**PROJECT REPORT**

* **TITLE:**

‘SERVICELY’

PROVISION OF SERVICES THAT SUITS YOU!

* **INTRODUCTION:**

Servicely - an online marketplace for all your everyday service needs. Our platform connects customers with a wide range of pre-screened and vetted professionals who can provide a variety of services, including plumbing, carpentry, mechanics, electrical work, home cleaning, landscaping, moving, and more. Our goal is to save our customers time and hassle and provide them with peace of mind knowing that their service needs are taken care of by highly skilled professionals.

* **WORKING:**

The Servicely website would work by allowing customers to create an account and login to the platform. Once logged in, customers would be able to browse through a directory of provided services. After choosing a service, service providers would be shown, each with their own profile that includes information such as their qualifications, experience, and reviews from previous customers.

Customers would also have the ability to submit appointment requests through the website. These requests would include details about the job such as the address and the date and time for the appointment. Service providers would then be able to view these requests and choose to accept or reject them.

Service providers would also have their own login and profile on the website, where they would be able to view and manage their appointment requests and monitor their progress. Once the job is complete, customers would be able to leave a review and rate the service provider.

**CUSTOMER FUNCTIONS:**

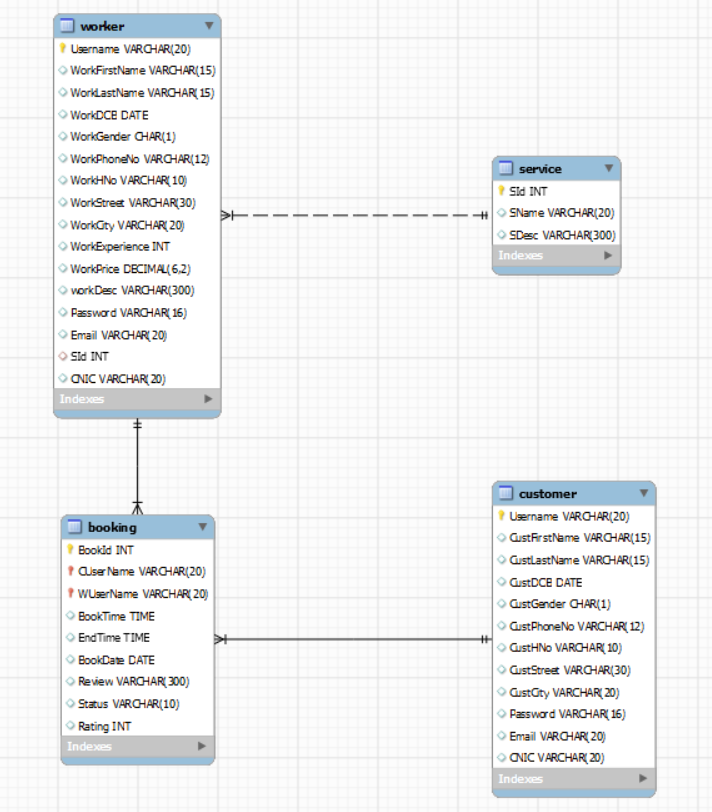
1. Create account. Name, Username, Password, Address, Phone, Email, Cnic, Dob, Gender.
2. On opening home page user will see different services provided. On clicking a particular service user will be directed to a page dedicated to that service.
3. A page dedicated for services will show the individuals that provided that service. By default, the service providers will be listed in order of descending ratings.
4. Above the listing of service providers, a dropdown menu will be provided to the user to order the listing according to his requirement.
5. List can be ordered by rating and price.
6. After selecting the service provider of choice, user can click in the hire button and schedule an appointment.
7. A calendar will show the time slots in which the service provider is available.

**SERVICE PROVIDER FUNCTIONS:**

1. Create account. Name, Username, Password, Address, Phone, Email, Cnic, Dob, Gender, Profession, Experience, Price/hour.
2. See previous work done.
3. See and manage bookings requests.
4. See their stats for example number of bookings made shown in a month, last 7 days.

