

# course Esilv - Angular & Node Js

## part 4: Advanced Web Features

[arnaud.nauwynck@gmail.com](mailto:arnaud.nauwynck@gmail.com)

# Outline

- Some Advanced features
  - AG-Grid
  - HighCharts
  - Code Scaffolding
  - Generating (Angular) Client from Swagger
  - Security: Cookie, Login, OAuth2, JWT, ...
  - RxJs, Auto completion, WebSocket
- Final Words : Angular - React - Vue.Js - Solid.Js

# Reminder

**4 x CMs - Lessons**

**5 x TDs - Hands on**

**1 Project submission .... deadline 31 dec**

Reminder ...  
expectation on project submission for Exam

**Your project MUST include at least  
1 advanced feature**

# Outline

- Some Advanced features



- AG-Grid

- HighCharts

- Code Scaffolding

- Generating (Angular) Client from Swagger

- Security: Cookie, Login, OAuth2, JWT, ...

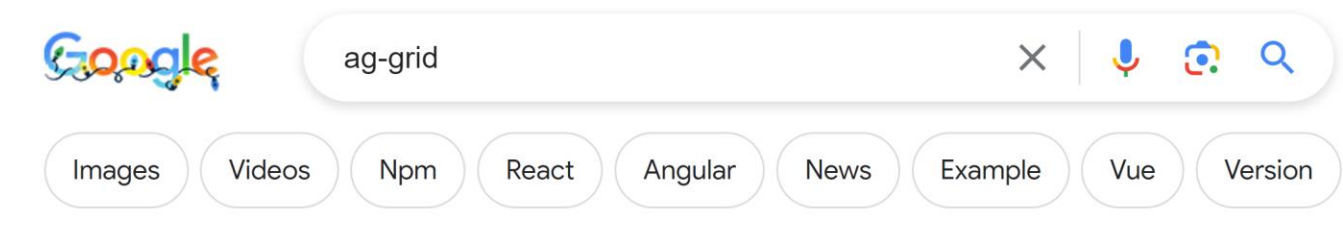
- RxJs, Auto completion, WebSocket

- Final Words : Angular - React - Vue.Js - Solid.Js

<https://www.ag-grid.com/>



# AG-Grid



About 164,000,000 results (0.48 seconds)



AG Grid

<https://www.ag-grid.com>

## Data Grid: AG Grid: High-Performance React Grid, Angular ...

**AG Grid** is a feature rich datagrid designed for the major JavaScript Frameworks. Version 31. 0.

Download v31 of the best Data Grid in the world now.

## Angular Data Grid

Our documentation will help you to get up and running with AG Grid ...

## React Data Grid: Documentation

Our documentation will help you to get up and running with AG Grid ...

## Demo

AG Grid is a feature-rich datagrid available in Community or ...

## Angular Data Grid: Quick Start

Angular Data GridQuick Start. angular logo · Container: for the ...

# https://ag-grid.com

← → ↻ ag-grid.com









Introducing our new AG Charts library! [Visit →](#)

**AG Grid**





Demo Docs API Blog Pricing

## The Best JavaScript Grid in the World

The professional choice for developers building enterprise applications

Stock	Timeline	Current	Last	% Change
SID capital		\$8.10	\$105.81	↓ 1205.85%
Uptick 500		\$52.11	\$125.17	↓ 140.19%
Xingyun BA		\$87.94	\$15.59	↑ 82.27%
Wiltshire 4500-kbm		\$66.56	\$115.18	↓ 73.04%
DT PI		\$55.07	\$118.92	↓ 115.94%
ESTA-MID		\$139.84	\$65.94	↑ 52.84%
Capra ibex		\$134.77	\$53.05	↑ 60.64%
NY composite index		\$28.46	\$36.92	↓ 29.72%

Get started

 React  Angular  Vue  JS Javascript



# Getting Started

The screenshot shows the AG Grid documentation website for Angular. The browser address bar shows the URL `ag-grid.com/angular-data-grid/getting-started/`. The page has a dark blue header with the AG Grid logo, a search bar, and navigation links for Demo, Docs, API, Blog, Pricing, and a user profile icon. A left sidebar contains a table of contents with sections like Overview, Get Started (with 'Quick Start' highlighted), Installation, Guides, Migration, Layout & Styling, Charting, and Core Features. The main content area is titled 'Angular Data Grid Quick Start' and includes a sub-header 'Create a grid in 60 Seconds'. It lists three requirements for creating a grid: Container, Styles, and Row Data & Column Definitions. Below this is an 'Install' section with the command `npm install ag-grid-angular`. The 'Create a Component' section shows a code snippet for an Angular component that uses AG Grid modules. A right sidebar lists the 'Quick Start' steps: Install, Create a Component, Row Data & Column Definitions, Rendering the Grid, Styling the Grid, Result, and Next Steps.

Angular Data Grid  
Quick Start

Create a grid in 60 Seconds

At a minimum, three things are required to create a grid:

- **Container:** for the grids placement in your application.
- **Styles:** to define the grid's theme & dimensions.
- **Row Data & Column Definitions:** to define the data and how it should be displayed.

**Install**

First, install the `ag-grid-angular` library:

```
npm install ag-grid-angular
```

**Create a Component**

Then, create a new component in your application with the required dependencies:

```
import { Component } from '@angular/core';
import { AgGridModule } from 'ag-grid-angular'; // Angular Grid Logic
import { ColDef } from 'ag-grid-community'; // Column Definitions Interface

@Component({
  selector: 'app-root',
  standalone: true,
  imports: [AgGridModule], // Add AG Grid Module to component
  styleUrls: ['./app.component.css'],
  template: ``
})


export class AppComponent {}
```


**Quick Start**


- Install
- Create a Component
- Row Data & Column Definitions
- Rendering the Grid
- Styling the Grid
- Result
- Next Steps

# https://ag-grid.com/example

← → ↻ ag-grid.com/example/ 🔒 ⌵ ☆ 🗖 A ⋮

**AG Grid** 

[Demo](#) [Documentation](#) [Pricing](#) [Blog](#) 

Data Size: 100000 Rows, 22 Co ⌵ Theme: Alpine ⌵ Filter:   [Take a video tour](#)

⋮ Drag here to set row groups

Participant			Game of Choice		Performance >		Rating
<input type="checkbox"/> Name	Language	Country	Game Name ⌵	Bought	Bank Balance		
<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="blu"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> Olivia Fisher	Maltese	🇲🇹 Malta	Tablut	×	\$79,428	★★	
<input type="checkbox"/> Lucy Beckham	French	🇫🇷 France	Tablut	×	\$80,350		
<input type="checkbox"/> Tony Hudson	Spanish	🇻🇪 Venezuela	Tablut	×	\$67,068	★★★	
<input type="checkbox"/> Lily Cage	Spanish	🇪🇸 Spain	Tablut	×	\$85,532	★★	
<input type="checkbox"/> Grace Kane	French	🇫🇷 France	Tablut	×	\$23,417	★★★★	
<input type="checkbox"/> Andrew Dalton	Portuguese	🇧🇷 Brazil	Tablut	×	\$84,588	★	
<input type="checkbox"/> Chloe Flanagan	French	🇫🇷 France	Tablut	×	\$60,967	★★	
<input type="checkbox"/> Ruby Greyson	French	🇫🇷 France	Tablut	×	\$2,692	★	

Rows: 1,470 of 100,000

☐ Pivot Mode

⌵ ☒

⌵ ☒ Participant

☒ Name

☒ Language

☒ Country

⌵ ☒ Game of Choice

☒ Game Name

☒ Bought

⌵ ☒ Performance

☒ Bank Balance

⋮ Row Groups



Drag here to set row groups

Σ Values

Drag here to aggregate

Columns ⌵ Filters

Did you notice? "Rows ... of 100,000"

<input type="checkbox"/>	Chloe Hanagan	French	 France
<input type="checkbox"/>	Ruby Greyson	French	 France
<div><div></div></div>			
Rows: 1,470 of 100,000			

# AG-Grid performances

## Naive

create millions of  
<div></div>  
<div></div>  
<div></div>  
<div></div>  
<div></div>

Use Browser to scroll page



## Virtual Scrolling

create ~50 only if visible  
<div ..translateY(0px)>  
<div ..translateY(41px)>  
<div ..translateY(82px)>

change "div.y" position  
when scrolling  
Scroll

# Your-own-table vs AG-Grid

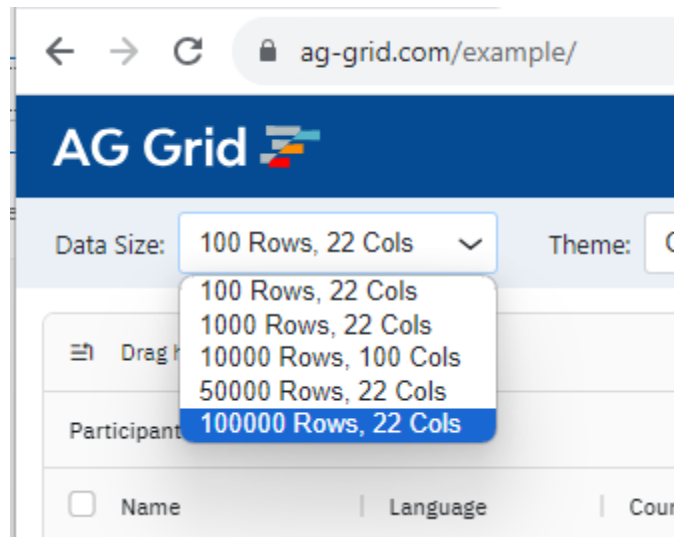


Name	Username	Email	Phone	Website
Sanna Cochran	Bret	Glovervillage.biz	1-778-796-8001 x56442	Midagood.org
Ervin Howell	Antonette	Shanemadness.biz	016-462-4581 x89125	amertalia.net
Clementine Bauch	Sonoma	Herthavilleysa.net	1-463-723-4447	randia.info
Patricia Lebsack	Karina	Julliana.DConner@bory.org	480-770-9023 x795	kale.biz
Chelsey Dietrich	Kenneth	Louis_Plattinger@amita.ca	(204)856-1239	demarco.info



```
<div *ngFor="let row of rows">
  <div class="row">{{row}}</div>
</div>
```

# Try online yourself



Then open Chrome DevTools

# Virtual Scroll

```
<div "style="transform: translateY(82px); ">
```

The screenshot displays the AG-Grid demo application. The table lists participants and their preferred games. The row for Kevin Flanagan is selected. A tooltip indicates the row's class names. The DevTools console shows the row's HTML structure, highlighting the 'style' attribute 'transform: translateY(0px);' on the first row and 'transform: translateY(41px);' on the second row.

