

Project Report
Data Base Management System (CS-5493)

Domestic Gas Management System

December 1, 2022

By- KrishnaSrinivas Chilakamarri (gix161)

Saisandeep Vemulakonda(ion054)

Saivardhan Reddy Kunduru(dmm203)

And Maheba Patan (dqd242)

Instructor- Dr. Ritu Arora

Table of Content

Title page	1
Table of content	1
Introduction	2
About	2
Importance	2
Target audience	2
Design	3-5
Implementation	6
Testing	6
Challenges	7
How to Run	7
Contributions	7
References	8

Introduction

1. About

The “Domestic Gas Management System” has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce hardships faced by this existing system. Moreover this system is designed for the particular need of the company to carry out operations in a smooth and effective manner. We have developed a web application interface with the database of the Domestic gas management system in Countries like United States. Our implementation covers all the requirements specified in the plan. The database includes the list of Admins, customers, dealers, and commonly known services. In the web application, the user will be able to deal with the propane gas booking that they would like to get it delivered by the dealer and they can look up for the booking info and can do payment in online using their credit or debit cards. They can also follow up the booking and can be able to address their queries against their bookings by using token service. To get the service up and running, we have used data from public domains like Walmart, Costco, Home Depot, Lowes, HEB. We have used these public domain services which are existing dealers in our database to assist users in better understanding of booking and making better selections.

2. Importance

Every organization, whether big or small , has challenges to overcome and managing the information of Booking, Customer, Gas, Payment, Number. Every Online Gas Booking System has different Customer needs, therefore we design exclusive employee management systems that are adapted to your managerial requirements. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. Also, for those busy executive who are always on to go , our systems come with the remote access features, which will allow you to manage workforce anytime, at all times. These systems will ultimately allow you to better manage resources. The project focuses on to automate its existing manual system by the help of computerized equipments and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with the easy accessing and manipulation of the same. Basically the project describes how to manage for good performance and better services for the clients.

