



Ministerie van Defensie

# *Crossplatform Mobile Apps*

## Module 2 – Framework 7



Peter Kassenaar – [info@kassenaar.com](mailto:info@kassenaar.com)

# Onze keuze, Framework 7

The image displays the Framework7 website and its mobile application. The website has an orange background with the Framework7 logo (a red circle with a white '7') and the text "Full featured HTML framework for building iOS & Android apps". It includes "Download" and "Get Started" buttons, a navigation bar with "Docs", "Templates", "Apps Showcase", and "Forum", and a PhpStorm advertisement. The mobile app, shown on the right, has a black header with "Framework7" and a search icon. It features a list of components: "About Framework7", "Accordion", "Action Sheet", "Autocomplete", "Badge", "Buttons", "Calendar / Date Picker", "Cards", and "Checkbox". The app also shows status bar information like "BELL", "4:21 PM", and "100%" battery.

**Framework7**

Full featured HTML framework  
for building iOS & Android apps

[Download](#) [Get Started](#)

[Docs](#) [Templates](#) [Apps Showcase](#) [Forum](#)

Lightning-smart  
PHP IDE  
**PhpStorm**  
Try FREE now

Try PhpStorm - the PHP  
IDE that gets your code.  
Free trial, license from  
\$119.  
ads via Carbon

MIT Licensed, v3.2.1 released on August 31, 2018

12,933 stars 2,781 forks Support Framework7

**Framework7**

About Framework7

COMPONENTS

- Accordion
- Action Sheet
- Autocomplete
- Badge
- Buttons
- Calendar / Date Picker
- Cards
- Checkbox

## Framework 7 ook weer : 2 delen



The diagram consists of three vertical rectangular blocks of different heights and colors. The first block on the left is blue and contains the text 'Componenten (HTML/CSS)'. The second block in the middle is yellow and contains the text 'Programmeren (JavaScript)'. The third block on the right is cyan and contains the text 'Case'. The blue and yellow blocks are of equal height, while the cyan block is significantly shorter.

