**ON DEMAND CAR WASH SYSTEM**

**LOW LEVEL DEDIGN (LLD)**



DOCUMENT APPROVAL

**Approvers of this document**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Department** | **Role** | **Signature** | **Date** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Document Change History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Version #** | **Author** | **Date** | **Description** |
| 1.0 | Abhiroop Jana |  | On-Demand Car Wash System LLD |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

1.0 Document Purpose

2.0 Intended Audience

3.0 Project Background, Objective(s)

3.1 Project Background

3.2 Project Objective

4.0 Design Pattern

5.0 Solution Diagram

6.0 Solution Steps

7.0 Classes/function name

8.0 Validations

9.0 Data model/Tables

10.0 API Canvas

11.0 Env variables

12.0 Integrations

13.0 AWS Role

14.0 HTTP Status Code

15.0 Unit Testing

16.0 Request

17.0 Response

# Document Purpose

This document describes the solution architecture for an On-demand car wash system.

# Intended Audience

This document is intended as a reference for the following roles and stakeholders who are interested in the On-demand car wash system technical architecture.

|  |  |
| --- | --- |
| **Role** | **Nature of Engagement of Stakeholders** |
| Product Owners/SME | Key stakeholders to ensure that the architecture is aligned with business goals. |
| Business Analysts | Business analysts are one of the stakeholders who are informed of the key architectural decisions. |
| Enterprise Architects | To enforce Customer management Platform Architecture is aligned to business goals and architecture, and architectural guidelines. |
| Solution Architects | To ensure solution design and architecture are aligned to business requirements and architectural guidelines. |
| Developers | Use Technical Architecture Document as the guiding document for detail design and implantation approach to align with On-demand car wash system |

# Project Background, Objective(s)

## Project Background

On-demand car wash system leads to providing the best car wash services at the customer’s doorstep. With Green Wash, you can get your car washed wherever you are at your convenience. Whether it's your home, salon, office or any other place, your car wash is just a few taps away

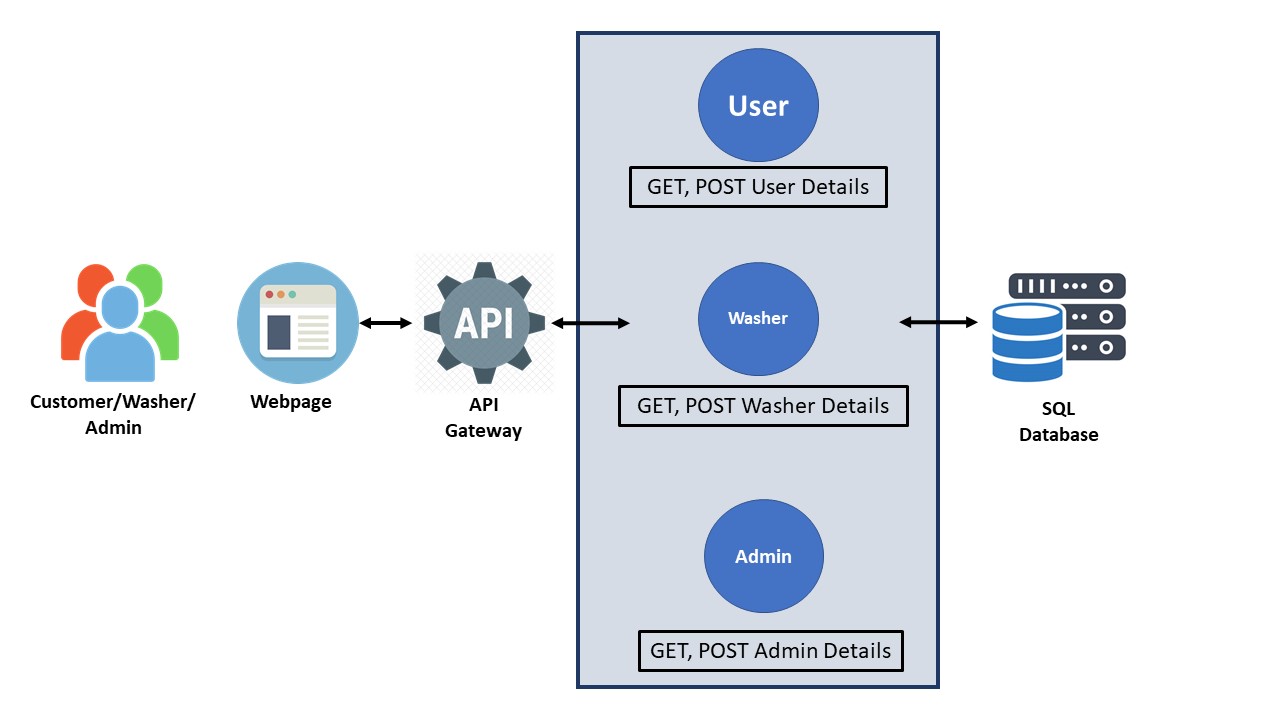
## Project Objective

## To develop an On-demand car wash system that will allow users to book a car wash at the specified time and location. The system is equipped with services like easy signup/in, editing their personal details, entering car details for further washes, selecting the date, time, and add-ons or package details of wash, one-tap booking, rating the wash services and many more.

