



Adventist University of Central Africa

P.O. Box 2461 Kigali, Rwanda | www.auca.ac.rw | info@auca.ac.rw

PURCHASE REQUEST APPLICATION



WEB TECHNOLOGY PROJECT



Done By

NSHUTI Kenny Herve, # 22979

Major in

Software Engineering

December 2023

PURCHASE REQUEST APPLICATION

Purchase request application is application that enables employees to submit purchase requests for goods or services they need to procure for their organization. It typically involves a workflow-based system where employees fill out electronic forms with details of their purchase requests, such as the item or service needed, quantity, price, budget code, and other relevant information.

Once a purchase request is submitted, it goes through an automated approval process where designated personnel, such as managers or procurement officers, review and approve or reject the request.

Function Requirement

- The system should allow the user to register into our system with First Names, Last Name Email and Password.
- The system must allow users to login into their accounts by entering their Email and Password.
- The system must allow Admin to login into he/she account by entering their email and password and Landing on user.
- The system should allow the employee to register the new request update and delete .
- The user(customer) should view the different request which was sent to system.

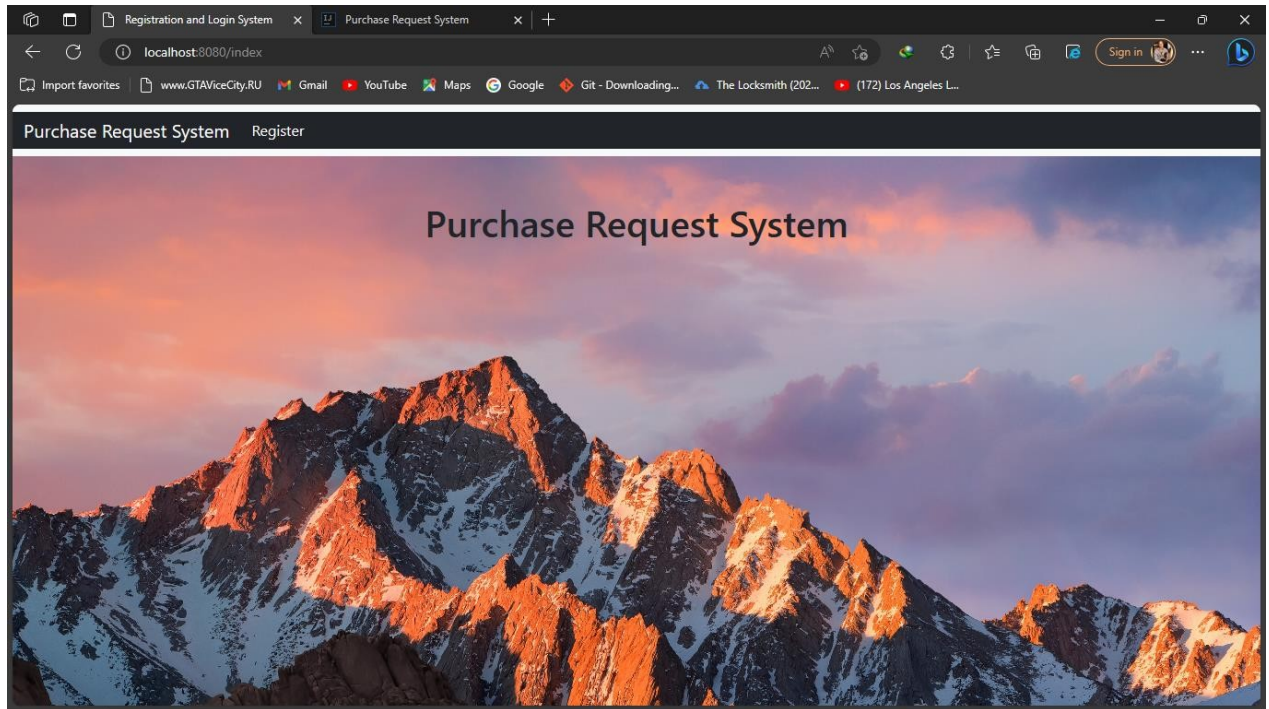
Non-Function Requirement

- The system must perform user request within 10 sec
- The system should not exceed 20 secs in case of downtime
- The user request will not exceed 2 clicks maximum to be completed
- The system must be scalable enough to support 100,000 citizens at the same time while maintaining optimal performance
- The system shall be accessible on laptops/tablets/mobile phones behavior
- The mean time to restore the system (MTTRS) following a failure must not be greater 10 minutes.
- All data inside the system shall be protected against malware attacks or unauthorized access

