



DOBRODOŠLI NA PREDAVANJE

JavaScript u praksi



OSIJEK
SOFTWARE
{ITY_



predavač: Alen Egredžija
Jägerova 2, Osijek
#CodeCamp

FACTORY U BROJKAMA



28

EMPLOYEES



2

OFFICES



100+

CLIENTS

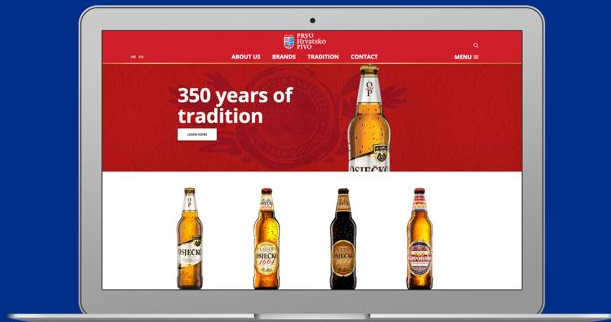
PROJEKTI

Index.me



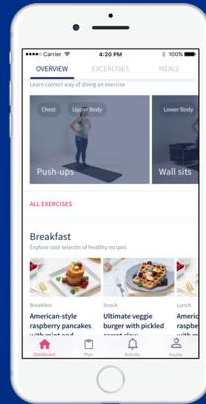
Najpopularnijem news portalu u Hrvatskoj, razvili smo mobilnu aplikaciju za još veće povezivanje sa svojim korisnicima

Osječka pivovara



Razvili smo web rješenje kako bi korisnici što prije došli do najdražeg piva

Sheypme



Mobilna aplikacija za startup koja sadrži personalizirane planove vježbanja i program jela

Node.js

Open-source, cross-platform
JavaScript run-time environment

Prva verzija - Svibanj 2009

Ryan Dahl

Node.js

“JavaScript everywhere” paradigm
event-driven, asinkroni I/O

npm

JavaScript package manager

mogućnost dijeljenja, instalacije, distribucije i organizacije paketa ("dependency-a")

npm

Tri komponente:

1. Website
2. Command Line Interface (CLI)
3. Registrar

Instalacija

(Node.js + npm)

<https://nodejs.org/en>

Kreiranje projekta

Unutar konzole:

```
npm init
```

ili

```
npm init -y
```

("default", bez dodatnih pitanja prilikom inicijalizacije)

package.json

“manifest” za Node.js ekosistem

<https://docs.npmjs.com/getting-started/using-a-package.json>

Webpack

“Module bundler” napisan za moderne Javascript aplikacije

Vrlo “**podesiv**”

Verzija **4**

Instalacija

```
npm install webpack --save-dev
```

i Webpack CLI:

```
npm install webpack-cli --save-dev
```

Webpack 4

“Zero configuration”

```
"scripts":{  
  "build": "webpack"  
}
```

```
npm run build
```

Webpack 4

“environment modes”

```
"scripts":{  
  "dev": "webpack --mode development",  
  "build": "webpack --mode production"  
}
```

Webpack 4

```
"scripts":{  
  "dev": "webpack --mode development ./src/index.js --output ./dist/main.js",  
  "build": "webpack --mode production ./src/index.js --output ./dist/main.js"  
}
```

webpack.config.js

konfiguracijski fajl

Koristi se automatski, mogućnost izmjene:

```
“scripts”: {  
  “build”: “webpack --config ./config/prod.config.js”,  
}
```


webpack.config.js

```
webpack.config.js •
1  const path = require('path');
2
3  module.exports = {
4    entry: [
5      './src/index-1.js',
6      './src/index-2.js',
7    ],
8    output: {
9      path: path.resolve(__dirname, 'dist'),
10     filename: 'main.js'
11   },
12  };
```

webpack loader

“out-of-the box” JavaScript;

preprocessing ostalih tipova datoteka

webpack plugin

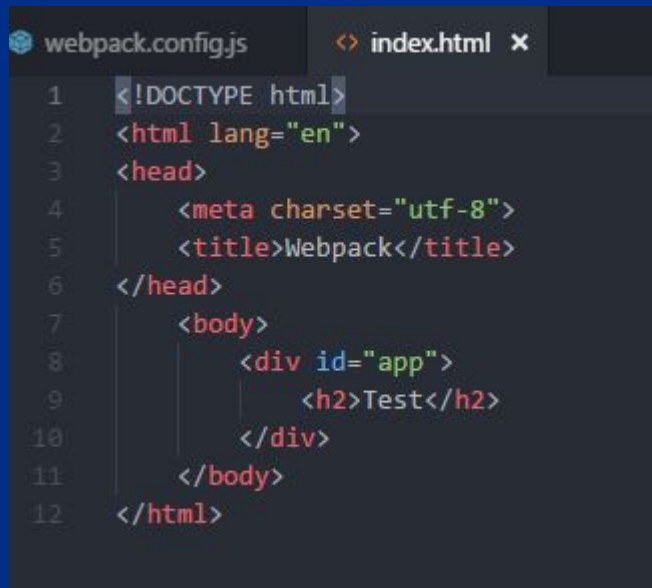
Webpack je praktički “izgrađen” na plugin-ima
“odrađuju” sve što loader ne može

