

**Module Code & Module Title**

**CS6PO5NT-Final Year Project**

**Assessment Weightage & Type**

**Final Year Project Proposal (25%)**

**Year and Semester 2022-23 Autumn**

**Student Name: Aviyan Khatiwada**

**London Met ID: 20048413**

**College ID:** [**np05cp4s210119@iic.edu.np**](mailto:np05cp4s210119@iic.edu.np)

**Assignment Submission Date: 12/28/2022**

**Internal Supervisor: Mr. Nishesh Bishwas**

**External Supervisor: Mr. Bibek Maharjan**

**Title: Easy Hire**

*I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked.*

***I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.***

**Table of Contents**

[1. Introduction 1](#_Toc66597)

[1.1 Introduction to the topic 1](#_Toc66598)

[1.2 Current Scenario 2](#_Toc66599)

[1.3 Problem Statement 2](#_Toc66600)

[1.5 Aim and Objectives 3](#_Toc66601)

[1.6 Report Structure 4](#_Toc66602)

[1.6.1. Introduction 4](#_Toc66603)

[1.6.2. Background/Literature review 4](#_Toc66604)

[1.6.3. Development to date 4](#_Toc66605)

[1.6.4. Analysis of progress 4](#_Toc66606)

[1.6.5. Future work 5](#_Toc66607)

[1.6.6. References 5](#_Toc66608)

[1.6.7. Appendix 5](#_Toc66609)

[2.Background 5](#_Toc66610)

[2.1 Technology used 5](#_Toc66611)

[2.1.1 Programming Language 5](#_Toc66612)

[2.1.2 UI 6](#_Toc66613)

[2.1.3 IDE 8](#_Toc66614)

[2.2.1 Considered Methodologies 10](#_Toc66615)

[2.2.1.1. Scrum Methodology 10](#_Toc66616)

[2.2.1.2 Spiral Methodology 12](#_Toc66617)

[2.2.1.3 Waterfall Model 13](#_Toc66618)

[2.2.2 Selected Methodology 15](#_Toc66619)

[2.2.2.1 Rational Unified Process (RUP) Methodology 15](#_Toc66620)

[3.3.2. Reasons for selecting RUP: 17](#_Toc66621)

[3.5.1 Register 23](#_Toc66622)

[3.5.2 System login 24](#_Toc66623)

[4. Analysis of progress 30](#_Toc66624)

[4.1 Progress Table 30](#_Toc66625)

[4.2 Progress Review 31](#_Toc66626)

[4.3 Progress Timeline 31](#_Toc66627)

[5. Future work 32](#_Toc66628)

[6.References 33](#_Toc66629)

[7. Appendix 35](#_Toc66630)

**Table of Figures**

Figure 1:Node js .............................................................................................................. 6

Figure 2:Visual Studio Code ............................................................................................ 8

Figure 3:Xampp ............................................................................................................... 9

Figure 4:Scrum Methodology ........................................................................................ 11

Figure 5:Spiral Methodology .......................................................................................... 13

Figure 6:Waterfall Model Methodology .......................................................................... 14

Figure 7:Rational Unified Process Methodology ............................................................ 15

Figure 8:Survey Image 1 ............................................................................................... 20

Figure 9:Survey image 2 ............................................................................................... 20

Figure 10:Survey image 3 ............................................................................................. 21

Figure 11:Use case Diagram ......................................................................................... 22

Figure 12:My logo .......................................................................................................... 23

Figure 13:UI Homepage ................................................................................................ 28

Figure 14:About us page ............................................................................................... 28

Figure 15:Contact us page ............................................................................................ 29

Figure 16:System Architecture Diagram ........................................................................ 29

Figure 17:System Perspective ....................................................................................... 38

Figure 18:Home page .................................................................................................... 53

Figure 19:Login page .................................................................................................... 54

Figure 20:Registration page .......................................................................................... 54

Figure 21:About us and Contact us page ...................................................................... 55

Figure 22:Dashboard page ............................................................................................ 55

Figure 23:Gantt Chart .................................................................................................... 56

Figure 24:Milestone Chart ............................................................................................. 56

Figure 25:Survey image 4 ............................................................................................. 57

Figure 26:survey image 5 .............................................................................................. 57

Figure 27:Survey image 6 ............................................................................................. 58

Figure 28:Survey image 7 ............................................................................................. 58

Figure 29:Survey image 8 ............................................................................................. 59

Figure 30:Survey image 9 ............................................................................................. 59 Figure 31:Survey image 10 ........................................................................................... 60

Figure 32:Suggestions and Feedback from Survey ....................................................... 60

**Table of Tables**

Table 1:Features Comparison between similar projects ................................................ 19

Table 2:Register highlevel ............................................................................................. 23

Table 3:System Login highlevel ..................................................................................... 24

Table 4:Update profile high level ................................................................................... 24

Table 5:view available worker high level ....................................................................... 25

Table 6:View Worker Details high level ......................................................................... 25

Table 7:Post vanacies high level ................................................................................... 25

Table 8: choose worker high level ................................................................................. 25

Table 9:make payment high level .................................................................................. 26

Table 10:apply for job high level .................................................................................... 26

Table 11:view available vacancy high level ................................................................... 26

Table 12:View Job status high level .............................................................................. 26

Table 13:delete/ban profiles high level .......................................................................... 27

Table 14:Notify updates high level ................................................................................ 27

Table 15:Progress table ................................................................................................ 31

Table 16:Register User.................................................................................................. 42

Table 17:Login ............................................................................................................... 43

Table 18:Update Profile ................................................................................................. 44

Table 19:View available worker ..................................................................................... 45

Table 20:Post vacancies ............................................................................................... 45

Table 21:Choose Workers ............................................................................................. 46

Table 22:Apply for Job .................................................................................................. 47

Table 23:Make Payment................................................................................................ 48

Table 24:View Available vacancies ............................................................................... 48

Table 25:View Job Status .............................................................................................. 49

Table 26:Delete/Ban profiles ......................................................................................... 49

Table 27:Notify updates................................................................................................. 50

Table 28:Performance reequirements ........................................................................... 52

Table 29:Safety Requirements ...................................................................................... 52

Table 30:Other software quality attributes ..................................................................... 53

# 1. Introduction

## 1.1 Introduction to the topic

We are all aware that technology is always developing, and as a result, more and more people are using mobile and web applications in various aspects of their daily lives. According to the kepis analysis, there were 11.51 million internet users in Nepal in January 2022. There are many various kinds of applications that have dominated the market, and many people are requesting new apps every day to make life easier. Among other things, Web-based applications are becoming more and more well-liked and helpful to all businesses and organizations. Still many peoples are searching for jobs manually. The manual way of searching for workers or jobs is very difficult task. As modernization takes over all systems, digitalization helps them better in a variety of ways. The “Easy

Hire” web application is one of the solutions that will help people to hire workers and helps workers to find jobs easily.

The goal of this web application is to digitalize the system of hiring professionals and distribute jobs to the professionals so that it could save time of the Users. The “Easy Hire” web application’s major aim is to make easier platform for both clients and freelancers with the smooth and streamline process to get a deal reducing the time, cost, and paperwork. As a result, the system has a very quick response time and operates very quickly. This application has a streamlined dashboard with all the professions details presented also with the professionals.

## 1.2 Current Scenario

Technology is the application of scientific knowledge to real-world problems or purposes. This is the age of advanced technology, and people from all walks of life are learning how to make the most of it to have better facilities and a more comfortable living. The usage of mobile phones is now in every human being's blood.

Many IT industries in the world are carrying out real-world projects to solve daily life issues.In Nepal, the similar situation exists. Manual job searching is timeconsuming and difficult for professionals or workers to do well. Additionally, the hiring process is so complicated, and one cannot imagine working as a freelancer.

The “Easy Hire” web application is one of the solutions that helps peoples to hire freelancers or workers as per their need and minimize the complexity of workers to find jobs. The system will handle the task such as adding and removing workers details and availability in the system, Preserving client’s details and payment history and other task necessary for the “Easy Hire” web application.

## 1.3 Problem Statement

Despite the fact that the world is getting increasingly digital, many people still struggle to discover the professionals they want to work with. In Nepal, the similar situation exists. Manual job searching is time-consuming and difficult for professionals or workers to do well. Additionally, the hiring process is so complicated, and one cannot imagine working as a freelancer.

These are some of the problems faced.

* When parents go to their work/office, most of the time they are worried about their children/babies and also their pets being alone at home.
* Traveling can be fun but without professional drivers, it could be a difficult journey and finding drivers at ease is always challenging.
* Functions are very important, and it should be entertaining as well but finding djs and musicians for instant functions is very challenging.
* For every professional mentioned above and more, freelancing is never made easy.
* Requires lots of time and manpower to search for freelancers mentioned above.

## 1.4 Project as a solution

The “Easy Hire” is a user-friendly web application. This system simplifies all functions for both clients and workers. It is really simple in design and to use. The “Easy Hire” web application is one of the solutions that helps peoples to hire freelancers or workers as per their need and minimize the complexity of workers to find jobs. The system will handle the task such as adding and removing workers details and availability in the system, Preserving client’s details and payment history and other task necessary for the “Easy Hire” web application.

This is a one stop, or one clicks solution for the people with their busy schedule. Clients can choose the workers by viewing their experience, ratings, and past works.. To add it will also respect the privacy of the clients and the professionals, since only authorized users will be able to access the system using their credentials.

## 1.5 Aim and Objectives

###### 1.5.1 Aim

The “Easy Hire” web application’s major aim is to make easier platform for both clients and freelancers with the smooth and streamline process to get a deal reducing the time, cost, and paperwork.

###### 1.5.2 Objectives

The objectives of the “Easy hire” is to make the hiring process smoother in the context of Nepal.

* To create a system that allows professionals to list their work location, experience and make themselves open to work.
* To store the details of the user.
* To allow users to rate the work of the professionals.
* To recommend and suggest nearby professionals according to the best ratings as well.
* To give a quick response with very accurate information on the professionals and clients

## 1.6 Report Structure

#### 1.6.1. Introduction

Introduction consists of information about project, problem domain, end users, project scope, aims and objectives of the project.

#### 1.6.2. Background/Literature review

Background/Literature review consists of information about ends users, understanding solution for problem, review of similar application, technology and methodology that has been used in this project.

#### 1.6.3. Development to date

In Development to date chapter, progress of project is included.

#### 1.6.4. Analysis of progress

Analysis of progress chapter include the progress analysis to obtain the information on the state of project, how it is being developed? This chapter helps to determine the actual progress of project according to Gantt chart.

#### 1.6.5. Future work

Future work includes the remaining task of this project to be carried out in future.

#### 1.6.6. References

In references section, it included all the references that are taken as reference while doing this project, report.

#### 1.6.7. Appendix

In appendix sections, it contains the brief description of SRS document, high level use case diagram and wireframes with screenshot. Similarly, relationship between entities are explained. Data dictionary of each entity is kept. Additionally, it included brief description of project as solution, considered and selected methodology. Thus, scanning images of survey done at parking location, screen shot of android and web code are kept here.

# 2.Background

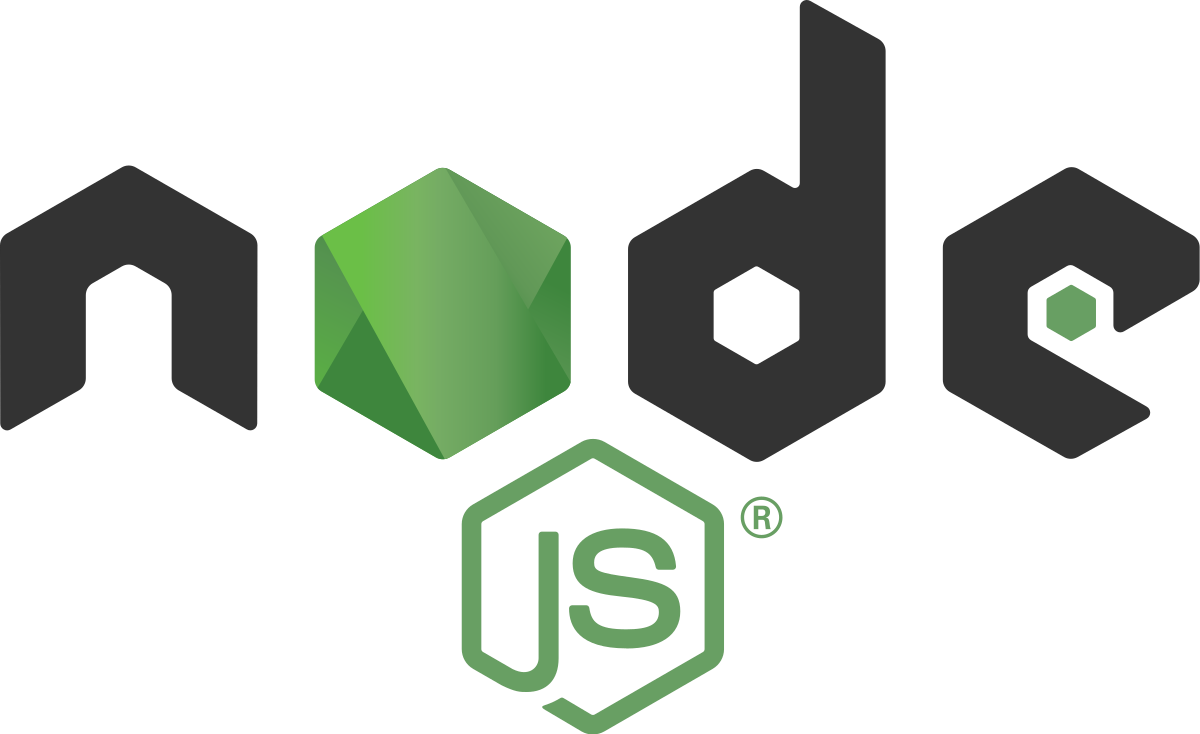
## 2.1 Technology used

### 2.1.1 Programming Language

1. **Nodejs**

Node.js (Node) is an [open-source,](https://www.techtarget.com/whatis/definition/open-source) cross-platform runtime environment for executing [JavaScript](https://www.theserverside.com/definition/JavaScript?_ga=2.163145724.1963875579.1671854332-1899672515.1650861734&_gl=1*dlrwlh*_ga*MTg5OTY3MjUxNS4xNjUwODYxNzM0*_ga_TQKE4GS5P9*MTY3MTg1NDMzMi40LjAuMTY3MTg1NDMzMi4wLjAuMA..) code. Developers can utilize JavaScript for both client-side and server-side code without having to learn another language because Node is widely used for server-side programming. Node is occasionally described as a programming language or a framework for software development, however none of these descriptions is accurate; it is solely a JavaScript runtime. One process handles all aspects of a Node application. Like many traditional server-side programs, Node does not start a new thread for each request. In this method, thread concurrency difficulties and the overhead multithreading causes are avoided, allowing a Node server to handle thousands of concurrent clients. (Robert

Sheldon, 2022)



*Figure 1:Node js*

1. **JavaScript**

When a web page displays more than simply static data for you to look at, such as realtime content updates, interactive maps, dynamic 2D/3D graphics, scrolling video jukeboxes, etc., JavaScript enables you to add complicated features to the website. As browsers improve daily, JavaScript has grown in popularity as a reliable platform for creating internet applications. By utilizing Google Maps as an illustration, we can better understand it. To disclose the information in Maps, users merely need to click and move the mouse; clicking makes the details apparent. These concepts are implemented using JavaScript. (tutorialspoint, 2022)

#### 2.1.2 UI

**HTML and CSS**

HTML is an acronym for Hypertext Markup Language. Building electronic documents (also known as pages) that are displayed on the World Wide Web requires the usage of the markup language HTML. Each page contains a list of hyperlinks, which are connections to other websites. You are currently viewing a web page that was made using a specific type of HTML. Cascading Style Sheets are used to change the page's appearance after HTML provides the page's basic framework. Consider HTML to be the bones (structure) of a web page, and CSS to be its skin (appearance). (Fran, 2021)

CSS, or Cascading Style Sheets, is a simple language meant to make the process of creating web pages presentable easier. Styles may be applied to web pages using CSS. More crucially, CSS allows you to accomplish this without relying on the HTML that sets up each web page. CSS is simple to learn and comprehend, but it gives you a lot of power over how an HTML document looks. (Fran, 2021)

