[Firebase and Firestore Masterclass](https://www.udemy.com/course/firebase-course/)

By

Angular University

[Udemy](https://www.udemy.com/course/firebase-course/)

Table of Contents

[Introduction 3](#_Toc38822378)

[Section 1: Course Kickoff - Development Environment Setup 5](#_Toc38822379)

[1.1 Serverless Angular with Firebase Course - Helicopter View 5](#_Toc38822380)

[1.2 Recommended Software Versions 5](#_Toc38822381)

[1.3 Firebase Course – Development Environment Setup 5](#_Toc38822382)

[1.4 The Typescript Jumpstart Ebook 5](#_Toc38822383)

[1.5 Setting Up our Firestore NoSQL Database 7](#_Toc38822384)

[Section 2: Introduction to the Firestore NoSQL Database 8](#_Toc38822385)

[2.1 The Firestore NoSQL Database - Documents and Collections 8](#_Toc38822386)

[2.2 Firestore Root Collections vs Nested Collections 8](#_Toc38822387)

[2.3 Querying Firestore Documents using the Firebase SDK 8](#_Toc38822388)

[2.4 Understanding Firestore Auto-generated Document Identifiers 8](#_Toc38822389)

[2.5 Querying Firestore Collections using the Firebase SDK 8](#_Toc38822390)

[Section 3: Angular Service Layers with AngularFire 9](#_Toc38822391)

[3.1 Setting Up AngularFire 9](#_Toc38822392)

[3.2 Querying a Firestore Collection using AngularFire 9](#_Toc38822393)

[3.3 The AngularFire API - snapshotChanges vs stateChanges 9](#_Toc38822394)

[3.4 Querying and Displaying Data with AngularFire - The Home Component 9](#_Toc38822395)

[3.5 AngularFire Observable Streams - An Important Feature 9](#_Toc38822396)

[3.6 Firestore Queries and Performance Guarantees - orderBy and array-contains 9](#_Toc38822397)

[3.7 Firestore Compound Queries - Understanding Firestore Indexes 9](#_Toc38822398)

[3.8 Understanding Firestore Querying Constraints and Invalid Query Error Messages 9](#_Toc38822399)

[3.9 The Course Component - Design Overview 9](#_Toc38822400)

[3.10 Initial Implementation of the Course Screen - the Course Resolver 10](#_Toc38822401)

[3.11 Course Component Lessons List - Querying a Nested Firestore Collection 10](#_Toc38822402)

[3.12 Course Component Conclusion - Pagination and Loading Indicator 10](#_Toc38822403)

[3.13 Firestore Data Modification Example - The Edit Course Dialog 10](#_Toc38822404)

[3.14 Edit Dialog Component Implementation – Conclusion 10](#_Toc38822405)

[3.15 Firebase SDK Offline Support - See it in Action 10](#_Toc38822406)

[3.16 Understanding Firestore Batched Writes - An Example 10](#_Toc38822407)

[3.17 Firestore Transactions - Transactions vs Batched Writes 10](#_Toc38822408)

[3.18 Firestore Persistent Document References - An Example 10](#_Toc38822409)

[Section 4: Firebase Authentication 11](#_Toc38822410)

[4.1 Firebase Authentication - Section Introduction 11](#_Toc38822411)

[4.2 Firebase UI In Action - Social Login plus Email and Password Authentication 11](#_Toc38822412)

[4.3 Authentication with Firebase UI - Implementation Conclusion 11](#_Toc38822413)

[4.4 The AngularFire Auth User Profile – a Practical Use Case 11](#_Toc38822414)

[4.5 AngularFire Auth Service - Displaying a User Profile Picture 11](#_Toc38822415)

[Section 5: Firestore Security Rules Deep Dive 12](#_Toc38822416)

[5.1 Security in a Serverless Architecture - Introduction to Firestore Security Rules 12](#_Toc38822417)

[5.2 Breaking Down the Firestore Security Rules Syntax 12](#_Toc38822418)

[5.3 Security Rules Fundamentals - How do Security Rules Work? 12](#_Toc38822419)

[5.4 Security Rules Fundamentals - Nested Rules and WildCards 12](#_Toc38822420)

[5.5 Security Rules - Making Data Visible Only to Authenticated Users 12](#_Toc38822421)

[5.6 Firebase Security Functions - an Example 12](#_Toc38822422)

[5.7 Authentication with a Pre-Approved User List 12](#_Toc38822423)

[5.8 User Authorization Roles - Admin Users vs Normal Users 12](#_Toc38822424)

[5.9 Adding a Schema to Firestore Data using Security Rules 12](#_Toc38822425)

[5.10 The Difference Between the Request and the Resource objects in Security Rules 13](#_Toc38822426)

[5.11 Installing the Firebase Tools - Firestore Command Line Deployment 13](#_Toc38822427)

[Section 6: Firebase Hosting 13](#_Toc38822428)

[6.1 Firebase Hosting - Benefits & Quick Deployment Guide 13](#_Toc38822429)

# Introduction

If you are looking for the **best backend to go along with Angular**, or even better if you are looking to do as little server-side development as possible and *focus mostly on the frontend*, then look no further: Firebase is the right choice for you.

