Devansh

G-23 (2010991860)

Gas Booking Website

12, February, 2023

Overview

This project aims to build a web-based application for booking and managing gas deliveries for customers. The application will allow users to create an account, view their booking history, and place new orders for gas deliveries. The application will also have an admin panel for managing customer accounts and monitoring delivery orders.

Technology Stack

The application will be built using the MERN stack (MongoDB, Express.js, React.js, and Node.js).

MongoDB: MongoDB will be used to store customer and order information.

Express.js: Express.js will be used as the back-end web framework for building the API endpoints for the application.

React.js: React.js will be used as the front-end framework for building the application's user interface.

Node.js: Node.js will be used as the runtime environment for the application.

Outline Of The Project

The application will have two main parts: the customer-facing part and the admin part.

Customer-Facing Part:

- Users will be able to create an account.
- The user will be able to log in.
- Users have a user-dashboard which displays their information and recently placed orders.
- Users will be able to view their booking history.
- Users will be able to place new orders for gas deliveries.
- Users will be able to give feedback.

Admin Part:

- Admin will be able to view all customer accounts.
- Admin will be able to monitor delivery orders.
- Admin will be able to update the status of orders.
- Admin will be able to manage the inventory of gas.
- Admin is able to view user feedbacks on their dashboard.

Endpoints:

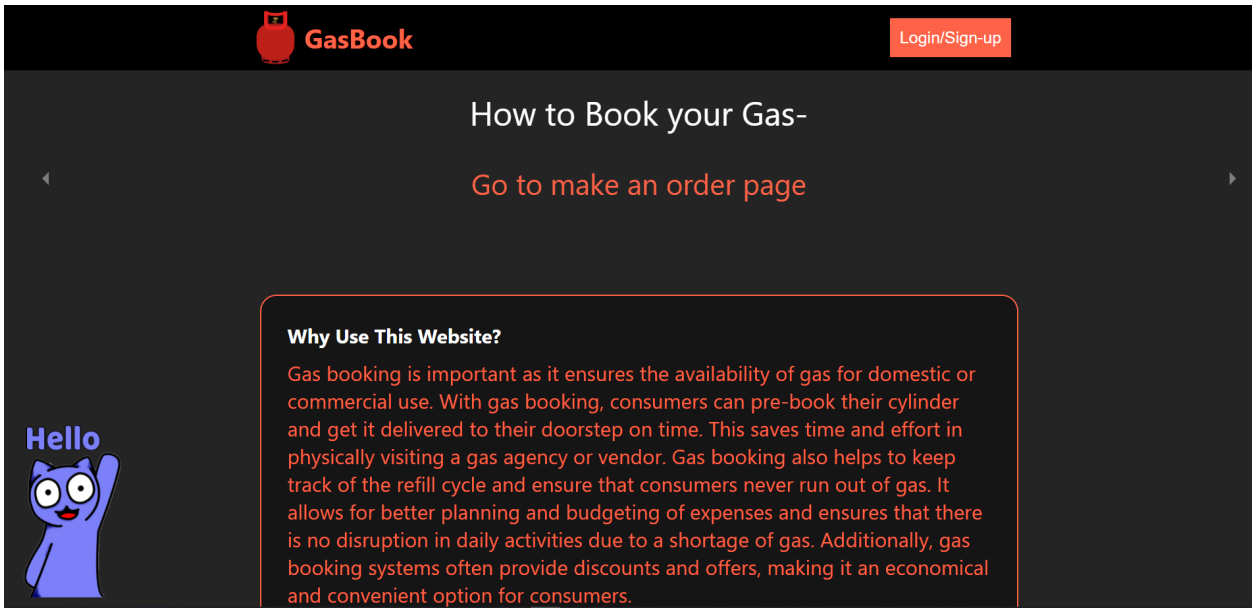
- **POST /register:** for customer registration
- **POST /login:** for customer login
- **GET /orders:** for fetching customer orders
- **POST /orders:** for creating a new order
- **PUT /orders/:id:** for updating order status
- **GET /admin/customers:** for fetching all customer
- **GET /admin/orders:** for fetching all orders
- **PUT /admin/inventory:** for updating the inventory
- **GET /feedback:** for fetching all the feedbacks