# Design Pattern

|  |  |  |
| --- | --- | --- |
| # | Name | Description |
|  |  |  |

# Solution Diagram



# Solution Steps

**Customer Login/Signup**

1. User will enter his/her name and password to login if he/she is and existing user or else he/she will enter all the details like name, address,email,phone number etc.
2. As the user clicks the Login/Signup button, the browser directs the request to is NewCustomer().
3. API gateway does the routing and forwards the request to is NewCutomer().
4. The is NewCustomer() will validate if the user exists in the database or not
   1. If function returns true, user will be directed to the userRegistration() function.
   2. If function returns false, the getUserDetails API will be called and the user will be logged in and greeted with his username and index page of the website.
5. userRegistration() - User will enter the required details email, Phone number, car details, car image, and payment details to register for a service.
6. The entered details will be validated by the function validateProfile().
7. On Successful validation and registration user will have to login again either using his loging credentials.

**Wash Now/Schedule Later**

1. User will first have to login.
2. As the user enters the details and clicks the login button, browser directs the request to Customer List API.
3. After validating the user login credentials, the user will be redirected to the webpage’s index page.
4. As the user clicks on the wash button, the user will be redirected to the wash\_request view.
5. The user needs to select car model, wash package, add-ons and some special instruction message for the washer.
6. The user then needs to click either on wash now or wash later button based on the needs.
   1. If user chooses wash now option the user needs to set a location for wash.
   2. If user chooses wash later option the user needs to set a location, date and time for wash.
7. The user will be then redirected to the Payments View. The user will be needed to fill up the payment details and on successful payment the user will receive a payment receipt.

**User Profile View/ update**

1. User will first have to login by entering the login credentials.
2. As the user enters the details and clicks the login button, browser directs the request to Customer List API.
3. After validating the user login credentials, the user will be redirected to the webpage’s index page.
4. As the user clicks on the profile button, the user will be redirected to the profile page.
5. The user can view or edit his personal details such as Name, email, Phone number, car details, car image, and payment details.
6. When the user clicks on update profile, validateProfile() functions will be called for validating the entered details.
7. The user’s personal details will be then updated in the database and finally shown to the user.

**Washer Login**

1. Washer will enter his login Id and password.
2. As the washer clicks on the login button, the browser directs the request to the washer login API.
3. Call reaches the API gateway.
4. API gateway does the routing and forwards the request to washer\_validate() function.
   1. If the function returns true, the browser displays the dashboard for the washer.
   2. If the function returns false, the browser displays the error message to the washer.

**Admin Login**

1. Admin will enter his login Id and password.
2. As the washer clicks on the login button, the browser directs the request to the washer login API.
3. Call reaches the API gateway.
4. API gateway does the routing and forwards the request to admin\_validate() function.
   1. If the function returns true, the browser displays the dashboard for the admin.
   2. If the function returns false, the browser displays the error message to the admin.

# 7.0 Classes/function

# Validations

# Data model/Table

|  |  |  |
| --- | --- | --- |
| T\_Admin | | |
| PK | admin\_Id | INT |
|  | username | VARCHAR(100) |
|  | email | VARCHAR(100) |
|  | password | VARCHAR(100) |

|  |  |  |
| --- | --- | --- |
| T\_Customer | | |
| PK | customer\_Id | INT |
|  | name | VARCHAR(100) |
|  | email | VARCHAR(100) |
|  | phn\_number | VARCHAR(100) |
|  | password | VARCHAR(100) |
|  | carModel | VARCHAR(100) |
|  | customerAddress | VARCHAR(100) |
|  | payment\_type/mode | VARCHAR(100) |

|  |  |  |
| --- | --- | --- |
| T\_Washer | | |
| PK | washer\_Id | INT |
|  | name | VARCHAR(100) |
|  | email | VARCHAR(100) |
|  | phn\_number | VARCHAR(100) |
|  | password | VARCHAR(100) |
|  | washerRating | VARCHAR(100) |
|  | status | VARCHAR(100) |

|  |  |  |
| --- | --- | --- |
| T\_OrderDetails | | |
| PK | order\_Id | INT |
|  | order\_status | VARCHAR(100) |
| FK | customer\_Id | INT |
|  | customer\_name | VARCHAR(100) |
|  | customer\_address | VARCHAR(100) |
|  | customer\_location | VARCHAR(100) |
| FK | washer\_Id | INT |
|  | washer\_name | VARCHAR(100) |
| FK | package\_Id | INT |
|  | package\_name | VARCHAR(100) |
|  | addOn | VARCHAR(100) |
|  | promoCode | VARCHAR(100) |
|  | addOn \_cost | MONEY |
|  | promoCode\_price | MONEY |
|  | wash\_cost | MONEY |
|  | order\_Date/Time | DATETIME |

|  |  |  |
| --- | --- | --- |
| T\_Package/Service | | |
| PK | package\_Id | INT |
|  | Package\_name | VARCHAR(100) |
|  | description | VARCHAR(100) |
|  | status | VARCHAR(100) |

|  |  |  |
| --- | --- | --- |
| T\_PaymentDetails | | |
| PK | payment/txn\_Id | INT |
| FK | order\_Id | INT |
|  | payment\_amount | MONEY |
|  | payment\_type/mode | VARCHAR(100) |
|  | payment\_date | DATETIME |
|  | payment\_status | VARCHAR(100) |

|  |  |  |
| --- | --- | --- |
| T\_CarDetails | | |
| PK | car\_Id | INT |
|  | type | VARCHAR(100) |
|  | price | MONEY |
|  | status | VARCHAR(100) |

|  |  |  |
| --- | --- | --- |
| T\_Promocode | | |
| PK | promoCode\_Id | INT |
|  | promoCode\_name | VARCHAR(100) |
|  | description | VARCHAR(100) |
|  | price | MONEY |
|  | status | VARCHAR(100) |

|  |  |  |
| --- | --- | --- |
| T\_AddOn | | |
| PK | addOn\_Id | INT |
|  | addOn\_name | VARCHAR(100) |
|  | cost | MONEY |
|  | status | VARCHAR(100) |