With Firebase, you will be able to query your database directly from the client and even directly modify data straight from the client *in a secure way*. You might not think that this would even be possible and that there always has to be a server involved in order for things to be secure, and that is true.

But that does not mean that you have to write the server yourself, and that is the whole point of the term *Serverless*. With Firebase you can write **essentially serverless applications**, in the sense that in the application code base there is hardly any server-side code at all when compared to a traditional REST-based application.

In this course, you are going to learn how to design and build Angular applications using a Serverless Architecture by leveraging the whole Firebase ecosystem: this includes the **Firestore NoSQL DataStore database**, Firebase Security Rules, Firebase Hosting, Firebase Storage and Firebase Cloud Functions.

**Course Overview**

This course will allow you to learn Firebase and NoSQL in a fun and practical way, as you **build an application using the Firebase SDK and AngularFire**. As we build the application, we will also discuss the main concepts of serverless design.

More than querying the database and displaying the data on the screen, we are going to **make the application secure** using Firebase Security Rules and Firebase Authentication, and we are even going to deploy it to production using Firebase Hosting.

This will all be done without writing a single line of server code! We are then going to implement a **complete example of file upload**using Firebase Storage, again all in a secure way and without a single line of server code.

But what if the image needs some resizing? This is better done on the server so we will implement a **Firebase Cloud Function** that does some image processing upon upload completion. We are also going to demonstrate how Firebase Cloud functions can be used to keep multiple parts of our database in sync using server-side trigger functions.

We are then going to talk about denormalization best practices that will help us design the database to allow multiple views of the same data.

The result is a highly maintainable real-time application with huge scalability, performance and security, written with hardly any server code at all.

**Table of Contents**

This course will go over the following topics:

* Introduction to NoSQL data modeling
* Documents vs Collections
* Firestore Unique Identifiers
* Querying a database using the Firebase SDK
* Angular Service Layer Design with AngularFire
* Offline support
* Pagination
* Indexes, Composite Indexes
* Data Modification with AngularFire
* Transactions
* Multi-path Updates
* Authentication with Firebase Authentication
* Securing Database access with Firebase Security Rules
* Production Deployment with Firebase Hosting
* Image file upload with Firebase Storage
* Firebase Storage Rules
* Server-side image processing with Firebase Cloud Functions
* Denormalization is normal - supporting multiple data views
* Firebase Could Functions HTTP endpoints

**What Will You Learn In this Course?**

After taking this course, you will feel comfortable **designing and using a NoSQL database** for your application using Firestore. More than that, you will be familiar with the overall concept of a Serverless Architecture, and you feel comfortable with the whole Firebase ecosystem.

You will be proficient in the complete Firebase ecosystem, which includes: Firestore, AngularFire, Firebase Security Rules, Firebase Storage & Firebase Cloud Functions. But most of all, you will know how to use these building blocks to design and implement a Serverless Architecture in your application.

**What you’ll learn**

* Code in Github repository with downloadable ZIP files per section
* Serverless Architecture Design
* NoSql Database (Firestore)
* Serverless Angular development with AngularFire
* Serverless File Upload with Firebase Storage
* a full drop-in Authentication solution: Firebase UI and Firebase Authentication
* Secure client-side data modification operations with Firestore Security Rules
* High performance, free SSL hosting with Firebase Hosting
* Image processing with Firebase Cloud Functions
* Database Triggers with Firebase Cloud Functions
* REST endpoints with Firebase Cloud Functions

**Are there any course requirements or prerequisites?**

* Just a little bit of Angular and Typescript

**Who this course is for:**

* Angular Developers looking for the best possible backend to go along with Angular

# Section 1: Course Kickoff - Development Environment Setup

## 1.1 Serverless Angular with Firebase Course - Helicopter View

Review video

## 1.2 Recommended Software Versions

Just a quick note to let you know that the current recommended node version for this course is:

* Node 12

Please use this version of node, which is compatible with the package-lock.json available in the repository. This will avoid any semantic versioning issues and ensure you a smooth installation and course taking experience.

