

1. INTRODUCTION

When someone need aid with small but major household tasks, the trouble arises when service skilled persons are unavailable or the trusted providers are impossible to find, who delivers consistently flawless service on instance.

Our online system for household services provides the most expedient and annoys free way to get your domestic work done. We aim to help in providing optimal solutions to all your household troubles with more efficiency, ease and majorly, a delicate touch. A single click system describes booking highly skilled in-house professionals and gets your service done on time.

Customers' overall willingness to pay is significantly and positively correlated with the expectation that fee-based services would be better, and with the belief that "pay for what you get" is the right thing to do. Keeping that in sense our proposed system is basically a marketplace for household services and it is the platform where the rates were standardized and there is no necessitate haggling over prices. Several aspects like plumbing, electrical works and carpentry services are involved in a system to provide happy and healthy home atmosphere in order to satisfy consumers

2. PRODUCT OVERVIEW AND SUMMARY

2.1 PURPOSE

Our project, “Konnect”, is a web-based online application which aims to connect users with the vendors for the daily household services. It includes bidding process. In our project user can browse the services he wants to use according to it he can also share his budget. From the vendor side one or more vendors will give their quotation to the user so the bidding process will show the sorted list of quotation to the customer so finally customer will select the vendor for the services he wants to use.

2.2. SCOPE

“Konnect” aims to deliver a web-based application that will connect the local service provider like electricians, carpenters, painter, laundrymen etc. with local customers. The main aim of the project is to provider an easy-to-use application for services provided for customers.

We often get frustrated while taking the appointment of service provider because there the many problems are occur, like the service provider is busy art somewhere else or his not receiving our call or his cost is very high according to problem. So in this project we will remove this headache. We intend to introduce dynamic pricing mechanism for this web application based on reverse auctioning to come to the most economic price of the services. This web application will help connect the local service providers with customers and boost up local economy. So, from the home customer can select the desired service he wants to use from the budget friendly vendors.

2.3. OVERVIEW

A. TECHNOLOGIES USED

i. FRONT END

- React Js
- Axios
- Material UI, Bootstrap

ii. BACK END

- Spring Boot
- SMTP

iii. DATABASE MANAGEMENT SYSTEM

- MySQL

B. FEATURES PROVIDED

i. FOR CUSTOMERS

- a. **Register**- Customer can register himself for the web application
- b. **Login**- Customer can login into web application by using registered email and password
- c. **Service**-Customer can select the desired service he wants to use
- d. **Budget**-Customer can write his budget for the service
- e. **Description**-Customer can enter the description of the work.
- f. **Lockout time**-Customer can enter the lockout time for the bidding process.
- g. **Place order**-Upon entering all the parameters about the services he want to use customer can place the order.
- h. **Start order**-Customer can give the order to the desired vendor
- i. **Cancel order**-Customer also can cancel the order

- j. **Active order**-Customer can see the active orders.
- k. **Check bids**-Customer can check the bids which vendors entered
- l. **Pending order**-Customer can see the pending orders.
- m. **Completed order**-Customer can see the Completed orders.
- n. **Rating**-Customer can give the rating to vendors.
- o. **Logout**-Customer can end the session.

ii. FOR VENDORS

- a. **Register**- Vendor can register himself or the use of web application
- b. **Login**- Vendor can login into web application by using registered email and password
- c. **Place bid**-Vendors can enter his budget for the services customer wants to use.
- d. **I'm out**-Vendor can also opt out from the bidding process
- e. **Pending order**-Vendor can see the pending orders.
- f. **Completed order**-Vendor can see the Completed orders.
- g. **Logout**-Vendor can end the session.

2.4. FEASIBILITY STUDY

Feasibility is the determination of whether a project is worth undertaking or not. Before actually recommending the new system, it is important to investigate if it is feasible to develop it. Before developing and implementing a system, we have to make sure that the system is feasible in the following ways:

A. TECHNICAL FEASIBILITY

In this type of feasibility study, we are going to check whether it is possible or not to develop the requested system with the available manpower, software, hardware, etc.

This project makes use of cross-platform software's and solutions like Java, and hence can run on any operating system. React, used in front-end, is swift and light weight framework when it comes to delivering the requested page as it doesn't reload the entire page for every HTTP request. It only re-renders the components that need to fetch new data. Also, as React is modular in nature, it is easy to develop new components and scale up existing components in order to add new features to the system.

The combination of Spring Boot for backend make for a fast, easy to set-up and reliable system to interact with the database, as they are secure and transactional in nature. Since the sensitive data of customers and vendors need to be stored in a robust and secure database, MySQL database management system was chosen as it is an industry standard.

B. OPERATIONAL FEASIBILITY

In this type of feasibility study, the operation of the system is considered. An analysis is performed on whether it is feasible for the user department to use the application. Thus, the proposed system is said to be operationally feasible only if clients are able to understand the system clearly and correctly, and can use it with ease. In the design of this project, we always kept user experience in mind. We made an effort to have a good user interface with consistent theme and alluring design to keep the users interested and engaged. In our project, the use of universally known icons and instructions that are easy to understand makes sure

that the user will not need any special technical know-how to use the application. We made sure that the information available throughout the application is arranged in a logically coherent and consistent manner, guaranteeing that the users will have a smooth and effortless experience and even enjoy using the application.

C. ECONOMIC FEASIBILITY

In this type of feasibility study, the benefits of the system to the organization are considered by taking into consideration the cost-benefit analysis. All the software and technologies used in our project free, open-source, and widely available, with each of the technologies having an extensive community support. This makes “Konnect” an economically feasible solution to the organizations that wish to implement it.

3. REQUIREMENTS FULFILLED

3.1 FUNCTIONAL REQUIREMENTS

Following are the functional requirements fulfilled by our project:

- Customers can place orders for require services with expected budget and lockout time.
- Based on their work to be done and quality of work feedbacks for vendor existing ones.
- Customers can see the how many bids are active for their order.
- Customers can see how many orders completed, pending and in active.
- Vendors can see which work is require and how much time for bid.
- Vendors can enter in the bid with budget and comments for work.
- Vendors are accessible for exit from the current bid.
- Vendors and customers are able to get bill by email with all detail.

3.2 NON-FUNCTIONAL REQUIREMENTS

Following are the non-functional requirements fulfilled by our project:

- Since the application uses lightweight and established software components that are also cross-platform, it is remarkably performs and has good support for every operating system.
- The use of React for front end and Spring Boot, Spring Data JPA and Hibernate for back end delivers quick response times to vendors and customers alike.
- application provides a consistent theme and user-friendly interface that anyone can grasp easily, even without a technical

4. PROJECT DESIGN

4.1 DATA MODEL

4.1.1 Tables Related to customers

1. Customers Table

	Field	Type	Null	Key	Default	Extra
▶	id	bigint	NO	PRI	NULL	auto_increment
	address	varchar(255)	YES		NULL	
	city	varchar(255)	YES		NULL	
	contact_no	varchar(255)	YES	UNI	NULL	
	email	varchar(255)	YES	UNI	NULL	
	name	varchar(255)	YES		NULL	
	password	varchar(255)	YES		NULL	
	pincode	varchar(255)	YES		NULL	
	state	varchar(255)	YES		NULL	

4.1.2 Tables Related to Orders

1. Orders Table

	Field	Type	Null	Key	Default	Extra
▶	id	bigint	NO	PRI	NULL	auto_increment
	budget	double	YES		NULL	
	customer_comments	varchar(255)	YES		NULL	
	final_amount	double	YES		NULL	
	lockout_time_in_minutes	int	NO		NULL	
	order_finalized_time	datetime(6)	YES		NULL	
	order_placed_time	datetime(6)	YES		NULL	
	rating	double	YES		NULL	
	temp_table_name	varchar(255)	YES	UNI	NULL	
	vendor_comments	varchar(255)	YES		NULL	
	customer_id	bigint	YES	MUL	NULL	
	final_vendor_id	bigint	YES	MUL	NULL	
	order_status_id	bigint	YES	MUL	NULL	
	service_id	bigint	YES	MUL	NULL	

2. Orders Status

	Field	Type	Null	Key	Default	Extra
▶	id	bigint	NO	PRI	NULL	auto_increment
	order_status_type	varchar(255)	YES		NULL	