- **POST /feedback:** for creating a new feedback from a user

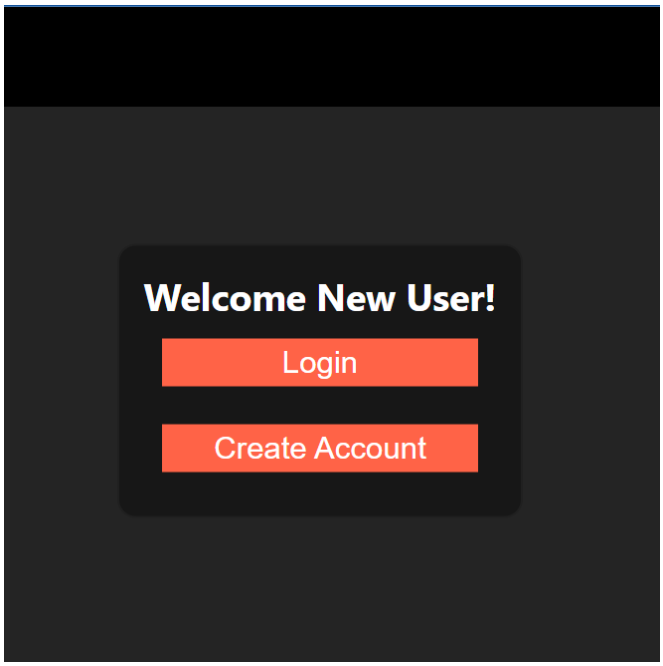
Working Of The Website:

General Part-

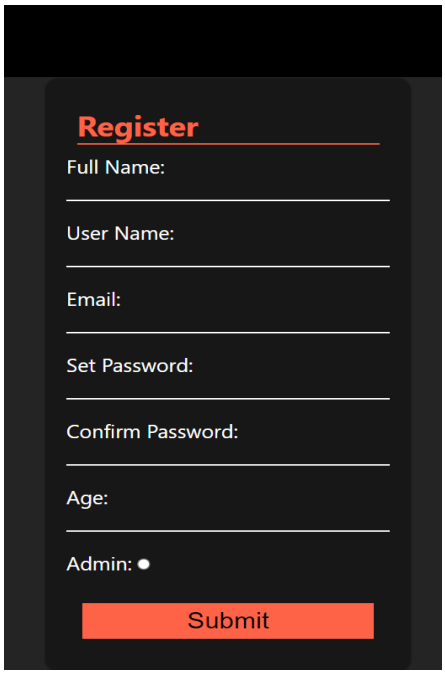
Home Page:



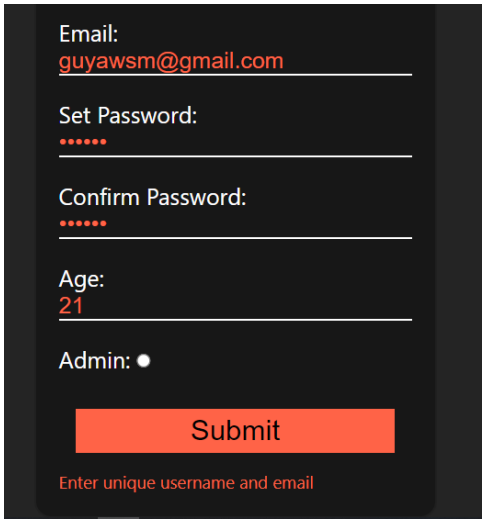
The simple home page contains a carousel, which guides people on how to book gas cylinders on the website. To access the user functions, one must make sure they are logged in, which is why the navbar currently has the Login/Sign-up button.



Register Page:



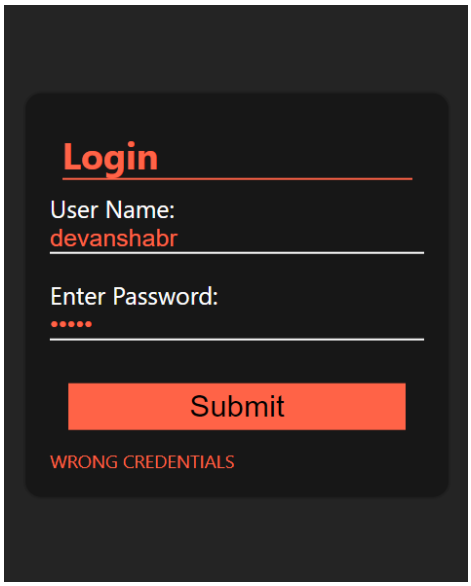
This is the common Register page, wherein a user can enter his/her details and create an account, the user name, and email fields must be **unique**, or the application throws a warning-



The password entered by the user is encrypted and then stored on MongoDB using a manually written encryption algorithm, which uses the Unicode of given characters in the password string, and a secret string, and then takes the XOR of the two to create an encrypted string. Example-

Here the string qwerty (the password here) gets encrypted, before being stored on MongoDB.

