**3.DESIGN**

## 3.1 Architecture

## 3.1.1 Software Architecture

The architecture of the system describe the major components relationships(structures), and how they interact with each other. Software architecture and design includes several contributory factors such as business strategy, quality attributes, human dynamics, design and IT environment. Software architecture typically refers to the bigger structures of a software system, and it deals with how multiple software processes cooperate to carry out their tasks. Software design refers to the smaller structures and it deals with the internal design of a single software process.

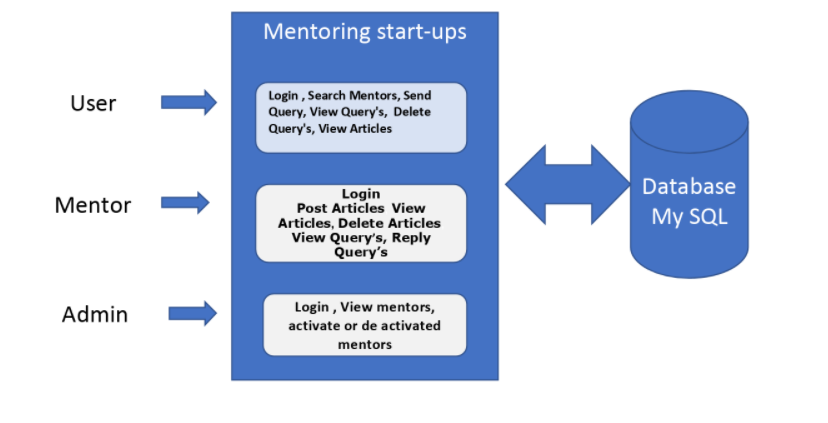


Fig 3.1 Software Architecture

## 3.1.2 Technical Architecture

Technical architecture refers to the structural process of designing and building system’s architecture with focus on the users and sponsors view of the environment. On the client-side user should have a web browser like Internet explorer in which display is done using HTML pages. From the client-side HTTP requests are sent to the server to perform different operations. At the server-side server contains servlets which have control and business logic. If information is to be retrieved, then it is done by executing SQL queries which return the required data from database. Using the data dynamic pages are built and it is sent as HTTP response to the client side.

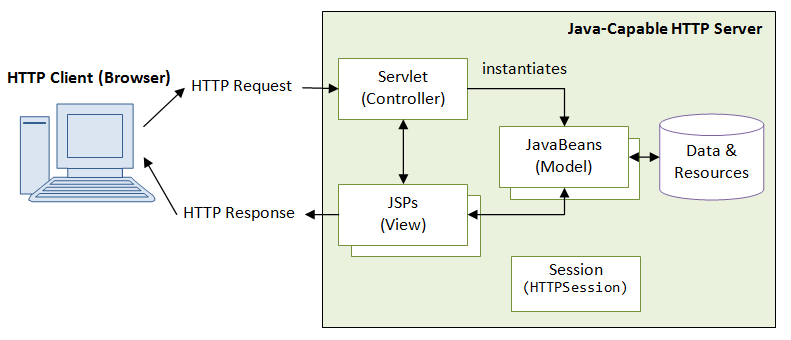


Fig 3.2 Technical Architecture

Design engineering deals with the Unified Modelling Language (UML) which is a standard language for writing software blue prints. The UML is a language for

* + - * Visualizing
      * Specifying
      * Constructing
      * Documenting the artifacts of a software intensive system.

The UML is a language which provides vocabulary and the rules for combining words in that vocabulary for the purpose of communication. A model language whose vocabulary and the rules focus on the conceptual and physical representation of a system. Modelling yields an understanding of a system.

## 3.2 Use case diagram

Use case diagrams are used to gather the requirements of a system including internal and external influences. A use case represents a functionality of a system. So, use case diagrams are used to describe the relationships among the functionalities and their internal/external controllers. These controllers are known as actors.

