# **3 Design**

# **Introduction to Design**

One of the important part while developing a project is Design. The second stage of the software development life cycle (SDLC) is design. The system is designed according to the constructed analysis and requirements. Design contains the structure of a system that is to be built which can be represented by various diagrams. Design stage helps a system to be built without having many complication and the errors can be identified easily. The features and function can be described in detail in design specification. Structural models and Behavioral models are mostly covered in design. The design should also include the design of the database system before implementation.

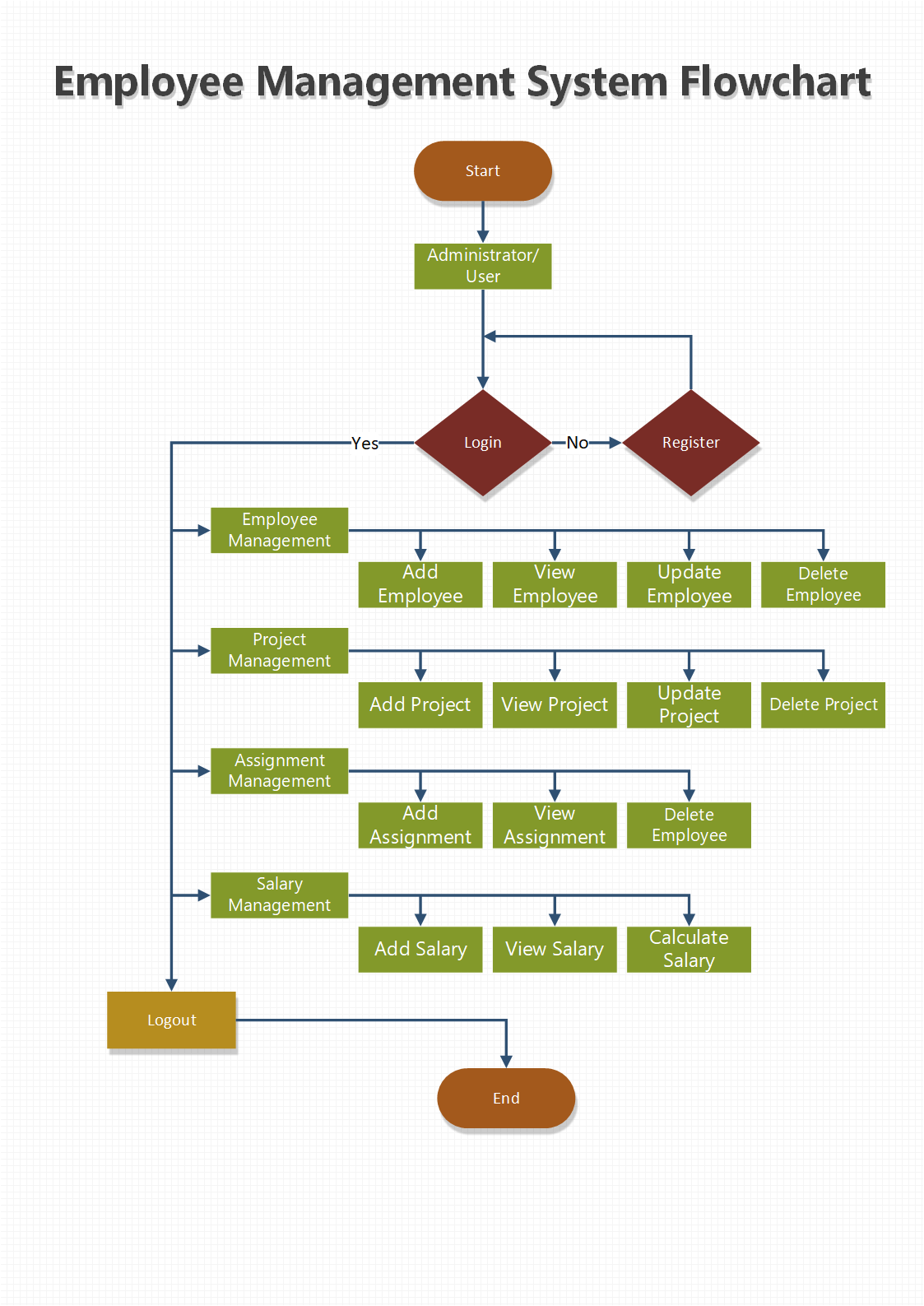
## **3.1 Structural Model**

The whole framework of the system is represented in structural model. It represents all the classes and objects of the system is represented in structural model and also shows their relationship with each other. Class diagram is one of the type of structural model. Different classes, attributes, operations and the relationship between these objects is represented in class diagram.