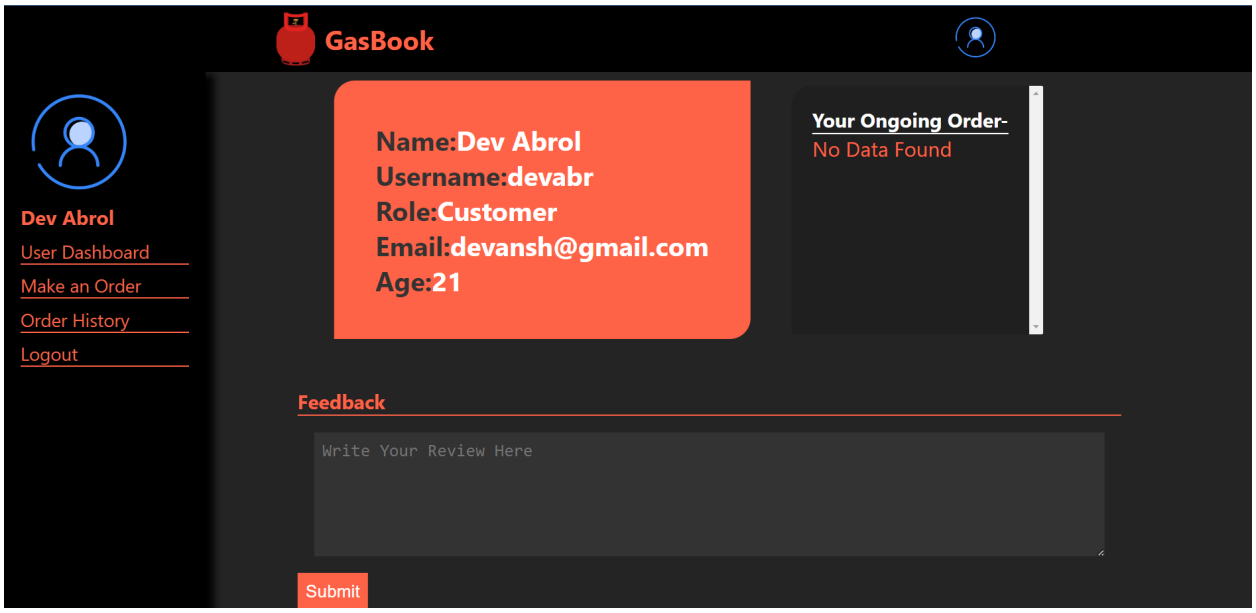
Login Page:



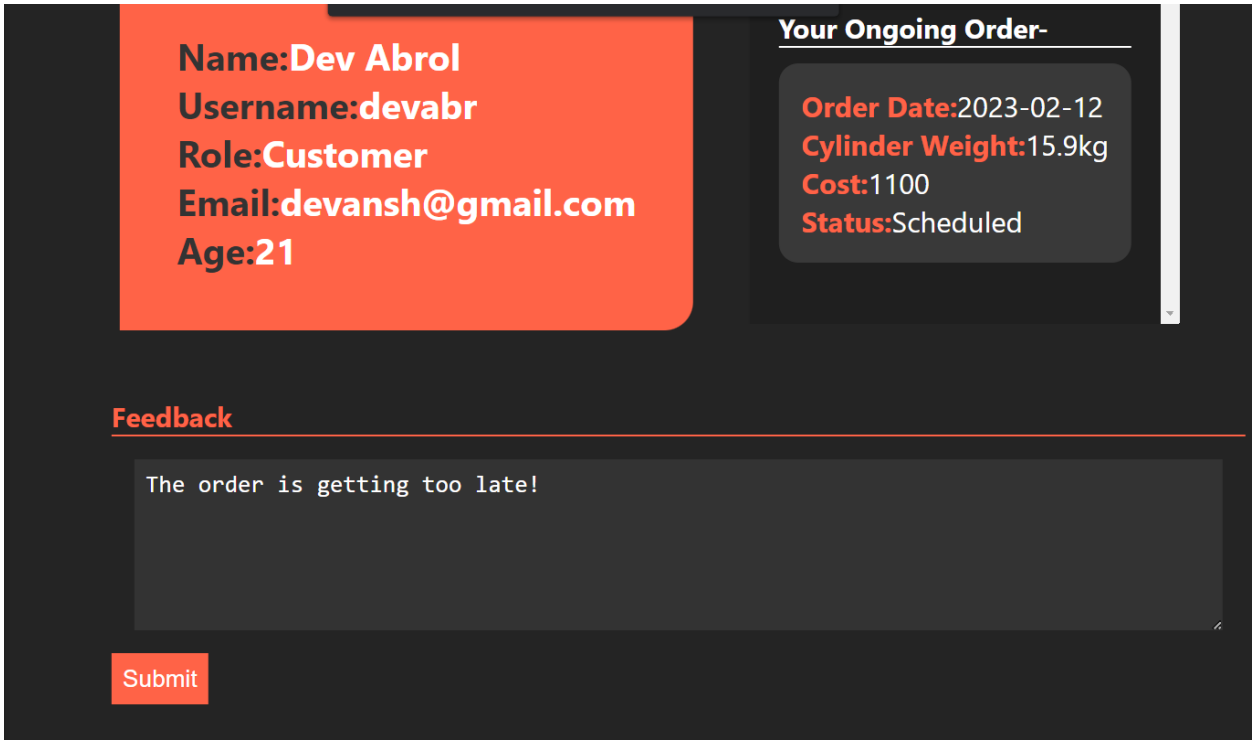
The simple login page checks user credentials and returns a message if any of the credentials entered are wrong. Once logged in, the username, and his/her role is saved in the local storage so that when they enter the site again, they remain logged in already. Once they log out, their information is also deleted from local storage.