DATABASE

* **ERD:**



* **TABLES:**

1. **SERVICE:**

This table stores the services available at our website

* + Sid: service id
  + SName: service name
  + SDesc: service description

1. **CUSTOMER:**

This table stores the data of customers who create an account on our website

* + Username: the username of customer accounts for logging in
  + Password: the password of customer account for logging in
  + CustFirstName
  + CustLastName
  + CustDOB
  + CustGender
  + CustPhoneNo: unique
  + CustHNo
  + CustStreet
  + CustCity
  + Email: unique
  + CNIC: unique

1. **WORKER:**

This table stores the data of workers/service providers who create an account on our website

* + Username: the username of worker accounts for logging in
  + Password: the password of worker account for logging in
  + WorkFirstName
  + WorkLastName
  + WorkDOB
  + WorkGender
  + WorkPhoneNo: unique
  + WorkHNo
  + WorkStreet
  + WorkCity
  + Email: unique
  + CNIC: unique
  + WorkExperience
  + WorkPrice
  + WorkDesc
  + SId: Foreign key from Service table that tells the profession of worker

1. **BOOKING:**

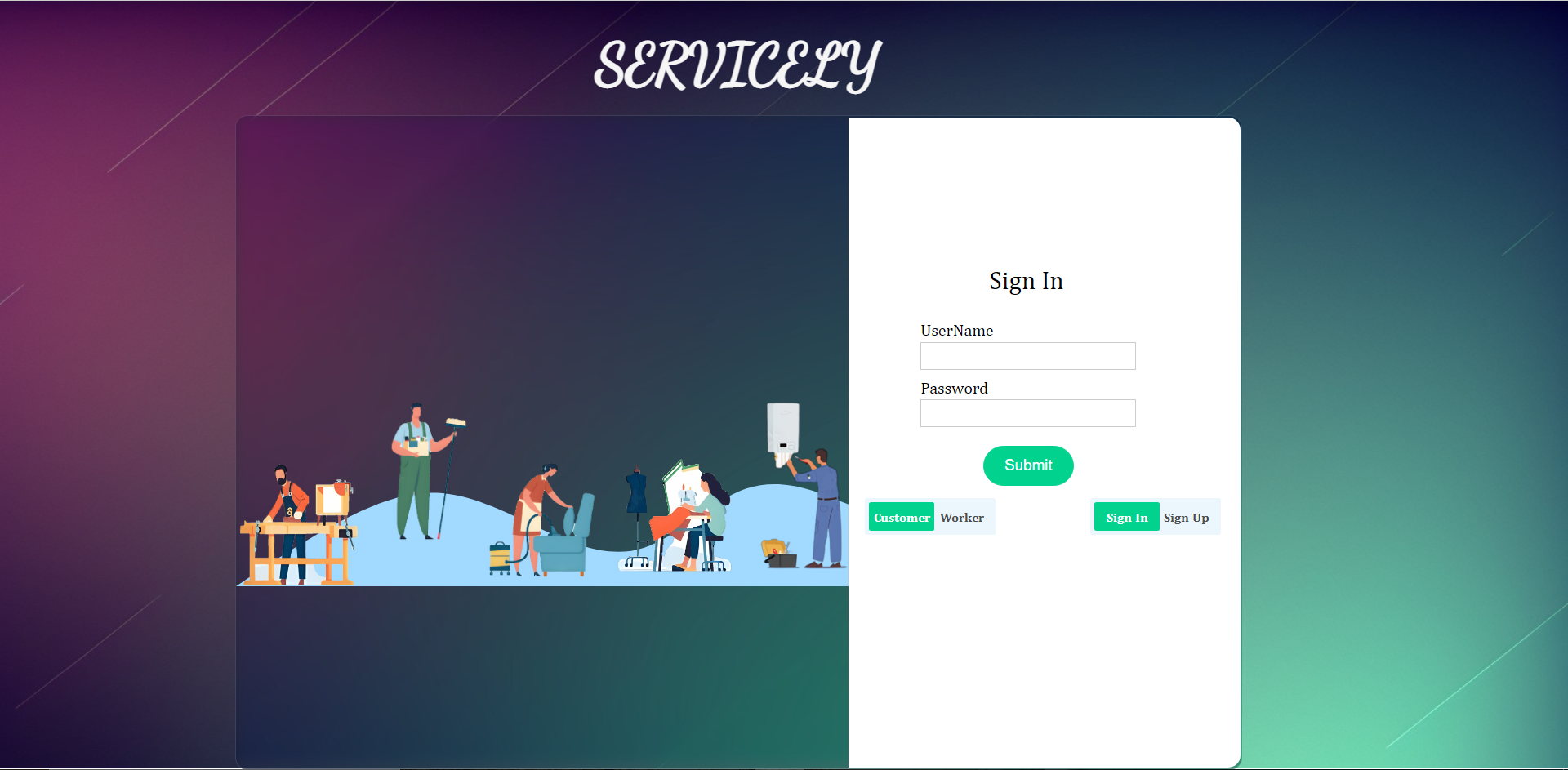
This table stores the details of bookings. It is a weak entity.

