

**UNIVERSITETI POLITEKNIK I TIRANES  
FAKULTETI I TEKNOLOGJISE SE INFORMACIONIT**

**DETYRE KURSI**

**Relacion**

**Zhvillim aplikacioni ne Cloud**

**Punoi:**

**Geri Meko**

**Pranoi:**

**Dr.Eduart Torba**

***Tirane, Qershor 2024***

## Relacioni për Projektin e Platformës Web

### Përmbledhje e Projektit

Projekti ka për qëllim ndërtimin e një platforme web që përfshin një infrastrukturë të vendosur në Docker dhe një aplikacion web të zhvilluar me teknologjinë Java ose Spring Boot. Platforma përmban module për autentikimin dhe regjistrimin e përdoruesve, administrimin e profileve dhe rolin e administratorit për menaxhimin e përdoruesve.

### Specifikat e Projektit:

#### 1. Infrastruktura në Docker

- **a. Ndërtimi i një imazhi për webserverin**
  - Krijimi i një skedari Dockerfile për webserverin që përfshin një bazë imazhi (eclipse-temurin:19-alpine) dhe artefaktet e ndërtuara të aplikacionit (jar file).
- **b. Ndërtimi i një imazhi për databazën**
  - Krijimi i një Dockerfile për bazën e të dhënave MySQL, që përfshin imazhin e MySQL dhe konfigurimet e nevojshme për krijimin dhe inicializimin e bazës së të dhënave.
- **c. Konfigurimi i komunikimit midis tyre (docker-compose)**
  - Krijimi i një skedari docker-compose.yml për orkestrimin e shërbimeve, duke specifikuar komunikimin midis konteinerëve të webserverit dhe bazës së të dhënave.

#### 2. Platforma Web

- **a. Teknologjia: Frameworku Spring Boot**
  - Zhvillimi i aplikacionit web duke përdorur Spring Boot si framework për menaxhimin e varësive, kontrollin e të dhënave dhe shërbimeve të sigurisë.
- **b. Moduli i autentikimit të userit duke vendosur email dhe password**
  - Implementimi i një ndërfaqe për login që lejon përdoruesit të autentikohen duke përdorur email dhe fjalëkalim.
- **c. Moduli i regjistrimit të userit**
  - Implementimi i një ndërfaqe për regjistrimin e përdoruesve që përfshin fushat: Emri, Atësia, Mbiemri, Nr Tel, Datëlindja, email, password.
- **d. Profil i përdoruesit**
  - Pas autentikimit, përdoruesi ridrejtohet drejt profilit të tij ku mund të shohë dhe të modifikojë të dhënat e tij personale, duke përfshirë edhe foton e profilit.
- **e. Roli i administratorit**
  - Implementimi i një ndërfaqeje për administratorin që i lejon të shohë një listë të të gjithë përdoruesve të regjistruar në formë tabele, si dhe mundësinë për të modifikuar ose fshirë përdorues.
- **f. Validimi i të dhënave**
  - Validimi i të dhënave të përdoruesit në modulën e regjistrimit dhe në profilin e tij:
    - Emri, atësia, mbiemri: duhet të përmbajnë vetëm shkronja dhe të kenë maksimumi 20 karaktere.

- Numri i telefonit: duhet të fillojë me +355 dhe të ketë 9 numra (formati: +35569xxxxxxx).
- Datëlindja: duhet të jetë në format date.
- Emaili: duhet të jetë në formatin e duhur të emailit.
- Password: duhet të ketë minimumi 8 karaktere, një shkronjë të madhe, një shkronjë të vogël, një numër dhe një karakter special.

## Hapat e Implementimit

### Hapi I: Ndërtimi i Ndërfaqeve të Login dhe Register në Spring Boot

- **Krijimi i Projektit Spring Boot:**
  - Inicimi i një projekti të ri në Spring Boot duke përdorur Spring Initializer dhe përzgjedhja e dependencies të nevojshme: Spring Web, Spring Data JPA, Spring Security, MySQL Driver.
- **Krijimi i Modeleve dhe Repositorëve:**
  - Krijimi i modelit User me fushat përkatëse dhe anotimi me @Entity dhe @Table.
  - Krijimi i repositorit UserRepository duke përdorur JpaRepository për operacionet CRUD.
- **Krijimi i Shërbimeve dhe Kontrolluesve:**
  - Implementimi i CustomUserDetailsService për logjikën e biznesit dhe përdorimin e repositorëve.
  - Krijimi i ApplicationController dhe AdminController për trajtimin e kërkesave HTTP.
- **Implementimi i Sigurisë:**
  - Konfigurimi i Spring Security për autentifikimin dhe autorizimin e përdoruesve, dhe krijimi i klasave për detajet e përdoruesit dhe menaxhimin e sesioneve.

### Hapi II: Krijimi i Database-it për Web-in e Ndërtuar

- **Dizajnimi i Bazës së të Dhënave:**
  - Përcaktimi i skemës së bazës së të dhënave dhe krijimi i tabelave të nevojshme duke përdorur MySQL Workbench.
- **Konfigurimi i Bazës së të Dhënave në Spring Boot:**
  - Konfigurimi i application.properties për të specifikuar detajet e lidhjes me bazën e të dhënave MySQL dhe testimi i lidhjes.

