DECLARATION

I Prashant Parashar (1BY18CS110) hereby declare that the Internship report entitled "**EMPLOYEE MANAGEMENT SYSTEM**" submitted to the BMS Institute of Technology & Management, Yelahanka, Bengaluru has been carried out by me and submittedin partial fulfillment of the course requirements for the award of degree in Bachelor of Engineering in Computer Science and Engineering of Visvesvaraya Technological University, Belagavi, during the academic year 2021 - 2022. The matter embodied in this report has not been submitted to any other university or institution for the award of any other degree.

Signature of the Student

Prashant Parashar
1BY18CS110

ACKNOWLEDGEMENT

I am happy to present this report on the internship program. This Program would not have been possible without the guidance, assistance, and suggestions of many individuals. I would like to express my deep sense of gratitude and indebtedness to each and every one who has helped during my internship program.

I gratefully thank my Internal Guide, **Dr Anjan Krishnamurthy, Assistant Professor, Department of Computer Science and Engineering BMSIT&M** for his guidance, support, and advice.

I sincerely convey my thanks to **Dr. Thippeswamy G, Head of the Department, of Computer Science and Engineering, BMSIT&M** for his constant encouragement and support.

I express my heartfelt gratitude and sincere thanks to **Dr. Mohan Babu G. N, Principal of BMSIT&M** for his constant encouragement and inspiration.

I heartily thank my External Guide **Mr. Sharath Puthiyakunnon, Product Manager, Epicor Software,** for his constant support, encouragement, and advice.

Finally, I would like to thank my parents, friends, and all those who are involved in the successful completion of the internship program.

Prashant Parashar

1BY18CS110

CERTIFICATE

DocuSign Envelope ID: F4EF31D0-7A83-4AF3-87F0-A0E890D0BE27

EPICOR

February 24, 2022

PERSONAL AND CONFIDENTIAL TO:

Prashant Parashar
Behind high school, ganesh ward, Kareli
Narsinghpur
487221

Appointment Letter for Internship

Dear **Prashant**,

We are pleased to offer you an internship opportunity with Epicor Software India (P) Limited ("Epicor" Or "The Company") pursuant to the terms and conditions set out in this Internship Letter.

Terms & Conditions

Position and Commencement of Employment

In your role as an **"Intern"**, you will report to a designated Manager. We expect your first day of internship to be **March 01, 2022** (the "Start Date"). Your place of posting will be at Epicor's Bangalore office.

You will be governed by the rules, regulations and other Company policies (together the "Company Policy") of Epicor as applicable, enforced, amended or altered from time to time during the course of your internship. This offer of Internship is subject to the following documentation being signed and obtained from you before or on the day of joining (as outlined below you on or before the agreed upon start date.

- Worldwide Code of Business Conduct
- Proprietary Rights Agreement
- Proof of your stated qualifications

Please bring these documents with you on the day of Commencement of Internship.

1. Remuneration

Starting from the date of Internship, you will be paid a Fixed monthly stipend of **INR 25,000** . The remuneration will be taxable as per the existing

ABSTRACT

"Employee Salary Management System" is designed to make the existing manual system automatic with the help of computerized equipment and full-edged computer software, fulfilling their requirements, so that their valuable data and information can be stored for a longer period with easy access and manipulation of the same.

The required software is easily available and easy to work with. This web application can maintain and view computerized records without getting redundant entries. The project describes how to manage user data for good performance and provide better services for the client.

TABLE OF CONTENTS

Description	Page No.
Acknowledgement	**i**
Internship Certificate	**ii**
Abstract	**iii**
Table of Contents	**iv**
List of Figures	**v**
List of Tables	**v**

Chapter No.	Chapter Name	Page No.
1	**About the company**	**1 - 5**
	1.1 About Company	1
	1.2 Expertise	2-4
	1.3 Services	5
	1.4 Products	5
2	**About the Department**	**6 - 10**
	2.1 Department insight	6
	2.2 Software development	7-9
	2.3 FullStack using C#	9-10
3	**Tasks performed**	**11 – 25**
	3.1 Introduction	11
	3.2 Technologies used	12-13
	3.3 Requirement Specification	14
	3.4 Design Methodology	15
	3.5 Implementation	16-19
	3.6 Results and Discussions (Include Snapshots)	19-25
4	**Reflection Notes**	**26-29**
5	**Conclusion**	
	References	

List of Figures

Figure Number	Figure Names	Page No
3.6.1	Front Page	20
3.6.2	Add Leave Form	20
3.6.3 3.6.3.1 3.6.3.2	Login Page Login Successful Login Unsuccessful	21
3.6.4 3.6.4.1 3.6.4.2	Department Table Add Department Form Edit Department Form	22-23
3.6.5	Employee Table	24
3.6.6	Paygrade Table	24
3.6.7	Payroll Table	25
3.6.8	Leave Table	25

List of Tables

Table Number	Table Name	Page No
3.5.4.1	Department Table	17
3.5.4.2	Employee Table	17
3.5.4.3	Paygrade Table	18
3.5.4.4	Payroll Table	18
3.5.4.5	Leave Table	18
3.5.4.6	Login Table	19

CHAPTER 1
ABOUT THE COMPANY

1.1 About Epicor Software

Epicor Software Corporation is a business software company based in Austin; Texas founded in 1972. Its products are aimed at the manufacturing, distribution, retail and services industries.

Epicor provides enterprise resource planning (ERP), customer relationship management (CRM), supply chain management (SCM), and human capital management (HCM) software to business customers in both software as a service (SaaS) and on-premises deployment models.

Epicor also released the plan 'Fit for the Future,' intended to support stakeholders in the group develop market sustainability and thrive in the new environment.

Providers involve resellers, device integrators, and business partners who can use the Epicor software range and services to introduce, plan, and build transformational IT strategies.

Stephen Murphy (CEO)

Steve Murphy joined Epicor Software Corporation as Chief Executive Officer in October 2017, bringing over 20 years of technology industry executive management experience to the role. As CEO, Murphy is responsible for providing a long-term strategic vision for the company-with a focus on customer experience and delivering innovative products, services and support that drive business growth.

Before joining Epicor, Murphy served as president of OpenText, a $2B global leader in enterprise information management with 12,000 employees worldwide, where he was responsible for all customer-facing activities including sales, marketing, partners and alliances, global technical services, customer support, professional services, sales operations and cloud etc.…

1.2 Expertise

Aerospace And Defense

Epicor understands that as an aerospace and defense (A&D) manufacturer, you must identify, consider, and respond to a new set of challenges each day. Change in this industry is a given; managing the change with tools and services to distinguish your business from local and global competition is fundamental. With increasing pressure to reduce prices in a business climate of expanding raw material cost and reduced labor availability, businesses like yours are streamlining and adopting new technology to automate business processes for more competitive lead times and to reduce waste in the organization

Automotive

The automotive manufacturing industry continues to exceed expectations worldwide. However, several key challenges are present, including new technically advanced vehicles, demographic shifts and preferences, and hybrid models and options. Epicor has the software solutions to maximize your efficiency and flexibility and keep your business on pace with the global automotive market.

Discrete Manufacturing

Discrete manufacturing is a complex business. Epicor has powerful software solutions with the functionality to manage that complexity—available in the cloud, on premises or as a managed service.

- Optimize lean manufacturing to focus on the key priorities to make smarter decisions, eliminate waste and increase customer satisfaction
- Track, measure, and monitor your entire business, from shop floor to top floor and from raw materials to final product
- Access and visualize critical information anytime or anywhere from any device
- Improve production planning to eliminate bottlenecks and idle equipment
- Reduce costs and streamline processes while you grow revenue and increase profits.

Construction and Engineering

The Engineering and Construction industry is an ever-shifting industry sector, marked by cycles of growth and decline, and, most recently, acquisition and consolidation. Experts are optimistic and claim that engineering and construction firms will make increasing investments in new technologies for more efficient operations and to compete in the global market. Epicor has the software solutions you need to differentiate your business and meet or exceed market demand.