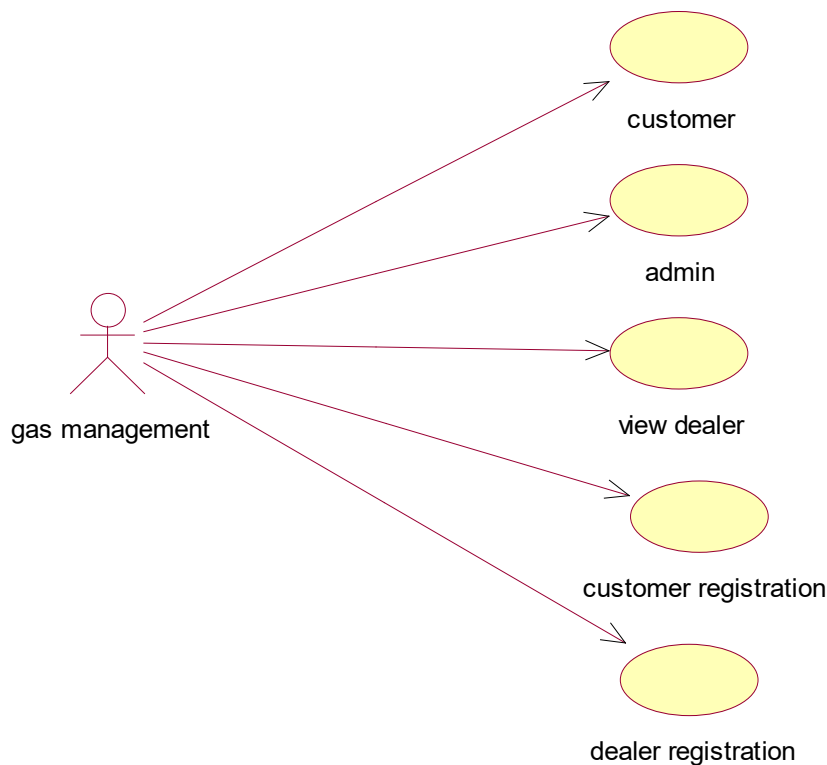
3. Target audience

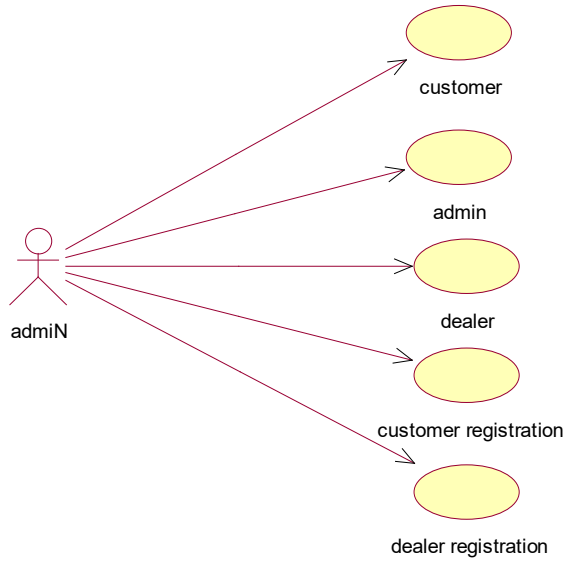
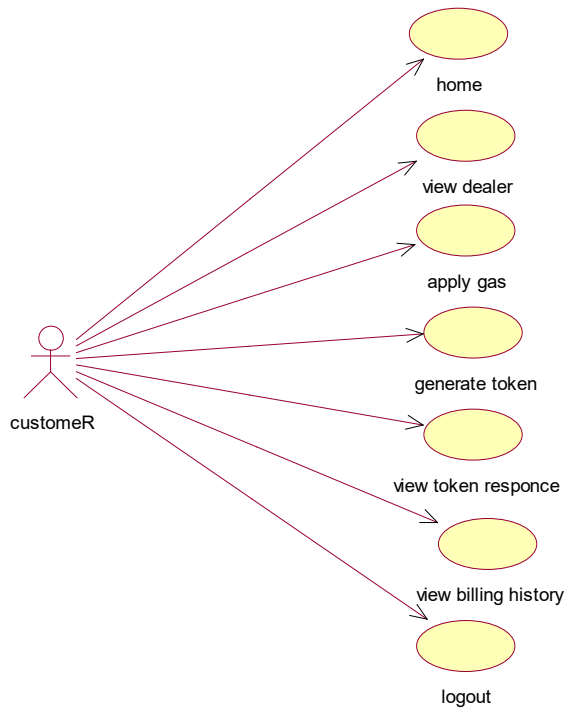
Our target audience are the people and the major business holders in the market who deals with

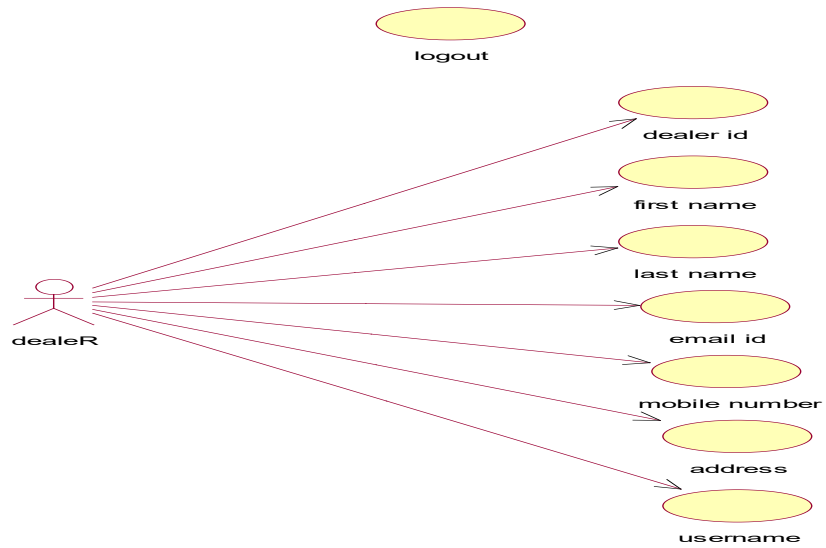
the supply chain in the entire industries. It will help a person to know the management of passed year perfectly and vividly. It also helps in the current all works relative to Gas Booking System. It will be also reduced the cost of collecting the management & collection procedure will go on smoothly

Design

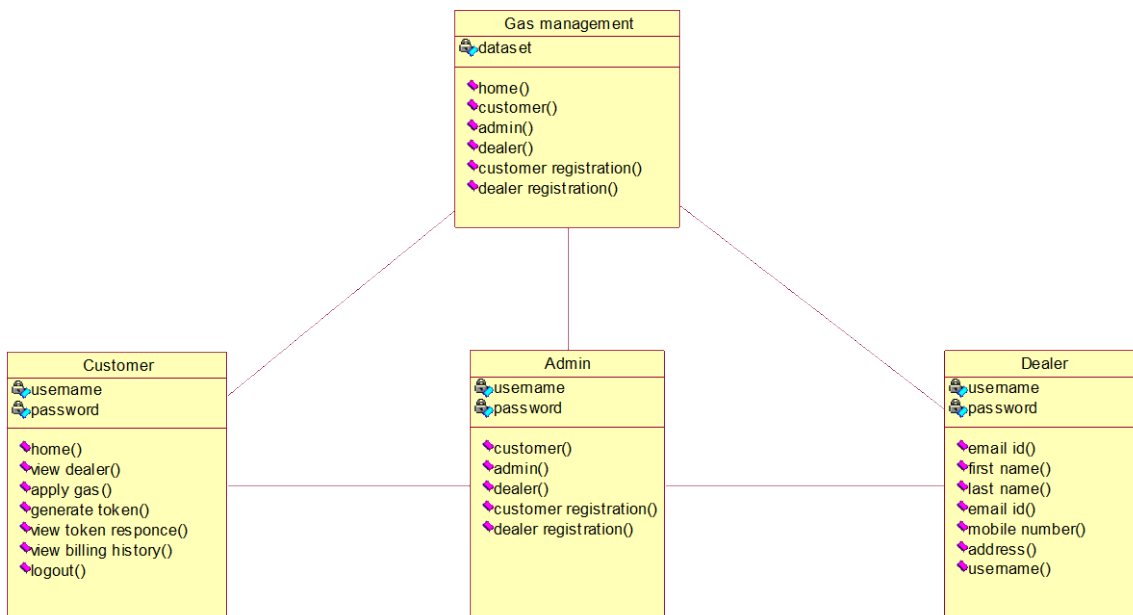
Below is the design architecture for our project. We create a User Interface (UI) which sends and HTTP request to the Backend Service where the backend receives HTTP requests from the browser. The HTTP headers or request body of the queries may contain data. The goal could be to request new data or to send data created by the user to the backend. HTTP requests are created and sent from within the user's browser. Each request receives a response, which includes information from the HTTP headers and the request content. Those responses are returned to the user's browser from the backend.