User Side-

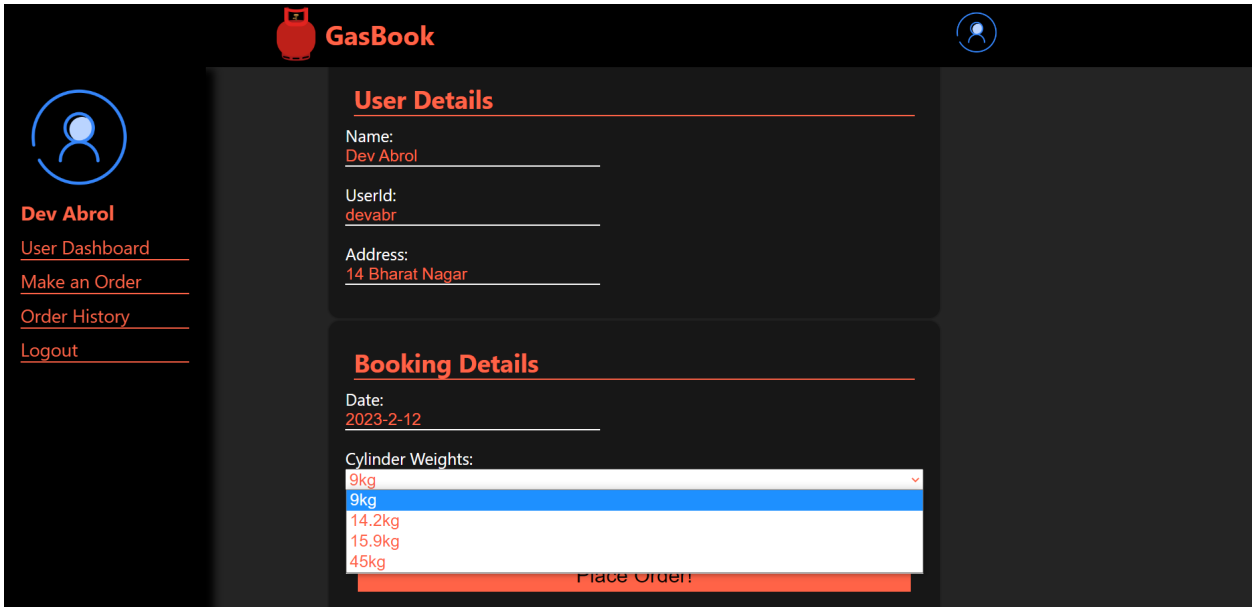
User Dashboard-



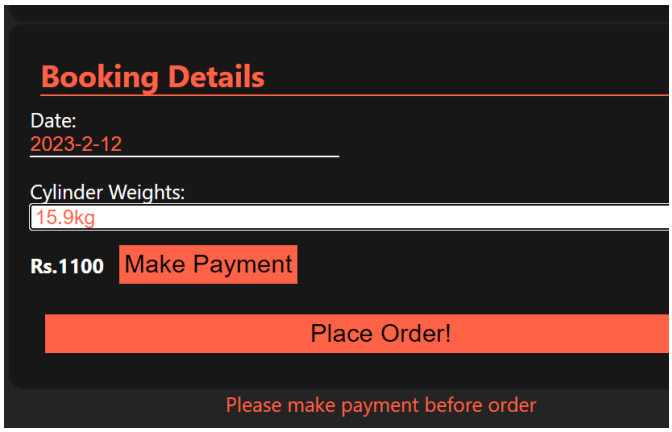
The user dashboard contains three things- User information, their recently placed orders (which aren't delivered yet), and a feedback form.