## **3.1.1 Class Diagram**

## **3.1.2 Flowchart**

A flow chart can be defined as graphical representation of any activities and action that is involved in a project. The main objective of flow chart is to show the overview of the systems and provides reference point for the people dealing with the project.



The flow chart is for the project that is to be developed. Different functions and features like employee management (add, view, update, delete), project management (add, view, update, delete), assignment management (add, view, delete) and salary management (add, view, calculate) can be used by the logged in admin.

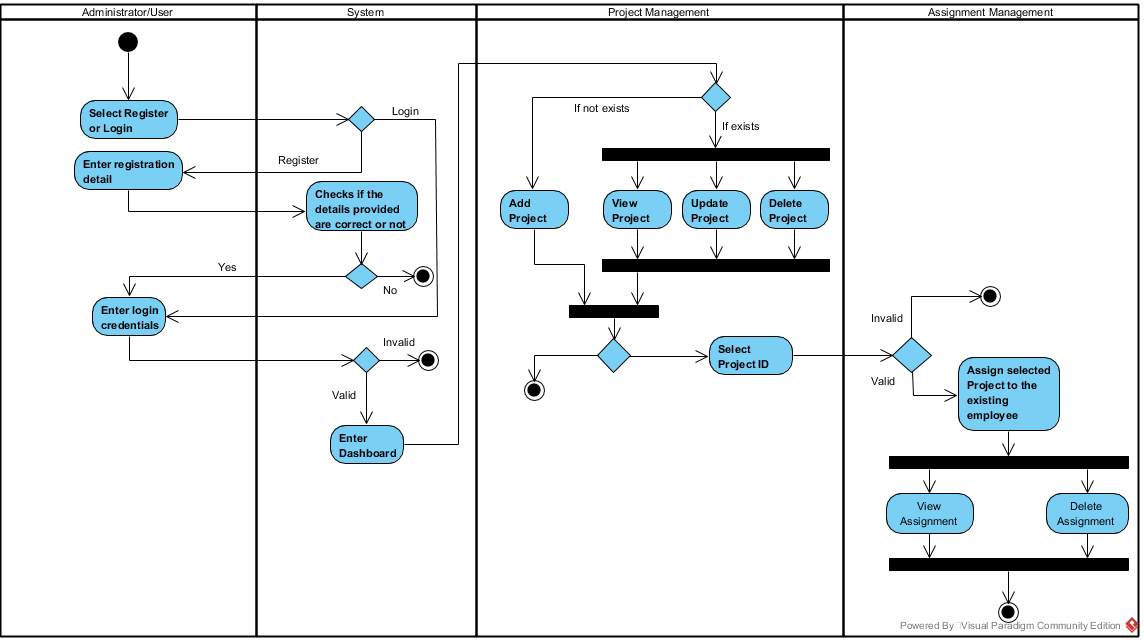
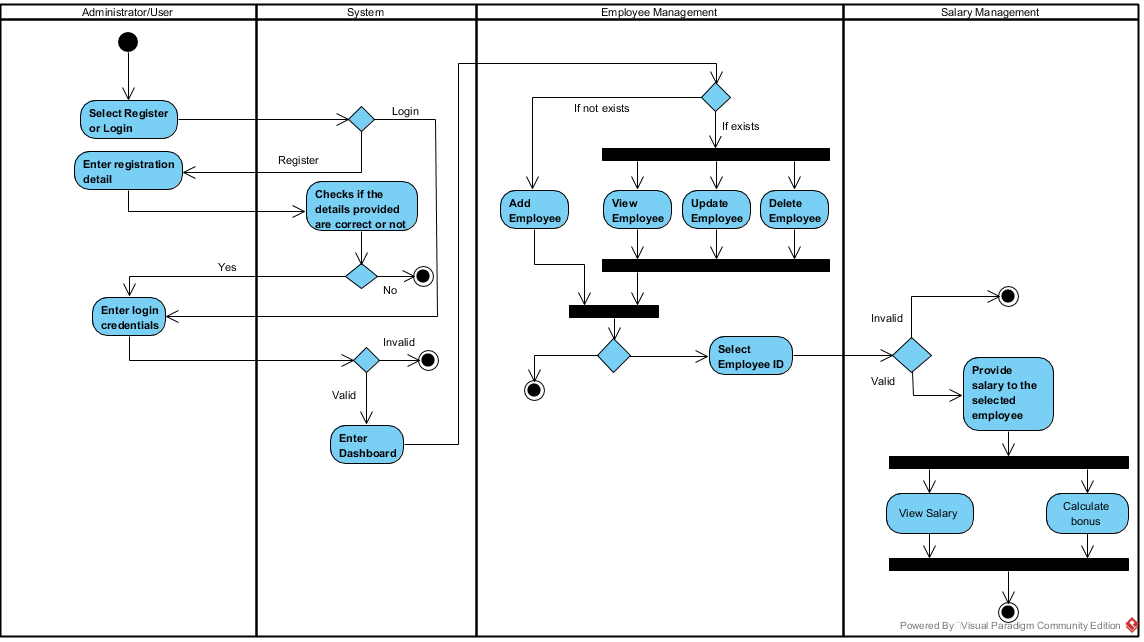
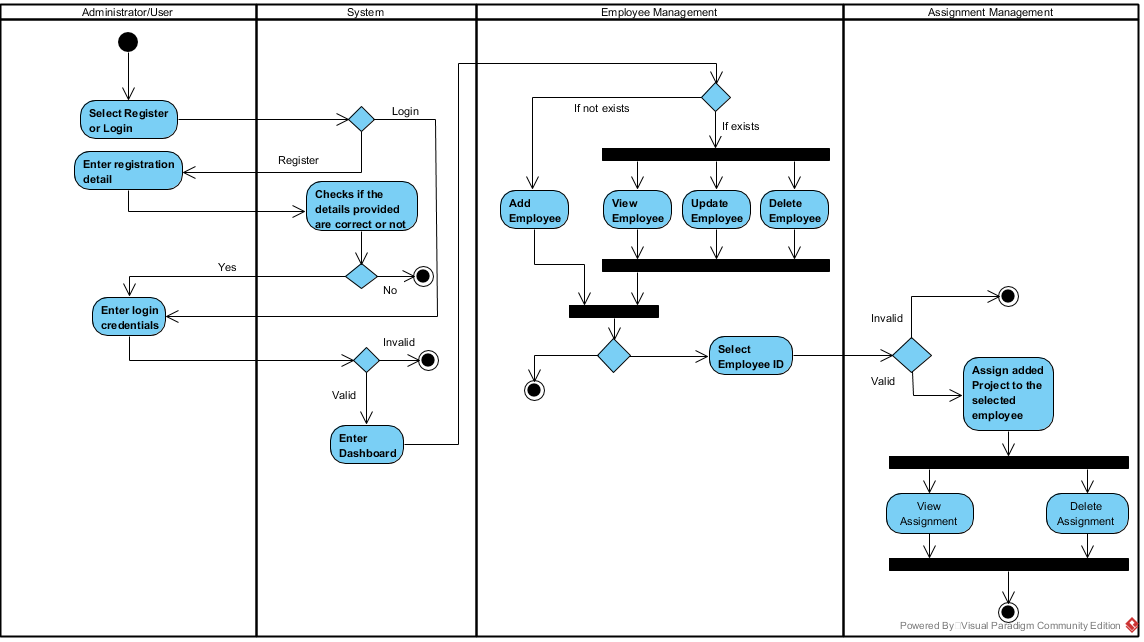
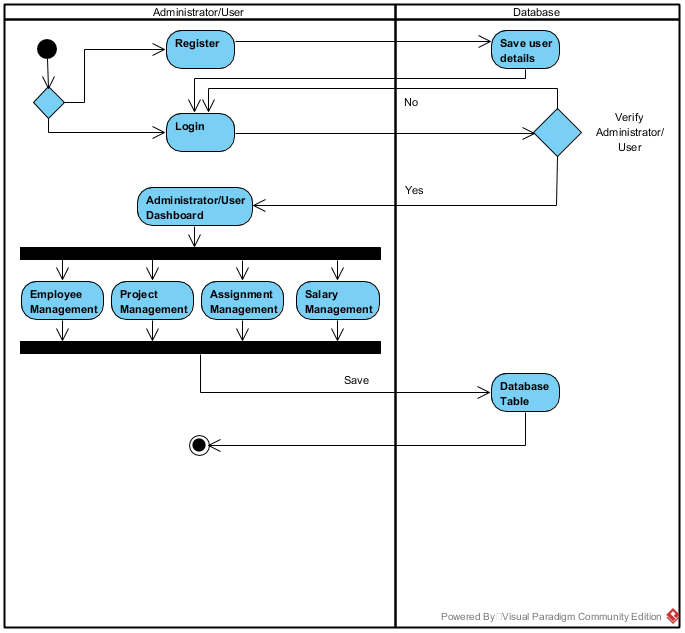
## **3.2 Behavioral Model**

Behavioral model can be defined as any graphical representation that shows the internal structure and features of the system. The dynamic sequence of flow of the system is shown. The two types of behavioral model diagram are sequence diagram and activity diagram.

