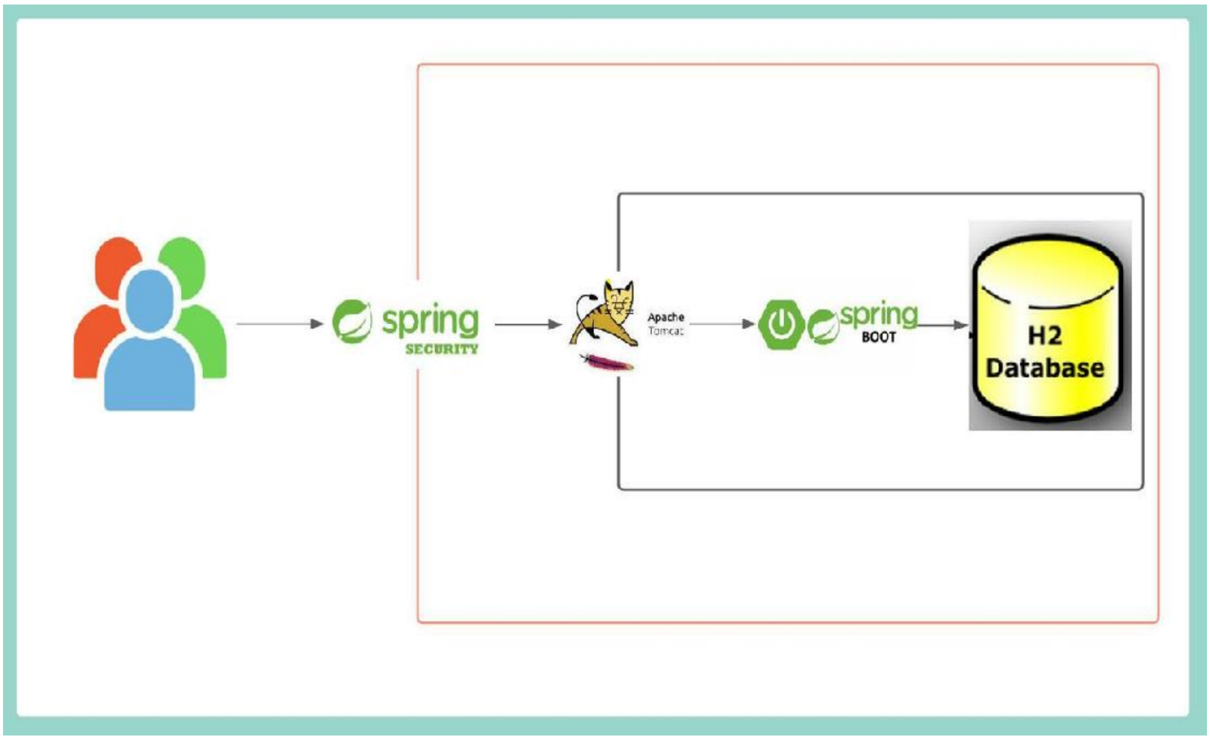
Online ToDo Application

This Spring boot application helps to Manage Todo's for the Users, and this application is protected by spring-boot security

Features Implementation

- Developed User registration module implementation
- Developed a Login module implementation
- Developed a **Todo CRUD** operations implementation
    - ➢ Add Todo for a User
    - ➢ Update Todo for a User
    - ➢ Delete Todo for a User
    - ➢ View Todo for a User

Architecture



Technologies and Tools

Technologies and Tools used to develop this application

Operating System	Windows 10 Home
Programming Languages	Core Java -8
Application Frameworks	Spring-boot 2.1.2, Spring-security
Template engine	Thymeleaf
Logging Framework	Apache Log4j
Testing Frameworks	Junit-5, Mockito-2
Web Designing Tools	HTML, CSS, JSP, Scriptlets
Code Review Plugin	SonarQube Server/SonarLint Client
IDE/Development Tools	Spring Tool Suite
Version Control System	Github
Database	H2 In-memory DB
Application Server	Tomcat 8
Dependency management	Maven-3
Design patterns	Factory Method Pattern, Singleton pattern, Proxy pattern, Template pattern

I have used two view resolvers ThymeleafViewResolver and InternalResourceViewResolver to process Jsp. Since lot of legacy application still sitting in JSP, just want to show how we can accommodate both Thymeleaf and JSP in single Spring-Boot application, then we can work on migrating any low performance JSP module into effective thymeleaf module, Since Thymeleaf offers a set of Spring integrations that allow you to use it as a fully-featured substitute for JSP in Spring MVC applications .In this application Login & registration module built using Thymeleaf and Todo module built using JSP.

Used Junit-5 for better documentation logging with minimal annotations.

This Application not developed only for UI users. We can leverage this applications APIs to integrate with other applications in an enterprise, hence exposed few apis to consume this application functionality, I have exposed few APIs for now. but I can able to expose all the functionality as APIs in future to integrate with other applications. so that for a company it will single common todo application.

Source Code : prabagaran-sellamuthu/online-todo-application (github.com)

Online-todo-application Launch & Test

Online Todo Spring Boot Application

# Prerequisites:

JDK-8 and Tomcat-8 installed on your machine

# How to start

git clone https://github.com/prabagaran-sellamuthu/online-todo-application.git

cd online-todo-application

mvn clean install

To test using spring-boot embedded tomcat

mvn spring-boot:run

Launch this Url in browser

http://localhost:8080/login

To test using external tomcat 8

Place the war file in tomcat webapps folder

Launch this Url in browser

http://localhost:8080/online-todo-application/login

Embedded tomcat
Swagger UI
http://localhost:8080/swagger-ui.html
DB UI
http://localhost:8080/h2-console

Username : sa
Password: password

External Tomcat
Swagger UI
http://localhost:8080/online-todo-application/swagger-ui.html
DB UI
http://localhost:8080/online-todo-application/h2-console

Username : sa
Password: password