- Complete visibility of project-based resources including assets, inventory, materials and labor.
- Comprehensive project management, including multilevel work breakdown structures, robust resource scheduling, and effective contract management.
- Accurate, timely billing of project costs through seamless collection of time, material, and expense transactions.
- Efficient bid management through embedded customer relationship management (CRM) capabilities including visibility of historical bids.
- Flexible, strong revenue recognition capabilities including profitability tracking and margin control.
- Comprehensive asset management, which improves reliability and reduces costs to maintain equipment.

Industrial Machinery

Thousands of companies worldwide manufacture the sophisticated machines used in agriculture, mining, construction, power generation and oil and gas production. A commitment to technical innovation is essential to continued leadership in a competitive global marketplace. Epicor has the software solutions to help manage complex engineering requirements, ensure production efficiency, and meet or exceed customer expectations while keeping costs low.

- Support lean manufacturing and automate business processes to reduce waste in the enterprise, resulting in reduced labor and material costs

- Satisfy customer requirements, control costs, and improve processes with a robust quality management system
- Track detailed cost estimates and capture actual costs throughout the life of the contract or project for comparison against the estimated costs
- Integrated Product Lifecycle Management (PLM), helps you control complex designs and specifications and minimize the impacts of engineering changes
- Improve production management, review and update production plan as demand changes across your multi-site and global operations.

Make To Order Manufacture

Make-to-order (MTO) manufacturers operate in a dynamic landscape where a number of trends new and old are shaping the future. You need to keep pace with these changes and take advantage of them or risk falling behind. In this eBook, we look at how an enterprise resource planning (ERP) solution can help you:

- Deliver a great customer experience
- Operate as efficient as possible
- Drive business growth and profitability.

Process Manufacturing

Today's process manufacturers are employing ingenuity, agility and technology to succeed in an ever-evolving environment of regulatory requirements and consumer trends. Gains in productivity and reduction in time to market are key concerns. Epicor has flexible software solutions that enable you to improve the efficiency of your supply chain and as well as increase visibility and end-to-end traceability.

- Manage your entire business processes end-to-end from time of order through production and delivery
- Improve yield and quality in recipe-based manufacturing
- Accommodate rapid changes in customer and consumer trends to maximize sales growth

- Plan and control stock, production, planning and distribution
- Manage, maintain, and improve production accurately and in real time
- Meet tough regulatory standards where origin and use of every ingredient must be recorded
- Forecast demand pull scheduling and cost allocation across co-products and by-products.

1.3 Services

- Business Consulting Services
- Education Services
- Cybersecurity
- Upgrades and Migrations
- Implementation Services
- Product Development Services

1.4 Products

- Epicor Kinetic
- Epicor iScala
- Epicor Advanced MES
- Epicor iNet Suite
- Epicor ECM
- Epicor PartExpert GFX
- Epicor Vision
- Company
- Epicor Commerce Connect
- Epicor XL Connect
- Epicor HCM
- Clientele
- Vantage

CHAPTER-2

ABOUT THE DEPARTMENT

2.1 Department Insight

Epicor Software has different departments of work. The employees undergo training of basics for 1 to 2 Months and later get into the development phase. Most of the projects have an analysis, design, implementation, testing, validation and verification phase of each module. In meetings, any new ideas or any kind of improvements that are voiced by the employees get reflected to the implementation before the next phase gets started.

2.2 Software Development

There are a series of roles that exist in most software development processes. One team member may be filling many roles and some roles may be suppressed for a specific type of project, but all these roles exist in one form or another in every software development projects. Of course, each organization has its own take on these roles.

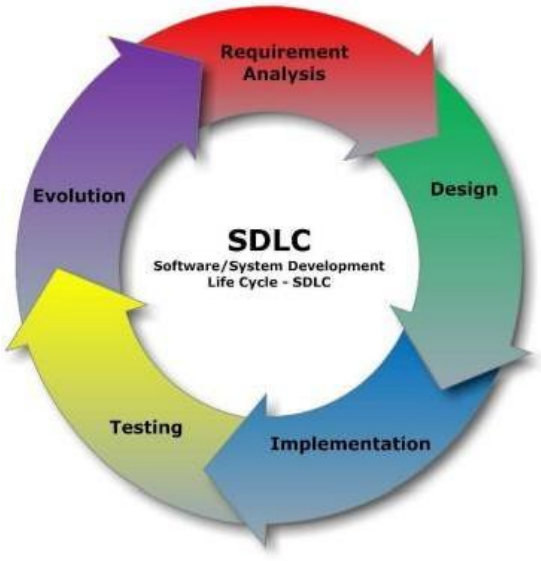


Figure 2.1 Software Development Cycle

As shown in Figure 2.1, Epicor Software has practiced a life cycle of developing any software or an application and that is followed as explained below: Software engineering task bridging the gap between system requirements engineering and software design. Provides software designer with a model of: • system information • function • behavior Model can be translated to data, architectural, and component-level designs. Expect to do a little bit of design during analysis and a little bit of analysis during design. The analysis phase defines the requirements of the system, independent of how these requirements will be accomplished. This phase defines the problem that the customer is trying to solve. The deliverable result at the end of this phase is a requirement document. Ideally, this document states in a clear and precise fashion what is to be built. This analysis represents the "what" phase.

The requirement document tries to capture the requirements from the customer's perspective by defining goals and interactions at a level removed from the implementation details. The requirement document may be expressed in a formal language based on mathematical logic. Traditionally, the requirement document is written in English or another written language. The requirement document does not specify the architectural or implementation details but specifies information at the higher level of description. The problem statement, the customer's expectations, and the criteria for success are examples of high-level descriptions.

There is a fuzzy line between high-level descriptions and low-level details. Sometimes, if an exact engineering detail needs to be specified, this detail will also appear in the requirement document. This is the exception and should not be the rule. These exceptions occur for many reasons including maintaining the consistency with other established systems, availability of particular options, customer's demands, and to establish, at the requirement level, a particular architecture vision.

An example of a low- level detail that might appear in the requirement document is the usage of particular vendor's product. particular vendor's product line, or the usage of some accepted computer industry standard, or a constraint on the image size of the application.

The requirement document states what the system should accomplish, independent of many of the details. The discovery process used in establishing the requirements during the analysis phase is best described as a refinement process than as a levels-of-detail process. Traditionally, the requirement document describes the things in the system and the actions that can be done on these things. Things might be expressed as objects in an object-based technology where data and algorithms are hidden behind hierarchical polymorphic methods. Alternatively, things might be expressed as services accessing databases in

the description of things in the system can be much more general and not confined to a particular technology. In a more general sense, this document describes the ontology that is the noun phrases and the verb phrases that will become the guidelines for defining the application specific protocol. Later, in the design phase, these requirement descriptions are mapped into computer science-based primitives, such as lists, stacks, trees, graphs, algorithms, and data structures. In the design phase the architecture is established. This phase starts with the requirement document delivered by the requirement phase and maps the requirements into architecture. The architecture defines the components, their interface and behaviors. The deliverable design document is the architecture. The design document describes a plan to implement the requirements. This phase represents the “how" phase. Details on computer programming languages and environments, machines, packages, application architecture, distributed architecture layering, memory size, platform, algorithms, data structures, global type definitions interfaces, and many other engineering details are established. The design may include the usage of existing components. The architectural team can now expand upon the information established in the requirement document. Using the typical and a typical scenario provided from the requirement document, performance tradeoffs can be accomplished as well as complexity of implementation trade-offs. Obviously, if an action is done, it needs to be done correctly and efficiently. A seldom used action needs to be implemented correctly, but it is not obvious what level of performance is required. The requirement document must guide this decision process. An example of a seldom used action which must be done with high performance is the emergency shutdown of a nuclear reactor. Analysing the trade-offs of necessary complexity allows for many things to remain simple which, in turn, will eventually lead to a higher quality product. The architecture team also converts the typical scenarios into a test plan. In our approach, the team, given a complete requirement document, must also indicate critical priorities for the implementation team. A critical implementation priority leads to a task that has to be done right. If it fails, the product fails. If it succeeds, the product might succeed. At the very least, the confidence level of the team producing a successful product will increase. This will keep the implementation team focused. Exactly how this information is conveyed is a skill based on experience more than a science based on fundamental foundations. In the implementation phase, the team builds the components either from scratch or by composition. Given the architecture do

flexibility. For example, a component may be narrowly designed for this particular system, or the component may be made more general to satisfy a reusability guideline.