## **3.2.1 Activity Diagram**

An activity diagram can be defined as graphical representation of the system's flow of control and actions/activities. Its main objective is to describe the activities in detail.

|  |  |  |
| --- | --- | --- |
| S.N. | Symbol | Name |
| 1 |  | Initial Point |
| 2 |  | Activity |
| 3 |  | Decision Symbol |
| 4 |  | Fork node |
| 5 |  | Join node |
| 6 |  | Activity flow |
| 7 |  | End Point |
| 8 |  | Swinlanes |

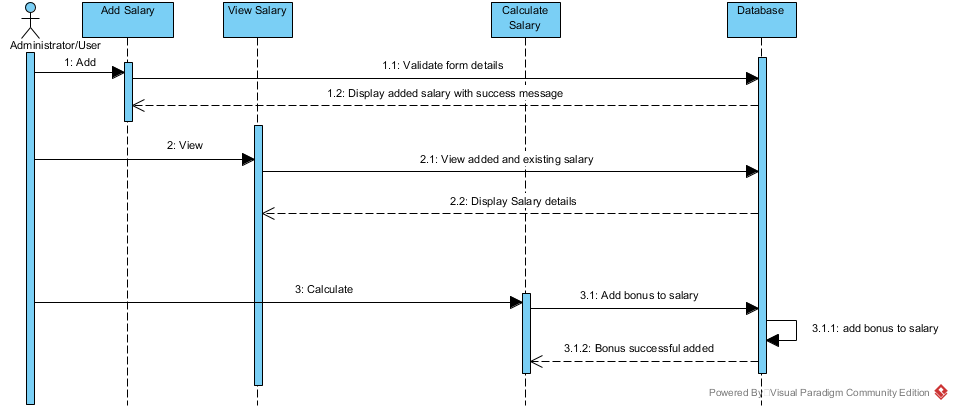
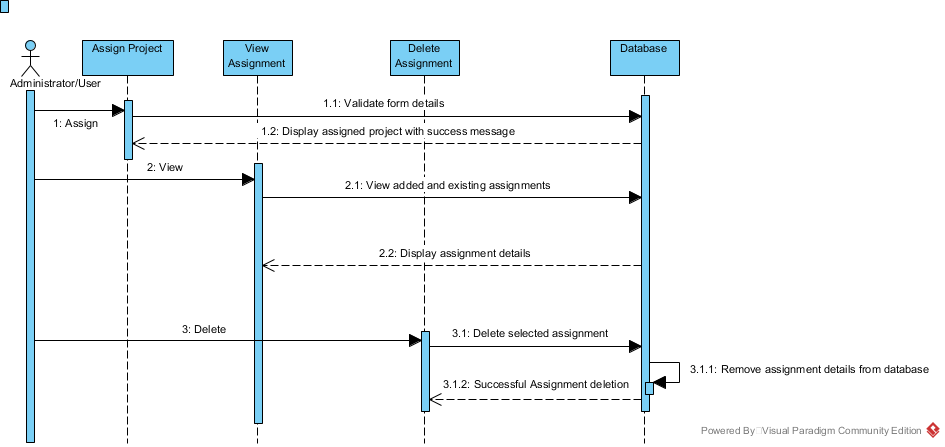
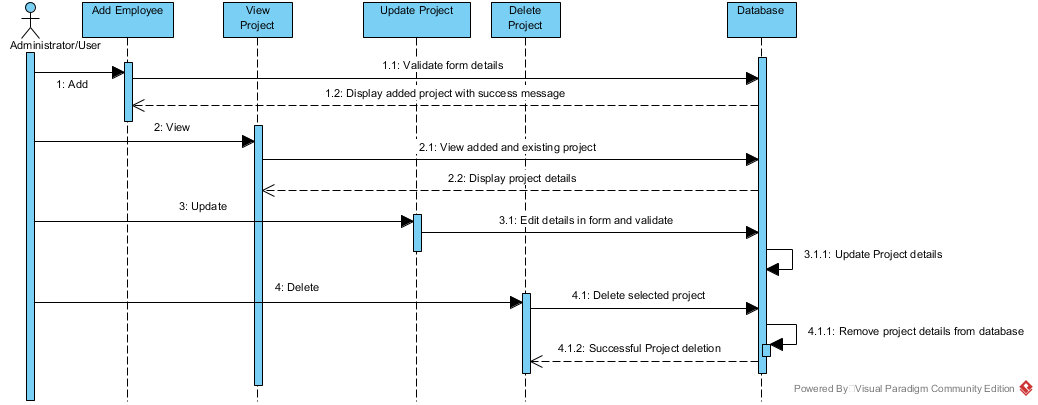
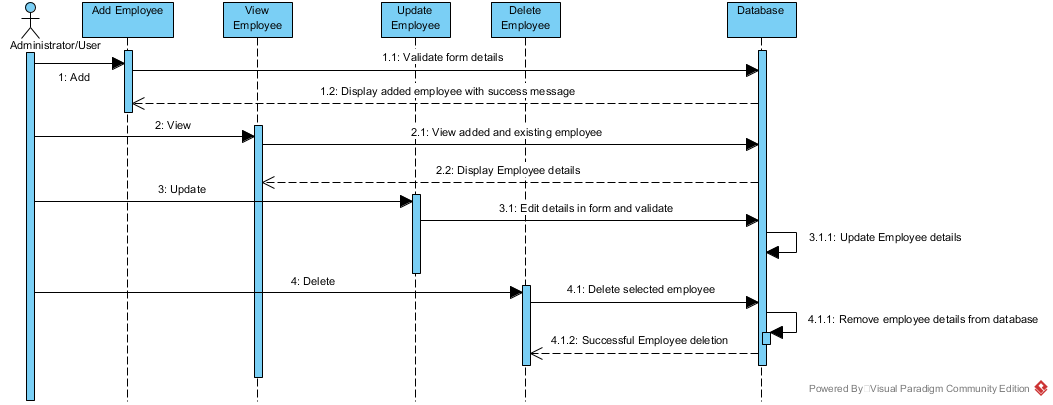
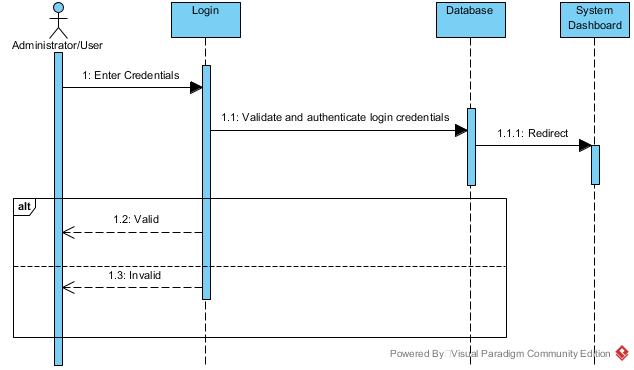


