f;

f =new Font("Arial",Font.BOLD,16);

l1.setFont(f);

l2.setFont(f);

b1.setFont(f);

b2.setFont(f);

t1.setFont(f);

t2.setFont(f);

p1=new JPanel();

p2=new JPanel();

p3=new JPanel();

p4=new JPanel();

setLayout(new BorderLayout());

p2.add(l1);

p2.add(t1);

p2.add(l2);

p2.add(t2);

add(p2,BorderLayout.CENTER);

p1.add(l3);

add(p1,BorderLayout.WEST);

p3.add(b1);

p3.add(b2);

add(p3,BorderLayout.SOUTH);

b1.addActionListener(this);

b2.addActionListener(this);

}

public void actionPerformed(ActionEvent ev)

{

try {

Class.forName("com.mysql.cj.jdbc.Driver");

Connection cn=DriverManager.getConnection("jdbc:mysql://localhost:3306/billing","root","");

if(ev.getSource()==b1)

{

String name=t1.getText();

String pass = t2.getText();

String q="select \* from login\_info where usr='"+name+"' and passwd='"+pass+"'";

Statement stm=cn.createStatement();

ResultSet set=stm.executeQuery(q);

if(set.next())

{

new homepage().setVisible(true);

this.setVisible(false);

}

else

{

JOptionPane.showMessageDialog(null,"invalid login");

setVisible(false);

}

}

else

{

this.setVisible(false);

new mainPage().setVisible(true);

}

}

catch(Exception e)

{

e.printStackTrace();

System.out.println("Error.........");

}

}

public static void main(String s[])

{

new loginpage().setVisible(true);

}

}

//new\_customer.java

package Electricity\_billing;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

import java.util.\*;

import java.io.\*;

public class new\_customer extends JFrame implements ActionListener{

JLabel j1,j2,j3,j4,j5,j6,j7,j8,j9;

JButton b1,b2;

JTextField t1,t2,t3,t4,t5,t6,t7,t8,t9;

JPanel p1,p2;

Font f,f1;

Choice c5;

new\_customer()

{

super("New customer Details");

setSize(800,400);

setLocation(40,40);

setResizable(false);

f=new Font("Arial",Font.BOLD,15);

f1=new Font("rockwell",Font.BOLD,30);

p1=new JPanel();

p1.setLayout(new GridLayout(9,2,10,10));

j1=new JLabel("Name");

t1=new JTextField(50);

j1.setFont(f);

t1.setFont(f);

p1.add(j1);

p1.add(t1);

j3=new JLabel("Address");

t3=new JTextField (50);

j3.setFont(f);

t3.setFont(f);

p1.add(j3);

p1.add(t3);

j4=new JLabel("State");

t4=new JTextField (15);

j4.setFont(f);

t4.setFont(f);

p1.add(j4);

p1.add(t4);

j5=new JLabel("City");

t5=new JTextField (15);

j5.setFont(f);

t5.setFont(f);

p1.add(j5);

p1.add(t5);

j6=new JLabel("Email");

t6=new JTextField (15);

j6.setFont(f);

t6.setFont(f);

p1.add(j6);

p1.add(t6);

j7=new JLabel("Phone no");

t7=new JTextField (15);

j7.setFont(f);

t7.setFont(f);

p1.add(j7);

p1.add(t7);

j9=new JLabel("New Customer Detail");

j9.setFont(f1);

j9.setHorizontalAlignment(JLabel.CENTER);

b1=new JButton("Submit");

b2=new JButton("Cancle");

b1.setFont(f);

b2.setFont(f);

p1.add(b1);

p1.add(b2);

b1.setBackground(Color.BLACK);

b1.setForeground(Color.WHITE);

b2.setBackground(Color.BLACK);

b2.setForeground(Color.WHITE);

p2=new JPanel();

p2.setLayout(new GridLayout(1,1));

ImageIcon img=new ImageIcon(ClassLoader.getSystemResource("Electricity\_billing/image/logo1.png"));

Image image=img.getImage().getScaledInstance(270,300,Image.SCALE\_DEFAULT);

ImageIcon img2=new ImageIcon(image);

j8=new JLabel(img2);

p2.add(j8);

setLayout(new BorderLayout(30,30));

add(j9,"North");

add(p1,"Center");

add(p2,"West");

b1.addActionListener(this);

b2.addActionListener(this);

}

public void actionPerformed(ActionEvent ev)

{

if(ev.getSource()==b1)

