

Electricity Company Management System

Basaveshwara Engineering College
Bagalkot

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
BY : KARTIK A MASHAL
USN : 2BA18CS025 (4th SEM: B-DIV)
2019-2020
DATE : 07/08/2020

ABSTRACT :

*This project is aimed at developing **Electricity Company Management System** which is a very important for a company. The Electricity Company Management System is an internet based application that can be accessed throughout the organization and even we can operate it by sitting at our Home. This system is useful for all the members of the company including employees as well as the customer. This project mainly focuses on the easy user interface with greater efficiency.*

So., our project is mainly concerned with creating customer account, creating the employee account, deleting customer account, deleting employee account, accepting complaints from the customer, accepting new customer request, generating bills for the customer, the admin can view all the employee details, all customer details, all branch details, and our project also facilitates online payment of the bills through various payment options such as google pay, phone pay, debit card, credit card.

We also facilitate the user that he can also be able to view all his history i.e., all his previous paid bills whether he had paid or not, on what date he paid and even through which payment mode he has paid.



-: INDEX :-

1. Introduction.....	4
2. Literature Survey.....	5-6
3. Problem Definition.....	7
4. Aim of the Project	7
5. Proposed Model	8-12
• Customer Table	8
• Employee Table	9
• Branch table and Location Table	10
• Ward and Contact Table	11
• Meter and Complain Table	12
6. Implementation	13-23
• IDE	13
• Front End Tools and their features	14-15
• Back End Tools and their features.....	17-23
7. References	24

Introduction :

Electrical Energy Management System (EEMS) widely refers to a computer system which is designed specifically for the automated control and monitoring of electric power and utility system. The scope may span from a load dispatch center to a group of power networks. Most of these energy management systems also provide decision making facilities for operator in the operation and control in real time. The data obtained from such actions are used to train operators in a control center and for performing engineering studies for futuristic actions like planning, optimization and maintenance scheduling, etc. on a frequent basis and to produce trend analysis and annual consumption forecasts.

Electricity Company Management System (ECMS) is a collection of computerized tools used to monitor, control, and optimize the performance of generation and transmission systems. This intelligent energy management software control system is designed to reduce complexity of the user and the employees so that they can easily manage the data of the entire company, energy consumption, improve the utilization of the system, increase reliability, and predict electrical system performance. Energy. Energy Management System had its origin in the need for electric utility companies to operate their generators as economically as possible. To operate the system as economically as possible required that the characteristics of all generating units be available in one location so that the most efficient units could be dispatched properly along with the less efficient. In addition, there was a requirement that the on/off scheduling of generating units be done in an efficient manner as well. Energy management systems can also provide metering, sub metering, and monitoring functions that allow facility and building managers to gather data and insight that allows them to make more informed decisions about energy activities across their sites.

Literature Survey :

Gilberto P. Azevedo [1] et al explains in the software development area, as in most fields of the computer industry, new technologies are trumpeted as revolutionary solutions almost daily, just to disappear silently some time later. This was not the case with open-architecture energy management systems (EMS). About 10 years after their conception, they have proven to be a successful technological approach. But this does not mean that all problems have been solved; in fact, this is a dynamic research area, in continuous evolution and still raising challenges for the near future.

Xiaofeng He [2] et al describes the changing requirements due to privatization and deregulations have created needs for analysing information from different sources within DW. These needs require new high performance solutions represented by the new data warehouse of SCADA/EMS system and its characteristics and structure outlined in the 50 paper. Utilities have started to take advantage of this new technique and many other plans to follow. As the industry gains experience from this new tool new applications will evolve on the SCADA/EMS system.

Jian Wu [3] et al describes the Supervision Control and Data Acquisition (SCADA) system is a communication and control system used for monitoring, operation and maintenance of energy infrastructure grids. Compared with traditional applications, a SCADA system has a harsh deadline for critical tasks. There is special time constraint for the real time database used in a SCADA system. The real time database in SCADA extends traditional database to include in-memory database. Such real time database management are designed to operate in the harsh environment of real time systems, with strict requirements for resource utilization, and are ready to provide the performance and reliability required by real-life applications. In this paper, the main principle of real time database has been introduced. Its implementation in power system SCADA is discussed and a sample database is briefly introduced.

Jim see [4] et al describes the Electric utilities are finding it increasingly necessary to better monitor, analyse and control their distribution systems. Planning and operation of the grid is increasing in complexity on one hand but subject to ever more binding constraints on the other. Real-time analysis is being seen as necessary to achieve acceptable operational efficiencies and quality of service.

EASHY YANG [5] et al describes the queuing models are introduced for evaluation of the performance and design of the supervisory control and data acquisition (SCADA) systems. Emphasis is placed on the applications of the queuing theory to the design and analysis of the system. Two SCADA systems were implemented and evaluated. Based on the concept of concurrent processing, a two-processor-based SCADA system is presented and discussed. The results of this analysis imply that a cost/ performance close to the optimum may be achieved by the use of queuing models.

Srinivas Medida [6] et al describes the Supervisory Control and Data Acquisition (SCADA) Systems employ a wide range of computer and communication technologies. Advances in these technologies have helped in improving the effectiveness of SCADA. One such recent technological development is Internet and the World Wide Web. True to its name, it has been successful in casting a web over the wide variety of platforms and applications used in the entire Information Technology industry. The network thus achieved, paved a way for the common interfaces to access the data available on the net.

