FlickTime

Giuseppe Daidone
Davood Sheibani

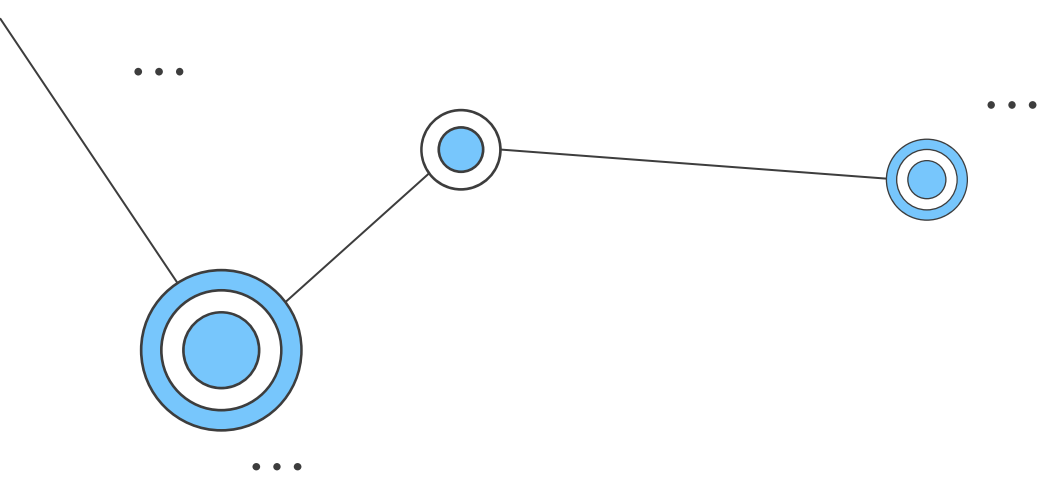
Project Overview

Project Overview

The "Flick Time" web application is designed to provide users with information about movies. It offers features such as user authentication, movie search, user preferences, and more.

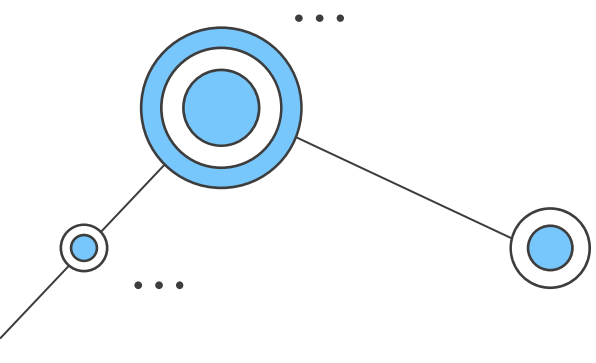
The application is built using the Spring Boot framework, which simplifies the development process and promotes best practices for building modern web applications.

Spring Boot



Spring Boot

- Spring Boot is a framework built on top of the Spring framework, designed to simplify the development of production-ready applications. It provides features for auto-configuration, embedded servers, and dependency management, allowing developers to focus on writing business logic rather than boilerplate code.

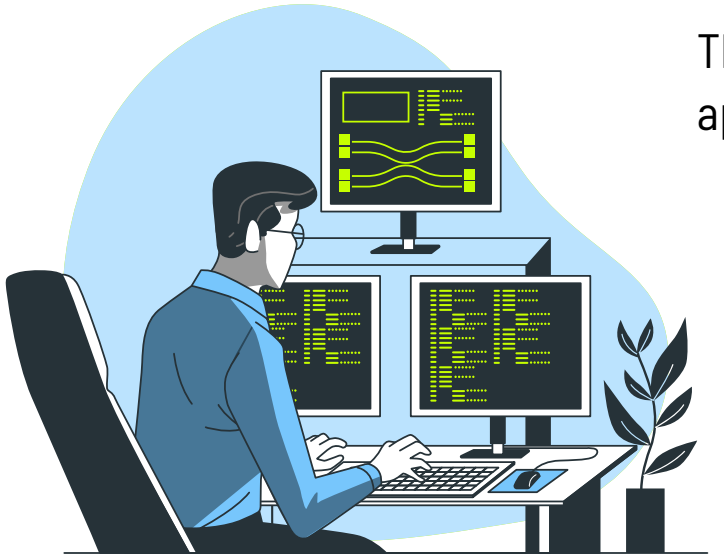


Explanation

Spring Boot provides a parent POM (spring-boot-starter-parent) that configures default settings and dependencies for Spring Boot applications.

