COMP 431/531: Web Development

Lecture 8: Intro to Angular

Mack Joyner (mjoyner@rice.edu)

https://www.clear.rice.edu/comp431
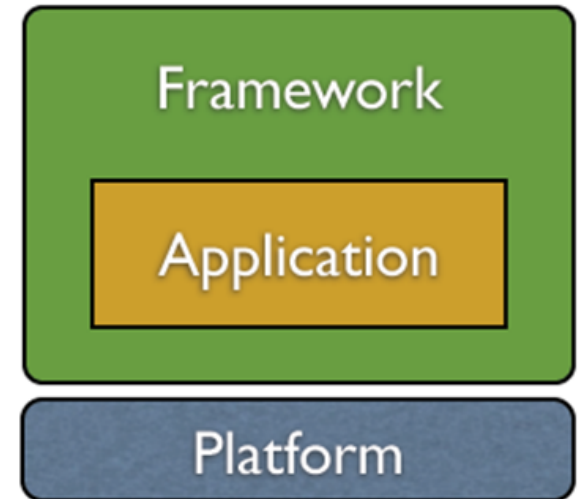
Announcements & Reminders

HW #3 (JavaScript Game) is due Thursday, September 25th at 11:59pm

classroom hw3 repo: https://classroom.github.com/a/wZT10Yqv

Node JS

- 2009 - Invented by Ryan Dahl at Joyent

- 2011 - npm created by Isaac Schlueter
  - included with Node JS

- Built on Google's V8 JavaScript engine

- Executes JavaScript outside of browser

- Single-threaded

- Non-blocking IO

- Node is a platform

Download Node.js

Run JavaScript Everywhere

Node.js® is a free, open-source, cross-platform JavaScript runtime environment that lets developers create servers, web apps, command line tools and scripts.

Get Node.js®

Get security support
for EOL Node.js versions

v22.19.0 Latest LTS   v24.7.0 Latest Release   Trademark Policy

https://nodejs.org/en

Model-View-Controller
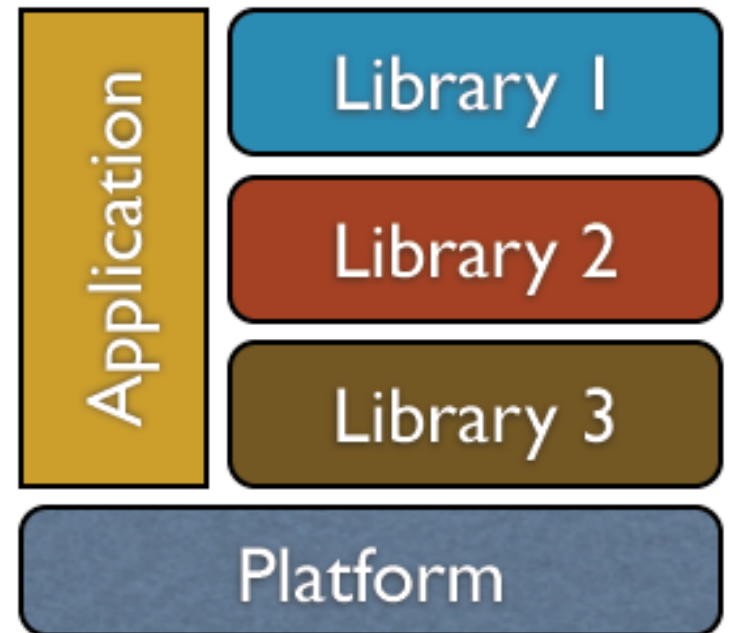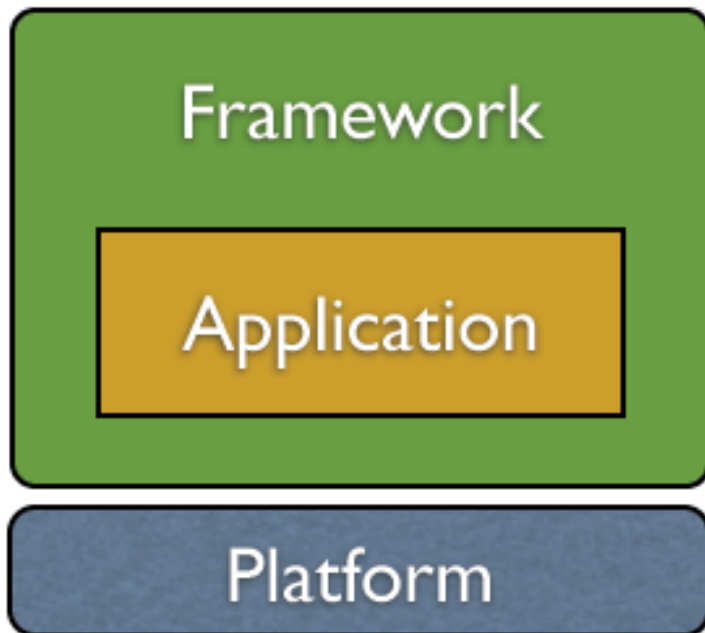
- Model contains the data

