TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS

A
FINAL REPORT
ON

HOME SEWA : AT-DOOR SERVICE DELIVERY SYSTEM WITH RANK BASED MATCH MAKING

SUBMITTED BY:
ARPAN POKHAREL (PUL075BCT015)
CHIRAG LAMSAL (PUL075BCT032)
BIBEK BASYAL (PUL075BCT097)
SAUGAT KAFLEY (PUL075BCT099)

SUBMITTED TO:
DEPARTMENT OF ELECTRONICS & COMPUTER ENGINEERING

APRIL, 2023

Page of Approval

TRIBHUVAN UNIVERSIY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS
DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

The undersigned certifies that they have read and recommended to the Institute of Engineering for acceptance of a project report entitled **"Home Sewa: at-door service delivery system with rank based match making"** submitted by **Arpan Pokharel**, **Chirag Lamsal**, **Bibek Basyal**, **Saugat Kafley** in partial fulfillment of the requirements for the Bachelor's degree in Electronics & Computer Engineering.

....

Surendra Shrestha, Phd.
Supervisor, Associate Professor
Department of Electronics and Computer Engineering,
Pulchowk Campus, IOE, TU.

....

Internal examiner
Department of Electronics and Computer Engineering,
Pulchowk Campus, IOE, TU.

....

External examiner
Department of Electronics and Computer Engineering,
Pulchowk Campus, IOE, TU.

Date of approval:

Copyright

The author has agreed that the Library, Department of Electronics and Computer Engineering, Pulchowk Campus, Institute of Engineering may make this report freely available for inspection. Moreover, the author has agreed that permission for extensive copying of this project report for scholarly purposes may be granted by the supervisors who supervised the project work recorded herein or, in their absence, by the Head of the Department wherein the project report was done. It is understood that the recognition will be given to the author of this report and to the Department of Electronics and Computer Engineering, Pulchowk Campus, Institute of Engineering in any use of the material of this project report. Copying or publication or the other use of this report for financial gain without approval of to the Department of Electronics and Computer Engineering, Pulchowk Campus, Institute of Engineering and author's written permission is prohibited.

Request for permission to copy or to make any other use of the material in this report in whole or in part should be addressed to:

Head
Department of Electronics and Computer Engineering
Pulchowk Campus, Institute of Engineering, TU
Lalitpur,Nepal.

Acknowledgments

We would like to express our deepest sense of gratitude towards our project supervisor, **Dr.Surendra Shrestha** for his continuous support, valuable feedback, and motivation behind the success of this project. His insightful suggestions have provided us with a wealth of wisdom, not only in completing this project but also in sharpening our skills as researchers and experimenters. We are appreciative to the Department of Electronics and Computer Engineering for providing us with such an opportunity to widen our knowledge and understanding beyond the scope of theories. We would also like to express our gratitude to all of our department's teachers and staff members who have supported us get to where we are now. All those who have inspired us to gather courage and move forward with the project will be remembered. We would like to honor those unnamed individuals, computer science researchers who, despite not being directly involved in our project, have assisted us in bringing the tools and technologies to this point.

Abstract

With the progress in technology and IT, lives have been easier. Regardless of the advancements, in the present context in Nepal, although tasks and services have been somehow available for us all, different problems have arisen like delay in services, various scams, untrustworthy service providers and much more. Home Sewa is an application to provide users an access to at-door service delivery from the expert professionals. This application is designed to serve as a platform to connect customers and provide them quality service from the professionals. On the demand of a service from a user, the matchmaking system matches the user with the appropriate service provider using Elo ranking algorithm that incorporates important parameters associated with the delivery of services by service provider such as location, elo-rating and time availability. Home Sewa uses collaborative filtering model that gives personalized recommendations based upon the usage behaviour of similar users. This platform aims to provide a convenient and seamless experience for the customers seeking expert services.

Keywords: *at-door, service, delivery, ranking algorithm, services, web-application*

Contents

List of Figures

List of Abbreviations

CBS	Customer Behaviour Score
CCA	Canonical Correlation Analysis
CMF	Correlated Matrix Factorization
GUI	Graphical User Interface
HS	Home Sewa
MAUT	Multi Attribute Utility Theory
MCA	Multi Criteria Analysis
MF	Matrix Factorization
MMR	Matchmaking Rating
SES	Servicer Effort Score
SVD	Singular Value Decomposition

1. Introduction

1.1 Background

The rapid pace of technological advancement has led to the development of innovative solutions in various sectors, including the service industry. In Nepal, where access to quality service providers can be a challenge, Home Sewa aims to bridge this gap by providing an online platform for at-door service delivery that is both convenient and reliable. The term "at-door service delivery" refers to an arrangement where services are delivered to the customers by service providers, upon their request, to their doors. Home Sewa is a platform to connect customers and freelance service providers for a wide range of service catalog which include plumbing, electrical repairs, cleaning, beauty salon, painting, carpentry, and so on. Home Sewa's concept is quite the same as ecommerce but focused on selling service rather than a product. This can potentially save time, money and the hustle for searching for professionals and having them do the work by making the entire process transparent and efficient.

1.2 Problem Statement

In the context of Nepal, the only way to call professionals for at-door services is to call a particular provider, quote for the service price, and ask them for the availability which is very tedious. It is not usually reliable for customers because the service providers may just lie about the availability and make them wait for a few days more and customers can do nothing. Furthermore, the other concerning problem is the price, they charge customers the price according to their will, not from the professional aspect.

1.3 Objectives

The objective of the project is to:

- Develop a web-based application to connect client and expert professionals to provide at-door home services.

- Implement a rank based matchmaking system to match orders and freelance service providers.

1.4 Scope

At-door service delivery is a growing need for almost all households. Delivering services at-door with few clicks will certainly help customers to save the time for searching. This will create a great experience for the working customers who want professional service and may not have time to spend searching for the best service providers to carry out certain tasks at their home. It can also be beneficial for those living away from home to help parents with the household chores that may require professionals to serve.

2. Literature Review

2.1 Related work

In recent years, there has been an increasing demand for online platforms that provide home services, such as cleaning, plumbing, and beauty services. Urban Company, founded in India in 2014, is one of the most popular platforms related to providing home services [1]. It provides customers with a variety of home services, including beauty, cleaning, and repair services. Urban Company allows customers to book services through their website or mobile application, and features a rating and review system to help customers select trustworthy service providers. Once a service is booked, a qualified and experienced professional is assigned to the job. The customer can track the professional's progress and payment is made through the platform. Urban Company ensures that all professionals on its platform go through a rigorous background check, verification process, and training program to ensure quality services to its customers. With an expanding network, Urban Company has expanded to multiple cities in India and has also launched services in other countries.

Handy, a platform that also provides home services, is based in the United States [2]. This platform provides a range of home services, including cleaning, handyman, and plumbing services. Handy allows customers to book services on-demand or schedule them in advance, and also provides a satisfaction guarantee to customers, promising to refund or re-do any unsatisfactory work. Handy's platform includes a mobile application, and it also has a website that provides access to its services.

TaskRabbit, a similar platform in the field of providing home service, is based in the United States and connects customers with skilled freelancers who can provide a wide range of services [3]. The platform features a rating and review system and allows customers to chat with service providers to discuss their needs. TaskRabbit has expanded to multiple cities in the United States and also operates in the United Kingdom. TaskRabbit operates on a peer-to-peer model, where users can post their task requirements and receive bids from qualified freelancers, known as Taskers. For Taskers, TaskRabbit offers a flexible and convenient way to find work. They can choose when and where they work, and they have control over the jobs they accept and can also set their own rates and can earn money by completing tasks that match their skills and expertise.

One of the Nepal-based service providing platform is SajiloSewa, which provides a wide range of home services to customers [4]. SajiloSewa allows customers to book services through their website or mobile application and provides a user-friendly interface for managing bookings and payments. The platform also features a rating and review system to help customers choose the best service providers. SajiloSewa is committed to providing high-quality services to its customers and ensures that all service providers are verified and trained professionals. The platform has been gaining popularity in Nepal due to its convenient and efficient services.

The Home Sewa platform, despite sharing some similarities with related platforms in providing a range of home services and connecting customers with skilled professionals, distinguishes itself with the use of rank-based matchmaking system. This ensures that customers are matched with the most appropriate service providers based on factors such as location, elo-rating, and availability. Additionally, Home Sewa's recommendation system provides personalized service suggestions based on the types of services similar customer has used.

2.2 Matchmaking System

The Elo matchmaking system, which was originally developed for chess, has been used in many different competitive games and sports as well as in ecommerce sector over the years. One of the earliest adaptations of the Elo system was for competitive Scrabble, which began using the system in the 1980s [5]. This has also been used in various other board games, such as Go and Shogi, as well as in video games, including League of Legends, Dota 2, and CS:GO. The strengths of the Elo system is its simplicity. The elo system assigns each player or team a rating based on their performance in previous matches, and then uses these ratings to predict the outcome of future matches. This allows players to be matched up against opponents of similar skill levels, which can lead to more competitive and enjoyable games. The Elo system remains a popular and effective tool for matchmaking in many competitive games and sports. By providing a simple and effective way to assess the skill levels of players and teams, the system helps to create more balanced and competitive matches, which can ultimately lead to a better overall experience for all participants. The Elo rating system, originally developed for chess, has also been applied to the e-commerce industry to predict the probability of successful transactions between buyers and sellers. The applications of the Elo rating system in e-commerce is in the field of online marketplaces. Online marketplaces such as eBay, Amazon, and Alibaba have implemented variations of the Elo system to help match buyers and sellers based on their respective reputations and transaction histories. The Elo system is used by eBay, Amazon, and Alibaba to power its recommendation engine, which is one of the key features of the Amazon shopping experience. When a customer searches for a product or browses a category, the recommendation engine uses the Elo system to analyze the customer's behavior and predict which products they are most likely to be interested in.

Weight-based matchmaking systems have been widely used in both the gaming and e-commerce industries to create personalized and efficient user experiences. A weight-based matchmaking system is a type of recommendation system that ranks users or items based on their skill level, experience, or preferences, and matches them with appropriate counterparts. Weight-based matchmaking systems have also been utilized in the gaming industry to match players of similar skill levels in online matches. Blizzard's Hearthstone matchmaking system uses an MMR (Matchmaking Rating) system to rank players based on their win-loss records, and matches them with opponents of similar skill levels. Weight-based matchmaking systems have been implemented in the education industry to match students with appropriate learning materials based on their learning preferences and abilities [6]. In the healthcare industry, these systems have been used to match patients with appropriate healthcare providers based on their medical history and needs [7].

2.3 Recommendation system

Recommendation systems are a popular and widely used technology in today's digital world. They help users find items of interest in large-scale systems, such as e-commerce websites, movie and music streaming services, social networks, and more. Recommendation systems have become an important part of online services, helping users find items of interest in large-scale systems. Amazon was one of the first companies to use collaborative filtering for its recommendation system. In 1998, it launched its "Customers who bought this item also bought" feature, which was based on item-to-item collaborative filtering[8]. Amazon's recommendation system has since evolved, incorporating both user-to-user and item-to-item collaborative filtering. The system analyzes users' purchase and browsing history, as well as the purchase and browsing history of similar users, to generate personalized recommendations.

Netflix is another company that has successfully implemented collaborative filtering for its recommendation system. In 2006, it launched the Netflix Prize [9], a competition to improve its recommendation system's accuracy. The winning solution was based on a combination of item-to-item and user-to-user collaborative filtering. Netflix's recommendation system analyzes users' viewing history and preferences, as well as the viewing history and preferences of similar users, to generate personalized movie and TV show recommendations. Collaborative filtering has proven to be a highly effective technique for improving recommendation systems across a wide range of industries. By analyzing users' past preferences and the preferences of similar users, companies can provide more personalized recommendations, improving user engagement and satisfaction.

3. General Theory

3.1 Elo-based MatchMaking Algorithm

The Elo rating system is a means of calculating players' relative skill levels in zero-sum games like chess. It bears the name of its inventor, physics professor Arpad Elo, a Hungarian-American. The Elo system was created as an improvement to the Harkness chess rating system. It is also used as a rating system in baseball, basketball, pool, table tennis, American football, association football, and numerous board games and esports. The Elo rating of a player is represented by a number that might change according to the results of rated games played. Every time a game is over, the victor subtracts points from the loser. The overall amount of points won or lost following a game is determined by the difference between the ratings of the winner and loser. Only a few rating points will be deducted from the lower-rated player if the person with the higher rating wins. However, numerous rating points will be transferred if the player with the lower rating pulls off an unexpected victory. In the event of a draw, the lower-rated player will also win a few points from the higher-rated player. This indicates that the rating system is self-improving. Players with excessively low or high ratings should eventually perform better or worse than the rating system forecasts and gain or lose rating points until the ratings accurately represent their playing ability. Elo ratings are not an absolute indicator of a player's ability; rather, they are only valid inside the rating pool in which they were derived.

Consider ratings of two players as RatingA and RatingB initially. The expected outcome of a match is calculated as in equation 3.1.

$$expected\_score = \frac{1}{1 + 10^{(RatingB - RatingA)/400}} \tag{3.1}$$

Thus the new updated rating of player A can be obtained using equation 3.2

$$new\_rating = RatingA + K * (score - expected\_score) \tag{3.2}$$

where New Rating is the updated rating of player A, K is a constant that determines the weight given to each match, score is the actual score of player A (1 for a win, 0 for a loss, or 0.5 for a draw), and expected score is calculated using the previous formula.

The constant K, also called Development coefficient or k-factor, determines how quickly a player's rating will adjust after each match. For example, a value of K = 16 means that a player's rating will change by a maximum of 16 points after each match, while a value of K = 32 means that the change can be up to 32 points. The value of K can be adjusted to give more weight to important matches, such as championship matches, and less weight to less important matches, such as practice matches. In the ELO rating system, the player's initial rating is calibrated based on their past performance. The ratings are updated after each match, and the ratings of all players will change over time as they play more matches and their skill levels change. The ELO system is designed to be self-correcting, so that as players improve, their ratings will naturally increase, while as they decline, their ratings will decrease. The ELO rating system is a powerful tool for estimating the relative skill levels of players in two-player games. It is widely used in many competitive games, including chess, Go, and video games, and it is based on a simple mathematical formula that uses the results of matches to estimate the expected outcome and adjust the ratings of the players involved. The ELO system is designed to be self-correcting, so that over time, the ratings of all players will reflect their true skill levels, leading to more accurate matches.