Dependencies like spring-boot-starter-data-jpa, spring-boot-starter-security, spring-boot-starter-thymeleaf, and spring-boot-starter-web include pre-configured libraries for common functionalities like data access, security, web development, and templating.

The spring-boot-maven-plugin facilitates building and packaging Spring Boot applications into executable JAR or WAR files.



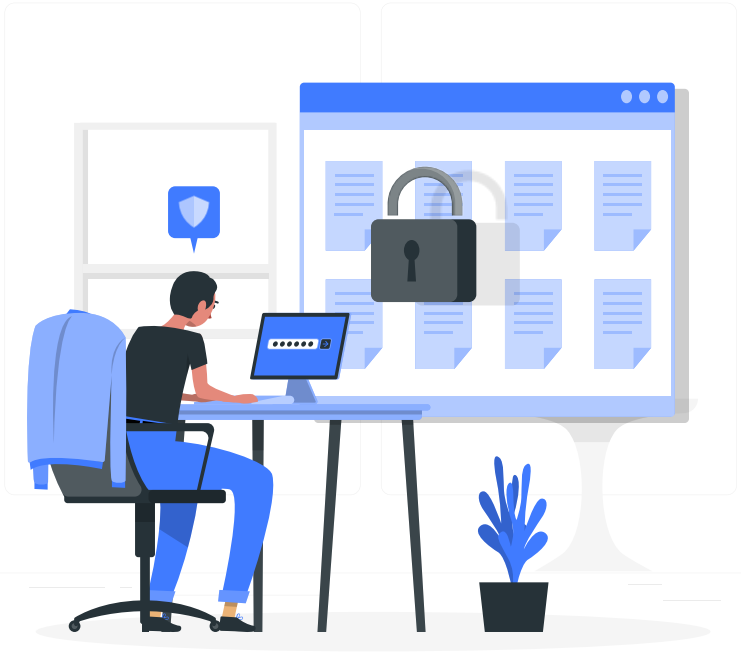
Key components

Authentication

Users can register an account and log in securely to access the application's features.

Spring Security is used for user authentication and authorization, ensuring that only authenticated users can access protected resources.

Controller Layer



Controllers handle incoming HTTP requests and delegate the processing to the appropriate service methods.

RESTful endpoints are defined to expose various functionalities of the application, such as movie search, user profile management, etc.

Service Layer

Services contain the business logic of the application, implementing functionalities such as movie search, user preferences management, etc. It orchestrates interactions between the Repository layer and other components, such as controllers and external services to perform CRUD (Create, Read, Update, Delete) operations on data entities.

Key Features:

- Implements business rules and logic.
- Encapsulates complex operations and workflows.
- Facilitates transaction management and ensures atomicity of operations.

Technologies Used:

- Spring Framework: Provides support for dependency injection, aspect-oriented programming and transaction management.
- Java: Utilizes core Java features and libraries to implement business logic.

Best Practices:

- Keep services focused and cohesive, adhering to the single responsibility principle.
- Avoid direct database access in services; delegate data access tasks to the Repository layer.
- Design service methods with clear input and output parameters to promote reusability and testability.

Repository Layer

The Repository layer is responsible for data access and persistence.

Repositories interface with the database to perform data access operations.

Spring Data JPA is used to simplify database interactions by providing repository interfaces with predefined CRUD (Create, Read, Update, Delete) methods.

Hibernate: Acts as the JPA provider and handles object-relational mapping (ORM) between Java objects and database tables.

Model Layer

The model layer represents the domain objects or entities used in the application.

Entities define the structure of data stored in the database, such as User, Movie, etc.

These entities are annotated with JPA annotations to specify the mapping between Java objects and database tables.

Data Validation

Data validation is performed to ensure the integrity and consistency of user input.

