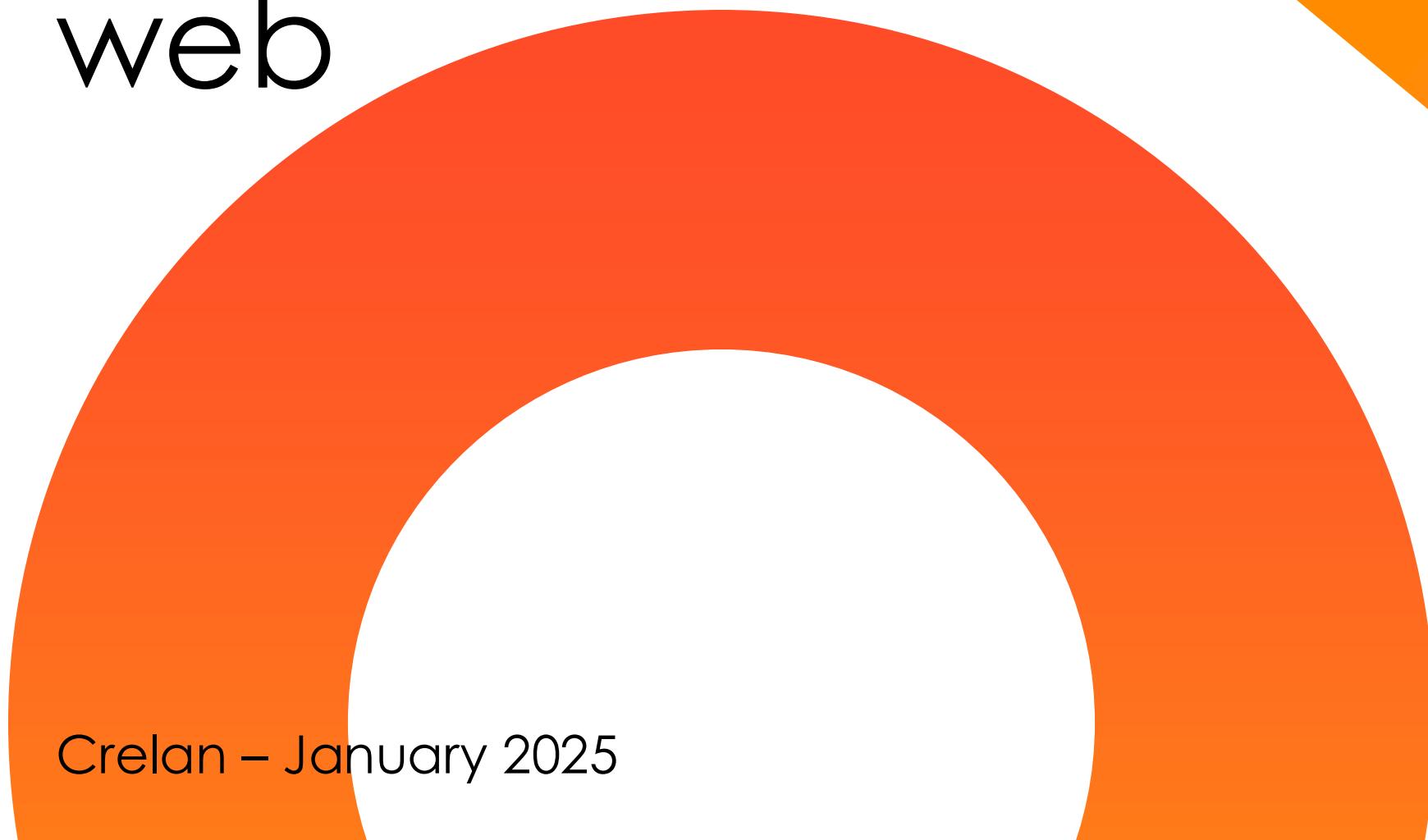


# Front end engineering for the web



Crelan – January 2025



# Introduction trainer

- Anton Thomas
- Angular developer since 2019



**IT board member**

Greenhill Capital



**Full stack developer**

[antonthomas.be](http://antonthomas.be)



**Full stack developer**

Farmad



# Introduction trainer

- Alexander Versweyveld
- Angular developer since 2020



2021

2023

acerta





# Ice Breaker – Agree or Disagree

de

# Ice Breaker – Agree or Disagree

Ik ben productiever  
tijdens remote work  
dan op kantoor /  
school

# Ice Breaker – Agree or Disagree

Ik ben productiever  
tijdens remote work dan  
op kantoor / school

Pair programming is  
essentieel voor de  
kwaliteit van de code

# Ice Breaker – Agree or Disagree

Ik ben productiever  
tijdens remote work dan  
op kantoor / school

Pair programming is  
essentieel voor de  
kwaliteit van de code

Het gebruiken van  
programmeer  
concepten is  
cruciaal, zelfs als dit  
meer complexiteit  
toevoegt

# Ice Breaker – Agree or Disagree

Ik ben productiever  
tijdens remote work dan  
op kantoor / school

Pair programming is  
essentieel voor de  
kwaliteit van de code

Het gebruiken van  
programmeer  
concurrentie  
zeer belangrijk  
complexe code

Developers zouden mee  
aan het (solution) design  
proces moeten werken

# Ice Breaker – Agree or Disagree

Ik ben productiever  
tijdens remote work dan  
op kantoor / school

Pair programming is  
essentieel voor de  
kwaliteit van de code

Het gebruiken van  
pairing en standup  
concreet voor  
zelfstandigheid  
compromis

Er is altijd maar 1 beste  
oplossing voor een  
programmeer probleem

Developers zijn  
meestal niet  
(solitair) program-

# Ice Breaker – Agree or Disagree

Ik ben productiever  
tijdens remote work dan  
op kantoor / school

Het gebruiken van  
programmamore

Er is altijd maar 1 beste  
manier om te doen

Pair programming is  
essentieel voor de  
kwaliteit van de code

Het is oke om code te copy  
pachten van het internet

# Ice Breaker – Agree or Disagree

Ik ben productiever  
tijdens remote work dan  
op kantoor / school

maar 1 beste  
ing voor een  
rammeer  
bleem

AI gaat in de toekomst  
developers vervangen

Pair programming is  
essentieel voor de  
kwaliteit van de code

Geen code te  
kosten van het  
internet

# Ice Breaker – Agree or Disagree

Ik ben productiever tijdens remote work dan op kantoor / school

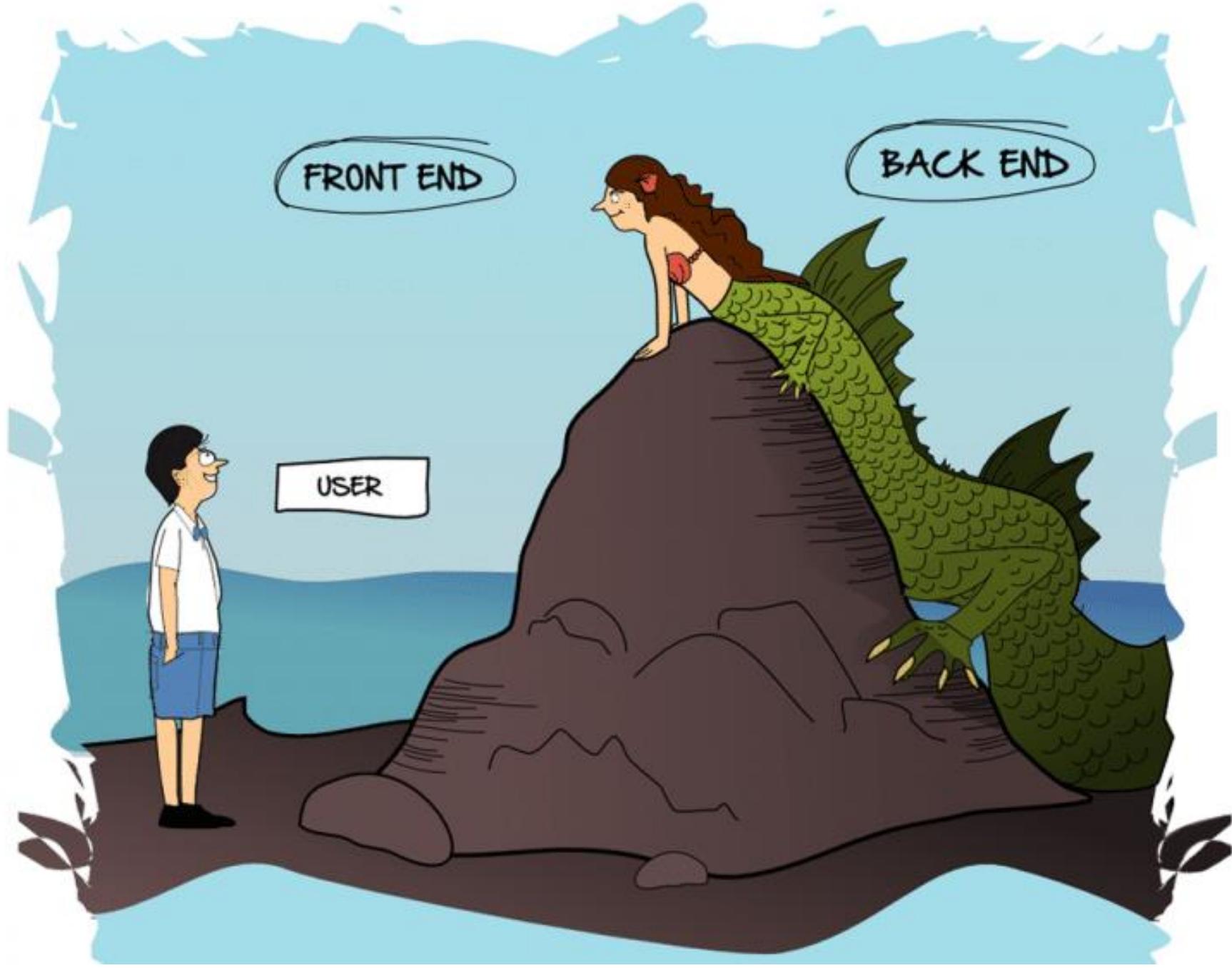
Pair programming is essentieel voor de kwaliteit van de code

Low level programmeertalen zijn beter dan high level programmeertalen

AI gaat in de toekomst developers vervangen

# Introduction attendees

- Tour de table
  - Who are you?
  - What do you expect of this course?
  - What are your thoughts on and experience with FEE?



WATCH OUT FOR THE UGLY MERMAID



de

# Agenda

1. JavaScript & TypeScript
2. Dependency management
3. Component design
4. Modern web frameworks
5. UX & UI
6. ALM
7. QA
8. Security (optional)

# Training resources

Code and slides

- <https://github.com/AE-nv/fee-for-the-web-angular>

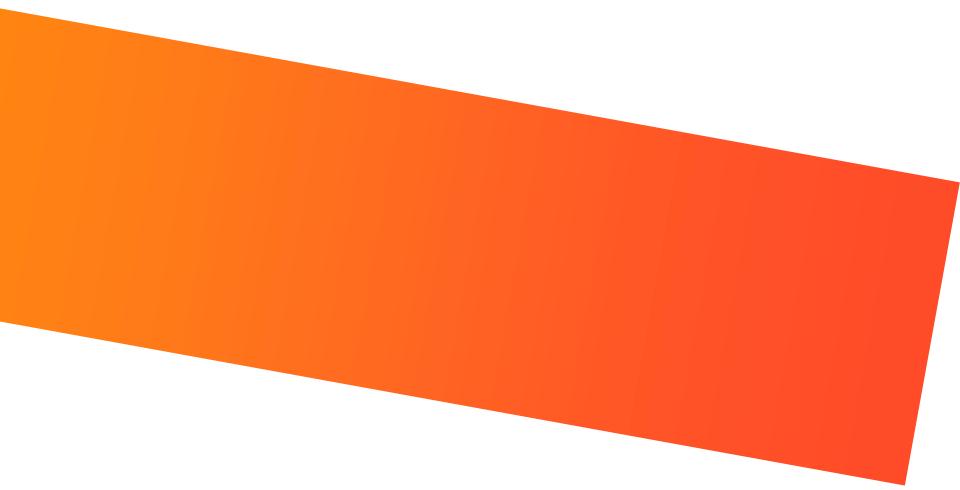
# IP disclaimer

The contents of this presentation can be downloaded but rights remain with AE; content cannot be duplicated for new courses.

# DAY 1

The basics

# JavaScript & TypeScript

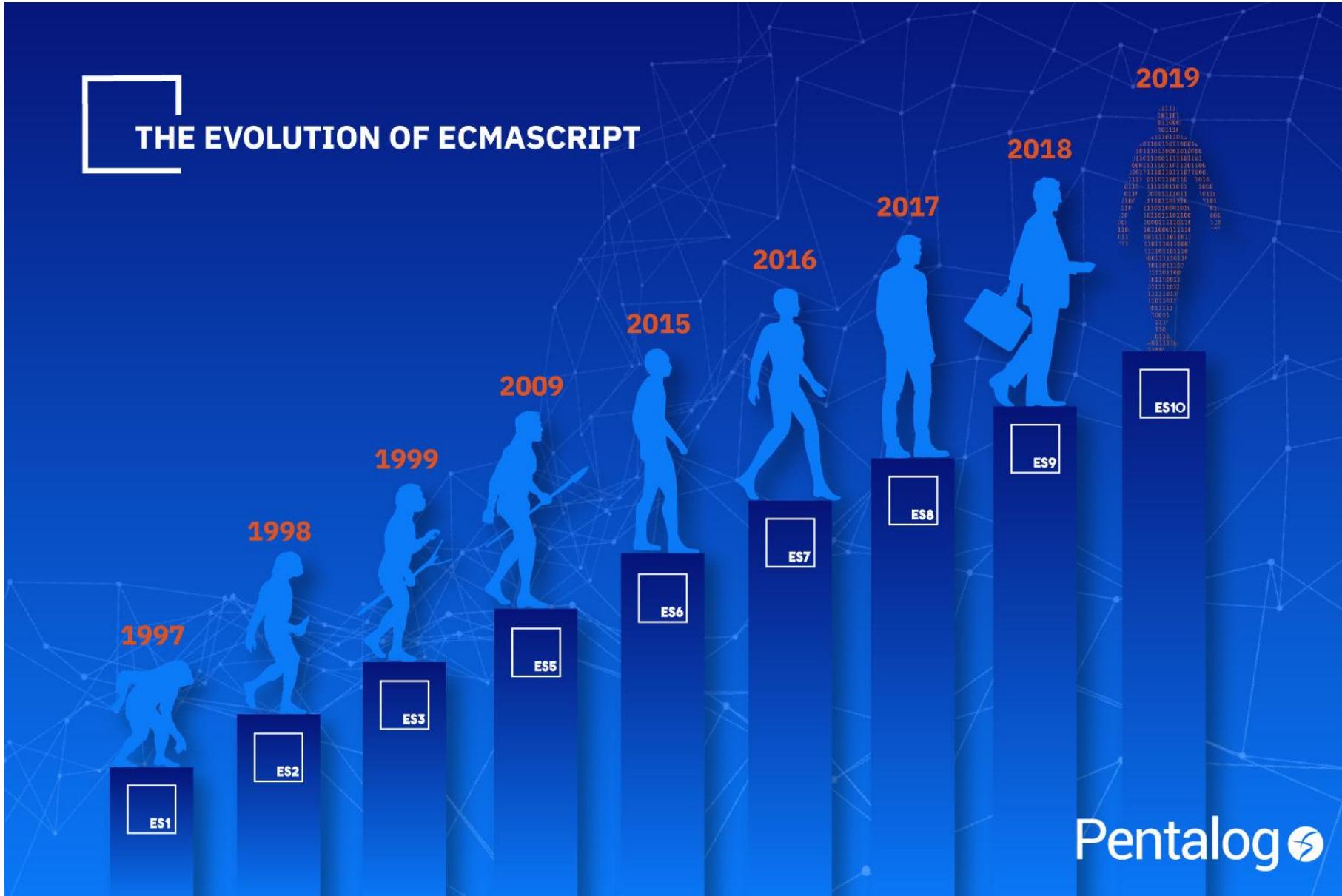




JavaScript is the language of the web.

It is built into every modern web browser and is thus available on almost every device.





# Modules

	<b>CommonJS require</b>	<b>ES Module import</b>
<b>CommonJS exports</b>	Works	Works <i>(Will need to explicitly use .mjs or .cjs extension for one of the modules)</i>
<b>ES Module export</b>	Error  <code>Error [ERR_REQUIRE_ESM]: Must use import to load ES Module</code>	Works

```
// CommonJS import example
const myModule = require('myModule');

//CommonJS export example
exports.myVariable = 10;

// ESM import example
import MyComponent from './myModule.js';

// ESM export example
export { myVariable, myClass };
```

# Supersets

- Languages that **compile** to plain JavaScript and provide syntactic sugar and utilities



TypeScript



Dart



elm





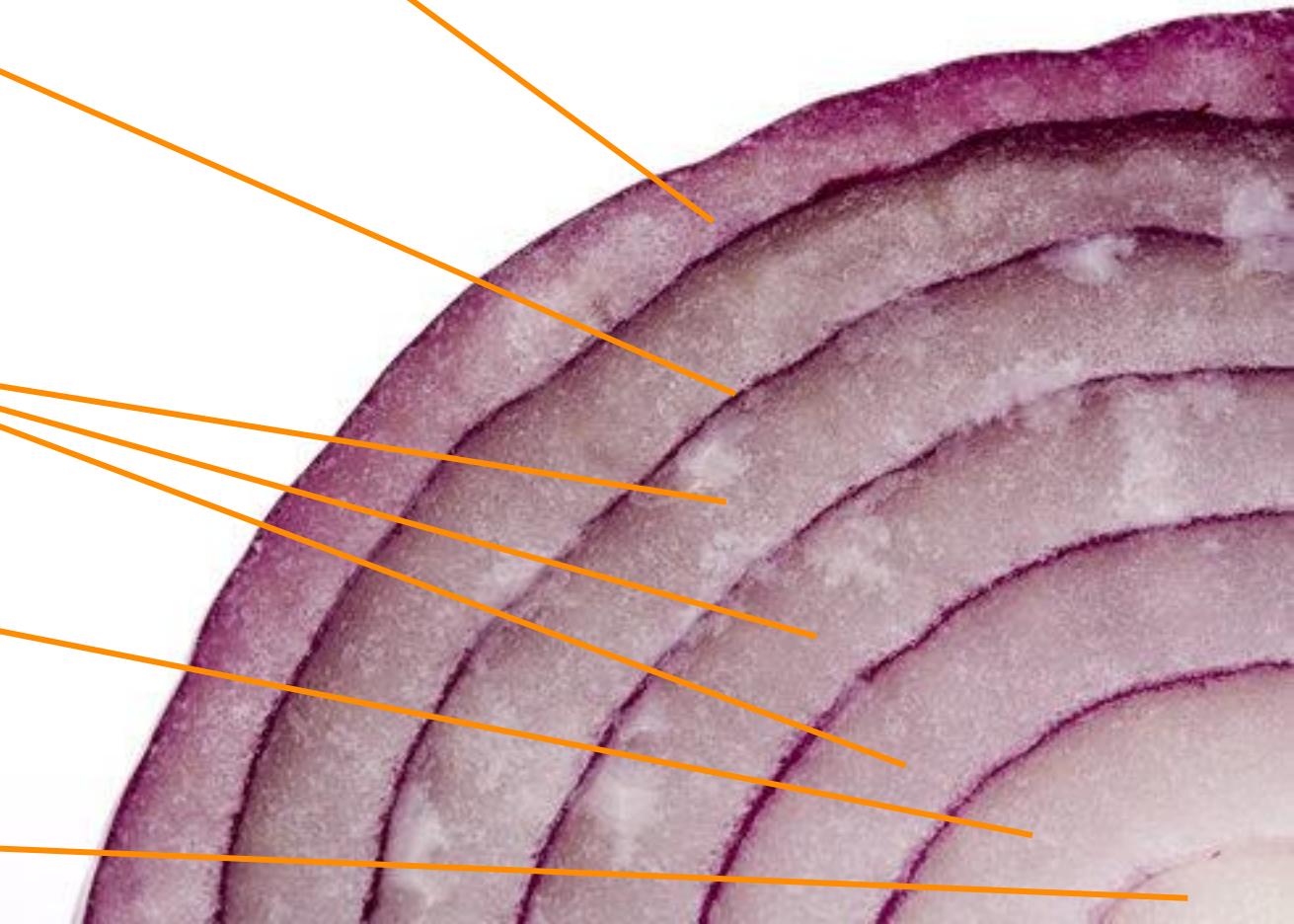
TypeScript

ES2025

ES...