- View presents the data to user in response to the Model

- Controller sends commands to update the Model (change data) or the View

Front-End JavaScript Frameworks

- JQuery, Bootstrap, Mustache

- Angular, React*, Vue

~Library

FRAMEWORK

React is more like a library than a framework

Angular

- Angular is a structural framework for dynamic web apps

- AngularJS first released in June 2012
  - Designed by Misko Hevery

- Angular

  - Framework completely rewritten (Angular 2, Sept 2016)

  - Component-based with templates

  - Uses Model-View View-Model (MVVM) rather than MVC

  - Mobile support

  - Version 19 (LTS, released Nov 2024)

Typescript

- JavaScript with types

- Supports ECMAScript 6 (ES 2015)

- Primitives types
  - string
  - boolean
  - number
  - undefined, null

- Non-primitive types
  - array ([], Array)
  - object

- Other types: any, Enum, void, …

- Classes are heavily used in Angular, methods don't need *function* in front

- **https://www.typescriptlang.org/docs/handbook/basic-types.html**

Creating an Angular app

- **Install Angular:** *sudo npm install -g @angular/cli*

- **Create a new application in a directory:** *ng new tictactoe*

```
? Which stylesheet format would you like to use? CSS           [
https://developer.mozilla.org/docs/Web/CSS               ]
? Do you want to enable Server-Side Rendering (SSR) and Static Site Generation
[(SSG/Prerendering)? No
CREATE tictactoe/README.md (1063 bytes)
CREATE tictactoe/.editorconfig (274 bytes)
CREATE tictactoe/.gitignore (587 bytes)
CREATE tictactoe/angular.json (2607 bytes)
CREATE tictactoe/package.json (1040 bytes)
CREATE tictactoe/tsconfig.json (857 bytes)
CREATE tictactoe/tsconfig.app.json (263 bytes)
CREATE tictactoe/tsconfig.spec.json (273 bytes)
CREATE tictactoe/.vscode/extensions.json (130 bytes)
CREATE tictactoe/.vscode/launch.json (470 bytes)
CREATE tictactoe/.vscode/tasks.json (938 bytes)
CREATE tictactoe/src/main.ts (250 bytes)
CREATE tictactoe/src/favicon.ico (15086 bytes)
CREATE tictactoe/src/index.html (295 bytes)
CREATE tictactoe/src/styles.css (80 bytes)
CREATE tictactoe/src/app/app.component.css (0 bytes)
CREATE tictactoe/src/app/app.component.html (19903 bytes)
CREATE tictactoe/src/app/app.component.spec.ts (925 bytes)
CREATE tictactoe/src/app/app.component.ts (305 bytes)
CREATE tictactoe/src/app/app.config.ts (227 bytes)
CREATE tictactoe/src/app/app.routes.ts (77 bytes)
CREATE tictactoe/src/assets/.gitkeep (0 bytes)
✔ Packages installed successfully.
```

Package.json

- File contains build config for app

- Scripts available to build, run, test app

- Dependencies needed for project

- Auto-generated by Angular

- Build and serve app on cmd line (http://localhost:4200)

  >> cd tictactoe

  >> sudo ng serve (use --open option to show in browser)

- Alternatively (npm run start)