{

String name=t1.getText();

String address=t3.getText();

String state=t4.getText();

String city=t5.getText();

String email=t6.getText();

String phone=t7.getText();

try

{

connectionclass ob=new connectionclass();

String q="insert into customer\_info(`name`,`address`,`state`,`city`,`email`,`phone`) values('"+name+"' , '"+address+"' , '"+state+"' , '"+city+"' , '"+email+"' , '"+phone+"')";

ob.stm.executeUpdate(q);

}

catch(Exception ex)

{

ex.printStackTrace();

}

JOptionPane.showMessageDialog(null,"Employe data inserted......");

setVisible(false);

}

if(ev.getSource()==b2)

{

System.exit(0);

}

}

public static void main(String args[])

{

new new\_customer().setVisible(true);

}

}

//customer\_detail.java

package Electricity\_billing;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

import java.util.\*;

import java.io.\*;

public class customer\_details extends JFrame implements ActionListener{

int i=0,j=0;

JTable t1;

JButton b1;

String x[]={"name","meterno","address","state","city","email","phone"};

String y[][]=new String[20][8];

customer\_details()

{

super("Customer\_details");

setSize(1200,650);

setLocation(100,100);

setResizable(false);

try

{

connectionclass obj=new connectionclass();

String q="select \* from customer\_info";

ResultSet res=obj.stm.executeQuery(q);

while(res.next())

{

y[i][j++]=res.getString("name");

y[i][j++]=res.getString("meterno");

y[i][j++]=res.getString("address");

y[i][j++]=res.getString("state");

y[i][j++]=res.getString("city");

y[i][j++]=res.getString("email");

y[i][j++]=res.getString("phone");

i++;

j=0;

}

}

catch(Exception e)

{

e.printStackTrace();

}

t1=new JTable(y,x);

b1=new JButton("print");

add(b1,"South");

JScrollPane sp=new JScrollPane(t1);

add(sp);

b1.addActionListener(this);

}

public void actionPerformed(ActionEvent ev)

{

if(ev.getSource()==b1)

{

try

{

t1.print();

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

}

public static void main(String args[])

{

new customer\_details().setVisible(true);

}

}

//pay\_bill.java

package Electricity\_billing;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

import java.util.\*;

import java.io.\*;

public class pay\_bill extends JFrame implements ActionListener{

JLabel l1,l2,l3,l4,l5;

JButton b1,b2;

JTextArea t1;

Choice c1,c2;

JPanel p,p1;

Font f,f1;

pay\_bill()

{

super("Pay bill");

setSize(1000,400);

setLocation(200,200);

setResizable(false);

f=new Font("rockwell",Font.BOLD,17);

f1=new Font("rockwell",Font.ITALIC,30);

l1=new JLabel("Meter Number");

l2=new JLabel("Month");

l3=new JLabel("Units Consumed");

l5=new JLabel("Calculates Electricity Bill");

l5.setHorizontalAlignment(JLabel.CENTER);

l1.setFont(f);

l2.setFont(f);

l3.setFont(f);

l5.setFont(f1);

c1=new Choice();

try {

connectionclass ob = new connectionclass();

ResultSet data = ob.stm.executeQuery("select `meterno` from `customer\_info`");

while(data.next()) {

c1.add(data.getString(1));

}

}catch(Exception e) {

System.out.print("Connection problem");

}

c1.setFont(f);

c2=new Choice();

c2.add("January");

c2.add("Febuary");

c2.add("March");

c2.add("April");

c2.add("May");

c2.add("June");

c2.add("July");

c2.add("August");

c2.add("September");

c2.add("October");

c2.add("November");

c2.add("December");

c2.setFont(f);

t1=new JTextArea();

t1.setFont(f);

b1=new JButton("Submit");

b2=new JButton("Cancle");

b1.setBackground(Color.BLACK);

b1.setForeground(Color.WHITE);

b2.setBackground(Color.BLACK);

b2.setForeground(Color.WHITE);

b1.setFont(f);

b2.setFont(f);

b1.setSize(50,50);

p=new JPanel();

p.setLayout(new GridLayout(4,2,30,30));

p.add(l1);

p.add(c1);

p.add(l2);

p.add(c2);

p.add(l3);

p.add(t1);

p.add(b1);

p.add(b2);

p1=new JPanel();

p1.setLayout(new GridLayout(1,1));

ImageIcon ig=new ImageIcon(ClassLoader.getSystemResource("Electricity\_billing/image/logo3.jpeg"));

