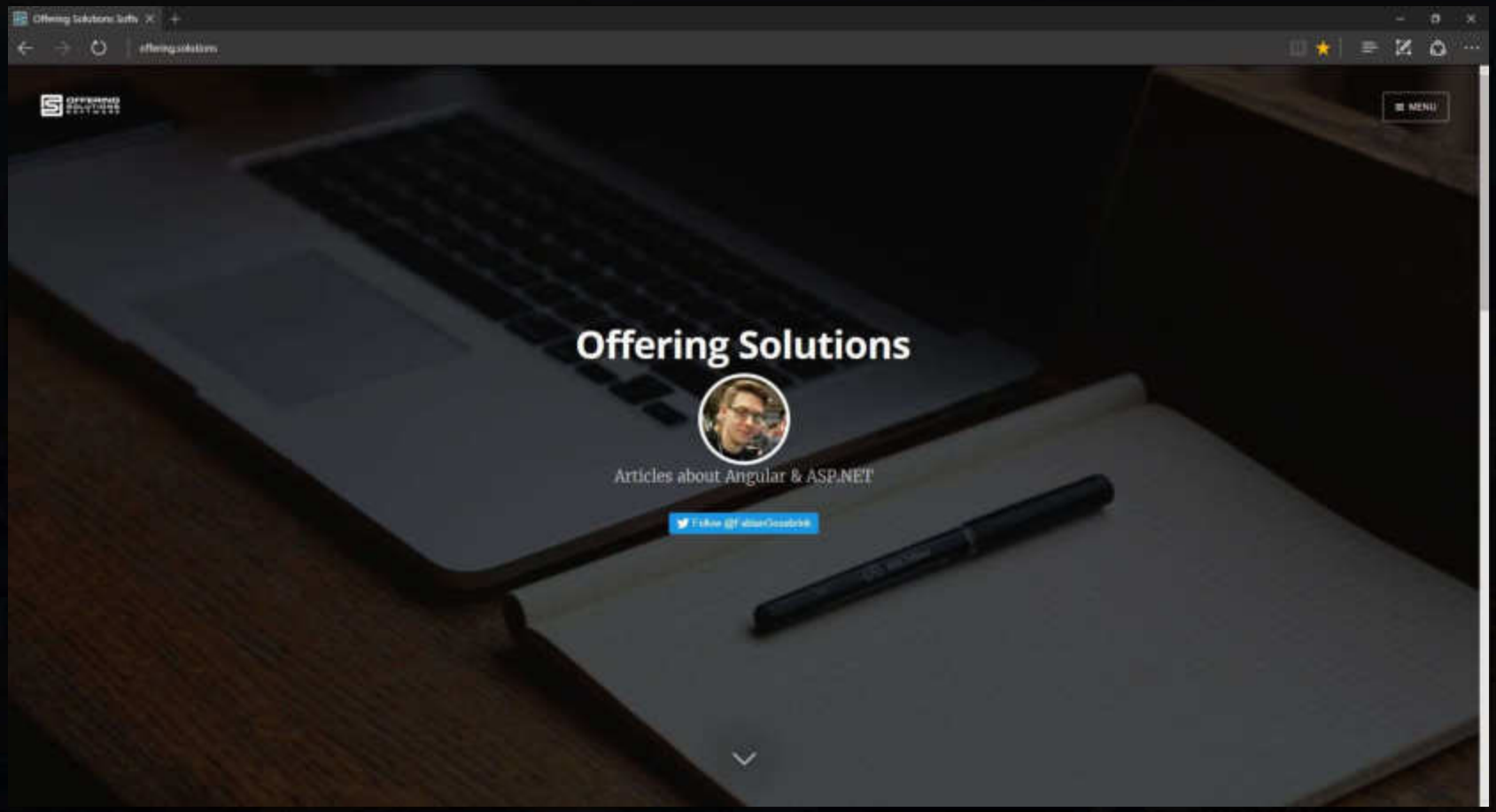
Business applications

for every system with

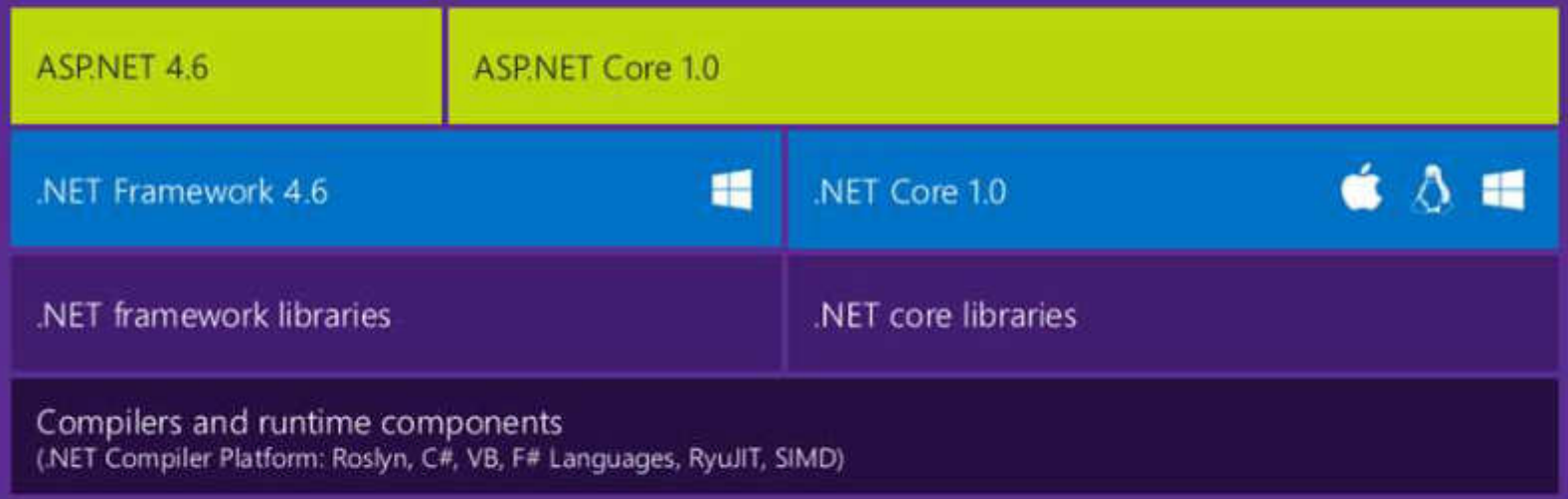
ASP.NET Core

&

@FabianGosebrink


Offering Solutions
MENU
Offering Solutions
Articles about Angular & ASP.NET

ASP.NET 4.6 and ASP.NET Core 1.0



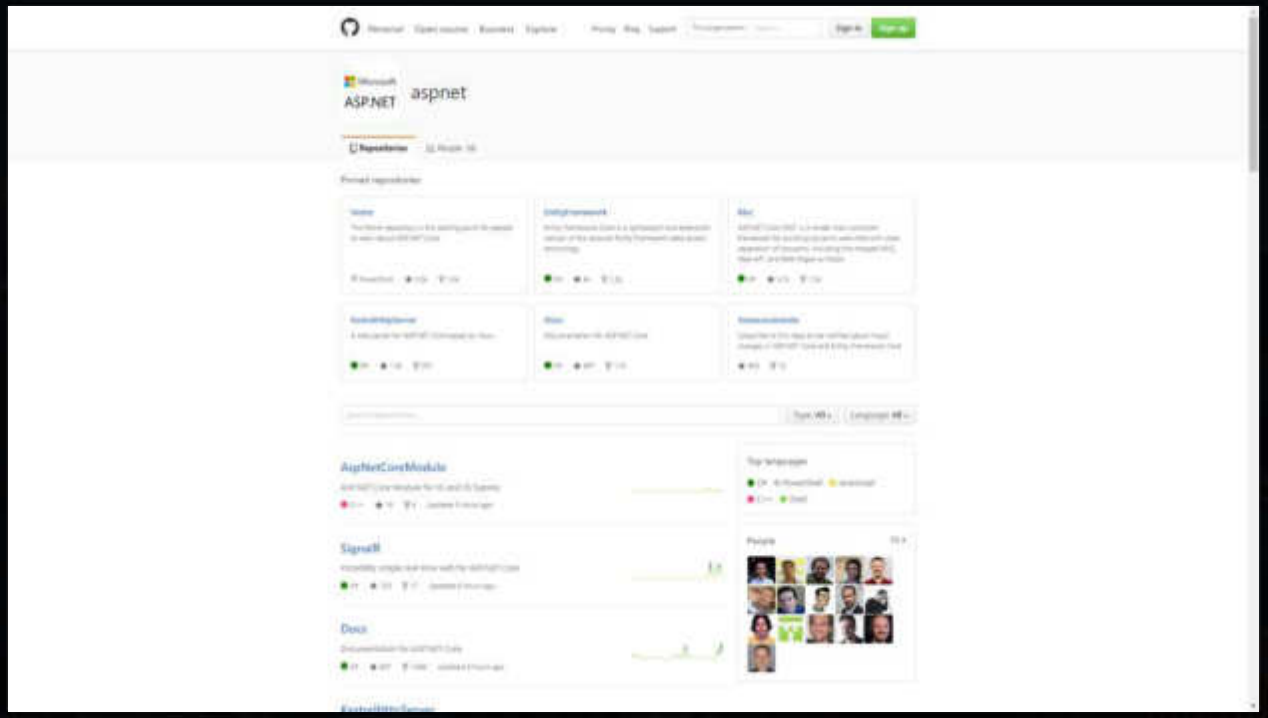
MVC + WebAPI + Web Pages

ASP.NET Core

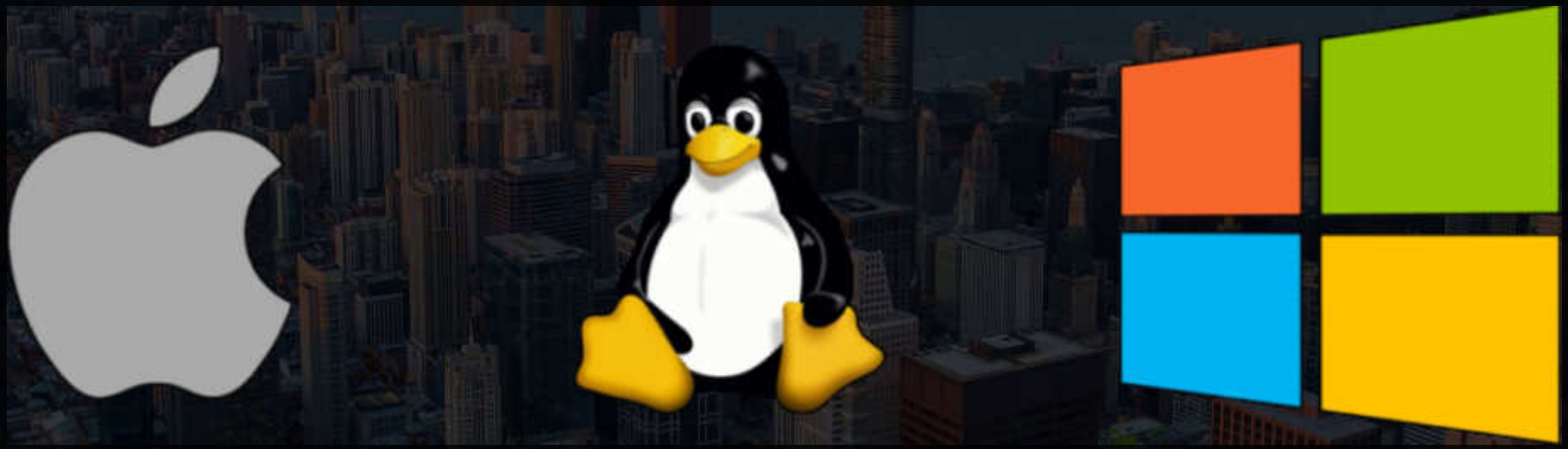

Modular

Faster Deployment Cycles

Open Source

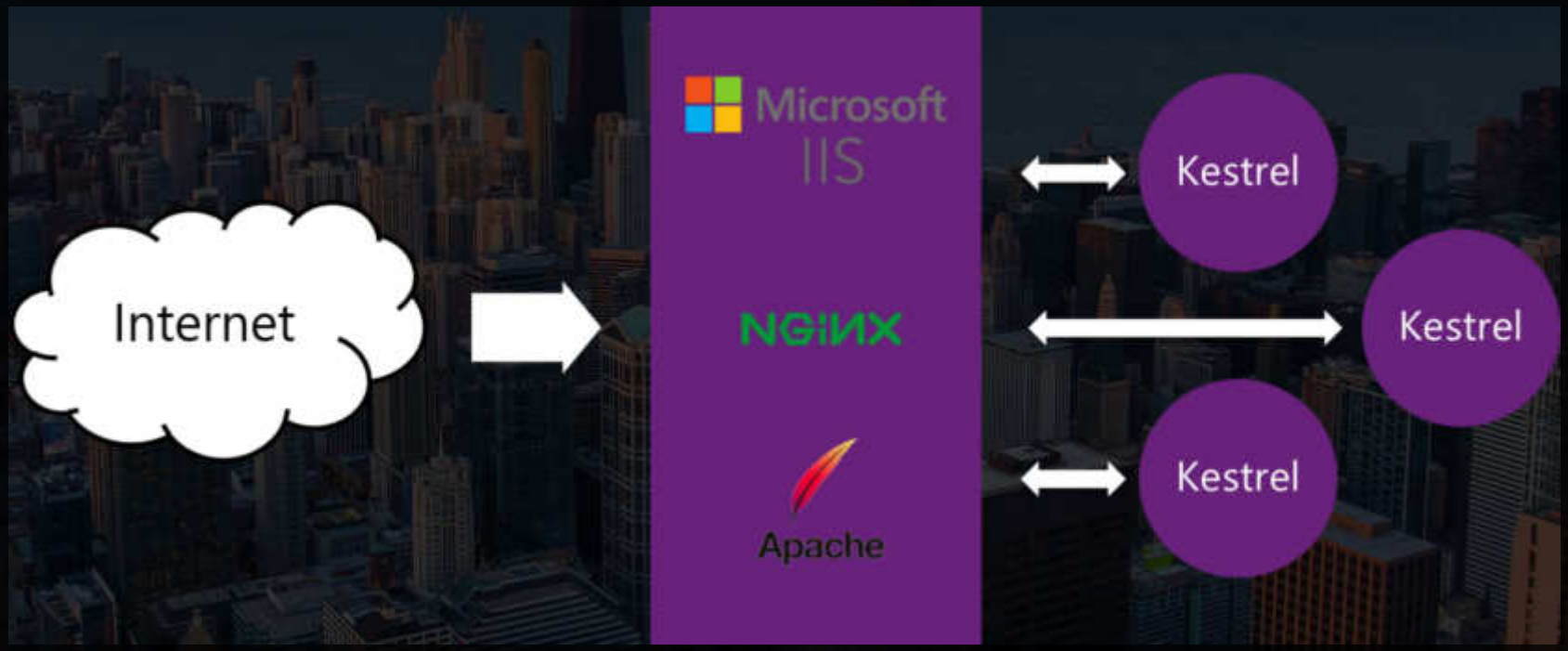


http://github.com/aspnet



Tooling

Kestrel


Internet
Microsoft
