**MULTI-SERVICE PROVIDER**

|  |  |
| --- | --- |
| **SCRUM MASTER:** | **SUBMITTED BY:** |
| Ms Sona Sebastian | ABHINAND K S |
| Assistant professor | S3RMCA -A |
| AJCE,Kanjirappally | ROLL-NO: 03 |

# ABSTRACT

A multiple provider system is a web application that facilitates the seamless booking and delivery of various services such as cleaning, plumbing, laundry and more. Clients can use this system to request specific services based on their needs and service providers can then accept these requests and assign appropriate workers to perform the requested tasks. Key features include detailed service listings, flexible booking and scheduling options, worker assignment management ,notifications transparent rating and feedback mechanisms, payment processing, and reputationbuilding through reviews. By seamlessly connecting clients with skilled professionals, the system enhances convenience for clients and expands business opportunities for service providers, fostering efficient and reliable service delivery across a range of industries.

# User

Clients are individuals or businesses seeking various services through the platform. Clients have access to explore available services, select what they need, and schedule appointments according to their preferences. Once the service is completed, clients can pay for the service and rate and provide feedback based on their experience.

# Service Provider

Service providers are businesses or individuals offering a variety of services through the platform. They create and manage service listings, specifying service descriptions, pricing, and availability.

They can review the request's specifics and decide whether to accept or decline it based on their availability and expertise. If accepted, service providers assign suitable workers to perform the task.

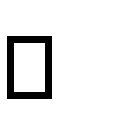
# Worker

Workers are individuals who carry out the actual tasks requested by clients. They receive assignments from service providers and have access to relevant task details such as location, time, and specific instructions.

# Admin

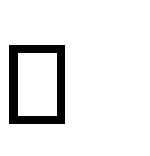
Administrators oversee and manage the overall operation of the platform. Admin verifies services from the service provider.

## MINI Project User

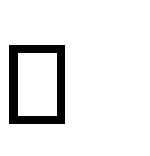
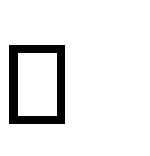
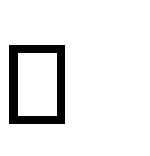
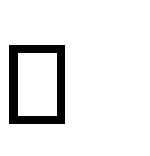
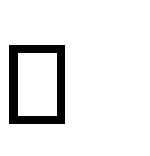
* Users view the list of services and also can search for the desired service
* Users select the service provider based on their rating
* Users schedule time for booking the service



* Users view the status of the service
* Users review and rate the service **Service providers**
* Service Provider can manage the employees working under them

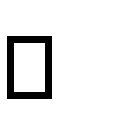


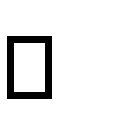
* They get notification from the users or workers



* Can view the list of available employee
* Assign work to the available employee
* View and respond to user reviews
* Manage appointment of users

## Worker

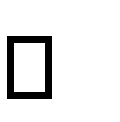
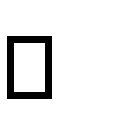
* The worker can accept or reject the work request
* Receive information of the service needed and the details of the user
* Create a report to the service provider



* Mark the completion of the service

## Admin

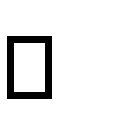
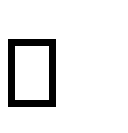
* Manage the users, service providers and workers
* Manage the booking of services
* Manage the user review and rating



# **MAIN Project**

## User

* Users pay after the completion of the service



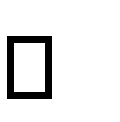
* User can reschedule or cancel the booking
* Users view the on demand services on the top of the service list(ML)

• Chatbot for user for enquiries(AI)

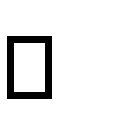
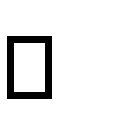
## Service providers

• Service Provider can register for new services

## Worker

• Receive real-time update about the appointment change, cancellation or rescheduling

## Admin

* Verify the new service registration by service providers
* Analyse the performance of service providers using feedback from users(ML)
* Can send important notification to users and service providers
* Can respond to user feedback

# **Technology**

Django

Machine Learning

Artificial Intelligence

References : https://verastar.co.uk [https://isg-one.com/articles/dealing-with-multiple-serviceproviders-a-necessary-evil](https://isg-one.com/articles/dealing-with-multiple-service-providers-a-necessary-evil)

Software Specifications

## Front-end : HTML,CSS,JavaScript Back-end : Python Django