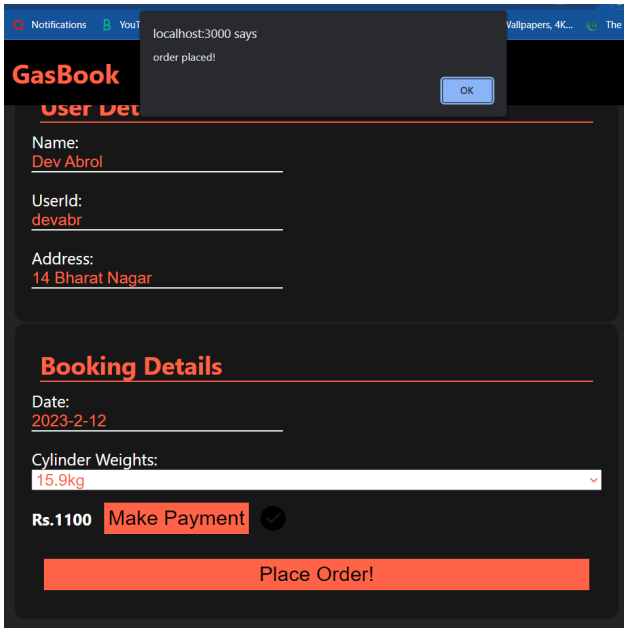


Making an order:



To make an order, one must fill out the form providing their details like Address, and select the weight of the cylinder they wish to order. The process doesn't proceed, if the user hasn't made any payments, and displays a message until the payment is made-

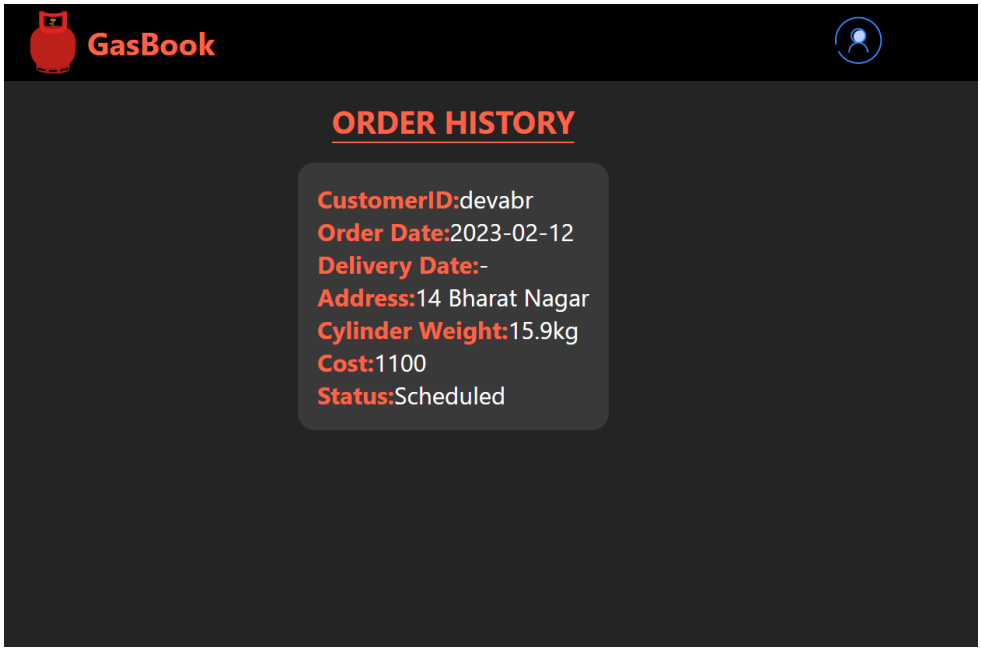




Once, the payment is made, the user can make the order, and an alert box gets displayed.

