



Get Hip with JHipster!

Matt Raible • <http://raibledesigns.com>



Photos by  McGinity Photo

 **RAIBLE DESIGNS**
ENTERPRISE OPEN SOURCE CONSULTING



Who is **Matt Raible**?

Father, Skier, Mountain
Biker, Whitewater Rafter

Web Framework Connoisseur

Founder of AppFuse



Bus Lover

Blogger on raibledesigns.com

What about **YOU**?

Are you a Java Developer?

How long have you been doing web development?

Do you like JSF? Do you like GWT?

Do you like JavaScript?



Do you want to be a hip Java Developer?

Java 8

Parallel Collections

JSR 310 Date and Time API

Functional Interfaces with default method

Lambda Expressions (a.k.a. Closures)

```
// sort a list by lastName  
List<Person> persons = ...;  
persons.sort((p1, p2) ->  
    p1.getLastName().compareTo(p2.getLastName()));
```

Nashorn JavaScript Engine



Hipster

n. noun

1. One who is exceptionally aware of or interested in the latest trends and tastes.



Latest trends in JavaLand?

Annotations

Environments: dev, test, production

Microservices

Container-less Deployment

Monitoring





Spring Boot Demo

Spring Boot



Create stand-alone Spring applications

Embeds Tomcat, Jetty or Undertow directly

Provides opinionated 'starter' POMs to simplify your Maven configuration

Also supports/promotes using Gradle

Spring Boot



Automatically configures Spring whenever possible

Provides production-ready features such as metrics, health checks and externalized configuration

Absolutely no code generation and no requirement for XML configuration

Grails 3.0 is based on Spring Boot

SPRING INITIALIZR @ start.spring.io

```
1. bash
~75% [mraible:~] $ curl start.spring.io

  ____  _
 / ___|| | | |
| |___| | | |
 \___ \| | | |
  ___) | | | |
 |____|_|_|_|

:: Spring Initializr :: https://start.spring.io

This service generates quickstart projects that can be easily customized.
Possible customizations include a project's dependencies, Java version, and
build system or build structure. See below for further details.

The services uses a HAL based hypermedia format to expose a set of resources
to interact with. If you access this root resource requesting application/json
as media type the response will contain the following links:
+-----+-----+
| Rel          | Description          |
+-----+-----+
| gradle-build  | Generate a Gradle build file          |
| gradle-project| Generate a Gradle based project archive|
| maven-build   | Generate a Maven pom.xml              |
```


Latest trends in Web Development?

