



# Getting STARTED with Angular

# hi!



## I am Matt Vaughn

Developer, Speaker, Consultant, PodCaster, Musician, Owned by Lukka



@angularlicious



github.com/buildmotion

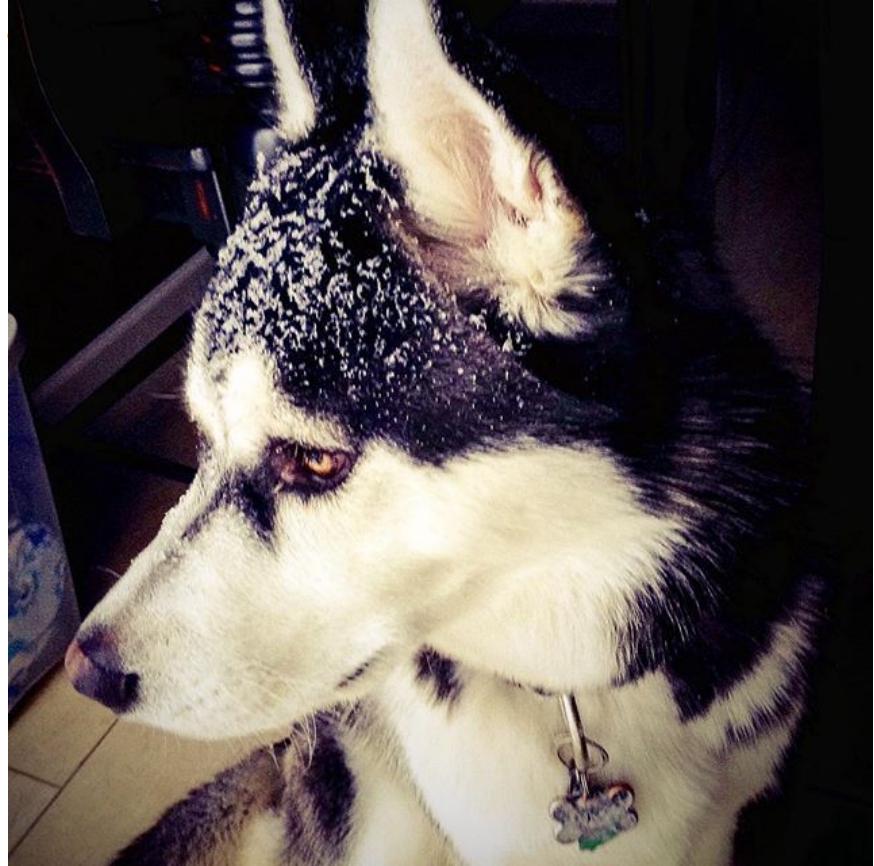


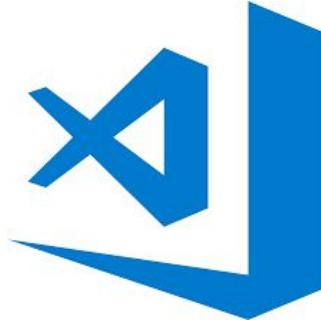
[http://www.angularlicious.us](http://www.angularlicious.com) OR [www.angularlicious.com](http://www.angularlicious.com)

# About ME

Profile:

- Web development since 1998
- .NET/C# since 2001
- Angular 2 since 2016
- Musician (Saxes, EWI)
- Husky: **lukka\_the\_husky**