2.3 Full Stack Development

Angular

Angular is a platform and framework for building single-page client applications using HTML and TypeScript. Angular is written in TypeScript. It implements core and optional functionality as a set of TypeScript libraries that you import into your applications.

The architecture of an Angular application relies on certain fundamental concepts. The basic building blocks of the Angular framework are Angular components that are organized into NgModules. NgModules collect related code into functional sets; an Angular application is defined by a set of NgModules. An application always has at least a root module that enables bootstrapping, and typically has many more feature modules.

Components define views, which are sets of screen elements that Angular can choose among and modify according to your program logic and data. Components use services, which provide specific functionality not directly related to views. Service providers can be injected into components as dependencies, making your code modular, reusable, and efficient.

Asp.net

ASP.NET is open-source, server-side web-application development to produce dynamic web pages. It was developed by Microsoft to allow programmers to build dynamic web sites, applications and services. The name stands for Active Server Pages Network Enabled Technologies.

It was first released in January 2002 with version 1.0 of the .NET Framework and is the successor to Microsoft's Active Server Pages (ASP) technology. ASP.NET is built on the Common Language Runtime (CLR), allowing programmers to write ASP.NET code using any supported .NET language. The ASP.NET SOAP extension framework allows ASP.NET compon

ASP.NET's successor is ASP.NET Core. It is a re-implementation of ASP.NET as a modular web framework, together with other frameworks like Entity Framework. The new framework uses the new open-source .NET Compiler Platform (codename "Roslyn") and is cross platform. ASP.NET MVC, ASP.NET Web API, and ASP.NET Web Pages (a platform using only Razor pages) have merged into a unified MVC 6.

Microsoft Sql Server

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications which may run either on the same computer or on another computer across a network (including the Internet). Microsoft markets at least a dozen different editions of Microsoft SQL Server, aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users.

Postman API

Postman is an API platform for building and using APIs. Postman simplifies each step of the API lifecycle and streamlines collaboration so you can create better APIs-faster.

CHAPTER 3

TASK PERFORMED

3.1 Introduction

The proposed project “Employee Salary Management System” has been developed to overcome the problems faced in the practicing of manual system. This software is built to eliminate and, in some cases, reduce the hardships faced by the existing system. Moreover, this system is designed for need of the company to carry out its operations in a smooth and effective manner.

This web application is reduced as much as possible to avoid errors while entering data. It also provides error message while entering invalid data. It is user-friendly as no formal knowledge is required to use the system.

Human resource challenges are faced by every organization which has to be overcome by the organization. Every organization has different employee and payroll management needs. Therefore, I have design exclusive Employee and payroll Management System that are adapted to the organization’s Managerial Requirements

The purpose of this document is to describe the functionality and specifications of the design of a web application for Managing Employees and their payroll. The expected audiences of this document are the developers and the admin of the web application. Now with the help of this system the admin has the information on his fingertips and can easily prepare a good record based on their requirements.

Finally, we can say that this system will not only automate the process but save the valuable time of the manager or the admin, which can be well utilized buy his institute. This will be an additional advantage and management of power based on their free time from his normal duty.

List of Tasks

- Study angular.js and C#
- Learning about company Architecture
- Knowledge Transfer sessions about the code structure
- Implemented project
- Rectified the project according to the suggestions and completed the work

GANTT CHART FOR INTERNSHIP SCHEDULE																
ACTIVITIES	FEBRUARY				MARCH				APRIL				MAY			
	WEEKS															
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Study javascript																
Study angular and c#																
KT Session																
implementation																
finalization																

Weekly Task performed

Week 1 (Learn javascript)

Study javascript and learn web development technologies like html, css, bootstrap.

JavaScript, often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. As of 2022, 98% of websites use JavaScript on the client side for web page behavior, often incorporating third-party libraries.

Week 2-7 (Study angular and asp.net)

Study angular js and c#. Angular and c# both are required to make a fullstack project with backend and frontend application. Learning in ongoing along with the implementation side by side. Our purpose is to learn from implementation.

3.1 Angular

Angular is a platform and framework for building single-page client applications using HTML and TypeScript. Angular is written in TypeScript. It implements core and optional functionality as a set of TypeScript libraries that you import into your applications.

The architecture of an Angular application relies on certain fundamental concepts. The basic building blocks of the Angular framework are Angular components that are organized into NgModules. NgModules collect related code into functional sets; an Angular application is defined by a set of NgModules. An application always has at least a root module that enables bootstrapping, and typically has many more feature modules.

Components define views, which are sets of screen elements that Angular can choose among and modify according to your program logic and data. Components use services, which provide specific functionality not directly related to views. Service providers can be injected into components as dependencies, making your code modular, reusable, and efficient.

Asp.net

ASP.NET is open-source, server-side web-application development to produce dynamic web pages. It was developed by Microsoft to allow programmers to build dynamic web sites, applications and services. The name stands for Active Server Pages Network Enabled Technologies.

It was first released in January 2002 with version 1.0 of the .NET Framework and is the successor to Microsoft's Active Server Pages (ASP) technology. ASP.NET is built on the Common Language Runtime (CLR), allowing programmers to write ASP.NET code using any supported .NET language.

The ASP.NET SOAP extension framework allows ASP.NET components to process SOAP messages.

Week 7-10 (Implementation)

Implemented the project with all the requirements.

The purpose of this design phase is to develop a clear understanding of what developer wants people to gain from his/her project. As the developer works on the project, the test for every design decision should be “Does this Design fulfil the ultimate purpose of the project”.

A purpose statement affects the design process by explaining what the developer wants the project to do, rather than describing the project itself. The Design document will verify that the current design meets all the explicit requirements contained in the system model as well as implicit requirement desired by the customer.

The project has backend database as sql server to store the information. Api’s are created to connect backend with the frontend and to communicate.

Week 10-12 (Finalization)

Finalized the project along with all the rectification that are required by the manager and the project mentor. Learning outcomes are discussed along with the KT sessions are provided.

Skills are tested along with the project and other technologies that could be used are discussed and projected is tested with its requirement, different ways that can be implemented are discussed. How company architecture is related to all these things are discussed.

3.2 Technology Used

Angular

Angular is a platform and framework for building single-page client applications using HTML and TypeScript. Angular is written in TypeScript. It implements core and optional functionality as a set of TypeScript libraries that you import into your applications.

The architecture of an Angular application relies on certain fundamental concepts. The basic building blocks of the Angular framework are Angular components that are organized into NgModules. NgModules collect related code into functional sets; an Angular application is defined by a set of NgModules. An application always has at least a root module that enables bootstrapping, and typically has many more feature modules.

Components define views, which are sets of screen elements that Angular can choose among and modify according to your program logic and data. Components use services, which provide specific functionality not directly related to views. Service providers can be injected into components as dependencies, making your code modular, reusable, and efficient.

Asp.net

ASP.NET is open-source, server-side web-application development to produce dynamic web pages. It was developed by Microsoft to allow programmers to build dynamic web sites, applications and services. The name stands for Active Server Pages Network Enabled Technologies.

