AngularJS - Introduction

@kevinderudder

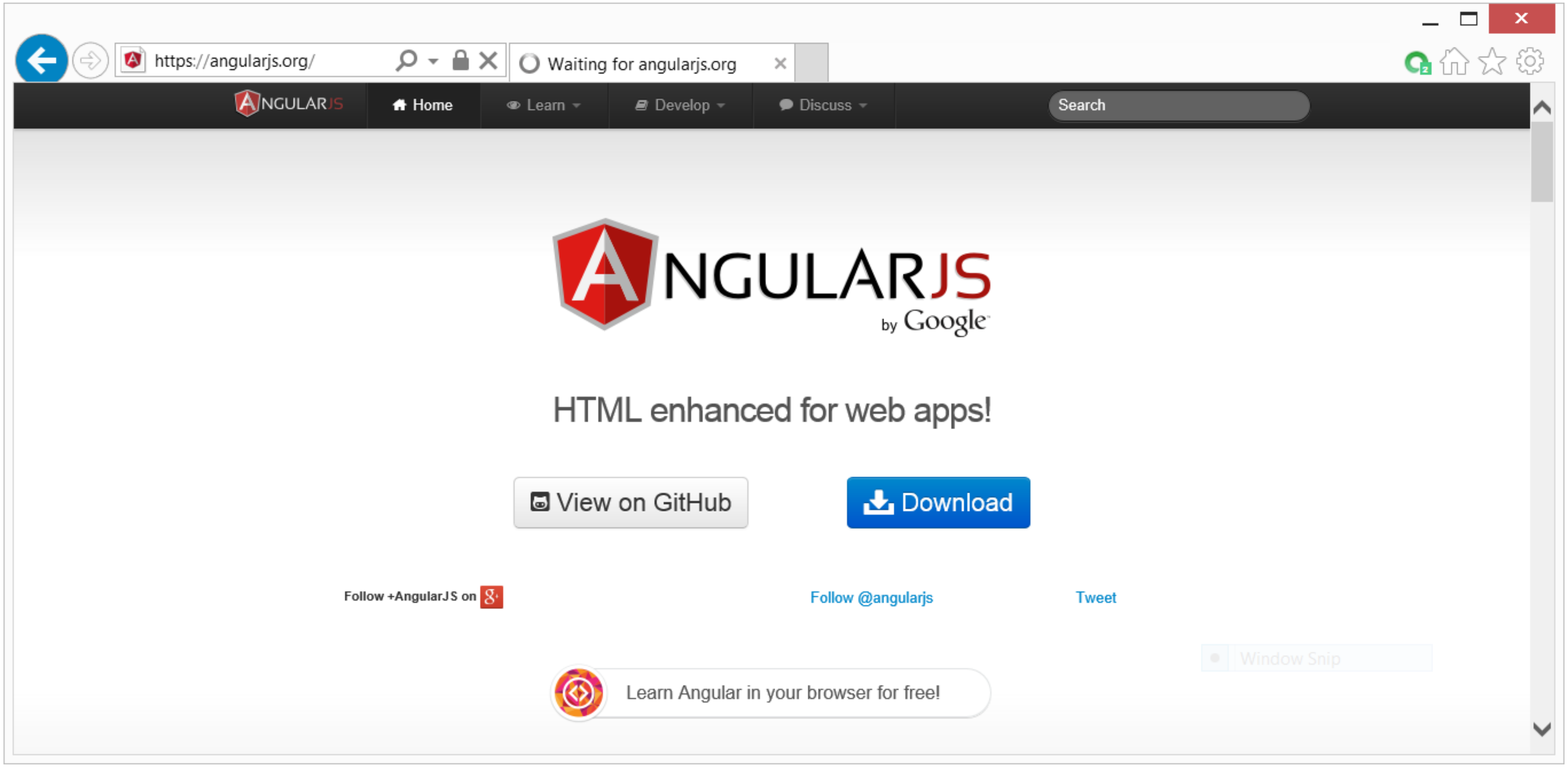
Agenda

- About Angular
- Angular vs Others
- Single Page Applications
- Architecture
- Expressions
- Directives
- Organizing your code