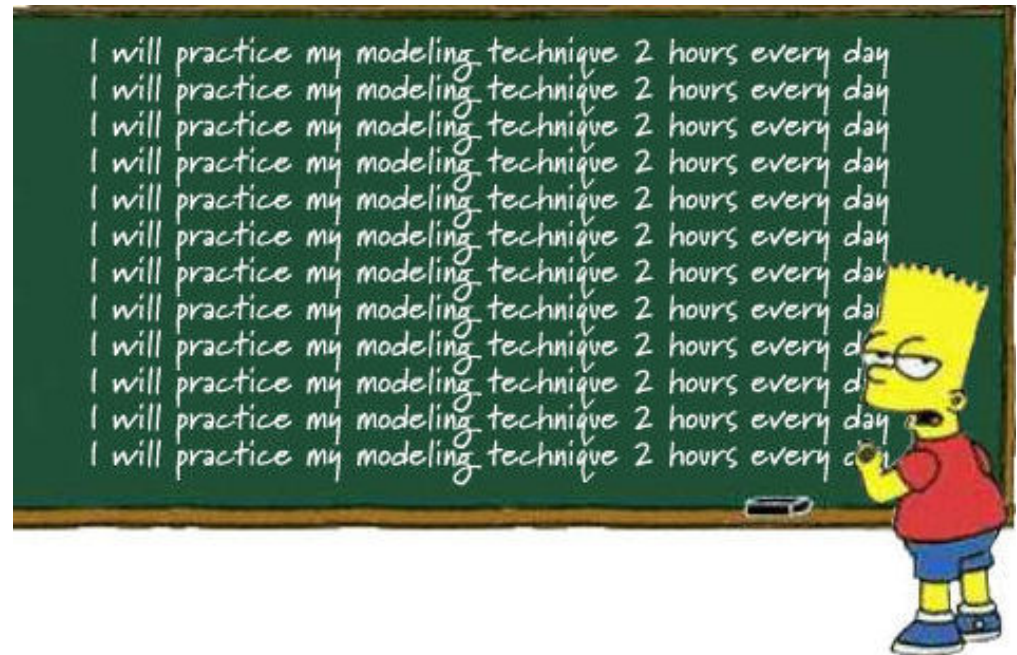
**Componenten  
(HTML/CSS)**

**Programmeren  
(JavaScript)**

**Case**

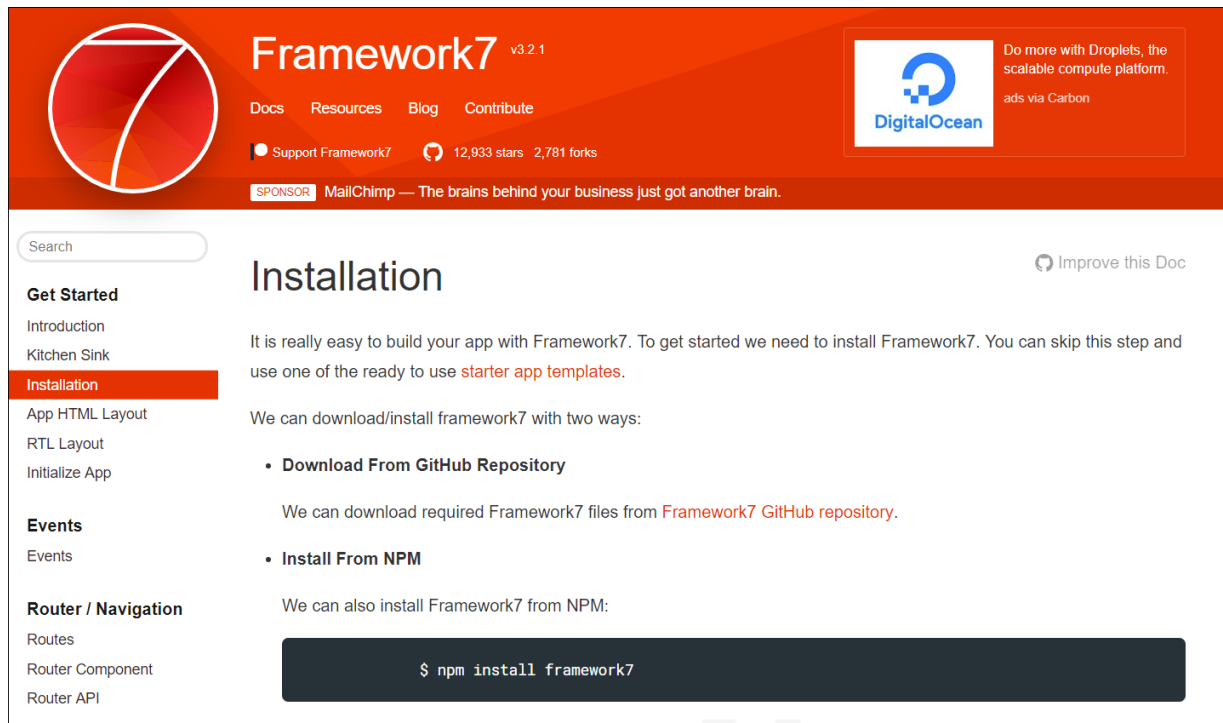
# Oefeningen

- Demo – eerste site/app maken.
- Daarna zelf doen.



# Getting Started

- Download & installatie - kan intimiderend overkomen.
  - Geen 'one click install', of een wizard
  - Kale Github-repository
- <https://framework7.io/docs/installation.html>
- Documentatie: Deel over ES Module en ES-Next kun je overslaan.



The screenshot shows the Framework7 v3.2.1 documentation page. The header is orange with the Framework7 logo, navigation links (Docs, Resources, Blog, Contribute), and a DigitalOcean advertisement. A sidebar on the left contains a search bar and a table of contents with 'Installation' highlighted. The main content area is titled 'Installation' and explains that it's easy to build an app with Framework7. It lists two ways to install: 'Download From GitHub Repository' and 'Install From NPM'. The NPM section includes a terminal command: `$ npm install framework7`.

Framework7 v3.2.1

Docs Resources Blog Contribute

Support Framework7 12,933 stars 2,781 forks

SPONSOR MailChimp — The brains behind your business just got another brain.

Search

Improve this Doc

**Get Started**

- Introduction
- Kitchen Sink
- Installation**
- App HTML Layout
- RTL Layout
- Initialize App

**Events**

- Events

**Router / Navigation**

- Routes
- Router Component
- Router API

## Installation

It is really easy to build your app with Framework7. To get started we need to install Framework7. You can skip this step and use one of the ready to use [starter app templates](#).

We can download/install framework7 with two ways:

- **Download From GitHub Repository**

We can download required Framework7 files from [Framework7 GitHub repository](#).

- **Install From NPM**

We can also install Framework7 from NPM:

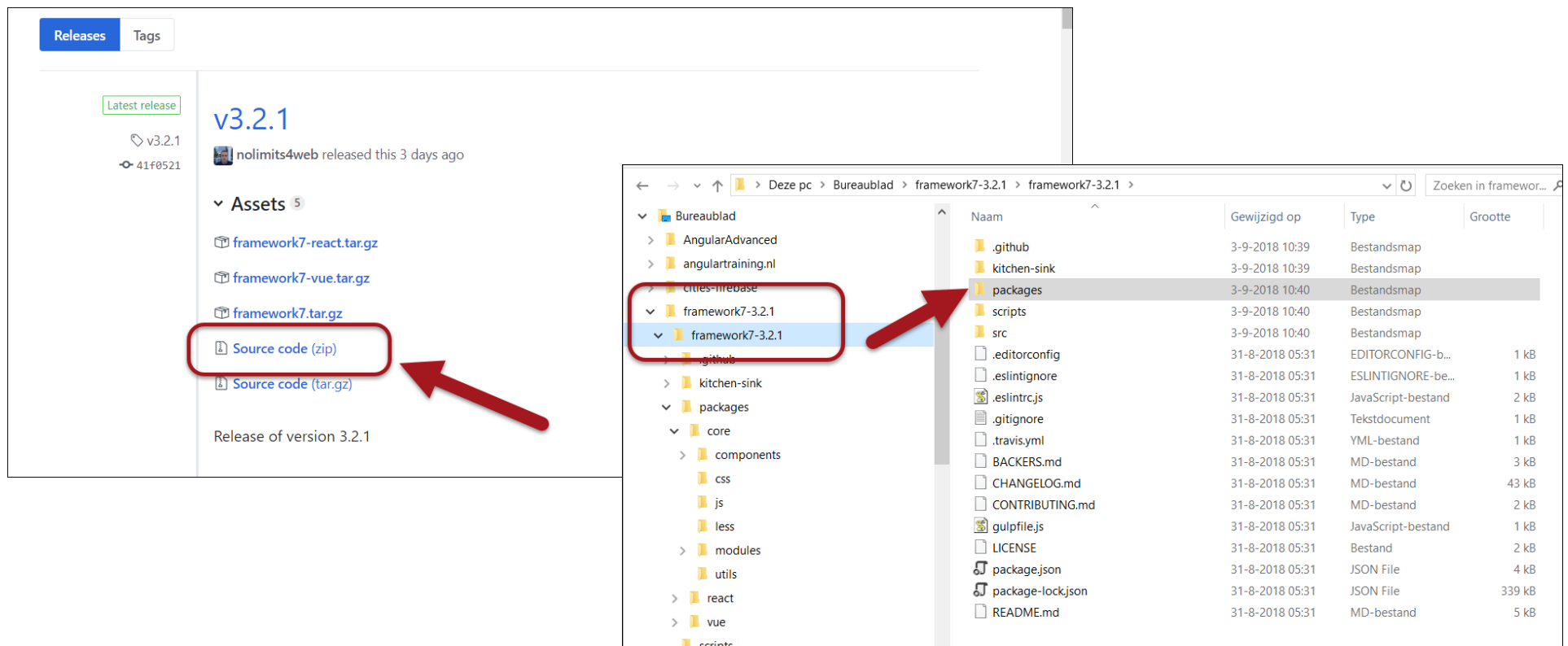
```
$ npm install framework7
```

# Werkwijze

## 1. Download Framework 7 files van de Github Repository

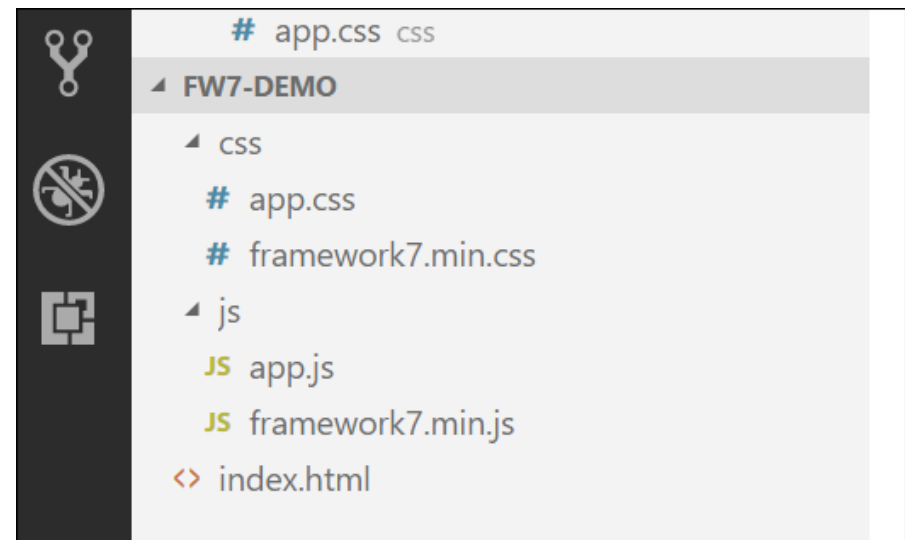
1. <https://github.com/framework7io/framework7/releases>

## 2. Unzip de bestanden op je bureaublad



# Maak een project in VS Code

- Maak een nieuwe map, bijvoorbeeld `fw7-demo`
- Maak hierin alvast de mappen `/css` en `/js`
- Kopieer vanuit de gedownloade map `/packages/core` de bestanden
  - `/js/framework7.min.js` en
  - `/css/framework7.min.css`
  - ...naar de mappen van je VS Code-project
- Maak zelf de lege bestanden `index.html`, `app.css` en `app.js`
- De structuur ziet er dan zo uit:



# Standaard App Lay-out kopiëren

- Ga naar <https://framework7.io/docs/app-layout.html> en kopieer de standaard HTML App Lay-out
- Plak deze in je eigen `index.html`

Search

Get Started

Introduction

Kitchen Sink

Installation

App HTML Layout

RTL Layout

Initialize App

Events

Events

Router / Navigation

Routes

Router Component

Router API

Components

App / Core

Accordion / Collapsible

Action Sheet / Actions

Autocomplete

Badge

App HTML Layout

First thing we should do for our App is to create `index.html` file with app's skeleton. Let's look at basic app layout:

```
<!DOCTYPE html>
<html>
  <head>
    <!-- Required meta tags-->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, minimum-scale=1, user-scalable=no, minimal-ui, viewport-fit=cover">
    <meta name="apple-mobile-web-app-capable" content="yes">
    <!-- Color theme for statusbar -->
    <meta name="theme-color" content="#2196f3">
    <!-- Your app title -->
    <title>My App</title>
    <!-- Path to Framework7 Library CSS -->
    <link rel="stylesheet" href="path/to/framework7.min.css">
    <!-- Path to your custom app styles-->
    <link rel="stylesheet" href="path/to/my-app.css">
  </head>
  <body>
    <!-- App root element -->
    <div id="app">
      <!-- Statusbar overlay -->
      <div class="statusbar"></div>

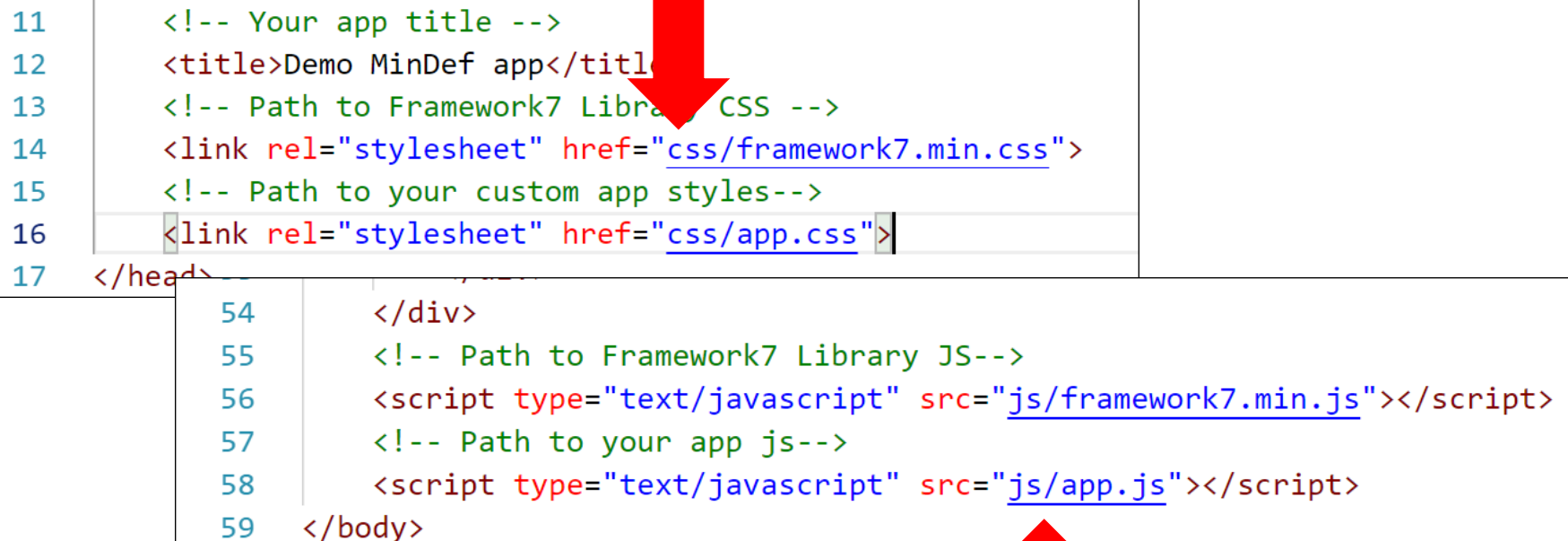
      <!-- Your main view, should have "view-main" class -->
```