The class diagram for the implementation part is



Implementation

After making a successful project plan we implemented and made the project meet the objectives of our plan. To get the basic structure of the web application we used a standard web development framework(JavaScript) with a styling framework(css,html). React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library that allows you to create user interfaces using UI components. We have integrated the API endpoints of the backend service using Javascript .

Testing

Testing of the project is done by running Run Simulation. There are no broken links on our web pages and all links work properly. Forms are formatted to make them easier to read. HTML and CSS have been thoroughly tested to ensure that search engines can simply crawl your site. On all pages of your site, menus, buttons, and links to different pages are simply visible and consistent and there are no spelling or grammatical problems in the content.

Test queries are reliably sent to the database, the client-side output is presented correctly and All application requests are handled by the test Web server without any service denials also the database queries return the expected results. All fields such as Customer, Connection, Two Factor Authentication are validated and does not take invalid values.

When creating, updating, or deleting data in a database, data integrity is preserved and the online application appropriately displays test data retrieved from the database.

The online application looks well in all browsers, and JavaScript, and authentication are all operating properly and the web application is compatible with a wide range of operating systems, including Windows, Linux, and Mac, as well as browsers like Firefox, Internet Explorer, and Safari.

Performance

Response times for website applications at various connection speeds, as well as load testing online applications to determine their behavior under typical and peak loads. When pushed to beyond usual loads at peak times, a web application was stress tested to discover its breaking point. Application recovers from a crash caused by peak traffic, and optimization techniques like gzip compression, browser and server-side cache are used to improve load times.

Challenges

One of the most difficult issues we faced was rectifying the errors which are caught under testing and really worked hard to get rid of the errors and make the project well. We tried deploying the project many times and ended up with failures. We tried AWS for Docker and couldn't able to finish it as we are facing different type of issues like kernel version mismatch. Finally, after failing several times we successfully deployed the project but couldn't able to make it out the docker deployment. But project is working fine and we are successfully able to run the project with minimizing all the issues we faced. Only the UI Layers of the app are covered by JavaScript. So we still needed to pick some other technologies to complete the development toolkit for the project.

How to Run

The project can be accessed at: <https://utsa-food-market.herokuapp.com/>

For Admin user = [admin](#) and Password = [root](#).

You can run application DomesticGasManagementSystem directly from NetBeans IDE.

We have deployed the project in EC2 Instance from where you can access the web service by simply running the engine.

Contributions

Sandeep and Maheba-

- Proposed and validated the table schema/structure.
- Set up the database and create/managed tables.
- Designed and implemented the database.

Saivardhan and Krishna Srinivas-

- Planned and developed UI interface for home page, login, booking etc.
- Then explored user interface development solutions and decided on the technology.
- Also explored the backend service alternatives and decided on the technology.

Krishna Srinivas and Maheba-

- Implemented the application by gathering all individual developments.
- Helped deploy the application.

Saivardhan and Sandeep-

- Taking care of testing the application end- end.
- Tested and validated all the use-cases and user flow.

Krishna Srinivas, Saivardhan, Sandeep and Maheba-

- Completely worked on gathering information about docker implementation and use of cloud to deploy application in cloud.
- Worked on different resources like Nginx, Apache Tomcat etc., to check the possibility of application in cloud.

References

[1] Front end development

<https://www.w3schools.com/html/>

[1] Java documentation for development

<https://docs.oracle.com/javase/8/docs/>

[2] Concepts of UML Diagrams used in project

<https://www.lucidchart.com/blog/types-of-UML-diagrams>

[3] Docker implementation

https://docs.docker.com/develop/develop-images/dockerfile_best-practices/

[4] AWS cloud User Interface

<https://aws.amazon.com/console/>