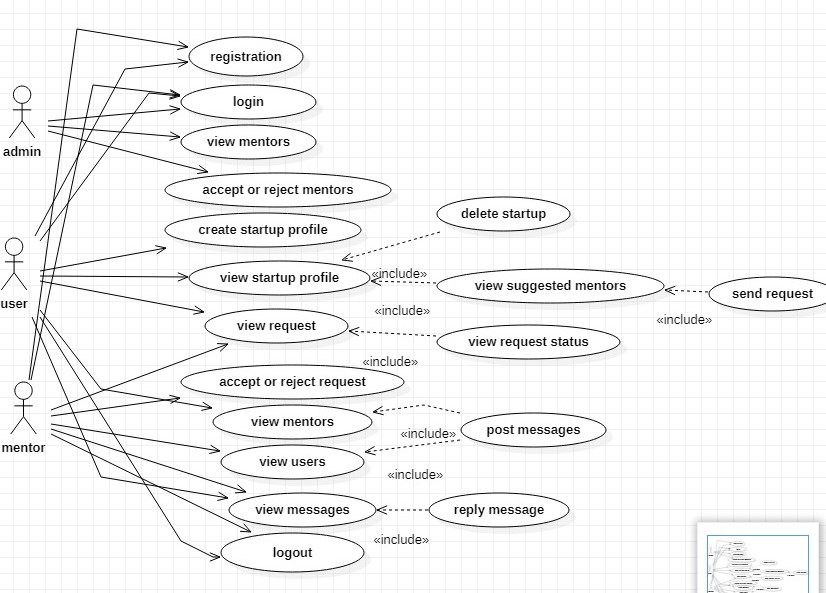
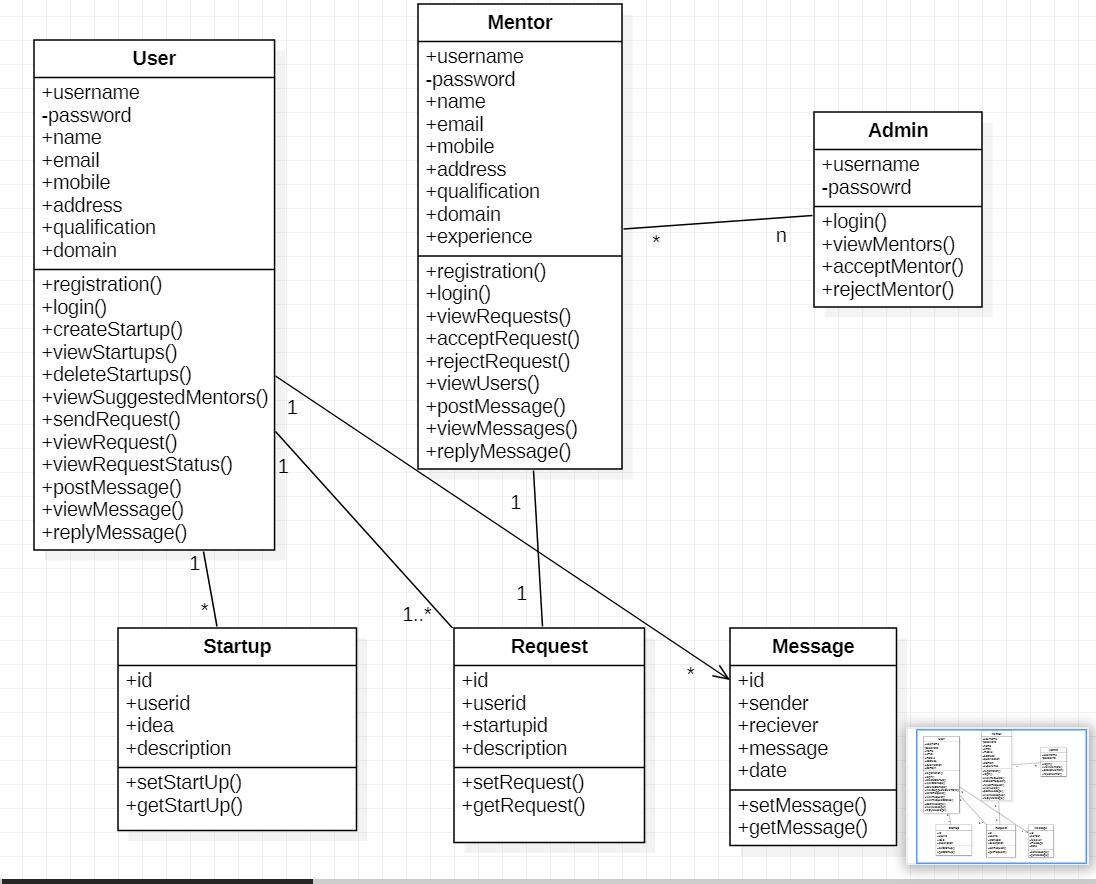