JavaScript MVC Frameworks

CSS 3 with Animations

Mobile First

Front-End Optimization

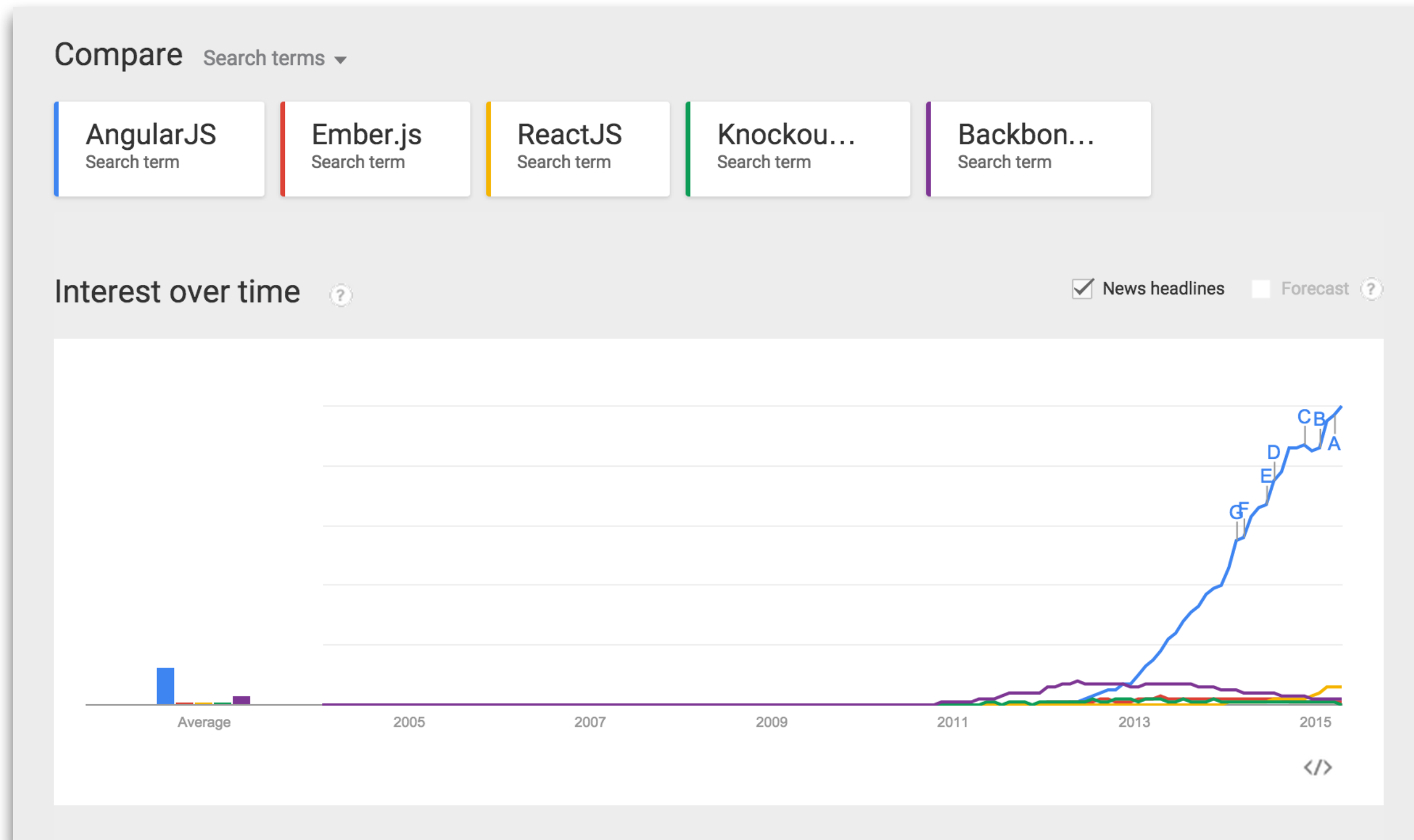
REST APIs



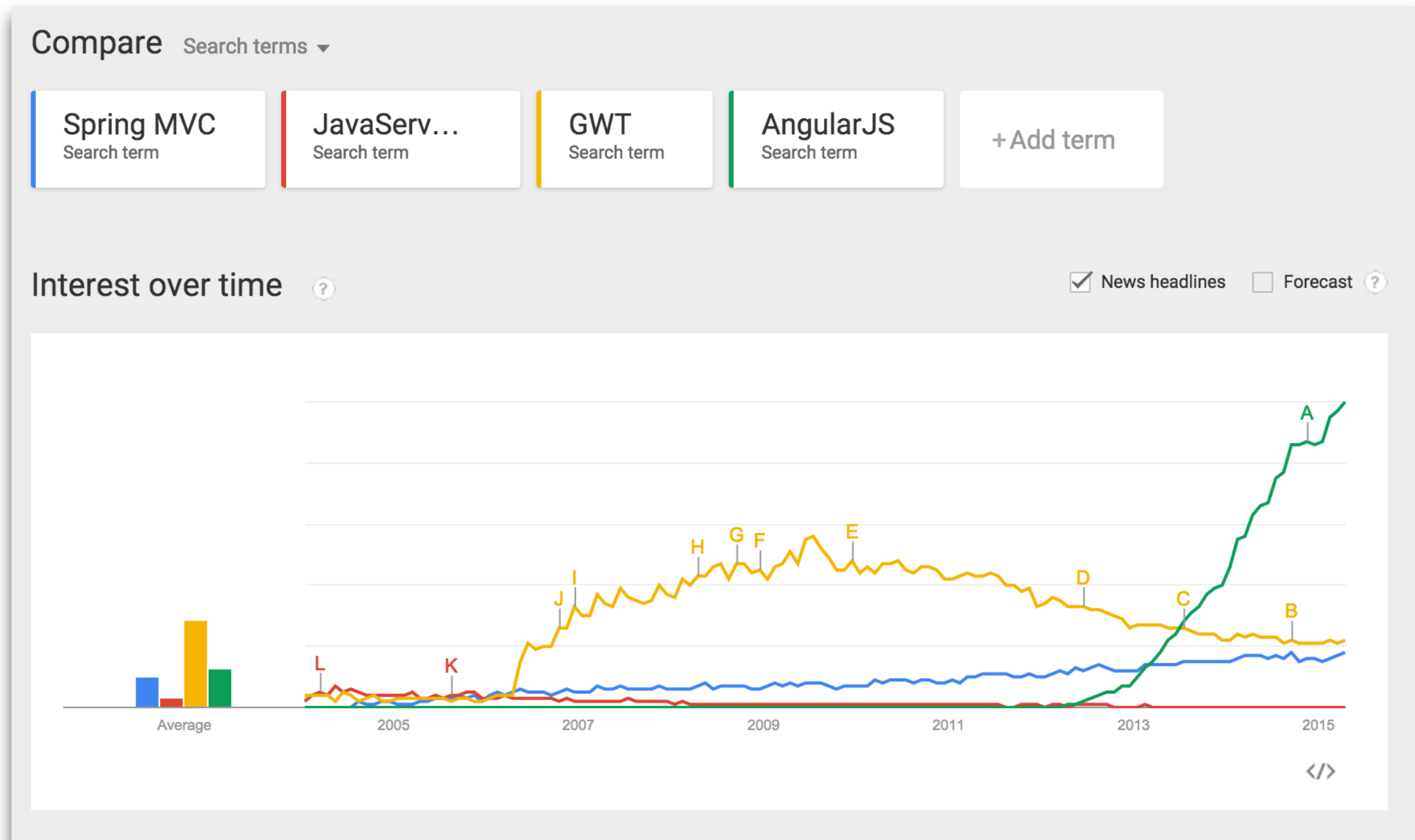


ANGULARJS
by Google

Google Trends



Google Trends



The most starred project on GitHub?