ES2

ES1



# Transpilers

- **Convert** one language to another (e.g. typescript -> javascript)
- Checks for semantic errors
- Optimize code
- Type checking

# TypeScript - transpiling

```
1  {
2    "compilerOptions": {
3      "resolveJsonModule": true,
4      "esModuleInterop": true,
5      "module": "commonjs",
6      "declaration": true,
7      "removeComments": true,
8      "emitDecoratorMetadata": true,
9      "experimentalDecorators": true,
10     "allowSyntheticDefaultImports": true,
11     "target": "ES2021",
12     "sourceMap": true,
13     "outDir": "./dist",
14     "baseUrl": "./",
15     "incremental": true,
16     "skipLibCheck": true,
17     "strictNullChecks": false,
18     "noImplicitAny": false,
19     "strictBindCallApply": false,
20     "forceConsistentCasingInFileNames": false,
21     "noFallthroughCasesInSwitch": false
22   }
23 }
```

# TypeScript - Types

```
let isDone: boolean = false;
```

```
let fullName: string = `Bob Bobbington`;
let age: number = 37;
```

```
let list: number[] = [1, 2, 3];
```

```
function warnUser(): void {
    alert("This is my warning message");
}
```

# TypeScript – Generics & Enums

```
function identity<T>(arg: T): T {  
    return arg;  
}
```

```
let list: Array<number> = [1, 2, 3];
```

```
1 enum Direction {  
2     Up,  
3     Down,  
4     Left,  
5     Right  
6 }  
  
var Direction;  
(function (Direction) {  
    Direction[Direction["Up"] = 0] = "Up";  
    Direction[Direction["Down"] = 1] = "Down";  
    Direction[Direction["Left"] = 2] = "Left";  
    Direction[Direction["Right"] = 3] = "Right";  
})(Direction || (Direction = {}));
```

```
1 enum Direction {  
2     Up = "Up",  
3     Down = "Down",  
4     Left = "Left",  
5     Right = "Right"  
6 }  
  
var Direction;  
(function (Direction) {  
    Direction["Up"] = "Up";  
    Direction["Down"] = "Down";  
    Direction["Left"] = "Left";  
    Direction["Right"] = "Right";  
})(Direction || (Direction = {}));
```

# TypeScript – Classes & interfaces

```
interface ClockInterface {  
    currentTime: Date;  
    setTime(d: Date);  
}  
  
class Clock implements ClockInterface {  
    currentTime: Date;  
    setTime(d: Date) {  
        this.currentTime = d;  
    }  
    constructor(h: number, m: number) { }  
}
```

# TypeScript – Constructor assignment

```
class TestClass {  
    private name: string;  
  
    constructor(name: string) {  
        this.name = name;  
    }  
}
```

```
class TestClass {  
    constructor(private name: string) {}  
}
```

# TypeScript – Optional properties & modifiers

de

```
interface SquareConfig {  
    color?: string;  
    width?: number;  
}
```

```
class Animal {  
    private name: string;  
    constructor(theName: string) { this.name = theName; }  
}
```

```
class Person {  
    protected name: string;  
    constructor(name: string) { this.name = name; }  
}
```

# Javascript tips: var vs let

- Var is global scoped
- Let is block scoped (const as well)
- Let is recommended because of predictability



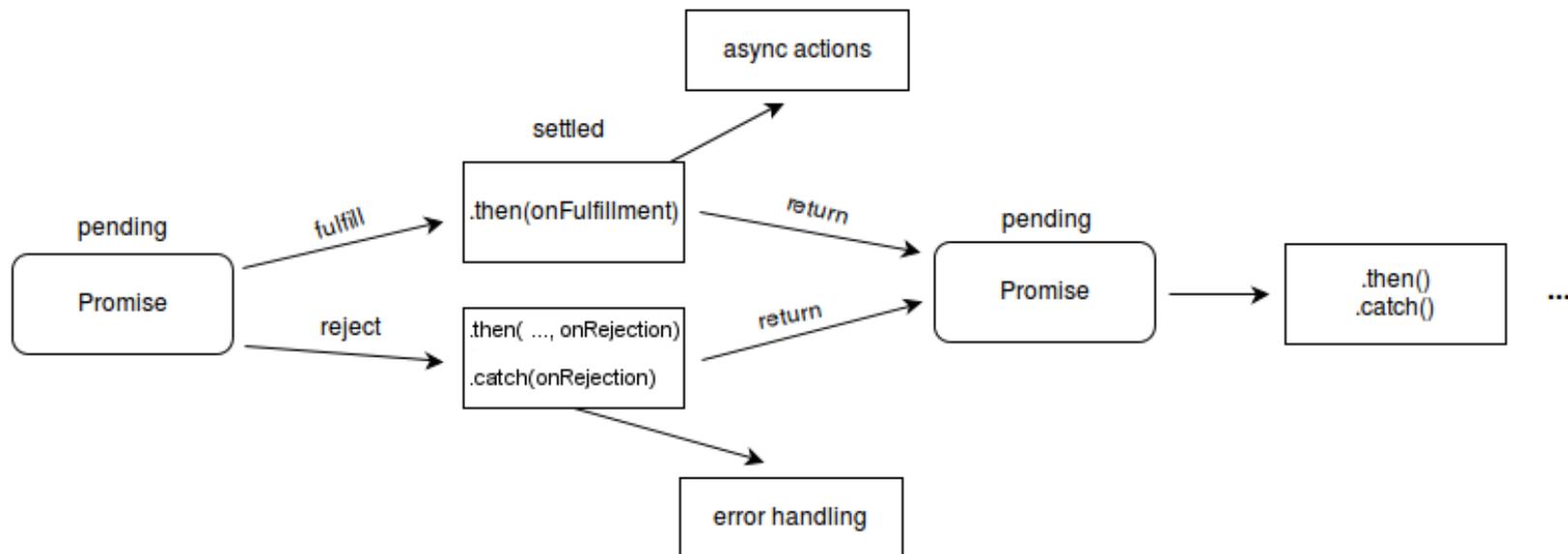
# Javascript tips

- Optional chaining
  - No need to check if a property is null/undefined
  - `let y = something?.value`
- Double pipe operator `||`
  - `let x = Something.value || "somethingelse"`
- Nullish coalescing operator `??`
  - *Nullish* want geldt voor zowel `null` as `undefined`



# Javascript tips: promises

- An asynchronous method which supplies a value at some point in the future
- await





# Exercises

## SETUP

```
git clone https://github.com/AE-nv/Front-end-engineering-for-the-web---basics
```

INSPECT THE CODE (together) branch js-ts

## RUN

```
node src/Laebrary.js
```

```
npm i -g typescript
```

Needed to compile TS files via commandline

# Exercises

- Implement the Typescript interface and class for Book
  - ISBN is optional
- Rename Laebrary.js to Laebrary.ts and add specific TS behaviour to it
  - Explicitly declare class properties books, users and reservations
    - Make them typed and use correct modifiers
  - Add modifiers and return types to all class functions
  - Use constructors to create books, users and reservations in the create\* functions



Extract data generation logic into separate class

# Exercises

COMPILE AND RUN

```
tsc
```

```
node dist/Laebrary.js
```

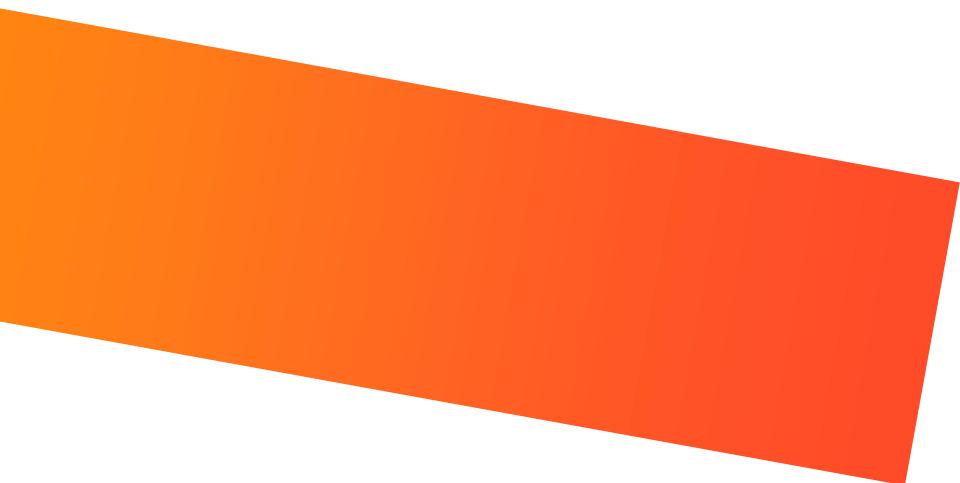
# Exercises - Solution

```
git checkout js-ts-solution
```

```
tsc
```

```
node dist
```

# Dependency management





ES 2015's import and export provide modularity to your codebase

But what about external dependencies?



# Package managers

Manage project dependencies for you across different development devices

- Weapon of Choice:



- Shipped with NodeJs
- Versioned independently of NodeJs
- Scans packages for security vulnerabilities

# Package management - NPM

## ○ Package.json

- Describes your package

```
"name": "laebrary",
"version": "1.0.0",
"description": "Laebrary",
"main": "index.js",
"private": true,
"repository": {
  "type": "git",
  "url": "git+https://github.com/AE-nv/laebrary.git"
},
"author": "AE nv",
"license": "ISC",
"bugs": {
  "url": "https://github.com/AE-nv/laebrary/issues"
},
"homepage": "https://github.com/AE-nv/laebrary#readme"
```

# Package management - NPM

## ○ Package.json

- Lists your dependencies

```
"dependencies": {  
  "body-parser": "^1.17.1",  
  "compression": "^1.6.2",  
  "express": "~4.15.2",  
  "lodash": "^4.17.4",  
  "log4js": "~1.1.1",  
  "mongodb": "^2.2.26",  
  "mongoose": "^4.10.4",  
  "mongoose-unique-validator": "~1.0.5",  
  "q": "^1.5.0"  
}
```

```
"devDependencies": {  
  "@types/body-parser": "^1.16.3",  
  "@types/bson": "^1.0.3",  
  "@types/compression": "^0.0.33",  
  "@types/express": "^4.0.35",  
  "@types/lodash": "^4.14.63",  
  "@types/log4js": "^0.0.32",  
  "@types/mongodb": "^2.2.2",  
  "@types/mongoose": "^4.7.13",  
  "@types/node": "^7.0.13",  
  "@types/q": "^1.0.0",  
  "gulp": "^3.9.1",  
  "gulp-nsp": "^2.4.2",  
  "gulp-typescript": "^3.1.6",  
  "typescript": "^2.2.2"  
}
```

# Package management - NPM

- **Package.json**
  - Provides scripts

```
"scripts": {  
  "start": "node index.js",  
  "build": "tsc index.ts"  
}
```

```
npm run <script-name>
```

# Package management - NPM

- Package-lock.json
  - Describes the exact dependency tree that was generated

```
{  
  "name": "laebrary",  
  "version": "1.0.0",  
  "lockfileVersion": 1,  
  "requires": true,  
  "dependencies": {  
    "@types/body-parser": {  
      "version": "1.17.0",  
      "resolved": "https://registry.npmjs.org/@types/body-parser/-/body-parser-1.17.0.tgz",  
      "integrity": "sha512-a2+YeUjPkztKJu5aIF2yArYFQQp8d51wZ7DavSHjFuY1mqVgidGyzEQ41JIVNy82fxj8yPgy2vJmfIywqESW6w==",  
      "dev": true,  
      "requires": {  
        "@types/connect": "3.4.32",  
        "@types/node": "7.0.62"  
      }  
    },  
    "@types/bson": {  
      "version": "1.0.8",  
      "resolved": "https://registry.npmjs.org/@types/bson/-/bson-1.0.8.tgz",  
      "integrity": "sha512-PMa4nkRhLagwsXQDDTzGbTyCIpej0ERznyAP9fyuGnlsmUbcc4Y25mdqjibYjkOPNyK/BWWUKneruaKHcs3Q8g==",  
      "dev": true  
    }  
  }  
}
```

# Package management - NPM

CLI

```
npm init
```

Generates a **package.json** file based on user input

# Package management - NPM

CLI

```
npm install <package-name>
```

- **Installs** latest version of an NPM package **into node\_modules**
- **Adds** the package **to dependencies** **in package.json**
- **Creates or updates** the **package-lock.json** file

Installs **specific version**

```
npm i <package-name>@<version>
```

Installs and **adds to devDependencies**

```
npm i <package-name> -D
```

Installs the package **globally**

```
npm i <package-name> -g
```

# Package management - NPM

CLI

- Packages can be found on <https://npmjs.com>
- Some packages are scoped

```
@somescope/somepackagename
```

# Package management - NPM

CLI

- Hooks are available
  - **Updates the version** in package.json and package-lock.json
  - **Creates a git tag**

```
npm version [<newversion> | major | minor | patch | premajor | preminor | prepatch | prerelease | from-git]
```

```
"scripts": {  
  "preversion": "npm test",  
  "version": "npm run build && git add -A dist",  
  "postversion": "git push && git push --tags && rm -rf build/temp"  
}
```

# Package management - NPM

CLI

- .npmignore can be specified

**Uploads** the contents of the **package** to **npm repo**

```
npm publish
```

**Creates a symlink** in the global folder  
to the package where this command was executed

```
npm link
```

Creates a **symbolic link from** globally installed  
**package-name to node\_modules**

```
npm link <package-name>
```

# Package management - NPM

CLI

```
# Create global symbolic aka symlink to dependency
cd ~/projects/some-dep
npm link # Step 1.

# Tell the application to use the symlink
cd ~/projects/my-app
npm link some-dep # Step 2.
```

# Package management - NPM

Consider using NVM (node version manager)

- **Consider using NVM (node version manager)**



- If you have multiple applications each using a different version of NodeJS / NPM and you want to easily switch between NodeJS versions
- If you want to easily upgrade to another version of NodeJS without running a new installed, but CLI based

# Package management - NPM

- Manage dependencies + versions easily

```
npm i npm-check-updates -g
```

```
$ ncu

express      4.12.x → 4.13.x
multer       ^0.1.8  →  ^1.0.1
react-bootstrap ^0.22.6 →  ^0.24.0
react-a11y   ^0.1.1  →  ^0.2.6
webpack      ~1.9.10 →  ~1.10.5
```

Run with -u to upgrade your package.json

```
$ ncu -u

express      4.12.x → 4.13.x
package.json upgraded
```

<https://www.npmjs.com/package/npm-check-updates>

## SETUP

```
git checkout dependency-management
```

## RUN

```
tsc
```

```
node dist
```

Check that the dates are in timezone GMT

1. Set up a new npm package called @ae/laebrary
2. Install your first dev dependency
  - **rimraf**: reproduces rm -rf behaviour on windows
3. Create your first NPM script
  - **start** – run the compiled index.js file with node
  - Foresee a pre and a **post hook** that do the compilation + cleanup
    - Prestart => tsc
    - Start => node dist
    - Poststart => rimraf dist

4. Install and use date-fns and date-fns-tz to create and format Dates
  - Create dates in local timezone
  - Display reservation start dates formatted as DD-MM-YYYY

```
import format from "date-fns/format";
```

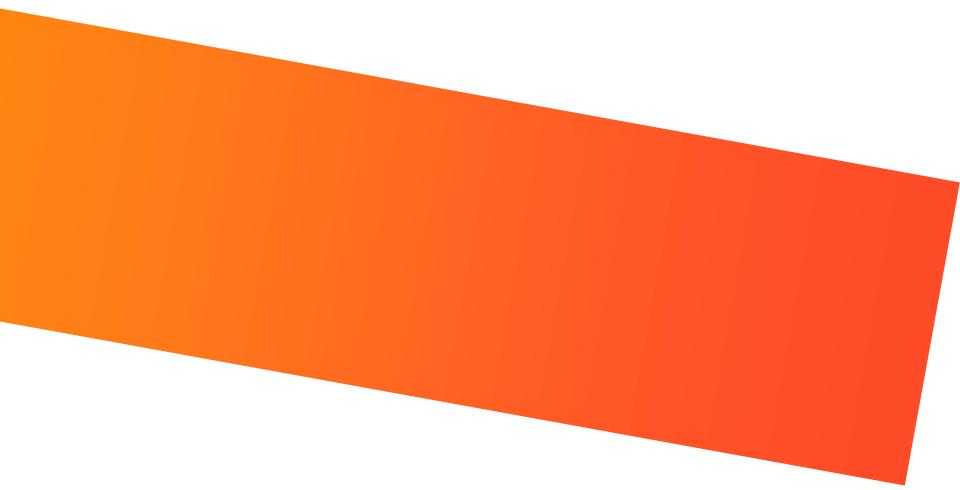
# Exercises

- SOLUTION

```
git checkout dependency-management-solution
```

```
npm start
```

# Component design





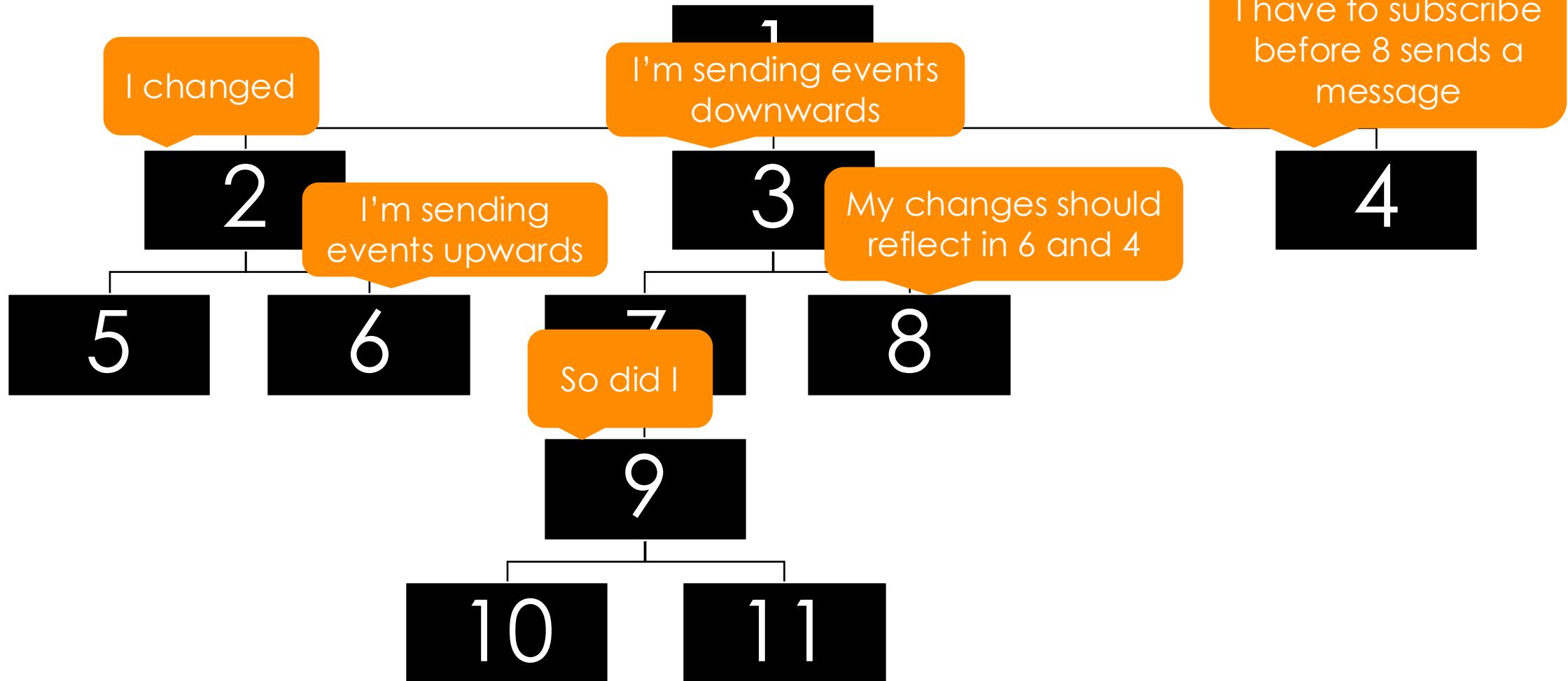
Component design is one of the most important aspects of a successful SPA



# Component design & atomic design?

- **Atomic design** = a method of breaking a visual design down into its smallest parts  
→ Parts are then **standardized** and built back up into increasingly larger, reusable objects.
- Parts:
  1. Atoms
  2. Molecules
  3. Organisms
  4. Templates
  5. Pages
- Atomic design is also called **component-based design**, where the various reusable objects are called **components**

# Components

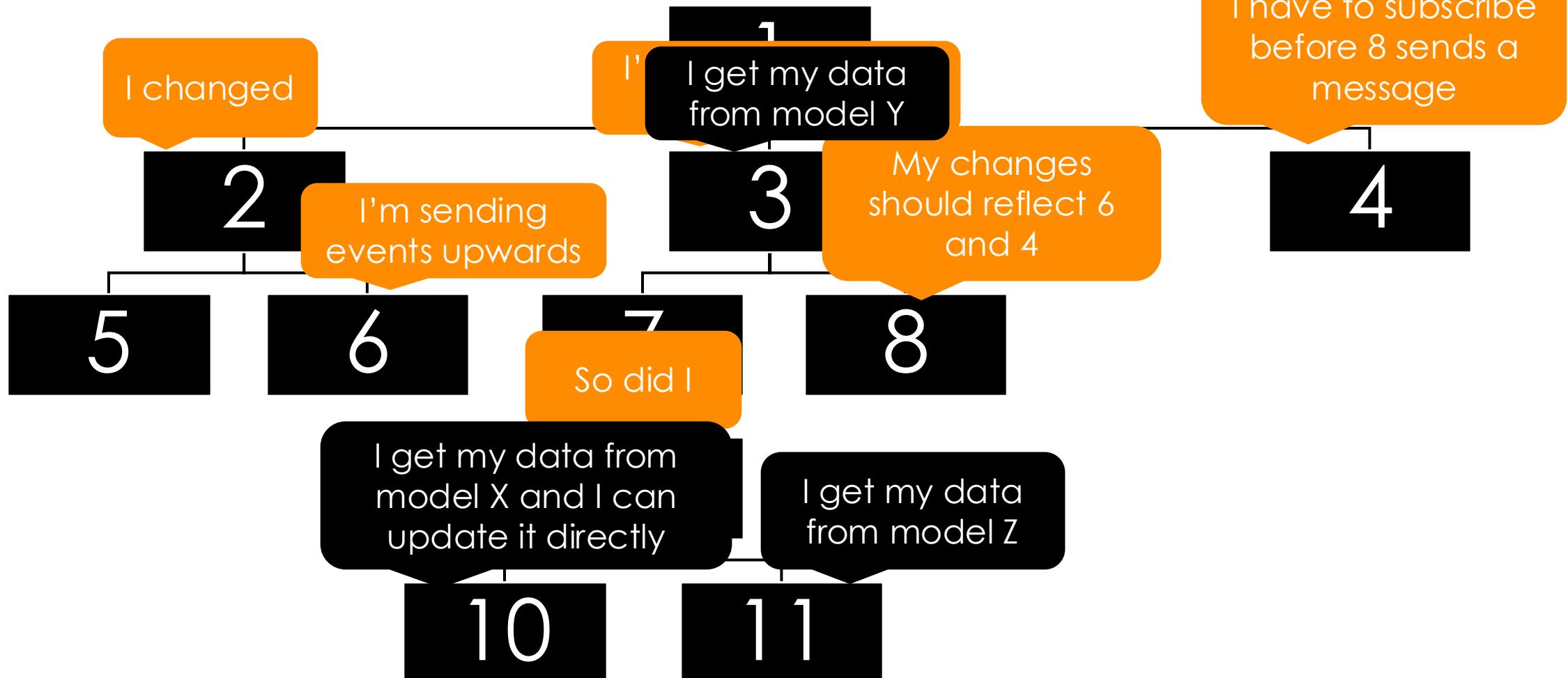


# Benefits of component design

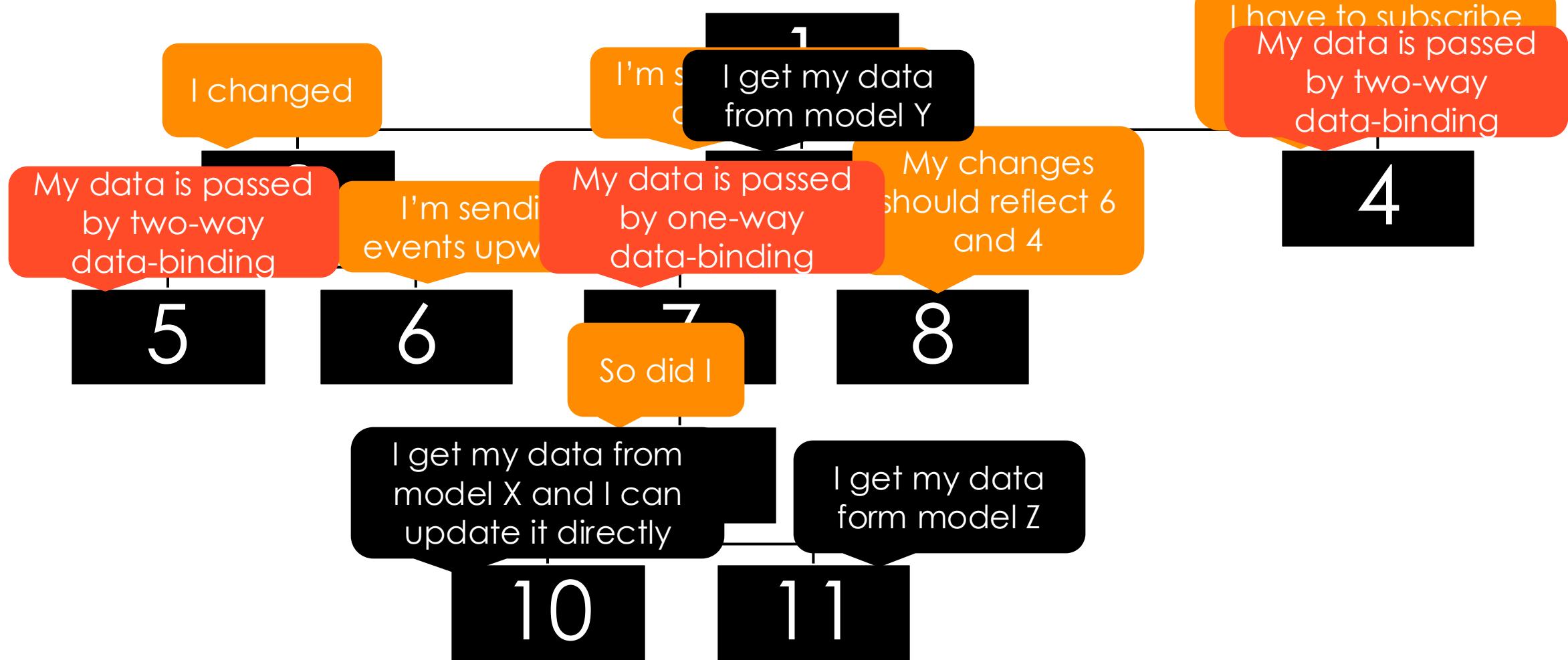
- Decrease complexity
- Increase maintainability
- Single responsibility (SoC)
- Reusability
- Testability



