TRIBHUVAN UNIVERSITY

Institute of Science and Technology

Shree Siddhanath Science Campus

Mahendranagar, Kanchanpur, Nepal

A PROJECT REPORT

On

"MNRSEWA"

Submitted to

Department of Computer Science and Information Technology

Shree Siddhanath Science Campus, Mahendranagar

In the partial Fulfillment of the Requirement for the Bachelor's Degree in Computer Science and Information Technology

Submitted by

LALIT NATH (16300/074)

NIRAJ BAHADUR PAL (16307/074)

PRADIP MALLA (16311/074)

Submitted Year

2078 B.S. (2022 A.D.)

MNRSEWA

[CSC 412]

A project report submitted for the partial fulfillment of the requirement for the degree of Bachelor of Science in Computer science & Information Technology awarded by Tribhuvan University.

Submitted By

Lalit Nath (16300/074)

Niraj Bahadur Pal (16307/074)

Pradip Malla(16311/074)

Submitted To

Siddhanath Science Campus

Department of Computer science

Affiliated to Tribhuvan University

Bheemdatta, Kanchanpur

SIDDHANATH SCIENCE CAMPUS
Institute of Science & Information Technology

April 2022

Date: .....

SUPERVISOR'S RECOMMENDATION

It is my pleasure to recommend that a report on "**mnrSewa**" has been prepared under my supervision by **Lalit Nath, Niraj Bdr. Pal, Pradip Malla** in partial fulfillment of the requirement of the degree of Bachelor of Science in Computer Science and Information Technology (BSc. CSIT). Their report is satisfactory to process for the future evaluation.

.....

Mr Keshav Datt Bhatta

Supervisor

Department of Computer Science & IT

Siddhanath Science
Campus

CERTIFICATE OF APPROVAL

The undersigned certifies that he has read and recommended to the Department of Computer Science and Information Technology for acceptance of a report entitled "**mnrSewa"** submitted by **Lalit Nath, Niraj Bdr. Pal and Pradip Malla** in partial fulfillment for the degree of Bachelor of Science in Computer Science and Information Technology (BSc.CSIT), Institute of Science and Technology, Tribhuvan University.

Dr Gajaram Damai

Campus Chief

Siddhanath Science Campus

Mr Keshav Datt Bhatt

Supervisor

Department of Computer Science & IT

Siddhanath Science Campus

Head of Department

Department of Computer Science & IT

Siddhanath Science Campus

External Examiner

ACKNOWLEDGEMENT

We would like to express our deepest appreciation to all those who provided us with the possibility to complete this report. A special gratitude we give to our final year project Supervisor, Mr Keshab Datt. Bhatt, whose contribution in stimulating suggestions and encouragement, helped us to coordinate our project especially in writing this report. Furthermore, we would also like to acknowledge with much appreciation the crucial role of the staff of Siddhanath Science Campus, who gave the permission to use all required equipment and the necessary materials to complete the task. We are thankful and fortunate enough to get constant support from our seniors and every teaching staff of B.Sc. CSIT department which helped us successfully complete our project. We would also like to extend our regards to all the non-teaching staff of B.Sc. CSIT department for their timely support. We appreciate the guidance given by other supervisors as well as the panels, especially in our project document which has improved our standard of report. Our thanks and appreciations also go to each and every one of our colleagues for their encouragement and support in developing the project.

With Respect,
Lalit Nath (16300/074)
Niraj Bdr. Pal (16307/074)
Pradip Malla (16311/074)

ABSTRACT

The project ‘mnrSewa’ aims to provide opportunities for hardworking Nepalese and help improve people’s status. The main objective of our project is to provide a platform that acts as a bridge between the service provider and the prospective client for showcasing their skills while empowering consumers by providing them with all the information to help them make an informed decision. mnrSewa provides a platform to connect professionals in Information Technology fields and citizens where citizens like us can find services, products or news/events all in one place.

It is a community-based platform where people connect in order to solve any kind of problem whether it is complex or simple. That is what mnrSewa is, an idea where you can find all that you need in one place, a full stop where all your needs end.

KEYWORDS: Service Provider, Platform, Online portal service, Service Recommendation System, Service Delivery,

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CI/CD: Continuous Integration/ Continuous Delivery

CRM: Customer Relationship Management

CSS: Cascading Style Sheet

DevOps: Development and Operation

FTP: File Transfer Protocol

HTML: Hypertext Markup Language

JS: JavaScript

SaaS: Software as a Service

SDLC: Software Development Lifecycle

TCP: Transmission Control Protocol

UDP: User Defined Protocol

UML: Unified Modeling Language

CHAPTER 1
INTRODUCTION

1.1. Background

The concept of Service providers using Information Technology has been around for quite a while, but it has continuously failed to meet customer expectations. Over time, improvements to the internet are leading platforms to lift many of the early workflow limitations and make scalability more possible. It has made online business processes faster and more reliable. Cost-efficiency, ease of use, and improvements to core functionality all led to exponential growth in Software as a Service (SaaS). Today, it's a practical option even for enterprise-level businesses.

The Project entitled "MNRSEWA" is a web application that allows a Web-based platform with rich features for service providers between the local expert in their fields and the citizens. The users as well as professionals (service providers) who wish to use this application, can register for the website and get the User ID and password. Whereas service providers can register as vendors (Freelancer, Company) to show their fields of services and their details. It also allows the users to find all types of services to their needs with other users at the same time. It provides the user with up-to-date information about the availability and schedule of any professional service provider experts. The project will be initially focused on a non-payment basis and later integrate a payment system too.

By using this project people can ask for services from professionals so that they can get rid of their problems. In MnrSewa`s platform, everything is done remotely so people can connect with professional service providers from any part of the world virtually.

MNRSEWA is a service provider model in which services like IT service, Electrician, Plumber, AC Cleaning & Repairing, Delivery System, Photographer, etc are provided over the internet, and makes them available to end-users over the internet.

In this project, a client can register to the application and use services based on his personal or enterprise needs. The subscription plans include freemium. Once one is registered, a user can search for his needs to the specific problem and only need to pay a charge of the service

provider vendor. As a role of service provider, a vendor can register and add the specifications about their ways of dealing with the problems. It provides flexibility to search for an effective solution for a problem as in review based, nearer one service or a cheaper one in terms of cost. Some of the features in the application will be authentication, lead registration, real-time service tracking, and invoicing and billing. The user can set some pre-information about their field of interest to get those field services as recommendations. For example, a user may choose some field of interest like IT services, Electrician, Plumber then He/She may get most specific recommendations about IT services, Electrician, Plumber.

This project will be built using the general-purpose language Laravel and its open-source PHP framework, which is robust, easy to understand, and easy to use. It follows a model-view-controller design pattern. HTML, CSS, Bootstrap 4, and JavaScript is the set of technologies considered in developing the user interface of our web application. The project will be hosted using the available cloud platform, and computing resources.

1.2. Problem Definition

In the current time all applications for every daily life can be found. But still, now people have to look or call for a local electrician, plumber, carpenter, painter, visit the hospital for nursing care, Software companies for IT services, etc., manually. Sometimes people have to spend the whole day to find a person who can solve small household problems like electricity problems, door problems, water problems, etc., Not every time people can find the person to solve the problem.

"**mnrSewa**?" is the name of the application which is designed and proposed to solve the current problem in searching for the man who can solve household and professional office problems.

As we know that Nepal is a developing country, so to contribute to the field of development we IT students have an idea to provide an online service delivery platform to improve the standard lives of people. So we started from our own city 'Mahendranagar' to make a step in the digital world.

Therefore, all these above problems can be accommodated through the use of the platform we wish to create "**mnrSewa**".

1.3. Objectives

The main aim of the application is to create a web application that is the medium for household and professional workers like carpenters, painters, electricians, plumbers, IT professionals, nurses, teachers, photographers, vehicle repair, etc., and the people for easily hiring the people when people face any problems in our daily lives as a household and professional office activities.

The project has aimed to fulfil the following objectives:

- To deliver on-demand services as user required and service provider tracking, and searching best effective solution provider through applications over the internet.
- To track nearer, cost-effective, and best-reviewed service providers to get their job done or problem solved.
- Establishing secure communication between the small service provider and the people i.e It's a platform with multiple vendors on it. Some employment opportunities for those who can solve electric, plumbing, problems,
- Detects the location and serves the nearest service provider to customers,
- To design a system that enables Review System so that user can get the best-reviewed service,
- This application is a multi-language system, as many local people are not perfect in English so they can easily access the application on native language i.e. Nepali.

1.4. Scope and Limitation

1.4.1. Scope

This project has scope within two targeted groups. One is a platform for business enterprises or service providers to increase their customers by online portal efficiently, and another group is the members or users who want to get service the best solution for their problems. The application can create a bridge between the users living in remote as well as urban areas to get a solution to problems with the professional service provider. MNRSEWA can help people solve household and office problems, track effective service providers, and solve any issues by querying the service provider. Both parties can easily

communicate with each other to understand the actual problem and get rid of it. Users can add and search services according to their needs.

The vendor or service provider works effectively to serve people through their experienced team members, create healthy relationships with customers, and finally assign tasks to the joined members.

On the other hand, the user meets their goals and objectives of solving their problem in a given set of time while following its guidelines.

1.4.2. Limitation

- mnrSewa can't support in all parts of Nepal and abroad yet

1.5. Development Methodology

The project follows the professional Agile project execution methodology to build a SaaS application.

Agile is an iterative approach to project management and software development that helps us

deliver products effectively. Instead of being overwhelmed by the whole system in a single take, we develop a single module in an effective way. Requirements, plans, and results are evaluated continuously by our team so that we have a chance to respond to our changes beforehand.

Agile Modeling (AM) is a practice-based methodology for modelling and documentation of software-based systems. It is intended to be a collection of values, principles, and practices for modeling software that can be applied to a software development project in a more flexible manner than traditional modeling methods.[1]

The steps of the Agile approach to MNRSEWA:

- **Set Requirements**: Surveys, User Observation, Document Analysis,
- **Design**: Executives are ready to support UX designers, Cross-functional, Research and testing,
- **Development**: code development and code review, source code management tools, code merging
- **Testing**: continuous testing tools that provide feedback on business risks, determine performance

- **Deployment**: code is deployed to a QA or testing environment that is accessible to specific users and as close as possible to a real-world environment. That way, users can continuously test the software and send it back for improvement,
- **Review**: meetings to review and demonstrate the user stories that the development team completed during the sprint.
- **Delivery**: an iterative approach to software delivery in which teams build software incrementally at the beginning of a project rather than ship it at once upon completion.
- **Feedback**: maintain communication throughout the development process, gather feedback (good and bad) on the app from different teams, identify areas for improvement, and increase developer productivity

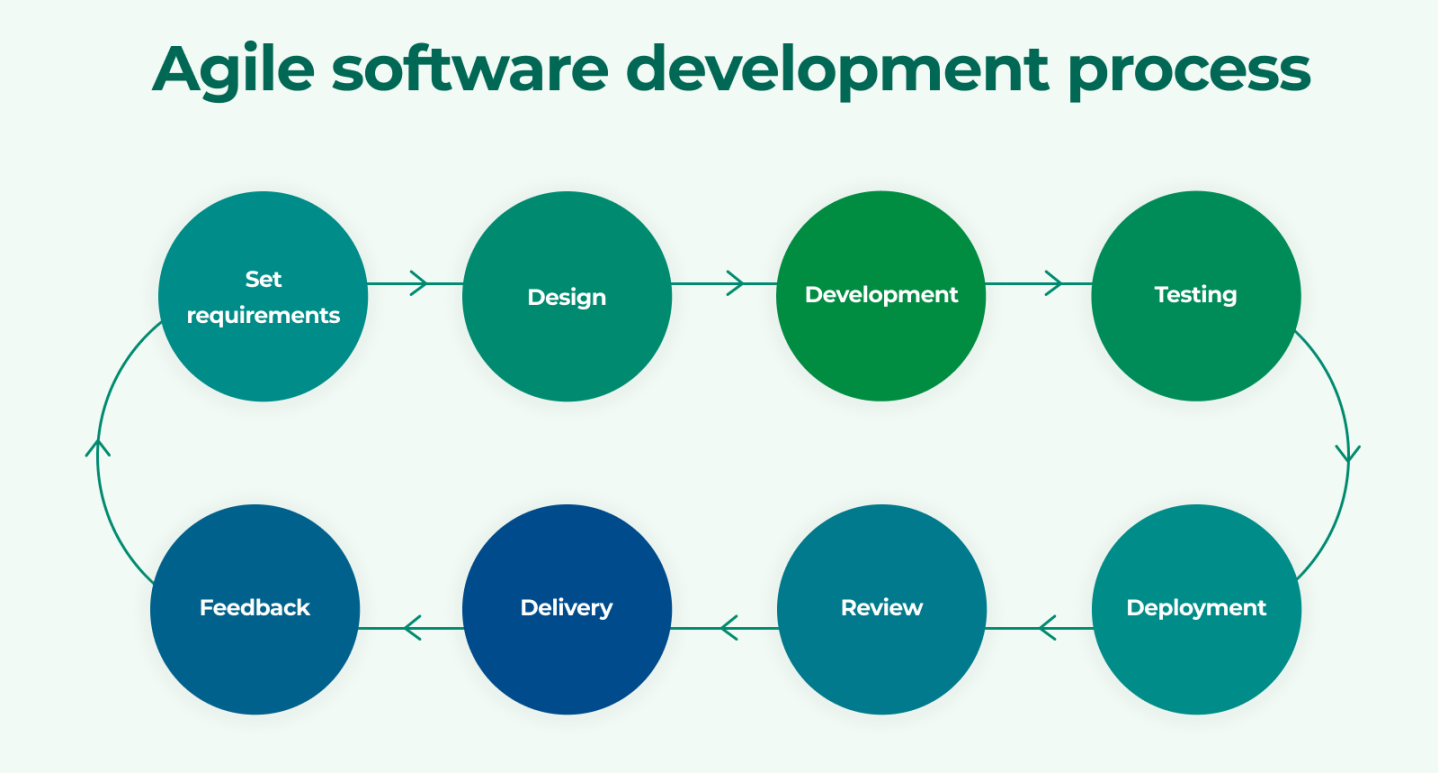


Figure 1: Agile Methodology

1.6. Report Organization

The report on "**MNRSEWA**" is based on six chapters. Each chapter follows the constructive building of this project. Chapter 1 gives an overview idea of our project. It anticipates and combines the main points to be described later in the chapters following. Similarly, chapter 2 usually contains the theoretical literature review. It gives the insight to distinguish the possible hypothesis, strategies, and shortfalls in the current research. Chapter 3 studies the system such that information can be analyzed, modeled and developed. It also gives enough information to replicate the study. It addresses the problems from chapter 1 and explains the objects of each experiment. Chapter 4 contains high-order UML diagrams to give an insight into the system design and algorithm being used while developing the system. Chapter 5 contains system testing, it discusses the execution of a program or system with the intent of finding errors. It also includes the examination of code as well as the execution of that code in various environments and conditions. Chapter 6 describes the significance of "**MNRSEWA**", moreover, discusses the future recommendations applicable to enhance the project.

CHAPTER 2
BACKGROUND AND LITERATURE REVIEW

2.1. Background Study

Now you can find all the applications of any everyday life. But people still have to manually find and call local electricians, plumbers, carpenters, painters, go to hospitals, manually visit IT service software companies, etc. They may need to spend the whole day finding someone who can solve a small household problem, such as an electrical problem, a door problem, or a water problem. Not every time people can find someone to solve the problem.

This project allows customers to register with the application and use the service based on their personal or business needs. Freemium is included in the subscription plan. Once registered, users can search for specific issues according to their needs and only pay the service provider's fee. As a service providers, providers can create and add specifications on how to address issues. It provides the flexibility to find an effective solution to your problem. B. Evaluation-based, close to service or cheap in terms of cost. Some of the application's features are authentication, lead registration, real-time service tracking, and billing and billing. Users can set up preliminary information about their area of interest to receive these field services as recommendations. For example, users can select areas of interest such as IT services, electricians, and plumbers to get the most specific recommendations for IT services, electricians, and plumbers.
"MnrSewa?" Is the name of an application being developed and proposed to solve the current problem of finding a man who can solve a home or professional office problem?