Search

Search

Repositories984,688

Code

Issues

Users

Languages

JavaScript191,462

Python104,410

Ruby94,569

Java87,919

PHP80,312

C48,210

C++40,402

Objective-C35,555

We've found 984,688 repository results

Sort: Most stars

twbs/bootstrap

CSS★79,716🔗31,143

The most popular HTML, CSS, and JavaScript framework for developing responsive, mobile first projects on the web.

Updated 10 hours ago

vhf/free-programming-books

★38,041🔗8,404

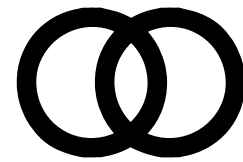
Updated 25 minutes ago

angular/angular.js

JavaScript★37,127🔗15,398

HTML enhanced for web apps

Wouldn't it be hip if...?



JHipster

<http://jhipster.github.io/>

Greetings, Java Hipster!



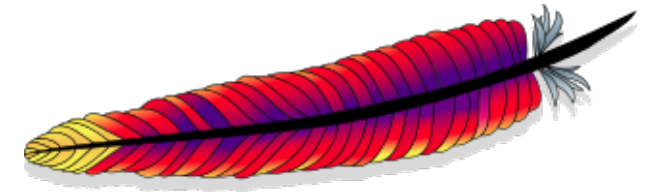
JHipster is a
Yeoman generator,



used to create a
Spring Boot + **AngularJS**
project.