# Components



# Components



# Component design guidelines

- Keep them **small**

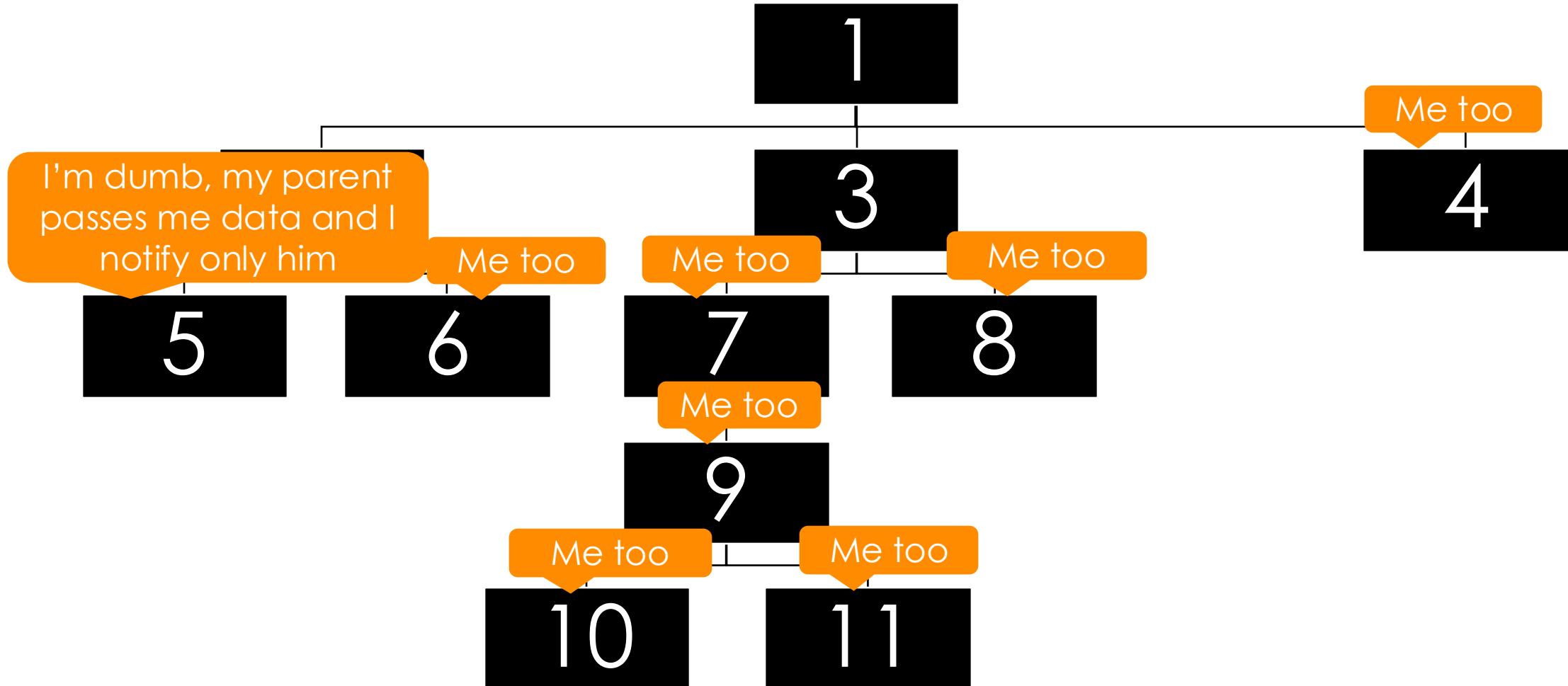


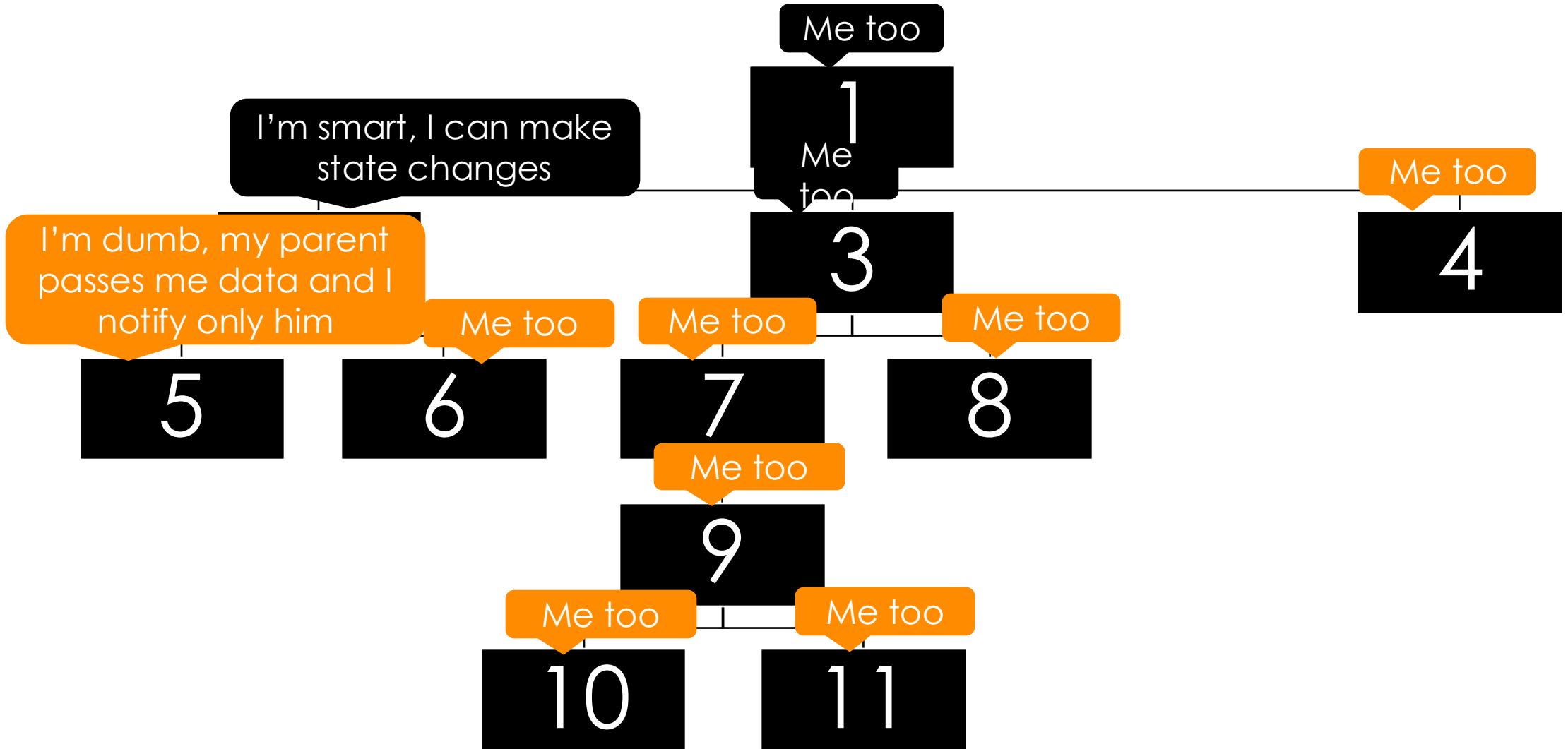
# Component design guidelines

- Keep them **dumb**



# Components





# Smart vs dumb components

## Smart

- Knows about the application
- Does data fetching (through services)
- Not reusable
- State management
- Has dependencies

## Dumb

- Doesn't know about the application
- Doesn't do data fetching
- Reusable
- Only communicates with direct parent
- Mostly no dependencies

Stays

Flights

Flight + Hotel

Car rentals

Attractions

Airport taxis

Leuven

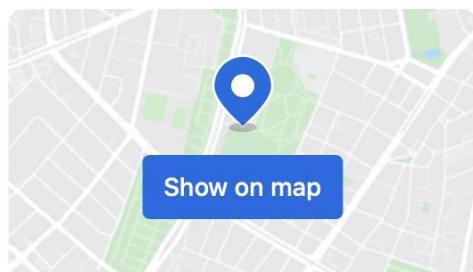


Thu 21 Dec — Fri 22 Dec

2 adults · 0 children · 1 room



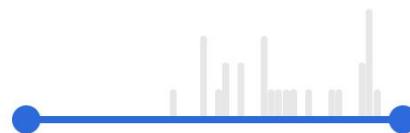
Search

[Home](#) > [Belgium](#) > [Flemish Brabant](#) > [Leuven](#) > Search results

## Filter by:

## Your budget (per night)

€ 60 - € 200+



## Popular filters

 Breakfast included

19

 Hotels

21

 4 stars

8

 Superb: 9+

2

Based on guest reviews

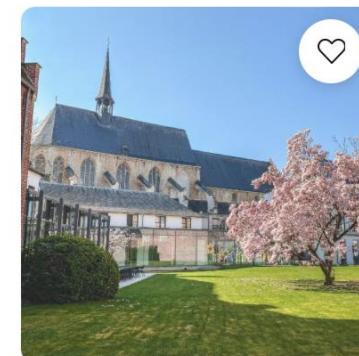
 Bed and breakfasts

3

## Leuven: 31 properties found

↓↑ Sort by: Our top picks ◇

Commission paid on bookings and other factors may affect property rankings. Learn about these ranking parameters [Find out more](#)

**Martin's Klooster** ★★★★ [Leuven](#) · [Show on map](#) · 300 m from centre

Travel Sustainable Level 3+

**Standard Double Room**

1 large double bed

**Free cancellation** **No prepayment needed** – pay at the property**Very good** 8.4  
2,423 reviews**Location 9.4**

1 night, 2 adults

€ 136

Includes taxes and charges

[See availability >](#)**pentahotel Leuven** ★★★★ [Leuven](#) · [Show on map](#) · 250 m from centre

Travel Sustainable Level 3+

**penta Standard Double or Twin Room**

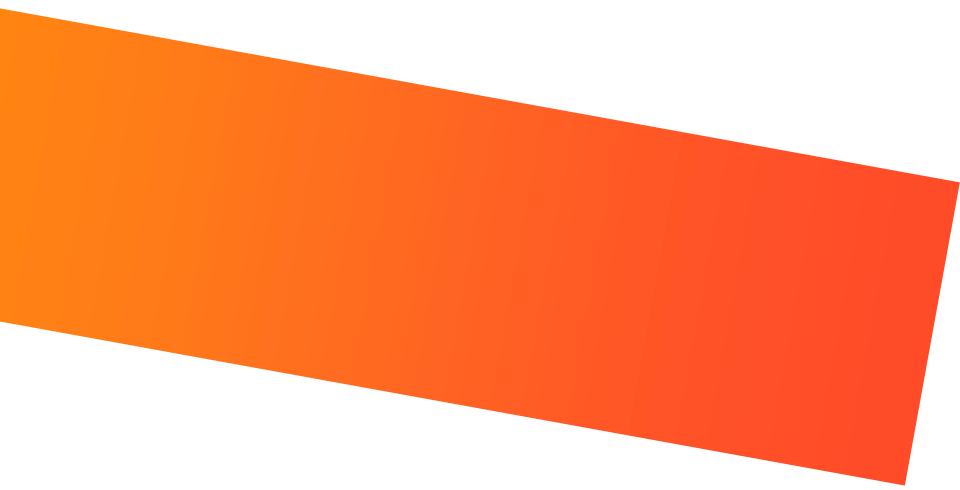
Beds: 1 double or 2 singles

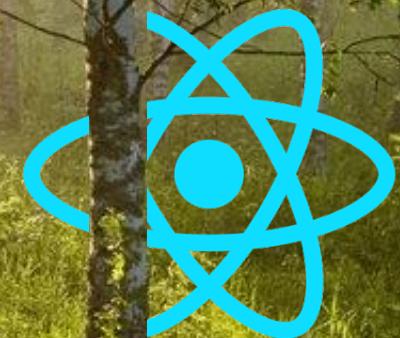
**Very good** 8.0  
3,441 reviews**Location 9.4**

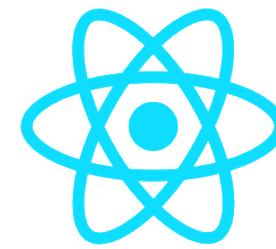
1 night, 2 adults

€ 138

# **Modern Web Frameworks**









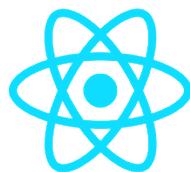
## *A Superheroic JavaScript MVW Framework*

Google, ° Sept 2016 (AngularJS: Oct 2010)



## *An Intuitive, Fast and Composable MVVM for building interactive interfaces*

Evan You, ° Feb 2014



## *A JavaScript library for building user interfaces (V)*

Facebook, ° Mar 2013



## *A radically different JS framework (MVW) that aims to improve developer experience and performance of Web apps*

Rich Harris, ° June 2023

# Frameworks vs. libraries



**Framework** – strong opinions on how to structure Application and a lot of functionality out of the box

*Complete solution, ready-to-go (if everything goes right)*



# Frameworks vs. libraries



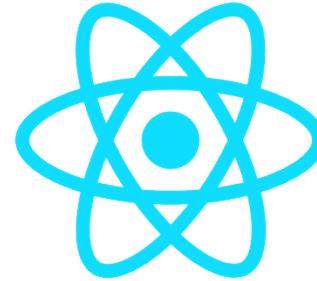
de



**Framework** – “Vue.js does not make large assumptions about much of anything”

“Vue.js comes with the perfect balance of what it will do for you and what you need to do yourself.”

# Frameworks vs. libraries



**Library** – Universally flexible **view** oriented library (“UI as a function”)

*With great flexibility comes great responsibility*



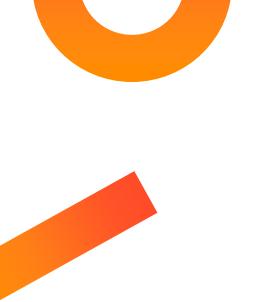
# Frameworks vs. libraries



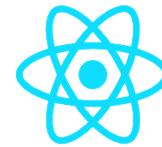
**Framework and library** –Combines the greatest elements of both categories into one tool

*A hybrid providing the flexibility and power of a library while offering the structure of a framework*





de



The  
Guardian

The Adobe logo, featuring a red 'A' inside a black hexagon with the word 'Adobe' in a bold black sans-serif font.

The Airbnb logo, a red stylized 'a' inside a white hexagon with the word 'airbnb' in a red sans-serif font.



euronews.



The Brave logo, a red lion's head inside a shield with the word 'brave' in a black sans-serif font below it.

Forbes

The Alibaba.com logo, an orange stylized cloud icon above the text 'Alibaba.com'.



The AIPRM logo, a black geometric icon resembling a stylized 'A' or a cube with four vertical bars, with the acronym 'AIPRM' in a black sans-serif font below it.



TESLA

The GitLab logo, a red and orange geometric icon next to the word 'GitLab' in a grey sans-serif font.

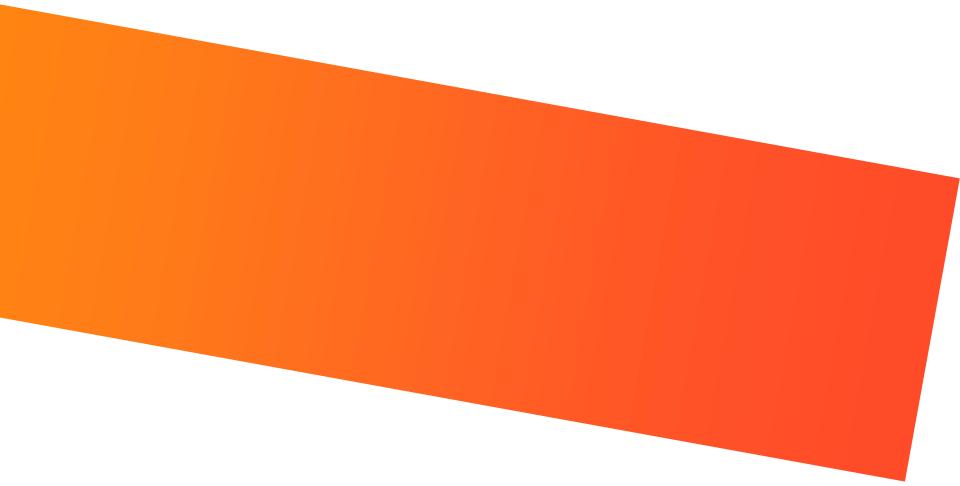
The Uber logo, a black circle with a white 'U' inside, with the word 'UBER' in a black sans-serif font below it.

PayPal



NETFLIX

# **UI & UX**



“

“I majored in engineering—it’s almost a badge of pride to build something that looks awful.”

”

- Erik D. Kennedy

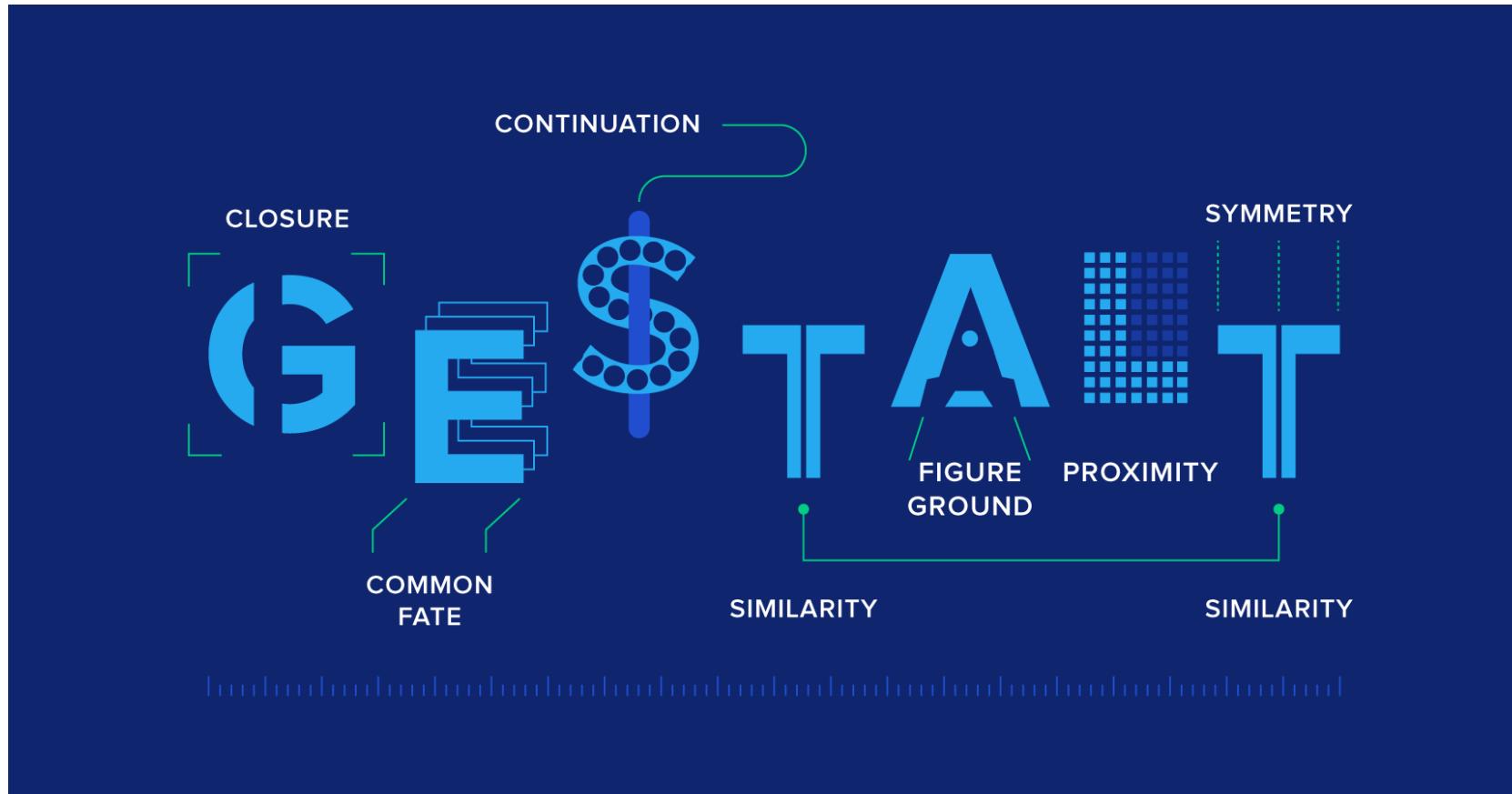


# UX Design – Create Desirable Interaction

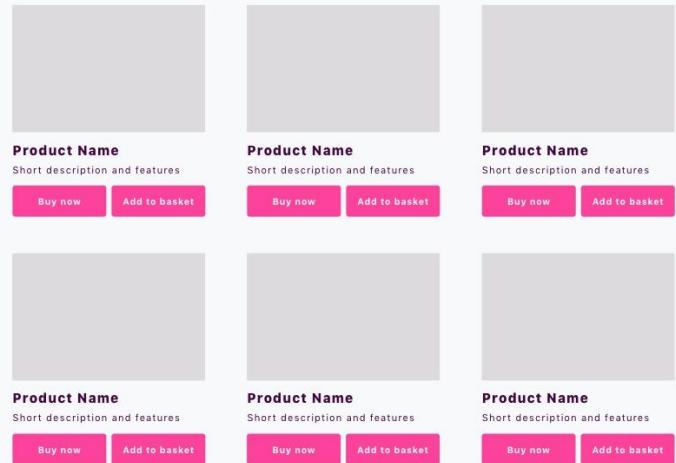
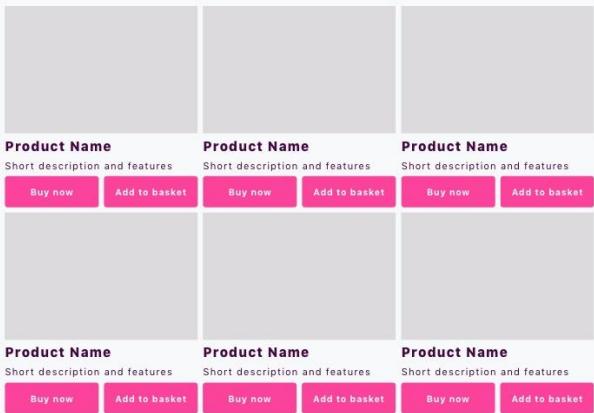
Do not create a perfect technical solution that the user hates...