2.2. Literature Review

A web application (or web app) is application software that runs on a web server, unlike computer-based software programs that are run locally on the operating system (OS) of the devices. Web applications are accessed by the user through a web browser with an active network connection. These applications are programmed using a client-server modelled structure—the user ("client") is provided services through an off-site server that is hosted by a third party. [2]

Web app development is rapidly growing. From retail, telecommunications, and e-commerce to insurance, healthcare, and government, organizations across industries must meet user expectations for real-time, convenient ways to conduct transactions and access information. Today, the web applications that unlock their value—are the most popular way for people and businesses to connect to the internet. To stay relevant, responsive, and successful, organizations need to develop the web applications that their customers, partners, and employees demand.
Yet web application development might seem daunting. Fortunately, by following a few basic guidelines and best practices, you can streamline your application development journey. [3]

MNRSEWA will help to overcome the above-stated problem which states the difficulties that people have faced in having the desired service with professionals in the desired domain. For this project, we researched various platforms such as social media and other talks with citizens. We found that they were not satisfied with the traditional problem-solving methods. To overcome this, we wished to provide a platform that connects people with expertise in their preferred domain.

Similarly, there is a usage amount of research and study on the framework, language, and tools that might be suitable to develop our platform. Since our application is UI heavy we found that Bootstrap and JS library perfect for our use case. Bootstrap is a JavaScript library for building user interfaces and it has a rich ecosystem of libraries that will be easier to integrate into our project. Similarly, the project is done in PHP/Laravel on the backend sense that it is one of the most effective tech stacks supporting web apps is Laravel Framework.

Also, mnrsewa uses some algorithms for the recommendation of services to the people. The recommendation would be based on the user's interest or other activities.

CHAPTER 3

SYSTEM ANALYSIS

3.1. System Analysis

3.1.1. Requirement Analysis

Requirement Analysis We've analyzed and validated the requirements and recorded and monitored the implementation throughout the project. mnrSewa have used the Customer Journey Mapping analysis technique which studies the business and its goals and how this gap can be closed. It also helped us to understand the customers' motivation, fears, and objections.

i. Functional Requirements

The system should help clients/vendors find new customers, win their business, and keep them happy by serving customer and prospect information and also enabling an online portal for users who are seeking the best solution for a problem or wants to get a better service nearby without visiting a physical workshop. Secure registration and profile management facilities for different users. The system should also provide details like vendor details, facilities, security, expertise, experiences of service providers, reviews, etc. to the local user. Generate a report about the registered thread to the admin and a response report to the user who has submitted his queries

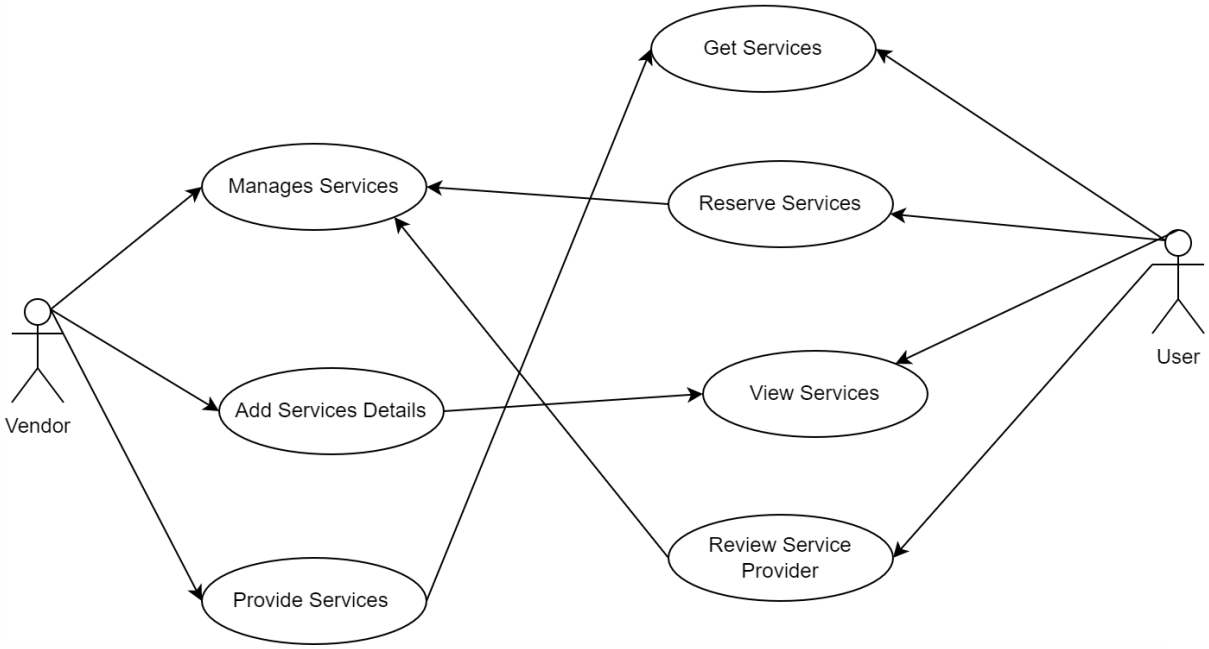


Figure 2: Use Case Diagram

ii. Non Functional Requirements

Scalability: SaaS is an on-demand business model, it's extremely difficult to predict the load on the system. This is where our system architects will have to leverage the cloud model to take advantage of the on-demand resource consumption model. Performance: Regardless of the type of application or the volume of processing that happens behind the screens. This application architect will have to consciously take into consideration the potential performance bottlenecks and implement designs that can help leverage concepts like asynchronous processing.

Availability: Availability of the application is the biggest concern, particularly if the application addresses a business-critical solution. Unplanned downtime of SaaS applications can lead to heavy loss for Customers and consequently can ruin the SaaS provider's business. Thus, this project should consider recovery techniques while designing the strategy. Auditing: From a SaaS provider perspective, the ownership of the system and its functioning is with the provider. Therefore, it's our responsibility to implement the appropriate measures to track the usage of the system and the events happening within it.

Load Balancing: SaaS applications require high computing

3.1.2. Feasibility Analysis

i. Technical Feasibility

The technical feasibility analysis is based on gaining an understanding of the existing technical resources and their applicability to the proposed system's planned needs. All the tools and software products required are easily available online. It needs an easily affordable Web server and Multi-Cloud architecture to develop. However, SaaS products are a complex undertaking. We need to figure out how all the moving parts interact to become a fully functional body.

Table 1: Hardware Requirements

Hardware Requirements	
System Architecture	Any standard x86 and x64 bit computer
Memory	4 GB RAM
Storage and Type	Minimum 1 GB free HDD/SSD space.

Table 2: Software Requirements

Software Requirements	
Web Browsers	Any Modern Browsers: i. Chrome - Latest stable release ii. Safari - Latest stable release ii. Firefox - Latest stable release iv. Edge - Latest stable release v. Opera- Latest stable release

Mobile Platforms	i. iOS- Version 10 or higher ii. Android- Version 5.0 or higher
Bandwidth Speed Requirements	12 Mbps (Internet Speed Requirement)
IOPS	55-180 IOPS

ii. Operational

This project will allow any user to use the website and search for the best services without the need to visit a physical store. Any naïve users can operate this website with bare minimum knowledge and no special kind of training is required. The system will operate over the internet thus making the user available with the best service provider (vendor). Hence this project is operationally feasible.

iii. Economic

SaaS app development is a complicated process. There are certain pitfalls we may face along the way. These challenges can increase SaaS development costs considerably if we won't address them on time.

Maintaining a comprehensive cloud, which would be important as the number of users continues to increase. Also, the details of the old customers and frequent subscribers need to be stored. A Heroku Postgres would be economical. We will be using the open-source framework to develop **MNRSEWA**, making the cost-focused only on cloud storage and hosting. After the application is live, the majority of our revenue will be generated from users paying for the software subscription.

Moreover, we are continuously aiming at identifying, exploring, and evaluating the solution of the project to save money and time.

iv. Schedule

Table 3: Gantt Chart

Activities	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12
Requirement Gathering and Analysis												
Design												
Implementation and Coding												
Testing												
Deployment												
Report Preparation												

3.1.3. Analysis

3.1.3.1. Class and Object Diagram

The class and object diagram visualizes the particular working functionality of the system. The UML diagram of the MNRSEWA only depicts the abstract view of the diagram where the object represents the static view whereas class represents the behavior of those objects.

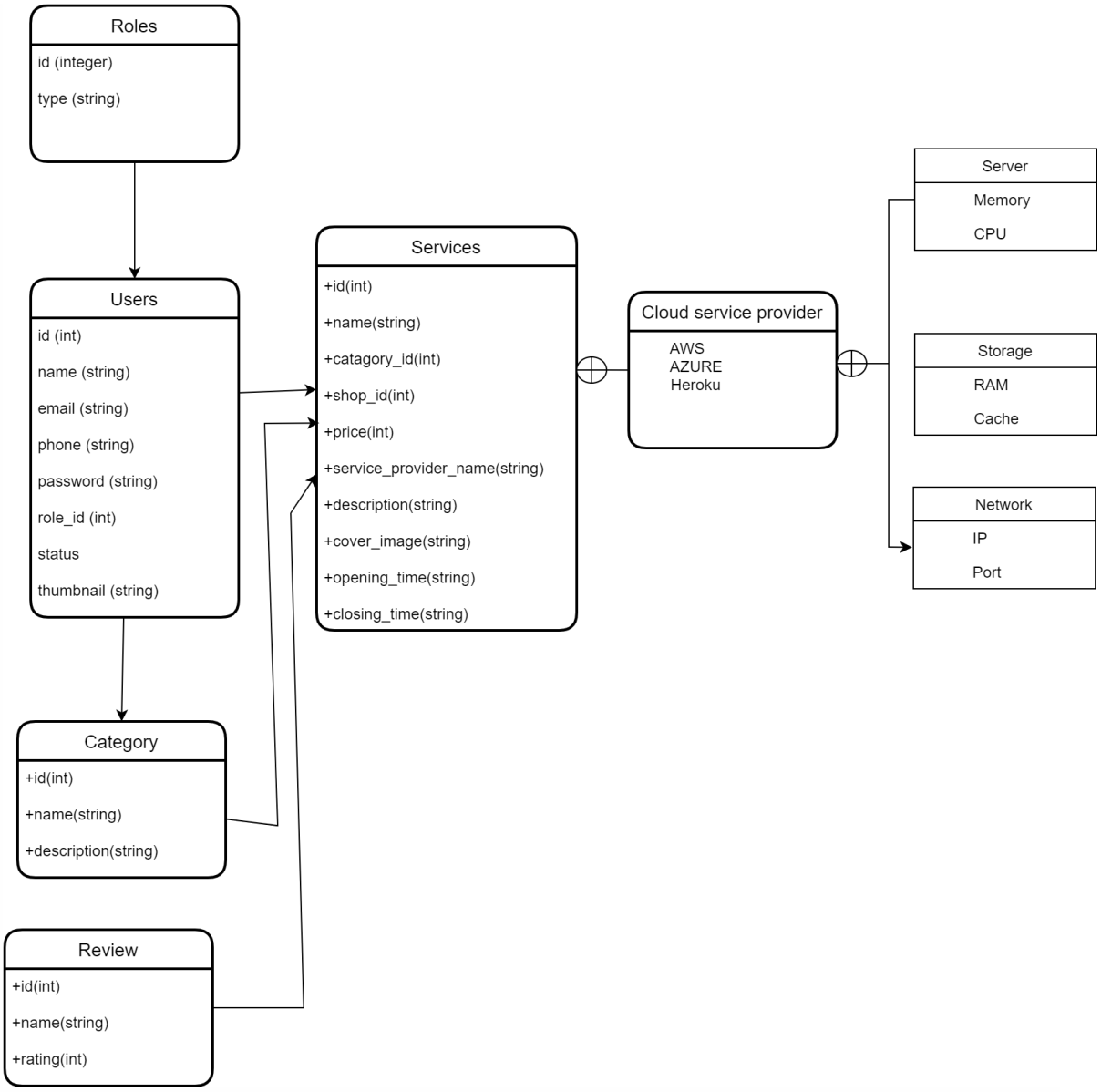


Figure 3: Class Diagram

3.1.3.2. State and Sequence Diagram

Sequence diagram shows how objects in the system or classes within the code interact with each other. Particularly these diagrams show diagrams in order. The sequence diagram used in this system helps to understand the existing and the requirements of the new features and applications.

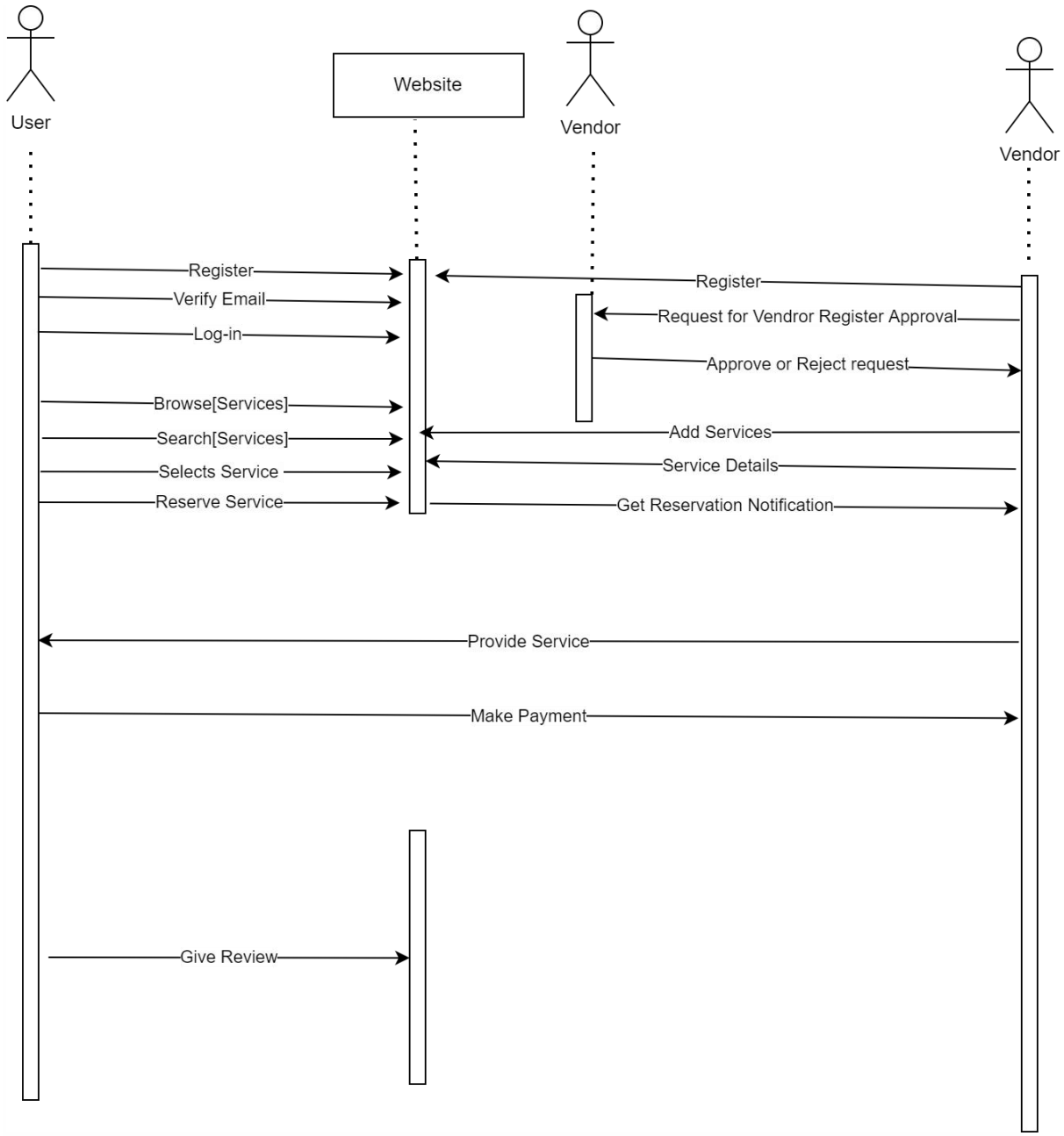


Figure 4: Sequence Diagram

3.1.3.3. Activity Diagram

The activity diagram shows the dynamic view of the system. Moreover, it is also represented as the flowchart which represents the flow of control among the activities in the system. In **MNRSEWA**, the activity diagram shows the operation of the services, plans, information, and joining processes in the diagrammatic form.