To begin with, the administrator/user logins to the system. If the admin is not registered than he should register in order to login to use the system. When the administrator logins, his credentials are verified to check if the administrator/user is legit. After the login is successful, he/she is redirected to administrator dashboard where four features are available in the system to be used. The administrator/user can use all the features where he/she will be able to add, view, update and delete employee details in employee management. Administrator/user will be able to add, view, update and delete project details in project management. Administrator/user will be able to assign projects to the employees’ available, view and delete assignment in assignment management. Administrator/user will be able to add, view and calculate salary in salary management. These are all the functions and features that are used by a logged in administrator/user.

## **3.2.2 Sequence Diagram**

Sequence diagram can be defined as operations that are carried out and interactions captured between objects in the content of collaboration. It is one of the most important diagram in design.

|  |  |  |
| --- | --- | --- |
| S.N. | Symbol | Name |
| 1 |  | Actor |
| 2 |  | Lifeline |
| 3 |  | Message |
| 4 |  | Self-Message |
| 5 |  | Recursive Message |
| 6 |  | Reply |
| 7 |  | Alt.Combined Fragment |



All the interaction between the administrator/user and the system are represented in the diagrams above. It shows how the features will be used by the administrator.

## **3.3 Data Modeling**

Database design can be defined as process of collecting the processes that will help the design, implementation and development of a database system.

## **3.3.1 Data Dictionary**

Employee Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Datatype** | **Primary Key** | **Foreign Key** | **Nullable** |
| Employee\_id | int(10) | Yes | No | No |
| EmployeeName | varchar(55) | No | No | Yes |
| Age | int(10) | No | No | Yes |
| Gender | varchar(55) | No | No | Yes |
| DateOfBirth | date | No | No | Yes |
| Address | varchar(55) | No | No | Yes |
| Qualification | varchar(55) | No | No | Yes |
| Designation | varchar(55) | No | No | Yes |

Administrator/User Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Datatype** | **Primary Key** | **Foreign Key** | **Nullable** |
| username | varchar(255) | No | No | Yes |
| password | varchar(255) | No | No | Yes |

Project Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Datatype** | **Primary Key** | **Foreign Key** | **Nullable** |
| Project\_id | int(10) | Yes | No | No |
| ProjectName | varchar(55) | No | Yes | No |
| ProjectDesc | varchar(155) | No | No | Yes |
| Startdate | date | No | No | Yes |
| Enddate | date | No | No | Yes |