Keep in mind a few rules of thumb!



# Proximity



Gestalt Principles

<https://uxmisfit.com>

# Similarity

**Product Name**  
Description and features  
[Edit](#) [Remove](#)



Gestalt Principles

<https://uxmisfit.com>

# Continuation

Home Services Contact



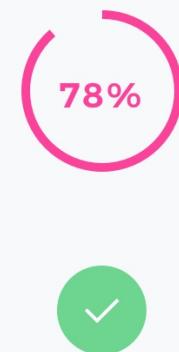
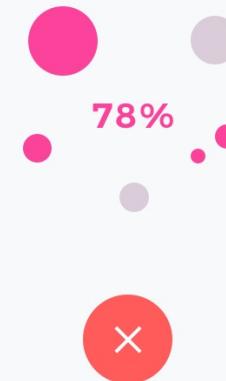
Gestalt Principles

Home Services Contact



<https://uxmisfit.com>

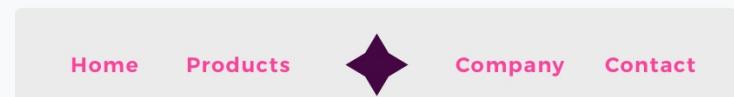
# Closure



Gestalt Principles

<https://uxmisfit.com>

# Symmetry



Gestalt Principles

<https://uxmisfit.com>

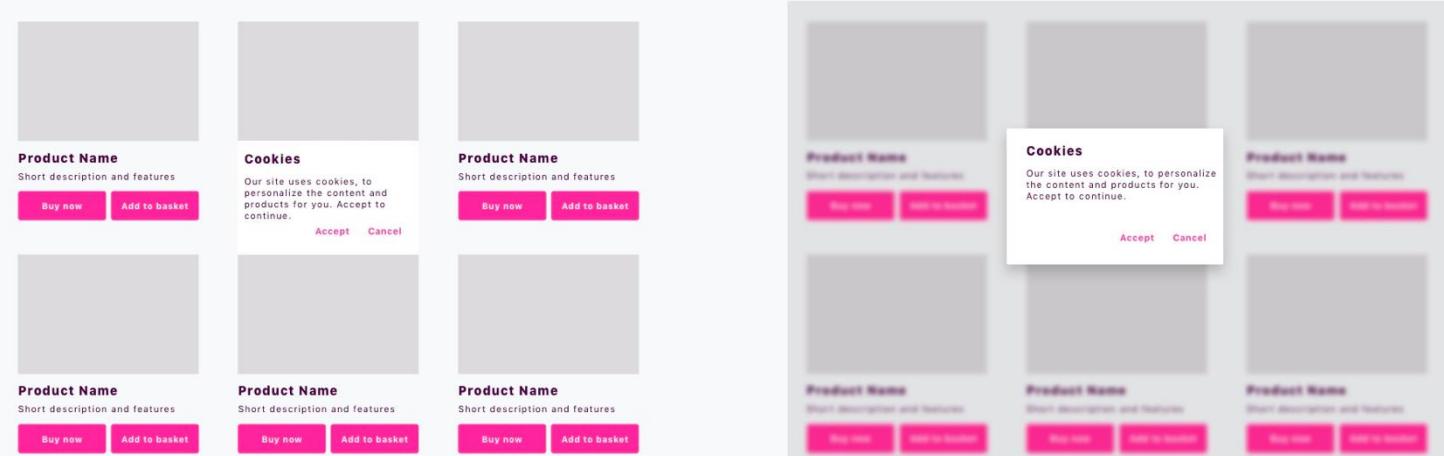
# Common Fate



Gestalt Principles

<https://uxmisfit.com>

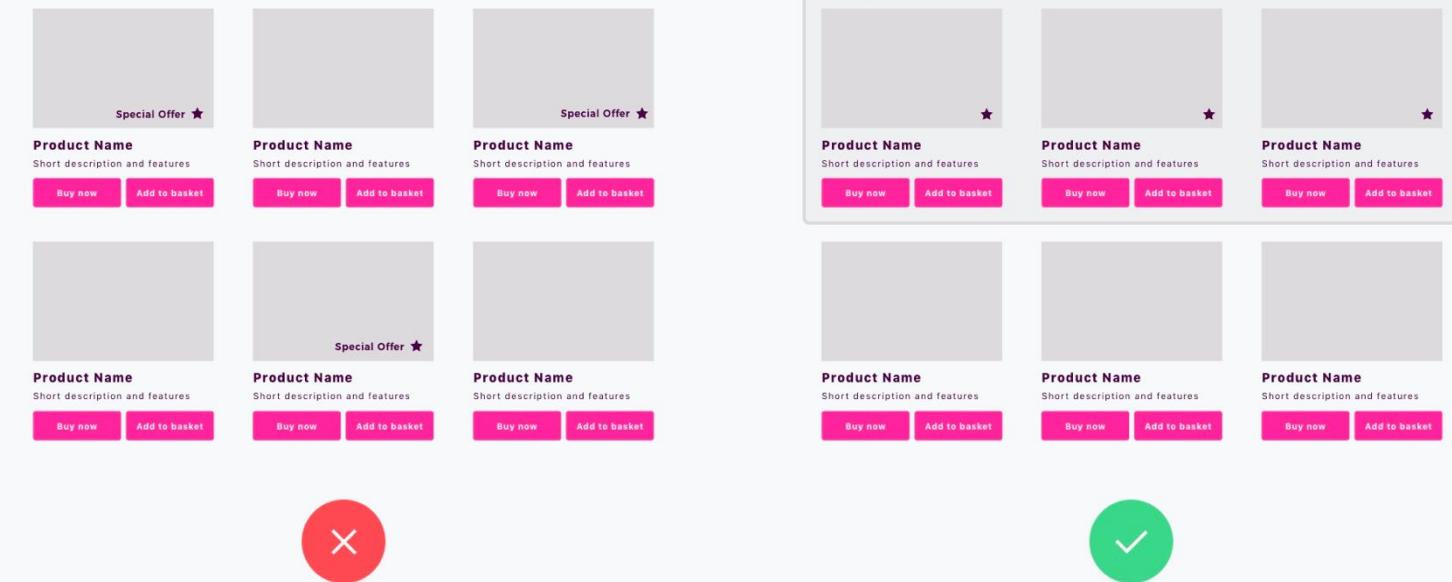
# Figure - Ground



Gestalt Principles

<https://uxmisfit.com>

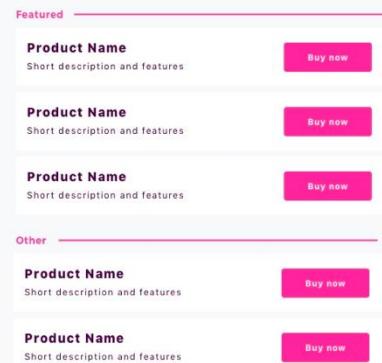
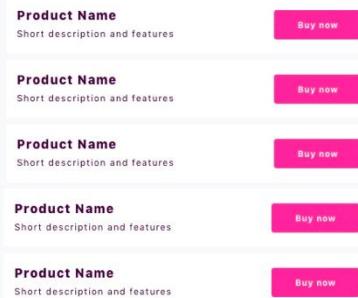
# Common Region



Gestalt Principles

<https://uxmisfit.com>

# Periodicity



Gestalt Principles

<https://uxmisfit.com>

# THE 10 HEURISTICS TO MEASURE



## Visibility

Show system status, tell what's happening. Give feedback to the user about what is going on (spinners etc.).



## Mapping

Use familiar language and provide metaphors that match the real world, and do so in a natural and logical order.



## User Control & Freedom

Support undo, redo and exit points to avoid mistakes. Help the user feel in charge. Avoid data-loss at all times.



## Consistency and standards

Use same UI patterns and language. Follow platform conventions, consistent words, actions.



## Error prevention

Help users avoid making mistakes. Prevent problems from occurring. Eliminate error-prone conditions or check for them before users commit to the action.



## Recognition rather than recall

Make information easier to discover



## Flexibility and ease of use

Make advanced tasks fluid and efficient. Advanced users like to use shortcuts for frequent actions.



## Aesthetic and minimalist design

Provide only necessary information in an elegant way. Simplicity is key.



## Help users recognize, diagnose and recover from errors

Errors are easily made. Be forgiving. If you have to show an error message, do it in plain language to indicate the problem and suggest solutions.



## Help and documentation

Use proactive and in-context hints to guide the user, avoid steep learning curves.



## Visibility

Show system status, tell what's happening. Give feedback to the user about what is going on (spinners etc.).

de

cheapTickets.be

Brussel (Alle luchthavens) → Tokyo (Tokio) (Alle Luchthavens)  
wo 08 mei, 2 passagiers, Economy

Jouw zoekopdrachten 4 Mijn account Service & Contact

Wij vergelijken nu alle airlines

**Veilig, vertrouwd én voordelig**

- ✓ Nummer 1 Travel Website van België
- ✓ Binnen 5 minuten je tickets per email
- ✓ Nederlandstalige klantenservice

TRUSTPILOT  
Goed: 8.3   
[1615 klantbeoordelingen op Trustpilot.com](#)

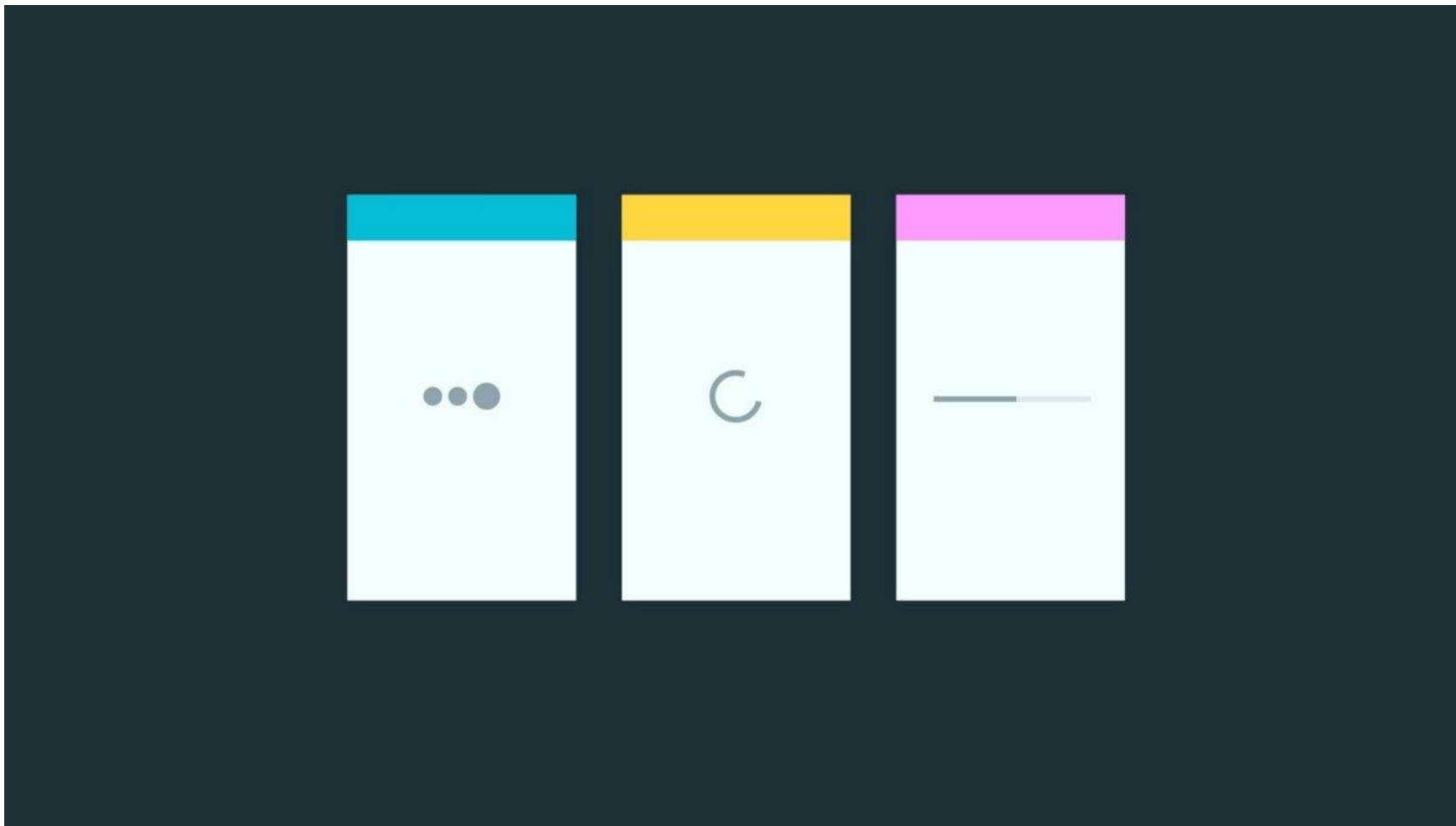
Waiting for searchproductsapi.travix.com...

Feedback



## Visibility

Show system status, tell what's happening. Give feedback to the user about what is going on (spinners etc.).





## Visibility

Show system status, tell what's happening. Give feedback to the user about what is going on (spinners etc.).

•••• AT&T ⌘ 6:15 PM ⌂ ↗ 68% 🔋

≡ My Drive ⚡

Uploads

119B99B6-A...CBD632.PNG X

Folders ↑ NAME

Final\_Home\_Album ···  
Modified May 28, 2016

Home\_Album ···  
Modified May 26, 2016

Marriage ···  
Modified Jan 14, 2016

PGPEx\_CVs ···  
★ Modified May 22, 2013



1 of 1 uploading...



## Mapping

Use familiar language and provide metaphors that match the real world, and do so in a natural and logical order.

de

## Bekijk al onze verzekeringen



[Voertuigen](#) | [Woning](#) | [Familie & gezondheid](#) | [Reizen & vrije tijd](#) | [Op het werk](#) | [Sparen & beleggen](#)



### Auto

- ✓ Omnimium zonder franchise
- ✓ 24u/24 Mobility bijstand (gratis)

[Info & Offerte >](#)



### Ethias Young Drivers

- ✓ Autoverzekering speciaal voor de jongeren
- ✓ Young Drivers Bonus

[Info & Offerte >](#)



### Assistance pechverhelping

- ✓ 24 uur per dag bijstand, vanaf uw woning
- ✓ Dekt tot drie voertuigen

[Info & Offerte >](#)



## Mapping

Use familiar language and provide metaphors that match the real world, and do so in a natural and logical order.





## User Control & Freedom

Support undo, redo and exit points to avoid mistakes. Help the user feel in charge. Avoid data-loss at all times.

A screenshot of a 'New Message' window from an email application. The window has a dark header bar with the title 'New Message' and standard window control buttons. Below the header, there are input fields for 'Recipients' and 'Subject'. A modal dialog box is centered over the message body, titled 'Attaching file'. It contains the message: 'Your file is larger than 25 MB. It will be sent as a [Google Drive](#) link.' Below this message, there is a preview of a file named 'really\_big\_attachment.mp4' which is 1.9G in size. A progress bar indicates the file is being processed. At the bottom right of the dialog, there is a 'Cancel' button.



### User Control & Freedom

Support undo, redo and exit points to avoid mistakes. Help the user feel in charge. Avoid data-loss at all times.



## Oprah

@Oprah

*Live Your Best Life*

Chicago, IL - <http://www.oprah.com>

Followed by [The Cut](#), [InStyle](#)

2,385

TWEETS

44

FOLLOWING

Follow



Oprah Winfrey

@MorningWXGuy  
treadmill watching



## Oprah

@Oprah

*Live Your Best Life*

Chicago, IL - <http://www.oprah.com>

Followed by [The Cut](#), [InStyle](#)

2,385

TWEETS

44

FOLLOWING

Following



Oprah Winfrey

@MorningWXGuy  
treadmill watching



## Oprah Winfrey

@Oprah

*Live Your Best Life*

Chicago, IL - <http://www.oprah.com>

Followed by [The Cut](#), [InStyle](#) and [Noah Everett](#).

2,385

TWEETS

44

FOLLOWING

10,473,141

FOLLOWERS

Unfollow



Oprah Winfrey

@Oprah  
@MorningWXGuy @cbsthismorning for s  
treadmill watching@CBSThisMorning w



Oprah Winfrey

In a commercial  
yourself? Readin



Oprah Winfrey

In a commercial  
yourself? Readin



Oprah Winfrey

@Oprah  
In a commercial  
yourself? Reading your answers on air #



## Consistency and standards

Use same UI patterns and language. Follow platform conventions, consistent words, actions.

de



Short fuzzy coat with lapel collar

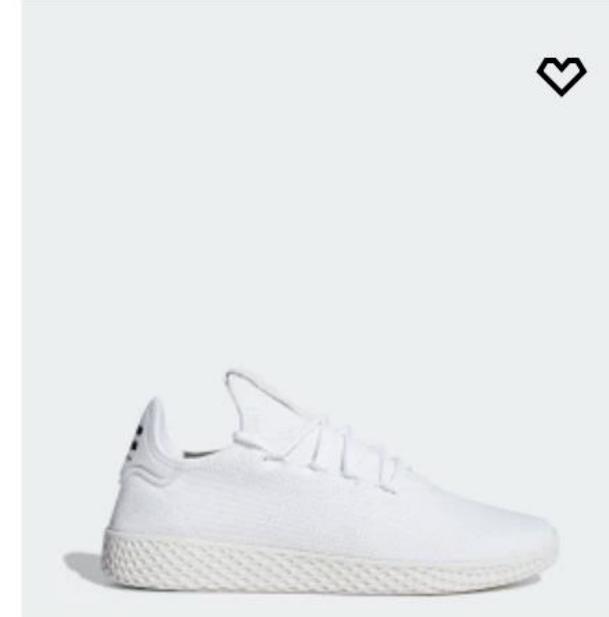
49,99 €



NIEUW

**Eastpak**  
Heuptas - night navy

€ 25,00



Originals

PHARRELL WILLIAMS TENNIS HU  
SH...

€ 99,95



Grote oorbellen  
€ 2,99





### Error prevention

Help users avoid making mistakes. Prevent problems from occurring. Eliminate error-prone conditions or check for them before users commit to the action.

mail.google.com says

It seems like you have forgotten to attach a file.

You wrote "see attachment" in your message, but there are no files attached. Send anyway?

Cancel

OK



### Error prevention

Help users avoid making mistakes. Prevent problems from occurring. Eliminate error-prone conditions or check for them before users commit to the action.

Compose new Tweet X

This is an extraordinarily long Tweet. It's very useful that Twitter will warn you before you reach the character limit for a Tweet, enab

Add photo Location disabled 1  Tweet

Compose new Tweet X

This circle icon might be the most annoying part of Twitter switching to 280  

    Tweet



Recognition rather than recall  
Make information easier to discover



JIRA Dashboards Projects Issues Boards Create Search ?

CURO\_Transacties Onder... CURO\_Transacties Ond...

Backlog Active sprints Releases Reports Issues Components

## All reports Switch report

### Agile

**Burndown Chart**  
Track the total work remaining and project the likelihood of achieving the sprint goal. This helps your team manage its progress and respond accordingly.

**Sprint Report**  
Understand the work completed or pushed back to the backlog in each sprint. This helps you determine if your team is overcommitting or if there is excessive scope creep.

**Velocity Chart**  
Track the amount of work completed from sprint to sprint. This helps you determine your team's velocity and estimate the work your team can realistically achieve in future sprints.

**Cumulative Flow Diagram**  
Shows the statuses of issues over time. This helps you identify potential bottlenecks that need to be investigated.

**Version Report**  
Track the projected release date for a version. This helps you monitor whether the version will release on time, so you can take action if work is falling behind.

**Epic Report**  
Understand the progress towards completing an epic over time. This helps you manage your team's progress by tracking the remaining incomplete/unestimated work.

**Control Chart**  
Shows the cycle time for your product, version or sprint. This helps you identify whether data from the current process can be used to determine future performance.