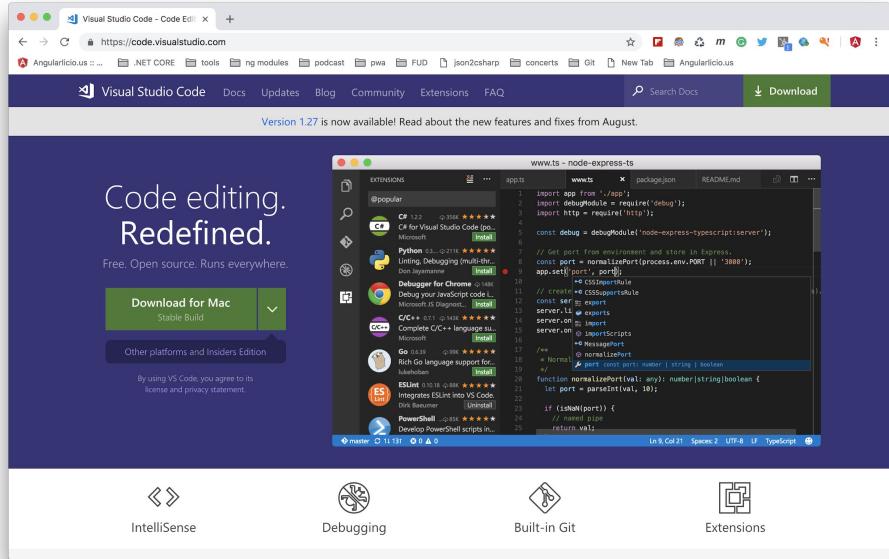
# DEV Environment

Things that you will need to get started



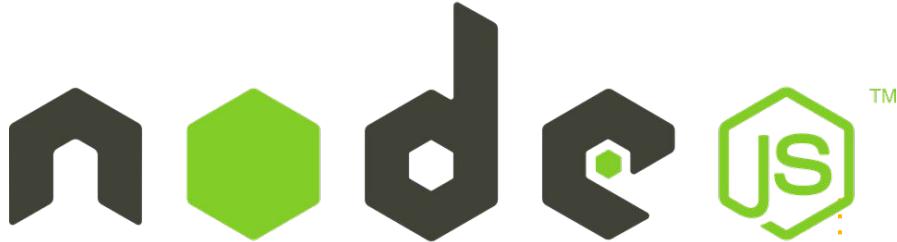
# Visual Studio Code

- Download and install.



# node.js

- Install nodejs...
- **Browsers** require Javascript to run the app - therefore, Typescript needs to be transpiled (which requires Nodejs).



# Install on a MAC?

Use [Brew.sh](#) to install nodejs  
and npm on a Mac.





# Install on a Windows PC?

Just [download](#) and install the latest LTS release.

LTS Recommended For Most Users	Current Latest Features
 Windows Installer <small>node-v8.12.0-x86.msi</small>	 macOS Installer <small>node-v8.12.0.pkg</small>



# Angular CLI

- Install the Angular CLI - globally.
- <https://cli.angular.io/>



```
1 npm install -g @angular/cli
```



# HOW IT WORKS

ANGULAR STUFF



# ng: How it works...

**Definition:** The things that make Angular work.

**Angular** is

- Module-driven
- Enterprise-level Framework
- Design Patterns
- Object-Oriented
- Tooling

**Benefits:**

- Opinionated.
- Well-defined.
- Worldwide communities.
- End-to-end Framework
- Learning Resources
- Conferences

# ng DETAILS



## Angular is

- Module-Driven Platform
- Packages
- ES6
- Strongly Typed
- Services
- HTTP Client
- Routing
- Components
- Provider-based DI

## Tooling:

- CLI
- Schematics
- Build
- Serving
- Testing (e2e, specification)

# ng APPLICATIONS



**Angular** is

- Single Page Application (SPA)
- Progressive Web App (PWA)
- Angular Universal (Server-side Rendering)
- Angular + Ionic (Hybrid Mobile)



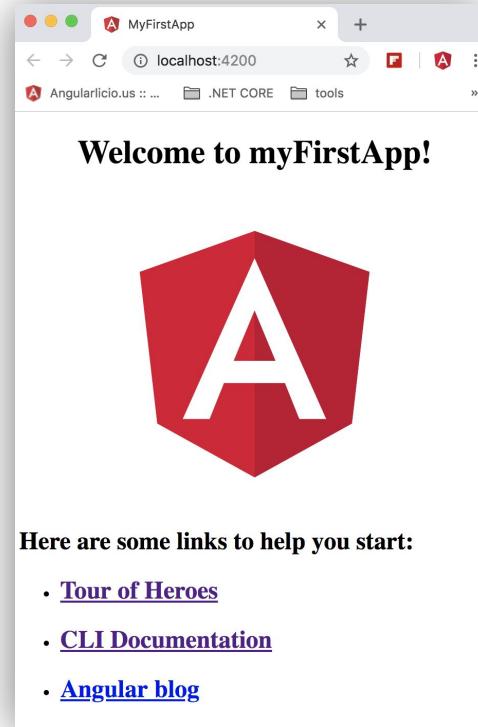
# At the END of the DAY?

WHAT DO WE GET AFTER ALL OF THIS?

# HOW does ng WORK?

A single page application.

- Index.html
- Javascript Bundles
  - Javascript
  - HTML
  - CSS
- Module loader...runtime
- HTTP Client access Web APIs