It was first released in January 2002 with version 1.0 of the .NET Framework and is the successor to Microsoft's Active Server Pages (ASP) technology. ASP.NET is built on the Common Language Runtime (CLR), allowing programmers to write ASP.NET code using any supported .NET language. The ASP.NET SOAP extension framework allows ASP.NET components to process SOAP messages.

ASP.NET's successor is ASP.NET Core. It is a re-implementation of ASP.NET as a modular web framework, together with other frameworks like Entity Framework. The new framework uses the new open-source .NET Compiler Platform (codename "Roslyn") and is cross platform. ASP.NET

MVC, ASP.NET Web API, and ASP.NET Web Pages (a platform using only Razor pages) have merged into a unified MVC 6.

Microsoft Sql Server

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications which may run either on the same computer or on another computer across a network (including the Internet).

Microsoft markets at least a dozen different editions of Microsoft SQL Server, aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users.

Postman API

Postman is an API platform for building and using APIs. Postman simplifies each step of the API lifecycle and streamlines collaboration so you can create better APIs—faster.

At Postman, we create with the same curiosity that we see in our users. We value transparency and honest communication about not only successes, but mistakes. In our work, we focus on specific goals that add up to a larger vision. Our inclusive work culture ensures that everyone is valued equally as important pieces of our final product. We are dedicated to delivering the best products.

3.3 Requirement Specification

Available Technologies:

- Asp.net with C#
- Microsoft Sql Server
- Angular 13
- Postman API

Hardware used:

- Processor: Intel i5 core vPro
- Ram : 16GB
- SSD: 512GB
- Operating System: Windows 10

Software Used:

- Editor: Microsoft Visual Studio
- Database: Microsoft SQL Server
- API: Postman API
- FrontEnd: Angular 13

3.4 Design Methodology

The purpose of this design phase is to develop a clear understanding of what developer wants people to gain from his/her project. As the developer works on the project, the test for every design decision should be “Does this Design fulfil the ultimate purpose of the project”.

A purpose statement affects the design process by explaining what the developer wants the project to do, rather than describing the project itself. The Design document will verify that the current design meets all the explicit requirements contained in the system model as well as implicit requirement desired by the customer.

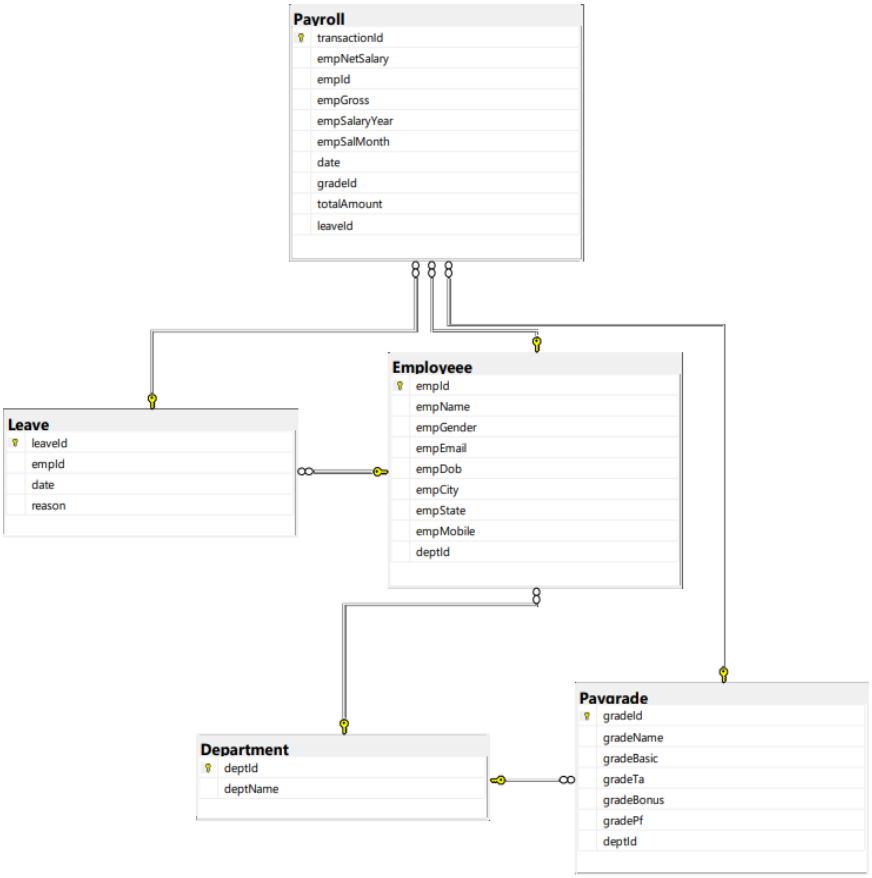


Figure 3.4.1 ER Diagram

3.5 Implementation

3.5.1 Creating Models

As a first part initialize the column name and their respective data types in separate models. In my project the respective models are,

1. Employee Model

2. Department model

3. Payroll Model

4. Paygrade Model

5. Leave Model

3.5.2 Generating the Controllers

After creating model for each model generate a controller using scaffolding method. This controller includes GET, POST, PUT and DELETE methods. In my project the respective controllers are,

1. Department Controller

2. Employee Controller

3. Payroll Controller

4. Paygrade Controller

5. Leave Controller

3.5.3 Add Migration and Update Database

Using NuGet package console the following commands are used to migrate and update database.

1.Add-Migration 1 (1 indicates the number of Migrations)

2.Update-Database

3.5.4 Microsoft MySQL Database

After updating database command in NuGet console all the tables are automatically added into the Database. We can check by connecting the database into our server using connection string code in the backend with Windows Authentication.

The respective tables that are automatically generated by the commands in project are,

3.5.4.1 Department Table

Department table has two columns, deptId and departmentName here deptId is a primary key.

deptId	Integer (32), Primary key
departmentName	Varchar (50)

3.5.4.2 Employee Table

Employee Table has 11 columns that has listed in below table, here empId is PrimaryKey,

deptId and gradeId is Foreign Key.

empId	Integer, Primary Key
empName	Varchar(max)
empGender	Integer
empDob	Datetime2
empcity	Varchar(max)
empState	Varchar(max)
empMobile	Varchar(max)
empEmail	Varchar(max)
deptId	Integer, Foreign key(department)
gradeId	Integer, Foreign key(paygrade)
joining Date	Integer

3.5.4.3 Paygrade Table

Paygrade table has 7 columns in that gradeId is a Primary Key and DeptId is a Foreign Key.

gradeId	Integer, Primary Key
gradeName	Varchar(max)
gradeBasic	Decimal
gradeTa	Decimal
gradeBonus	Decimal
gradePf	Decimal
deptId	Integer, Foreign Key (Department)

3.5.4.4 Payroll Table

Payroll table has 9 columns in that transactionId is a Primary Key, empId and gradeId is Foreign Key.

transactionId	Integer, Primary key
empNetSalary	Decimal
empId	Integer, Foreign key(Employee)
empGross	Decimal
empSalaryYear	Decimal
empSalaryMonth	Decimal
date	DateTime2
totalAmount	Decimal
gradeId	Integer, Foreign key(Paygrade)

3.5.4.5 Leave Table

Leave table has total 6 columns in that leaveId is a Primary Key and empId is a Foreign Key.

leaveId	Integer, Primary key
empId	Integer, Foreign key (Employee)

fromDate	DateTime2
toDate	DateTime2
reason	Varchar(max)
totalDays	Integer

3.5.4.6 Login Table

Login table consist of 2 columns that is listed below, in that Email is a Primary Key.

Email	Primary Key, Varchar (50)
Password	Varchar(max)

3.5.5 Postman API

This API is used to send the body in a JSON format through Web API and store it in a Database. It is something like testing the API through hard code part.