**Epic Burndown**  
Track the projected number of sprints required to complete the epic (optimized for Scrum). This helps you monitor whether the epic will release on time, so you can take action if work is falling behind.

**Release Burndown**  
Track the projected release date for a version (optimized for Scrum). This helps you monitor whether the version will release on time, so you can take action if work is falling behind.



Recognition rather than recall  
Make information easier to discover

de

## Your Recently Viewed Items and Featured Recommendations

Inspired by your browsing history

Page 2 of 9 | Start over



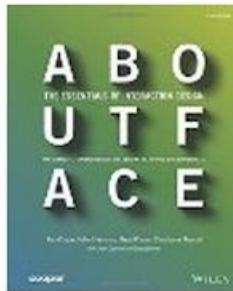
Growth Hacker Marketing:  
A Primer on the Future...

› Ryan Holiday

★★★★★ 37

Paperback

£8.99



About Face: The Essentials  
of...

› Alan Cooper

★★★★★ 3

Paperback

£23.11



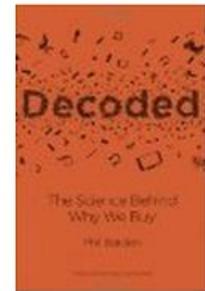
Venture Deals: Be Smarter  
Than Your...

› Dick Costolo

★★★★★ 4

Hardcover

£30.59



Decoded: The Science  
Behind Why We Buy

› Phil Barden

★★★★★ 43

Hardcover

£16.58



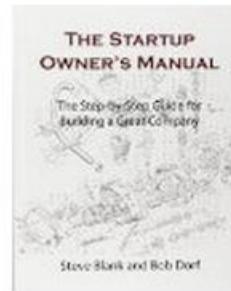
The Ten Principles  
Behind Great Customer...

› Matt Watkinson

★★★★★ 54

Paperback

£14.09



The Startup Owner's  
Manual: The...

› Steve Blank

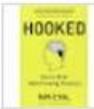
★★★★★ 22

Hardcover

£24.05



You  
viewed



› View or edit  
your browsing  
history



## Flexibility and ease of use

Make advanced tasks fluid and efficient. Advanced users like to use shortcuts for frequent actions.



## Adobe XD | KEYBOARD SHORTCUTS (Part - 2)



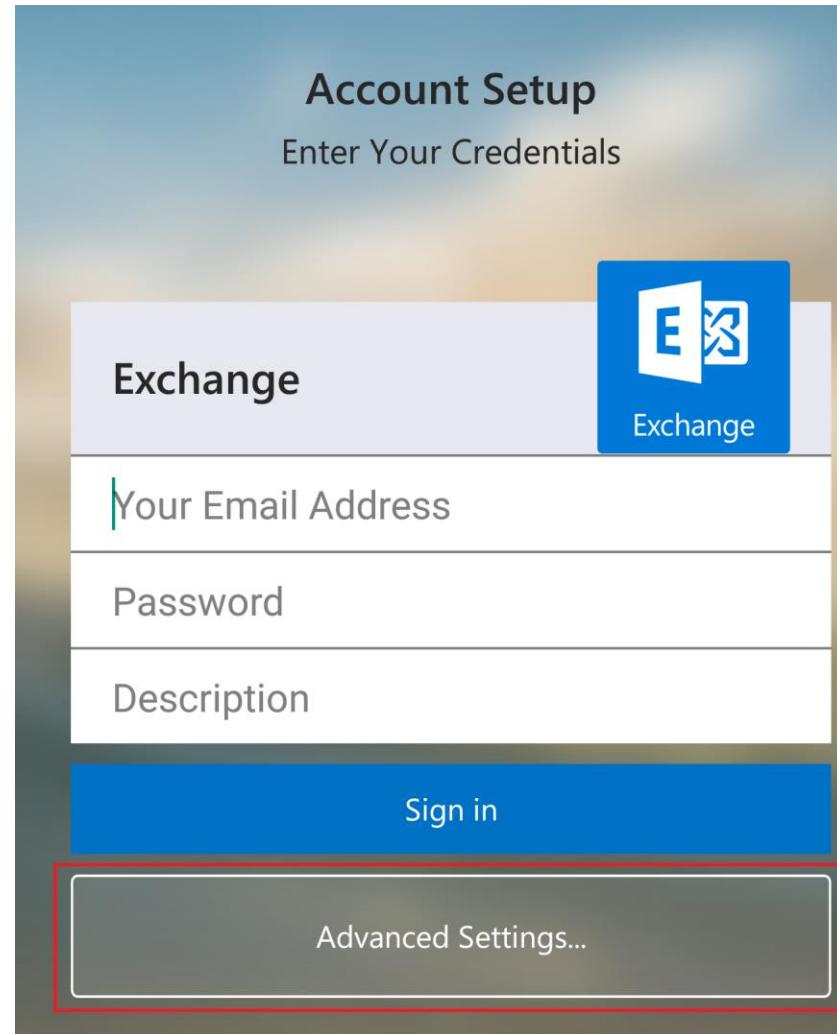
Path/Pen		
Action	Mac	Windows
Open Path/Pen panel	P	P
Convert Point	Double-click	Double-click
Asymmetric Control Point	Alt	Alt
Snap Control Point Angle	Shift	Shift
Snap Anchor Point Angle	Shift	Shift
Add	Cmd + Alt + U	Ctrl + Alt + U
Subtract	Cmd + Alt + S	Ctrl + Alt + S
Intersect	Cmd + Alt + I	Ctrl + Alt + I
Exclude Overlap	Cmd + Alt + X	Ctrl + Alt + X
Convert to Path	Cmd + 8	Ctrl + 8

Operations menu		
Action	Mac	Windows
From Center	Alt	Alt
Copy Paste Object	Alt + Drag	Alt + Drag
Constrain	Shift	Shift
Edit Text	Enter	Enter
Constrain Rotate	Shift (15°)	Shift (15°)
Line Constrain Rotate	Shift (45°)	Shift (45°)
Constrain From Center	Shift + Alt	Shift + Alt
Direct Select	Cmd	Ctrl
Switch between Design and Prototype mode	Cmd + Tab	Ctrl + Tab



### Flexibility and ease of use

Make advanced tasks fluid and efficient. Advanced users like to use shortcuts for frequent actions.





Aesthetic and minimalist design

Provide only necessary information in an elegant way.

Simplicity is key.

# Google

Microphone icon

Google Search

I'm Feeling Lucky

Google offered in: [Nederlands](#) [Français](#) [Deutsch](#)



### Aesthetic and minimalist design

Provide only necessary information in an elegant way.  
Simplicity is key.

The image shows a minimalist web interface for a file sharing service. The background features a repeating pattern of green and blue wavy lines. In the top right corner, there is a navigation bar with three buttons: "Help", "Over ons", and "Plus?". On the left side, there is a large white input field containing a blue button with a plus sign and the text "Voeg bestanden toe". Below this input field, there is a "Bericht" (Report) section and a "Versturen" (Send) button. The overall design is clean and modern, emphasizing simplicity and elegance.



### Help users recognize, diagnose and recover from errors

Errors are easily made. Be forgiving. If you have to show an error message, do it in plain language to indicate the problem and suggest solutions.

I'm gonna have to go ahead and ask you to use a different browser.



Yaaaaah... matthew, what's happening?

We went ahead and stopped supporting Internet Explorer 7 & 8 in the Spiceworks app, so if you could just go ahead and upgrade IE or install Chrome Frame, that would be great...

(Neither take long to install, and both make Spiceworks so much faster!)

[Upgrade Internet Explorer](#) [Install Chrome Frame](#)



## Help users recognize, diagnose and recover from errors

Errors are easily made. Be forgiving. If you have to show an error message, do it in plain language to indicate the problem and suggest solutions.

Vodafone IN 5:46

YouTube

You're offline

The next time you're online, try saving some videos that you can watch without an Internet connection.

RETRY

Home Trending Subscriptions Inbox Library

No connection

< O □



[Sign Up](#) [To download](#)

## Register

[or log in to your account](#)

First Name

Last name

unvalidemailaddress.com

! Please include an '@' in the email address. 'unvalidemailaddress.com' is missing an '@'.

This page is protected with reCAPTCHA and subject to the Google [Privacy Policy](#) and [Terms of Service](#).

I accept the [Dropbox terms](#)

**Register**

[Register with Google](#)



## Help and documentation

Use proactive and in-context hints to guide the user, avoid steep learning curves.



Help Center

[Submit a request](#) [Back to WeTransfer.com](#) [Work with us?](#) [English \(US\) ▾](#)

What can we help you with?

# WeTransfer Support



## Knowledge Base

Get down to the basics & the nitty gritty with these articles. Carefully curated to get you on your way again.



## WeTransfer Plus

Everything you ever wanted to know about Plus, setting up a profile page and plenty more is stored away here.



## Apps & Mobile

We've got mobile apps, mac apps, appetizers - you name it! Find them, and all the answers to your questions, here.



## Troubleshooting

Is our service getting you down or are you having a rough day? Don't worry - We've got things covered here.



## Security/Privacy



## Paper® and Paste™

WeTransfer Status

● All Systems Operational

recently viewed articles

How many GB can I upload?

Popular Articles

How do I download a transfer that was sent to me?

Can I download a transfer that expired?

Embedding a button to your Plus Profile on your own website

I'm a Plus user but people cannot send transfers over 2GB to me

How do I delete multiple transfers?



## Help and documentation

Use proactive and in-context hints to guide the user, avoid steep learning curves.



slackbot

☆ | ❤ active | slackbot



Today

Anish 3:45 PM

I need help

slackbot 3:45 PM

I can help by answering simple questions about how Slack works. I'm just a bot, though! If you need more help, try our [Help Center](#) for loads of useful information about Slack – it's easy to search! Or simply type /feedback followed by your question or comment, and a human person will get back to you. 😊

Anish 3:46 PM

Giude me how it works

slackbot 3:46 PM

I searched for that on our Help Center. Perhaps these articles will help:

- [Improve company culture with Slack](#)
- [Message and file threads](#)

Anish 3:46 PM

How can i share files with team

slackbot 3:46 PM

Hi! Slack has a wonderful built-in file uploader. There are a couple ways to share any file with your workspace:

- Copy and paste images directly into your message box.
- Tap the plus button by the message box and select an option under [Add a file from...](#)
- On desktop, drag and drop a file into your Slack window. ✨



Want to learn more about sharing files in general or from Google Drive, Dropbox, and Box? Head to our Help Center article, [Upload and share files](#).



Message @slackbot



121

# UI Design – Create Desirable Look and Feel

- A functionally nice solution is a good first step but it is better to make it look nice as well!



# UI Design

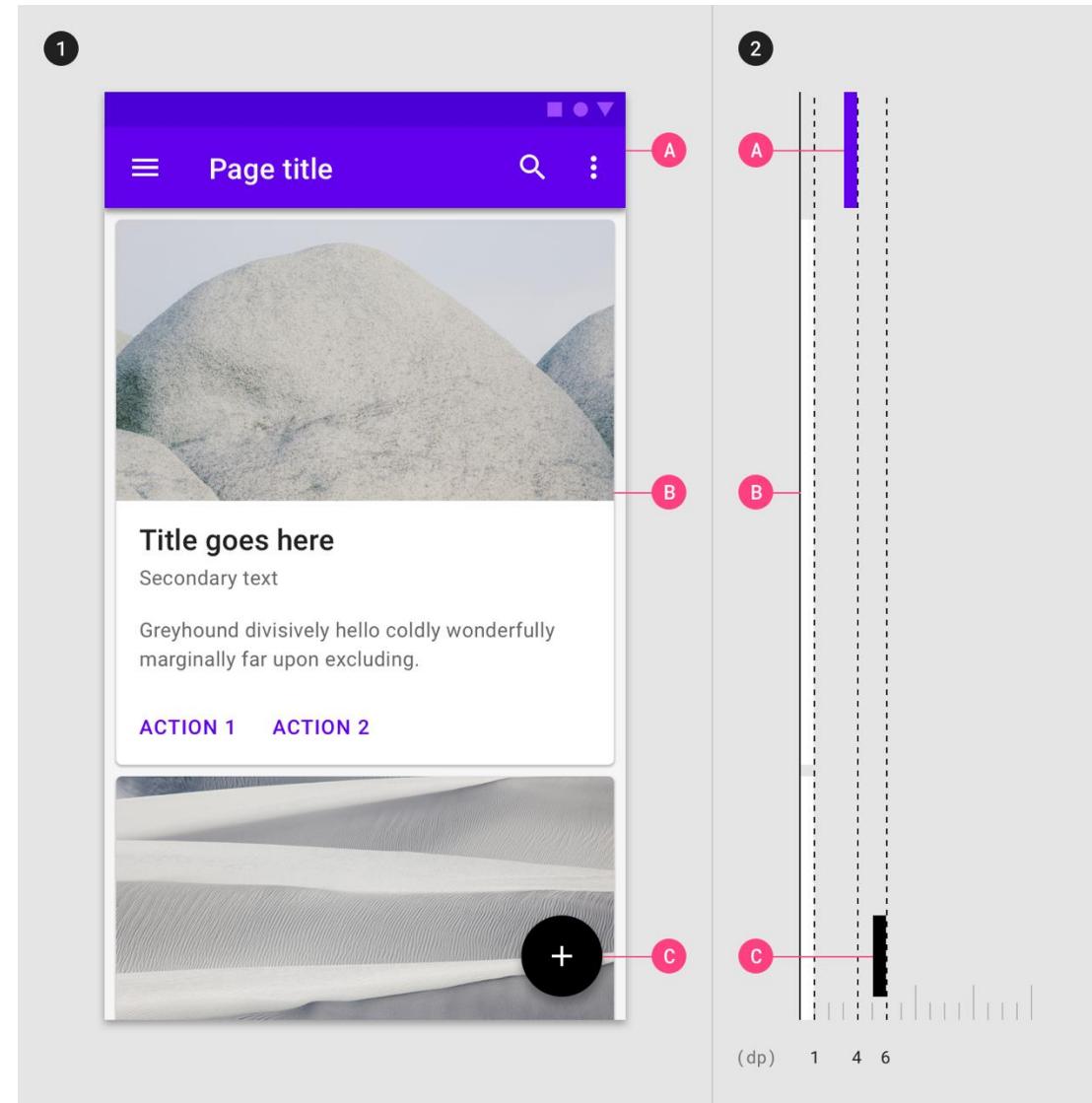
7 rules for gorgeous design by Erik D. Kennedy

1. Light comes from the sky
2. Black and white first
3. Double your whitespace
4. Learn the methods of overlaying text on images
5. Make text pop—and un-pop
6. Only use good fonts
7. Steal like an artist

<https://learnui.design/blog/7-rules-for-creating-gorgeous-ui-part-1.html>

<https://learnui.design/blog/7-rules-for-creating-gorgeous-ui-part-2.html>

# 1. Lights comes from the sky



# 2. Black and white first



## I'm a headline about this section

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin pharetra lorem non luctus condimentum. Cras egestas laoreet tincidunt. Quisque mollis lorem quis elit porttitor dapibus et at ante.

## 2. Black and white first

- Complementary
- Monochromatic
- ...

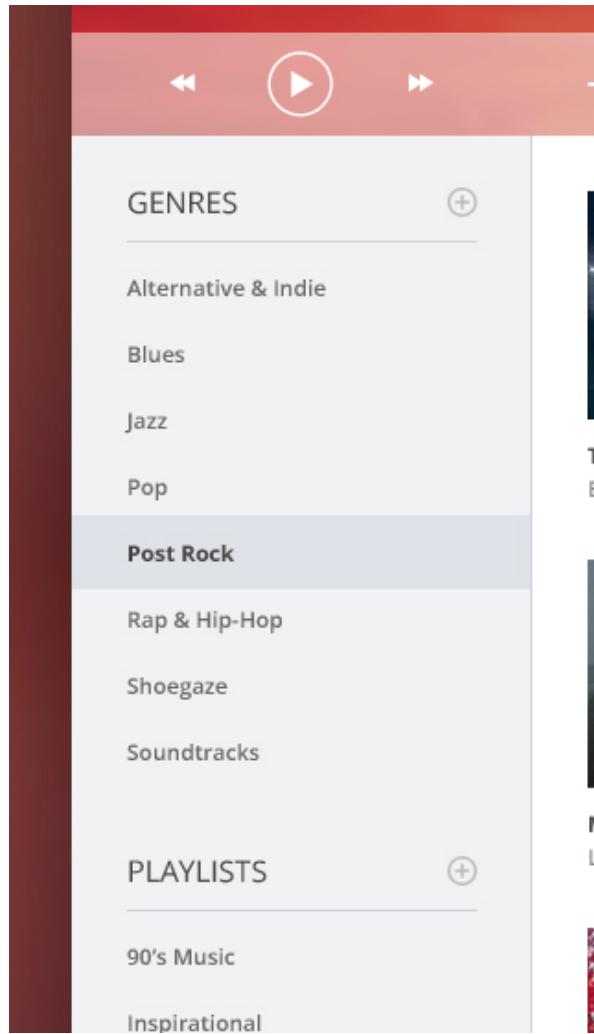


[Color wheel - color theory and calculator | Canva Colors](#)  
[Coolors - The super fast color palettes generator!](#)

# 3. Double your whitespace

The screenshot shows a Wikipedia article for "Martinique". The page has a clean layout with a large main content area. On the left, there's a sidebar with a large "W" logo and the word "WIKIPEDIA" followed by "The Free Encyclopedia". Below this is a "CONTENTS" section with links to "Abstract", "History", "European contact", "Major century", "The revolution", "Governance", "Geography", "Subdivisions", "Mount pélée", "Geopolitics", "Economics", "Demographics", "Culture", and "References". The main content area starts with the title "Martinique" and a redirect notice: "'Martinique' redirects here. For others uses, see [Martinique](#)". The text describes Martinique as an island in the Lesser Antilles in the eastern Caribbean Sea, mentioning its land area, population, and status as an overseas department of France. Below this is a "History" section with a heading "European contact" and a paragraph about the island's history from Arawaks to Columbus. To the right of the main content is a sidebar titled "MARTINIQUE" featuring a tropical beach photo, a "Country" section (France), a world map with a highlighted location for Martinique, and a "DENSITY" section showing 340 hab/km². At the bottom is a thumbnail image of a tropical sunset over a beach.

# 3. Double your whitespace



# 4. Overlaying text images

The screenshot shows the homepage of the OTTER website. At the top left is the brand name 'OTTER'. A horizontal navigation bar at the top right includes links for Home, Workshops, Products, Journal, About, Press, Contact, and Basket / 0. The main visual is a photograph of a grey and orange tent pitched on a grassy hillside overlooking the ocean. Overlaid on the center of the image is the text 'PITCH UP WITH US' in a light, sans-serif font. Below the image is a white footer bar containing four icons: a surfboard, a wrench and screwdriver, a red arrow pointing up, and two speech bubbles. Underneath these icons are the corresponding menu items: SURFBOARDS, WORKSHOPS, ABOUT, and CONTACT.