3.2 Recommendation System

A recommender system is a framework that filters data using a variety of algorithms and then recommends the user the information that is most pertinent to them. Systems for recommendations are useful customisation tools that are frequently current and based on the preferences of current customers. These systems have proven to be quite beneficial in a variety of e-commerce, educational, entertainment, media, book, film, and product-related fields.

3.2.1 Content-based Filtering

Content-based filtering is a type of recommendation system that utilizes item characteristics to suggest other items that are similar to ones that a user has previously shown interest in or provided explicit feedback on. This approach involves analyzing features such as genre, author, artist, or keywords of items that the user has interacted with and using that information to recommend other items that share similar characteristics.

The recommendation system of Home Sewa works by building a profile of the user's preferences based on the features of the items they have previously consumed, and then using that profile to find other items with similar features. For example, if a user frequently reads science fiction novels, the system will recommend other science fiction novels with similar themes or authors. This approach is particularly useful when there is a large amount of data available about the items being recommended and when there is a need to make recommendations based on specific characteristics of those items.

The content-based filtering can provide personalized recommendations to users based on their previous actions or explicit feedback, by utilizing item features to suggest similar items that the user may be interested in. It lacks to provide suggestions based upon the similar user activities .

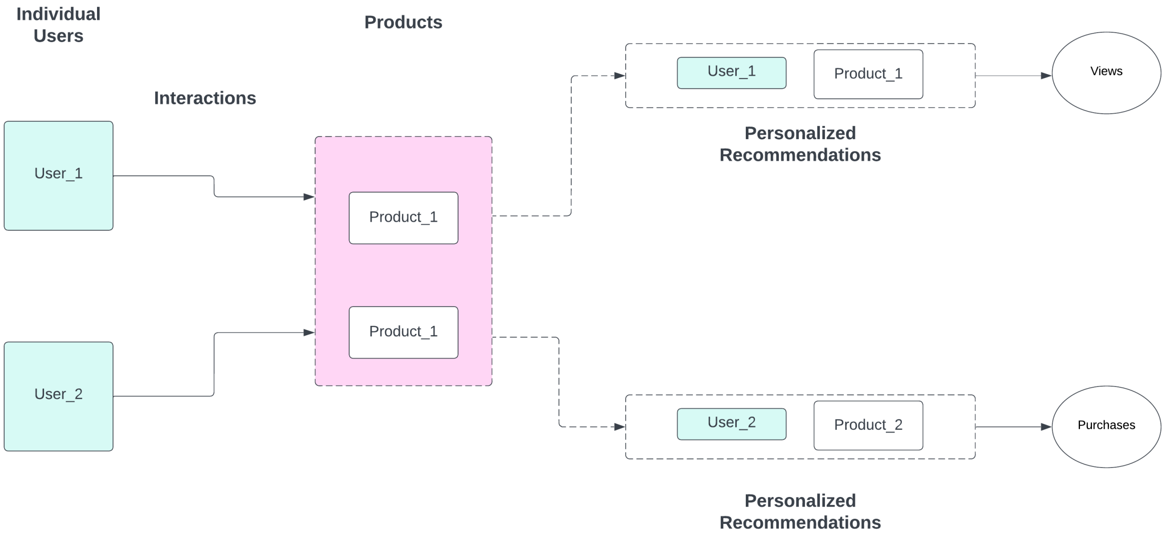


Figure 1: Block diagram for Content Based Filtering

3.2.2 Collaborative Filtering

Collaborative filtering is a recommendation system that utilizes similarities between both users and items simultaneously to provide personalized recommendations. This approach differs from content-based filtering, which only considers the features of items when making recommendations.

In collaborative filtering, the system analyzes the behavior of multiple users to identify patterns and similarities in their preferences. It then uses this information to recommend items that other users with similar preferences have enjoyed. This method allows for serendipitous recommendations, where the system can suggest items to a user based on the interests of a similar user, even if the items are not explicitly related. To achieve this, collaborative filtering models embeddings are used, which are low-dimensional representations of users and items that capture their preferences and characteristics. These embeddings can be learned automatically through machine learning algorithms, without relying on hand-engineering of features.

Considering a movie recommendation system in which the training data consists of a feedback matrix in which each row represents the user data and the column represents an item which corresponds to a movie. The feedback about movies falls into one of two categories:

- **Explicit** — users specify how much they liked a particular movie by providing a numerical rating.
- **Implicit** — if a user watches a movie or consumes an item , the system infers that the user is interested.

To simplify, assume that the feedback matrix is binary; a value of 1 indicates interest in the movie and 0 indicates the user has unobserved or has not rated the movie.

When a user visits the homepage, the system should recommend movies based on both:

- similarity to movies the user has liked in the past
- movies that similar users liked

A scalar value between -1 and 1 is assigned to each movie, indicating whether the movie is intended for children (negative values) or adults (positive values). Similarly, each user is assigned a scalar value between -1 and 1, indicating their interest in children's movies (closer to -1) or adult movies (closer to +1). The movie embedding and user embedding are then multiplied to calculate a score. Movies with a higher score (closer to 1) are expected to be liked more by the user.

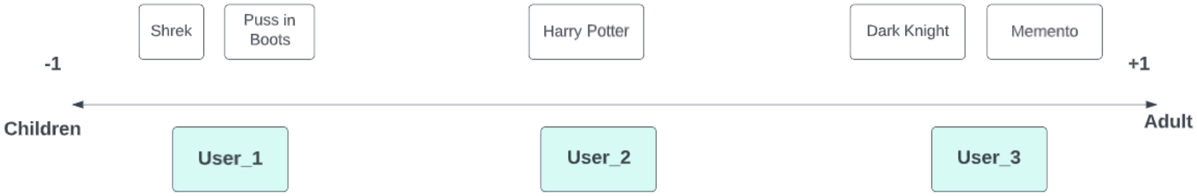


Figure 2: 1D Embedding Representation

In Figure 3, each checkmark identifies a movie that a particular user watched. The third and fourth users have preferences that are explained by this feature. The third user prefers movies for children and the fourth user prefers movies for adults. However, the first and second users' preferences are not well explained by this single feature.

		-.2	-.8	-.1	-.9	1
		Movies				
.1	u1	✓		✓	✓	
0	u2		✓			✓
-1	u3	✓	✓	✓		
1	u4				✓	✓

Figure 3: Rating Matrix

3.2.3 2D Embedding

One feature was not enough to explain the preferences of all users. To overcome this problem, a second feature is added: the degree to which each movie is a blockbuster or an arthouse movie. With a second feature, it can now be represented each movie with the two-dimensional embedding as shown in Figure 4.

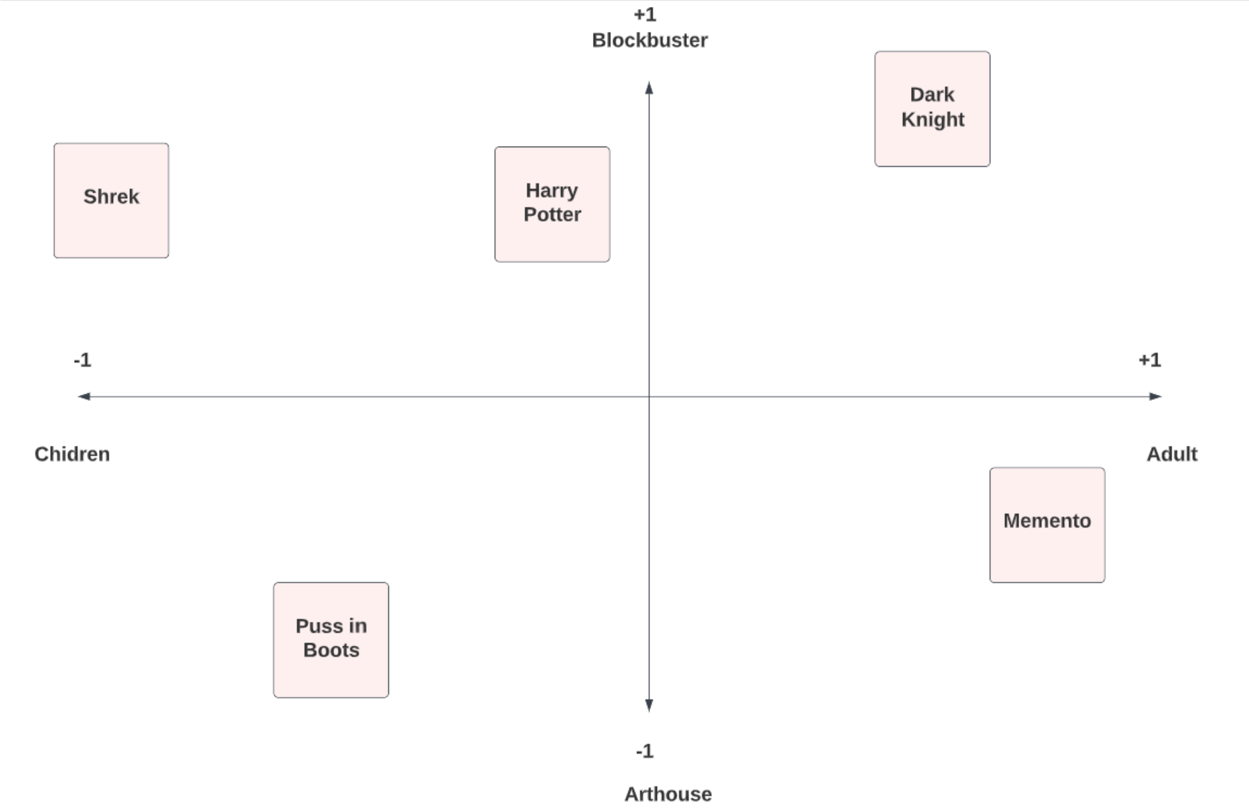
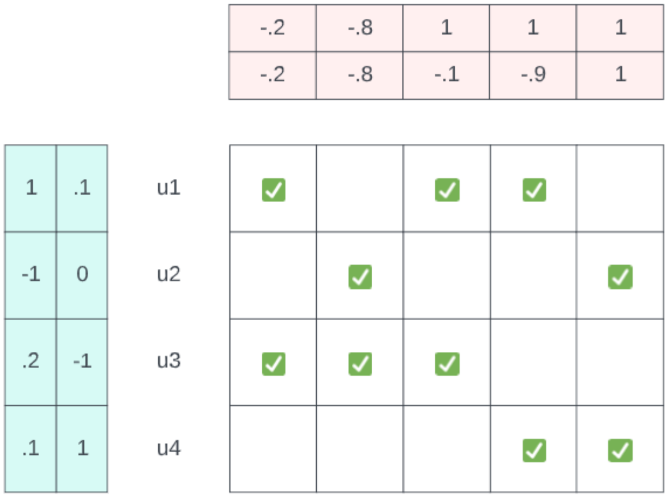


Figure 4: 2D Embedding

The Feedback Matrix: For each (user, item) pair, the dot product of the user embedding and the item embedding is assigned close to 1 when the user has watched the movie, and to 0 otherwise.



-.2	-.8	1	1	1
-.2	-.8	-.1	-.9	1

1	.1	u1	✓		✓	✓	
-1	0	u2		✓			✓
.2	-1	u3	✓	✓	✓		
.1	1	u4				✓	✓

Figure 5: Collaborative Matrix

The collaborative nature of the approach is evident in the process of learning embeddings. When the embedding vectors for the movies are fixed, the model can learn the embedding vector for each user to best represent their preferences. This process results in embeddings of users with similar preferences being located close together in the embedding space. On the other hand, when the embeddings for the users are fixed, the model can learn the movie embeddings that best represent the feedback matrix. As a result, embeddings of movies liked by similar users will be located close to each other in the embedding space.

This approach is collaborative in that it utilizes the feedback and preferences of multiple users to learn and improve the embeddings of both users and items simultaneously. By doing so, it can provide more accurate and personalized recommendations to users based on the preferences of other similar users, and the characteristics of items they have previously engaged with .

3.2.4 Matrix Factorization

Matrix factorization is considered a simple embedding model that involves factoring a matrix into two lower-dimensional matrices that represent user and item embeddings. This factorization process is performed by minimizing the difference between the original matrix and its approximation using the lower-dimensional matrices. As a result it can provide personalized recommendations to the users as well as recommend popular items that other users have been using that might have similar usage behavior to that user. Given the feedback matrix $A \in R^{mxn}$, where:

- m is the number of users (or queries) and
- n is the number of items

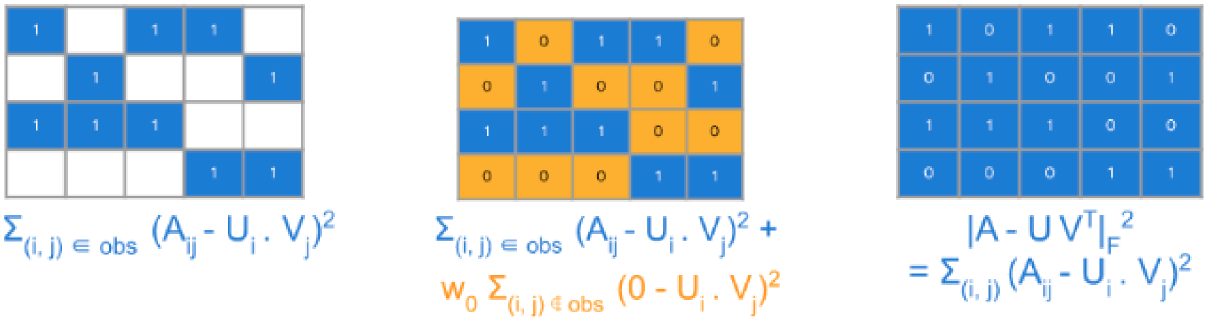
The embeddings are learned such that the product UV^T is a good approximation of the feedback matrix A. The Observed entry of (i , j) is $U.V^T$ simply the dot product of the embeddings of user and item, which you want to be close to $A_{i,j}$.

3.2.5 Choosing Objective Function

One intuitive objective function is the squared distance. Minimizing the sum of squared errors over all pairs of observed entries, as in the equation 3.3.

$$min(U\epsilon R^{m\times d}, V\epsilon R^{n\times d}) \sum_{(i,j)\epsilon obs} (A_{ij} - \langle U_i, V_j \rangle)^2 \tag{3.3}$$

In the objective function, the sum is observed over pairs (i, j), that is, over non-zero values in the feedback matrix. However, only summing over values of one is not a good idea, a matrix of all ones will have a minimal loss and produce a model that can't make effective recommendations and that generalizes poorly.