3.5.6 Angular

To make the project user friendly implementing a frontend part using angular in this project. Here we creating a forms for all the table to perform an operation such as GET, POST, PUT and DELETE.

So here Angular and Visual Studio hosting a different localhost. So here we are using a CROSS tool which connect these localhosts and unblocking the server and it makes cross platform very easy.

3.6 Results and Discussion (Include Snapshot)

This chapter (3.6) consist of working snapshots of the Employee Salary Management.

3.6.1 Front Page

Once the project is open Front page will appear it consist of “Add Leave Form” and “Admin” Login Button as shown in the Figure 3.6.1

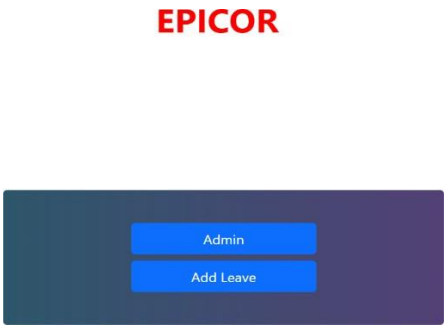


Figure 3.6.1 Front Page

3.6.2 Add Leave Button

This Add Leave is global for all the employees, any employee can apply leave through this Add Leave form as shown in the Figure 3.6.2

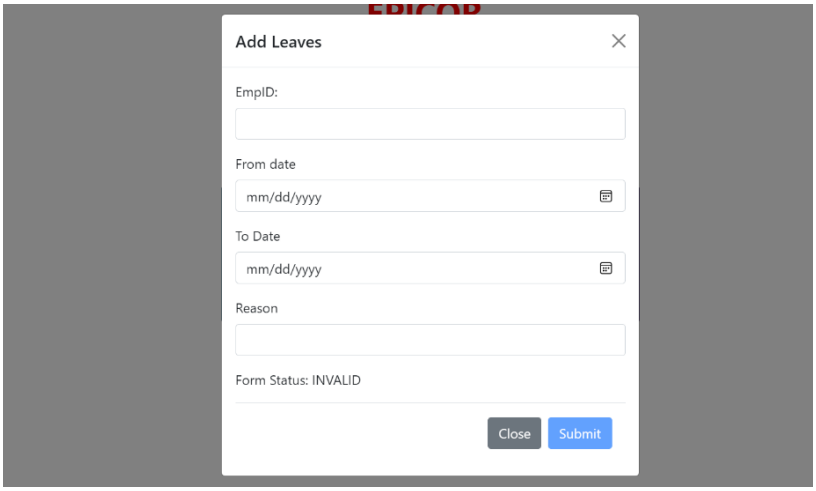


Figure 3.6.2 Add Leave Form

3.6.3 Login Page

Admin can login using their credentials properly to access all the details of Database. If admin credentials were wrong this shows the alert message and cannot access the database. These two Figures 3.6.3.1 Login Success and Figure 3.6.3.2 Login Unsuccessful.

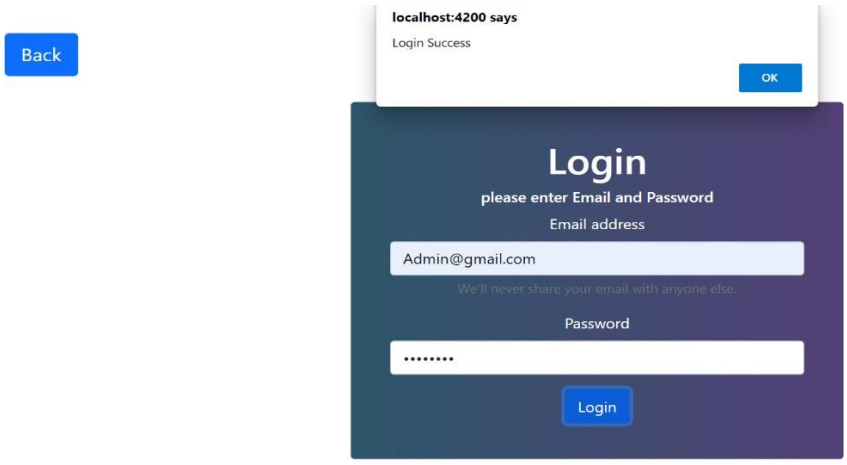


Figure 3.6.3.1 Login Successful

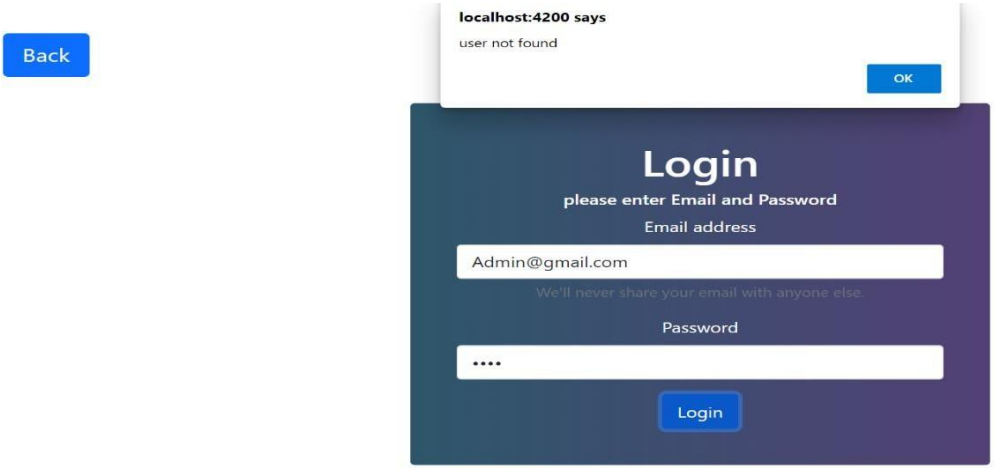


Figure 3.6.3.2 Login Unsuccessful

3.6.4 Department Table

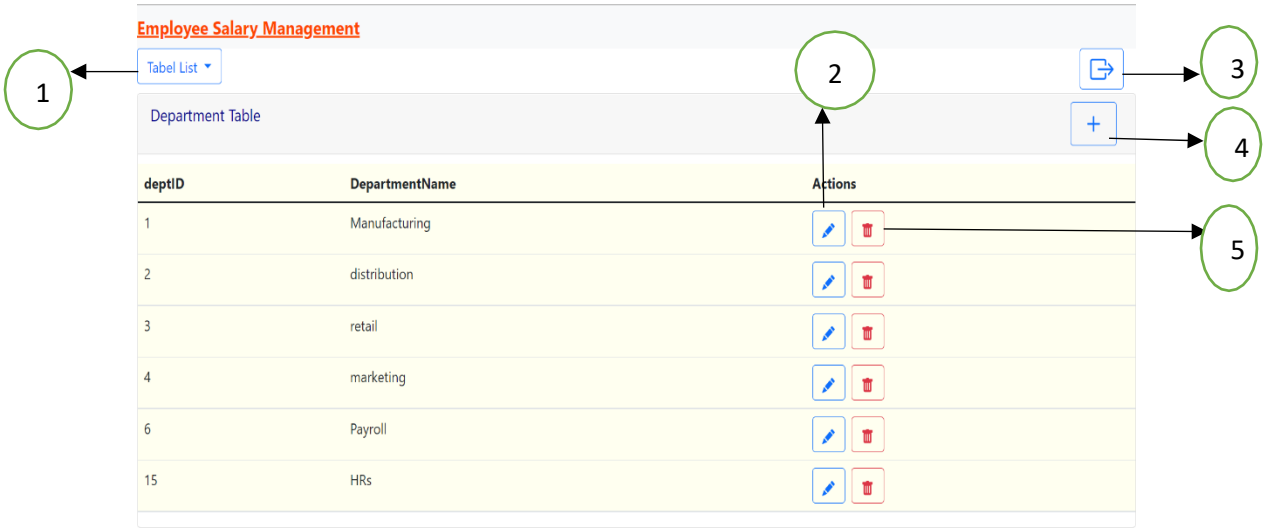


Figure 3.6.4 Department Table

1. **Table List:** It is dropdown button here we can select the table list and it will redirect to that selected table.
2. **Edit button:** By clicking this edit form will appear and edit the details
3. **Logout button:** By clicking this we can exit from the page it will redirect to frontpage.
4. **Add button:** By clicking it Add Form will appear, so using this admin can add new department.
5. **Delete button:** By clicking this admin can delete the data.