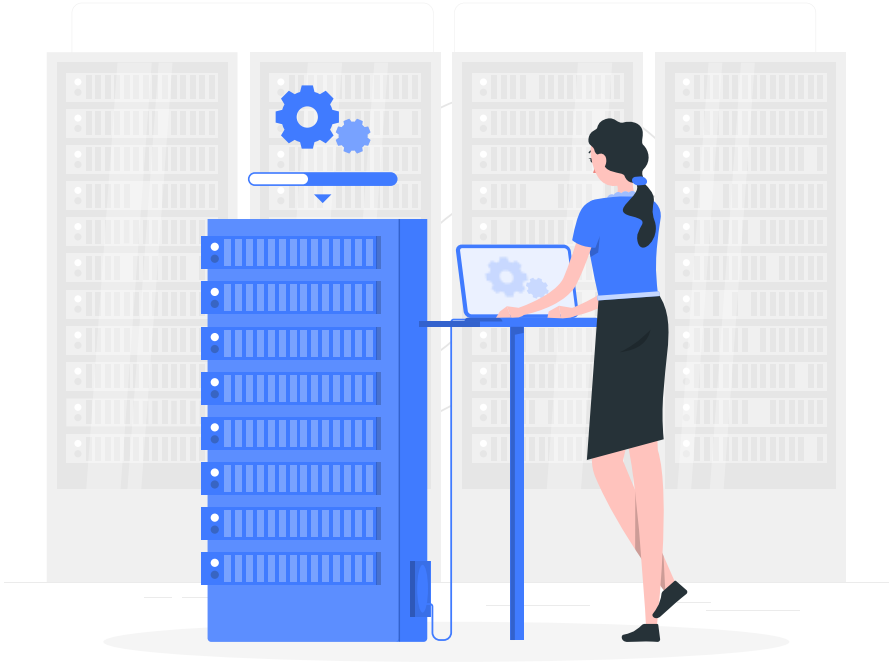
Spring Boot Validation is used to define validation constraints on domain model attributes and automatically validate incoming request data.

Templating Engine

Thymeleaf is used as the templating engine for server-side rendering of HTML templates.

Thymeleaf templates are dynamic and can include expressions to render data retrieved from the backend.

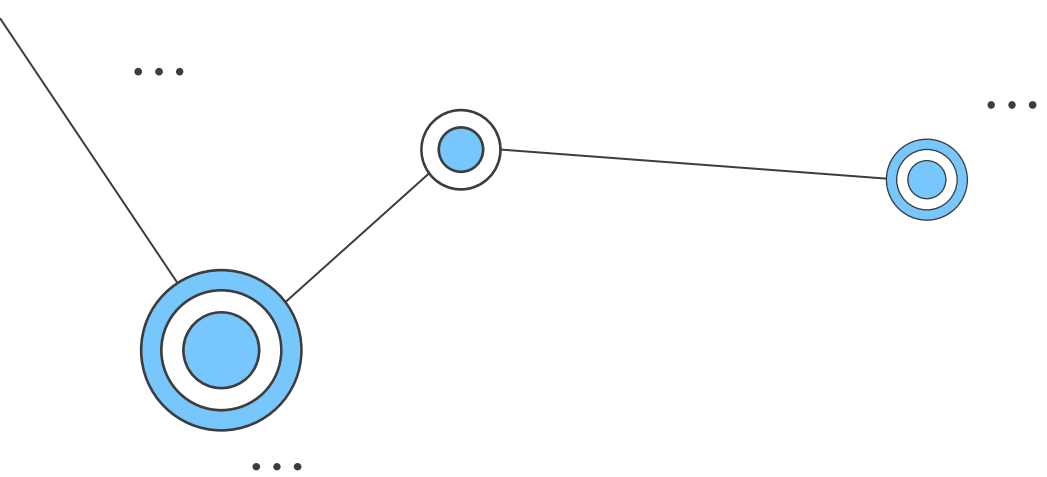
Database



PostgreSQL is used as the backend database to store application data, such as user information, movie details, etc.

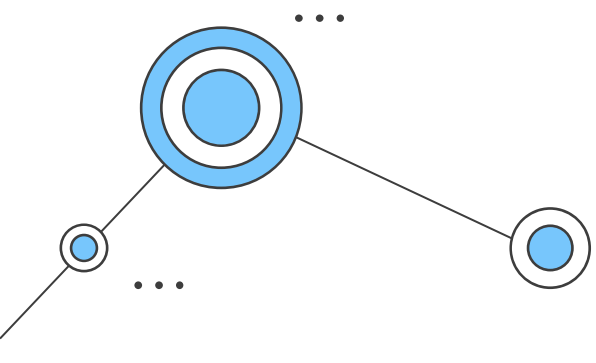
Spring Boot's auto-configuration simplifies database setup and connection management.

