

A
Summer Internship Report
On
" INTERN MANAGEMENT PROJECT"

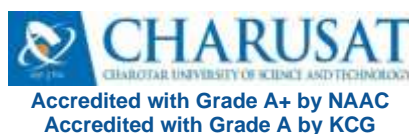
(IT346 – Summer Internship - I)

Prepared by
Dev Nandurbarkar (21IT096)

Under the Supervision of
Assistant Professor
Mikin Patel

Submitted to
Charotar University of Science & Technology (CHARUSAT)
for the Partial Fulfillment of the Requirements for the
Degree of Bachelor of Technology (B.Tech.)
for Semester 5

Submitted at



**SMT. KUNDANBEN DINSHA PATEL DEPARTMENT OF
INFORMATION TECHNOLOGY**

Chandubhai S. Patel Institute of Technology (CSPIT)
Faculty of Technology & Engineering (FTE), CHARUSAT
At: Changa, Dist: Anand, Pin: 388421.
August, 2023



Accredited with Grade A by NAAC
Accredited with Grade A by KCG

CERTIFICATE

This is to certify that the report entitled “**Intern Management Project**” is a bonafied work carried out by **Dev Nandurbarkar (21IT096)** under the guidance and supervision of **Assistant Professor Mikin Patel & Mr.Prerak Patel** for the subject **Summer Internship – I (IT346)** of 5th Semester of Bachelor of Technology in **Department of Information** at Chandubhai S. Patel Institute of Technology (CSPIT), Faculty of Technology & Engineering (FTE) – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himsel, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred by the examiner(s).

Under the supervision of,

Mikin Patel
Assistant Professor
Smt. Kundanben Dinsha Patel Department of
Information Technology
CSPIT, FTE, CHARUSAT, Changa, Gujarat

Mr.Prerak Patel
Head of ITE
Linde Engineering India

Dr. Parth Shah
Head of Department (IT)
CHARUSAT, Changa, Gujarat.

Chandubhai S. Patel Institute of Technology (CSPIT)
Faculty of Technology & Engineering (FTE), CHARUSAT

At: Changa, Ta. Petlad, Dist. Anand, Pin: 388421. Gujarat.



Linde Engineering India Pvt. Ltd. "Linde House", Near Nilamber Circle, Vasna, Vadodara.

26 June 2023

CERTIFICATE

This is to certify that Mr. Dev Nandurbarkar has undergone internship at Linde Engineering India Pvt. Ltd., Vadodara for a period of 26 days starting from 29/05/2023 to 23/06/2023.

During his internship, He was found to be hard working and sincere in his approach.

We wish him all the best for his future.

For Linde Engineering India Pvt. Ltd.

**Shashikant
Patange**

Digitally signed by Shashikant
Patange
DN: cn=Shashikant Patange, ou=
User, email=Shashikant.Patange
@linde.com
Date: 2023.06.26 10:06:28 +05'30'

Shashikant Patange
Director – Human Resources

Linde Engineering India Pvt. Ltd.
Linde House,
Near Nilamber Circle,
Vasna - Gotri Road, Vasna,
Vadodara- 391410, Gujarat, India
Phone +91 (265) 305 6789
Fax +91 (265) 298 1757
Website www.linde-india.com
Email Id: india@linde-le.com
Corporate Identity Number: U64203GJ1987PTC097873

ACKNOWLEDGEMENT

First I would like to thank Prerak Patel, Head of ITE, Linde Engineering India. And also Shashikant Patange, Director – Director Human Resource, Linde Engineering India. For giving me the opportunity to do an internship within the organization. Also, I would like to thank Dr. Parth Shah and Sanket Suthar Sir for helping me to give permission to intern at Linde Engineering India.

I perceive this opportunity as a big milestone in my career development. I will strive to use gained skills and knowledge in the best possible way, and I will continue to work on their improvement, in order to attain desired career objectives.

Finally, I would like to express my special thanks to my family for helping me in all aspects and appreciating me to spend my time in the work during my internship time.

Thanks,

Dev Nandurbarkar (21IT096)

Chandubhai S Patel Institute of Technology

ABSTRACT

I carried out my internship at Linde Engineering India, Linde is a multi-national company that provides IT solutions and automation requirements to help the company grow. This internship provided an opportunity to know how a project is executed in a MNC scale and the project expectations they have and the work culture and the work flow.

During my internship period, I had to work and lead other college students as my fellow interns and we are given tasks as a team. That helped me to gain experience in how to work in a team, and how to be a good team member.

My role was to take care of the front end and divide the whole project in sprints and make sure the project is meeting deadlines and user expectations as well as present the project whenever needed

In conclusion, this was an opportunity to develop and enhance skills and competencies in my career field which I actually achieved.