Webpack & HTML

```
npm install html-webpack-plugin html-loader --save-dev
```

```
1  const HtmlWebPackPlugin = require("html-webpack-plugin");
2  const path = require('path');
3
4  module.exports = {
5    entry: [
6      './src/index-1.js',
7      './src/index-2.js',
8    ],
9    output: {
10      path: path.resolve(__dirname, 'dist'),
11      filename: 'main.js'
12    },
13    module: {
14      rules: [
15        {
16          test: /\.html$/,
17          use: [
18            {
19              loader: "html-loader",
20              options: { minimize: false }
21            }
22          ]
23        },
24      ]
25    },
26    plugins: [
27      new HtmlWebPackPlugin({
28        template: "./src/index.html",
29        filename: "./index.html"
30      })
31    ],
32  };
```

./src/index.html



The image shows a code editor with two tabs: 'webpack.config.js' and 'index.html'. The 'index.html' tab is active, displaying the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="utf-8">
5     <title>Webpack</title>
6   </head>
7   <body>
8     <div id="app">
9       <h2>Test</h2>
10    </div>
11  </body>
12 </html>
```

Webpack & HTML

```
npm run build
```

Webpack & CSS

```
npm install style-loader css-loader --save-dev
```


webpack.config.js • {} package.json <> index.html

```
1  const HtmlWebpackPlugin = require("html-webpack-plugin");
2  const path = require('path');
3
4  module.exports = {
5    entry: [
6      './src/index-1.js',
7      './src/index-2.js',
8    ],
9    output: {
10     path: path.resolve(__dirname, 'dist'),
11     filename: 'main.js'
12   },
13   module: {
14     rules: [
15       {
16         test: /\.html$/,
17         use: [
18           {
19             loader: "html-loader",
20             options: { minimize: false }
21           }
22         ]
23       },
24       {
25         test: /\.css$/,
26         use: [ 'style-loader', 'css-loader' ]
27       }
28     ]
29   },
30   plugins: [
31     new HtmlWebpackPlugin({
32       template: "./src/index.html",
33       filename: "./index.html"
34     })
35   ],
36 };
```

webpack.config.js # style.css x <> index.html

```
1 body{
2   background-color: blue;
3 }
4
```

webpack.config.js JS index-1.js x <> index.html

```
1 import './style.css';
2
3 alert('data-1');
```

Webpack & CSS

```
npm run build
```

webpack-dev-server

development server sa live reloadom

```
npm install webpack-dev-server --save-dev
```

```
"scripts":{  
  "start": "webpack-dev-server --mode development  
--open",  
  "build": "webpack --mode production"  
}
```

webpack-dev-server

```
devServer: {  
  open: true,  
  port: 3000,  
  contentBase: './dist'  
},
```



webpack.config.js



package.json x



index.html

```
1  {
2    "name": "js-workshop",
3    "version": "1.0.0",
4    "description": "Simple showcase application",
5    "scripts": {
6      "start": "webpack-dev-server --mode development",
7      "build": "webpack --mode production"
8    },
9    "keywords": [
10     "webpack",
11     "npm",
12     "sass",
13     "babel"
14   ],
15   "author": "Egredžija Alen",
16   "license": "ISC",
17   "dependencies": {
18     "npm": "^6.1.0",
19     "update": "^0.7.4",
20     "webpack": "^4.12.0"
21   },
22   "devDependencies": {
23     "css-loader": "^0.28.11",
24     "html-loader": "^0.5.5",
25     "html-webpack-plugin": "^3.2.0",
26     "style-loader": "^0.21.0",
27     "webpack-cli": "^3.0.4",
28     "webpack-dev-server": "^3.1.4"
29   }
30 }
31
```

```
webpack.config.js x {} package.json <> index.html
1  const HtmlWebpackPlugin = require("html-webpack-plugin");
2  const path = require('path');
3
4  module.exports = {
5    entry: [
6      './src/index-1.js',
7      './src/index-2.js',
8    ],
9    output: {
10      path: path.resolve(__dirname, 'dist'),
11      filename: 'main.js'
12    },
13    module: {
14      rules: [
15        {
16          test: /\.html$/,
17          use: [
18            {
19              loader: "html-loader",
20              options: { minimize: false }
21            }
22          ]
23        },
24        {
25          test: /\.css$/,
26          use: [ 'style-loader', 'css-loader' ]
27        }
28      ]
29    },
30    plugins: [
31      new HtmlWebpackPlugin({
32        template: "./src/index.html",
33        filename: "./index.html"
34      })
35    ],
36    devServer: {
37      open: true,
38      port: 3000,
39      contentBase: './dist'
40    },
41  };
```


Izvori

<https://nodejs.org/en/>

<https://www.npmjs.com/>

<https://webpack.js.org/>

TRAŽITE PRAKSU?



BACKEND
DEVELOPER



WORDPRESS
DEVELOPER



FRONTEND
DEVELOPER

prijavi se na **factory.hr/jobs**

Kontakti: alen.egredzija@factory.hr

<http://factory.hr/blog/>

<https://github.com/factoryhr/education> - materijali za učenje

Pitanja?

Package.json; razlika između dependency-a i devDependency-a?

Koje environment modove webpack koristi i po čemu se razlikuju?

Što nam govori --open zastavica kod webpack-dev-server-a?

Priprema za sljedeću radionicu

Pogledati SASS/SCSS preprocesore i BABEL te ES6 (ECMAScript 2015).