JHipster Project



Web Statistics, March 2015

270K Page Views

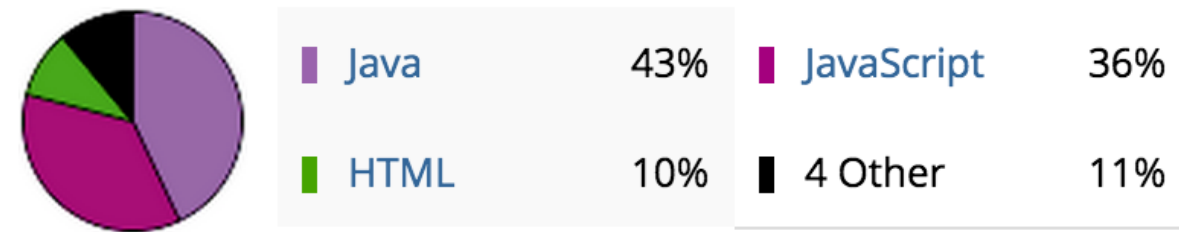
10K Downloads

GitHub Statistics

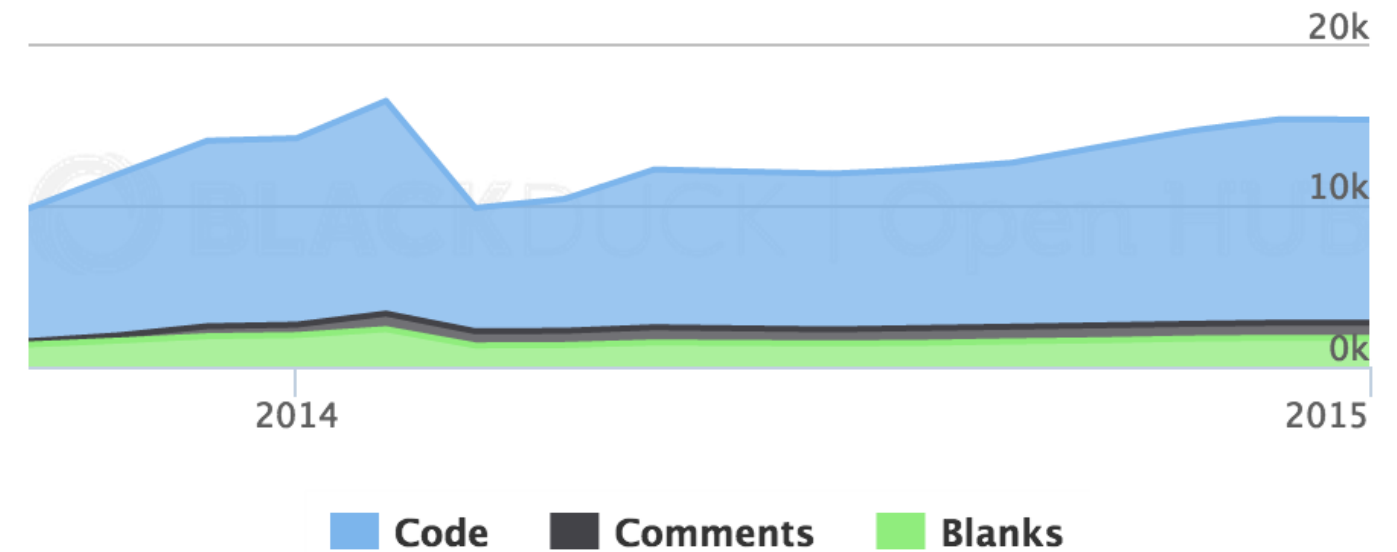
1871 Stars

106 Contributors

Languages



Lines of Code





JHipster

Foundational Frameworks

Spring Boot

Spring Security

AngularJS

Bootstrap

Bower

Metrics

Project Options

Java 7 or Java 8

Maven or Gradle