Fig 3.3 Use case Diagram

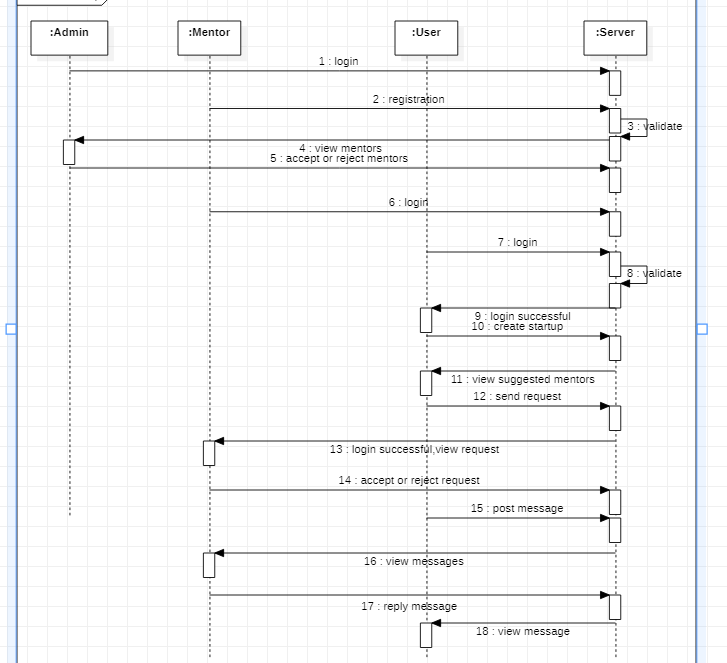
## 3.3 Class diagrams

A class diagram in the Unified Modelling Language is a type of a static structure diagram that describes the structure of a system showing the system’s classes, their attributes, operations (or methods), and the relationship among the classes.



