WikiAnime

Presentado Por:

Gabriel Arturo García Quintero - 1151654

Ivan Mauricio Uribe - 1151612

Daniel Calderon Ospinal - 1151687

Presentado a:

Ing. Carlos Rene Angarita

Universidad Francisco de Paula Santander

Facultad de Ingeniería

Ingeniería de Sistemas

Cúcuta

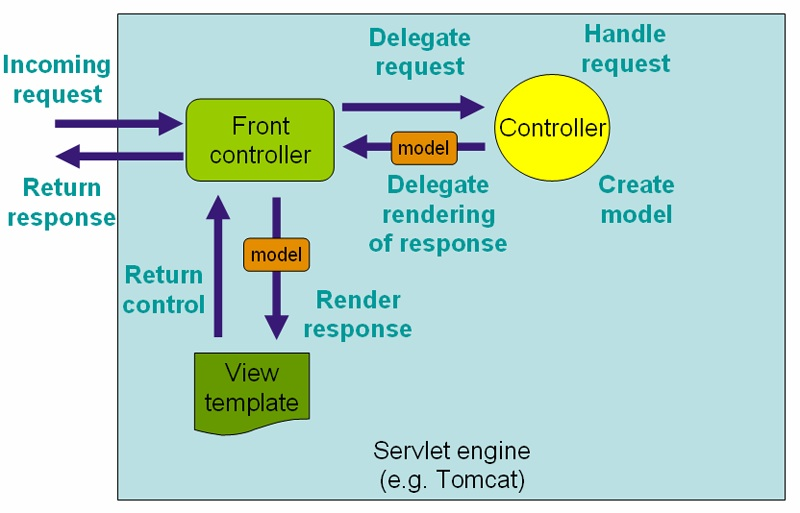
2020

Enunciado

La plataforma WikiAnime es muy reconocida por la gran interacción que proporciona a sus usuarios en la cual ellos mismos son los encargados de enriquecer los diversos aspectos de la plataforma.

La plataforma permite a los usuarios nuevos realizar la inscripción de la plataforma. Una vez registrado puede iniciar sesión en la misma y acceder a las opciones de agregar animes, agregar temporada a un anime y ver detalladamente todos los animes/temporadas que han sido publicados por diversos usuario dando a conocer quién es el autor de dicha publicación ofreciendo en el perfil una lista de los animes publicados por el usuario.

Arquitectura



En el proyecto implementamos la arquitectura **MVC** o Modelo-Vista-Controlador que, utiliza 3 componentes o capas principales (Vistas, Models y Controladores) separa la lógica de la aplicación de la lógica de la vista en una aplicación.

El modelo que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.

La vista, o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con éste.

El controlador, que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno.

Por otro lado, está el patrón de diseño **Front Controller** que se basa en usar un controlador como punto inicial para la gestión de las peticiones. El controlador gestiona estas peticiones, y realiza algunas funciones como: comprobación de restricciones de seguridad, manejo de errores, mapear y delegación de las peticiones a otros componentes de la aplicación que se encargarán de generar la vista adecuada para el usuario, es la encargada de estar al pendiente de la petición generada por el usuario mediante la url y dar una repuesta.

Se divide en 2 diferentes objetos el Front Controller y el Dispatcher. En ese caso, El Front Controller acepta todos los requerimientos de un cliente y realiza la autenticación, y el Dispatcher direcciona los requerimientos a manejadores apropiada