**Framework**

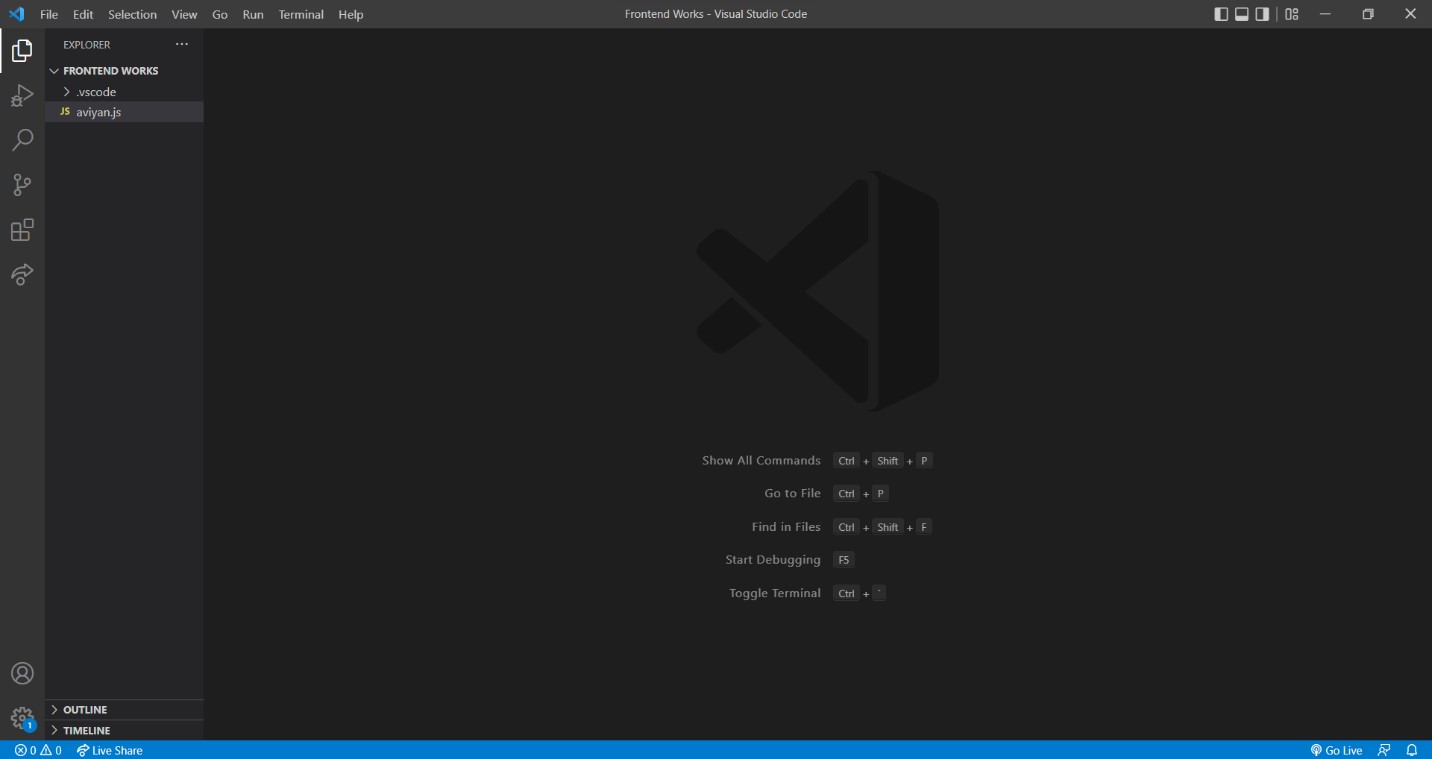
**a) Vue js**

Vue is a JavaScript framework for building user interfaces. It builds on top of standard HTML, CSS, and JavaScript and provides a declarative and component-based programming model that helps you efficiently develop user interfaces, be they simple or complex. (Vue.js, 2022)

#### 2.1.3 IDE

**a) Visual Studio Code**

The free open-source text editor Visual Studio Code from Microsoft is frequently abbreviated as "VS Code." VS Code is compatible with Windows, Linux, and macOS. Although the editor is basic, it has a number of powerful features that have helped VS Code rise to the top of the development environment tool rankings in recent years. Along with CSS, Go, and Docker files, VS Code supports a wide variety of programming languages, including Java, C++, and Python. Furthermore, you may add on and even create new extensions for VS Code, such as code linters, debuggers, and cloud and web development support. (Martin Heller, 2022)



*Figure 2:Visual Studio Code*

**2.1.4 Database**

**Web Server**

XAMPP is a well-known cross-platform web server that enables programmers to build and test code on a local web serverPHP and Perl are represented by the Ps in the acronym XAMPP, which stands for Cross-Platform, Apache, MySQL, PHP, and Perl. It is

*Figure 3:Xampp*

a collection of open-source online solutions that includes Apache server, MariaDB, PHP, and Perl modules along with distributions of Apache for multiple servers and commandline executables. It works on various platforms and is accessible in 11 different languages, including IA-32 for Windows, x64 for Mac OS X, and x64 for Linux. It is a platform that provides an appropriate environment for testing and confirming the functionality of projects based on Apache, Perl, MySQL, and PHP on the host's system. (javaTpoint, 2022)

**2.2 Methodology**

Software development methodologies are essential in the development of software. The software development methodology is a framework for structuring, planning, and controlling the process of developing an information system. Each methodology has its own set of pros and limitations. The primary goal of these techniques is to provide smooth software development in accordance with the project requirements. The main issue of this software development process in this type of development methodology is that it does not include any technical aspects but requires proper planning for the software development lifecycle.

Among all methodologies, Rational unified Process Methodology (RUP) was selected because it offers a standardized and systematic approach to creating software systems, as well as a detailed plan for each process in the development process.

### 2.2.1 Considered Methodologies

##### 2.2.1.1. Scrum Methodology

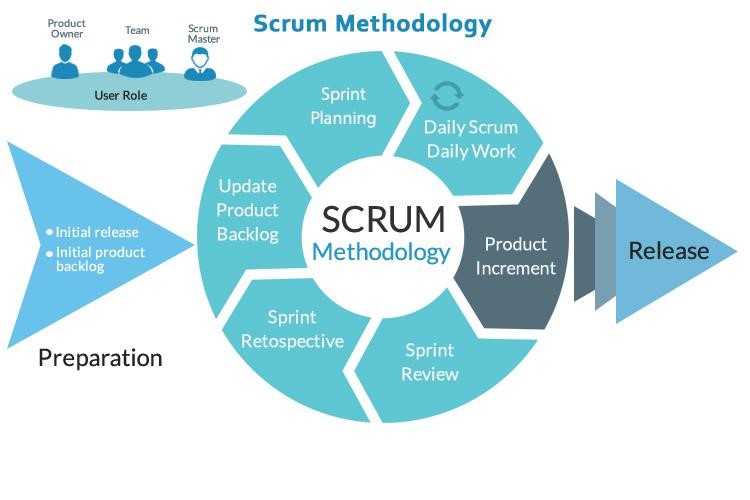
Scrum is an agile project management method that is commonly used in software development. Agile software development with Scrum is frequently seen as a methodology; however, instead of considering Scrum as a methodology, consider it a framework for managing a process. Scrum is based on a self-organizing, multi-functional team. The scrum team is self-organizing in the sense that there is no general team leader who chooses who does what work or how an issue is solved. These are decisions made by the team as a whole. In Scrum, a team is cross-functional, which means that everyone is required to take a feature from concept to implementation. (PROJECTPRACTICAL, 2022)

**Advantages of Scrum Methodology:**

* Scrum may help teams in completing project deliverables in a timely and effective manner.
* Scrum ensures that time and money are spent wisely.
* Large projects are broken down into smaller sprints.
* During the sprint review, developments are coded and tested. This works effectively for quickly development projects.
* Short sprints make it much easier to make adjustments based on feedback. (PROJECTPRACTICAL, 2022)

**Disadvantages of Scrum Methodology:**

* Due of the lack of a fixed end-date, Scrum frequently leads to scope creep.
* If they aren't dedicated or cooperative, there's a good probability the project will fail. •



*Figure 4:Scrum Methodology* Adopting the Scrum framework in big groups is difficult.

* Any team member who departs in the middle of a project might have a significant negative influence on the project.
* Quality is difficult to implement until the team goes through a rigorous testing procedure.

(PROJECTPRACTICAL, 2022)

##### 2.2.1.2 Spiral Methodology

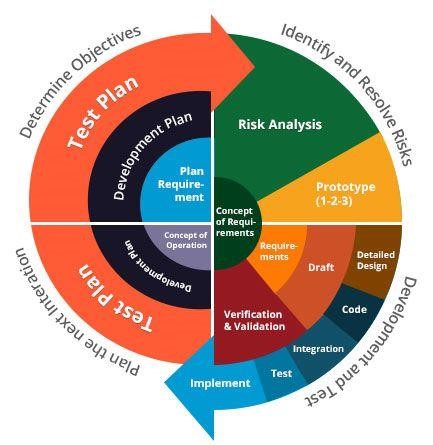
The spiral model is one of the most important Software Development Life Cycle models that supports risk management. It seems to be a spiral with multiple loops in its diagrammatic form. The precise number of spiral loops is uncertain and varies from project to project. Each spiral loop is referred to as a Phase in the software development process. The project manager can change the number of phases required to build the product based on the project risks. Because the project manager chooses the number of stages dynamically, the project manager plays a significant rolein developing a product utilizing the spiral model. (RajKumar, 2020)

**Advantages of Spiral Methodology:**

* Additional functionality or changes can be done at a later stage.
* Because the prototype is built in smaller fragments, cost estimating becomes simple.
* Risk management is aided by continuous or recurrent development.
* Spiral development is quick, and features are introduced in a systematic manner.
* There is always space for customer input. (RajKumar, 2020)

**Disadvantages of Spiral Methodology:**

* There is a chance that the timeline or budget will not be met.
* Spiral development is best suited for large projects, but it also needs risk assessment competence.
* The spiral model protocol must be closely followed in order for it to function properly.
* Documentation is more extensive since there are intermediary steps.
* Spiral software development is not recommended for small projects since it might be expensive. (RajKumar, 2020)



*Figure 5:Spiral Methodology*

##### 2.2.1.3 Waterfall Model

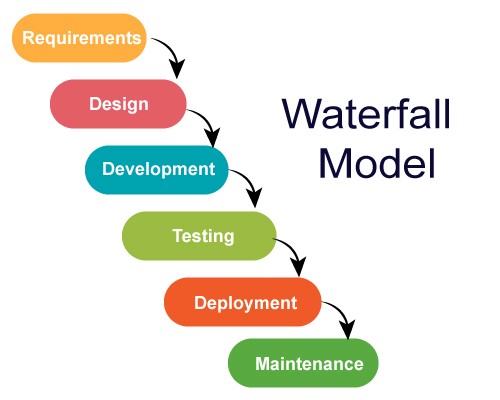
The Waterfall Model was the first to be offered as a Process Model. It's also known as a linear sequential life cycle model. It is quite simple to grasp and apply. In a waterfall model, each phase must be finished before the next one can begin, and the stages must not overlap. The Waterfall model was the first SDLC technique utilized in software development. The waterfall Model depicts the software development process in a sequential, linear flow. This indicates that any step of the development process may begin only after the preceding phase has been completed. The stages in this waterfall model do not overlap. (Matthew Martin, 2022)

**Advantages of Waterfall Model:**

* Simple and easy to understand and use.
* Because of the model's rigidity, it is simple to manage. Each phase includes its own set of deliverables and a review process.
* One phase at a time is processed and completed.
* Works effectively for smaller projects with well-defined requirements. • Tasks are simple to organize. (Matthew Martin, 2022)

**Disadvantages of Waterfall Model:**

* There is a lot of danger and uncertainty.
* This is not an appropriate approach for complex, object-oriented programs.
* Poor model for long-term projects.
* Not appropriate for projects with a moderate to high risk of change in requirements.
* Changing the scope of a project during its life cycle might lead to its destruction. (Matthew Martin, 2022)

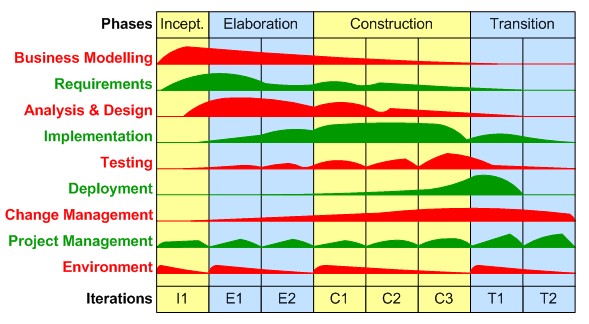


*Figure 6:Waterfall Model Methodology*

### 2.2.2 Selected Methodology

##### 2.2.2.1 Rational Unified Process (RUP) Methodology

RUP is a software development methodology that is agile. The project life cycle is divided into four phases according to RUP. All 6 major development disciplines are carried out at each phase: business modeling, requirements, analysis and design, implementation, testing, and deployment. Certain processes, however, are more critical and need more time at each phase. Each of the four phases has a primary goal that must be met before the project can go on to the next phase. RUP's major objective is to provide high-quality software within a reasonable price and time limit. If required, each of the life cycle phases can be repeated until the main objective is achieved. (Master2Teach, 2022) The project is completed once the transition phase is successfully completed So, this methodology is chosen to develop Easy Hire . Basically, RUP methodology has four different phases which are listed below:



*Figure 7:Rational Unified Process Methodology*

* **Inception:** It is the first phase of the development phase. Basically, we define the design and core idea of the project at this phase. We will also decide on the project's purpose, success criteria, risk assessment, scheduled time, and resources needed to execute it, etc. It is similar to a project evaluation. (Master2Teach, 2022)

* **Elaboration:** During the elaboration phase, the design of the project and the resources required are analyzed further. This phase attempts to assess the system's requirements and design, establish the project plan, and reduce the project's greatest risk aspects. It is, without a doubt, the most essential of all phases since it represents the transition from low risk to high-risk. It is also the time at which we must determine whether or not to begin construction (development and coding). (Master2Teach, 2022)
* **Construction:** The application framework is built in its entirety during the construction phase. We will be finally ready to develop all components and features and merge them into the application at this phase. It is a manufacturing method in which we focus on managing resources in order to maximize costs, schedules, and quality. The application is designed, written, and tested successfully. User Manual and fully completed software system are some of the results of the development phase. (Master2Teach, 2022)

* **Transaction:** The objective of the transition phase is to transfer the product to its new user. As soon as the user starts using the system, problems almost always arise that require changes to be made to the system. The goal, however, is to ensure a positive and smooth transition to the user. This requires the management of all bug fixes and problem solutions, as well as the completion of certain earlier postponed features. It is the deployment procedure. (Master2Teach, 2022)

Advantages of RUP Methodology

* Good documentation is provided, and the procedure is finished independently.
* It helps with risk management.
* Because the components are reused, the overall time is less.
* Training and tutorials are both excellent forms of online help. (Master2Teach, 2022)

Disadvantages of RUP Methodology

* The use of an experienced team is recommended due to the complexity of the process.
* The process that is difficult to arrange appropriately.
* increased reliance on risk management.
* Repeated integration is challenging. (Master2Teach, 2022)

#### 3.3.2. Reasons for selecting RUP:

* RUP methodology gives extensive support for identifying and eliminating risk factors in an earlier phase of the development lifecycle.
* It enables you to manage with changing requirements, whether they come from the customer or the project itself.
* It requires integration to take place throughout the software development process, particularly during the construction phase.
* It highlights the need of proper documentation

**3.3 Review of Similar Project** **3.3.1 Similar Project**

a) Rojgari

Rojgari is a mobile app working to empower job seekers with a wide range of employment related information and service, particularly jobs counselling, soft skills training, and employment matchmaking. Rojgari enlists and connects job seekers and employers in Nepal through an online portal www.rojgari.com, Rojgari mobile Application, 'Rojgari

Pasal', a brick and mortar physical outlet branded as ‘Employment Shop’, Ghumti Rojgari Pasal , a mobile pasal in Van which run Inside Nepal in that places where Rojgari Pasal is no presence. And it lacks many features like notifications, video and photo upload etc.

b) Find A Worker – Workers find job

Find A Worker – Workers find job is a mobile app to post job vacancies and hire workers. Find A Worker is the easiest and best way for employers to post job vacancies and get hired fast. It is also the easiest way to get hired for the exact job they’ve been looking for. Find A Worker is 100% free for workers. This app helps workers with all types of jobs like part time, full time, seasonal, automotive, restaurant, retail, hospitality, warehouse, office, entry level, customer service jobs or if they are just looking for side work. Basically, anything they are looking for. And it still lacks features like notifications, Recommendation, photo and video uploads, etc

c) Jobit – Work and Hire

Jobit is a mobile app beautifully designed to help people find thier next job or hire thier next employee, easier than ever. For employers, they've made it faster than ever before to find the best candidates for any field by allowing search based on skills, experience, education or whatever thier requirements may be.People can post a job in one step and start receiving applicants immediately. For job seekers, one-tap application process is the fastest in the world. People will be able to go through tons of different jobs in no time and find the best opportunities for them! It still lacks features like Recommendation, Video and photo upload, etc.