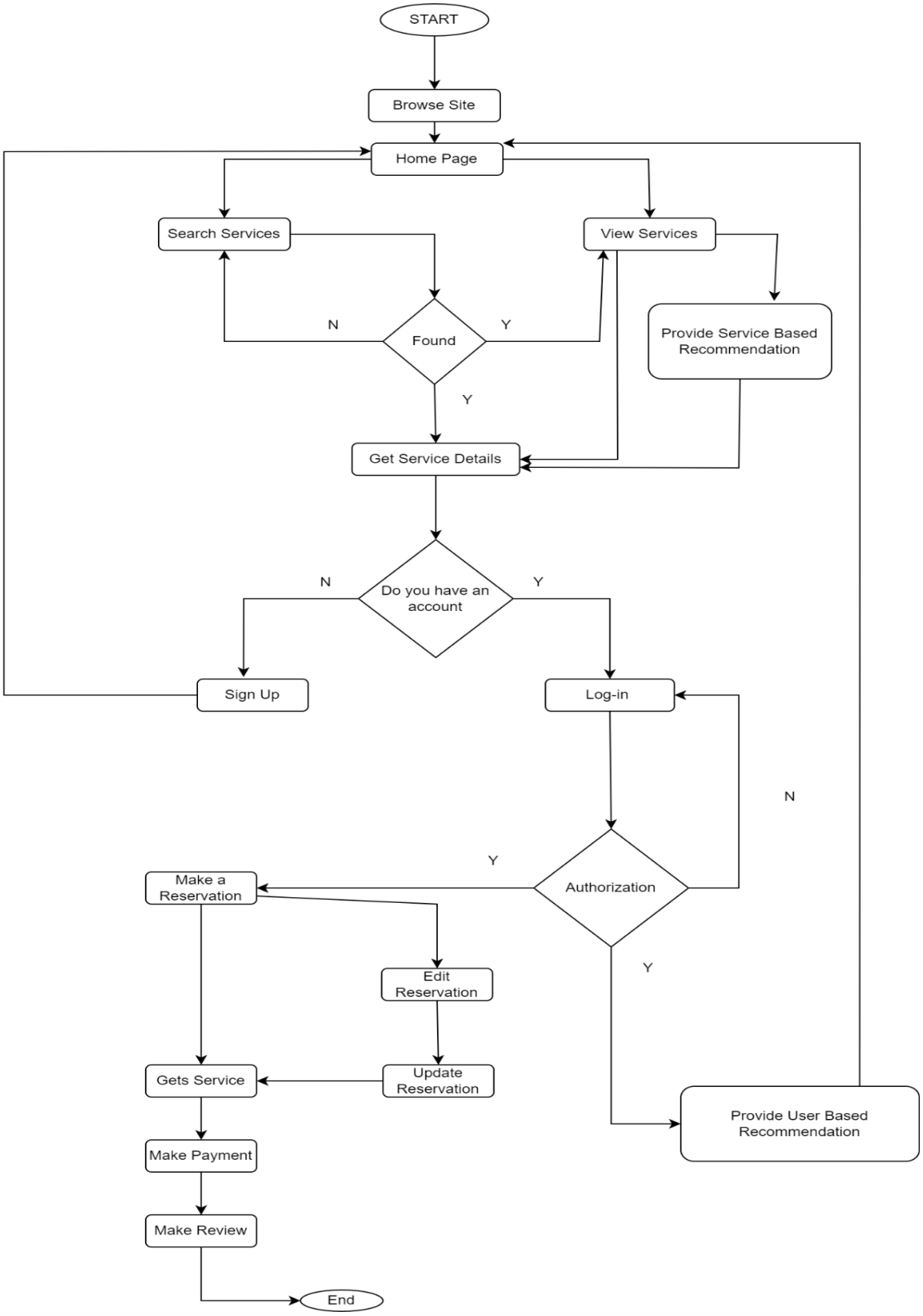


Figure 5: Activity Diagram

CHAPTER 4

SYSTEM DESIGN

4.1. Design

4.1.1. Refinement of Class, Object, Sequence, and Activity Diagrams

The refinement class diagram depicts the shallow description of each important module used in web app development. These refinements further help in enhancing and smoothing the overall workflow.

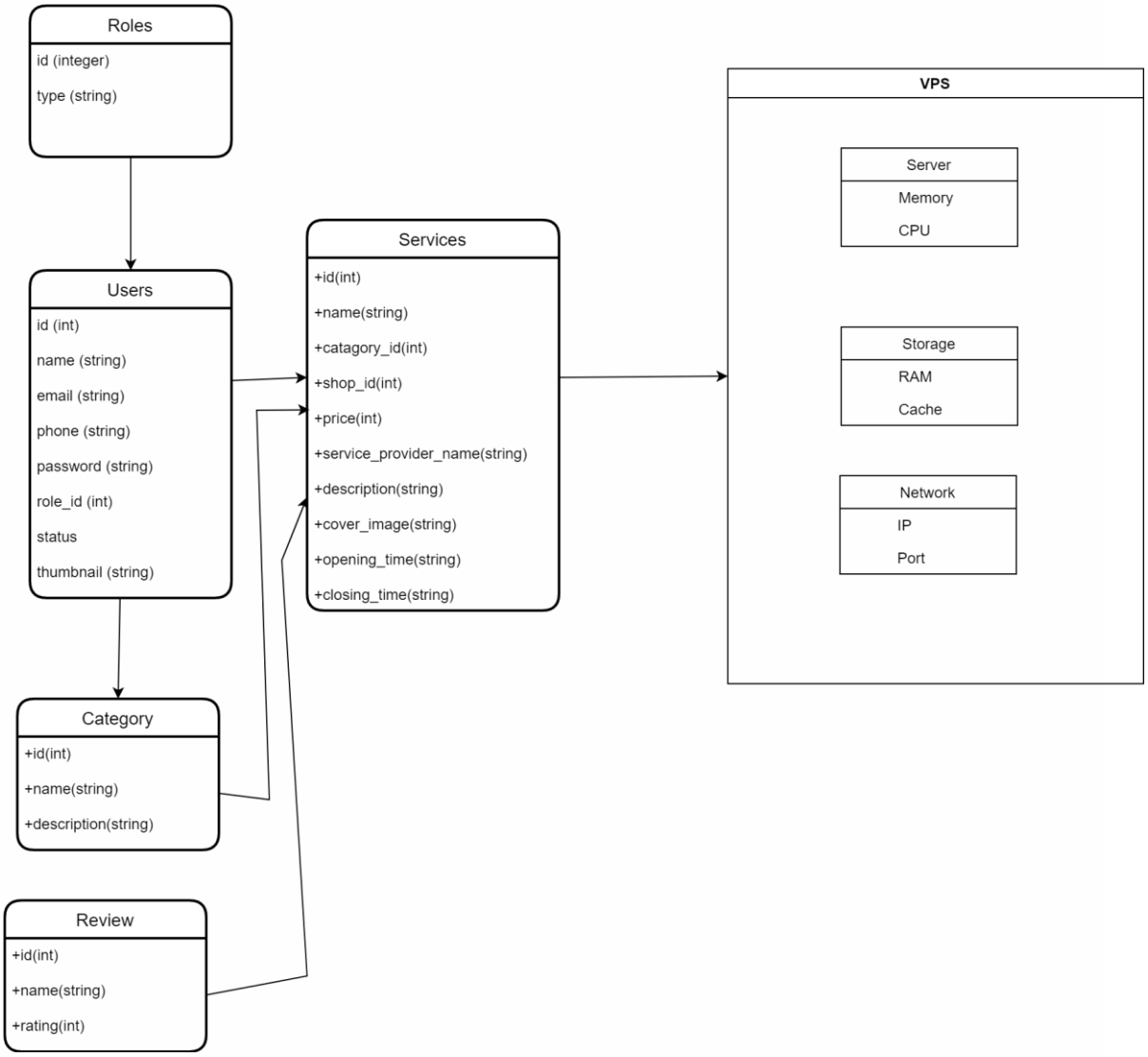


Figure 6: Simplified Class Diagram

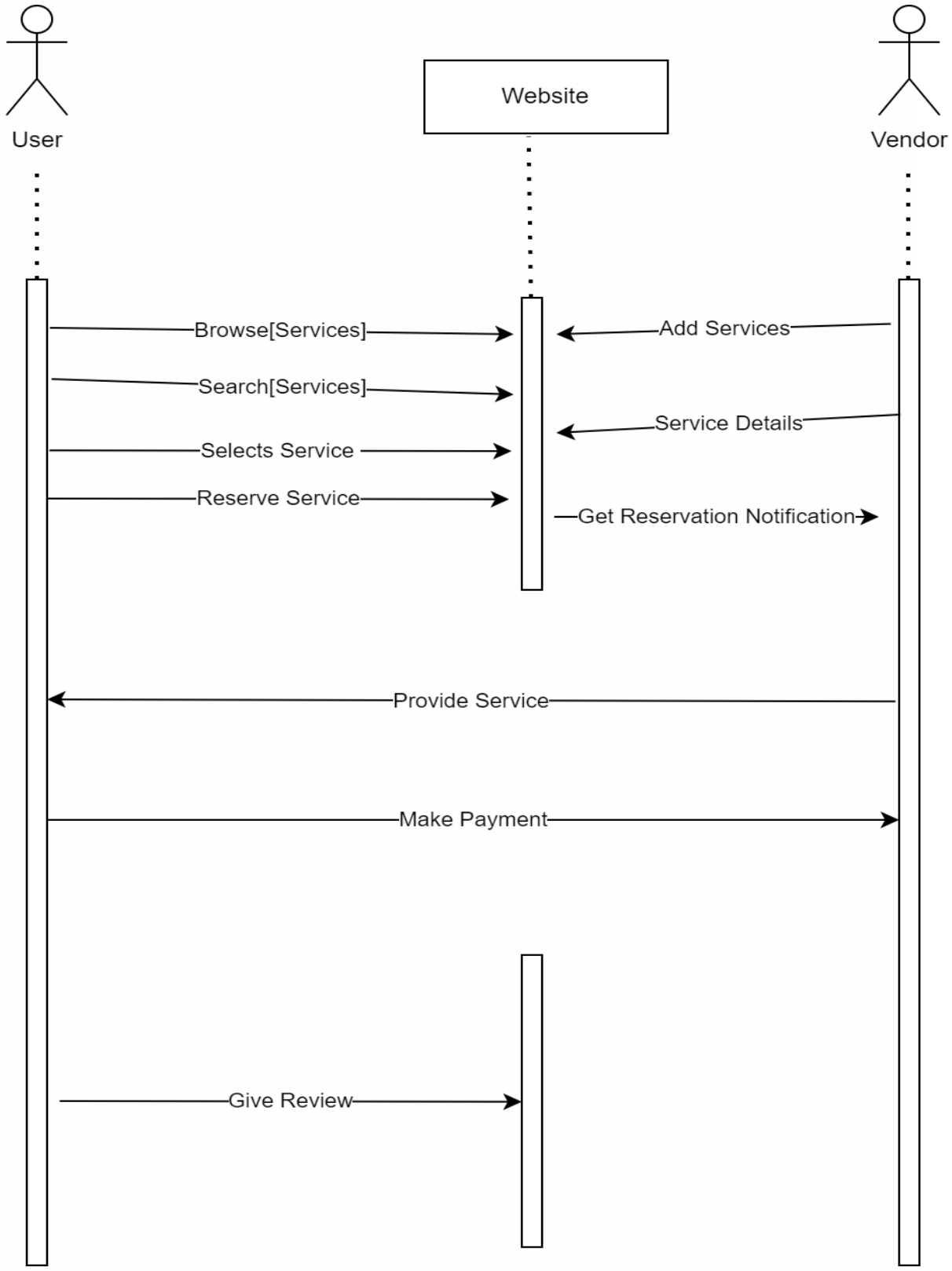


Figure 7: Simplified Sequence Diagram

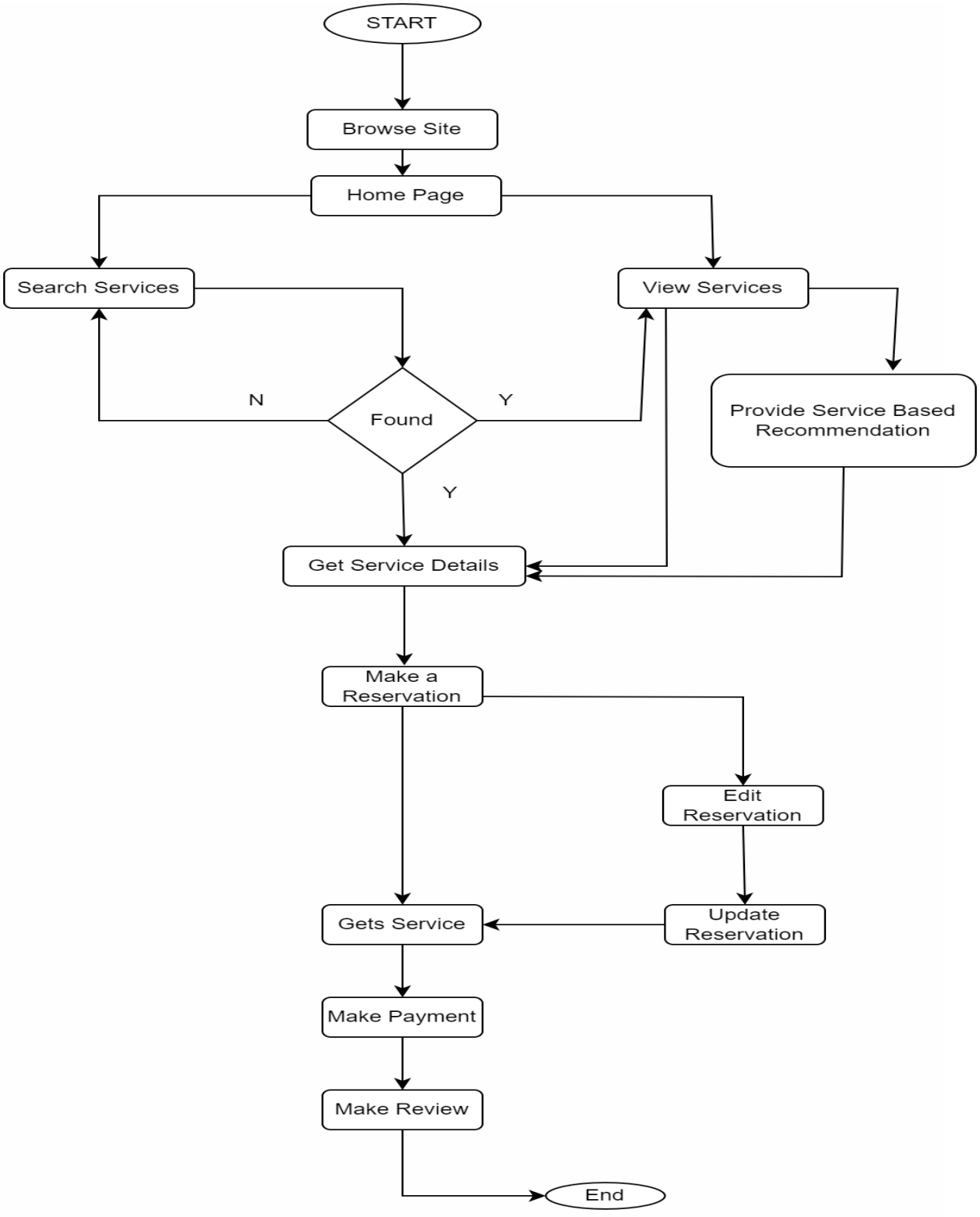


Figure 8: Simplified Activity Diagram

4.1.2. Component Diagram

The component diagram helps to visualize the physical components of the system and their dependency relationship. The **MNRSEWA** has the components such as a database, network, cloud server, services, and various parties involved to make a complete architecture.