Table of Contents

Acknowledgement	ii
Abstract.....	iii
Description of company / organization.....	vi
Chapter 1 Introduction.....	1
1.1 Internship objectives.....	1
1.2 What is front end development, SDLC, and sprint.....	2
1.2.1 Front end development.....	2
1.2.2 SDLC (Software Development Life Cycle)	2
1.2.3 Sprints(in Agile Development).....	2
1.3 Responsibilities.....	3
1.3.1 Front end development.....	3
1.3.2 SDLC (Software Development Life Cycle).....	3
1.3.3 Sprints(in Agile Development).....	4
Chapter 2 Projects	5
2.1 Introduction to tools, technologies and languages used.....	5
2.1.1 Front-end – HTML5, CSS AND JS.....	5
2.1.2 Back-end & Database	5
2.1.3 API – Fetch API.....	6
2.1.4 Tools.....	6
2.2 Flow of Project.....	7
2.2.1 Project Initiation.....	7
2.2.2 Requirement Gathering and Analysis.....	8
2.2.3 Sprint 1.....	9
2.2.4 Sprint 2.....	10
2.2.5 Sprint 3.....	11
2.2.6 Sprint 4.....	11
Chapter 3 Sample Screenshot.....	13
3.1 Front End.....	13
3.2 Back End.....	18
Chapter 4 Conclusion	19

List of Figures

Fig 3.1 Home Page.....	13
Fig 3.2 Concerned Person Page.....	13
Fig 3.3 Page where a concerned person can enter dates.....	14
Fig 3.4 Page where a concerned person can enter availability.....	14
Fig 3.5 Page where a concerned person can update any changes on.....	15
Fig 3.6 Page where a concerned person can enter overlapping dates to modify.....	15
Fig 3.7 Calander view to see availability.....	16
Fig 3.8 Page where employee can give details.....	16
Fig 3.9 Concerned persons candidate approval or rejection page.....	17
Fig 3.10 Database Schema.....	18

DESCRIPTION OF COMPANY

Linde Engineering India is the Indian subsidiary of the multi-national company Linde Engineering, which is a global engineering company specializing in industrial gases, process plants, and related services. Linde Engineering India operates within the broader Linde Group, a leading global industrial gas and engineering company.

Linde Engineering India's primary focus is on providing engineering solutions for various industries, including petrochemicals, refining, natural gas, industrial gases, and other chemical processes. The company offers a wide range of services, including feasibility studies, engineering design, procurement, construction, and project management. They are known for their expertise in designing and building large-scale industrial plants, gas processing facilities, and liquefied natural gas (LNG) facilities.

Additionally, Linde Engineering India is actively involved in the development and implementation of sustainable technologies and solutions to reduce environmental impact. They aim to promote cleaner and more efficient processes in the industries they serve.

CHAPTER 1 INTRODUCTION

1.1 INTERNSHIP OBJECTIVES

- Internships are utilized in a number of different career fields, including architecture, engineering, healthcare, economics, advertising, and many more.
- Internships are generally thought to be reserved for college students looking to gain experience in a particular field. However, a wide array of people can benefit from Training Internships in order to receive real-world experience and develop their skills.
- An objective for this internship is to gain knowledge through real-life experience in various areas of user experience while keeping the educational background in view.
- In a large organization, the recruitment, and turnover of interns across different departments can pose a significant challenge. This problem statement aims to address the need for efficient management and oversight of the intern workforce, specifically focusing on tracking and maintaining the number of interns in each department.
- The primary issue lies in accurately monitoring the influx and outflow of interns within the organization. The project aims to help in having a clear understanding of the number of interns joining and leaving each department to ensure proper resource allocation, workload management, and a smooth transition process, as Linde is a large organization, which typically comprises multiple departments and functions distributed across different locations. With interns rotating through various departments and potentially having different start and end dates, it becomes crucial to establish a streamlined process for tracking their movement and maintaining a balanced intern distribution across departments.
- By effectively managing the number of interns in each department, the organization can ensure that workloads are appropriately distributed, projects are adequately supported, and interns receive a valuable learning experience. It also enables the HR or internship management team to plan and allocate resources effectively, such as mentors or supervisors, workspace, and training materials.
- The solution is to imply the need for a system or process that enables the organization to track the recruitment and turnover of interns across departments accurately.

1.2 WHAT IS FRONT-END DEVELOPMENT, SDLC, AND SPRINTS

1.2.1 Front-end Development:

Front-end development, also known as client-side development, refers to the process of creating the user interface (UI) and user experience (UX) of a website or web application. Front-end developers work on the visual and interactive elements that users directly interact with in their web browsers. This includes designing and implementing layouts, navigation menus, buttons, forms, and other graphical elements using HTML, CSS, and JavaScript. Front-end development focuses on making the website or application visually appealing, user-friendly, and responsive across different devices.

1.2.2 SDLC (Software Development Life Cycle):

The Software Development Life Cycle (SDLC) is a structured and systematic approach to software development. It outlines the various stages and activities involved in the creation, maintenance, and eventual retirement of software applications. The SDLC typically includes the following phases:

- a. Requirements Gathering: Gathering and analyzing the requirements of the software from stakeholders and end-users.
- b. Design: Creating a detailed design specification based on the gathered requirements.
- c. Implementation (Coding): Writing the actual code for the software based on the design specifications.
- d. Testing: Conducting various testing activities to identify and fix bugs and ensure the software meets the specified requirements.
- e. Deployment: Releasing the software for use by end-users.
- f. Maintenance: Ongoing support and updates to keep the software running smoothly and address issues that arise post-deployment.

The SDLC provides a systematic approach to software development, ensuring that the process is well-organized, efficient, and meets the needs of the stakeholders.

1.2.3 Sprints (in Agile Development):

Sprints are time-boxed, iterative periods in Agile software development during which a development team works on specific tasks or user stories. Agile is an iterative and incremental approach to software development, and sprints are a fundamental part of it. Each sprint typically lasts for two to four weeks, and at the beginning of a sprint, the development team selects a set of prioritized tasks from the product backlog to work on during that period. The goal is to complete these tasks by the end of the sprint and deliver a potentially shippable increment of the software.