4.1.3 Tables Related to Vendors

1. Vendors Table

	Field	Type	Null	Key	Default	Extra
▶	id	bigint	NO	PRI	NULL	auto_increment
	address	varchar(255)	YES		NULL	
	city	varchar(255)	YES		NULL	
	contact_no	varchar(255)	YES	UNI	NULL	
	email	varchar(255)	YES	UNI	NULL	
	name	varchar(255)	YES		NULL	
	password	varchar(255)	YES		NULL	
	pincode	varchar(255)	YES		NULL	
	rating	double	YES		NULL	
	state	varchar(255)	YES		NULL	

2. Vendors Service Table

	Field	Type	Null	Key	Default	Extra
▶	vendor_id	bigint	NO	PRI	NULL	
	services_id	bigint	NO	PRI	NULL	

3. Service Table

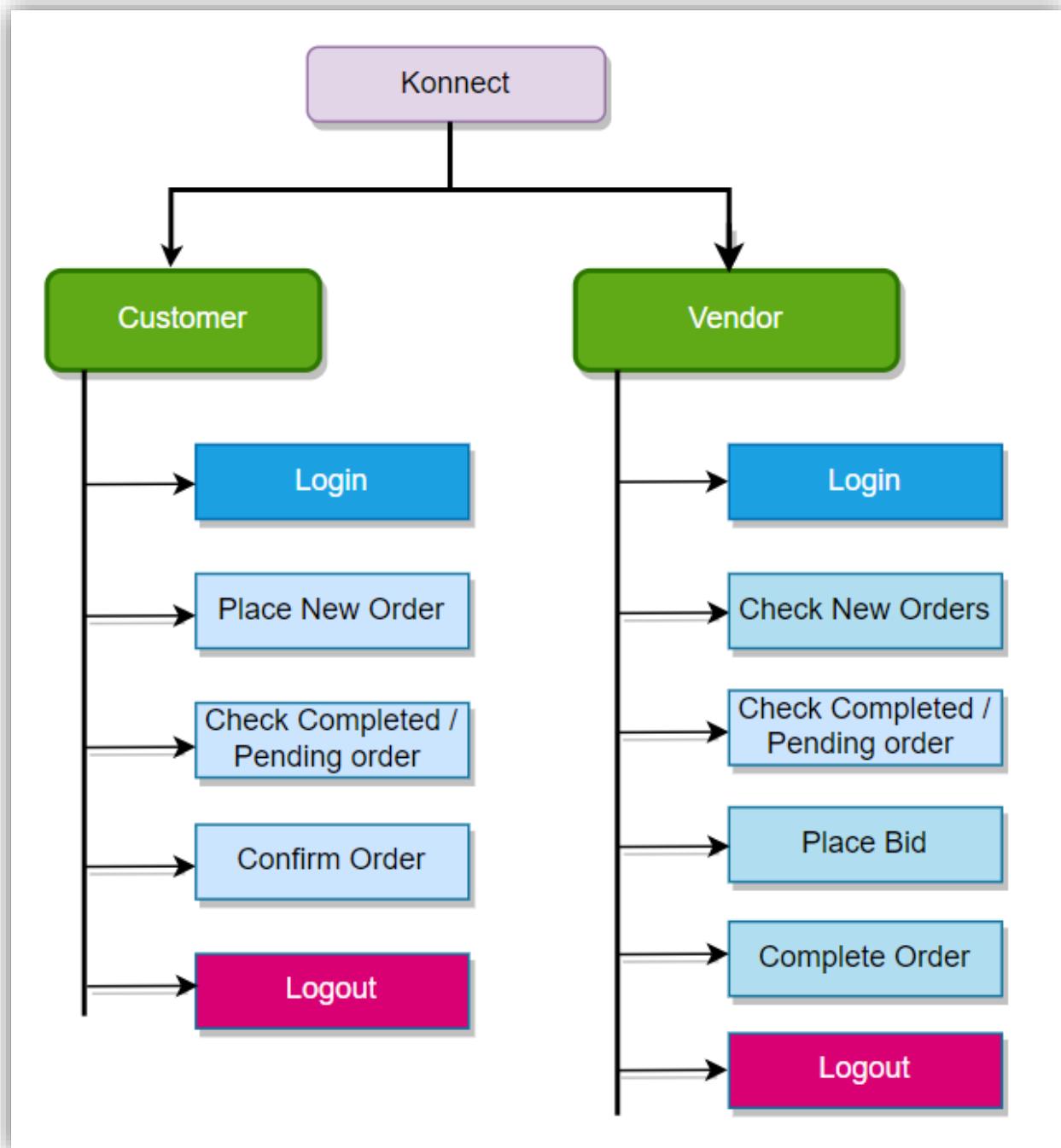
	Field	Type	Null	Key	Default	Extra
▶	id	bigint	NO	PRI	NULL	auto_increment
	service_type	varchar(255)	YES		NULL	

4.1.4 Tables Related to Bidding

1. Bid Table

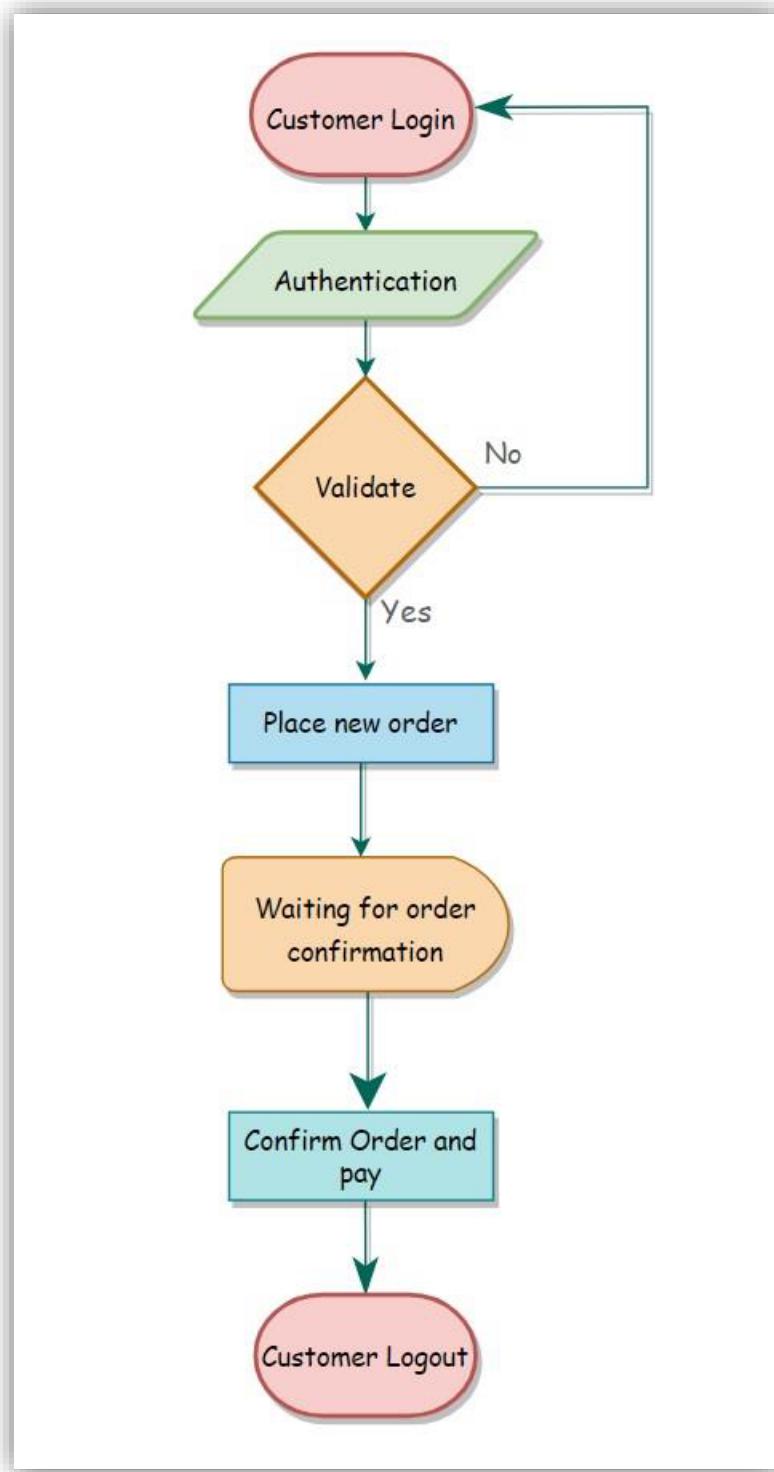
	Field	Type	Null	Key	Default	Extra
▶	id	bigint	NO	PRI	NULL	auto_increment
	bid_amount	double	YES		NULL	
	bid_placed_time	datetime(6)	YES		NULL	
	vendor_comments	varchar(255)	YES		NULL	
	order_id	bigint	YES	MUL	NULL	
	vendor_id	bigint	YES	MUL	NULL	