IIS
NGINX
Apache
Kestrel
Kestrel
Kestrel

dotnet cli

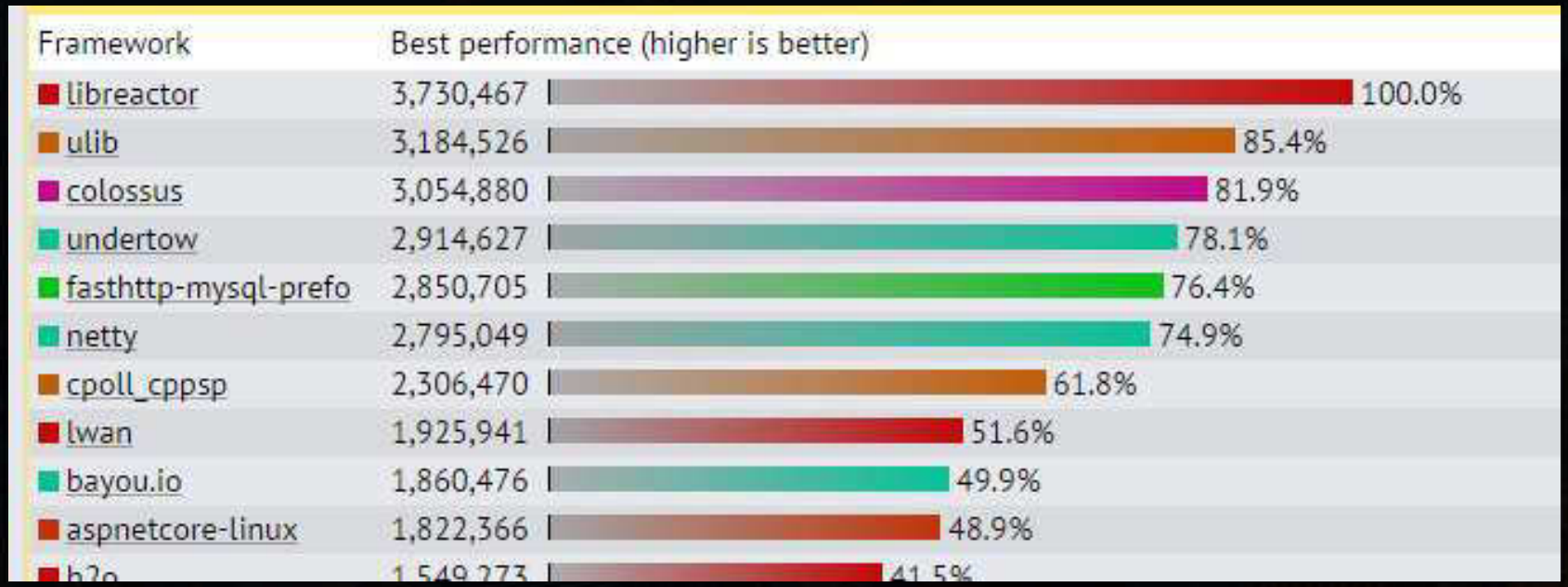
```
using System;

namespace console
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
        }
    }
}
```

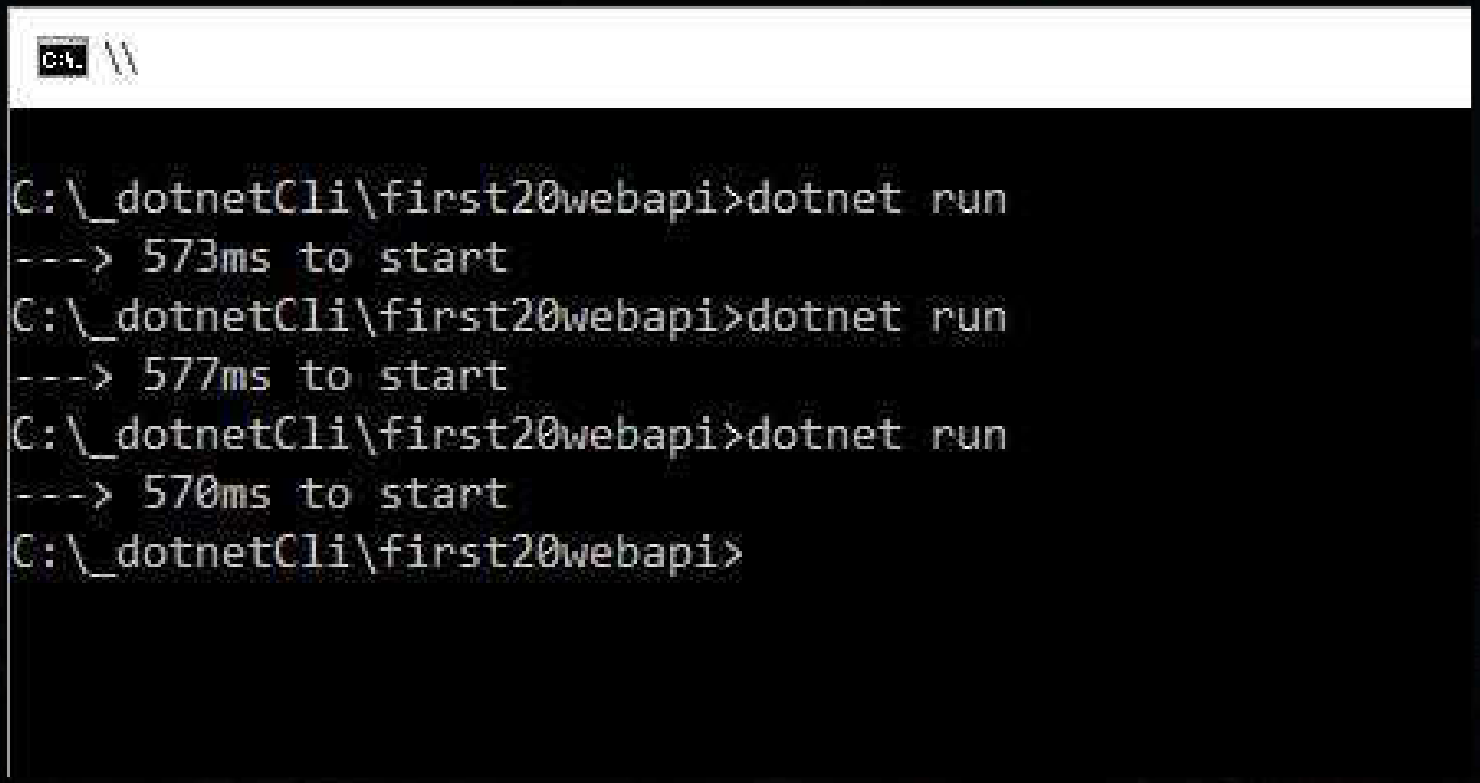
dotnet cli

```
public class Program
{
    public static void Main(string[] args)
    {
        BuildWebHost(args).Run();
    }

    public static IWebHost BuildWebHost(string[] args) =>
        WebHost.CreateDefaultBuilder(args)
            .UseStartup<Startup>()
            .Build();
}
```

```
public static IWebHostBuilder CreateDefaultBuilder(string[] args)
{
    var builder = new WebHostBuilder()
        .UseKestrel()
        .UseContentRoot(Directory.GetCurrentDirectory())
        .ConfigureAppConfiguration((hostingContext, config) => { /*...*/ })
        .ConfigureLogging((hostingContext, logging) => { /*...*/ })
        .UseIISIntegration()
        .UseDefaultServiceProvider((context, options) => { /*...*/ })
        .ConfigureServices(services =>
        {
            services.AddTransient</*...*/>();
        });

    return builder;
}
```