Google

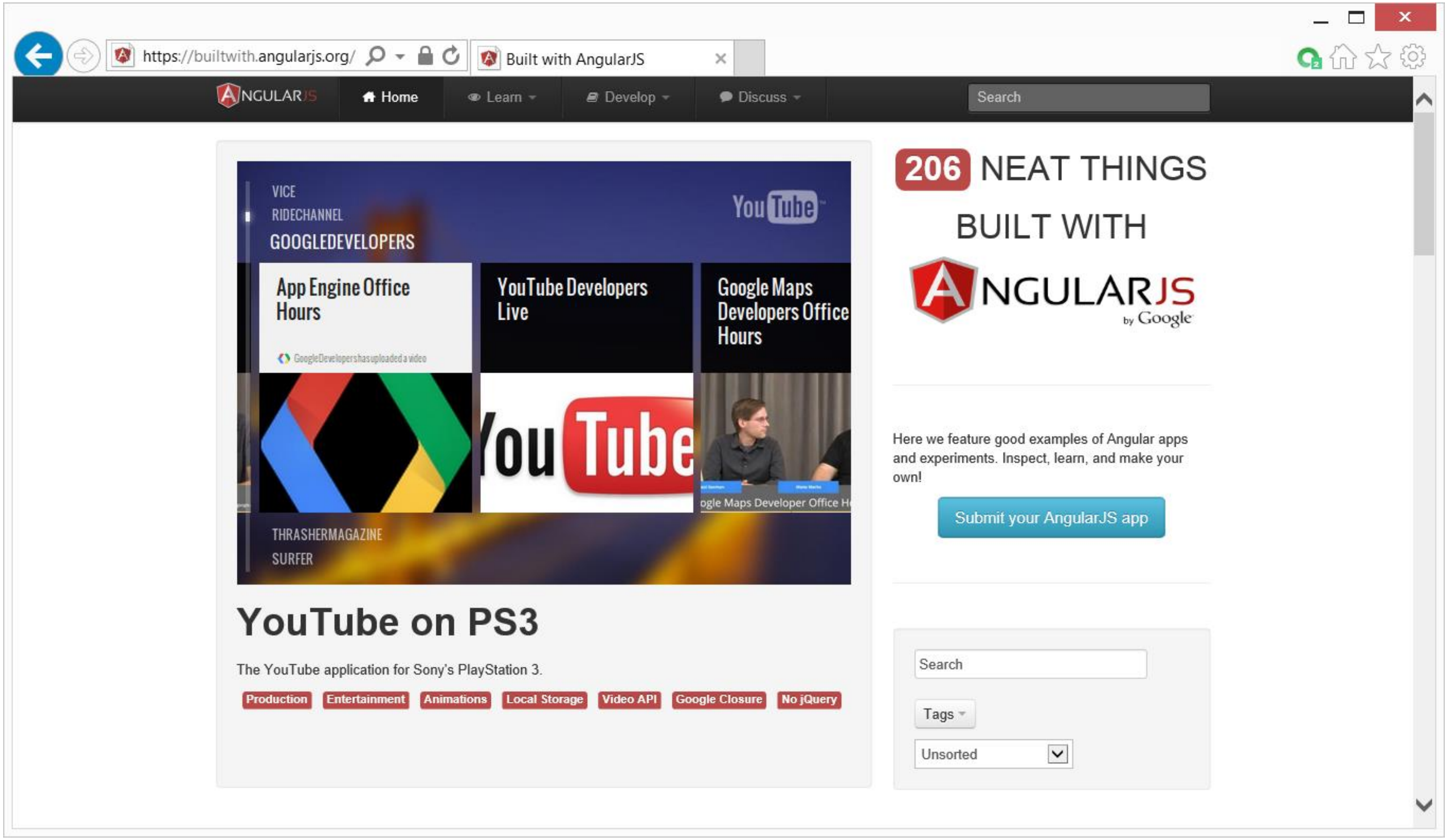
- Started as a project at Google
  - https://angularjs.org

- At this moment, it is an open source project
  - Anyone can contribute
  - https://github.com/angular/angular.js



HTML enhanced for web apps!

Learn Angular in your browser for free!



206 NEAT THINGS BUILT WITH ANGULARJS by Google

YouTube on PS3

The YouTube application for Sony's PlayStation 3.

Production Entertainment Animations Local Storage Video API Google Closure No jQuery

Here we feature good examples of Angular apps and experiments. Inspect, learn, and make your own!