Technology Used

- Spring Security
- Thymleaf
- Mysql
- Spring boot
- IntelliJ IDE

Home Page: Localhost:8080/index



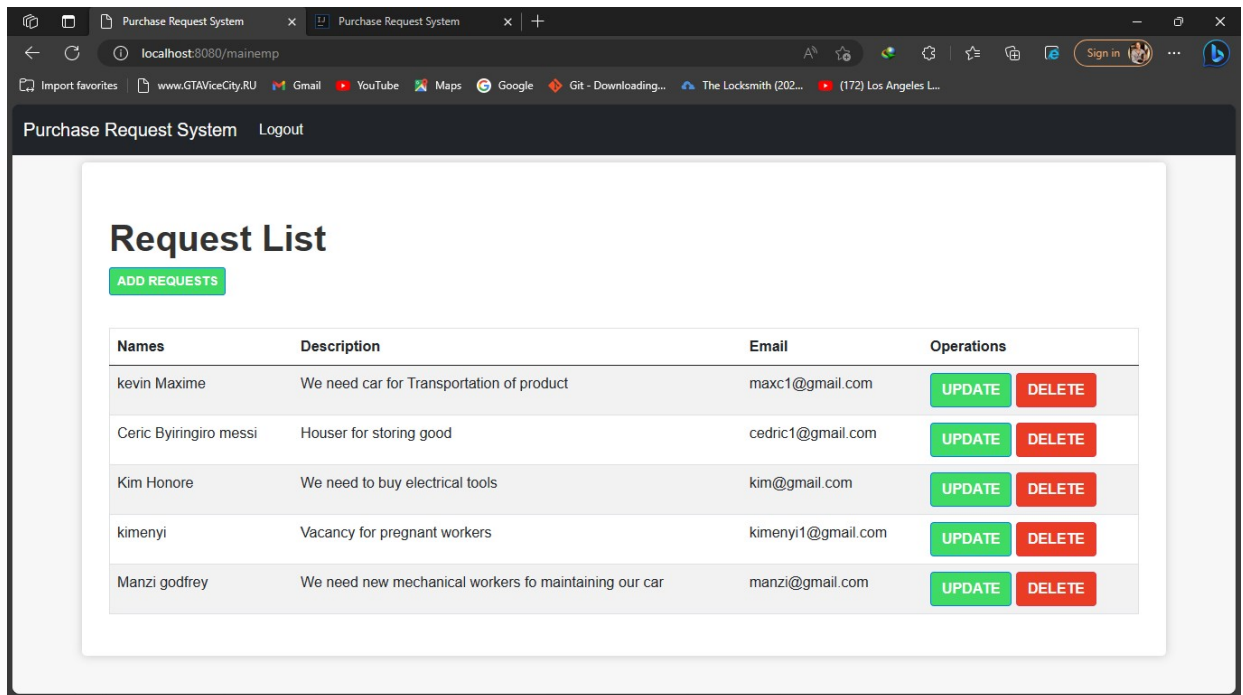
Registration

The screenshot shows a web browser window with two tabs, both titled "Purchase Request System". The address bar shows "localhost:8080/register". The browser's toolbar includes various icons and a "Sign in" button. The page header is "Purchase Request System Login". The main content area features a "Registration" form with the following fields: "First Name" (placeholder: "Enter first name"), "Last Name" (placeholder: "Enter last name"), "Email" (value: "kimenyi1@gmail.com"), and "Password" (placeholder: "*****"). Below the password field is a blue "Register" button and a link that says "Already registered? [Login here](#)".