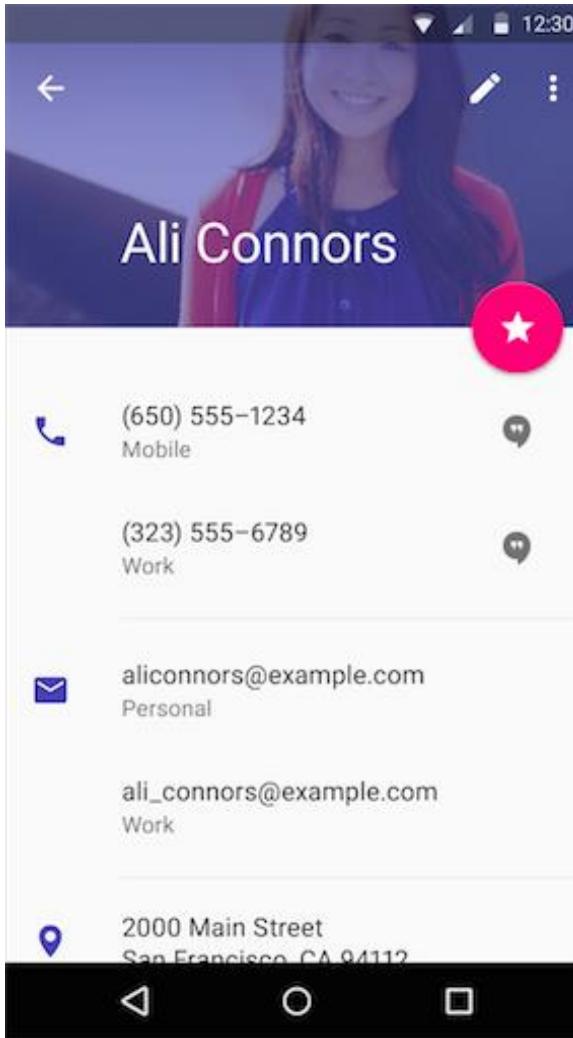
OTTER

Home Workshops Products Journal About Press Contact Basket / 0

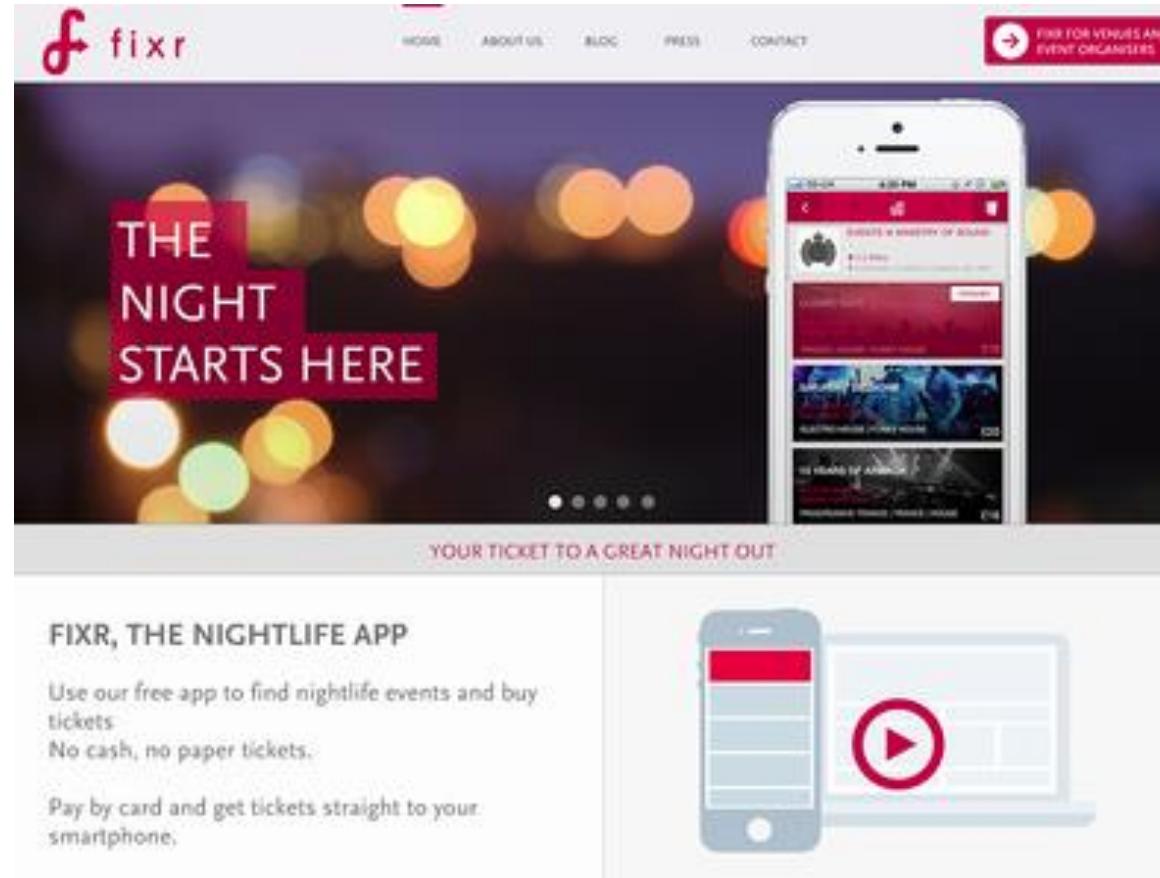
PITCH UP WITH US

SURFBOARDS WORKSHOPS ABOUT CONTACT

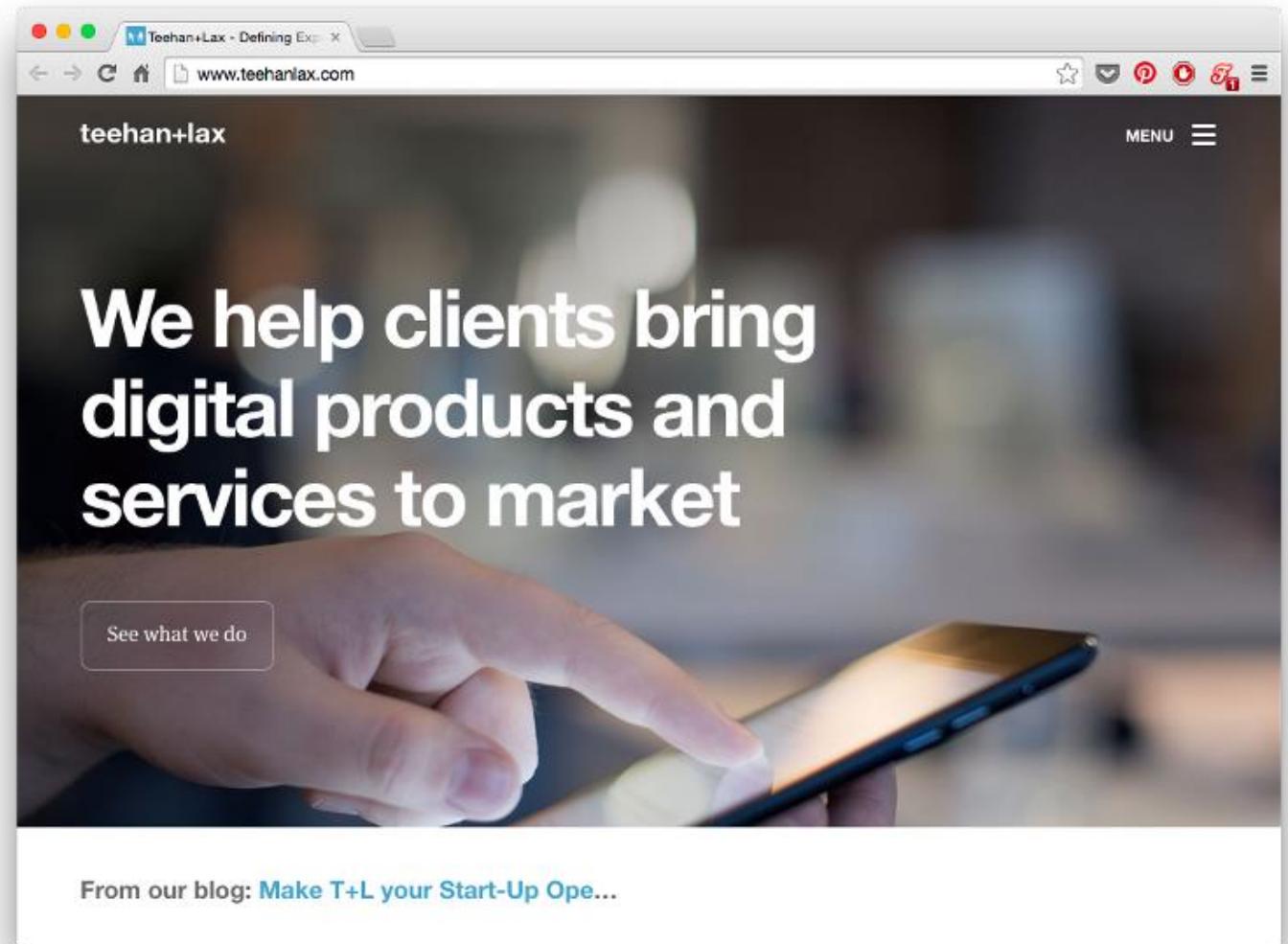
# 4. Overlaying text/images - overlay



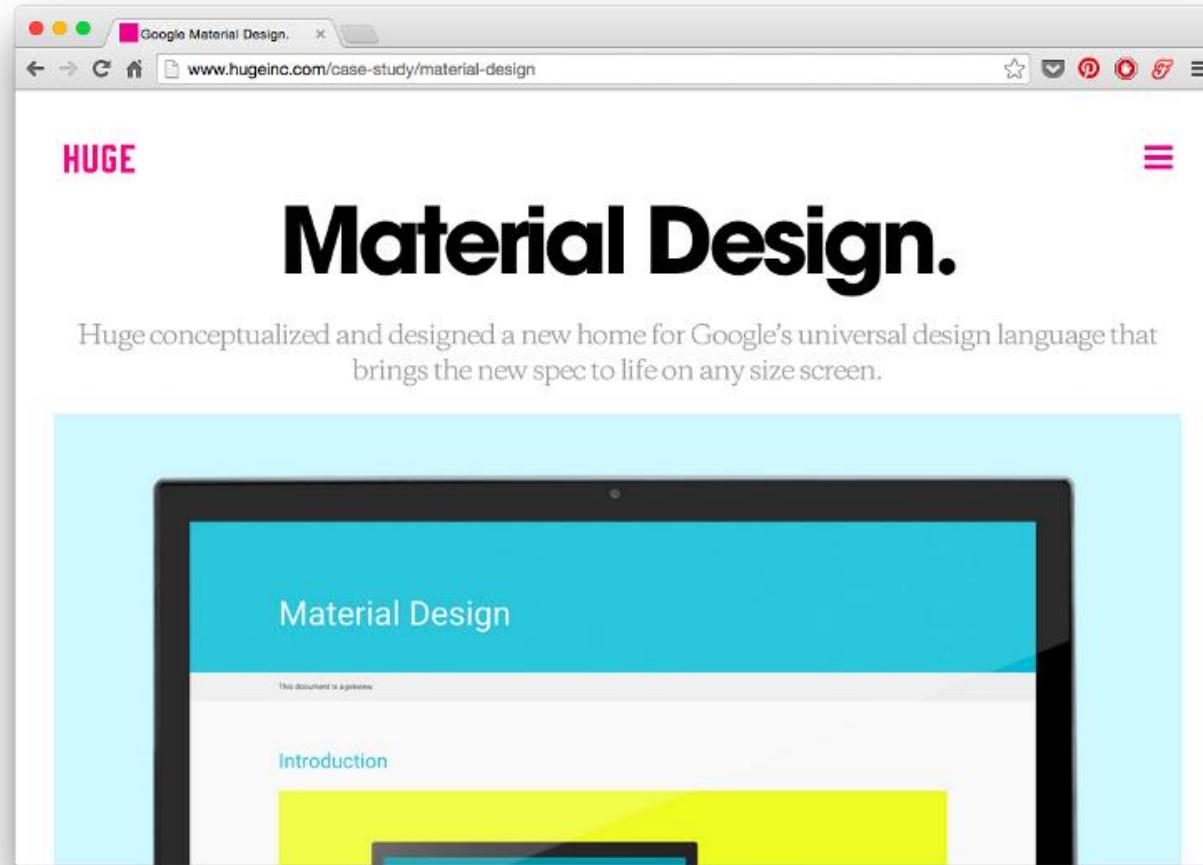
# 4. Overlaying text images – text in a box



## 4. Overlaying text images – blur the image



# 5. Make text pop and un-pop



# 6. Only good fonts

The screenshot shows a web browser displaying the BonFX website at <https://bonfx.com/best-fonts/>. The page features a header with a call to action about a book on font combinations, followed by a title 'The 100 Best Fonts (in a huge sortable table)'. Below this is a table with three rows, each listing a font: Helvetica (rank 1), Garamond (rank 2), and Frutiger (rank 3). Each row includes the font's name, designer, category, classification, foundry, and a small orange circular icon with a plus sign.

RANK	TYPEFACE NAME	DATE	TYPE DESIGNER	CATEGORY	CLASSIFICATION	FOUNDRY
1	Helvetica	1957	Max Miedinger	Sans-serif	Neo-Grotesque	Haas Typefoundry
2	Garamond	1530	Claude Garamond	Serif	Old Style	
3	Frutiger	1977	Adrian Frutiger	Sans-serif	Humanist	Linotype

<https://bonfx.com/best-fonts/>

## 7. Steal like an artist

- Dribbble
- Pttrns

# UI Design

My two cents:

- “7. Steal like an artist” – Just enough, don’t plagiarize!
  - <https://dribbble.com/>
  - <https://land-book.com/>
  - <https://interfaces.pro/>
  - ...

- <https://cantunsee.space/>

# Let's get technical

css  
is  
awesome

```
#wife {  
    right: 100%;  
    margin: 0;  
}
```

```
#usa + #mexico {  
    border: 1px dashed;  
}
```

```
#mario.mushroom {  
    transform: scale(200%);  
}
```

```
.hobbit {  
    height: 50%;  
}  
.hobbit #foot {  
    width: 200%;  
}
```

```
.push-up-bra {  
    margin-top: -25%;  
    overflow: visible;  
}
```

```
#big-bang::before {  
    content: " ";  
}
```

# CSS [Cascading Style Sheets]

- CSS is primitive and incomplete
  - Building a function, reusing a definition or inheritance are hard to achieve
- Pre-processors help to achieve writing reusable, maintainable and extensible codes in CSS
  - They extend CSS with
    - Variables
    - Operators
    - Interpolations
    - Functions
    - Mixins

# Preprocessors



# Preprocessors - SASS

## Variables

```
$font-size: 16px;  
  
div {  
  font-size: $font-size;  
}
```

Sass

## Nesting

```
$link-color: #999;  
$link-hover: #229ed3;  
  
ul {  
  margin: 0;  
  
  li {  
    float: left;  
  }  
  
  a {  
    color: $link-color;  
  
    &:hover {  
      color: $link-hover;  
    }  
  }  
}
```

Sass

# Preprocessors - SASS

## Mixins

```
@mixin bordered($width) {  
    border: $width solid #ddd;  
  
    &:hover {  
        border-color: #999;  
    }  
}  
  
h1 {  
    @include bordered(5px);  
}
```

Sass

## Extends

```
.block { margin: 10px 5px; }  
  
p {  
    @extend .block;  
    border: 1px solid #eee;  
}  
  
ul, ol {  
    @extend .block;  
    color: #333;  
    text-transform: uppercase;  
}
```

Sass

# Preprocessors - SASS

## Color operations

```
saturate($color, $amount)  
desaturate($color, $amount)  
lighten($color, $amount)  
darken($color, $amount)  
adjust-hue($color, $amount)  
opacify($color, $amount)  
transparentize($color, $amount)  
mix($color1, $color2[, $amount])  
grayscale($color)  
complement($color)
```

Sass

## If/Else statements

```
@if lightness($color) > 30% {  
  background-color: black;  
}  
  
@else {  
  background-color: white;  
}
```

Sass

# Preprocessors - SASS

## Loops

```
@for $i from 1px to 3px {  
  .border-#{$i} {  
    border: $i solid blue;  
  }  
}
```

Sass

## Math

```
1cm * 1em => 1 cm * em  
2in * 3in => 6 in * in  
(1cm / 1em) * 4em => 4cm  
2in + 3cm + 2pc => 3.514in  
3in / 2in => 1.5
```

Sass

## Imports

```
@import "library";  
@import "mixins/mixin.scss";  
@import "reset.css";
```

Sass

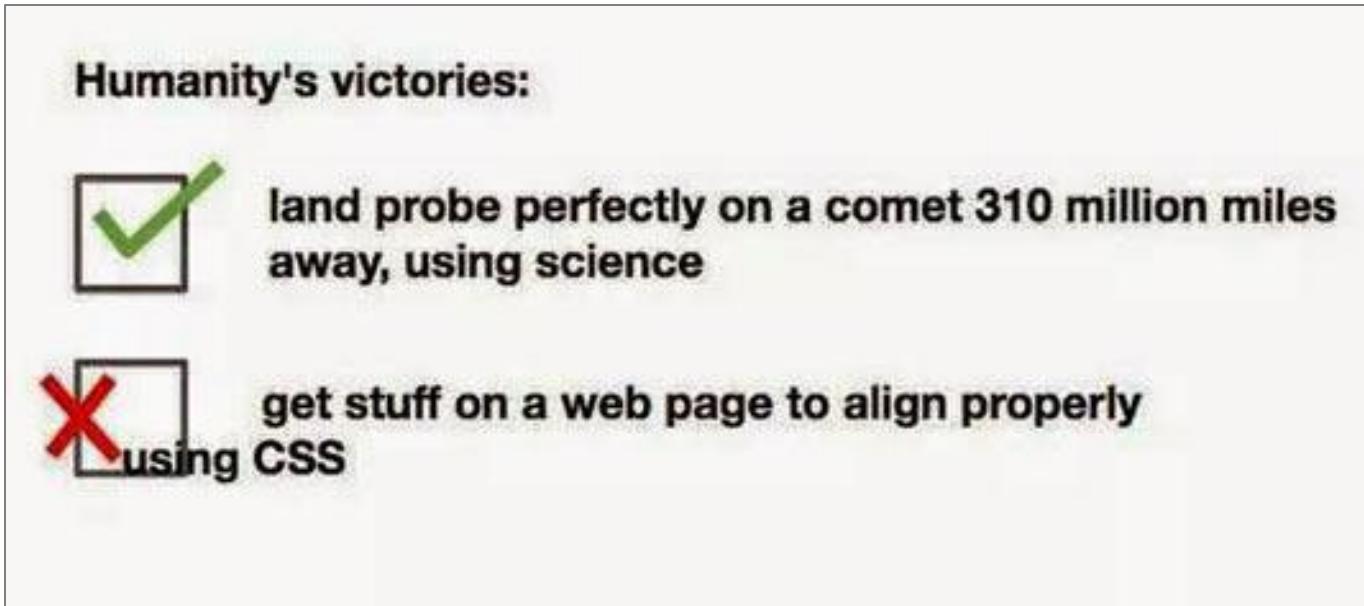
SORRY

THERE ARE NO  
SILVER BULLETS....

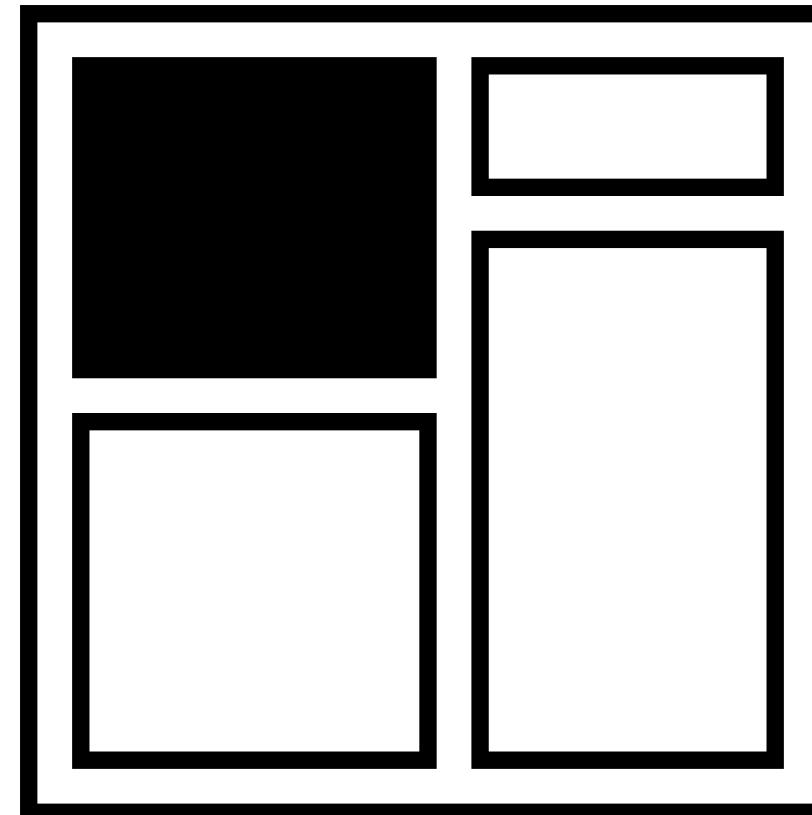


# Preprocessors – Missing features

- Preprocessors don't fix layouting issues

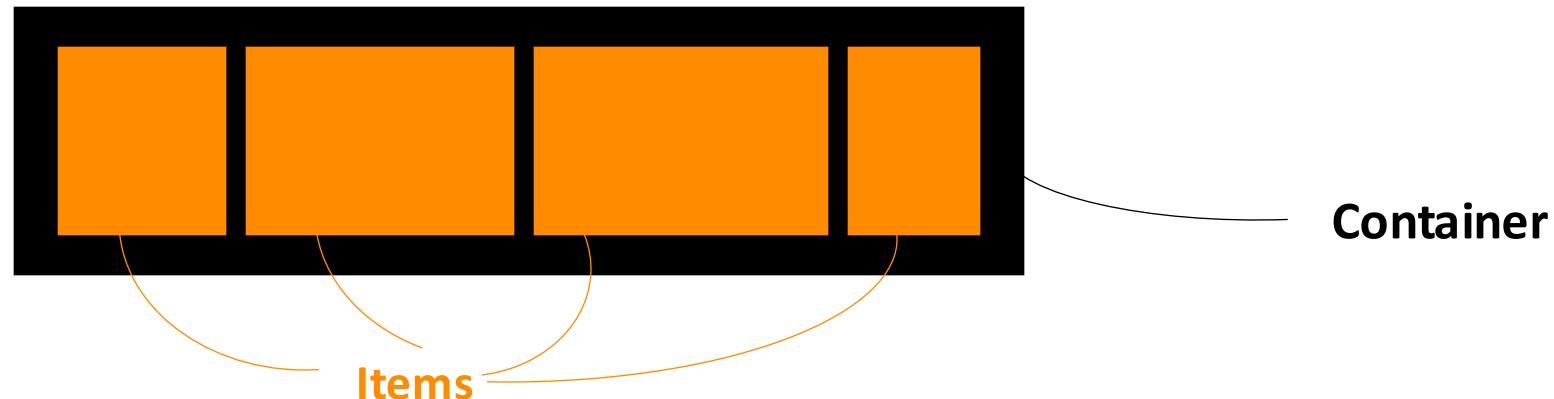


# CSS – Layout mechanisms



# CSS -flexbox

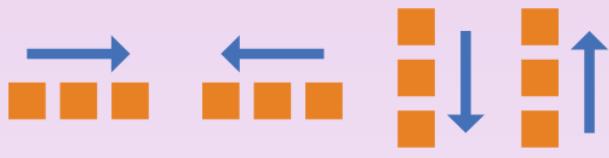
- A more efficient way to
  - Lay out
  - Align
  - Distribute space among items in a container
  - ... even when their size is unknown and/or dynamic



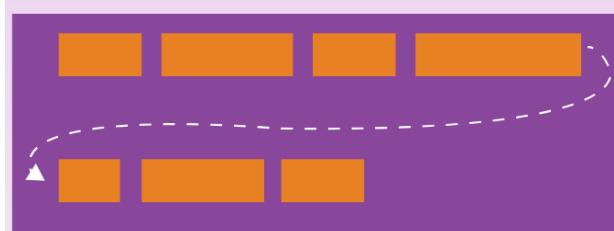
# CSS - Flexbox

```
.container {  
    display: flex;  
}
```

