



WOQOD

Mobile Application Architecture Document

(Backend & Backoffice)

Contents

1.	Document Control	4
1.1	Document Identification	4
1.2	Document Change History	4
2.	Document Overview	4
3.	Overall Architecture	4
3.1	Production	4
3.2	UAT	6
4.	Project Architecture	7
4.1	Project APIs Flow	7
4.2	Backend	8
4.2.1	Description	8
4.2.2	Project Structure	8
4.2.3	Dependencies used	9
4.3	Back-office	13
4.3.1	Description	13
4.3.2	Project Structure	13
4.3.3	Dependencies Used	14
5.	Configuration & maintenance of the system	17
5.1	Database configuration	17
5.2	Active Directory Integration	17
5.3	Payment Gateway	17
5.3.1	CyberSource payment	18
5.3.2	QPay payment configuration	19
5.3.3	Payment call-back URLs	20
5.3.4	In Progress Transaction handling	21
5.4	Swagger documentation	22
5.4.1	Description	22
5.4.2	Swagger configuration	22
5.5	Base URLs configuration	22
5.6	Azure File Storage	22
5.6.1	Description	23
5.6.2	Configuration	23
6.	Features	23
6.1	Login	23
6.2	Registration	24
6.3	Activate Account	25
6.4	Forget Password	25
6.5	Home	26
6.6	News	26
6.7	Profile	27
6.8	Reset Password	27

6.9	Supermarket	28
6.10	Shafaf Retailers	29
6.11	Bulk LPG	29
6.12	Fuel Prices	29
6.13	Promotions	29
6.14	Tenders	29
6.15	Feedback	30
6.16	Settings	30
6.17	Notification	31
6.18	Survey	32
6.19	WOQODe	32
6.19.1	WOQODe Top-up	32
6.19.2	WOQODe Transaction History	35
6.20	Fahes	35
6.20.1	Fahes & Petrol Stations	36
6.20.2	Fahes Add New Car	36
6.20.3	Fahes new inspection online payment	37
6.20.4	Fahes list of receipts	38
6.20.5	Fahes Inspection Report	38
6.20.6	Fahes Booking	39
6.20.7	Fahes Cancel/Modify Booking	39
7.	Integration points:	40
7.1	Fahes System	40
7.1.1	API Links	40
7.1.2	API Request & Response	42
7.2	RPS WOQODe System	54
7.2.1	Integration URLs	54
7.2.2	API Request & Response	55
7.3	MOTC System	64
7.3.1	Description	64
7.3.2	Integration URL	64
7.3.3	MOTC API Request & Response	6465
7.4	WOQOD API ("SharePoint")	66
7.4.1	Description	66
7.4.2	Integration URLs	66

1. Document Control

1.1 Document Identification

Title:	WOQOD Mobile Architecture Document
File Name:	WOQOD Mobile Architecture Document
Document Type:	Technical Specification Document

1.2 Document Change History

Date	Version	Description	Author
10.01.2023	V1.0	Initial version	Nada ACHOUR & Asma MACHERKI
27.01.2023	V2.0	Updated version	Nada ACHOUR & Asma MACHERKI

2. Document Overview

WOQOD mobile application is a convenient and user-friendly tool for customers of WOQOD, a leading fuel retail company in Qatar. The app allows users to purchase fuel and access other services on the go. Some of the key features of the app include:

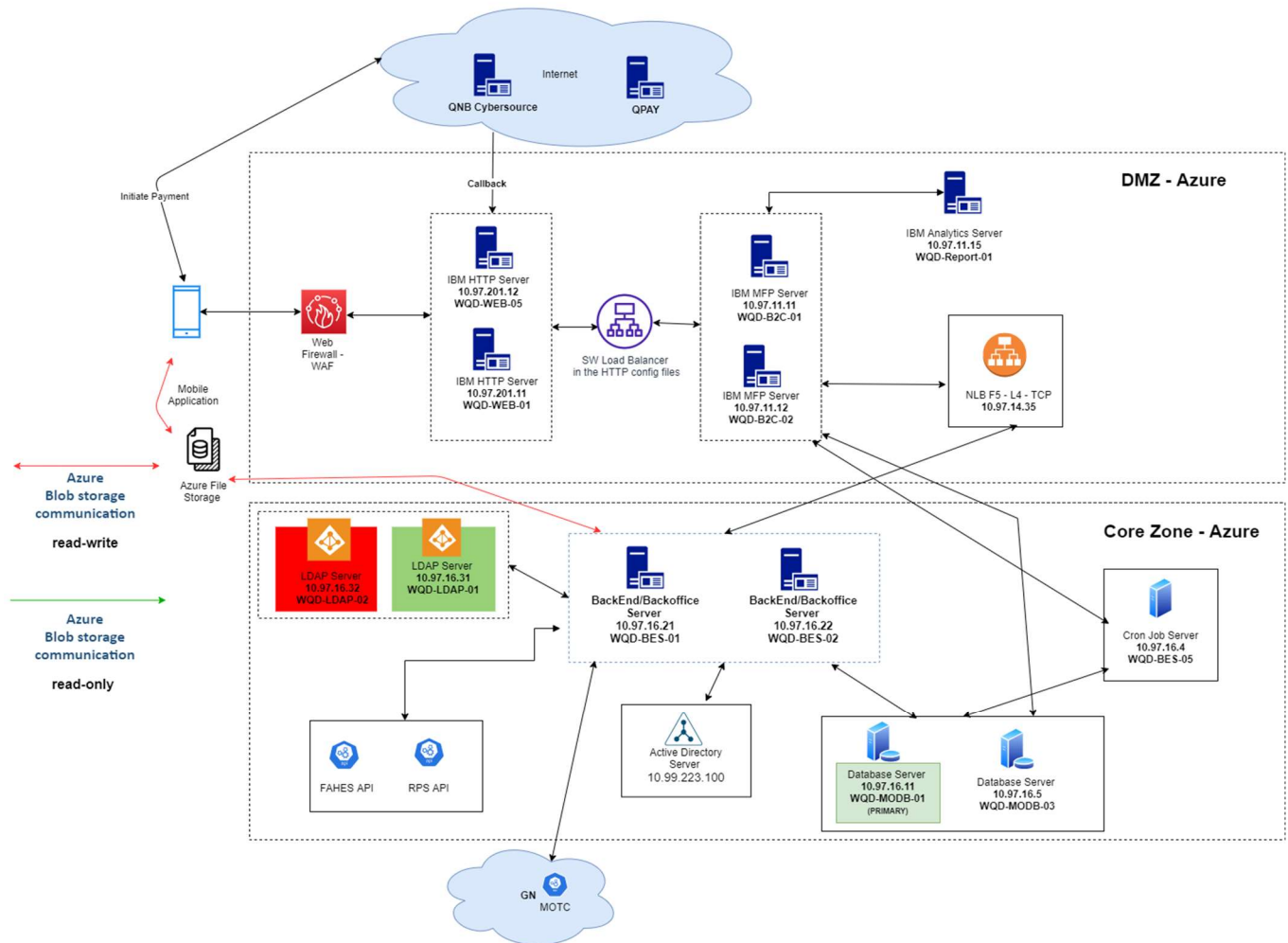
- Locate nearby WOQOD fuel stations: The app's map feature allows users to easily locate the nearest WOQOD fuel station, view the available services and get directions to the station.
- View fuel prices: Users can view current fuel prices and compare prices with other fuel stations in the area.
- Purchase fuel directly from the app: Users can purchase fuel directly from the app using their WOQOD card or other electronic payment options.
- Manage WOQOD card: Users can view their transaction history and manage their WOQOD card from the app.
- Real-time notifications: The app provides real-time notifications to inform users of any updates or offers.

This document serves as an input for the development team to introduce WOQOD project configurations & maintenance.

Also, it presents a functional guide for the communication between APIs and MFP.

3. Overall Architecture

3.1 Production

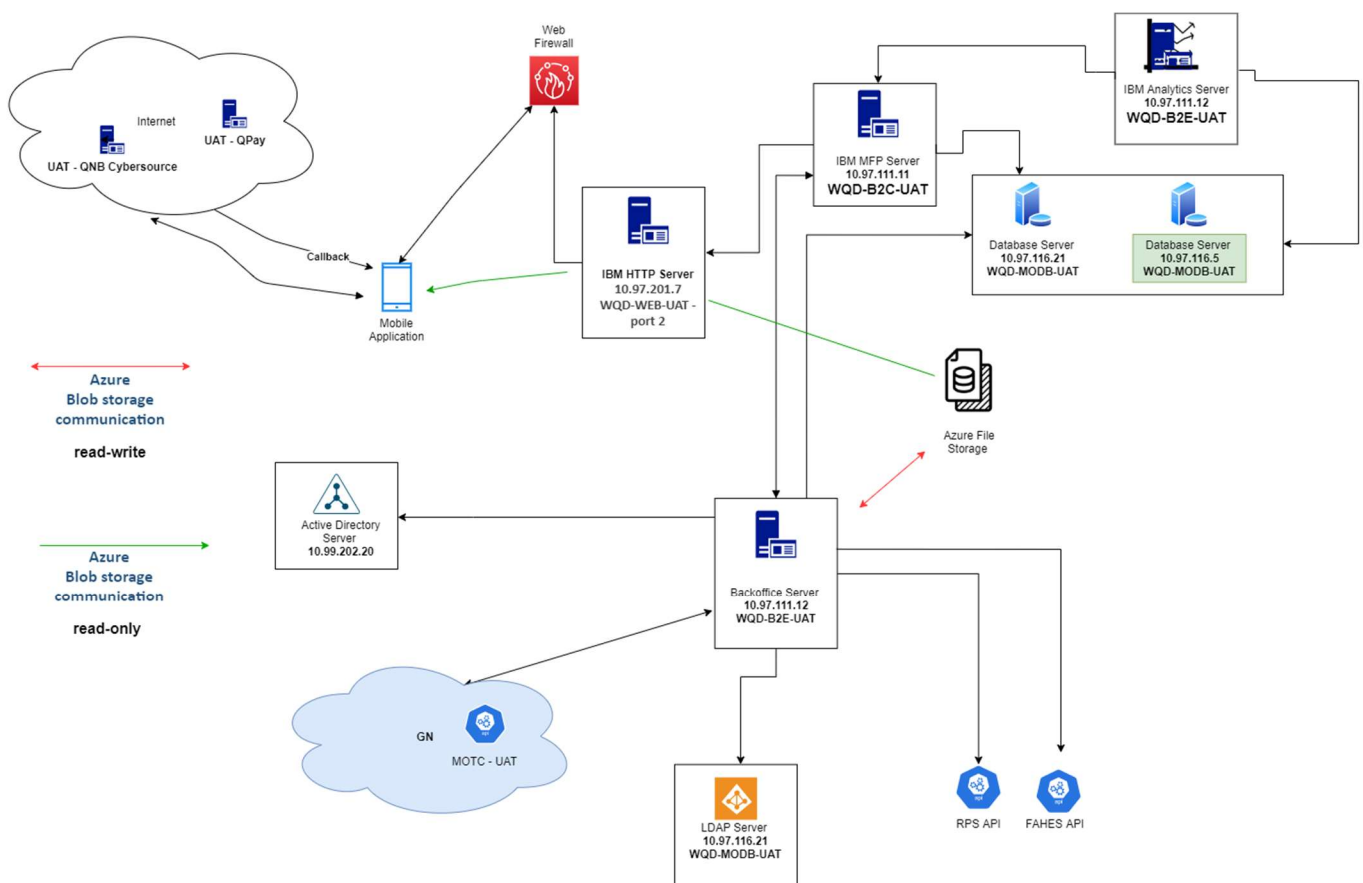


Note: This architecture diagram is verified on 30 January 2023.

Name	Role	Address
Web Firewall	Control incoming and outgoing network traffic	-
IBM HTTP Server	Control and secure HTTP requests	10.97.201.11 & 10.97.201.12
IBM MFP Server	<ul style="list-style-type: none"> - Deploy adapters - Give access to other production backend systems and securing APIs through adapters. - Send notifications to devices 	10.97.11.11 & 10.97.11.12
BackOffice Server	<ul style="list-style-type: none"> - Used for backend and back-office deployment - Ensure that all company data and information are secure and maintained 	10.97.16.21 & 10.97.16.22
IBM Analytics Server	Used for big data storage and analytics management	10.97.11.15
LDAP Server	Save and get mobile app users for production environment	10.97.16.31 & 10.97.16.32
Active Directory Server	Save and get Back-office users	10.99.202.20
Database Server	Save all data	10.97.16.11

Cron Job Server	Dedicated to Fahes and WOQOD payment inprogress cron job	10.97.16.4
Cron Job Server	Dedicated to WOQOD SharePoint cron job	10.97.16.21
Azure File Storage	Store application unstructured data (media files used in the application including mobile app content medial files	https://wqdstoragewebprd.blob.core.windows.net/wqdstoragewebprdcontent
MOTC	Ministry of Transport and Communications-Head Quarter	https://stgapi.gdx.gov.qa:9443/gdx/prod/v1/CustomerValidationWS/validateCustomer
CyberSource	Gateway to process the payment with credit card and master cards	https://secureacceptance.cybersource.com/pay
QPay	Gateway to process the payment with debit card	https://www.qpay.gov.qa/QPayOnePC/PaymentPayServlet

3.2 UAT



Name	Role	Address
Web Firewall	Control incoming and outgoing network traffic	-
IBM HTTP Server	Control and secure HTTP requests	10.97.201.12
IBM MFP Server	-Deploy adapters	10.97.111.11

	<ul style="list-style-type: none"> -Give access to other UAT backend systems and securing APIs through adapters. - Send notifications to devices 	
BackOffice Server	<ul style="list-style-type: none"> - Used for backend and back-office deployment - Ensure that all company data and information are secure and maintained 	10.97.111.12
IBM Analytics Server	Used for big data storage and analytics management	10.97.111.12
LDAP Server	Save and get mobile app users for UAT environment	10.97.116.21
Active Directory Server	Save and get Back-office users Active Directory integration details: ldap.base=OU=MobileApp,DC=Woqod,DC=com ldap.url=ldap://10.99.202.20:389 ldap.userSearchBase=OU=MobileApp,DC=Woqod,DC=com	10.99.202.20
Database Server	Save all data	10.97.116.21
Azure File Storage	Store application unstructured data	https://wqdstoragewebuat.blob.core.windows.net/wqdstoragewebuatcont
MOTC	Ministry of Transport and Communications-Head Quarter	https://stgapi.gdx.gov.qa:9443/gdx/prod/v1/CustomerValidationWS/validateCustomer
Cyber Source	Gateway to process the payment with credit card & master card	https://testsecureacceptance.cybersource.com/pay
QPay	Gateway to process the payment with debit card	https://pgtest3.qcb.gov.qa/QPayOnePC/PaymentPayServlet

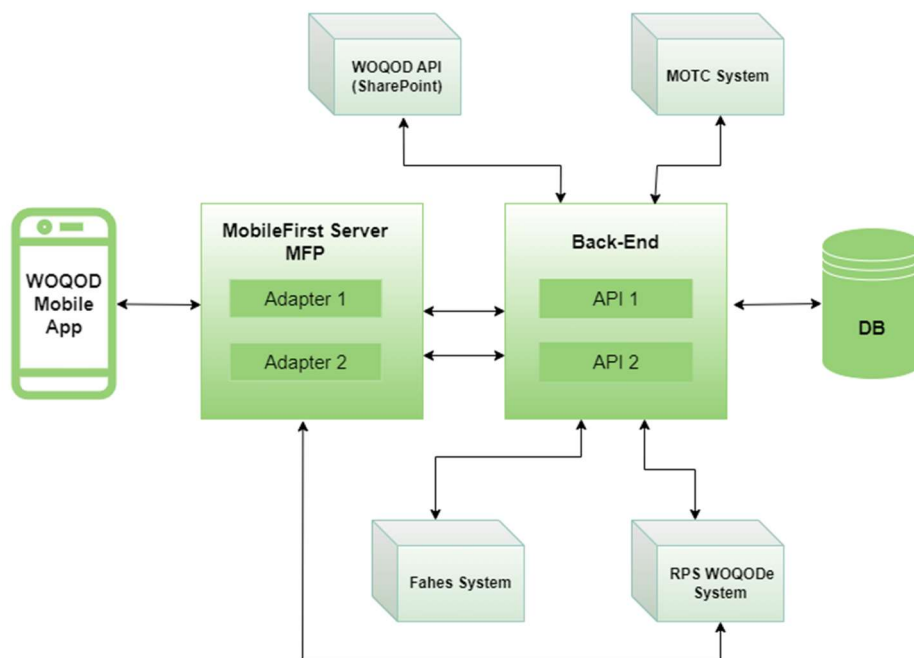
4. Project Architecture

4.1 Project APIs Flow

WOQOD mobile application communicates with backend servers through APIs. The mobile-first API(**MFP**) server serves as a bridge between the mobile application and the backend server, allowing the mobile application to send and receive data from the backend server through **REST API** calls. The data exchanged between the mobile application and the server is in **JSON** format.