**3.3.2 Comparison Between System.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Features** | **Rojgari** | **Jobit-Work and Hire** | **Find a Worker** | **Easy Hire** |
| Photos, videos  upload | No | No | No | Yes |
| Login | Yes | Yes | Yes | Yes |
| Background Check | Yes | No | No | Yes |
| Communication tool | Yes | Yes | Yes | Yes |
| Streamlined  Dashboard | Yes | No | No | Yes |
| Plan Page with  Ratings | No | No | No | Yes |
| Recommendation  Feature | No | No | No | Yes |
| Notification via email | No | Yes | No | Yes |
| Easy to use | Yes | Yes | Yes | Yes |
| Mobile app | Yes | Yes | Yes | No |

*Table 1:Features Comparison between similar projects*

2. Development to Date

3.1 Requirement Gathering

The requirement of this project is gathered by conducting online surveys inside and outside the college using google Forms from December 26, 2022. While conducting an online survey inside and outside the college, 53 people responded to the survey form. The survey consists of questionnaires that include project practicability and feasibility.

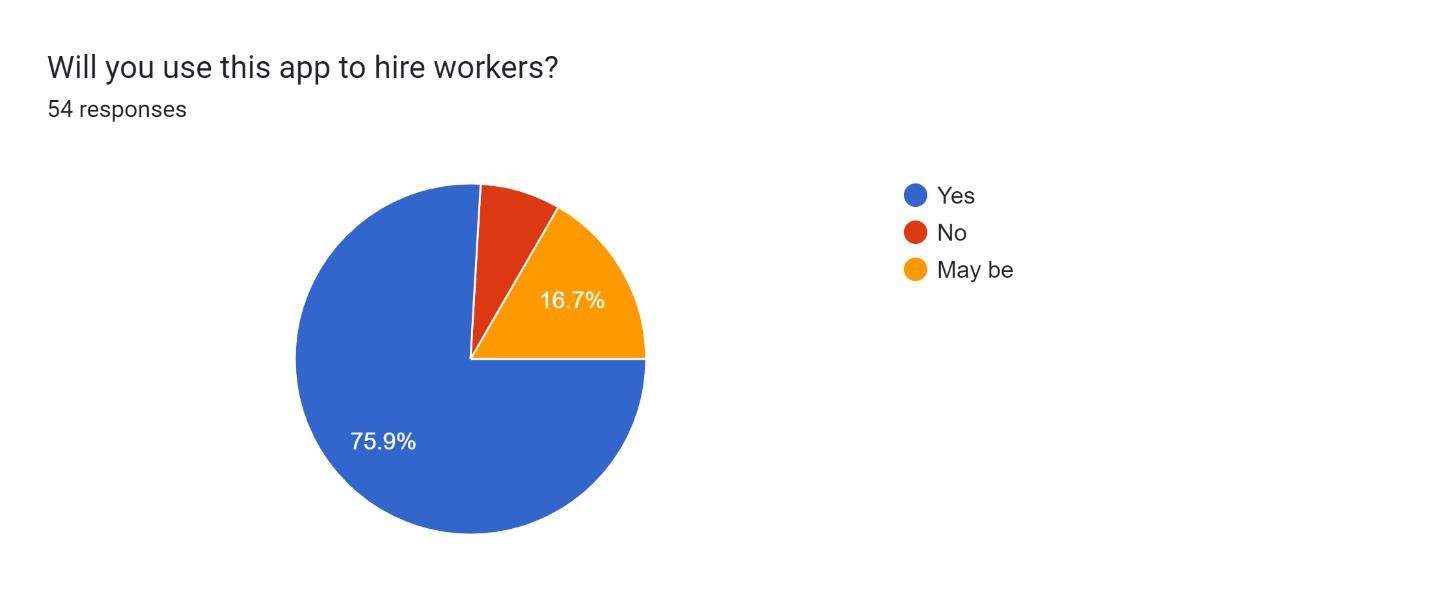
3.1.1 Conducted Online Survey

1. The survey response encourages me to create a web application that can replace manual work with a digital system. In a pre-survey form, among 53 participants, 75% of participants find it very helpful.



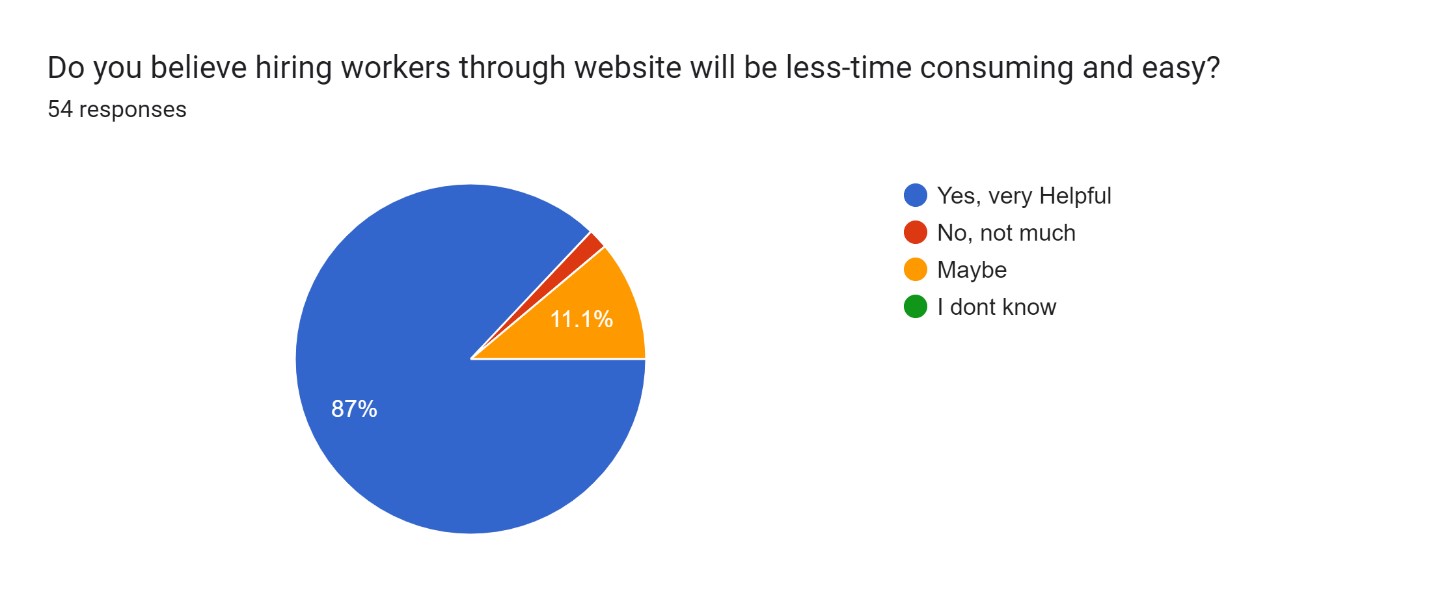
*Figure 8:Survey Image 1*

1. About 75% of people will use this app to hire workers according to the responses.



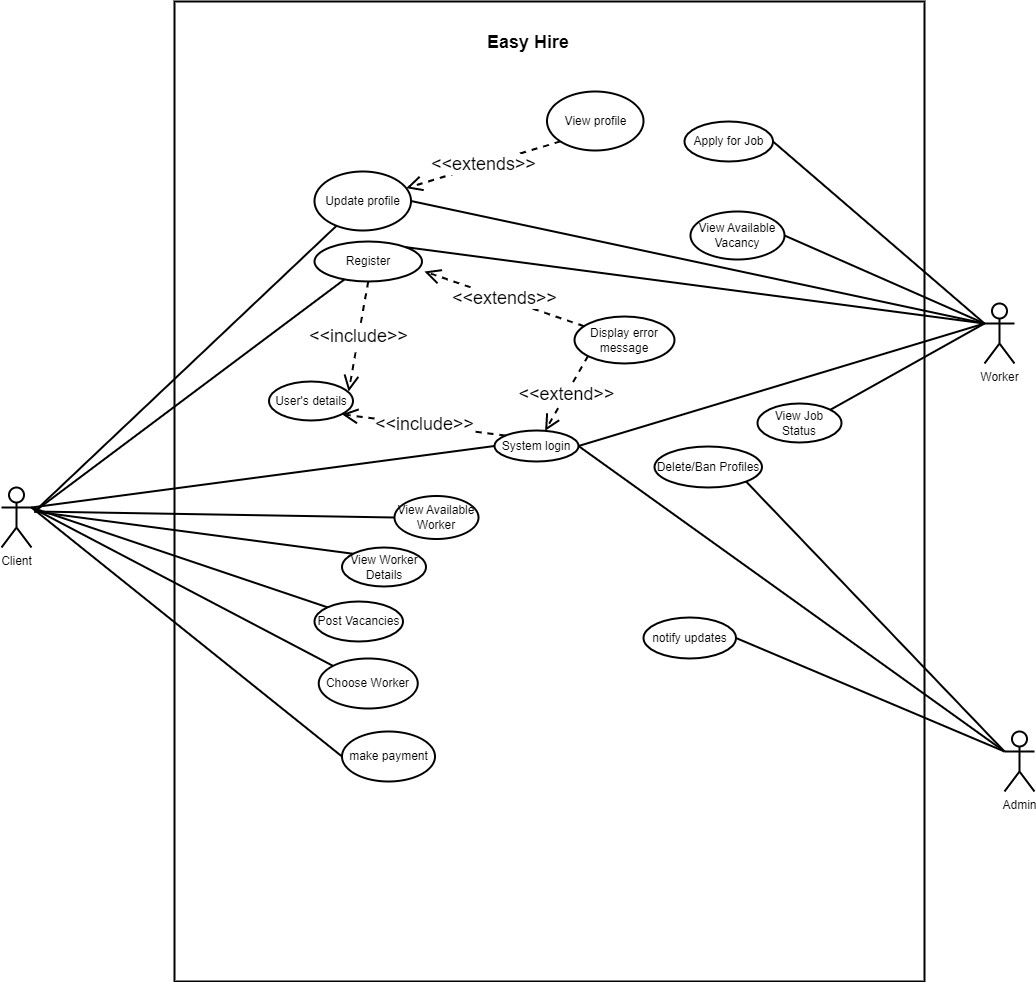
*Figure 9:Survey image 2*

1. Most people from the pre-survey form believe that hiring workers through website will be less-time consuming and easy. Among 54 people, 87% think it will be very helpful, and 11% think it maybe helpful.



*Figure 10:Survey image 3*

3.2 Use Case



*Figure 11:Initial Use case Diagram*

Diagram

Description automatically generated

Figure 12:Final Use case diagram

Shape

Description automatically generated with low confidence3.3 Designed Logo

*Figure 12:My logo*

**3.4 SRS Document**

The full description of the SRS document is kept in the Appendix section.

3.5 High-Level Use Case Diagram

Easy hire high-level use cases are listed below. This use case description since it contains limited information about the use case.

#### 3.5.1 Register

|  |  |
| --- | --- |
| **Use case:** | Register |
| **Actor:** | Client, worker |
| **Description:** | The system requests the client and the worker their name, email, password, and other relevant data during registration. |

*Table 2:Register highlevel*

#### 3.5.2 System login

|  |  |
| --- | --- |
| **Use case:** | System login |
| **Actor:** | Client, worker, and Admin |
| **Description:** | The client, the worker and the admin login to the system using verified login password and essentials details. |

*Table 3:System Login highlevel*

3.5.3 Update Profile

|  |  |
| --- | --- |
| **Use case:** | Update Profile |
| **Actor:** | Client , worker |
| **Description:** | In the system, The client and the worker can update their details in the profile section. |

*Table 4:Update profile high level*

3.5.4 View Available Worker

|  |  |
| --- | --- |
| **Use case:** | View Available workers |
| **Actor:** | Client |
| **Description:** | In the system, The client will be able to view the availability of the workers to hire them. |

*Table 5:view available worker high level*

3.5.5 View Worker Details

|  |  |
| --- | --- |
| **Use case:** | View Workers Details |
| **Actor:** | Client |
| **Description:** | In the system, The client will be able to view the details of the workers like; Name, Experience, Ratings, etc to know them before hiring. |

*Table 6:View Worker Details high level*

3.5.6 Post vacancies

|  |  |
| --- | --- |
| **Use case:** | Post Vacancies |
| **Actor:** | Client |
| **Description:** | In the system, The client will be able to post vacancies for the workers to hire them. |

*Table 7:Post vanacies high level*

3.5.7 Choose Worker

|  |  |
| --- | --- |
| **Use case:** | Choose Worker |
| **Actor:** | Client |
| **Description:** | In the system, The client will be able to choose workers among many job applications with their efficiency. |

*Table 8: choose worker high level*

3.5.8 Make Payment

|  |  |
| --- | --- |
| **Use case:** | Make payment |
| **Actor:** | Client |
| **Description:** | In the system, The client will be able to pay the workers through Hand cash system. |

*Table 9:make payment high level*

3.5.9 Apply for Job

|  |  |
| --- | --- |
| **Use case:** | Apply for job |
| **Actor:** | Worker |
| **Description:** | In the system, The workers will be able to apply for job or they can upload their job application whenever it is necessary. |

*Table 10:apply for job high level*

3.5.10 View Available vacancy

|  |  |
| --- | --- |
| **Use case:** | View available vacancy |
| **Actor:** | Worker |
| **Description:** | In the system, The Workers will be able to view the available vacancies and can apply with their interest. |

*Table 11:view available vacancy high level*

3.5.11 View Job Status

|  |  |
| --- | --- |
| **Use case:** | View Job Status |
| **Actor:** | Worker |
| **Description:** | In the system, The workers will be able to view their job status, whether they are selected or not for the jobs they applied. |

*Table 12:View Job status high level*

3.5.12 Delete/Ban Profiles

|  |  |
| --- | --- |
| **Use case:** | Delete/Ban Profiles |
| **Actor:** | Admin |
| **Description:** | In the system, Admin will be able to delete or ban the profiles of both clients and workers at anytime with valid reasons. |