Name	Language	Country	Game Name
Kevin Flanagan	Spanish	Uruguay	Rithmomachy
Bricker McGee	French	France	Kalah
Dimple Unalkat	Portuguese	Portugal	Game of the Generals
Gil Lopes	Spanish	Colombia	Hare and Hounds
Sophie Beckham	English	Ireland	Sugoroku
Isabelle Black	French	France	Nine Men's Morris
Emily Braxton	Maltese	Malta	Blockade
Olivia Brennan	French	France	Patolli
Lily Brock	Italian	Italy	YINSH
Chloe Bryson	Greek	Greece	Downfall
Isabella Cadwell	English	Ireland	Gipf
Amelia Cage	English	Ireland	Shogi

# AG-grid virtual-scroll

The image displays the AG-Grid virtual scroll example, showing the DOM structure and the grid interface.

**DOM Structure (Elements Panel):**

```
<div class="ag-pinned-right-cols-container ag-hidden" role="presentation" aria-hidden="true" style="height: 4.1e+06px; width: 0px; max-width: 0px; min-width: 0px;"></div>  
<!-- AG Row Container fullWidth -->  
<div class="ag-full-width-container" role="presentation" style="height: 4.1e+06px;"></div>  
<!-- AG Fake Vertical Scroll -->  
<div class="ag-body-vertical-scroll" aria-hidden="true" style="width: 21px; max-width: 21px; min-width: 21px;"> flex == $0  
  <div class="ag-body-vertical-scroll-viewport" ref="eViewport" style="width: 21px; max-width: 21px; min-width: 21px;"> ...</div>  
</div>  
<!-- AG Sticky Top -->  
<div class="ag-sticky-top" role="presentation" style="height: 0px; top: 142px; width: calc(100% - 21px);"> ... flex</div>  
<!-- AG Pinned Bottom -->  
<div class="ag-floating-bottom" role="presentation" style="height: 0px; min-height: 0px;"></div>
```

**Grid Interface (Left Panel):**

The grid shows a table with columns: Name, Language, Country, Game of Choice, Game Name, Bought, Performance, Bank Balance, Extra Info 1, Extra Info 2, Rating, Total Winnings, and Monthly Breakdown. The table is scrollable, and the DOM structure shows the virtual scroll implementation.

**Accessibility (Left Panel):**

div.ag-body-vertical-scroll-viewport  
ort  
Color #181D1F  
Font 14px "IBM Plex Sans", -apple-system, Blink...

**Accessibility Tree (Left Panel):**

- Name
- Language
- Country
- Game of Choice
- Game Name
- Bought
- Performance
- Bank Balance
- Extra Info 1
- Extra Info 2
- Rating
- Total Winnings
- Monthly Breakdown
- Jan
- Feb
- Mar
- Apr

**Row Groups (Left Panel):**

Row Groups

**DOM Structure (Elements Panel):**

div.ag-body-vertical-scroll-viewport  
div.ag-body-vertical-scroll  
div.ag-sticky-top  
div.ag-floating-bottom



# AG-Grid Features

## CORE FEATURES

- › Columns
- › Rows
- › Filtering
- › Selection
- › Rendering
- › Editing
- › Client-Side Data
- › Scrolling
- › Interactivity


---

## ADVANCED FEATURES

- › Group & Pivot
- › Master Detail
- › Accessories
- › Server-Side Data
- › Import & Export
- › Components

# Demo Example : Sorting, Export to Excel


# Outline

- Some Advanced features
  - AG-Grid
  -  • HighCharts
  - Code Scaffolding
  - Generating (Angular) Client from Swagger
  - Security: Cookie, Login, OAuth2, JWT, ...
  - RxJs, Auto completion, WebSocket
- Final Words : Angular - React - Vue.Js - Solid.Js

# HighCharts

<https://www.highcharts.com>


# google HighCharts



ImagesVideosAngularPythonFreeExamplesLine chartNpmOnline

All filtersToolsSafeSearch

About 108,000,000 results (0.37 seconds)

**Highcharts**  
<https://www.highcharts.com>

## Highcharts: Interactive charting library

The Only Charting Library You Need. The **Highcharts** library comes with all the tools you need to create reliable and secure data visualizations. Built on ...

Results from highcharts.com

### Highcharts demos

Check out Highcharts demos and examples to learn how to create ...

### Highcharts Documentation


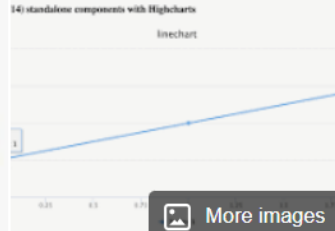

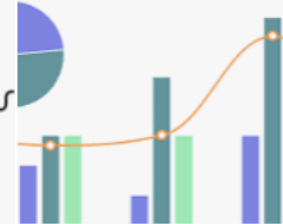
Charting ... Our core library. Includes all standard chart ...

### Download

Installation - Highcharts file service - Chart Chooser - Forum - Demo

### Highcharts Core Charting Library

Highcharts, the core library of our product suite, is a pure ...

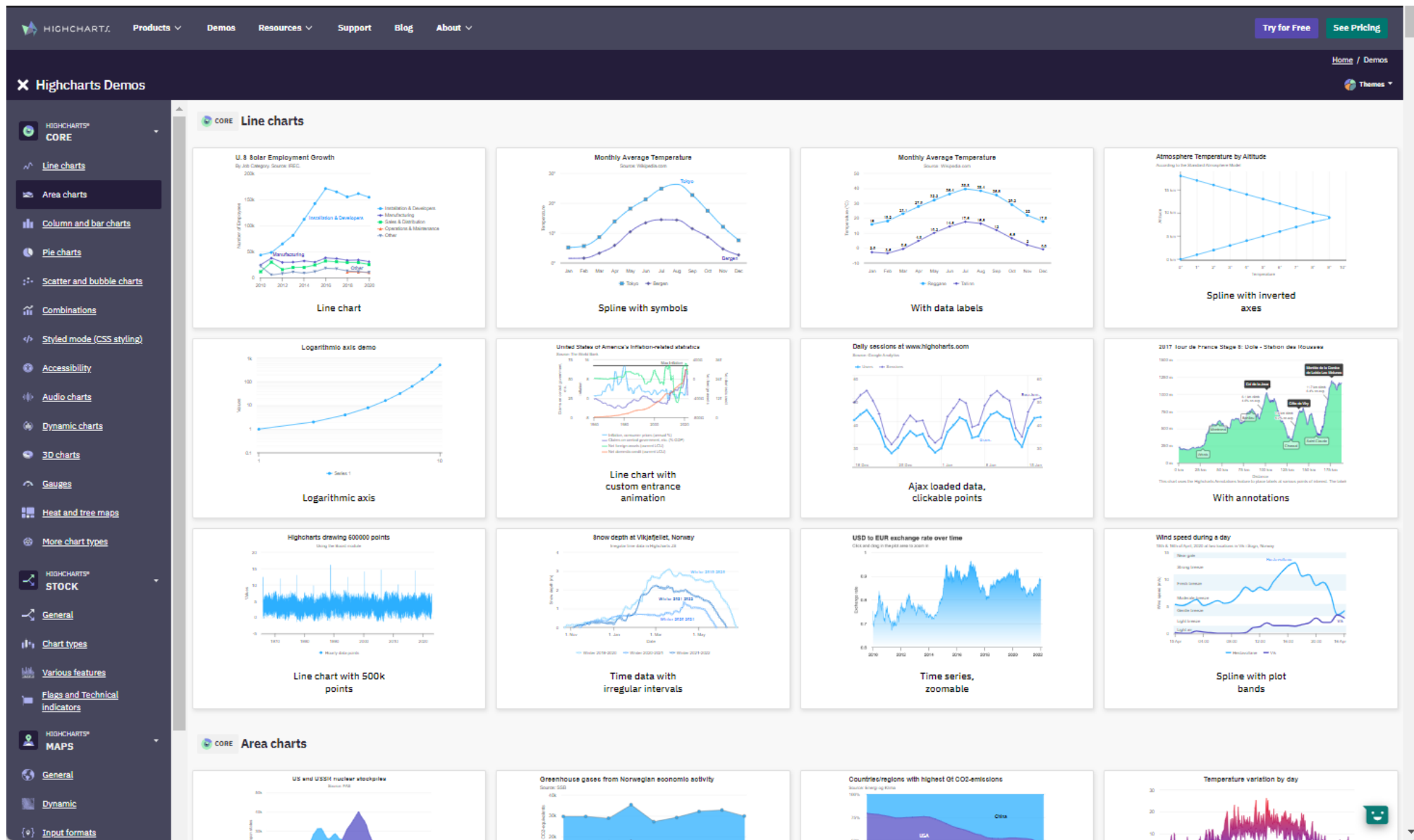
**HIGHCHARTS**

**Highcharts**  
Software

Highcharts is a software library for charting written in pure JavaScript, first released in 2009. The license is proprietary. It is free for personal/non-commercial uses and paid for commercial applications. [Wikipedia](#)

People also ask :

# <https://www.highcharts.com/demo>



# HighChart Features



HIGHCHARTS®  
CORE



[Line charts](#)



[Area charts](#)



[Column and bar charts](#)



[Pie charts](#)



[Scatter and bubble charts](#)



[Combinations](#)



[Styled mode \(CSS styling\)](#)



[Accessibility](#)



[Audio charts](#)



[Dynamic charts](#)



[3D charts](#)



[Gauges](#)



[Heat and tree maps](#)



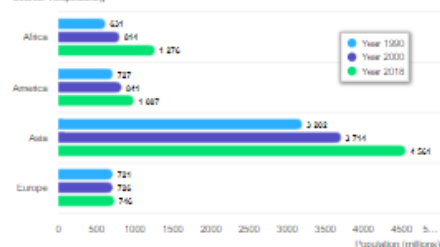
[More chart types](#)



CORE Column and bar charts

Historic World Population by Region

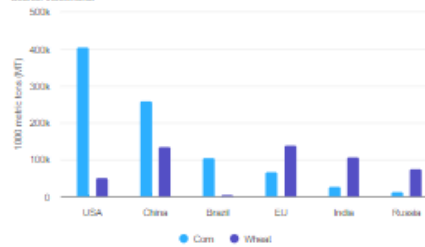
Source: Wikipedia.org



Basic bar

Corn vs wheat estimated production for 2020

Source: indonemundi



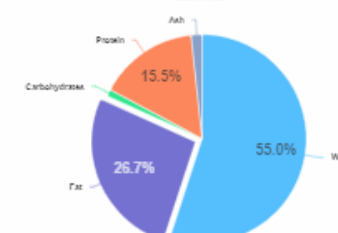
Basic column



CORE Pie charts

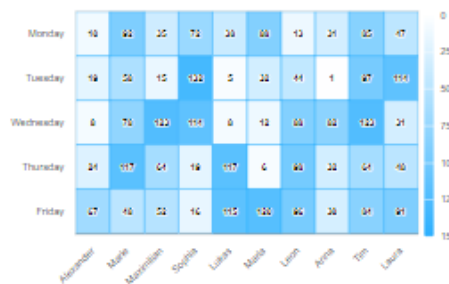
Egg Yolk Composition

Source: MCF1



Pie chart

Sales per employee per weekday



Heat map

Day temperature in Oslo, Norway July 2023

Temperature variation all day through July



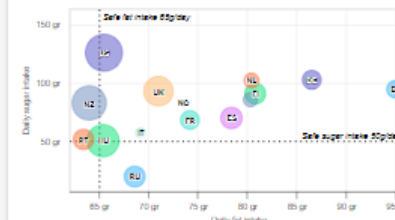
Calendar Heatmap



CORE Scatter and bubble charts

Sugar and fat intake per country

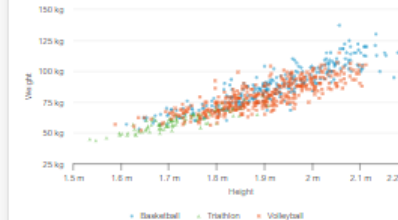
Source: Eurostat and OECD



Bubble chart

Olympics athletes by height and weight

Source: The Guardian



Scatter plot

# https://github.com/highcharts/highcharts-angular

The screenshot shows the GitHub repository page for `highcharts/highcharts-angular`. The repository is public and has 408 stars, 115 forks, and 31 watchers. The main content area displays a list of files and folders, including `.github/ISSUE_TEMPLATE`, `e2e`, `highcharts-angular`, `js`, `src`, `tasks`, `.browserslistrc`, `.editorconfig`, and `.gitignore`. The right sidebar shows the repository's about section, including the README, license, activity, and releases. The latest release is `v4.0.0`, marked as the latest version, and was released 3 weeks ago.