The Backend server communicates with WOQOD external systems (Fahes system, WOQODE system, MOTC system and WOQOD SharePoint system) via **REST/SOAP API** calls.

WOQOD Mobile APIs Flow



4.2 Backend

4.2.1 Description

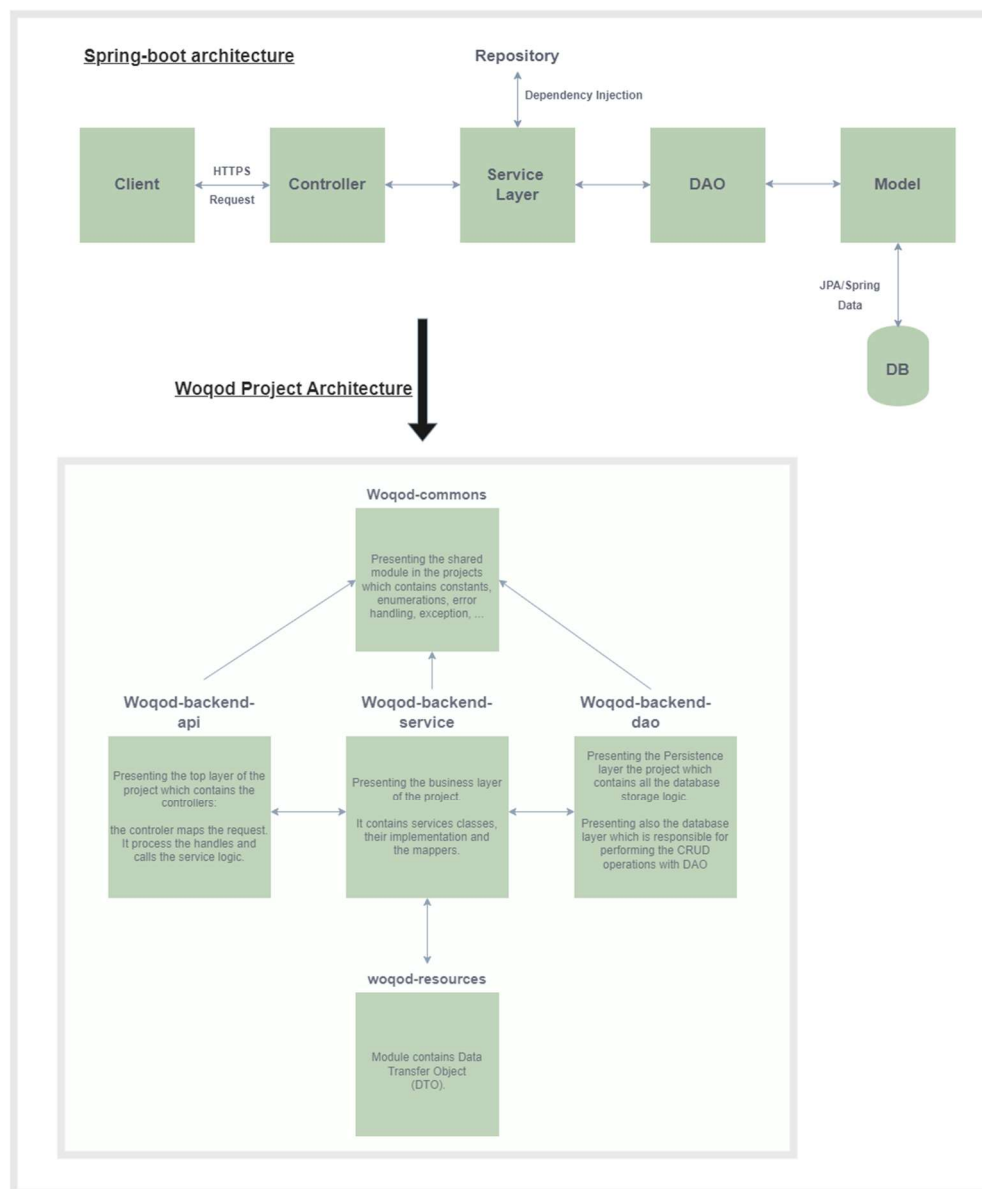
In WOQOD project, we built an application based on **a modular design and MVC pattern**, developed with **Java 8** as a programming language and **Spring boot** as a framework.

At the core of Spring Boot is the **Spring Application Context**, which is responsible for managing the lifecycle of all the beans in the application. These beans are created and configured through configuration files. This application also provides built-in features, such as data access, security (added to a project through the use of "starters"), etc.

4.2.2 Project Structure

In the project description we talked about the modularization of the project and the clean architecture approach, as we can see in the diagram below there is a:

- **Shared module 'woqod-commons'**: which is a module that contains all the resources and classes that are needed across all the app features, such as exception/error handling classes, and other util classes (LapUtils, FileManagmentUtils, PinCodeGeneratorUtils, etc.)
- **API layer 'woqod-backend-api'**: responsible for exposing the application's functionalities/features as a RESTful API. The API layer receives HTTP requests from client applications, processes the requests, and returns HTTP responses to the clients.
- **Service layer 'woqod-backend-service'**: the business layer of the project, it contains service classes, their implementation and mappers.
- **DAO layer 'woqod-backend-dao'**: stands for Data Access Object, and refers to a component that is responsible for accessing and manipulating data stored in the database. It is implemented as a Repository, and used to perform database operations, such as retrieving, creating, updating, and deleting data. The DAO layer interacts with the database using the persistence technology: **JPA (Java Persistence API) and hibernate**.
- **Resources layer: 'woqod-ressources'**: represents a specific resource in an application. For example, a resource can represent a user, a survey, and can be accessed and manipulated through RESTful APIs.



4.2.3 Dependencies used

Dependency	Description	URL Documentation	Version
Spring Boot Starter Parent	Parent pom providing dependency and plugin management for applications built with Maven	https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-starter-parent	2.7.6
Spring Security Core	Spring Security is a powerful and highly customizable authentication and access-control framework. It provides protection against attacks like session fixation,	https://mvnrepository.com/artifact/org.springframework.security/spring-security-core	5.7.5

	clickjacking, cross site request forgery.		
Querydsl Core Module	core module for querydsl	https://mvnrepository.com/artifact/com.querydsl/querydsl-core	5.0.0
SureFire JUnitCore Runner	SureFire JUnitCore (JUnit 4.7+) Runner	https://mvnrepository.com/artifact/org.apache.maven.surefire/surefire-junit47	2.19.1
Ojdbc8	Oracle JDBC Driver compatible with JDK8, JDK9, and JDK11	https://mvnrepository.com/artifact/com.oracle.ojdbc/ojdbc8	19.3.0.0
SpringFox Swagger2	JSON API documentation for spring-based applications	https://mvnrepository.com/artifact/io.springfox/springfox-swagger2	2.9.2
SpringFox Swagger UI	JSON API documentation for spring-based applications	https://mvnrepository.com/artifact/io.springfox/springfox-swagger-ui	2.9.2
Apache Log4j Core	Implementation for Apache Log4J, a highly configurable logging tool that focuses on performance and low garbage generation. It has a plugin architecture that makes it extensible and supports asynchronous logging based on LMAX Disruptor	https://mvnrepository.com/artifact/org.apache.logging.log4j/log4j-core	2.17.2
Apache Log4j API	API for Apache Log4J, a highly configurable logging tool that focuses on performance and low garbage generation. It has a plugin architecture that makes it extensible and supports asynchronous logging based on LMAX Disruptor	https://mvnrepository.com/artifact/org.apache.logging.log4j/log4j-api	2.17.2
Querydsl JPA Support	JPA support for Querydsl	https://mvnrepository.com/artifact/com.querydsl/querydsl-jpa	5.0.0
Querydsl APT Support	APT based Source code generation for Querydsl	https://mvnrepository.com/artifact/com.querydsl/querydsl-apt	5.0.0
Unirest Java	Simplified, lightweight HTTP client library	https://mvnrepository.com/artifact/com.mashape.unirest/unirest-java	1.4.9
Apache Commons Lang	Apache Commons Lang, a package of Java utility classes for the classes that are in java.lang's hierarchy, or are considered to be so standard a	https://mvnrepository.com/artifact/org.apache.commons/commons-lang3	3.12.0

	s to justify existence in java.lang.		
Apache Commons Collections	The Apache Commons Collections package contains types that extend and augment the Java Collections Framework.	https://mvnrepository.com/artifact/org.apache.commons/commons-collections4	4.4
CyberSource SDK Java	Simple Order API Client	https://mvnrepository.com/artifact/com.cybersource/cyber-source-sdk-java	6.2.11
Jackson Modules: Java 8	Parent pom for Jackson modules needed to support Java 8 features and types	https://mvnrepository.com/artifact/com.fasterxml.jackson.module/jackson-modules-java8	2.13.4
CyberSource REST Client Java	Cybersource Rest Client SDK	https://mvnrepository.com/artifact/com.cybersource/cyber-source-rest-client-java	0.0.40
ModelMapper	ModelMapper	https://mvnrepository.com/artifact/org.modelmapper/modelmapper	2.3.6
JTDS	JTDS is an open source 100% pure Java (type 4) JDBC 3.0 driver for Microsoft SQL Server (6.5, 7, 2000, 2005, 2008, 2012) and Sybase ASE (10, 11, 12, 15). JTDS is based on FreeTDS and is currently the fastest production-ready JDBC driver for SQL Server and Sybase. JTDS is 100% JDBC 3.0 compatible, supporting forward-only and scrollable/updateable ResultSets and implementing all the Database Metadata and ResultSetMetadata methods.	https://mvnrepository.com/artifact/net.sourceforge.jtds/jtds	1.3.1
Apache POI API Based On OPC and OOXML Schemas	Apache POI - Java API To Access Microsoft Format Files	https://mvnrepository.com/artifact/org.apache.poi/poi-ooxml	3.17
Spring Boot Starter Mail	Starter for using Java Mail and Spring Framework's email sending support	https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-starter-mail	2.7.6
Spring WS Support	Spring WS Support	https://mvnrepository.com/artifact/org.springframework.ws/spring-ws-support	3.1.4
Microsoft Azure Storage Client SDK	SDK for Microsoft Azure Storage Clients	https://mvnrepository.com/artifact/com.microsoft.azure/azure-storage	8.4.0

Azure Spring Boot Starter For Azure Storage	Spring Boot Starter for Azure Storage	https://mvnrepository.com/artifact/com.azure.spring/azure-spring-boot-starter-storage	3.8.0
iText 7 Kernel	iText 7 Kernel	https://mvnrepository.com/artifact/com.itextpdf/kernel	7.1.12
pdfHTML	pdfHTML is an iText 7 add-on that lets you to parse (X)HTML snippets and the associated CSS and converts them to PDF.	https://mvnrepository.com/artifact/com.itextpdf/html2pdf	3.0.1
JSch	JSch is a pure Java implementation of SSH2	https://mvnrepository.com/artifact/com.jcraft/jsch	0.1.46
iText Core	A Free Java-PDF library	https://mvnrepository.com/artifact/com.itextpdf/itextpdf	5.5.13
Unirest Java	Simplified, lightweight HTTP client library	https://mvnrepository.com/artifact/com.mashape/unirest-java	1.4.9
Apache PDFBox	The Apache PDFBox library is an open-source Java tool for working with PDF documents.	https://mvnrepository.com/artifact/org.apache.pdfbox/pdfbox	2.0.4
JasperReports Library	Free Java Reporting Library	https://mvnrepository.com/artifact/net.sf.jasperreports/jasperreports	6.0.0
ZXing Core	Core barcode encoding/decoding library	https://mvnrepository.com/artifact/com.google.zxing/core	3.3.0
ZXing Java SE Extensions	Java SE-specific extensions to core ZXing library	https://mvnrepository.com/artifact/com.google.zxing/javase	3.3.0
Tomcat Embed Core	Core Tomcat implementation	https://mvnrepository.com/artifact/org.apache.tomcat.embed/tomcat-embed-core	8.5.43
Apache HttpClient	HttpClient provides an efficient, up-to-date, and feature-rich package implementing the client side of the most recent HTTP standards and recommendations.	https://mvnrepository.com/artifact/org.apache.httpcomponents/httpclient	4.5.13
Spring WS Security	Spring WS Security	https://mvnrepository.com/artifact/org.springframework.ws/spring-ws-security	3.1.4
XMLSec	XMLSec	https://mvnrepository.com/artifact/org.wso2.santuario/xmlsec	2.0.8

4.3 Back-office

4.3.1 Description

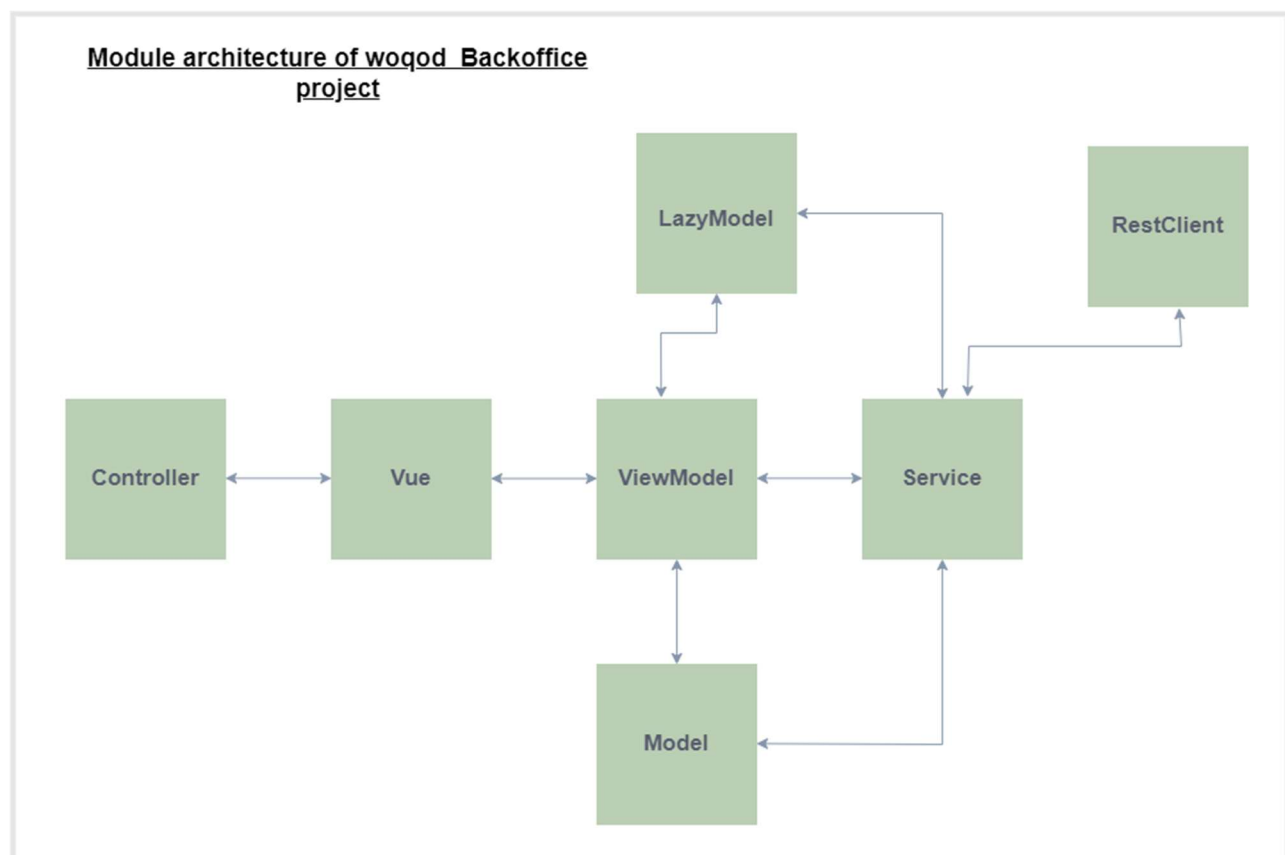
In WOQOD Backoffice project, we built a Spring boot application based on a **modular design and MVC pattern**.

