Easy builder

## By SEMESTER-10th of Fifth Year M.Sc. (CA & IT)

**(2021-2022)**

**GROUP ID - 22**

|  |  |
| --- | --- |
| **Student Name** | **Roll No** |
| **Prajapati Jaypal P.** | **5052** |

### PROJECT GUIDE: - Jeet Vyas

### Name of Company :- KCS

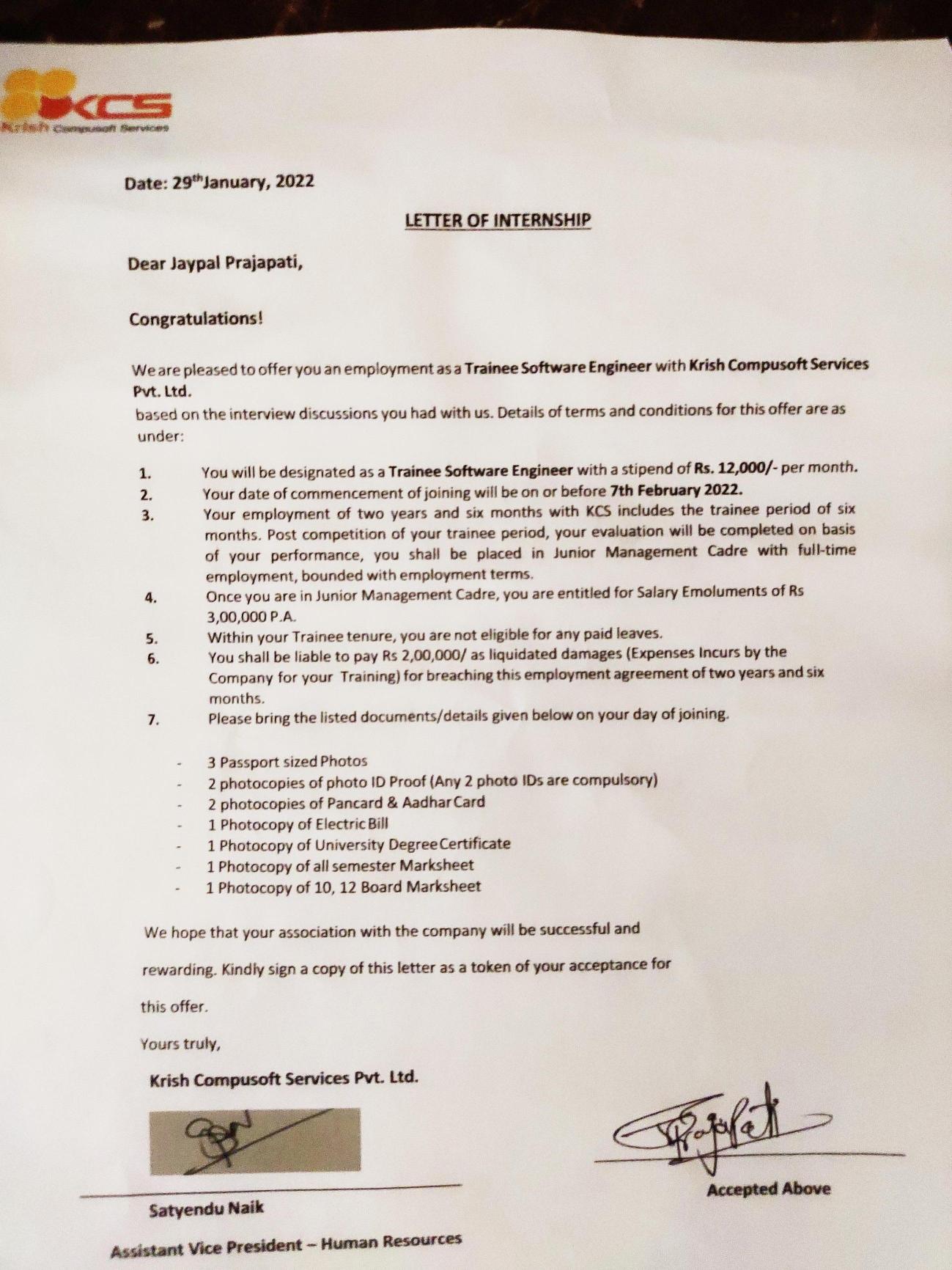
**SUBMITTED TO:-**



**K.S. School OF Business Management**

**M.Sc. In Computer Application and Information Technology**

**Certificate from Company**



**Acknowledgement**

The primary experience of developing our first project was quite interesting and awesome as we are developing our project we would like pay our gratitude to all those who have contributed in our project. Initially starting we would like to be thankful to our internal faculties for their guidance and time contributed in our project development. Further, we would like be grateful to our course External Guide **Mr. Jeet Vyas** for her motivation given during project.

We would like to be grateful to Unity Info way for their support and co-operation for getting to know the requirements, to let us know that there is a need for developing this type of systems. At last we would like to thank our college K.S. School of Business Management for helping us in providing all the necessary equipment.

**Your Sincerely Jaypal Prajapati**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | |  |
| K.S SCHOOL OF BUSINESS MANAGEMENT M.Sc(CA & IT)  Index | | | | | | |  |
|  |  | **No.** | **Title** | | | **Page No.** |  |
| 1 | Introduction | | | 5 |  |
|  | 1.1 | Organization Profile | | 5 |  |
|  | 1.2 | Introduction | | 6 |  |
|  |  | 1.2.1 | Project Definition | 6 |  |
|  |  | 1.2.2 | Importance of System | 6 |  |
|  |  | 1.2.3 | Background/Objective of System | 7 |  |
|  | 1.3 | System Details | | 8 |  |
|  |  | 1.3.1 | Existing System | 8 |  |
|  |  | 1.3.2 | Proposed System | 8 |  |
|  | 1.4 | Scope of System | | 10 |  |
| 2 | Proposed System Requirement Gathering | | | 11 |  |
|  | 2.1 | Stakeholder of System | | 11 |  |
|  | 2.2 | Consolidated List of Requirement | | 11 |  |
| 3 | System Management & Planning | | | 12 |  |
|  | 3.1 | Hardware – Software Requirement | | 12 |  |
|  | 3.2 | System Planning | | 13 |  |
|  |  | 3.2.1 | Work Breakdown Structure | 13 |  |
|  |  | 3.2.2 | Gantt Chart | 14 |  |
|  | 3.3 | Process Model | | 14 |  |
| 4 | System Analysis and Design | | | 16 |  |
|  | 4.1 | E-R Diagram | | 16 |  |
|  | 4.2 | UML (Unified Modelling Language) | | 17 |  |
|  | 4.3 | System flow diagram | | 24 |  |
|  | 4.4 | Data Dictionary | | 25 |  |
|  | 4.5 | User interface (Wire Frames) | | 33 |  |
|  | 4.6 | System Navigation | | 45 |  |
|  | 4.7 | Front End Design | |  |  |
| P a g e 4 | 46 | | | | | | |  |

**CHAPTER-1 INTRODUCTION**

* 1. **Organization profile**



ESMSYS

KCS was founded in 2001, as an exclusive consultant firm

And an Enterprise Service Management company, and with many years of experience in the field of IT consultancy, Brand Development and Financial Services has now emerged as.

KCS has contributed to technology engagements with clients across a wide range of industry verticals and has helped unlock their business potential with innovation. The teams at KCS have the expertise and in-depth knowledge for various industry domains while contributing with several technology solutions like digital transformation, cloud migration to software modernization and more.

KCS is a complete web solution provider company in india having team of young, energetic and dynamic members, focused on providing high notch web Development, Software Solutions, Mobile Applications Development, Internet Marketing service and Project Training. With a team of such experience and knowledgeable members they always try and deliver high quality of project, exceeding client’s expectations.

KCS is a company with a strong specialization in web-development and software development. It also provides project training to the students. Committed to innovate. KCS works hard to its clients to help them to leverage its business process productivity.

## Introduction

* + 1. **Project Definition**

### Definition: “Employee Management System”

* *The objective of “Employee Management System” is to design a scheduling system for a work center.
* The Easy Leave is an Internet based application that can be accessed throughout the organization. This system can be used to automate the workflow of leave applications and their approvals. The periodic crediting of leave is also automated.
* There are features like email notifications, cancellation of leave, report generators etc. in this Tool.
* Employees are the backbone of any the company, management of employee performance plays a major role in deciding the success of the organization.

## Importance of System

* + - * It maintains the information about the personal details of their employees.
      * Employee Management System is a distributed application, developed to maintain the

Details of employees working in any department.

* + - * The system also manages the company employee details like their personal details and leave details.
      * The aim is to automate its existing manual system with the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulate of the same.

## Background/Objective of System

* + - * To reduce workload of a manually work.
      * To reduce paperwork and reduce time for process.
      * The system makes the overall project management much easier and

Flexible.

* + - * Easy to understand by leave are approved or not.
      * All employee Leave details maintain in System.
      * Easy to manage employee details.