During a sprint, the team holds daily stand-up meetings to discuss progress, challenges, and plans. At the end of the sprint, the team demonstrates the completed work to stakeholders, and feedback is collected. The development team also holds a sprint retrospective to discuss what went well, what could be improved, and how to make the next sprint more effective.

The use of sprints in Agile development allows for continuous delivery of software features, frequent stakeholder feedback, and adaptability to changing requirements, resulting in a more flexible and responsive development process.

1.3 RESPONSIBILITIES

1.3.1 Front-end Development:

Responsibilities of Front-end Developers include:

- Implementing the visual and interactive elements of a website or web application based on design specifications.
- Writing clean, maintainable, and efficient HTML, CSS, and JavaScript code.
- Ensuring the website or application is responsive and works seamlessly across different devices and screen sizes.
- Optimizing the performance of the front-end code for faster loading times.
- Collaborating with UI/UX designers and back-end developers to ensure a smooth user experience.
- Conducting cross-browser testing and fixing any compatibility issues.
- Staying updated with the latest front-end technologies and best practices.
- Work on the API and server-side code as well

1.3.2 SDLC (Software Development Life Cycle):

Responsibilities within the SDLC can vary depending on the role and phase of development. Key responsibilities include:

- Requirements Gathering: Engaging with stakeholders to elicit, analyze, and document software requirements.
- Design: Creating detailed technical specifications and architectural designs based on the requirements.
- Implementation (Coding): Writing, reviewing, and testing the code to build the software application.
- Testing: Conducting different types of testing, such as unit testing, integration testing, and user acceptance testing, to ensure the software's quality and identify defects.
- Deployment: Preparing the software for production deployment and managing the release process.
- Maintenance: Addressing and resolving issues reported by end-users and providing ongoing support to keep the software functioning effectively.
- Throughout the SDLC, collaboration, and communication among team members, stakeholders, and project managers are essential for successful software development.

1.3.3 Sprints (in Agile Development):

Responsibilities within the context of Sprints in Agile Development include:

- **Product Owner:** Defining and prioritizing the user stories in the product backlog based on business value and stakeholder needs.
- **Scrum Master:** Facilitating the sprint planning, daily stand-up, and sprint retrospective meetings. Ensuring that the Agile principles and practices are followed.
- **Development Team:** Select user stories from the product backlog to be worked on during the sprint. Collaborating to design, develop, test, and deliver the sprint increment.
- **During the Sprint:**
 - **Daily Stand-up:** Each team member shares their progress, challenges, and plans for the day.
 - **Sprint Review:** The development team presents the completed work to stakeholders and collects feedback.
 - **Sprint Retrospective:** The team reflects on the sprint and identifies areas for improvement in their process.
- **In summary,** Front-end Developers focus on creating the user interface and experience, SDLC defines the overall software development process, and Sprints are specific time-boxed iterations within the Agile framework where development tasks are executed and delivered. Collaboration and effective communication are key elements for success in all of these responsibilities.

CHAPTER 2 PROJECTS

2.1 INTRODUCTION TO TOOLS, TECHNOLOGIES AND LANGUAGES USED

2.1.1 FRONT-END – HTML 5, CSS AND JAVASCRIPT

HTML5, CSS, and JavaScript are foundational technologies for web development. HTML5 (HyperText Markup Language) is the latest version of the markup language used to structure and present content on the web, allowing developers to define the page's layout, headings, paragraphs, images, and links. CSS (Cascading Style Sheets) complements HTML by controlling the presentation and layout of the HTML elements, enabling developers to customize fonts, colors, spacing, and overall visual appearance. JavaScript, on the other hand, is a powerful scripting language that adds interactivity and dynamic behavior to web pages, facilitating actions like form validation, animations, and handling user interactions. Together, these three technologies form the core of modern web development, enabling developers to create visually appealing, responsive, and interactive websites and applications.

2.1.2 Back-End & Database- ASP.NET Core 6, Microsoft SQL, Node JS

ASP.NET Core 6, Microsoft SQL, and Node.js are prominent technologies used in modern web development. ASP.NET Core 6 is a cross-platform, open-source framework for building web applications, APIs, and microservices. It offers improved performance, modular architecture, and supports cloud-based deployment. Developers can use C# to build robust and scalable applications on various platforms.

Microsoft SQL (Structured Query Language) is a relational database management system widely used to store and manage structured data efficiently. It provides a reliable, secure, and scalable database solution, enabling developers to create complex data-driven applications and perform efficient data retrieval and manipulation operations.

Node.js is a server-side runtime environment built on Chrome's V8 JavaScript engine. It allows developers to use JavaScript on the server-side, providing a non-blocking, event-driven architecture that handles concurrent connections effectively. Node.js is popular for building scalable and high-performance applications, especially real-time applications and APIs.

Together, these technologies offer a powerful stack for building web applications. ASP.NET Core 6 provides the server-side framework, Microsoft SQL handles data storage and management, and Node.js allows for scalable server-side JavaScript execution, making it possible to create modern and feature-rich web applications that meet the demands of today's dynamic and data-intensive web development landscape.

2.1.3 API – FETCH API

The Fetch API is a built-in JavaScript interface that allows developers to make asynchronous HTTP requests to servers and fetch resources, such as data or files, from web services or APIs. It provides a modern alternative to traditional XMLHttpRequest, simplifying the process of handling network requests and responses. The Fetch API uses Promises, enabling a more concise and readable code structure. Developers can customize requests by specifying request headers, methods, and other options. When a response is received, it can be parsed and processed using different methods like `response.json()`, `response.text()`, or `response.blob()`. The Fetch API's native browser support and simplicity have made it a preferred choice for fetching data and interacting with web services in modern web development, powering various applications, and enabling dynamic content on webpages.