4.3.2 Project Structure

Each feature (Fahes, WOQODe Top-up ...) in the BO have its own module: modules cannot communicate between each other and there is only one module that all the other module can see which is the "shared" module(woqod-backoffice-commons). Each module (Feature) has seven main layers (Controller, View, Vue Model, Lazy Model, Model, Service, Rest Client):

- **Model:** this module represents the data of the application, it contains POJOs and JPA entities used to carry data between the Controller and the View.
- **Controller:** this module is responsible for handling incoming HTTP requests, processing them and returning appropriate responses.
- **View:** responsible for presenting the data from the Model to the user using **JSF as a view technology**.
- **View Model** referees the data that is passed from the Controller to the View for rendering.
For example, if you have a REST endpoint that returns data about a User, the user data could be stored in a View Model class, and then passed to the View for rendering in an HTML template.
- **Lazy Model:** Lazy Initialization, by default in Spring, all the defined beans, and their dependencies, are created when the application context is created. In contrast, when we configure a bean with lazy initialization, the bean will only be created, and its dependencies injected, once they're needed.
- **Service:** component that implements the business logic of an application.
- **Rest Client:** Rest Template is used to create applications that consume RESTful Web Services.

This diagram shows the flow of the communication data between the different layers:



4.3.3 Dependencies Used

Dependency	Description	URL Documentation	Version
Spring Boot Starter Parent	Parent pom providing dependency and plugin management for applications built with Maven	https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-starter-parent	2.7.6
Spring LDAP	Spring LDAP is a Java library for simplifying LDAP operations, based on the pattern of Spring's JdbcTemplate. The framework relieves the user of common chores, such as looking up and closing contexts, looping through results, encoding/decoding values and filters, and more. The LdapTemplate class encapsulates all the plumbing work involved in traditional LDAP programming, such as creating a DirContext, looping through NamingEnumerations, handling exceptions and cleaning up resources.	https://mvnrepository.com/artifact/org.springframework.ldap/spring-ldap	2.4.1
PrimeFaces	PrimeFaces	https://mvnrepository.com/artifact/org.primefaces/primefaces	8.0
All Themes	Aggregate PrimeFaces themes and build "all-in-one" theme JAR.	https://mvnrepository.com/artifact/org.primefaces.s.themes/all-themes	1.0.10
PrimeFaces Extensions	PrimeFaces Extensions components and utilities for PrimeFaces.	https://mvnrepository.com/artifact/org.primefaces.s.extensions/primefaces-extensions	8.0
Hibernate Validator Engine Relocation Artifact	Hibernate Validator Engine Relocation Artifact	https://mvnrepository.com/artifact/org.hibernate/hibernate-validator	6.5.2.Final
Spring Boot Starter AOP	Starter for aspect-oriented programming with Spring AOP and AspectJ	https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-starter-aop	2.7.6
Spring Data Core	Spring Data makes it easy to use data access technologies, relational and non-relational databases, map-reduce frameworks, and cloud-based data services.	https://mvnrepository.com/artifact/org.springframework.data/spring-data-commons	2.7.6
Spring Boot Starter Data JPA	Starter for using Spring Data JPA with Hibernate	https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-starter-data-jpa	2.7.6

Spring Security Core	Spring Security is a powerful and highly customizable authentication and access-control framework. It provides protection against attacks like session fixation, clickjacking, cross site request forgery	https://mvnrepository.com/artifact/org.springframework.security/spring-security-core	5.7.5
Spring Boot Starter Security	Starter for using Spring Security	https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-starter-security	2.7.6
Spring Security LDAP	Spring Security	https://mvnrepository.com/artifact/org.springframework.security/spring-security-ldap	5.8.0
Spring HATEOAS	Library to support implementing representations for hyper-text driven REST web services.	https://mvnrepository.com/artifact/org.springframework.hateoas/spring-hateoas	1.5.2
AspectJ Weaver	The AspectJ weaver applies aspects to Java classes. It can be used as a Java agent in order to apply load-time weaving (LTW) during class-loading and also contains the AspectJ runtime classes.	https://mvnrepository.com/artifact/org.aspectj/aspectjweaver	1.9.7
Spring Cloud Starter OpenFeign	Spring Cloud Starter OpenFeign	https://mvnrepository.com/artifact/org.springframework.cloud/spring-cloud-starter-openfeign	3.1.4
PrimeFaces Spring Boot Starter	PrimeFaces Spring Boot Starter	https://mvnrepository.com/artifact/org.joinfaces/primefaces-spring-boot-starter	4.7.6
JSTL	JSTL	https://mvnrepository.com/artifact/javax.servlet/jstl	1.2
JUnit Jupiter API	JUnit Jupiter is the API for writing tests using JUnit 5.	https://mvnrepository.com/artifact/org.junit.jupiter/junit-jupiter-api	5.1.0
JasperReports Library	Free Java Reporting Library	https://mvnrepository.com/artifact/net.sf.jasperreports/jasperreports	6.15.0
JasperReports Font Extension	JasperReports Font Extension	https://mvnrepository.com/artifact/net.sf.jasperreports/jasperreports-fonts	6.15.0
Jackson Datatype: JSR310	Add-on module to support JSR-310 (Java 8 Date & Time API) data types.	https://mvnrepository.com/artifact/com.fasterxml.jackson.datatype/jackson-datatype-jsr310	2.13.4
Apache Commons IO	The Apache Commons IO library contains utility classes, stream implementations, file filters, file comparators, endian transformation classes, and much more.	https://mvnrepository.com/artifact/commons-io/commons-io	2.6
ModelMapper	ModelMapper	https://mvnrepository.com/artifact/org.modelmapper/modelmapper	2.3.6
Apache POI Common	Apache POI - Java API To Access Microsoft Format Files	https://mvnrepository.com/artifact/org.apache.poi/poi	5.2.3

Apache POI API Based on OPC and OOXML Schemas	Apache POI - Java API To Access Microsoft Format Files	https://mvnrepository.com/artifact/org.apache.poi/poi-ooxml	5.2.3
iText, A Free Java PDF Library	iText, a free Java-PDF library	https://mvnrepository.com/artifact/com.lowagie/itext	2.1.7
Apache HttpClient	HttpClient provides an efficient, up-to-date, and feature-rich package implementing the client side of the most recent HTTP standards and recommendations.	https://mvnrepository.com/artifact/org.apache.httpcomponents/httpclient	4.5.13
Apache Commons IO	The Apache Commons IO library contains utility classes, stream implementations, file filters, file comparators, endian transformation classes, and much more.	https://mvnrepository.com/artifact/commons-io/commons-io	2.11.0
Jsoup Java HTML Parser	jsoup is a Java library for working with real-world HTML. It provides a very convenient API for fetching URLs and extracting and manipulating data, using the best of HTML5 DOM methods and CSS selectors. jsoup implements the WHATWG HTML5 specification, and parses HTML to the same DOM as modern browsers do.	https://mvnrepository.com/artifact/org.jsoup/jsoup	1.15.3
Project Lombok	Spice up your java: Automatic Resource Management, automatic generation of getters, setters, equals, hashCode and toString, and more!	https://mvnrepository.com/artifact/org.projectlombok/lombok	1.18.24
Xerces2 J	Xerces2 provides high performance, fully compliant XML parsers in the Apache Xerces family. This new version of Xerces continues to build upon the Xerces Native Interface (XNI), a complete framework for building parser components and configurations that is extremely modular and easy to program. The Apache Xerces2 parser is the reference implementation of XNI but other parser components, configurations, and parsers can be written using the Xerces Native Interface.	https://mvnrepository.com/artifact/xerces/xercesImpl	2.12.2
Spring Web MVC	Spring webmvc contains Spring's model-view-controller (MVC) and REST Web Services implementation for web applications. It provides a clean separ	https://mvnrepository.com/artifact/org.springframework/spring-webmvc	5.3.2

	ation between domain model code and web forms and integrates with all of the other features of the Spring Framework.		
Apache Commons Validator	Apache Commons Validator provides the building blocks for both client-side validation and server-side data validation. It may be used standalone or with a framework like Struts.	https://mvnrepository.com/artifact/commons-validator/commons-validator	1.7

5. Configuration & maintenance of the system

5.1 Database configuration

❖ DEV:

- **URL:** `jdbc:oracle:thin:@172.16.20.57:1521:ORCL`
- **Username:** `woqod_dev`
- **Password:** `woqod_dev`

❖ UAT:

- **URL:** `jdbc:oracle:thin:@10.97.116.5:1521/mbldbua`
- **Username:** `APPDATA0311`
- **Password:** `APPDATA`

❖ Preprod:

- **URL:** `jdbc:oracle:thin:@10.97.116.21:1521/mbldbua`
- **Username:** `PREPRODAPPDATA`
- **Password:** `appdata123`

❖ PROD:

- **URL:** `jdbc:oracle:thin:@10.97.16.11:1521/mbldbprod`
- **Username:** `APPDATA`
- **Password:** `APPDATA`

5.2 Active Directory Integration

❖ Description

Active Directory (AD) is a Microsoft directory service used for storing and organizing information about resources in a network, such as users. AD is designed to provide centralized authentication and authorization for Windows-based networks and provides a range of services, including user account management, policy enforcement, and security administration.

❖ Configuration

```
ldap.base=OU=MobileApp,DC=Woqod,DC=com
ldap.url=ldap://10.99.202.20:389
ldap.userSearchBase=OU=MobileApp,DC=Woqod,DC=com
```

5.3 Payment Gateway

5.3.1 CyberSource payment

5.3.1.1 Description

CyberSource is a payment gateway that allows merchants to process credit card securely. Integrating CyberSource into a web application involves configuring the application to communicate with the CyberSource payment gateway and handle the various steps of the payment process.

Steps for integrating CyberSource:

1. Sign up for a CyberSource account and obtain the necessary credentials (merchant ID and keys) to connect to the payment gateway.
2. Add the CyberSource SDK or library to the application. This will typically involve adding a dependency to the application's (Backend) build file (pom.xml) and configuring the application (Mobile application) to use the SDK.
3. Create a payment form in the application that allows users to enter their credit card information and other details required for the payment.
4. Implement the code for handling the payment process in the application. This typically involves creating a payment request, submitting it to the CyberSource payment gateway, and handling the response.
5. Implement security measures such as encryption and tokenization to protect sensitive information.
6. Test the integration with CyberSource test environment and then production environment.
7. Update the application to handle different scenarios like success, failure, or cancelation of the transaction.

5.3.1.2 CyberSource Credentials for Fahes Payment

CyberSource credentials are the necessary credentials (profile ID and keys) required to connect to the payment gateway. These credentials are configured in the BE side as properties in the application configuration files.

❖ **PROD:**

Profile ID	6A326091-141A-4423-A4D8-590914169580
Secret Key	8e0d2008919248b8b8f8022bd844ec7f9947d98996a54bd0b66fc4b08c40e20a6291353fdec24778ab86c4178748e081439307617e5648e6ab6429e53297c0efdc07a3500f842819a8c8ffd71013b5583e33ee f89c742ac815726956ade450c188352535c1f4e6f97a041d8fd92f509b4250d2a7a3b4ed9b1ad916d052a7b01
Access Key	cfa38d0433713a02b2d9c2f848798899

❖ **UAT (UWOQODB2C Runtime):**

Profile ID	BD7107E3-E57A-4505-B1B6-3E97610A0265
Secret Key	48a7fd9d5bba452d8b9d8fd9b86577acbf92f1318edf4bbdbf3b5870cbb910f61a42f2f5d774c9bbb0d032ac3dcaa03a16a7b7747594baa83466aa81948ddf27e7b32f37038453ebfaeabe92ff41f4867be75515c3643f1bacb7368becadac10cb58d4e707b46dfa85e1ebb3001797d8bbd90aa3a6b4f0092ed8d0c8a9b8c0c
Access Key	395d8c526a653b2ebb7a35f8e6f624f7

❖ **UAT (SWOQOD Runtime):**

Profile ID	F494F19E-5A11-4198-AF33-0B6B63A7E24C
Secret Key	b8780f3851394f059fccdfb0503ed6c011f903116581430582e2830cb8e090c15dd442929b584167b6702ce012b2654dc0bb45adde3d47b89b34c658c391344e23d609db806145c9b2df33f9425a02e21e7a4cd58af340078bc750ef4bea36e05ced2fd5146941ed819443a0cba4a9b37e8842f924d7433da80789408bf2639d
Access Key	f7f45454c48e3e34a5e7ce87457b3303

5.3.1.3 CyberSource Credentials for WOQOD Payment

❖ **PROD:**

Profile ID	56C1ED3C-1169-40B3-8652-2A4BAB631206
Secret Key	2248a81b0fb944b3b1f485762e43067671c83051304e4f0dabe4442c0f580dd9019cecf16cde4a57bd6c21335011f2f94babd1b08e294afbb292a31b6817112b5c5535cdeea8434d962c95cf105bb453edfaa4d839c242d2b27524804f759fd464ca1b684036404f94a86b7bd783ce0dcd0cefffc60344dabf3ed0d1e8e8dacd
Access Key	99d20317a2cd3700b539c80c1bd08190

❖ **UAT (UWOQODB2C Runtime):**

Profile ID	94E17A6D-0A37-4C0E-B003-28FBC25E63C4
Secret Key	e631050519f34f8096746b963ca8ab78c8a7d660fd954b91aaebdbd186a3bf6e6067dd4f65a04a2b8909d3681654d49daf9eda21fc534802bfb210bd1043d78eccce3695593c47d19aa49b018fd0410a3f7ab6a8ad1e4421ace6cd6489670ece9b8e826db11d4daeb75213d9412f7cfc3f18e8ee3d0c4ae9b0777424b4216369
Access Key	a81cdb41264837fea58c6eb72bf6f35a

❖ **UAT (SWOQOD Runtime):**

Profile ID	4D124593-8335-4873-ADC3-F152F5221C23
Secret Key	a76e86884659428485235cc4408a85b16532ce35d0934307b7df49222dc9ef00fde377b34de84af88a4a29b85cf468fcb2a294268bd04be0a1f8847e62e14c41b4afb9a453024d2abefaf49e66b34f41365139e3b8cc4f2bb0c8735edcd91cc6159dbd2159254f0ea9264243be82473cbe90ee15d8da4c908fa07ecf423f7bb9
Access Key	75f0aa0fc7d438038579aab92097c027

5.3.2 QPay payment configuration

5.3.2.1 Description

Qpay is a payment gateway provided by Qatar National Bank (QNB) that allows merchants to process payments from customers in Qatar. Integrating Qpay into a mobile application involves configuring the application to communicate with the Qpay payment gateway and handle the various steps of the payment process.

Steps for integrating Qpay:

1. Sign up for a Qpay account and obtain the necessary credentials (merchant ID and keys) to connect to the payment gateway.
2. Add the Qpay SDK and library to the application. This will typically involve adding a dependency to the application's build file (pom.xml) and configuring the application to use the SDK.
3. Create a payment form in the web application that allows customers to enter their credit card information and other details required for the payment.
4. Implement the code for handling the payment process in the application. This typically involves creating a payment request, submitting it to the Qpay payment gateway, and handling the response.
5. Implement security measures such as encryption and tokenization to protect sensitive information.
6. Test the integration with Qpay test environment and then production environment.
7. Update the application to handle different scenarios like success, failure, or cancelation of the transaction.
8. Update the application to handle refunds.

5.3.2.2 QPay Merchant configuration

Description	Value(s)/Note	
	Staging (UAT)	Production
Referrer URL	http://10.97.111.12:8080/	http://10.97.16.21:8080/ http://10.97.16.22:8080/
PG Redirect Pay URL	https://pgtest3.qcb.gov.qa/QPayOneWeb/EZConnectRequestServlet	https://www.qpay.gov.qa/QPayOneWeb/EZConnectRequestServlet
PG Inquiry & Refund Action URL	https://pgtest3.qcb.gov.qa/QPayOneWeb/EZConnectRequestServlet	https://www.qpay.gov.qa/QPayOneWeb/EZConnectRequestServlet

5.3.2.3 QPay Credentials for Fahes Payment

This section contains the necessary credentials (merchant ID for both Fahes & WOQODe, secret keys and redirection URLs) required to connect to the payment gateway. These credentials are configured in the BE side as properties in the application configuration files.