1		1	1	
	1			1
1	1	1		
			1	1

$\Sigma_{(i, j) \in obs} (A_{ij} - U_i . V_j)^2$

1	0	1	1	0
0	1	0	0	1
1	1	1	0	0
0	0	0	1	1

$\Sigma_{(i, j) \in obs} (A_{ij} - U_i . V_j)^2 +$
$w_0 \Sigma_{(i, j) \notin obs} (0 - U_i . V_j)^2$

1	0	1	1	0
0	1	0	0	1
1	1	1	0	0
0	0	0	1	1

$|A - U V^T|_F^2$
$= \Sigma_{(i, j)} (A_{ij} - U_i . V_j)^2$

(a) Observed-only MF

(b) Weighted MF

(c) SVD

Figure 6: SVD representation

3.2.6 Regularization

Regularization is a machine learning technique that is commonly used to prevent overfitting, which occurs when a model becomes too complex and fits the training data too closely, resulting in poor performance on new, unseen data. In the context of matrix factorization, regularization involves adding a penalty term to the objective function that encourages the learned factors to have a certain property, such as sparsity or smoothness. This penalty term helps to control the complexity of the model and can improve its generalization performance by reducing overfitting.

Regularization of the model parameters. This is a common l2 regularization term on the embedding matrices, given by equation 3.4.

$$r(U,V) = \frac{1}{N}\sum_i \|U_i\|^2 + \frac{1}{M}\sum_j \|V_j\|^2 \tag{3.4}$$

3.3 Celery

Celery is a Python-based task queue that allows developers to perform tasks asynchronously, without blocking the main thread [10]. Tasks are broken down into smaller, independent units of work that can be performed concurrently, allowing developers to improve application performance and scalability.

Celery uses a distributed architecture that consists of a client, a broker, and workers. The client is responsible for submitting tasks to the task queue, while the broker is responsible for managing the tasks and distributing them to workers. Workers are responsible for executing the tasks and returning the results to the broker. Celery supports multiple brokers, including RabbitMQ, Redis, and Apache Kafka, which allows developers to choose the broker that best fits their needs. Additionally, Celery supports multiple result backends, including Redis, MongoDB, and SQL databases, allowing developers to store task results for later.

One of the key benefits of Celery is its ability to improve application performance and scalability. By offloading time-consuming tasks to the background, the main thread can continue processing other user requests, leading to faster response times and a better user experience. Additionally, Celery can scale horizontally by adding more workers to the task queue, allowing developers to handle an increasing number of requests without sacrificing performance. Celery also supports task routing, which allows developers to route tasks to

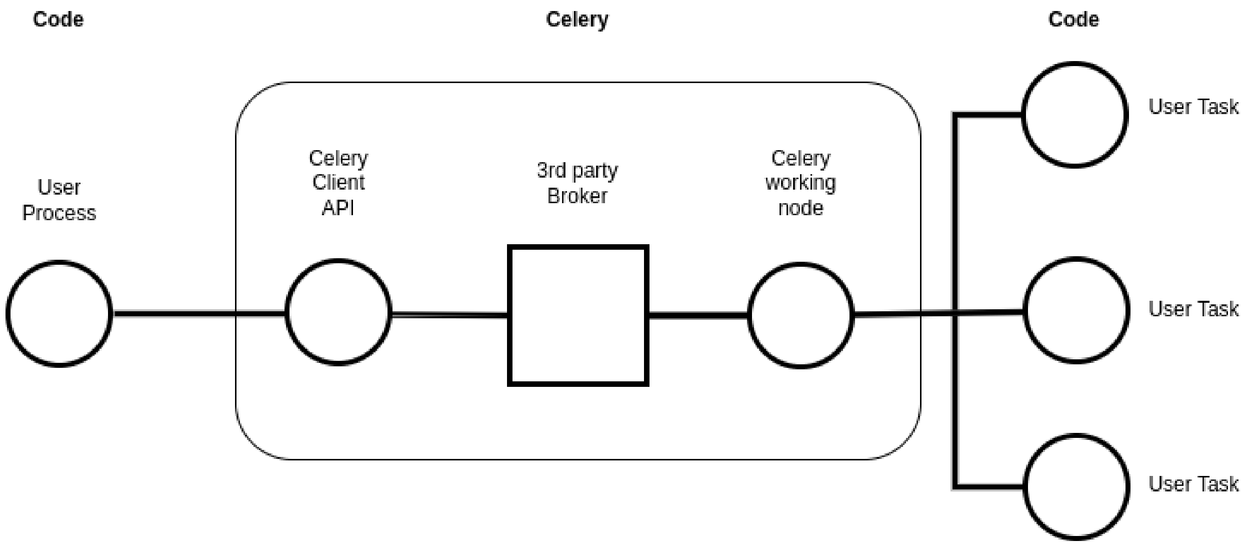


Figure 7: Working Principle of Celery

specific workers based on various criteria, such as task priority, worker availability, and task type. This makes it possible to optimize the task execution process and ensure that tasks are processed as efficiently as possible.

4. Hardware and Software Requirements

To develop a service delivery platform like Home Sewa various hardware and software components are needed to work together co-ordinating with each other. These components must serve the objective of the platform providing users with a reliable service uptime and persistence performance whenever needed. These components include:

4.1 Web Server

For anyone on the internet to access the system, it needs to be served in a computer which can manage, allow connections to the application and handle the requests. This computer which has a wide range of responsibility to make the application accessible is referred as the web server. Web Servers can be both with linux OS and a windows OS. But, the more preferrable option is to go with the linux OS because it is more managable and grants the administrator more grasp to the underlying features of the system. For Home Sewa, a linux web server with ubuntu as the OS is used. Typical configuration of the server would look like as: 2v CPU, 4 GB RAM. This server is responsible for handling requests for application users in the beginning phase of release.

4.2 Database

The database is an essential component for any web application. It is responsible for storing and managing the data that the application uses. For Home Sewa, a SQL-based database called PostgreSQL is used. PostgreSQL is a powerful database that can handle large amounts of data and many connections simultaneously. The hardware requirements for the database depend on the expected number of connections and the amount of data to be stored.

4.3 Celery

Celery is a task queue that is used for distributing tasks across a cluster or multiple machines. It is written in Python and requires the installation of the Celery package. Celery works with a message broker, and Redis is used as the message broker for the Home Sewa project. Redis is a reliable and scalable message broker that can handle large volumes of messages and connections.

4.4 Redis

Redis is an in-memory data store that is used for caching frequently accessed data. It is often used to store key-value pairs, counters, and other types of data. For Home Sewa, Redis is used as a cache to speed up database queries and improve the overall performance of the application. The hardware requirements for Redis depend on the expected number of connections and the amount of data to be stored.

4.5 Matchmaking Server

The matchmaking server is responsible for matching service requests from users with the appropriate freelancer service providers. It is built using Python and the Flask web framework. The server exposes a RESTful API that allows it to communicate with other components of the system. The hardware requirements for the matchmaking server depend on the expected number of connections and the amount of data to be processed.

4.6 Recommendation Server

The recommendation server is responsible for providing personalized recommendations to users based on their previous service requests. It uses a collaborative filtering algorithm to generate recommendations. The server is built using Python and the Flask web framework. The hardware requirements for the recommendation server depend on the expected number of connections and the amount of data to be processed.

4.7 Frontend

The frontend of the Home Sewa is an web-application which can be acessed from a browser from a computer or a mobile phone. For the frontend part, popular javascript library called ReactJS has been used.

4.8 Backend

Backend is the most important part of an application. As our application has matchmaking and recommendation system integrated, it is crucial that we choose a backend which is able to perform well for our requirements preserving the fun to develop it. A popular python web framework called Django is used as the backend of this application because python being ideal to work with.

5. Methodology

5.1 System Architecture

System Architecture of HomeSewa follows a three-tier architecture which includes a client, a server and a database layer. Three tier architecture is suitable for horizontal scalability and provides a layer level flexibility to incorporate large and dependent features in the system. The overall architecture of the HomeSewa system can be seen in Figure 8.

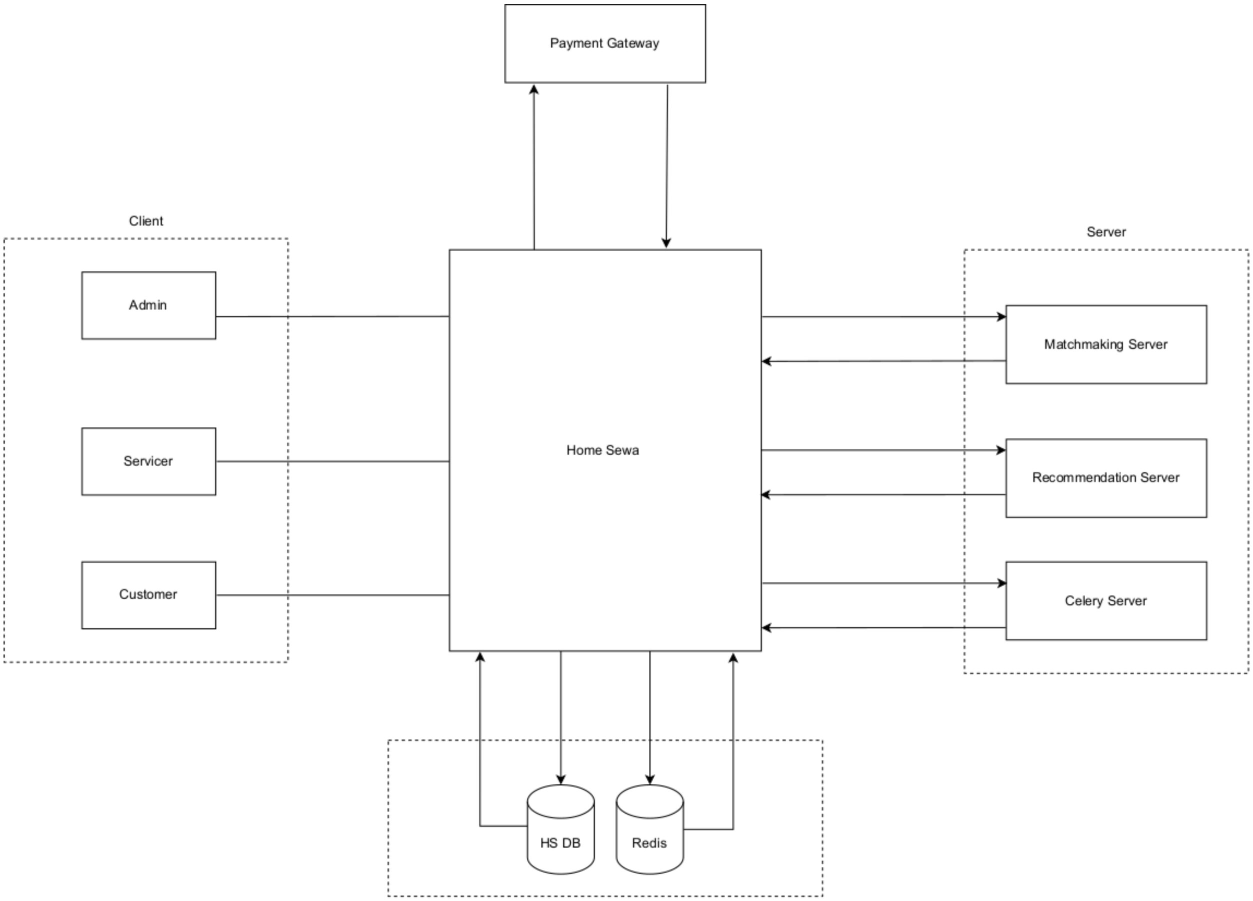


Figure 8: System Block Diagram of Home Sewa

The Home Sewa system can be basically divided into four subsystems:

Client:

A client is a graphical user interface with which users interact with the system. Home Sewa client system provides an interface for users according to their roles. Roles can be customer, service provider or an administrator of the Home Sewa. Each interaction can be described as a view of the client system. Any user, to access the system must be authenticated from the server. For this, they are first needed to register to the system with the required details and also are required to verify their contact details. After the registration process, an account for the asked role whether customer or service provider. With those account credentials users can now access their dashboard presented as a view. These views can be described as:

1. **Customer View**: As the name suggests, the customer interactions refer to as customer view. Home Sewa allows a customer to order a service, view its details, pay for the order and provide feedback to the service providers. After login, the customer can access the dashboard and can get service recommendations according to the customer's activity history or the preferences. Customers can also search for the required service according to their need. For the selected service, the customer orders the service by specifying the address, appointment time and service type. For the required service selected by the customer, matchmaking is performed. Matchmaking algorithm matches the service provider based on the parameters (location, appointment time, availability and elo-rating). After the delivery of the service the customer makes payment to the Home Sewa system through the supported payment gateway. The customer rates the servicer according to their performance. These ratings are used for determining the ranks which helps in future matchmaking.

2. **Servicer View**: In order to provide a service, a servicer has to be registered. In the registration form necessary details such as name, address, citizenship. They should also provide the relevant certificates of skills completion and previous work experience. After filling up the signup form it goes for verification which is done by the Home Sewa staff and admins. When the servicer logins, he/she can see the recent order requests in the dashboard. A servicer is prompted with a new order request if a customer has requested for a service matched from the match-making algorithm. After the job has been accepted by the servicer, a delivery status is generated which is sent to the customer. The servicer responds with the appointment details i.e, date and time . After the service has been provided to the customer, an invoice is generated according to the service cost and any extra materials used. The customer can see the generated invoice by the servicer. After successful payment, the servicer receives the payment from the supported payment gateway. The customer and servicer rate each other according to the quality of service provided, performance and experience. The ratings play a crucial

role in the recommendation and suggestions to other similar customers. They are also used in determining the preference(rank) in the match-making process.

3. **Admin View**: An admin is able to manage all the users including servicers and customers and also the services provided . Admin can verify newly registered users and can allow selected users to work in the platform. Admin also provides the transaction management payment mechanism. In this account, the accounting admin is responsible for receiving the service payment from the customers and correspondingly providing the pay to the service providers. Also, the admin is able to monitor and view the reports and analytics for the services or users or other entities aiding in decision-making. Admin is also responsible for managing the underlying data and making manipulations whenever necessary in the system.

Server:

The server is the most integral part of the Home Sewa System. It is responsible for handling all the data processing and client requests. In the homesewa system, the server is responsible for Matchmaking, Recommendation, Tasks execution. The payment is handled by the respective payment gateway of the choice. Matchmaking is done based on the skills and experience of the service provider. The recommendation system works on the data of the customers' previous orders history. All the incoming requests in the server are served in the incoming order. It is handled by Celery. Furthermore, an in-memory database called redis has been used to handle the caching, channeling and acts as a broker to improve the server performance.

Database:

Home Sewa uses PostgreSQL in order to store data. PostgreSQL is a relational database management system which is efficient for handling more database connections and provides flexibility options during scaling.

5.2 Matchmaking System

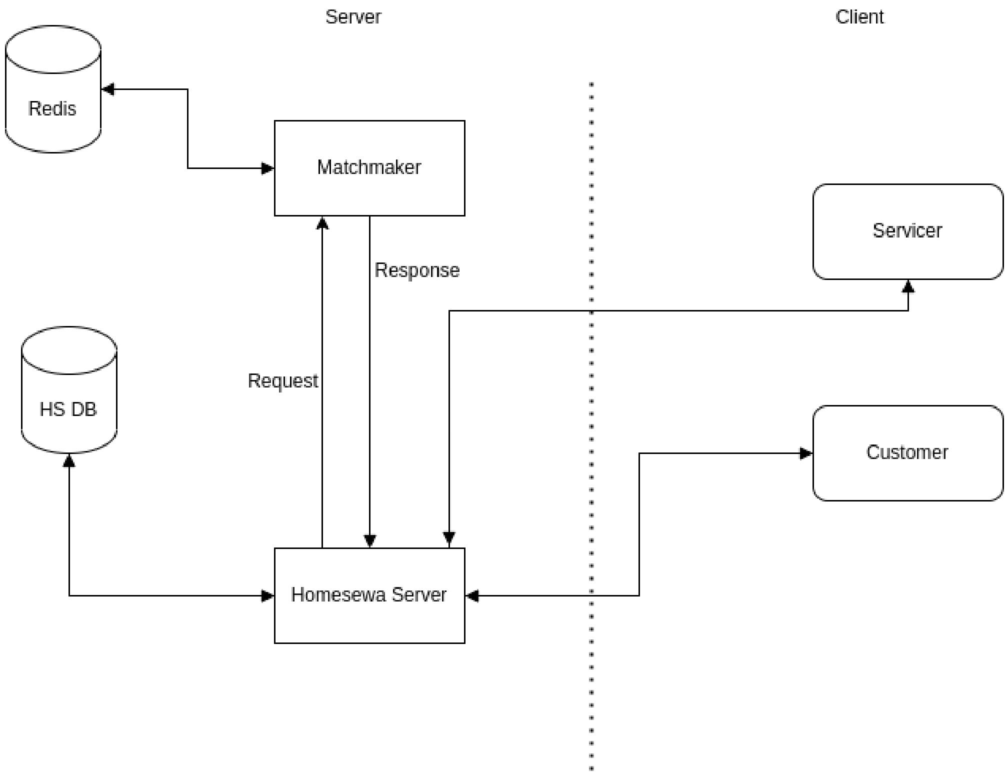


Figure 9: Matchmaking Flow Diagram

Home Sewa Server: Home Sewa Server is the central server that is responsible for managing and coordinating all of the other servers in the Home Sewa platform. It communicates with the matchmaking server to facilitate the matchmaking process between customers and service providers.

Matchmaking Server: Matchmaking Server is responsible for performing the matchmaking process between customers and service providers based on their preferences and availability. The matchmaking server receives requests from the Home Sewa server and returns the best match for each request. Matchmaking server also performs User Elo Calibration and Initialization, which is a measure of a user's skill level and is used in the matchmaking process to ensure that customers are matched with service providers of a similar skill level. The Home Sewa platform uses an Elo rating system to calculate and update user skill levels. The Home Sewa server is responsible for calibrating and initializing each user's Elo rating when they first join the platform. As users continue to use the platform and complete jobs, their Elo rating will be updated to reflect their current skill level. The Home Sewa server is responsible for updating each user's Elo rating and providing a rank or score based on their skill level. This allows customers to easily identify the most skilled service providers on the platform.

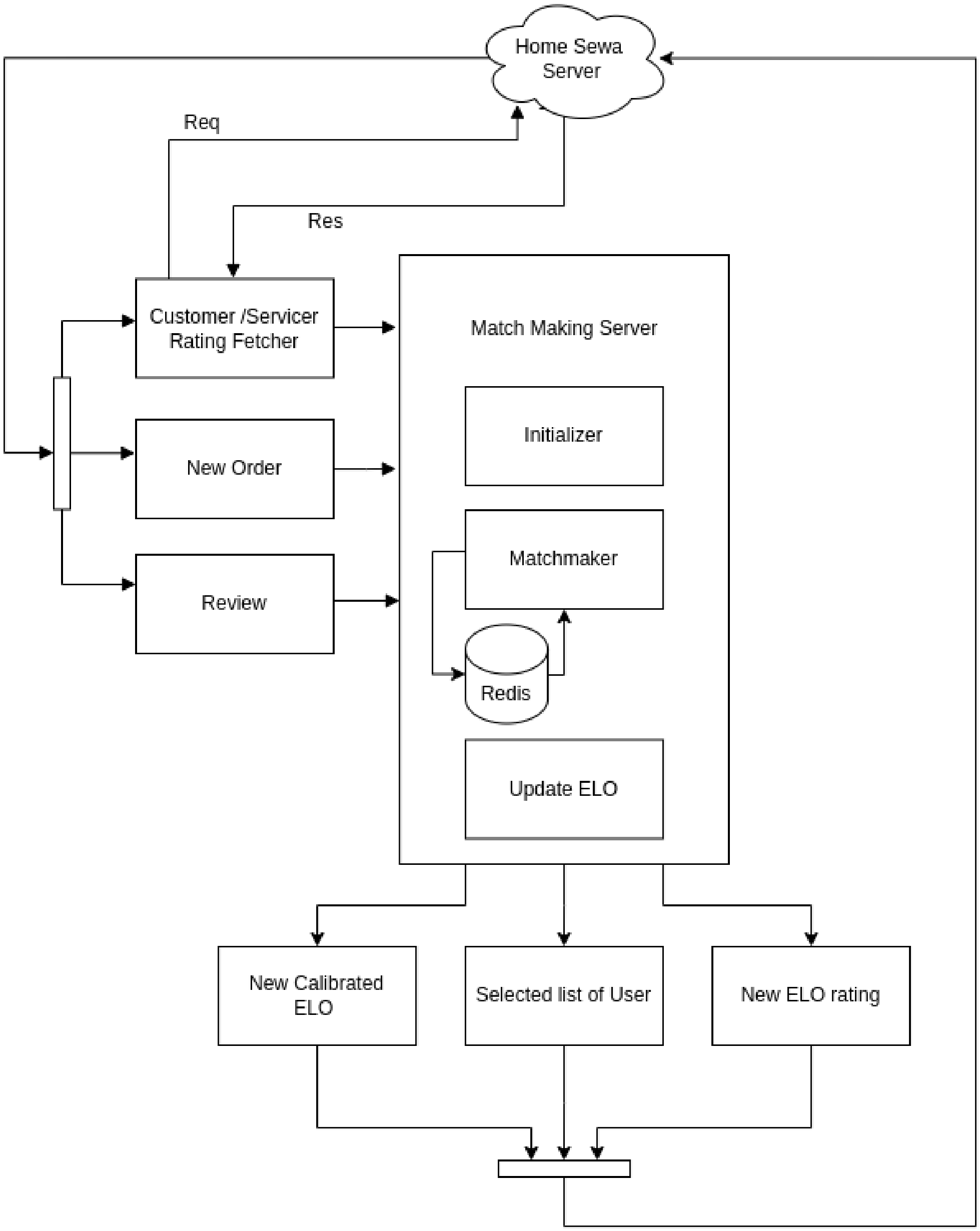


Figure 10: Block Diagram of Home Sewa Matchmaking Server

Matchmaker

In the MatchMaker block, the actual matchmaking process takes place. In Figure 10 when Home Sewa server sends the request for the matchmaking, the matchmaker block gets triggered. The request from the Home Sewa contains the order information, customer's list with their necessary information(i.e skills, availability, elo points, location). At first location is filtered out. Servicers within the radius of 10KM are filtered out whereas others are disqualified. Then Skills are matched between the servicer and orders. Servicers having similar skill are filtered. Then on the basis of elo points the servicers are matched. The matchmaking block sends response the list of users that are matched to the home sewa server.

Update ELO

The update of elo is performed in Home Sewa system in the flow as: when a servicer completes the order, a form is provided to the customer which calculates the Servicer Effort Score (SES) and a form is provided to the servicer which calculates the Customer Behaviour Score (CBS).Then expected score is calculated using the expected formula. Then the new rating of the customer or servicer is then updated.

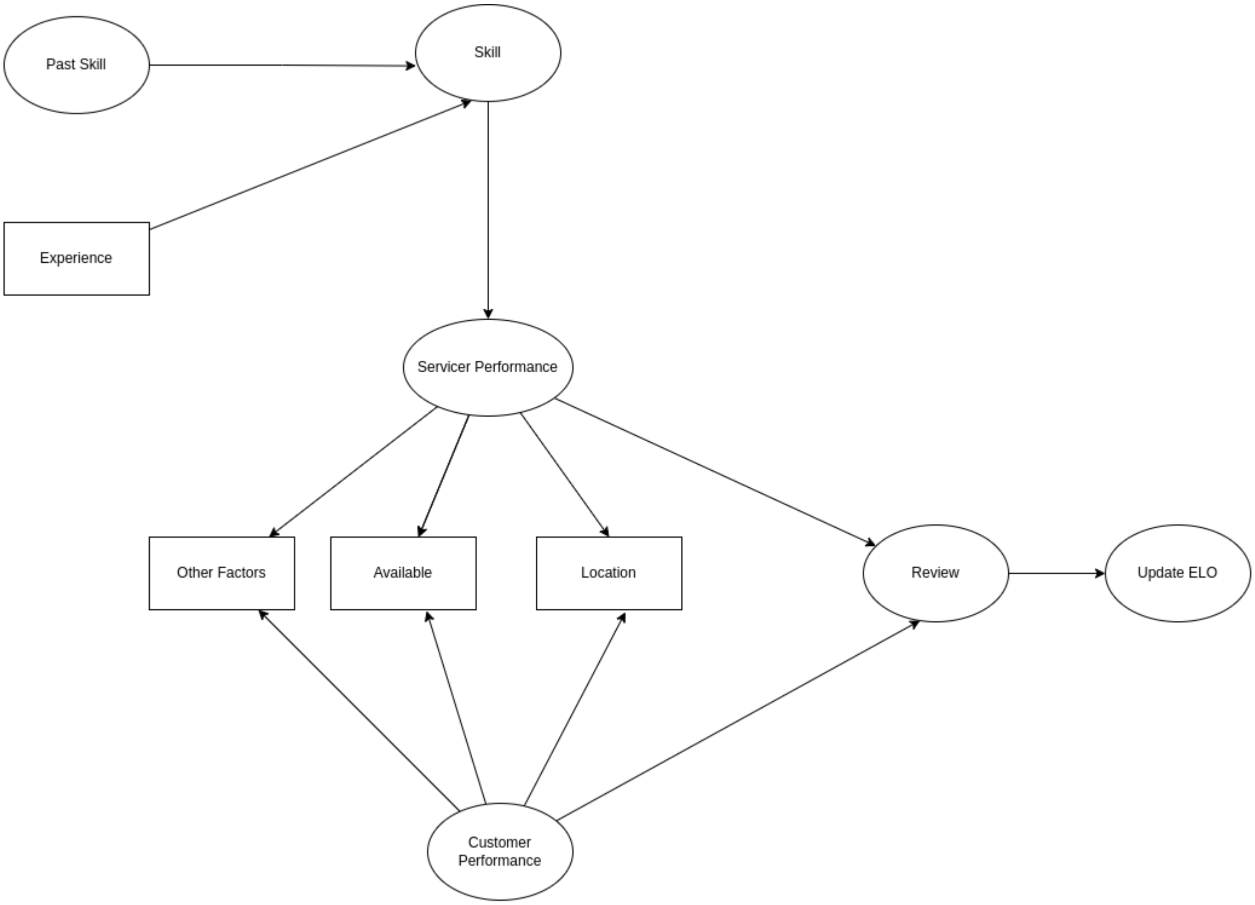


Figure 11: Flow diagram for Updating ELO rating

Factors for Matchmaking

The availability of both customers and service providers was determined by checking their profiles for their preferred working hours, days, and time zones. The algorithm then checked for overlapping availability and matched customers with service providers whose schedules aligned.

The algorithm checked the online status of both customers and service providers. This information was obtained from their profiles or by tracking their activity on the platform. The algorithm matched customers with service providers who were currently online or had been active recently to increase the chances of a successful match.

The algorithm also took into account the skills of both customers and service providers. This information was obtained from their profiles, where they listed their areas of expertise, experience, and qualifications. The algorithm checked for overlapping skills and matched customers with service providers whose skills complemented each other.

Location was another important criterion used by the algorithm to match customers with service providers. This information was obtained from their profiles, where they listed their city, state, or country. The algorithm checked for proximity and matched customers with service providers who were located in the same or nearby regions to reduce the time and cost of communication and transportation.

The algorithm considered the elo ratings and reviews of both customers and service providers. This information was obtained from their profiles or by collecting feedback from previous clients. The algorithm matched customers with service providers who had similar ratings and positive reviews to increase the likelihood of a successful match.