Image img=ig.getImage().getScaledInstance(270,270,Image.SCALE\_DEFAULT);

ImageIcon ig2=new ImageIcon(img);

l4=new JLabel(ig2);

p1.add(l4);

setLayout(new BorderLayout(30,30));

add(l5,"North");

add(p1,"West");

add(p,"Center");

b1.addActionListener(this);

b2.addActionListener(this);

}

public void actionPerformed(ActionEvent ev)

{

if(ev.getSource()==b1)

{

try

{

connectionclass obj=new connectionclass();

String meterno=c1.getSelectedItem();

String month=c2.getSelectedItem();

// This query for checking data already available of not...........

ResultSet billingData = obj.stm.executeQuery("select \* from `bill` where `meterno`="+meterno+" and `month` = '"+month+"'");

if(billingData.next()) {

JOptionPane.showMessageDialog(null, "This data already exists..", "Warning", JOptionPane.WARNING\_MESSAGE);

}else {

int units=Integer.parseInt(t1.getText());

int amount=(units\*7)+98+42+112+48;

String q="insert into bill value('"+month+"','"+units+"','"+amount+"','"+meterno+"')";

obj.stm.executeUpdate(q);

JOptionPane.showMessageDialog(null,"Data succesfull inserted\n --------------------------------"+"\n Meter Number = "+meterno+"\n Units = "+units+"\n Amount = "+amount);

setVisible(false);

}

}

catch(Exception ex)

{

// if field is empty then exception occured..................

JOptionPane.showMessageDialog(null, "All Field Required", "Warning",JOptionPane.WARNING\_MESSAGE);

}

}

if(ev.getSource()==b2)

{

JOptionPane.showMessageDialog(null,"You pressed cancel......");

setVisible(false);

}

}

public static void main(String args[])

{

new pay\_bill().setVisible(true);

}

}

//generate\_bill.java

package Electricity\_billing;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

import java.util.\*;

import java.io.\*;

public class generate\_bill extends JFrame implements ActionListener{

JLabel l1,l2;

JButton b1,b2;

JTextArea t1;

Choice c1,c2;

JPanel p,p1;

Font f;

generate\_bill()

{

super("Generate Bill");

setSize(450,700);

setResizable(false);

setLocation(20,20);

p=new JPanel();

f=new Font("senserif",Font.BOLD,16);

p1=new JPanel();

l1=new JLabel("meter\_no");

l2=new JLabel("month");

l2.setFont(f);

l1.setFont(f);

c1=new Choice();

connectionclass ob=new connectionclass();

String q="SELECT `meterno` FROM `customer\_info`";

try {

ResultSet set = ob.stm.executeQuery(q);

//Dynamic data adding on choice button....

while(set.next()) {

c1.add(set.getString(1));

// System.out.print(set.getString(1));

}

}catch(Exception e) {

e.printStackTrace();

}

c1.setFont(f);

c2=new Choice();

c2.add("january");

c2.add("febuary");

c2.add("march");

c2.add("april");

c2.add("may");

c2.add("june");

c2.add("july");

c2.add("august");

c2.add("september");

c2.add("october");

c2.add("november");

c2.add("december");

c2.setFont(f);

t1=new JTextArea(50,15);

t1.setFont(f);

ImageIcon i1=new ImageIcon(ClassLoader.getSystemResource("Electricity\_billing/image/logo7.png"));

Image img1=i1.getImage().getScaledInstance(20,20,Image.SCALE\_DEFAULT);

ImageIcon i2=new ImageIcon(img1);

b1=new JButton("PRINT",i2);

ImageIcon i3=new ImageIcon(ClassLoader.getSystemResource("Electricity\_billing/image/logo2.png"));

Image img2=i3.getImage().getScaledInstance(20,20,Image.SCALE\_DEFAULT);

ImageIcon i4=new ImageIcon(img2);

b2=new JButton("SHOW",i4);

b1.setFont(f);

b2.setFont(f);

setLayout(new BorderLayout());

p.add(l1);

p.add(c1);

p.add(l2);

p.add(c2);

add(p,"North");

p1.add(b1);

p1.add(b2);

add(p1,"South");

add(t1);

JScrollPane sp=new JScrollPane(t1);

add(sp);

b1.addActionListener(this);

b2.addActionListener(this);

}

public void actionPerformed(ActionEvent ev)

{

if(ev.getSource()==b2)