## 1.3 Firebase Course – Development Environment Setup

Review video and rough notes for Github repo link and install/test commands

(Cannot download video for some reason :/)

## 1.4 The Typescript Jumpstart Ebook

This course will be in the **Typescript language**, which is closely related to other languages that you are *likely already familiar with*.

But it that is not the case, we have here a **E-Book aimed at getting you started quickly with the language**, so that you can focus on the Advanced Angular features presented during the course.

Before starting to set up your development environment, please take a moment to download the **Typescript Jumpstart Ebook**, that is available together with this video course:

[Click Here To Start Immediate PDF Download](https://s3-us-west-1.amazonaws.com/angular-university/typescript-ebook/Typescript_Jumpstart_Book_Udemy.pdf)

This book covers the *Typescript Type System* and the multiples *Type Definitions,* which are the most unique parts of the language that are a bit different from other closely related languages.

Note: This is **not** required reading before continuing the video course.

This book is in the PDF format and the **code is fully in text and not in images**, so everything is searchable. This format is ideal for taking it to work and using it during your daily development,  please enjoy.

**This Book In a Nutshell**

This book is aimed at developers that want to get a *deeper understanding of Typescript*. If you have been trying to learn or use Typescript and would now like to go deeper into the language and learn how to make the most of its **powerful type system**, then this book is for you.

**What is the core value proposition of this book?**

Sit down in *one evening* with this book and learn the **key aspects** of the language and its type system that might take months of experience and long StackOverflow sessions to gather the hard way.

Become a lot more comfortable using Typescript on a daily basis, learn quickly the fundamentals of the language so that you can focus on other things in your project.

**Why a Fast-track Guide to Typescript?**

Typescript combines many of the best features of **statically typed**languages, together with some of the best features of **dynamically typed** languages.

So, this means that if you already know one of the following: JavaScript /ES6, Java, C#, Ruby Python, you will notice many *overlapping features.* So you already know a lot about Typescript, and only really need to learn **what is unique about it.**

**A Deceivingly Familiar Language!**

Many times, developers can just jump right into the language without any formal training, because the **language looks so familiar**. And this is very often the case in the Angular Ecosystem for example, where we often just start using the language straight away.

But you might notice that some things just don't work as expected: for example, compiler error messages show up for something that apparently should just work.

The problem is that the Typescript type system **works in a very different way** than the type systems of the most popular statically typed languages, and there are good reasons for that.

The familiarity with other languages is a great feature, but that **familiarity alone might not be enough** for a comfortable development experience.

To get the most out of Typescript, we really need to take a moment to **dive deeper into its type system**, and that is what this book is specifically about.

**I'm a JavaScript Developer, is this book for me?**

If you are afraid that using Typescript means a lot of ceremony and verbosity for just a bit of tooling, in this book you will learn that we can actually have the **best of both worlds in Typescript**:

we can write very concise code with almost no type annotations, but still benefit from all the tooling like auto-completion and refactoring working out-of-the-box.

**Table Of Contents**

Have a look at the table of contents below, to give you an idea about the book:

* Section 1 - Introduction
* Section 2 - The Typescript Type System
* A Simple Example - Why Doesn't This Work?
* Key Concept 1 - Type Inference
* Key Concept 2
* Structural Subtyping - How are types defined?
* Key Concept 3 - Type Compatibility Section
* Section 3 - Typescript Type Definition
* What are the multiple scenarios for Typescript Type Definitions?
* How do I use libraries that don't have Type Definitions available?
* How does the Any Type work?
* What is the relation between Type Definitions and Npm?
* Do we really need type annotations to get type-safety?
* Why Type safety does not mean more ceremony?
* The biggest advantage of Typescript
* How to make the most of Typescript Type Definitions
* What is @types, when should I use it and why?
* What happened to the typings executable and DefinitivelyTyped ?
* What are compiler opt-in types, when should I use them and why?
* Why do I sometimes get this 'duplicate type definition' error?
* Handling the gap between libraries and the compiler
* Guidelines for Using the multiple Type Definitions available
* How to make sure our programs leverage type safety effectively?
* Section 4 - Conclusions
* Final Thoughts
* Bonus Content - Typescript Video List
* Bonus Content - Free Angular For Beginners Course

**Any questions or feedback about the book?**

If you have **any questions about the book** or would like to send me some **feedback** on it, please send me a Direct Message using the Udemy application.

