Top 4 cool projects to build to enrich your full-stack web development skills

Every good mentor will tell you that the best way to learn programming skills is to build projects, but you might not know what to build, this article is here to help you with that!

Carmine Zaccagnino
Jun 14, 2018 · 4 min read

You've been told thousands of times that you just need to build new projects all the time to learn and acquire critical skills for your job but you really can't think of any cool project you can build.
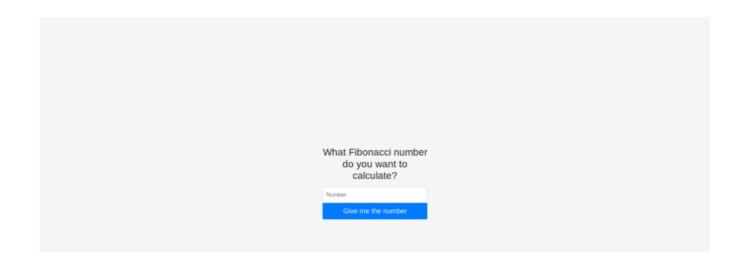
Maybe you don't know what you're capable of building, maybe you are just out of ideas.

This list is here for you: if you start building these projects you'll learn new skills and also get new ideas to expand your experience and knowledge.

I'll sort the project from the easiest to the hardest, so you can build these projects in order, or skip one or two of them if they're too easy for you.

The point of this article is to give you ideas so you can learn by yourself using the resources you can find on-line, I won't give you code to copy and paste, only hints about the technologies to use and how to use them, now that you have the ideas you have to investigate how to make them reality or you'll actually learn and remember nothing.

Project #1 (Basic Level): Simple web app

The input for the app

For this one I'll give you more guidance than for the other projects.

The first cool project you should build, if you haven't ever built such a project, is a simple front-end/back-end web app with a simple text input and an output.

You should build a back-end like I showed in this article about how to build an API that generates Fibonacci numbers and a front-end using JQuery to make a GET Request to the API and make basic DOM manipulations to show the number you got from the server.

For hosting the back-end I'd recommend Heroku since it's free and quite easy to use and for the front-end GitHub Pages is a great option.

This is a very simple project but will teach you the basics of how to make your server interact with your front-end web pages and will give you hints about how to build a "real" website.

Project #2 (Easy): Basic One-Page Blog

What I recommend you build at this stage is a simple one-page list of articles you fetch from a database.

For this project you need the same skills and basic knowledge of the previous app, but you'll start working with databases.

I don't recommend using an ORM, when starting out it's good to use SQL and make queries manually before you start using libraries to avoid that work.

These projects are simple enough that worrying about queries shouldn't be a huge problem.
One thing you can do at this stage is to start using a front-end framework like Bootstrap to make better-looking websites without too much effort and use their examples to learn more about how to make your site look more like something serious and professional.

Project #3 (Bridge to Intermediate): A Full Blog and CMS

At this point you need to learn how to use page templating and load articles from the database as they are requested by the user.

You should also build an admin page with an authentication system that involves username, passwords, sessions and cookies, something that looks, feels and works like any other websites you log in to.

For this project you should start using ORM's in addition to page templates, with this project you'll probably start having your own ideas about which projects to build.

Project #4 (Intermediate and Higher): A Website offering real life or external services using third-party API's

You can interact with server and databases and have used page templates and are now ready for the next step: build something useful for your users.

Now you should start interacting with third-party API's and services to provide a service to your users: copy the idea from another website, make a social network, a business review site, an e-commerce website, start handling user sign-up and use Twilio to send confirmation SMS's, start using OAuth for easier sign-up and log-in and start building something useful to show off when you look for work or get a great business idea and build it.

The conclusions

To learn how to code you need to code, and the more you build things, the more you'll learn what you can build and you'll acquire fundamental skills for the programming industry.

If you've never built a web project you might not know where to start, but if you start from some project ideas and investigate how to build them you will also start getting your own ideas based on your knowledge and experience building other projects.

Follow me on twitter at @carminezacc and stay updated on my Medium profile for more development advice coming out soon.

Web Development   Full Stack   Full Stack Developer   CMS   Cms Development