highcharts / highcharts-angular

Type to search

Code Issues 9 Pull requests Actions Projects Security Insights

highcharts-angular Public

Watch 31 Fork 115 Star 408

master 8 branches 25 tags

Go to file Add file Code

karolokolodziej Changed name in the readme. ce78091 4 days ago 224 commits

.github/ISSUE_TEMPLATE	feat: Add issue templates. Closes #100.	5 years ago
e2e	updated to angular 11	3 years ago
highcharts-angular	chore(release): 4.0.0	3 weeks ago
js	feat: Added reference to @highcharts/map-collection and updated doc...	5 years ago
src	Updated highcharts.	last month
tasks	Updated produciton build.	last year
.browserslistrc	Upadted demo to Angular 15.	last year
.editorconfig	Created branch for do-it-yourself.	6 years ago
.gitignore	Updated to Angular 13.	2 years ago

About

Highcharts official integration for Angular

- Readme
- View license
- Activity
- 408 stars
- 31 watching
- 115 forks

Report repository

Releases 14

v4.0.0 Latest 3 weeks ago

+ 13 releases



# Getting Started

## npm install -s highcharts-angular

### Installing

Get package from NPM in your Angular app:

```
npm install highcharts-angular --save
```



In your app.module.ts add the HighchartsChartModule:

```
...  
import { HighchartsChartModule } from 'highcharts-angular';  
  
@NgModule({  
  imports: [  
    ...  
    HighchartsChartModule
```



# <https://github.com/highcharts/highcharts-angular#hello-world-demo>

## Hello world demo

To create a simple demo start with [installing](#).

Next for `app.component.ts` 's HTML template use:

```
<highcharts-chart
  [Highcharts]="Highcharts"
  [options]="chartOptions"

  style="width: 100%; height: 400px; display: block;"
></highcharts-chart>
```

and export variables:

```
export class AppComponent {
  Highcharts: typeof Highcharts = Highcharts;
  chartOptions: Highcharts.Options = {
    series: [{
      data: [1, 2, 3],
      type: 'line'
    }]
  };
  ...
}
```



# Notice: Alternatives To HighCharts

**HTML <SVG>**

**HTML canvas**

builtin features, but too low-level to use yourself

**d3js**


also low-level and very complex

(redundant with Angular \*ngFor="let .. of ... ")

**chart.js**

**& MANY others libraries**

# Outline

- Some Advanced features
  - AG-Grid
  - HighCharts
  -  • Code Scaffolding
  - Generating (Angular) Client from Swagger
  - Security: Cookie, Login, OAuth2, JWT, ...
  - RxJs, Auto completion, WebSocket
- Final Words : Angular - React - Vue.Js - Solid.Js

# Code Scaffolding

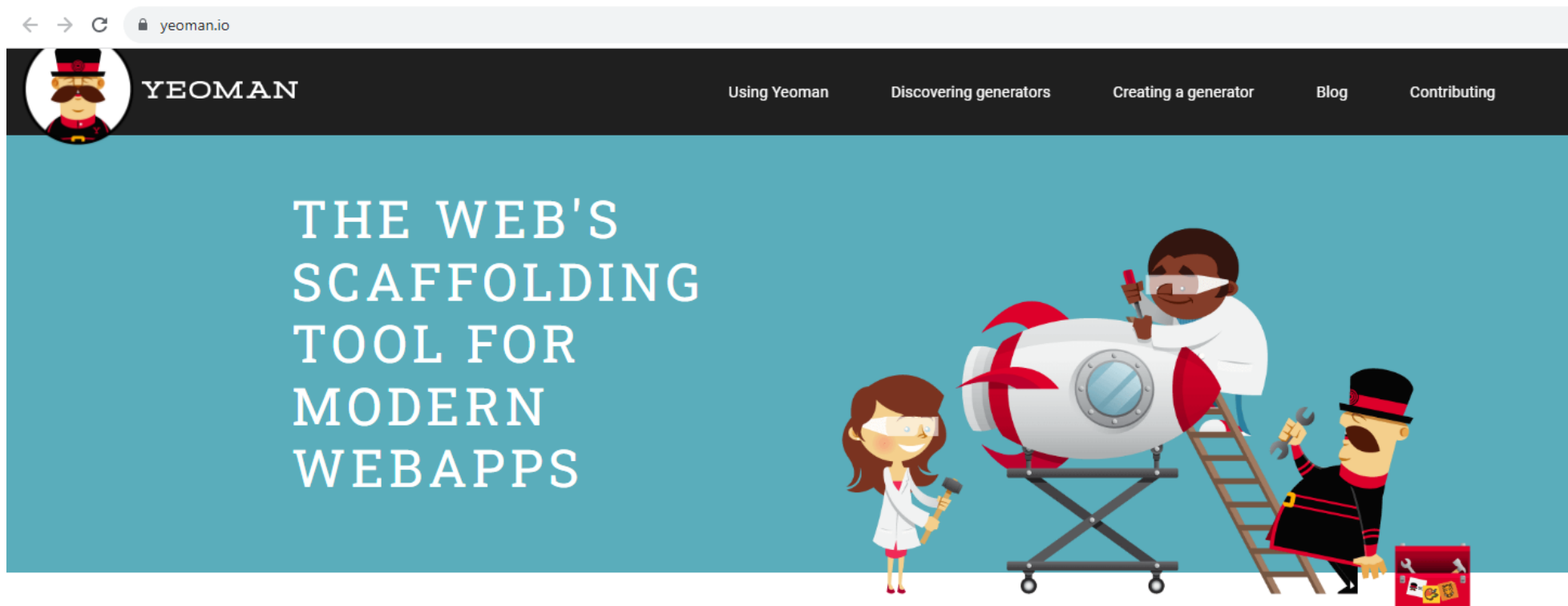
\$ ng new

\$ ng g c = \$ ng generate component

Your first scaffolding tools !  
a.k.a Code Generator

# More Scaffolding.. Yeoman

## <https://yeoman.io>



[Get started](#) and then [find a generator](#) for your webapp.  
Generators are available for [Angular](#), [Backbone](#), [React](#), [Polymer](#)  
and over [5600+ other projects](#).

One-line install using [npm](#):

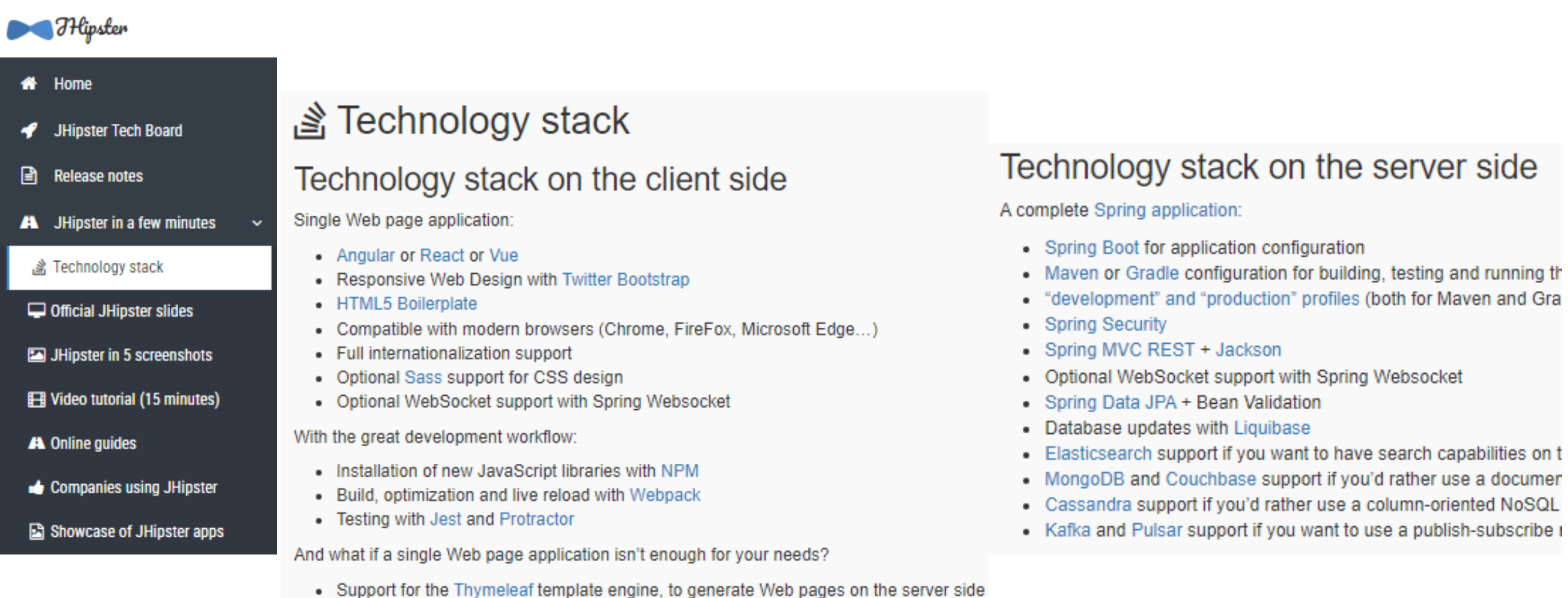
```
npm install -g yo
```

# More Scaffolding ... JHipster

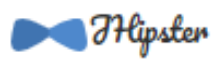
<https://jhipster.tech>



# JHipster in Few minutes > Technology Stack



The screenshot shows the JHipster website interface. On the left is a dark sidebar with navigation links: Home, JHipster Tech Board, Release notes, JHipster in a few minutes (with a dropdown arrow), Technology stack (highlighted with a white background), Official JHipster slides, JHipster in 5 screenshots, Video tutorial (15 minutes), Online guides, Companies using JHipster, and Showcase of JHipster apps. The main content area has a light gray header for 'Technology stack' with a stack of books icon. Below this, the section 'Technology stack on the client side' is highlighted. It lists technologies for a 'Single Web page application' and a 'great development workflow'. To the right, the 'Technology stack on the server side' section lists technologies for a 'complete Spring application'.