# App lay-out aanpassen

Let er op dat je in de `<head>` het pad naar de CSS-bestanden aanpast!

Let er op dat je onderin het pad naar de JavaScript-bestanden aanpast!



```
11 <!-- Your app title -->
12 <title>Demo MinDef app</title>
13 <!-- Path to Framework7 Library CSS -->
14 <link rel="stylesheet" href="css/framework7.min.css">
15 <!-- Path to your custom app styles-->
16 <link rel="stylesheet" href="css/app.css">
17 </head>

54 </div>
55 <!-- Path to Framework7 Library JS-->
56 <script type="text/javascript" src="js/framework7.min.js"></script>
57 <!-- Path to your app js-->
58 <script type="text/javascript" src="js/app.js"></script>
59 </body>
```

# JavaScript toevoegen

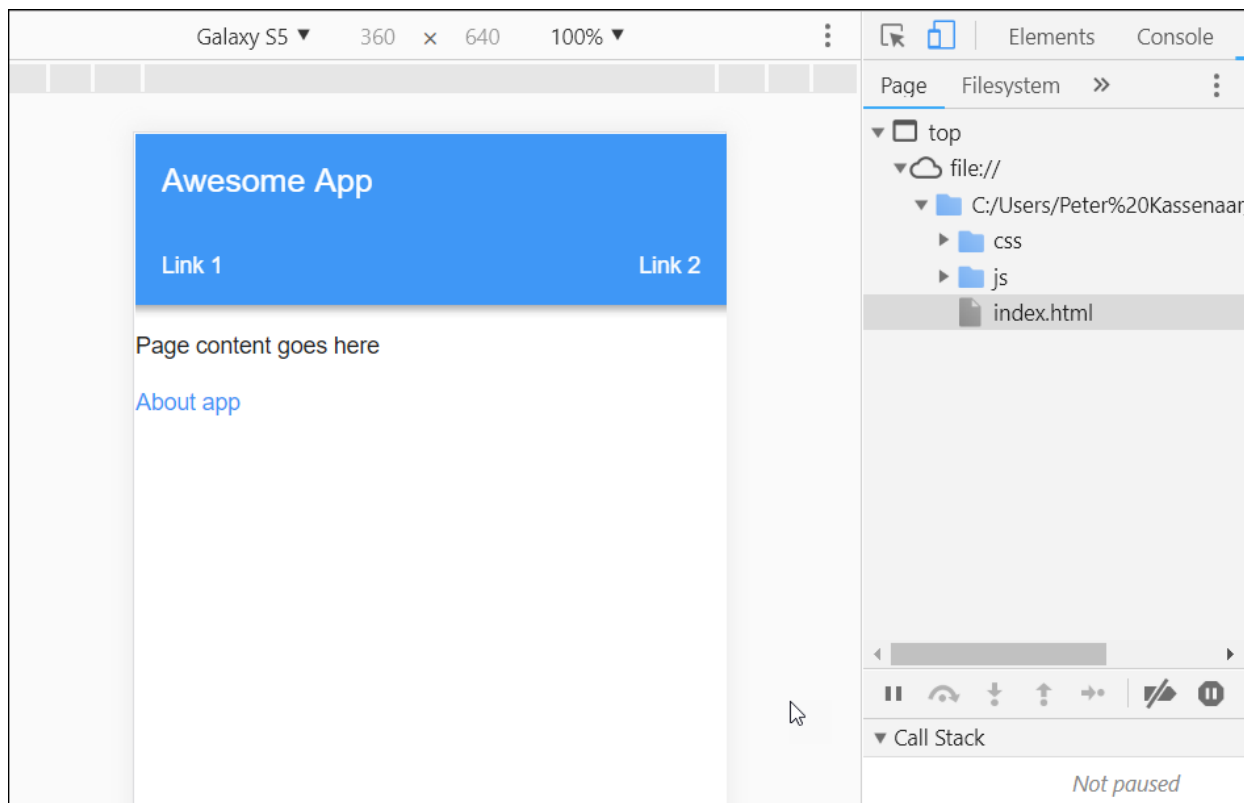
- Framework 7 werkt niet zonder initialisatie in JavaScript.
- Open je eigen `app.js`-bestand
- Ga naar <https://framework7.io/docs/init-app.html> en kopieer de voorbeeldcode naar je eigen bestand

```
var app = new Framework7({
  // App root element
  root: '#app',
  // App Name
  name: 'My App',
  // App id
  id: 'com.myapp.test',
  // Enable swipe panel
  panel: {
    swipe: 'left',
  },
  // Add default routes
  routes: [
    {
      path: '/about/',
      url: 'about.html',
    },
  ],
  // ... other parameters
});

var mainView = app.views.create('.view-main');
```