4.2 FUNCTIONAL DECOMPOSITION DIAGRAM

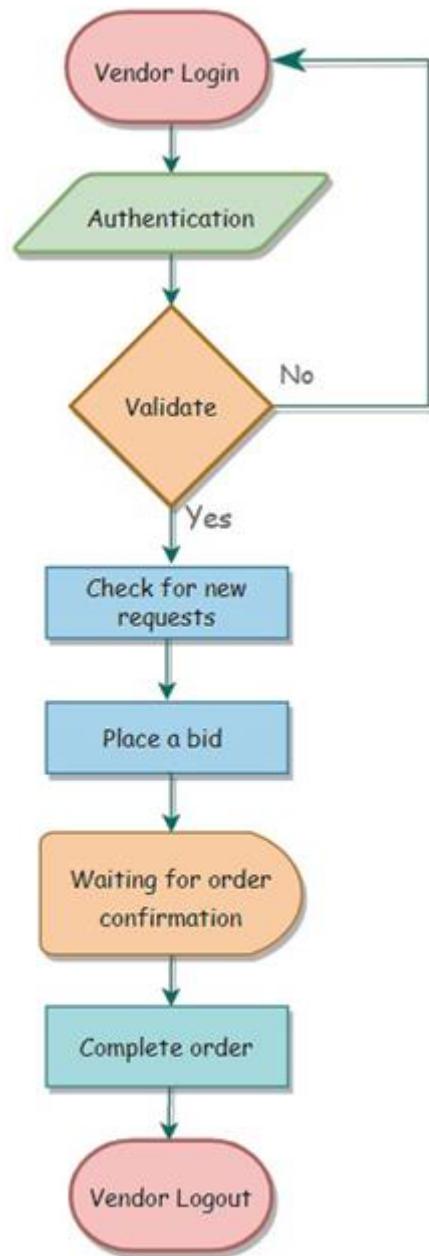


4.3 ACTIVITY DIAGRAMS

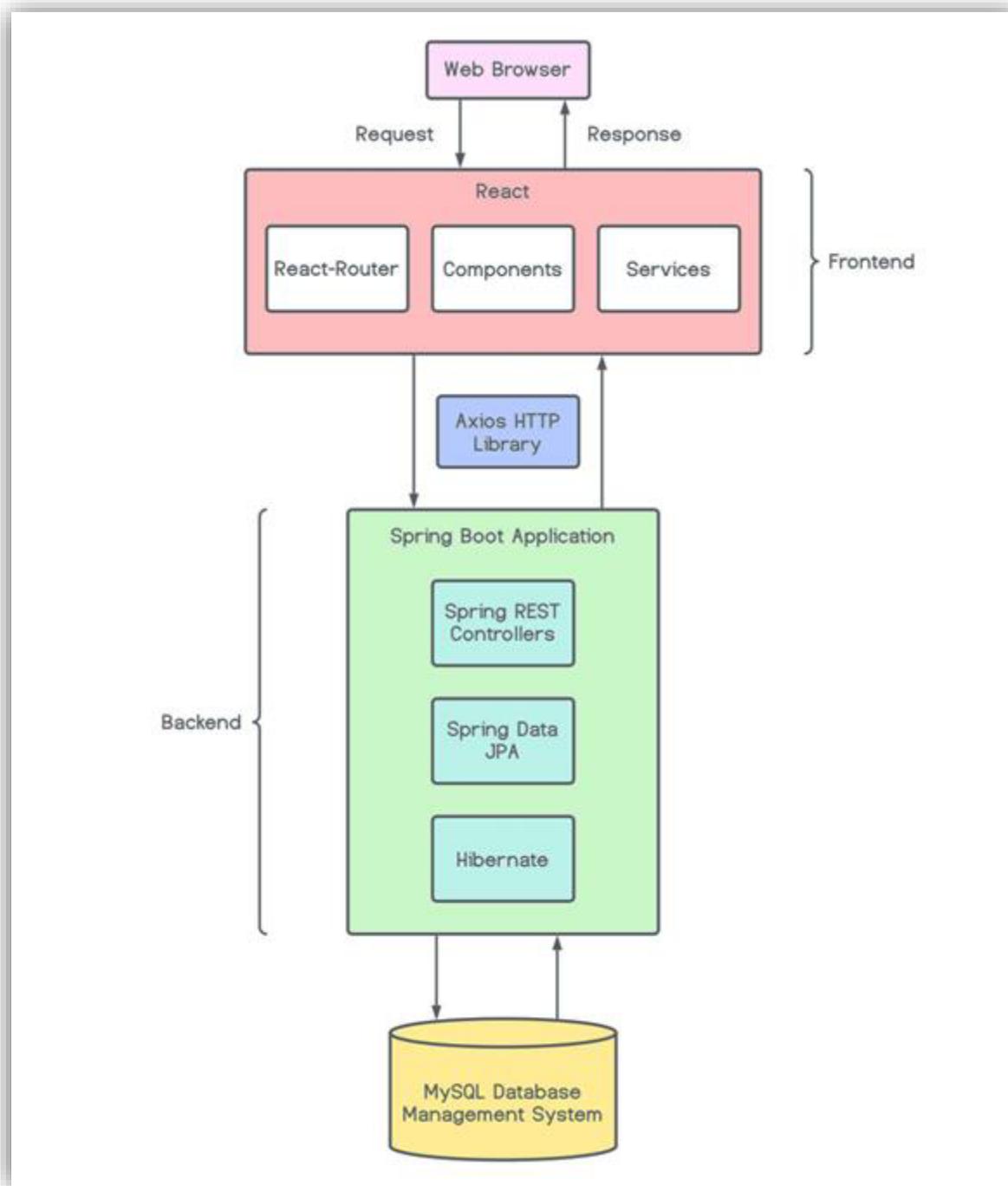
1. Customer Activity Diagram.