Description	Value(s)/Note	
	Staging (UAT)	Production
Merchant ID	7954014131	
Secret Key	N2YzMjVkJkZDVkMWQ2NTU0NDFiNGI4NTNj	MzI0ZTkzMzk5OGU5YmJmYzYyNzA2NzBm
ExtraFieldsF14	http://10.97.111.11:9080/uwoqodb2c/api/adapters/fahesregistration/qpay/success	https://mobileapp1.woqod.com/newwoqod/api/adapters/fahesregistration/qpay/success

5.3.2.4 QPay Credentials for WOQODe Payment

Description	Value(s)/Note	
	Staging (UAT)	Production
Merchant ID	3333066131	
Secret Key	YWZhZjNhMjM5OTE5MzVIZGU3YWNhZmZh	MzE4YmFhM2U0N2I2ZjdYTRjZDRiM2Rh
ExtraFieldsF14	http://10.97.111.11:9080/uwoqodb2c/api/adapters/woqodetopup/qpay/success	https://mobileapp1.woqod.com/newwoqod/api/adapters/woqodetopup/qpay/success

5.3.3 Payment call-back URLs

QPay and CyberSource both provide electronic payment solutions, and both have a feature known as call-back URL.

A call-back URL is a feature provided by QPay and CyberSource to allow merchants to receive real-time notifications of payment transactions made through the platform. When a customer makes a payment using the QPay or CyberSource payment

gateway, the system sends a call-back, to a specified URL. This URL is set up by the merchant and is used by QPay or CyberSource to communicate the status of the payment transaction to the merchant.

Payment Gateway	UAT	PROD
CyberSource	<p>Fahes call-back URL: https://mobileuat.wogod.com/uwoqodb2c/api/adapters/woqodetopup/prtransactionlog/success</p> <p>WOQODe Top-up call-back URL: https://mobileuat.wogod.com/uwoqodb2c/api/adapters/woqodetopup/success</p>	<p>Fahes call-back URL: https://mobileapp1.wogod.com/newwoqod/api/adapters/woqodetopup/prtransactionlog/success</p> <p>WOQODe Top-up call-back URL: https://mobileapp1.wogod.com/newwoqod/api/adapters/woqodetopup/success</p>
QPay	<p>PG Redirect Pay URL: https://pgtest3.qcb.gov.qa/QPayOnePC/PaymentPayServlet</p> <p>PG Inquiry & Refund Action URL: https://pgtest3.qcb.gov.qa/QPayOneWeb/EZConnectRequestServlet</p> <p>Fahes pre-registration call-back URL: https://mobileuat.wogod.com/uwoqodb2c/api/adapters/fahesregistration/qpay/success</p> <p>WOQODe Top-up call-back URL: https://mobileuat.wogod.com/uwoqodb2c/api/adapters/woqodetopup/qpay/success</p>	<p>PG Redirect Pay URL: https://www.qpay.gov.qa/QPayOnePC/PaymentPayServlet</p> <p>PG Inquiry & Refund Action URL: https://www.qpay.gov.qa/QPayOneWeb/EZConnectRequestServlet</p> <p>Fahes pre-registration call-back URL: https://mobileapp1.wogod.com/newwoqod/api/adapters/fahesregistration/qpay/success</p> <p>WOQODe Top-up call-back URL: https://mobileapp1.wogod.com/newwoqod/api/adapters/woqodetopup/qpay/success</p>

5.3.4 In Progress Transaction handling

5.3.4.1 CyberSource

An In Progress transaction indicates that the payment is still being processed by the system and a final decision has not been reached. The merchant should check the status of the transaction again at a later time to determine if it has been **approved or declined**.

CyberSource has the ability to handle in-progress transactions through its API. This can be done using the CyberSource **Search Payments API** that will initiate the desired action and receive a response indicating the status of the transaction.

In our project, we have implemented an **In Progress script** to handle those transactions, as they can occur for a variety of reasons such as network connectivity issues, or a delay in the processing time for the payment due to a high volume of transactions.

The script will check the status of the transaction after **20 minutes** of its creation date and update it in the database.

5.3.4.2 QPay

To handle In Progress transactions in QPay payment, we have implemented an In Progress script to periodically check the status of the transaction using the QPay **Inquiry API call**:

Once the transaction status changes to In Progress, we wait for a certain period of time before checking the status again. The waiting time is based on the time limit set by QPay (20 minutes).

- **If the status of the transaction received from the Inquiry call is PAID**, we proceed with updating the user's balance in case of WOQODe Top-up payment or Fahes pre-registration process in case of Fahes payment.

- If the status of the transaction received from the Inquiry call is **FAILED**, we update the status in the database and save the necessary information on why the transaction was declined.

5.4 Swagger documentation

5.4.1 Description

Swagger documentation is the process of generating human-readable documentation from the Swagger specification. This documentation is generated in a format that is easy to read and understand, and it includes information such as the resources, operations, and parameters of the API, as well as example responses. The purpose of integrating Swagger documentation is to generate interactive API documentation that describes the functionality of the application's RESTful endpoints. This documentation can be used by developers to understand how to use the API and test out requests and responses.

5.4.2 Swagger configuration

❖ UAT:

- Swagger API URL: <http://10.97.111.12:8080/BE2020/swagger-ui.html>
- Username: woqod
- Password: woqod

❖ Preprod:

- Swagger API URL: <http://10.97.111.12:8181/BE/swagger-ui.html>
- Username: woqod
- Password: woqod

❖ PROD:

- Swagger API URL: <http://10.97.16.21:8080/BE/swagger-ui.html> / <http://10.97.16.22:8080/BE/swagger-ui.html>
- Username: woqod
- Password: woqod

5.5 Base URLs configuration

❖ UAT:

- Mobile First Base URL: (uwoqodb2c Runtime): <https://mobileuat1.woqod.com/uwoqodb2c/api/adapters/>
- Base URL - Back End UAT: <http://10.97.111.12:8080/BE2020>

❖ Preprod:

- Mobile First Base URL: (SWoqod Runtime): <https://mobileuat1.woqod.com/SWoqod/api/adapters/>
- Base URL - Back End UAT: <http://10.97.111.12:8080/BE>

❖ PROD:

- Mobile First Base URL: <https://mobileapp1.woqod.com/pwoqodb2c/api/adapters/>
This URL will be used as the base URL of all application API used below in the features section as **"MFP-Base-URL"**
- Base URL - Back End: <http://10.97.16.21:8080/BE> / <http://10.97.16.22:8080/BE>
This URL will be mentioned as **"Backend-Base-URL"** in the features section.

5.6 Azure File Storage

5.6.1 Description

Azure file storage is a fully managed, scalable, and redundant object storage service provided by Microsoft Azure used to store unstructured data such as images, videos, audio, and documents. Blobs can be accessed via HTTP/HTTPS and can be managed using Azure Storage REST API or Azure Storage SDKs.

We used this storage service to store mobile application banner's images and videos.

5.6.2 Configuration

❖ UAT:

Azure storage account name	wqdstoragewebuat
Azure storage container name	wqdstoragewebuatcont
Endpoint	https://wqdstoragewebuat.blob.core.windows.net/
Azure storage blob endpoint	https://wqdstoragewebuat.blob.core.windows.net/wqdstoragewebuatcont
SAS Token	sp=racwdl&st=2021-09-12T07:11:08Z&se=2023-09-03T15:11:08Z&spr=https&sv=2020-08-04&sr=c&sig=vii6OfxjUS3BpzReg7wWmNfk8GoZad9b17Q0rmRcnhM%3D
Container name	wqdstoragewebuatcont

❖ PROD:

Azure storage account name	wqdstoragewebprd
Azure storage container name	wqdstoragewebprdcont
Endpoint	https://wqdstoragewebprd.blob.core.windows.net/
Azure storage blob endpoint	https://wqdstoragewebprd.blob.core.windows.net/wqdstoragewebprdcont
SAS Token	sp=racwdl&st=2022-03-01T08:28:19Z&se=2024-03-05T16:28:19Z&spr=https&sv=2020-08-04&sr=c&sig=JAjj9RxEueXf4eCbPwHR6wj9DIDHnbq7JH2XBo9ZOn0%3D
Container name	wqdstoragewebprdcont

	SAS Token	SAS URL
RO access	sp=rl&st=2022-03-01T04:52:01Z&se=2024-03-05T12:52:01Z&spr=https&sv=2020-08-04&sr=c&sig=b4%2FA9x1LJ8SvqDS03AEdoxaKMkJZ2xOWZ6p%2FLbS1IEA%3D	https://wqdstoragewebprd.blob.core.windows.net/wqdstoragewebprdcont?sp=rl&st=2022-03-01T04:52:01Z&se=2024-03-05T12:52:01Z&spr=https&sv=2020-08-04&sr=c&sig=b4%2FA9x1LJ8SvqDS03AEdoxaKMkJZ2xOWZ6p%2FLbS1IEA%3D
Full access	sp=racwdl&st=2022-03-01T08:28:19Z&se=2024-03-05T16:28:19Z&spr=https&sv=2020-08-04&sr=c&sig=JAjj9RxEueXf4eCbPwHR6wj9DIDHnbq7JH2XBo9ZOn0%3D	https://wqdstoragewebprd.blob.core.windows.net/wqdstoragewebprdcont?sp=racwdl&st=2022-03-01T08:28:19Z&se=2024-03-05T16:28:19Z&spr=https&sv=2020-08-04&sr=c&sig=JAjj9RxEueXf4eCbPwHR6wj9DIDHnbq7JH2XBo9ZOn0%3D

6. Features

6.1 Login

Login module is used to allow the user to sign in into WOQOD app.

If the user has an account, he can use his [Username, Mobile number, Email or QID] to perform the login, otherwise there is a sign-up button to create a new account.

The option to login with biometric is active once the user logs-in then accepts to enable this option and it depends on the availability of this option on the device itself.

- **APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/checkFingerPrint	Backend-Base-URL/bio/checkFingerPrint	API to verify if the user has activated the login with fingerprint.	POST	"bioPin": "string", "customPwd": "string", "deviceId": "string", "username": "string"	Boolean
MFP-Base-URL/users/getProfilePhoto	Backend-Base-URL/users/getProfilePhoto	API to get the user profile photo after login.	GET	"username": "string"	String (User Profile Photo)
MFP-Base-URL/ApiAdapter/login/verifyCredentials	Backend-Base-URL/login/verifyCredentials	API to verify user credentials		{ "bioPin": "string", "deviceId": "string", "deviceType": "string", "email": "string", "ipAddress": "string", "password": "string", "qid": "string", "type": "Username_Auth", "username": "string" }	Boolean
MFP-Base-URL/ApiAdapter/pincode/login/send	Backend-Base-URL/pincode/login/send	API to send otp to the user mobile number		{ "biostatus": true, "creationDate": "2023-01-11T14:02:28.453Z", "id": 0, "ipAddress": "string", "language": "ar", "pincode": "string", "type": "Username_Auth", "used": true, "username": "string", "wrongAttempt": 0 }	Boolean
MFP-Base-URL/ApiAdapter/pincode/login/validatePinCode	Backend-Base-URL/pincode/login/validatePinCode	API to validate the pin code		{ "biostatus": true, "creationDate": "2023-01-11T14:15:05.279Z", "id": 0, "ipAddress": "string", "language": "ar", "pincode": "string", "type": "Username_Auth", "used": true, "username": "string", "wrongAttempt": 0 }	Boolean

6.2 Registration

Registration is divided into three steps, the first step to validate the QID and mobile phone number, the second step to check the user's validity and the last step to register the mobile user.

- **APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/register/checkQidValidity	Backend-Base-URL/checkqidvalidity	API to check if the QID and Mobile number match.	GET	number: string, qid: string	Boolean
MFP-Base-URL/register/register	Backend-Base-URL/register	API to register the user information.	POST	language: string, qid: string	Boolean
MFP-Base-URL/users/checkUser	Backend-Base-URL/users/checkUser	API to check if the mobile number and QID are in the correct form.	POST	{ "email": "string", "phone": "string", "qid": "string", "username": "string" }	Boolean
MFP-Base-URL/area	Backend-Base-URL/area	API to get list of areas.	GET	arabicTitle: String title: String	List of areas

6.3 Activate Account

This functionality comes up right after the three registration steps, where the user has to confirm his account through getting and entering the confirmation code received on his phone by SMS.

- **APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/register/activate	Backend-Base-URL/activate	API to verify the activation code that the user entered.	POST	{ "deviceSerial": "string", "deviceType": "string", "pinCode": "string", "username": "string" }	Boolean
MFP-Base-URL/register/sendpincode	Backend-Base-URL/sendpincode	API to resend the activation code.	PUT	language: string, username: string	Boolean

6.4 Forget Password

When the user forgets his password, there is a button in the login screen that redirect the user to the forget password screen, a verification of the user will happen in the first step, the user will receive a code on his phone and once that code is validated, the user will be asked to enter a new password.

This functionality comes up right after the three registration steps, where the user has to confirm his account through getting and entering the confirmation code received on his phone by SMS.

- **APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/register/recover	Backend-Base-URL/recover	API to send a verification code via email.	POST	{ "language": "ar", "newpassword": "string", "pincode": "string", "type": "Username_Auth", "username": "string" }	Boolean

MFP-Base-URL /register/checkrecoverycode	Backend-Base-URL/ checkrecoverycode	API to validate the verification code.	POST	{ "language": "ar", "newpassword": "string", "pincode": "string", "type": "Username_Auth", "username": "string" }	Boolean
MFP-Base-URL/ register/sendrecoverycode	Backend-Base-URL/ sendrecoverycode	API to update the forgotten password.	POST	{ "language": "ar", "newpassword": "string", "pincode": "string", "type": "Username_Auth", "username": "string" }	Boolean

6.5 Home

Home module contains dynamic content configurable from Back-office.

- APIs:

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/RspServices/BalanceInquiry		API to get the current users account balance.	POST	Qid: String Mobile: String	Balance Inquiry Response
MFP-Base-URL ApiAdapter/home/topbanner	Backend-Base-URL/topbanner/active	API to get Top Banner content.	GET	No parameters	List of active top banners
MFP-Base-URL/ ApiAdapter/petrol/prices	Backend-Base-URL/petrol/prices	API to get Fuel Prices content	GET	No parameters	List of fuel prices
MFP-Base-URL/ ApiAdapter/home/businessbanner	Backend-Base-URL/businessbanner/activebanner	API to get Ads Banner content	GET	No parameters	List of active business banners

6.6 News

The news feature is responsible for retrieving the list of news to be displayed in the mobile application, these news are configured from the Backoffice.

- APIs:

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter/news	Backend-Base-URL/ news	API to get a list of active news.	GET	request: offset: Integer pageNumber: Integer pageSize: Integer paged: Boolean sort.sorted: Boolean sort.unsorted: Boolean unpaged: Boolean	List of news
MFP-Base-URL/ ApiAdapter/news/view	Backend-Base-URL/ news/view	API to update news view count.		id: String	Boolean

6.7 Profile

This module is responsible for fetching and updating mobile user information.

- Integration points and APIs:

MFP URL	Backend URL	Description	Method	Request	Response
<u>MFP-Base-URL</u> /ApiAdapter/users	<u>Backend-Base-URL</u> /users	API to update user information.	PUT	{ "adress1Label": "string", "birthdate": "timestamp", "firstName": "string", "id": 0, "idArea": "string", "ipAddress": "string", "lastName": "string", "poBox": "string" }	Current User Resource
<u>MFP-Base-URL</u> ApiAdapter/users/updatePhoto	<u>Backend-Base-URL</u> /users/updateProfilePhoto	API to update the profile photo.	PUT	{ "photo": "string", "username": "string" }	Current User Resource
<u>MFP-Base-URL</u> / ApiAdapter/area	<u>Backend-Base-URL</u> / area	API to get list of areas.	GET		List of areas
<u>MFP-Base-URL</u> / ApiAdapter/register/checkQidAndMobileValidity	<u>Backend-Base-URL</u> / checkQidAndMobileValidity	API to check if mobile number and QID are valid.	GET	umber: String qid: String	Boolean
<u>MFP-Base-URL</u> / ApiAdapter/pincode/sendPinCodeToUpdateUserEmail	<u>MFP-Base-URL</u> / sendPinCodeToUpdateUserEmail	API to send pin code when changing user email.	POST	{ "email": "string", "ipAddress": "string", "language": "ar", "phone": "string", "username": "string" }	Boolean
<u>MFP-Base-URL</u> / ApiAdapter/pincode/sendPinCodeToUpdateUserPhoneNumber	<u>MFP-Base-URL</u> / sendPinCodeToUpdateUserPhone	API to send pin code when changing user mobile number.	POST	{ "email": "string", "ipAddress": "string", "language": "ar", "phone": "string", "username": "string" }	Boolean
<u>MFP-Base-URL</u> / ApiAdapter/ users/updatePhoneNumber	<u>MFP-Base-URL</u> / updatePhoneNumber	API to update user mobile number.	PUT	{ "email": "string", "ipAddress": "string", "language": "ar", "phone": "string", "username": "string" }	Boolean
<u>MFP-Base-URL</u> / ApiAdapter/users/updateEmail	<u>MFP-Base-URL</u> / updateEmail	API to update user email.	PUT	{ "email": "string", "ipAddress": "string", "language": "ar", "phone": "string", "username": "string" }	Boolean

6.8 Reset Password

The reset password feature allows users to reset their password in case they have forgotten it. The integration with an LDAP server can provide an efficient and secure way to store and manage the user credentials for this feature.