## System Details

* + 1. **Existing System**
    - The Problem definition for designing the system is to maintain data of employee, to make easy controlling employees, to divide jobs and access control of employees, to use technology for accurate and timely processing by fully privacy and full authorities’ access.
    - The objective of the project is to set up employee information system about status of employee Leave.
    - To eliminate or reduce as much as possible the hardships of existing system and avoid errors while entering data. In existing method employee management are employee record are maintaining in records. It’s a manual process. Complicated to search the employee leave.
    - This increases the paperwork & maintaining the records becomes tedious.

## Proposed System

* The proposed system is internet-based system so employee can also participate in this system and show their information.
* The proposed system provides detail general information's about the employee.
* It enhances the managements in adding, viewing, and updating employee’s details and generates various reports regarding employee’s leave.
* The periodic crediting of leave is also automated. There are features like email notifications, cancellation of leave, automatic approval of leave, report generators etc. in this Tool.
* Provides the searching facilities based on Employee name.

## Scope of System

* The Dashboard displays different data depending on the Role of Employee.
* Different reports regarding leave can be seen from section.
* Employees can also edit personal details.
* Employees can apply for leaves, check their leave balance, approve team members’

Leave applications, and see a few reports related to Leaves depending on

their Roles.

* + - **Limitations: -**
* Complex outcomes & demands cannot be satisfactorily fulfilled to the users.
* Limited up to process department of company group only.
* **Advantages of Leave Management System:**
* Easy to maintain in Centralized Database Leave applications can be submitted via network.
* Leave application can be approved easily and notified to respective Admin.
* The leave applicant as well as the approver can view the balance leave and past leave applications.

## CHAPTER-2 PROPOSED SYSTEM REQUIREMENT GATHERING

* 1. **Stakeholder of System**

We have analyzed the employee workflow and getting mostly requirement on

company work environment.

1. **Admin**
   * The dashboard displays different data depending on the role employee.
   * Information includes personal and leave.
   * See a few reports related to Leaves depending on employees.
   * Can be managed Employees total leave, total pending leave, total canceled leave, and total approved leave.
   * Manage Employees details, employee leave, manage all admin and profile.

## Employee

* + Employee information is added to EMS when a new employee joins the organization.
  + Information includes personal details.
  + The dashboard displays different data depending on employee apply total leave, leave status and total approved leave.
  + Employees can apply for leaves, check their leave balance.
  + The approve team members’ leave applications, and see a few reports related to Leaves depending on their Roles.
  + There is a reporting section that shows few reports

## 

## Consolidated List of Requirement

* + - System must be user friendly.
    - System saves time and energy to the company.
    - System should have security.
    - Manage and update the information of the employees.
    - System is create to employees leave apply.
    - All information about Employees and Admin are fulfilled in System.

## CHAPTER-3 SYSTEM MANAGEMENT & PLANNING

* 1. **Hardware – Software Requirement**

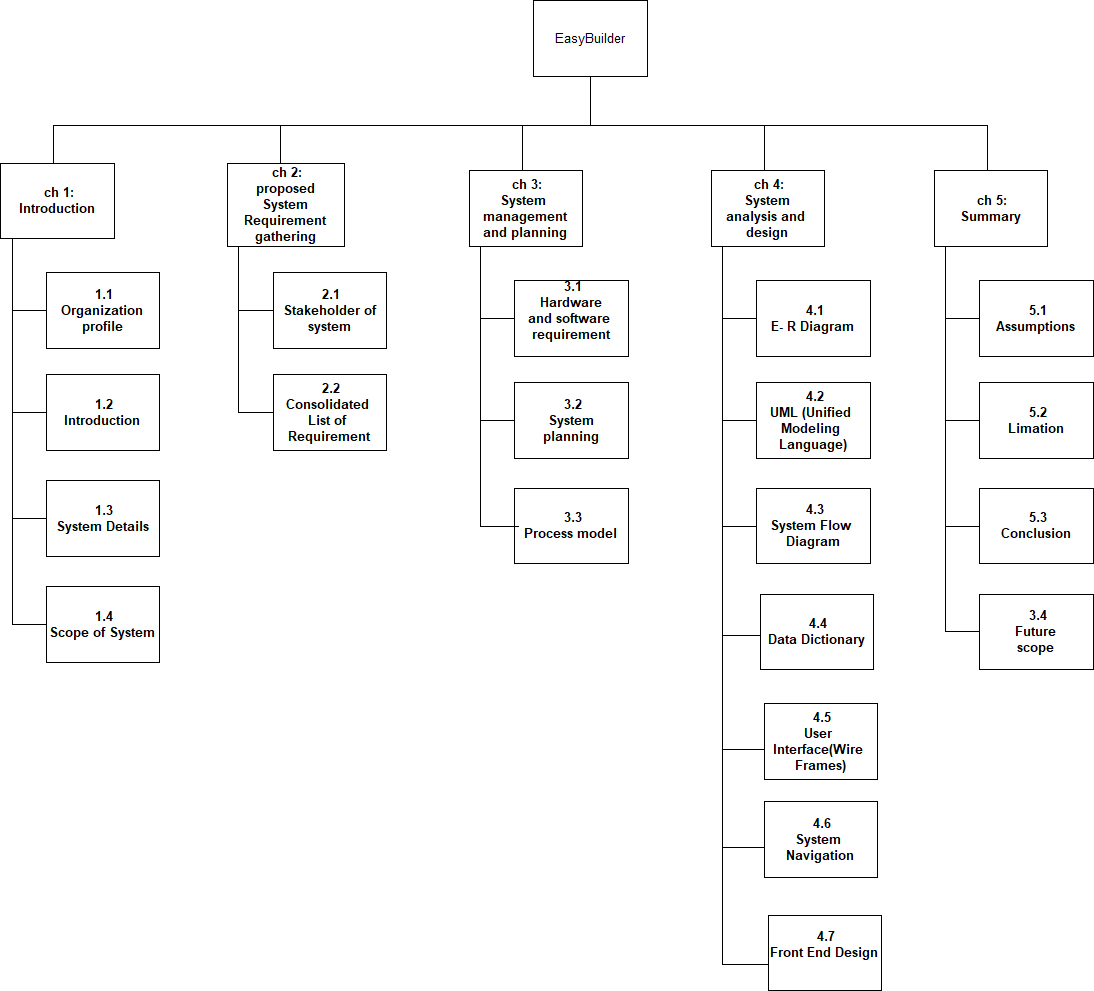
### Software Requirement:

|  |  |
| --- | --- |
| Platform | PHP |
| Operating System | Windows Version above 7 |
| Front-End | Sublime/Visual studio code |
| Server | Xampp |
| Browser | Chrome, Internet explorer, opera, mozila-firefox |
| Database | MySQL |

* + 1. **Hardware Requirement:**

|  |  |
| --- | --- |
| Processor | Core i3 |
| Hard-Disk | 25 GB |
| RAM | 2 GB or more |

* 1. **System Planning**
     1. **Work Breakdown Structure:**



* + 1. **Gantt Chart:**

Gantt charts are use full for planning and scheduling projects. They help you assess how long a project should take. Determine the resources needed, and plan the order in which you’ll complete tasks. They are also helpful for managing the dependencies between tasks.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Id** | **Task name** | **Jan 2022** | | **Feb 2022** | | | **Mar 2022** | | | **Apr 2022** |
| 1 | Definition |  |  | | | | | | | |
| 2 | Analysis |  | | |  | | | | | |
| 3 | Planning |  | | |  | | |  | | |
| 4 | UML Diagrams |  | | | |  | | |  | |
| 5 | Designing |  | | | | | | | | |

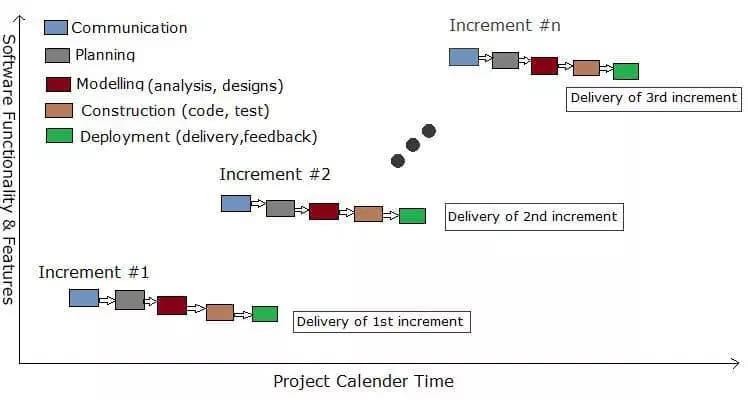
**Figure 3.2.2 Gantt Chart**

## Process Model

As new requirements can arise in future incremental models are used. With the help of it we can fulfil maximum user requirements. In every increment the needs of the client are kept in mind and more features and functions are added. These increments form a base for customer evaluation. Many features can be added after the development of the system that serves the main purpose. If there are less number of employees to work on the project Incremental development model is very useful to complete the project before the deadline.