Authentication Type: cookie-based or OAuth2

Type of Database: SQL or NoSQL

Caching: EhCache or Hazelcast

Grunt or Gulp.js

Yeoman

<http://yeoman.io/>

The web's scaffolding tool for modern webapps

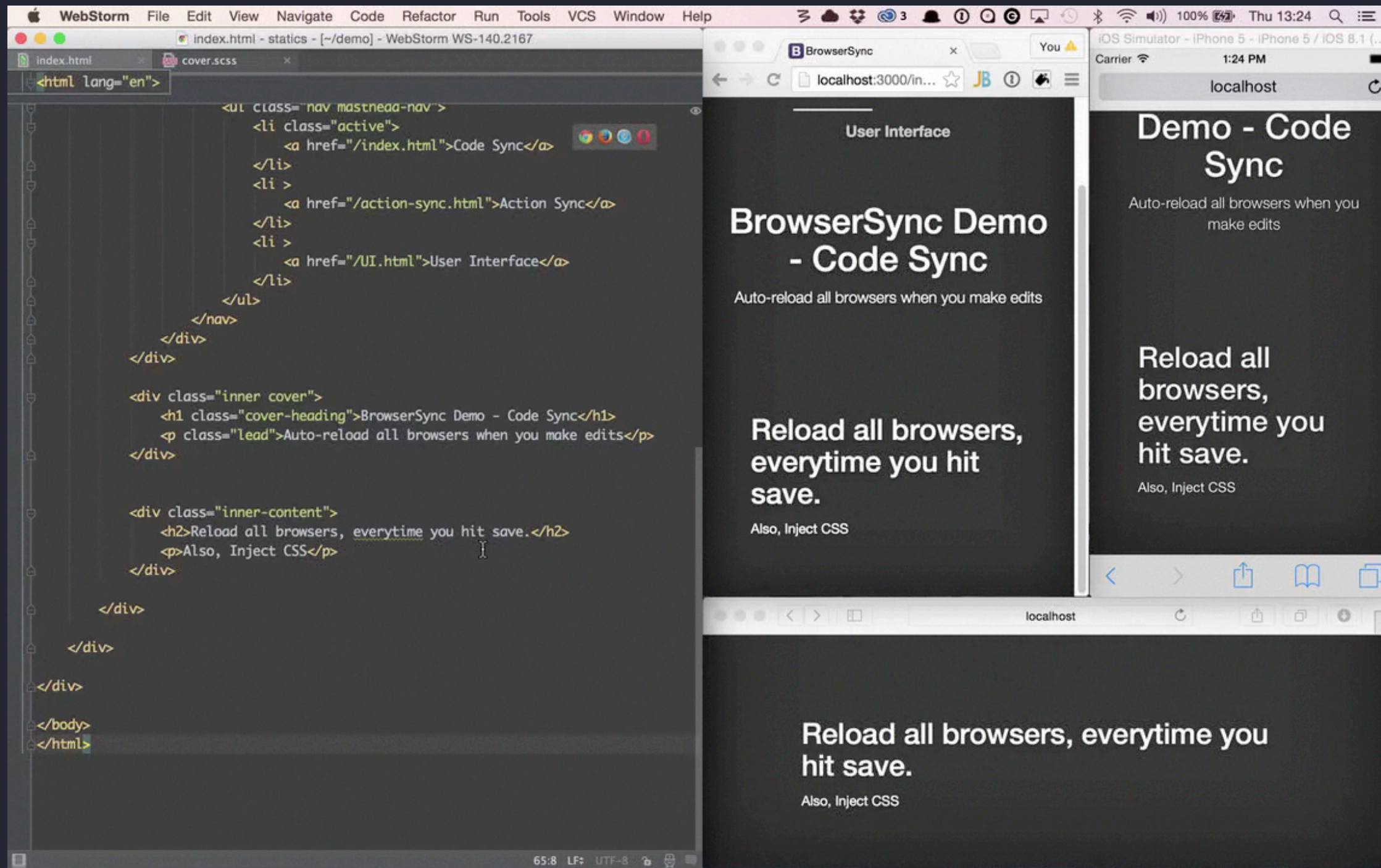
Helps you kickstart new projects

Promotes the Yeoman workflow

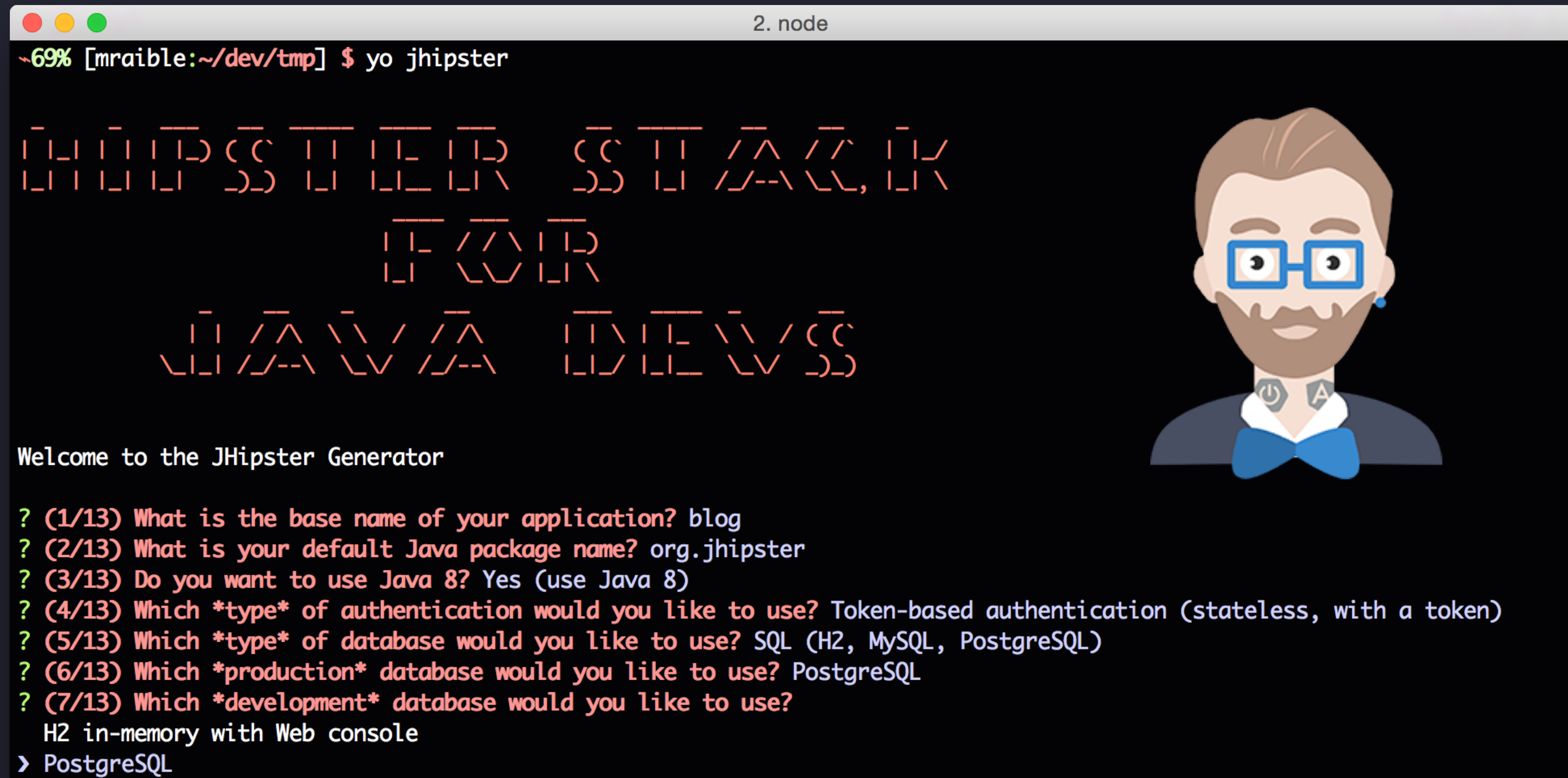


BrowserSync

<http://www.browsersync.io/>



Demo Time!