- Install everything in package.json

  >> npm install --verbose

IntelliJ uses angular.json to locate scripts

```
1  {
2      "name": "tictactoe",
3      "version": "0.0.0",
4      "scripts": {
5          "ng": "ng",
6          "start": "ng serve",
7          "build": "ng build",
8          "test": "ng test",
9          "lint": "ng lint",
10         "e2e": "ng e2e"
11     },
12     "private": true,
13     "dependencies": {
14         "@angular/animations": "~8.1.2",
15         "@angular/common": "~8.1.2",
16         "@angular/compiler": "~8.1.2",
17         "@angular/core": "~8.1.2",
18         "@angular/forms": "~8.1.2",
19         "@angular/platform-browser": "~8.1.2",
20         "@angular/platform-browser-dynamic": "~8.1.2",
21         "@angular/router": "~8.1.2",
22         "rxjs": "~6.4.0",
23         "tslib": "^1.9.0",
24         "zone.js": "~0.9.1"
25     },
```

```
> ng serve

i ｢wds｣: Project is running at http://localhost:4200/webpack-dev-server/
i ｢wds｣: webpack output is served from /
i ｢wds｣: 404s will fallback to //index.html

chunk {main} main.js, main.js.map (main) 11.6 kB [initial] [rendered]
chunk {polyfills} polyfills.js, polyfills.js.map (polyfills) 251 kB [initial] [rendered]
chunk {runtime} runtime.js, runtime.js.map (runtime) 6.09 kB [entry] [rendered]
chunk {styles} styles.js, styles.js.map (styles) 16.3 kB [initial] [rendered]
chunk {vendor} vendor.js, vendor.js.map (vendor) 4.07 MB [initial] [rendered]
Date: 2019-09-18T21:55:25.371Z - Hash: a65dab538def3fe88034 - Time: 13666ms
** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **
i ｢wdm｣: Compiled successfully.
```

Welcome to tictactoe!

Angular App index.html

- Root page served

- Component defines selector

- Template contains HTML

```
1   <!doctype html>
2   <html lang="en">
3   <head>
4     <meta charset="utf-8">
5     <title>Tictactoe</title>
6     <base href="/">
7
8     <meta name="viewport" content="width=device-width, initial-scale=1">
9     <link rel="icon" type="image/x-icon"
10  </head>
11  <body>
12    <app-root></app-root>
13  </body>
14  </html>
```

Tag defined in root component

Root Component (app.component.ts)

- Component, template (view)

- Creates component selector (html tag)

- Tightly coupled with template

- Contains fields (data) accessible by template

- Replaces tag (selector) in html with template

```
1   import { Component } from '@angular/core';
2   import { RouterOutlet } from '@angular/router';
3
4   @Component({
5     selector: 'app-root',
6     standalone: true,
7     imports: [RouterOutlet],
8     templateUrl: './app.component.html'
9     styleUrl: './app.co
10  })
11  export class AppComponent {
12    title = 'tictactoe';
13  }
```

Template is usually placed in another file

Component has title field that is accessible by the template

View: Root Template (app.component.html)

- Template used by root component

```
<h1>Hello, {{ title }}</h1>
<p>Congratulations! Your app is running. 🎉</p>
```

- Angular supports data binding using templates

Hello, tictactoe

Congratulations! Your app is running. 🎉

- Use all the normal html tags

- Router outlet tag incorporates other component templates based on current route (url)

```
<router-outlet />
```

Generating Angular Components

Generate new component:

```
>> cd tictactoe/
>> ng generate component board
```

```
CREATE src/app/board/board.component.css (0 bytes)
CREATE src/app/board/board.component.html (20 bytes)
CREATE src/app/board/board.component.spec.ts (589 bytes)
CREATE src/app/board/board.component.ts (230 bytes)
```

Board Component

```
1      import { Component } from '@angular/core';
2
3      @Component({
4        selector: 'app-board',
5        standalone: true,
6        imports: [],
7        templateUrl: './board.component.html',
8        styleUrl: './board.component.css'
9      })
10     export class BoardComponent {
11
12     }
```

Routing (app.routes.ts)

- Navigate between components, render different views

```
1    import { Routes } from '@angular/router';
2
3    export const routes: Routes = [];
```

Defines routes relative to base URL

- Uses browser URL to navigate to client view

```
import {BoardComponent} from "./board/board.component";

export const routes: Routes = [{path: '', component: BoardComponent}];
```

Buttons

- Buttons can used inside or outside of forms

- Buttons outside of form

  - Not recognized: *onclick="handleBtnClick()"*

  - Not recognized: *click="handleBtnClick()"* will do nothing

  - Recognized: *(click)="handleBtnClick()"* calls component *handleBtnClick()* method

  - Passing variable arguments to *(click)="handleBtnClick(0)"* is allowed