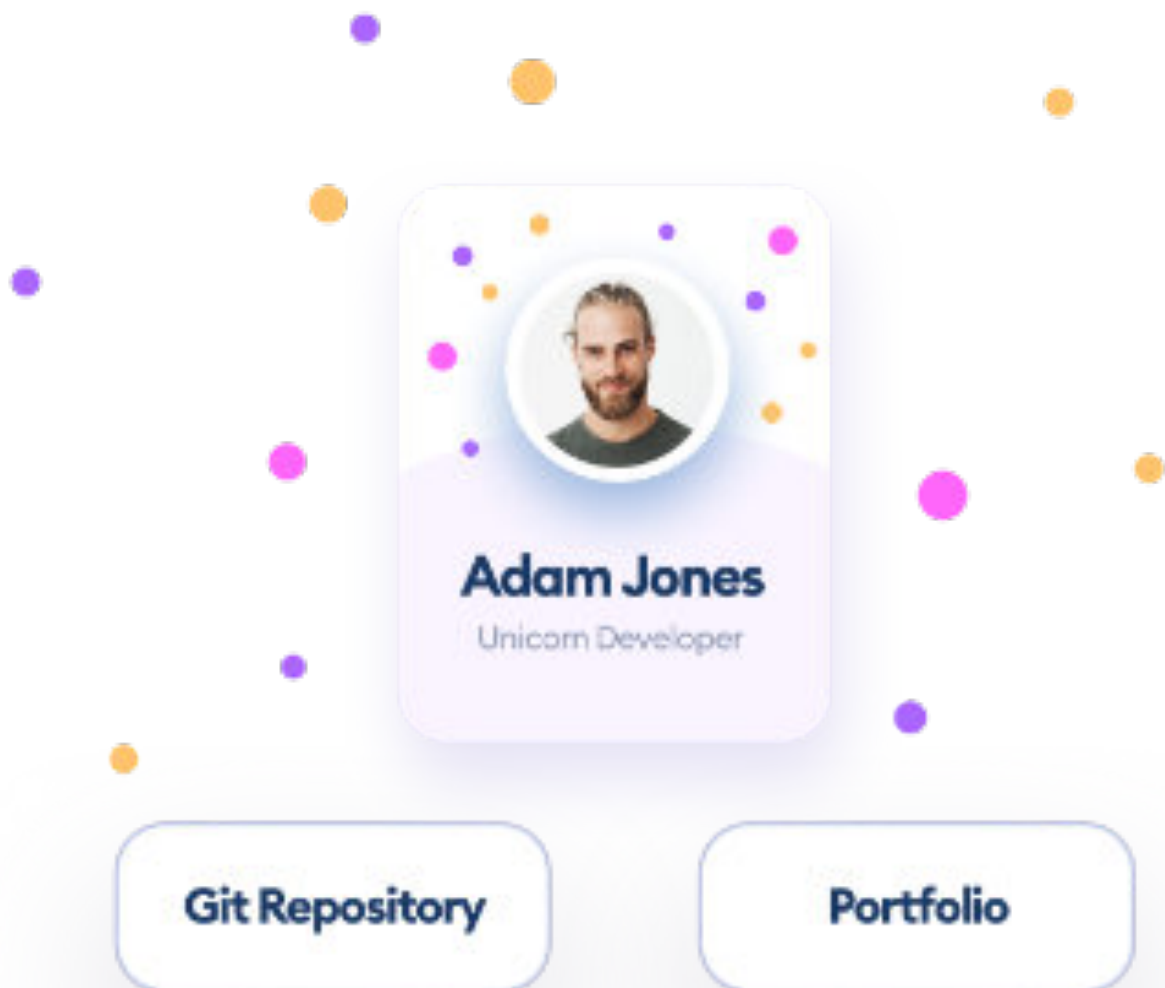# 4

Creating a portfolio

Creating a portfolio

Your portfolio is a critical part of your job seeking. If your portfolio is excellent and you'll get lucky, you might get an invite for the interview even without sending an application. And if you send an application and get through the first resume review, an impressive portfolio is what lets you stand out amongst dozens of other developers.

Software developers portfolio is made of two major parts - **Git repository** and **portfolio website**. You absolutely need to have both - one allows the recruiter to see your code, and the other the final effects of your work.

Building great Git repository

Your Git repository is a record of your growth and the projects you've made. It's a significant selling point, so you need to make it work in your favor.

The best place to have your Git repository is GitHub. It's almost an industry standard, and getting practice using it will come in handy. To make your GitHub stand out, you should follow a few rules. Firstly, make small, frequent commits. Some recruiters look at your activity and seeing someone who commits a lot is impressive. Also, making small commits is a good practice in real-life projects.

Consider hiding your first projects and the projects you didn't finish or are not proud of. As long as you have at least few projects, it's better to show only your best work. If you don't want to hide projects, at least pin the best ones. It will make them stand out. You can find the instruction on how to pin the projects here.