William J. Ackerman [7] describes the use of IEDs in substations will impose many changes on the design and implementation of the SCADA/EMS/DMS system. The increased data volume will require new methods for data transmission, processing and storage. New methods for checking out the transmission of data from a substation IED to a master station are required if the task is to be accomplished in a reasonable 52 amount of time. Newer EDs already incorporate facilities to accomplish this. On the positive side, the availability of additional data will lead to more accurate and reliable application programs such as state estimation, load flows, contingency analysis, etc. The improvements in these programs will, in the long run, result in the ability to operate a power system with greater safety, reliability and economy.

Problem Definition :

In India, the current electricity billing system is completely manual. The electric meters are situated in the houses, offices and factories etc. The energy meter reading is collected by meter readers on their fortnightly or monthly visits to the premises. This system has disadvantage of appointing meter reader to take the meter reading, effects consumer privacy etc. In this direction this project undertakes the meter reading without human intervention.

Aim of the project :

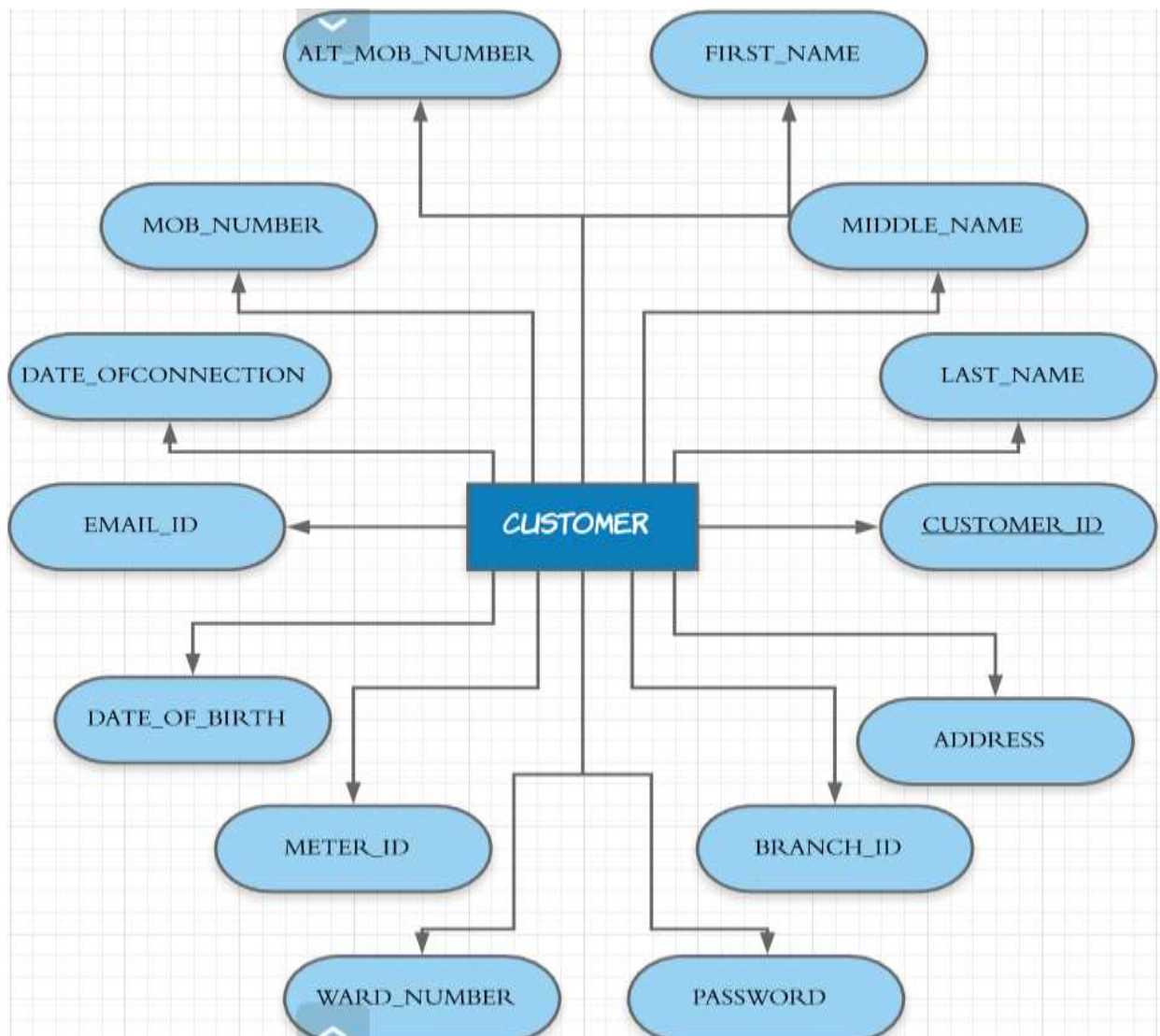
The objective of this project is

- To develop a internet based Automatic electricity billing system.
- The bill for the power consumed is automatically sent to the consumer Account.
- Within a limited period if the bill is not paid then the power will be automatically turned off.