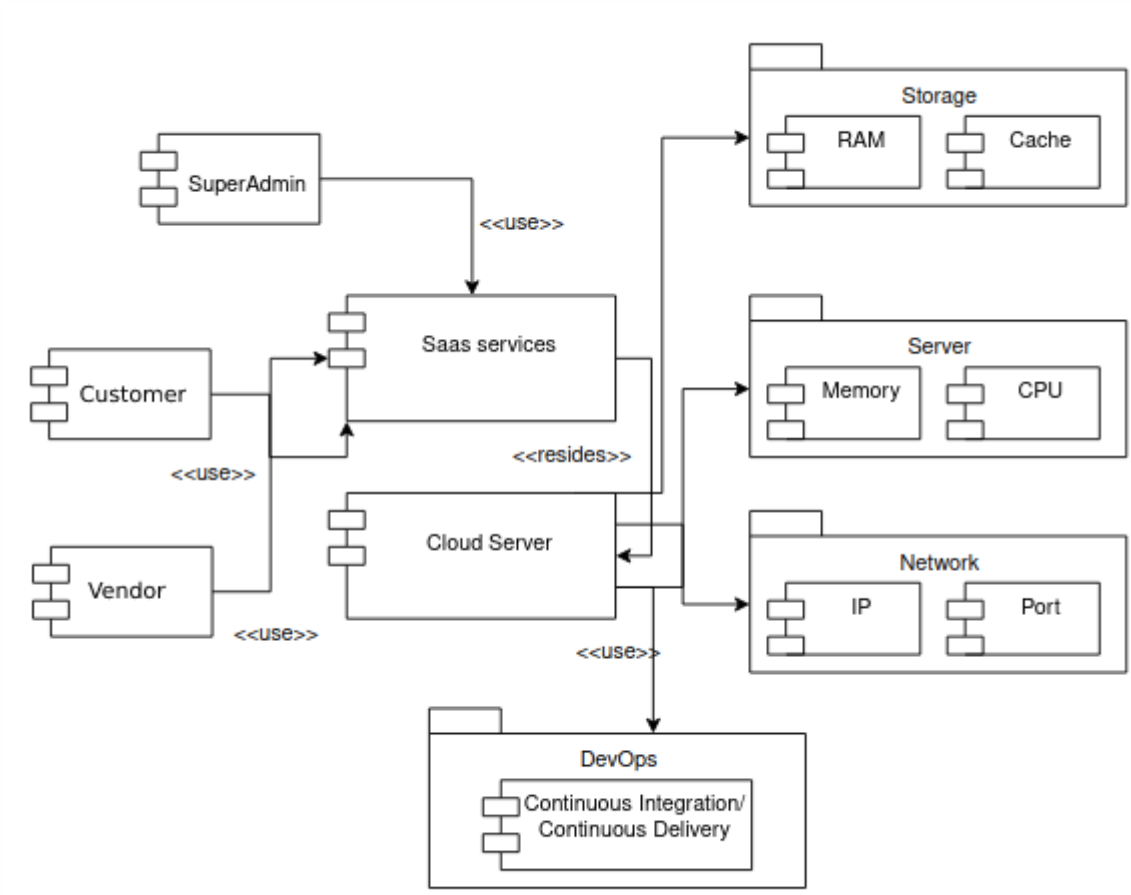


Figure 9: Component Diagram

4.1.3. Deployment Diagram

The deployment diagram shows the execution architecture of the system. It includes nodes such as hardware and software components and the middleware for the system execution. In the **MNRSEWA**, the cloud infrastructure excels to provide necessary hardware components whereas the software is developed on Laravel programming language which is then deployed in the SaaS domain.

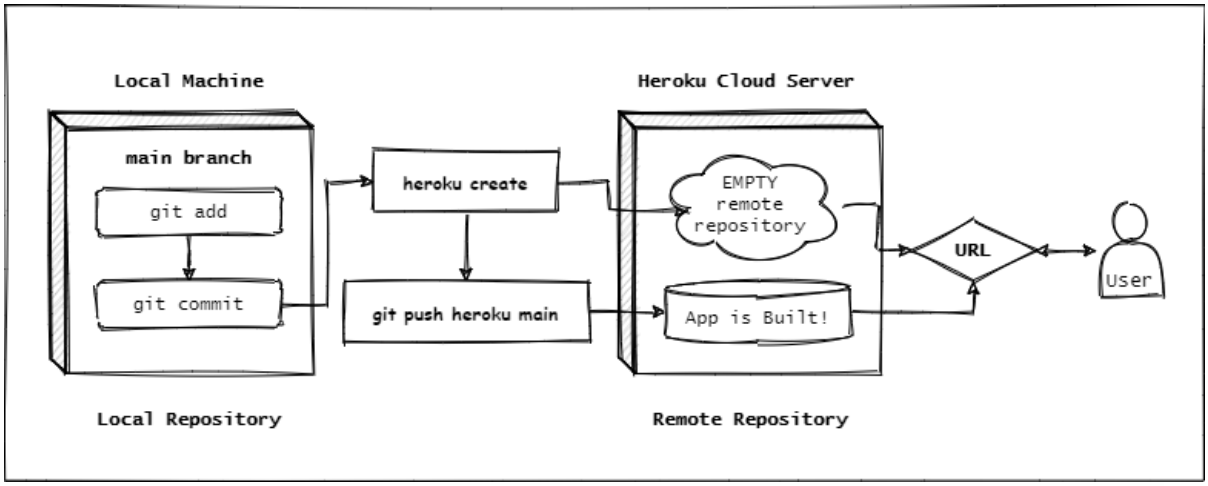


Figure 10: Deployment Diagram

4.2. Algorithm

4.2.1. Service Recommendation Algorithm

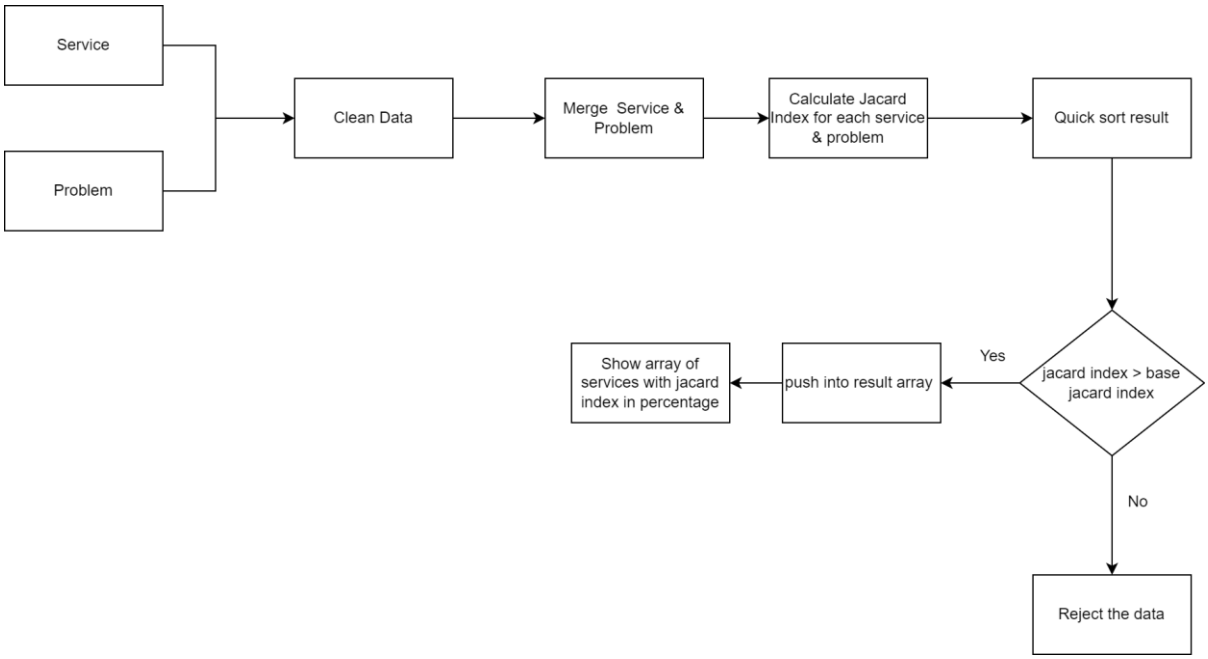


Figure 11: Service Recommendation Algorithm

The above diagram shows the core recommendation algorithm that we have used in this application. We have two data sets: service data set and problem data set. These data sets might contains extra information that we may not need like social media links, positions, and so on. So, we need to clean those data, in our case we have used the Danfo.js library. After that, we need to merge the properties like the service problem. Location and so on. Then after that, we need to calculate the Jaccard Index between the service and the problem.

We have discussed the algorithm in the latter part of the report. This will result in the Jaccard coefficient between 0 and 1. Since this unusual decimal value may not be good for the user experience we have to multiply it by 100 to get the value in percentage. Also, we have a certain base Jaccard index which is 10% in our case. If we have a Jaccard index calculated greater than the base Jaccard index we will recommend that vendor to the customer else we will ignore that vendor. In the end, we have to sort the array of vendors using Quicksort, in descending order of the Jaccard index calculated.

4.2.1.1. Collaborative Filtering Technique

We need to use collaborative filtering for our purpose. Collaborative filtering models compute their predictions using a dataset of feedback from users to items. We categorize these models further in how they process this data. Let's see the benefits and drawbacks of each approach.

a) User-User

The most commonly used recommendation algorithm follows the "people like you, like that" logic. We call it a "user-user" algorithm because it recommends an item to a user if similar users liked this item before. The similarity between two users is computed from the number of items they have in common in the dataset. This algorithm is very efficient when the number of users is way smaller than the number of items. You can think of a medium-sized online shop with millions of products. The major 16 drawback is that adding a new user is expensive since it requires updating all similarities between users.

b) Item-Item

The "item-item" algorithm uses the same approach but reverses the view between users and items. It follows the logic "if you like this, you might also like that". It recommends items that are similar to the ones you previously liked. As before the similarity between two items is computed using the number of users they have in common in the dataset. This algorithm is best when the number of items is way smaller than the number of users, such as in large-scale online shops. It is well suited if your items don't change too much, since you can pre-compute the full table of item-item similarities and then serve recommendations in real-time. Updating this table for adding a new item is unfortunately hard. [4]

c) User-Item

There are multiple forms of "user-item" recommendation algorithms that combine both approaches to generate recommendations. In our case, we can map its main principle as "If the vendor has similar skills or speaks similar languages or is from the same country then

you may also prefer that vendor". Once the users and their preferences are precomputed recommendations can be served in real-time. There are many approaches to calculate this like the matrix factorization method, and Jaccard index calculation. Matrix factorization methods are too verbose and don't serve well to our purpose. Since we have to calculate the similarity between the two user preferences but recommend the whole user at the end Jaccard Similarity algorithm serves our purpose well. [5]

4.2.2. Quick Sort

We can use Quicksort instead of JavaScript default sort as it will be different on different browsers and Node.js environments. For practical use, ease of implementation might be sacrificed for the sake of efficiency. On a theoretical basis, we can determine the number of element comparisons and swaps to compare performance. Additionally, actual running time will be influenced by other factors, such as caching performance and branch mispredictions.

We would be using Hoare's Partition scheme to sort the list of service providers (vendor) based on their Jaccard index. As the Lomuto partition scheme is more compact and easier to understand, it is frequently used in the partition process of Quicksort. But this scheme degrades to O(n2)

when the array is already sorted or when the array has all equal elements. In this post, a much more efficient Hoare partition scheme is discussed. Hoare Partition Scheme Hoare uses two indices that start at the ends of the array being partitioned, then move toward each other until they detect an inversion: a pair of elements, one greater than the pivot, one smaller, in the wrong order relative to each other. The inverted elements are then swapped. When the indices meet, the algorithm stops and returns the final index.

Hoare's scheme is more efficient than Lomuto's partition scheme because it does three times fewer swaps on average, and it creates efficient partitions even when all values are equal. But like Lomuto's partition scheme, Hoare partitioning also causes Quicksort to degrade to O(n2) when the input array is already sorted; it also doesn't produce a stable sort. Note that in this scheme, the pivot's final location is not necessarily at the index that was returned, and the next two segments that the main algorithm recurs on are [low…pivot] and [pivot+1…high] as opposed to [low…pivot-1] and [pivot+1…high] as in Lomuto's Scheme.

Pseudocode:

```
partition(arr[], lo, hi)
pivot = arr[lo]
i = lo - 1 // Initialize left index
j = hi + 1 // Initialize right index
// Find a value in left side greater
// than pivot
 Do
 i = i + 1
while arr[i] pivot
if i >= j then
return j
swap arr[i] with arr[j]
```

4.2.3. Quicksort algorithm using Hoare's partitioning scheme

Hoare partition is an algorithm that is used to partition an array about a pivot. All elements smaller than the pivot are on its left (in any order) and all elements greater than the pivot are on its right (in any order). Hoare uses two indices that start at the ends of the array being partitioned, then move toward each other, until they detect an inversion: a pair of elements, one greater than or equal to the pivot, one lesser or equal, that are in the wrong order relative to each other. The inverted elements are then swapped. When the indices meet, the algorithm stops and returns the final index.

The indices i and j run towards each other until they cross, which always happens at the pivot. This effectively divides the array into two parts: A left part which is scanned by i and aright part scanned by j. Now, a swap is done exactly for every pair of "misplaced" elements, i.e., a large element (larger than the pivot, thus belonging in the right partition) which is currently located in the left part and a small element located in the right part.

4.2.2.1. Algorithm of Hoare Partition

While partitioning arrays using the Hoare Partition scheme, we make an assumption:" The pivot element is always the first element of the array."

Pseudocode:

```
fun quicksort (input: T [], low : int, high : int)
 if (low < high)
 p: = partition (input, low, high)
 quicksort (input, low, p) // Note that this is different than when using Lomuto
 quicksort (input, p + 1, high)
19
fun partition (input: T [], low: int, high: int): int
 pivot Point: = floor ((high + low) / 2)
 pivot: = input [pivot Point]
 high++
 low--
 loop while True
 low++
 loop while (input[low] < pivot)
 high--
 loop while (input[high] > pivot)
 if (low >= high)
 return high
 swap(input[low], input[high])
```

4.2.4. Jaccard Similarity Algorithm:

Jaccard Similarity is a common proximity measurement used to compute the similarity between two objects, such as two text documents. Jaccard similarity can be used to find the similarity between two asymmetric binary vectors or to find the similarity between two sets. In literature, Jaccard similarity, symbolized by, can also be referred to as Jaccard Index, Jaccard Coefficient, Jaccard Dissimilarity, and Jaccard Distance.

Jaccard Similarity is frequently used in data science applications. Example use cases for Jaccard Similarity:- Text mining: find the similarity between two text documents using the number of terms used in both documents.

- E-Commerce: from a market database of thousands of customers and millions of items, find similar customers via their purchase history.
- Recommendation System: Movie recommendation algorithms employ the Jaccard coefficient to find similar customers if they rented or rated highly many of the same movies. The Jaccard similarity measures the similarity between two sets of data to see which members are shared and distinct. The Jaccard similarity is calculated by dividing the number of observations in both sets by the number of observations in either set. In other words, the Jaccard similarity can be computed as the size of the intersection divided by the size of the union of two sets. This can be written in set notation using intersection (A∩B) and unions (A∪B) of two sets:

J (A, B) =|A∩B||A∪B|

Where |A∩B| gives the number of members shared between both sets and |A∪B| gives the total number of members in both sets (shared and unshared). The Jaccard Similarity will be 0 if the two sets don't share any values and 1 if the two sets are identical. The set may contain either numerical values or strings.[6]

Pseudo Code:

```
Jaccard Similarity(doc1, doc2):
 # List the unique words in a document
 words_doc1 = set(doc1.lower().split())
 words_doc2 = set(doc2.lower().split())
 # Find the intersection of words list of doc1 & doc2
 intersection = words_doc1.intersection(words_doc2)
 # Find the union of words list of doc1 & doc2
 union = words_doc1.union(words_doc2)
 # Calculate Jaccard similarity score
 # Using length of intersection set divided by length of union set
 return float (Len(intersection)) / Len(union)
```

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1. Implementation

5.1.1. Tools Used

CASE as in Computer-Aided Software Engineering is a tool that helps with the analysis and design of software systems. Is a piece of software that helps in the development and maintenance of other software products? Code generators, Unified Modeling Language (UML) tools, refactoring tools, and model transformation tools are examples of such tools [4].