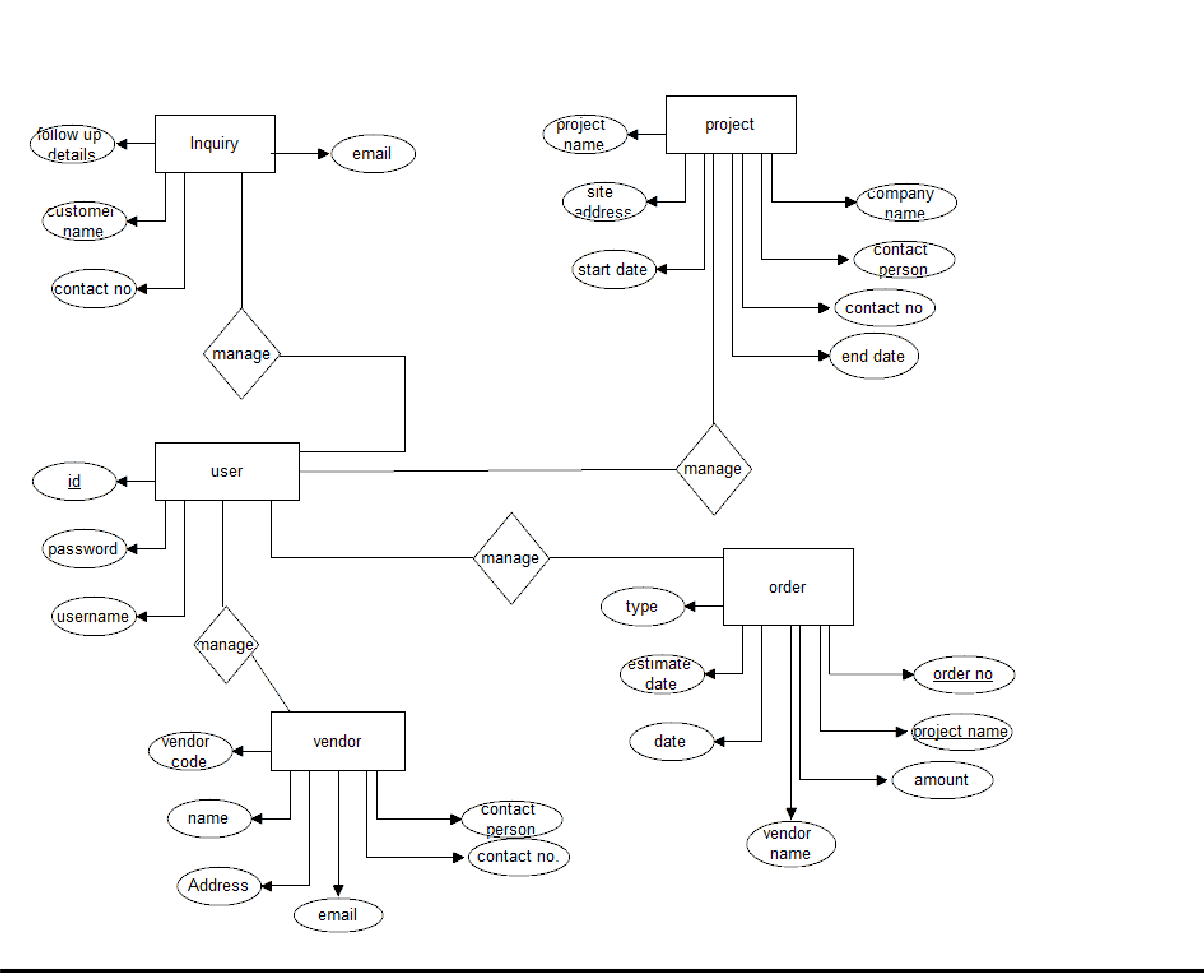
Incremental model:

* This model is more flexible – less costly to change scope and requirements.
* It is easier to test and debug during a smaller Iteration.
* In this model customer can respond to each built.
* Lowers initial delivery cost.
* This process model helps to manage technical risks.
* Initial product delivery is faster.



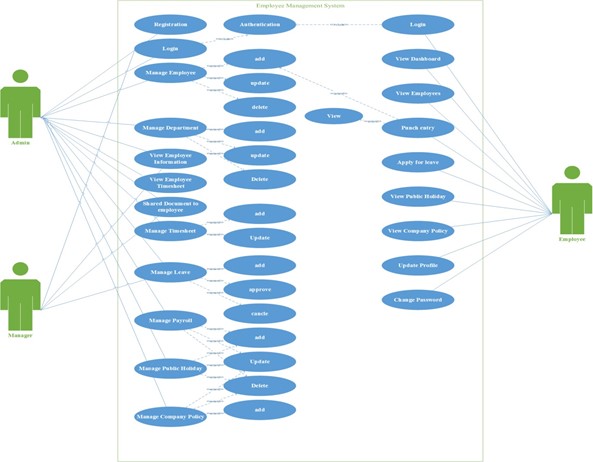
# CHAPTER-4 SYSTEM ANALYSIS AND DESIGN

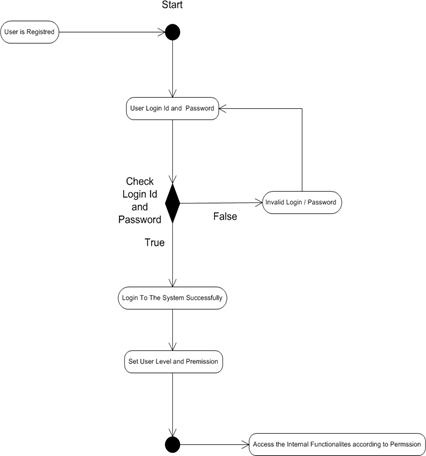
## E- R Diagram:



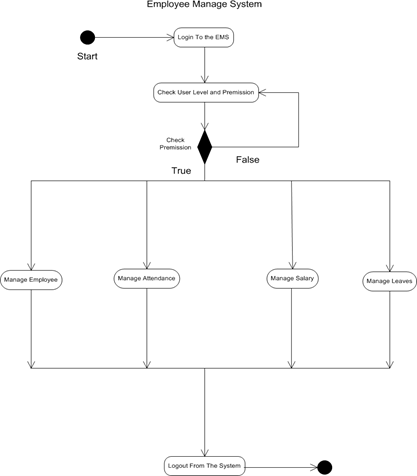
* 1. **UML (UNIFIED MODELING LANGUAGE**

### USECASE Diagram:

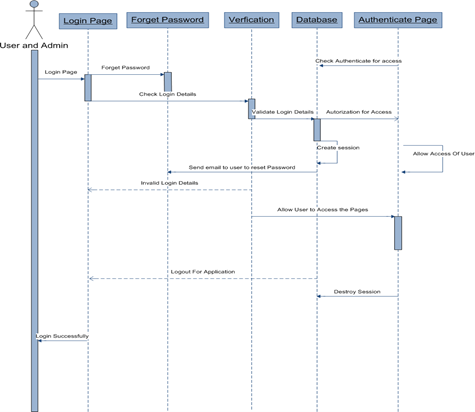
* **Employees Management System** 
  + 1. **Activity Diagram:**
* **ADMIN**