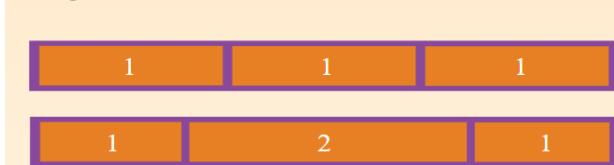
flex-direction



flex-wrap



flex-grow



## justify-content

flex-start



flex-end



center



space-between



space-around



## align-items

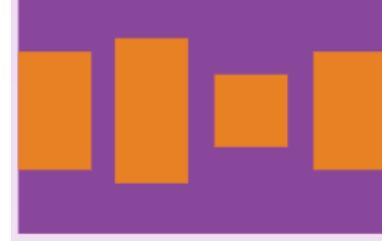
flex-start



flex-end



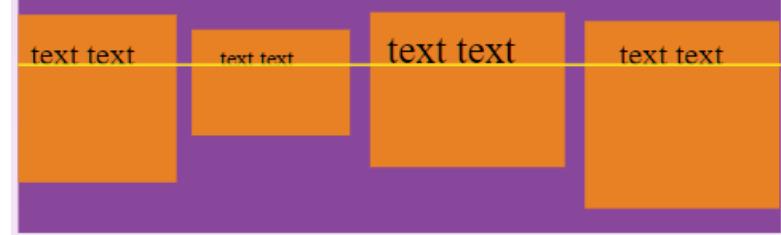
center



stretch

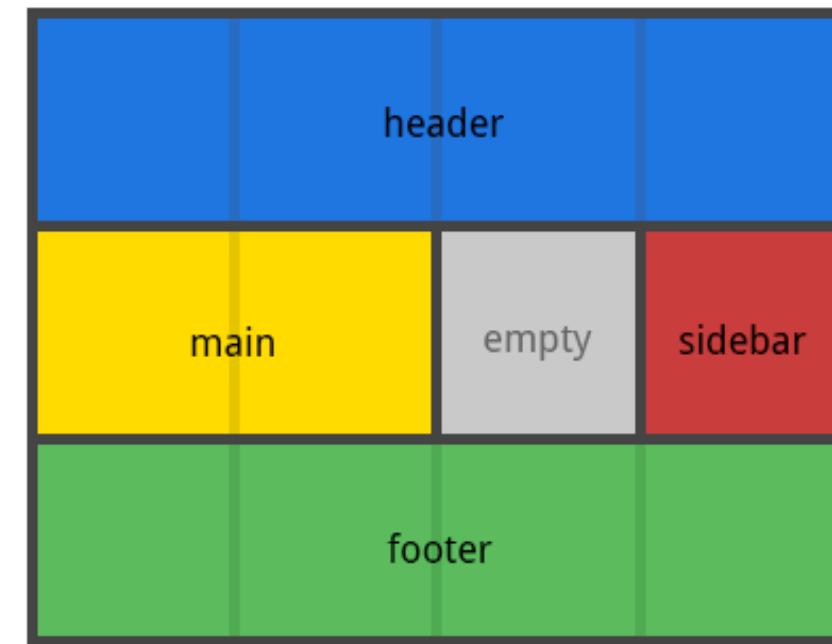
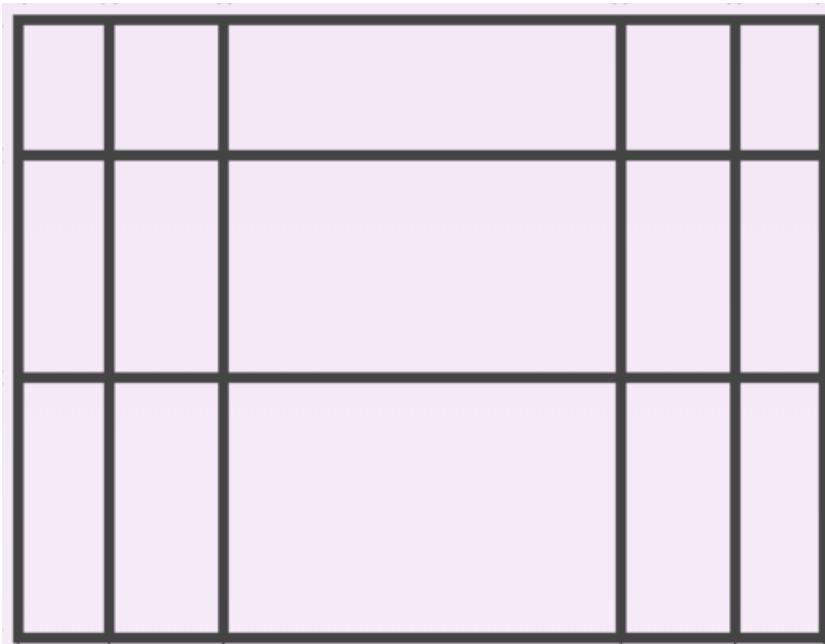


baseline



# CSS - Grid

- 2-dimensional system
  - Can handle rows and columns

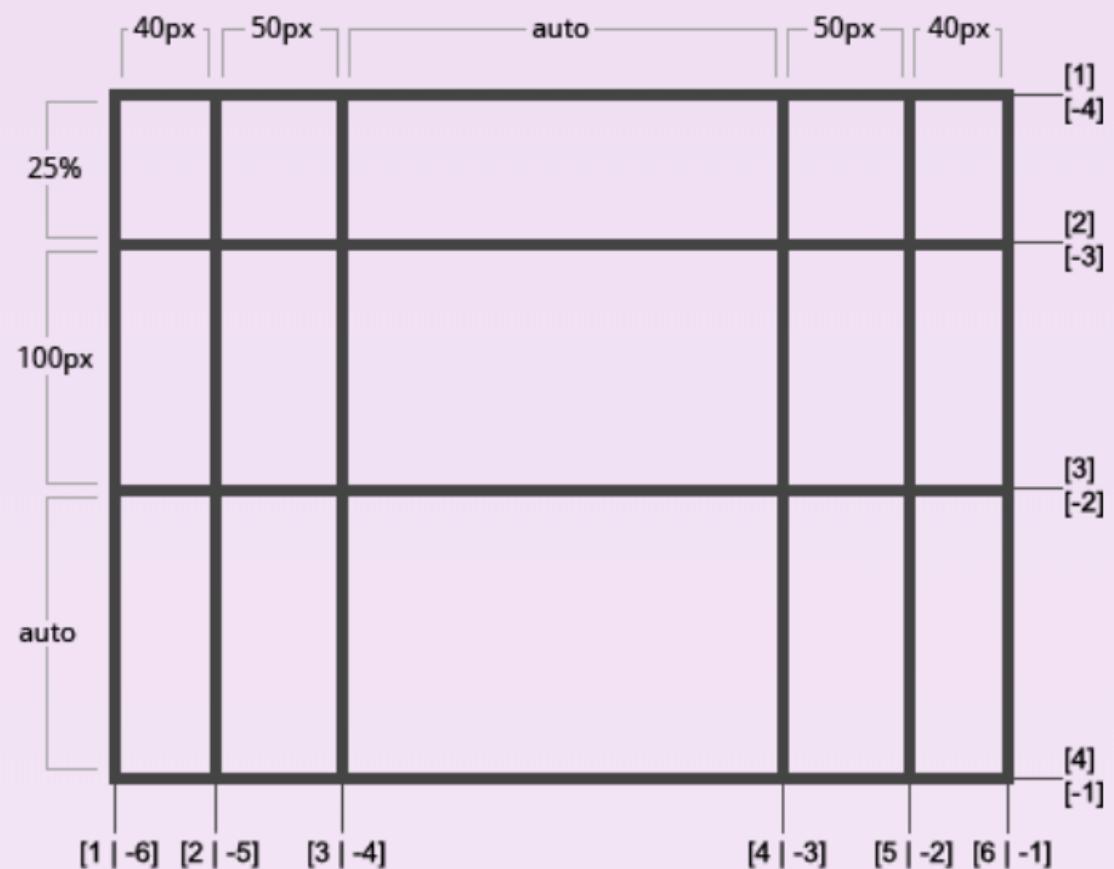


# CSS - Grid

```
.container {  
    display: grid | inline-grid;  
}
```

css

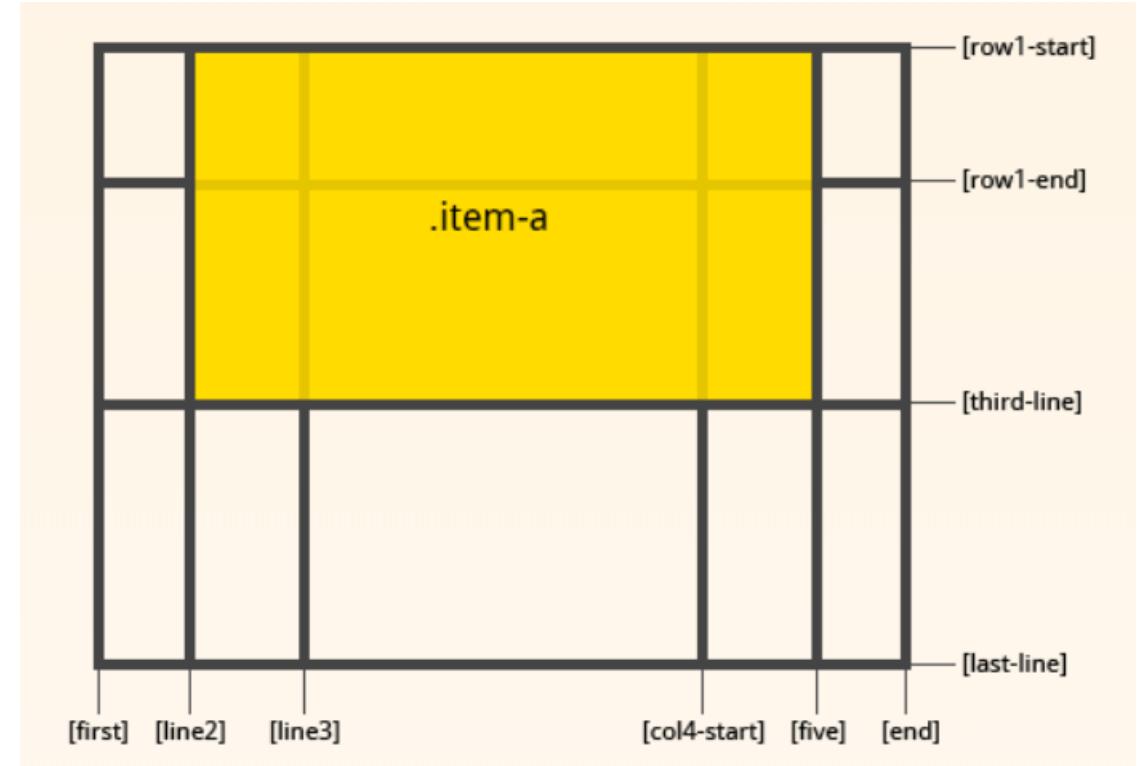
```
.container {  
    grid-template-columns: 40px 50px auto 50px 40px;  
    grid-template-rows: 25% 100px auto;  
}
```



# CSS - Grid

css

```
.item-a {  
    grid-column-start: 2;  
    grid-column-end: five;  
    grid-row-start: row1-start  
    grid-row-end: 3  
}
```



- <https://flexboxfroggy.com/>
- <https://cssgridgarden.com/>
- <https://cssgrid-generator.netlify.app>

# CSS – Design & components



- A toolkit used to build responsive, mobile first web apps
- Provides
  - Grid layout based on flexbox
  - Classes for typography, responsive images, and tables
  - UI components such as buttons, breadcrumbs, cards, dropdowns, form components, modals, navbars, tooltips, ...

# CSS – Design & components



- A visual language that synthesizes the classic principles of good design with the innovation of technology and science.
- Provides
  - Design principles describing how to use shadows, elevation, animations, shapes, icons,...
  - Libraries of pre-styled components

e.g. <https://material.io/design/>

# CSS – Design & components



## Tailwind CSS

- A utility-first CSS framework packed with classes like **flex**, **pt-4**, **text-center** and **rotate-90** that can be composed to build any design, directly in your markup.
- Provides
  - An API for your design system
  - It's tiny in production
  - A unique styling for your application

e.g. <https://tailwindcss.com/>

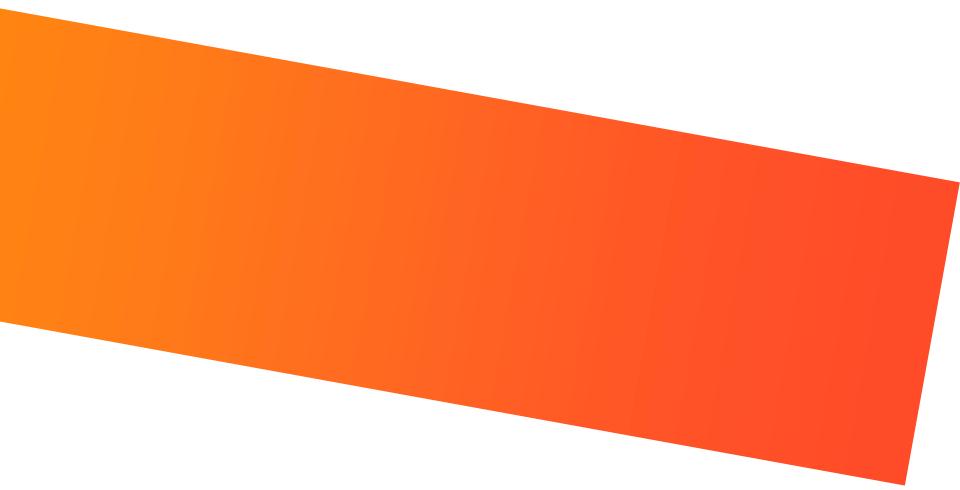
# Component libraries

- Material Design (<https://material.angular.io> )
- NGX Bootstrap (<https://github.com/valor-software/ngx-bootstrap>)
- Prime NG (<https://www.primefaces.org/primeng/>)
- NG Bootstrap (<https://ng-bootstrap.github.io/#/home>)
- Onsen UI (<https://onsen.io/angular2/>)
- Vaadin Components (<https://vaadin.com/docs/v7/elements/angular2-polymer/overview.html>)
- NG-ZORRO (<https://github.com/NG-ZORRO/ng-zorro-antd>)
- NG Lightning (<https://github.com/ng-lightning/ng-lightning>)
- NG Semantic-UI (<https://github.com/vladotesanovic/ngSemantic>)
- Clarity (<https://github.com/vmware/clarity>)
- NG2 Charts (<https://valor-software.com/ng2-charts>)
- ....

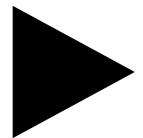
# CSS – Icons



**ALM**



# ALM – Use cases



**Run application locally**



**Lint and test application**



**Build application**



**Clean build output**



**Build application  
optimized for production**



**Run production app locally**



**Deploy application**

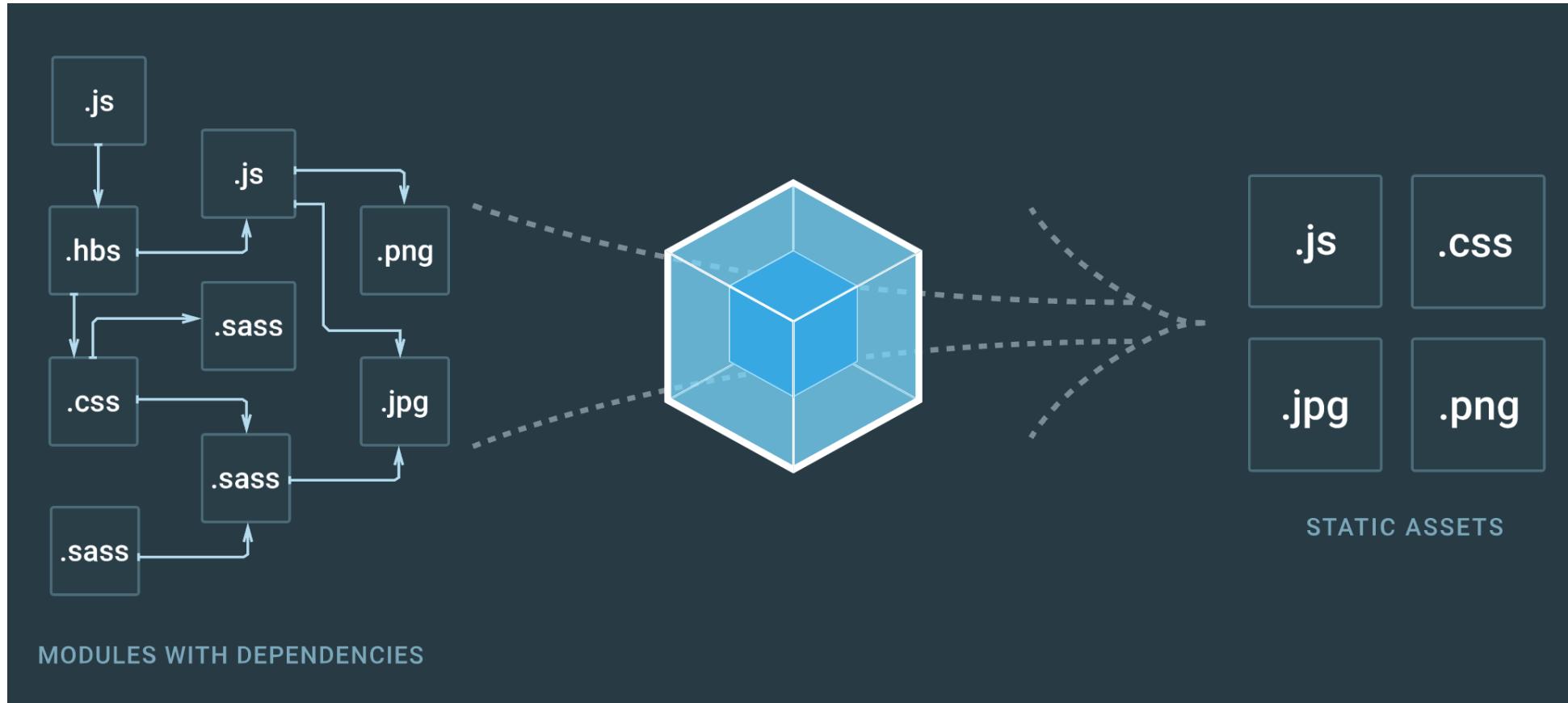
# ALM – building blocks

- Every use case consists of specific building blocks (aka tasks)

Linting	Testing	E2E testing	Variable substitution
Compilation	Bundling	Code splitting	Package installation
Caching	Minification	Tree shaking	Deploying
Copying files	Cleaning output	Packaging	Spinning up a web server

# ALM - Task runners, bundlers, module loaders

de



# Alternatives



**PARCEL**  
Blazing fast, zero configuration web application bundler



Vite



Snowpack



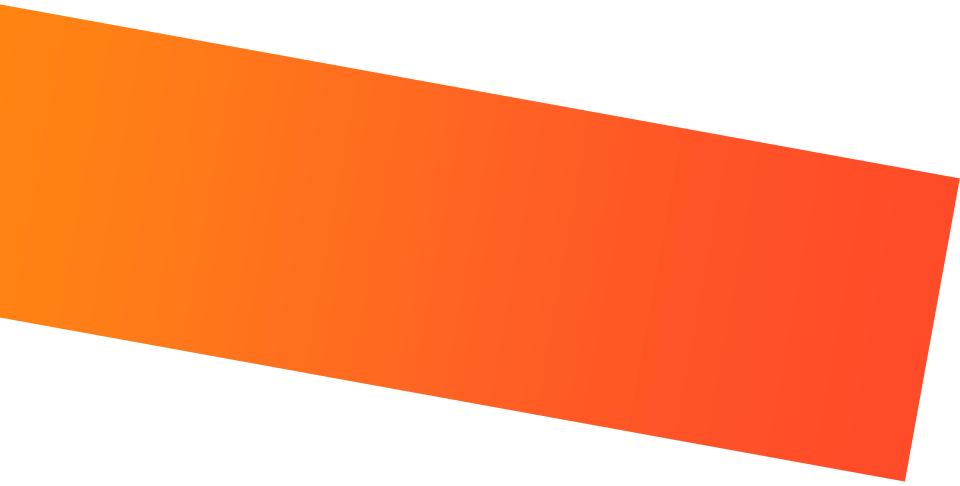
**GRUNT**

# ALM – Script execution

## npm scripts

```
"scripts": {  
  "ng": "ng",  
  "start": "node server.js",  
  "start-local": "ng serve --proxy-config proxy.conf.json",  
  "build": "ng build",  
  "build-prod": "ng build --prod",  
  "clean": "rimraf dist",  
  "clean-build": "npm run clean && npm install && npm run build-prod",  
  "test": "ng test",  
  "lint": "ng lint",  
  "e2e": "ng e2e",  
  "postinstall": "npm run build-prod"  
}
```

**QA**





**Linting** is the process of checking the source code for Programmatic as well as Stylistic errors.

This is most helpful in identifying coding mistakes and violations against code style agreements.



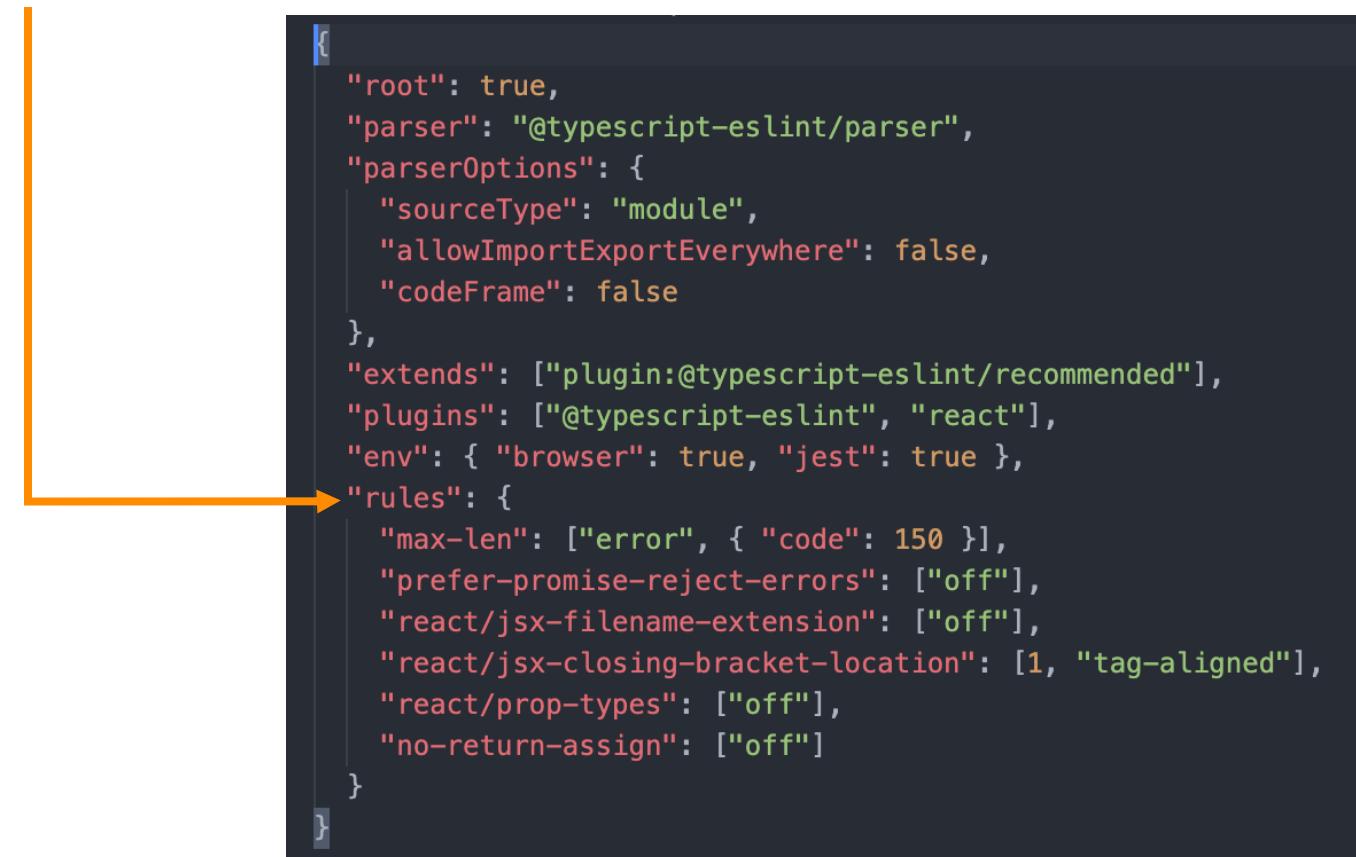
# Linting



Deprecated in favor of  
ESLint

# ESLint

- Rules are configured in `eslintrc.json/.js`



```
{  
  "root": true,  
  "parser": "@typescript-eslint/parser",  
  "parserOptions": {  
    "sourceType": "module",  
    "allowImportExportEverywhere": false,  
    "codeFrame": false  
  },  
  "extends": ["plugin:@typescript-eslint/recommended"],  
  "plugins": ["@typescript-eslint", "react"],  
  "env": { "browser": true, "jest": true },  
  "rules": {  
    "max-len": ["error", { "code": 150 }],  
    "prefer-promise-reject-errors": ["off"],  
    "react/jsx-filename-extension": ["off"],  
    "react/jsx-closing-bracket-location": [1, "tag-aligned"],  
    "react/prop-types": ["off"],  
    "no-return-assign": ["off"]  
  }  
}
```

# ESLint

- Built-in presets can be extended
- Lots of available configurations, e.g. Airbnb
- Custom rules can be added
  - e.g. Codelyzer for Angular following Angular styleguide

```
{  
  "extends": [ "codelyzer" ],  
  "rules": {  
    "component-class-suffix": true,  
    "component-max-inline-declarations": true,  
    "component-selector": [ true, "element", "sg", "kebab-case" ],  
  }  
}
```

# Testing



SINON.JS



# Testing

Example: Anatomy of a Jasmine spec

```
describe("A spec (with setup and tear-down)", () => {  
  let foo;  
  
  beforeEach(()=> {  
    foo = 0;  
    foo += 1;  
  });  
  
  afterEach(()=> {  
    foo = 0;  
  });  
  
  it("can have more than one expectation", ()=> {  
    expect(foo).toEqual(1);  
    expect(foo).toEqual(2);  
  });  
});
```

The diagram illustrates the structure of a Jasmine test suite with the following annotations:

- A curved orange arrow points from the opening brace of the `describe` block to the text *A test suite*.
- A curved orange arrow points from the `beforeEach` block to the text *Setup*.
- A curved orange arrow points from the `afterEach` block to the text *Tear down*.
- A curved orange arrow points from the `it` block to the text *Test logic (aka spec)*.
- A curved orange arrow points from the final closing brace to the text *Expectations*.