5.1.1.1. Programing Language Tools

The **MNRSEWA** uses Laravel, a PHP framework as the primary language to build the backend of the application. For the front-end task, mnrSewa uses HTML, Bootstrap 4 CSS for the structure and styling of the web application, and JS for the front-end for the web interface. Laravel is an open-source framework from Php. The Laravel application architecture follows MVC (Model, View, and Controller) to serve the application on the website.

5.1.1.2. Database Tools

Initially we were using MySQL database at the development phase of our application, later on, at the production level, we are using PostgreSQL to server the application data. It is an open-source object-relational database and is flexible with our Laravel integration.

5.1.1.3. Drawing Tools

We are using Figma for the UI/UX design of our web application. For the drawing of our diagrams we are using Flowchart Maker & Online Diagram Software, an online development tool for making flowcharts, Class Diagram, DFD, and UML.

5.1.1.4. Deployment Tools

For the deployment we are using CI/CD approach which uses tools such as Postgres database server, Heroku, GitHub, and Git.

5.1.2. Implementation Details of Modules

MNRSEWA uses various models to design the complete system, these include:

FOR USER:-

- **Module 1: Register New User (Email, Password)**

  Upon registration, the user has to enter the valid email, password, and repeat the password. When a user has entered the data a verification mail is sent to the user's email by which the user can verify their identity.

- **Module 2: Log in User (Email, Password)**

  Upon login, the Login module needs an email and password. If the credentials match our database which is encrypted then the login is successful.

- **Module 3: Add Interested Fields of Services**

  In this module, a user who registered to the application can select particular fields of interest which are pre-set by the vendor or a service provider so that later they should be recommended by those fields of services.

- **Module 4: Visit Home Page**

  In this module, users can visit a home (main) page of our application where they can find different services, and details, get recommendations, get best-reviewed services providers, etc. moreover they can operate the entire application through this page.

- **Module 5: Service Reservation**

  In this module, users can find their required service and make a reservation of service and make contact with the service provider throughout the application.

- **Module 6: Gets Service**

  In this module, a user gets the problem solved by the service provider and pays their charges.

- **Module 7: Review Service Provider or a Freelancer**

  In this module, users can give reviews based on the facility or satisfaction level they get from services providers and help them to build a greater profile.

FOR Vendor / Freelancer:-

- **Module 1: Register New User (Email, Password)**

  Upon registration, the vendor has to enter the valid email, and password and repeat the password. When a user has entered the data a verification mail is sent to the user's email by which the user can verify their identity and a notification is sent to

the admin of the application to verify identity, certificates, experiences, etc and the admin can accept or reject the vendor's request.

- **Module 2: Login Vendor(Email, Password)**

  Upon login, the Login module needs an email and password. If the credentials match our database which is encrypted then the login is successful.

- **Module 3: Dashboard**

  In the Dashboard module, the daily reservation logs, reservation logs based on month, and team performance can be seen. Also, it tracks the team members of the workshop.

- **Module 4: Add Service Details**

  In this module, the vendor can add information about work on the relevant service or a field. Under the single vendor, one can add multiple fields of services as a workshop can perform multiple services and respectively multiple team members in the field or services.

5.2. Testing

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that the software product is defect-free. It involves the execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps, or missing requirements in contrast to actual requirements.

5.2.1 Test Cases for Unit Testing

Unit test basic approach is followed by the programmer to test the unit of the program. It helps developers to know whether the individual unit of the code is working properly or not. Various unit test cases are shown below:

5.2.1.1. User & Vendor Login and Registration Testing

Table 4: User & Vendor Login and Registration Testing

SN.	Test Cases	Feature	Description	Steps To Execute	Expected Results	Remarks
1	TC-01	User Interface	Check all the text boxes and buttons	Check Page	• UI should be perfect • Text boxes and buttons should be aligned	Pass
2	TC-02	Required Fields	Check the required fields by not filling any data.	1. Enter an invalid username 2. Enter the correct password 3. Click on the Login Button	User should not log in and should show a proper error message	Pass
3	TC-03	User Login	Check When passing a correct email and invalid password	1. Enter valid email 2. Enter incorrect password 3. Click on the Login Button	User should not log in and should show a proper error message	Pass
4	TC-04	User Interface	Check Keeping Password	1. Enter a valid username 2. Do not enter password 3. Click on the Login Button	User should not log in and should show a proper error message	Pass
5	TC-05	User Login	Check when passing the correct email and password	1. Enter a valid username 2. Enter a valid password	User should log in	Pass

				3. Click on the Login Button		
6	TC-06	User Login	Check if the password is entered in encrypted	1. Enter a valid username 2. Enter password 3. Click on Login Button	Password is entered in encrypted form	Pass
7	TC-07	Signup Option for new users	Check whether the signup link for the new user is working	Click Signup link	Clicking the signup link takes the user to signup page successfully	Pass
8	TC-08	Vendor Registration	Check whether the request for approval goes to admin or not	Accept or Reject by admin	Notification of approval is sent to the admin	Pass

5.2.1.2. Add services testing

Table 5: Add Services Testing

SN.	Test Cases	Feature	Description	Steps To Execute	Expected Results	Remarks
1	TC–01	Add service	Click on add service to create a new service	Create service and provide service name	Service should be successfully added	Pass
2	TC-02	Service Details	Click on team details to see service and plans	Enter service details	Service details should successfully be added	Pass

3	TC-03	Edit Service	Click on Edit button	Click on an editing service and edit required data	The services should be updated	Pass
4	TC-04	Delete Service	Click on Delete Button	Click on the delete button to delete service	The service will be deleted	Pass

5.2.1.3. Select Interested Category by User testing

Table 6: Select Interested Category by User Testing

SN.	Test Cases	Feature	Description	Steps To Execute	Expected Results	Remarks
1	TC-01	My Profile	Go to My Profile or Dashboard	Click on my profile to add a category that they needed	The project should be successfully added.	Pass
2	TC-02	Choose Categories	Click on Categories	Click on only those categories that are needed	Selected categories services are recommende d	Pass

5.2.1.4. Language Change testing

Table 7: Language Change Testing

SN.	Test Cases	Feature	Description	Steps To Execute	Expected Results	Remarks
1	TC-01	Select English	English Language is selected	Click on en on menu bar to select English language	Language is changed to English	Pass

2	TC-02	Select Nepali	Nepali Language is selected	Click on np on the menu bar to select Nepali language	Language is changed to Nepali	Pass

5.2.1.5. Dashboard testing

Table 8: Dashboard Testing

SN.	Test Cases	Feature	Description	Steps To Execute	Expected Results	Remarks
1.	TC-01	Dashboard	View details log	Click on the dashboard to view details of services, reservations	View services and reservations notification	Pass

5.2.1.6. Service Reserving Testing

Table 9: Service Reserving Testing

SN.	Test Cases	Feature	Description	Steps To Execute	Expected Results	Remarks
1.	TC-01	Book a Service	Find a service and make a reservation of service	- Search for the service they want to get - Make a reservation	The service is reserved and notification is sent to the vendor	Pass

5.2.1.7. Review Service Testing

Table 10: Service Reserving Testing

SN.	Test Cases	Feature	Description	Steps To Execute	Expected Results	Remarks
1.	TC-01	Give Review	Give a review after getting served	Review based on the satisfaction level of service provider	Review is updated	Pass

5.2.2. Test Cases for System and Acceptance Testing

Acceptance testing is a testing technique performed to determine whether or not the software system has met the required specifications. The main purpose of this test is to evaluate the system's compliance with the business requirements and verify if it has met the required criteria for delivery to end-users. The test case is given below:

Table 11: System and Acceptance Testing

SN.	Test Cases	Feature	Description	Steps To Execute	Expected Results	Remarks
1	TC-01	User login and registration	User signup with username and password, it gets saved to the database, the user now login with the same credential	Click on signup and fill the form, fill login form and login to the dashboard.	User has successfully authenticated	Pass

2	TC-02	User searches a service	User search for service and view details and find out best service	Read each detail and experience of the service provider to analyze the service provider among various	Check which service provider is best to take	Pass
3.	TC-03	User book a service	Customer book a service and expects to get service soon	Notification is sent to the vendor to give service to the user	Service reservation is made successfully	Pass
4	TC-04	Notification to Vendor	Vendor get notification from user of booking	User gets the address of user and go to serve to	Vendor to give service and expect for payment	Pass

5.3. Result Analysis

We have done a number of manual testing of our application and web server. The application performs efficiently in any given adverse condition. The failure rate is relatively zero to none.

After the project team tests, the product and the product passes each testing phase, the product is ready to go live. The methods used by developers to build, test and deploy new code will impact how fast a product can respond to changes in customer preferences or requirements and the quality of each change.

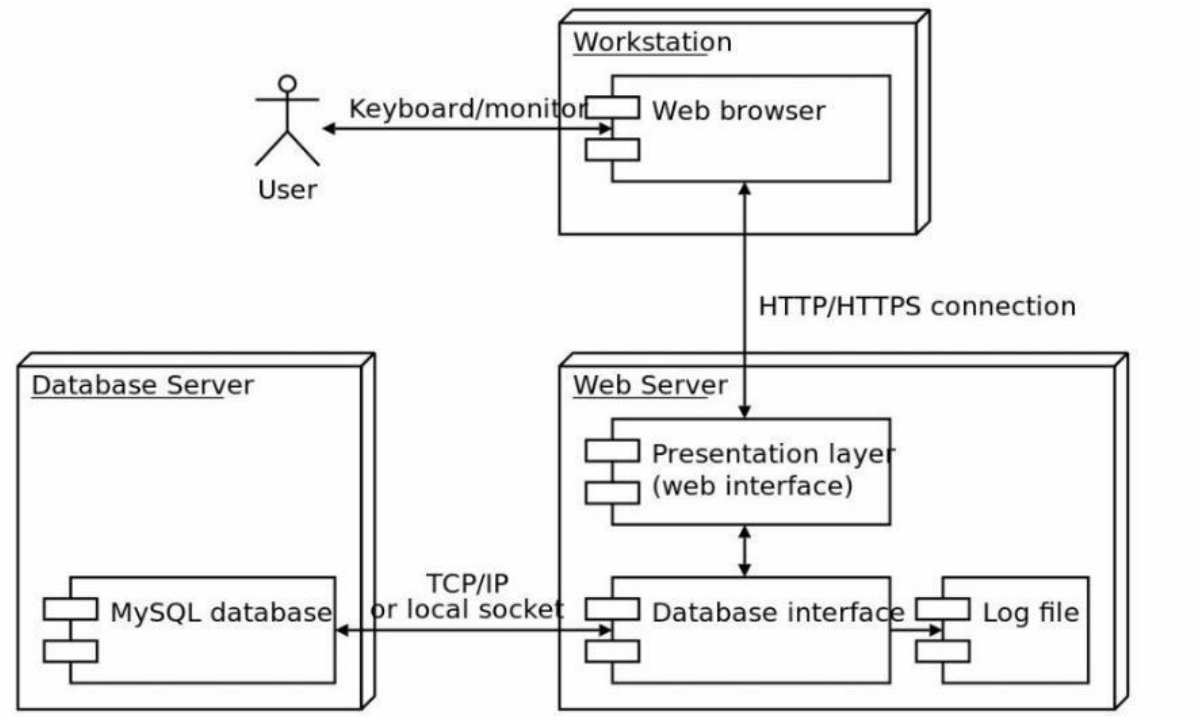


Figure 12: Deployment workflow of MNRSEWA

For the version control we have used Git and GitHub platforms. The GitHub runs all the actions for CI/CD and if the tests are passed then they will be pushed to the main branch. Then the GitHub bot runs [6] on the main branch and the application gets deployed to the Heroku server.

As shown in the above figure we have deployed the backend and frontend of our application on the Heroku server. Similarly, we have hosted our PostgreSQL database in Heroku. The deployed version of the site can be viewed at: https://mnrsewa.herokuapp.com/

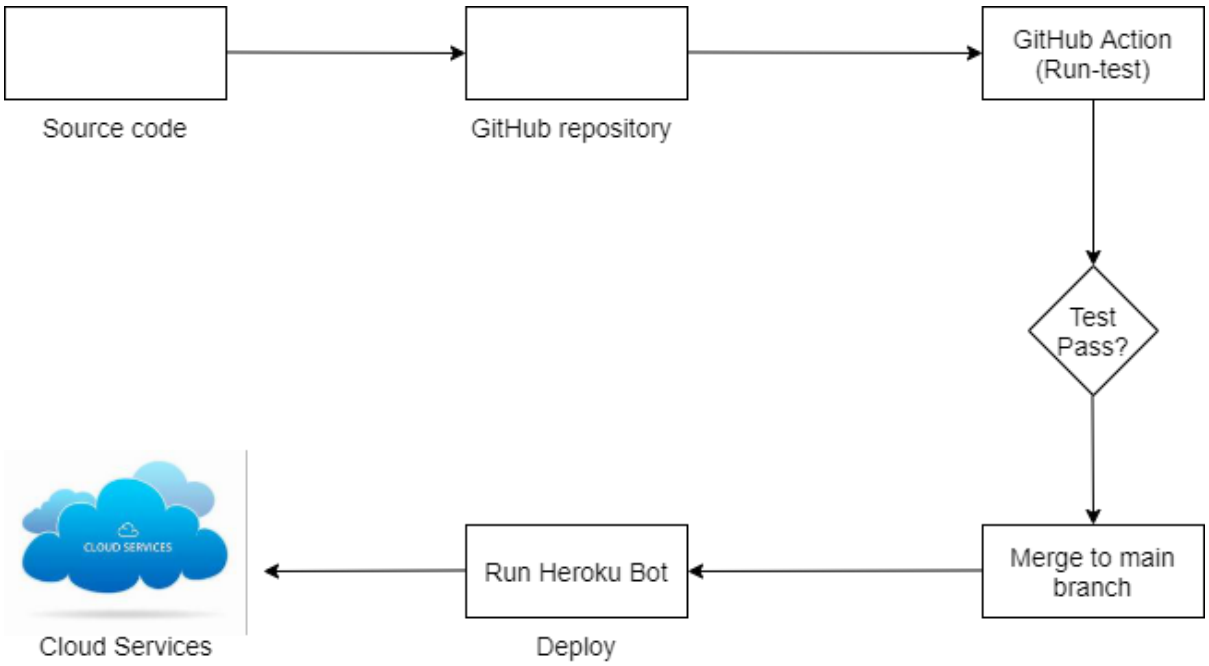


Figure 13: Cloud Architecture of MNRSEWA

CHAPTER 6
CONCLUSION AND RECOMMENDATION

6.1. Conclusion

Now you can find all the applications of any everyday life. But people still have to manually find and call local electricians, plumbers, carpenters, painters, go to hospitals, manually visit IT service software companies, etc... They may need to spend the whole day finding someone who can solve a small household problem, such as an electrical problem, a door problem, or a water problem. Not every time people can find someone to solve the problem.

This project allows customers to register with the application and use the service based on their personal or business needs. Freemium is included in the subscription plan. Once registered, users can search for specific issues according to their needs and only pay the service provider's fee. As a service providers, providers can create and add specifications on how to address issues. It provides the flexibility to find an effective solution to your problem. B. Evaluation-based, close to service or cheap in terms of cost. Some of the application's features are authentication, lead registration, real-time service tracking, and billing and billing. Users can set up preliminary information about their area of interest to receive these field services as recommendations. For example, users can select areas of interest such as IT services, electricians, and plumbers to get the most specific recommendations for IT services, electricians, and plumbers.
"Mnr Sewa?" Is the name of an application being developed and proposed to solve the current problem of finding a man who can solve a home or professional office problem?

6.2. Future Recommendations

There is always room for improvement, and so do our applications. We can add a number of functions and features also, and improve the existing ones. The feature we can add to the existing project is the Ticket management system, advanced review system, Revenue system, online payment, and real-time Order tracking giving a complete solution to real problems faced while taking the mnrSewa. We are starting this idea in the real field which will be starting from our own location in Mahendranagar then we will expand all over the country.

REFERENCES

[1]"Agile Modeling (AM) Home Page: Effective Practices for Modeling and Documentation." Agile Modeling (AM) Home Page: Effective Practices for Modeling and Documentation. N.p., n.d. Web. 7 Nov. 2013. <http://www.agilemodeling.com>.

[2]Wikipedia, “Service-Based Recommendation System”, Google. https://en.wikipedia.org/wiki/Web_application (accessed June 4, 2021)

[3]G. X. Zhijun Zhang and P. Zhang, "Research on E-Commerce Platform-Based Personalization," Hindawi, p. 7, 2016.