Speeed


Framework
Best performance (higher is better)
libreactor 3,730,467 100.0%
ulib 3,184,526 85.4%
colossus 3,054,880 81.9%
undertow 2,914,627 78.1%
fasthttp-mysql-prefo 2,850,705 76.4%
netty 2,795,049 74.9%
cpoll_cppsp 2,306,470 61.8%
lwan 1,925,941 51.6%
bayou.io 1,860,476 49.9%
aspnetcore-linux 1,822,366 48.9%
h2o 1,549,273 41.5%


\\
C:_dotnetCli\first20webapi>dotnet run
---> 573ms to start
C:_dotnetCli\first20webapi>dotnet run
---> 577ms to start
C:_dotnetCli\first20webapi>dotnet run
---> 570ms to start
C:_dotnetCli\first20webapi>


BBC
MAKE YOUR GIF AT YOUREPEAT.COM

Configuration based on Environment

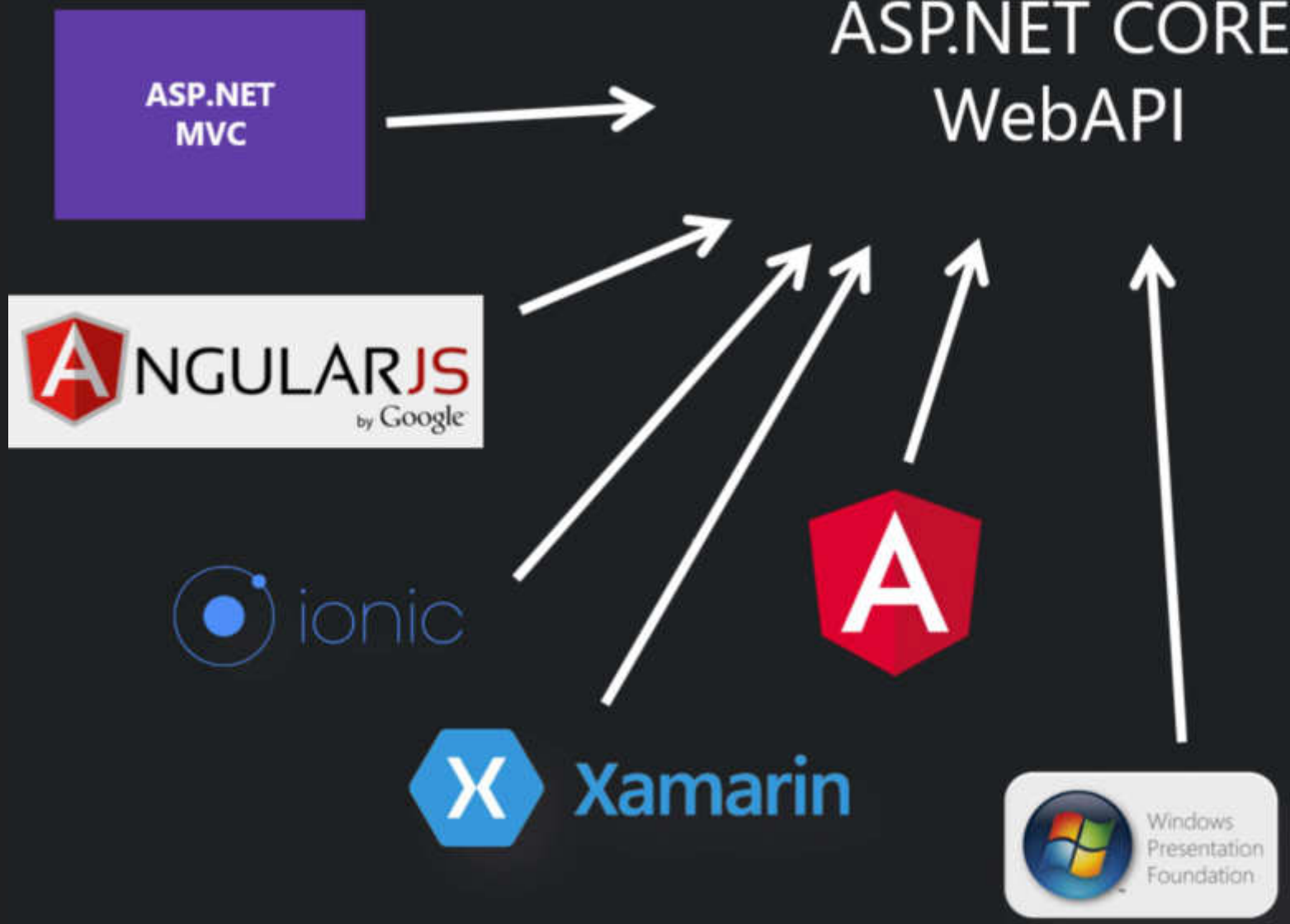
Dependency Injection

Startup.cs

```
1. public class Startup
2. {
3.     public Startup(IConfiguration configuration)
4.     {
5.         Configuration = configuration;
6.     }
7.
8.     public IConfiguration Configuration { get; }
9.
10.    public void ConfigureServices(IServiceCollection services)
11.    {

12.            services.AddScoped<IRepository, Repository>();
```

REST API

```
1. [Route("api/[controller]")]
2. public class FoodsController : Controller
3. {
4.     private readonly IFoodRepository _foodRepository;
5.
6.     public FoodsController(IFoodRepository foodRepository)
7.     {
8.         _foodRepository = foodRepository;
9.     }
10.
11.    [HttpGet]

12.    public IActionResult Get()
```


ASP.NET
MVC
ASP.NET CORE
WebAPI
ANGULARJS
by Google
ionic
Xamarin
Windows
Presentation
Foundation