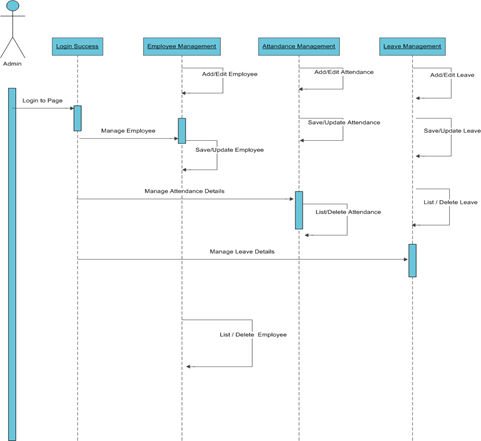
* **Supervisor**



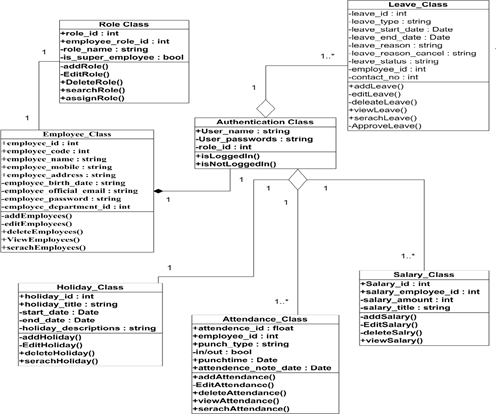
* + 1. **Sequence Diagram:**



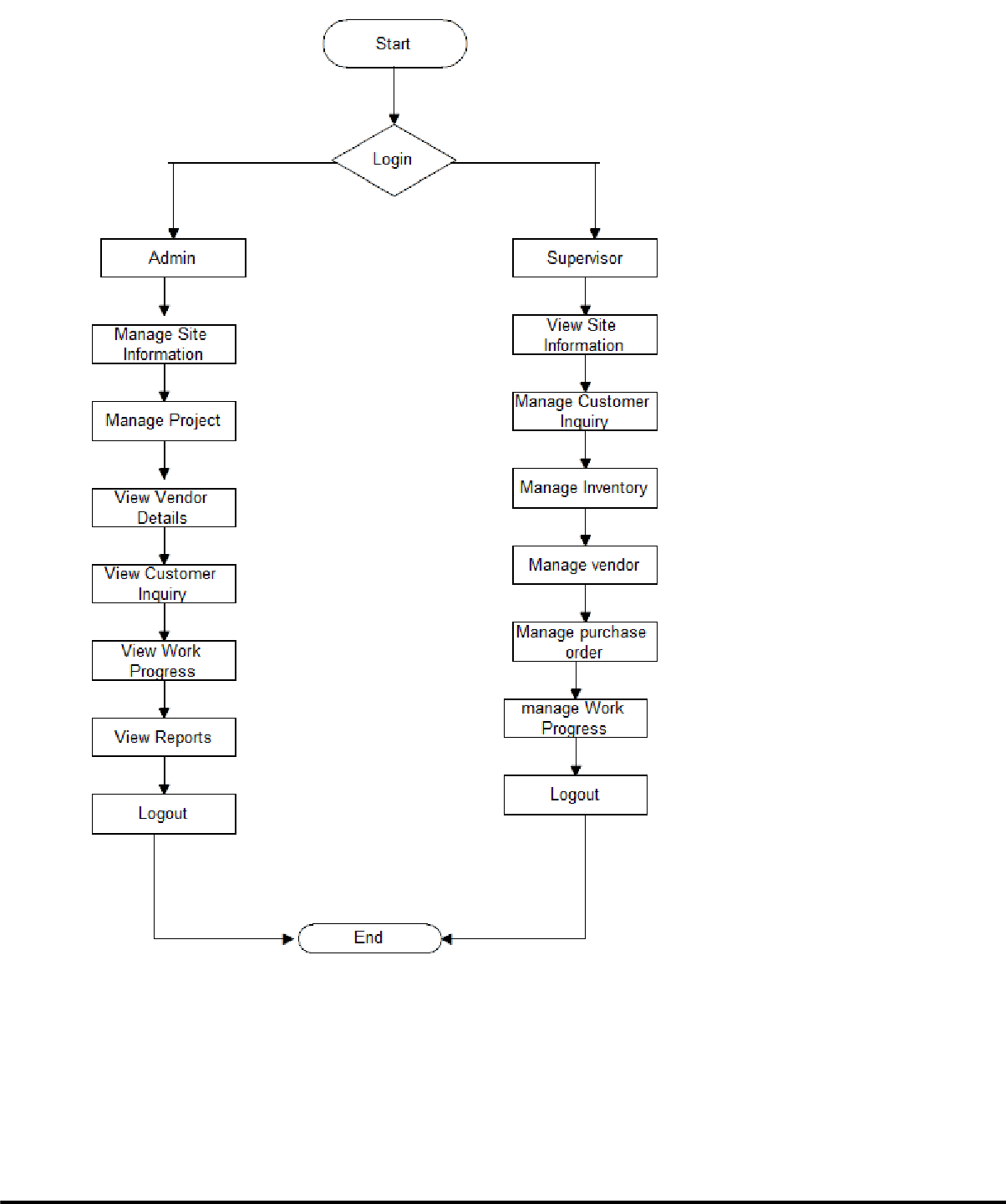
* **Supervisor**



* + 1. **Class Diagram:**



* 1. **System Flow Diagram**



* 1. **Data Dictionary**
  2. **Data Dictionary**

|  |  |
| --- | --- |
| **Sr. No** | **Table Name** |
| 1. | **departments\_master** |
| 2. | **designation** |
| 3. | **user\_roles** |
| 4. | **employee\_details** |
| 5. | **employee\_officials\_information** |
| 6. | **document** |
| 7. | **permanent\_address** |
| 8. | **add\_timesheet** |
| 9 | **apply\_for\_leave** |
| 10. | **update\_salary\_of\_employee** |
| 11. | **professional\_tax** |
| 12. | **teams** |
| 13. | **teams\_members** |
| 14. | **company\_policies** |
| 15. | **public\_holiday\_details** |

* + 1. **Table Name: departments\_master Description:- to store different departments**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type (with size)** | **Constraint** | **Description** | **Sample Data** |
| Department\_id | Int(11) | Primary Key | Id of Department | 1 |
| Department Name | Varchar(50) | Not Null | Add Department Name | Accounts |

* + 1. **Table Name: designation**

**Description:- to store designation details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type (with size)** | **Constraint** | **Description** | **Sample Data** |
| Designation\_id | int(10) | Primary key | Id for designation | 1 |
| designation | varchar(100) | Not Null | Select the employee  designation | Like Designer, Developer |

### User Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **id** | Auto\_Number | Primary Key | 11 |
| **firstname** | varchar | Not null | 50 |
| **lastname** | varchar | Not null | 50 |
| **username** | varchar | Not null | 50 |
| **password** | varchar | Not null | 50 |
| **usertype** | Varchar | Not null | 50 |
| **state\_id** | Int | Foreign key | 3 |
| **city\_id** | Int | Foreign key | 3 |
| **birthdate** | Date | Not null | - |
| **Mobile** | Int | Not null | 11 |
| **email** | Varchar | Not null | 50 |
| **Address** | Varchar | Not null | 255 |
| **Zipcode** | Int | Not nul | 11 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Gender** | Varchar | Not null | 6 |
| **Isactive** | Int | Not null | 1 |
| **Doi** | Datetime | Not null | - |
| **Dou** | Datetime | Not null | - |

**Site Profile Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **Site\_id** | Int | Primary Key | 11 |
| **Company\_name** | Varchar | Not null | 50 |
| **Project\_name** | Varchar | Not null | 100 |
| **Pan** | Varchar | Not null | 50 |
| **Tin** | Varchar | Not null | 50 |
| **St** | Varchar | Not null | 50 |
| **Gstnaumber** | Varchar |  | 50 |
| **Doi** | Datetime | Not null | - |
| **Dou** | Datetime | Not null | 10 |

**Tower Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **Tower\_id** | Int | Primary Key | 11 |
| **Tower\_name** | Varchar | Not null | 50 |
| **Total\_floor** | Int | Not null | 11 |
| **No\_of\_flats\_perfloor** | Int | Not null | 11 |
| **Totalflats\_perfloor** | Int | Not null | 11 |
| **Isactive** | Int | Not null | 2 |
| **Doi** | Datetime | Not null | - |
| **Dou** | Datetime | Not null | 10 |

**Property Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **property\_id** | Int | Primary Key | 11 |
| **property\_name** | Varchar | Not null | 150 |
| **Municipal\_charge** | Double | Not null | (10,2) |
| **Maintaince\_charge** | Double | Not null | (10,2) |
| **Isactive** | Int | Not null | 2 |
| **Doi** | Datetime | Not null | - |
| **Dou** | Datetime | Not null | 10 |

**Property type Details Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **Property\_type\_id** | Int | Primary Key | 11 |
| **Property\_id** | Int | Foreign key | 11 |
| **Area\_type** | Varchar | Not null | 50 |
| **Area\_unit** | Varchar | Not null | 50 |
| **Per\_area\_rate** | Double | Not null | - |
| **Area\_size** | Double | Not null | - |
| **Area\_rate** | Double | Not null | - |

**Customer Inquiry Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **Inquiry\_id** | Auto\_Number | Primary Key | 11 |
| **Name** | varchar | Not null | 150 |
| **Address** | varchar | Not null | 150 |
| **Proof** | varchar | Not null | 150 |
| **Occupation** | varchar | Not null | 200 |
| **Religion** | Varchar | Not null | 50 |
| **state\_id** | Int | Foreign key | 3 |
| **city\_id** | Int | Foreign key | 3 |
| **Birthdate** | Date | Not null | - |
| **Age** | Int | Not null | 3 |
| **Mobile** | Int | Not null | 11 |
| **Home\_contact\_no** | Int | Not null | 11 |
| **Email** | Varchar | Not null | 50 |
| **Property\_id** | Int | Foreign key | 11 |
| **Followup** | varchar | Not nul | 150 |
| **Hear\_about us** | Varchar | Not null | 150 |
| **Status** | ENUM('Inquiry', 'Possible Member',  'Member') | Not null | - |
| **Inquiry\_date** | Date | Not null | - |
| **Sale\_date** | Date | - | - |
| **Is\_active** | Int | Not null | 3 |

**Vendor Details Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **vendor\_id** | Auto\_Number | Primary Key | 11 |
| **Org\_name** | Varchar | Not null | 150 |
| **Org\_address** | Varchar | - | 150 |
| **Org\_pincode** | Int | - | 10 |
| **Proof** | varchar | Not null | 150 |
| **Occupation** | varchar | Not null | 200 |
| **Religion** | Varchar | Not null | 50 |
| **state\_id** | Int | Foreign key | 3 |
| **city\_id** | Int | Foreign key | 3 |
| **Birthdate** | Date | Not null | - |
| **Age** | Int | Not null | 3 |
| **Mobile** | Int | Not null | 11 |
| **Home\_contact\_no** | Int | Not null | 11 |
| **Email** | Varchar | Not null | 50 |
| **Property\_id** | Int | Foreign key | 11 |
| **Followup** | varchar | Not nul | 150 |
| **Hear\_about us** | Varchar | Not null | 150 |
| **Status** | ENUM('Inquiry',  'Possible Member', 'Member') | Not null | - |
| **Inquiry\_date** | Date | Not null | - |
| **Sale\_date** | Date | - | - |
| **Is\_active** | Int | Not null | 3 |

