

# Angular: Getting Started

---

## Introduction



**Deborah Kurata**

Consultant | Speaker | Author | MVP | GDE

@deborahkurata



**WHAT?**

**WHERE?**

**WHO?**

**WHEN?**

**WHY?**

**HOW?**

# Angular Is ...



**A JavaScript framework**

**For building client-side applications**

**Using HTML, CSS and TypeScript**

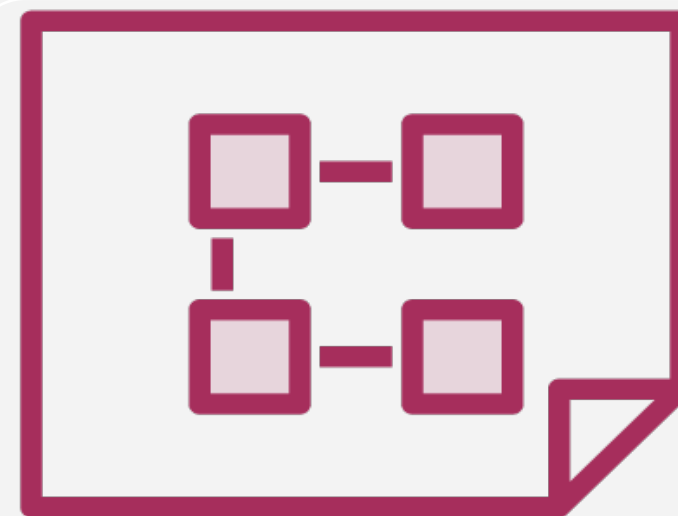
# Why Angular?



