JHipster (Spring, Angular)

This **JHipster Training Course with Spring and Angular** will teach you how to build modern web applications and microservices with Spring and Angular using JHipster, which sits in between the two and can generate templates for both the front and back-end. It is also used to control database changes, the architecture of the application and much more.

Prerequisites

- Needless to say, your Java knowledge should be up to scratch – at least on Java 8 Programming level.
- This course assumes and therefore only touches on Web API protocols. If you do not know Web Services, you must do Spring REST API to complete your full-stack.
- This course only touches on Angular. If you do not know Angular, you must do Angular to complete your full-stack.

Further Training

Spring MVC, Spring REST API, Angular, Java Web Services

Contents

Introduction to Modern Web Application Development

- Modern full-stack web development
- Web architecture patterns
- Monolithic web architecture
- Microservice architecture
- Choosing the right pattern
- When to choose a monolithic architecture
- When to choose a microservice architecture

Getting Started with JHipster

- Why JHipster?

- Goal and adoption of JHipster
- Introduction to technologies available
- Client-side technologies
- HTML5 and CSS3
- HTML5
- CSS3
- Sass
- Bootstrap
- MVVM framework
- Angular
- React
- Build tools
- Webpack
- BrowserSync
- Testing tools
- Karma
- Protractor
- Internationalization
- Server-side technologies
- Spring Framework
- Spring Boot
- Spring Security
- Spring MVC
- Spring data
- Security
- JWT
- Session
- OAuth2
- Build tools
- Maven
- Gradle
- Hibernate
- Liquibase
- Caching
- Ehcache
- Hazelcast
- Infinispan

- Swagger
- Thymeleaf
- Dropwizard metrics
- WebSocket
- Kafka
- Testing frameworks
- JUnit
- Gatling
- Cucumber
- Introduction to database options
- SQL databases
- H2
- MySQL
- MariaDB
- PostgreSQL
- MS SQL
- Oracle
- NoSQL databases
- MongoDB
- Cassandra
- Elasticsearch
- Installation and setup
- Prerequisites
- Installation procedure
- Java 8
- Git
- Node.js
- Yarn
- Docker
- IDE configuration
- System setup
- Installation of JHipster

Building Monolithic Web Applications with JHipster

- Application generation
- Step 1 – preparing the workspace

- Step 2 – generating code using JHipster
- Server-side options
- Client-side options
- Internationalization options
- Testing
- Modules
- Code walkthrough
- File structure
- Server-side source code
- Java source
- Resources
- client-side source code
- Starting the application
- Application modules
- Home and Login modules
- Account modules
- Settings
- Password
- Registration
- Admin module
- User management
- Metrics
- Health
- Configuration
- Audits
- Logs
- API
- Running generated tests
- Server-side tests
- Client-side tests

Entity Modeling with JHipster Domain Language

- Introduction to JDL
- DSL grammar for JDL
- Entity modelling with JDL
- Relationship management

- DTO, service, and pagination options
- JDL Studio
- Use case entity model with an explanation
- Entities
- Relationships
- Options for entities
- Entity generation with JHipster
- Generated code walkthrough
- Server-side source code
- Domain class for the entity
- Repository interface for the entity
- Service class for the entity
- Resource class for the entity
- Client side
- TypeScript model class for the entity
- Angular services for the entity
- Angular components of the entity
- Angular route for the entity
- Angular module for the entity
- Generated pages
- Running generated tests

Customization and Further Development

- Live reload for development
- Spring Boot DevTools
- Webpack dev server and BrowserSync
- Setting up live reload for an application
- Customizing the Angular frontend for an entity
- Editing an entity using the JHipster entity sub-generator
- Changing the look and feel of the application
- Adding a new i18n language
- Authorization with Spring Security
- Limiting access to entities
- Limiting access to create/edit/delete entities
- Limiting access to data of other users