**Purchase order Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **p\_id** | Int | Primary Key | 11 |
| **Porder\_id** | Varchar | Not null | 150 |
| **Order\_date** | Date | Not null | - |
| **Vendor\_id** | Int | Not null | 11 |
| **Total\_Quantity** | Int | Not null | 2 |
| **Status** | Enum(pending,  received) | Not null | - |
| **Payment\_status** | Enum(penuding,  invoice\_recorded) | Not null | - |

**Work Progress Category Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **cat\_id** | Int | Primary Key | 11 |
| **Category\_name** | Varchar | Not null | 150 |
| **Is\_active** | Int | Not null | 2 |

**Work Progress Subcategory Table**

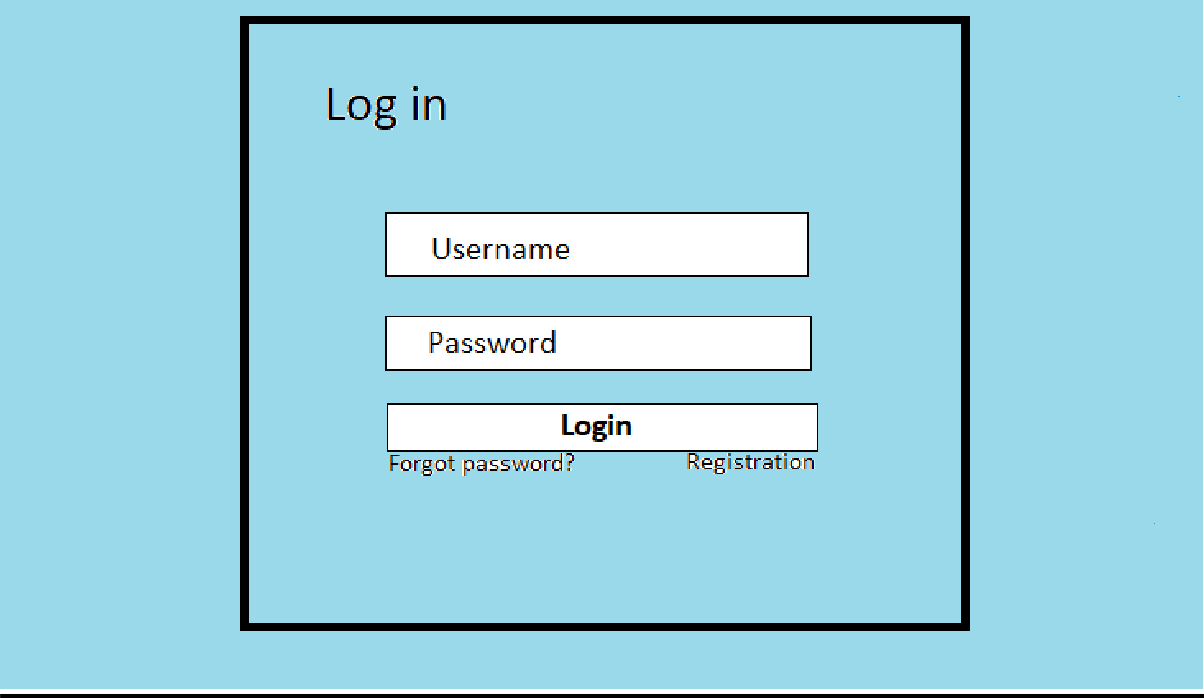
|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **Sub\_cat\_id** | Int | Primary Key | 11 |
| **c\_id** | Int | Foreign key | 11 |
| **subcategory\_name** | Varchar | Not null | 150 |
| **Is\_active** | Int | Not null | 2 |

**Work Progress Table**

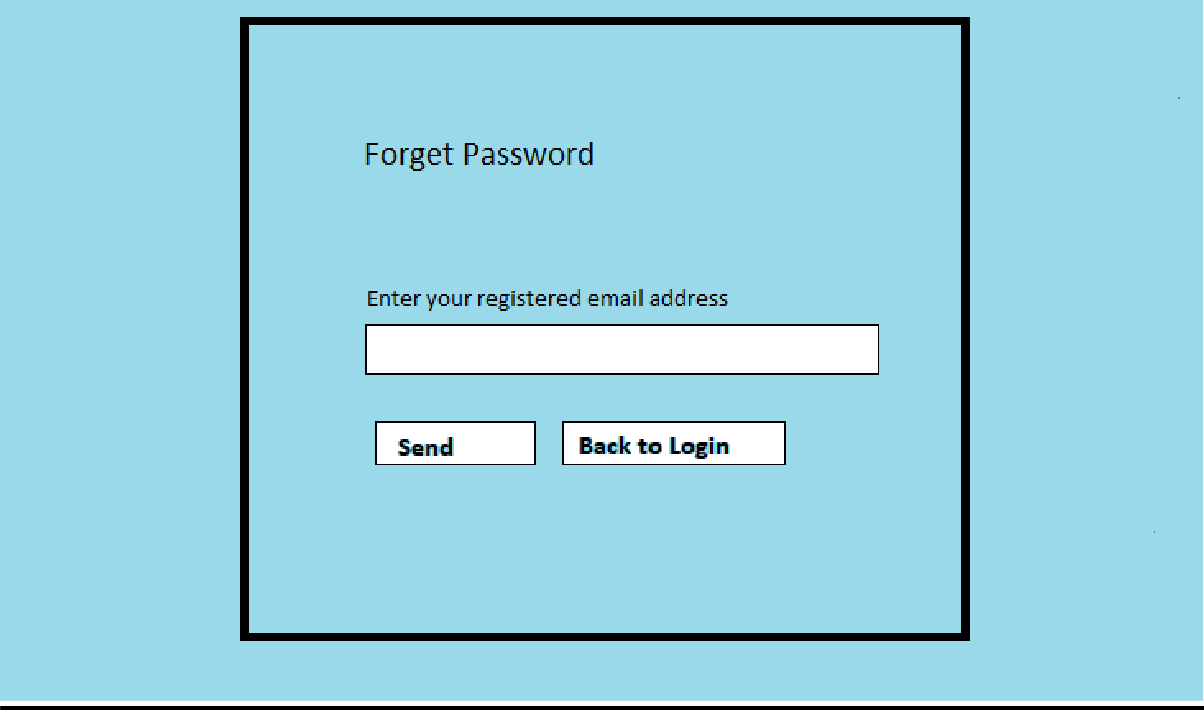
|  |  |  |  |
| --- | --- | --- | --- |
| **Fields** | **Data-Type** | **Constrains** | **size** |
| **wp\_id** | Int | Primary Key | 11 |
| **Cat\_id** | Int | Foreign key | 11 |
| **Sub\_cat\_id** | Int | Foreign key | 11 |
| **item\_id** | Int | Foreign key | 11 |
| **Vendor\_id** | Int | Foreign Key | 150 |
| **Used\_qty** | Int | Not null | 11 |
| **Unit** | Varchar | Not null | 100 |
| **Doi** | Datetime | - | - |
| **Dou** | Datetime | - | - |

* 1. **User Interface (Wire Frames)**

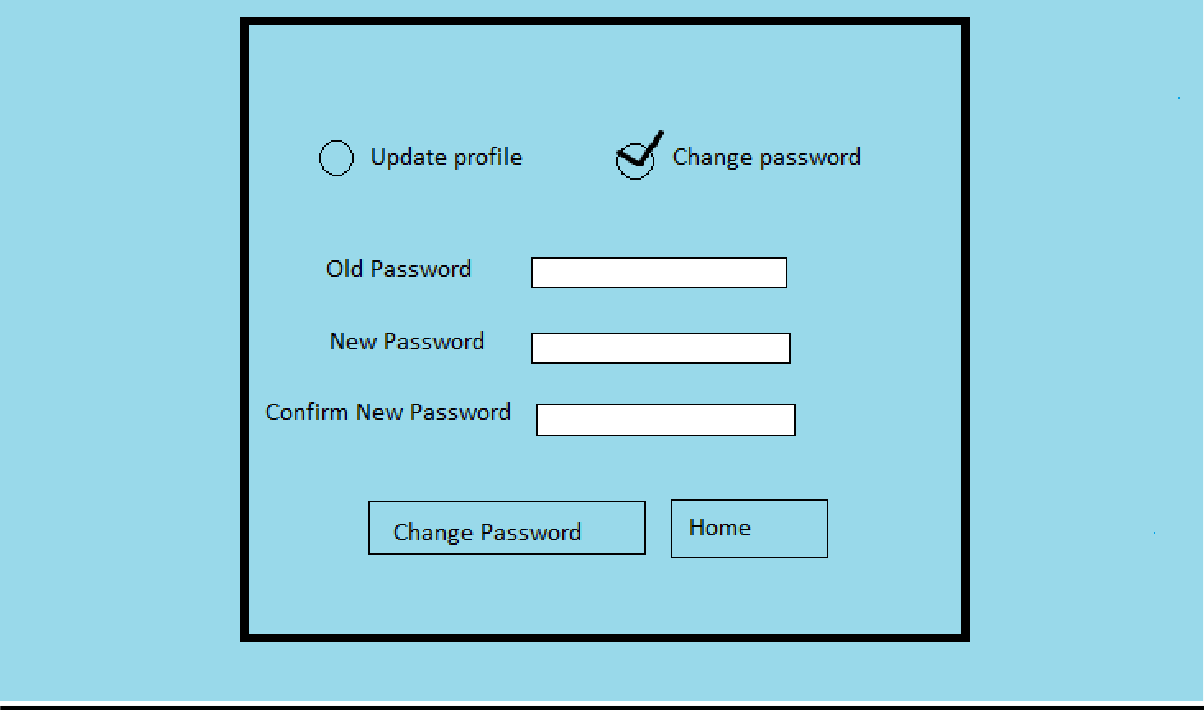
1. Login Page:



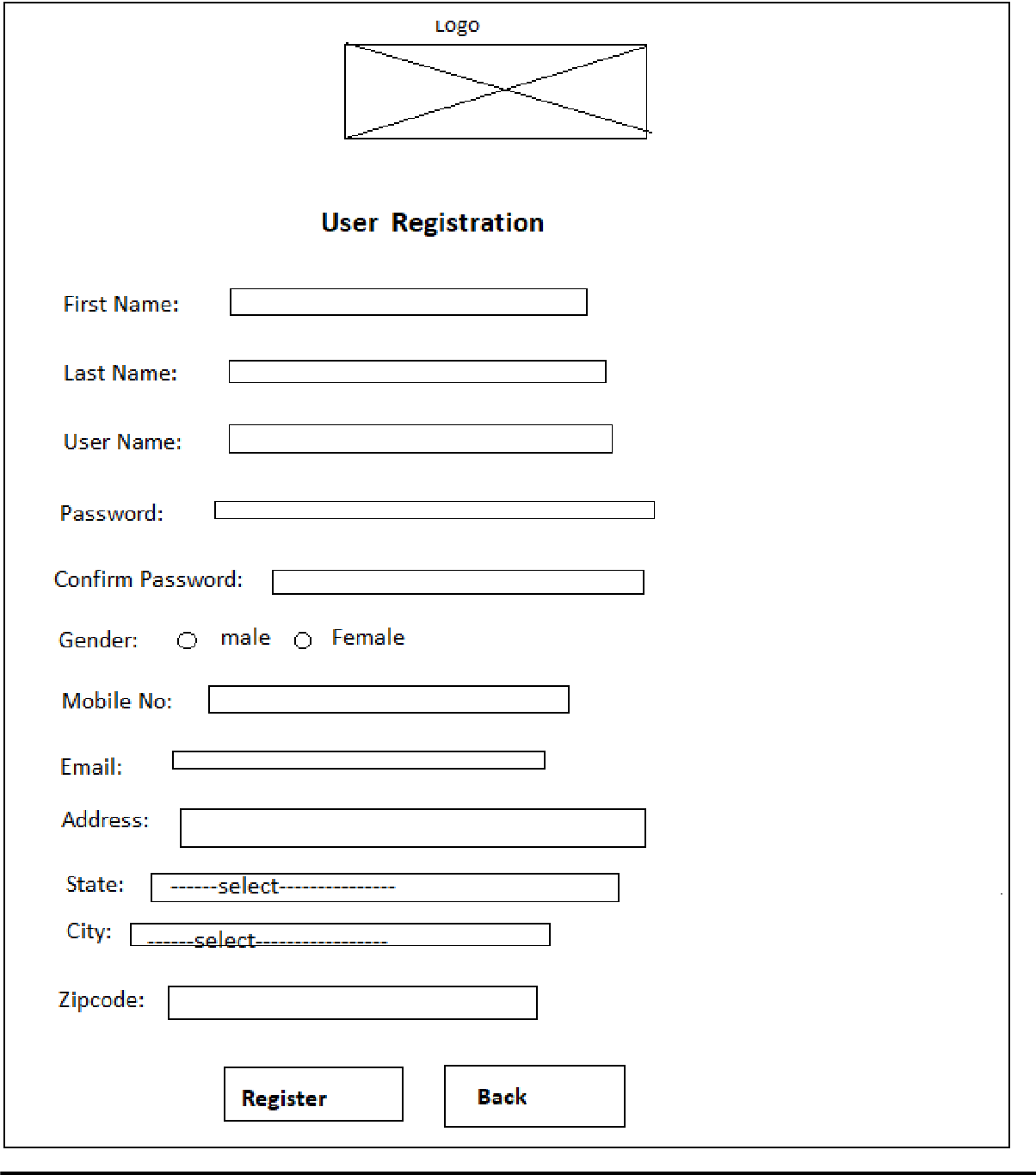
1. Forget Password Page:



1. Reset Password

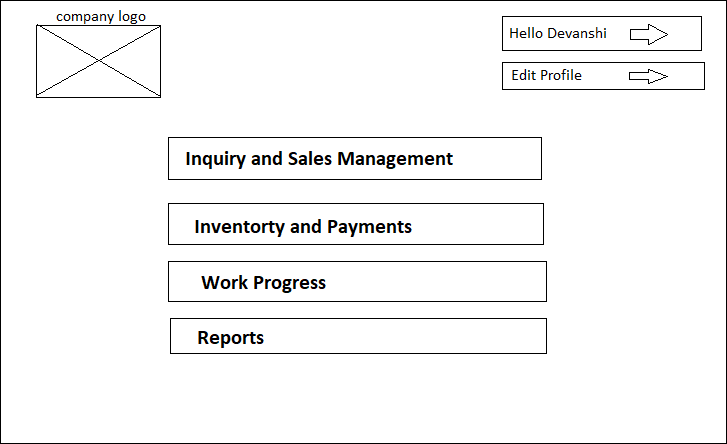


Registration Page:

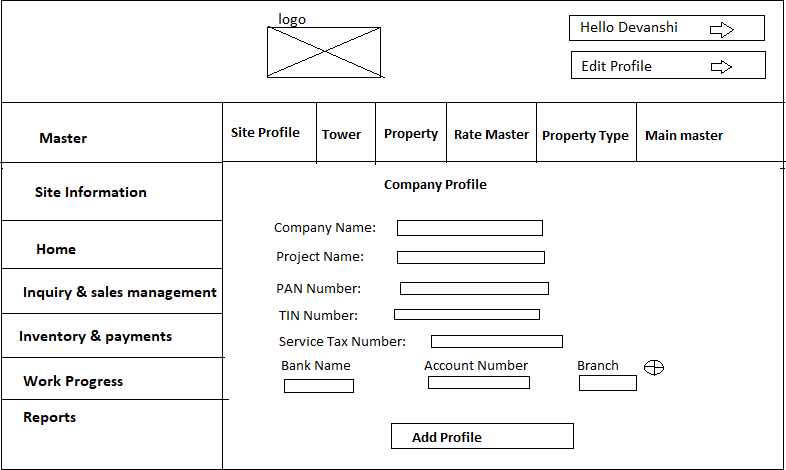


## Admin side

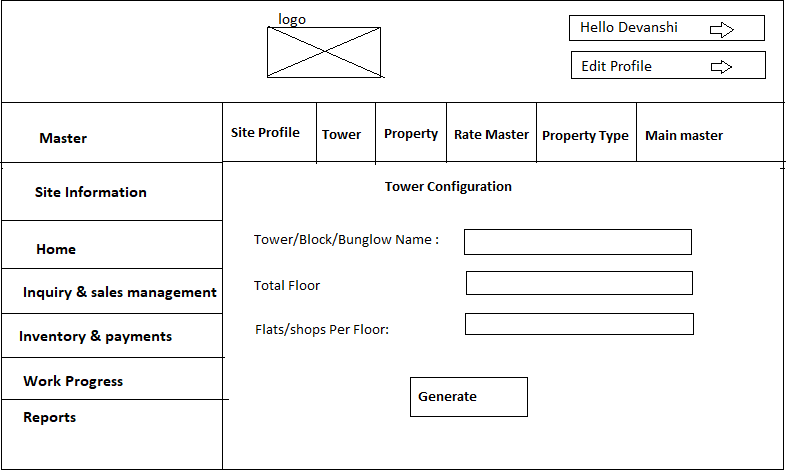
1. Home Page:



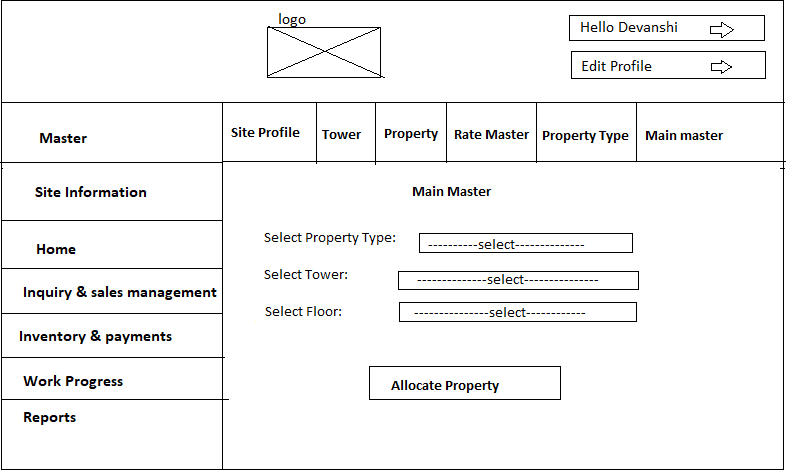
1. Company profile Page:



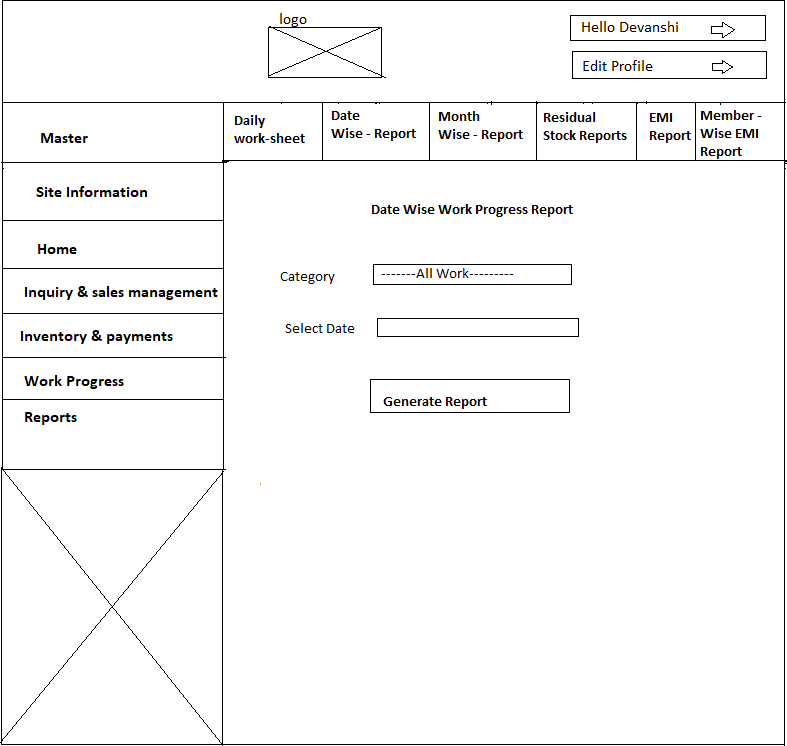
1. Add Tower Page:



1. Main master Page : In this page add property type in different tower and also add how many floor of this tower also you can add.