* + BookId
  + CUserName: Foreign key from Customer table to tell which customer made an appointment
  + WUserName: Foreign key from Worker table to tell that for which worker the appointment is made
  + BookTime
  + EndTime
  + BookDate
  + Review
  + Status: ‘Pending’, ‘Accepted’, ‘Rejected’ or ‘Completed’
  + Rating: Rating given by customer having range 1 to 5
* Conversion of ERD to relational schema was considered effectively by keeping in mind all the essential factors for conversion of ERD to relational model. Meanwhile, Normalization was also kept in mind, due to which all anomalies were removed, and our schema is normalized up to 3NF.

FRONT END

Our website is composed of four pages

* **LOGIN:**



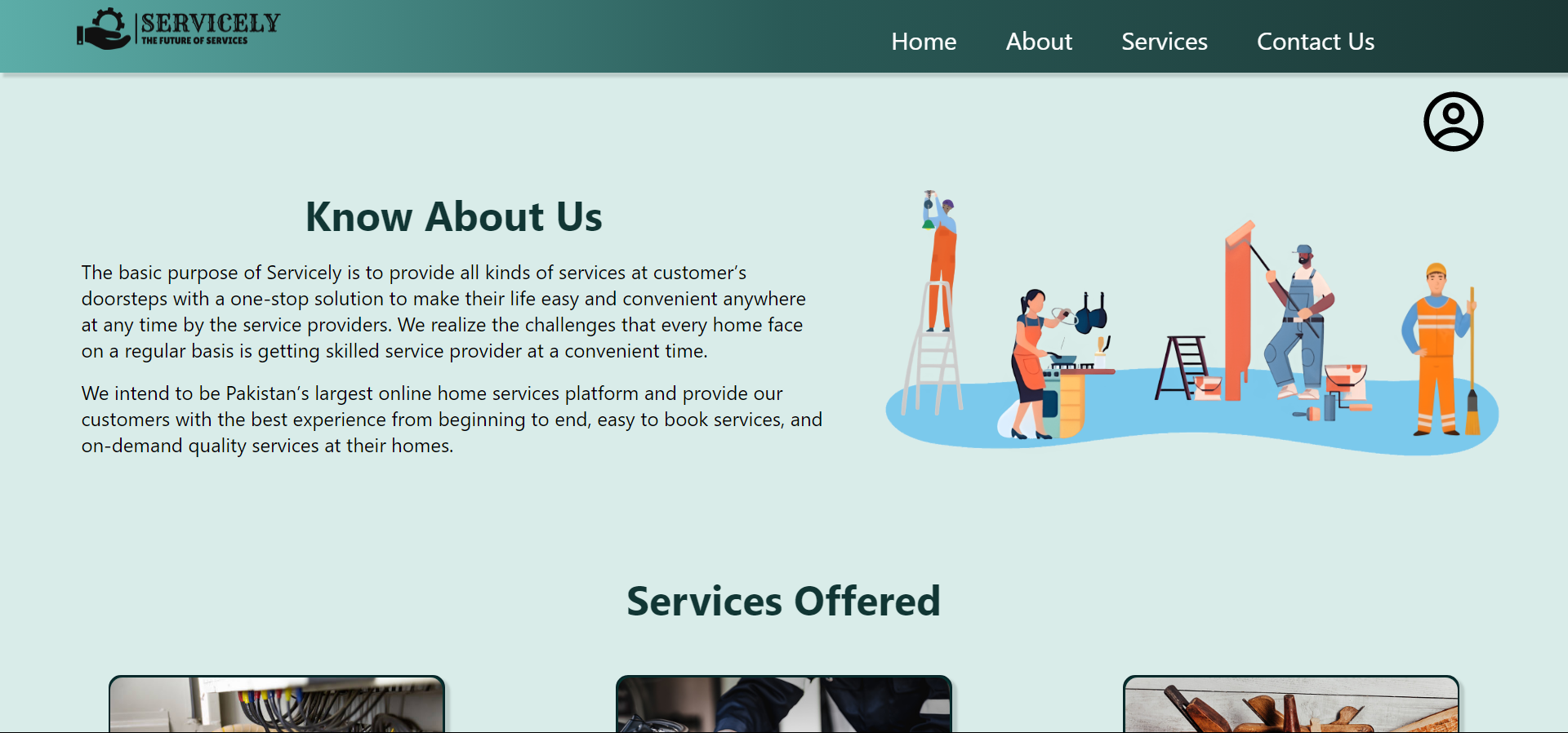
**FUNCTIONS:**

1. Customer login/sing-up
2. Worker login/sign-up

**BACK-END METHODS:**

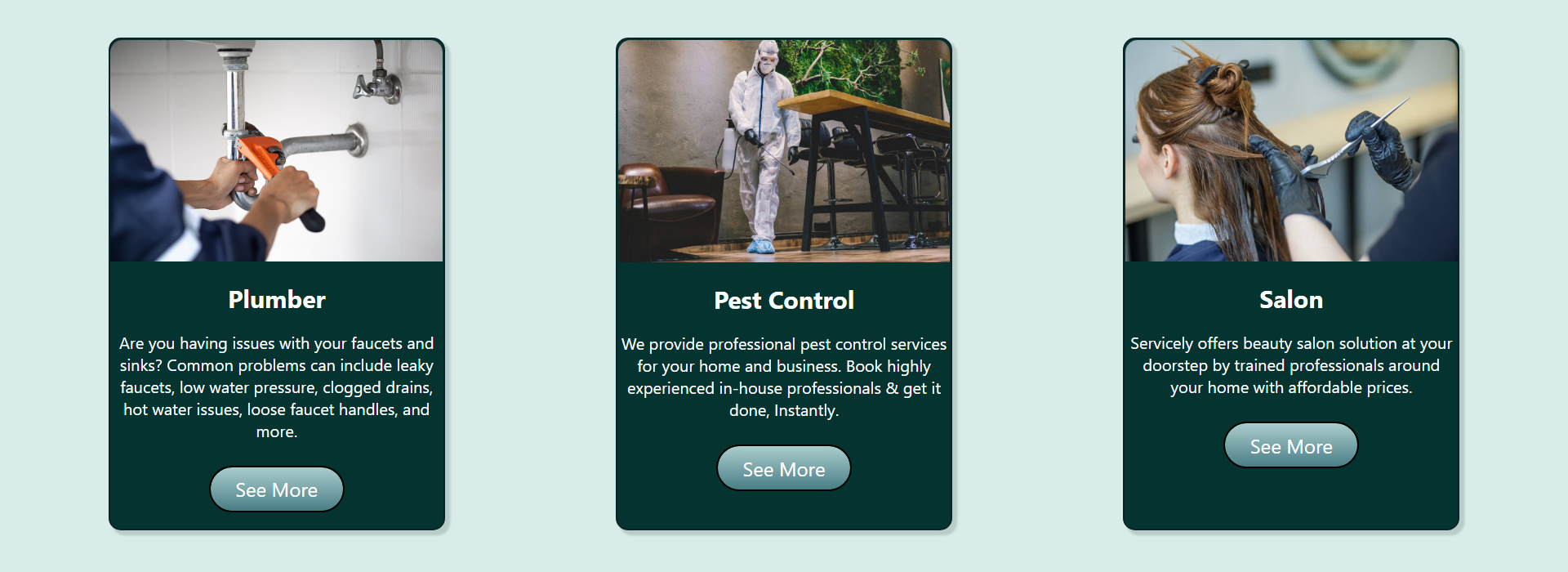
1. login: for customer/worker login
2. createCustomer: for customer sign-up
3. createWorker: for worker sign-up