Demo Goals

Generate a basic blog application

Look at its configuration

Generate the CRUD entities

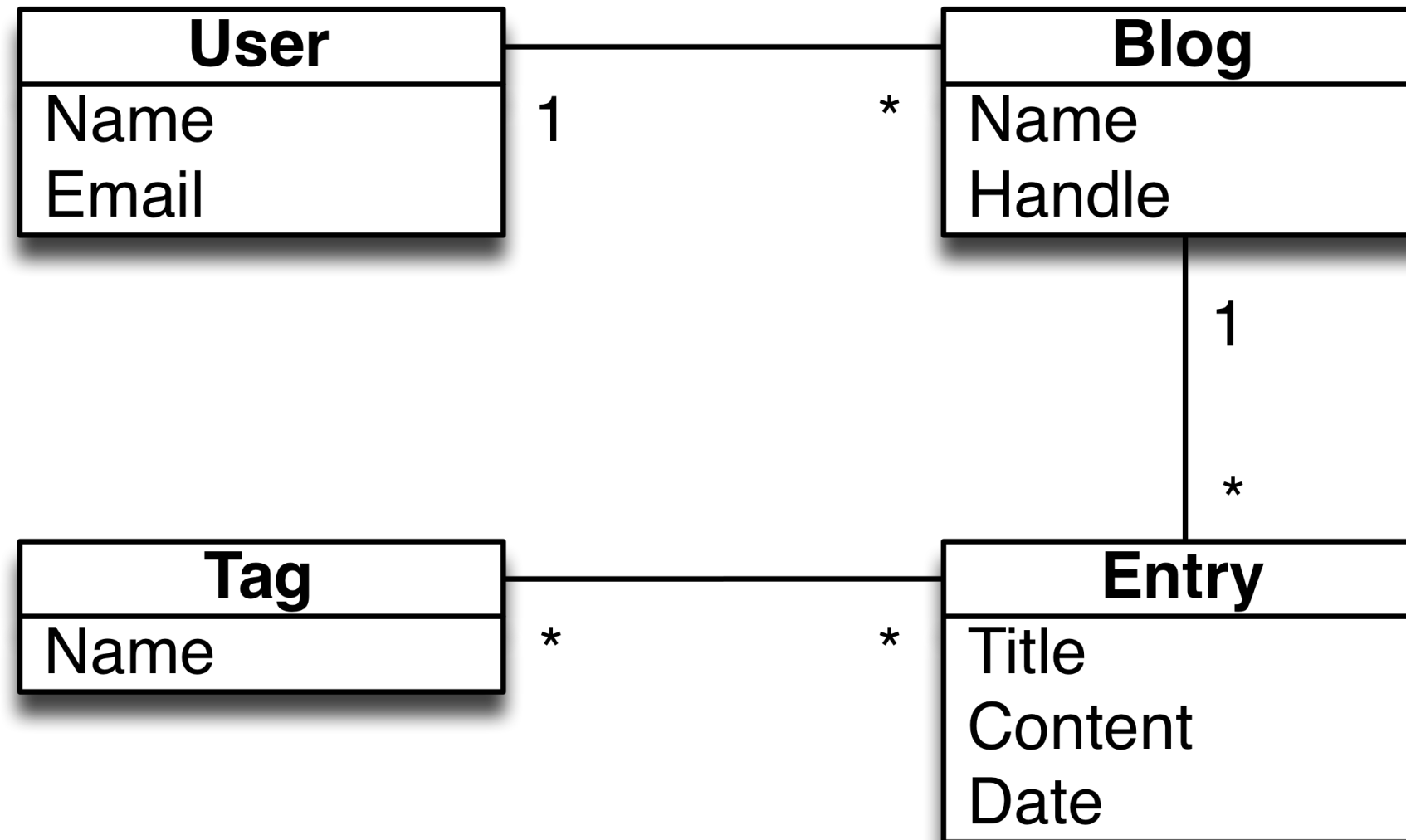
Limit blogs to current user, allow HTML

Deploy to  heroku



... all in 20 minutes!

Demo Application



Load Testing with Gatling



High Performance

Scala, Akka and Netty

Nice HTML-based Reporting

Scenario Recorder

```
val scn = scenario("Test the Blog entity")
  .exec(http("First unauthenticated request")
    .get("/api/account")
    .headers(headers_http)
    .check(status.is(401)))
  .pause(10)
  .exec(http("Authentication")
    .post("/api/authenticate")
    .headers(headers_http_authentication)
    .formParam("username", "admin")
    .formParam("password", "admin")
    .check(jsonPath("$.token").saveAs("x_auth_token")))
  .pause(1)
  .exec(http("Authenticated request")
    .get("/api/account")
    .headers(headers_http_authenticated)
    .check(status.is(200)))
  .pause(10)
  .repeat(2) {
    exec(http("Get all blogs")
      .get("/api/blogs")
      .headers(headers_http_authenticated)
      .check(status.is(200)))
    .pause(10 seconds, 20 seconds)
    .exec(http("Create new blog")
      .put("/api/blogs"))
```

What about UI Testing?



