



**COMPREHENSIVE TESTING
ANGULAR EĞİTİMİ
3 GÜN**



Digital Vizyon
Akademi

www.digitalvizyon.net



İçindekiler

Eğitim Hakkında.....	3
Neler Öğreneceksiniz?	3
Ön Koşullar	4
Kimler Katılmalı.....	4
Outline	5
Angular Testing Overview	5
Setup, tools and Technologies	5
Unit Testing.....	5
End To End (E2E) testing	5
PART 1: UNIT TESTS Unit tests with Jest	5
Testing of reactivity	6
PART 2: COMPONENT & INTEGRATION TESTS	6
Component Tests	6
Advanced component tests	6
Integration tests	6
Best Practices.....	6
PART 3: END-TO-END TESTS WITH CYPRESS & VISUAL REGRESSION.....	6
Visual regression with storybook	6
Cypress	7
Cypress Advanced.....	7
Cypress and Beyond	7

Eđitim Hakkında

Comprehensive Testing in Angular Eđitimi, Angular uygulamalarınızın kalitesini ve güvenilirliğini artırmak için test tekniklerini öğrenmek isteyen geliştiriciler için ideal bir kaynaktır. Angular ile web uygulamaları geliştirmeye aşına olanlar için tasarlanmış bu eğitim, test süreçlerinin derinlemesine anlaşılmasını sağlar.

Eđitimin ilk bölümünde, Angular'da testin önemini ve temel kavramlarını ele alıyoruz. Katılımcılara, test süreçlerinin neden kritik olduğunu ve Angular uygulamalarının kalitesini nasıl etkilediğini anlatıyoruz.

Daha sonrasında, Angular'da birim testleri (unit testing) ve bunların nasıl yazılacağı üzerinde duruyoruz. Jasmine test çerçevesini ve Karma test çalıştırıcısını kullanarak, servislerin, bileşenlerin ve direktiflerin nasıl test edildiğini öğretiyoruz. Ayrıca, asenkron kod ve bağımlılıkları nasıl test edebileceğinizi de öğrenirsiniz.

Eđitimde ayrıca entegrasyon testlerine de yer veriyoruz. Angular uygulamalarında daha karmaşık bileşen ve servis etkileşimlerinin nasıl test edileceğini gösteriyoruz. Protractor gibi araçlarla eşleşen end-to-end (E2E) testlerin nasıl gerçekleştirildiğini ve uygulamanın genel iş akışının nasıl doğrulandığını anlatıyoruz.

Eđitimde ayrıca, testlerin hızını ve verimliliğini artırmak için çeşitli stratejilere ve araçlara odaklanıyoruz. Katılımcılar, sürekli entegrasyon sistemlerine testleri nasıl entegre edeceklerini ve otomatik olarak çalıştırabileceklerini öğrenirler.

Comprehensive Testing in Angular Eđitimi, ayrıca kod kaplama tekniklerini ve kodunuzun ne kadarının testler tarafından kapsandığını nasıl ölçebileceğinizi de kapsar.

Eđitim, gerçek dünya örnekleri üzerinde çalışma fırsatı sunar; katılımcılar, Angular ile ilgili öğrendikleri test kavramlarını uygulamalı projelerde uygularlar. Bu, test süreçlerinin, test etme stratejilerinin ve en iyi uygulamaların daha iyi anlaşılmasına yardımcı olur.

Comprehensive Testing in Angular Eđitimi, Angular uygulamalarınızı güvenilir ve sağlam bir şekilde test etmek için gereken bilgi ve becerileri kazandırır. Eğitim, katılımcılara hem teorik bilgi hem de pratik deneyim kazandırarak Angular tabanlı projelerde test süreçlerinin etkili bir şekilde uygulanmasına olanak tanır.

Neler Öğreneceksiniz?

- Angular'ın temel kavramları: Componentler, Modüller, Databinding, Services ve Dependency Injection gibi temel Angular kavramlarını anlamaya ve uygulamaya yönelik çalışmalar yapacaksınız.
- Front-end geliştirme: Angular ile kullanıcı arayüzlerinin geliştirilmesi, routing, formlar, API çağrıları, görsel efektler ve diğer önemli front-end kavramları hakkında bilgi sahibi olacaksınız.



- Back-end geliştirme: Web API kullanarak RESTful API'lerin nasıl oluşturulacağını öğreneceksiniz ve Angular front-end ile back-end arasındaki veri akışını nasıl yapacağınızı öğreneceksiniz.
- Uygulama geliştirme: Tam stack (front-end ve back-end) uygulamaların nasıl geliştirileceğini ve deploy edileceğini öğrenerek, Angular ve Web API teknolojilerini uygulamalı olarak öğrenme fırsatı bulacaksınız.
- Proje çalışması: Eğitim sürecinde bir proje üzerinde çalışacaksınız ve Angular ve Web API teknolojilerini uygulamalı olarak öğreneceksiniz.