*Table 13:delete/ban profiles high level*

3.5.13 Notify updates

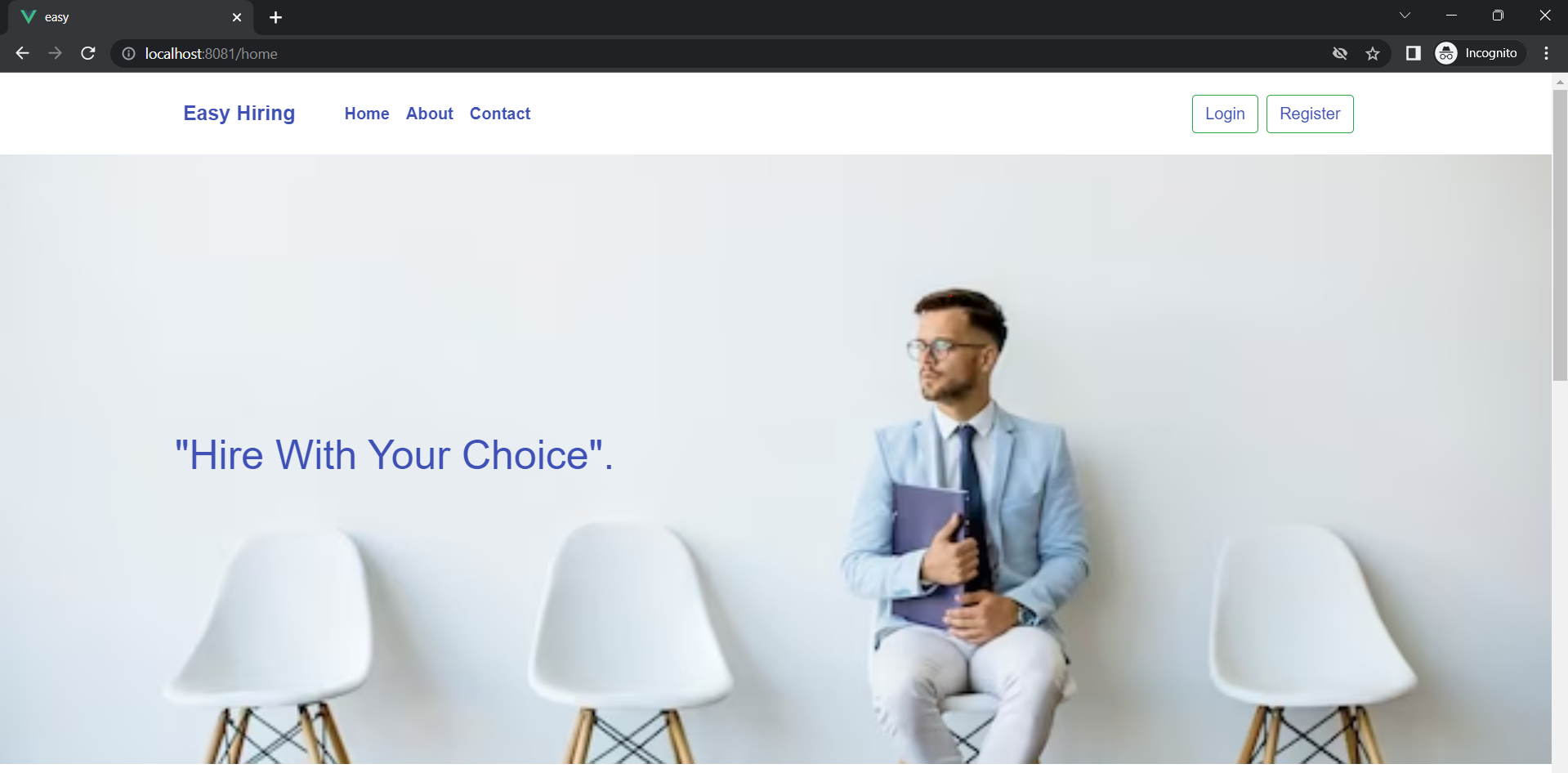
|  |  |
| --- | --- |
| **Use case:** | Notify updates |
| **Actor:** | Admin |
| **Description:** | In the system, Admin will be able to send notification to clients as well as workers through email and message. |

*Table 14:Notify updates high level*

**3.6 Wireframes**

The image of wireframes is kept in the appendix section.

3.8 Initial UI



*Figure 13:UI Homepage*

Graphical user interface, website

Description automatically generated

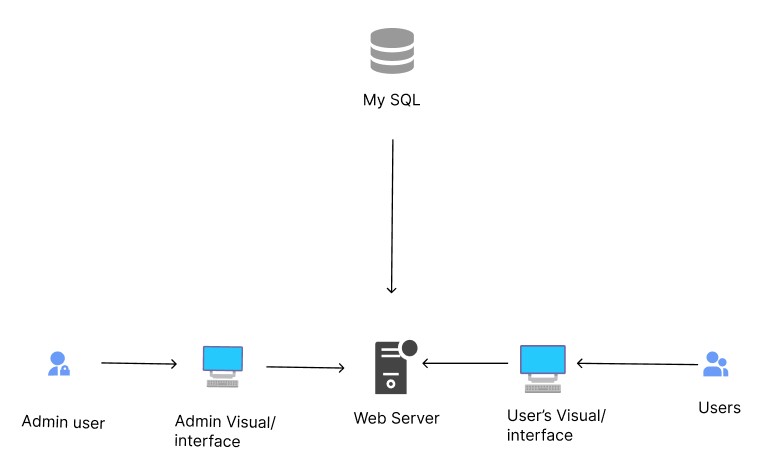
*Figure 14:About us page*

Graphical user interface, application

Description automatically generated

*Figure 15:Contact us page*

3.8 System Architecture Diagram



*Figure 16:System Architecture Diagram*

**Test Plan**

After developing the system, It is very much prominent to test the system before deploying it to the client because a testing helps to find out the flaws and bugs into the system by which we can fix the bugs and flaws. So, the proper test case and test scenario is made to test each features of the system by which we can increase the quality of the system. This testing phase is carried out in all project development before deploying it to the general public in order to find out whether all the functions and features implemented in the project works in the correct manner or not. Testing the system is so risky and concentrative work because if some test cases are missed to test than it may create a bad impression towards client is some failure occurs in client side.

**Unit Testing, Test Plans**

|  |  |
| --- | --- |
| **Test Case** | **Objective** |
| User Register | As a user,I should be able to register into the system |
| Test 2 | As a user,I should be able,to login into the system |
| Test 3 | As a admin, I should be able to login admin into the system |
| Test 4 | As an admin, I should be able to add location into the system by admin |
| Test 5 | As a admin, I should be able to remove location into the system by admin |
| Test 6 | As a admin, I should be able to add blogs into the system by admin |
| Test 7 | As a admin, I should be able to Send mass notifications by admin |
| Test 8 | As a admin, I should be able to add job category into the system by admin |
| Test 9 | As a admin, I should be able to remove job category into the system by admin |
| Test 10 | As a admin, I should be able to add salary range into the system by admin |
| Test 11 | As a admin, I should be able to remove salary range into the system by admin |
| Test 12 | As a admin, I should be able to confirm the new users in waiting list into the system by admin |
| Test 13 | As a admin, I should be able to remove registered users from the system by admin |
| Test 14 | As a admin, I should be able to update username and password from the system by admin |
| Test 15 | As a user,I should be able to update the user profile |
| Test 16 | As a user,I should be able to post a job into the system |
| Test 17 | As a user,I should be able to apply job into the system |
| Test 18 | As a user,I should be able to search jobs by location name |
| Test 19 | As a user,I should be able to search jobs by category name |
| Test 20 | As a user,I should be able to give rating to the users. |
| Test 21 | To check registration validation |
| Test 22 | To check login validation |
| Test 23 | As a user, I should be able to check my job applied history. |
| Test 24 | As a user, I should be able to check my job created history. |
| Test 25 | As a user, |

**Critical Analysis**

As a final-year student, completing my project on "Easy Hire" was a significant milestone for me. This project was aimed at developing a user-friendly web application that would simplify the process of hiring freelancers or workers for clients and simplify the job search process for workers. In this critical analysis, I will provide a detailed account of my journey in completing this project.

The first step in the process was to conduct extensive research on the requirements of the project. I analyzed similar web applications available in the market, studied their features and functionalities, and identified the gaps that could be filled through the "Easy Hire" web application. This research helped me in developing a clear understanding of the project requirements and the technologies that would be best suited for the project.

After conducting the research, I chose Vue.js as the frontend framework and Node.js as the backend technology to develop the application. Vue.js is a progressive JavaScript framework used for building user interfaces, while Node.js is a popular backend technology used for building scalable and high-performance web applications. I chose these technologies based on their popularity, compatibility, and ease of use.

Once the technologies were chosen, I began the development process. I followed the Rational Unified Process (RUP) methodology to evaluate the tasks involved in the project. This helped me in breaking down the project into smaller, more manageable tasks, which I then prioritized based on their importance and complexity.

I created a detailed project plan and timeline for the development process. This plan included all the tasks involved in the development process, their deadlines, and the resources required to complete them. This helped me in keeping track of the progress of the project and ensured that I completed the project within the given time frame.

I used Visual Studio Code as my development platform, which provided me with all the necessary tools for the development process. I found it easy to use and very user-friendly. I used various plugins and extensions to make the coding process more efficient and effective.

During the development process, I faced several challenges. One of the main challenges was integrating the frontend and backend parts of the application. This required me to have a deep understanding of the technologies and their interaction with each other. I spent considerable time on this task, but it was worth it as it helped me learn a lot about the technologies and their working.

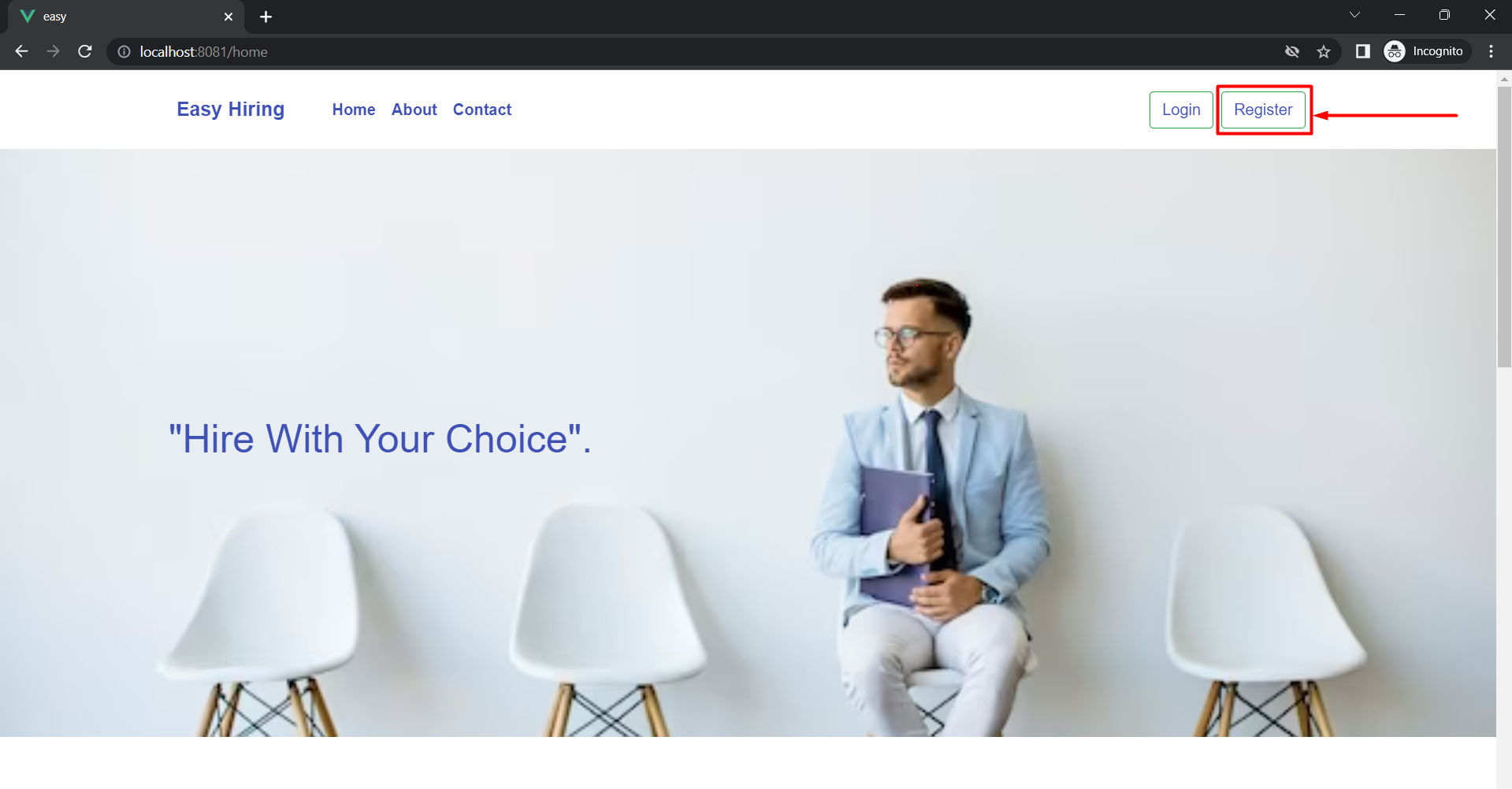
Another challenge I faced was designing the user interface of the application. I realized that designing a user-friendly and visually appealing interface was critical to the success of the application. I used various design principles and best practices to overcome this challenge, and the final design was well received by the users.

|  |  |
| --- | --- |
| **Test Case** | **Objective** |
| User Register | As a user,I should be able to register into the system |
| User Login | As a user,I should be able,to login into the system |
| Admin Login | As a admin, I should be able to login admin into the system |
| Add location list by admin | As an admin, I should be able to city admin |
| Delete location list by admin | As a admin, I should be able to remove location into the system by admin |
| Add blog by admin | As a admin, I should be able to add blogs into the system by admin |
| Send mass notification my admin | As a admin, I should be able to Send mass notifications by admin |
| Add Job category list by admin | As a admin, I should be able to add job category into the system by admin |
| Delete job category list by admin | As a admin, I should be able to remove job category into the system by admin |
| Add salary list by admin | As a admin, I should be able to add salary range into the system by admin |
| Delete salary list by admin | As a admin, I should be able to remove salary range into the system by admin |
| User approve by admin | As a admin, I should be able to confirm the new users in waiting list into the system by admin |
| User delete by admin | As a admin, I should be able to remove registered users from the system by admin |
| Update profile by admin | As a admin, I should be able to update username and password from the system by admin |
| Update profile by user | As a user,I should be able to update the user profile |
| Post job by user | As a user,I should be able to post a job into the system |
| Apply job by user | As a user,I should be able to apply job into the system |
| Search job (Location) | As a user,I should be able to search jobs by location name |
| Search job (category) | As a user,I should be able to search jobs by category name |
| Give ratings by user | As a user,I should be able to give rating to the users. |
| Invalid Registration | To check registration validation |
| Invalid Login | To check login validation |
| Check job applied history | As a user, I should be able to check my job applied history. |
| Check Job created history | As a user, I should be able to check my job created history. |
| Test 25 | As a user, |
|  |  |
|  |  |

**Unit Testing Test Execution**

**Test to register user into the system.**

|  |  |
| --- | --- |
| Objective | To check whether user can be registered in the system or not. |
| Action | Click on the register button on the home page. Enter valid user credentials in the register form. Click on register |
| Expected Result | The user should be created, and its credentials should store in database and the system should redirect the user into login page. |
| Actual Result | The system redirected the user into login page with a registration successful message. |
| Conclusion | Test Successful. |



Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

**Test to login user into the system.**

|  |  |
| --- | --- |
| Objective | To check whether user can be login in the system to not. |
| Action | Enter valid user credentials in the login form. Email: amitkhatiwada01@gmail.com  Password : Amit@1234 |
| Expected Result | The system should redirect the user into dashboard page after logged in. |
| Actual Result | The system redirected the user into dashboard page with a loggin successful message. |
| Conclusion | Test Successful. |