AngularJS

- So, it’s a JavaScript Library
- But, it is kind off disrespectful to call it just a JavaScript Library
  - It is an MV* JavaScript Framework
  - It is an opinionated framework which means that they guide you to use in a specific way (eg: don’t touch the DOM)

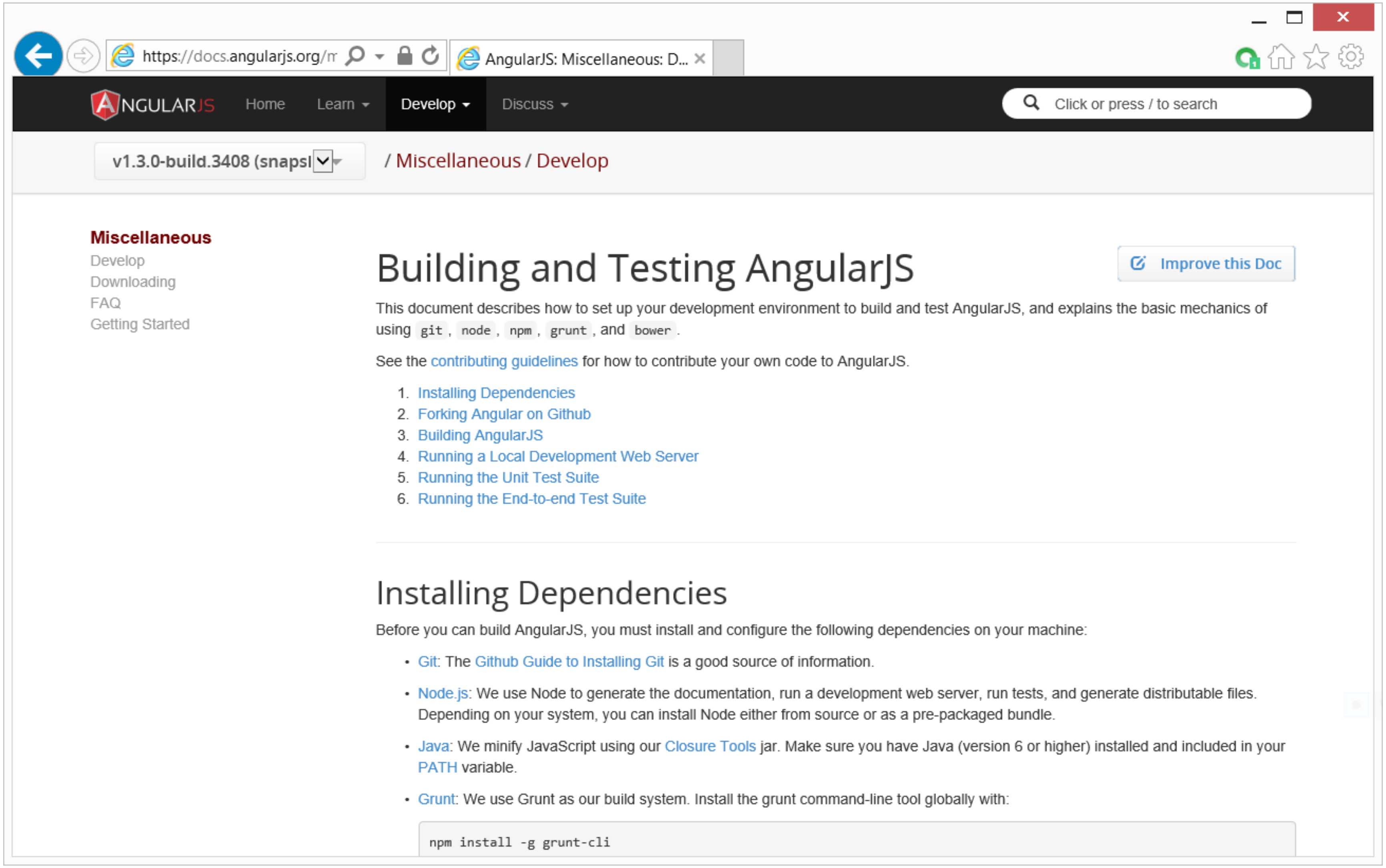
MV *

- Model View Controller
- Model View ViewModel
- Model View Presenter
- Model View Whatever
  - Whatever works for you?

Open Source

- Started by google
  - They know how to work with the WEB

- Benefits from the work of the community


https://docs.angularjs.org/m
AngularJS: Miscellaneous: D...
ANGULARJS
Home
Learn
Develop
Discuss
Click or press / to search
v1.3.0-build.3408 (snapsl
/ Miscellaneous / Develop
Miscellaneous
Develop
Downloading
FAQ
Getting Started
Improve this Doc

Building and Testing AngularJS

This document describes how to set up your development environment to build and test AngularJS, and explains the basic mechanics of using `git`, `node`, `npm`, `grunt`, and `bower`.

See the contributing guidelines for how to contribute your own code to AngularJS.