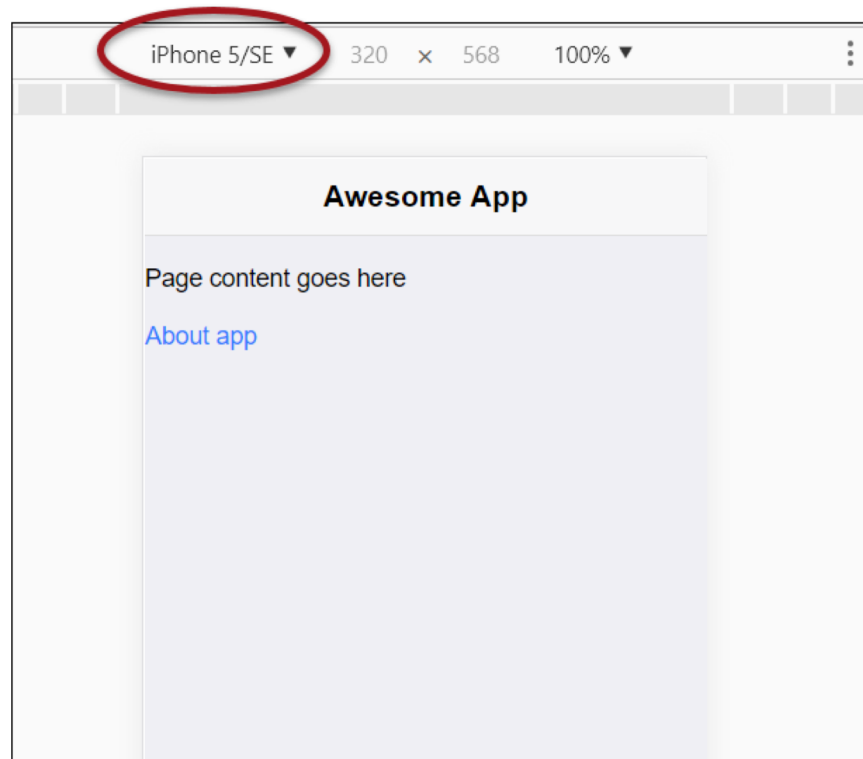
# Testen in browser

- Open de pagina index.html in de browser, vanuit VS Code
  - (met de extensie is dat via Alt+B of Alt+Shift+B)
- Hij ziet er als volgt uit:

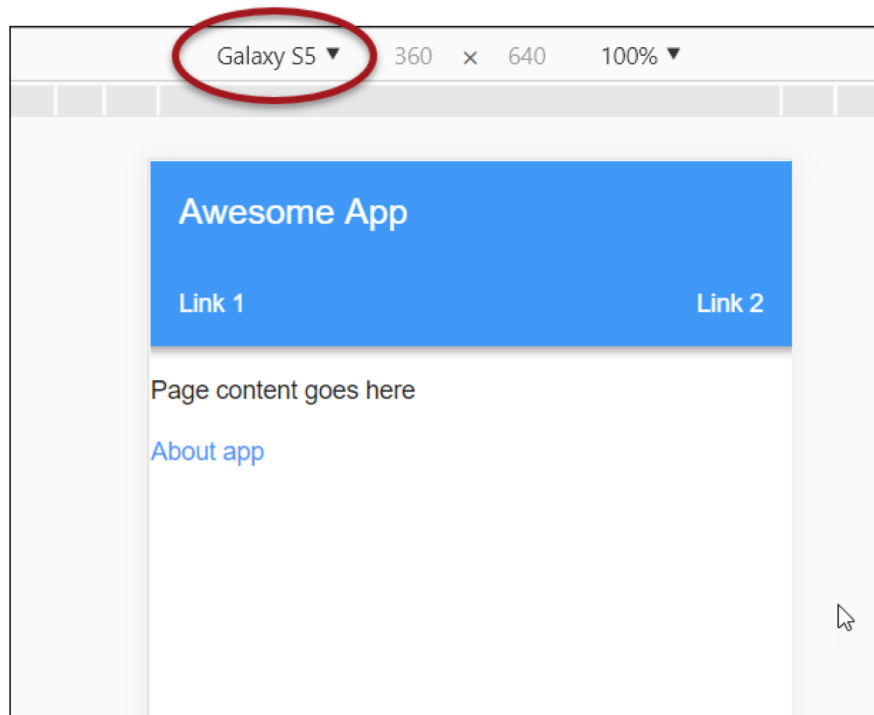


# Test verschillende platforms

## iOS Device

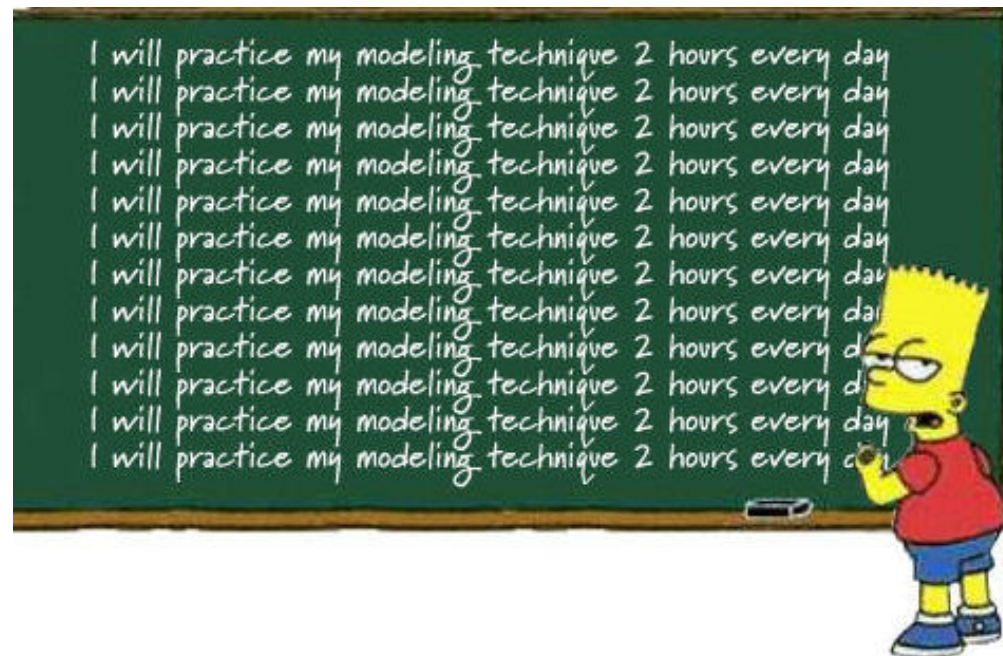


## Android Device



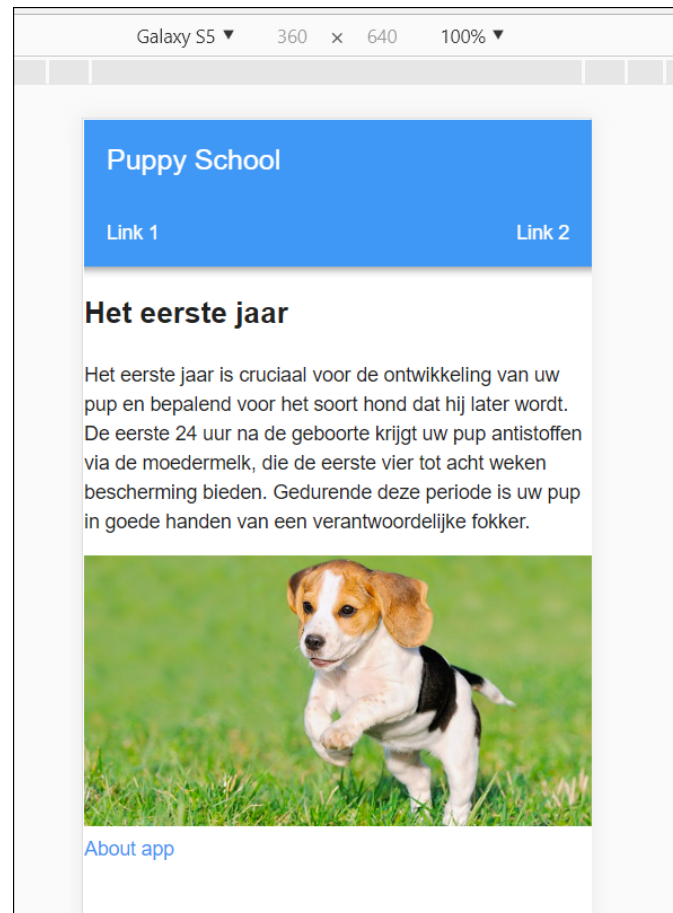
# Oefening

- Zelf doen
  - Maak een standaard App op basis van online voorbeeld
  - App heeft nog geen functionaliteit!



# 1. Pagina uitbreiden

- Breid de pagina uit met enige tekst en afbeeldingen
- Rest van de componenten en opmaak komt later nog!



## 2. Nieuwe pagina toevoegen

- Maak een nieuwe pagina `about.html`
- Deze is al voorgedefinieerd in de routing-table (zie `app.js`)
- Hij heeft wel speciale kenmerken:
  - Gebruik een `<div class="page">`
  - Voeg de klasse `.navbar` en `.navbar-inner` toe in de page
  - Gebruik een `<div class="page-content">` voor de inhoud
- Serveer de site over `http`!
  - Gebruik een webserver
  - Pagina's worden via AJAX geladen. Dit werkt niet op het `file://`-protocol

```
<div class="page" data-name="about">
  <div class="navbar">
    <div class="navbar-inner sliding">
      <div class="left">
        <a href="#" class="link back">
          <i class="icon icon-back"></i>
          <span class="ios-only">Back</span>
        </a>
      </div>
      <div class="title">Over ons</div>
    </div>
  </div>
  <div class="page-content">
    <h1>Over ons</h1>
    ...
  </div>
</div>
```