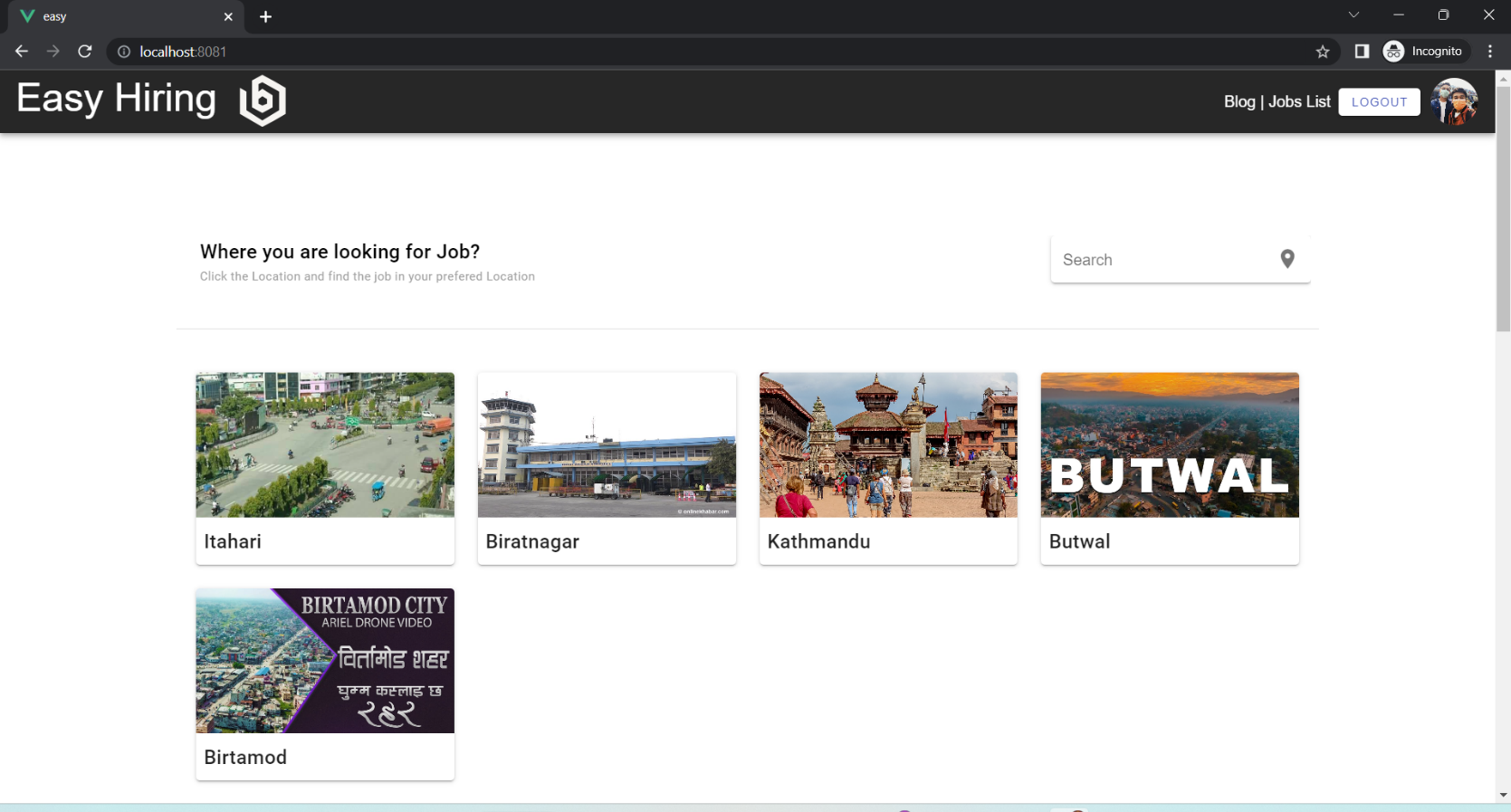
Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, application, Teams

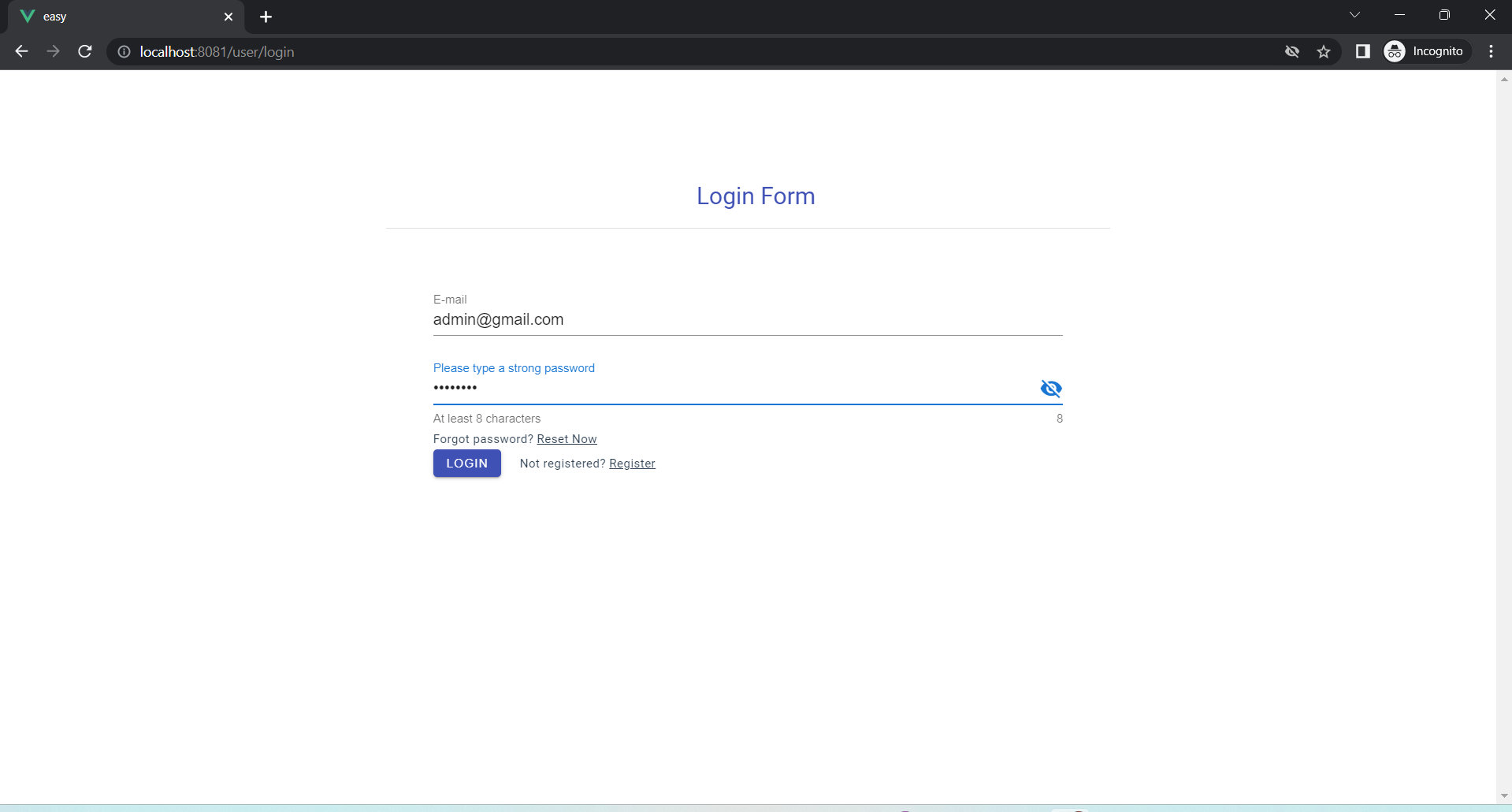
Description automatically generated

|  |  |
| --- | --- |
| Objective | To check whether admin can be login in the system to not. |
| Action | Enter valid user credentials in the login form. Email: [admin@gmail.com](mailto:admin@gmail.com)  Password: password |
| Expected Result | The system should redirect the admin into admin dashboard page after logged in. |
| Actual Result | The system redirected the admin into admin dashboard page with a loggedin successfully message. |
| Conclusion | Test Successful. |

**Test to login admin into the system**

Graphical user interface, application, Teams

Description automatically generated



Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, application

Description automatically generated

|  |  |
| --- | --- |
| Objective | To check whether admin can add location into the system. |
| Action | Open the dashboard of admin panel and click on city. After that, click on add city button. Enter the city name in the text field and then select Image of the city. Click on post city button |
| Expected Result | The system should redirect the admin into city list page after posting new city with a city added successfully message. |
| Actual Result | The system redirected the admin into city list page after posting new city with a city added successfully message. |
| Conclusion | Test Successful. |

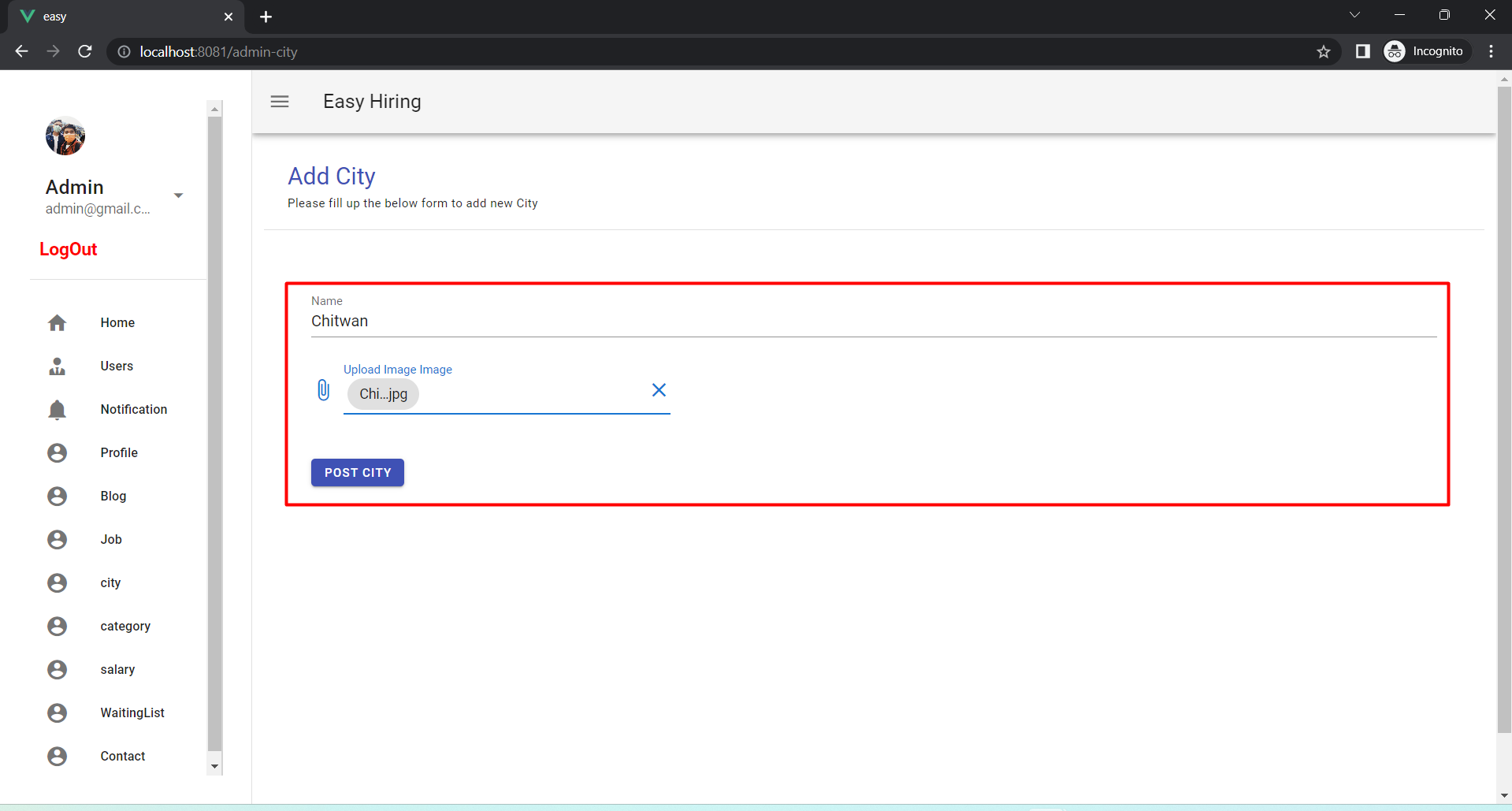
**Test to add location into the system by admin.**

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated



Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, application

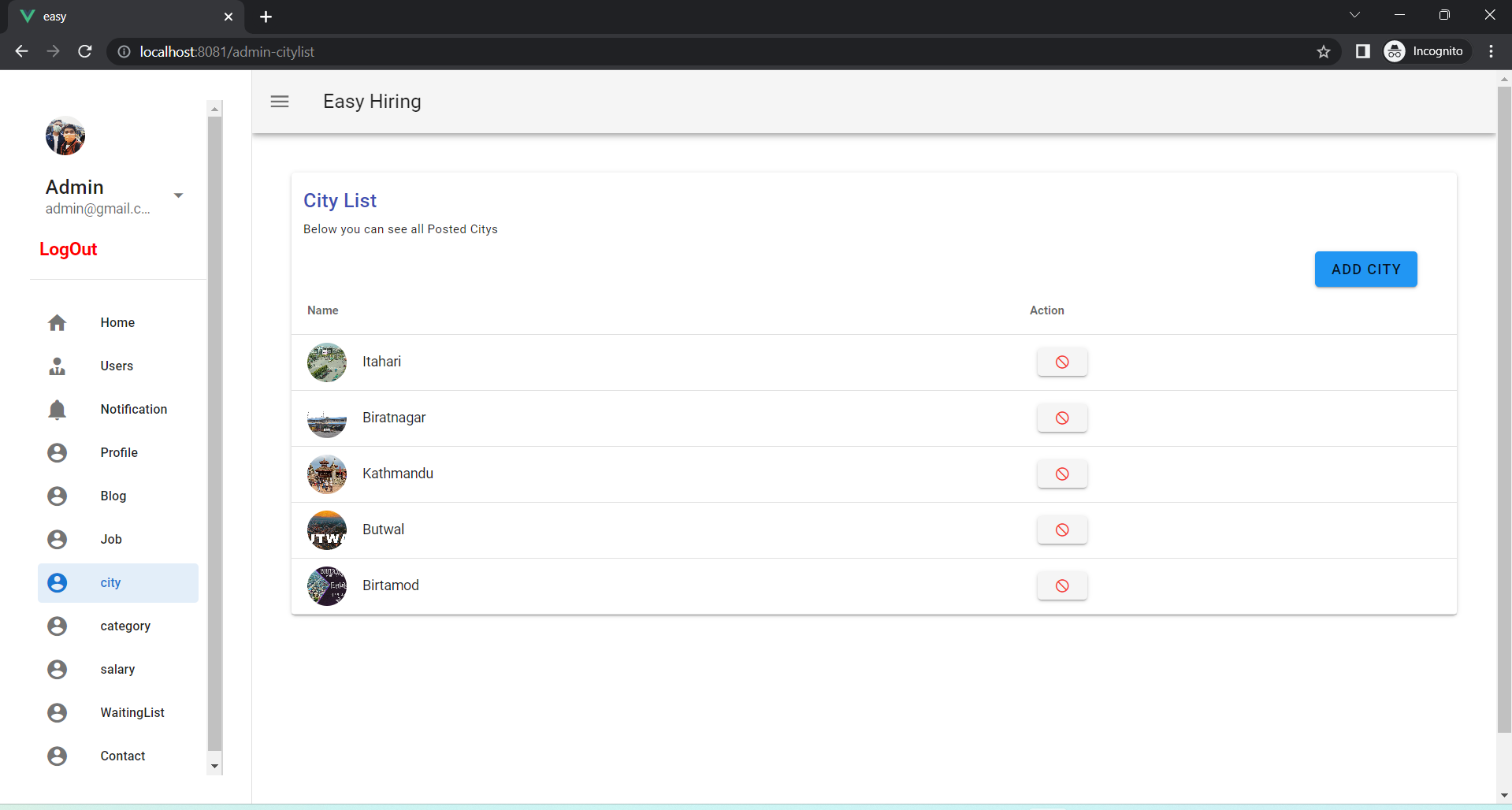
Description automatically generated

**Test to delete city from City List by admin.**

|  |  |
| --- | --- |
| Objective | To check whether admin can delete city from city list into the system or not. |
| Action | Open the dashboard of admin panel and click on city. After that, click on delete button. |
| Expected Result | The system should delete the city from the city list. |
| Actual Result | The system deleted the city from the city list. |
| Conclusion | Test Successful. |

Graphical user interface, text, application

Description automatically generated



**Critical Analysis**

As a final-year student, completing my project on "Easy Hire" was a significant milestone for me. This project was aimed at developing a user-friendly web application that would simplify the process of hiring freelancers or workers for clients and simplify the job search process for workers. In this critical analysis, I will provide a detailed account of my journey in completing this project.

The first step in the process was to conduct extensive research on the requirements of the project. I analyzed similar web applications available in the market, studied their features and functionalities, and identified the gaps that could be filled through the "Easy Hire" web application. This research helped me in developing a clear understanding of the project requirements and the technologies that would be best suited for the project.

After conducting the research, I chose Vue.js as the frontend framework and Node.js as the backend technology to develop the application. Vue.js is a progressive JavaScript framework used for building user interfaces, while Node.js is a popular backend technology used for building scalable and high-performance web applications. I chose these technologies based on their popularity, compatibility, and ease of use.