 JHipster

Home

JHipster Tech Board

Release notes

JHipster in a few minutes

Technology stack

Official JHipster slides

JHipster in 5 screenshots

Video tutorial (15 minutes)

Online guides

Companies using JHipster

Showcase of JHipster apps

## Technology stack

### Technology stack on the client side

Single Web page application:

- [Angular](#) or [React](#) or [Vue](#)
- Responsive Web Design with [Twitter Bootstrap](#)
- [HTML5 Boilerplate](#)
- Compatible with modern browsers (Chrome, FireFox, Microsoft Edge...)
- Full internationalization support
- Optional [Sass](#) support for CSS design
- Optional WebSocket support with Spring WebSocket

With the great development workflow:

- Installation of new JavaScript libraries with [NPM](#)
- Build, optimization and live reload with [Webpack](#)
- Testing with [Jest](#) and [Protractor](#)

And what if a single Web page application isn't enough for your needs?

- Support for the [Thymeleaf](#) template engine, to generate Web pages on the server side

### Technology stack on the server side

A complete [Spring](#) application:

- [Spring Boot](#) for application configuration
- [Maven](#) or [Gradle](#) configuration for building, testing and running the application
- "development" and "production" profiles (both for Maven and Gradle)
- [Spring Security](#)
- [Spring MVC REST + Jackson](#)
- Optional WebSocket support with Spring WebSocket
- [Spring Data JPA](#) + Bean Validation
- Database updates with [Liquibase](#)
- [Elasticsearch](#) support if you want to have search capabilities on the server side
- [MongoDB](#) and [Couchbase](#) support if you'd rather use a document-oriented NoSQL database
- [Cassandra](#) support if you'd rather use a column-oriented NoSQL database
- [Kafka](#) and [Pulsar](#) support if you want to use a publish-subscribe messaging system



# JHipster Slides - Front End Technologies

<https://www.jhipster.tech/presentation/#/>



## NPM

- Fast and reliable dependency management
- Used to install and run all client-side tools

```
npm install -g generator-jhipster
```

## Yeoman

### YEOMAN PROVIDES APPLICATION *GENERATORS*

- Hundreds of generators are available
- Mostly geared toward JavaScript front-end applications
- The top-rated generators have excellent quality

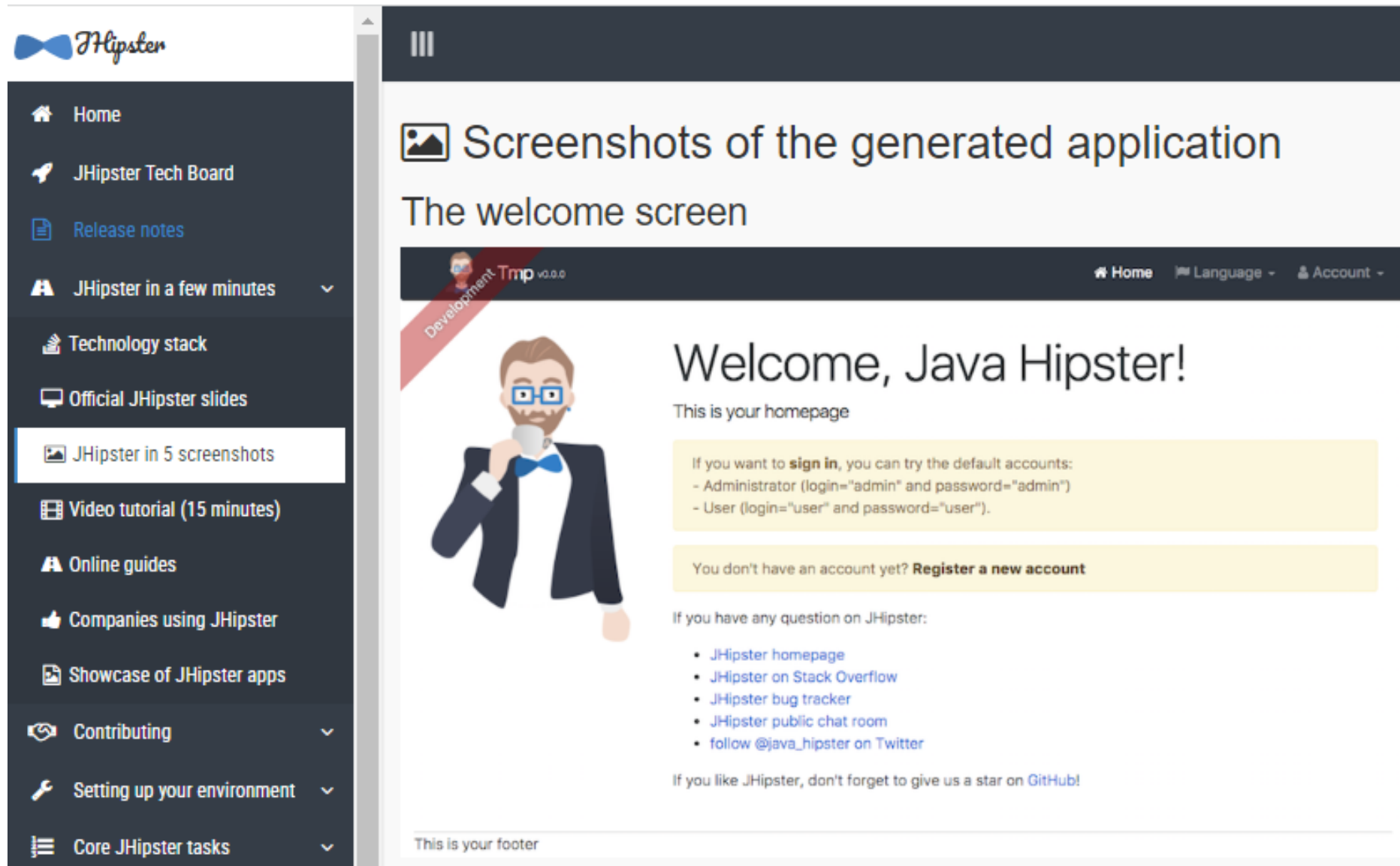
jhipster

## Angular and React

### THE 2 MOST POPULAR JAVASCRIPT FRAMEWORKS

- Both Angular and React are supported by JHipster
- Powerful & easy to learn
- Data binding, form validation, i18n... all out of the box

# JHipster in 5 Screenshots: 1/5



The image shows a screenshot of the JHipster website. On the left is a dark sidebar with a navigation menu. The main content area on the right shows the 'Screenshots of the generated application' section, specifically 'The welcome screen'. The welcome screen features a cartoon character, a 'Welcome, Java Hipster!' message, and instructions on how to sign in or register.

**Navigation Menu (Left Sidebar):**

- Home
- JHipster Tech Board
- Release notes
- JHipster in a few minutes
- Technology stack
- Official JHipster slides
- JHipster in 5 screenshots**
- Video tutorial (15 minutes)
- Online guides
- Companies using JHipster
- Showcase of JHipster apps
- Contributing
- Setting up your environment
- Core JHipster tasks

**Main Content Area:**

## Screenshots of the generated application

### The welcome screen

**Development Tmp v0.0.0** Home Language Account

# Welcome, Java Hipster!

This is your homepage

If you want to **sign in**, you can try the default accounts:

- Administrator (login="admin" and password="admin")
- User (login="user" and password="user").

You don't have an account yet? **Register a new account**

If you have any question on JHipster:


- [JHipster homepage](#)
- [JHipster on Stack Overflow](#)
- [JHipster bug tracker](#)
- [JHipster public chat room](#)
- [follow @java\\_hipster on Twitter](#)

If you like JHipster, don't forget to give us a star on [GitHub!](#)

This is your footer

# JHispter in 5 Screenshots: 2/5

## The generated “Book” entity

Development Tmp v0.0.0

[Home](#) [Entities](#) [Administration](#) [Language](#) [Account](#)

### Books

[+ Create a new Book](#)


ID	Title	Description	Publication Date	Price	Author	
2	Inferno	Inferno is a 2013 mystery thriller novel by American author Dan Brown and the fourth book in his Robert Langdon series, following Angels & Demons, The Da Vinci Code and The Lost Symbol.	May 14, 2013	45	Dan Brown	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
4	King Lear	King Lear is a tragedy written by William Shakespeare. It depicts the gradual descent into madness of the title character.	Dec 30, 1604	10	William Shakespeare	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
5	Gulliver's Travels	It is Swift's best known full-length work, and a classic of English literature.	Oct 26, 1726	15	Jonathan Swift	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>

Showing 1 - 3 of 3 items.

[1](#)

# JHispter in 5 Screenshots: 3/5

One of the generated forms

 **JHispter** v0.0.0

[Home](#) [Entities](#) [Administration](#) [Language](#) [Account](#)