Jasmine

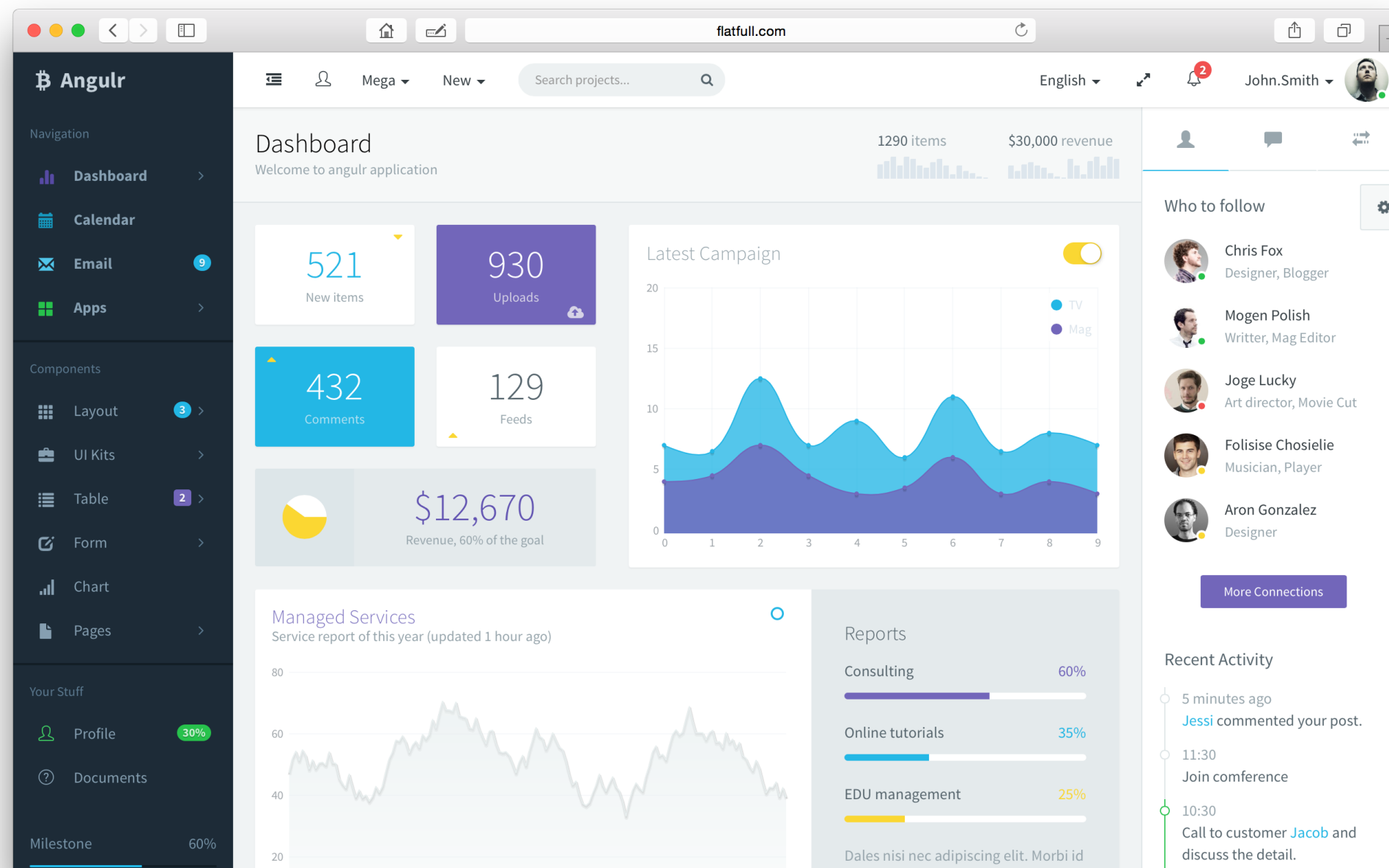


Protractor
end to end testing for AngularJS

<https://github.com/jhipster/generator-jhipster/issues/897>

How sweet is JHipster?!

AngularJS, Java 8, Microservices - what more could you want?



Angular 2.0

```
<input type="text" [value]="firstName">
```

```
<button (click)="addPerson()">Add</button>
```

```
<input type="checkbox" [checked]="someProperty">
```

Concepts Eliminated in Angular 2.0

Controllers

Directive Definition Object

\$scope

angular.module

jqLite



Angular 2: Template Syntax

A screenshot of a video player interface. The left pane shows TypeScript code for an Angular 2 component named 'Hello'. The right pane shows a browser preview of the component running on localhost:9000. The browser shows a form with a 'Name:' label, a text input, a 'Hello !' message, and a 'Click me' button. The video player has a play button in the center and a progress bar at the bottom.

```
@Component({
  selector: 'hello'
})
@Template({
  inline: `
    <div>
      <label>Name:</label>
      <input type="text" #ref>
      <hr>
      <h1>Hello {{ref.value}}!</h1>
    </div>

    <button id="button" (custom)="onCustom()" (click)="onClick(ref)">Click me
  `
})
export class Hello {
  constructor() {
    var button = document.querySelector("body /deep/ #button")
    setInterval(function() {
      button.dispatchEvent(new Event("custom"));
    }, 1000);
  }

  onClick(value) {
    console.log(value);
  }

  onCustom() {
    console.log("custom");
  }
}
```