Entity Relationship Diagrams :

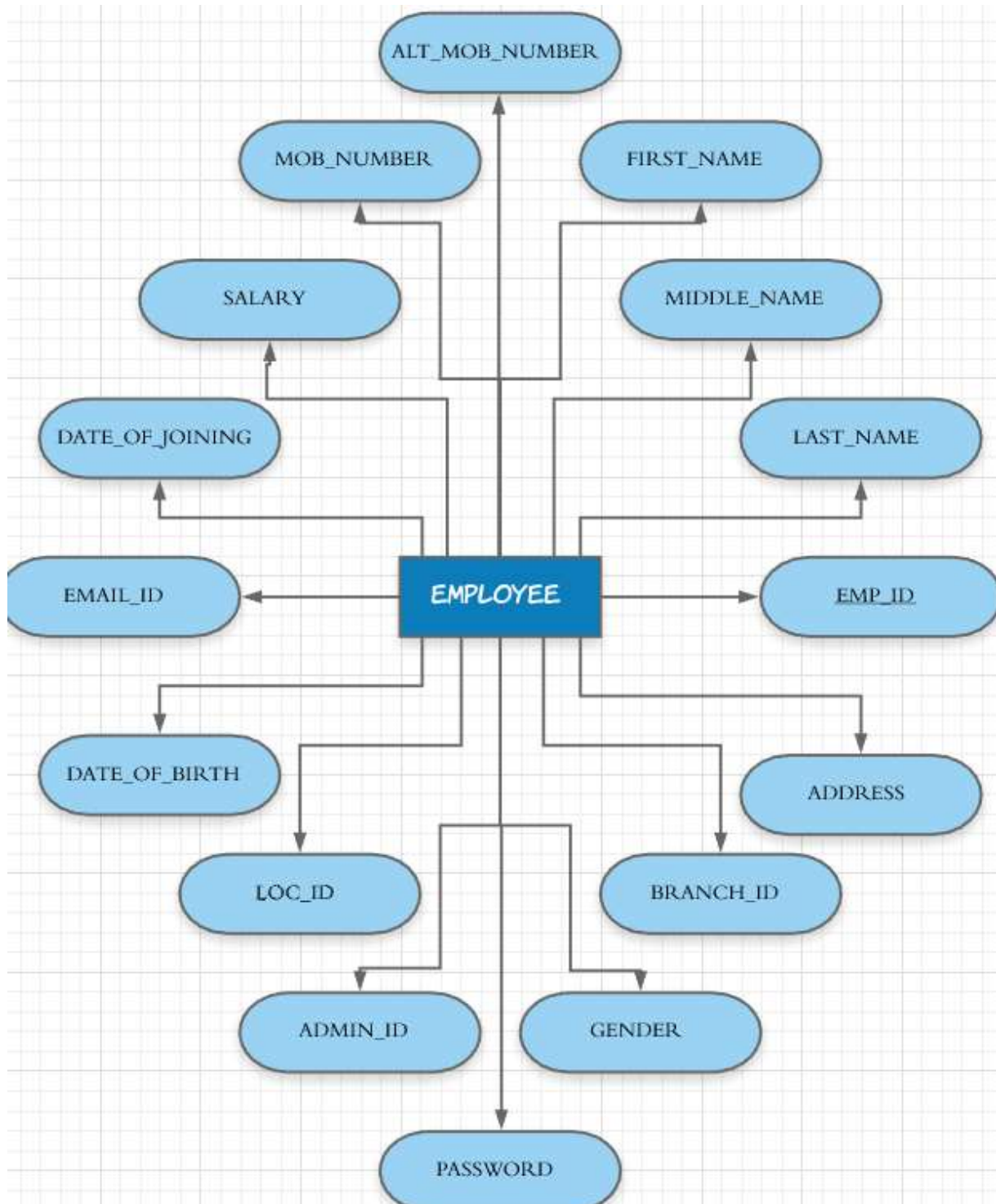
1. Customer Table :

The **Customer table** mainly consists attributes like first, middle and last name of the customer, address, Email id, contact number, date of birth, date of connection, customer id etc.. where customer id is the **PRIMARY KEY ATTRIBUTE** of the table and branch id is the **FOREIGN KEY ATTRIBUTE** referencing branch id of **branch** table..