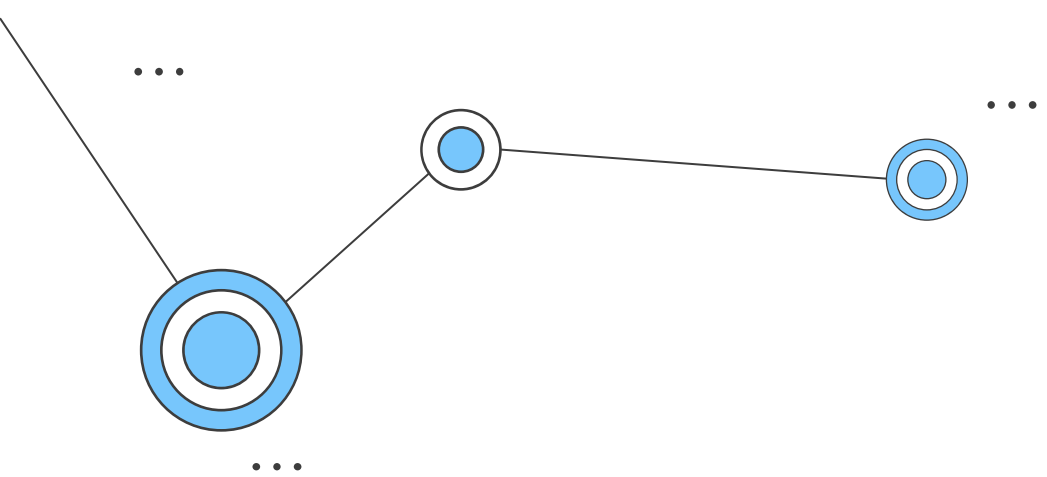
Technologies Used



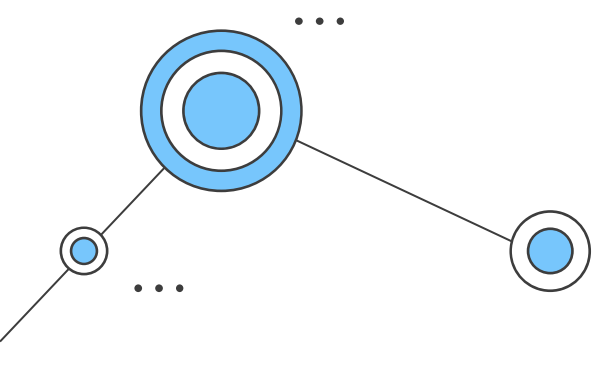
Spring Boot

- Description: Spring Boot is a powerful framework for building Java-based applications with minimal setup and configuration.
- Usage: Spring Boot simplifies application development by providing opinionated defaults and auto-configuration, allowing developers to focus on writing business logic rather than infrastructure code.
- Dependency: spring-boot-starter-parent serves as the parent project for Spring Boot applications, managing dependencies and providing default configurations.