**Como lo hago correr.**

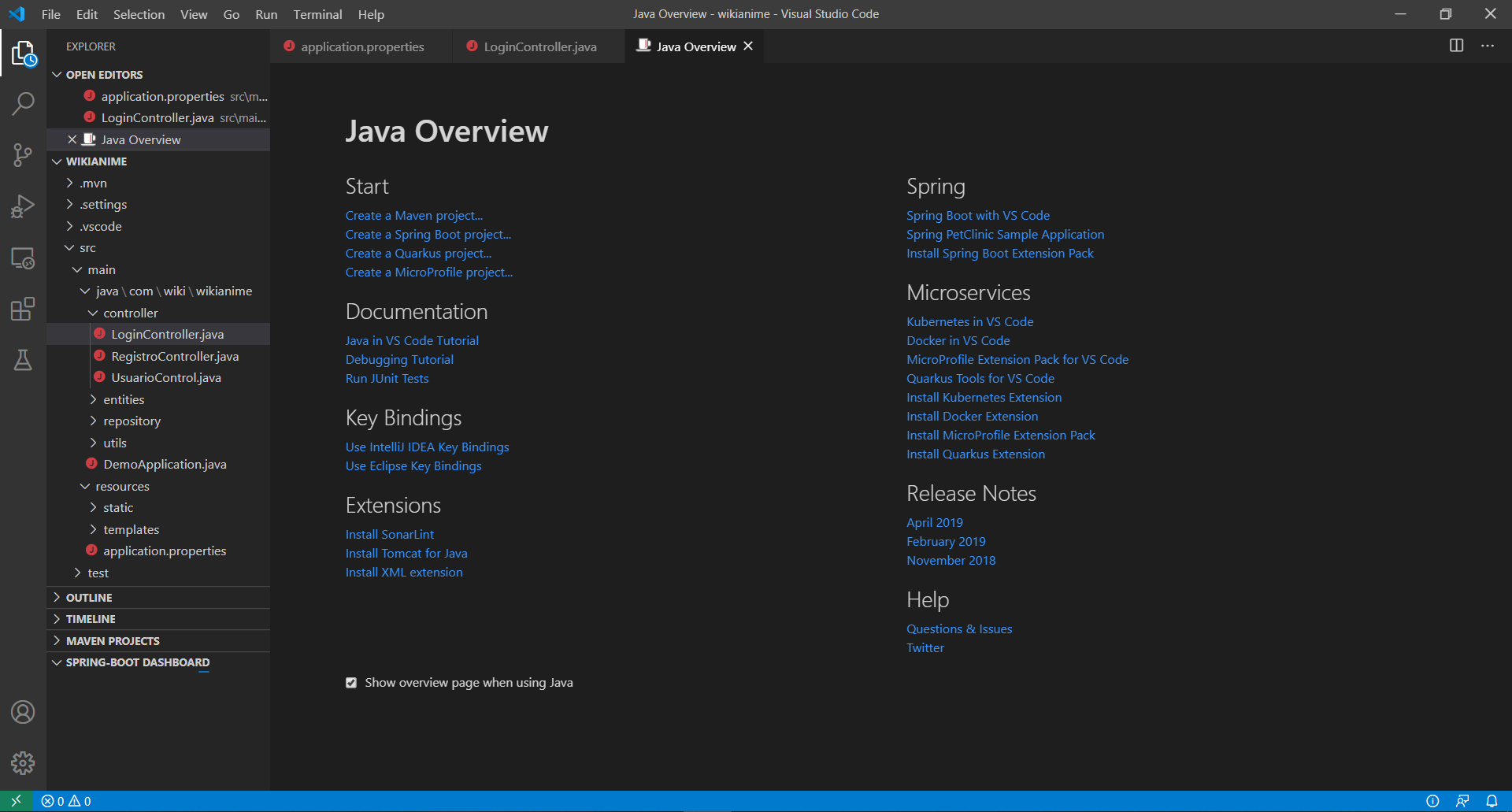
Para hacer correr el proyecto debemos descargar la aplicación XAMPP en nuestro ordenador, la cual trae una serie de herramientas como Apache Web Server Versión 2.4.41, MySQL Database Versión 8.0.17, phpMyAdmin Database Manager Version 4.9.1.

Iniciamos el Visual Code Studio (Primero hay que tener todos los plugins instalados, a continuación, decimos cuales fueron) Luego en Spring DashBoard le damos a “Start”, en el navegador buscamos “localhost:8082” y nos direccionar a la página principal.