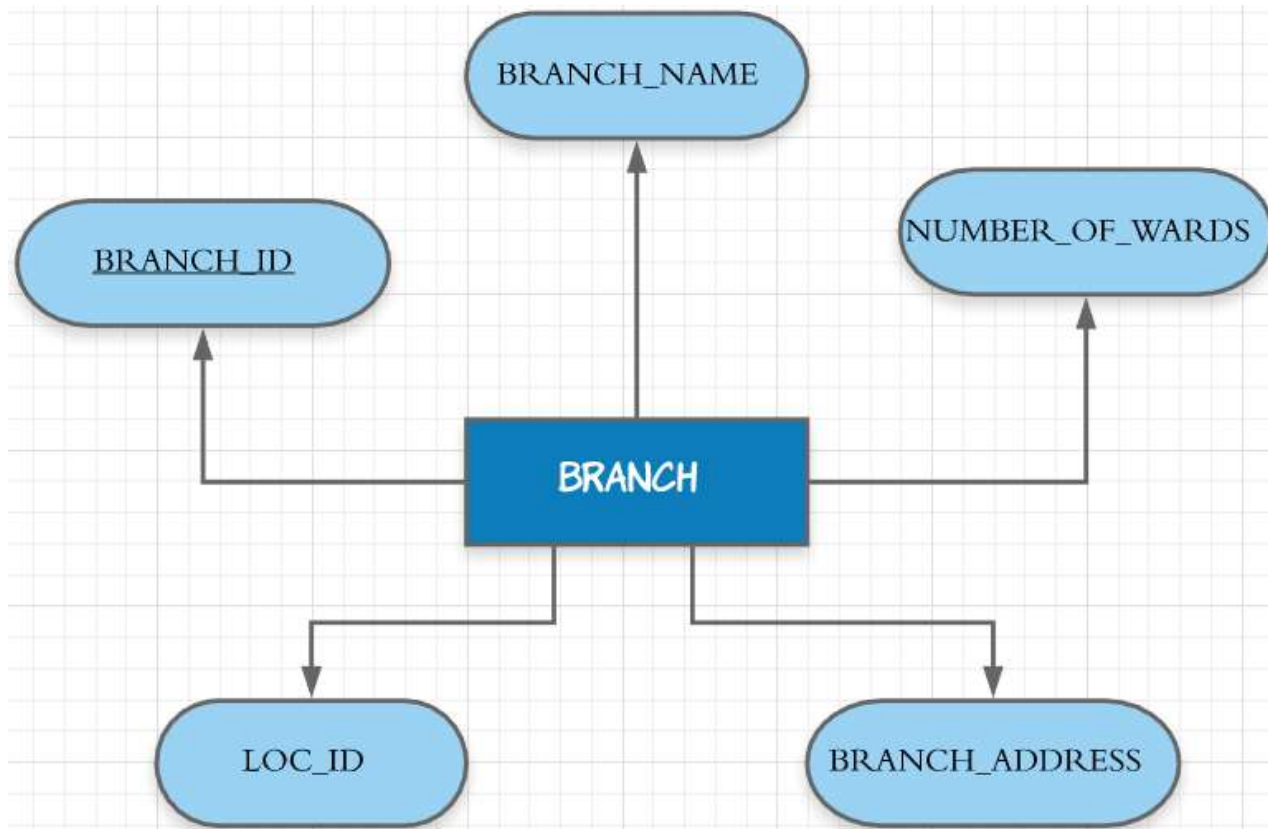
2. Employee Table :

The **Employee Table** mainly consists like first, middle, last name, date of joining, salary, employee id etc.. where employee id is the **PRIMARY KEY** and branch id, location id are the **FOREIGN KEY ATTRIBUTE** referencing the **branch id** of branch table and **location id** of location table.



3. Branch table :

The **branch table** mainly consists of branch name, number of wards, branch id, location id and branch address where branch id is the **PRIMARY KEY** attribute and location id is the **FOREIGN KEY** attribute referencing location id of location table.



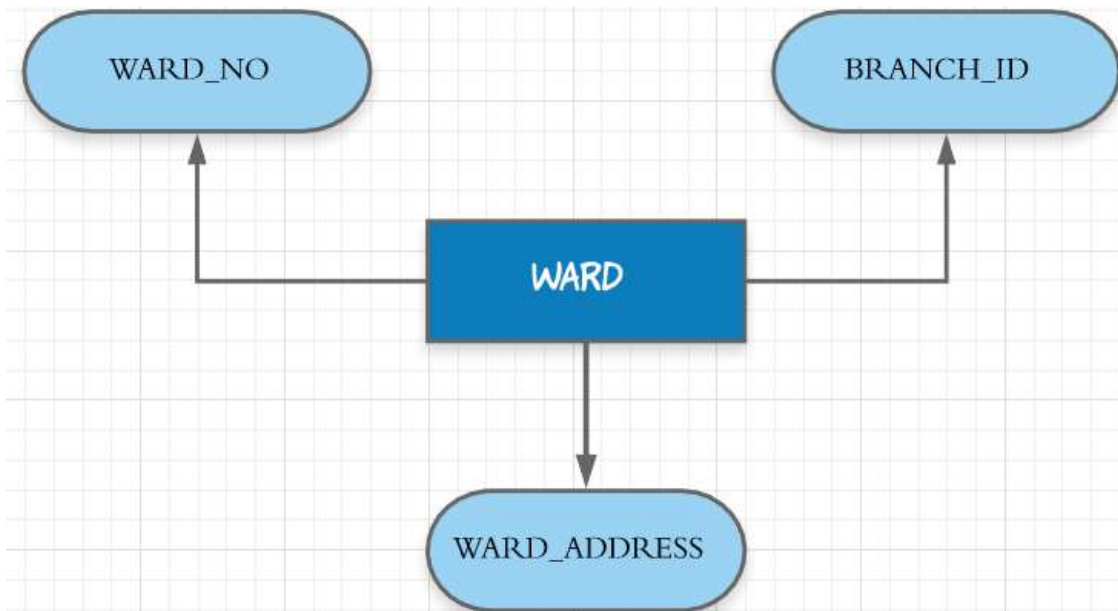
4. Location table :

In **Location table** there are only two attributes they are, Location id and Location name out of which location id is the **PRIMARY KEY** and there are **NO FOREIGN KEYS**.