2.1.4 Tool(s)- Visual Studio Integrated Development Environment

Visual Studio is a comprehensive Integrated Development Environment (IDE) created by Microsoft for software development across multiple platforms. It offers a wide range of tools and features to support various programming languages such as C#, C++, JavaScript, Python, and more. With its intuitive interface, developers can efficiently write, debug, and test code, as well as collaborate with team members using built-in version control integration. Visual Studio provides powerful code editors, intelligent code completion, and real-time debugging capabilities, making it easier for developers to write high-quality code. It also offers extensive frameworks, libraries, and extensions to enhance productivity and cater to specific development needs, making it a favored choice for professional software development projects.

2.2 FLOW OF THE PROJECT

2.2.1 Project Initiation (Days 1-2):

29/05 – 30/05

Project Scope, Objectives, and Deliverables:

Scope:

The project aims to develop and implement an efficient intern management system within a large organization to track the recruitment, onboarding, and turnover of interns across different departments. The system will enable accurate monitoring of the number of interns joining and leaving each department and facilitate effective resource allocation, workload management, and a smooth transition process.

Objectives:

Develop a centralized intern management system: Create a website that enables the organization to track and manage the intern workforce across departments. The system should provide real-time data and analytics for effective decision-making.

Streamline onboarding and offboarding processes: Establish standardized procedures for intern onboarding and offboarding, ensuring a smooth transition experience. This includes developing clear guidelines, providing necessary resources and training materials, and facilitating knowledge transfer between interns and their respective departments.

Deliverables:

Intern management system: Develop and deploy a web solution that enables the organization to track and manage interns across departments. The system should include features for capturing intern details, department assignments, start and end dates, and relevant analytics.

Resource allocation framework:

Establish a framework or mechanism for optimizing the allocation of interns to different departments. This includes guidelines for workload balancing, project alignment, and skill matching.

Workload Division:

Dividing the work we needed to do among 3 people and planning accordingly milestones so there is flexibility and adaptability and regular evaluation and feedback

2.2.2 Requirements Gathering and Analysis (Days 3-5): 31/05 – 02/06

Requirement Analysis:

Engage with Prerak Sir to gather detailed requirements. This involves understanding the specific needs and expectations regarding tracking intern recruitment and turnover, resource allocation, workload management, and transition processes.

Requirement Prioritization:

Prioritize the identified requirements based on their importance and impact on the overall project. Consider factors such as the criticality of accurately tracking intern movement, maintaining workload balance, and ensuring a positive learning experience for interns.

Risk Identification:

Assess potential risks and challenges associated with the project. These may include data accuracy issues, lack of integration with existing systems, resistance to change, resource limitations, or potential privacy concerns. Identify and document these risks to mitigate them effectively during the project execution.

Use Cases:

Create and identify use cases to capture the functional requirements from the perspective of different stakeholders. User stories provide a clear understanding of how the system should behave and the expected outcomes. Use cases outline specific scenarios and interactions with the intern management system.

Functional Specifications:

Based on the requirements and user stories/use cases, develop detailed functional specifications. These specifications describe the specific functionalities, features, and interactions of the intern management system. It should cover aspects such as intern registration, department assignment, intern movement tracking, reporting, and data analytics.

Validation and Verification:

Validate the requirements with the stakeholders to ensure they accurately represent their needs and expectations. Verify that the requirements align with the overall goals of the project and address the identified problem statement effectively.

Project Lifecycle:

Establish a process to handle any modifications or additions to the requirements throughout the project lifecycle. Implement mechanisms to evaluate and incorporate changes while considering their impact on the project timeline, resources, and stakeholders' needs.

By thoroughly analyzing the requirements, prioritizing them, and identifying potential risks, you can ensure a clear understanding of the project objectives and the necessary functionalities of the intern management system. This approach helps in minimizing ambiguity, setting clear expectations, and lay the foundation for successful project execution.

2.2.3 Sprint 1 - Design and Planning (Days 6-10): 03/06 – 07/06**Determine Sprint Duration:**

Decide on when we show sir the planning which typically, happened one to three times a week. Consider the project complexity, team capacity, and other factors when determining the sprint duration.

Task Estimation:

Estimate the effort required for each task in the sprint backlog. This helps in managing workload distribution and setting realistic expectations for the sprint as well as meeting expectations and test case scenarios set by sir

Task Assignment:

Assign the tasks from the sprint backlog to individual team members based on their skills and the role given.

Define the Architecture and Design of the Solution:

We decided to use HTML 5, CSS, and JavaScript in the front end as it was comfortable with all three of the members and also as we can use ASP.NET Core 6 and Microsoft SQL in the backend which is currently in the majority of Linde Engineering Projects also considering scalability, security, integration requirements, and future expansion were considered later on we also planned the flow of the code

After creating wireframes and a database schema for the intern management system, we sought feedback and validation from Prerak Sir. The next step involves presenting the wireframes and database schema to the supervisor, incorporating any suggested changes, and improving upon identified shortcomings

2.2.4 Sprint 2 - Development (Days 11-15):**08/06 – 12/06****Set up Development Environment:**

Prepare the development environment by installing the necessary tools, frameworks, and libraries required for the development of the intern management system. In our case we used VS Code IDE and live server to run the website

Task Breakdown and Assignment:

Break down the development tasks based on the design specifications and functional requirements. Siddarth and Dev developed the front end for two different ends and Vrushali coded the backend and took care of the database. We ensured that we aligned with the sprint goals.