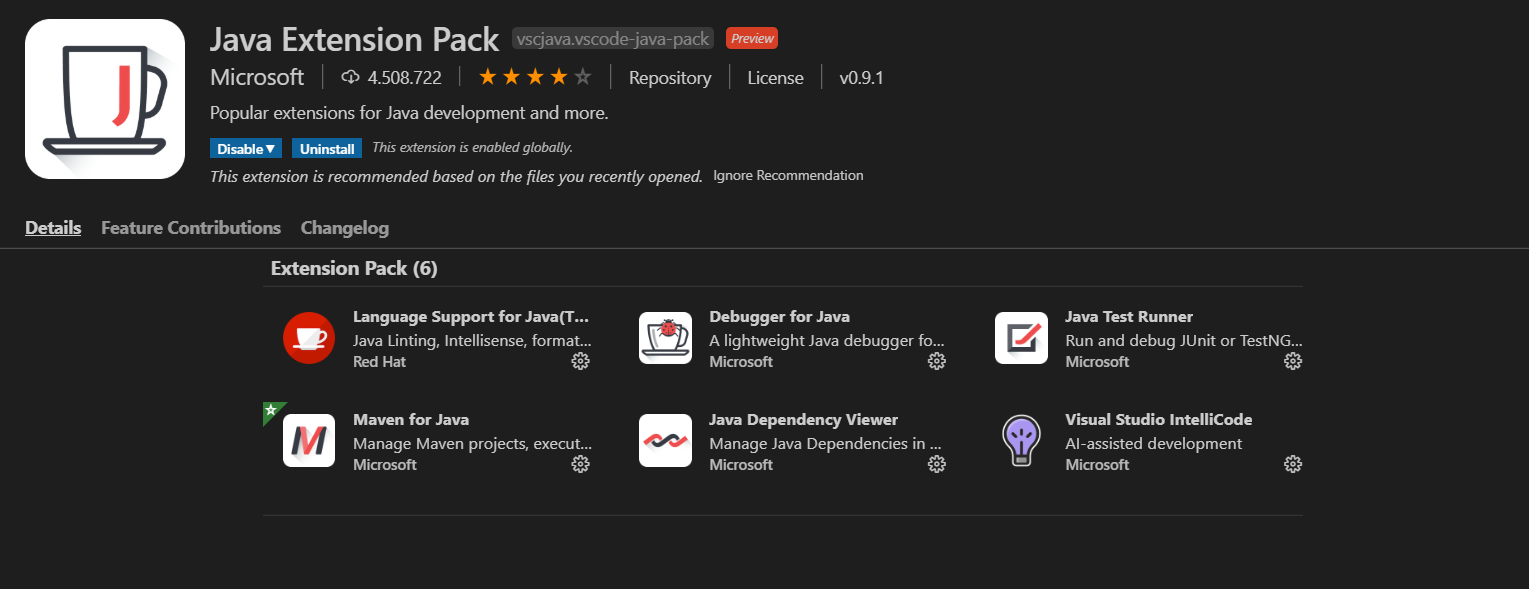
**Tecnología y Recursos.**

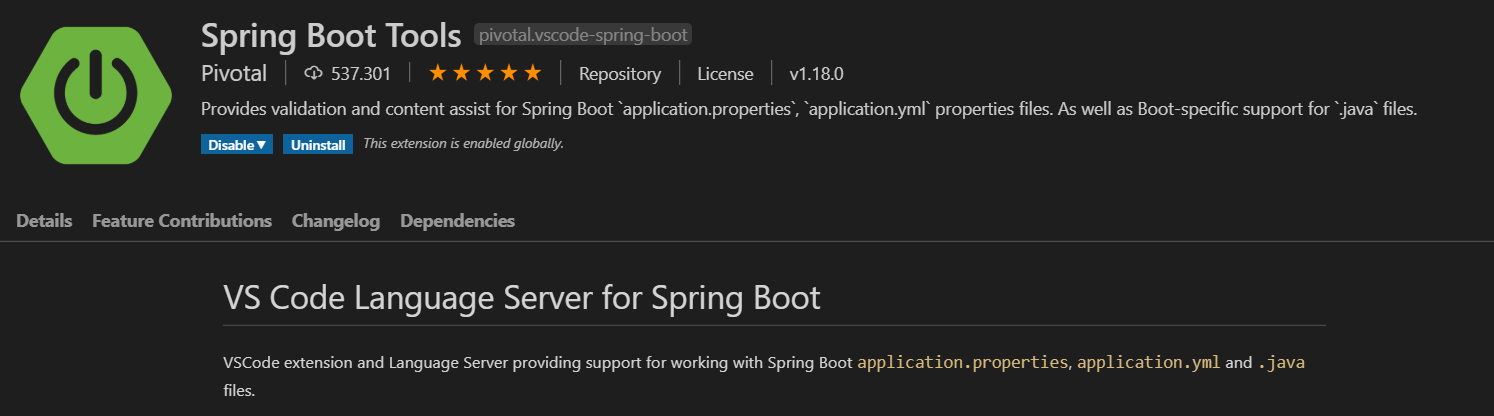
Esta sección estará dividida en dos partes, Frontend y Backend. Incluyendo el entorno de trabajo que se empleó:

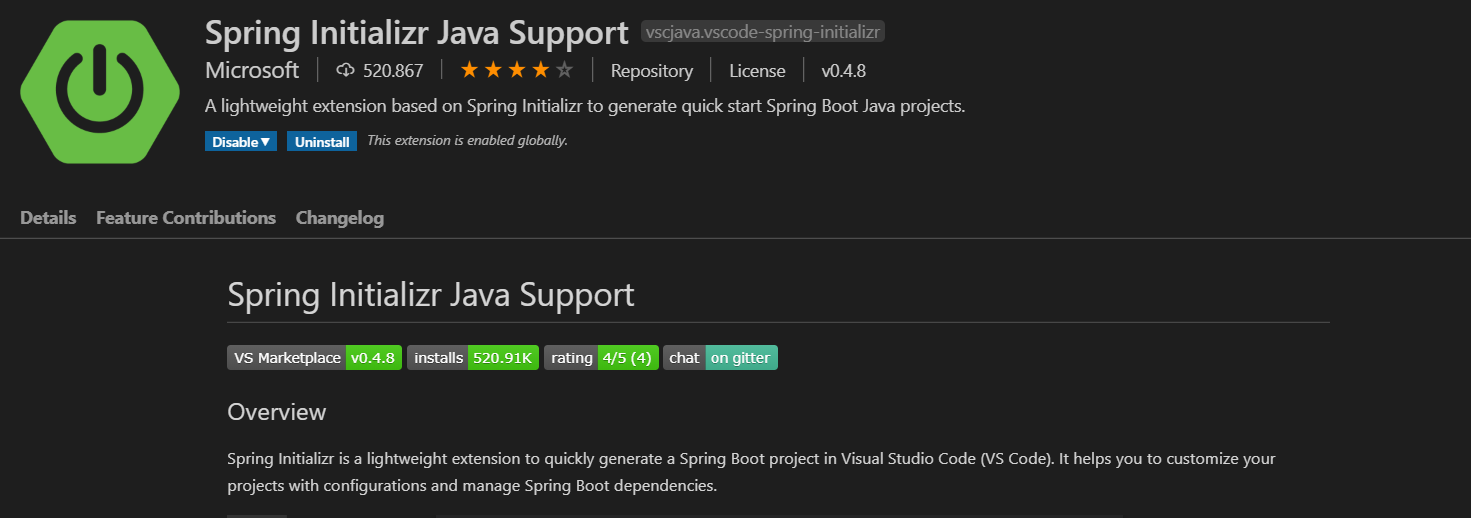
IDE de Trabajo: Visual Code Studio.