Assignment Table

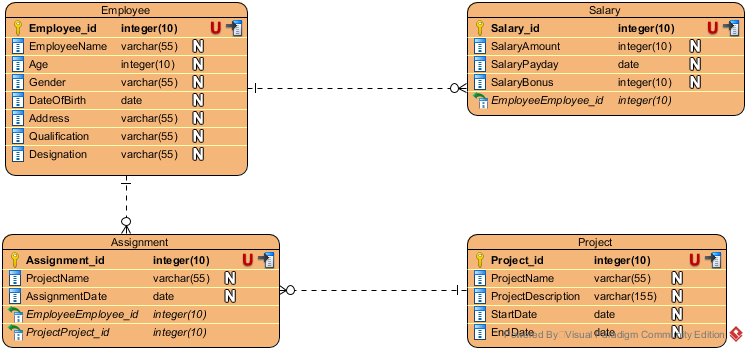
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Datatype** | **Primary Key** | **Foreign Key** | **Nullable** |
| Assignment\_id | int(10) | Yes | No | No |
| Employee\_id | int(10) | No | Yes | No |
| Project\_id | int(10) | No | Yes | No |
| ProjectName | varchar(155) | No | No | Yes |
| AssignmentDate | date | No | No | Yes |

Salary Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Datatype** | **Primary Key** | **Foreign Key** | **Nullable** |
| Salary\_id | int(10) | Yes | No | No |
| SalaryAmount | int(10) | No | No | Yes |
| SalaryPayday | date | No | No | Yes |
| SalaryBonus | int(10) | No | No | Yes |