**Expressive HTML**



**Powerful Data Binding**



**Modular By Design**



**Built-in Back-End Integration**

# Module Overview



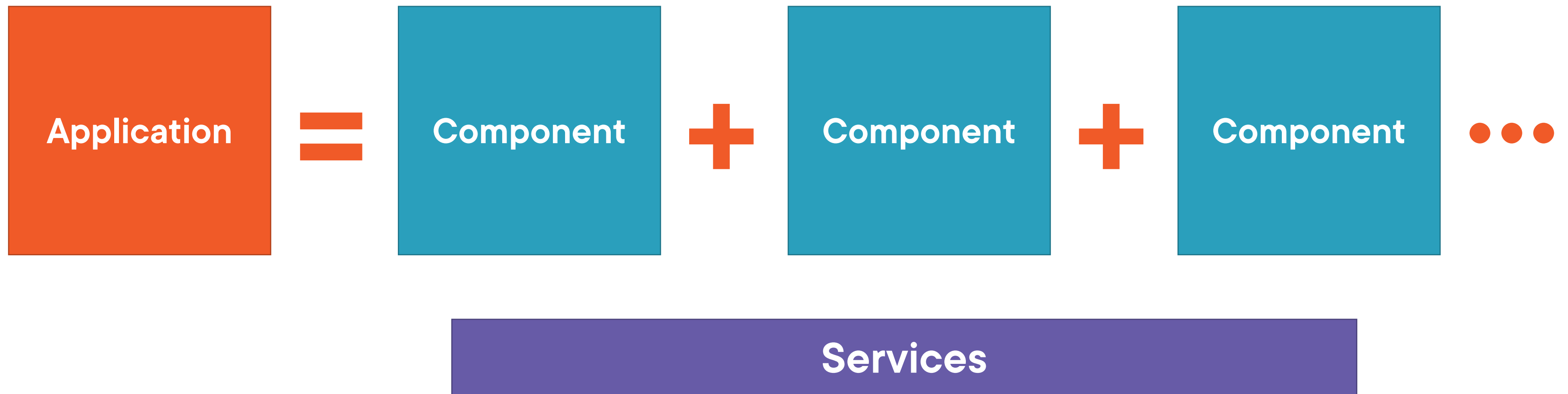
**Anatomy of an Angular application**

**Getting the most from this course**

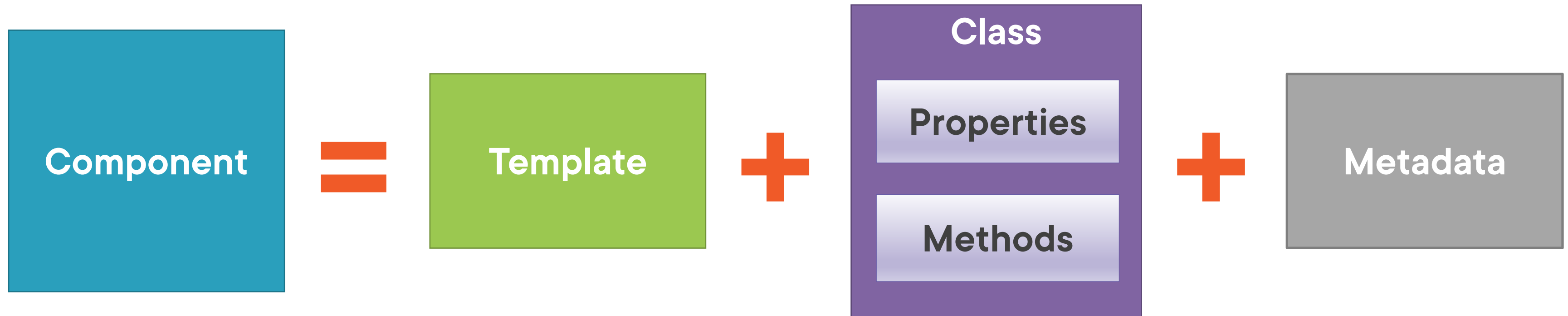
**Sample application**

**Course outline**

# Anatomy of an Angular Application



# Component



# Prerequisites



## Required

- JavaScript basics
- HTML basics
- CSS basics



## Helpful

- Object-oriented programming (OOP) concepts
- C++, C#, Java, ...



## Not required

- Prior knowledge of Angular
- Prior knowledge of TypeScript



# Thoughts? Comments? Questions?

## Angular: Getting Started

by Deborah Kurata

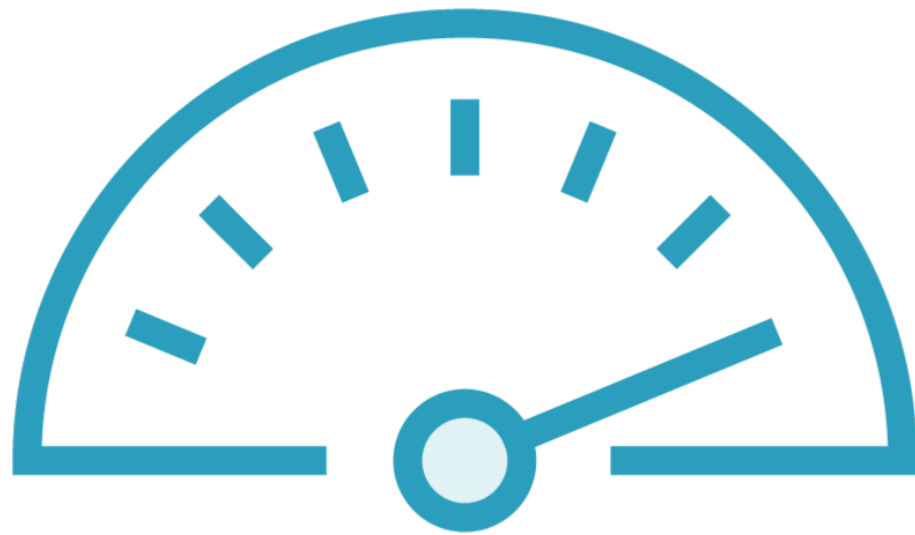
Angular is one of the fastest, most popular open source web app frameworks today, and knowing how to use it is essential for developers. You'll learn how to create components and user interfaces, data-binding, retrieving data using HTTP, and more.