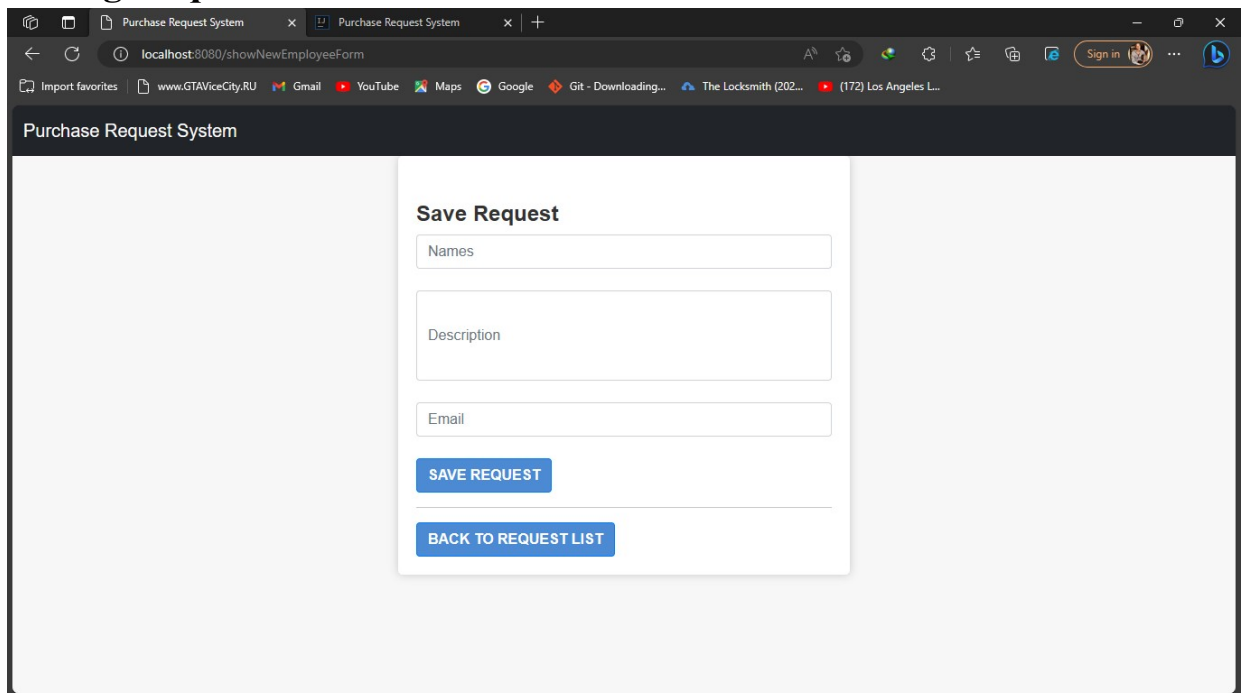
Login

The screenshot shows a web browser window with two tabs, both titled "Purchase Request System". The address bar shows "localhost:8080/login". The browser's toolbar includes various icons and a "Sign in" button. The page header is "Purchase Request System Register". The main content area features a "Login Form" with the following fields: "Email" (value: "kimenyi1@gmail.com") and "Password" (placeholder: "*****"). Below the password field is a blue "Submit" button and a link that says "Not registered ? [Register/Signup here](#)".