1. Installing Dependencies
2. Forking Angular on Github
3. Building AngularJS
4. Running a Local Development Web Server
5. Running the Unit Test Suite
6. Running the End-to-end Test Suite

Installing Dependencies

Before you can build AngularJS, you must install and configure the following dependencies on your machine:

- Git: The Github Guide to Installing Git is a good source of information.
- Node.js: We use Node to generate the documentation, run a development web server, run tests, and generate distributable files. Depending on your system, you can install Node either from source or as a pre-packaged bundle.
- Java: We minify JavaScript using our Closure Tools jar. Make sure you have Java (version 6 or higher) installed and included in your PATH variable.
- Grunt: We use Grunt as our build system. Install the grunt command-line tool globally with:

```
npm install -g grunt-cli
```

Benefits of using AngularJS

- Open Source
  - Already mentioned this

- Comprehensive

- Testable
  - Built with testing in mind

- Extendable
  - Add your own elements and properties
  - Called directives

ANGULARJS

by Google

has been designed to **build applications**
instead of documents

Angular Introduction

ANGULAR VS OTHERS

Angular vs Others

- Angular is independent from other JavaScript libraries
  - Twitter bootstrap is not
- Focusses on the MV* pattern

Angular vs jQuery

Angular is a **Framework** while jQuery is a **library**

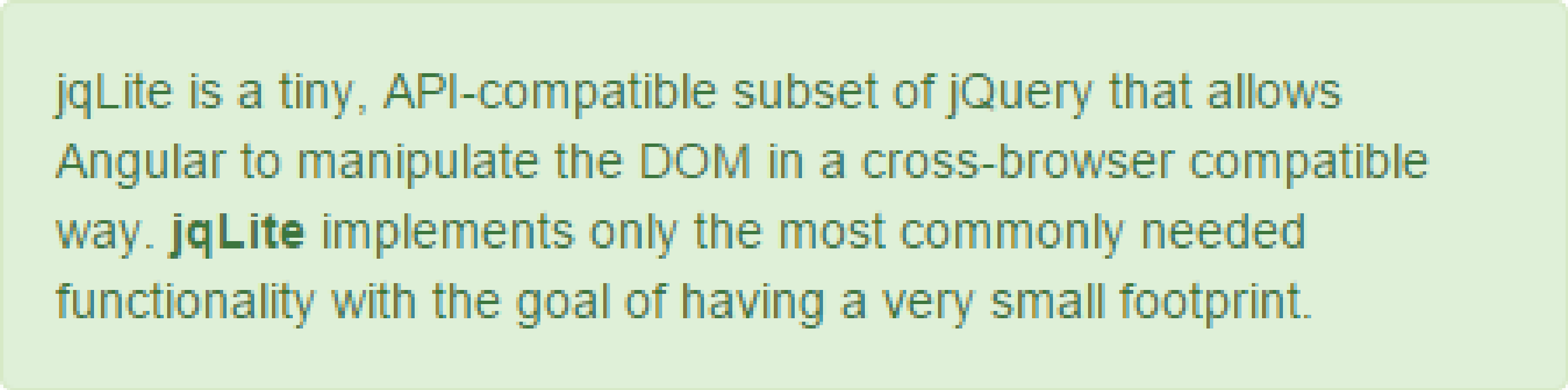
Angular vs jQuery

- Depends on what type of Websites your building
- The way you use jQuery is totally different from the way you build your applications with jQuery

Angular uses jQuery??

- Or you can use your version of jQuery

- If not
  - Angular falls back to its own implementation of the subset of jQuery
  - jQLite



