# Approach and Design Pattern

To develop Todo List web application I have used the Spring Boot open source framework. Spring Boot makes it easy to create stand-alone, production-grade Spring based Applications that you can just run. Spring framework provides a comprehensive infrastructure support for developing Java applications.

# Why Spring Boot?

SpringBoot’s powerful auto-configuration mechanism makes it very easy to get started with a Spring-based application. More importantly, SpringBoot offers a wide array of Starters which is more than sufficient for many applications. One of the primary goals of Spring and SpringBoot is to make things easier. Spring portfolio has it’s own powerful Web MVC framework, Spring Security framework but the majority of its other projects are to provide higher level abstractions to make using them easier.

# MVC Architecture

A Spring MVC is a Java framework which is used to build web applications. It follows the Model-View-Controller design pattern. It implements all the basic features of a core spring framework like Inversion of Control, Dependency Injection.

A Spring MVC provides an elegant solution to use MVC in spring framework by the help of **DispatcherServlet**. Here, **DispatcherServlet** is a class that receives the incoming request and maps it to the right resource such as controllers, models, and views.

# Spring Data JPA

* To interact with the database I have used Spring data JPA(Java Persistence API).It provides JPA template class to integrate spring application with JPA.
* It gives abstract repositories that are implemented at run time by spring container and perform CRUD operations. As a developer we have to provide the abstract methods in the interface.
* The biggest advantage is Spring data JPA reduces the amount of boiler plate code required to write data access layer.

# Views and Database

In this Todo list project I have used H2 in memory database. The DB consists of two tables *Users* and *Tasks* table to hold the user and task data respectively. H2 database makes it handy for the developers to test the application. For the views I have used Java Server Pages JSP, because performance is significantly better because JSP allows embedding Dynamic Elements in HTML Pages itself instead of having separate CGI files. Regular HTML, that is static HTML, does not contain dynamic [information](https://ecomputernotes.com/fundamental/information-technology/what-do-you-mean-by-data-and-information). So it does not react to user input and is also not fit for accessing server side resources. JSP contains both static and non-static content. As static part, it contains HTML.

**What Extra I Would Have Done**

Spring Security

Spring Security is the primary choice for implementing application-level security in Spring applications. Generally, its purpose is to offer you a highly customizable way of implementing authentication, authorization, and protection against common attacks. Spring Security is open-source software released under the Apache 2.0 license. The real power of Spring Security is found in how easily it can be extended to meet custom requirements. Spring security is comprehensive, makes it easy for Servlets API integration and provides protection against attacks. Due to time constraints I could not implement spring security. Spring provides authentication frameworks one of them is JWT based authentication .where a token will be generated at server side and validated for each user.

# Logging

Logging is the process of writing log messages during the execution of a program to a central place. This logging allows you to report and persist error and warning messages as well as info messages (e.g., runtime statistics) so that the messages can later be retrieved and analyzed. The object which performs the logging in applications is typically just called Logger. Loggers will be very useful mainly in production environment.

**Automation Of Deployment Using Jenkins and AWS**

Jenkins is a self-contained, open source automation server which can be used to automate all sorts of tasks related to building, testing, and delivering or deploying software.

Jenkins can be installed through native system packages, Docker, or even run standalone by any machine with a Java Runtime Environment (JRE) installed. [Amazon web service](https://www.simplilearn.com/tutorials/aws-tutorial/aws-fundamentals) is an online platform that provides scalable and cost-effective cloud computing solutions.AWS is a broadly adopted cloud platform that offers several on-demand operations like compute power, database storage, content delivery, etc., to help corporates scale and grow.

# Session Management

Spring Session provides an API and implementations for managing a user’s session information. while also making it trivial to support clustered sessions without being tied to an application container-specific solution Spring Session decouples the session management logic from the application, making it more tolerant.

**Microservices and Spring Cloud**

Spring Cloud is **a framework for building robust cloud applications** and it provides a solution to the commonly encountered patterns when developing a distributed system. Spring Cloud framework provides tools for developers to quickly build both cloud applications and microservice-based applications.

**Screen Shots of Todo web Application**

Graphical user interface, application

Description automatically generated

**Loign Page**

Graphical user interface

Description automatically generated

**Home Page**

A screenshot of a computer

Description automatically generated**View Task Page**

Graphical user interface, application

Description automatically generated

**Add Task Page**

Graphical user interface, application, website

Description automatically generated

**Edit Task Page**

A screenshot of a computer

Description automatically generated

**Edited Task**

Graphical user interface, application

Description automatically generated

**Deleted Task**

A screenshot of a computer

Description automatically generated

**Newly Added Task**

GIT HUB Link: https://github.com/NayanKumar1992/ToDoWebApp

URL: <http://localhost:8080/>

If war file doesn’t work you can run the application from command line

java -jar target/ TodoManagerProject-0.0.1-SNAPSHOT.jar