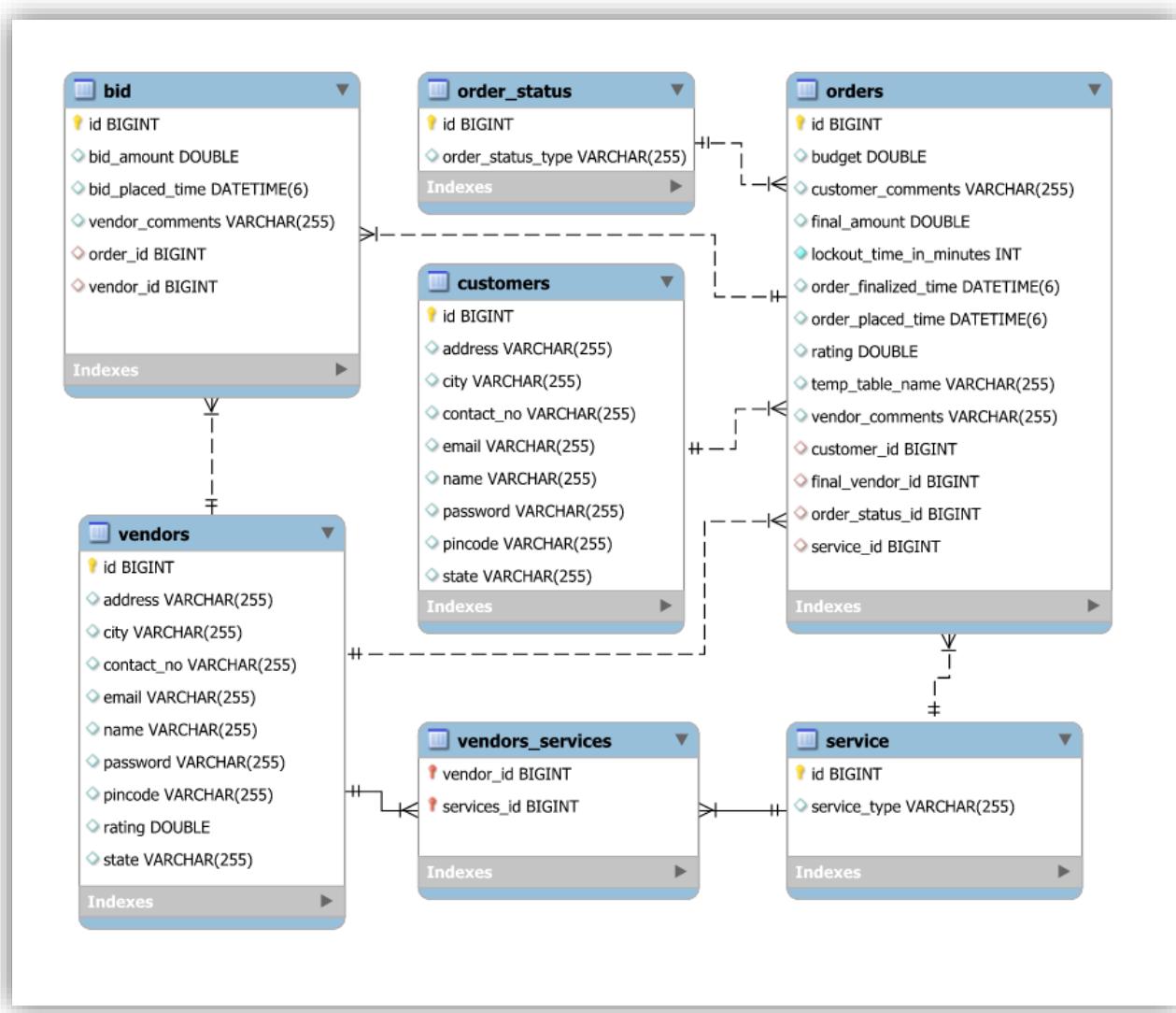
2. Vendor Activity Diagram



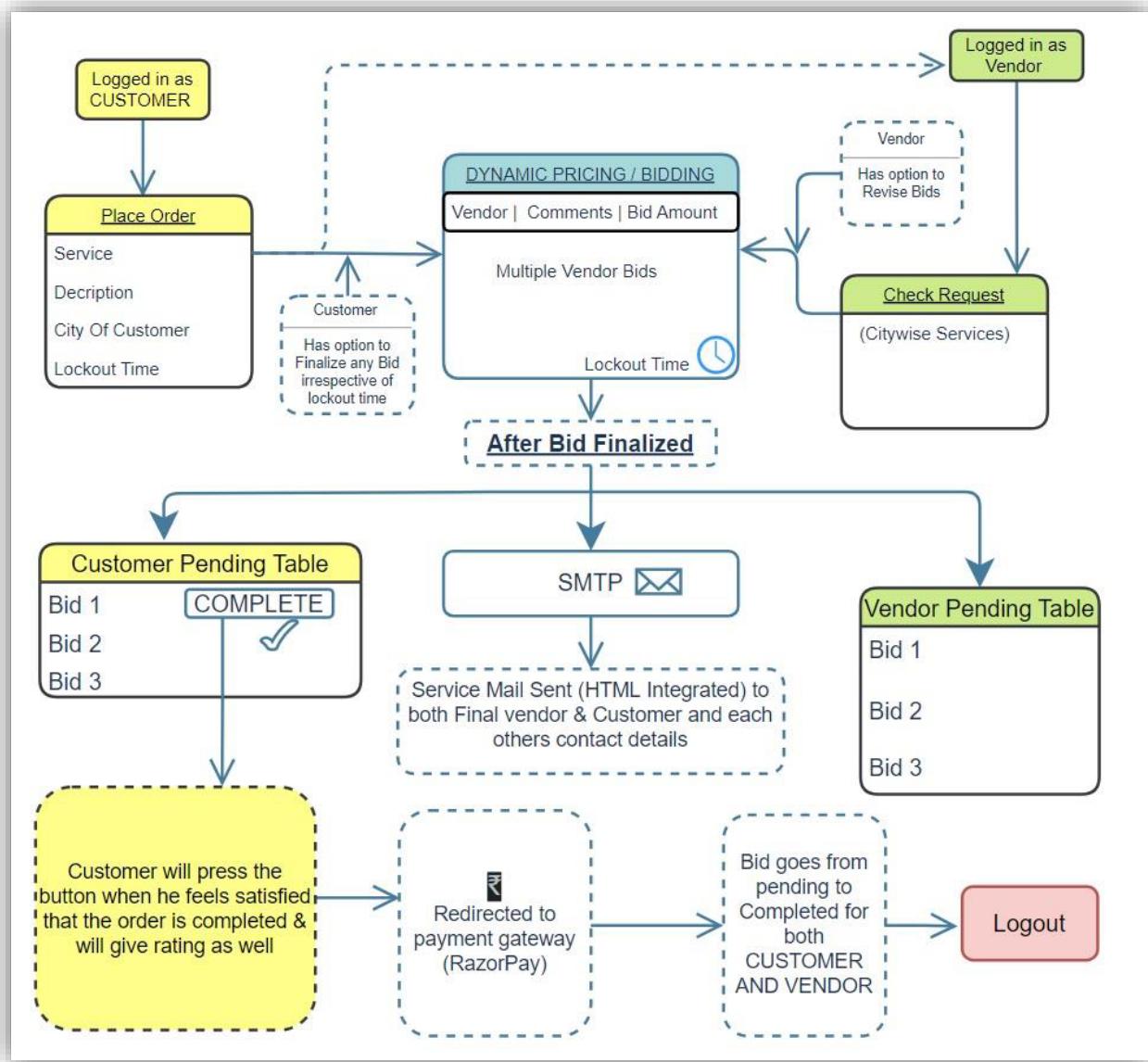
4.4 PROJECT ARCHITECTURE:



4.5 ER DIAGRAM



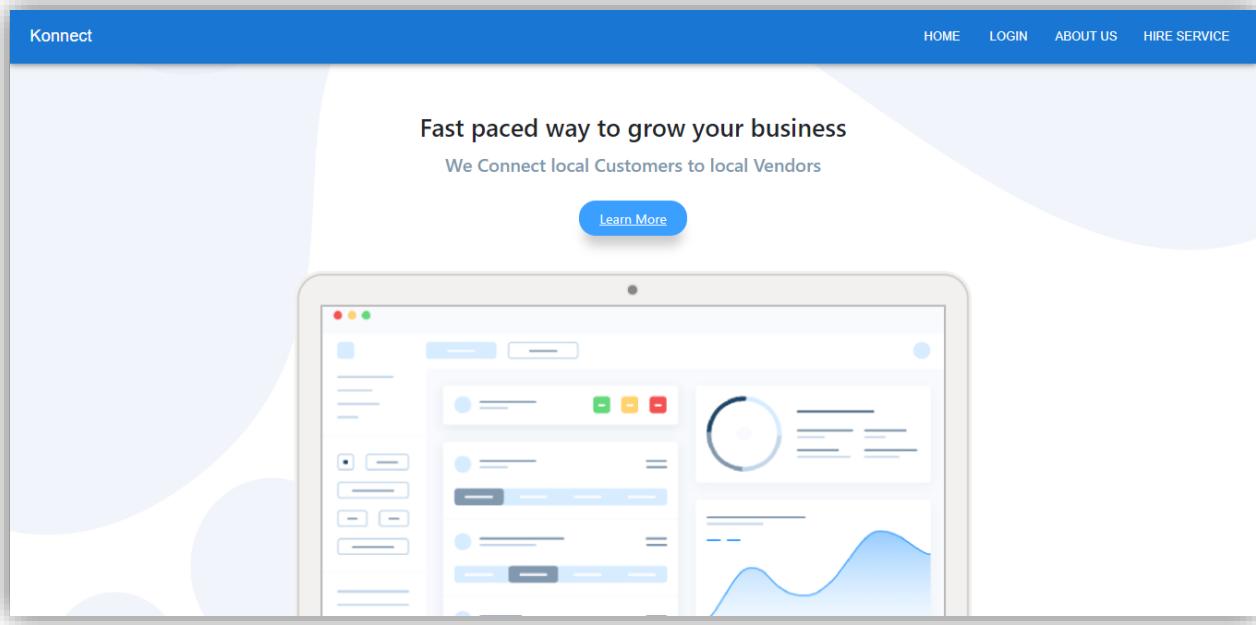
4.6 DATA FLOW DIAGRAM



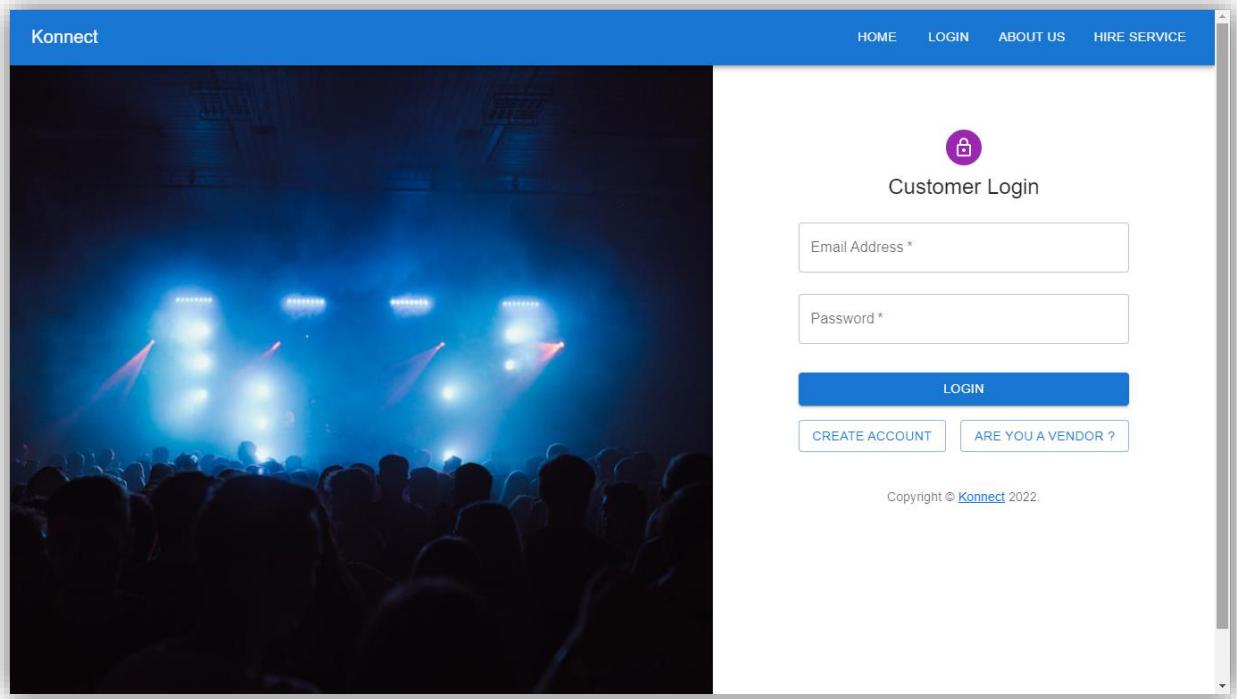
5. PROJECT SCREENSHOTS

CUSTOMER

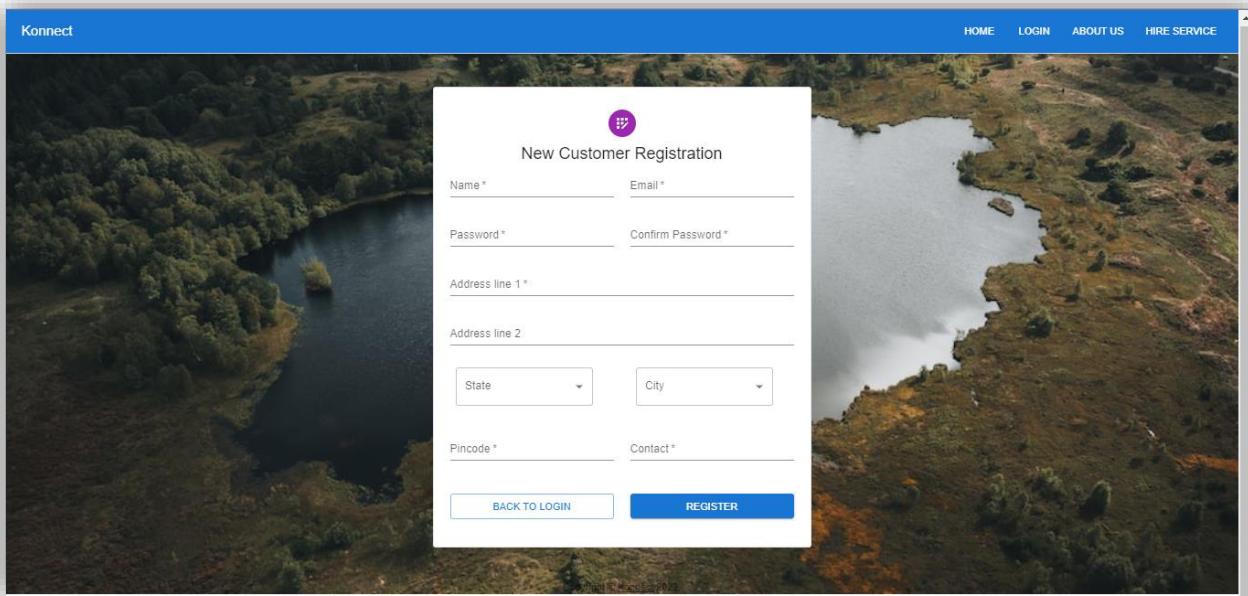
1. Home Page



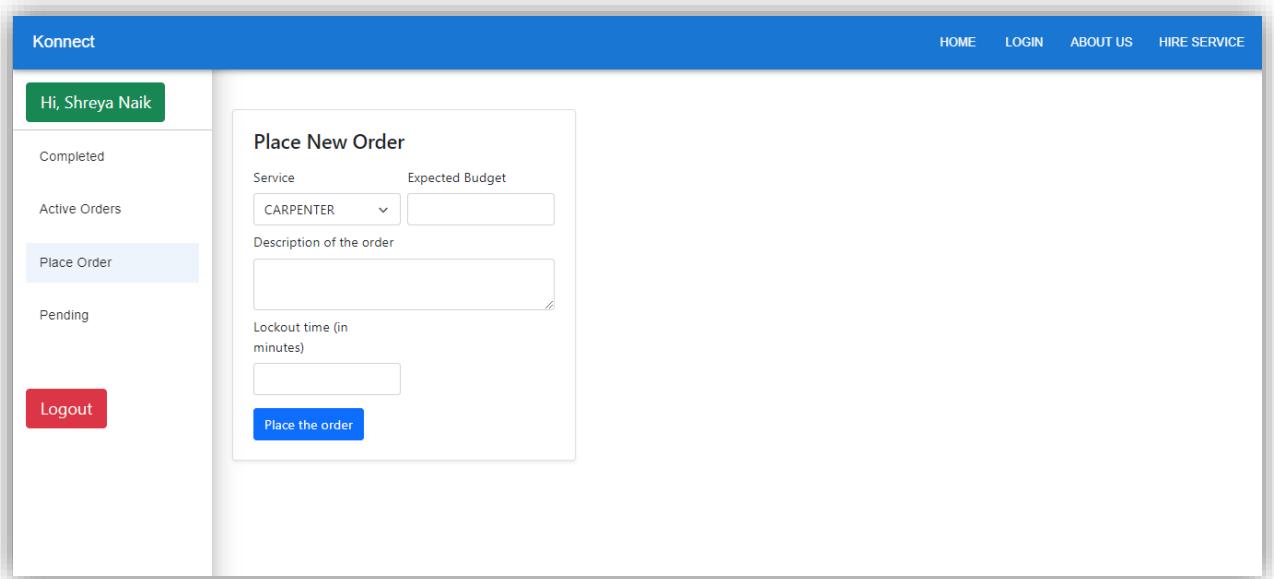
2. Customer Login



3. Customer Registration



4. Place Order



5. Active Orders

The screenshot shows the Konnect platform interface. At the top, there is a blue header bar with the word "Konnect" on the left and navigation links "HOME", "LOGIN", "ABOUT US", and "HIRE SERVICE" on the right. Below the header, on the left side, is a sidebar with a dark green header containing the text "Hi, Shreya Naik". The sidebar has several options: "Completed" (dark grey background), "Active Orders" (light grey background, currently selected), "Place Order" (white background), and "Pending" (white background). At the bottom of the sidebar is a red "Logout" button. The main content area on the right displays a table titled "Order Description" with columns: "Order Description", "Service", "Active Bids", "Current Min Bid", "Time Remaining", and "Bidding". There are two rows in the table:

Order Description	Service	Active Bids	Current Min Bid	Time Remaining	Bidding
Sweeping and cleaning	HOUSEKEEPING			0:18:50	<button>CHECK BIDS</button>
Fan fixing in 6 rooms	ELECTRICIAN			0:24:50	<button>CHECK BIDS</button>

6. Active Bids

The screenshot shows the Konnect platform interface with a modal window open over a dark grey background. The modal has a white header with the number "#2". Inside the modal, the "Order Description" is listed as "Sweeping and cleaning" with a budget of "200". To the right of the description is a timer showing "0:7:55". Below this, there is a table with columns: "Vend...", "Vendor Name", "Vendor Rating", "Vendor Comments", "Bid Placed Time", "Bid Amount ↑", and "Select Vendor". Two entries are shown:

Vend...	Vendor Name	Vendor Rating	Vendor Comments	Bid Placed Time	Bid Amount ↑	Select Vendor
2	Vivek Sharma	★★★☆☆	I can do this in 170	2022-09-26T22:1...	170	<button>START ORDER</button>
1	Bhakti Mungale	★★★★★	I can do this	2022-09-26T22:0...	200	<button>START ORDER</button>

At the bottom of the modal, there are buttons for "ROWS PER PAGE: 100", "1-2 of 2", and navigation arrows. A "CANCEL ORDER" button is located at the bottom left of the modal.