any

data

any

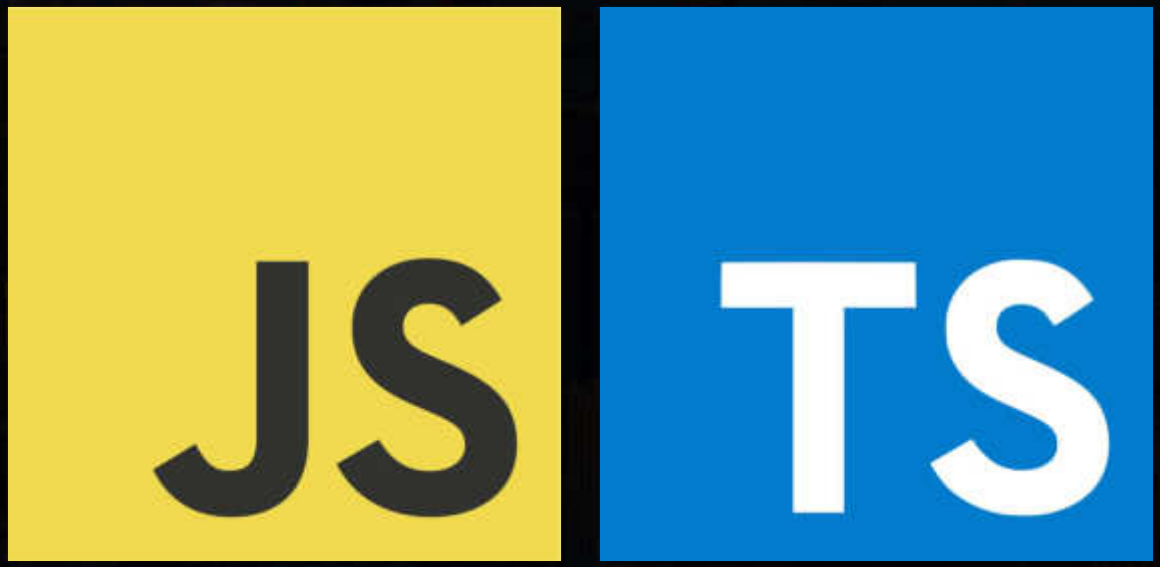
time

any

where

Web
Desktop
Mobile

Web
Desktop
Mobile


JS
TS

@Component

Stateful & Stateless

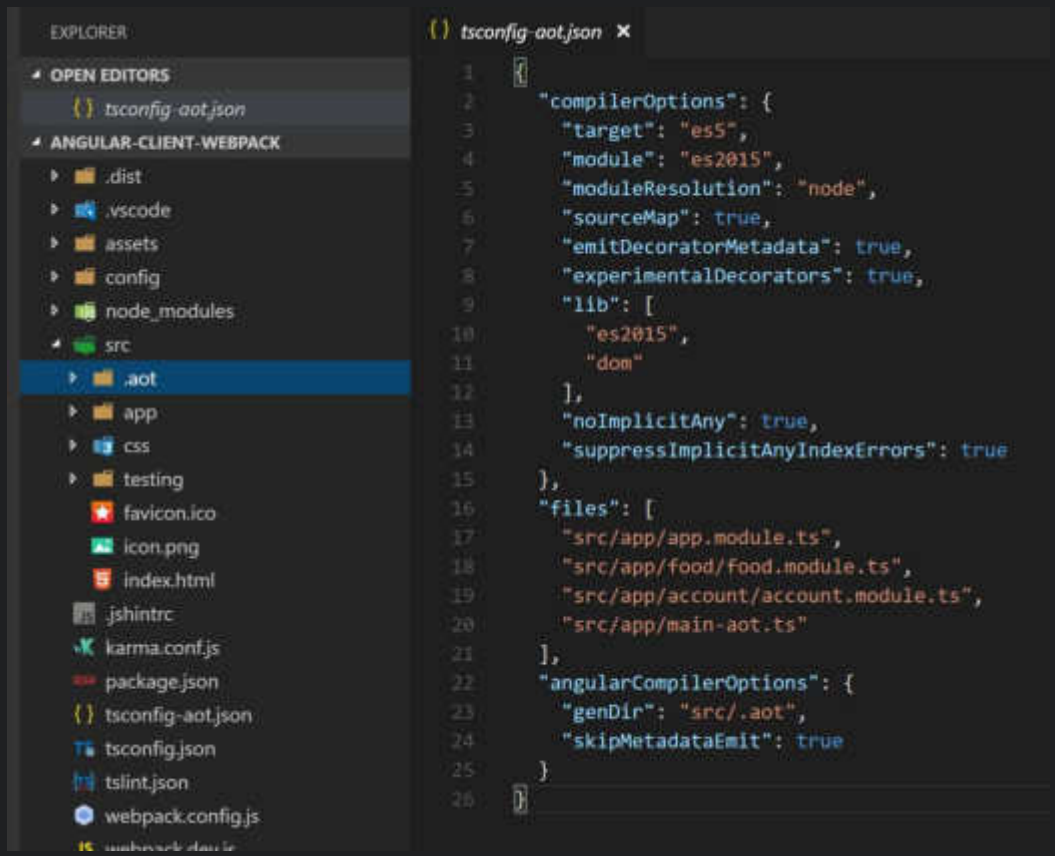
One way dataflow

@NgModule

Dependency Injection

How to get production ready?

Ahead of Time Compilation


EXPLORER
OPEN EDITORS
tsconfig-aot.json
ANGULAR-CLIENT-WEBPACK
.dist
.vscode
assets
config
node_modules
src
.aot
app
css
testing
favicon.ico
icon.png
index.html
.jshintrc
karma.conf.js
package.json
tsconfig-aot.json
tsconfig.json
tslint.json
webpack.config.js
tsconfig-aot.json
{
"compilerOptions": {
"target": "es5",
"module": "es2015",
"moduleResolution": "node",
"sourceMap": true,
"emitDecoratorMetadata": true,
"experimentalDecorators": true,
"lib": [
"es2015",
"dom"
],
"noImplicitAny": true,
"suppressImplicitAnyIndexErrors": true
},
"files": [
"src/app/app.module.ts",
"src/app/food/food.module.ts",
"src/app/account/account.module.ts",
"src/app/main-aot.ts"
],
"angularCompilerOptions": {
"genDir": "src/.aot",
"skipMetadataEmit": true
}
}

package.json

```
{
  "name": "angular-client-webpack-aot-ts",
  "version": "1.0.0",
  "scripts": {
    // ...
    "ngc": "ngc -p tsconfig-aot.json",
    // ...
  },
  "license": "ISC",
  "dependencies": {
    "@angular/compiler-cli": "4.0.3",
    "@angular/platform-server": "4.0.3",
    // ...
  },
  "devDependencies": {
    // ...
```

Treeshaking

Imports
Don't

```
import Rx from "rxjs/Rx";

// Use everything from RxJS here
```

Imports

Do

```
import { Observable } from "rxjs/Observable";
import "rxjs/add/observable/interval";
import "rxjs/add/operator/take";
import "rxjs/add/operator/map";
import "rxjs/add/operator/bufferCount"
```

Lazy Loading

Lazy Loading - Angular

```
import { Routes } from '@angular/router';
import { AuthGuard } from './shared/guards/authentication.guard';

export const AppRoutes: Routes = [
    { path: '', redirectTo: 'home', pathMatch: 'full' },
    { path: 'food', loadChildren: './food/food.module#FoodModule' },
    { path: 'account', loadChildren: './account/account.module#AccountModule' },
    {
      path: '**',
      redirectTo: 'home'
    }
];

```


webpack

Treeshaking - webpack.config

```
module.exports = {
    entry: {
        'app': './src/app/main-aot.ts' // AoT compilation
    },

    output: {
        // ...
    },

    resolve: {
        // ...
    },

    module: {
        rules: [
            // ...
```

Compression

```
module.exports = {
    // ...
    plugins: [
        // ...
        new CompressionPlugin()
    ]
};
```

1-14e52c5a.chunk.js	200	script	app-14e52c5a.b...	3.3 KB	162 ms
app-14e52c5a.bundle.js	200	script	(index)	183 KB	440 ms