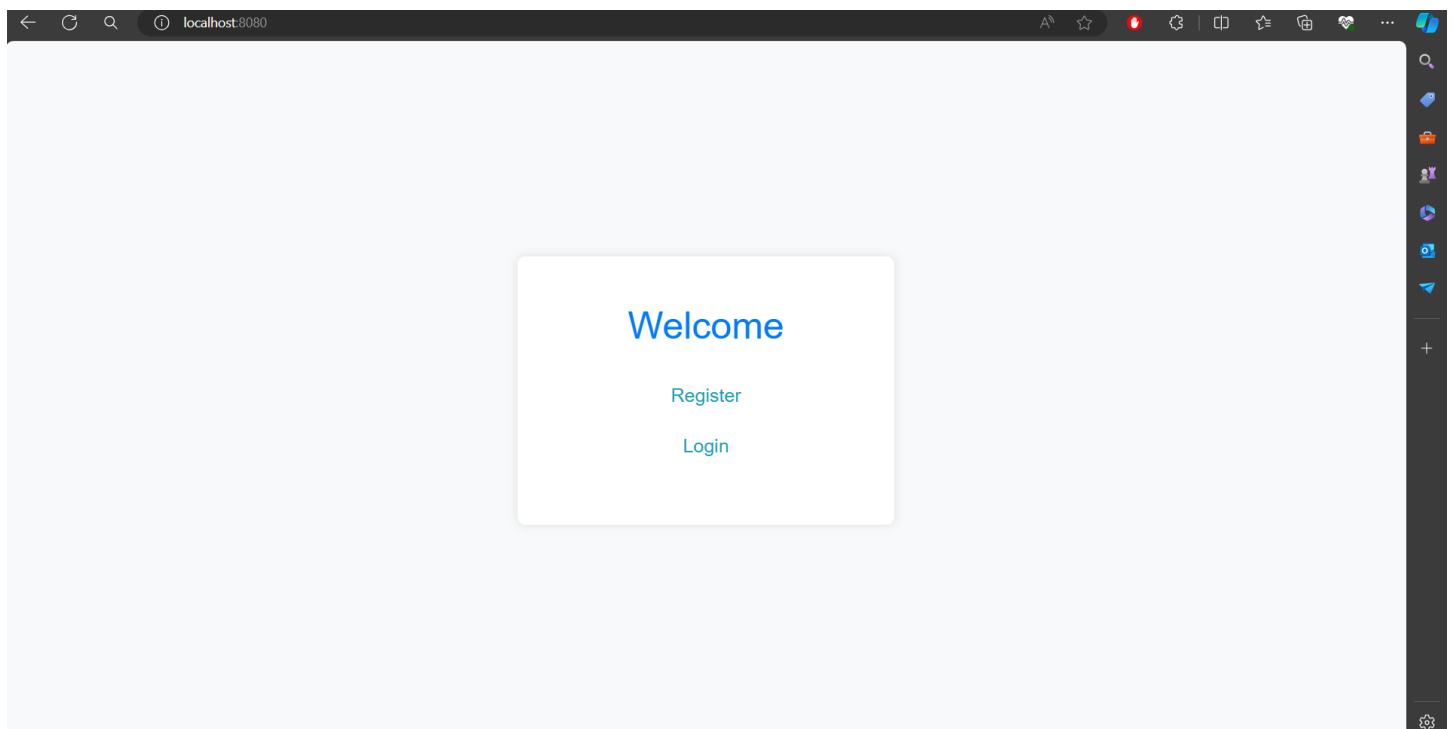
### Hapi III: Konfigurimi i Platformës Web në Docker

- **Krijimi i Dockerfile:**
  - Krijimi i Dockerfile për webserverin dhe për bazën e të dhënave në një folder DB.
- **Krijimi i docker-compose.yml:**
  - Krijimi i skedarit docker-compose.yml për orkestrimin e shërbimeve të aplikacionit dhe bazës së të dhënave MySQL.
- **Ekzekutimi i Docker Compose:**

- Ekzekutimi i komandës `docker-compose up --build` për të nisur të gjitha shërbimet dhe verifikimi i funksionimit të aplikacionit duke aksesuar URL-në përkatëse.

## Përfundimi

Projekti i platformës web është ndërtuar me sukses duke përdorur teknologjinë Spring Boot dhe MySQL për menaxhimin e të dhënave. Infrastruktura e vendosur në Docker siguron një mjedis të izoluar dhe të shkallëzueshëm për aplikacionin dhe bazën e të dhënave. Përdoruesit mund të regjistrohen, autentikohen dhe menaxhojnë të dhënat e tyre, ndërsa administratori ka mundësinë të menaxhojë të gjithë përdoruesit e platformës.