Agile Practices and Daily Stand-up Meetings:

Follow Agile practices throughout the development process. We conducted daily stand-up meetings where the team members provide updates on their progress, discuss any challenges or roadblocks they are facing, and coordinate their efforts. This promotes effective communication, collaboration, and transparency within the team.

Iterative Development:

We adopted an iterative approach to development, where functionality is built incrementally in small increments or iterations. Each iteration should deliver a usable and testable portion of the intern management system. This allows for frequent feedback, course correction, and early detection of issues as well as making it easy to understand or update the flow of the program.

2.2.5 Sprint 3 - Testing and Quality Assurance (Days 16-20): 13/06 – 17/06

Test Planning:

We defined a comprehensive test plan that outlines the testing approach, test objectives, test scope, and test deliverables. Identify the functional and non-functional aspects of the system that need to be tested. Determine the testing techniques, tools, and environments required for conducting effective testing.

Unit Testing:

We performed unit testing to validate the individual components or units of the intern management system. This involves testing each unit in isolation to ensure they function correctly and produce the expected outputs. Unit tests should cover a range of scenarios and edge cases to verify the behaviour of the components.

Integration Testing:

Conducted integration testing to verify the interactions and data flow between different components or modules of the system. This testing ensures that the integrated parts of the system work together seamlessly. Test scenarios that involve the communication and integration points between modules.

System Testing:

Performed system testing to evaluate the entire intern management system as a whole. This involves testing the system's end-to-end functionality and validating its compliance with the specified requirements. Test various user scenarios, input combinations, and business workflows to ensure the system performs as expected.

2.2.5 Sprint 4 - Deployment and User Acceptance (Days 21-25): 18/06 – 22/06

Infrastructure Assessment:

Evaluate the existing infrastructure and determine if it meets the requirements for deploying the intern management system. Consider factors such as hardware, network resources, databases, servers, and security measures.

Scalability and Performance:

Assess the scalability and performance requirements of the intern management system. Ensure that the infrastructure can handle the expected number of concurrent users, data volumes, and system load.

Data Management:

Establish appropriate database management systems or data storage solutions to support the intern management system's data storage and retrieval needs. Ensure that data backups, replication, and disaster recovery mechanisms are in place to safeguard critical data.

Security Measures:

We implemented robust security measures to protect sensitive data and ensure the system's security. To achieve this we used Windows based activation, it also makes sure no other person out of Linde can access or one person can't modify data even if he has someone else's user account

**2.2.6 Sprint 5 – Documentation and Project Closure (Days 26-31):
23/06 – 28/06**

Development Completion:

Review the project's development tasks and identify any pending features, functionalities, or enhancements that need to be implemented. Coordinate with the development team to finalize the remaining tasks and ensure that they are integrated into the system.

Testing and Bug-Fixing:

Conduct thorough testing of the system to identify any bugs, errors, or inconsistencies. Collaborate with the testing team to address the identified issues and ensure that they are resolved satisfactorily. Perform regression testing to verify that the fixes have not introduced new problems.

Quality Assurance:

Review the overall quality of the developed system by conducting comprehensive quality assurance checks. Ensure that the system meets the defined requirements, adheres to coding standards, and follows best practices. Perform performance testing, security testing, and compatibility testing, if applicable.

Accurate and up-to-date project documentation is essential for future reference and maintenance of the intern management system. Here's an elaboration of the activities involved:

Documentation Review:

Review the existing project documentation, including design documents, technical specifications, user manuals, installation guides, and any other relevant documentation. Identify any outdated or incomplete information that needs to be revised.

Document Updates:

Update the project documentation to reflect the final system architecture, design, and functionality. Incorporate any changes made during the development and testing phases. Ensure that the documentation is comprehensive, easy to understand, and serves as a reliable reference for future users and maintainers of the system.

CHAPTER 3 SAMPLE SCREENSHOTS

3.1 FRONT END

Concerned person: The employee is responsible to set a time frame and availability of Interns each month and accept or reject an intern.

Concerned Person View:

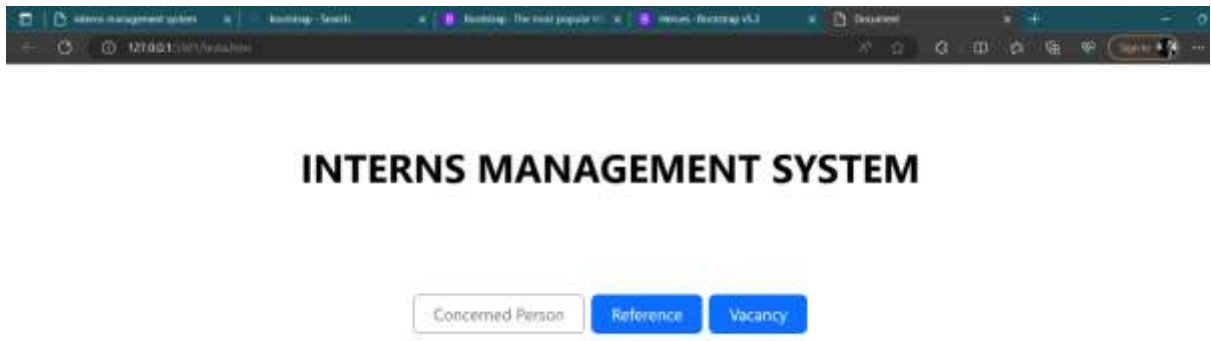


Fig 3.1 Home Page

Employee View:

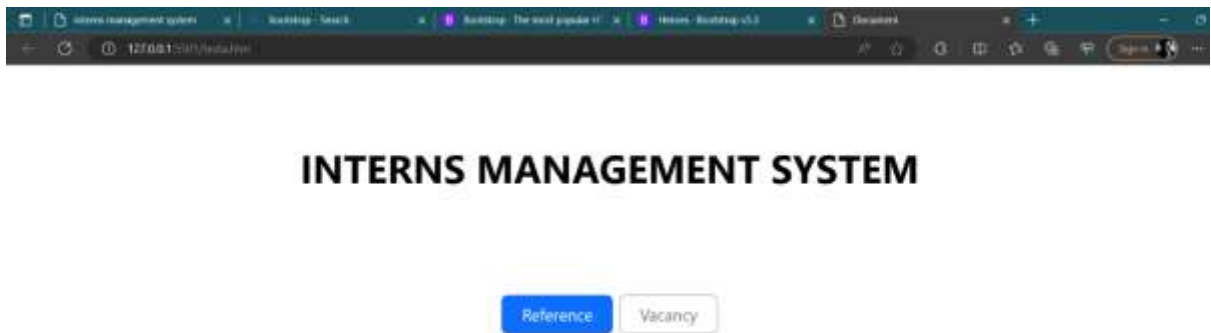


Fig 3.2 Concerned Person Page

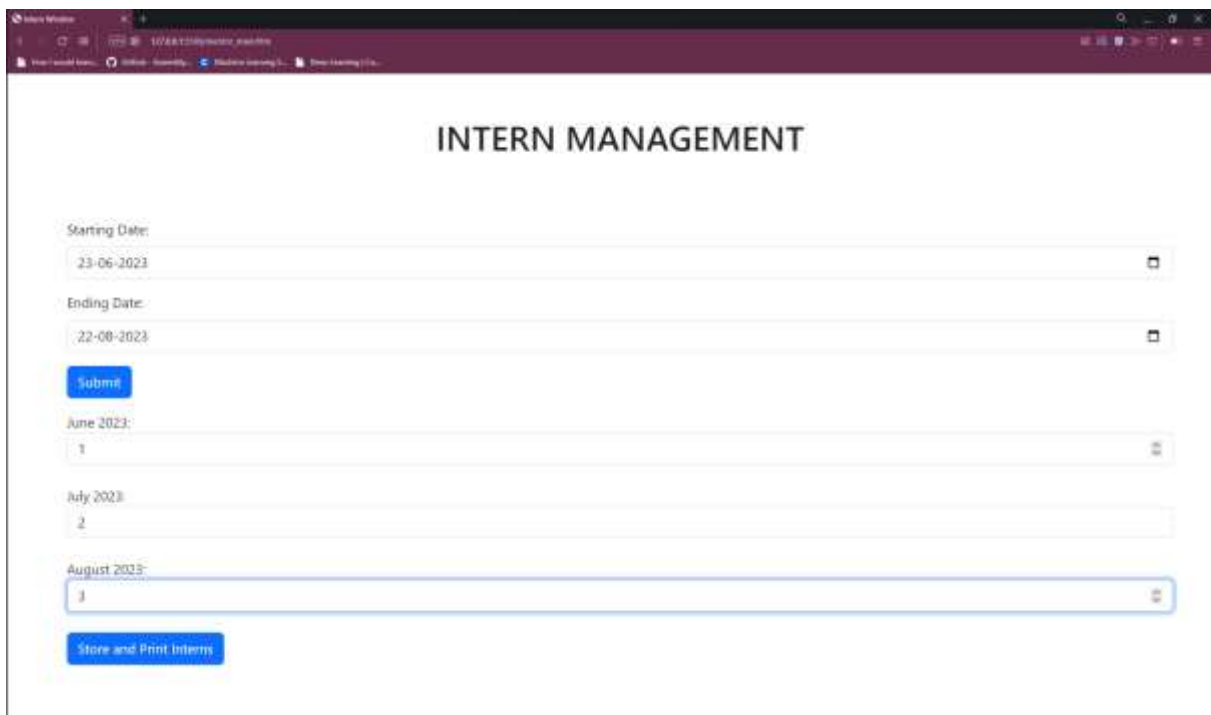
The concerned person will see this page upon opening the project, by clicking :
Concerned Person we will see the form where we can give in the date window.
Reference, the employee from Linde can fill in the form.
Vacancy, for seeing the calendar view of vacant position



The screenshot shows a web browser window with the title "INTERN MANAGEMENT". The page has a dark purple header. Below the header, the title "INTERN MANAGEMENT" is centered. There are two date input fields: "Starting Date:" and "Ending Date:". Both fields have a placeholder "dd-mm-yyyy" and a calendar icon on the right. Below these fields is a blue "Submit" button.

Fig 3.3 Page where a concerned person can enter dates

Here the concerned person can enter the start date and end date in format (dd-mm-yyyy) in such a way that the end date isn't before the start date and that the start day can't be before the present day, and the appropriate alert is shown in case after the alert error is resolved then only the submit button will activate.



The screenshot shows the same "INTERN MANAGEMENT" page. The "Starting Date:" field now contains "23-06-2023" and the "Ending Date:" field contains "22-08-2023". The "Submit" button is still present. Below these fields are three more input fields for months: "June 2023:", "July 2023:", and "August 2023:". Each field has a placeholder "1", "2", and "3" respectively. Below these fields is a blue "Store and Print Interns" button.