7. Pending Orders

The screenshot shows the Konnect application interface. On the left, a sidebar menu includes 'Completed', 'Active Orders', 'Place Order', 'Pending' (which is selected and highlighted in blue), and 'Logout'. The main content area displays a table of pending orders:

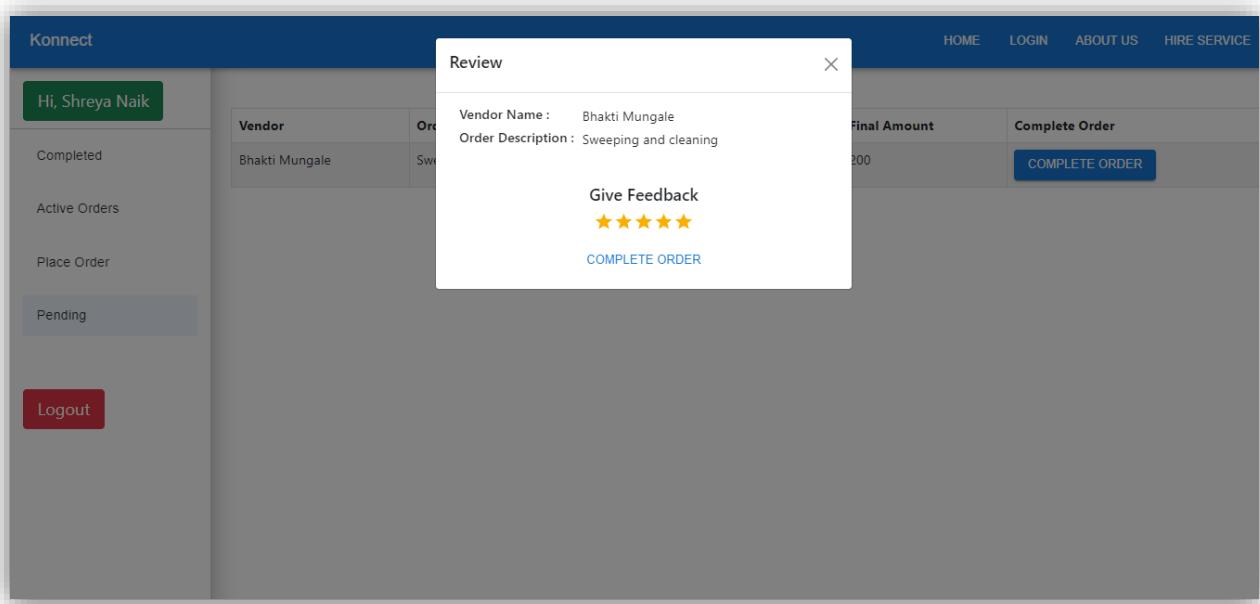
Vendor	Order Description	Service	Final Amount	Complete Order
Bhakti Mungale	Sweeping and cleaning	HOUSEKEEPING	200	COMPLETE ORDER

8. Completed Order

The screenshot shows the Konnect application interface. On the left, a sidebar menu includes 'Completed' (which is selected and highlighted in blue), 'Active Orders', 'Place Order', 'Pending', and 'Logout'. The main content area displays a table of completed orders:

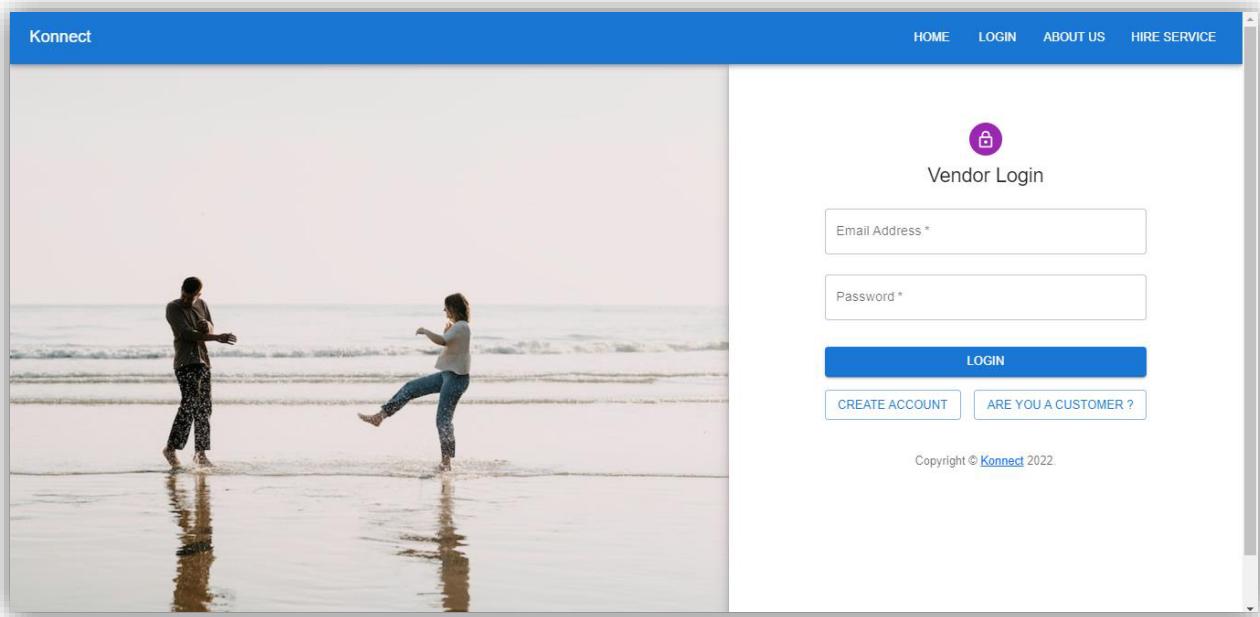
Vendor Name	Order Description	Final Amount	Order Placed Time	Order Completed Time	Service
Bhakti Mungale	Sweeping	170	2022-09-26T16:09:12.60594	2022-09-26T16:11:36.628615	HOUSEKEEPING
Bhakti Mungale	Sweeping and cleaning	200	2022-09-26T22:01:42.020011	2022-09-26T22:14:32.977244	HOUSEKEEPING