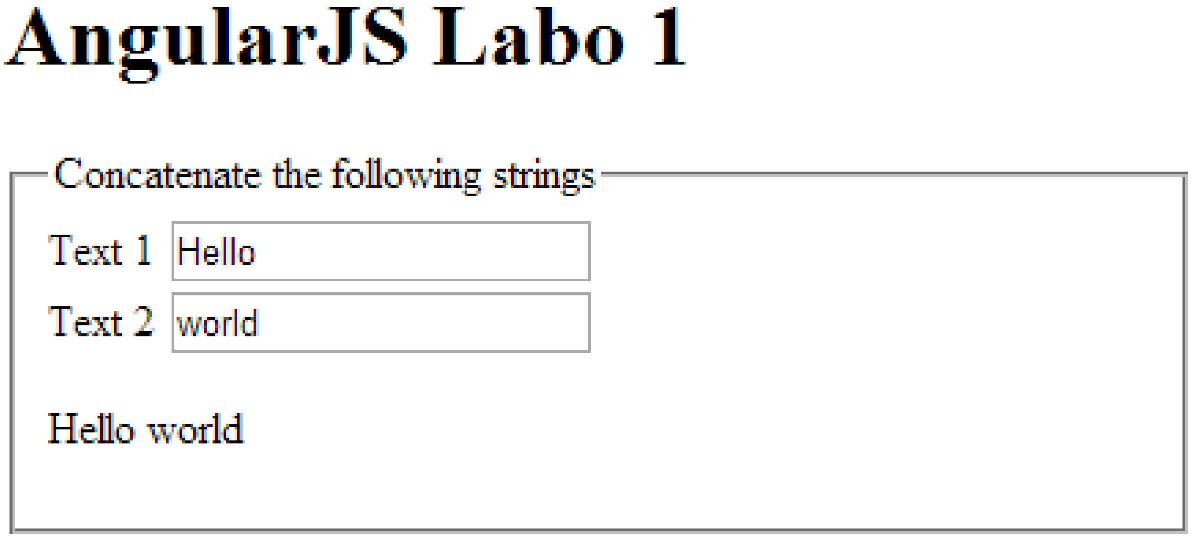
jqLite is a tiny, API-compatible subset of jQuery that allows Angular to manipulate the DOM in a cross-browser compatible way. **jqLite** implements only the most commonly needed functionality with the goal of having a very small footprint.