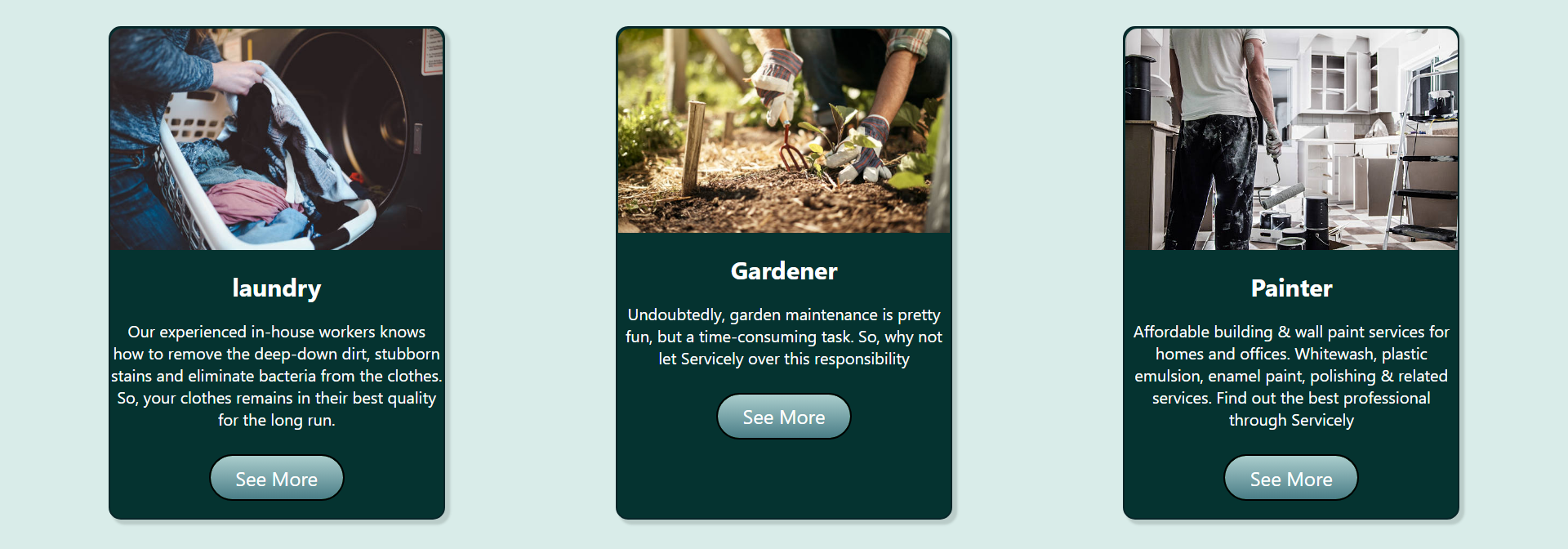
* **CUSTOMER HOME:**

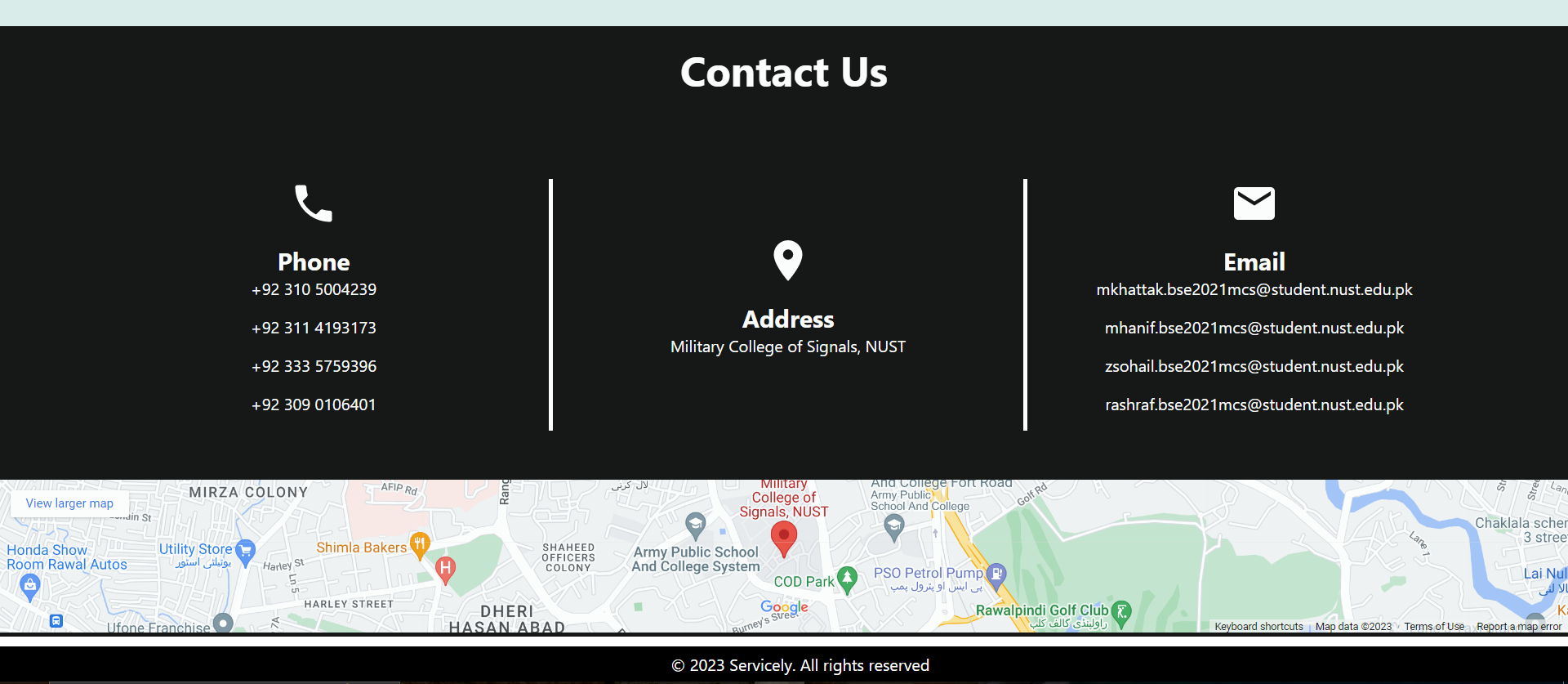


Graphical user interface, application

Description automatically generated