← Over ons

## Over ons

### Kassenaar IT Services

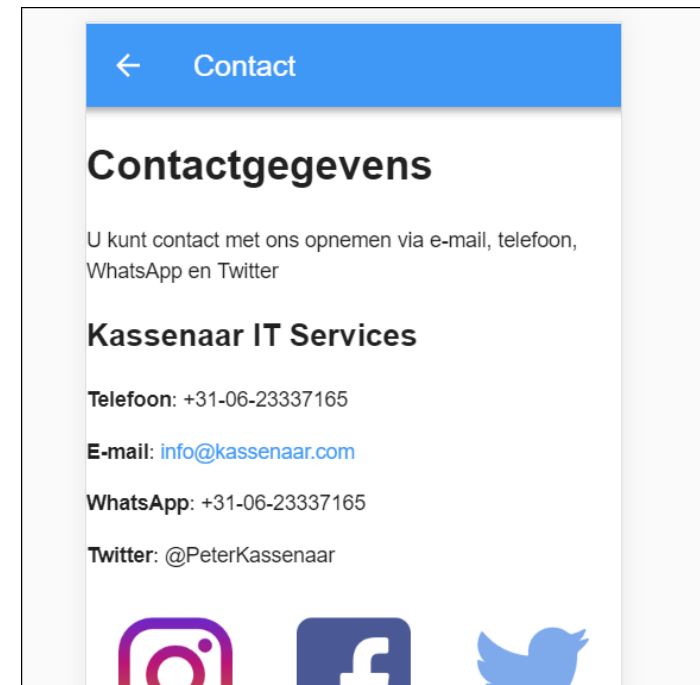
*Laarstraat 47-47a  
7201 CB Zutphen*



### 3. Extra pagina's toevoegen

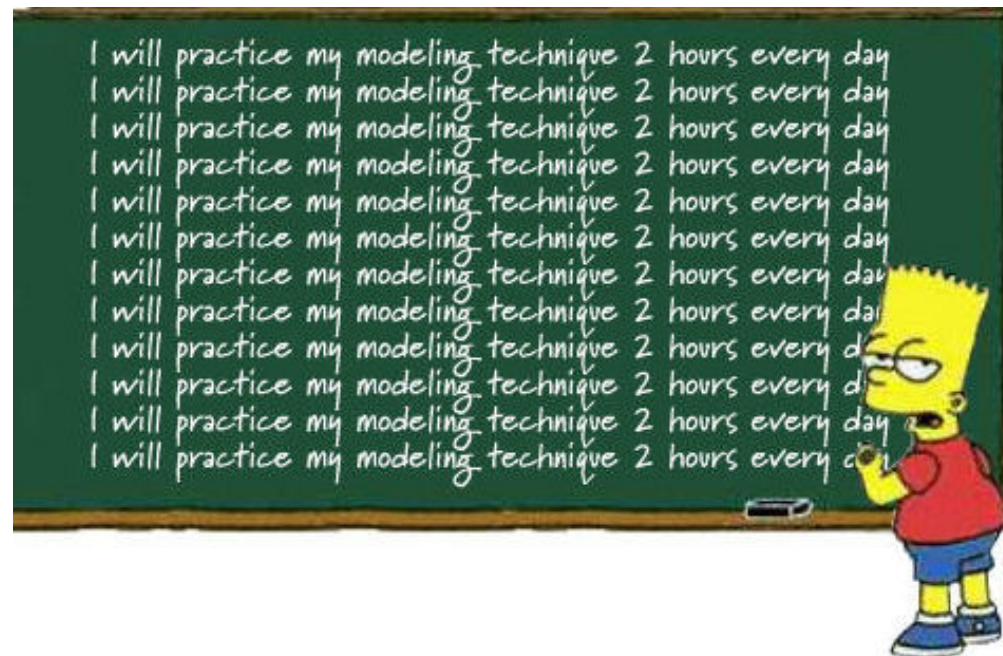
1. Breidt de array `routes` in `app.js` uit met de nieuwe route en nieuwe pagina('s).
2. Maak de nieuwe pagina's en koppel ze via de homepage.
3. Bijvoorbeeld:

```
// app.js - Add default routes
routes: [
  {
    path: '/about/',
    url: 'about.html'
  },
  {
    path: '/contact/',
    url: 'contact.html'
  }
]
```



# Oefening

- Zelf doen
  - Voeg twee pagina's toe aan de app
  - Maak de pagina's, denk aan het uitbreiden van de route-tabel!



# Configuratie parameters

- Elke Framework 7-applicatie heeft een `app`-object.
- Dit moet je opgeven in het JavaScript-bestand
- Herken de syntaxis. `var app = new Framework7();`
  - Aanroep van een functie
- De functie heeft een object als parameter

```
var app = new Framework7({  
  // App root element  
  root: '#app',  
  // App Name  
  name: 'My App',  
  // App id  
  id: 'com.myapp.test',  
  // Enable swipe panel  
  panel: {  
    swipe: 'left',  
  },  
  // Add default routes  
  routes: [  
    {  
      path: '/about/',  
      url: 'about.html',  
    },  
  ],  
  // ... other parameters  
});
```

```
// app.js
var app = new Framework7({
  root: '#app', // <-- Verwijst naar <div id="app">
  name: 'My App', // <-- Naam van de applicatie
  id: 'com.myapp.test', // <-- naam van de bundle, voor publicatie in App Stores
  panel: {
    swipe: 'left' // <-- // Swipe panel vanaf links (als je dit gebruikt)
  },
  routes: [
    // <-- Routes/pagina's binnen de app
    {
      path: '/about/',
      url: 'about.html'
    },
    {
      path: '/contact/',
      url: 'contact.html'
    }
  ]
  // <-- overige parameters
});
```

```
// Elke app heeft een main view, verwijst naar <div class="view view-main">
var mainView = app.views.create('.view-main');
```

# Overige parameters

Hoef je niet uit je hoofd te kennen, maar moet weten waar je ze kunt vinden

<https://framework7.io/docs/app.html>

## App Parameters

Lets look at the list of available parameters:

Parameter	Type	Default	Description
<b>root</b>	string	body	App root element. If you main app layout is not a direct child of the <code>&lt;body&gt;</code> then it is required to specify root element here
<b>id</b>	string	io.framework7.testapp	App bundle id.
<b>name</b>	string	Framework7	App name. Can be used by other components, e.g. as the default title for <code>Dialog</code> component.
<b>version</b>	string	1.0.0	App version. Can be used by other components.
<b>theme</b>	string	auto	App theme. Can be <code>ios</code> , <code>md</code> or <code>auto</code> . In case of <code>auto</code> it will use iOS theme for iOS devices and MD theme for all other devices.
<b>language</b>	string		App language. Can be used by other components. By default equal to the current browser/webview language (i.e. <code>navigator.language</code> ).
<b>routes</b>	array	[]	Array with default routes to all views.
<b>data</b>	function		App root data. Must be a function that returns an object with root data. For example:

```
var app = new Framework7({
  data: function () {
    return {
      username: 'vladimir'
```

# Een Framework 7 app

app is nu een globaal object

Je kunt de methodes, properties en events raadplegen

## App Methods & Properties

Returned Framework7 instance `app` contains a lot of useful properties and methods:

### Properties

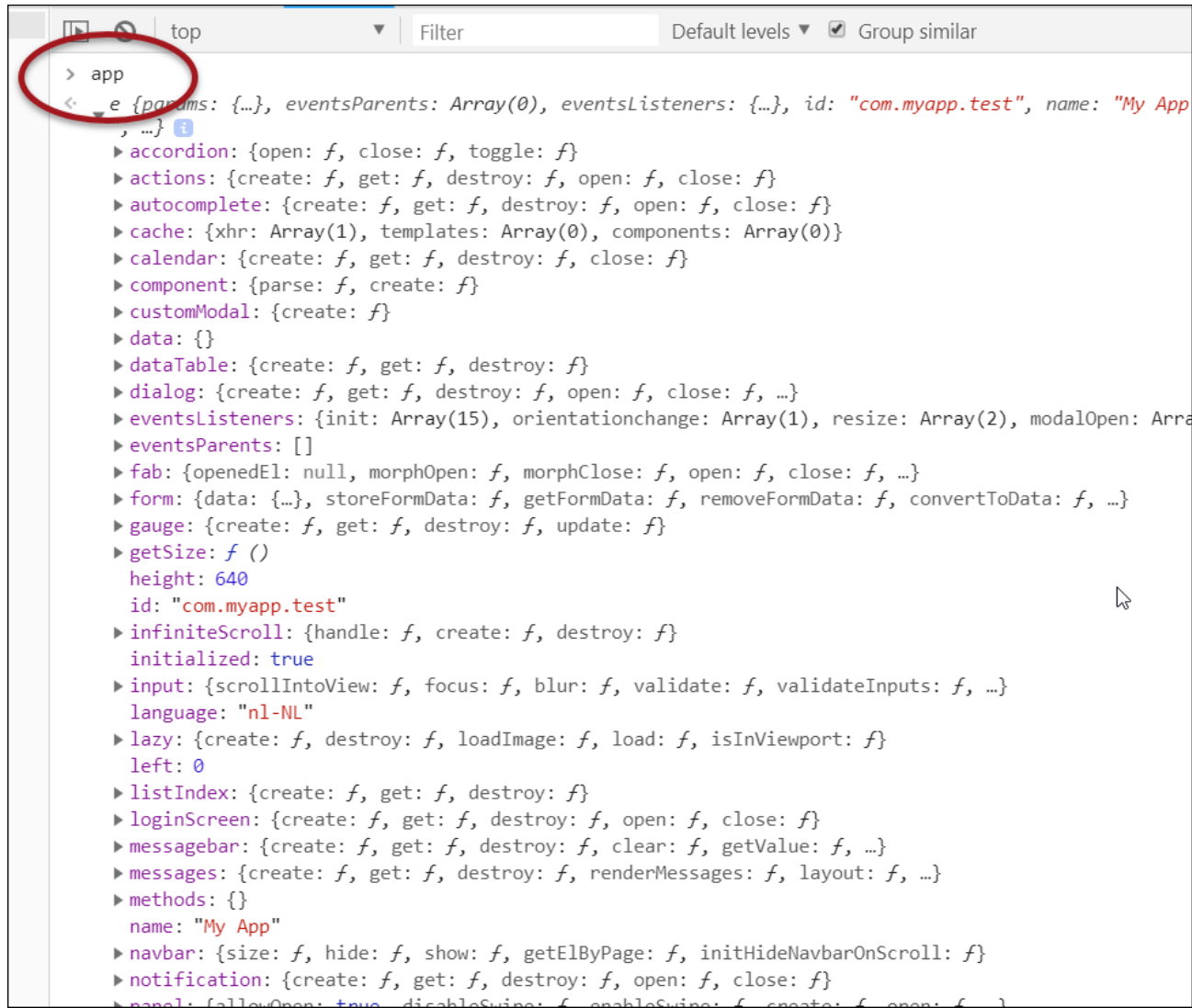
<code>app.id</code>	App ID passed in parameters
<code>app.name</code>	App name passed in parameters
<code>app.version</code>	App version
<code>app.routes</code>	App routes
<code>app.language</code>	App language

## App Events

App instance emits the following core events:

Event	Arguments	Description
<code>init</code>		Event will be fired on app initialization. Automatically after <code>new Framework7()</code> or after <code>app.init()</code> if you disabled auto init.
<code>resize</code>		Event will be fired on app resize (window resize).
<code>orientationchange</code>		Event will be fired on app orientation change (window orientation change).
<code>click</code>	<code>event</code>	Event will be fired on app click
<code>touchstart:active</code>	<code>event</code>	Event will be fired on touch start (mousedown) event added as active listener (possible to prevent default)

# App inspecteren via console





# Oefening

- Pas de naam en bundle-id aan van je app.  
Inspecteer dit in de console
- Gebruik de event handler `app.on('resize', function() {...})`.
  - Toon de huidige hoogte en breedte in de console bij een wisseling van schermgrootte