# BUILDING **blocks**

TOOLS, COMPONENTS, AND FRAMEWORKS

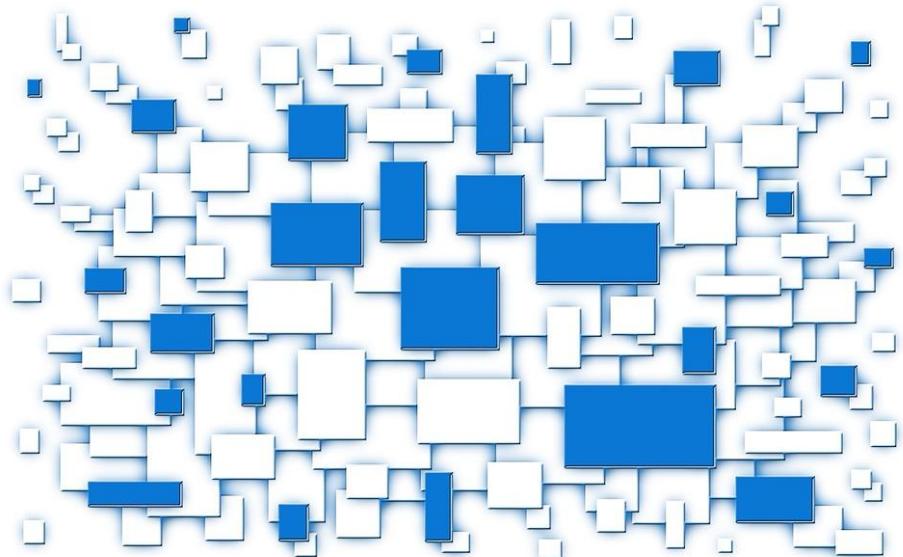


# WHAT is A MODULE?

**Definition:** A module is a collection of ***related*** things - organized together to provide or perform some related functionalities.

**Javascript:** It is part of the ECMAScript specification 2015.

- ES6
- ESM6
- EcmaScript 2015





# WHAT is an ng Module?

**Definition:** A container for components, services, classes, pipes, directives.

**Angular Modules** can *import* and use other modules - and what those modules expose (components, services, classes).

## Benefits:

- Modular-driven
- Encapsulates implementation details.
- Controls what is exposed



# WHAT is a COMPONENT?

**Definition:** A component is a unit that provides functional and visual elements for your application.

**Angular Component:** Is a specialized set of items:

- **Class** with @Component Decorator
- **HTML** → Template/View
- **CSS**

## Benefits:

- Can *extend* a base class.
- Container-Presentational.
- Composable.
- Autonomous/Independent Communication
  - Parent-to-child
  - Child-to-parent
- Data Binding



# WHAT is a SERVICE?

**Definition:** A SERVICE provides sets of functionality (i.e., API), that allows for the encapsulation of logic.

**Angular Service:** Is a specialized class that is injected via DI:

- **Class** with `@Injectable` Decorator
- Provides an **API** to components or other services.

## Benefits:

- Can implement an *interface*.
- Can *extend* a base class.
- Participates in Angular Dependency Injection mechanism.
- Can have configuration/setup data.
- Can be scoped for application, module, service, or component level.



# WHAT is ROUTING?

**Definition:** Mechanism to navigate from one view to another.

**Angular Routing Module:** Is a specialized module with a routing service:

- Configurable routes for navigation.
- URL-based
- Links
- State and Events

## Benefits:

- Built-in mechanism for navigation.
- Route guards to protect sensitive information.



# WHAT is HttpClient?

**Definition:** Mechanism to communicate with backend systems using HTTP.

**Angular HttpClient:** Is a specialized module with capabilities of constructing and handling HTTP requests and responses.

## Benefits:

- Built-in mechanism for HTTP communication.
- HTTP request/response error handling.
- Use to load JSON data
- Stream-based using Observables (rxjs)
- Security (XSRF, Headers)
- Supports mocking

# first app

LET'S DO THIS...





# USE the CLI...





# APPLICATION configuration

CONFIGURATION FOR YOUR ANGULAR APPLICATION



# WHAT is package.json?

**Definition:** A file that lists the npm packages used by your application..

- Dependencies for runtime and development
- Scripts for building, testing, and running

## Benefits:

- Built-in mechanism for HTTP communication.
- HTTP request/response error handling.
- Use to load JSON data
- Stream-based using Observables (rxjs)
- Security (XSRF, Headers)
- Supports mocking



# WHAT is `tsconfig.json`?

**Definition:** TypeScript compiler configuration for the Angular app (`tsconfig.app.json`) and for the unit tests (`tsconfig.spec.json`).

- Compiler Options
- File Targets
  - Includes
  - Excludes

## Resources:

- Learn more at:  
<https://www.typescriptlang.org/docs/handbook/tsconfig-json.html>



# WHAT is tslint.json?

**Definition:** TSLint is an extensible static analysis tool that checks Typescript code for readability, maintainability, and functionality errors.

- Configurable
- Rule based
- Type checking
- Support custom rules

## Resources:

- Learn more at:  
<https://palantir.github.io/tslint/>





# WHAT is angular.json?

**Definition:** Use to define the definition of all projects in an Angular Workspace.

- Add multiple applications
- Add multiple libraries
  - Shared
- Configure each application/library distinctly
- Shared configuration
- Environment Configurations
- Build Configurations

## Resources:

- [Schema Definition](#)
- [Schema Documentation](#)
- [Blog post...](#)



# WHAT is **tsconfig.app.json?**

**Definition:** Provides application-level Typescript build configuration..

- Extend the global tsconfig.json
- Override/customize the configuration items specific to the application.



# WHAT is environment.ts?

**Definition:** A target environment file for environment-based configurations.

- Debug (default)
- Production
- Allows for customized configurations by environment
- Build configuration uses specified target
- Load configuration from JSON files.