HomeSewa Matchmaking Flow

The matchmaking flow in the Home Sewa system begins with the customer. When the customer requests for a service, the matchmaking process is initiated. First of all, from the list of the available servicers that are open to work are filtered out. Then the servicers that fall under some radius are determined using Haversine distance formula, which uses the coordinates of the customer as the origin and finds all the servicers within the radius. The servicers are then arranged on the basis of the similarity between them. The more the servicer's ELO rating is closer to the customer, the more is it likely that the customer

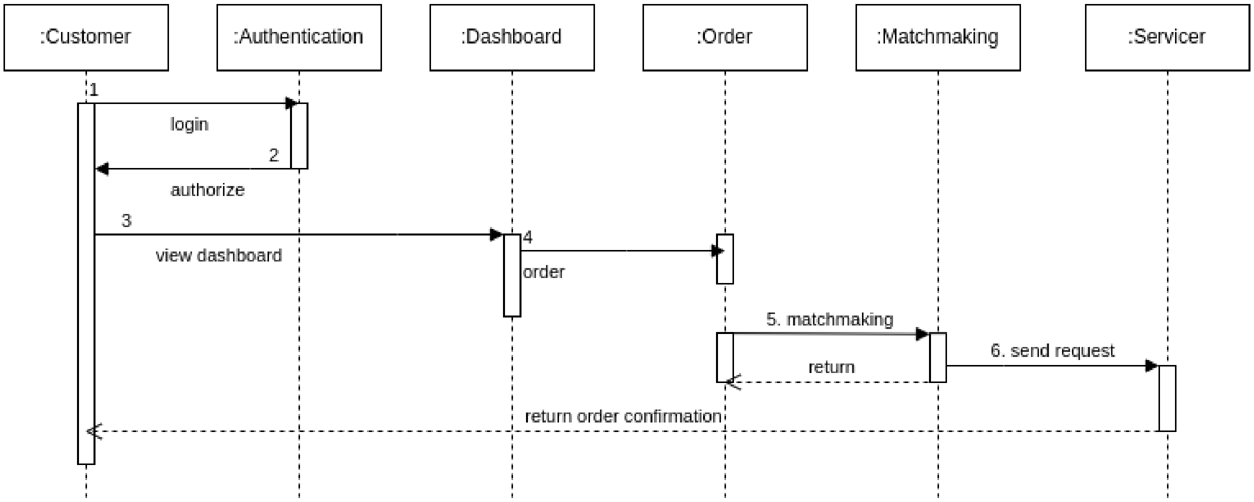


Figure 12: Matchmaking Sequence diagram

will have similar or even better service experience. For any matchmaking process, the ELO rating score of both servicers as well as customers is required. Therefore, their ELO has to be initialized according to their skills and experience in their respective jobs. The matchmaking server will inform all the potential candidates (servicers) for the order delivery via SMS or email so that the servicer gets informed that there is an available order. After the completion of the job, the payment of the order has to be completed. The servicer and customer will be requested to give review of the order experience which helps in updating the ELO rating.

Consider ratings of customer and servicer as Rating-A and Rating-B initially. The expected outcome of an order is calculated using the equation 5.1.

$$expected\_score = \frac{1}{1 + 10^{(RatingB - RatingA)/400}} \tag{5.1}$$

where RatingA and RatingB are the elo ratings of servicer and customer Thus the new updated rating of customer can be obtained as in equation 5.2

$$NewRating = RatingA + K * (score - expected\_score) \tag{5.2}$$

where New Rating is the updated rating of a customer, K is a constant that determines the weight given to each match, score is the actual review of a person that has been given to the servicer and expected score is calculated using the previous formula.

5.3 Entity Relationship Diagram

The Home Sewa ER diagram demonstrates the relationship between the entities of the system.

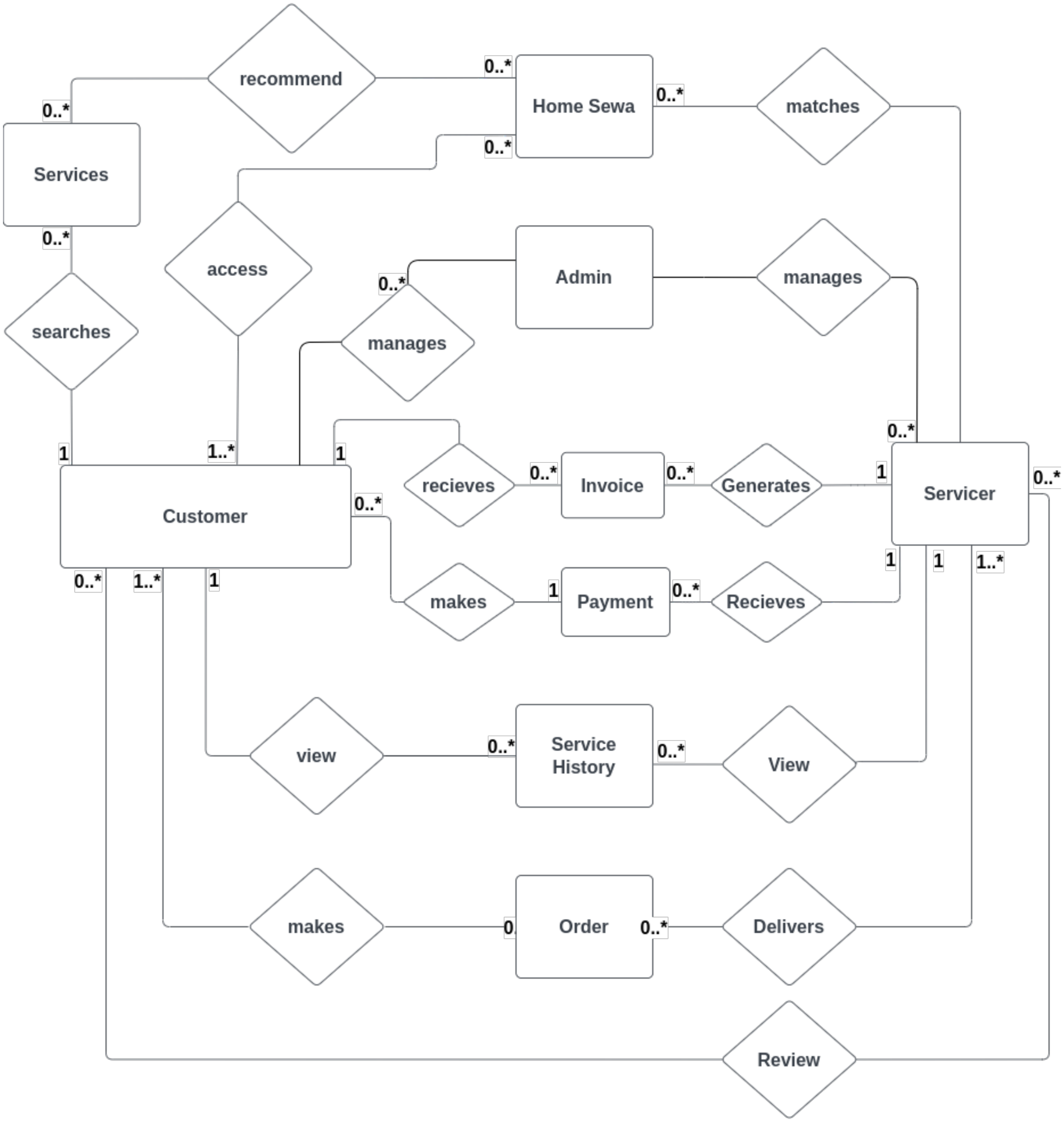


Figure 13: Entity-Relationship Diagram

The ER diagram for a Home Sewa shows how it caters the data used in various service and management activities via database design (ERD). It reveals the database relationships between the system's entities. This demonstrates database logical structure by identifying entities, properties, and interactions. This ER Diagram is a graphical representation of all System entities' relationships. The ER Diagram is used to design and troubleshoot the relational database of the system. The entities of Home Sewa are defined as:

- **Services**: Services refers to the various types of services that are being delivered to the customer's doorstep.

- **Servicer**: A servicer refers to the individual who is responsible for providing the at-door service. They are a service provider, and they are responsible for delivering the ordered goods or providing the service.

- **Order**: An order refers to the request made by the customer for a service. It contains the details of the item ordered, the price, and the delivery address.

- **Customer**: A customer refers to the person who places the order for the at-door service. They could be an individual or a business entity, and they are responsible for providing accurate information about their order, including their contact details and delivery address.

- **Invoice**: An invoice is a document generated by the servicer and customer that provides the details of the services, the price, and the payment details. It can be viewed by the customer after the service has been provided or fulfilled.

- **Payment**: Payment refers to the exchange of money between the customer and the servicer for the at-door service. It could be made through various payment methods, such as cash on delivery or online payment.

- **Service History**: Service history refers to the record of all the services that a customer has received from a particular service provider.

The Relationship of Home Sewa are defined as:

- **Servicer:**

  - A Servicer can have a profile that lists their name, contact information, and other details about their qualifications and experience.

  - A Servicer can have a rating or feedback score based on the quality of their services and their interactions with Customers.
  - A Servicer can be assigned to multiple Orders, but can only perform one service at a time.
  - A Servicer can view their Service History to see all the Orders and Services they have completed in the past.
  - A Servicer can receive Payments for completed Services, and can view their Payment History to see all the payments they have received from the Homesewa system.

- **Order:**
  - An Order is created by a Customer and includes information about the requested service.
  - An Order can have one or more Services, which are assigned to a Servicer for completion.
  - An Order can have a status that tracks its progress, such as Pending, In Progress, or Completed.
  - An Order can be cancelled by the Customer before it is assigned to a Servicer.
  - An Order can have a Feedback section where the Customer can rate and provide comments on the completed service.

- **Customer:**
  - A Customer can create and manage Orders for different services and view their Order History to see all the services they have requested in the past.
  - A Customer can view their Payment History to see all the payments they have made for completed services.
  - A Customer can have a profile that includes their name, contact information, and any preferences or notes related to their service requests.
  - A Customer can have a Service History that lists all the services they have received

from different Servicers over time.

- **Admin:**
  - An Admin can view and manage all the Orders, Services, Invoices, and Payments in the system.
  - An Admin can assign Servicers to Orders based on their availability and skill set.
  - An Admin can generate Invoices for completed Services and manage the Payment process.
  - An Admin can view system analytics and reports on Service quality, payment trends, and other metrics.
  - An Admin can adjust system settings, such as Service fees, Payment methods, and Service categories.
- **Invoice:**
  - An Invoice is generated for each completed Order and includes details about the service, the Servicer, and the total cost.
- **Payment:**
  - A Payment is associated with an Invoice and completes the transaction for a completed service.
  - A Payment can have a status that tracks its progress.
  - A Payment can be made using different Payment methods, such as cash on delivery and online payment.
  - A Payment can have a Transaction ID or Receipt Number that serves as a unique identifier for the payment.
  - A Payment can be processed automatically by the system.
- **Service History:**
  - A Service History lists all the Services completed by a particular Servicer over time.

- A Service History includes information about the Orders and Customers associated with each completed Service.

- A Service History also records the date and time of each completed Service.

5.4 Recommendation System

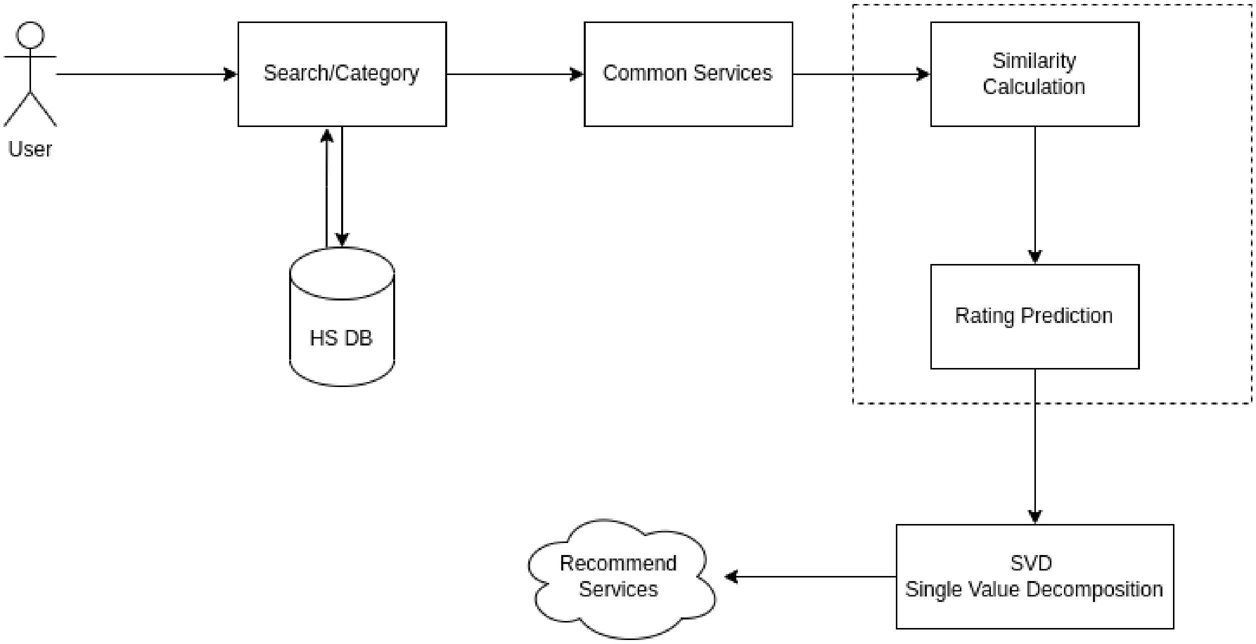


Figure 14: Working of Recommendation System

Data Collection:

In the development of a recommendation system, the first step involves the collection of data based on user behavior, such as ratings and purchases of services. This data is then stored in a database for future use in the recommendation system.

Data Pre-processing:

Data pre-processing is an important step in preparing data for use in a recommendation system. It involves cleaning the data to remove duplicates and inconsistencies, as well as converting the data into a suitable format. One common method of preparing data for use in a recommendation system is to convert user and item information into embedding matrices. These matrices represent the information in a way that can be easily processed and used in machine learning algorithms to generate recommendations.

$$A \approx UV^\top \text{ with } U = \left[\begin{array}{c} u_1 \\ \hline \vdots \\ \hline u_N \end{array}\right] \text{ and } V = \left[\begin{array}{c} v_1 \\ \hline \vdots \\ \hline v_M \end{array}\right].$$

where,
N is the number of users,
M is the number of services,
A_{ij} is the rating of the j th services by the i th user,
each row U_i is a d -dimensional vector (embedding) representing user i ,
each row V_j is a d -dimensional vector (embedding) representing movie j ,
the prediction of the model for the (i,j) pair is the dot product $<U_i, V_j>$.

The rating matrix could be very large and, in general, most of the entries remain unobserved, since a given user will only rate a small subset of services . For efficient representation, a sparseTensor is used from the TensorFlow library. A SparseTensor uses three tensors to represent the matrix: tf.SparseTensor(indices, values, dense_shape) represents a tensor, where a value $A_{ij} = a$ is encoded by setting $indices[k] = [i, j]$ and $values[k] = a$.

Model Selection:

The next step is to select a suitable model for the recommendation system. Collaborative filtering involves recommending items based on the behavior of similar users. This allows for serendipitous recommendations; that is, collaborative filtering models can recommend an item to user A based on the interests of a similar user B. Furthermore, the embeddings can be learned automatically, without relying on hand-engineering of features. But only the Matrix Factorization model might produce spurious results. The items with high norm value (popular items) might be recommended more frequently, which might be irrelevant to some users . Therefore , a regularization is also performed to make correct predictions and make better suggestions.

Evaluation:

The next step is to evaluate the performance of the selected model. The data is thus split into training and testing sets, and the accuracy of the model is measured using metrics such as Root Mean Squared Error (RMSE).