Once the technologies were chosen, I began the development process. I followed the Rational Unified Process (RUP) methodology to evaluate the tasks involved in the project. This helped me in breaking down the project into smaller, more manageable tasks, which I then prioritized based on their importance and complexity.

I created a detailed project plan and timeline for the development process. This plan included all the tasks involved in the development process, their deadlines, and the resources required to complete them. This helped me in keeping track of the progress of the project and ensured that I completed the project within the given time frame.

I used Visual Studio Code as my development platform, which provided me with all the necessary tools for the development process. I found it easy to use and very user-friendly. I used various plugins and extensions to make the coding process more efficient and effective.

During the development process, I faced several challenges. One of the main challenges was integrating the frontend and backend parts of the application. This required me to have a deep understanding of the technologies and their interaction with each other. I spent considerable time on this task, but it was worth it as it helped me learn a lot about the technologies and their working.

Another challenge I faced was designing the user interface of the application. I realized that designing a user-friendly and visually appealing interface was critical to the success of the application. I used various design principles and best practices to overcome this challenge, and the final design was well received by the users.

# 4. Analysis of progress

Analysis of progress sections includes the progress to date of the final year project and its development.

## 4.1 Progress Table

|  |  |  |  |
| --- | --- | --- | --- |
| **S.N** | **Task** | **Status** | **Progress(%)** |
| 1. | Topic selection | Completed | 100% |
| 2. | Feasibility Study | Completed | 100% |
| 3. | Cost and time estimation | Completed | 100% |
| 4. | Research on a similar project | Completed | 100% |
| 5. | Finalize proposal | Completed | 100% |
| 6. | Conduct online survey | Completed | 100% |
| 7. | Design Logo for the project | Completed | 100% |
| 8. | Develop a use case diagram of the overall system | Completed | 100% |
| 9. | Develop SRS Document | Completed | 100% |
| 10. | Develop a High-Level use case diagram | Completed | 100% |
| 11. | Develop Wireframe | Completed | 100% |
| 12. | Finalized Initial ERD | Completed | 100% |
| 13. | System architecture diagram development | Completed | 100% |
| 14. | Development of application | Not completed | 5% |
| 15. | Finalize Interim Report | Not completed | 0% |
| 16. | Prepare test cases and testing. | Not completed | 0% |
| 17. | Finalize fyp report | Not completed | 0% |
| 18. | Review and Refinement | Not completed | 0% |
| 19. | Final fyp submission | Not completed | 0% |

*Table 15:Progress table*

## 4.2 Progress Review

The topic for the final year project was chosen in the early stages, and a feasibility study was performed to see whether the concept was practical. A pre-survey was undertaken to compile the project requirements. An online Google form was used for the pre-survey, which was intended to find out the exact issue they were having and the potential value of my solution. Cost estimates and calculations were made based on the requirements. In addition, research was conducted on related projects to pinpoint the qualities they have, the ones they didn't, and the biggest problems.

I find it quite helpful to determine the features of the current project by doing a pre-survey and researching like projects. Following the identification of all the project's players, a use case was created, a wireframe was created, and an SRS document was started. A high-

Level Use Case was developed following the project supervisor’s feedback on the use case. And project Ui development was started.

## 4.3 Progress Timeline

According to the Gantt Chart that was included in the proposal, the project's work has been completed on schedule.

# 5. Future work

Most RUP development lifecycle phases, such as inception and elaboration, are completed on the project's first day. According to the progress report, the following tasks must be done in the next iteration of the development lifecycle:

1. **Application development.**

The remaining front-end coding and back-end development need to be completed in the estimated time in this phase.

1. **Prepared test cases and testing.**

Test cases are created for system testing to determine whether or not the built system is error-free. If an error or problem is discovered during testing, the system bug or error will be resolved by developers in order to make the produced system error-free.

1. **FYP report finalization.**

The FYP report will be finalized in accordance with the Gantt Chart. While completing the FYP report, the Use Case diagram, ERD, System architecture diagram, collaboration diagram, test cases, and so on will be completed.

1. **Review and refinement of project report and submission.**

The FYP report will be modified in response to supervisor feedback in order to improve it and create well-formatted documentation. The final year project will be submitted in by the set deadline. Which contained files and folders such as the Project folder and Development folder, as well as the final documentation report in pdf format. Evidence of the project is kept in the project folder and may contain screenshots from stack overflow, journal publications, and other sources. The development of web application (Easy in learning) code is included in the development folder.

# 6.References

Fran. (2021). *What are HTML and CSS used for?* Retrieved from Future Learn: https://www.futurelearn.com/info/blog/what-are-html-css-basics-of-coding

javaTpoint. (2022). *XAMPP TUTORIAL*. Retrieved from

https://www.javatpoint.com/xampp

Martin Heller. (2022). *What is Visual Studio Code? Microsoft’s extensible code editor*. Retrieved from InfoWorld: https://www.infoworld.com/article/3666488/what-isvisual-studio-code-microsofts-extensible-code-editor.html

Master2Teach. (202). *RUP*. Retrieved from Master2Teach:

https://master2teach.com/software-engineering/rational-unified-process-rup/

Master2Teach. (2022). *Rational Unified Process*. Retrieved from Master2teach: https://master2teach.com/software-engineering/rational-unified-process-rup/

Matthew Martin. (2022). *What is Waterfall Model in SDLC? Advantages and Disadvantages*. Retrieved from GURU99: https://www.guru99.com/what-is-sdlcor-waterfall-model.html

mozilla. (2022). *What is JavaScript?* Retrieved 12 15, 2022, from

https://developer.mozilla.org/en-

US/docs/Learn/JavaScript/First\_steps/What\_is\_JavaScript

PROJECTPRACTICAL. (2022). *Advantage and Disadvantages*. Retrieved from https://www.projectpractical.com/advantages-and-disadvantages-of-scrummethodology/

RajKumar. (2020). *Advantages and Disadvantages of using spiral model*. Retrieved from GeeksforGeeks: https://www.geeksforgeeks.org/advantages-and-disadvantagesof-using-spiral-model/

RajKumar. (n.d.). *Advantages and Disadvantages*. Retrieved from

https://www.geeksforgeeks.org/advantages-and-disadvantages-of-using-spiralmodel/

Robert Sheldon. (2022). *WhatIs.com*. Retrieved from TechTarget:

https://www.techtarget.com/whatis/definition/Nodejs#:~:text=js%20(Node)%3F,Node.,to%20learn%20an%20additional%20language.

tutorialspoint. (2022). *javascript-Overview*. Retrieved from

https://www.tutorialspoint.com/javascript/javascript\_overview.htm

Vue.js. (2022). *What is Vue?* Retrieved from https://vuejs.org/guide/introduction.html

# 7. Appendix

**7.1.1. Proect Title: Easy Hire**

**7.1.2. Category: Web application 7.1.3. Introduction**

**7.1.3.1. Purpose:**

Software Requirements Specification (SRS) documents are made to provide specific information on system functionalities and non-functionalities. It outlines how the software should work and what its purposes are. This article lists the features required for a web application. This document goes into great depth on the hardware, software, and other technical requirements.

7.1.3.2. Intended Audiences and Reading Suggestions:

This guide is designed to help system developers create web applications that reduce manual work for users, enable them to hire employees or freelancers based on their needs, and lessen the difficulty of obtaining employment for workers. Clients and employees are the end users of this online application. The development plan for the system, i.e. Easy Hire, will be contained in this paper. Because of this, users may use this document to learn about the features needed in web applications.

**7.1.3.3. Project Scope:**

The scope of this SRS includes the features and functionality of the Easy Hire web application, as well as the constraints and assumptions that apply to its development and use. The features that are required for the developed web application are listed below.

* A streamlined dashboard.
* The plan page where the profession details are displayed with their ratings.
* Display login page where admin, Clients and professionals can login to the system data.
* Display the professional profile after they get registered to the system.
* Display the professional list along with the rate and hour.
* View payment details and feedback from clients.
* Recommend and display the professionals with their nearby availability.
* Notify via email.
* Video and photo uploading of professionals’ recent works

###### 7.1.3.4. Existing System

The present system does not satisfy the problem faced by the people need of student which are very important in their study life. Many people are still using manual ways to write their work done in their daily basis.

###### 7.1.3.5. Proposed System

The main aim of the proposed system is to update the existing system with some useful, needed features and by making the user-friendly, easy-to-use web application. There are many advantages of Easy Hire clients, workers, and admin which are listed below:

**The advantage for the Client.**

1. Client can view a list of available workers, including their ratings, experience, and past work examples.
2. Client can request services from specific Workers, including details about the project and desired timeframe.
3. Client can view a history of payments made to workers, including the amount paid and the date of the transaction.
4. Client can to view feedback left by other clients for workers, including ratings and comments. **The advantage for workers**
5. Workers can create a profile that includes information about their identity, work experience, and skills.
6. Workers can able to view a list of job listings posted by clients, including details about the project and desired skills.
7. Admin can accept or decline requests for services from clients

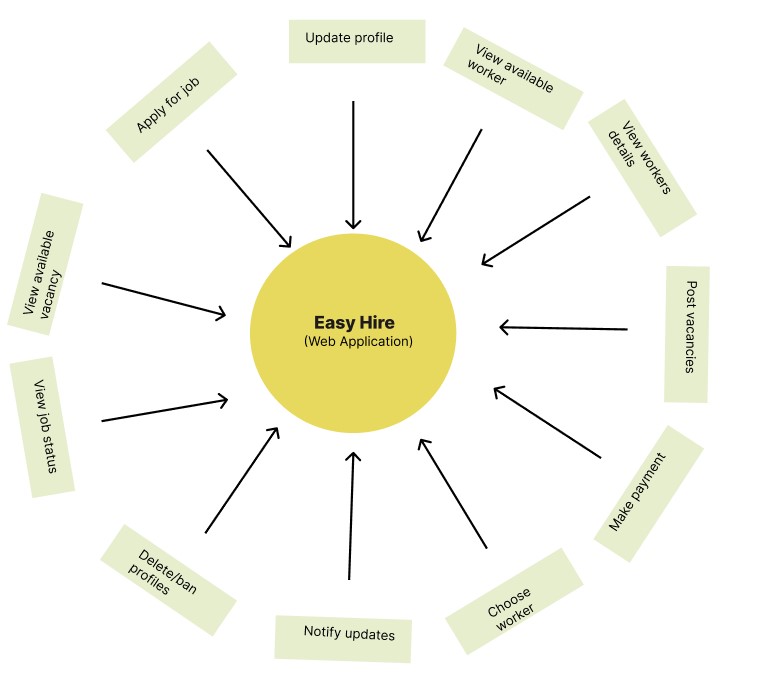
**The advantage of Admin**

1. Admin shall be able to add or remove workers from the system, including their contact information and availability.
2. Admin can view and manage payment history for both clients and Workers including the ability to resolve disputes.
3. Admin able to access and update client and workers information, including contact details and payment preferences min can

.

###### 7.1.4. Overall Description

7.1.4.1. System Perspective



*Figure 17:System Perspective*

7.1.4.2. System Features

Some of the features that should be included in the web application are listed below:

7.1.4.2.1. Register User

The web application will facilitate users(Worker and client) to register to the system by requesting the user’s name, email, password, and other relevant data.

7.1.4.2.2. Update profile

The web application will facilitate the users the (client and the worker) can update their details in the profile section.

7.1.4.2.3. View available worker

The web application will facilitate the user(client) to able to view the availability of the workers to hire them.

7.1.4.2.4. View worker Details

The web application will facilitate the user (client ) to view the details of the workers like; Name, Experience, Ratings, etc to know them before hiring.

7.1.4.2.5. Post vacancies

The web application will facilitate the client to post vacancies for the workers to hire them. **7.1.4.2.6. Choose worker**

The web application will facilitate the user to choose workers among many job applications with their efficiency.

7.1.4.2.7. Make payment

The web application will facilitate the user the client will be able to pay the workers through Hand cash system,

7.1.4.2.8. Apply for job

The web application will facilitate the workers to apply for job or they can upload their job application whenever it is necessary.

7.1.4.2.9. View available vacancy

In the system, The Workers will be able to view the available vacancies and can apply with their interest.

7.1.4.2.10. View job status

The web application will facilitate the workers will be able to view their job status, whether they are selected or not for the jobs they applied.

7.1.4.2.11. Delete/Ban Profiles

The web application will facilitate admin to delete or ban the profiles of both clients and workers at anytime with valid reasons.

**7.1.4.2.12.** Notify updates

The web application will facilitate adminto send notification to clients as well as workers through email and message.

7.1.4.3. User class and characteristics

Easy Hire web applications will be mainly used by the user client and worker and others will be Admin. Basically, there will be two users. According to the users their roles are listed below:

**UC1: Users (Client)**

**UC1.1**

Can log in or register to the system.

**UC1.2**

Can view available workers and their details.

**UC1.3**

Can choose workers according to the ratings.

**UC1.4**

Can post vacancies to hire workers.

**UC1.5**

Can make payment.

**UC1.6**

Can update their profile whenever needed.

**UC2: Admin.**

**UC2.1**

Can log in to the system.

**UC2.2**

Can delete/ban the profiles of workers as well as clients.

**UC2.3**

Can send notifications about the system via email.

**UC3: Workers.**

**UC3.1**

Can log in or register to the system.

**UC3.2**

Can view available vacancies.

**UC3.3**

Can apply for the job vacancy according to their need.

**UC3.4**

Can view job status.

**UC3.5**

Can update their profile whenever needed.

###### 7.1.4.4. Operating Environment

**OE1:** In the present context, Easy Hiring is focused on hiring workers for different work on web platforms only. Further planning is to make this web application on the mobile app and run on a different platform.

###### 7.1.4.5. Assumption and Dependencies

**AS1:** The main targeted audience of this application is users(Client, workers).

**AS2:** The software created will be a web application.

**AS3:** Developers and researchers that are interested in expanding this system with new features can access the documents and reports for it.