## Resources:

- [Deployment](#)
- [Application Settings](#)



# THE BACKEND

EVERY APPLICATION NEEDS A GOOD BACKEND



# Option #1: WEB API

**Definition:** Use Microsoft's ASP.NET Web API to .

- Support for .NET Core
- Uses C#/NET
- Controller-based
- Works well with JSON
- Integrates with Entity Framework

## Resources:

- [Getting Started](#)
- [Building Web APIs with .NET Core](#)
- [Pluralsight: Introduction to ASP.Net Web API](#)



# Option #2: SERVERLESS

**Definition:** A Serverless approach doesn't require a dedicated backend for the application.

- You can use Firebase as a database (BaaS)
  - Supports Functions
- Use Web APIs from any supported platform to compose functionality for the application
  - Contentful, MailChimp, Firebase, Google

## Resources:

- [Angular and Firebase Awesomeness](#)
- Contentful
- SendGrid
- Stripe
- Google
- AWS



# Option #3: OTHER

**Definition:** There are many other options out there.

- Node/Express
- Ruby on Rails
- Django for Python
- Flask (Python)
- Phoenix Framework
- Laravel

## Resources:

- [Node backend](#).
- [List of Backend Frameworks](#)



# HOSTING angular

HOSTING ANGULAR APPLICATION → NOTHING BUT OPTIONS



# Hosting OPTIONS

**Definition:** add content here.

- On-premise
- Azure
- Heroku
- Firebase
- AWS
- Github Pages

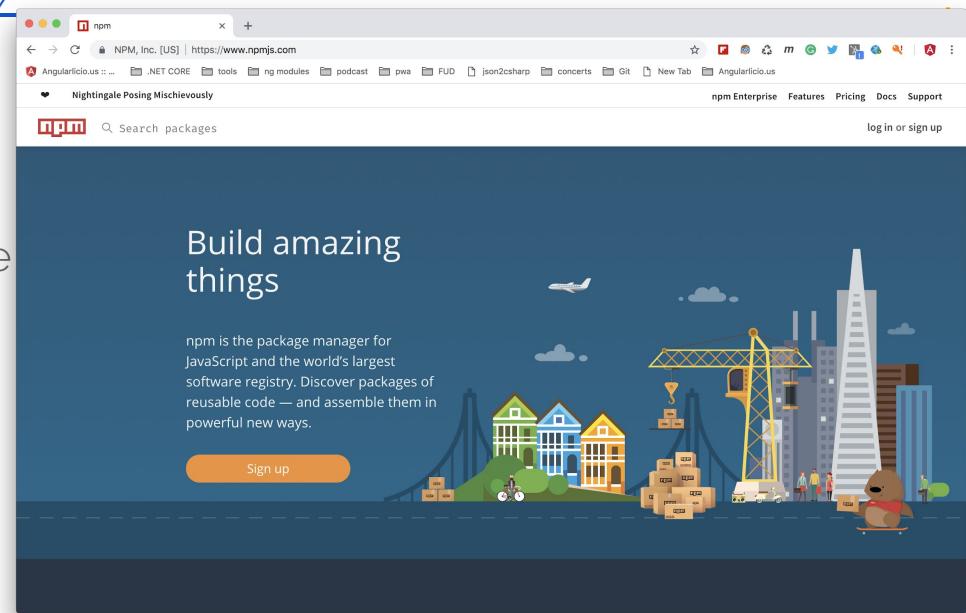


# USEFUL things

OTHER USEFUL THINGS TO HELP YOU BUILD NG APPS

# npm

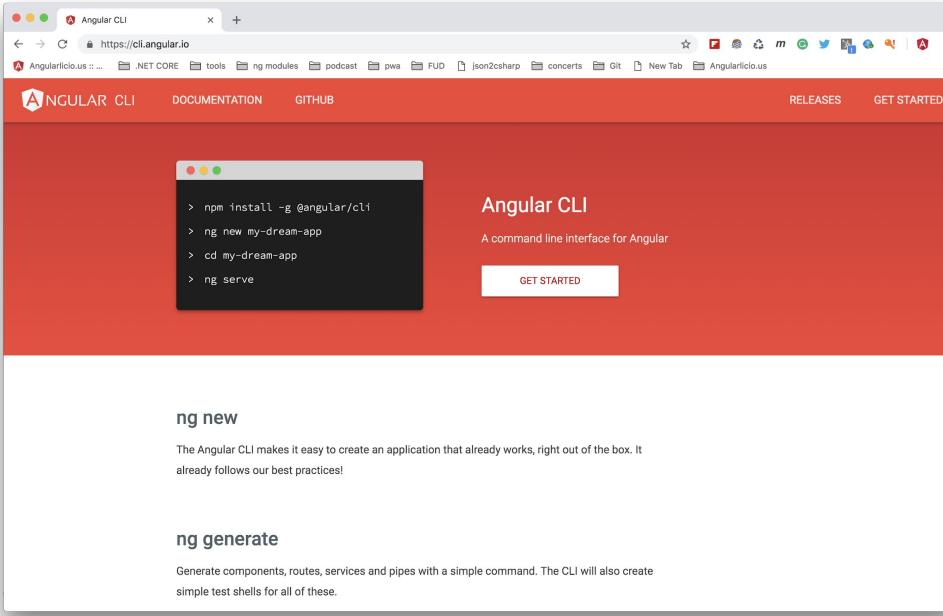
- NPM (node package manager)
  - <https://www.npmjs.com/>
- Javascript Packages
- Angular-specific packages
- Publish custom packages
  - Custom libraries
- Public and/or Private package repositories
- Package Versioning