## User settings for [admin]

First Name

Last Name

E-mail

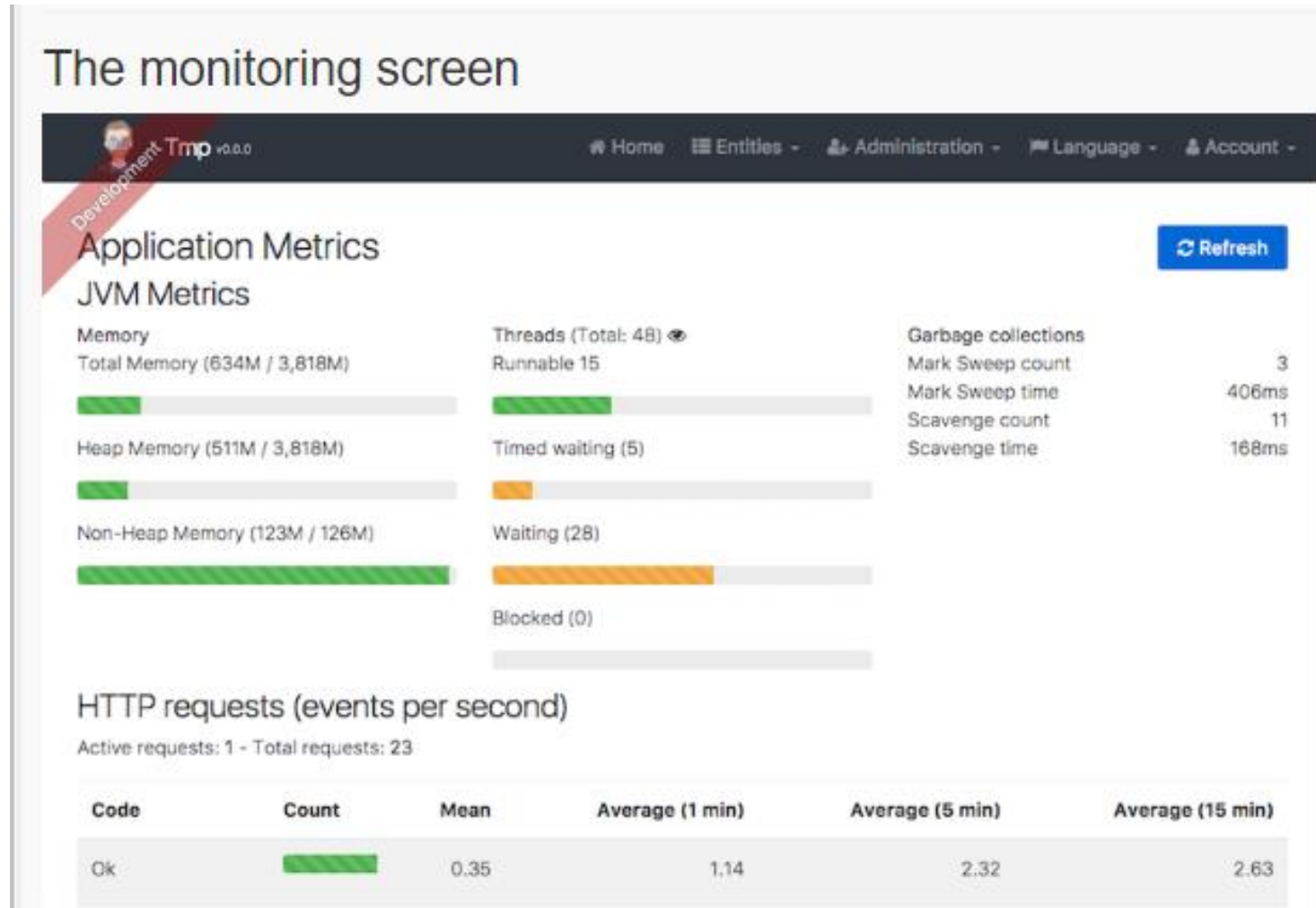
Your e-mail is required.

Language

Save

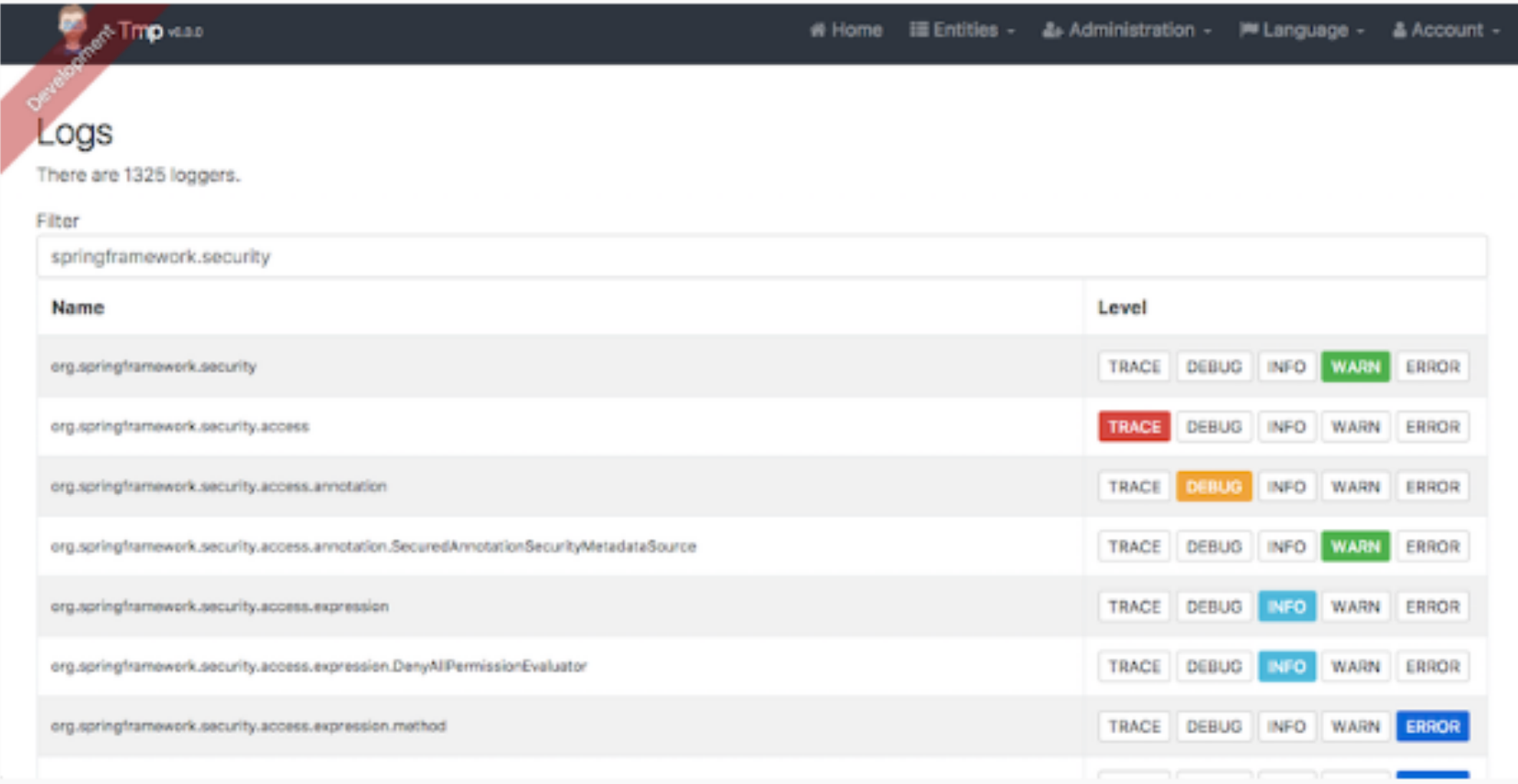
This is your footer

# JHispter in 5 Screenshots: 4/5



# JHispter in 5 Screenshots: 5/5

## The log management screen



The screenshot shows the 'Logs' management interface in JHispter. At the top, there's a dark navigation bar with links for Home, Entities, Administration, Language, and Account. Below this, a red banner indicates 'Development Tmp v0.3.0'. The main section is titled 'Logs' and states 'There are 1325 loggers.' A filter input field contains 'springframework.security'. Below the filter is a table listing loggers and their log levels. The table has two columns: 'Name' and 'Level'. The 'Level' column contains buttons for TRACE, DEBUG, INFO, WARN, and ERROR, with one button highlighted in each row to indicate the current log level.

Name	Level
org.springframework.security	TRACE DEBUG INFO <b>WARN</b> ERROR
org.springframework.security.access	<b>TRACE</b> DEBUG INFO WARN ERROR
org.springframework.security.access.annotation	TRACE <b>DEBUG</b> INFO WARN ERROR
org.springframework.security.access.annotation.SecuredAnnotationSecurityMetadataSource	TRACE DEBUG INFO <b>WARN</b> ERROR
org.springframework.security.access.expression	TRACE DEBUG <b>INFO</b> WARN ERROR
org.springframework.security.access.expression.DenyAllPermissionEvaluator	TRACE DEBUG <b>INFO</b> WARN ERROR
org.springframework.security.access.expression.method	TRACE DEBUG INFO WARN <b>ERROR</b>

# JHipster Demo

(case 1/ with local installation)

```
$ npm install -g generator-jhipster
```


```
$ jhipster  
.... prompt....
```

```
$ jhipster add ...
```

```
$ jhipster jdl <<your-file>>.jdl
```

# case 2 / with JHipster Online


## <https://start.jhipster.tech/generate-application>


 **JHipster Online** v2.26.0


JHipster Online is the best place to generate **JHipster** applications, with no installation required!


No more command line: fill an easy-to-use Web form, JHipster Online generates a complete **JHipster** application on your GitHub account.

Check out **JHipster statistics** to have an overview of what JHipster users are building.

 [Create Application](#)

 [Create Azure Application](#)

 [JDL Studio](#)

 [JHipster Statistics](#)

### Application generation

#### Project configuration

Application name

hipsterSampleApplication

Repository name

hipster-sample-application

#### Application type

Which type of application would you like to create?

Monolithic application (recommended for simple projects) ▼

#### Server side options

What is your default Java package name?

com.mycompany.myapp

Download as Zip file

Reset



# <https://start.jhipster.tech/jdl-studio>

← → ↺ start.jhipster.tech/jdl-studio/

JDL-Studio

<Select template>

Please sign in for more features!

```
1
2 entity Region {
3   regionName String
4 }
5
6 entity Country {
7   countryName String
8 }
9
10 // an ignored comment
11 /** not an ignored comment */
12 entity Location {
13   streetAddress String,
14   postalCode String,
15   city String,
16   stateProvince String
17 }
18
19 entity Department {
20   departmentName String required
21 }
22
23 /**
24  * Task entity.
25  * @author The JHipster team.
```

UML Class Diagram:

- JobHistory**: startDate : Instant, endDate : Instant, language : Language
- Department**: departmentName : String\*
- Location**: streetAddress : String, postalCode : String, city : String, stateProvince : String
- Country**: countryName : String
- Region**: regionName : String
- Employee**: firstName : String, lastName : String, email : String, phoneNumber : String, hireDate : Instant, salary : Long, commissionPct : Long
- Job**: jobTitle : String, minSalary : Long, maxSalary : Long
- Task**: title : String, description : String

Language Enum:

- FRENCH
- ENGLISH
- SPANISH

# Step 1/ Compile backend

## \$ mvn package

```
$ mvn package
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.mycompany.myapp:jhipster-sample-application >-----
[INFO] Building Jhipster Sample Application 0.0.1-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO] The original artifact has been renamed to C:\web\demo-jhipster\jhipsterSample.jar.original
[INFO]
[INFO] --- modernizer-maven-plugin:2.7.0:modernizer (modernizer) @ jhipster-sample-application ---
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 6.574 s
[INFO] Finished at: 2023-12-02T18:08:06+01:00
[INFO] -----
```

# Step 2/ Run backend

## \$ mvn spring-boot:run -Pdev

```
$ mvn spring-boot:run -Pdev
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.mycompany.myapp:jhipster-sample-application >-----
[INFO] Building Jhipster Sample Application 0.0.1-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] >>> spring-boot-maven-plugin:3.1.5:run (default-cli) > test-compile @ jhipster-sample-application >>>

2023-12-02T18:13:20.679+01:00 INFO 13436 --- [ restartedMain] c.m.myapp.JhipsterSampleApplicationApp : Started
n 6.889 seconds (process running for 7.296)
2023-12-02T18:13:20.684+01:00 INFO 13436 --- [ restartedMain] c.m.myapp.JhipsterSampleApplicationApp :
-----
Application 'jhipsterSampleApplication' is running! Access URLs:
Local:      http://localhost:8080/
External:   http://192.168.0.10:8080/
Profile(s): [dev, api-docs]
-----
```