Order History-

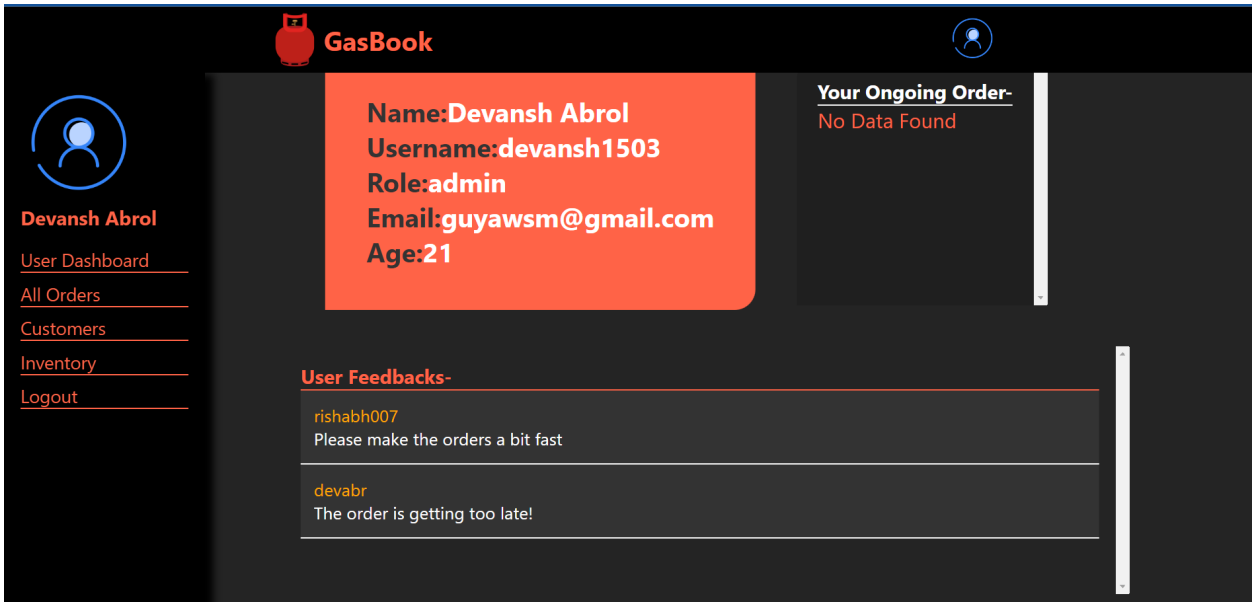
The user can view their order history to gather information on all the orders they made in the past. The orders which aren't been delivered yet display no delivery date and are status-scheduled.



Admin Part-

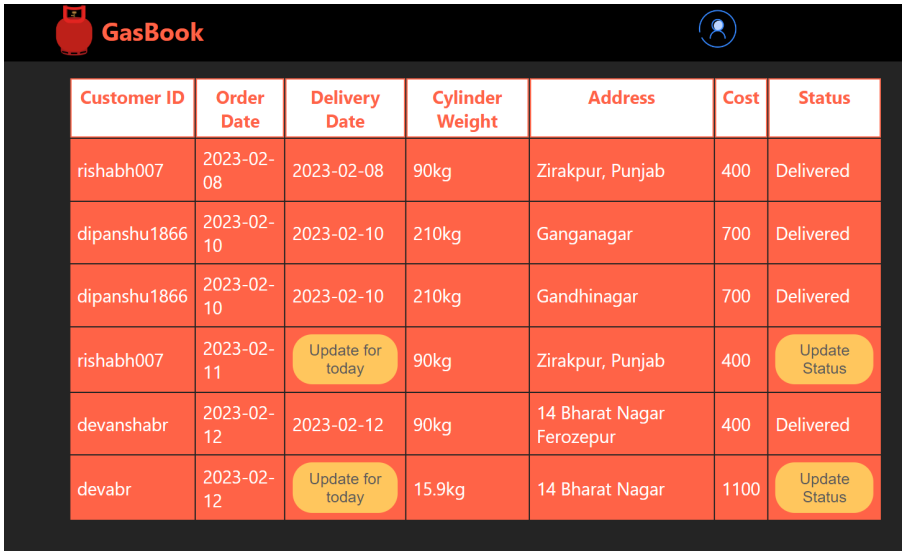
User Dashboard:

The user dashboard is almost the same as that of the user, with a few differences that the admin has no current orders placed, and they can view all feedback given by various users-



Monitor Orders:

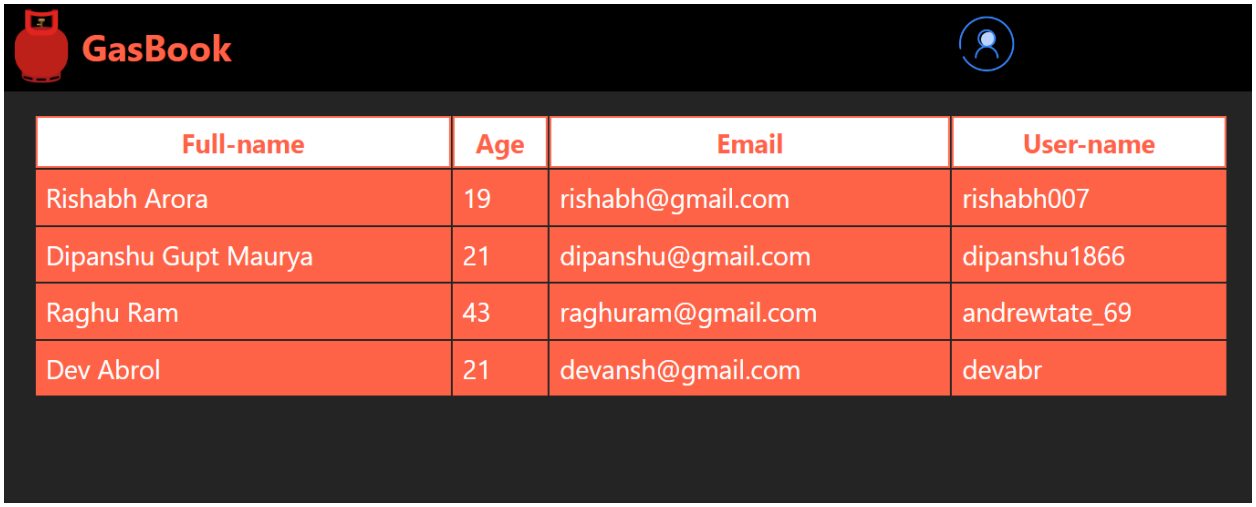
The admin can monitor orders, update their status to delivered and update the delivery date on the current day (which is assumed to be their delivery date)-



devabr	2023-02-12	2023-02-12	15.9kg	14 Bharat Nagar	1100	Delivered

View All Customers-

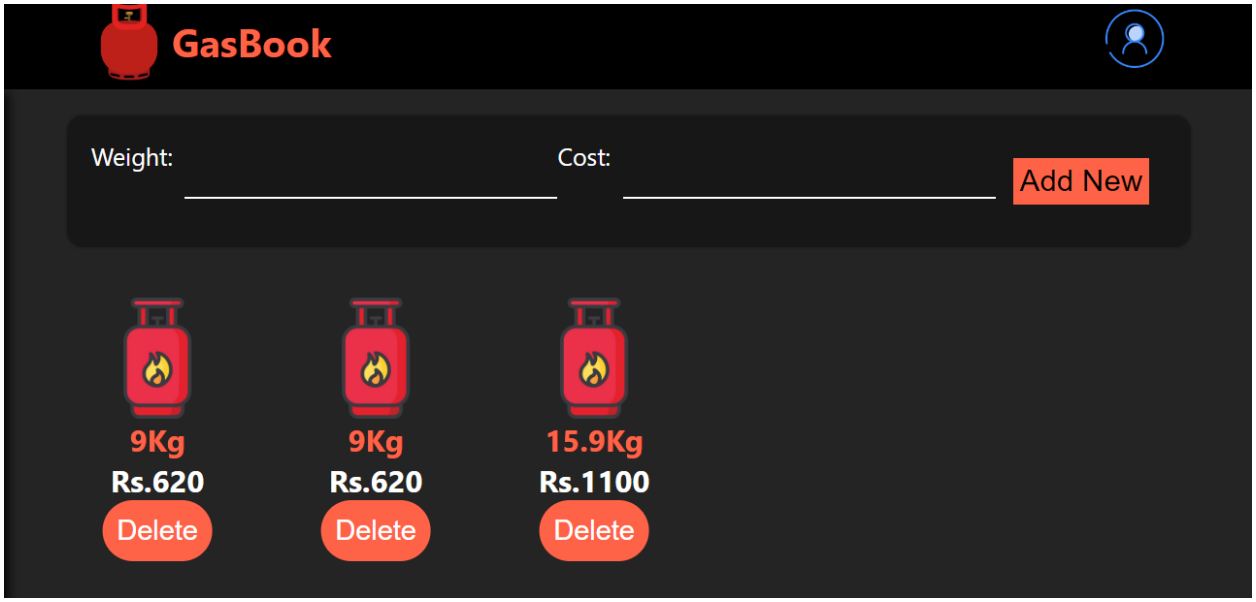
The admin is able to view all the customers and their data (except their passwords)-