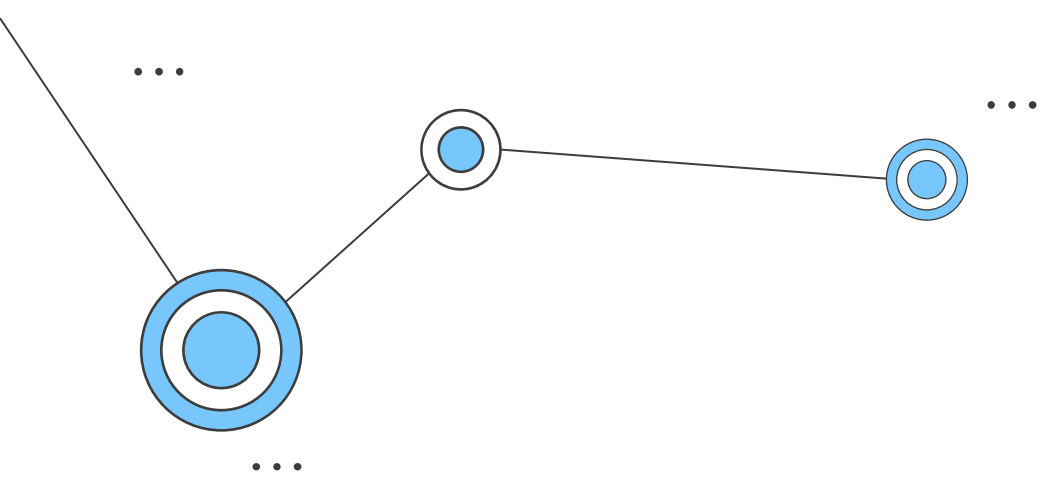




Spring Data JPA

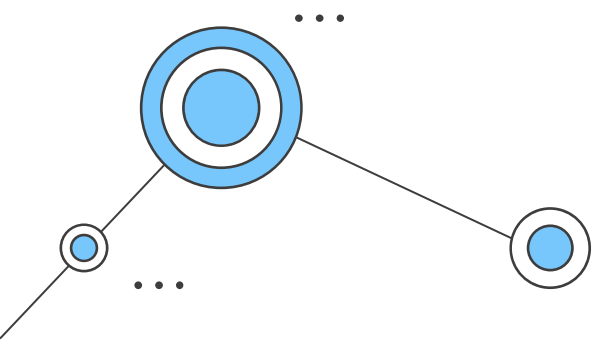


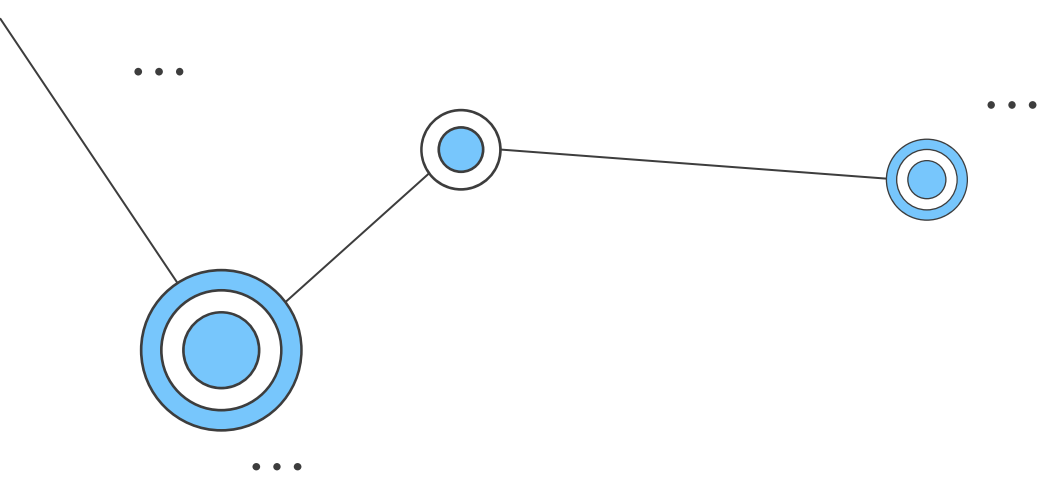
- Description: Spring Data JPA simplifies the implementation of data access layers in Spring applications by providing a repository abstraction over JPA (Java Persistence API).
- Usage: Developers can define repositories with minimal code using interfaces, allowing Spring Data JPA to generate necessary CRUD (Create, Read, Update, Delete) operations automatically.
- Dependency: spring-boot-starter-data-jpa includes Spring Data JPA and its required dependencies for database interaction.



Spring Security

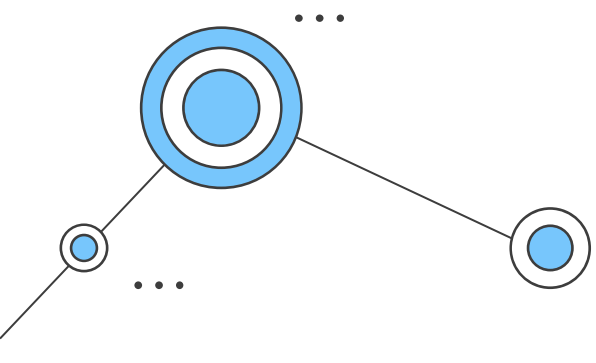
- Description: Spring Security is a powerful authentication and access control framework for Java applications.
- Usage: Spring Security enables developers to secure their applications by handling authentication, authorization, and other security concerns.
- Dependency: spring-boot-starter-security integrates Spring Security with Spring Boot applications, providing out-of-the-box security features.