###### 7.1.4.6. Functional Requirements 7.1.4.6.1. Register user

|  |  |
| --- | --- |
| **Req.ID** | **Requirement Description** |
| **FR.01** | Users(clients, workers) can register through this application.   |  |  | | --- | --- | | **Sys.Req. ID** | **System Requirement** | | SR.01 | The user (student) can view the registration page. | | SR.02 | The System should check whether the Compulsory filling area is entered or not. | | SR.03 | The System should check provided information is valid or not. | | SR.04 | The System Should display a message about whether the user registration is successful or not. | |

*Table 16:Register User*

7.1.4.6.2. Login (Users/admin)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req.ID** | **Requirement Description** | | | |
| **FR.02** | users who have registered for the system and admin should log in to  use this system. | | | |
|  | **Sys.Req. ID** | **System Requirement** |  |
| SR.05 | The users can view the login page. |
| SR.06 | The System should check whether the Compulsory filling area is entered or not. |
| SR.07 | The System should check provided information is valid or not. |
| SR.08 | The System Should display a message about whether the login is successful or not. |
| SR.09 | If the user entered details are valid then verify the user. |
| SR.10 | If the user is a client, workers then display their dashboard otherwise the admin dashboard should be displayed. |

*Table 17:Login*

7.1.4.6.3. Update Profile

|  |  |
| --- | --- |
| **Req.ID** | **Requirement Description** |
| **FR.03** | users who have an account in this web app can update their profile  details if needed.   |  |  | | --- | --- | | **Sys.Req. ID** | **System Requirement** | | SR.11 | Users should be able to see their profiles. | | SR.12 | A user should be able to edit their information details. | | SR.13 | The System should make a change and display. | |

*Table 18:Update Profile*

7.1.4.6.4. View available worker

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req.ID** | **Requirement Description** | | | |
| **FR.04** | Clients can view available workers for their needed work and hire them. | | | |
|  | **Sys.Req. ID** | **System Requirement** |  |
| SR.14 | When the client clicks on the available worker’s button then the system shall display the list of available workers. |
| SR.15 | If the client wants to view the workers details, then the system shall show the worker’s details list. |
| SR.16 | Client should view the list of available workers with their details. |

*Table 19:View available worker*

**7.1.4.6.5. Post vacancies.**

|  |  |
| --- | --- |
| **Req.ID** | **Requirement Description** |
| **FR.05** | The client can post vacancies for the workers to hire them.   |  |  | | --- | --- | | **Sys.Req. ID** | **System Requirement** | | SR.17 | When the client clicks on post vacancy then the system shall show the vacancy form to create. | | SR.18 | When the vacancy form is created the system should post the vacancy form. | | SR.19 | The client should view the vacancy form they created and edit them if needed. | |

*Table 20:Post vacancies*

7.1.4.6.6. Choose workers

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req.ID** | **Requirement Description** | | | |
| **FR.07** | The client can choose workers among many job applications with their efficiency. | | | |
|  | **Sys.Req. ID** | **System Requirement** |  |
| SR.20 | When the client post the vacancy, the system should show the list of job applications. |
| SR.21 | When the client checks out the worker details with their ratings then the system should display choose button. |
| SR.22 | When the client choose the worker, the system should send notification to the worker via email. |

*Table 21:Choose Workers*

**7.1.4.6.7. Apply for job.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req.ID** | **Requirement Description** | | | |
| **FR.07** | The worker can upload their job application whenever it is necessary and apply for a job. | | | |
|  | **Sys.Req. ID** | **System Requirement** |  |
| SR.23 | The system shall show the job application to the worker. |
| SR.24 | When the worker fills out the job application, the system should show the upload button. |
| SR.25 | When the worker’s job application is accepted, the system should send the notification to the workers via email. |

*Table 22:Apply for Job*

**7.1.4.6.8. Make payment.**

|  |  |
| --- | --- |
| **Req.ID** | **Requirement Description** |
| **FR.08** | The client pay the worker through hand cash system.   |  |  | | --- | --- | | **Sys.Req. ID** | **System Requirement** | | SR.26 | When the client makes payment through the hand cash system, they can press the payment done button. | | SR.27 | The system will verify the payment and store in payment history. | |

*Table 23:Make Payment*

**7.1.4.6.9. View available vacancy.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req.ID** | **Requirement Description** | | | |
| **FR.09** | Workers can view the available vacancy and can apply for the vacancy with their interest. | | | |
|  | **Sys.Req. ID** | **System Requirement** |  |
| SR.28 | When the job vacancy is uploaded the system should display it to the workers . |
| SR.29 | When the system displays the job vacancy the workers can view the available vacancy and apply for it. |
| SR.30 | System should store the data they have submitted |

*Table 24:View Available vacancies*

**7.1.4.6.10. View job Status.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req.ID** | **Requirement Description** | | | |
| **FR.10** | Workers can view their job status, whether they are selected or not for the jobs they applied. | | | |
|  | **Sys.Req. ID** | **System Requirement** |  |
| SR.31 | When the system displays the job vacancy the workers can view the available vacancy and apply for it. |
| SR.32 | When the workers application is accepted or rejected, the system shall show the their job status. |
| SR.33 | When the system shows their job status, the workers can view it. |

*Table 25:View Job Status*

**7.1.4.6.11. Delete/ban profiles.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req.ID** | **Requirement Description** | | | |
| **FR.10** | Admin can able to delete or ban the profiles of both clients and workers at any time with valid reasons. | | | |
|  | **Sys.Req. ID** | **System Requirement** |  |
| SR.34 | When the admin encounter any unnecessary action by client or the workers, admin can delete/ban profile of both (clients, workers). |
| SR.35 | The System should hide the profile of the deleted workers/ clients. |
| SR.36 | The system will store the data of deleted profile. |

*Table 26:Delete/Ban profiles*

**7.1.4.6.10. Notify updates.**

|  |  |
| --- | --- |
| **Req.ID** | **Requirement Description** |
| **FR.10** | Admin can send notifications to clients as well as workers about the updates made in the system through email and messages.   |  |  | | --- | --- | | **Sys.Req. ID** | **System Requirement** | | SR.37 | System should display the notification option when the admin makes any updates in the system. | | SR.38 | The system should send the notification via email. | | SR.39 | Users should view the notification about the changes made in the system. | |

*Table 27:Notify updates*

###### 7.1.4.7. External Interfaces Requirements

**7.1.4.7.1. Hardware.**

This application has been developed to that helps peoples to hire freelancers or workers as per their need and minimize the complexity of workers to find jobs. So, to use this web application they must have their own laptop or desktop computer with an internet connection. Hardware devices are required for this project are:

* Processor: 2.0 GHz or higher.
* Memory: 4 GB or higher.
* Storage: 100 GB or higher.

**7.1.4.7.2. Software.**

Standard resources will be utilized to create and maintain an Easy hire database. For Coding necessitates the development of a web application I required. Similarly, MySQL databases are used to store user data and information that is required by the program.

* HTML5.
* CSS3.
* JavaScript.
* Vuejs.
* Nodejs.
* MySQL.

7.1.4.7.4. Other Non-Functional Requirements

7.1.4.7.4.1. Performance Requirements

|  |  |  |
| --- | --- | --- |
| **Req.ID** | **Requirement Description** | **Priority** |
| **PR.01** | If the user is not logged in or registered, the application should display the login page or registration page initially. | Should |
| **PR.02** | After successfully logging in, the application will show the users' dashboard. | Should |
| **PR.03** | All the features of web applications will be functional, For the users. | Should |

*Table 28:Performance reequirements*

7.1.4.7.4.2. Safety Requirements

|  |  |  |
| --- | --- | --- |
| **Req.ID** | **Requirement Description** | **Priority** |
| **SR.01** | Advertisements should not be displayed while using the application. | Should |
| **SR.02** | The application should be made free from any attacks. | Could. |
| **SR.03** | The application will take permission from users if any personal data of user need to access. | Should. |

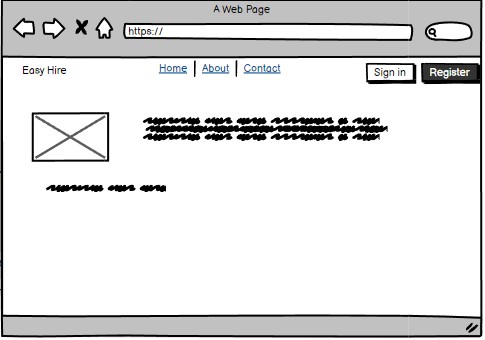
*Table 29:Safety Requirements*

7.1.4.7.4.3. Other Software Quality Attributes

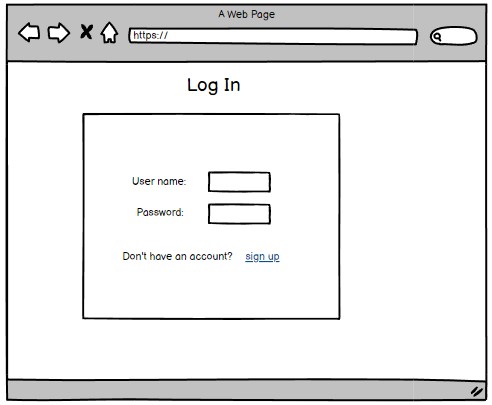
|  |  |  |
| --- | --- | --- |
| **Req.ID** | **Requirement Description** | **Priority** |
| **SQA.01** | The requirements specified for this project must be completed. | Should |
| **SQA.02** | The application should be responsive and easy to operate. | Should |
| **SQA.03** | If needed, new functionality and modern UI might be added to the updated applications. | Could |