The above 5 points are similar in all the tables. The snapshots of Edit, Delete and Add as shown below in Figure 3.6.4.[1|2].

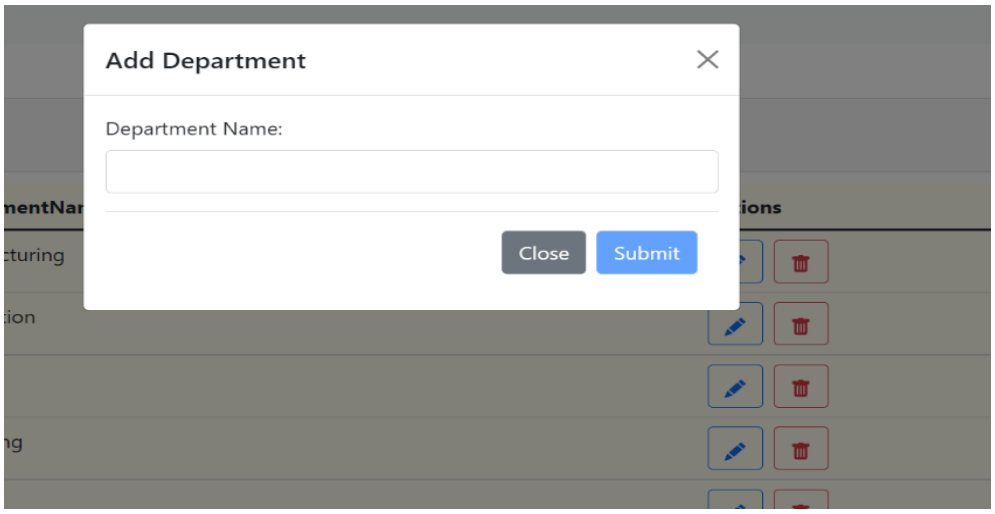


Figure 3.6.4.1 Add Department Form

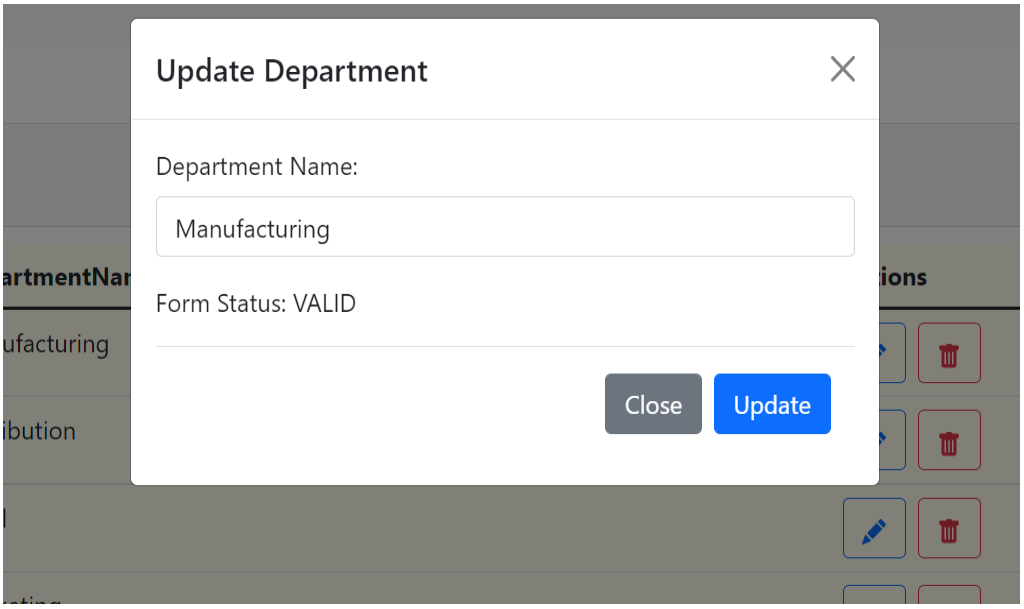
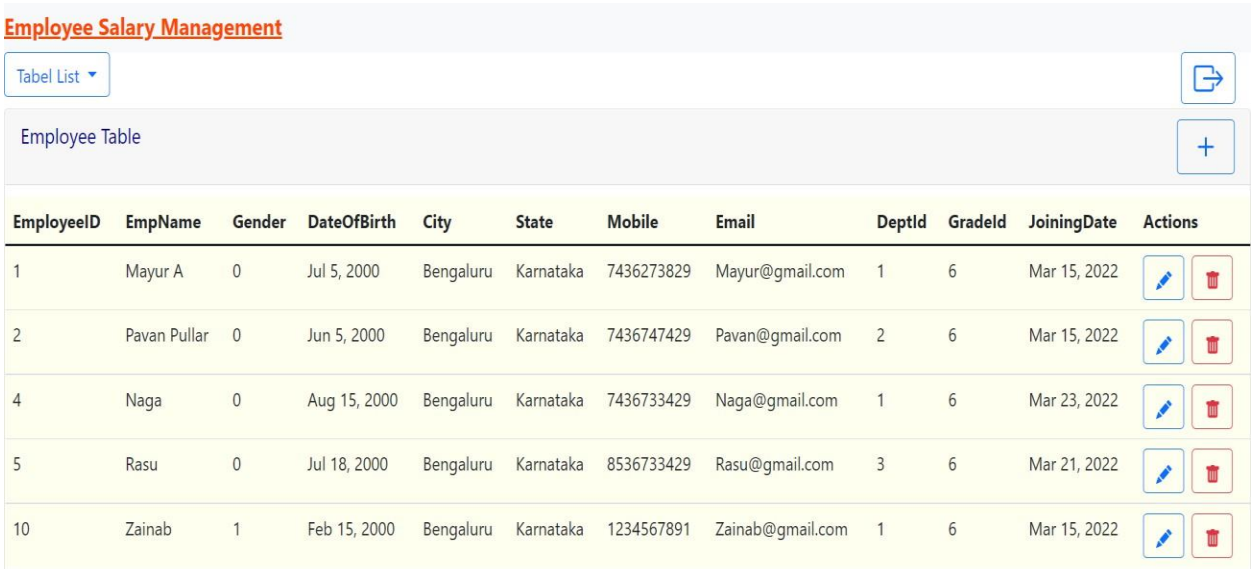


Figure 3.6.4.2 Edit Department Form

3.6.5 Employee Table

Here admin can access Add, Edit and Delete options in employee database.