Dashboard of User



Saving Request



Updating request

Employee Management System x Purchase Request System x +

localhost:8080/showFormForUpdate/2

Purchase Request System

Update Employee

kevin Maxime

We need car for Transportation of product

maxc1@gmail.com

UPDATE EMPLOYEE

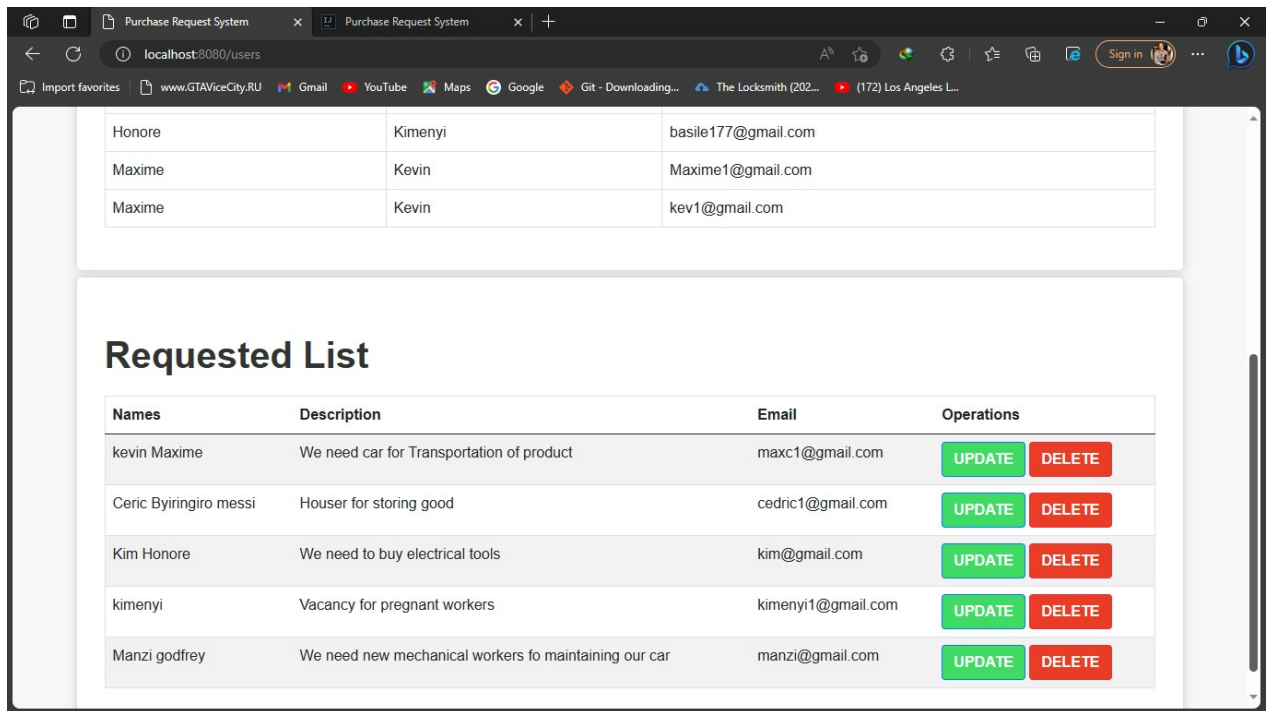
BACK TO EMPLOYEE LIST

Dashboard of The Admin

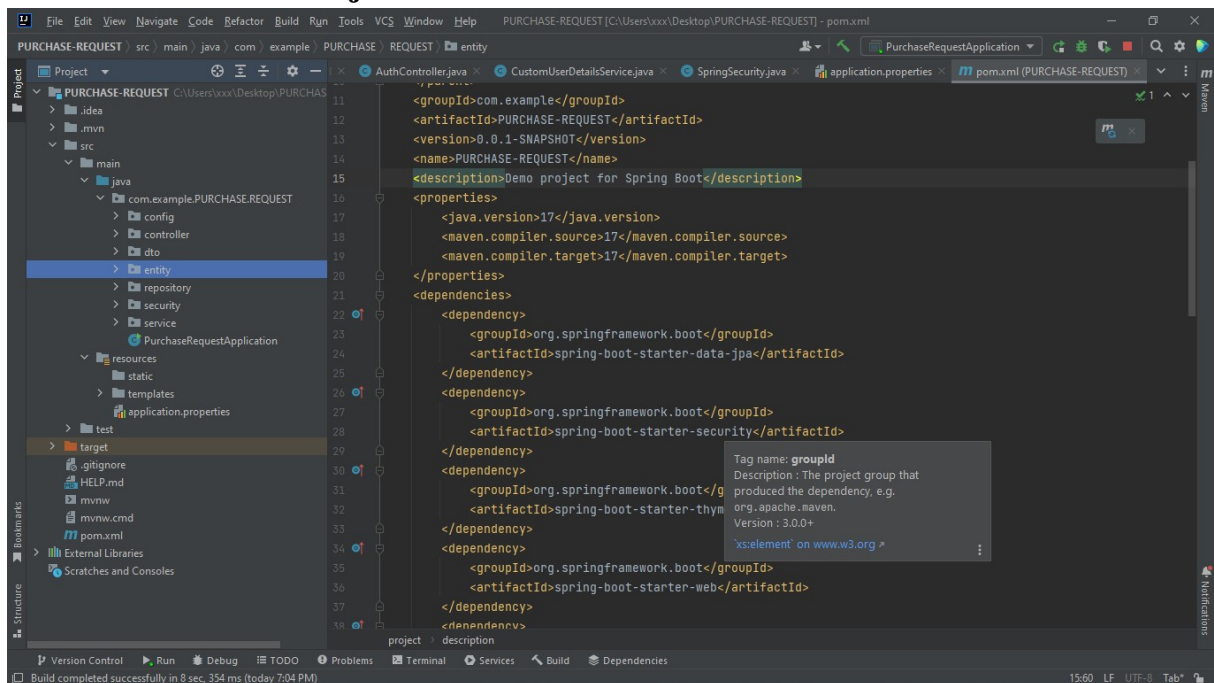
Purchase Request System Logout

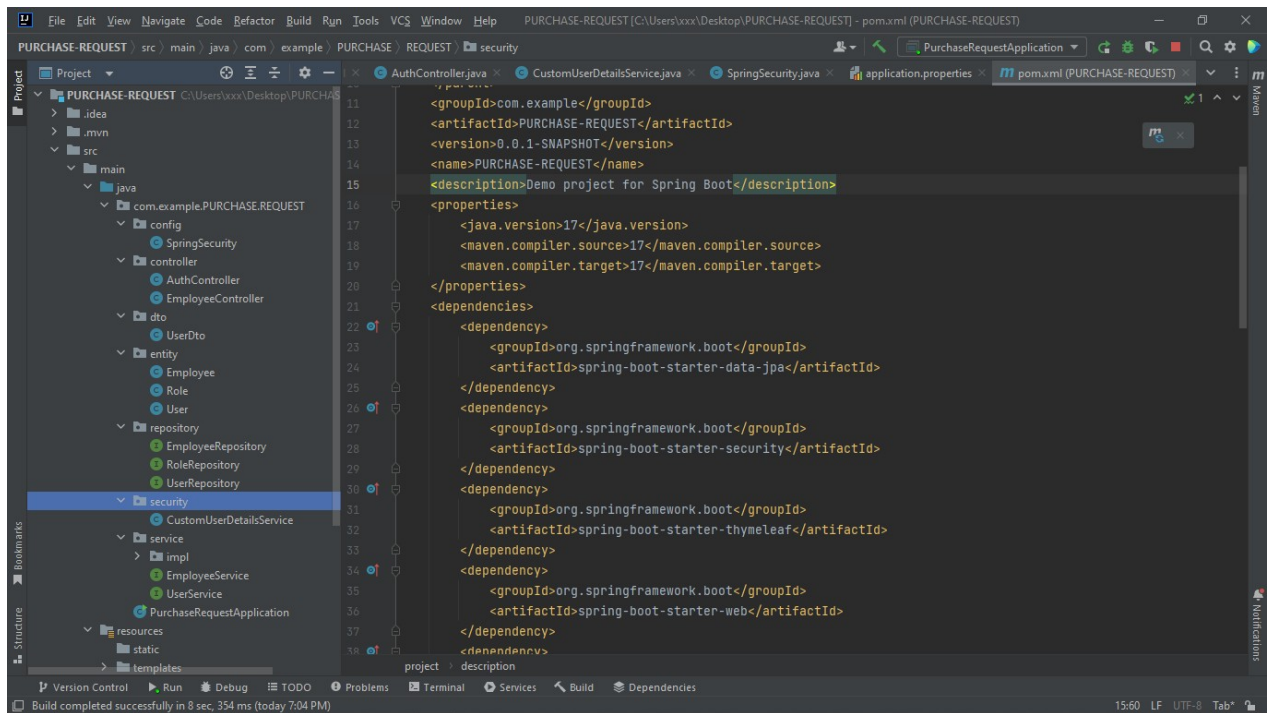
List of Registered Users

First Name	Last Name	Email
honore	basile	honore23@gmail.com
Cedric	Messi	cedric1@gmail.com
Max	kev	max1@gmail.com
admin	admin	admin@gmail.com
honore	basile	honore1@gmail.com
kimenyi	basile	kimenyi1@gmail.com
basile	basile	basile1@gmail.com
Honore	Kimenyi	basile177@gmail.com
Maxime	Kevin	Maxime1@gmail.com
Maxime	Kevin	kev1@gmail.com



Structure of the Project





Structure of the Database : Database name is request

Table: roles

Columns:
 id: bigint AI PK
 name: varchar(255)

#	Time	Action	Message	Duration / Fetch
14	18:30:33	SELECT * FROM request.users LIMIT 0, 1000	10 row(s) returned	0.000 sec / 0.000 sec
15	18:34:50	SELECT * FROM request.roles LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
16	18:34:56	SELECT * FROM request.users LIMIT 0, 1000	10 row(s) returned	0.000 sec / 0.000 sec
17	18:35:31	SELECT * FROM request.roles LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
18	18:36:35	INSERT INTO 'request'.'roles' ('id', 'name') VALUES ('1', 'ADMIN')	1062: Duplicate entry '1' for key 'roles PRIMARY'	
19	22:08:31	SELECT * FROM request.users LIMIT 0, 1000	10 row(s) returned	0.079 sec / 0.015 sec

Project Plan Schedule

S/N	ACTIVITIES	WEEK1	WEEK2	WEEK3	WEEK4
1	Task 1				
2	Task 2				
3	Task 3				
4	Task 4				
5	Submit Project				