Here's an overview of how the password reset feature is integrated with the LDAP server:

- User initiates password reset request: The user triggers the password reset process by providing their username.
- Validation of user identity: the backend application validates the user's identity by checking the information(username) provided against the LDAP directory and the database.
- Generation of verification code: If the user's identity is confirmed, a verification pin code is generated and sent to the user's phone number.
- User inputs the verification code then verification of the verification code in the backend side.
- Update of LDAP password, upon successful validation of the verification code, the backend application updates the user's password in the LDAP directory.

• APIs:

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter/pincode/send	Backend-Base-URL/ pincode/send	API to send a code via email.	POST	{ "biostatus": true, "creationDate": "2023-01-12T09:05:21.399Z", "id": 0, "ipAddress": "string", "language": "ar", "pincode": "string", "type": "Username_Auth", "used": true, "username": "string", "wrongAttempt": 0 }	Boolean
MFP-Base-URL ApiAdapter/pincode/validatePinCode	Backend-Base-URL / pincode/validatePinCode	API to validate the code sent via email.	POST	{ "biostatus": true, "creationDate": "2023-01-12T09:07:18.468Z", "id": 0, "ipAddress": "string", "language": "ar", "pincode": "string", "type": "Username_Auth", "used": true, "username": "string", "wrongAttempt": 0 }	Boolean
MFP-Base-URL/ ApiAdapter/users/password	Backend-Base-URL/ users/password	API to update the Password.	PUT	{ "email": "string", "ipAddress": "string", "language": "ar", "phone": "string", "username": "string" }	Boolean

6.9 Supermarket

This feature is responsible for showing all Shafaf supermarkets with their details and showing them on a map with the option to filter the pins by name and distance.

• APIs:

MFP URL	Backend URL	Description	Method	Request	Response
---------	-------------	-------------	--------	---------	----------

MFP-Base-URL/ ApiAdapter/supermarket	Backend-Base-URL/ SuperMarket	API to get the list of Shafaf supermarkets.	GET	No parameters	List of Shafaf supermarkets.
------------------------------------------------	-----------------------------------------	---------------------------------------------	-----	---------------	------------------------------

6.10 Shafaf Retailers

Shafaf retailers feature is responsible for showing the list of Shafaf retailers in the mobile app with the possibility to view each supermarket on the map and calling it.

- APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter/retailers	Backend-Base-URL/ retailers	API to get the list of Shafaf Retailers	GET	No parameters	List of Shafaf retailers.

6.11 Bulk LPG

Bulk LPG feature is responsible for retrieving information about the Bulk LPG and the list of contractors with their contact information.

- APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter/contractors	Backend-Base-URL/ Contractor	API to get a list of Bulk LPG contractors	GET	No parameters	List of Contractors.

6.12 Fuel Prices

Fuel prices feature is responsible for showing the list of fuel prices retrieved from SharePoint.

- APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter/petrol/prices	Backend-Base-URL/ petrol/prices	API to get a list of Fuel Prices.	GET	No parameters	List of fuel prices.

6.13 Promotions

Promotions feature is responsible for displaying the list of available promotions retrieved from SharePoint.

- APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter/promotions	Backend-Base-URL/ promotions	API to get a list of current promotions.	GET	No parameters	List of promotions.

6.14 Tenders

Tenders feature is responsible for showing the list of tenders retrieved from SharePoint.

- APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
<u>MFP-Base-URL/</u> ApiAdapter/tenders/tenders	<u>Backend-Base-URL/</u> tenders/tenders	API to get a list of tenders.	GET	offset: Integer pageNumber: Integer pageSize: Integer paged: Boolean sort.sorted: Boolean sort.unsorted: Boolean unpaged: Boolean	List of tenders.

6.15 Feedback

Feedback functionality allows the user to send new feedback from the mobile application and get the list of his feedbacks with their details.

- **APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
<u>MFP-Base-URL/</u> feedback/newFeedback	<u>Backend-Base-URL/</u> feedbacks	API to add new feedback.	POST	{ "category": "Complaint", "connected": true, "contactNumber": "string", "content": "string", "creationDate": "2023-01-12T09:47:03.919Z", "email": "string", "feedbackNumber": "string", "feedbackResponseResources": [{ "adminName": "string", "content": "string", "creationDate": "2023-01-12T09:47:03.919Z", "feedbackId": 0, "id": 0, "ticketNumber": "string" }], "file": "string", "id": 0, "language": "ar", "name": "string", "qid": "string", "sector": "PETROL_STATION", "status": true, "ticketNumber": "string", "userName": "string" }	Boolean
<u>MFP-Base-URL/</u> ApiAdapter/feedback/lazyFeedbackHistory	<u>Backend-Base-URL/</u> /feedbacks/lazyFeedbackHistory	API to get a list of the current user feedbacks.	GET	username: String	List of Feedbacks
<u>MFP-Base-URL/</u> feedback/getFeedbackById	<u>Backend-Base-URL/</u> feedbacks/getFeedbackById	API to get feedback by id.	GET	id: String	Feedback response resource

6.16 Settings

Tenders feature is responsible for showing the list of tenders retrieved from SharePoint.

- **APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter/bio/updateStatus	Backend-Base-URL/ /bio/status	API to update the state of biometric authentication.	PUT	{ "deviceId": "string", "username": "string" }	Boolean
MFP-Base-URL/ ApiAdapter/pincode/login/send	Backend-Base-URL/ pincode/login/send	API to send otp to the user mobile number		{ "biostatus": true, "creationDate": "2023-01-11T14:02:28.453Z", "id": 0, "ipAddress": "string", "language": "ar", "pincode": "string", "type": "Username_Auth", "used": true, "username": "string", "wrongAttempt": 0 }	Boolean
MFP-Base-URL/ ApiAdapter/pincode/login/validatePinCode	Backend-Base-URL/ pincode/login/validatePinCode	API to validate the pin code		{ "biostatus": true, "creationDate": "2023-01-11T14:15:05.279Z", "id": 0, "ipAddress": "string", "language": "ar", "pincode": "string", "type": "Username_Auth", "used": true, "username": "string", "wrongAttempt": 0 }	Boolean
MFP-Base-URL/ ApiAdapter/users/deleteUser	Backend-Base-URL/ /users/delete	API to delete the user	DELETE	username: String	Boolean

6.17 Notification

Notification functionality is responsible for fetching the list of all the received push notifications from BO.

- **APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter/notification/listByUsername	Backend-Base-URL/ notification/listByUsername	API to get all the notification history corresponding to this user.	GET	deviceId: String offset: Integer pageNumber: Integer pageSize: Integer paged: Boolean sort.sorted: Boolean sort.unsorted: Boolean unpaged: Boolean	List of Notifications sent to the user
MFP-Base-URL/ ApiAdapter/notification/anonymous	Backend-Base-URL/ notification/anonymous	API to get all the public notification history.	GET	deviceId: String offset: Integer pageNumber: Integer pageSize: Integer paged: Boolean sort.sorted: Boolean sort.unsorted: Boolean	List of public notifications

				unpaged: Boolean	
MFP-Base-URL/ ApiAdapter /notification/updateAll	Backend-Base-URL/ /notification/updateAll	API to update the status of all notifications to "read".	GET	connected: Boolean deviceid: String	Boolean
MFP-Base-URL/ ApiAdapter/notification/u pdateStatu	Backend-Base-URL/ notification/updateStat us	API to update the status of a specific notification to "read".		{ "deviceid": "string", "idnotif": "string", "uniqueid": "string", "username": "string" }	Boolean

6.18 Survey

This module is responsible for creating new survey from the BO, sending the survey to all users or only to WOQODe users. Mobile users receive the survey as a notification and can respond to questions of the survey. A survey can have one or multiple questions and each question can have single or multiple answers.

- **APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ /surveys/surveysResponse	Backend-Base-URL/ surveysResponses	API to submit the user answer of the survey.	POST	{ "answredAt": "2023-01-12T10:31:05.130Z", "deviceid": "string", "id": 0, "questionResponseResources": [{ "id": 0, "questionId": 0, "responses": ["string"] }], "surveyId": 0 }	Boolean

6.19 WOQODe

6.19.1 WOQODe Top-up

This functionality allows the user to make a Top-up into his WOQODe account by putting the amount of money that he wants to add to the account and make the payment with his credit card and by respecting the min/max amount configured from the BO.

- **APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/RspServices/BalanceInquiry		API to get the current users account balance.	POST	Qid: String Mobile: String	Current User Account Balance
MFP-Base-URL/ ApiAdapter/updateWoqodeUser	Backend-Base-URL/ updateWoqodeUser	API to update the user woqode column in the DB (Connected Mode)	POST	isWoqode: boolean mobile: String qid: String	Boolean
MFP-Base-URL/ ApiAdapter/getMaxMin	Backend-Base-URL/ amount/getMaxMin	API to get the min and max amount of money	GET	No parameters	Minimum and maximum

		the user can put into their WOQODE account.			amount of topup
MFP-Base-URL/ ApiAdapter/isDebitCardPaymentHidden	Backend-Base-URL/ debit-card-payment-config	API to get is the debit should be shown or not	GET	No parameters	Boolean
MFP-Base-URL/ ApiAdapter/date		API to get the current date	GET		Date Model
MFP-Base-URL/ ApiAdapter/createTransactionUUID	Backend-Base-URL/ /payment/createTransactionUUID	API to create a payment transaction.	POST	{ "amount": 0, "authReversal": "string", "authReversalStatus": "string", "createdDate": "2023-01-12T11:00:56.107Z", "currency": "string", "description": "string", "email": "string", "id": 0, "lastModifiedDate": "2023-01-12T11:00:56.107Z", "mobile": "string", "qid": "string", "referenceNumber": "string", "rpsStatus": "string", "transactionID": "string", "transactionSearchDescription": "string", "transactionStatus": "INPROGRESS", "transactionUUID": "string", "userID": "string" }	Boolean
MFP-Base-URL/ ApiAdapter/signature2		API to get the payment signature to be used in the payment process (used in the PostCreateTransactionUUIDUseCase)	POST		
MFP-Base-URL/ ApiAdapter/getSignatureParams	Backend-Base-URL/ fahes/getSingParam	API to get the payment parameters such as the payment secret key.	GET	name: String	Signature Params
MFP-Base-URL/ ApiAdapter/updateTransactionUUID	Backend-Base-URL/ /payment/updateTransactionUUID	API to cancel a user transaction	PUT	RPSStatus: String authReversal: String statusEnum: TransactionStatusEnum transationID: String transactionUUID: String	Transaction Log Resource
MFP-Base-URL/ ApiAdapter/qpay/createTransaction	Backend-Base-URL/ payment/qpay/createTransaction	API to create a QPay payment transaction.	POST	{ "amount": 0, "approvalCode": "string", "authReversalStatus": "string", "createdDate": "2023-01-12T11:13:33.144Z", "currency": "string", "description": "string", "email": "string", "id": 0,	Boolean

				<pre> "lastModifiedDate": "2023-01- 12T11:13:33.144Z", "mobile": "string", "pun": "string", "qid": "string", "referenceNumber": "string", "refundStatus": "PENDING", "refundStatusCode": "string", "refundStatusMessage": "string", "rpsStatus": "string", "transactionID": "string", "transactionSearchDescrip tion": "string", "transactionStatus": "INPROGRESS", "transactionStatusCode": "string", "transactionStatusMessag e": "string", "userID": "string" } </pre>	
<u>MFP-Base-URL/</u> ApiAdapter/getQpaySigna tureParam	<u>Backend-Base-URL/</u> fahes/getSingParam	API to get the qpay payment parameters such as the payment secret key.	GET	name: String	Signature Params
<u>MFP-Base-URL/</u> woqodetopup/updatePun	<u>Backend-Base-URL/</u> payement/qpay/update TopupQpayTransaction	API to cancel a user transaction.	PUT	RPSStatus: String pun: String statusEnum: TransactionStatusEnum statusCode:String statusMessage:String transationID: String transactionUUID: String	WOQODe QPay Transaction Resource
<u>MFP-Base-URL/</u> woqodetopup/success	<u>Backend-Base-URL/</u> payement/update TransactionUUID	API to receive payment response details for CyberSource.	POST	decision: String req_transaction_uuid: String auth_response: String req_reference_number: String req_amount: String transaction_id: String reason_code: String auth_trans_ref_no: String req_card_expiry_date: String auth_amount: String bill_trans_ref_no: String req_card_type: String payer_authentication_pare s_status: String req_profile_id: String signed_date_time: String req_bill_to_address_line1: String req_bill_to_address_city: String payer_authentication_xid: String	BalanceIncrea seValue, CivilID,CivilMo bileNumber,T ransactionID, decision,refer ence

				req_customer_ip_address: String payer_authentication_validate_result: String req_bill_to_address_country: String req_bill_to_address_postal_code: String req_payment_method: String	
MFP-Base-URL/ woqodetopup/qpay/success	Backend-Base-URL/ payment/qpay/updateTopupQpayTransaction	API to receive payment response details for QPay.	POST	Response.Amount: String Response.CurrencyCode: String Response.PUN: String Response.EZConnectResponseDate: String Response.ConfirmationID: String Response.MerchantModuleSessionID: String Response.Status: String Response.StatusMessage: String Response.MerchantID: String Response.BankID: String Response.Lang: String Response.AcquirerID: String Response.ItemID: String Response.CardNumber: String Response.CardExpiryDate: String Response.CardHolderName: String Response.AgentID: String Response.SecureHash: String	amount, decision, pun, responseDate, statusCode, statusMessage

6.19.2 WOQODE Transaction History

This functionality is responsible for fetching the transaction history related to his WOQODE account.

- Integration points and APIs:**

MFP URL	Description	Method	Request	Response
MFP-Base-URL/ RpsAdapterJava/getCustomerByFleetName	API to get the current users list of transaction history.	POST	Qid: String	User's Transaction History
MFP-Base-URL/ RpsAdapter/getAllSalesTransactions	API get transaction sales by QID and plate number.	POST	plateNumbersSt: String fleetName: String	List of sales transactions by plate number

6.20 Fahes

6.20.1 Fahes & Petrol Stations

Fahes & petrol stations list retrieved from SharePoint to be displayed is Mobile app's map with their details.

- APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
<u>MFP-Base-URL/</u> ApiAdapter /fahesStations	<u>Backend-Base-URL/</u> fahes/station	API to get the list of all Fahes stations.	GET	No parameters	List of Fahes Stations
<u>MFP-Base-URL/</u> ApiAdapter /supermarket	<u>Backend-Base-URL/</u> SuperMarket	API to get the list of supermarkets.	GET	No parameters	List of SuperMarkets
<u>MFP-Base-URL/</u> ApiAdapter/stations	<u>Backend-Base-URL/</u> Stations	API to get the list of WOQOD's Petrol stations.	GET	category: StationCategoryEnum (FAHES, SIDRA, MARINE, PETROL...)	List of Petrol station

6.20.2 Fahes Add New Car

This functionality allows connected mobile users to add new cars.

- APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
<u>MFP-Base-URL/</u> fahes/plateTypes	<u>Backend-Base-URL/</u> plate	API to get plates types list	GET	No parameters	List of plate types
<u>MFP-Base-URL/</u> ApiAdapter/fahes/plateTypes	<u>Backend-Base-URL/</u> car/getCarsByQid	API to get the list of cars by qid.	GET	qid: String	List of Cars
<u>MFP-Base-URL/</u> ApiAdapter /fahes/checkCar	<u>Backend-Base-URL/</u> car/checkCar	API to check car	GET	plate_number: String plate_type_id: String qid: String	Car Resource
<u>MFP-Base-URL/</u> ApiAdapter/pincode/send	<u>Backend-Base-URL/</u> pincode/send	API to send verification code via phone number	POST	{ "biostatus": true, "creationDate": "2023-01-12T09:05:21.399Z", "id": 0, "ipAddress": "string", "language": "ar", "pincode": "string", "type": "Username_Auth", "used": true, "username": "string", "wrongAttempt": 0 }	Boolean
<u>MFP-Base-URL/</u> ApiAdapter/pincode/validatePinCode	<u>Backend-Base-URL/</u> pincode/validatePinCode	API to validated the entered code verification	POST	{ "biostatus": true, "creationDate": "2023-01-12T09:07:18.468Z", "id": 0, "ipAddress": "string", "language": "ar", "pincode": "string", "type": "Username_Auth", "used": true, "username": "string", "wrongAttempt": 0 }	Boolean

MFP-Base-URL/ ApiAdapter /fahes/addCar	Backend-Base-URL/ car/register	API to submit the car information.	POST	Car Resource pincode: String qid: String	Boolean
-----------------------------------------------------	------------------------------------------	------------------------------------	------	------------------------------------------------	---------

6.20.3 Fahes new inspection online payment

This functionality allows mobile users to create new inspection and pay the inspection cost with debit or credit card for both connected and guest mode.

- APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter /fahes/plateTypes	Backend-Base-URL/ plate	API to get plates types list	GET	No parameters	List of plate types
MFP-Base-URL/ ApiAdapter/fahes/isOwner	Backend-Base-URL/ car/isOwner	API to check is owner	GET	plate_number: String plate_type_id: String qid: String	Car Resource
MFP-Base-URL/ ApiAdapter/fahes/inspection/fee	Backend-Base-URL/ inspection/fee	API to get fees	GET	plate_number: String plate_type_id: String qid: String	Inspection Resource
MFP-Base-URL/ ApiAdapter/isDebitCardPaymentHidden	Backend-Base-URL/ debit-card-payment-config	API to get is the debit should be shown or not	GET	No parameters	Boolean
MFP-Base-URL/ ApiAdapter/date		API to get the current date	GET		Date Model
MFP-Base-URL/ ApiAdapter/getSignatureParam	Backend-Base-URL/ fahes/getSingParam	API to get the payment parameters such as the payment secret key	GET	name: String	Signature Params
MFP-Base-URL/ ApiAdapter/signature2		API to get the payment signature to be used in the payment process	POST		
MFP-Base-URL/ ApiAdapter/fahes/prtransactionlog/creditcard/save	Backend-Base-URL/ prtransactionlog/creditcard/save	API to create the new transaction in the database	POST	anonym: String plate_number: String plate_type: String prTransactionLogResource: prTransactionLogResource	Boolean
MFP-Base-URL/ ApiAdapter/fahes/prtransactionlog/updateTransactionUUID	Backend-Base-URL/ prtransactionlog/updateTransactionUUID	API to cancel the transaction	PUT	status: TransactionStatusEnum transactionUUID: String	Boolean
MFP-Base-URL/ ApiAdapter/getQpaySignatureParam	Backend-Base-URL/ fahes/getQpaySignatureParam	API to get the signature of the QPay transaction	GET	name: String action: String amount: String lang: String merchantModuleSessionID: String pun: String quantity: String transactionRequestDate: String	Fahes Resource
MFP-Base-URL/ ApiAdapter/fahes/qpaytransaction/debitcard/save	Backend-Base-URL/ qpay/prtransaction/debitcard/save	API to create new QPay transaction in the database	POST	anonym: Boolean plate_number: String plate_type: String resource: FahesQpayTransactionResource	Boolean

MFP-Base-URL/ fahesregistration/qpay/cancelTransaction	Backend-Base-URL/ qpay/prtransaction/updatePun	API to cancel a debit card transaction	PUT	pun: String statusCode: String statusMessage: String	Boolean
MFP-Base-URL/ fahesregistration/qpay/success	Backend-Base-URL/	API to receive payment response details for QPay.		GPR, GTR, QNB, Response, Response_AR, UPR, UTRS, UTRV, amount, decision, exception, pun, receipt, reference, responseDate, statusCode, statusMessage	
MFP-Base-URL/ /woqodetopup/prtransactionlog/success	Backend-Base-URL/	API to receive payment response details for cyberSource.		GPR, GTR, QNB, Response, Response_AR, UPR, UTRS, UTRV, barcode, decision, exception, Receipt, reference, transactionUUID	

6.20.4 Fahes list of receipts

This functionality allows mobile users to get the list of receipts for Fahes payment.

- APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter/fahes/preRegistration/findByQid	Backend-Base-URL/ preregistration/findByQid	API to get the list of Fahes receipts	GET	Qid: String	Preregistration Resource
MFP-Base-URL/ ApiAdapter/fahes/preregistration/generatePDF	Backend-Base-URL/ preregistration/generatePDF	API to generate the pdf of the selected receipt	GET	reference_number: String	String
MFP-Base-URL/ ApiAdapter/fahes/preRegistration/sendMail	Backend-Base-URL/ preregistration/sendMail	API to send receipt by email	POST	{ "connected": true, "email": "string", "referencenumber": "string" }	Boolean

6.20.5 Fahes Inspection Report

This functionality allows mobile users to get the list of cars and report details for each selected car.

- APIs:**

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ ApiAdapter/fahes/listCars	Backend-Base-URL/ car/getCarsByQid	API to get the list of Fahes cars	GET	qid: String	List of cars
MFP-Base-URL/ ApiAdapter/fahes/inspection/detailInspection	Backend-Base-URL/ inspection/details	API to get the details of inspection report	GET	inspection_id: String	Inspection Resource
MFP-Base-URL/ ApiAdapter/fahes/inspection/historyInspectionsByPlate	Backend-Base-URL/ inspection/listDetailsByPlate	API to get the inspection's information	GET	owner_id: String plate_number: String plate_type_id: String	List of inspections

6.20.6 Fahes Booking

This functionality allows mobile users to book appointment for Fahes inspection.

- APIs:

MFP URL	Backend URL	Description	Method	Request	Response
MFP-Base-URL/ApiAdapter/reservation/isReservationHidden	Backend-Base-URL/reservation-config	API to check if booking feature is active	GET	No parameters	Boolean
MFP-Base-URL/ApiAdapter/reservation/vehicleDetails	Backend-Base-URL/reservation/vehicleDetails	API to check the ownership of the car before proceeding to booking	POST	customerId: String plateNumber: String plateType: String	Car Resource
MFP-Base-URL/ApiAdapter/reservation/vehicleRegistrationValidity	Backend-Base-URL/reservation/vehicleRegistrationValidity	API to check the vehicle registration validity	GET	customerId: String plateNumber: String plateType: String	Vehicle Registration Resource
MFP-Base-URL/ApiAdapter/reservation/getPreReservationReference	Backend-Base-URL/reservation/getPreReservationReference	API to save and get ID for the ongoing booking	GET	{ "customerId": "string", "mobileNumber": "string", "plateNumber": "string", "plateType": "string", "preReservationReferenceId": 0, "vehicleShapeId": "string" }	PreReservation Resource
MFP-Base-URL/ApiAdapter/reservation/otpValidation	Backend-Base-URL/reservation/otpValidation	API for OTP verification	POST	otp: String preReservationReferenceId: String	Boolean
MFP-Base-URL/ApiAdapter/reservation/getAvailableStations	Backend-Base-URL/reservation/getAvailableStations	API to get list of Fahes available stations	GET	preReservationReferenceId: String	List of available stations
MFP-Base-URL/ApiAdapter/reservation/getAppointmentDates	Backend-Base-URL/reservation/getAppointmentDates	API to get available appointments dates for specific station	GET	preReservationReferenceId: String stationId: String	List of Appointment Dates
MFP-Base-URL/ApiAdapter/reservation/getSlots	Backend-Base-URL/reservation/getSlots	API to get available time slots for specific station and date	GET	date: String stationId: String preReservationReferenceId: String	List of available slots
MFP-Base-URL/ApiAdapter/reservation	Backend-Base-URL/reservation/reservation	API to save the booking	POST	vehicleShapeName: String preReservationReferenceId: String userId: String slotId: String	Reservation Resource

6.20.7 Fahes Cancel/Modify Booking

This functionality allows mobile users to modify or cancel his appointment for Fahes inspection.

- APIs:

MFP URL	Backend URL	Description	Method	Request	Response
---------	-------------	-------------	--------	---------	----------

MFP-Base-URL/ ApiAdapter/ /reservation/detailsByCustomer	Backend-Base-URL/ reservation/getDetailsByCustomer	API to check available bookings for specific user	POST	customerId: String	Reservation Resource
MFP-Base-URL/ ApiAdapter /reservation/rescheduling	Backend-Base-URL/ reservation/rescheduling	API to modify specific booking	GET	reservationId: String	Reservation ID
MFP-Base-URL/ ApiAdapter /reservation/cancellation	Backend-Base-URL/ reservation/cancellation	API to cancel specific booking	POST	reservationId: String	Boolean
MFP-Base-URL/ ApiAdapter /reservation/getAvailableStations	Backend-Base-URL/ reservation/getAvailableStations	API to get list of Fahes available stations	GET	preReservationReferenceId: String	List of available stations
MFP-Base-URL/ ApiAdapter /reservation/getAppointmentDates	Backend-Base-URL/ reservation/getAppointmentDates	API to get available appointments dates for specific station	GET	preReservationReferenceId: String stationId: String	List of Appointment Dates
MFP-Base-URL/ ApiAdapter /reservation/getSlots	Backend-Base-URL/ reservation/getSlots	API to get available time slots for specific station and date	GET	date: String stationId: String preReservationReferenceId: String	List of available slots
MFP-Base-URL/ ApiAdapter /reservation/reservation	Backend-Base-URL/ reservation/reservation	API to save the booking	POST	vehicleShapeName: String preReservationReferenceId: String userId: String slotId: String	Reservation Resource

7. Integration points:

7.1 Fahes System

7.1.1 API Links

	FAHES API	BE API Link (UAT)	BE API Link (PROD)	Http Method
Booking				
1	api/v1/account/gbm	No direct URL	No direct URL	
2	api/v1/reservation/plate-type-with-shape-code	http://10.97.111.12:8181/BE/reservation/getPlateTypeWithShape	http://10.97.16.21:8080/BE/reservation/getPlateTypeWithShape	GET
3	api/v1/reservation/vehicle-details	http://10.97.111.12:8181/BE/reservation/vehiculeDetails	http://10.97.16.21:8080/BE/reservation/vehiculeDetails	POST
4	api/v1/reservation/vehicle-registration-validity	http://10.97.111.12:8181/BE/reservation/vehiculeRegistrationValidity	http://10.97.16.21:8080/BE/reservation/vehiculeRegistrationValidity	POST
5	api/v1/reservation/pre-reservation	http://10.97.111.12:8181/BE/reservation/getPreReservationReference	http://10.97.16.21:8181/BE/reservation/getPreReservationReference	POST

6	api/v1/reservation/otp-validation	http://10.97.111.12:8181/BE/reservation/otpValidation	http://10.97.16.21:8080/BE/reservation/otpValidation	POST
7	api/v1/reservation/otp	http://10.97.111.12:8181/BE/reservation/resendOtp	http://10.97.16.21:8080/BE/reservation/resendOtp	PUT
8	api/v1/reservation/stations	http://10.97.111.12:8181/BE/reservation/getAvailableStations	http://10.97.16.21:8080/BE/reservation/getAvailableStations	GET
9	api/v1/reservation/appointment-dates	http://10.97.111.12:8181/BE/reservation/getAppointmentDates	http://10.97.16.21:8080/BE/reservation/getAppointmentDates	GET
10	api/v1/reservation/slots	http://10.97.111.12:8181/BE/reservation/getSlots	http://10.97.16.21:8080/BE/reservation/getSlots	GET
11	api/v1/reservation/can-pay-online	http://10.97.111.12:8181/BE/reservation/canPayOnline	http://10.97.16.21:8080/BE/reservation/canPayOnline	POST
12	api/v1/reservation/reservation	http://10.97.111.12:8181/BE/reservation/reservation	http://10.97.16.21:8080/BE/reservation/reservation	POST
13	api/v1/reservation/details	http://10.97.111.12:8181/BE/reservation/getDetails	http://10.97.16.21:8080/BE/reservation/getDetails	POST
14	api/v1/reservation/cancellation	http://10.97.111.12:8181/BE/reservation/cancellation	http://10.97.16.21:8080/BE/reservation/cancellation	POST
15	api/v1/reservation/rescheduling	http://10.97.111.12:8181/BE/reservation/rescheduling	http://10.97.16.21:8080/BE/reservation/rescheduling	GET
16	api/v1/reservation/available-by-customer	http://10.97.111.12:8181/BE/reservation/availableByCustomer	http://10.97.16.21:8080/BE/reservation/availableByCustomer	GET
17	api/v1/reservation/details-by-customer	http://10.97.111.12:8181/BE/reservation/getDetailsByCustomer	http://10.97.16.21:8080/BE/reservation/getDetailsByCustomer	POST
Pre-registration				
1	/token	There is no direct url	There is no direct url	
2	/api/gbm/GetPlateTypes	http://10.97.111.12:8181/BE/plate	http://10.97.16.21:8080/BE/plate	GET
3	/api/gbm/GetVehicleCategories	No direct URL	No direct URL	
4	/api/gbm/GetVehicleDetailsByVIN	No direct URL	No direct URL	
5	/api/gbm/GetVehicleDetails	No direct URL	No direct URL	
6	/api/gbm/IsOwner	http://10.97.111.12:8181/BE/car/isOwner	http://10.97.16.21:8080/BE/car/isOwner	GET
7	/api/gbm/GetInspectionDetails	http://10.97.111.12:8181/BE/inspection/listDetails	http://10.97.16.21:8080/BE/inspection/listDetails	GET
8	/api/gbm/GetInspectionDefectDetails	http://10.97.111.12:8181/BE/inspection/details	http://10.97.16.21:8080/BE/inspection/details	GET

9	/api/gbm/GetFee	http://10.97.111.12:8181/BE/inspection/fee	http://10.97.16.21:8080/BE/inspection/fee	GET
10	/api/gbm/SavePreRegistration	No direct URL Call-back URL : https://mobileuat1.woqod.com/SWoqod/api/adapters/woqodetopup/prtransactionlog/success	No direct URL	
11	/api/gbm/IsOwnerAddToPage	http://10.97.111.12:8181/BE/car/isOwnerAddCar	http://10.97.16.21:8080/BE/car/isOwnerAddCar	GET

7.1.2 API Request & Response

7.1.2.1 Authentication

This API returns a Bearer Token to access the other service APIs.

Request

Method	URL
POST	api/v1/account/gbm

Response

Status Code	Output Values
200	<pre>{ "data": { "jwtToken": string } }</pre>
401	<pre>{ "request": object, "succeeded": bool, "message": "Invalid UserName/Password.", "errors": null, "data": null }</pre>

7.1.2.2 Plate Types with vehicle Shape Codes

This API returns a list of plate types which vehicle shapes.

Request

Method	URL
GET	api/v1/reservation/plate-type-with-shape-code

Type	Parameter Name	Parameter Type	Comments
------	----------------	----------------	----------

Header	authorization	string	<bearer <access_token>>
--------	---------------	--------	-------------------------

Response

Status Code	Output Values
200	<pre>{ "data": [{ "plateTypeId": int, "plateTypeNameEn": string, "plateTypeNameAr": string, "vehicleShapes": [{ "id": int, "nameEn": string, "nameAr": string }] }] }</pre>

7.1.2.3 Vehicle Details

This API returns the vehicle Details.

Request

Method	URL
GET	api/v1/reservation/vehicle-details

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Query	customerId	int	
	plateNumber	string	
	plateTypeId	int	

Response

Status Code	Output Values
200	<pre>{ "request": null, "succeeded": true, "message": null, "errors": null, "data": { "registrationExpiresOn": date, // yyyy-mm-dd } }</pre>

	<pre> "expiryStatus": int, // 0 = Already expired // 1 = Expiring soon, // 2 = Expiring after 30+ days "vin": string } } </pre>
400	<pre> { "data": { "errorId": "", "errorEn": "Invalid Vehicle Owner.", "errorAr": "يرجى اعادة ادخال البيانات الصحيحة. رقم اللوحة و بيانات المالك غير متطابقتين" } } </pre>
404	<pre> { "data": { "errorId": "", "errorEn": "Vehicle details not found in Mol.", "errorAr": "" } } </pre>

7.1.2.4 Vehicle Registration Validity

This API returns the Vehicle Registration Validity Status. If "shouldGetUserConfirmationToProceed" is TRUE, prompt to be shown and get the confirmation from the user before proceeding with the Reservation flow.

