

High Level Document

**User Management**

# Introduction

* + Overview
* User Management API to manage user operations.
* Objective
* Provides HLD for User Management API.
* In scope
* JUnit partial coverage
* Security partial coverage
* Exception Handling partial coverage
* Containerization
* Out of scope
* JUnit full coverage
* Testing Metrics
* Security full coverage
* Logging

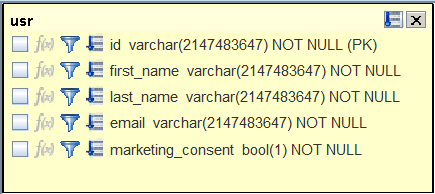
# Used Technologies

* Web
* Spring Rest
* Spring Data
* Spring Boot
* Spring Security
* Web Server
* Spring Boot Embedded Tomcat
* Database
* PostgreSQL
* Source Control
* GitHub
* SW Project Management Tool
* Maven
* Unit Testing Framework
* Spring Test
* Containerization
  + Docker

# Source Code

https://github.com/AbdElHalimHeba/SaqayaRepo.git

# ER Diagram



# Running Instructions

* Open bash
* Create a shared network between containers:

docker network create spring-postgres

* Change directory to user-api\src\main\docker\db to build PostgreSQL DB image:

docker build -t postgres\_db .

* Run PostgreSQL DB container:

docker run -d --network spring-postgres -p 5432:5432 --name db postgres\_db

* Change directory to user-api to package Spring Boot application & copy jar from target to src/main/docker:

mvnw clean package

cp target/user-api-0.0.1-SNAPSHOT.jar src/main/docker

* Change directory to user-api\src\main\docker to build Spring Boot application image:

docker build -t system-design .

* Run Spring Boot application container:

docker run -d --network spring-postgres -p 8080:8080 --name app user-api