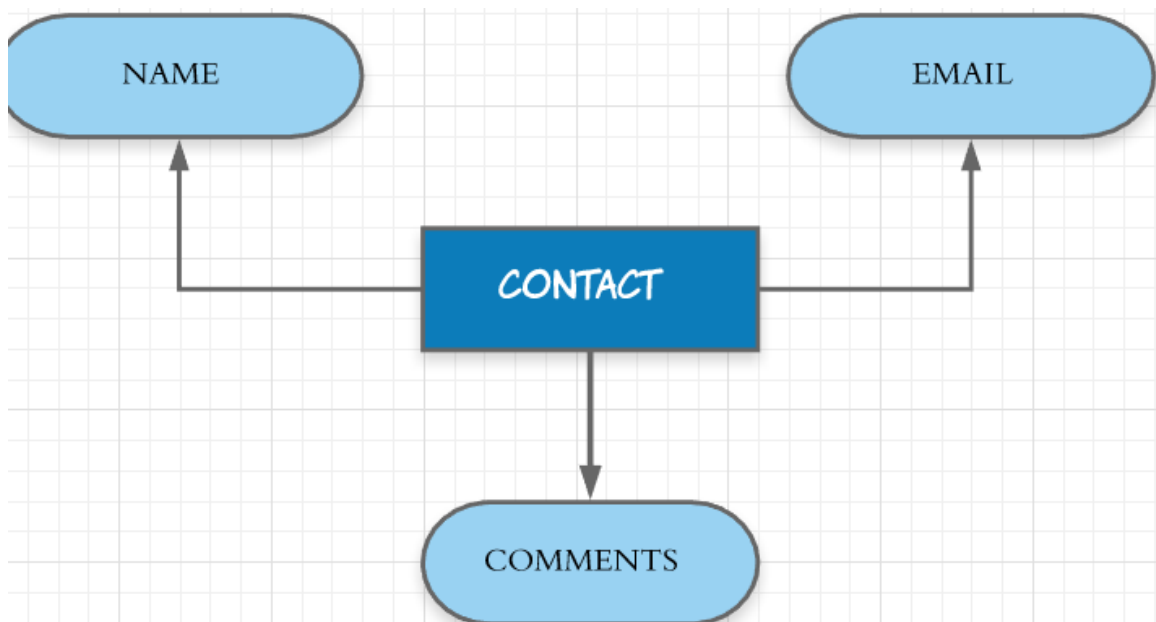
5. Ward table :

The **Ward table** consists of three attributes i.e., ward number, branch id and ward address where branch id is the **FOREIGN KEY** referencing the branch id of branch table.



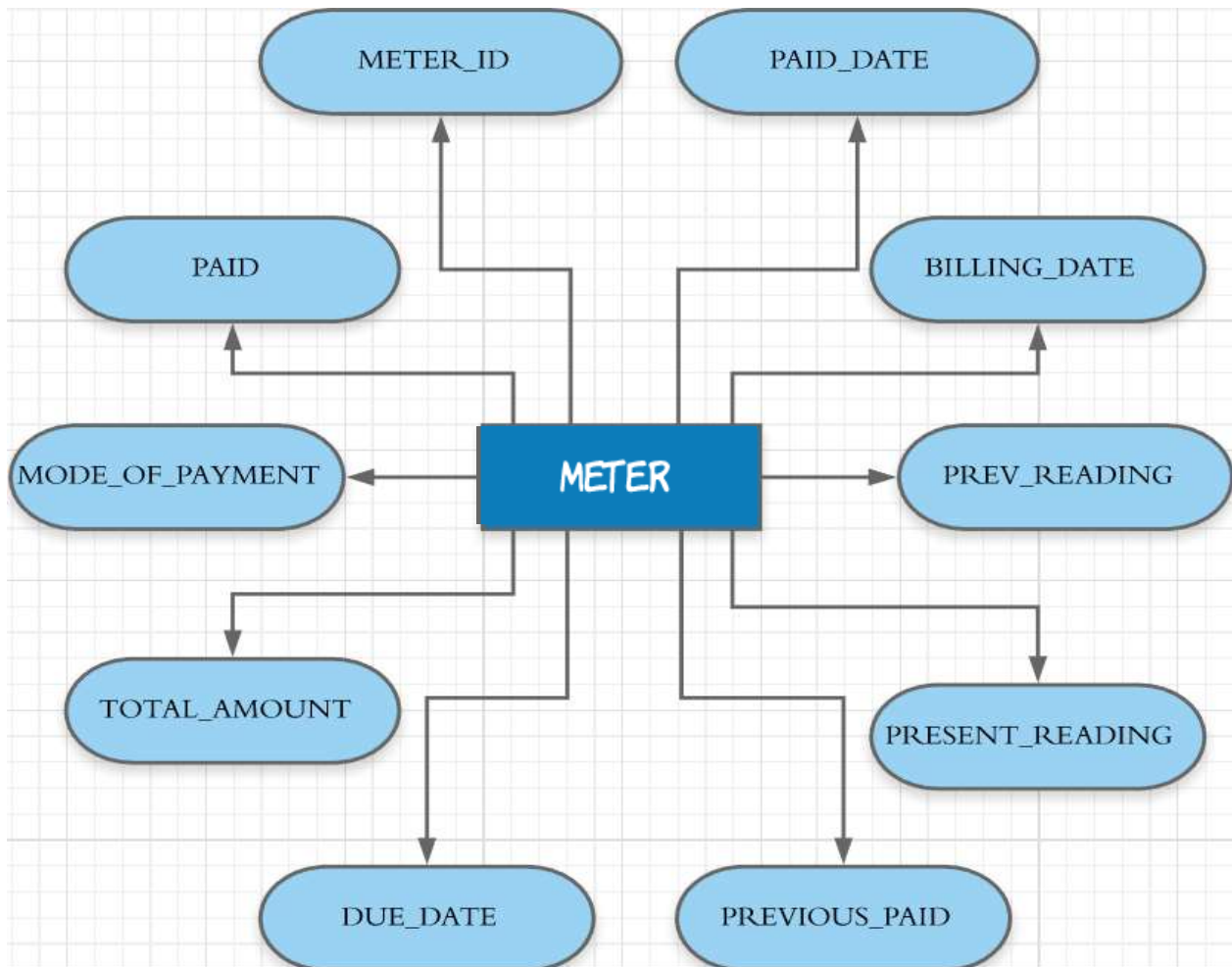
6. Contact table :