9. FeedBack

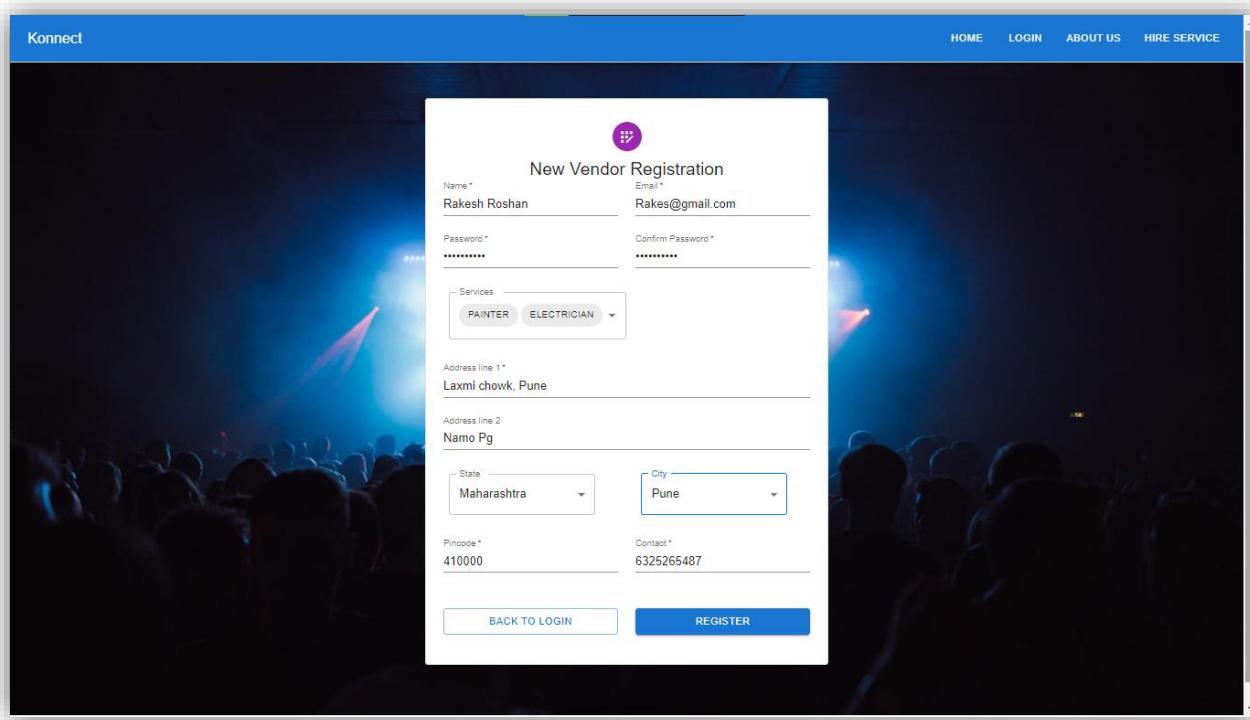


VENDOR

1. Vendor login



2. Vendor Registration

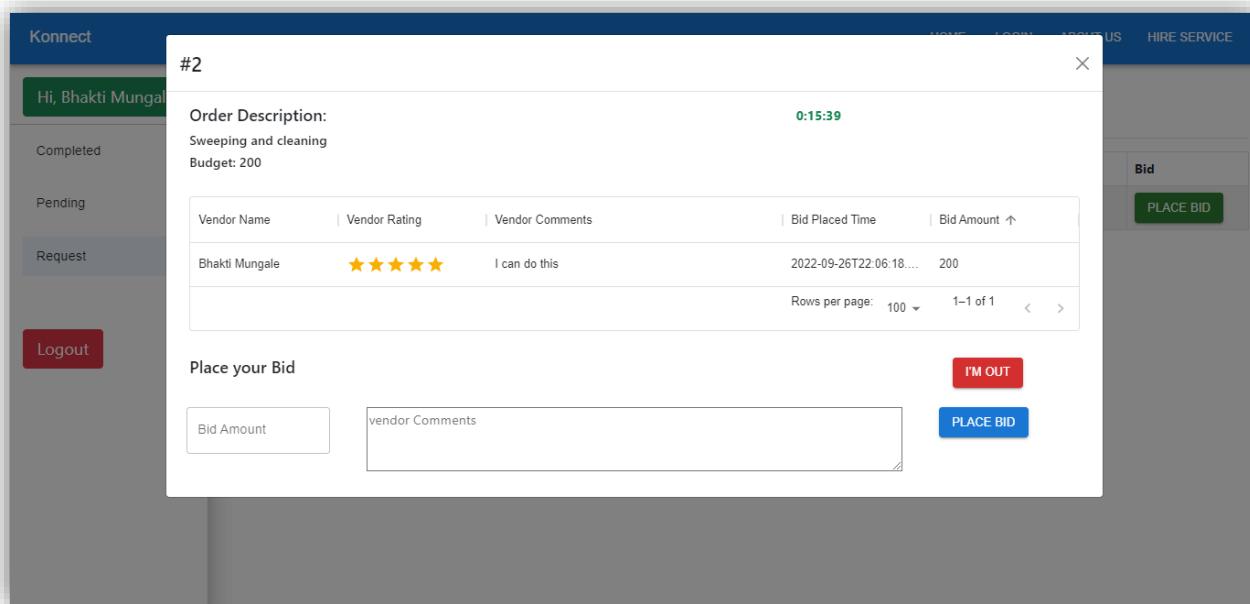


The screenshot shows the 'New Vendor Registration' form on the Konnect platform. The background features a dark, atmospheric image of a concert or event with blue stage lights. The registration form is centered and contains the following fields:

- Name*: Rakesh Roshan
- Email*: Rakes@gmail.com
- Password*: [REDACTED]
- Confirm Password*: [REDACTED]
- Services: PAINTER, ELECTRICIAN
- Address Line 1*: Laxmi chowk, Pune
- Address Line 2: Namo Pg
- State: Maharashtra
- City: Pune
- Pincode*: 410000
- Contact*: 6325265487

At the bottom are two buttons: 'BACK TO LOGIN' and 'REGISTER'.

3. Place Bid



The screenshot shows the 'Place Bid' interface for a specific request. The top bar shows the user 'Hi, Bhakti Mungale'. The left sidebar has tabs for 'Completed', 'Pending', and 'Request', with 'Request' currently selected. A red 'Logout' button is also visible.

The main area displays the following information:

- #2
- Order Description: Sweeping and cleaning
- Budget: 200
- Time: 0:15:39
- Vendor Name: Bhakti Mungale
- Vendor Rating: ★★★★★
- Vendor Comments: I can do this
- Bid Placed Time: 2022-09-26T22:06:18....
- Bid Amount: 200

Below this, there are buttons for 'I'M OUT' and 'PLACE BID'. At the bottom, there are input fields for 'Bid Amount' and 'vendor Comments'.

4. Pending Orders

The screenshot shows the Konnect platform interface for a user named Bhakti Mungale. The left sidebar has buttons for 'Completed', 'Pending' (which is selected), and 'Request'. A 'Logout' button is at the bottom. The main area is titled 'HOUSEKEEPING' and shows a table for a pending order. The table columns are Customer Name, Order Description, Customer Address, Active Bids, Current Minimum Bid, Time Remaining, and Bid. The data is: Customer Name - Shreya Naik, Order Description - Sweeping and cleaning, Customer Address - Mantri colony fulai nagar, Active Bids - -, Current Minimum Bid - -, Time Remaining - 0:16:52, and a 'PLACE BID' button.

HOUSEKEEPING				LAUNDRYMAN		
Customer Name	Order Description	Customer Address	Active Bids	Current Minimum Bid	Time Remaining	Bid
Shreya Naik	Sweeping and cleaning	Mantri colony fulai nagar	-	-	0:16:52	<button>PLACE BID</button>