# ng CLI

- <https://cli.angular.io/>



The screenshot shows the Angular CLI homepage. At the top, there's a navigation bar with links for 'Angular CLI', 'DOCUMENTATION', 'GITHUB', 'RELEASES', and 'GET STARTED'. Below the navigation, there's a large image of a Mac terminal window displaying the following command sequence:

```
> npm install -g @angular/cli  
> ng new my-dream-app  
> cd my-dream-app  
> ng serve
```

To the right of the terminal image, the text 'Angular CLI' is displayed, followed by 'A command line interface for Angular' and a 'GET STARTED' button. Below this section, there are two more sections: 'ng new' and 'ng generate'.

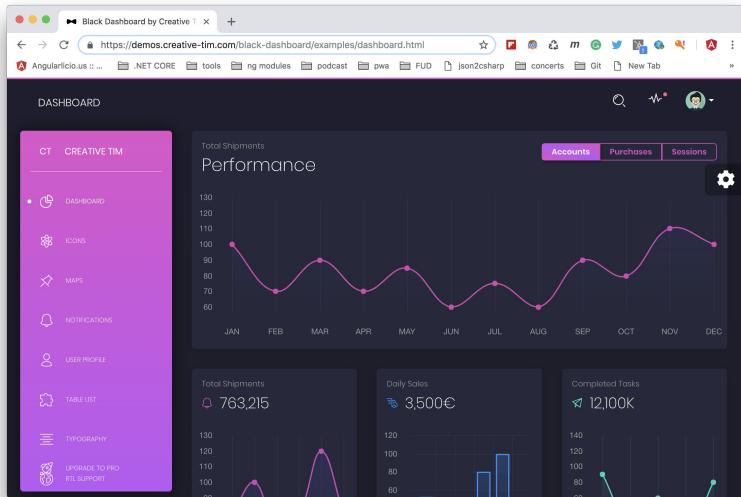
**ng new**  
The Angular CLI makes it easy to create an application that already works, right out of the box. It already follows our best practices!

**ng generate**  
Generate components, routes, services and pipes with a simple command. The CLI will also create simple test shells for all of these.

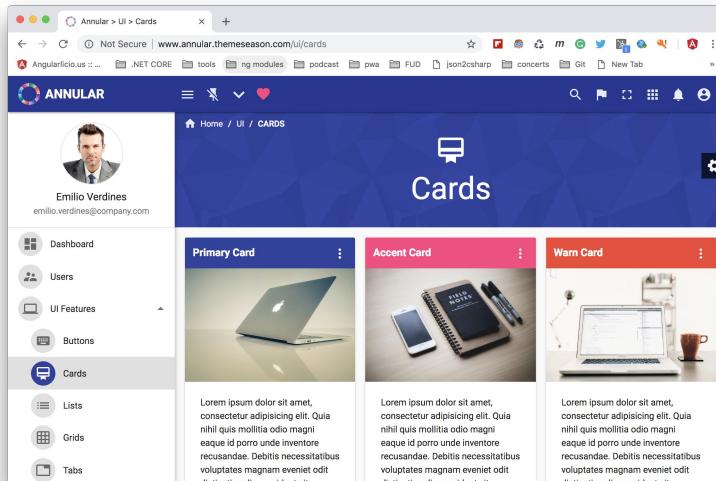


# ng TEMPLATES

- Bootstrap [www.creative-tim.com](http://www.creative-tim.com)
- Material Design [www.material.angular.io](http://www.material.angular.io)



A screenshot of a web browser displaying the 'Black Dashboard' theme from Creative-Tim. The dashboard has a dark purple header with a search bar and user profile icon. On the left is a sidebar with purple icons for Dashboard, Icons, Maps, Notifications, User Profile, Table List, Typography, and Upgrade to PRO. The main area features a large chart titled 'Performance' showing total shipments from January to December. Below it are three smaller cards: 'Total Shipments' (763,215), 'Daily Sales' (3,500€), and 'Completed Tasks' (12,100K).