## 1.5 Setting Up our Firestore NoSQL Database

Review video

# Section 2: Introduction to the Firestore NoSQL Database

## 2.1 The Firestore NoSQL Database - Documents and Collections

Review video

## 2.2 Firestore Root Collections vs Nested Collections

Review video

## 2.3 Querying Firestore Documents using the Firebase SDK

Review video and code

## 2.4 Understanding Firestore Auto-generated Document Identifiers

Review video

## 2.5 Querying Firestore Collections using the Firebase SDK

Review video and code

# Section 3: Angular Service Layers with AngularFire

## 3.1 Setting Up AngularFire

Review video and code

## 3.2 Querying a Firestore Collection using AngularFire

Review video and code

## 3.3 The AngularFire API - snapshotChanges vs stateChanges

Review video and code

## 3.4 Querying and Displaying Data with AngularFire - The Home Component

Review video and code

## 3.5 AngularFire Observable Streams - An Important Feature

Review video and code

## 3.6 Firestore Queries and Performance Guarantees - orderBy and array-contains

Review video and code

## 3.7 Firestore Compound Queries - Understanding Firestore Indexes

Review video and code

## 3.8 Understanding Firestore Querying Constraints and Invalid Query Error Messages

Review video and code

## 3.9 The Course Component - Design Overview

Review video and code

## 3.10 Initial Implementation of the Course Screen - the Course Resolver

Review video and code

## 3.11 Course Component Lessons List - Querying a Nested Firestore Collection

Review video and code

## 3.12 Course Component Conclusion - Pagination and Loading Indicator

Review video and code

## 3.13 Firestore Data Modification Example - The Edit Course Dialog

Review video and code

## 3.14 Edit Dialog Component Implementation – Conclusion

Review video and code

## 3.15 Firebase SDK Offline Support - See it in Action

Review video and code

## 3.16 Understanding Firestore Batched Writes - An Example

Review video and code

## 3.17 Firestore Transactions - Transactions vs Batched Writes

Review video and code

## 3.18 Firestore Persistent Document References - An Example

Review video and code

# Section 4: Firebase Authentication

## 4.1 Firebase Authentication - Section Introduction

Review video and code

## 4.2 Firebase UI In Action - Social Login plus Email and Password Authentication

Review video and code

## 4.3 Authentication with Firebase UI - Implementation Conclusion

Review video and code

## 4.4 The AngularFire Auth User Profile – a Practical Use Case

Review video and code

## 4.5 AngularFire Auth Service - Displaying a User Profile Picture

Review video and code

# Section 5: Firestore Security Rules Deep Dive

*\*Code for this section refers to Firebase Security Rules in Firebase Console. I will include a final copy of these rules in the repo.*