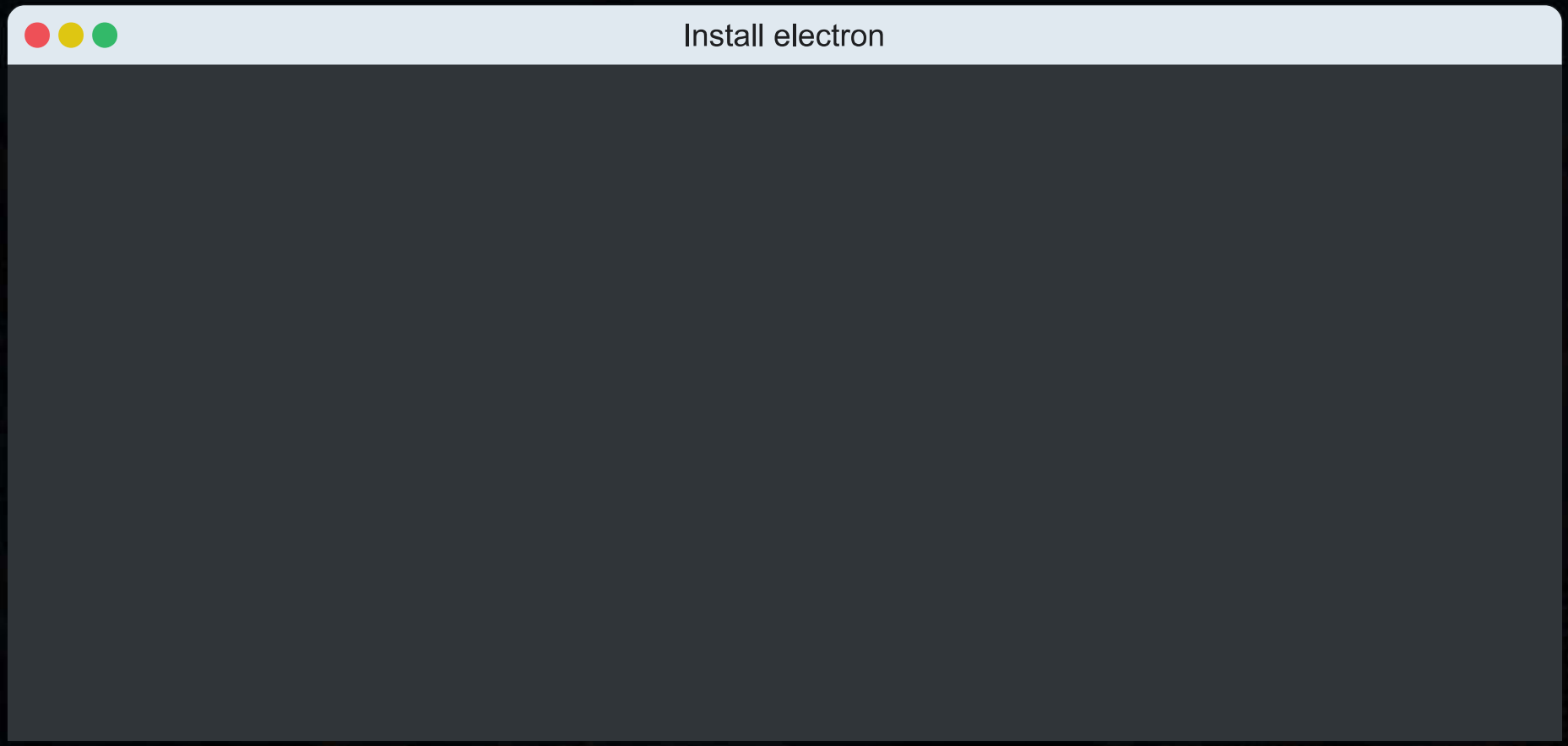
Web
Desktop
Mobile


ELECTRON

Why

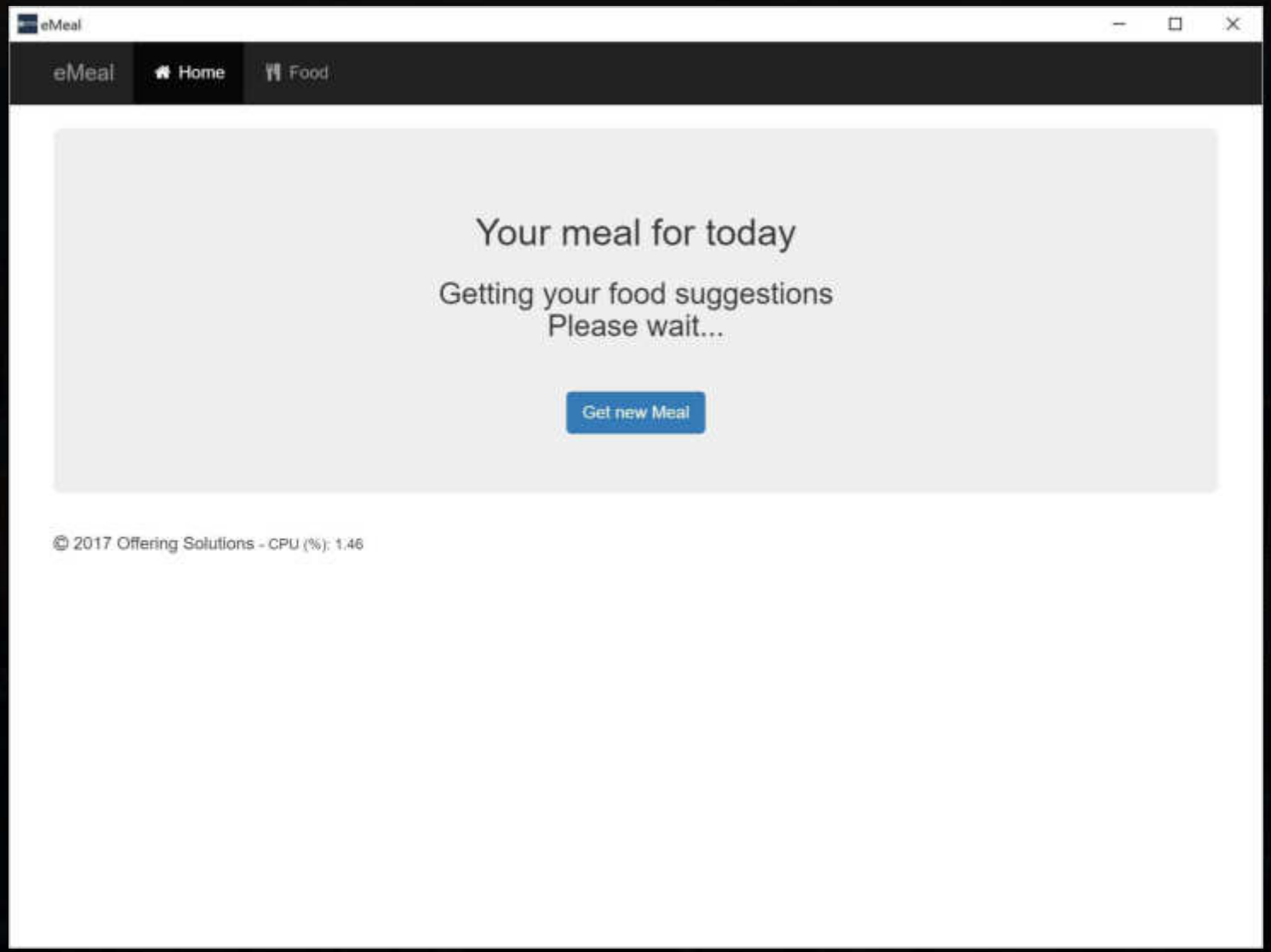
- web technology
- web developers & skills
- aaand...


HTML
CSS
JS
Renderer Process
IPC
node
JS
Main Process
NodeJS
Electron API
Operating System


Install electron

index.js

```
1. const { app, BrowserWindow, globalShortcut, Menu } = require('electron');
2.
3. const path = require('path');
4. const url = require('url');
5. const cpuValues = require('./cpuValues');
6. const trayIcon = require('./trayIcon');
7.
8. let mainWindow = null;
9.
10. app.on('window-all-closed', function () {
11.   if (process.platform !== 'darwin') {
12.     app.quit();
13.   }
14. });
15.
16. app.on('ready', function () {
```


eMeal
eMeal
Home
Food
Your meal for today
Getting your food suggestions
Please wait...
Get new Meal
© 2017 Offering Solutions - CPU (%): 1.46

Notifications

```
showNotification(type: MessageType, title: string, message: string, ...): void {
    if (!Notification) {
        alert('Desktop notifications not available in your browser. Try Chromium.');
        return;
    }

    let messageBody: NotificationOptions = {};

    messageBody.body = message;

    if (icon) {
        messageBody.icon = icon;
    }

    let titleToShow = MessageType[type] + ': ' + title;

```

Send values

```
1. const cpuValues = require('./cpuValues');
2.
3. app.on('ready', function () {
4.
5.   mainWindow = new BrowserWindow({
6.     width: 1024,
7.     height: 768,
8.   });
9.
10.  mainWindow.loadURL('file://' + __dirname + '/index.html');
11.
12.  // ...
13.  startSendCpuValues();
14. });
15.
16. let startSendCpuValues = () => {
```

Receive values

```
1. import { Injectable, NgZone, EventEmitter } from '@angular/core';
2. import { ElectronService } from 'ngx-electron';
3. import { PlatformInformationProvider } from './platformInformation.provider';
4.
5. @Injectable()
6. export class CpuValueService {
7.
8.     public onNewCpuValue = new EventEmitter<string>();
9.
10.    // ...
11.
12.    private registerCpuEvent() {
13.        if (this.electronService.ipcRenderer) {
14.            this.electronService.ipcRenderer.on('newCpuValue', (event, data) => {
15.                // console.log(data);
16.                this.onNewCpuValue.emit(data);
```

But you have variation points!