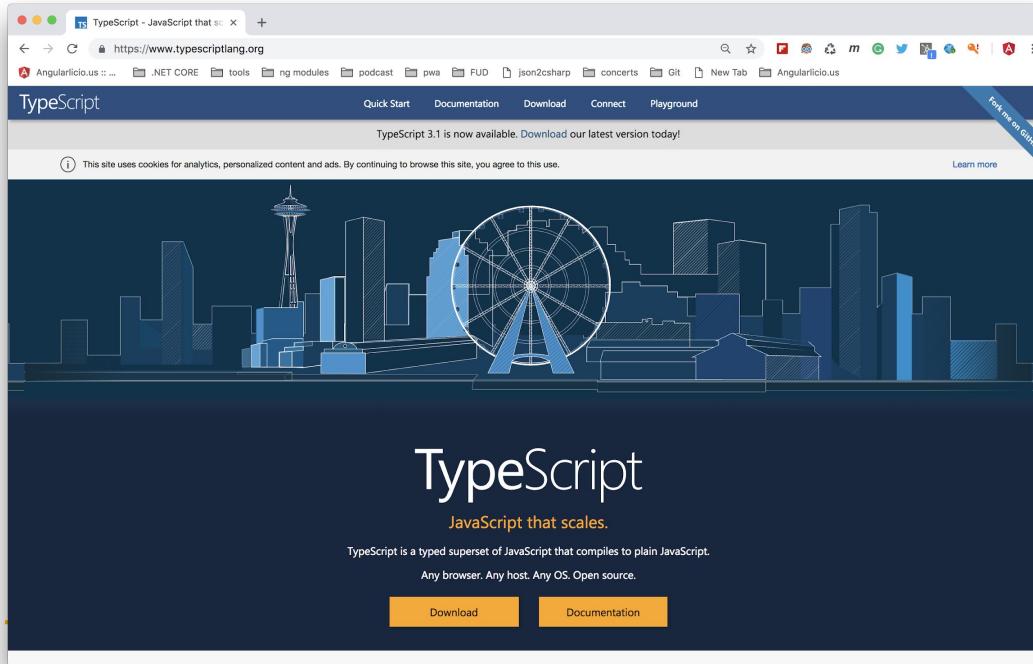


A screenshot of a web browser displaying the 'Cards' component from Annular. The page has a blue header with the 'ANNULAR' logo and navigation links for Home, UI, and CARDS. On the left is a sidebar with icons for Dashboard, Users, UI Features, Buttons, Cards (which is selected and highlighted in blue), Lists, Grids, and Tabs. The main content area shows three cards: 'Primary Card' featuring a laptop, 'Accent Card' featuring a smartphone and notebook, and 'Warn Card' featuring a laptop and coffee cup. Each card has placeholder text below it.



# TypeScript

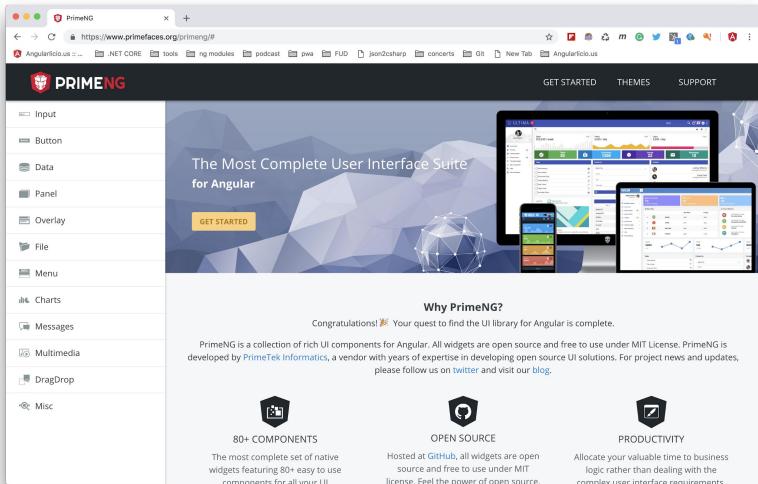
- <https://www.typescriptlang.org/>



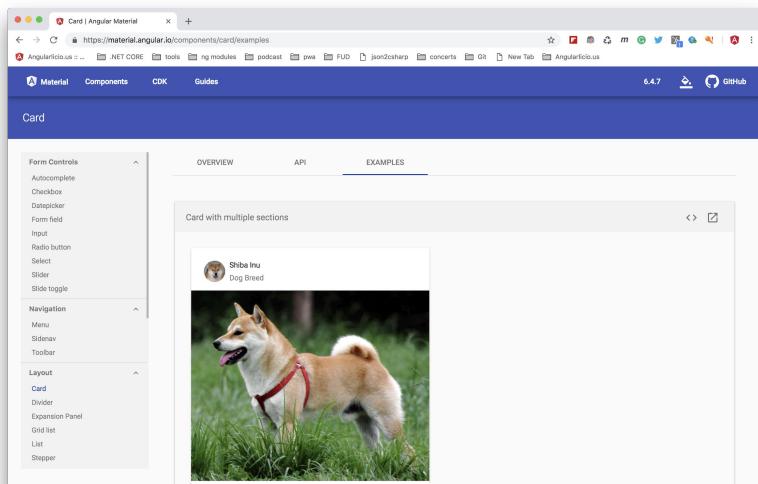


# ng CONTROLS

- Prime Ng: <https://www.primefaces.org/primeng/#/>
- Material Design [www.material.angular.io](https://material.angular.io)