[Resume Course](#) [Bookmarked](#) [Add to Channel](#) [Download Course](#) [Schedule Reminder](#)

[Table of contents](#) [Description](#) [Transcript](#) [Exercise files](#) [Discussion](#) [Learning Check](#) [Related Courses](#)

@deborahkurata

# Adjust the Speed



**Too fast?**



**Or too slow?**



**Just Right**

# Checklist

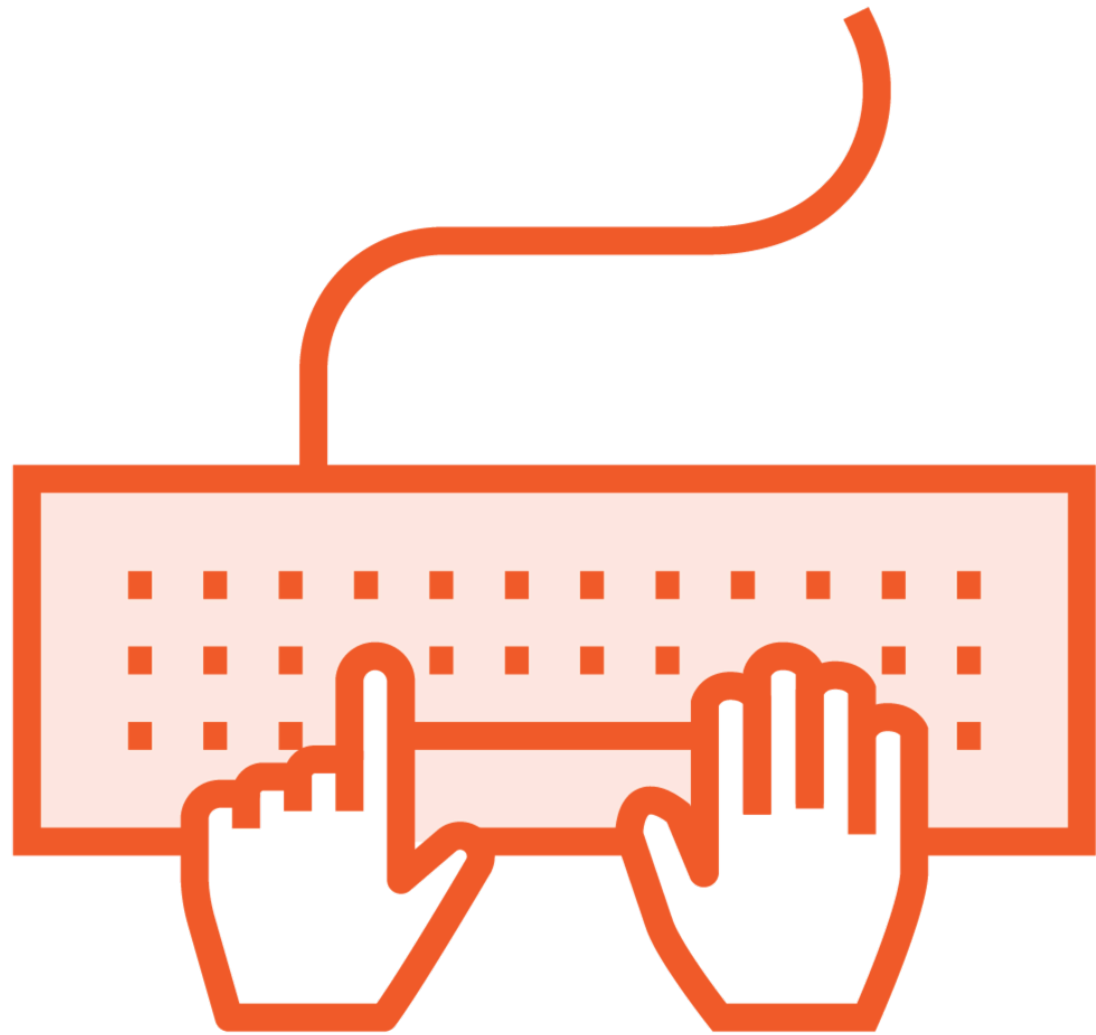


Review module concepts

Code along assistance

Revisit as you build

# Coding Along



**Code along with the demos throughout this course**

**Not required**

**Helpful for practicing what you learn**

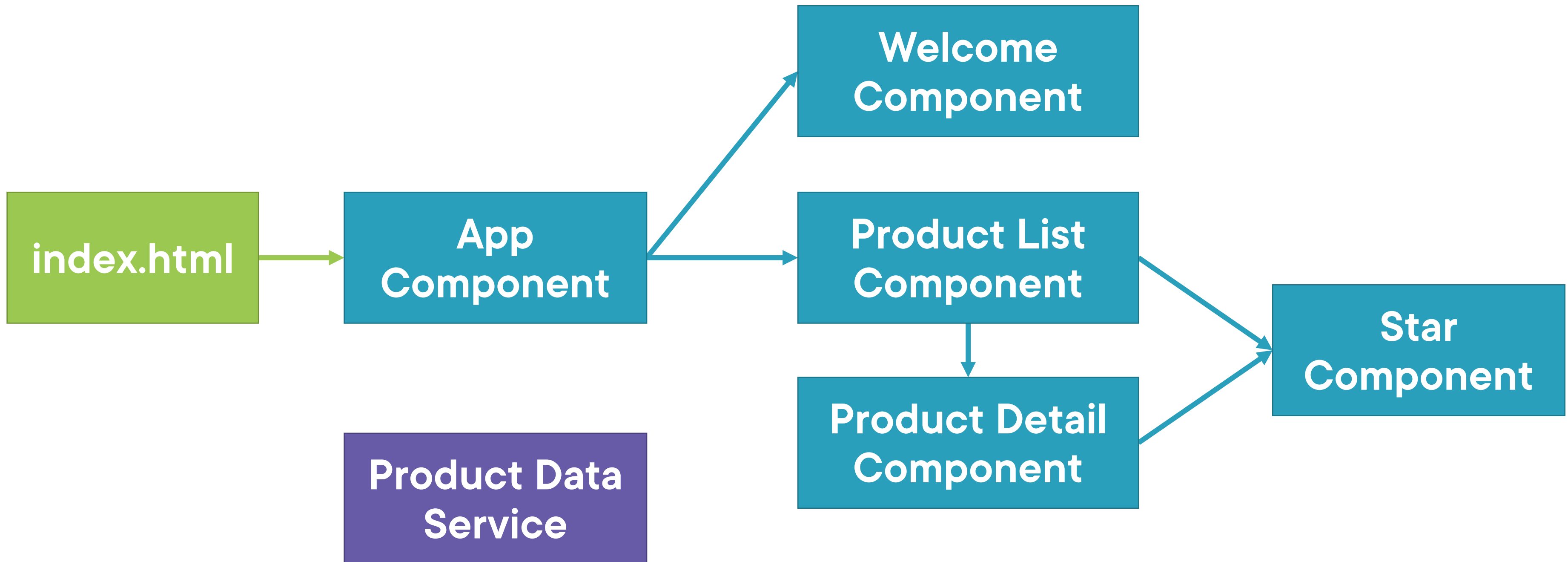
**You'll have a working sample application**

# Demo



**Sample application in action**

# Sample Application Architecture



# Course Outline



**First Things First**

**Introduction to Components**

**Templates, Interpolation, and Directives**

**Data Binding & Pipes**

**More on Components**

**Building Nested Components**

**Services and Dependency Injection**

**Retrieving Data Using HTTP**

**Navigation and Routing**

**Angular Modules**

**Building, Testing and Deploying with the CLI**





Coming up next ...

**First Things First**