The **contact table** consists of three attributes name, email and comments. So that if a user needs have a connection or any queries they get connected.



7. Meter table :

The **meter table stores** complete history of all the customers including their previous bills, present bills paid or not, mode of payment etc.. so that every user can view his history and can be able to check.



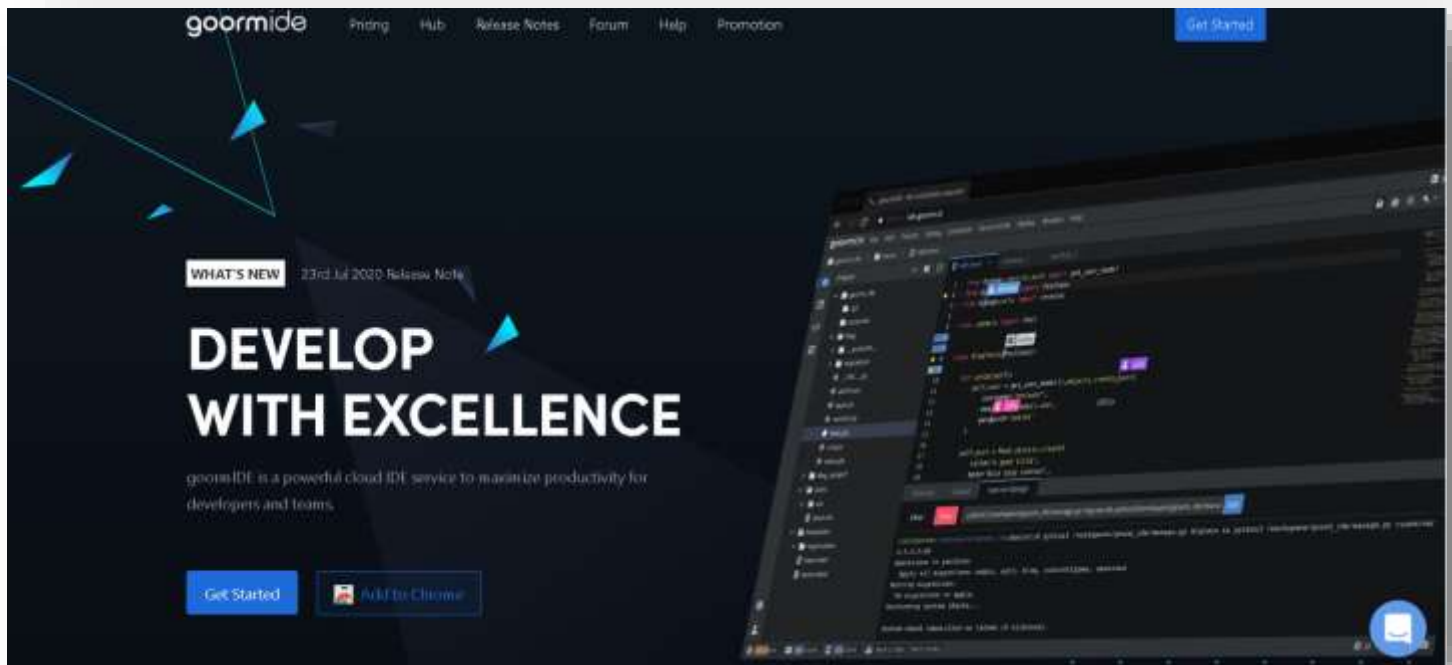
8. Complain Table :

The **complain table** stores the complaints of the customer and which can be visible for the admin of that location and is would be easy for him to resolve their problem and this table consists of customer id which is the **FOREIGN KEY** referencing customer id of customer table.



Implementation :

1. IDE used for project (GOORM IDE) :



goormIDE is a web-based cloud programming tool. It offers powerful collaboration capabilities, making it easy to program concurrently with other programmers.

With goormIDE, you can develop software more efficiently in a business unit. For example, developers can share opinions in real time through communication functions and edit the source code at the same time.

Also, developers and testers can fix bugs in the same project or files in real time and test it right away. In addition, customized goormIDE can increase efficiency of your businesses and organizations.

Front End Tools and their features :

1. HTML :



HTML stands for Hyper Text Markup Language. It is used to design web pages using markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. Markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most of markup (e.g. HTML) languages are human readable. Language uses tags to define what manipulation has to be done on the text.