DEMO

ANGULARJS

LAB

- Modify the source of Demo 01 to get the following result



AngularJS Labo 1

Concatenate the following strings

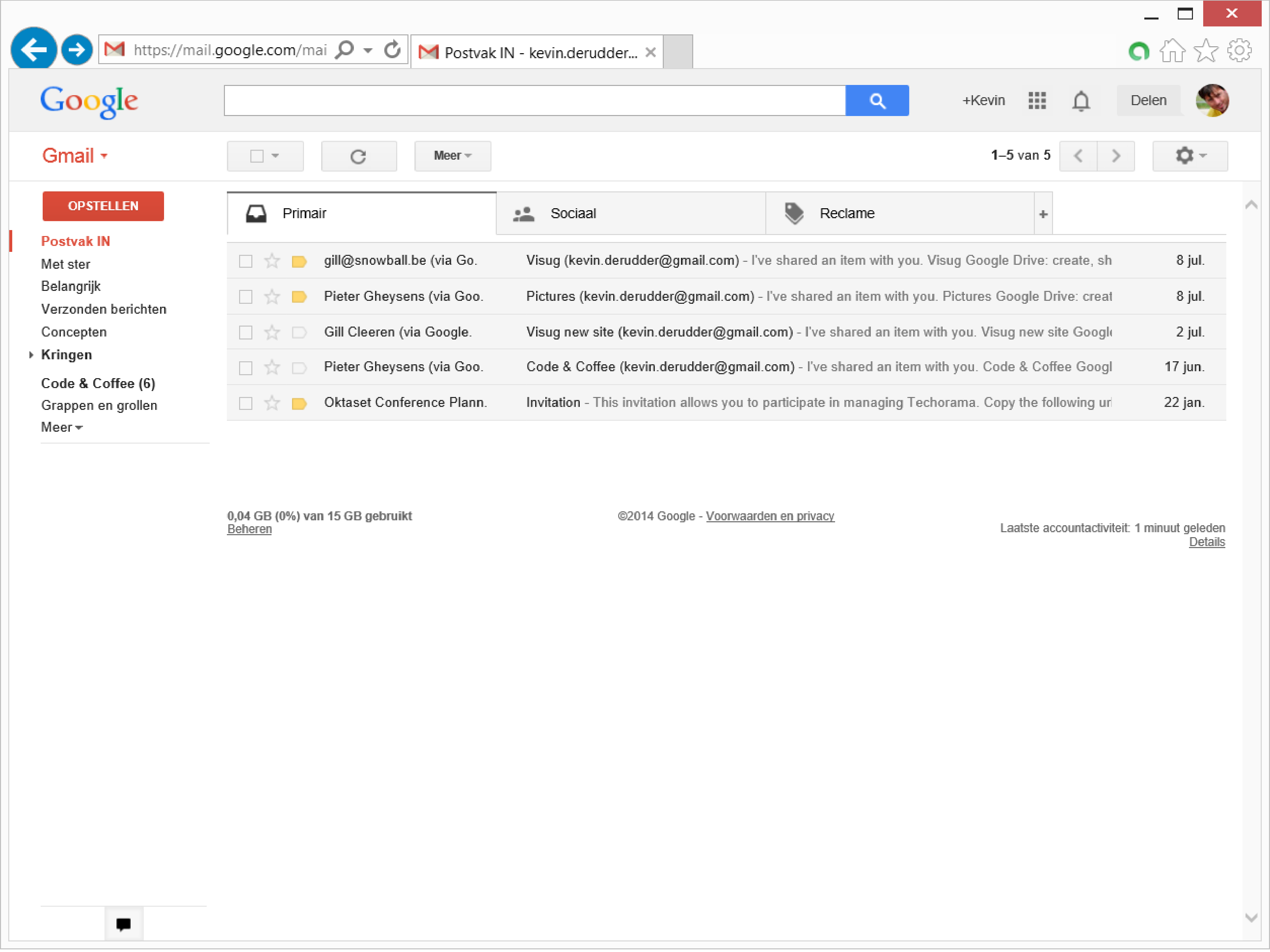
Text 1: Hello

Text 2: world

Hello world

AngularJS - Introduction

SINGLE PAGE APPLICATIONS



gill@snowball.be (via Go.	Visug (kevin.derudder@gmail.com) - I've shared an item with you. Visug Google Drive: create, sh	8 jul.
Pieter Gheysens (via Goo.	Pictures (kevin.derudder@gmail.com) - I've shared an item with you. Pictures Google Drive: creat	8 jul.
Gill Cleeren (via Google.	Visug new site (kevin.derudder@gmail.com) - I've shared an item with you. Visug new site Googl	2 jul.
Pieter Gheysens (via Goo.	Code & Coffee (kevin.derudder@gmail.com) - I've shared an item with you. Code & Coffee Googl	17 jun.
Oktaset Conference Plann.	Invitation - This invitation allows you to participate in managing Techorama. Copy the following ur	22 jan.

0,04 GB (0%) van 15 GB gebruikt
Beheren

©2014 Google - Voorwaarden en privacy

Laatste accountactiviteit: 1 minuut geleden
Details

Single Page Application

- A web application that fits on 1 page
  - The idea that you don’t have page reloads or post backs
- Initial page render
  - A SPA can redraw any part of the UI without a refresh
  - This is considered as a shell (skeleton)
- Provide native-application like experience

Single Page Application

- Data is separated from the presentation
  - Bring data back and forward
  - Separation of concerns (MV*)

- Lots, really lots of asynchronous requests

Advantages

- Quick experience for the user
  - You are loading the bare minimum
  - Load data & views on demand

- Performance and user experience
  - Because you only load what is needed
  - Less bandwith

Disadvantages (challenges)

- Search Engine Optimization
  - Your data must be on the page
- Browser History
- Requires JavaScript
- Memory leaks
  - Getting data in, adding stuff to the dom
  - Dynamically adding listeners, …

AngularJS - Introduction

ARCHITECTURE

Architecture

Services

Controllers

Views and Directives

Reusable logic

Tasks with

Architecture

- 2-way (data) binding
  - Keeps model and view in sync

- Dependency Injection
  - Important feature in AngularJS

AngularJS - Introduction

GETTING STARTED

Only 2 things to enable Angular

- Script tag pointing to angular.js

```
    </footer>
    <script src="Scripts/angular.min.js"></script>
</body>
```