Request

Method	URL
GET	api/v1/reservation/vehicle-registration-validity

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Query	customerId	int	
	plateNumber	string	
	plateTypeId	int	

Response

Status Code	Output Values
200	<pre> { "request": null, "succeeded": true, "message": null, "errors": null, "data": { "shouldGetUserConfirmationToProceed": bool, "confirmationPromptEn": string, "confirmationPromptAr": string, </pre>

	<pre> } } </pre>
400	<pre> { "data": { "errorId": "", "errorEn": "Invalid Vehicle Owner.", "errorAr": "يرجى اعادة ادخال البيانات الصحيحة. رقم اللوحة و بيانات المالك غير متطابقتين" } } </pre>
400	<pre> { "data": { "errorId": "", "errorEn": "Reservation Open.", "errorAr": "يوم 30 صلاحية استمارة هذه السيارة سارية لأكثر من " } } </pre>
404	<pre> { "data": { "errorId": "", "errorEn": "Vehicle details not found in Mol.", "errorAr": "" } } </pre>

7.1.2.5 Pre-Reservation Reference

This API returns the Pre-Reservation Reference ID which should be pass along with the other Reservation calls.

Request

Method	URL
GET	api/v1/reservation/pre-reservation

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Query	customerId	int	
	mobileNumber	string	
	plateNumber	string	
	plateTypeId	int	
	vehicleShapeId	int	

Response

Status Code	Output Values
200	<pre> { "request": null, "succeeded": true, </pre>

	<pre> "message": null, "errors": null, "data": { "preReservationReferenceId": long, "otpRetriesLeftCount": int } } </pre>
400	<pre> { "data": { "errorId": "", "errorEn": "Invalid Vehicle Owner.", "errorAr": "يرجى اعادة ادخال البيانات الصحيحة. رقم اللوحة و بيانات المالك غير متطابقتين" } } </pre>
400	<pre> { "data": { "errorId": "", "errorEn": "Reservation Open.", "errorAr": "يوم 30 صلاحية استمارة هذه السيارة سارية لأكثر من" } } </pre>
404	<pre> { "data": { "errorId": "", "errorEn": "Vehicle details not found in Mol.", "errorAr": "" } } </pre>

7.1.2.6 Pre-Reservation Reference

This API validate the OTP.

Request

Method	URL
POST	api/v1/reservation/otp-validation

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Query	preReservationReferenceId	long	Pre-Reservation Reference ID
	otp	int	6 digits long (range 100000-999999)

Response

Status Code	Output Values
200	<pre> { "data": { </pre>

	<pre>"validated": bool } }</pre>
--	----------------------------------

7.1.2.7 Resend OTP

Request

Method	URL
PUT	api/v1/reservation/otp

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Body	preReservationReferenceId	long	Pre-Reservation Reference ID

Response

Status Code	Output Values
200	<pre>{ "data": { "sms-sent": true } }</pre>
400	<pre>{ "data": { "errorId": "", "errorEn": "No OTP Retries Left.", "errorAr": "" } }</pre>

7.1.2.8 Available Stations

This API returns all the available stations for a given vehicle.

Request

Method	URL
GET	api/v1/reservation/stations

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Query	preReservationReferenceId	long	Pre-Reservation Reference ID

Response

Status Code	Output Values
-------------	---------------

200	<pre>{ "data": [{ "stationId": int, "stationNameEn": string, "stationNameAr": string }] }</pre>
400	<pre>{ "data": { "errorId": "", "errorEn": "OTP is not validated.", "errorAr": "" } }</pre>

7.1.2.9 Available Appointment Dates

This API returns a list of all the available appointment dates for a given station.

Request

Method	URL
GET	api/v1/reservation/appointment-dates

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Query	preReservationReferenceId	long	Pre-Reservation Reference ID
	stationId	int	

Response

Status Code	Output Values
200	<pre>{ "data": [{ "appointmentDate": date, // yyyy-mm-dd }] }</pre>
400	<pre>{ "data": { "errorId": "", "errorEn": "OTP is not validated.", "errorAr": "" } }</pre>

7.1.2.10 Available Slots

This API returns a list of available slots for a given station & date. If the date is current date, this will return +1 hour from the current time.

Request

Method	URL
GET	api/v1/reservation/slots

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Query	preReservationReferenceId	long	
	stationId	int	
	appointmentDate	Date	In Ansi Format (YYYY-MM-DD)

Response

Status Code	Output Values
200	<pre>{ "data": [{ "slotId": long, "hourFrom": int, // hour in 24 hours format "hourTo": int, // hour in 24 hours format "slotTimeEn": string, "slotTimeAr": string }] }</pre>
400	<pre>{ "data": { "errorId": "", "errorEn": "OTP is not validated.", "errorAr": "" } }</pre>

7.1.2.11 Can Pay Online

This API returns the details of the appointment and a flag indicating whether the Online Payment can be made or not.

Request

Method	URL
GET	api/v1/reservation/can-pay-online

Type	Parameter Name	Parameter Type	Comments
------	----------------	----------------	----------

Header	authorization	string	<bearer <access_token>>
Query	preReservationReferenceId	long	
	slotId	long	

Response

Status Code	Output Values
200	<pre>{ "data": { "canPayOnline": bool, "stationNameEn": string, "stationNameAr": string, "appointmentDate": date, // yyyy-mm-dd "hourFrom": int, // hour in 24 hours format "hourTo": int, // hour in 24 hours format "slotTimeEn": string, "slotTimeAr": string } }</pre>
400	<pre>{ "data": { "errorId": "", "errorEn": "OTP is not validated.", "errorAr": "" } }</pre>

7.1.2.12 Save the Reservation

This API saves the reservation and returns the reservation Id.

Request

Method	URL
POST	api/v1/reservation/reservation

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Body	preReservationReferenceId	long	Pre-Reservation Reference ID
	slotId	long	

Response

Status Code	Output Values
200	<pre>{ "data": { "reservationId": long, "vin": string, } }</pre>

	<pre> "licencePlate": string, "licencePlateTypeId": int, "vehicleShapeId": int, "pid": string, "mobileNumber": string, "stationNameEn": string, "stationNameAr": string, "appointmentDate": date, // yyyy-mm-dd "hourFrom": int, // hour in 24 hours format "hourTo": int, // hour in 24 hours format "canPayOnline": bool, "slotTimeEn": string, "slotTimeAr": string } </pre>
400	<pre> { "data": { "errorId": "", "errorEn": "OTP is not validated.", "errorAr": "" } } </pre>
403	<pre> { "data": { "errorId": "", "errorEn": "This endpoint was called more than once.", "errorAr": "" } } </pre>

7.1.2.13 Get Reservation Details

This API returns the reservation details to do cancellation/re-scheduling.

Request

Method	URL
GET	api/v1/reservation/details

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Query	customerId	int	
	mobileNumber	string	
	plateNumber	string	
	plateTypeId	int	

Response

Status Code	Output Values
-------------	---------------

200	<pre>{ "data": { "reservationId": long, "stationNameEn": string, "stationNameAr": string, "appointmentDate": date, // yyyy-mm-dd "hourFrom": int, // hour in 24 hours format "hourTo": int, // hour in 24 hours format "slotTimeEn": string, "slotTimeAr": string } }</pre>
404	<pre>{ "data": { "errorId": "", "errorEn": "Reservation not found.", "errorAr": "" } }</pre>

7.1.2.14 Cancellation

This API cancel a given reservation.

Request

Method	URL
POST	api/v1/reservation/cancellation

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Body	reservationId	long	Reservation Reference ID

Response

Status Code	Output Values
200	<pre>{ "data": { "cancelled ": bool, } }</pre>

7.1.2.15 Re-scheduling

This API returns a new Pre-Reservation ID. Available Station (api/v1/reservation/stations) call should be made after this one and continue the sequence to complete the Reservation.

Request

Method	URL
GET	api/v1/reservation/rescheduling

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Body	reservationId	long	Reservation Reference ID

Response

Status Code	Output Values
200	<pre>{ "data": { "preReservationReferenceId": long } }</pre>

7.1.2.16 Reservation Availability

This API returns any reservation available for a given customer.

Request

Method	URL
GET	api/v1/reservation/available-by-customer

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Query	customerId	String	

Response

Status Code	Output Values
200	<pre>{ "data": { "reservationAvailable": bool } }</pre>

7.1.2.17 Reservation Details

This API returns a list of all the available reservation details for a given customer.

Request

Method	URL
--------	-----

GET	api/v1/reservation/details-by-customer
-----	--------------------------------------------------------

Type	Parameter Name	Parameter Type	Comments
Header	authorization	string	<bearer <access_token>>
Query	customerId	String	

Response

Status Code	Output Values
200	<pre>{ "data": [{ "reservationId": long, "vin": string, "plateNumber": string, "plateTypeNameEn": string, "plateTypeNameAr": string, "customerId": string, "mobileNumber": string, "stationNameEn": string, "stationNameAr": string, "appointmentDate": date, // yyyy-mm-dd "hourFrom": int, // hour in 24 hours format "hourTo": int, // hour in 24 hours format "slotTimeEn": string, "slotTimeAr": string }] }</pre>

7.2 RPS WOQODe System

7.2.1 Integration URLs

WOQODe API URL	MFP API(UAT)	MFP API(PROD)	Http Method
RPSIntegrationService.asmx Action : BalanceInquiry	https://mobileuat1.woqod.com/Swoqod/api/adapters/RspServices/BalanceInquiry	https://mobileapp1.woqod.com/newwoqod/api/adapters/RspServices/BalanceInquiry	POST
RPSIntegrationService.asmx Action : BalanceUpdate	https://mobileuat1.woqod.com/Swoqod/api/adapters/woqodetopup/success	https://mobileapp1.woqod.com/newwoqod/api/adapters/woqodetopup/success	POST
wsmobileapp/rps/woqode.asmx Action: Get_Customer_ByFleetName	https://mobileuat1.woqod.com/Swoqod/api/adapters/RpsAdapterJava/getCustomerByFleetName	https://mobileapp1.woqod.com/newwoqod/api/adapters/RpsAdapterJava/getCustomerByFleetName	POST

wsmobileapp/rps/woqode.asmx Action: Get SalesTransactions	https://mobileuat1.woqod.com/SWoqod/api/adapters/RpsAdapterJava/getSaleTransactions	https://mobileapp1.woqod.com/newwoqod/api/adapters/RpsAdapterJava/getSaleTransactions	POST
------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------

7.2.2 API Request & Response

7.2.2.1 Get Balance (Balance Inquiry)

The API "Get Balance" provides information about WOQODE account (FleetID, CivilName, Balance...).

API link: <http://10.99.104.191:85/RPSIntegrationService.asmx>

Request

Method (POST or GET)	URL
POST	RPSIntegrationService.asmx

	Parameter Name	Parameter Type	value
Body	username	string	AuthName
	password	string	AuthPass

Type	Parameter Name	Parameter Type	Comments
body	contentType	text/xml; charset=utf-8	
	content	string	<p>Example:</p> <pre>'<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:tem="http://tempuri.org/">' + '<soap:Header>' + '<tem:AuthInfo>' + '<tem:AuthName>' + AuthName + '</tem:AuthName>' + '<tem:AuthPass>' + AuthPass + '</tem:AuthPass>' + '</tem:AuthInfo>' + '</soap:Header>' + '<soap:Body>' + '<tem:AccountInquiry>' + '<tem:CivilID>' + CivilID + '</tem:CivilID>' + '<tem:CivilMobileNumber>' + CivilMobileNumber + '</tem:CivilMobileNumber>' + '<tem:TransactionID>' + TransactionID + '</tem:TransactionID>' + '<tem:Origin>' + Origin + '</tem:Origin>' + '<tem:ChannelID>' + ChannelID + '</tem:ChannelID>' + '</tem:AccountInquiry>' + '</soap:Body>' + '</soap:Envelope>'</pre>

Response

Status Code	Parameter Name	Parameter Type	Comments
-------------	----------------	----------------	----------

200	JSON	<p>Exemple success:</p> <pre>{ "statusReason": "OK", "responseHeaders": { "Cache-Control": "private, max-age=0", "Server": "Microsoft-IIS/8.5", "X-AspNet-Version": "4.0.30319", "Content-Length": "557", "Date": "Tue, 29 Sep 2020 08:49:53 GMT", "Content-Type": "application/soap+xml; charset=utf-8", "X-Powered-By": "ASP.NET" }, "isSuccessful": true, "responseTime": 1647, "totalTime": 1648, "warnings": [], "Envelope": { "": "http://www.w3.org/2001/XMLSchema", "Body": { "AccountInquiryResponse": { "": "http://tempuri.org/", "AccountInquiryResult": { "status": 0, "FleetID": "9786", "CivilName": "28278800821- Test App", "ResponseGeneral": { "ResponseCode": "1", "ResponseMessage": "9786" }, "Balance": "0" } } } }, "errors": [], "info": [], "statusCode": 200 }</pre>
-----	------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

			<p>Example fail:</p> <pre>{ "statusReason": "OK", "responseHeaders": { "Cache-Control": "private, max-age=0", "Server": "Microsoft-IIS/8.5", "X-AspNet-Version": "4.0.30319", "Content-Length": "581", "Date": "Tue, 03 Nov 2020 13:57:23 GMT", "Content-Type": "application/soap+xml; charset=utf-8", "X-Powered-By": "ASP.NET" }, "isSuccessful": true, "responseTime": 458, "totalTime": 459, "warnings": [], "Envelope": { "": "http://www.w3.org/2001/XMLSchema", "Body": { "AccountInquiryResponse": { "": "http://tempuri.org/", "AccountInquiryResult": { "FleetID": "0", "CivilName": "", "ResponseGeneral": { "ResponseCode": "0", "ResponseMessage": "Civil (28578800059) with (50064646) phone number is not valid" } }, "Balance": "0" } } }, "errors": [], "info": [], "statusCode": 200 }</pre>
--	--	--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

7.2.2.2 Balance Update

The API “Update Balance” updates the balance of a WOQODE account.

API link: <http://10.99.104.191:85/RPSIntegrationService.asmx>

Request

Method (POST or GET)	URL
POST	RPSIntegrationService.asmx

	Parameter Name	Parameter Type	value
Body	username	string	AuthName
	password	string	AuthPass

Type	Parameter Name	Parameter Type	Comments
	contentType	text/xml; charset=utf-8	

body	content	string	<p>Exemple :</p> <pre> <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"> <soap:Header> <AuthInfo xmlns="http://tempuri.org/"> <AuthName>' + AuthName + '</AuthName> <AuthPass>' + AuthPass + '</AuthPass> </AuthInfo> <wsa:Action>http://tempuri.org/BalanceUpdate"</wsa:Action> <wsa:MessageID>urn:uuid:2c16293d-807e-4a69-95a2-6267b7c79a89</wsa:MessageID> <wsa:ReplyTo> <wsa:Address>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:Address> </wsa:ReplyTo> <wsa:To>http://10.99.108.109:84/RPSIntegrationService.asmx</wsa:To> <wsse:Security soap:mustUnderstand="1"> <wsu:Timestamp wsu:Id="Timestamp-676008ec-22a5-4bd9-8de3-47fda0824d0c"> <wsu:Created>2022-07-01T13:34:00Z</wsu:Created> <wsu:Expires>2022-07-01T13:39:00Z</wsu:Expires> </wsu:Timestamp> <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="SecurityToken-e48a25da-efd9-4371-815c-acab46bc8f30"> <wsse:Username>V1NTRUNXT1FPRA==</wsse:Username> <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordDigest">x0lIBr+GILxI04aHelXZZNYissl=</wsse:Password> <wsse:Nonce>q4j9LP3gI9QiDEL5HqHEew==</wsse:Nonce> <wsu:Created>2022-07-01T13:34:00Z</wsu:Created> </wsse:UsernameToken> </wsse:Security> </soap:Header><soap:Body><BalanceUpdate xmlns="http://tempuri.org/"> <CivilID>27899900031</CivilID><CivilMobileNumber>55000000</CivilMobileNumber><BalanceIncreaseValue>23.0</BalanceIncreaseValue><TransactionID>1656682388056bcbcb7b</TransactionID><Origin>WOQ-MOBILE</Origin><ChannelID>WOQ-MOBILE</ChannelID></BalanceUpdate></soap:Body> </soap:Envelope> </pre>
------	---------	--------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Response

Status Code	Parameter Name	Parameter Type	Comments
200		JSON	Exemple success:

			<p>Exemple fail:</p> <pre><?xml version="1.0" encoding="utf-8"?><soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"> <soap:Header><wsa:Action>http://schemas.xmlsoap.org/ws/2004/08/addressing/fault</wsa:Action> <wsa:MessageID>urn:uuid:623a2267-5717-4a04-8678-d992b4de58ab</wsa:MessageID><wsa:RelatesTo>urn:uuid:9da3affb-59cc-4cb8-9d16-e90d90c3ab61</wsa:RelatesTo> <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To><wsse:Security><wsu:Timestamp wsu:Id="Timestamp-2916104a-f9e7-4759-bf48-05b5be642430"> <wsu:Created>2022-10-10T09:42:09Z</wsu:Created><wsu:Expires>2022-10-10T09:47:09Z</wsu:Expires></wsu:Timestamp> </wsse:Security></soap:Header><soap:Body><soap:Fault><faultcode>soap:Server</faultcode> <faultstring>Server was unable to process request. ---&gt; ERROR [GetValidateToBalanceUpdate] : Execution Timeout Expired. The timeout period elapsed prior to completion of the operation or the server is not responding. State: SELECT RID FROM dbo.VehicleDetailsWS_ReqRes with(nolock) WHERE TransactionID ='3D7A779A-36A3-482A-8670-B1119DA8FDED' </faultstring><detail/></soap:Fault></soap:Body></soap:Envelope></pre>
--	--	--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

7.2.2.3 Get Customer by fleet name

The API "Get Customer by Fleet Name" provides information about WOQOD customer by fleet name.