*Table 30:Other software quality attributes*

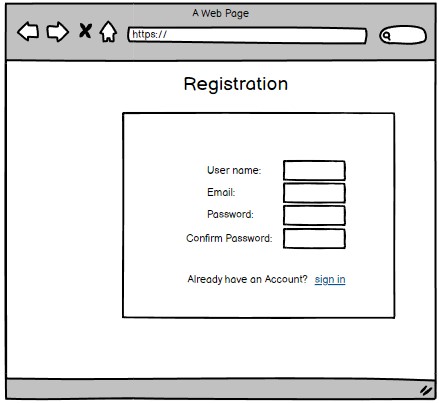
###### 7.2 Wireframes



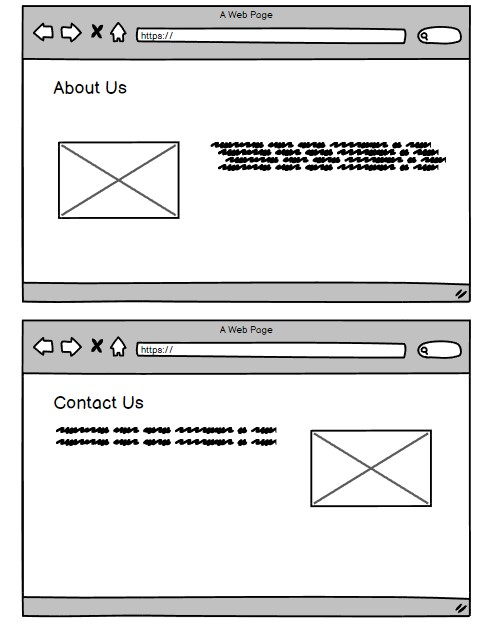
*Figure 18:Home page*



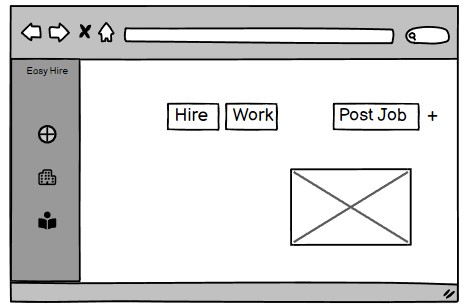
*Figure 19:Login page*



*Figure 20:Registration page*

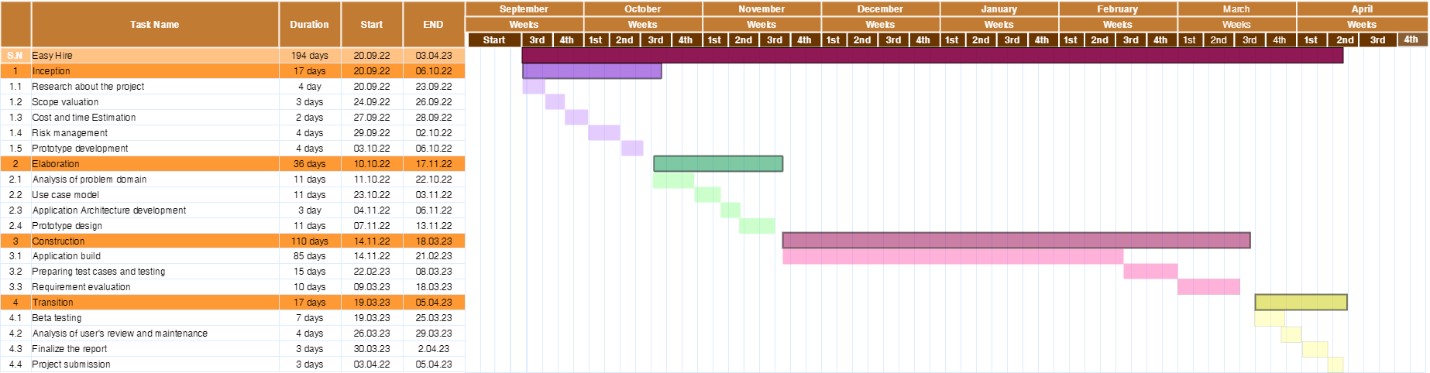


*Figure 21:About us and Contact us page*



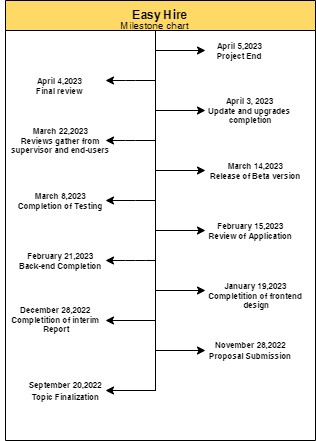
*Figure 22:Dashboard page*

###### 7.3 Revision of project Gantt Chart



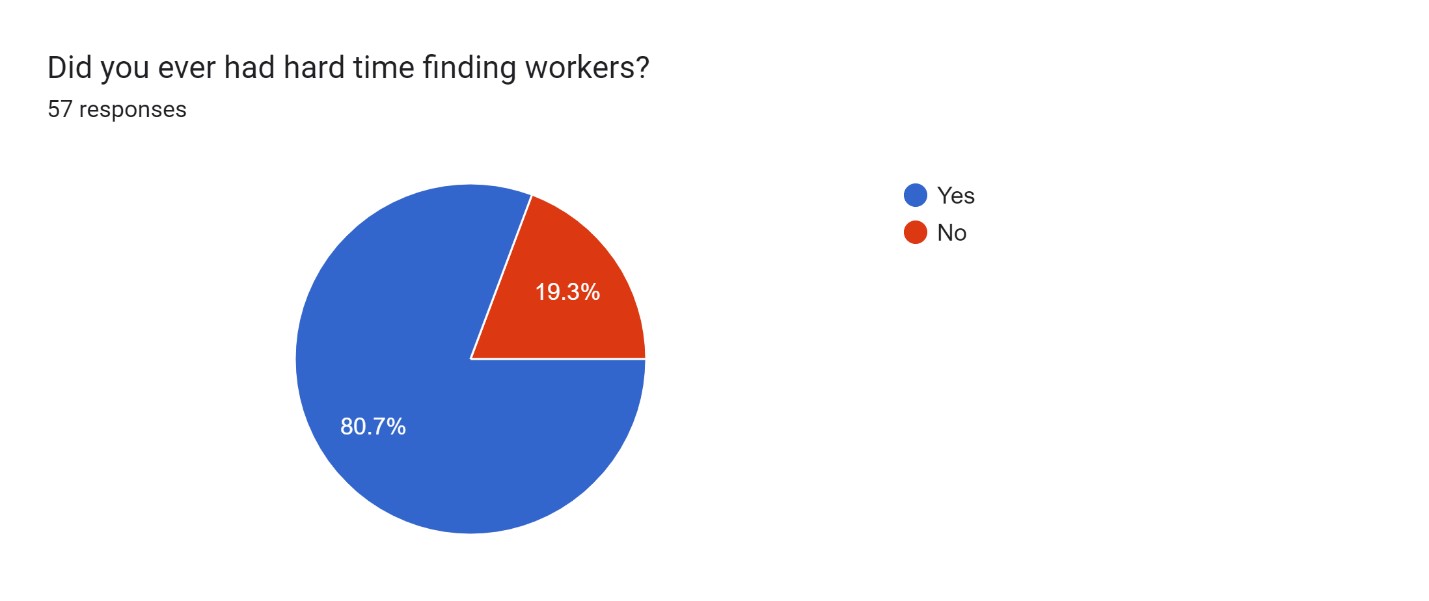
*Figure 23:Gantt Chart*

###### 7.4 Milestone Chart

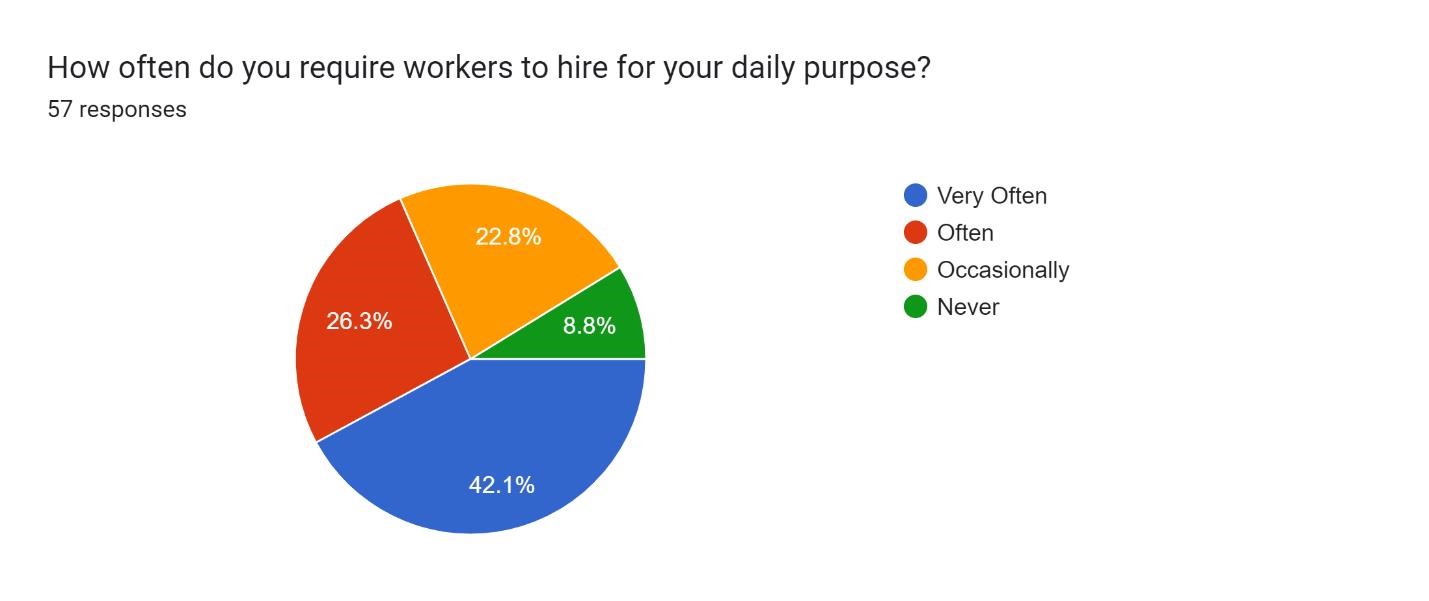


*Figure 24:Milestone Chart*

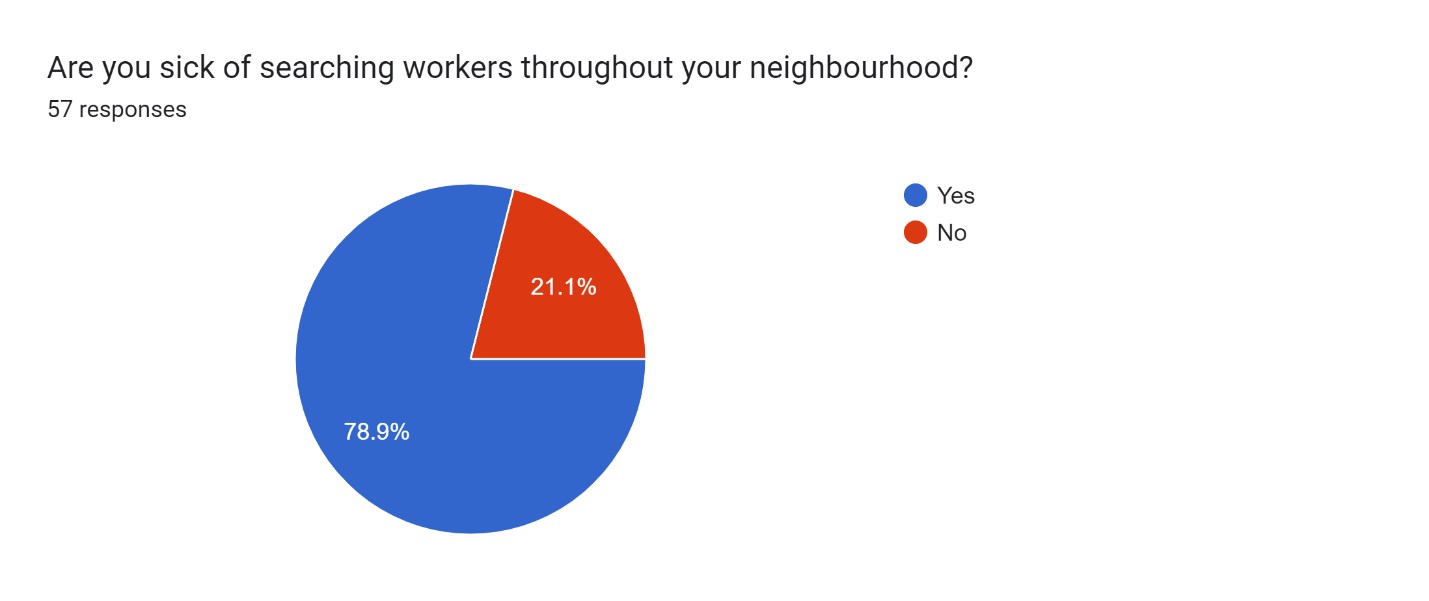
###### 7.5 Survey



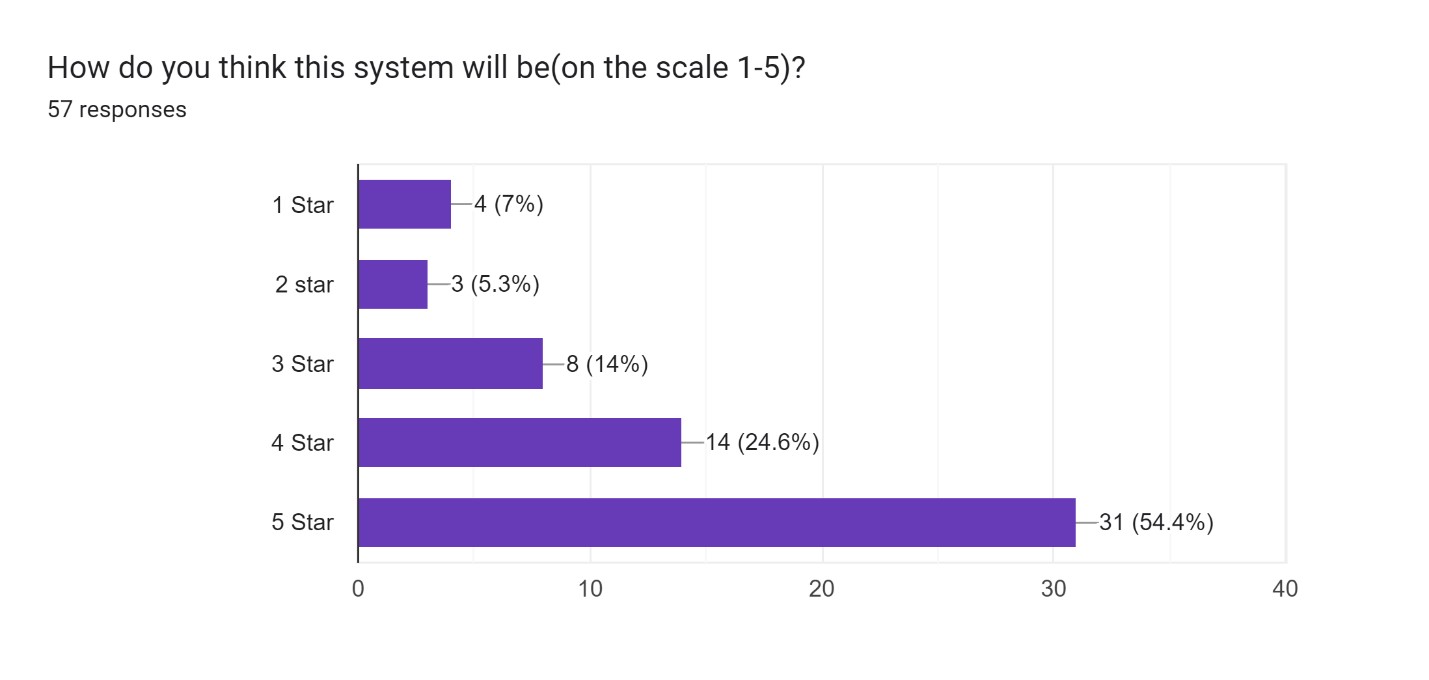
*Figure 25:Survey image 4*



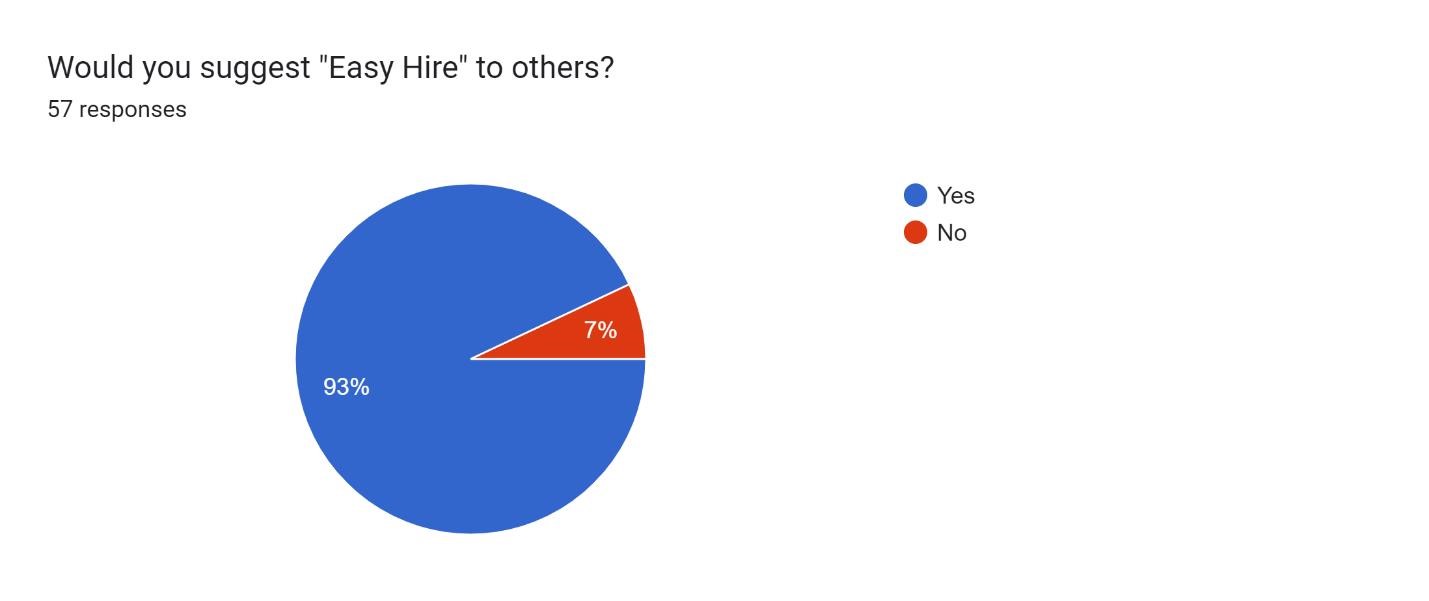
*Figure 26:survey image 5*



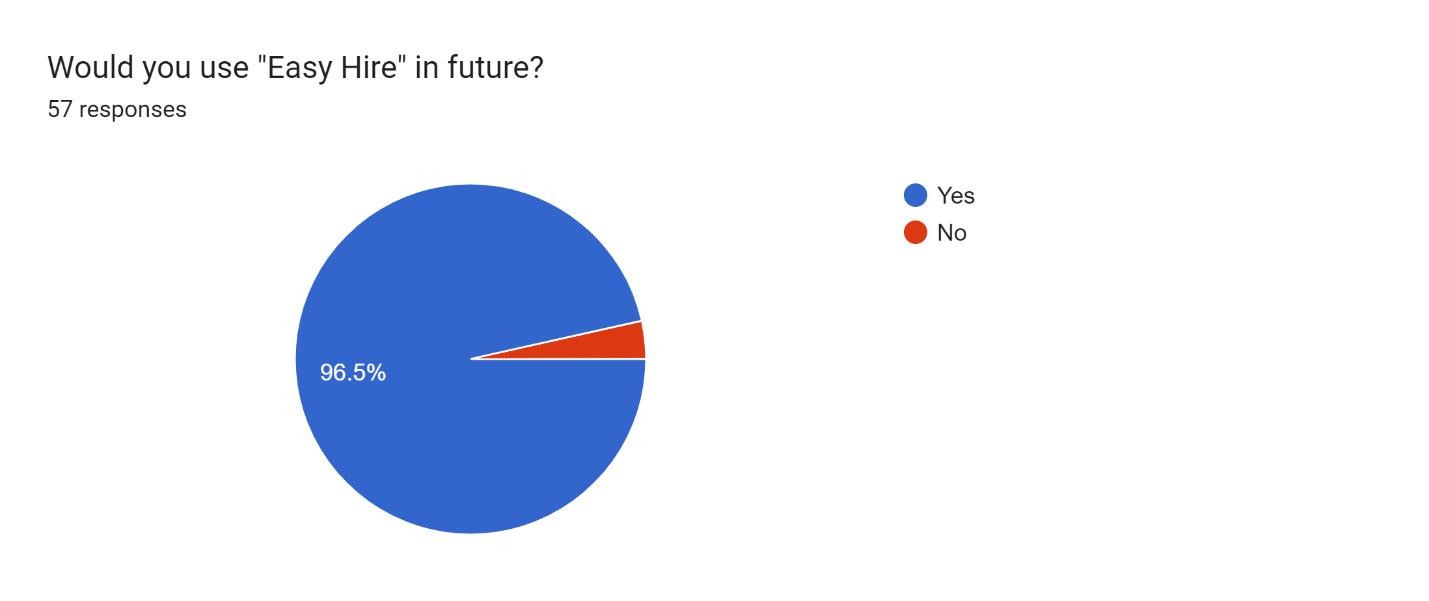
*Figure 27:Survey image 6*



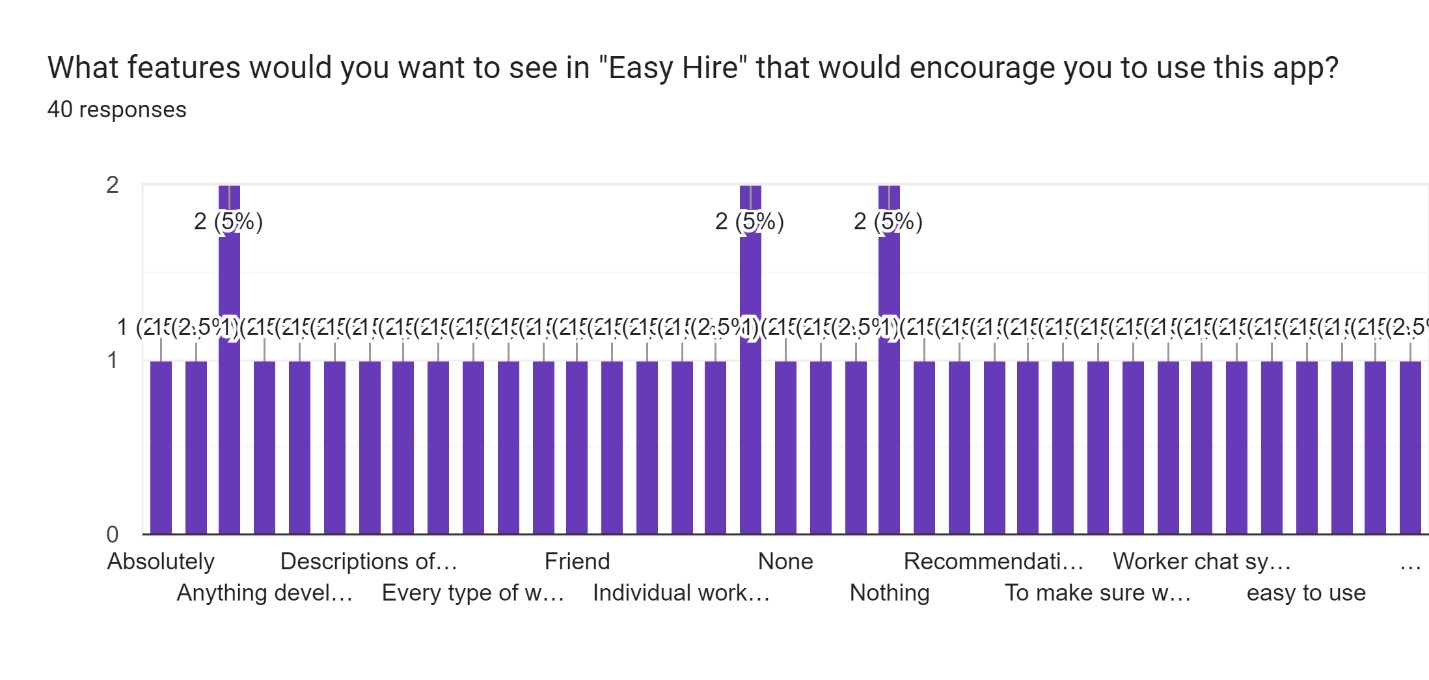
*Figure 28:Survey image 7*



*Figure 29:Survey image 8*



*Figure 30:Survey image 9*



*Figure 31:Survey image 10*



*Figure 32:Suggestions and Feedback from Survey*