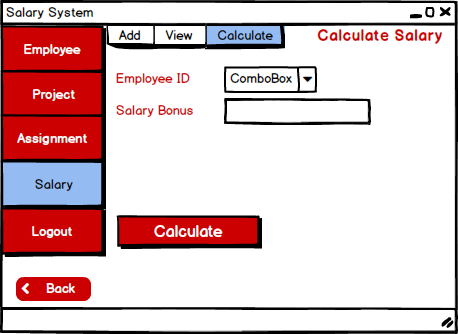
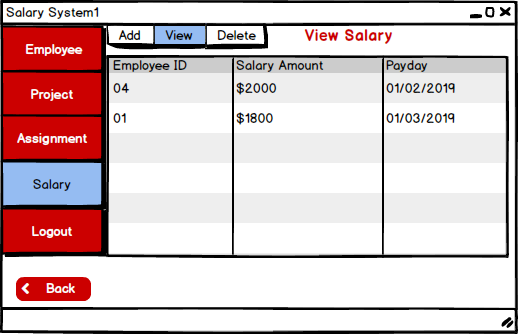
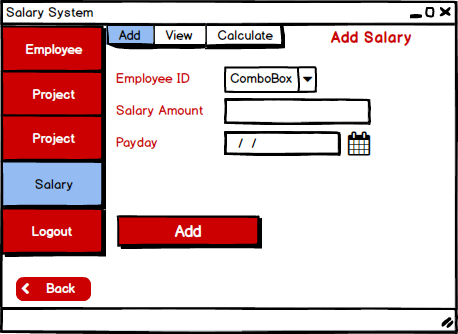
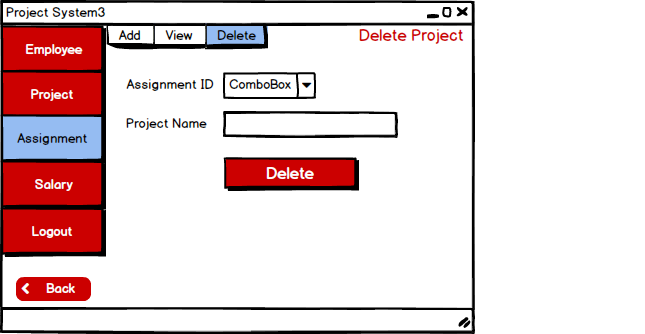
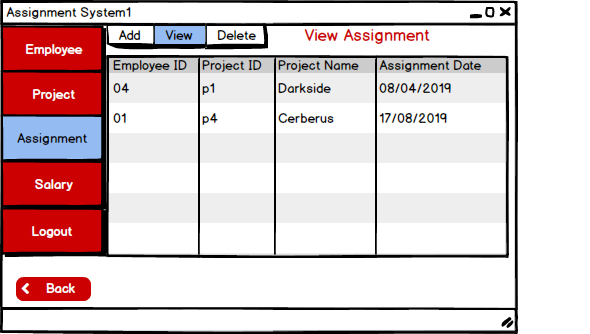
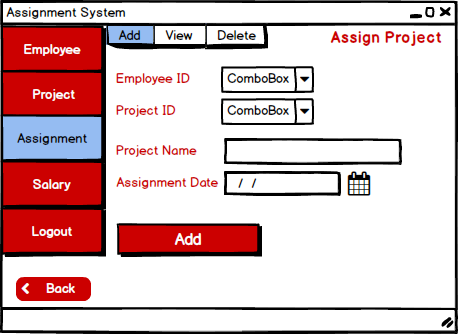
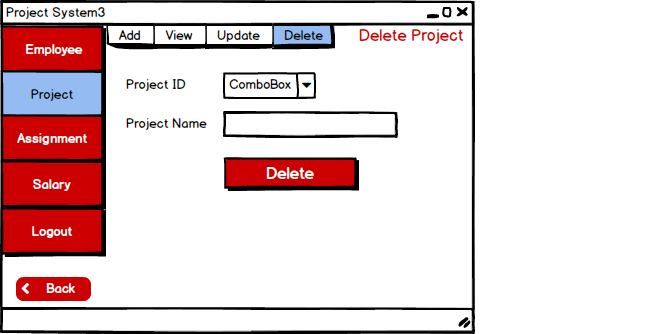
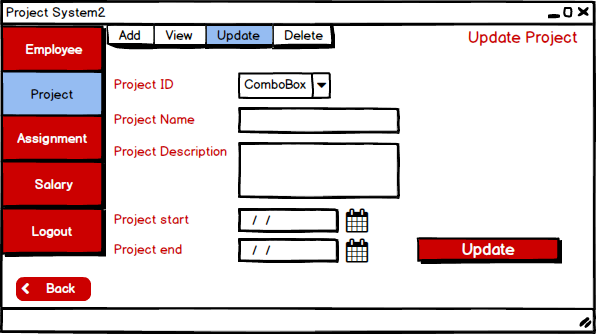
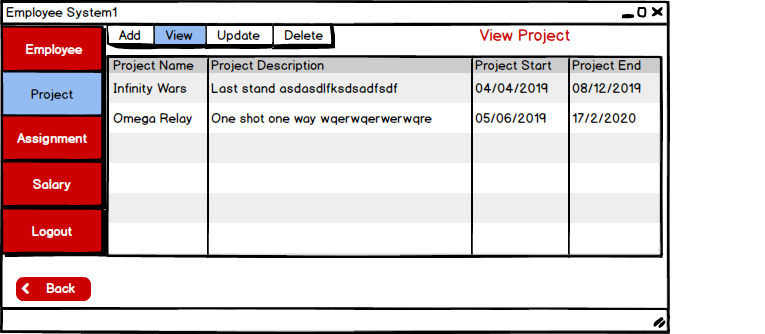
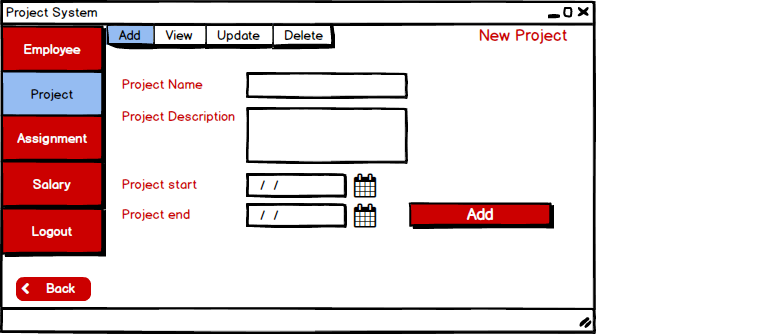
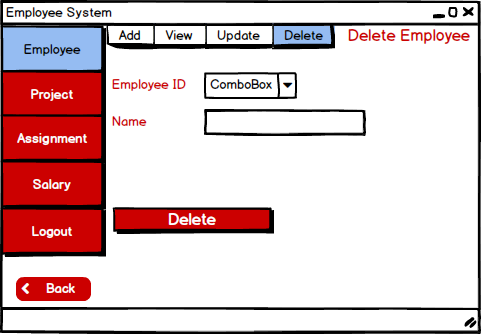
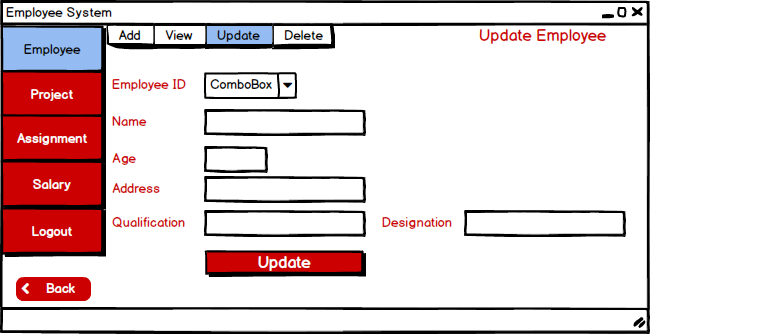
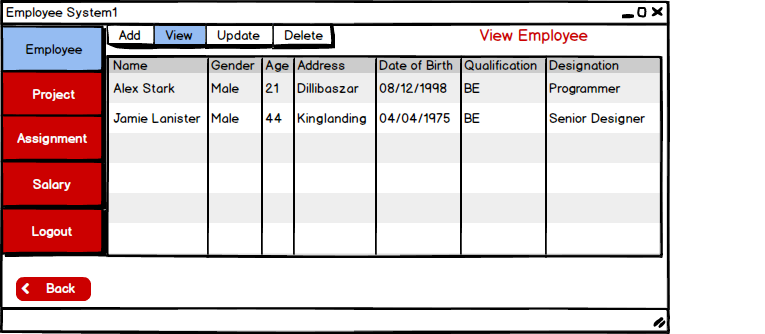
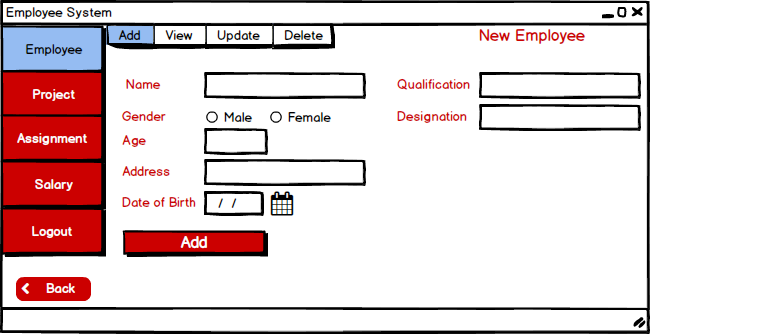
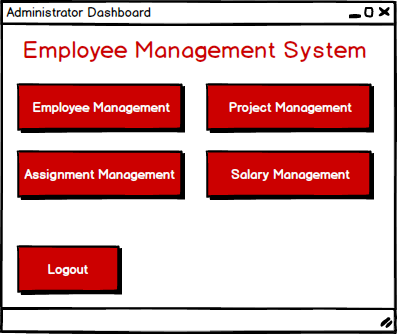
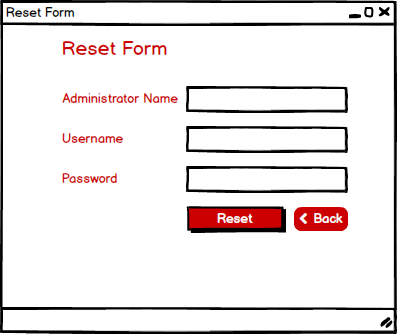
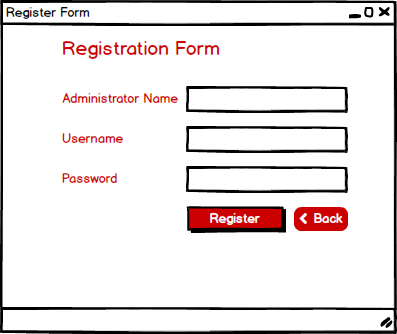
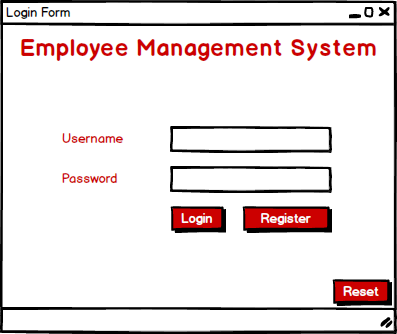
## **3.3.2 E.R. Diagram**

An Entity Relationship Diagram can be defined as a visual representation of a database where tables, entities, entities type and relationships between tables.



The ER diagram shown above is the ER diagram for the database of the system. There are four main tables in this database system. Employee, project, assignment and salary are the tables’ name. The employee tables shows the detailed information of the employees where employee id is the primary key for this table. The project table shows the detailed information of the projects where project id is the primary key for this table. The assignment table shows the detailed information of the projects assigned to the available employee where assignment id is the primary key for this table. The salary table shows the information of the employee's salary where salary id is the primary key for this table.

## **3.4 Prototype Design**



## **3.5 Architectural Model**