[4] Overview on CASE Tools, IGI Global https://www.igi-global.com/dictionary/reverse-engineering-of-object-oriented-code/3411

[5] P. Harrington, Machine Learning in Action, Shelter Island, New York 11964:Manning Publications Co., 2012.

[6] Deploy project on Heroku as a CI/CD, https://devcenter.heroku.com/articles/getting-started-with-laravel

[7] IEEE Referencing: Reference formats & examples https://libraryguides.vu.edu.au/ieeereferencing/formats

APPENDIX I (User)

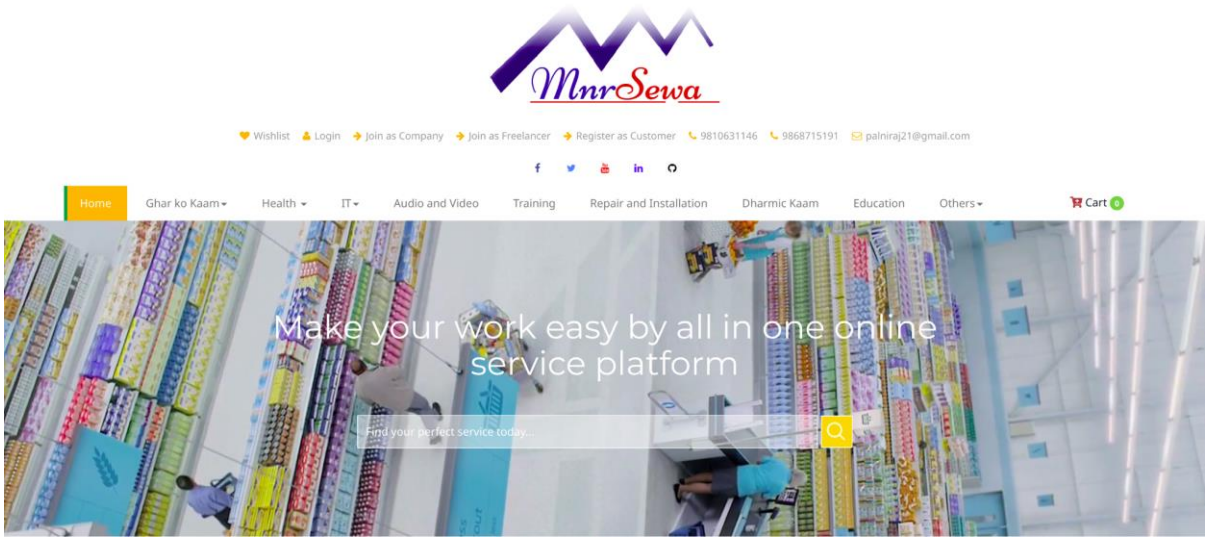


Figure: MNRSEWA -Homepage

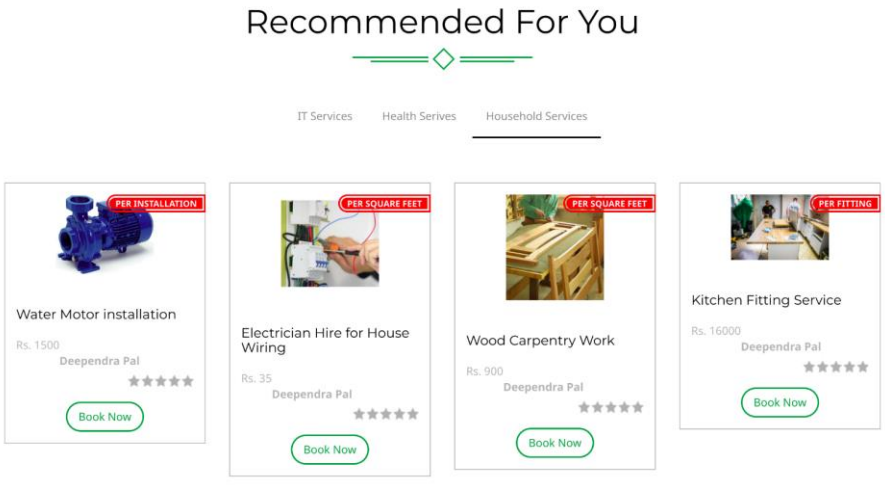


Figure: service-Recommended from Homepage

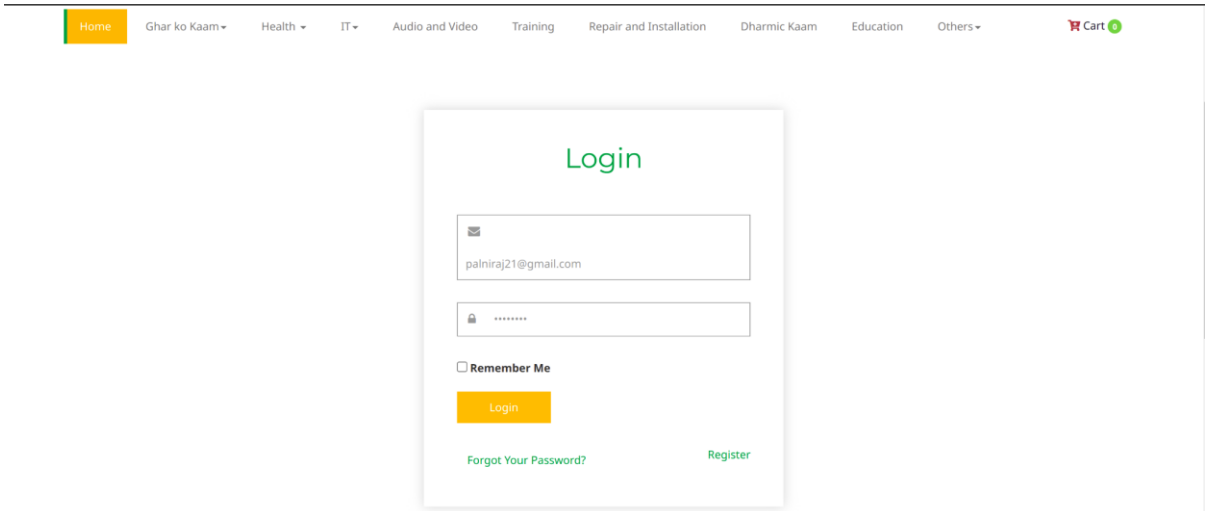


Figure: Login

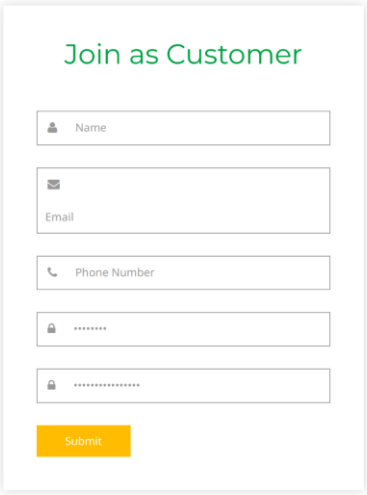
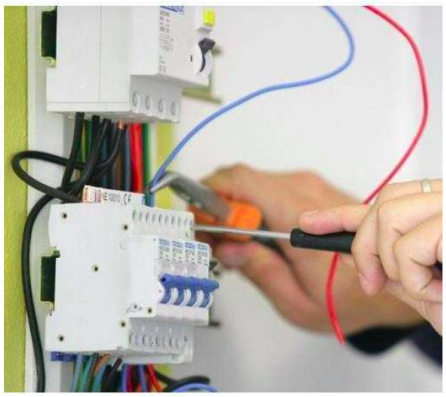


Figure: Signup



Electrician Hire for House Wiring

Deependra Pal

Rs.35

QUICK OVERVIEW:

Do you need to install new house wiring or repair old ones? This service is perfect for you if you want safe installation of house wiring for your house or office. The price is mentioned in per square foot basis which will help you make a total estimation. The price is only my service charge and not for the product needed for the job. I provide my service all over Mahendranagar so in case you need a professional electrician, please book my service from 10 am to 6 pm.