Deployment:

Therefore the recommendation system is then deployed in a working environment. This involves integrating the system into the e-commerce platform and ensuring that it is reliable, scalable and provides recommendations based upon the user behaviour.

5.5 Celery

Celery is a powerful tool that has been integrated into the Home Sewa app to enhance its functionality and efficiency. Celery is an asynchronous task queue that enables the app to perform tasks in the background, freeing up the main thread to handle other user requests. The tasks for that Celery has been used for in the Home Sewa app are:

- **Compress Image**: When a user uploads an image to the app, it is compressed using Celery to reduce its size and make it easier to store and load. This task is performed in the background, without interrupting the user's experience.

- **Matchmaking Task**: When a user requests a service, Celery is used to match them with a suitable service provider based on their location, availability, and expertise. This task is performed in real-time, ensuring that the user is matched with the best possible service provider.

- **Matchmaking Notification**: Once a suitable service provider has been found, Celery has been used to send a notification to the user and the service provider, informing them of the match. This task can be performed quickly and efficiently, ensuring that the user and the service provider can begin communicating and arranging the service.

- **Send Email**: Celery has been used to send emails to users and service providers, such as confirmation emails, matchmaking notification emails. This task is performed in the background, freeing up the main thread to handle other user requests

- **Send SMS**: Celery has been used to send SMS to users and service providers, such as OTP, matchmaking notification. This task is performed in the background, freeing up the main thread to handle other user requests

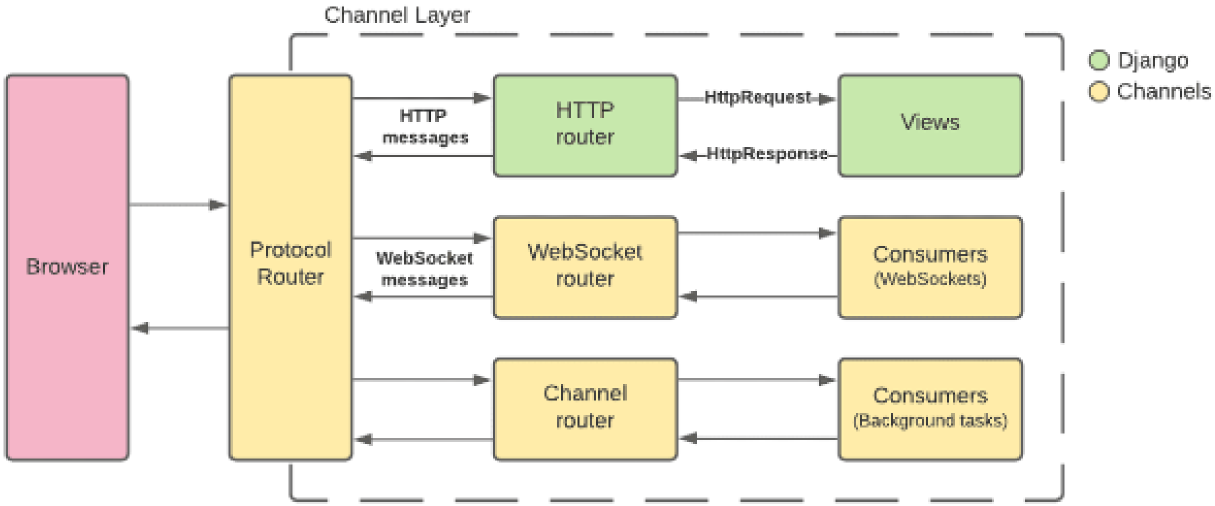


Figure 15: Working Principle of Chat and Notification Systems

Chat System:

In a traditional HTTP-based web application, the client sends a request to the server, and the server sends a response back to the client. This works well for many types of applications, but it's not well-suited for real-time communication, such as a chat system. Websocket is a protocol that enables two-way communication between a client and a server over a single, long-lived connection. This means that the server can send data to the client at any time, without waiting for the client to make a request. A chat system was implemented using websocket in Django, Django Channels is used. Django Channels provides a way to handle websocket connections using a concept called "consumers." Consumers are similar to views in Django, but they handle websocket connections instead of HTTP requests. When a client connects to the chat system, the server creates a new websocket connection and assigns it a unique ID. The client sends a message to the server, indicating that it wants to join a chat room or start a new chat. The server then creates a new group for the chat room or chat, and adds the websocket connection to the group. When a client sends a message to the chat room or chat, the server sends the message to all the other clients in the group. The other clients receive the message and display it in their chat window. If a client disconnects from the chat system, the server removes the websocket connection from the group. This ensures that messages are only sent to active clients.

Notification System

A notification system is a way for a web application to inform users of new events or updates, such as new messages, new orders, etc. Traditionally, notification systems have relied on periodic polling, where the client sends requests to the server at regular intervals to check for new updates. This approach can be inefficient, as it generates a lot of unnecessary traffic and can cause delays in delivering updates. Using websocket, a real-time notification system is built that delivers updates to clients as soon as they occur, without the need for periodic polling. A notification system was implemented using Django and websocket, we used Django Channels. A consumer create the handles for the websocket connections, and use it to send real-time updates to clients. When a client connects to our notification system, the server creates a new websocket connection and assigns it a unique ID. The client sends a message to the server, indicating that it wants to subscribe to certain types of notifications, such as new messages or new followers. When an event occurs that triggers a new notification, the server sends the notification to all the clients that have subscribed to that type of notification. The clients receive the notification and display it to the user. If a client disconnects from the notification system, the server removes the WebSocket connection from the list of active clients. This ensures that notifications are only sent to active clients.

6. Results and Discussion

For matchmaking between servicers and customers in Home Sewa, we initially used a weight-based rating system. However, it was discovered that this system had several significant disadvantages that were impacting the quality of matches and the experience of our users. One of the main drawbacks of the weight-based system was its complexity. The system required many variables to be taken into account, such as the servicer's availability, the location of the Customer, the servicer's skills, servicer feedback and many more. This complexity made it difficult for the accurate matchmaking.Another issue with the weight-based system was its instability. Ratings fluctuated dramatically after each match, leading to unpredictable matchmaking and unbalanced matches. This instability was frustrating and difficult to track progress over time. The weight-based system was subjective, relying on assessments of a servicer's skill level that could vary widely depending on the individual making the assessment. This led to biased matchmaking and unfair matches.Due to significant drawbacks, a decision was made to switch to an Elo-based rating system for matchmaking between servicers and customers in HomeSewa. The Elo system was simpler than the weight-based system, with a straightforward formula that was easy to understand and implement too. This made it more accessible to our customers. The ELO system was highly accurate in predicting the outcome of matches, taking into account the skill levels of both servicers and Customers and adjusting their ratings accordingly. This ensured that matches were fair and competitive.The ELO system was stable over time, with ratings that were less prone to dramatic fluctuations. This made it easier to track progress and maintain a consistent and reliable matchmaking system.

For the recommendation, initially a Collaborative Filtering Matrix Factorization (CFMF) model was built which ran for 1000 iterations. It came up with train error of 0.126281 and test error of 6.252879. The hyper parameters of the model was embedding dimensions , initial standard deviation and learning rate which was chosen to be 35, 0.4 and 0.01 respectively. The Figure 16 shows the train-test plot. The graph shows high error in the test phase since the data to feed the model was relatively small. While the training phase seemed to have less error.

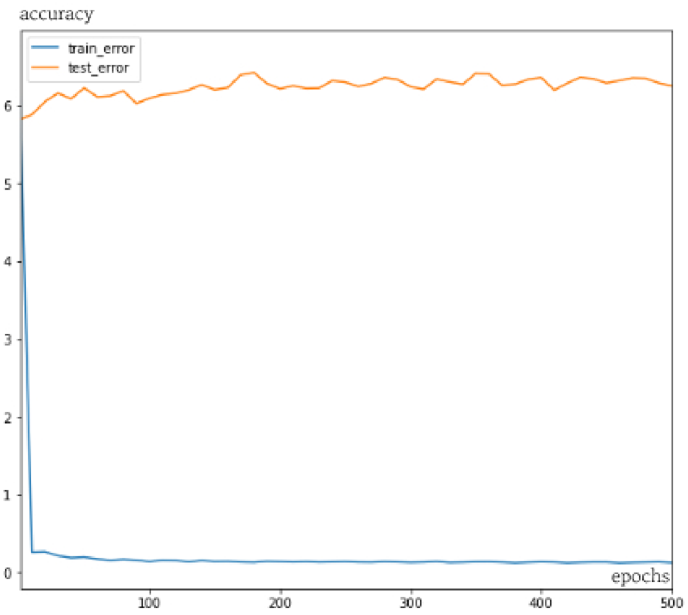


Figure 16: Train-Test Plot (Collaborative Filtering with Regularization)

7. Conclusion

Home Sewa is a promising application that can address the problems of delay in services, scams, and untrustworthy service providers in Nepal. The application's personalized recommendation system along with ELO ranking algorithm will help users to connect with appropriate and reliable service providers quickly. With the increasing reliance on technology and IT, such innovative solutions can significantly improve the quality of life and streamline service delivery. Overall, Home Sewa is a step in the right direction towards creating a reliable and efficient service delivery system in Nepal.

8. Future Enhancement

1. In-app Communication: To enhance the user experience and ensure that service requests are properly fulfilled, Home Sewa could integrate an in-app communication system that allows users to directly communicate with the service provider. This could help users clarify their needs and ensure that the service provider fully understands the

2. Quality Assurance: Home Sewa could implement a quality assurance system that monitors the performance of service providers and ensures that they are meeting the agreed-upon service standards. This could help build trust with users and ensure that they receive high-quality services every time.

3. Optimizing Home Sewa's recommendation system with AI techniques, such as content based filtering using matrix factorization or deep learning, could improve the accuracy and relevance of the platform's suggestions. This could increase user satisfaction and loyalty, attract new users, and help Home Sewa stay competitive in a market where AI-powered recommendation systems are becoming more common.

References

[1] Techinasia Urban company. Creating home services market india. https://www.techinasia.com/urban-company-creating-home-services-market-india. [Online; accessed 10-January-2023].

[2] Handy. About us. https://www.handy.com/about-us. [Online; accessed 20-January-2023].

[3] Task Rabbit. About us. https://www.taskrabbit.com/about. [Online; accessed 21-January-2023].

[4] Sajilo Sewa. About us. https://www.sajilosewa.com/aboutUs. [Online; accessed 21-January-2023].

[5] Elo A. E. The rating of chessplayers, past and present. 1978.

[6] Z. Zhu S. Zhang, R. Zhao and W. Wang. Weight-based collaborative filtering recommendation method for education resource. *IEEE Access*, 2019.

[7] D. H. Park D. Kim and C. Lee. A weight-based matchmaking system for the healthcare industry. in proceedings of the 2020 international conference on big data and smart computing (bigcomp). *IEEE Access*, pages 410–415, 2020.

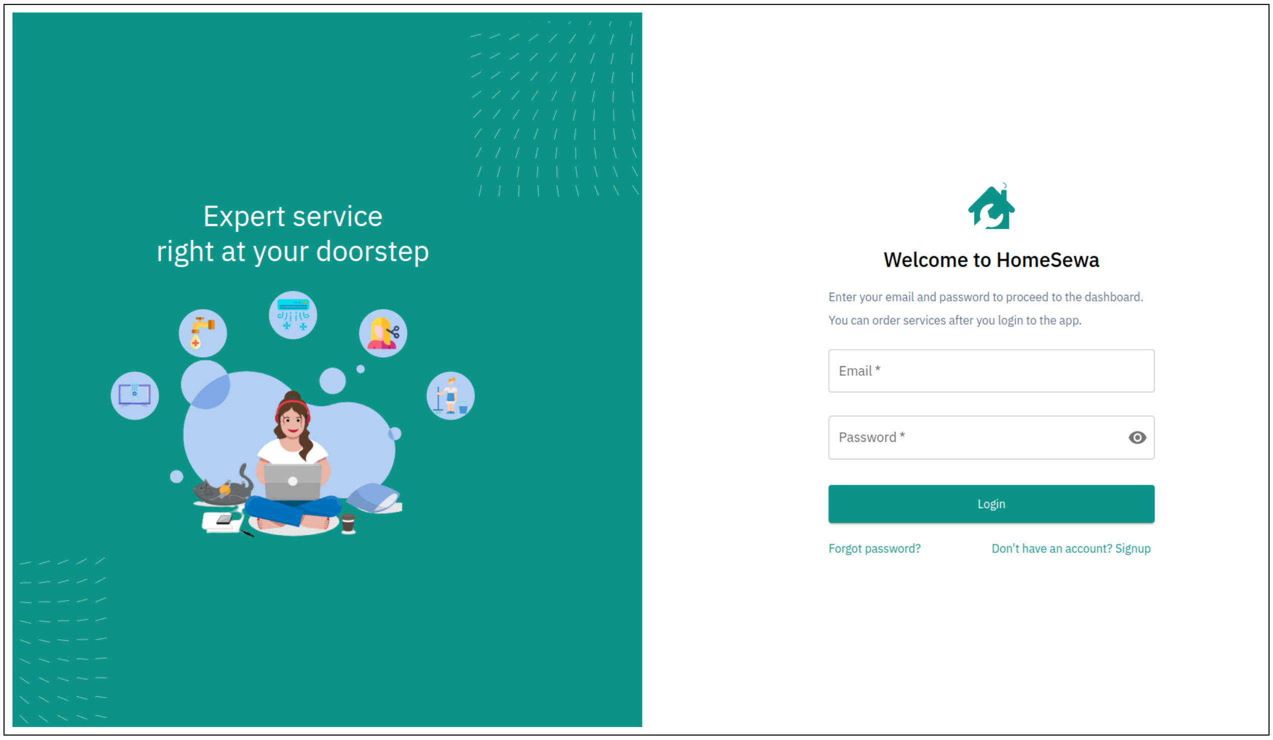
[8] Smith B. York J Linden, G. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, pages 76–80, 2003.

[9] Netflix prize. https://www.netflixprize.com/.

[10] J. Smith. Celery: An asynchronous task queue for python. journal of web development. 15(2):45–53, 2022.