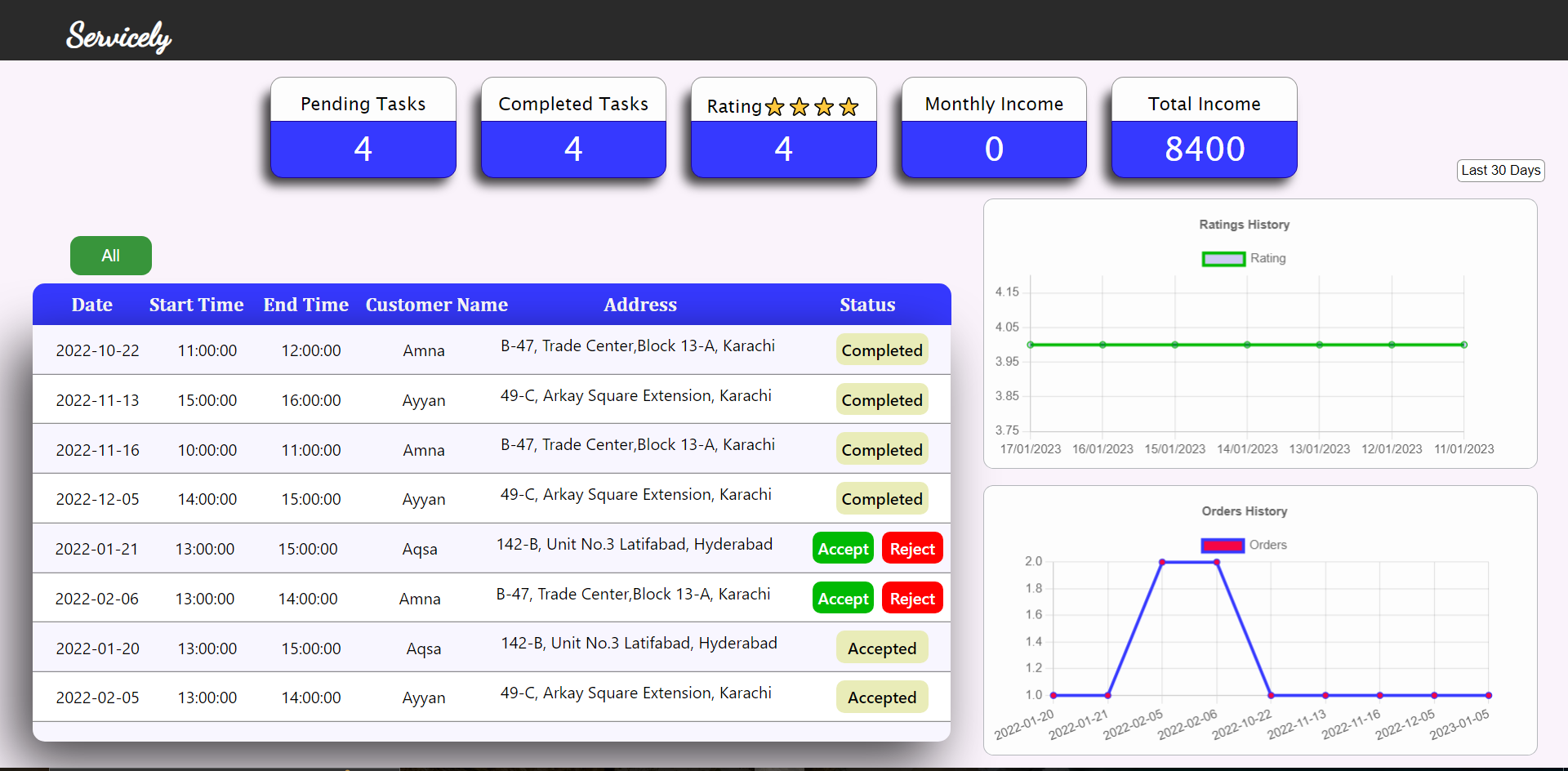
**FUNCTIONS:**

1. Display customer profile
2. Display all services

**BACK-END METHODS:**

1. getCustomer: for getting all information of customer that logged-in
2. getServices: for getting information of all services
3. getReviewBook: for getting information of all pending reviews
4. updateReview: to add the review and rating given in the booking table