{

String meterno=c1.getSelectedItem();

String month=c2.getSelectedItem();

t1.setText("\tNoida Power Company Limited\nELECTRICITY BILL FOR THE MONTH OF "+month+",2020\n\n");

try

{

connectionclass con=new connectionclass();

String q1="select \* from customer\_info where meterno ='"+meterno+"'";

ResultSet rset2=con.stm.executeQuery(q1);

while(rset2.next())

{

t1.append("\nName :"+rset2.getString("name"));

t1.append("\nMeter\_number :"+rset2.getString("meterno"));

t1.append("\nAddress :"+rset2.getString("address"));

t1.append("\nState :"+rset2.getString("state"));

t1.append("\nCity :"+rset2.getString("city"));

t1.append("\nEmail :"+rset2.getString("email"));

t1.append("\nPhone :"+rset2.getString("phone"));

}

t1.append("\n------------------------------------------------------------------------");

String q2="select \* from tax";

ResultSet rset3=con.stm.executeQuery(q2);

while(rset3.next())

{

t1.append("\nMeter location :"+rset3.getString("meter\_location"));

t1.append("\nMeter type :"+rset3.getString("meter\_type"));

t1.append("\nPhase code :"+rset3.getString("phase\_code"));

t1.append("\nBill type :"+rset3.getString("bill\_type"));

t1.append("\nMonth :"+rset3.getString("month"));

t1.append("\nDivision name :"+rset3.getString("division\_name"));

t1.append("\n\n--------------------------------------------------------------------");

t1.append("\nMeter rent :"+rset3.getString("meter\_rent"));

t1.append("\nService rent :"+rset3.getString("service\_rent"));

t1.append("\nGST :"+rset3.getString("gst"));

t1.append("\nMCB rent :"+rset3.getString("mcb\_rent"));

}

t1.append("\n");

String q="select \* from bill where meterno ='"+meterno+"' and month='"+month+"'";

ResultSet rset4=con.stm.executeQuery(q);

if(rset4.next()) {

ResultSet rset41=con.stm.executeQuery(q);

while(rset41.next())

{

t1.append("\nMeter no : "+rset41.getString("meterno"));

t1.append("\nCurrent Month : "+rset41.getString("month"));

t1.append("\nUnits Consumed : "+rset41.getString("units"));

t1.append("\ncharges : "+rset41.getString("amount"));

t1.append("\n\n--------------------------------------------------------------------------");

t1.append("\nTotal PAYBILL : "+rset41.getString("amount"));

t1.append("\n\n\n Please Update Your Mobile Number You Can Pay \nyour Bill Online at Nearest JAN SUVIDHA KENDRA");

t1.setForeground(Color.black);

}

}else {

t1.append(" \n Please pay your due bill ! \n There are no any record found !"+"\n+-----------------------------------------------------------------------+");

t1.setForeground(Color.red);

}

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

if(ev.getSource()==b1)

{

try

{

t1.print();

}

catch(Exception exc)

{

exc.printStackTrace();

}

}

}

public static void main(String args[])

{

new generate\_bill().setVisible(true);

}

}

**CONCLUSION**

**Main achievements of this project:-**

Administrative staff can view all the details of consumer bill details and can search for any result. Different types of reports can be obtained from the report menu. Different types of reports can be obtained from the report menu. All manual and paper works in the billing branch can be avoided by implementing this software. Since this is built in java, it is platform independent and it can make workable by simply installing JRE.

ABOUT THIS PROJECT:-

1. It is simple and user friendly
2. Platform independent
3. vide scope for future expansion
4. manual as well as paper works can be fully eliminated in the billing branch
5. accuracy and reliability are surely increased
6. it make sure that unauthorized personal cannot execute this program

System security refers to the technical innovations and procedures applied to hardware and operating system. To protect against deliberate or accidental damage from a defined threat. In contrast, data security is the protection of data, some loss, disclosure, modification and destruction. The system security problem can be provided into four related issues.

1. Security
2. integrity
3. privacy
4. access procedures

Using the power tools of JAVA and SQL server “NPCL BILLING SYSTYEM” as developed with a high degree of accuracy and user friendliness. Though the system is developed for domestic and commercial tariffs, a full provision is given for a full fledged billing system used at any billing centre of NPCL. This software provides advancement in NPCL Boards revenue collection and accounting of the sale. In future all provided options can be included and activated to increase the functionality of the NPCL ELECTRICITY BILLING SYSTEM.