## 5.1 Security in a Serverless Architecture - Introduction to Firestore Security Rules

Review video and code

## 5.2 Breaking Down the Firestore Security Rules Syntax

Review video and code

## 5.3 Security Rules Fundamentals - How do Security Rules Work?

Review video and code

## 5.4 Security Rules Fundamentals - Nested Rules and WildCards

Review video and code

## 5.5 Security Rules - Making Data Visible Only to Authenticated Users

Review video and code

## 5.6 Firebase Security Functions - an Example

Review video and code

## 5.7 Authentication with a Pre-Approved User List

Review video and code

## 5.8 User Authorization Roles - Admin Users vs Normal Users

Review video and code

## 5.9 Adding a Schema to Firestore Data using Security Rules

Review video and code

## 5.10 The Difference Between the Request and the Resource objects in Security Rules

Review video and code

## 5.11 Installing the Firebase Tools - Firestore Command Line Deployment

Review video and code

# Section 6: Firebase Hosting

## 6.1 Firebase Hosting - Benefits & Quick Deployment Guide

Review video

# Section 7: Firebase Storage Deep Dive (File Upload)

## 7.1 What is Firebase Storage? Learn its Key Features

Review video

## 7.2 File Upload with AngularFire - Step by Step Implementation

Review video and code

## 7.3 Implementing a File Upload Percentage Indicator with Angular Material

Review video and code

## 7.4 Get a Firebase Storage Revocable Download Url

Review video and code

## 7.5 Firebase Storage Security Rules – File Upload Conclusion

Review video and code

# Section 8: Firebase Cloud Functions Deep Dive

## 8.1 Firebase Cloud Functions - Hello World

Review video and code

## 8.2 Firebase Cloud Function REST Endpoint with Express

Review video and code

## 8.3 Firebase Cloud Functions Local Emulator and Production Deployment

Review video and code

## 8.4 Implementing a Firestore Database Trigger with Firebase Cloud Functions

Review video and code

## 8.5 Firebase Cloud Functions - Automatic Document Counter Implementation

Review video and code

## 8.6 Firebase Cloud Functions - Implementing a File Upload Trigger

Review video and code

## 8.7 Download a File from Firebase Storage from inside a Firebase Cloud Function

Review video and code

## 8.8 Image Processing with ImageMagick from inside a Firebase Cloud Function

Review video and code

## 8.9 Image Processing of Thumbnail with Firebase Cloud Functions – Implementation

Review video and code

## 8.10 Image Processing with Firebase Cloud Functions – Demo

Review video and code

# Section 9: Conclusion & Bonus

## 9.1 Other Courses

Review video

## 9.2 Bonus Lecture

**Updated on May 2020**

These Bonus coupons have been recently updated. Coupons *expire monthly now*, so if you happen to click on an expired coupon please send me a direct message or a question here on Udemy and I will send you back a new coupon ASAP.

**Udemy Discount Coupons**

Please find here *coupons for all the Angular University courses,*at the best current price on the platform!

So you can always come back here and grab a new coupon at any time:

[NestJs in Practice (with MongoDB)](https://www.udemy.com/course/nestjs-course/?couponCode=BONUS_MAY_2020)

[Angular Testing Masterclass](https://www.udemy.com/course/angular-testing-course/?couponCode=BONUS_MAY_2020)

[NgRx (with NgRx Data) - The Complete Guide](https://www.udemy.com/course/ngrx-course/?couponCode=BONUS_MAY_2020)

[Angular Core Deep Dive Course](https://www.udemy.com/course/angular-course/?couponCode=BONUS_MAY_2020)

[RxJs In Practice Course](https://www.udemy.com/course/rxjs-course/?couponCode=BONUS_MAY_2020)

[Stripe In Practice](https://www.udemy.com/course/stripe-course/?couponCode=BONUS_MAY_2020)

[Angular Universal Masterclass](https://www.udemy.com/course/angular-universal-course/?couponCode=BONUS_MAY_2020)

[Angular Material Masterclass](https://www.udemy.com/course/angular-material-course/?couponCode=BONUS_MAY_2020)

[Angular Progressive Web Applications (PWA) Masterclass](https://www.udemy.com/course/angular-pwa-course/?couponCode=BONUS_MAY_2020)

[Angular Security Masterclass](https://www.udemy.com/course/angular-security/?couponCode=BONUS_MAY_2020)

[Angular Forms Jumpstart](https://www.udemy.com/course/angular-forms-course/?couponCode=BONUS_MAY_2020)

[Angular Router Jumpstart](https://www.udemy.com/course/angular-router/?couponCode=BONUS_MAY_2020)

[Angular Advanced Masterclass](https://www.udemy.com/course/angular-advanced-masterclass/?couponCode=BONUS_MAY_2020)

[Reactive Angular Course (with RxJs)](https://www.udemy.com/course/rxjs-reactive-angular-course/?couponCode=BONUS_MAY_2020)

[Typescript Masterclass](https://www.udemy.com/course/complete-typescript-2-course/?couponCode=BONUS_MAY_2020)

**The Angular University Newsletter**

If you want to keep up to date with the Angular ecosystem with timely news, educational material in both video and written form and get **useful PDFs** on many angular sub-topics, then the Angular University Newsletter is a great resource for you.

I invite you to go to our main website and click the "Newsletter" button on the top blue menu:

[Click Here to Subscribe to the Angular University Newsletter](https://angular-university.io/)

**FREE Angular For Beginners Course (2.5h)**

All the material available here is also available at the Angular University website. New courses and blog posts are getting published all the time on many Angular sub-topics.

At the main website you can watch the Angular for Beginners course, which is available for free.

In general, about *25% of all our video lessons are freely available* on the Website, and you can watch them for free without even having to be logged in to the website:

[Click Here to View the Angular University Website](https://angular-university.io/)

**The Angular University Youtube Channel**

All the free lessons available on the main Website are also regularly published on our YouTube Channel:

[Click Here to View the Angular University YouTube Channel](https://www.youtube.com/channel/UC3cEGKhg3OERn-ihVsJcb7A?view_as=subscriber)

And now for the final conclusion lesson of the course, where we are going to summarize everything that we have learned and talk about some of the key takeaways learned throughout this course.

Thank you for watching and I hope you have enjoyed the courses!

Kind Regards,

Vasco

Angular University

## 9.3 Conclusions and Key Takeaways

Review video