Appendix

Appendix A

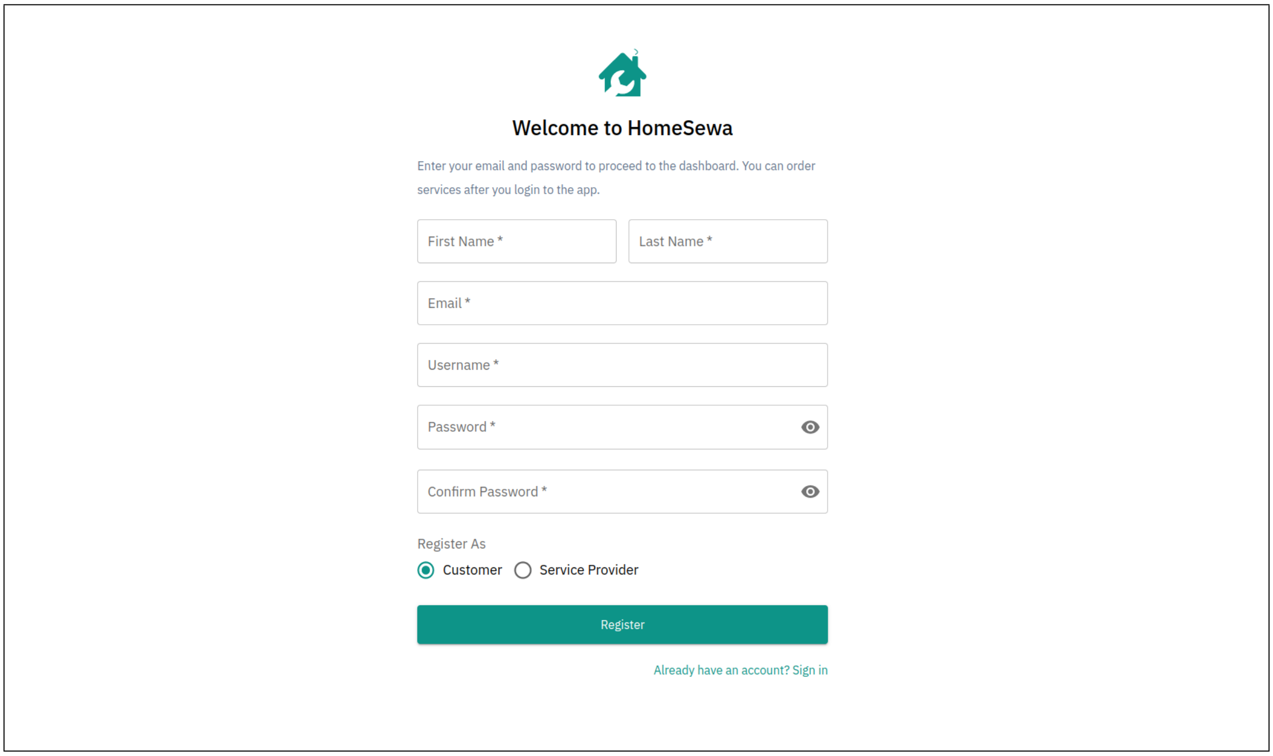


Expert service
right at your doorstep

Welcome to HomeSewa

Enter your email and password to proceed to the dashboard.
You can order services after you login to the app.

Figure A1: Login Screen



Welcome to HomeSewa

Enter your email and password to proceed to the dashboard. You can order services after you login to the app.