Sign Out

## User Profile

User ID	E-mail	First Name	Last Name	Profile Picture	Actions
2	admin@admin.com	Administrator	admin		<a href="#">View/Edit Profile</a>

List of Users

Sign Out

## List of Users

User ID	E-mail	First Name	Last Name	Actions	
2	admin@admin.com	Administrator	admin	<a href="#">View/Edit Profile</a>	<a href="#">Delete</a>
3	user1@users.com	user1	users	<a href="#">View/Edit Profile</a>	<a href="#">Delete</a>
7	user2@users.com	user2	users	<a href="#">View/Edit Profile</a>	<a href="#">Delete</a>
8	user9@users.com	user9	users	<a href="#">View/Edit Profile</a>	<a href="#">Delete</a>
9	user10@users.com	user10	users	<a href="#">View/Edit Profile</a>	<a href="#">Delete</a>
10	user11@users.com	user11	users	<a href="#">View/Edit Profile</a>	<a href="#">Delete</a>

## Please sign in

user1@users.com

.....

Sign in

## User Profile

**First Name:**

user1

**Last Name:**

users

**Father's Name:**

Devops

**Phone Number:**

+355691234567

**Email:**

user1@users.com

**Birthday:**

01 / 01 / 2000



**Profile Picture:** Choose File No file chosen

Update Profile

Cancel

localhost:8080/register

User Registration - Sign Up

E-mail:

user1@users.com

Password:

.....

Role:

User

First Name:

Last Name:

Birthday:

mm/dd/yyyy

Phone Number:

Father's Name:

Upload Profile Picture

Choose File

No file chosen

Sign Up

EXPLORER

DETYRE KURSI

db

spring-boot-registration-login

mvn

settings

bin

src

target

uploads

classpath

env

gitignore

project

Dockerfile

mvnw

mvnw.cmd

pom.xml

README.md

uploads

Detyre Kursi.pdf

docker-compose.yml

OUTLINE

TIMELINE

SERVICES

JAVA PROJECTS

MAVEN

Command Prompt

Microsoft Windows [Version 10.0.22631.3672]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\HP-Laptop>docker ps  
CONTAINER ID IMAGE PORTS NAMES  
44aa29145db1 detyrekursi-spring-boot-registration-login "java -jar SpringBoo..." 23 hours ago  
Up 4 minutes 0.0.0.0:8080->8080/tcp detyrekursi-spring-boot-registration-login-1  
a7ef3fb48054 detyrekursi-db "docker-entrypoint.s..." 2 days ago  
Up 4 minutes 33060/tcp, 0.0.0.0:3307->3306/tcp detyrekursi-db-1  
  
C:\Users\HP-Laptop>docker inspect -f '{{range.NetworkSettings.Networks}}{{.IPAddress}}{{end}}' a7ef3fb48054  
'172.18.0.2'  
  
C:\Users\HP-Laptop>S

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

spring-boot-registration-login-1 | 2024-06-09 12:41:38 [http-nio-8080-exec-9] DEBUG o.s.o.j.s.OpenEntityManagerInViewInterceptor - Closing JPA EntityManager in OpenEntityManagerInViewInterceptor  
spring-boot-registration-login-1 | 2024-06-09 12:41:38 [http-nio-8080-exec-9] DEBUG o.s.web.servlet.DispatcherServlet  
spring-boot-registration-login-1 | 2024-06-09 12:41:38 [http-nio-8080-exec-9] DEBUG o.s.s.w.a.ExceptionTranslation  
spring-boot-registration-login-1 | 2024-06-09 12:41:38 [http-nio-8080-exec-9] DEBUG o.s.s.w.c.SecurityContextPersistenceManager  
w cleared, as request processing completed  
spring-boot-registration-login-1 | 2024-06-09 12:41:38 [http-nio-8080-exec-1] DEBUG o.s.s.w.h.writers.HstsHeaderWriter  
it did not match the requestMatcher org.springframework.security.web.header.writers.HstsHeaderWriter\$SecureRequestMatcher  
spring-boot-registration-login-1 | 2024-06-09 12:41:38 [http-nio-8080-exec-1] DEBUG o.s.s.w.c.HttpSessionSecurityContextRepository  
empty or contents are anonymous - context will not be stored in HttpSession.  
spring-boot-registration-login-1 | 2024-06-09 12:41:38 [http-nio-8080-exec-1] DEBUG o.s.o.j.s.OpenEntityManagerInViewInterceptor  
ger in OpenEntityManagerInViewInterceptor  
spring-boot-registration-login-1 | 2024-06-09 12:41:38 [http-nio-8080-exec-1] DEBUG o.s.web.servlet.DispatcherServlet  
spring-boot-registration-login-1 | 2024-06-09 12:41:38 [http-nio-8080-exec-1] DEBUG o.s.s.w.a.ExceptionTranslation  
spring-boot-registration-login-1 | 2024-06-09 12:41:38 [http-nio-8080-exec-1] DEBUG o.s.s.w.c.SecurityContextPersistenceManager  
w cleared, as request processing completed

docker-compose

Snipping Tool

Screenshot copied to clipboard and saved  
Select here to mark up and share.

Ln 5, Col 91

Spaces: 4

UTF-8

ONLY

Markdown

Go Live

Prettier