<https://egghead.io/lessons/angularjs-angular-2-template-syntax>

The JHipster Mini-Book

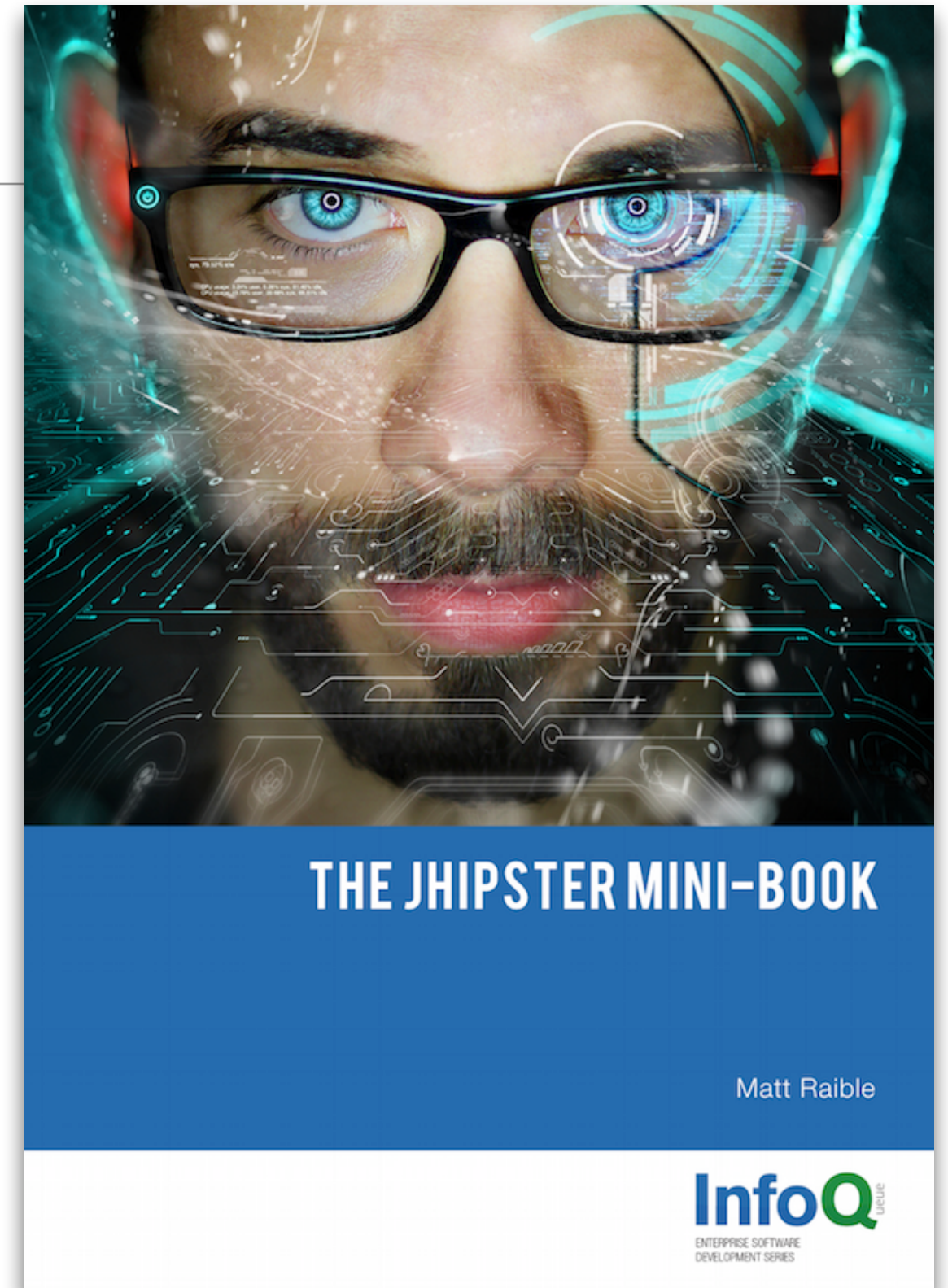
Written by yours truly

Quick and to the point

Less than 150 pages

Developing a Real World App

Free Download from **InfoQ**^{ueue}



Learn More

 https://twitter.com/java_hipster

 <http://jhipster.github.io/>

 <https://github.com/jhipster/generator-jhipster>

 <https://www.youtube.com/watch?v=R3jm2qmqctl>

 <https://angular.io/>

Questions?

Stay hip by following me!

 <http://raibledesigns.com>

 [@mraible](#)

Presentations

 <http://slideshare.net/mraible>

Code

 <http://github.com/mraible>



Devovx4Kids Denver

Teaching Kids to Program

Java, Minecraft, robots, oh my!

Non-profit, looking for speakers!



<http://www.meetup.com/Devovx4Kids-Denver/>