Book Now

Figure: Service Details

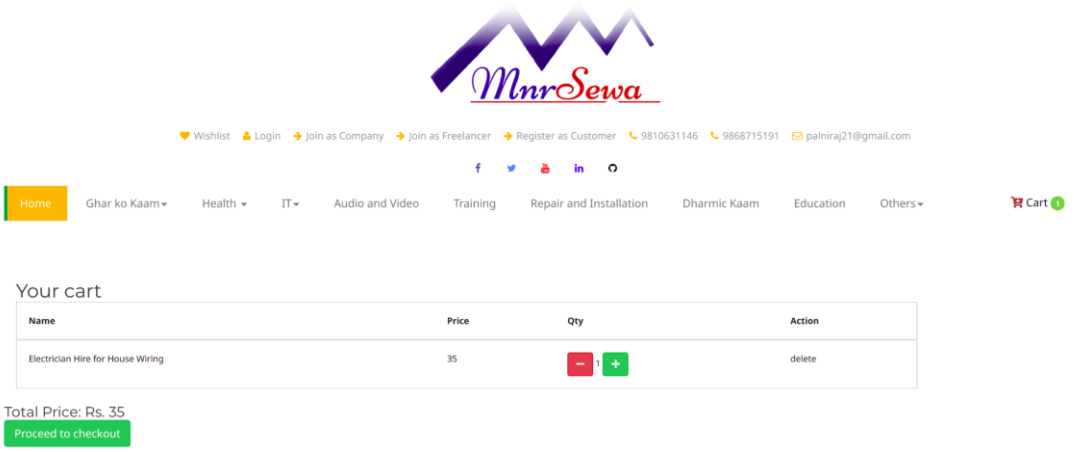


Figure: Service in Cart

Figure: Checkout Form

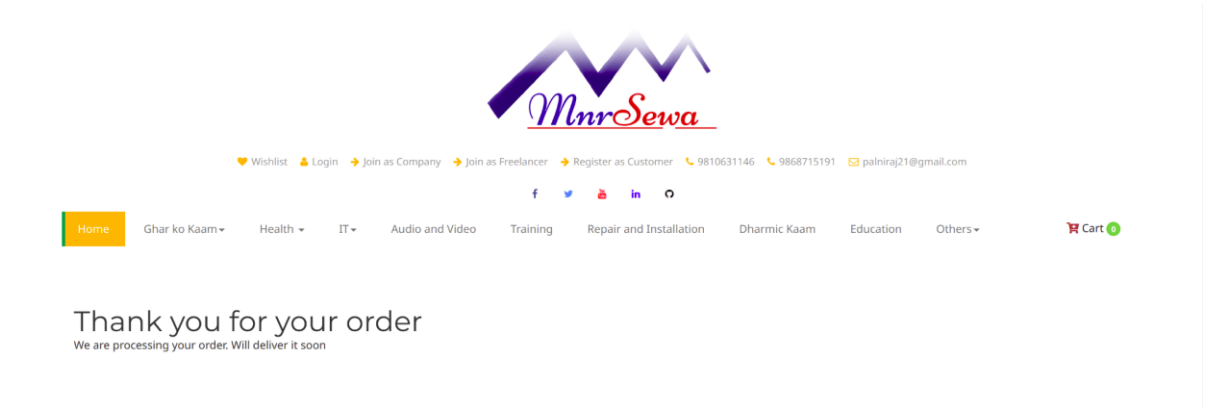


Figure: After Service Booked

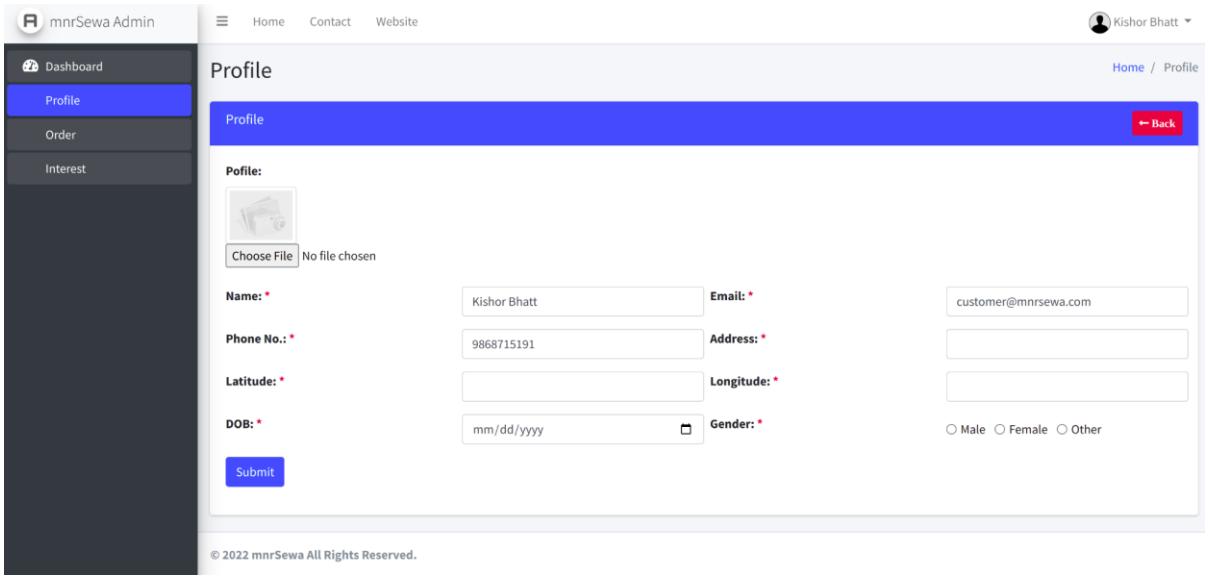


Figure: Profile Dashboard

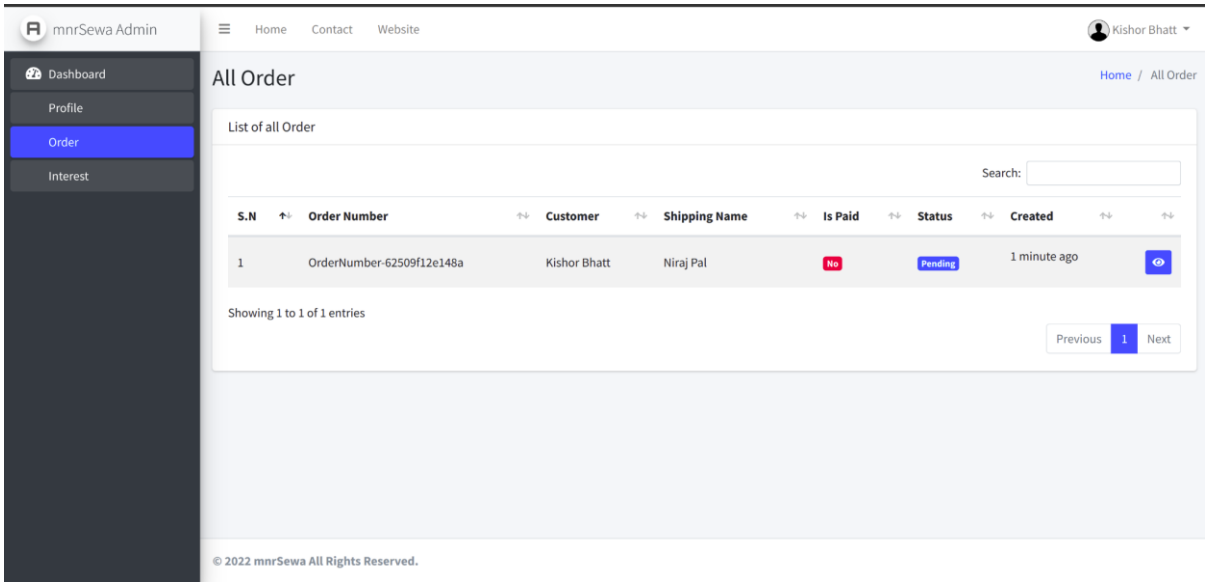


Figure: Reserved Services

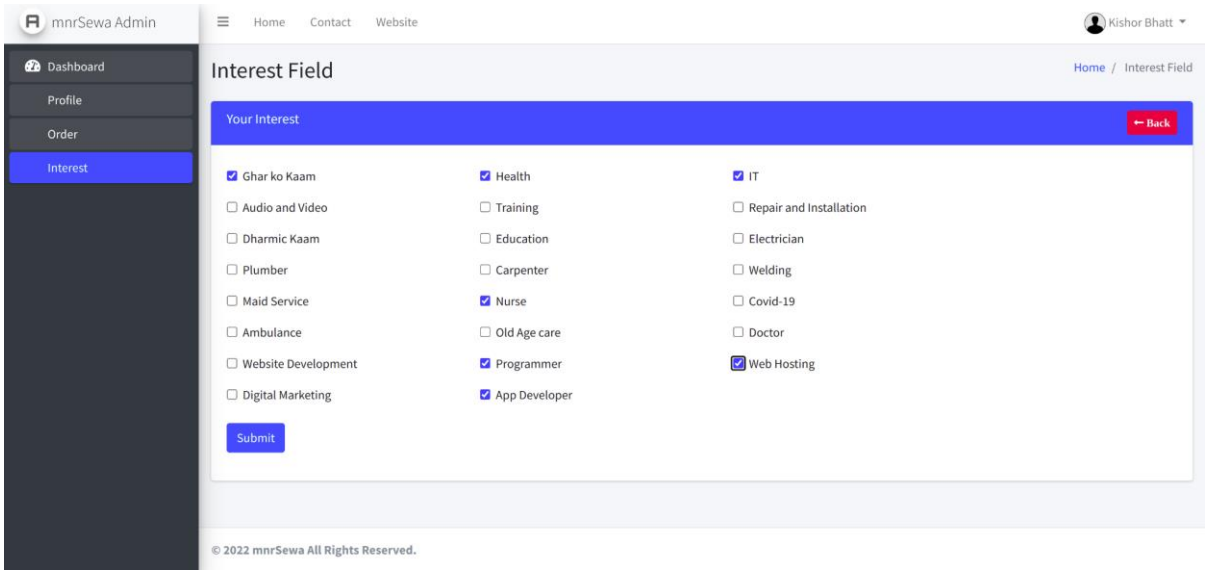


Figure: Choose Category

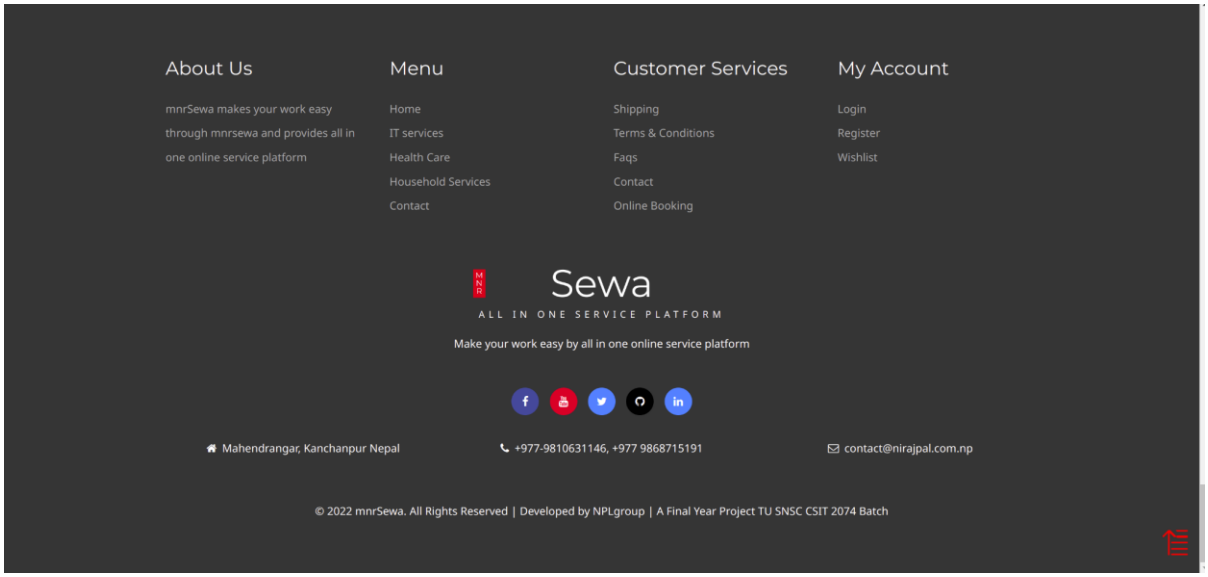


Figure: Footer

APPENDIX II (Admin)

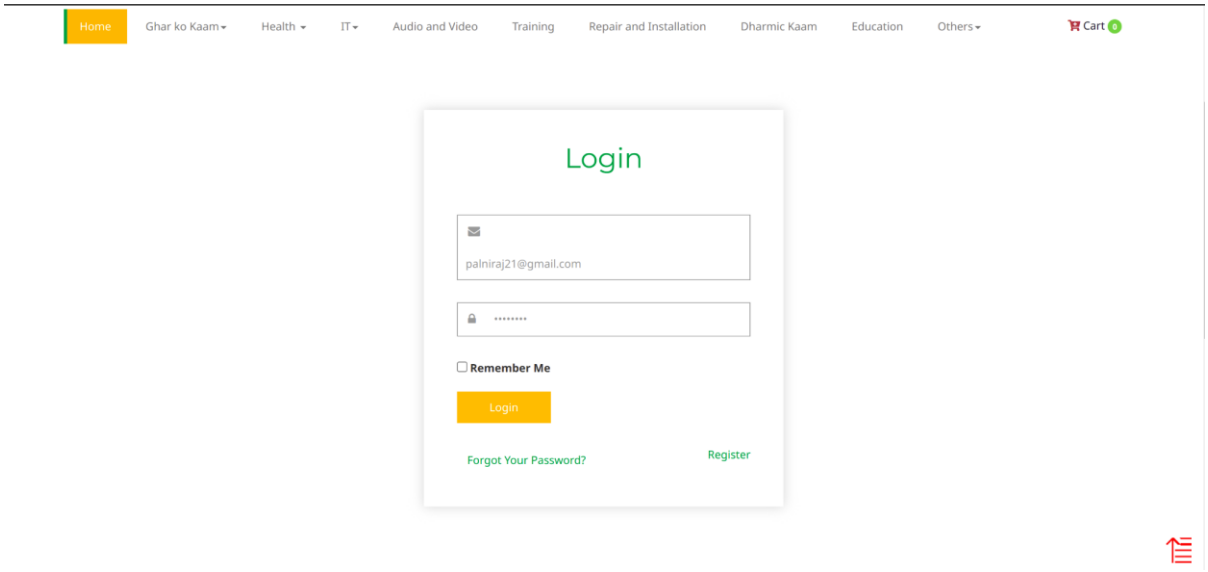


Figure: Signup/Login

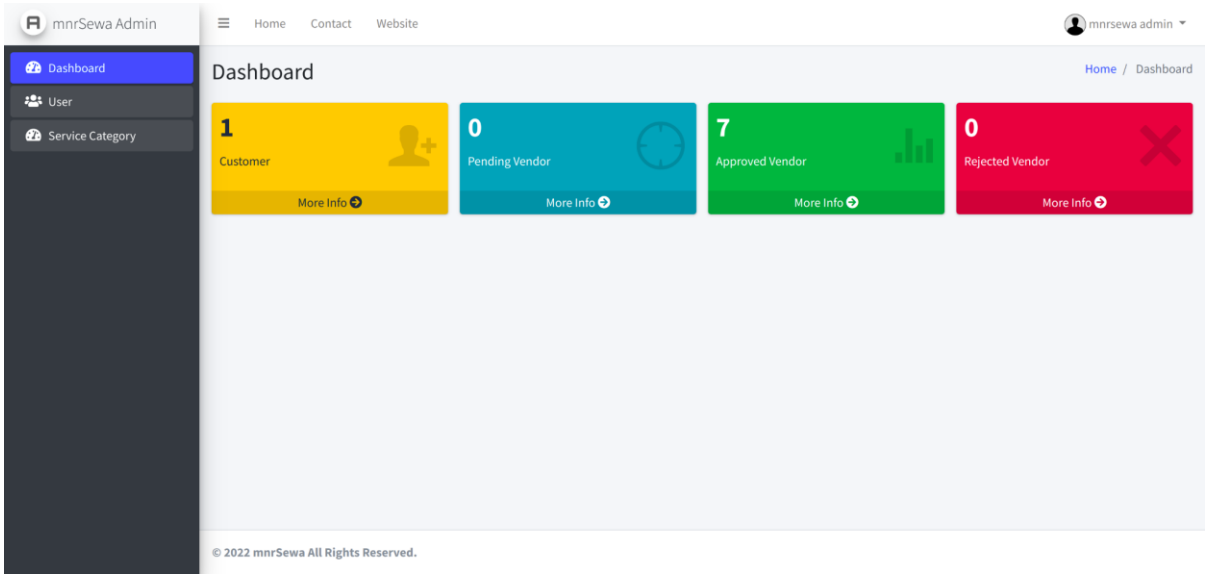


Figure: Dashboard

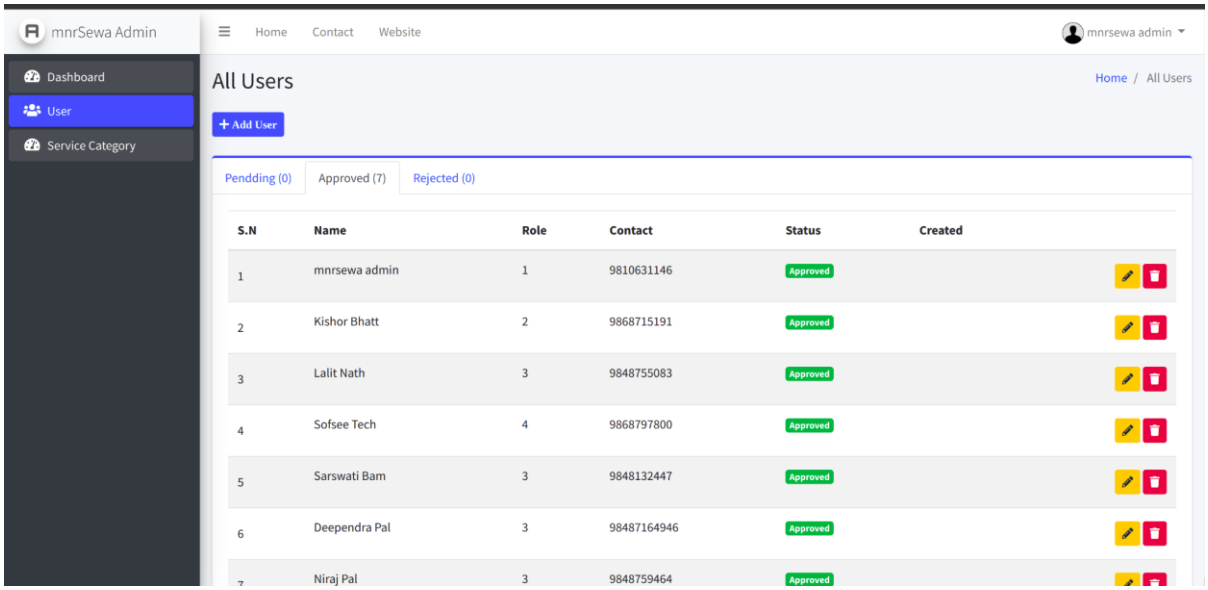


Figure: Notification of approval for Vendor List

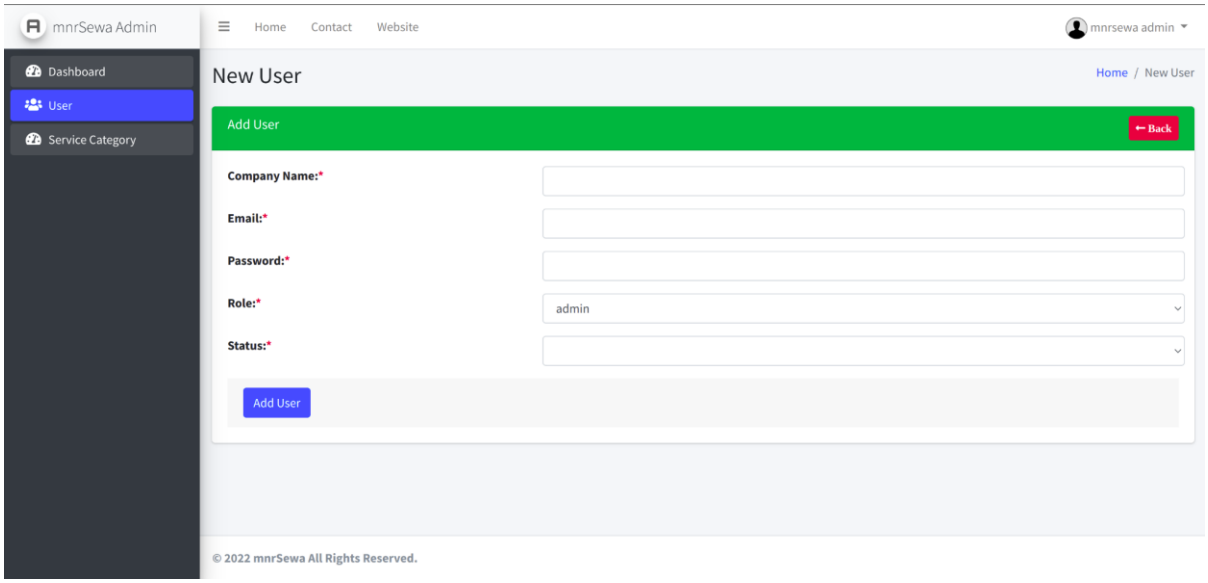


Figure: Add New Vendor

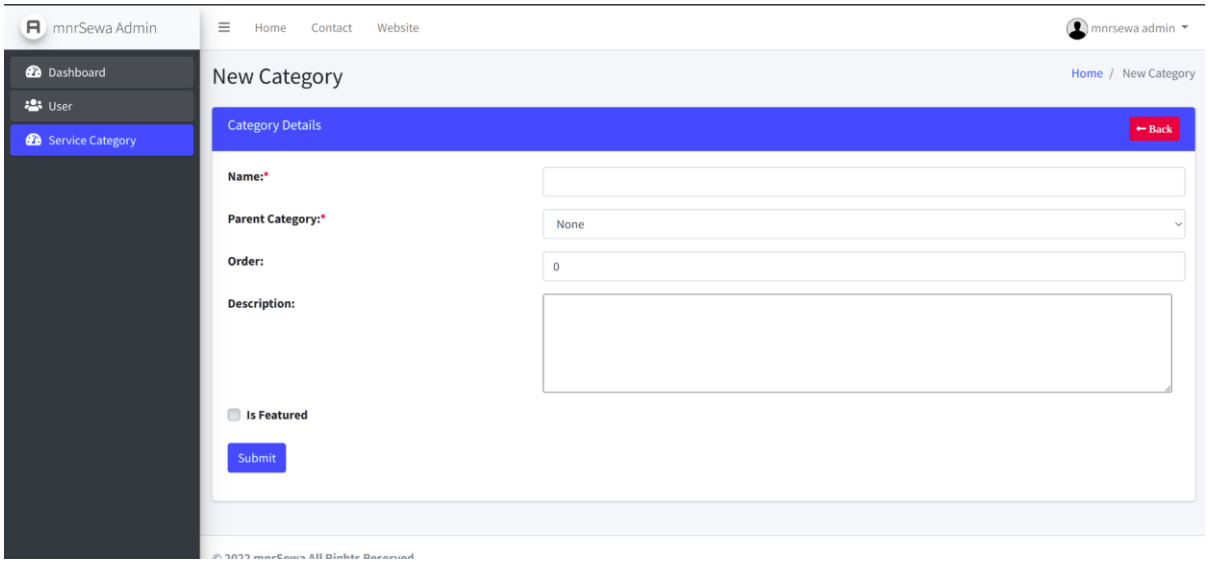


Figure: Add New Category

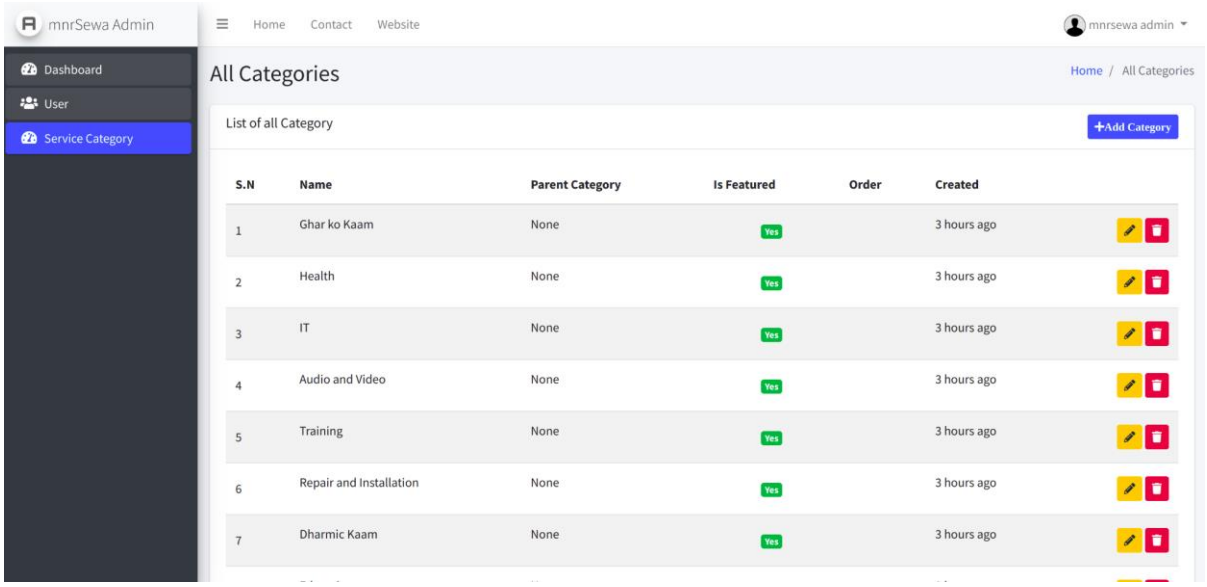


Figure: Category List

APPENDIX III (Vendor)