## Step 3/ Test Backend

http://localhost:8080/management/health

```
$ curl http://localhost:8080/management/health/liveness  
{"status":"UP"}
```



localhost:8080/management/health/liveness

```
{"status":"UP"}
```

Step 4/ compile FrontEnd  
\$ npm install  
\$ npm start

```
$ npm install  
[redacted] \ idealTree:jhipsterSampleApplication: sill idealTree buildDeps
```

```
added 2162 packages, and audited 2163 packages in 4m
```

```
306 packages are looking for funding  
  run `npm fund` for details
```

```
8 moderate severity vulnerabilities
```

# Step 5/ Start the FrontEnd

## \$ npm start

```
$ npm start
```

```
> jhipster-sample-application@0.0.1-SNAPSHOT start  
> ng serve --hmr
```

```
** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **
```

```
✓ Compiled successfully.
```

```
Notifications are disabled
```

```
Reason: DisabledByGroupPolicy Please make sure that the app id is set correctly.
```

```
Command Line: C:\web\demo-jhipster\jhipsterSampleApplication\node_modules\node-notifier\vendor\snoreToast\snoretoast.exe -p C:\web\demo-jhipster\jhipsterSampleApplication\webpack\logo-jhipster.png -t "Jhipster Sample Application"
```

```
[Browsersync] Proxying: http://localhost:4200
```

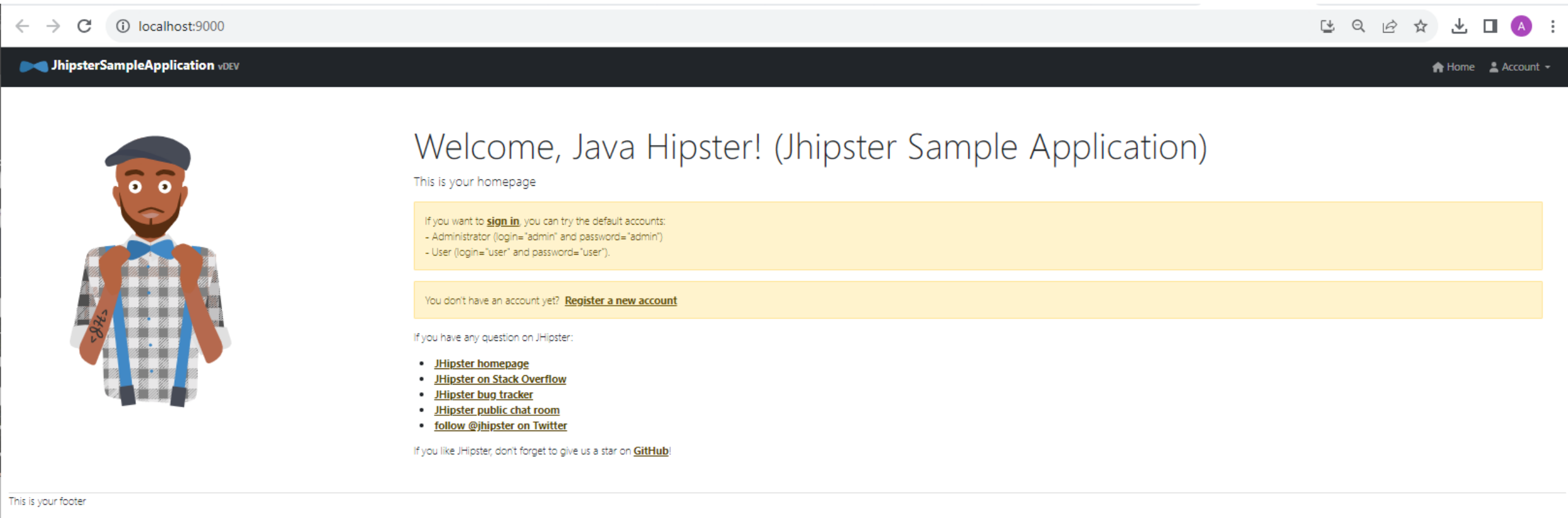
```
[Browsersync] Access URLs:
```

```
-----  
Local: http://localhost:9000  
External: http://192.168.0.10:9000  
-----
```

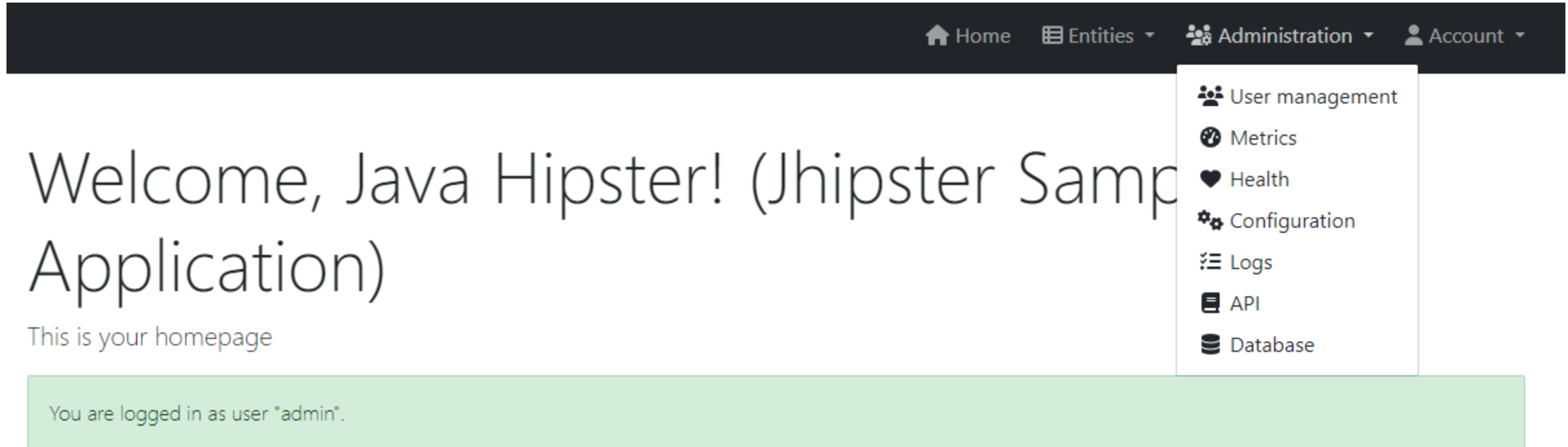
```
UI: http://localhost:3001  
UI External: http://localhost:3001  
-----
```

# Step 6/ open Browser

## http://localhost:4200



# Step 7/ login as "admin/admin" or "user/user"



The screenshot displays the Java Hipster! application interface. At the top, a dark navigation bar contains links for Home, Entities, Administration, and Account. The Administration dropdown menu is open, showing options like User management, Metrics, Health, Configuration, Logs, API, and Database. The main content area features a large welcome message and a confirmation of the current user.

Home Entities Administration Account

User management  
Metrics  
Health  
Configuration  
Logs  
API  
Database

## Welcome, Java Hipster! (Jhipster Sample Application)

This is your homepage

You are logged in as user "admin".



# Step 8/ import jdl ... recompile & restart \$ jhipster jdl \${jdlFile}

```
$ jhipster jdl ../jhipster-jdl.jdl  
WARNING! Since JHipster v8, the jhipster command will not use the locally installed generator-jhipster.  
If you want to execute the locally installed generator-jhipster, run: npx jhipster
```



<https://www.jhipster.tech>

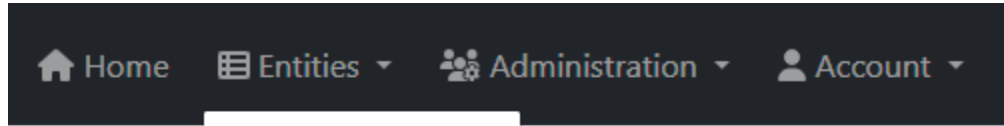
Welcome to JHipster v8.0.0

```
INFO! Generating jdl ../jhipster-jdl.jdl
```

```
INFO! Generating entities Region Country Location Department Task Employee Job JobHistory
```

```
create src\test\java\com\mycompany\myapp\web\rest\LocationResourceIT.java  
create src\test\java\com\mycompany\myapp\domain\EmployeeTestSamples.java  
create src\test\java\com\mycompany\myapp\domain\JobTest.java  
create src\main\webapp\app\entities\enumerations\language.model.ts  
create src\main\webapp\app\entities\region\service\region.service.ts  
create src\main\webapp\app\entities\country\update\country-update.component.spec.ts  
create src\main\webapp\app\entities\region\region.model.ts  
create src\main\webapp\app\entities\country\detail\country-detail.component.ts  
create src\main\webapp\app\entities\region\service\region.service.spec.ts
```

# Step 9/ navigate Entities Table - View - EditForm



- \* Region
- \* Country
- \* Location
- \* **Department**
- \* Task
- \* Employee
- \* Job
- \* Job History



## Departments

[Refresh list](#) [+ Create a new Department](#)

ID ^	Department Name ↕	Location ↕	
1	boastfully fumbling supposing		<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
2	ponder finally unfortunately		<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
3	rosy bubbly inasmuch		<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
4	plus arid		<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
5	sour		<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
6	ick boohoo obvious		<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
			<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
			<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
			<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>

## Create or edit a Department

ID

1

Department Name

dept 1


Location

1

[Cancel](#)

[Save](#)

# Outline

- Some Advanced features
  - AG-Grid
  - HighCharts
  - Code Scaffolding
  -  • Generating (Angular) Client from Swagger
  - Security: Cookie, Login, OAuth2, JWT, ...
  - RxJs, Auto completion, WebSocket
- Final Words : Angular - React - Vue.Js - Solid.Js

# Reminder ... swagger-ui.html /api-docs

would you like to use Maven or Gradle for building the backend?

Maven

Which other technologies would you like to use?