# Testing

Unit tests are important in front end as well!

Examples:

- Test form logic (validators, disabling,...)
- Test services and utilities
- Test initialisation of view
- Test authentication/authorization logic
- ...

# Most useful types of FE tests

- Testing flows in complex applications
  - Selecting A should bring me to flow 1
  - Selecting B should bring me to flow 2
- Button behaviour
  - Calling an api
  - Doing a function
  - Showing a new component
- Testing business logic
  - Filters
  - Sorts

# Exercises

```
git checkout qa
```

# Exercises

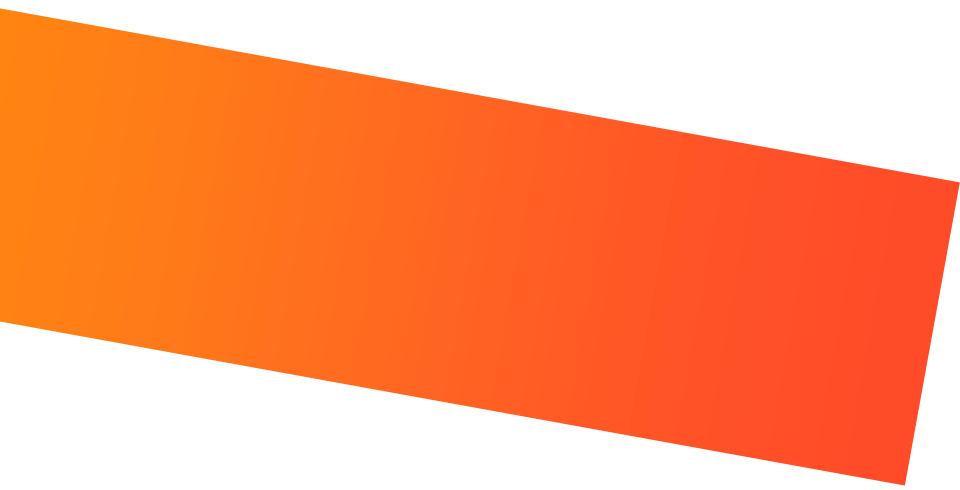
1. Install *tslint* as **dev dependency**
2. Execute **npm run lint** and fix issues

# Solution

```
git checkout qa-solutions
```

```
npm run lint
```

# Security



# Client-Side Security

- All front-end code is at the client side (unless you use SSR)!
  - Do not put secrets in your code
  - Use established security models and libraries
  - Never ever trust the user
- Fortify your APIs and check all user input that arrives
- Do not underestimate the effect of attacks originating at the client side!
- Always keep your framework and dependencies up to date!

# XSS – Cross Site Scripting

- Injection of malicious scripts into app
  - Via URL params
  - Via input fields
- Most frameworks prevent XSS attacks to occur
  - Content can be sanitized but should be used with care

# CORS – Cross Origin Resource Sharing

- Only allow access to API from trusted origins
  - Via http header **access-control-allow-origin**
- Only allow use of specific http verbs
  - Via http header **access-control-allow-method**
- Only allow specific headers
  - Via http header **access-control-allow-headers**

# CSP – Content Security Policy

<https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP>

- Only allow content from defined sources
  - Script-src
    - 'self'
    - 'none'
  - Style-src
    - 'unsafe-inline'
  - Font-src
    - Mycrelan.crelan.be
  - Connect-src
    - \*.crelan.be
  - Img-src
    - Default-src
- Via meta tag or header set by API: **Content-security-policy**

# NSP

- Node security platform
  - Identifies known vulnerabilities to node modules used in project
  - Available via CLI tool or gulp plugin
  - FYI: Dependabot in Github is similar
  - Build-in in npm: npm audit

High	Arbitrary File Overwrite
Package	tar
Patched in	>=4.4.2
Dependency of	npm-lifecycle
Path	npm-lifecycle > node-gyp > tar
More info	<a href="https://npmjs.com/advisories/803">https://npmjs.com/advisories/803</a>

Found 19 vulnerabilities (8 moderate, 11 high) in 11360 scanned packages  
run `npm audit fix` to fix 4 of them.  
12 vulnerabilities require semver-major dependency updates.  
3 vulnerabilities require manual review. See the full report for details.

## [Security] Bump bower from 1.8.2 to 1.8.8 #80

Merged dependabot merged 1 commit into master from dependabot/npm\_and\_yarn/bower-1.8.8 7 days ago

Conversation 0 Commits 1 Checks 0 Files changed 1 +2 -2

dependabot bot commented 7 days ago

Bumps bower from 1.8.2 to 1.8.8. This update includes security fixes.

Vulnerabilities fixed  
Sourced from [The Node Security Working Group](#).

Arbitrary File Write Through Archive Extraction  
attackers can write arbitrary files when a malicious archive is extracted.

Affected versions: <1.8.7

Release notes  
Commits  
Maintainer changes  
compatibility 88%

Dependabot will resolve any conflicts with this PR as long as you don't alter it yourself. You can also trigger a rebase manually by commenting @dependabot rebase.

If all status checks pass Dependabot will automatically merge this pull request.

Dependabot commands and options

[Security] Bump bower from 1.8.2 to 1.8.8 ... Verified e464683

dependabot bot added dependencies security labels 7 days ago

Subscriptions

No reviews

No one assigned

Labels

dependencies security

Projects

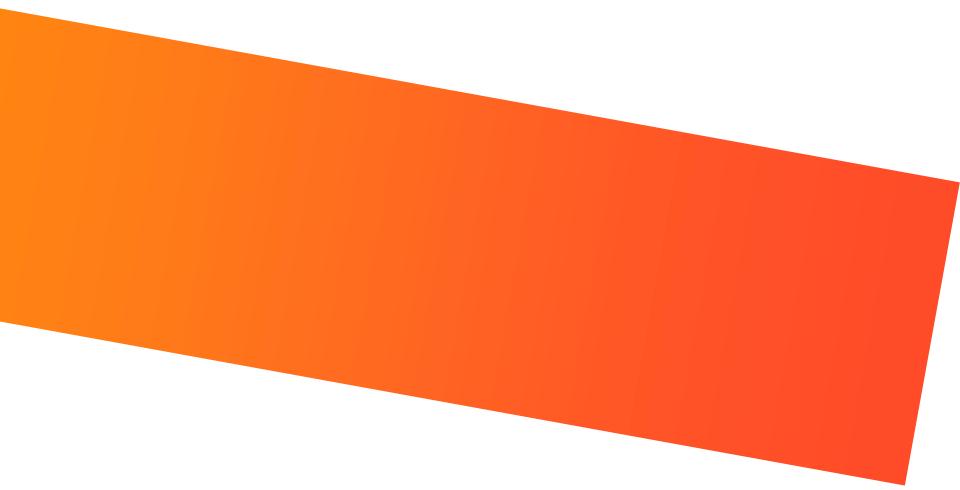
No milestone

Notifications

You're not receiving notifications from this thread.

1 participant

# Performance Optimization

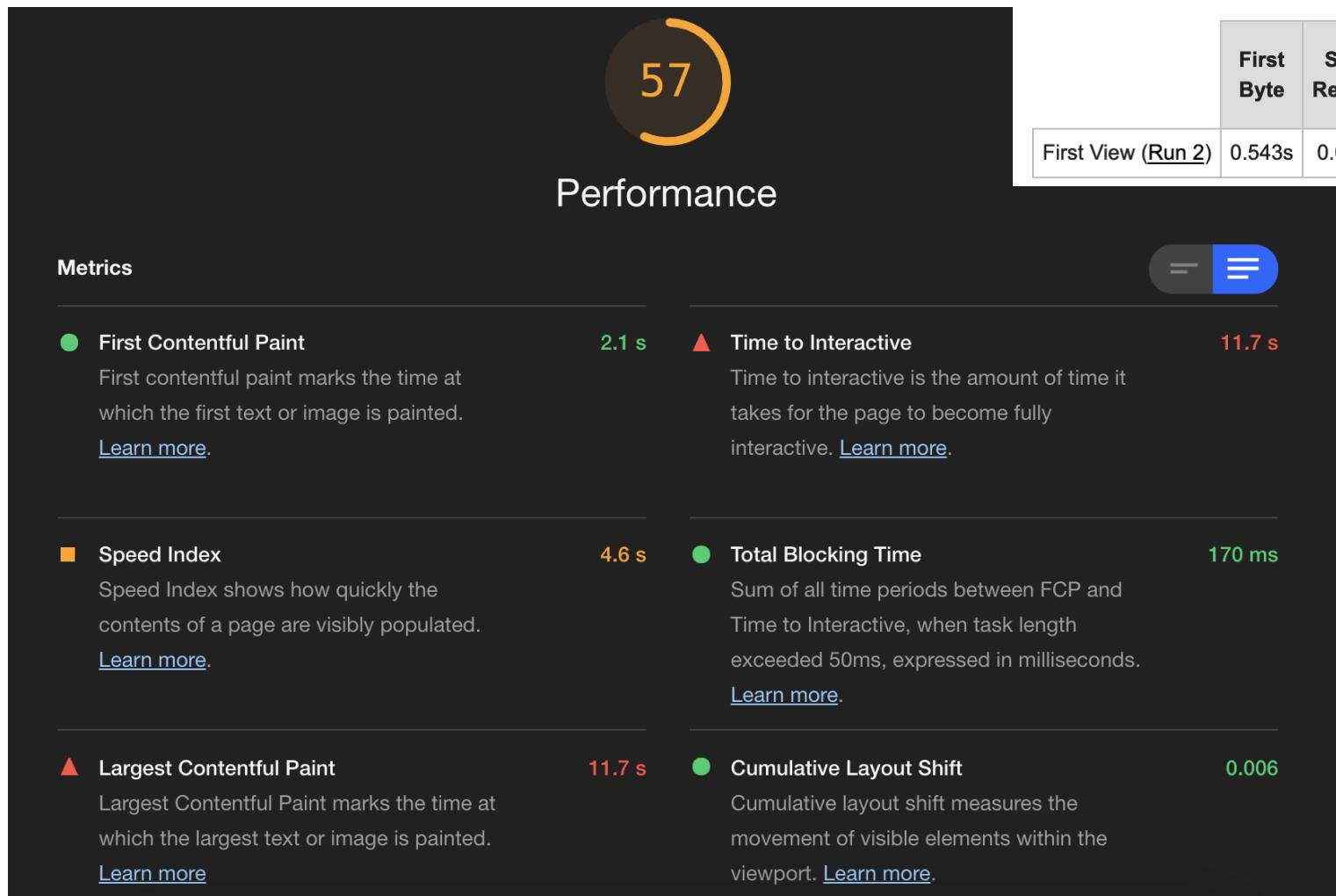


# Test your sites performance

- Different metrics to determine performance (source: <http://web.dev>)
  - First Contentful Paint
  - First Meaningful Paint
  - Speed Index
  - First CPU Idle
  - Time to Interactive
  - Max Potential First Input Delay
  - Total Blocking Time
  - Largest Contentful Paint
- Different tools and websites to measure performance
  - Lighthouse: built-in in Chrome (powerful tool)
  - For more detail: Performance tab in your Browser's DevTools
  - Other websites like: <https://www.webpagetest.org/>

# Test your sites performance

ae.be's performance



Performance Results (Median Run - SpeedIndex)

	First Byte	Start Render	<u>First Contentful Paint</u>	Speed Index	<u>Web Vitals</u>		Document Complete		
					Largest Contentful Paint	Cumulative Layout Shift	Time	Requests	Bytes In
First View (Run 2)	0.543s	0.000s	1.099s	0.000s	5.657s	0.011	5.730s	62	3,859 KB

# Test your sites performance

Lighthouse gives concrete tips on how to improve

▲ Serve static assets with an efficient cache policy — 22 resources found

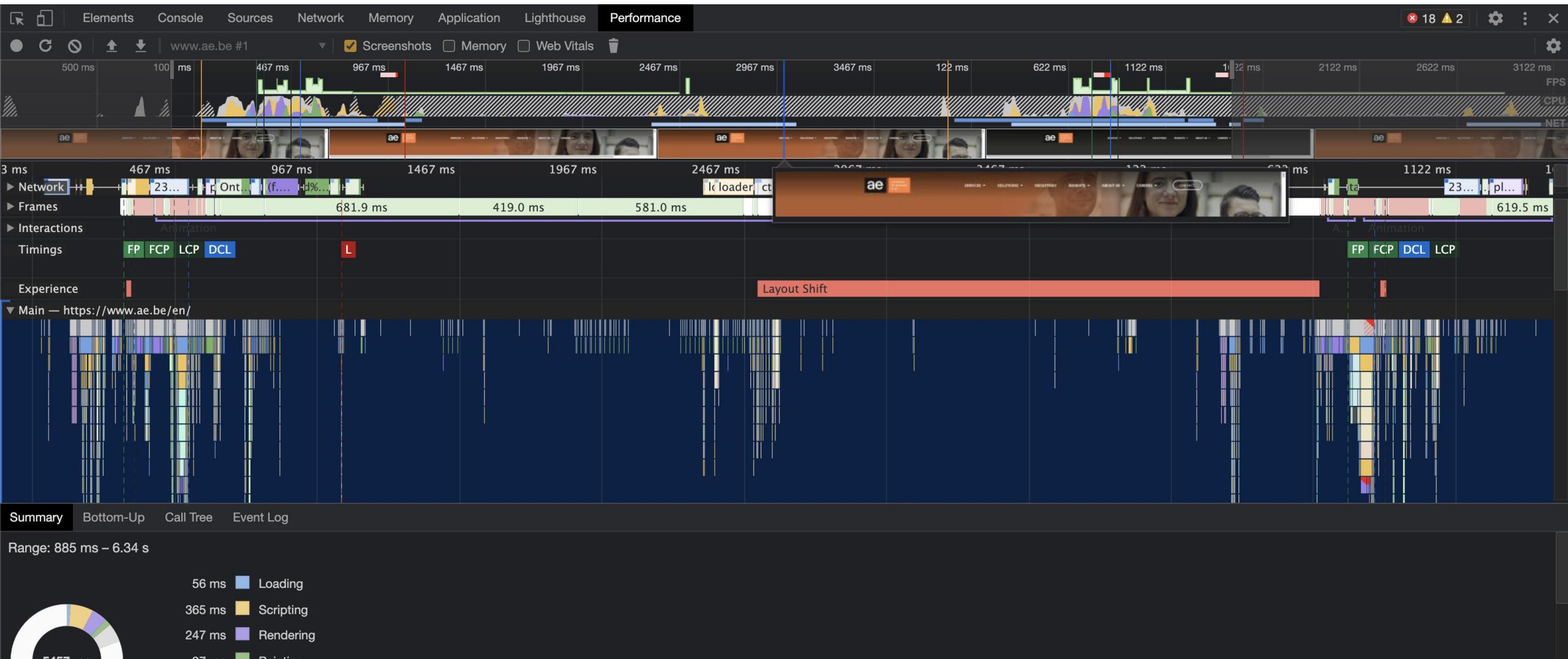
A long cache lifetime can speed up repeat visits to your page. [Learn more](#).

Show 3rd-party resources (13)

URL	Cache TTL	Transfer size
...88dw4xyxdvhvt79/mobile-more-menu-white.png (dl.dropboxusercontent.com)	1 m	2 KiB
...scriptloader/2015084.js (www.ae.be)	1 m	1 KiB
...cta/current.js (www.ae.be)	10 m	4 KiB
...js/8043754.js (www.iubenda.com)	1 d	0 KiB
/hubfs/BCC%20Background%20image.png (www.ae.be)	14 d	1,099 KiB
...2015084/Whitenpaper%20Ecosystems%20-		

# Test your sites performance

AE's performance



# Optimization tips

## 1. Image optimization

- Compress, compress, compress ! No need to serve 4K images if your target device is a smartphone
- Use plugins (e.g. `npm i compress-images`) or compress manually (e.g. ImageAlpha)
- Take advantage of responsive images using HTML `srcset` and `sizes`
- Use a preloader (in a smart way)
- More concrete
  - Replace GIFs with videos (animated gifs can be huge)
  - Use webp if possible (`npm install imagemin-webp`)
  - Serve images with correct dimensions

# Optimization tips

## 2. Reduce HTTP requests / consider adaptive serving

- Combine requests if possible or write a BFF (backend-for-frontend) if appropriate
- Perform smart asynchronous calls
- Only call APIs when necessary
- Implement caching strategies in FE and BE
- Adaptive serving:
  - Adapting which assets you're serving to users based on the quality of their connection: e.g. Netflix lowering quality (while playing) when connection is bad
  - Using Network Information API  
→ `var connection = navigator.connection`

# Optimization tips

## 2. Minify JS/CSS/...

- Built-in in most frameworks/bundlers (at least for production builds)
- Check out tools such as:
  - Terser: <https://github.com/terser/terser>
  - RollupJS: <https://rollupjs.org/guide/en/>
  - Browserify <http://browserify.org/>
- Verify your production builds: are they smaller/better/faster ... ? If not, something could be wrong with your configuration ☺

# Optimization tips

## 3. Use a CDN (where appropriate)

- CDN = content delivery network
- Refers to a geographically distributed group of servers which work together to provide fast delivery of (online) content
- Allows for the quick transfer of assets needed for loading Internet content including HTML pages, javascript files, stylesheets, images, and videos.
- Examples:
  - Cloudflare
  - ImageEngine
  - PageCDN

# Optimization tips

## 4. Avoid 301 redirects

- Simple: more redirects is more round-trip times
- Avoid them if possible



301

Moved Permanently

# Optimization tips

## 5. Consider server-side rendering

- Typically, fetching a web application will look like this (here: React)

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <link rel="shortcut icon" href="/favicon.ico">
    <title>React App</title>
  </head>
  <body>
    <div id="root"></div>
    <script src="/app.js"></script>
  </body>
</html>
```

- Next, the browser fetches the app.js file and loads the app



# Optimization tips

## 5. Consider server-side rendering

- This could become an issue if:
  - the end user is using a slow internet connection, e.g. mobile 3G
  - the end user is running an underpowered device, e.g. older smartphone
- Solution (or one of the possible solutions)  
= server-side rendering (SSR)
- Details would lead us too far, but basically
  - You return web pages with (dynamic!) content filled in
  - You refetch (and rerender) only when necessary
  - **Pros:** server is more powerful and thus can do computations instead of client
  - **Cons:** routing is more difficult
  - Check out: next.js, Gatsby, Angular Universal, Quasar, Nuxt.js ...



# Optimization tips

Lots & lots more ...

- Check out <https://web.dev/learn/> for a great overview and how-to's



The screenshot shows the "Learn" section of the web.dev website. At the top, there is a navigation bar with links for "Learn" (which is underlined in blue), "Measure", "Blog", and "About". There is also a search bar and a "SIGN IN" button. The main title "Learn" is centered above a subtitle: "Explore our structured learning paths to discover everything you need to know about building for the modern web." Below this, there is a section titled "Performance" featuring three collections:

- Collection • 11 resources**: **Web Vitals** (Icon: A smartphone with a graph overlay). Description: Essential metrics for a healthy site.
- Collection • 8 resources**: **Metrics** (Icon: A line chart with a dashed line). Description: Measuring performance and user experience.
- Collection • 60 resources**: **Fast load times** (Icon: A computer monitor with a progress bar and a downward arrow). Description: Techniques for improving site performance.

<https://tinyurl.com/fee-2025>

# Key takeaways

- Javascript is the language of the web because of its integration in webbrowsers
- Typescript is a very useful to avoid runtime type errors
- We use npm to manage our dependencies
- Component design is as if not more important than the code itself
- There are a lot of different FE-frameworks out there all with their own strengths and weaknesses
- To make an application look good you can use a simple few rules of thumb
- Security and testing in the front-end are as important as in the backend

# Final questions?

**THANK  
YOU**