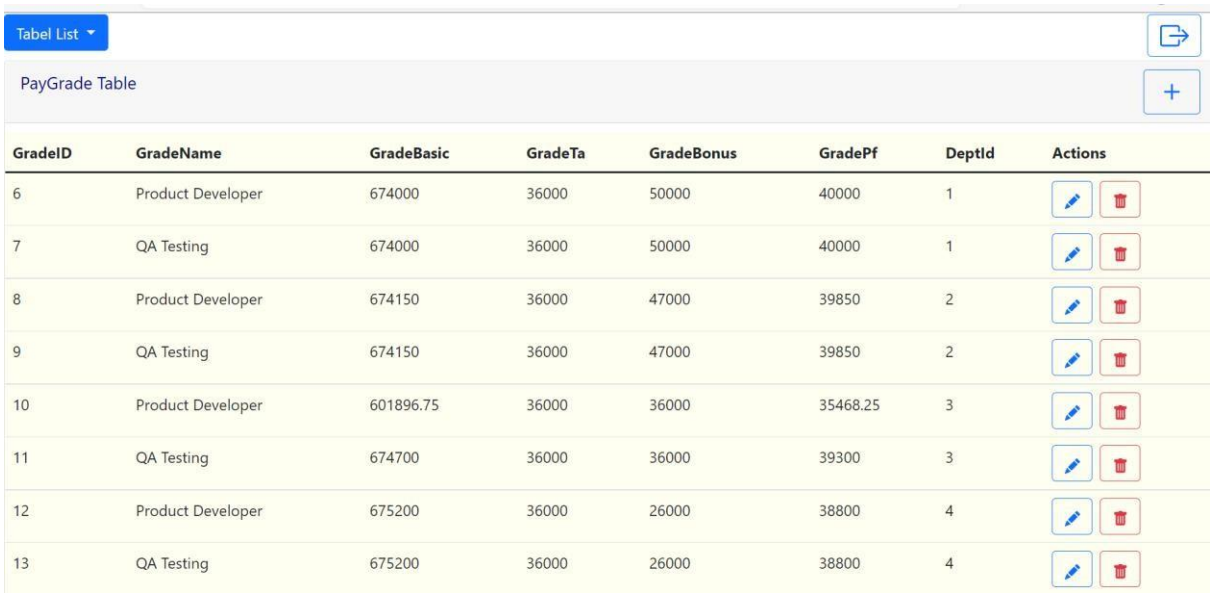


EmployeeID	EmpName	Gender	DateOfBirth	City	State	Mobile	Email	DeptId	GradeId	JoiningDate	Actions
1	Mayur A	0	Jul 5, 2000	Bengaluru	Karnataka	7436273829	Mayur@gmail.com	1	6	Mar 15, 2022	
2	Pavan Pullar	0	Jun 5, 2000	Bengaluru	Karnataka	7436747429	Pavan@gmail.com	2	6	Mar 15, 2022	
4	Naga	0	Aug 15, 2000	Bengaluru	Karnataka	7436733429	Naga@gmail.com	1	6	Mar 23, 2022	
5	Rasu	0	Jul 18, 2000	Bengaluru	Karnataka	8536733429	Rasu@gmail.com	3	6	Mar 21, 2022	
10	Zainab	1	Feb 15, 2000	Bengaluru	Karnataka	1234567891	Zainab@gmail.com	1	6	Mar 15, 2022	

Figure 3.6.5 Employee Table

3.6.6 Paygrade Table

Here admin can Add, Edit and Delete the Paygrade Details. As shown in the below Figure.

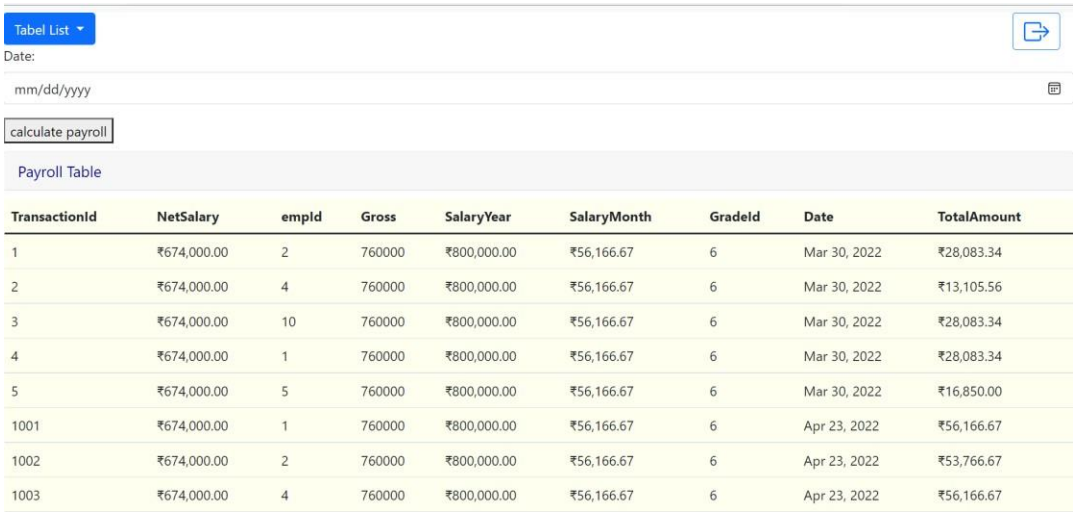


GradeID	GradeName	GradeBasic	GradeTa	GradeBonus	GradePf	DeptId	Actions
6	Product Developer	674000	36000	50000	40000	1	
7	QA Testing	674000	36000	50000	40000	1	
8	Product Developer	674150	36000	47000	39850	2	
9	QA Testing	674150	36000	47000	39850	2	
10	Product Developer	601896.75	36000	36000	35468.25	3	
11	QA Testing	674700	36000	36000	39300	3	
12	Product Developer	675200	36000	26000	38800	4	
13	QA Testing	675200	36000	26000	38800	4	

Figure 3.6.6 Paygrade Table

3.6.7 Payroll Table

Here admin can select their payroll date and by clicking calculate payroll button, it will calculate payrolls for all the employee by considering their no.of leaves and working days and it show in the table format.

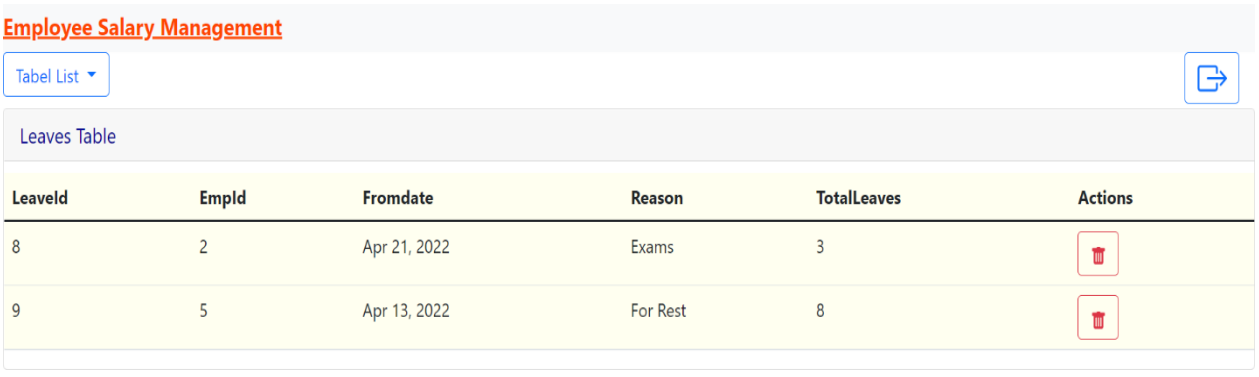


TransactionId	NetSalary	empId	Gross	SalaryYear	SalaryMonth	GradeId	Date	TotalAmount
1	₹674,000.00	2	760000	₹800,000.00	₹56,166.67	6	Mar 30, 2022	₹28,083.34
2	₹674,000.00	4	760000	₹800,000.00	₹56,166.67	6	Mar 30, 2022	₹13,105.56
3	₹674,000.00	10	760000	₹800,000.00	₹56,166.67	6	Mar 30, 2022	₹28,083.34
4	₹674,000.00	1	760000	₹800,000.00	₹56,166.67	6	Mar 30, 2022	₹28,083.34
5	₹674,000.00	5	760000	₹800,000.00	₹56,166.67	6	Mar 30, 2022	₹16,850.00
1001	₹674,000.00	1	760000	₹800,000.00	₹56,166.67	6	Apr 23, 2022	₹56,166.67
1002	₹674,000.00	2	760000	₹800,000.00	₹56,166.67	6	Apr 23, 2022	₹53,766.67
1003	₹674,000.00	4	760000	₹800,000.00	₹56,166.67	6	Apr 23, 2022	₹56,166.67

Figure 3.6.7 Payroll Table

3.6.8 Leave Table

Here admin can only access delete option does not edit one. As shown in the below figure.