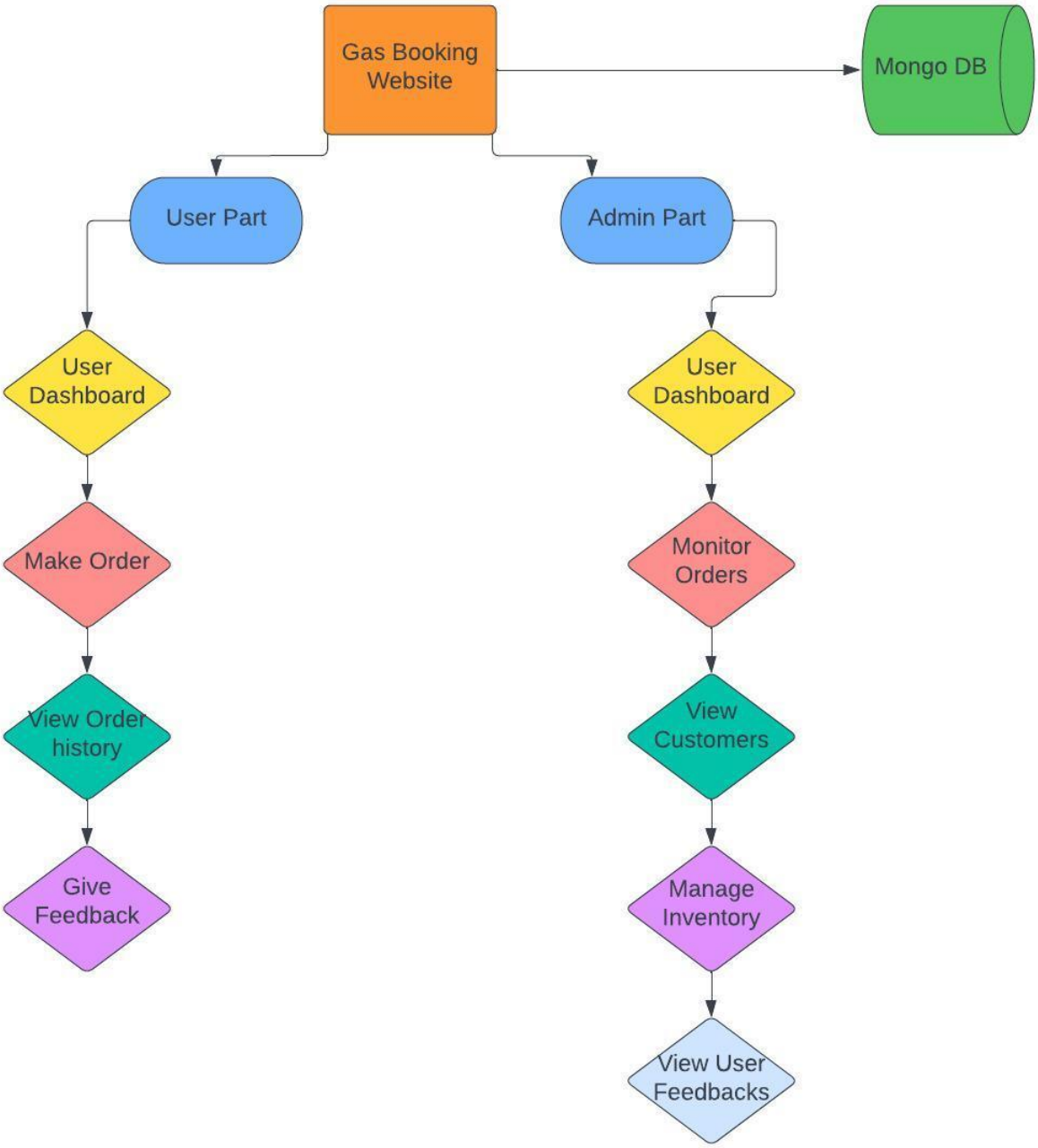
Full-name	Age	Email	User-name
Rishabh Arora	19	rishabh@gmail.com	rishabh007
Dipanshu Gupt Maurya	21	dipanshu@gmail.com	dipanshu1866
Raghu Ram	43	raghuram@gmail.com	andrewtate_69
Dev Abrol	21	devansh@gmail.com	devabr

Inventory-

The admin can update the inventory and view all the cylinders in inventory, they can manually update the number of cylinders by adding their weights and costs as per the orders monitored by them-



Flow Diagram:


Gas Booking Website
Mongo DB
User Part
Admin Part
User Dashboard
Make Order
View Order history
Give Feedback
User Dashboard
Monitor Orders
View Customers
Manage Inventory
View User Feedbacks

Conclusion:

The Gas Booking Web Project is a platform that will make the process of buying gas more convenient and efficient for users. The platform's user-friendly interface and various features will make it easy for users to book and purchase gas online. The platform will be built to the highest standards of security and performance and will be regularly maintained to ensure that it remains reliable and efficient.