Remember each project should have a readme file. Include technologies used, what the project is, and how to run it. Always have the live versions of the projects, so anyone can quickly see how it looks in action. Link to the live project should also be in your readme!

The final thing you can do to improve your GitHub is to pimp your profile. You can make it pretty by adding a readme file to the root of your profile. Here you can find instructions for doing this. You may add images to your profile page, links to your social media, or add technologies you use. Treat the main

profile page as your developer resume - add anything relevant (for more tips, check out the resume chapter of this book). You can even add your GitHub stats or get your latest blog posts (if you have a blog).

A portfolio website that stands out

The portfolio website should look good. Good is not equal to fancy - its goal is to showcase your work, so the projects should be the main focus. Go for the clear and clean design, and watch out for simple errors (for example typos or lack of favicon). Host your portfolio using some free solution like Netlify. It's good to have a custom domain, so consider spending a few dollars on getting it.

If you want to add some extra visuals like animations - go for it. But make sure you don't overwhelm users. The visual clutter destroys the excellent effect. Your portfolio is about showcasing your projects, not impressing with every animation skill you have.

The portfolio should be responsive and include a contact form or a chatbot - you want to make reaching out to you as easy as possible. Validate the inputs if you're using a contact form - otherwise, some people will make fun of you by sending empty messages.

The question I get asked a lot is whether you should use CSS frameworks or not in your portfolio. The answer is - it depends.

"

Pick interesting projects for your portfolio!

Frameworks will make creating clear and responsive designs easier. But if your special skill is being great with design, you should go with custom design and CSS.

On the other hand, If you're focused on JavaScript and are not skilled with design, you can go for frameworks and stand out some other way. Your portfolio needs to impress somehow, but it doesn't need to be a design.

As for the rest of the tech stack of your portfolio - it doesn't really matter. Pick anything that works for you - it may be a JavaScript framework, WordPress, or whatever else you like using.

Always remember that the crucial thing in the portfolio are the projects. Make them stand out, especially the ones you're proud of.

Picking great portfolio projects

Picking interesting projects to your portfolio is a perfect way to stand out from the crowd. I still remember seeing a simple puzzle game in one of the portfolios - our entire office started to play it. We invited the person to the interview without a moment of hesitation.

Your goal is to do this same - build something impressive enough for the company to invite you.

Avoid making apps that everyone does - weather apps, tic-tac-toe games, or to-do lists. They're great projects to learn to code but not good enough to make into a featured app in your portfolio. You need to be more original.

If you attended a Bootcamp, you probably had to build some final project. As a rule of thumb, avoid using this project as the main thing in your portfolio. Bootcamp projects tend to be inspired by teachers, and because of that, they're often repetitive. In my experience, I could recognize the Bootcamp projects at first glance - many candidates were finishing these same Bootcamps with similar projects.

That being said, if you believe your project is unique or you added a lot of great features, feel free to feature it.

To find an idea for the great app, think about the things you're interested in. Are you a lover of Netflix? Make a Netflix clone or a movie database like IMDB. Like playing card games? What about writing a poker game in JavaScript? Interested in investing? Build an app with stock data visualized as charts. You can easily find a free API to get the data about almost anything; check this or this list if you don't believe me.

An additional benefit of picking the project that's interesting to you is that you won't be discouraged easily. So choose wisely.

The other way to make a great app is to solve someone's problem. Maybe you've got a charity you like, but they don't have a website? Make it for them. They will get a great new page, and you can add testimonials to your portfolio. Nothing

sells better than other happy customers. The possibilities for new apps are endless!

If you need some more project ideas, here's the list of websites that can inspire you:

https://www.frontendmentor.io/

https://github.com/florinpop17/app-ideas

https://github.com/tastejs/awesome-app-ideas

https://github.com/karan/Projects

You don't need to have many projects in your portfolio - the recruiter will probably just glance at it. The tricky part is to make the projects really great. Make sure to have a clean design and make it responsive. Give the users the best possible experience irrespectively the device they use.

Pay special attention to the quality of your code - remove obsolete comments or console logs. Test your application in Lighthouse - try to achieve perfect or near-perfect scores in every category. Check out the quality of your HTML using tools to analyze the semantics of the page like W3C Validator.

Don't forget about the social media metadata. You want to share your portfolio on different platforms, so make sure that the links look beautiful.

These may look like tiny details, but they matter.

When you've picked and coded your best projects. It's time to present them.

For every project, you have to make a description. It should contain information about the tech stack you picked and the project functionality. If your project has some additional features, like tests, add information about it. This same would go with external libraries; if you used something complex, like D3.js, write about it and justify your choice.

It's also a good idea to highlight what you've learned in the project. If you're building a single-page application, you probably had to master client-side routing. If your app uses some external API, you've learned to handle asynchronous requests. Whatever it is, write it.