- ng-app attribute in your HTML

```
<body ng-app>
```

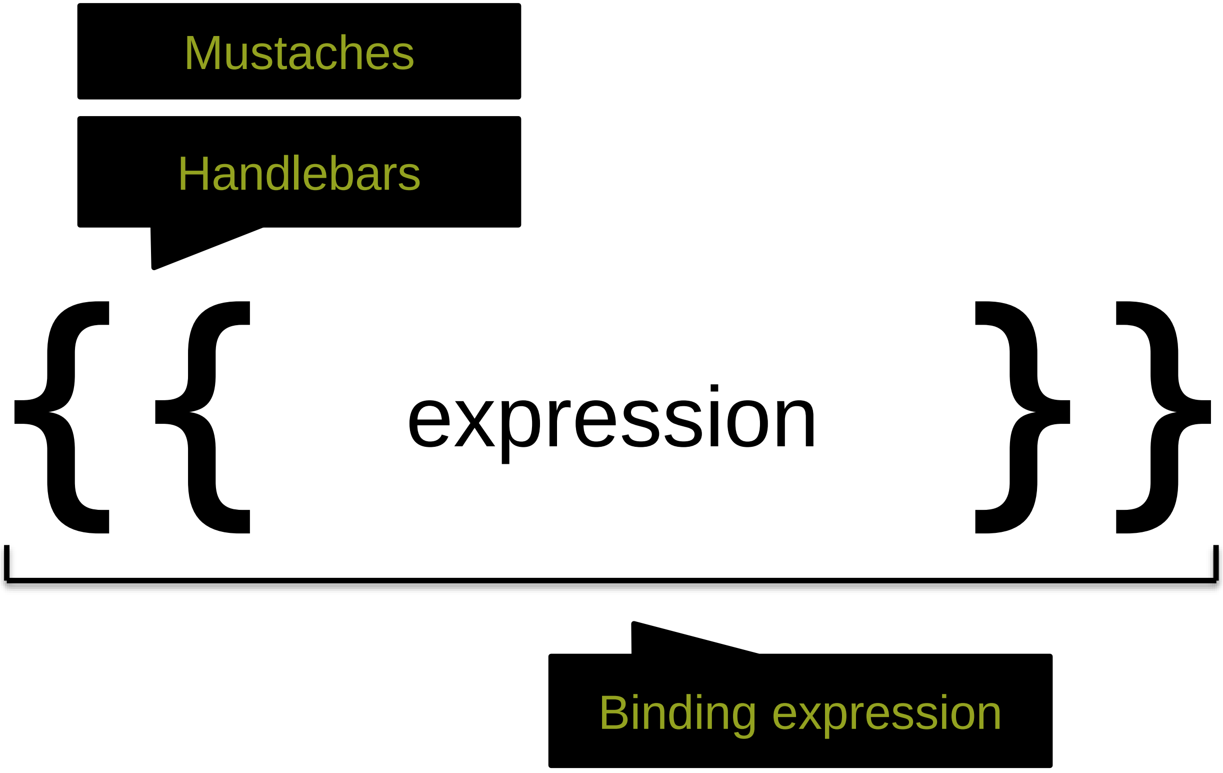
  – ng-app is an angular directive
  – ng is short for Angular
  – more valid would be data-ng-app

angular.js script file

- The only script necessary for your angular applications
  - No dependencies to other libraries
- Use the minified version on your live web application
- Use the uncompressed version if you modify angular
  - Not a best practice

AngularJS - Introduction

EXPRESSIONS



Mustaches

Handlebars

{{ expression }}

Binding expression

Binding Expressions

- Angular looks for the handlebars and tries to evaluate the expression inside

- Expressions can be anything
  - Additions, subtractions, …
  - Visualize data coming from a datasource

Angular Introduction

DIRECTIVES

ng-app

- ng-app is an angular directive
  - To be more specific: the application directive

- Can be set on any element
  - Only 1 application directive per page

```
<html ng-app>
```

```
<section ng-app>
    area with angular functionality
</section>
```

- Angular only controls that part of the DOM where the attribute is set

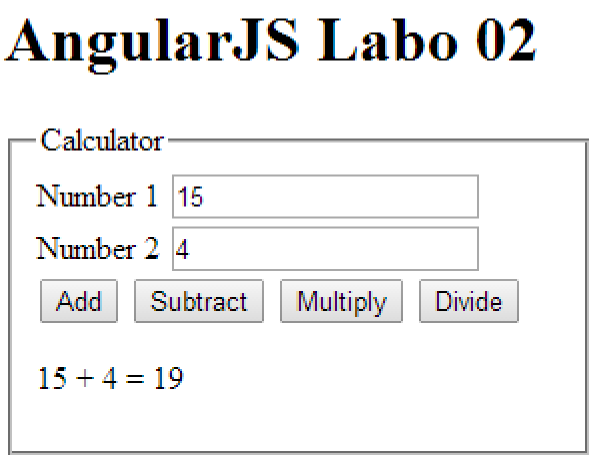
ng-model

- Your model with the data that has been sent to the page

- Default model created by angular when no model is applied

ng-click

- Your click-event connection between the page and the controller
  - More on this later