- My third me ... to my second me

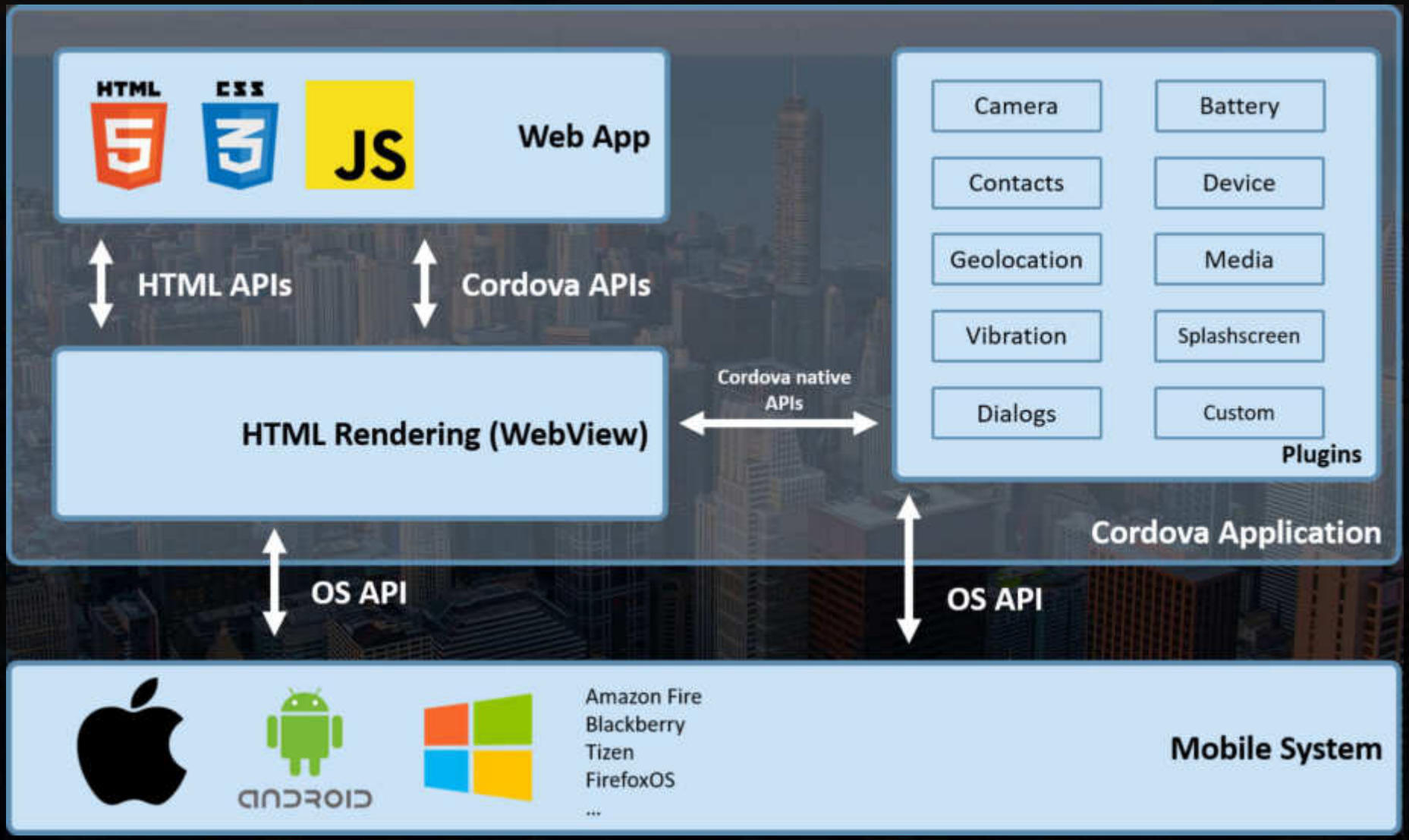
Interface / Abstract class

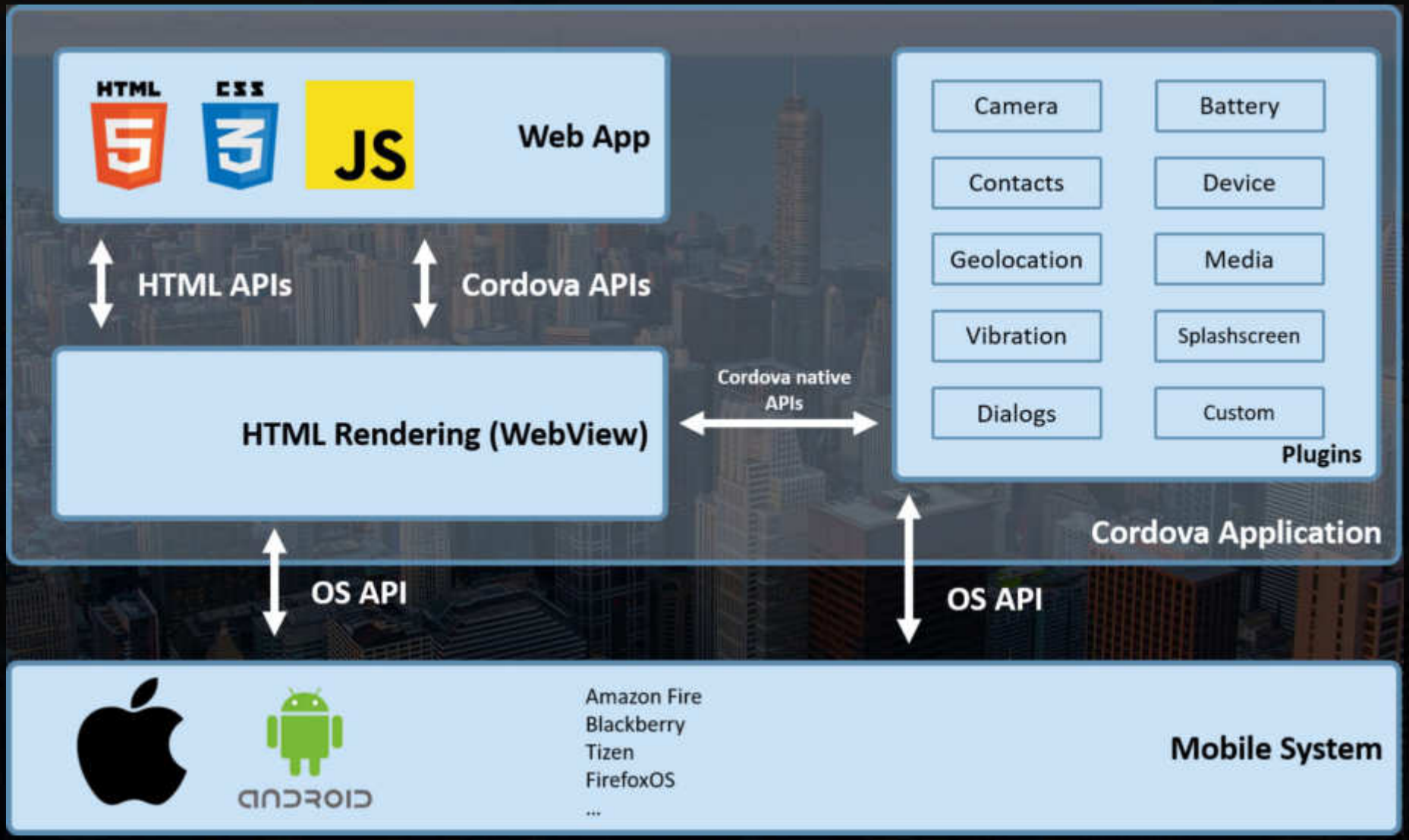
```
export function notificationFactory(): AbstractNotificationService {
    let platformProvider: // ...

    if (platformProvider.isElectron) {
        return new DesktopNotificationService();
    }

    return new WebAndMobileNotificationService();
};

interface INotificationService {
    showNotification(...): void;
}

export abstract class AbstractNotificationService implements INotificationService {
    abstract showNotification(...): void;
```