API link: wsmobileapp/rps/woqode.asmx

Action: Get_Customer_ByFleetName

Request

Method (POST or GET)	URL
POST	wsmobileapp/rps/woqode.asmx

	Parameter Name	Parameter Type	value
Body	sFleetName	string	Fleet name

Type	Parameter Name	Parameter Type	Comments
	contentType	text/xml; charset=utf-8	

body	content	string	Exemple: <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:woqv="http://woqv-web-services/"> <soapenv:Header/> <soapenv:Body> <woqv:Get_Customer_ByFleetName> <!--Optional:--> <woqv:sFleetName>27899900031</woqv:sFleetName> </woqv:Get_Customer_ByFleetName> </soapenv:Body> </soapenv:Envelope>
------	---------	--------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Response

Status Code	Parameter Name	Parameter Type	Comments
200		JSON	Exemple success: { "isSuccessful": true, "header": { "statusCode": "000" }, "body": { "result": [{ "fleetName": "27899900031", "vustomerName": "WOQODe INDIVIDUALS PREPAID", "customerCode": "12348", "fowOrder": "0", "fleetID": "73504", "froupID": "42013", "fleetCode": "027572", "groupName": "SHEREEN TAHA AL-ATTAR ", "licensePlateNr": "01566926", "customerID": "515", "vehicleID": "196297", "customerORACLE_ID": "1890225", "id": "TCustomer1", "groupCode": "037315" }] } }

7.2.2.4 Get Sales Transactions

The API "Get Sales Transactions" provides information about sales transaction history.

API link: wsmobileapp/rps/woqode.asmx

Action: Get_SalesTransactions

Request

Method (POST or GET)	URL
POST	wsmobileapp/rps/woqode.asmx

	Parameter Name	Parameter Type	value
Body	sPlateNumber	string	Plate number

Type	Parameter Name	Parameter Type	Comments
	contentType	text/xml; charset=utf-8	

body	content	string	<p>Exemple:</p> <pre><soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:woqv="http://woqv-web-services/"> <soapenv:Header/> <soapenv:Body> <woqv:Get_SalesTransactions> <!--Optional:--> <woqv:sPlateNumber>trfyguijo</woqv:sPlateNumber> </woqv:Get_SalesTransactions> </soapenv:Body> </soapenv:Envelope></pre>
------	---------	--------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Response

Status Code	Parameter Name	Parameter Type	Comments
200		JSON	<p>Exemple success:</p> <pre>{ "isSuccessful": true, "header": { "statusCode": "000" }, "body": { "result": [{ "fleetName": "27899900031", "stationCode": "000928", "total": "164.5400", "licensePlateNr": "01566926", "saleEnd": "2023-01-15T17:52:17+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "64724716", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "149.1200", "licensePlateNr": "01566926", "saleEnd": "2023-01-07T10:06:31+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "64402362", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "146.3700", "licensePlateNr": "01566926", "saleEnd": "2022-12-29T16:30:32+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "64094572", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "131.0400", "licensePlateNr": "01566926", "saleEnd": "2022-12-17T23:58:31+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "63655313", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }] } }</pre>

			<pre> "fleetName": "27899900031", "stationCode": "000928", "total": "132.3000", "licensePlateNr": "01566926", "saleEnd": "2022-12-11T19:55:01+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "63408260", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "152.7100", "licensePlateNr": "01566926", "saleEnd": "2022-12-03T23:16:05+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "63077935", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000927", "total": "165.6300", "licensePlateNr": "01566926", "saleEnd": "2022-11-22T12:59:47+03:00", "customerCode": "12348", "stationName": "Hazm Al Markhiya", "customerORACLE_ID": "1890225", "recID": "62591938", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "122.2400", "licensePlateNr": "01566926", "saleEnd": "2022-11-13T12:55:27+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "62185294", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "50.0000", "licensePlateNr": "01566926", "saleEnd": "2022-11-13T06:02:00+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "62166408", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "001117", "total": "131.4600", "licensePlateNr": "01566926", "saleEnd": "2022-11-02T15:53:37+03:00", "customerCode": "12348", "stationName": "Baaya", "customerORACLE_ID": "1890225", "recID": "61715883", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" } </pre>
--	--	--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

			<pre> "fleetName": "27899900031", "stationCode": "000928", "total": "157.1900", "licensePlateNr": "01566926", "saleEnd": "2022-10-24T20:58:58+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "61323183", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "134.1900", "licensePlateNr": "01566926", "saleEnd": "2022-10-13T15:01:01+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "60836413", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "111.4700", "licensePlateNr": "01566926", "saleEnd": "2022-10-07T17:27:10+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "60565049", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000174", "total": "163.5300", "licensePlateNr": "01566926", "saleEnd": "2022-09-28T09:16:44+03:00", "customerCode": "12348", "stationName": "Old Rayyan", "customerORACLE_ID": "1890225", "recID": "60154833", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "159.2400", "licensePlateNr": "01566926", "saleEnd": "2022-09-14T07:01:32+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "59537513", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "126.9700", "licensePlateNr": "01566926", "saleEnd": "2022-09-03T22:12:32+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "59075237", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { </pre>
--	--	--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

			<pre> "fleetName": "27899900031", "stationCode": "000928", "total": "165.2500", "licensePlateNr": "01566926", "saleEnd": "2022-08-25T20:58:05+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "58706110", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "158.0300", "licensePlateNr": "01566926", "saleEnd": "2022-08-14T06:33:51+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "58199780", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "133.4800", "licensePlateNr": "01566926", "saleEnd": "2022-08-01T20:44:34+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "57718899", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" }, { "fleetName": "27899900031", "stationCode": "000928", "total": "126.6500", "licensePlateNr": "01566926", "saleEnd": "2022-07-23T21:30:56+03:00", "customerCode": "12348", "stationName": "Al Aziziyah", "customerORACLE_ID": "1890225", "recID": "57375223", "customerName": "WOQODe INDIVIDUALS PREPAID", "fleetCode": "027572" } } } </pre>
--	--	--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

7.3 MOTC System

7.3.1 Description

The Ministry of Transport and Communications (MOTC) of Qatar works closely with the company WOQOD to integrate its mobile application with the country's transportation and communication systems. This system is used to verify user's details (QID and Mobile Number).

7.3.2 Integration URL

<https://stgapi.gdx.gov.qa:9443/gdx/prod/v1/CustomerValidationWS/validateCustomer>

7.3.3 MOTC API Request & Response

Request

Method (POST or GET)	URL
POST	https://stgapi.gdx.gov.qa:9443/gdx/prod/v1/CustomerValidationWS/validateCustomer

	Parameter Name	Parameter Type	value
Body	qid	string	QID
	mobileNo	string	Mobile number

Type	Parameter Name	Parameter Type	Comments
body	contentType	text/xml; charset=utf-8	
	content	string	Example: <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"> <SOAP-ENV:Header><wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" SOAP-ENV:mustUnderstand="1"><wsu:Timestamp wsu:Id="TS-d7c3d432-d90b-42a4-83c7-28a801e003a8"><wsu:Created>2022-11-18T09:45:03.965Z</wsu:Created><wsu:Expires>2022-11-18T09:50:03.965Z</wsu:Expires></wsu:Timestamp><wsse:UsernameToken wsu:Id="UsernameToken-6a2c6942-b11e-49bc-959b-60bc835273eb"><wsse:Username>WoqodStgUser</wsse:Username><wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">wydj@hdjs1254</wsse:Password></wsse:UsernameToken></wsse:Security></SOAP-ENV:Header><SOAP-ENV:Body><ns3:validateCustomer xmlns:ns3="urn:igov.gov.qa/customervalidation/inf/v1_0" xmlns:ns4="http://woqv-web-services/" xmlns:=""><validateCustomerRequest><QID>27899900031</QID><mobileNo>55865890</mobileNo></validateCustomerRequest></ns3:validateCustomer></SOAP-ENV:Body></SOAP-ENV:Envelope>

Response

Status Code	Parameter Name	Parameter Type	Comments
200		JSON	Exemple success: <NS1:Envelope xmlns:NS1="http://schemas.xmlsoap.org/soap/envelope/"><NS1:Body><NS2:validateCustomerResponse xmlns:NS2="urn:igov.gov.qa/customervalidation/inf/v1_0"><result><statusCode>0000</statusCode><statusMessage>SUCCESS</statusMessage></result></NS2:validateCustomerResponse></NS1:Body></NS1:Envelope>
112		JSON	Example failure: <NS1:Envelope xmlns:NS1="http://schemas.xmlsoap.org/soap/env

			<pre> <NS1:Body><NS2:validateCustomerResponse xmlns:NS2="urn:igov.gov.qa/customervalidation/in f/v1_0"><result><statusCode>1001</statusCode>< statusMessage>Invalid combination of QID and Mobile Number</statusMessage></result></NS2:validate CustomerResponse></NS1:Body></NS1:Envelope> </pre>
500		JSON	<pre> <NS1:Envelope xmlns:NS1="http://schemas.xmlsoap.org/soap/env elope/"><NS1:Body><NS2:validateCustomerRespo nse xmlns:NS2="urn:igov.gov.qa/customervalidation/in f/v1_0"><result><statusCode>1000</statusCode>< statusMessage>Technical Error while accessing Service Providers</statusMessage></result></NS2:validate CustomerResponse></NS1:Body></NS1:Envelope> </pre>

7.4 WOQOD API (“SharePoint”)

7.4.1 Description

WOQOD API ‘SharePoint’ is used to access and retrieve information we are displaying in the mobile application.

To access SharePoint data using SharePoint API, you need to:

1. Obtain the URL for the SharePoint site you want to access.
2. Make API calls to retrieve the data, such as promotions and petrol station information.
3. Parse the response from the API and display the information in your mobile application.

7.4.2 Integration URLs

These APIs are public and can be accessed directly by typing the SharePoint link into your navigator or by importing the below collection link to your postman:

[WOQOD API \(Sharepoint\).postman collection.json](#)

Data	BE URL	SharePoint URL
Petrol Stations	<u>Backend-Base-URL/</u> crons?category=STATION	https://www.woqod.com/_api/web/lists/GetByTitle('Service Stations')/items
Promotions	<u>Backend-Base-URL/</u> crons?category=PROMOTION	Promotions: https://www.woqod.com/_api/web/lists/GetByTitle('Promotions')/items Promotions item URL: https://www.woqod.com/_api/web/lists/GetByTitle('Promotions')/Items(itemNumber)/FieldValuesAsHtml
Petrol Prices	<u>Backend-Base-URL/</u> crons?category=PRICES	https://www.woqod.com/_api/web/lists/GetByTitle('Petrol Prices')/items
Retailers	<u>Backend-Base-URL/</u> crons?category=RETAILERS	https://www.woqod.com/_api/web/lists/GetByTitle('Retailers List')/items
Shafaf Supermarkets	<u>Backend-Base-URL/</u> crons?category=SUPERMARKET	https://www.woqod.com/_api/web/lists/GetByTitle('Buy Shafaf')/items
Fahes Stations	<u>Backend-Base-URL/</u> crons?category=FAHES	https://www.fahes.com.qa/_api/web/lists/GetByTitle('Service Stations')/items
Petrol Types	<u>Backend-Base-URL/</u> crons?category=PRICES	https://www.woqod.com/_api/web/lists/GetByTitle('Petrol types')/items
Tenders	<u>Backend-Base-URL/</u> crons?category=TENDERS	https://www.woqod.com/EN/Business/Tenders/_api/web/lists/GetByTitle('Pages')/items

7.5 Eurolandir

7.5.1 Description

Eurolandir is a standardized service, which is provided as a URL link. The query result can be in 3 formats: XML, CSV and JSON. Used for stock prices.

7.5.2 Integration URLs

Stock Prices	Backend-Base-URL/ crons?category=STOCKPRICES	https://tools.eurolandir.com/tools/pricefeed/?companycode=qa-qfls&format=json
---------------------	--------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

7.5.3 Stock Prices Eurolandir API Request & Response

Request

Method (POST or GET)	URL
GET	https://tools.eurolandir.com/tools/pricefeed/?companycode=qa-qfls&format=json

	Parameter Name	Parameter Type	value
Body	qid	string	QID
	mobileNo	string	Mobile number

Type	Parameter Name	Parameter Type	Comments
body	content Type	application/json	
	content	string	Example: <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"><SOAP-ENV:Header><wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" SOAP-ENV:mustUnderstand="1"><wsu:Timestamp wsu:id="TS-d7c3d432-d90b-42a4-83c7-28a801e003a8"><wsu:Created>2022-11-18T09:45:03.965Z</wsu:Created><wsu:Expires>2022-11-18T09:50:03.965Z</wsu:Expires></wsu:Timestamp><wsse:UsernameToken wsu:id="UsernameToken-6a2c6942-b11e-49bc-959b-60bc835273eb"><wsse:Username>WoqodStgUser</wsse:Username><wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">wydj@hdjs1254</wsse:Password></wsse:UsernameToken></wsse:Security></SOAP-ENV:Header><SOAP-ENV:Body><ns3:validateCustomer xmlns:ns3="urn:igov.gov.qa/customervalidation/inf/v1_0" xmlns:ns4="http://woqv-web-services/" xmlns=""><validateCustomerRequest><QID>27899900031</QID><mobileNo>55865890</mobileNo></validateCustomerRequest></ns3:validateCustomer></SOAP-ENV:Body></SOAP-ENV:Envelope>

Response

Status Code	Parameter Name	Parameter Type	Comments
200		JSON	<p>Exemple success:</p> <pre><200,{Qatar Fuel=SharePointStockPricesResponse(currency=QAR, bid=17.8400, ask=17.8600, change=-0.1100, changePercent=-0.6128, last=17.8400, high=17.9200, low=17.8000, volume=572126, previousClose=17.9500, iSIN=QA0001200771, symbol=QFLS, marketName=Qatar, noShares=994255760, marketCap=17737522758, open=17.9200, date=2023-02-07 13:30:30)},[cache-control:"private, max-age=60", content-type:"application/json; charset=utf-8", expires:"Tue, 07 Feb 2023 10:45:07 GMT", last-modified:"Tue, 07 Feb 2023 10:44:07 GMT", vary:"*", access-control-allow-origin:"*", access-control-allow-headers:"Content-Type", access-control-allow-methods:"GET,POST,OPTIONS", x-xss-protection:"1; mode=block", strict-transport-security:"max-age=15552000", public-key-pins:"max-age=30; pin-sha256="/+lEr3O32RTitEm7vaWdldE1/ln2DYs4CyT5wBB2qc="; pin-sha256="J44drAHqhdb4p5IHsd1cT/Trk/vZjly8y6ajSis5Hhc="; pin-sha256="Ok5YY3wU3OgqPbkRzNxbfAzOwi7Tp5Zn8kk8EA31tk0="; report-uri="http://tools.euroland.com/tools/pkp-report/default.aspx""], date:"Tue, 07 Feb 2023 10:44:06 GMT", content-length:"376"]></pre>

Request

Method	URL
GET	https://tools.eurolandir.com/tools/pricefeed/?companycode=qa-qfls&format=json

Type	Value
Header	<[Content-Type:"application/json", Accept:"application/json;odata=verbose;charset=UTF-8", User-Agent:"Chrome/58.0.3029.81"]>
Body	null

Response

Status Code	Output Values
200	{Qatar Fuel=SharePointStockPricesResponse(currency=QAR, bid=17.8400, ask=17.8600, change=-0.1100, changePercent=-0.6128, last=17.8400, high=17.9200, low=17.8000, volume=572126, previousClose=17.9500, iSIN=QA0001200771, symbol=QFLS, marketName=Qatar, noShares=994255760, marketCap=17737522758, open=17.9200, date=2023-02-07 13:30:30)}