API first development using OpenAPI-generator



Search engine using Elasticsearch



WebSockets using Spring Websocket



Asynchronous messages using Apache Kafka

← → ↻ openapi-generator.tech



OpenAPI Generator

Getting Started

Generators

Roadmap

FAQ

Team

Blog

API

## OpenAPI Generator

Generate clients, servers, and documentation from OpenAPI 2.0/3.x documents

Try It Out

Install

Generators

Customization

Integrations

# Generators

← → ↻ 🔒 openapi-generator.tech/docs/generators



OpenAPI Generator

Getting Started

Generators

Roadmap

FAQ

Team

Blog

API

Getting Started



Extending



Contributing



About



Releases



API



Generators List

🏠 > API > Generators List

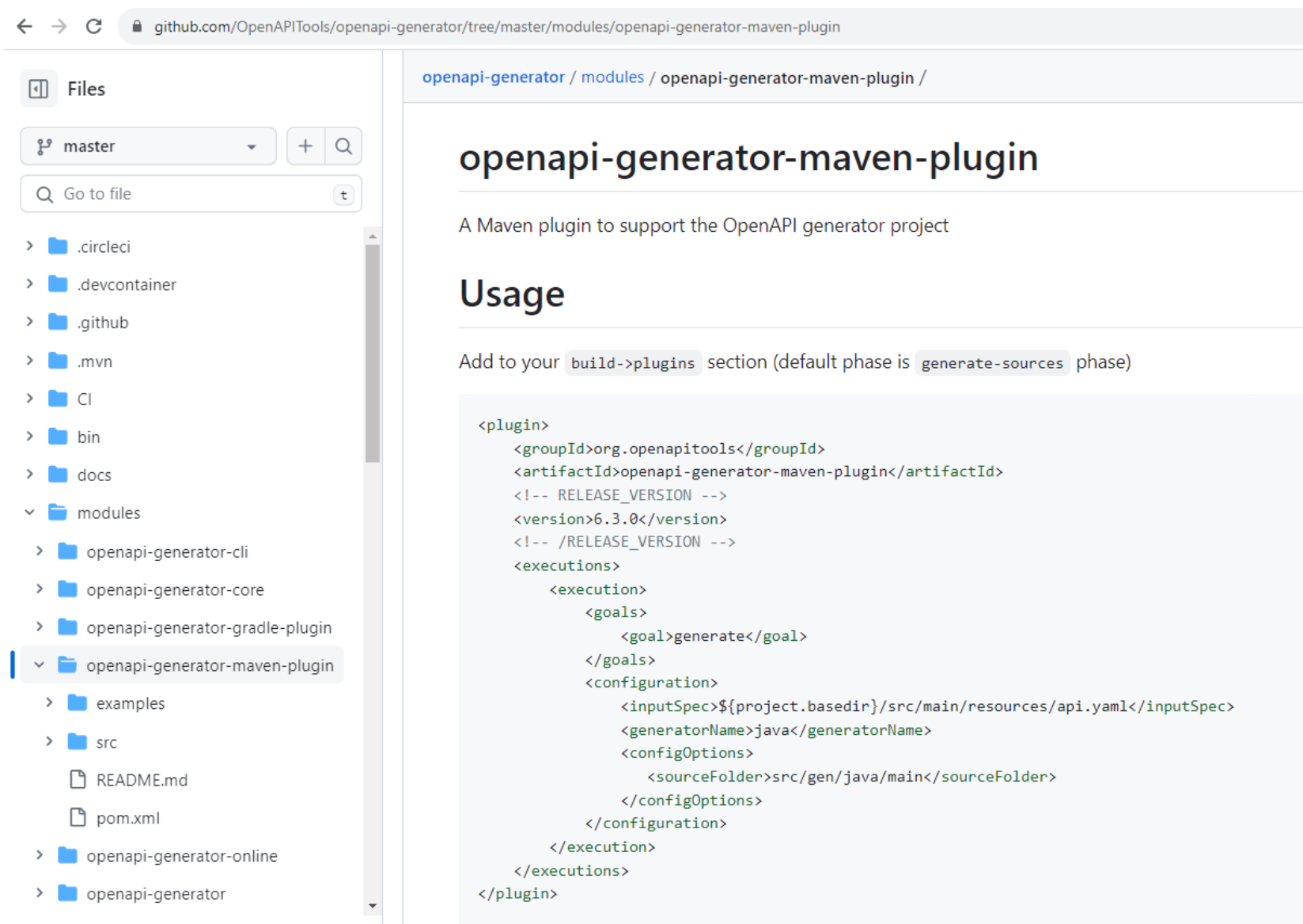
## Generators List

The following generators are available:

### CLIENT generators

- [ada](#)
- [android](#)
- [apex](#)
- [bash](#)
- [c](#)
- [scala](#)
- [swift-combine](#)
- [swift5](#)
- [typescript \(experimental\)](#)
- [typescript-angular](#)
- [typescript-aurelia](#)
- [typescript-axios](#)
- [typescript-fetch](#)

# ... other generators (maven)



The screenshot shows the GitHub repository page for the `openapi-generator-maven-plugin`. The left sidebar displays the file tree with the `modules/openapi-generator-maven-plugin` directory selected. The main content area shows the plugin's name, a brief description, and a usage example in a Maven `pom.xml` file.

github.com/OpenAPITools/openapi-generator/tree/master/modules/openapi-generator-maven-plugin

Files

master

Go to file

- > .circleci
- > .devcontainer
- > .github
- > .mvn
- > CI
- > bin
- > docs
- ▼ modules
  - > openapi-generator-cli
  - > openapi-generator-core
  - > openapi-generator-gradle-plugin
  - ▼ openapi-generator-maven-plugin
    - > examples
    - > src
    - README.md
    - pom.xml
  - > openapi-generator-online
  - > openapi-generator

openapi-generator / modules / openapi-generator-maven-plugin /

## openapi-generator-maven-plugin


A Maven plugin to support the OpenAPI generator project

## Usage

Add to your `build->plugins` section (default phase is `generate-sources` phase)

```
<plugin>
  <groupId>org.openapitools</groupId>
  <artifactId>openapi-generator-maven-plugin</artifactId>
  <!-- RELEASE_VERSION -->
  <version>6.3.0</version>
  <!-- /RELEASE_VERSION -->
  <executions>
    <execution>
      <goals>
        <goal>generate</goal>
      </goals>
      <configuration>
        <inputSpec>${project.basedir}/src/main/resources/api.yaml</inputSpec>
        <generatorName>java</generatorName>
        <configOptions>
          <sourceFolder>src/gen/java/main</sourceFolder>
        </configOptions>
      </configuration>
    </execution>
  </executions>
</plugin>
```

# Outline

- Some Advanced features
  - AG-Grid
  - HighCharts
  - Code Scaffolding
  - Generating (Angular) Client from Swagger
  -  • Security: Cookie, Login, OAuth2, JWT, ...
  - RxJs, Auto completion, WebSocket
- Final Words : Angular - React - Vue.Js - Solid.Js

Security ... should not be done half way

Leave it to Experts



# Security Basics

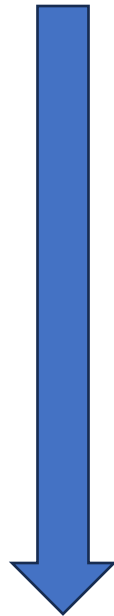
## Encryption

Confidentiality  
No Man-In-The-Middle



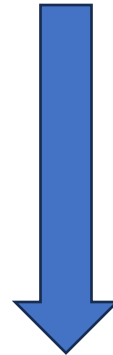
**httpS://**  
**certificate Trust**

## Authentication



**user**  
**password**

Delegated  
Authentication



**OAuth2**  
**flows**

Authenticated  
Session



**JWT**  
**token**  
**(Cookie)**

## Authorization



**relation in code**  
**Resource**

**-> Permission**

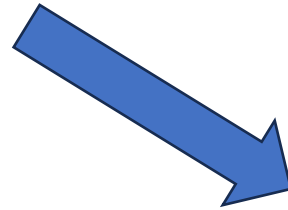
**-> Group -> Users**

# Http Headers

"Authorization=Basic <user:pass>"

"Authorization=Bearer <token>"

```
$ curl -v -u user:pass http://localhost:8080
```



password in Base64 (~clear text) on Http Header

# Demo "Basic <Base64(user:password)>"

```
$ curl -v -u user:user http://localhost:8080/  
* Trying [::1]:8080...  
* Connected to localhost (:::1) port 8080  
* Server auth using Basic with user 'user'  
> GET / HTTP/1.1  
> Host: localhost:8080  
> Authorization: Basic dXNlcjp1c2Vy  
> User-Agent: curl/8.4.0  
> Accept: */*  
>  
< HTTP/1.1 200 OK
```

```
$ echo dXNlcjp1c2Vy | base64 -d  
user:user
```

# Demo: Un-authenticated

```
$ curl -v http://localhost:8080/management/liveness
* Trying [::1]:8080...
* Connected to localhost (::1) port 8080
> GET /management/liveness HTTP/1.1
> Host: localhost:8080
> User-Agent: curl/8.4.0
> Accept: */*
>
< HTTP/1.1 401 Unauthorized
```


# Demo ... 401

( no password, bad password, bad user, or unauthorized )

```
$ curl -v -u user:pass http://localhost:8080/management/liveness
* Trying [::1]:8080...
* Connected to localhost (::1) port 8080
* Server auth using Basic with user 'user'
> GET /management/liveness HTTP/1.1
> Host: localhost:8080
> Authorization: Basic dXNlcjpwYXNz
> User-Agent: curl/8.4.0
> Accept: */*
>
< HTTP/1.1 401 Unauthorized
```

Demo of Login in Jhipster  
( if time ok ? )

# Outline

- Some Advanced features
  - AG-Grid
  - HighCharts
  - Code Scaffolding
  - Generating (Angular) Client from Swagger
  - Security: Cookie, Login, OAuth2, JWT, ...
  -  • RxJs, Auto completion, WebSocket
- Final Words : Angular - React - Vue.Js - Solid.Js

# RxJs : Asynchronous programming with Observable & Subscribe, is ~~Complex~~ Powerful

The screenshot shows the official RxJS website. The browser address bar displays 'rxjs.dev/guide/overview'. The website has a dark blue header with the RxJS logo and navigation links: OVERVIEW, REFERENCE, and TEAM. A search bar is located on the right side of the header. The left sidebar contains a list of navigation items: OVERVIEW (selected), Observables, Observer, Operators, Subscription, Subjects, Scheduler, Marble Testing, INSTALLATION, IMPORTING, REFERENCE, GLOSSARY, OPERATOR DECISION TREE, DEPRECATIONS & BREAKING CHANGES, DETAILED CHANGE LIST, and CODE OF CONDUCT. The main content area is titled 'Introduction' and contains the following text:

RxJS is a library for composing asynchronous and event-based programs by using observable sequences. It provides one core type, the **Observable**, satellite types (Observer, Schedulers, Subjects) and operators inspired by Array methods (`map`, `filter`, `reduce`, `every`, etc) to allow handling asynchronous events as collections.

*Think of RxJS as Lodash for events.*

ReactiveX combines the **Observer pattern** with the **Iterator pattern** and **functional programming with collections** to fill the need for an ideal way of managing sequences of events.

The essential concepts in RxJS which solve async event management are:

- **Observable**: represents the idea of an invokable collection of future values or events.
- **Observer**: is a collection of callbacks that knows how to listen to values delivered by the Observable.
- **Subscription**: represents the execution of an Observable, is primarily useful for cancelling the execution.
- **Operators**: are pure functions that enable a functional programming style of dealing with collections with operations like `map`, `filter`, `concat`, `reduce`, etc.
- **Subject**: is equivalent to an EventEmitter, and the only way of multicasting a value or event to multiple Observers.
- **Schedulers**: are centralized dispatchers to control concurrency, allowing us to coordinate when computation happens on e.g. `setTimeout` or `requestAnimationFrame` or others.

The right sidebar contains a list of navigation items: Introduction (selected), First examples, Purity, Flow, and Values.