LeaveId	EmpId	Fromdate	Reason	TotalLeaves	Actions
8	2	Apr 21, 2022	Exams	3	
9	5	Apr 13, 2022	For Rest	8	

Figure 3.6.8 Leave Table

CHAPTER 4

REFLECTION NOTES

4.1 EXPERIENCE

As a part of academics under VTU, B.E, 8th semester students need to do their internship program for a duration of 2 months in a company or an organization to get exposure to concepts in real time applications. Recently I got to know about Epicor Software through LinkedIn, and they are providing internship on new technologies such as C#, ASP.net, Angular and Azure.

The chief coordinator and the trainers in Epicor Software introduced us to the new concept in Full stack Development. As a part of this training program, I was introduced to the C#, Asp.net, Angular and Azure. After the introduction of concepts, I was given assessments to solve. I was made to work on different projects that are related to new technologies. The trainers were very supportive when I faced problems. Hands on training on each module helped us understand the concepts even better. The prior knowledge of programming helped me to understand Full Stack development better.

We first worked on few basic concepts in C#, Angular and learnt their features and uses. Different methods of integrating different parts of the software were taught by the trainers. While working hands on, I enjoyed the casual debates and discussions with the trainers and fellow interns which brought more joy to learning.

After completing the training, I along with my partner undertook a mini project on Full stack development with the guidance of our trainer and I am happy to say that I was successfully completed that mini project.

This internship was a new experience along my journey to achieve technical knowledge and contribute in a significant way to the development of the society. It helped me broaden my thinking in all horizons of Engineering. It was beneficial for me, and I take this opportunity to be grateful and thankful to all the people who have helped me along this amazing journey.

4.2 ASSESSMENT

The assessment of an internship is very important for any internship program. Assessment is nothing but weighing the pros and cons of the experience.

I would like to assess my internship as follows:

- Guides were helpful, approachable, provided feedback on my progress and skills.
- Level of responsibilities was consistent with my abilities.
- Fellow workers accepted my work and treated me appropriately.
- This internship provided me the opportunity to use my academic preparation.
- This provided me the opportunity to develop my interpersonal and technical skills.
- This was a valuable learning experience, and I grew as a professional.
- It helped me define my career goals and next step in my career.
- This provides a way to identify areas for personal and professional growth.
- I had the opportunity to use the resources that are not available on campus.
- This internship provided an opportunity for self-directed learning.
- This internship may have a good impact on my academic and career plans.

Teaching approaches may vary significantly from academic program to another. But one approach that seems to be universally valued across disciplines is that of the internship. Internship experiences help us connect learning to life, as they provide with the opportunity to apply the knowledge and skills they have learnt in the classroom to real-life situations.

At the beginning I did not have any experience of working at industry. I understood the functioning better, like the organization structure and setting up projects. In the beginning the dependence and uncertainty was annoying, but it forced me to be flexible and see what other things I could do.

4.3 LIST OF TECHNICAL OUTCOMES

The internship has provided the total reflection of the theoretical knowledge in the practical form. It has developed the awareness, understanding and capacity in the specific domain of technology called Industrial automation and well it has refined the skills effectively to endeavor professionally through technical achievement.

The technical outcomes are as briefed below:

- The ability to apply the knowledge of engineering.
- The ability to design and conduct experiments, as well as to analyze and interpret data.
- The ability to identify, formulate and solve industrial problems.
- The understanding of professional and ethical responsibility.
- Recognition towards the need for, and an ability to engage in, life-long learning.
- Knowledge of contemporary issues.
- The ability to use the techniques, skills, and modern tools necessary for industries.
- The ability to work in one or more significant application domains.
- The ability to identify, formulate and solve engineering problems.

4.4 NON-TECHNICAL SKILLS

As we all are very much familiar that the non-technical skills such as the verbal and written communication, personality development, time management and resource utilization are very important for any employee. Here I would like to express regarding my learning goals, how much I learned and what else should I have to improve.

At the beginning I did not have any experience of working within any industry, but later I understood better the functioning like organization structure and setting up projects. Trying to operate as a non-profit organization, I saw the importance of financial support and personal

capacity. I got to experience the usage of industrial standard techniques of coding. I learned actual process of designing any project. Before my internship started my ideas did not match the experiences have gained during my internship. This internship was an introduction to the actual work field for me. I have learned to work in a related organization and apply my knowledge into practice.

Relevant materials and guidance by the lecturers helped in shaping the report. Teachers gave us an outline to create report and alarmed us to do it in timely basis which immensely helped me. It was a fabulous experience with Silfra technologies. I gained lot of things both technically and non-technically. Throughout this journey reports were made to the internal guide in the college who immensely supported for shaping this report.

CHAPTER – 5

CONCLUSION

This project is built keeping in mind that it is to be used by only two user ,one is the admin and another is employees. It is built for use in medium scale organization where the number of employees is somewhat limited. According to the requested requirement the admin can add, manipulate, update and delete all employee data in his organization. The admin can add new departments and delete them. The admin can also add predefined pay grades for the employees. The required records can be easily viewed by the admin anytime time he wants in an instant. The payment of the employee is based on monthly basis. Numerous validations implemented would enable the admin to enter accurate data. The main objective of this framework is to save time, make the system cost effective and management records efficiently.

In the post-processing stage, I began by resolving minor problems. I did it to familiarise myself with the process' flow, the necessary architecture and design, and to investigate unfamiliar technologies like PHP, MySQL etc. As I was going through the programme, I picked up a lot of new ideas on Full-Stack development and its advantages. The programme exposed me to real-world issues and gave me additional hands-on experience, both of which helped me expand or improve my practical knowledge.

Some of the key aspects of the organization as well the internship are:

- Result-driven and goal-focused approach to work.
- Enforcing discipline of performance management.
- Supportive employees with a wide range of training and further education opportunities.
- Similar treatment and similar benefits for all the employees as well as for interns.

During the internship, the benefit I have achieved is to test out a job, employment setting, management style and other aspects of the workplace.

REFERENCES

[1]Description of the company at url https://www.epicor.com/en-in/ and last visited at 01/07/2022

[2]Description of the course at url https://angularjs.org/ and last visited at 01/07/2022

[3]Description of the course at url https://ionicframework.com/ and last visited at 01/07/2022

[4]Description of the website at url https://javascript.info/ and last visited at 01/07/2022

[5]Description of the website at url https://www.codecademy.com/catalog/language/html-css and last visited at 01/07/2022

[6]Description of the website at url https://www.geeksforgeeks.org/angular-8-introduction/ and last visited at 01/07/2022

[7]Description of the website at url https://www.geeksforgeeks.org/javascript/ and last visited at 01/07/2022

[8]Description of the website at url https://www.geeksforgeeks.org/html/ and last visited at 01/07/2022

[9]Description of the website at url https://github.com/angular/angular and last visited at 01/07/2022

[10]Description of the website at url https://www.udemy.com/course/the-complete-guide-to-angular-2/ and last visited at 01/07/2022

[11]Description of the website at url https://www.w3schools.com/css/ and last visited at 01/07/2022

[12]Description of the website at url https://www.epicor.com/en-in/ and last visited at 01/07/2022

[13]Description of the website at url https://www.geeksforgeeks.org/how-to-become-a-front-end-developer and last visited at 01/07/2022

[14]Description of the website at url https://www.geeksforgeeks.org/ultimate-guide-git-github/ and last visited at 01/07/2022

[15]Description of the website at url https://www.javatpoint.com/angular-8 and last visited at 01/07/2022