Fig 3.4 Page where a concerned person can enter availability

After entering two dates and we get a list of months between (and including) the start date and end date where we can enter the number of interns (only whole numbers are allowed to be entered) after all the months are filled with a number then the button labeled Store and Print Interns will activate.



INTERN MANAGEMENT

Starting Date:
23-06-2023

Ending Date:
22-08-2023

Submit

June 2023:
1

July 2023:
2

August 2023:
3

Store and Print Interns

June 2023: 1
July 2023: 2
August 2023: 3
Total Interns: 6

Fig 3.5 Page where a concerned person can update any changes

After clicking on the button labeled “Store and Print” we get an overview of all the months between the start date and end date and the number of interns in each month and the Total Interns(sum) of the total number of interns in the months.

If we select a date that was already selected, in this case, the next date is between months 7 and 9 we already get the numbers entered in the previous months.



INTERN MANAGEMENT

Starting Date:
13-07-2023

Ending Date:
20-09-2023

Submit

July 2023:
2

August 2023:
3

September 2023:
3

Fig 3.6 Page where a concerned person can enter overlapping dates to modify

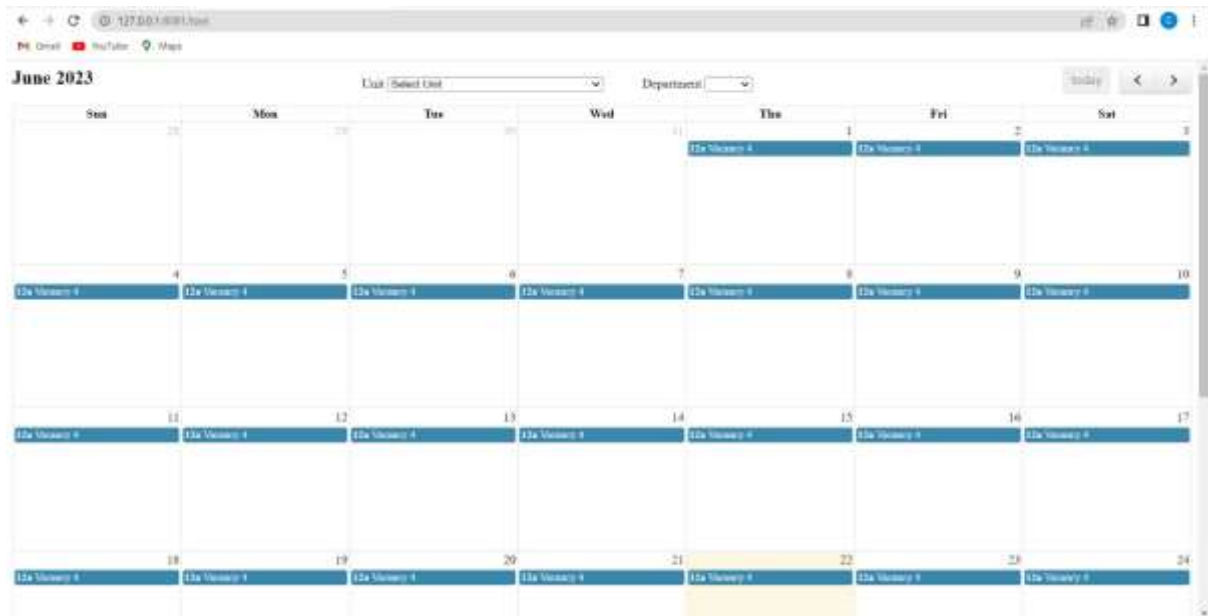


Fig 3.7 Calander view to see availability

The employee who wants to give a reference will check the Calander where he can enter the Unit and Department to shortlist internship entries and also view the number of vacancies for the given duration, the employee can note down the start date and end date and go towards the form section to fill out the form

Fig 3.8 Page where employee can give details

The employee who wants to give a reference will fill up the form above and mention all the details and dates according to the Calander above and then the submit button will be activated, on clicking the submit button the information will be available to the concerned person.

INTERN APPROVAL/REJECTION

INTERN NAME

- Full Name: Name Surname
- Date of Birth: 13-12-2002
- Email: test@gmail.com
- Mobile Number: +91 9876543210
- Gender: Male
- Current Qualification: B.Tech IT(Pursuing)
- Branch: Information Technology
- Unit: Information Technology
- Department: IT Application Service LEI
- Start Date: 29-05-2023
- End Date: 28-06-2023
- [Resume](#)

Fig 3.9 Concerned persons candidate approval or rejection page

The data from the form will be seen like this all the information will be provided to the concerned person and he can click on accept or reject to accept or reject an employee.

On clicking accept an automated mail will be sent to the employee who gave the reference and the intern saying that his internship report is accepted and he will get a follow-through call after some days.

On clicking reject an automated mail will be sent to the employee who gave a reference that the intern whose reference he gave is rejected.