* **WORKER HOME:**



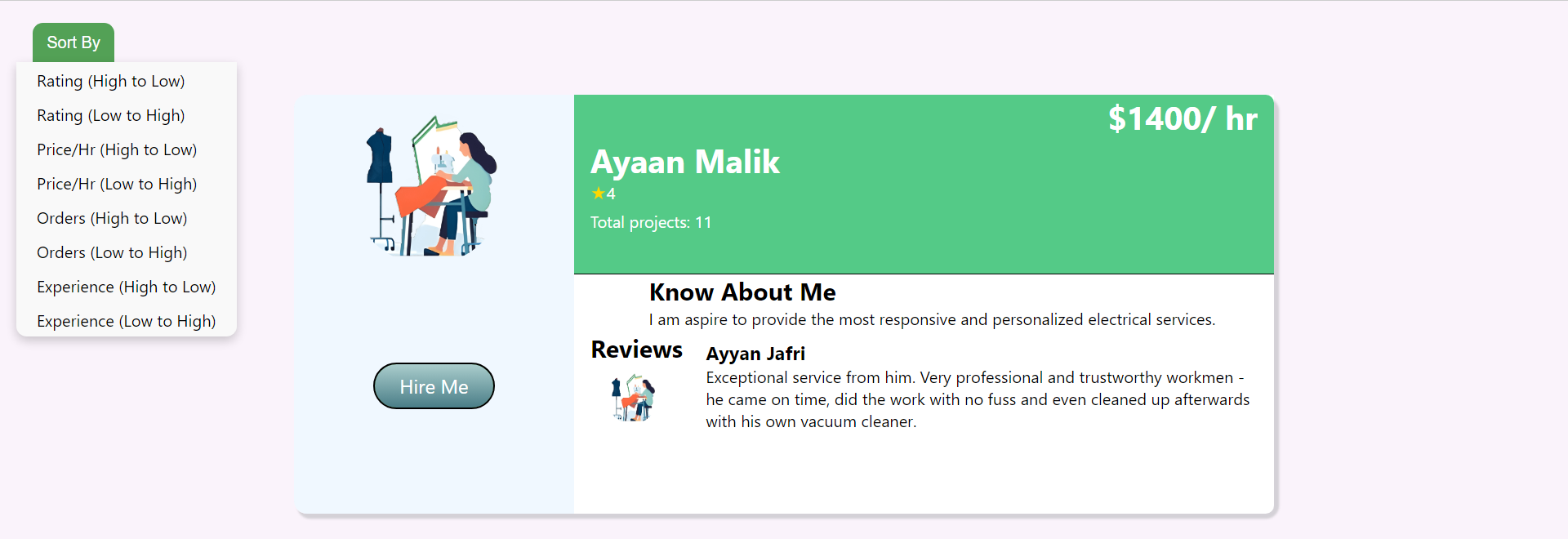
**FUNCTIONS:**

* 1. Display worker profile
  2. Display all the bookings of worker
  3. Display stats of rating and no. of orders

**BACK-END METHODS:**

1. getBooking: for getting all bookings of worker that logged-in
2. updateBooking: update bookings when worker accepts or rejects them
3. getOrderChartData: for getting information to plot Order-Chart
4. getRatingChartData: for getting information to plot Rating-Chart

* **WORKER CARDS:**





**FUNCTIONS:**

1. Display worker profiles for the particular service
2. Schedule booking for customer
3. Save booking requests

**BACK-END METHODS:**

1. getWorkerDetails: for getting information about all workers of the particular service
2. getWorkerBookings: for getting all bookings of worker customer clicked on
3. saveBooking: add a new booking in Booking table when user hires a worker