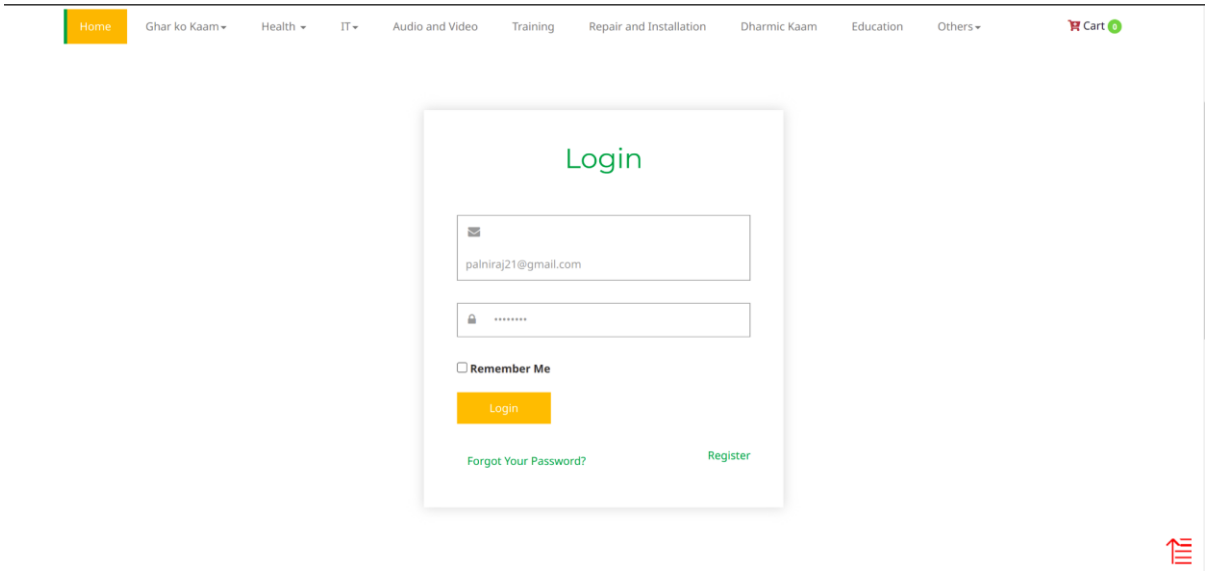


Figure: Login

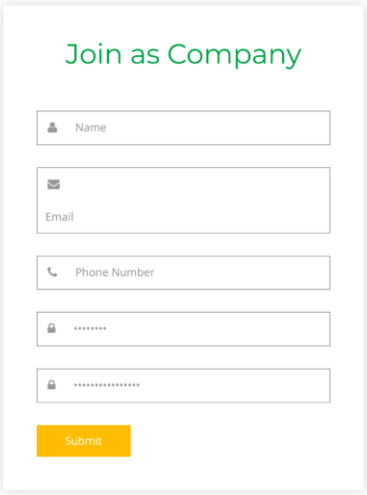


Figure: Register as Company

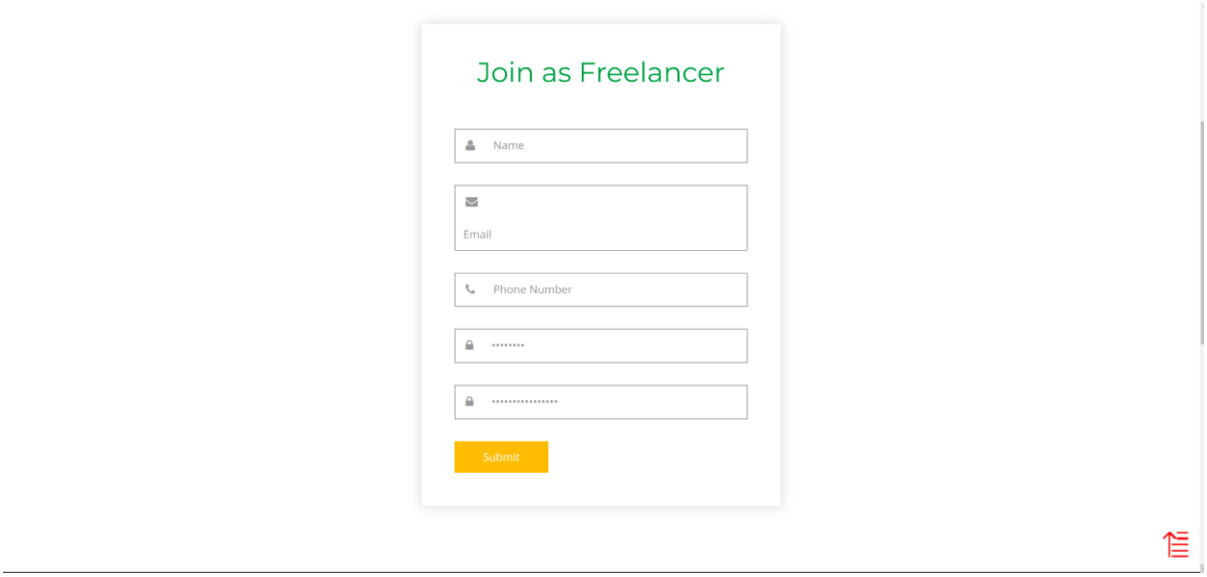


Figure: Register as Freelancer

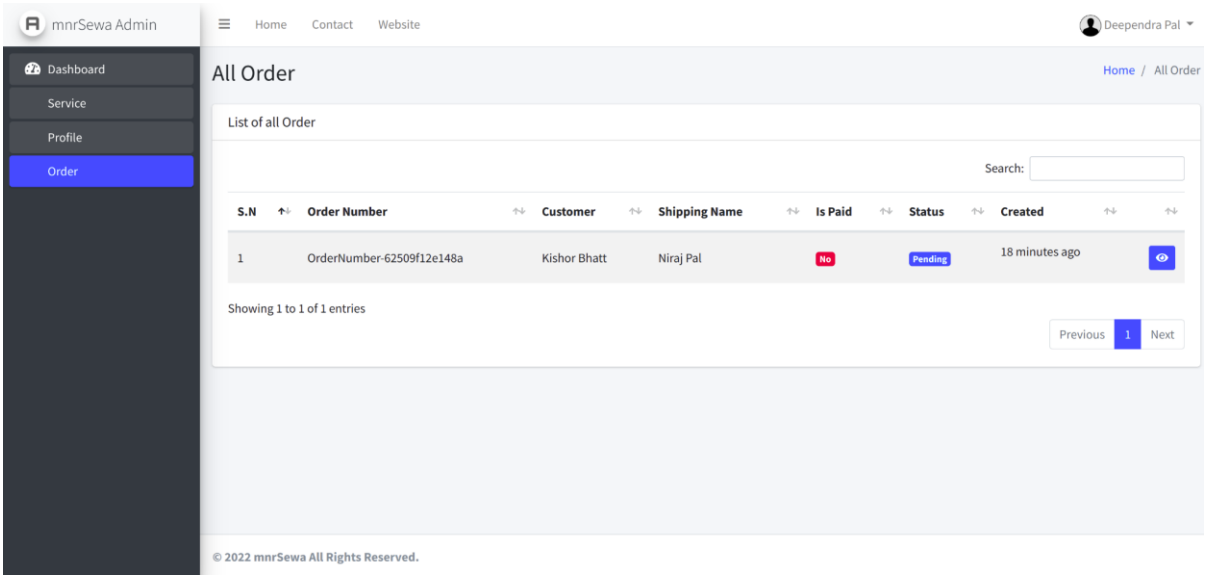


Figure: Order Placed Details

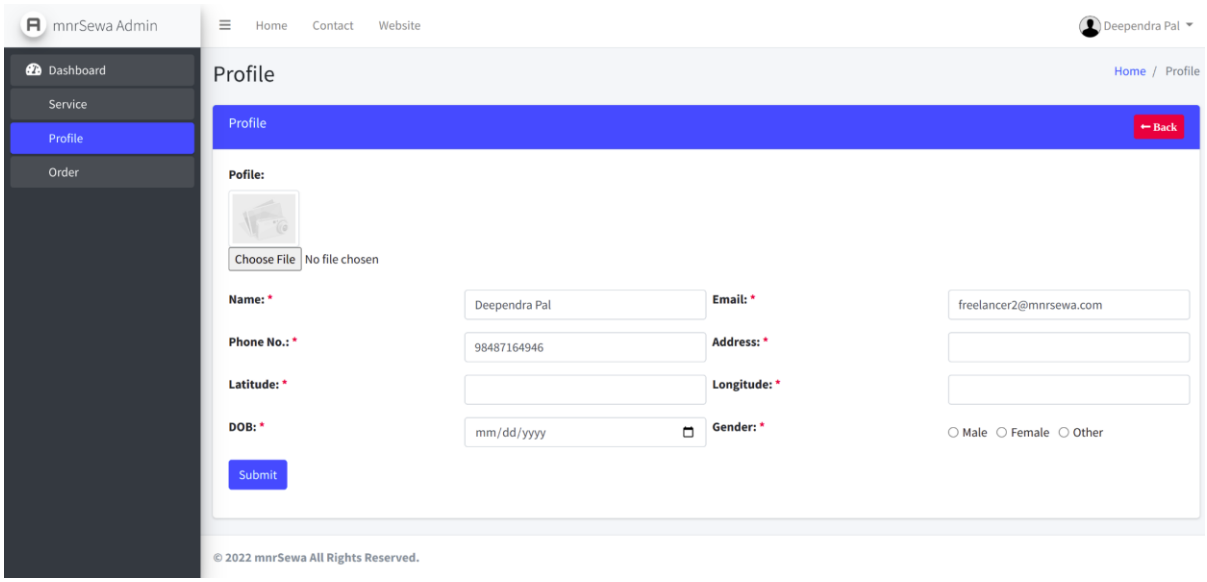


Figure: Profile Dashboard

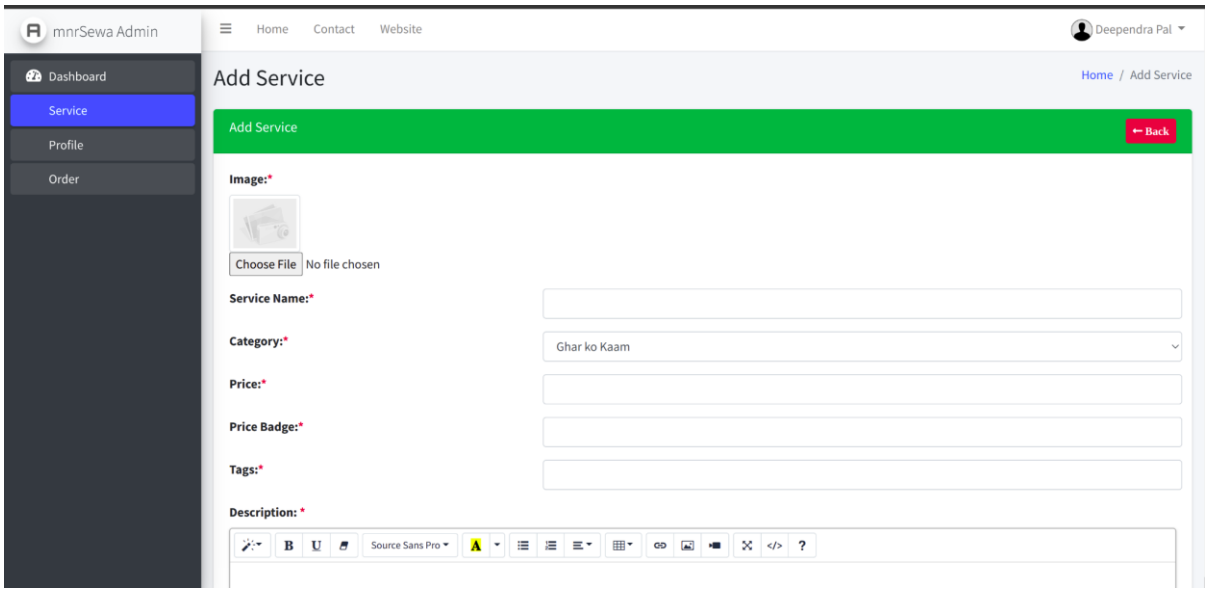


Figure: Add New Services

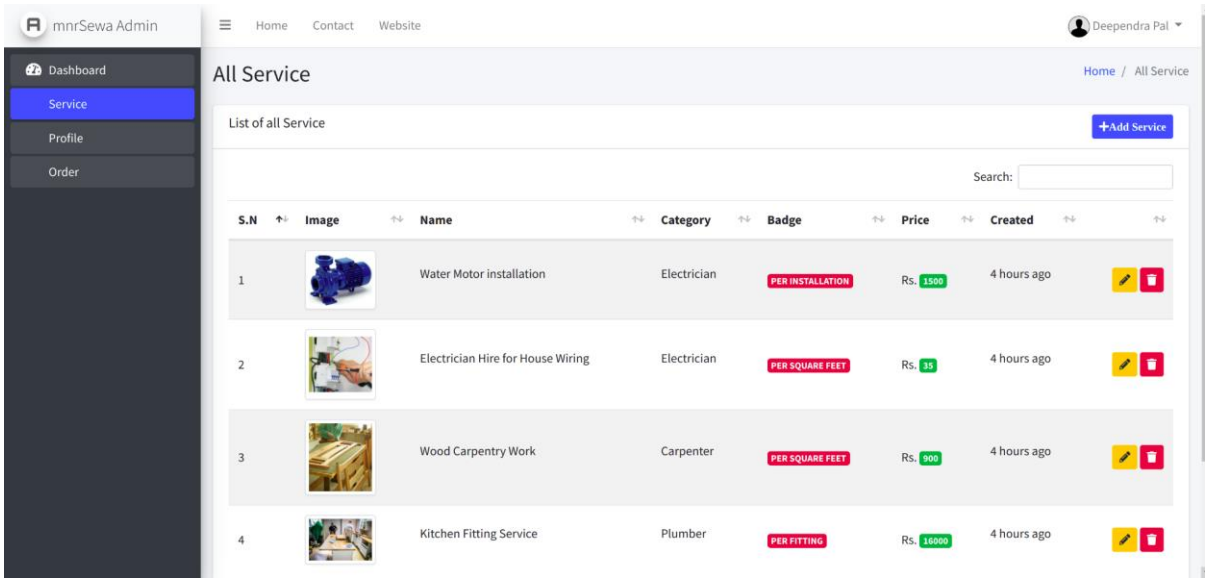


Figure: Services View