Bu eğitim programı, Angular ve Web API teknolojilerini kullanarak tam stack uygulamalarının nasıl geliştirileceğini öğrenmek isteyen web geliştiriciler için ideal bir eğitimidir.

Ön Koşullar

Angular ve Web API ile Full-Stack Applications Bootcamp eğitimi için önkoşullar şunlar olabilir:

- HTML, CSS ve JavaScript'in temel kavramlarına sahip olmak
- Web uygulamaları ve veritabanı kavramları hakkında bilgi sahibi olmak
- İlgili yazılım araçlarını (örneğin, Visual Studio Code, Node.js) kullanma becerisi

Bu önkoşullar yoksa, öncelikle bu alanlardaki bilginizi artırmanız ve pratik yapmanız tavsiye edilir. Ayrıca, Angular ve Web API hakkında temel bilgi sahibi olmak eğitimi daha verimli ve kolay hale getirecektir.

Kimler Katılmalı

Angular ve Web API ile Full-Stack Applications Bootcamp eğitimi, aşağıdaki kişiler için uygun olabilir:

- Web geliştiricileri: Angular ve Web API kullanarak full-stack web uygulamalarının nasıl geliştirileceğini öğrenmek isteyen web geliştiricileri.
- Başlangıç seviyesindeki Full-Stack Geliştiriciler: Angular ve Web API kullanarak tamamlayıcı teknolojileri öğrenmek isteyen başlangıç seviyesindeki full-stack geliştiriciler.
- İşletme ve IT profesyonelleri: Angular ve Web API kullanarak web uygulamalarının nasıl oluşturulacağını anlamak ve bu teknolojileri işletmelerinde kullanmak isteyen işletme ve IT profesyonelleri.

Eğitimi almaya karar vererseniz, Angular ve Web API konuları hakkında ilgi ve tutku duymanız, öğrenmeye istekli olmanız ve uygulamalı projelere odaklanmaya hazır olmanız önemlidir.



Outline

Angular Testing Overview

Setup, tools and Technologies

Unit Testing

- Organization and Conventions
- Jasmine
 - Built-In Matchers
 - Setup and Teardown
 - Disabled and Focused Tests
- Karma
- Command line parameters
- Testing Pipes
- Testing Components
- Testing Change Detection
- Testing Services
- Testing Asynchronous Code
- Testing Http
- Testing Forms
- Testing Routing
- Angular Test Bed
- Run your tests in a headless browser for automated tests environments (Continuous Integration)
- Code coverage report and Code coverage enforcement
- Mocks & Spies
 - Mock with fake class
 - Mock with spy
 - The Auto Spies Library

End To End (E2E) testing

- Protractor
- Cypress

PART 1: UNIT TESTS Unit tests with Jest

- Jasmine Jest
- Test Coverage
- Extending Jest
- Asynchronicity
- fakeAsync waitForAsync



- Mocking: mock, jest.fn, jest.spyOn
- Stubbing

Testing of reactivity

- RxJS Marbles for pipe operators
- Use of RxJS Marbles in normal unit tests
- Unit tests for NGRX

PART 2: COMPONENT & INTEGRATION TESTS

Component Tests

- TestBed and Dependency Injection
- DOM interaction
- Change detection
- Component Snapshots

Advanced component tests

- Nested components
- Testing @Input and @Output
- Testing directives and pipes
- Factory methods for reducing boilerplate code
- Spectator
- Testing Library
- Harnesses
- Reuse harnesses

Integration tests

- HttpTestingController
- RouterTestingModule
- NgRx and the MockStore

Best Practices

- Two approaches: London & Detroit
- Test strategies: when should I use what?
- Testable architecture

PART 3: END-TO-END TESTS WITH CYPRESS & VISUAL REGRESSION

Visual regression with storybook

- Multiple Jest configurations
- Puppeteer
- Storybook



- StoryShots

Cypress

- Overview of existing E2E frameworks
- Difference between Web-Driver and Chrome DevTools protocol
- Cypress and its “awaiting” feature
- The cy object
- Implicit and explicit assertions

Cypress Advanced

- Page Object Models
- cy expand object
- Avoid Flaky Tests
- Mocking of HTTP requests
- Reuse Angular Harnesses

Cypress and Beyond

- Optimal test design
- Conditional tests
- Direct communication with the backend
- Unit tests in connection with Storybook
- Unit tests with Component Test Runner
- Test Coverage