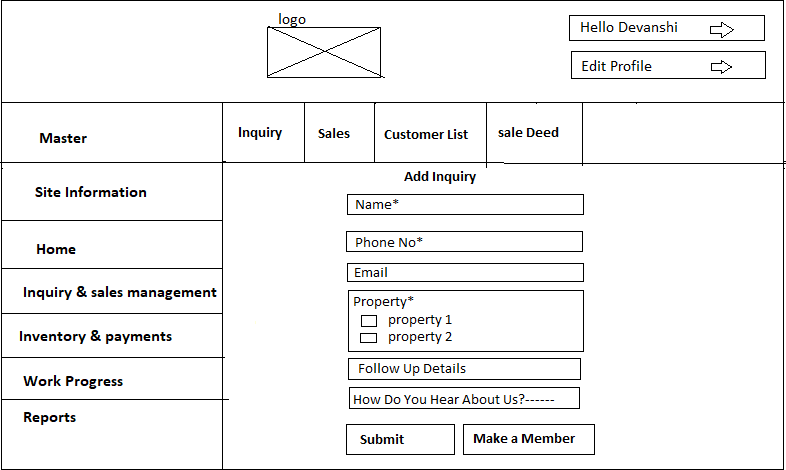


1. Report Page : In this page admin get all report for date wise and month wise also EMI report can get easily.

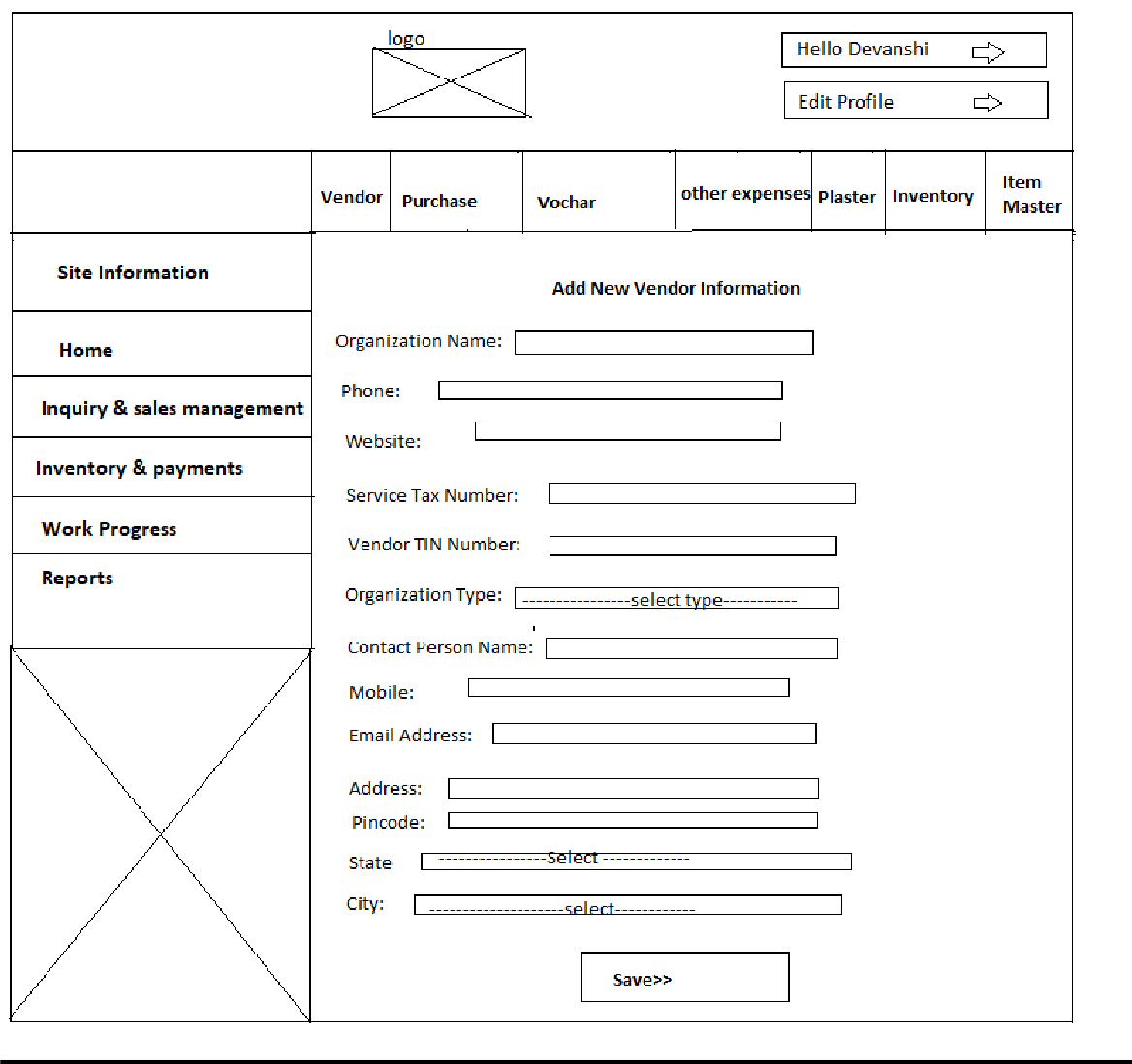


## Supervisor Side

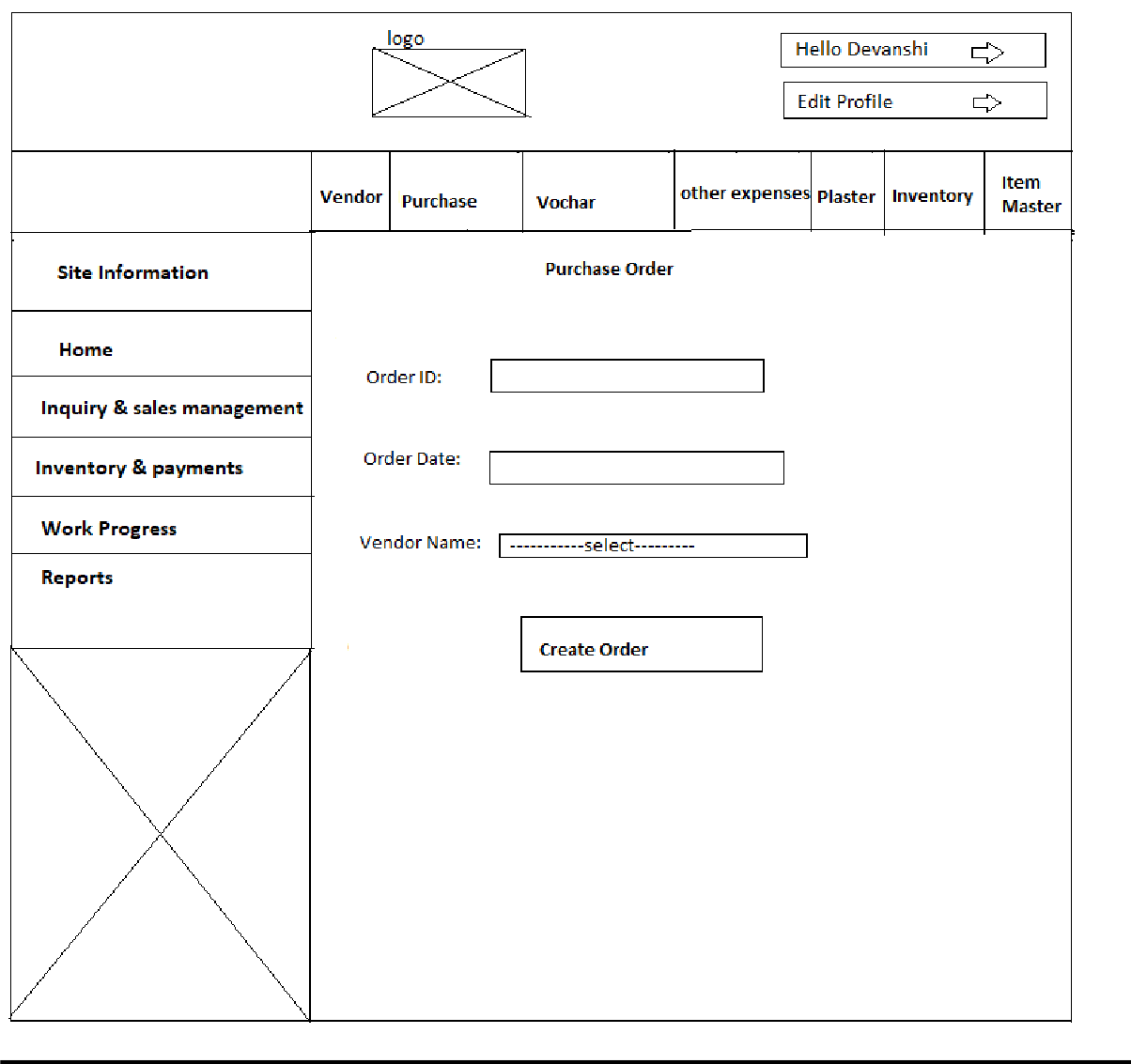
1. Inquiry Page : Below display add inquiry page. In this page supervisor add customer Inquiry and also mention property type .



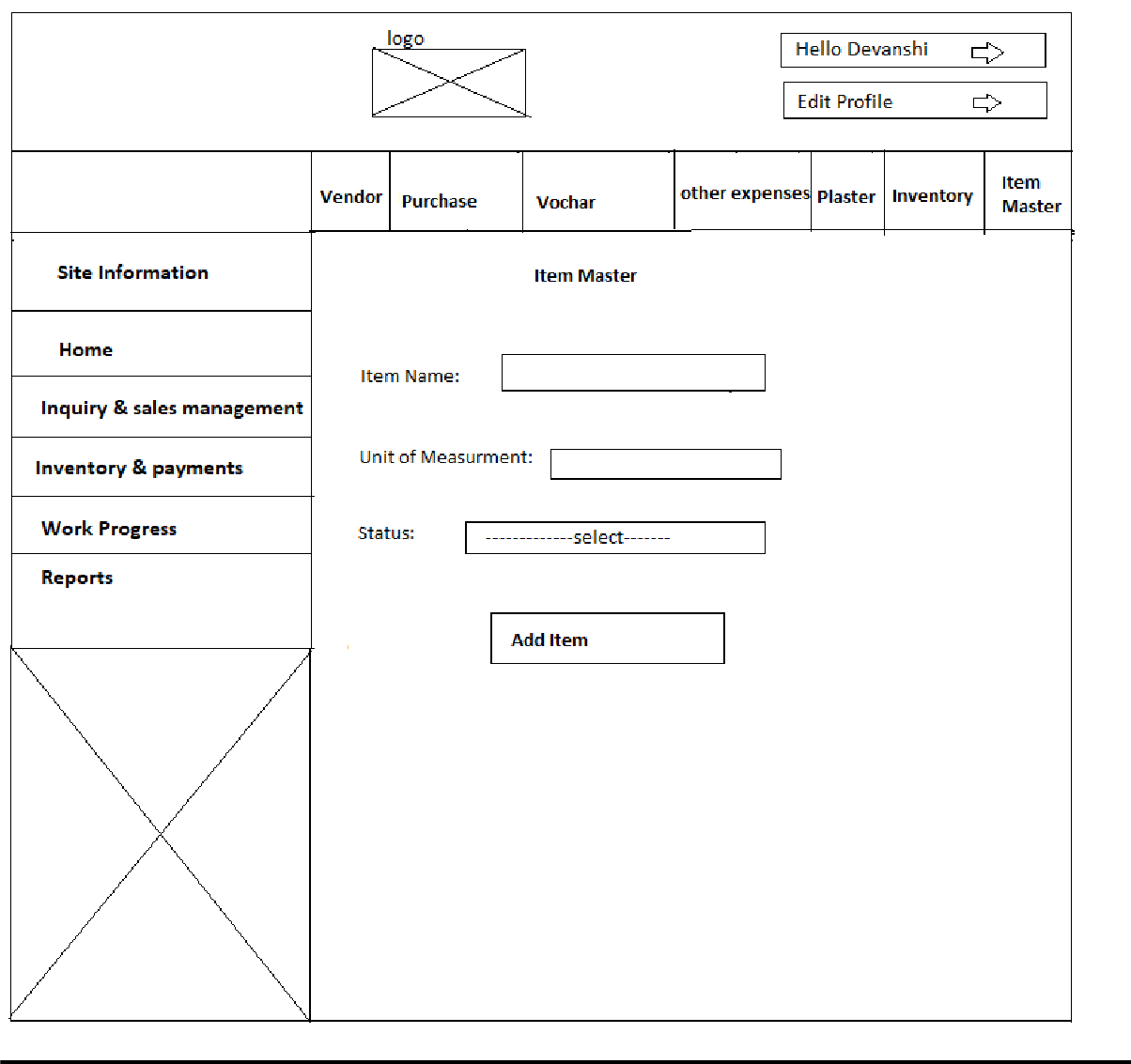
1. Vendor page:

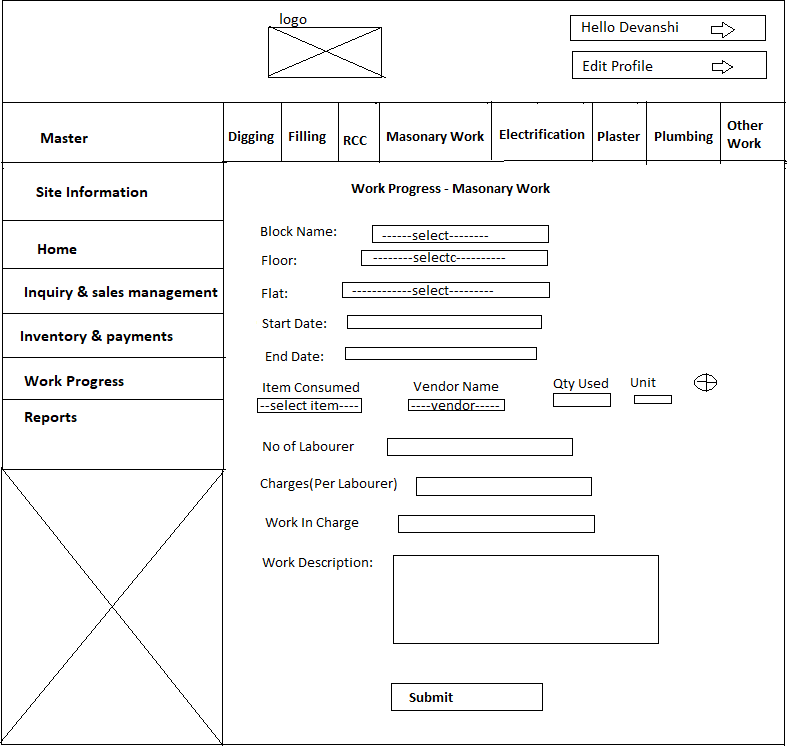


1. Purchase order page:



1. Item Page:



1. Add Work Page: In this page supervisor add work on site and construction related. And also add how many laborer work for particular work and how many charges pay to them.

## System Navigation:

