|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Software Requirement Specifications**  **BUILD EASE**  **Version: [1.5]**   |  |  | | --- | --- | | Project Code | F23-1139B | | Supervisor | Ms.Sobia Iftihkar | | Co Supervisor |  | |  |  | | Project Team | Hussain Tahir (20k-0185)  Ayaz Hassan (20K-1044)  Syed Arsalan (20k-1718) | | Submission Date | -December-2023 | |

[Instructions]

* No section of template should be deleted. You can write ‘Not applicable’ if a section is not applicable to your project. But all sections must exist in the final document.
* All comments/examples mentioned in square brackets ([]) are in the template for explanation purposes and must be replaced / removed in final document.
* This’ Instruction’ section should also be removed in final document.
* MS-Word Reviewing feature must be used to get the document reviewed by supervisors or co-supervisors.

Document History

|  |  |  |  |
| --- | --- | --- | --- |
| ***Version*** | ***Name of Person*** | ***Date*** | ***Description of change*** |
| 1.0 | Hussain Tahir | 28 Nov 23 | Started Working on the SRS |
| 1.1 | Syed Arsalan | 29 Dec 23 | Added Few Details |
| 1.2 | Syed Arsalan | 1 Dec 23 | Added Functional Hierarchy |
| 1.3 | Hussain Tahir | 3 Dec 23 | Added Scope, Functional and Non Functional Requirements |
| 1.4 | Ayaz Hassan | 4 Dec 23 | Added Use Cases |
| 1.5 | Ayaz Hassan | 5 Dec 23 | Reviewed Document |

Distribution List

|  |  |  |
| --- | --- | --- |
| ***Name*** | ***Role*** | |
| Ms. Sobia Iftikhar | | Supervisor |
|  | | Co- Supervisor |
|  | |  |

Document Sign-Off

|  |  |  |
| --- | --- | --- |
| ***Version*** | ***Sign-off Authority*** | ***Sign-off Date*** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Table of Contents**

[1.](#_30j0zll)Introduction 7

[*1.1.*](#_3znysh7) *Purpose of Document 7*

[*1.2.*](#_2et92p0) *Intended Audience 7  
1.3 Abbreviations ………………………………………………………………………………………...7*

[*1.4.*](#_tyjcwt) *Document Convention 7*

[2.](#_3dy6vkm)Overall System Description 8

[*2.1.*](#_1t3h5sf) *Project Background 8*

[*2.2.*](#_4d34og8) *Project Scope 8*

[*2.3.*](#_2s8eyo1) *Not In Scope 8*

[*2.4.*](#_17dp8vu) *Project Objectives 8*

[*2.5.*](#_3rdcrjn) *Stakeholders 8*

[*2.6.*](#_26in1rg) *Operating Environment 8*

[*2.7.*](#_35nkun2) *System Constraints 8*

[*2.8.*](#_1ksv4uv) *Assumptions & Dependencies 8*

[3.](#_2jxsxqh)External Interface Requirements 9

[*3.1.*](#_z337ya) *Hardware Interfaces 9*

[*3.2.*](#_3j2qqm3) *Software Interfaces 9*

[*3.3.*](#_1y810tw) *Communications Interfaces 9*

[4.](#_4i7ojhp)Functional Requirements 10

[4.1.](#_2xcytpi)Functional Hierarchy 10

[*4.2.*](#_1ci93xb) *Use Cases 10*

[*4.2.1.*](#_3whwml4) *[Title of use case] 10*

[5.](#_2bn6wsx)Non-functional Requirements 11

[*5.1.*](#_qsh70q) *Performance Requirements 11*

[*5.2.*](#_3as4poj) *Safety Requirements 11*

[*5.3.*](#_1pxezwc) *Security Requirements 11*

[*5.4.*](#_49x2ik5) *User Documentation 11*

[6.](#_147n2zr)References 12

[7.](#_3o7alnk)Appendices 13

1. **Introduction**

* 1. **Purpose of Document**

The purpose of this document is to specify all the requirements of our project in a detailed format, including its goals, scope and services. In a detailed format all the requirements of the stakeholders would be mentioned, such as system functionalities and its implementations.

* 1. **Intended Audience**

The targeted audience for this document will be the private construction contractors, the clients requiring construction services and the administrators of the platform.

**1.3 Abbreviations**

Not Applicable.

* 1. **Document Convention**

The font used for the headings and the body is Arial. The headings have a font size of 16 while the subheadings have a font size of 12. Following the paragraphs, it has a font size of 10. This format is maintained throughout the document.

1. **Overall System Description**
   1. **Project Background**

There is currently an enormous amount of financial resources and human resources required for the construction of the buildings on industrial scale or on an individual scale. However people that need to build houses for themselves tend to hire private contractors and people spend their life savings or huge amount. In the construction business this a big problem to select the right contractor as there was previously no platform available for the clients and the service providers to interact in addition to that there was no comparative data available for the clients to help them select the right contractor for their construction projects. The construction service provider also had a limited customer base. This project will revolutionize how the construction business in Karachi operates. This project aims to build an interactive platform for the use of both clients and service providers that will help them to make quick and efficient decisions that will help them financially by providing real time data in helping in the decision-making process.

.

* 1. **Project Scope**

Our project build ease aims to enable the clients and the construction service providers to connect to a digitally available platform that allows both the client and the service provider to reach mutual contractual conditions and decide for themselves to choose the desired individuals for their projects. The client will be able to register in the system by just entering the required information such as name, phone number, address, the area in which they are looking for the desired contractor for work. The contractor will using the same credentials to register in the system except the contractor will be able to choose from the type of construction such as house or flat, ratings will help the client to choose the right contractor that will help them and the percentage of project completion. A project cost calculator will estimate the cost of each project.

* 1. **Not In Scope**

This project will not enable you sign a contract based between two individuals. This project will not allow to make any kind of payments or include any type of paywalls. This will not build a virtual project or simulate a building a project by itself. This project will not be responsible for any delays or monitor any legal or illegal activities.

* 1. **Project Objectives**

The objective of this project is to provide with an interactive digital platform that will help enable the clients that are looking for construction contractor for building their homes on an individual level. We want to establish an online platform that is available to all users that want to provide or utilize the construction services that are being offered by the platform. Our objective is to make effective communication possible between the clients and the service providers. We want to help the clients make successful decisions meanwhile selecting the contractor and avoid undesirable contractors, we want make sure the users find the right service provider or client with minimal effort.

* 1. **Stakeholders**

Following are the stakeholder of this application:

* Service provider
* Developers
* Construction Workers.
* Legal Authorities.
  1. **Operating Environment**

The project requires the users to only have a smartphone, computer or laptop without any constraint on the operating system as the application will have the ability to run on Android, Windows or iOS devices and any web browser. However, users must have access to the internet as the application will not run without a stable WIFI connection.

* 1. **System Constraints**

The system has been developed by the project members of the university due to lack of the whole software team system is subject to some of the constraints. The system will not any payment information. The system will not enable users to pay for a service. The system will enable the text conversation between the clients and the service providers. The system has only the web app available for this time.

* 1. **Assumptions & Dependencies**

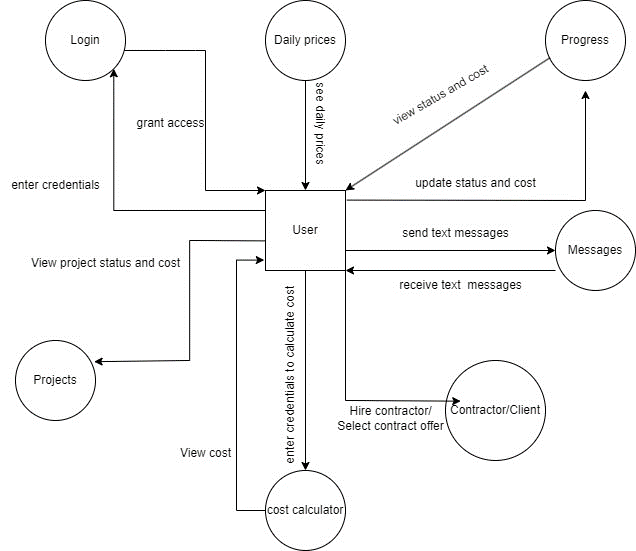
We assume that the users off our website:

* Have a stable internet connection
* Understand English and Urdu
* Users for this website are local contractor

**System Dependencies**:

* Our system requires a hosting service that is able to connect to the servers.
* System depends on a stable internet connection.
* Requires an active database connection to save the changes made.

1. **External Interface Requirements**



* 1. **Hardware Interfaces**

For this web application all you need is a computer or any other device with an internet connection to access it

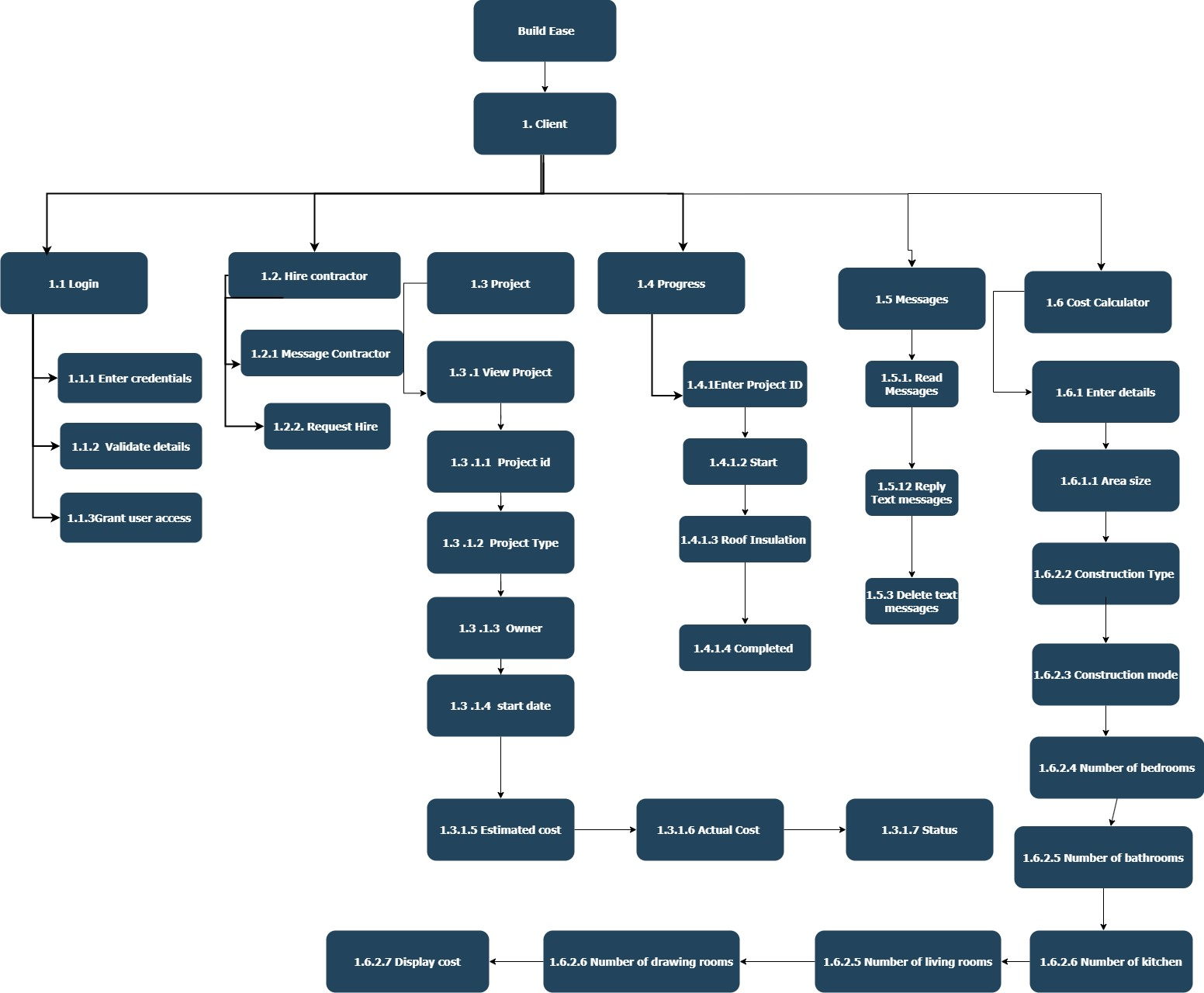
* 1. **Software Interfaces**

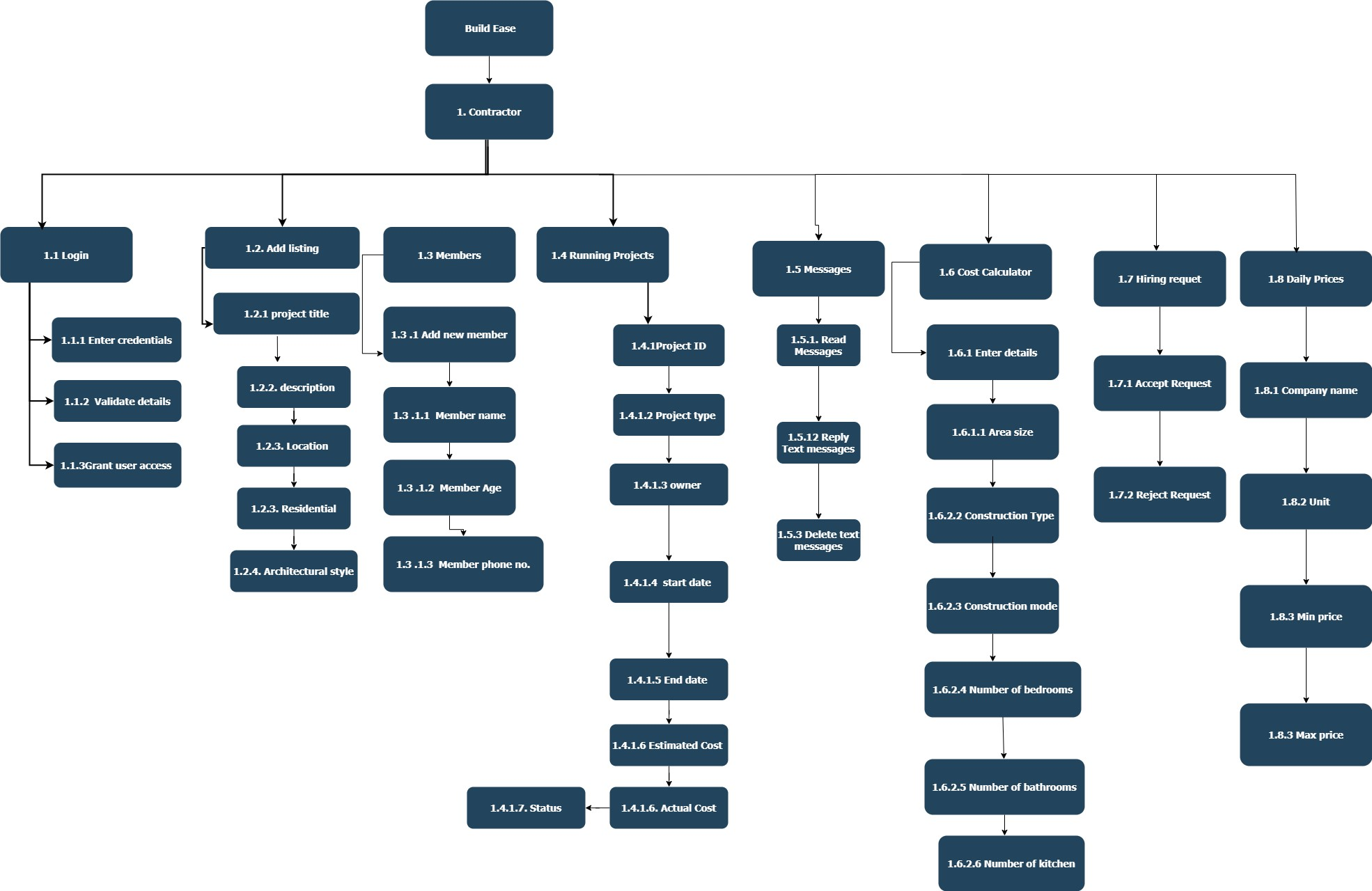
The system will use MongoDB ATLAS for storing the system's database. As it is a web application it can run on any operating system. The system will be implemented using React, html, CSS and we will use JavaScript. The recommender system will use python language and its required libraries. The data base used will be relational to show complex relationships.

* 1. **Communications Interfaces**

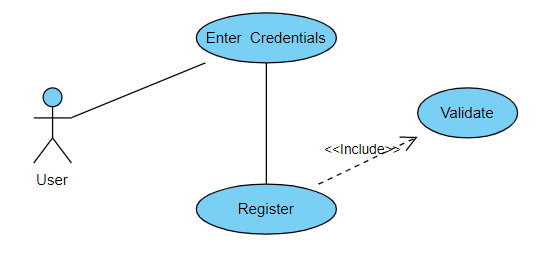
The application will communicate with the server using Express APIs. The front end and user communication will be handled by React. Mongo Db’s REST API will communicate between database and external applications. React will communicate

1. **Functional Requirements**
   1. **Functional Hierarchy**



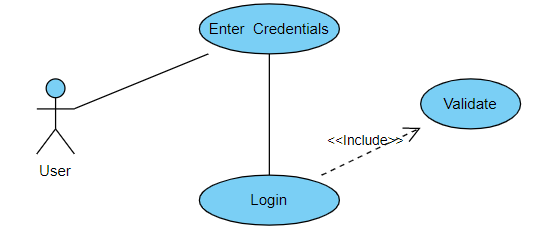


* 1. **Use Cases**
     1. **Registration**



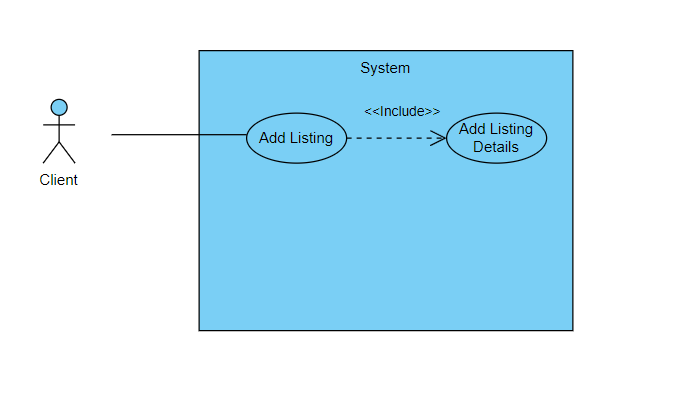
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4.2.1 : Registration** | | | | |
| **Use case Id:** | | *4.2.1* | | |
| **Actors:**  *Clients and Contractors* | | | | |
| **Feature:**  *Sign-Up* | | | | |
| **Pre-condition:** | | *-* | | |
| **Scenarios:** Before using the application, users like, clients and contractors have to register on it using their university allotted email address or roll number. As soon as they register, the account gets verified and ready to be logged in for next time. | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | *First of all the user must click on the sign-up to enter their details and register* | | | *Redirects the user to the registration form.* |
| **2.** | Enter email | | | Email or roll number’s format will be checked. |
| **3.** | Enter password | | | Password format will be checked |
| **4.** | Enter Name | | | Checks if the field is not empty. |
| **5.** | Click the button to register | | | Data will be verified and the user will be redirected to a specific interface upon successful creation. |
| **Alternate Scenarios:** | | | | |
| **1a:** Password hash will be created which is then saved in the database.    **2a:** Unique IDs will be generated by the system and will be assigned to each user. | | | | |
| **Post Conditions:** Successful creation of an account which can be used to login to the system at a later date. | | | | |
| **Step#** | **Description** | | | |
|  | The user will have a system generated ID which will act as the primary key in the database. | | | |
|  | A record corresponding to the user's credentials will be entered into the database. | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | *-* | |

* + 1. **Log-In**

****

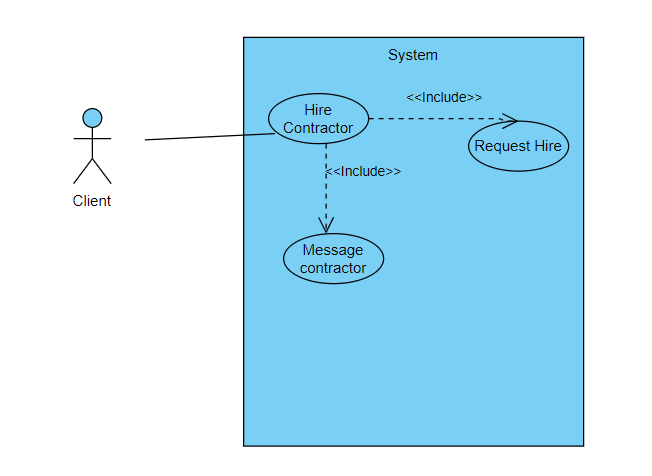
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4.2.2 : Log-In** | | | | |
| **Use case Id:** | | *4.2.2* | | |
| **Actors:**  *Client, Contractor and Admin* | | | | |
| **Feature:**  *Log-In* | | | | |
| **Pre-condition:** | | *The user needs to be registered before to Log-In.* | | |
| **Scenarios:** Considering the user has registered already, he/she logs in to use the web application and perform further actions. | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | *First of all the user must click on the log-in to enter their details and procees* | | | *Redirects the user to the log-in page.* |
| **2.** | Enter email address or roll number | | | Check if the email is correct and matches. |
| **3.** | Enter password | | | Check if the password matches. |
| **4.** | Click on the button to log-in | | | Authenticate and the user and proceed with the login process. |
| **Alternate Scenarios:** | | | | |
| **1a:** Email will be first checked if it exists in the database.  **2a:** Password entered and the password stored in the database of the respective roll number will be checked if they match. | | | | |
| **Post Conditions:** Successful authentication will lead the user to log into the system and use its services. | | | | |
| **Step#** | **Description** | | | |
|  | Access to use the system will be granted to the user according to its type. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | *4.2.1* | |

* + 1. **Add Listing**

****

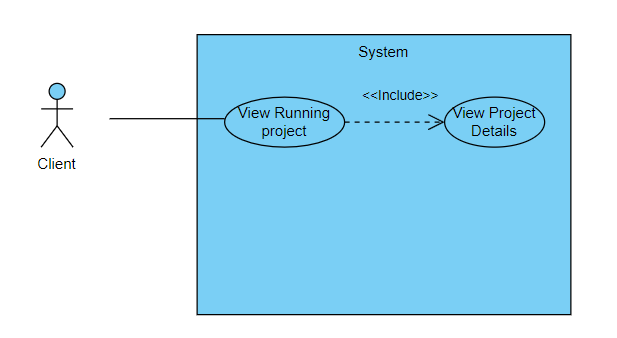
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4.2.3 : Add Listing** | | | | |
| **Use case Id:** | | *4.2.3* | | |
| **Actors:**  *Users – Client* | | | | |
| **Feature:**  *Add Listing* | | | | |
| **Pre-condition:** | | *The user type needs to be client and they need to be logged in.* | | |
| **Scenarios:** Admin logs into the system and adds course in the course list. | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | *User clicks on the add new listing button,* | | | *Redirects the user to the add list course form.* |
| **2.** | Enter listing Details | | | Takes all the details as a response. |
| **3.** | User clicks on the button to the submit the details | | | Validates the listing and uploads it onto the database. |
| **Alternate Scenarios:** | | | | |
| **Post Conditions:** After successful validation the new Listing is added to the list. | | | | |
| **Step#** | **Description** | | | |
|  | The listing is available for the Contractors to view it and its details. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | *4.2.2* | |

* + 1. **Hire Contractor**

****

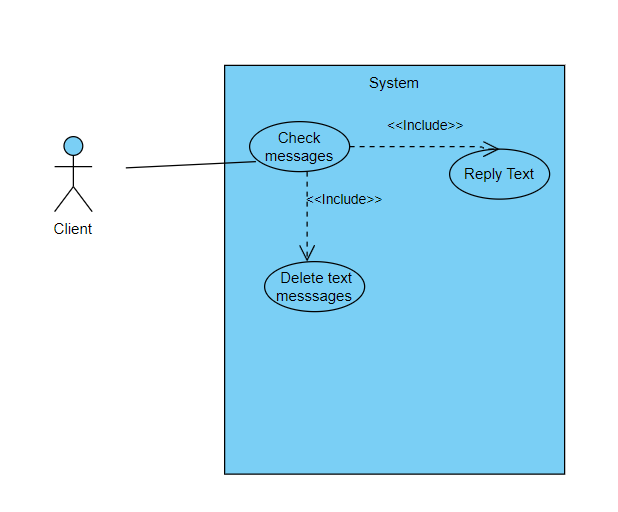
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4.2.4 :Hire Contractor** | | | | |
| **Use case Id:** | | *4.2.4* | | |
| **Actors:**  *Users – Client* | | | | |
| **Feature:** *Hire Contractor* | | | | |
| **Pre-condition:** | | *The user type needs to be Client and they need to be logged in.*  *And the Contactor profile must be selected.* | | |
| **Scenarios:** Client logs into the system and hires the contractor. | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | *User clicks contactor card,* | | | *Redirects the user to the hire contractor form.* |
| **2.** | Message or Request to hire | | | Takes sends the hiring request or a text message. |
| **3.** | User clicks on the button to the submit the details | | | Sends the text message or the request |
| **Alternate Scenarios:** | | | | |
| **1a:** When editing course, teachers names would be specified too for proper listing.  **2a:** Proper details need to be entered for valid display of details. | | | | |
| **Post Conditions:** After successful execution a hiring request is sent or the text message option is opened | | | | |
| **Step#** | **Description** | | | |
|  | The contracter is being hired or contacted. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | |  | |

* + 1. **View Running Project**

****

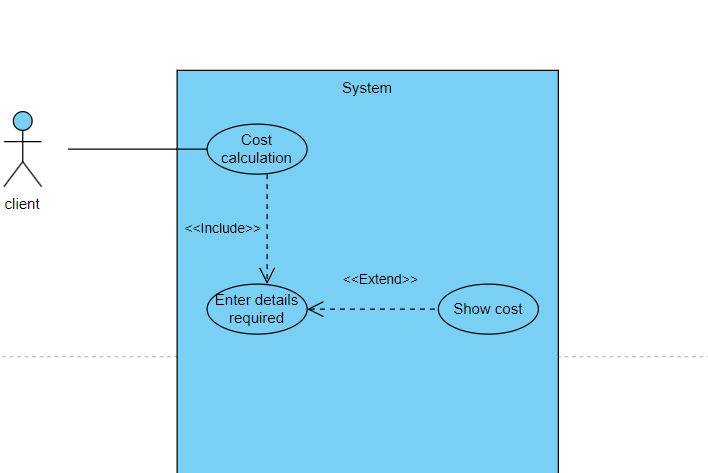
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4.2.5 : View Running Project** | | | | |
| **Use case Id:** | | *4.2.5* | | |
| **Actors:**  *Users – Client* | | | | |
| **Feature:**  *View Running Project* | | | | |
| **Pre-condition:** | | *The user type needs to be a client and they need to be logged in.*  *Project must be selected by the user.* | | |
| **Scenarios:** Admin logs into the system and deletes course from the course list. | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | *User clicks on the Running Project* | | | *Redirects the user to the project details window* |
| **2.** |  | | |  |
| **Alternate Scenarios: There are no projects so there is nothing to be shown.** | | | | |
| **1a:** User must be doing a project with a chosen contractor  **2a:** - | | | | |
| **Post Conditions:** After successful completion the details are shown to the user. | | | | |
| **Step#** | **Description** | | | |
|  | The details like owner, start date , end date of the project are shown to client. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | *4.2.2 and 4.2.1* | |

* + 1. **Check Messages**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4.2.6 : Check messages** | | | | |
| **Use case Id:** | | *4.2.6* | | |
| **Actors:**  *Users – Client/contactor* | | | | |
| **Feature:**  *Check Texts* | | | | |
| **Pre-condition:** | | *The user type needs to be a client and they need to be logged in.* | | |
| **Scenarios:** Client wants to view the messages. | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | *User clicks on the text messages button.* | | | *Redirects the user to the inbox of the text messages.* |
| **2.** | Clicks on reply section | | | Opens the text box for typing message. |
| **3.** | User clicks on delete button. | | | The Text disappears as it is deleted. |
| **Alternate Scenarios:** | | | | |
| **1a:** The client has no text messages in their inbox.  **2a: -** | | | | |
| **Post Conditions:** After successful completion the user can choose what to do with the messages. | | | | |
| **Step#** | **Description** | | | |
|  | The user can now respond or delete messages. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | *4.2.2* | |

* + 1. **Calculate Cost**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4.2.7 : Calculate Cost** | | | | |
| **Use case Id:** | | *4.2.7* | | |
| **Actors:**  *Users – Contractor/Client* | | | | |
| **Feature:**  *Cost Calculator* | | | | |
| **Pre-condition:** | | *The user type needs to be a client or Contractor and they need to be logged in.* | | |
| **Scenarios:** User Logs into the system and calculates the using the cost calculator. | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | *User clicks on cost calculator button* | | | *Redirects the user to open the cost calculator window* |
| **2.** | User selects and enter the details | | | Takes the input. |
| **3.** | User clicks on the button to the calculate the cost button to see the cost. | | | It shows the cost based on an equation and displays the cost. |
| **Alternate Scenarios:** | | | | |
| 1a: When calculating the cost the correct details are not entered.  2a: - The highlighted details are not entered. | | | | |
| **Post Conditions:** After successful entering of the values. | | | | |
| **Step#** | **Description** | | | |
|  | The cost is calculated and shown to the user. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | *4.2.2* | |

* + 1. **Add Member**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4.2.8 : Add Member** | | | | |
| **Use case Id:** | | *4.2.8* | | |
| **Actors:**  *Users - Contractor* | | | | |
| **Feature:**  *Add Member* | | | | |
| **Pre-condition:** | | *The user type needs to be Contractor and they need to be logged in.* | | |
| **Scenarios:** Admin logs into the system and adds the members. | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | *User clicks on the add new members button,* | | | *Redirects the user to the add new members* |
| **2.** | Enter member details. | | | Takes all the details as a response. |
| **3.** | User clicks on the save the details button | | | Saves the details and then add members in the database. |
| **Alternate Scenarios:** | | | | |
| **1a:** When adding new members the contact numbers must be in digits only. | | | | |
| **Post Conditions:** After successful addition new member is added. | | | | |
| **Step#** | **Description** | | | |
|  | The project members are now available and users can view them. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | *4.2.2* | |

1. **Non-functional Requirements**
   1. **Performance Requirements**

The application will have interactive interfaces that will be user-friendly and pages will be loaded instantly.Delays occur in the software will be minimal or non-existent due the fast speed of the servers. The response will be and connection with the server active to make everything smooth as possible

* 1. **Safety Requirements**

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the data must be backed up. Implementing secure user authentication and authorization mechanisms within the Node.js

* 1. **Security Requirements**

Proper input validation will be done to ensure that only legitimate users access the system's services. During registration Server-Side Validation will be done to ensure that all the necessary information was filled in. The password and email address compile with our expected format standards which are: passwords should contain at least 8 characters and must be alphanumeric, the email prefix as well as the email domain must be valid. Furthermore, the password hash will be stored in the database and upon login a hash of the entered password will be calculated and compared with the one stored in the database.

* 1. **User Documentation**

The user documentation that will be provided in the end will be:

1. User Manual
2. Online help

1. **References**

Not applicable here.

1. **Appendices**

Not applicable here.