HTML is a markup language which is used by the browser to manipulate text, images and other content to display it in required format. HTML was created by Tim Berners-Lee in 1991. The first ever version of HTML was HTML 1.0 but the first standard version was HTML 2.0 which was published in 1999.

Has the years passed now currently we have HTML5 which was introduced in 2014, which has more features than previous versions.

2. CSS :



Cascading Style **S**heets, fondly referred to as **CSS**, is a simply designed language intended to simplify the process of making web pages presentable. CSS allows you to apply styles to web pages. More importantly, CSS enables you to do this independent of the HTML that makes up each web page.

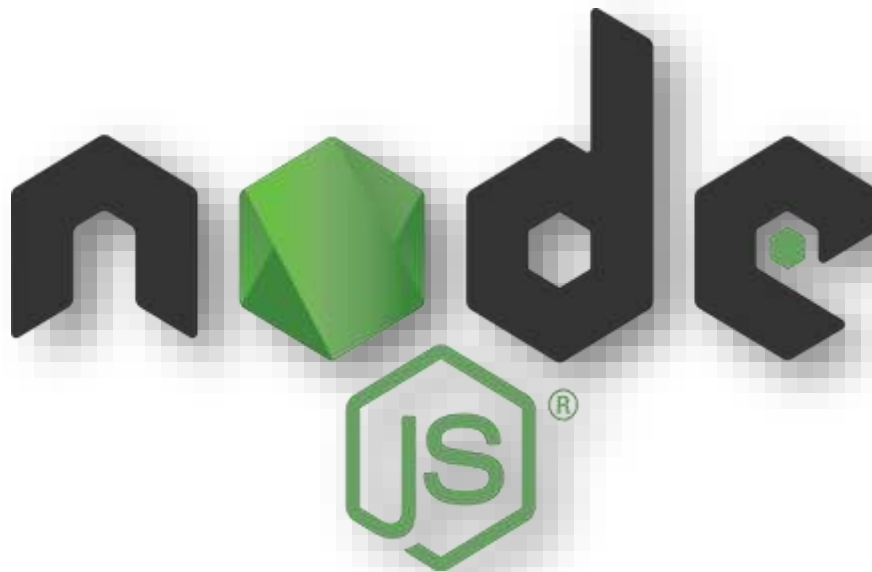
CSS is easy to learn and understood but it provides powerful control over the presentation of an HTML document.

WHY CSS?

- CSS saves time : You can write CSS once and reuse same sheet in multiple HTML pages.
- Easy Maintenance : To make a global change simply change the style, and all elements in all the webpages will be updated automatically.
- Search Engines : CSS is considered as clean coding technique, which means search engines won't have to struggle to "read" its content.
- Superior styles to HTML : CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
- Offline Browsing : CSS can store web applications locally with the help of offline cache. Using of this we can view offline websites.

Back end tools with its's features :

1. NODE.JS :



Node.js is a server-side platform built on Google Chrome's JavaScript Engine (V8 Engine). Node.js was developed by Ryan Dahl in 2009 . The definition of Node.js as supplied by its [official documentation](#) is as follows –

Node.js is a platform built on [Chrome's JavaScript runtime](#) for easily building fast and scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.

Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

Node.js = Runtime Environment + JavaScript Library.

Features of Node.js :

Following are some of the important features that make Node.js the first choice of software architects.

- Asynchronous and Event Driven – All APIs of Node.js library are asynchronous, that is, non-blocking. It essentially means a Node.js based server never waits for an API to return data. The server moves to the next API after calling it and a notification mechanism of Events of Node.js helps the server to get a response from the previous API call.
- Very Fast – Being built on Google Chrome's V8 JavaScript Engine, Node.js library is very fast in code execution.
- Single Threaded but Highly Scalable – Node.js uses a single threaded model with event looping. Event mechanism helps the server to respond in a non-blocking way and makes the server highly scalable as opposed to traditional servers which create limited threads to handle requests. Node.js uses a single threaded program and the same program can provide service to a much larger number of requests than traditional servers like Apache HTTP Server.
- No Buffering – Node.js applications never buffer any data. These applications simply output the data in chunks.

2. Express Frame Work :



[Express](#) is the most popular [Node](#) web framework, and is the underlying library for a number of other popular [Node web frameworks](#). It provides mechanisms to:

- Write handlers for requests with different HTTP verbs at different URL paths (routes).
- Integrate with "view" rendering engines in order to generate responses by inserting data into templates.
- Set common web application settings like the port to use for connecting, and the location of templates that are used for rendering the response.
- Add additional request processing "middleware" at any point within the request handling pipeline.

While [Express](#) itself is fairly minimalist, developers have created compatible middleware packages to address almost any web development problem. There are libraries to work with cookies, sessions, user logins, URL parameters, POST data, security headers, and [many](#) more.

3. MySQL :



MySQL is a relational database management system (RDBMS) based on the SQL (Structured Query Language) queries. It is one of the most popular languages for accessing and managing the records in the table. MySQL is open-source and free software under the GNU license. Oracle Company supports it.