DI & Factory pattern

```
1. import {
2.     AbstractNotificationService,
3.     notificationFactory
4. } from './services/notification.service';
5.
6. // ...
7.
8. providers: [
9.     // ...
10.    {
11.        provide: AbstractNotificationService,
12.        useFactory: notificationFactory
13.    }
14. ]
```

Web
Desktop
Mobile

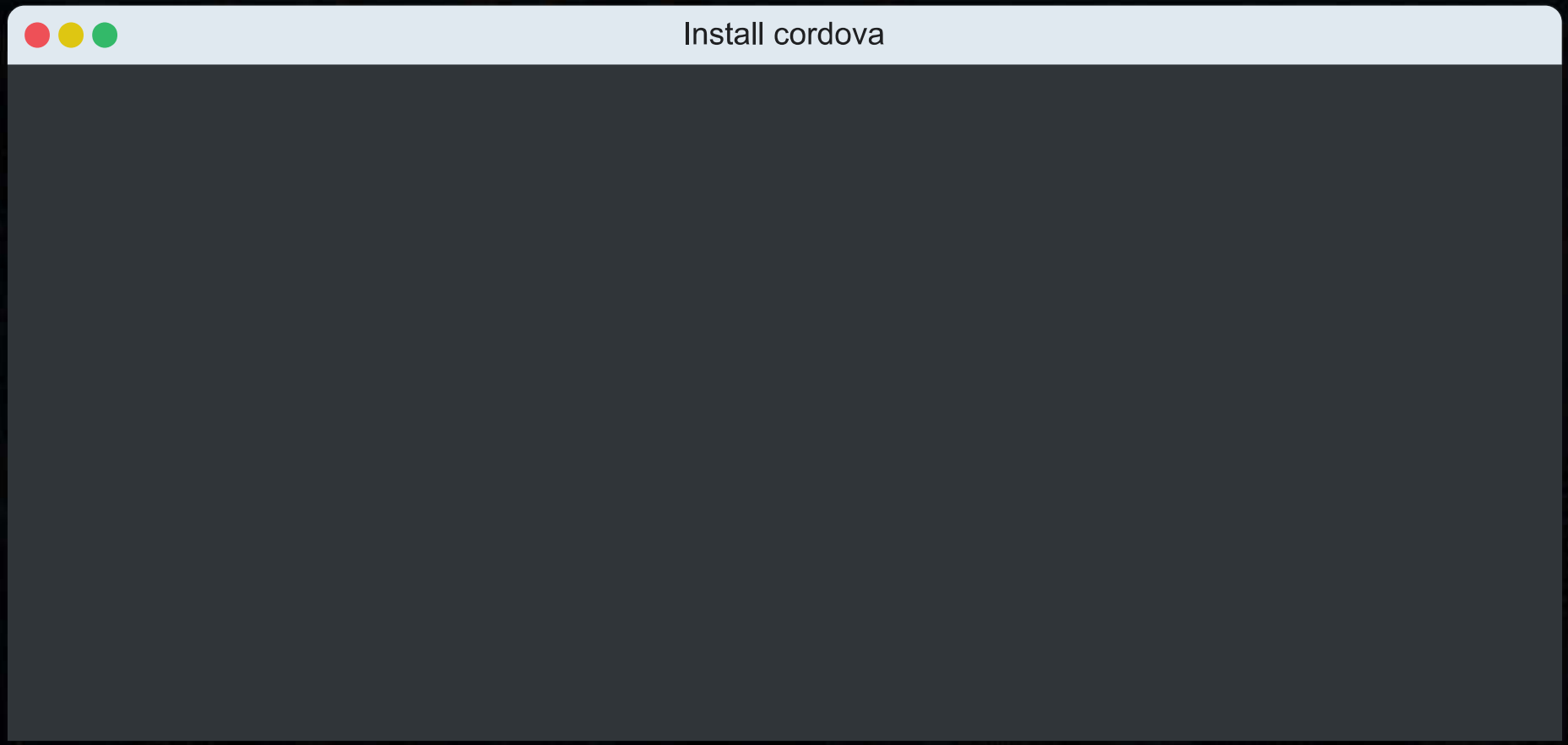

HTML
CSS
JS
Web App
HTML APIs
Cordova APIs
HTML Rendering (WebView)
Cordova native APIs
Camera
Battery
Contacts
Device
Geolocation
Media
Vibration
Splashscreen
Dialogs
Custom
Plugins
Cordova Application
OS API
OS API
android
Amazon Fire
Blackberry
Tizen
FirefoxOS
...
Mobile System

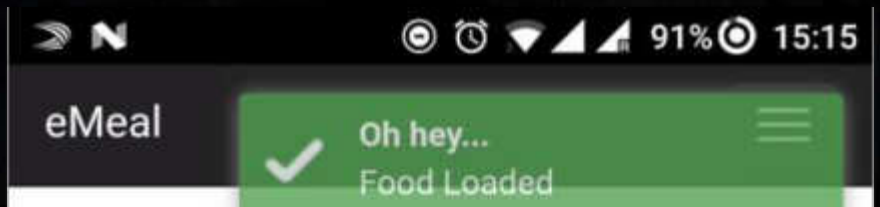

HTML
CSS
JS
Web App
HTML APIs
Cordova APIs
HTML Rendering (WebView)
Cordova native APIs
Camera
Battery
Contacts
Device
Geolocation
Media
Vibration
Splashscreen
Dialogs
Custom
Plugins
Cordova Application
OS API
OS API
ANDROID
Amazon Fire
Blackberry
Tizen
FirefoxOS
...
Mobile System

Cordova config

```
<?xml version="1.0" encoding="utf-8"?>
    <widget xmlns="http://www.w3.org/ns/widgets"
    xmlns:cdv="http://cordova.apache.org/ns/1.0"
    id="offering.solutions" version="0.0.1">

    <name>eMeal</name>
    <description>
        A small demo app to give yoou FoodSuggestions
    </description>
    <author email="mail@offering.solutions"
        href="http://fabian-gosebrink.com">
        Offering.Solutions
    </author>

    <content src="index.html"/>

```

Cordova CLI




91%
15:15
eMeal
Oh hey...
Food Loaded

Demo

The ~~future~~ present

- Offline First
- Progressive Web Apps
- ...


every data
everytime
everywhere


Le Fin

@FabianGosebrink
@SwissAngular

https://swissangular.com
https://offering.solutions
https://github.com/FabianGosebrink

Demos

https://github.com/FabianGosebrink/.NETDay-2017