First Name * Last Name *

Email *

Username *

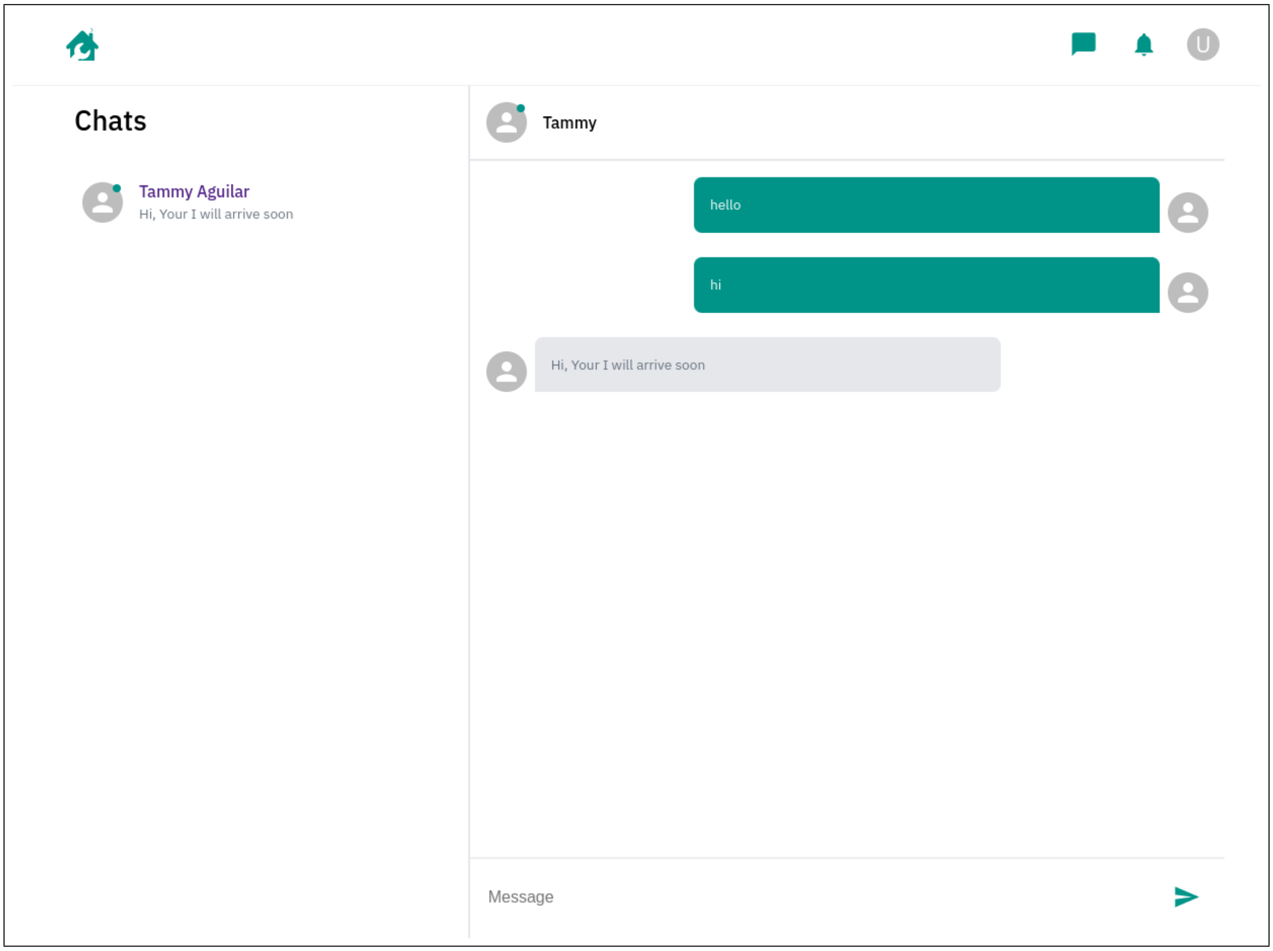
Password *

Confirm Password *

Register As

● Customer ○ Service Provider

Figure A2: Register Screen



Tammy Aguilar
Hi, Your I will arrive soon

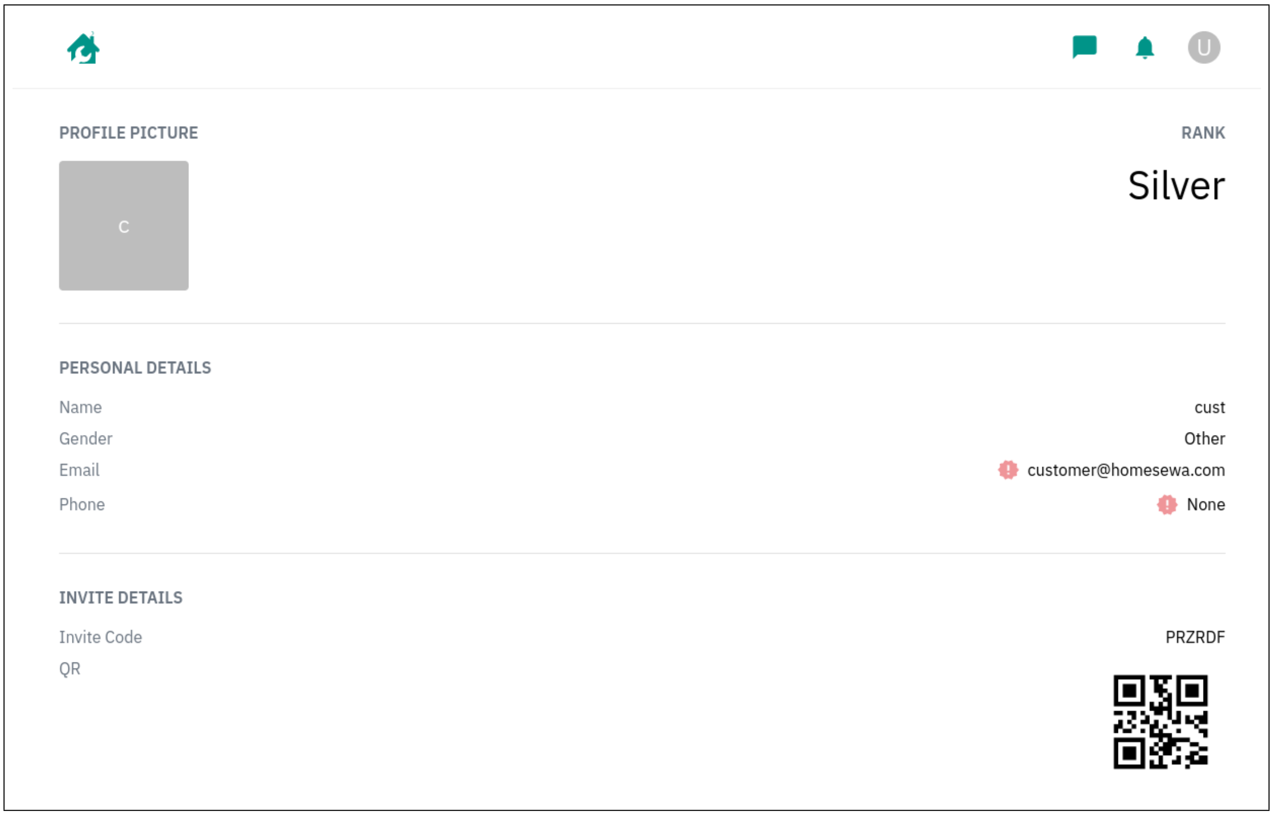
Tammy

hello

hi

Hi, Your I will arrive soon

Figure A3: Chat Screen



PROFILE PICTURE

RANK

Silver

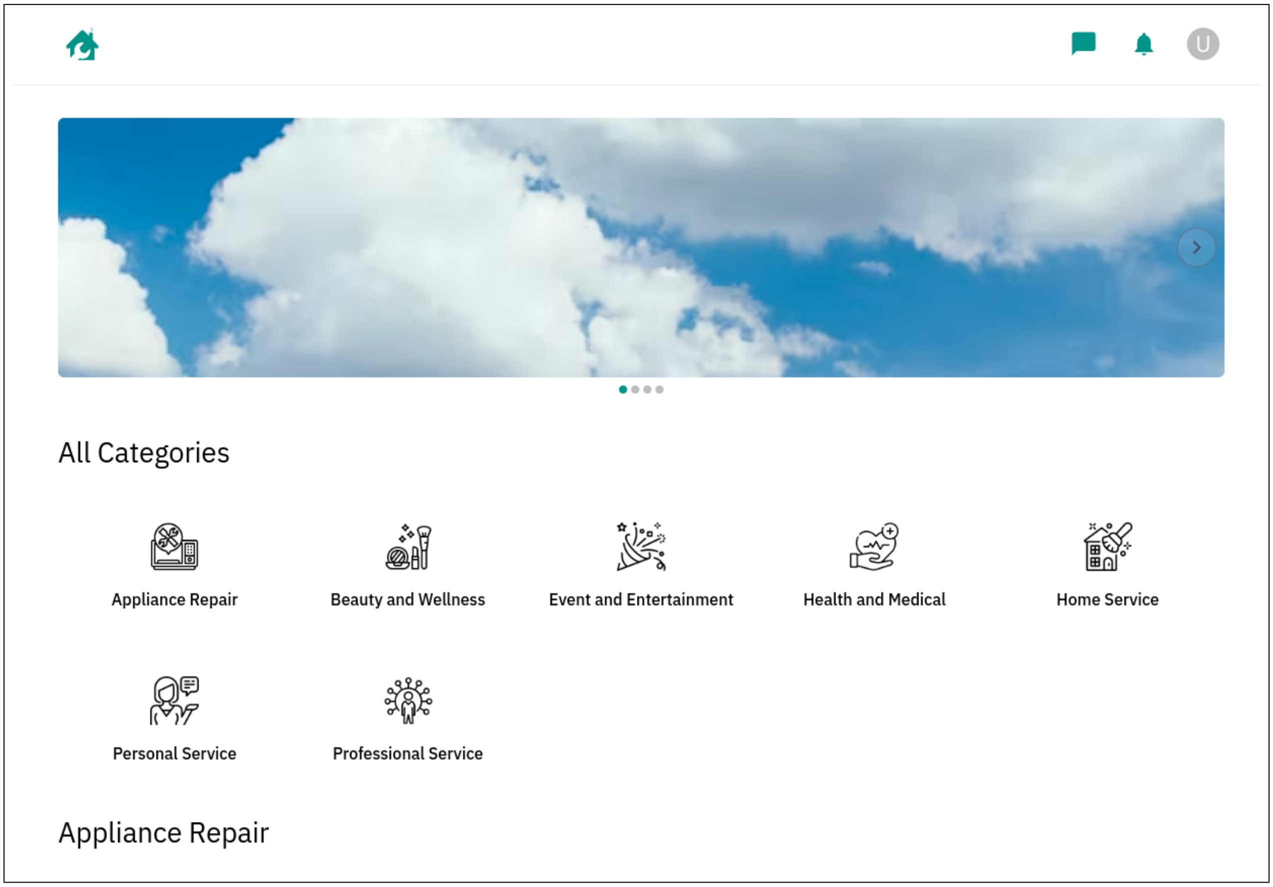
PERSONAL DETAILS

Name	cust
Gender	Other
Email	customer@homesewa.com
Phone	None

INVITE DETAILS

Invite Code	PRZRDF
QR	

Figure A4: User Profile



All Categories

Appliance Repair

Beauty and Wellness

Event and Entertainment

Health and Medical

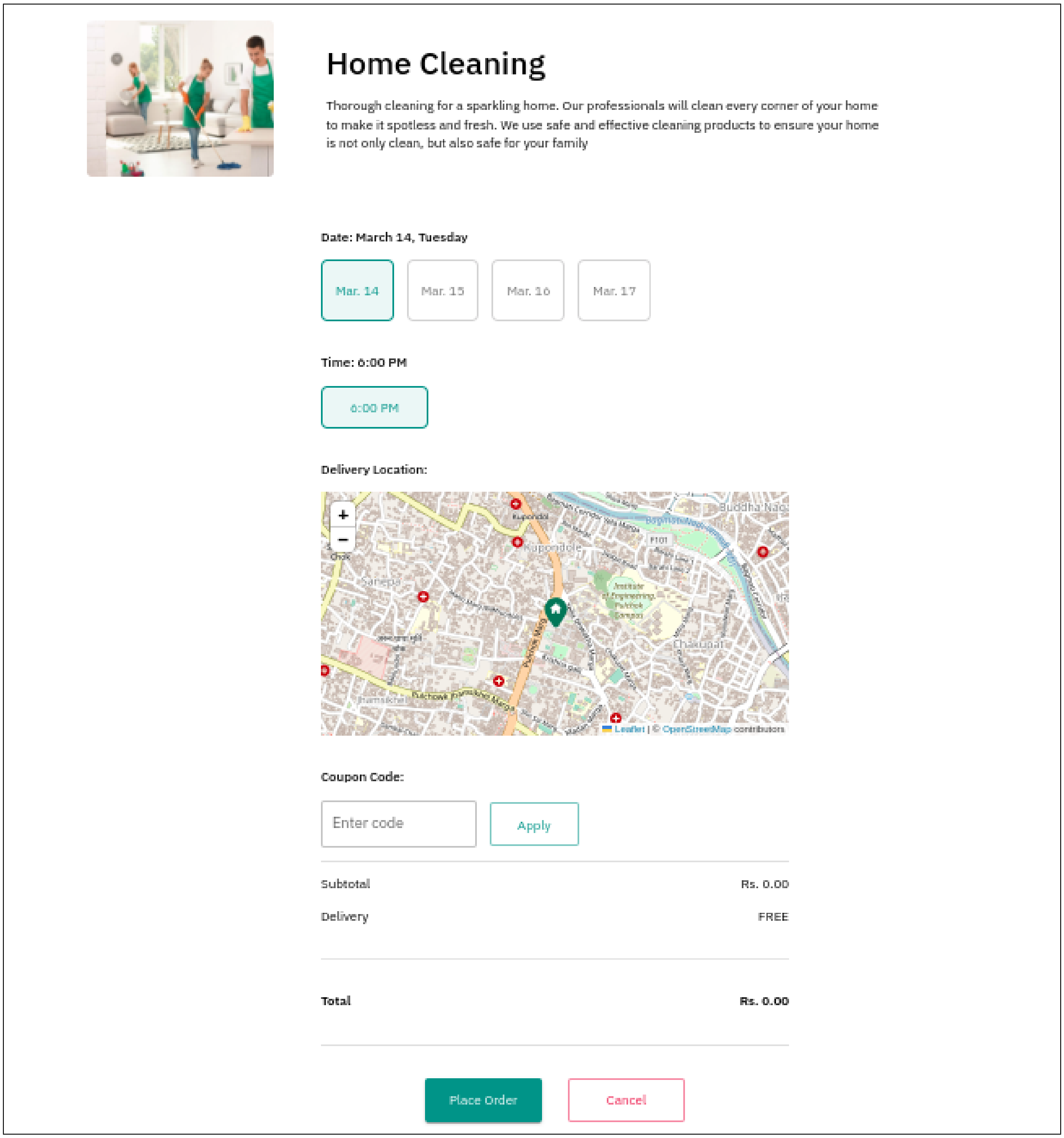
Home Service

Personal Service

Professional Service

Appliance Repair

Figure A5: Customer Dashboard



Home Cleaning

Thorough cleaning for a sparkling home. Our professionals will clean every corner of your home to make it spotless and fresh. We use safe and effective cleaning products to ensure your home is not only clean, but also safe for your family

Date: March 14, Tuesday

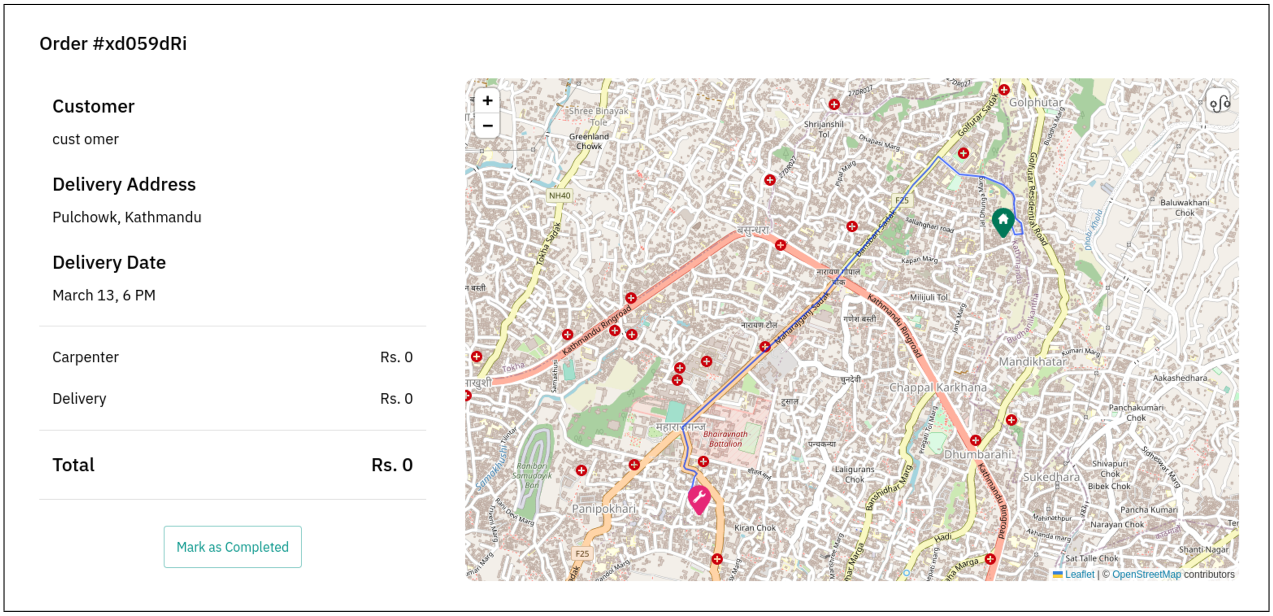
Time: 6:00 PM

Delivery Location:

Coupon Code:

Subtotal	Rs. 0.00
Delivery	FREE
Total	**Rs. 0.00**

Figure A6: Order Form



Order #xd059dRi

Customer

cust omer

Delivery Address

Pulchowk, Kathmandu

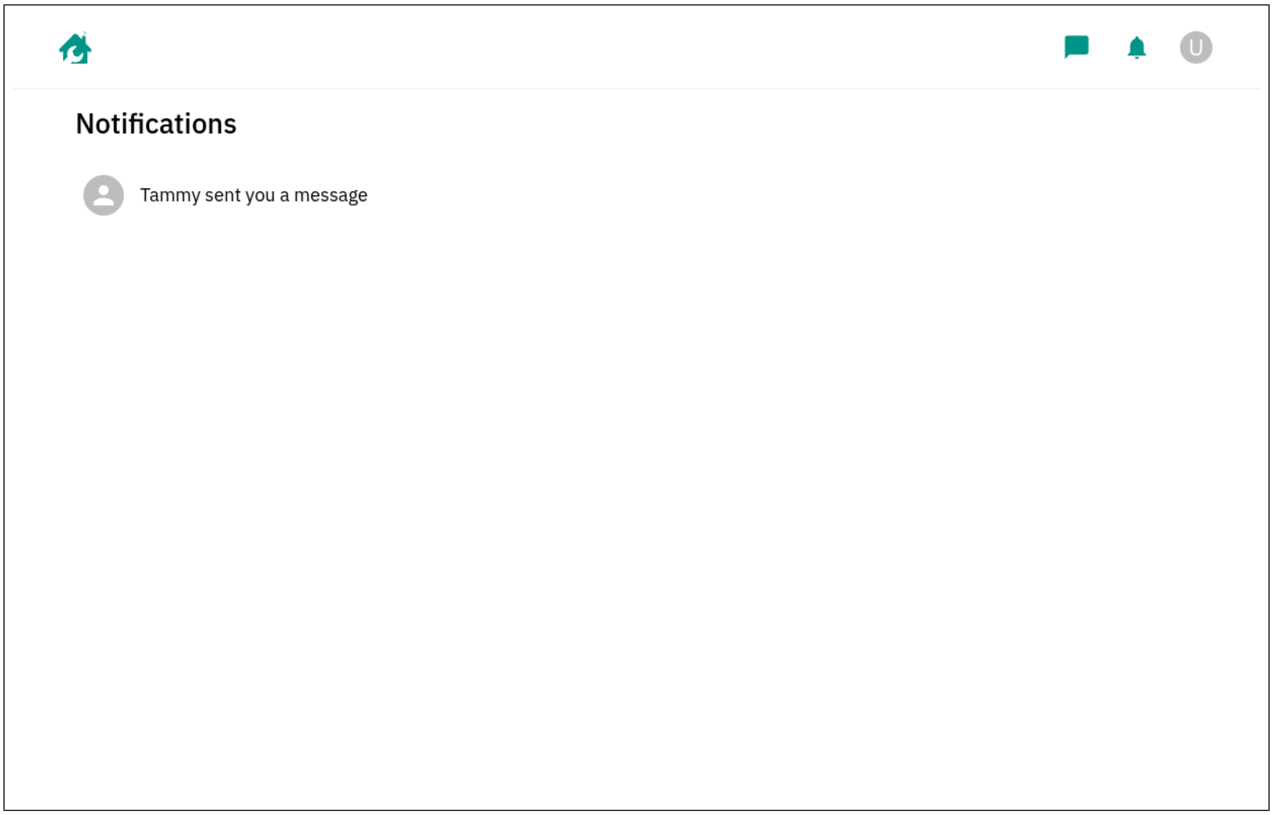
Delivery Date

March 13, 6 PM

Carpenter	Rs. 0
Delivery	Rs. 0
Total	**Rs. 0**

Mark as Completed

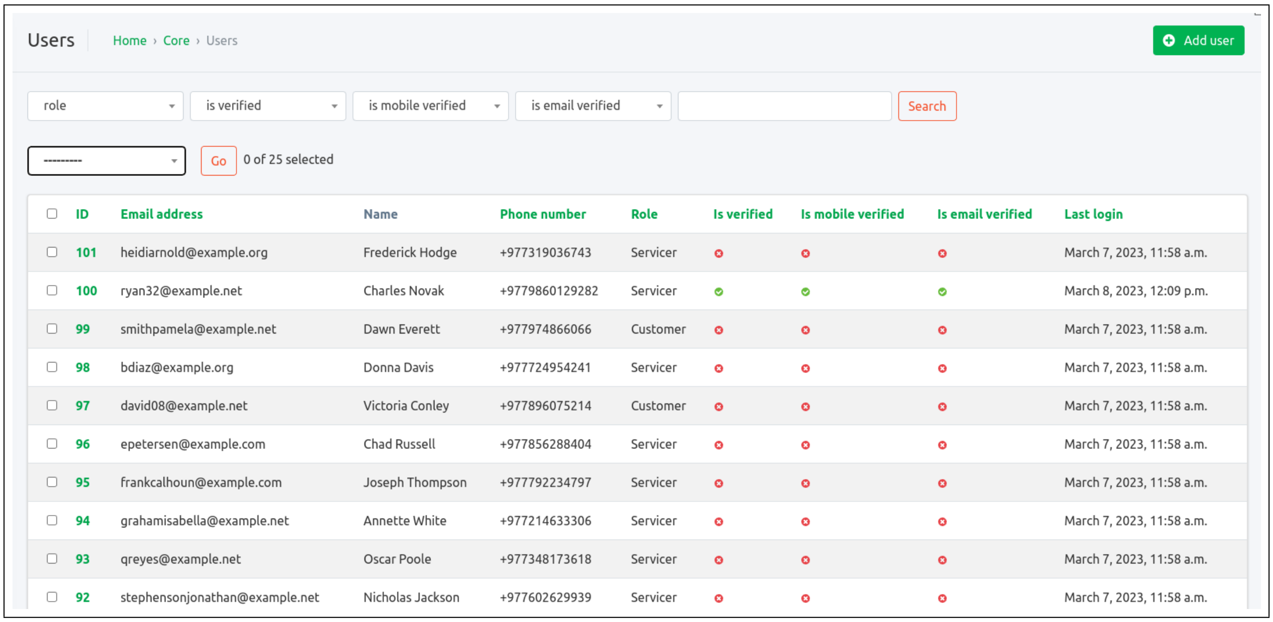
Figure A7: Booking Details



Notifications

Tammy sent you a message

Figure A8: Notification Screen



0 of 25 selected

□	ID	Email address	Name	Phone number	Role	Is verified	Is mobile verified	Is email verified	Last login
□	101	heidiarnold@example.org	Frederick Hodge	+977319036743	Servicer	✗	✗	✗	March 7, 2023, 11:58 a.m.
□	100	ryan32@example.net	Charles Novak	+9779860129282	Servicer	✓	✓	✓	March 8, 2023, 12:09 p.m.
□	99	smithpamela@example.net	Dawn Everett	+977974866066	Customer	✗	✗	✗	March 7, 2023, 11:58 a.m.
□	98	bdiaz@example.org	Donna Davis	+977724954241	Servicer	✗	✗	✗	March 7, 2023, 11:58 a.m.
□	97	david08@example.net	Victoria Conley	+977896075214	Customer	✗	✗	✗	March 7, 2023, 11:58 a.m.
□	96	epetersen@example.com	Chad Russell	+977856288404	Servicer	✗	✗	✗	March 7, 2023, 11:58 a.m.
□	95	frankcalhoun@example.com	Joseph Thompson	+977792234797	Servicer	✗	✗	✗	March 7, 2023, 11:58 a.m.
□	94	grahamisabella@example.net	Annette White	+977214633306	Servicer	✗	✗	✗	March 7, 2023, 11:58 a.m.
□	93	qreyes@example.net	Oscar Poole	+977348173618	Servicer	✗	✗	✗	March 7, 2023, 11:58 a.m.
□	92	stephensonjonathan@example.net	Nicholas Jackson	+977602629939	Servicer	✗	✗	✗	March 7, 2023, 11:58 a.m.

Figure A9: User List Screen