The screenshot shows the PrimeNG website. The header features the PrimeNG logo and navigation links for GET STARTED, THEMES, and SUPPORT. A sidebar on the left lists various UI components: Input, Button, Data, Panel, Overlay, File, Menu, Charts, Messages, Multimedia, DragDrop, and Misc. The main content area has a blue background with a geometric pattern and displays the text "The Most Complete User Interface Suite for Angular". It includes a "GET STARTED" button and three sections: "Why PrimeNG?", "80+ COMPONENTS", "OPEN SOURCE", and "PRODUCTIVITY".

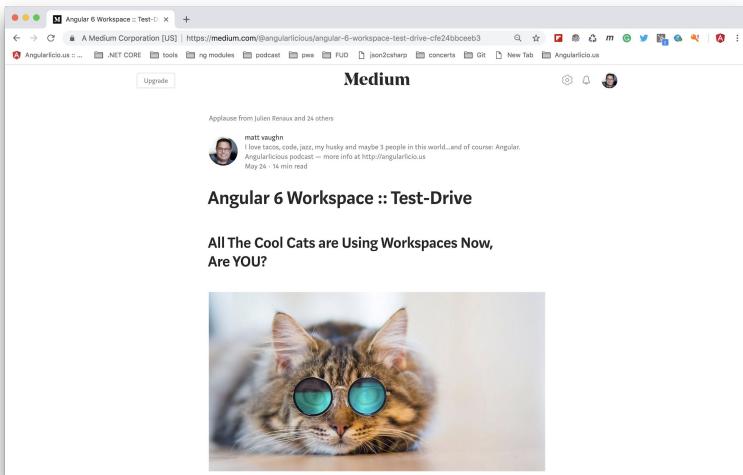


The screenshot shows the Material Design website under the Card component. The header includes the Angular logo and navigation links for Components, CDK, and Guides. The main content area shows the "Card" component with multiple sections. On the left, a sidebar lists components like Form Controls, Navigation, Layout, and Card. The main content area shows a card example featuring a Shiba Inu dog.

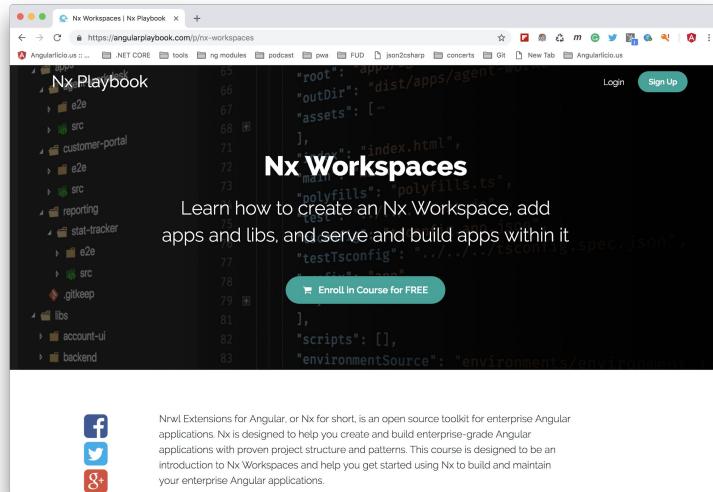


# ng WORKSPACE

- Default with Angular 6
- Nrwl.io Nx : <https://nrwl.io/nx>



A screenshot of a Medium post titled "Angular 6 Workspace :: Test-Drive". The post discusses the benefits of using workspaces in Angular 6. It features a photo of a cat wearing sunglasses. The URL in the browser bar is <https://medium.com/@angularicious/angular-6-workspace-test-drive-cfe24bceeb3>.



A screenshot of the Nx Playbook course page. The page shows a file tree for an Nx workspace and a snippet of the workspace's configuration file (nx.json). The configuration file includes details like root, outDir, assets, index.html, polyfills.ts, testTsConfig, and environmentSource. A call-to-action button says "Enroll in Course for FREE". Below the code editor, there's a brief description of Nx Workspaces and social sharing icons for Facebook, Twitter, and Google+.