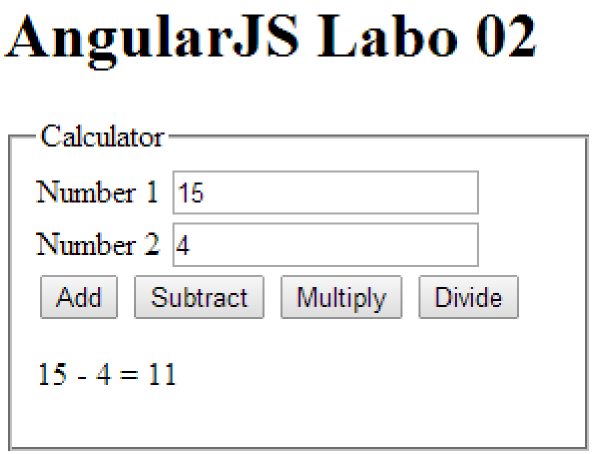
AngularJS Labo 02

Calculator

Number 1: 15

Number 2: 4

15 + 4 = 19



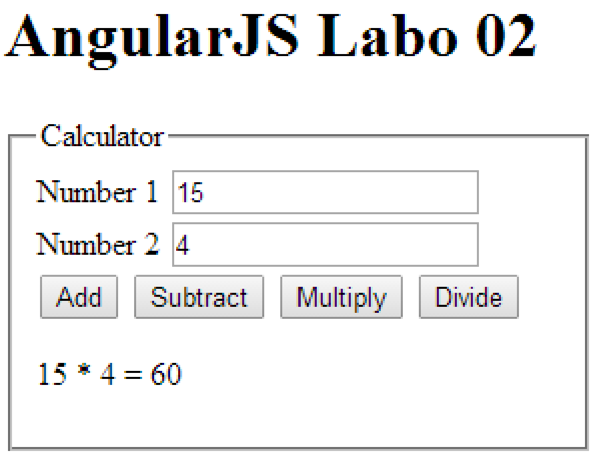
AngularJS Labo 02

Calculator

Number 1: 15

Number 2: 4

15 - 4 = 11



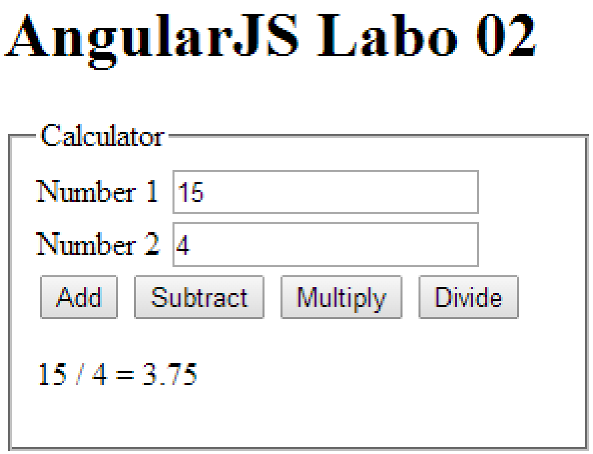
AngularJS Labo 02

Calculator

Number 1: 15

Number 2: 4

15 * 4 = 60



AngularJS Labo 02

Calculator

Number 1: 15

Number 2: 4

15 / 4 = 3.75

AngularJS - Introduction

ORGANIZING YOUR CODE

Seed Project

- Start point when creating Angular Applications
  - In a far distance, think of twitter bootstrap for angular
  - Or creating an MVC project in Visual Studio
- Most of the templates are created for smaller projects
  - With some additional organization, also usable for bigger projects
- Has its own web server

Angular Seed

```
> App
    > js
        > controllers
        > services
        > filters
        > directives
    > css
    > partials
    > img
    > lib
```

Writing all controllers in 1 file makes it hard to find a controller or service

You don’t have any idea how big your application is by looking at the directory

Angular Seed

```
> App
    > js
        > controller1
        > controller2
        > service1
        > …
    > css
    > partials
    > img
    > lib
```

Works well for tiny applications

If you have lots of file, your js folder is like a garbage dump

Angular by feature

```
> App
    > authentication
        > authCtrl.js
        > authSvc.js
        > authDirectives
        > auth.html
    > course
        > courseCtrl.js
        > …
    > css
    > partials
    > img
    > lib
```

Works from medium to large sites

Is good modular

Files are spread out through the app

Angular by feature

```
> App
    > js
        > controller1
        > controller2
        > service1
        > …
    > css
    > partials
    > img
    > lib
```

Works well for tiny applications

If you have lots of files, your js folder is like a garbage dump

Angular by feature then type

```
> App
    > course
        > CourseDetails
            > controllers
                > ctrl1
            > services
                > svc1
    > css
    > partials
    > img
    > lib
```

Angular by feature

```
> App
    > course
        > coursedetails
            > coursedetailsCtrl
            > coursedetailsSvc.js
            > coursedetails.html
        > courseoverview
            > …
    > css
    > partials
    > img
    > lib
```