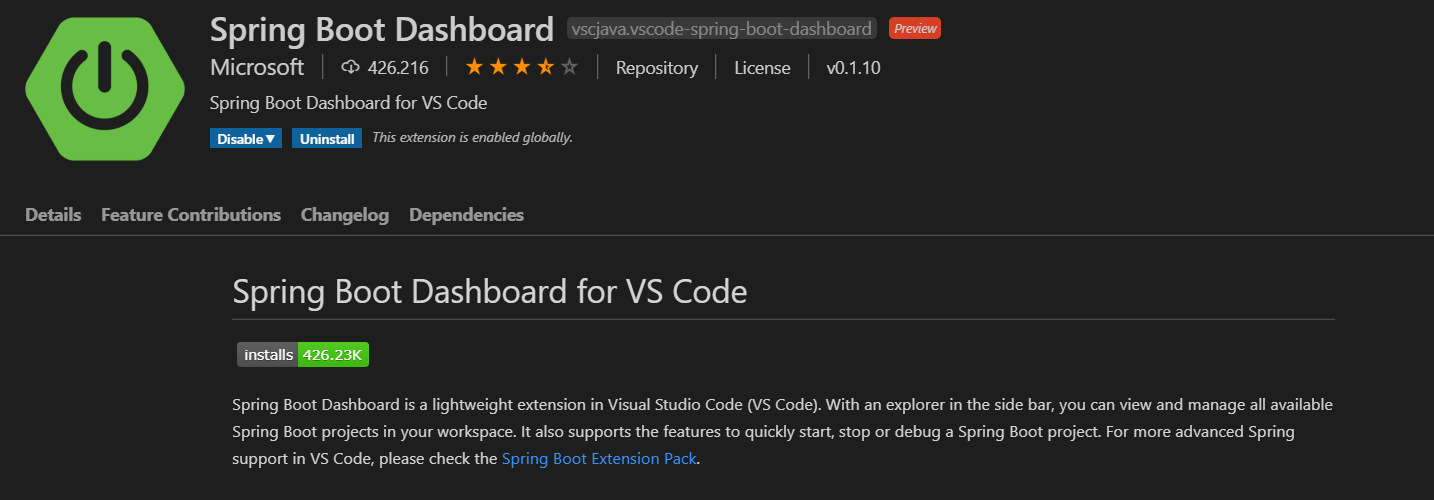


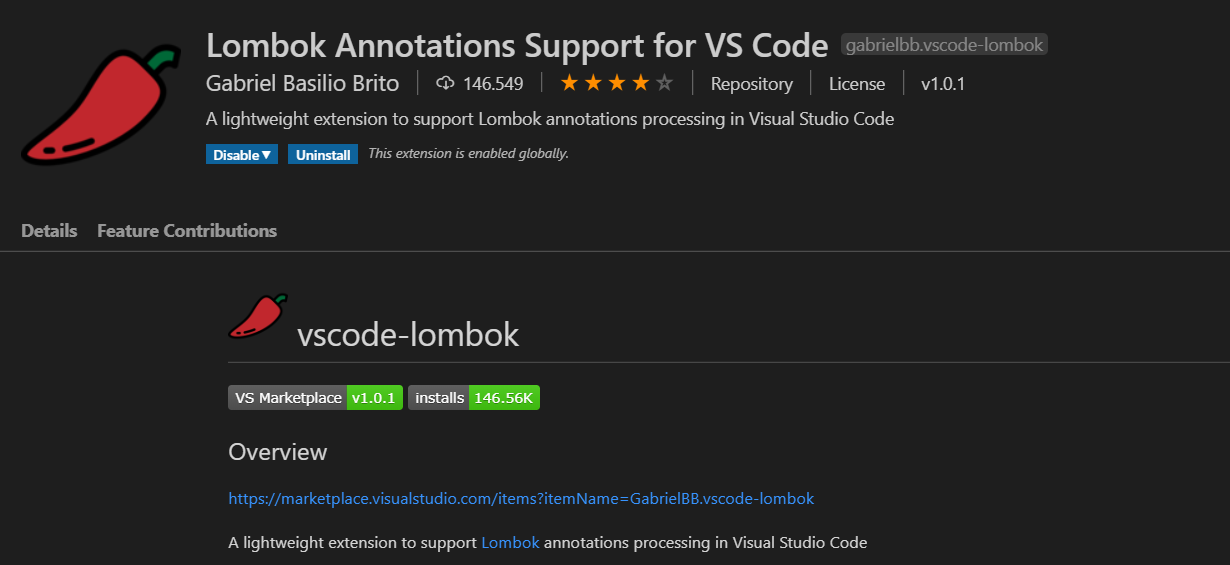
El cual se descargaron los siguientes plugins para el trabajo de Spring Boot.











Igualmente el link para la instalación de Spring Boot en Visual Code Studio

<https://code.visualstudio.com/docs/java/java-spring-boot>

La instalación de lombok para las anotaciones es importante para que el IDE reconozca esta librería.

Para el uso del **Backen** se realizó con Java 8, Java 8 es la versión más reciente de Java que incluye nuevas características, mejoras y correcciones de bugs para mejorar la eficacia en el desarrollo y la ejecución de programas Java; con el Framework Spring que es un framework para el desarrollo de aplicaciones y contenedor de inversión de control, de código abierto para la plataforma Java.

En el caso de **Frontend** utilizamos en el desarrollo del aplicativo el lenguaje HTML, CSS.(Frontend) y como framework Bootstrap, esta última es una excelente herramienta para crear interfaces de usuario limpias y totalmente adaptables a todo tipo de dispositivos y pantallas, sea cual sea su tamaño. Además, Bootstrap ofrece las herramientas necesarias para crear cualquier tipo de sitio web utilizando los estilos y elementos de sus librerías. Continuando con los lenguajes HTML es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. Se trata de las siglas que corresponden a HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto.

Páginas de referencia

La guía o referencia que utilizamos a lo largo del desarrollo de la aplicación, fueron una serie de videotutoriales que implementan los patrones de diseño Front Controller y MVC, a continuación, se dejarán los enlaces respectivos al curso realizado.

Video 1

<https://www.youtube.com/watch?v=ksLYIavT2L0&t=220s>

Video 2

<https://www.youtube.com/watch?v=zQjZFHo8eO4&t=302s>

Video 3

<https://www.youtube.com/watch?v=3JZZ6tkajdY&t=6s>

Video 4

<https://www.youtube.com/watch?v=f8g4MKKlYMg&t=911s>

Video 5

<https://www.youtube.com/watch?v=C1iiftjczTM&t=401s>

**Referencias:**

[**http://cristianruizblog.com/spring-boot-security-login/**](http://cristianruizblog.com/spring-boot-security-login/)

[**https://www.mitocode.com/**](https://www.mitocode.com/)

[**https://spring.io/projects/spring-framework**](https://spring.io/projects/spring-framework)

[**https://stackoverflow.com/questions/tagged/spring**](https://stackoverflow.com/questions/tagged/spring)

[**https://www.thymeleaf.org/documentation.html**](https://www.thymeleaf.org/documentation.html)