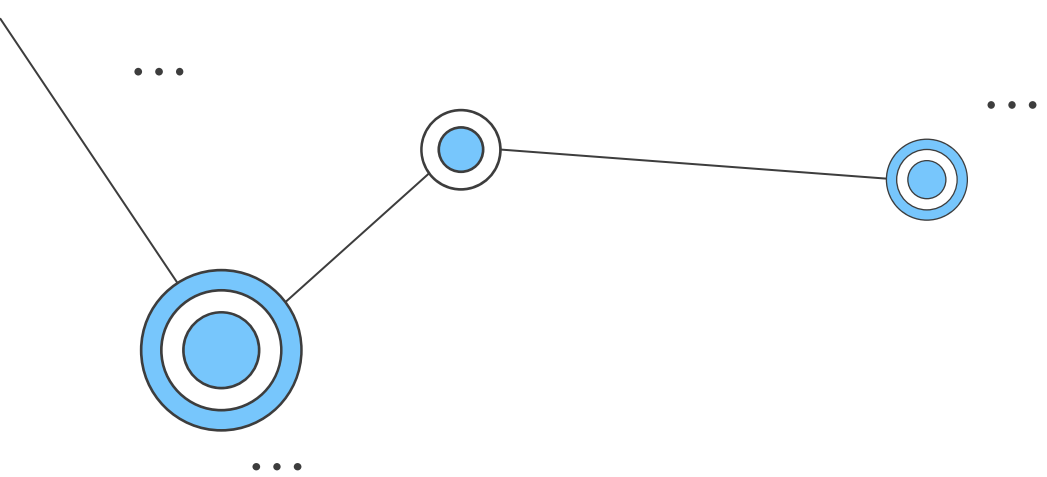
Thymeleaf

- Description: Thymeleaf is a modern server-side Java template engine for web and standalone environments.
- Usage: Thymeleaf simplifies the creation of dynamic web pages by allowing developers to write natural templates with dynamic expressions.
- Dependency: spring-boot-starter-thymeleaf includes Thymeleaf and its integration with Spring Boot for server-side rendering of HTML templates.



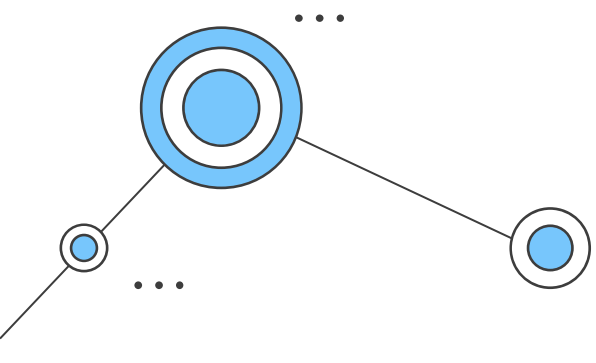
Spring Boot Validation

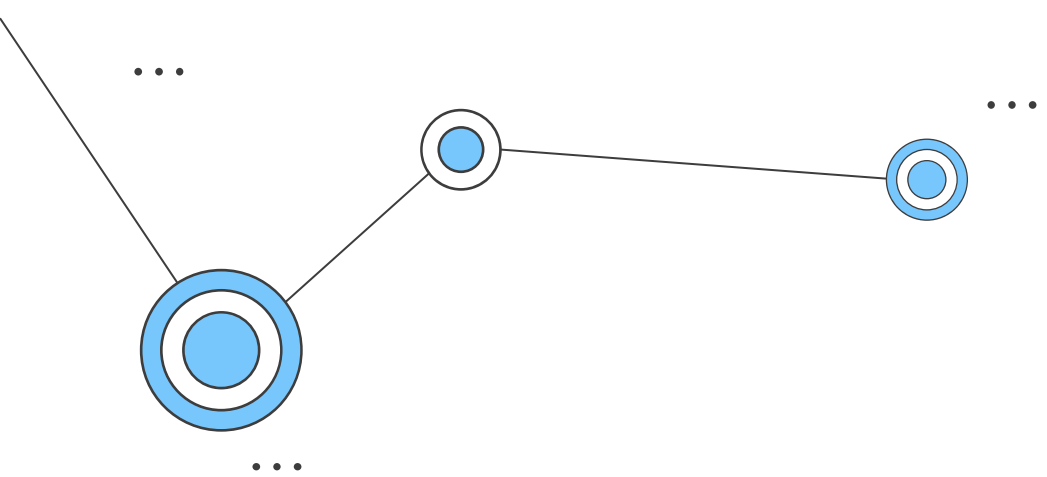
- Description: Spring Boot Validation provides support for data validation in Spring Boot applications.
- Usage: Developers can define validation constraints on domain model attributes using annotations, with Spring Boot Validation automatically validating incoming request data.
- Dependency: spring-boot-starter-validation includes Spring Boot's validation module for data validation.