The following are the most important features of MySQL:

[MySQL](#) is a relational database management system. This database language is based on the [SQL](#) queries to access and manage the records of the table.

a) Easy to use

MySQL is easy to use. We have to get only the basic knowledge of SQL. We can build and interact with MySQL by using only a few simple SQL statements.

b) It is secure

MySQL consists of a solid data security layer that protects sensitive data from intruders. Also, passwords are encrypted in MySQL.

c) Client/ Server Architecture

MySQL follows the working of a client/server architecture. There is a database server (MySQL) and arbitrarily many clients (application programs), which communicate with the server; that is, they can query data, save changes, etc.

d) Free to download

MySQL is free to use so that we can download it from MySQL official website without any cost.

e) It is scalable

MySQL supports multi-threading that makes it easily scalable. It can handle almost any amount of data, up to as much as 50 million rows or more. The default file size limit is about 4 GB. However, we can increase this number to a theoretical limit of 8 TB of data.

f) Speed

MySQL is considered one of the very fast database languages, backed by a large number of the benchmark test.

g) High Flexibility

MySQL supports a large number of embedded applications, which makes MySQL very flexible.

h) Compatible on many operating systems

MySQL is compatible to run on many operating systems, like Novell NetWare, Windows* Linux*, many varieties of UNIX* (such as Sun* Solaris*, AIX, and DEC* UNIX), OS/2, FreeBSD*, and others. MySQL also provides a facility that the clients can run on the same computer as the server or on another computer (communication via a local network or the Internet).

i) Allows roll-back

MySQL allows transactions to be rolled back, commit, and crash recovery.

j) Memory efficiency

Its efficiency is high because it has a very low memory leakage problem.

k) High Performance

MySQL is faster, more reliable, and cheaper because of its unique storage engine architecture. It provides very high-performance results in comparison to other databases without losing an essential functionality of the software. It has fast loading utilities because of the different cache memory.

l) High Productivity

MySQL uses Triggers, Stored procedures, and views that allow the developer to give higher productivity.

m) Platform Independent

It can download, install, and execute on most of the available operating systems.

n) Partitioning

This feature improves the performance and provides fast management of the large database.

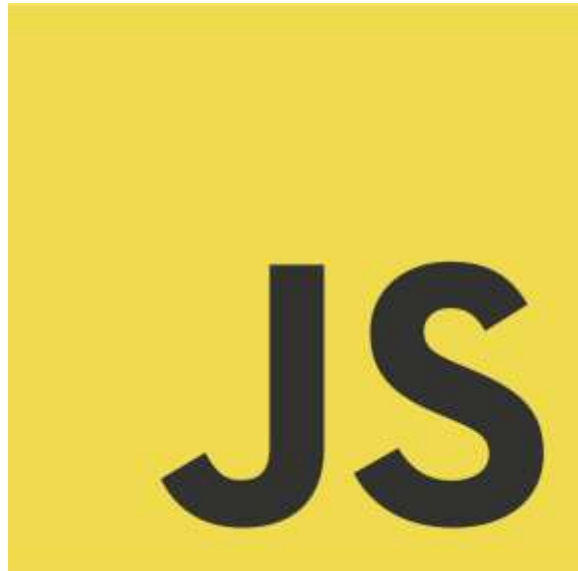
o) GUI Support

MySQL provides a unified visual database graphical user interface tool named "**MySQL Workbench**" to work with database architects, developers, and Database Administrators. [MySQL Workbench](#) provides SQL development, data modeling, data migration, and comprehensive administration tools for server configuration, user administration, backup, and many more. MySQL has a fully GUI supports from MySQL Server version 5.6 and higher.

p) Dual Password Support

MySQL version 8.0 provides support for dual passwords: one is the current password, and another is a secondary password, which allows us to transition to the new password.

4. Java Script :



- JavaScript often abbreviated as JS, is a [programming language](#) that conforms to the [ECMAScript](#) specification.^[7] JavaScript is [high-level](#), often [just-in-time compiled](#), and [multi-paradigm](#). It has [curly-bracket syntax](#), [dynamic typing](#), [prototype-based object-orientation](#), and [first-class functions](#).
- Alongside [HTML](#) and [CSS](#), JavaScript is one of the core technologies of the [World Wide Web](#).^[8] JavaScript enables interactive [web pages](#) and is an essential part of [web applications](#). The vast majority of [websites](#) use it for [client-side](#) page behavior,^[9] and all major [web browsers](#) have a dedicated [JavaScript engine](#) to execute it.
- As a multi-paradigm language, JavaScript supports [event-driven](#), [functional](#), and [imperative programming styles](#). It has [application programming interfaces](#) (APIs) for working with text, dates, [regular expressions](#), standard [data structures](#), and the [Document Object Model](#) (DOM). However, the language itself does not include any [input/output](#) (I/O), such as [networking](#), [storage](#), or [graphics](#) facilities, as the host environment (usually a web browser) provides those APIs.
- JavaScript engines were originally used only in web browsers, but they are now embedded in some [servers](#), usually via [Node.js](#). They are also embedded in a variety of applications created with [frameworks](#) such as [Electron](#) and [Cordova](#).
- Although there are similarities between JavaScript and [Java](#), including language name, [syntax](#), and respective [standard libraries](#), the two languages are distinct and differ greatly in design.

References :

- <https://ide.goorm.io/>
- <https://www.codecademy.com/learn/learn-html>
- <https://www.codecademy.com/learn/learn-css>
- <https://www.tutorialspoint.com/nodejs/index.htm>
- <https://www.javatpoint.com/expressjs-tutorial>
- <https://www.tutorialspoint.com/mysql/index.htm>
- <https://stackoverflow.com/>