# Example RxJs operator: debounceTime(millis)

← → ↺ 🔒 rxjs.dev/api/index/function/debounceTime

☰ RxJS OVERVIEW REFERENCE TEAM

OVERVIEW  
Observables  
Observer  
Operators  
Subscription  
Subjects  
Scheduler  
Marble Testing  
INSTALLATION  
IMPORTING  
REFERENCE  
GLOSSARY  
OPERATOR DECISION TREE  
DEPRECATIONS & BREAKING CHANGES  
DETAILED CHANGE LIST  
CODE OF CONDUCT

stable (v7.8.1)

API / rxjs/index

## debounceTime

FUNCTION STABLE OPERATOR

Emits a notification from the source Observable only after a particular time span has passed without another source emission.

`debounceTime<T>(dueTime: number, scheduler: SchedulerLike = asyncScheduler): MonoTypeOperatorFunction<T>`

Parameters

<code>dueTime</code>	<code>number</code>	The timeout duration in milliseconds (or the time unit determined internally by the optional scheduler) for the window of time required to wait for emission silence before emitting the most recent source value.
<code>scheduler</code>	<code>SchedulerLike</code>	Optional. Default is <code>asyncScheduler</code> . The <code>SchedulerLike</code> to use for managing the timers that handle the timeout for each value.

Returns

`MonoTypeOperatorFunction<T>`: A function that returns an Observable that delays the emissions of the source Observable by the specified `dueTime`, and may drop some values if they occur too frequently.

Description

It's like `delay`, but passes only the most recent notification from each burst of emissions.

debounceTime delays notifications emitted by the source Observable, but drops previous pending delayed emissions if a new notification arrives on the source Observable. This operator is useful for the debounceTime(T) use case. On the other hand, it does not drop emissions that occur after the T duration has passed since the previous emission. On the other hand, it does not drop emissions that occur after the T duration has passed since the previous emission.

example debounceTime()

Doing Auto-completion -> trigger http Call(s)  
but NOT for each char typed

... cf Also `.distinctUntilChanged()`  
and `switchMap()` !!

[https://stackblitz.com/edit/  
autocomplete-suggestions-seach-with-debouncing  
?file=src%2Fapp%2Fapp.component.ts](https://stackblitz.com/edit/autocomplete-suggestions-seach-with-debouncing?file=src%2Fapp%2Fapp.component.ts)

```
this.bookForm.controls["searchtext"].valueChanges
  .pipe(
    debounceTime(1000),
    distinctUntilChanged((curr, prev) => {
      return curr.toLowerCase() === prev.toLowerCase();
    }),
    switchMap(text => {
      this.searchstring.push(text);
      if (
        this.searchstring[this.searchstring.length - 1] !==
        this.searchstring[this.searchstring.length - 2]
      ) {
        return this.bookservice.getBookBySearchText(text);
      } else {
        return [];
      }
    })
  )
  .subscribe(res => {
    this.books = res;
  });
```

# Outline

- Some Advanced features
  - AG-Grid
  - HighCharts
  - Code Scaffolding
  - Generating (Angular) Client from Swagger
  - Security: Cookie, Login, OAuth2, JWT, ...
  - RxJs, Auto completion, WebSocket

 Final Words : Angular - React - Vue.Js - Solid.Js

# Final Words

## Thank you Tim

Sir

**Tim Berners-Lee**

OM KBE FRS FREng FRSA DFBCS RDI



Berners-Lee in 2014

<b>Born</b>	Timothy John Berners-Lee 8 June 1955 (age 68) <a href="#">London</a> , England
<b>Other names</b>	TimBL TBL
<b>Education</b>	<a href="#">The Queen's College, Oxford</a> (BA)
<b>Known for</b>	Invention of the <a href="#">World Wide Web</a>

# Render to Cesar what belongs to Cesar

**Brendan Eich** ([/ˈaɪk/](#); born July 4, 1961)<sup>[1]</sup> is an American computer programmer and technology executive. He created the [JavaScript programming language](#) and co-founded the [Mozilla](#) project, the [Mozilla](#)

**Brendan Eich**



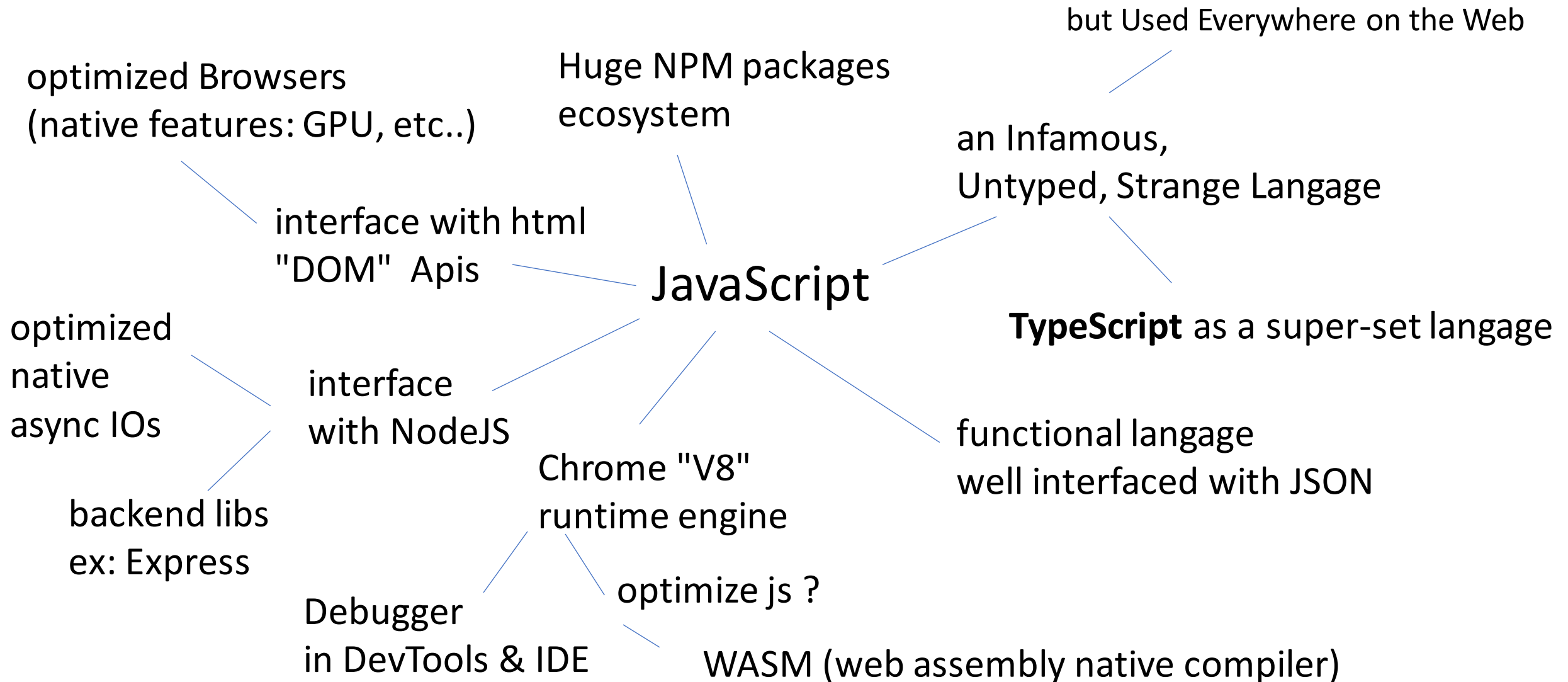
Eich in 2012

<b>Born</b>	4 July 1961 (age 62) <a href="#">Pittsburgh, Pennsylvania</a> , US
<b>Alma mater</b>	<a href="#">University of Illinois Urbana-Champaign</a> <a href="#">Santa Clara University</a>
<b>Website</b>	<a href="#">brendaneich.com</a> <a href="#">↗</a>

# Web Fundamental



# JavaScript MindMap





# TypeScript - JavaScript Schizofrenia Troller



# Backend - Frontend Schizofrenia Troller

**Back-end Developer**

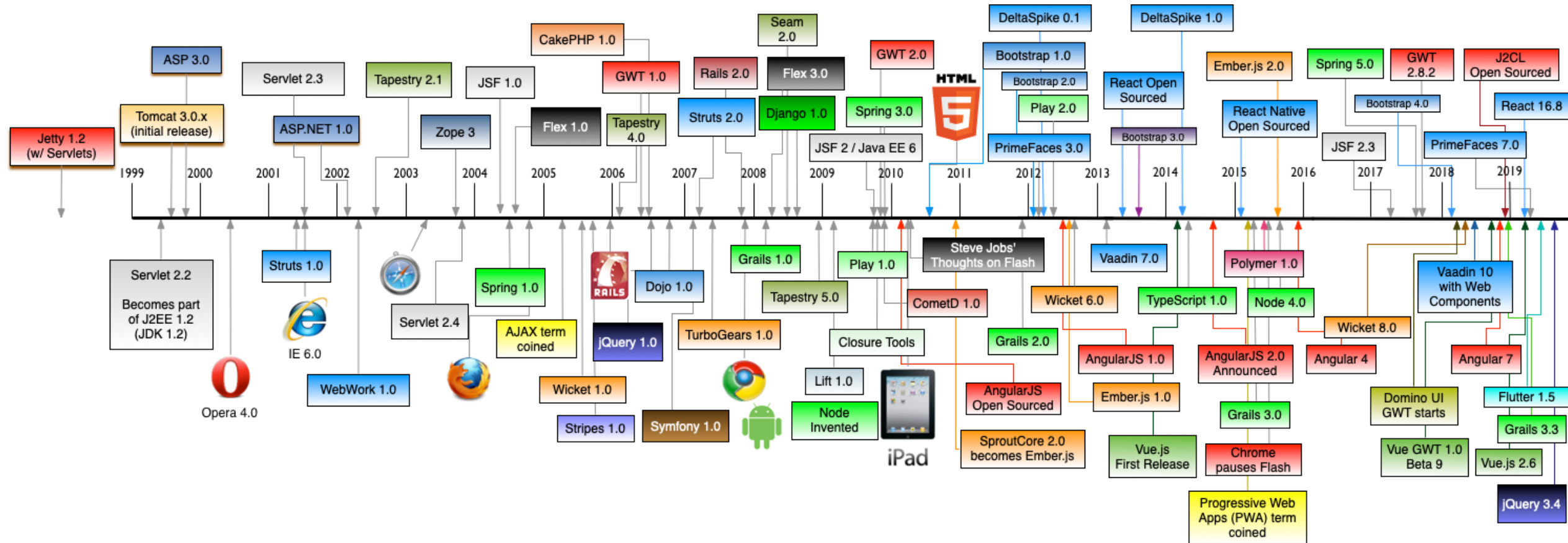


**Front-end Developer**



**Full-Stack Developer**

# Web Frameworks ... 1999 to 2020



# Web Frameworks - 2023 and Future ?



# Conclusion

Frameworks change every year

The Web will live long : Html + Javascript + Css

# Questions

[arnaud.nauwynck@gmail.com](mailto:arnaud.nauwynck@gmail.com)