PostgreSQL

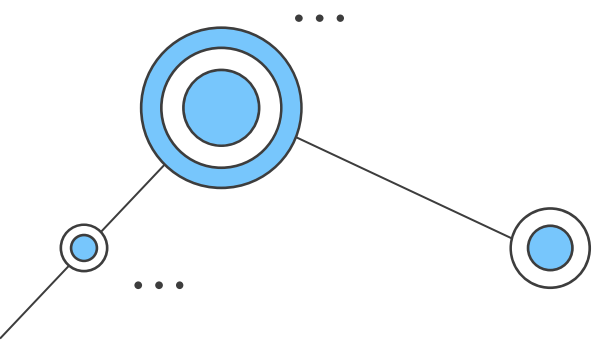
- Description: PostgreSQL is a powerful open-source relational database management system (RDBMS).
- Usage: PostgreSQL is used as the backend database to store application data, such as user information, movie details, etc.
- Dependency: postgresql includes the PostgreSQL JDBC driver required for connecting to a PostgreSQL database.

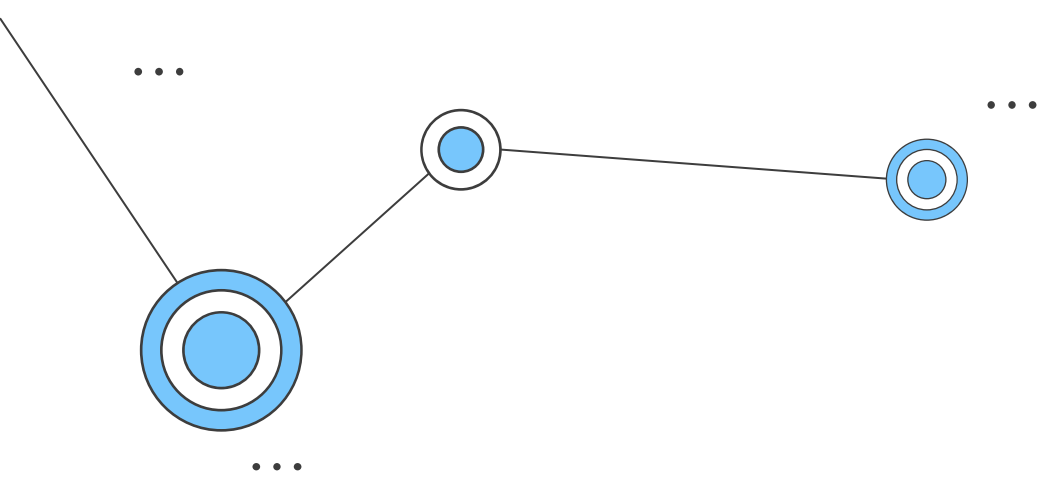




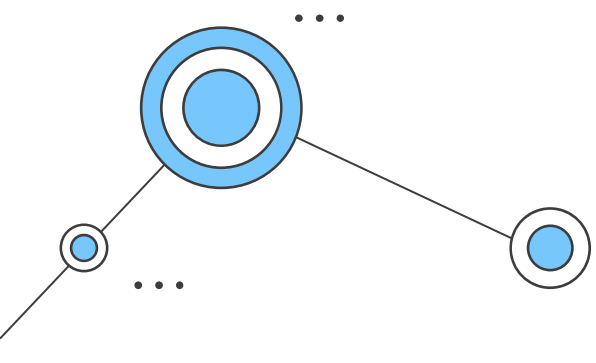
Testing Dependencies

- Description: Testing dependencies include libraries and frameworks for testing Spring Boot applications.
- Usage: These dependencies facilitate unit testing and integration testing of Spring Boot applications.
- Dependencies: spring-boot-starter-test and spring-security-test provide testing support for Spring Boot and Spring Security, respectively.





Maven Plugin



Maven is a build automation tool used primarily for Java projects. It simplifies the process of managing dependencies, compiling code, and packaging applications. Maven uses a project object model (POM) file to describe the structure and dependencies of a project.

- Description: Maven plugins extend the functionality of Maven by executing specific tasks during the build process.
- Usage: The spring-boot-maven-plugin is used to build and package Spring Boot applications into executable JAR or WAR files.
- Plugin: The <build> section configures Maven plugins, with spring-boot-maven-plugin specified to manage the Spring Boot application build process.