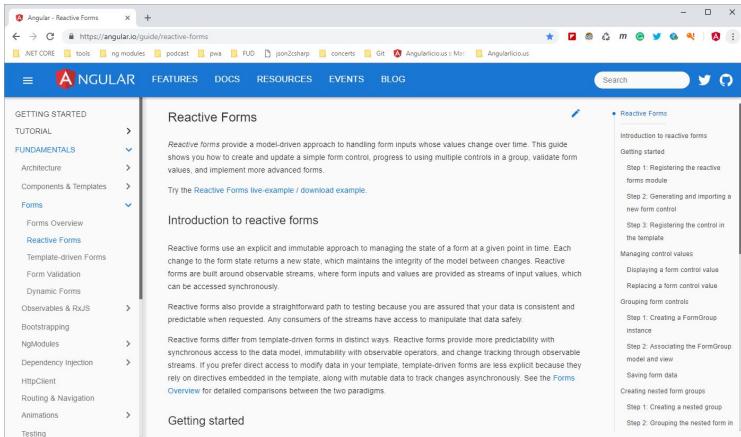
```
root": "dist/apps/agency",
"outDir": "dist",
"assets": [
  "src/assets",
  "index.html",
  "polyfills.ts",
  "main.ts"
],
"polyfills": "polyfills.ts",
"testTsConfig": "test.tsconfig.spec.json",
"scripts": [],
"environmentSource": "environments/environment.ts",
```

Nx Workspaces  
Learn how to create an Nx Workspace, add apps and libs, and serve and build apps within it



# ng FORMS

- (2) Options
  - Template-Driven
  - ReactiveForms



A screenshot of a web browser displaying the Angular Reactive Forms guide. The URL is https://angular.io/guide/reactive-forms. The page has a navigation bar with links for FEATURES, DOCS, RESOURCES, EVENTS, and BLOG. On the left, there's a sidebar with a tree view of Angular documentation categories like TUTORIAL, FUNDAMENTALS, and FORMS. The main content area is titled "Reactive Forms" and contains several sections: "Introduction to reactive forms", "Reactive forms provide a model-driven approach to handling form inputs whose values change over time.", "Try the Reactive Forms live-example / download example.", "Introduction to reactive forms", "Reactive forms use an explicit and immutable approach to managing the state of a form at a given point in time.", "Reactive forms also provide a straightforward path to testing because you are assured that your data is consistent and predictable when requested.", "Reactive forms differ from template-driven forms in distinct ways.", "Getting started". To the right of the main content, there's a sidebar with a list of "Reactive Forms" sub-sections such as "Introduction to reactive forms", "Getting started", "Step 1: Registering the reactive forms module", "Step 2: Generating and importing a new form control", "Step 3: Registering the control in the template", "Managing control values", "Displaying a form control value", "Replacing a form control value", "Grouping form controls", "Step 1: Creating a FormGroup instance", "Step 2: Associating the FormGroup model and view", "Saving form data", "Creating nested form groups", "Step 1: Creating a nested group", and "Step 2: Grouping the nested form in".

## Resources:

- [ReactiveForms Basics](#)
- [Todd Motto @ NG-NL 2017](#)
- [Reactive Thinking...](#)



# More RESOURCES

- <https://angular.io/resources>

The screenshot shows a web browser window titled "Angular - EXPLORE ANGULAR" displaying the resources page at <https://angular.io/resources>. The page lists several tools and libraries:

- Compodoc**  
This tool generates dedicated documentation for Angular applications.
- Lite-server**  
Lightweight development only Node.js® server
- NinjaCodeGen - Angular CRUD Generator**  
Generate several types of CRUD apps complete with e2e testing using template-sets for Angular, Material Design, Bootstrap, Kendo UI, Ionic, ...
- Nx**  
Nx (Nrw Extensions for Angular) is an open source toolkit built on top of Angular CLI to help enterprise teams develop Angular at scale.
- UI-jar - Test Driven Style Guide Development**  
A drop in module to automatically create a living style guide based on the test you write for your components.
- Universal for ASP.NET**  
This package provides facilities for developers building Angular applications on ASP.NET.
- Data Libraries**

“

*BE SO GOOD  
THEY CANNOT  
IGNORE  
YOU.*



# Angular RESOURCES

- [Angular Quickstart Guide](#)
- [Angular CLI](#)
- [Nodejs download](#)
- [TypeScript](#)
- [Angular Architecture](#)
  - [Modules](#)
  - [Components](#)
  - [Services](#)
  - [DI](#)
- [Routes](#)
- [HTTP Client](#)



# Presentation Resources

MORE RESOURCES

# Presentation **RESOURCES**



- [https://github.com/angularlicious/getting-started-wi  
th-angular](https://github.com/angularlicious/getting-started-with-angular)

# Thanks!



## Any questions?

You can find me at:

Twitter: @angularlicious

Email: [matt@angularlicio.us](mailto:matt@angularlicio.us)

Github: [www.github.com/angularlicious](https://www.github.com/angularlicious)

Web: [www.angularlicio.us](http://www.angularlicio.us) or [www.angularlicious.com](http://www.angularlicious.com)