Testing and Continuous Integration

- Fixing and running tests
- Continuous integration
- CI/CD tools
- Jenkins
- Travis CI
- GitLab CI
- CircleCI
- Setting up Jenkins
- Creating a Jenkins pipeline using JHipster
- The Jenkinsfile and its stages
- Setting up the Jenkinsfile in a Jenkins server

Going into Production

- An Introduction to Docker
- Docker containers
- The Dockerfile
- The Docker Hub
- Docker compose
- Starting the production database with Docker
- An introduction to Spring profiles
- Packaging the application for local deployment
- Building and deploying using Docker
- Building and deploying an executable archive
- Upgrading to the newest version of JHipster
- An introduction to deployment options supported by JHipster
- Heroku
- Cloud Foundry
- Amazon Web Services
- Production deployment to Heroku cloud

Introduction to Microservice Server-Side Technologies

- Microservice applications versus monoliths
- Building blocks of a microservice architecture

- Service registry
- Service discovery
- Health check
- Dynamic routing and resiliency
- Security
- Fault tolerance and failover
- JHipster Registry
- Netflix Eureka server
- Spring cloud config server
- HashiCorp Consul
- Service discovery
- Health discovery
- K/V store
- Multiple data centers
- JHipster Gateway
- Netflix Zuul
- Hystrix
- JHipster Console
- Elasticsearch
- Logstash
- Kibana
- Zipkin
- Prometheus
- JHipster UAA server

Building Microservices with JHipster

- Application architecture
- Gateway application generation
- Converting a monolithic application to a microservice gateway
- Application generation
- Generating a new Gateway
- Gateway configuration
- JWT authentication
- How JWT works
- Microservice application – Invoice Service with MySQL

database
- Application generation
- Microservice configuration
- Microservice application – notification service with NoSQL
- database
- Application generation
- Microservice configuration

Working with Microservices

- Setting up JHipster Registry locally
- Using a pre-packaged WAR file
- Building from source
- Docker mode
- Running a generated application locally
- Gateway application pages
- JHipster Registry pages
- System status
- Below renew threshold
- Instances registered
- General info and health
- Application listing page
- Metrics page
- Health page
- Configuration page
- Logs page
- Swagger API endpoints
- Running invoice and notification applications locally
- Modeling entities in JDL
- Entity generation on microservices
- Explaining the generated code
- Gateway application
- Explaining the generated pages

Deploying with Docker Compose

- Introducing microservice deployment options

- A short introduction to Docker Compose
- Kickstarting Kubernetes
- Introducing OpenShift
- Explaining Rancher
- Generated Docker Compose files
- Walking through the generated files
- Building and deploying everything to Docker locally
- Generating docker-compose files for microservices
- Features of the deployed application
- JHipster console demo
- Scaling up with Docker Swarm

Deploying to the Cloud with Kubernetes

- Generating Kubernetes configuration files with JHipster
- Walking through the generated files
- Deploying the application to Google Cloud with Kubernetes

Using React for the Client-Side

- Generating an application with React client side
- Technical stack and source code
- Technical stacks
- Using TypeScript
- State management with Redux and friends
- Routing with React Router
- HTTP requests using Axios
- Bootstrap components using Reactstrap
- Unit testing setup
- Generating source code
- Generating an entity with React client side

Best Practices with JHipster

- The next steps to pursue
- Adding a shopping cart for the application
- Improving end-to-end tests
- Improving the CI/CD pipeline

- Building a JHipster module
- Best practices to keep in mind
- Choosing a client-side framework
- Choosing a database option
- Architecture considerations
- Security considerations
- Deployment and maintenance
- General best practices

Duration and pricing

In Price Group A

Certificate

Read about our certificates

Bookings

You can download the course registration form on our home page or by clicking here

Brochure
You may download a pdf copy of this page by clicking on the pdf icon at the top of the page.

Questions
Please email us

Schedule

On the calendar below. If your browser doesn’t display the calendar below, please click on this link or try using Google Chrome, alternatively please enquire via our Contact Us page.