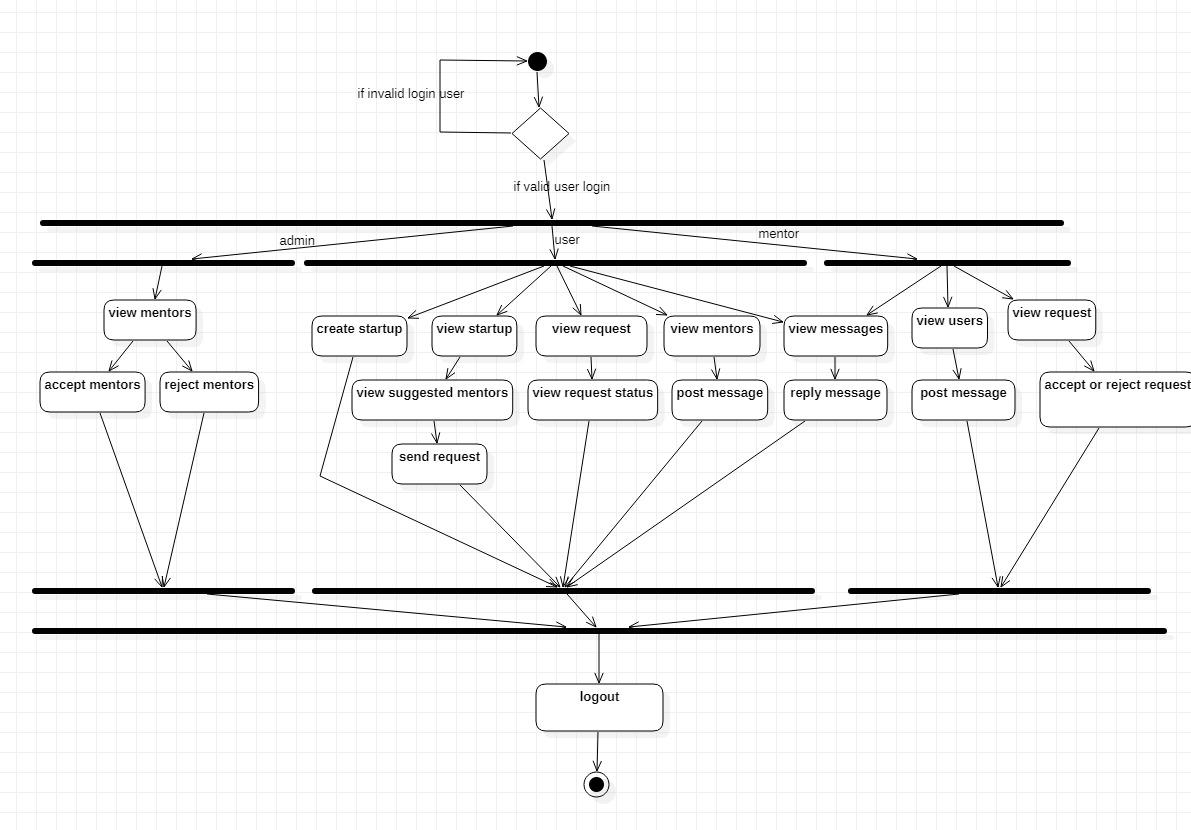
## 3.4 Sequence diagrams

A sequence is Unified Modelling Language (UML) is a kind of interaction diagram that show how processes operate with another and in what order. It is a construct of a Message Sequence Chart.

****

## 3.5 Activity diagram:

Activity diagrams are graphical representation of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams can be used to describe the business and operational step- by-step workflows of components in a system. An activity diagram shows the overall flows of control.



## 3.6 Database Design:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | type | key | null | default | Extra |
| Username | varchar | Pri |  |  |  |
| Password | varchar |  |  |  |  |
| Name | varchar |  |  |  |  |
| Email | varchar |  |  |  |  |
| Mobile | bigint |  |  |  |  |
| Address | varchar |  |  |  |  |
| Experience | varchar |  |  |  |  |
| qualification | varchar |  |  |  |  |
| domain | varchar |  |  |  |  |

## Table 4.5.1 DATABASE TABLE FOR MENTORS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | type | key | null | default | Extra |
| Id | int |  |  |  |  |
| Sender | Varchar |  |  |  |  |
| Receiver | varchar |  |  |  |  |
| Message | varchar |  |  |  |  |
| Date | date |  |  |  |  |

Table 4.5.2 DATABASE TABLE FOR MESSAGES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | type | key | N Null | default | Extra |
| Id | int |  |  |  |  |
| User id | varchar |  |  |  |  |
| S Startup id | Int |  |  |  |  |
| Description | varchar |  |  |  |  |

Table 4.5.3 DATABASE TABLE FOR REQUEST

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | type | key | null | default | Extra |
| Id | int |  |  |  |  |
| User id | varchar |  |  |  |  |
| Startup id | Int |  |  |  |  |
| Description | varchar |  |  |  |  |

Table 4.5.4 DATABASE TABLE FOR STARTUP

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | type | key | null | default | Extra |
| Username | varchar |  |  |  |  |
| Password | varchar |  |  |  |  |
| Name | varchar |  |  |  |  |
| Email | varchar |  |  |  |  |
| Mobile | Big int |  |  |  |  |
| Address | varchar |  |  |  |  |
| Experience | varchar |  |  |  |  |
| qualification | varchar |  |  |  |  |
| domain | varchar |  |  |  |  |

Table 4.5.5 DATABASE TABLE FOR USER