3.2 BACK END

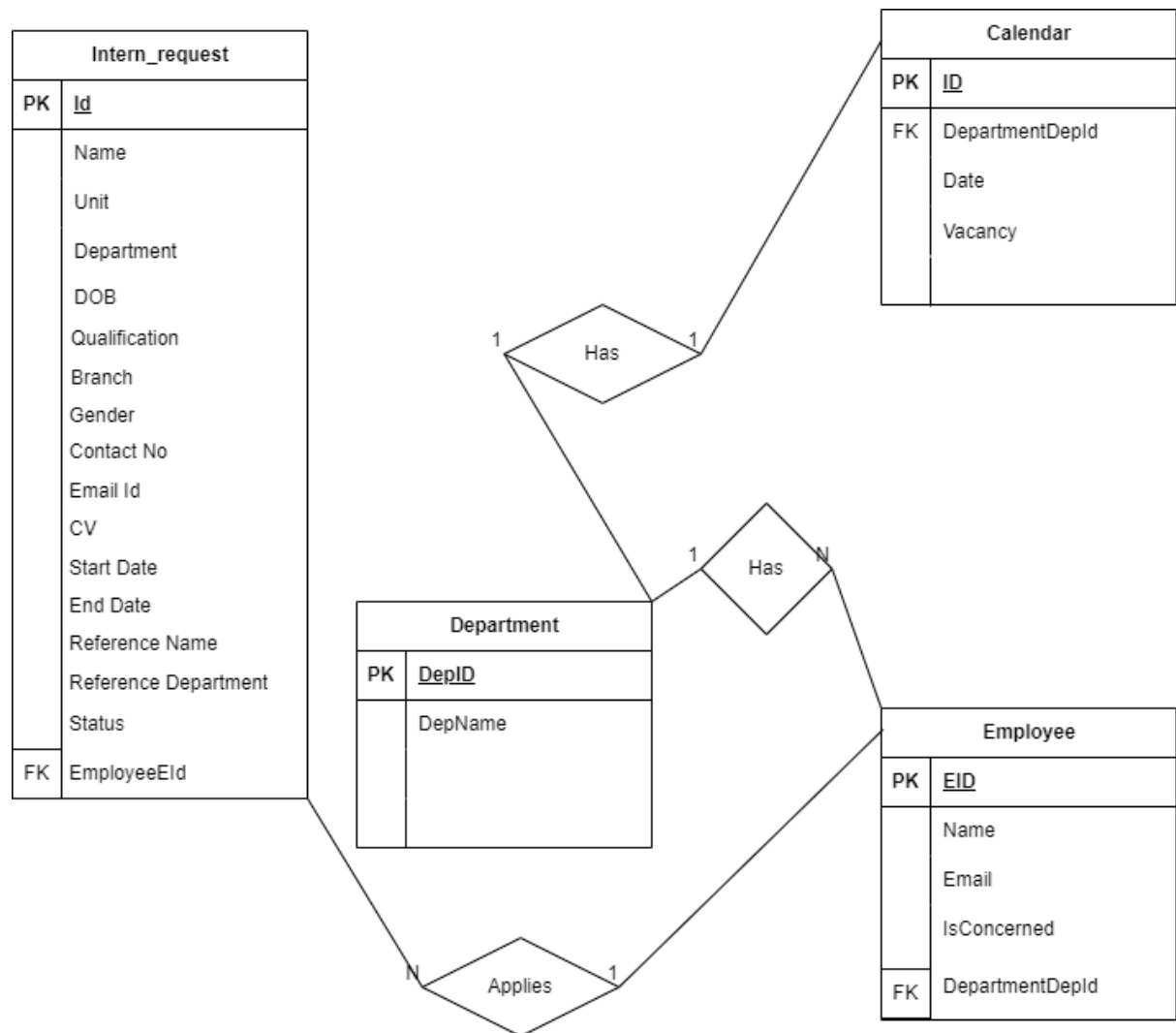


Fig 3.10 Database schema

CHAPTER 4 CONCLUSION

In conclusion, my internship at Linde Engineering India has been an invaluable learning experience, where I had the opportunity to work on diverse and cutting-edge technologies. During my internship, I was actively involved in front-end development using HTML5, CSS, and JavaScript, crafting visually appealing and user-friendly interfaces. Additionally, I gained hands-on experience with back-end and database technologies, including ASP.NET Core 6, Microsoft SQL, and Node.js, building robust and scalable applications. Working with Fetch API allowed me to create efficient and seamless interactions with various APIs. The Visual Studio Integrated Development Environment proved to be a powerful tool, enhancing my productivity and coding experience.

I am sincerely grateful to the entire team at Linde Engineering India for providing me with this opportunity and for their guidance and support throughout my internship.

CHAPTER 5 REFERENCE

Front-End Development:

MDN Web Docs: A comprehensive resource by Mozilla for learning HTML, CSS, and JavaScript. It's often considered one of the best references for web development.

Website: MDN Web Docs

W3Schools: A widely-used online resource with tutorials and references on HTML, CSS, JavaScript, and more.

Website: W3Schools

CSS-Tricks: A blog and community for web designers and developers, providing tutorials, tips, and tricks for CSS and front-end development.

Website: CSS-Tricks

DevDocs: A single-page collection of various web development documentation, including HTML, CSS, and JavaScript references.

Website: DevDocs

JavaScript Specific:

JavaScript MDN Web Docs: A comprehensive guide to JavaScript, including in-depth documentation and tutorials.

Website: JavaScript MDN Web Docs

Eloquent JavaScript: A free online book by Marijn Haverbeke that covers JavaScript in depth, suitable for beginners and intermediate developers.

Website: Eloquent JavaScript

Software Development:

Stack Overflow: An online community where developers can ask and answer technical questions related to software development.

Website: Stack Overflow

GitHub: A platform for version control and collaboration, hosting a wide range of open-source projects that you can explore and learn from.

Website: GitHub

Git Documentation: The official documentation for the Git version control system, which is essential for modern software development workflows.

Website: Git Documentation