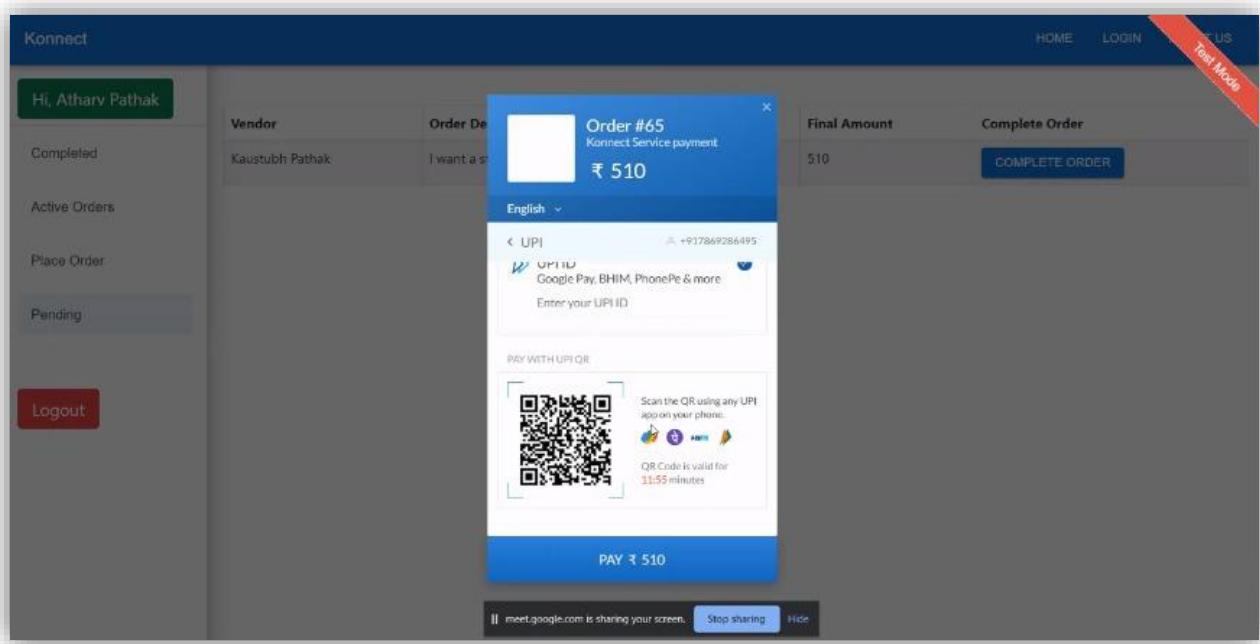
5. Completed Orders

The screenshot shows the Konnect platform interface for a user named Bhakti Mungale. The left sidebar has buttons for 'Completed' (selected), 'Pending', and 'Request'. A 'Logout' button is at the bottom. The main area is titled 'HOUSEKEEPING' and shows a table of completed orders. The table columns are Customer Name, Order Description, Customer Address, Customer Contact, Final Amount, and Rating Given. The data is: Customer Name - Shreya Naik, Order Description - Sweeping, Customer Address - Mantri colony fulai nagar, Customer Contact - 9874563210, Final Amount - 170, and a 5-star rating. Another row shows: Customer Name - Shreya Naik, Order Description - Sweeping and cleaning, Customer Address - Mantri colony fulai nagar, Customer Contact - 9874563210, Final Amount - 200, and a 5-star rating.

HOUSEKEEPING				LAUNDRYMAN	
Customer Name	Order Description	Customer Address	Customer Contact	Final Amount	Rating Given
Shreya Naik	Sweeping	Mantri colony fulai nagar	9874563210	170	
Shreya Naik	Sweeping and cleaning	Mantri colony fulai nagar	9874563210	200	

Responses

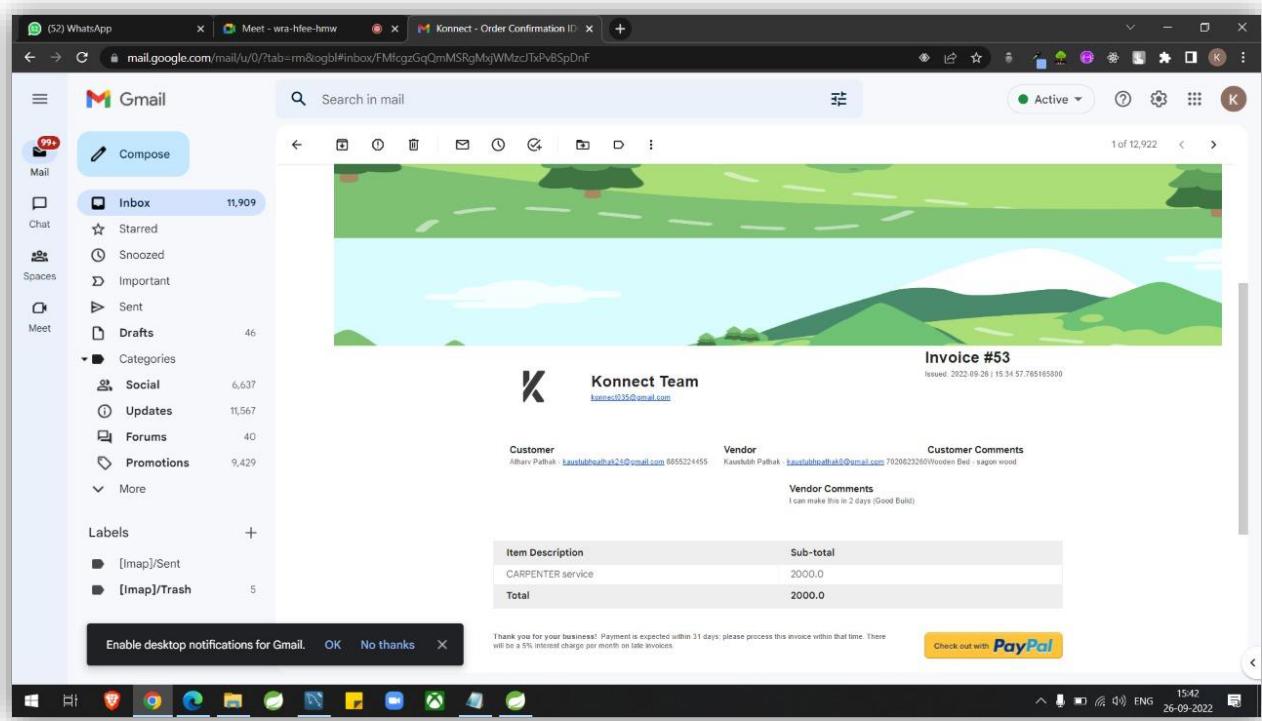
1. Payment



2. RazorPay Dashboard

A screenshot of the RazorPay dashboard. The left sidebar lists various features: Account Activation, Home, Transactions (selected), Settlements, Invoices, Payment Links, Payment Pages, Payment Button, Route, Subscriptions, QR Codes (marked as NEW), Smart Collect, Customers, and Offers. The main area shows a table of transactions. The table has columns for Payment Id, Razorpay Order Id, Amount, Email, Contact, Created At, and Status. There are 4 rows of data, all marked as 'Captured'. The first row: pay_KNHL100Y3xaLnK, order_KNHL1cH3iIYCcg, ₹ 510.00, kaustubhpathak24@gmail.com, +917869286495, 28 Sep 2022, 03:27:10 pm, Captured. The second row: pay_KNHb0vCm1QuK86, order_KNHbessr7wL0i, ₹ 210.00, kaustubhpathak24@gmail.com, +917869286495, 28 Sep 2022, 03:15:21 pm, Captured. The third row: pay_KNHb5kLYK08ca0, order_KNHbgatQshuJqv, ₹ 600.00, kaustubhpathak24@gmail.com, +917869286495, 28 Sep 2022, 03:13:31 pm, Captured. The fourth row: pay_KNGt8Sp0ClnCn, order_KNgSyZyo1711ka, ₹ 455.00, kaustubhpathak24@gmail.com, +917788556655, 28 Sep 2022, 03:00:57 pm, Captured. At the top, there's a message: 'YOU'RE IN TEST MODE' with a toggle switch, and 'Test Mode' buttons. Below the table, it says 'Showing 1 - 4'.

3. E-Mail



6. TESTING

A number of software testing strategies have been proposed in the literature. All provide you with a template for testing and all have the following generic one of the main purposes of testing is to validate and verify that the system works as intended. No program or system design is perfect. However, if we implement the system without proper testing, then it may cause problems and lead to a bad user experience. Testing and checking outcomes of each test gives us the best chance to detect and correct errors before the system is implemented in a production environments. In the course of our project, we made an effort to manually test each component. In all cases, we obtained the desired results as demonstrated below.

1. Customer Test Cases

No	Description	Outcome	Result
1.	Register as Customer	New customer details saved in the database.	Passed
2.	Login as Customer	Fetched authenticated user details saved in database.	Passed
3.	Select Service	Fetch desired service from database	Passed
4.	Budget	Budget in the box	Passed
5.	Description	Description about order in checkbox	Passed
6.	Lockout time	Lockout time in checkbox	Passed
7.	Place the order	Place the order for bidding	Passed
8.	Active orders	Fetch the active orders from database	Passed
9.	Pending	Fetch the pending orders from database	Passed
10.	Completed	Fetch the completed orders from database	Passed
11.	Logout	The session was cleared.	Passed

2. Vendor Test Cases

No.	Description	Outcome	Result
1.	Sign in as Vendor	Fetched authenticated user details saved in database.	Passed
2.	Request	Fetch the details about the order from the database	Passed
3.	Pending	Fetch the details of pending order from database	Passed
4.	Completed	Fetch the details about the completed orders from database	Passed
5.	Logout	The session was cleared.	Passed

7. FUTURE SCOPE

While our website currently focuses on services, In future we want to broaden the horizons of our website by including the categories of products as well. This will allow marketplace to determine dynamic pricing of every commodity which the customer desires.

Currently our app applies the model of "Pay when service is done" but we would like to integrate a payment gateway which would take the payment from the customer beforehand but hold it till the service is not completed. Once the service is done the required amount will be credited to the vendor.

Furthermore, we would like to add a live tracking functionality on the lines of food delivery services like Zomato, which will track your service provider once he departs to fulfill your requirement.