A common issue faced by our students is how to demonstrate passion and interest in coding before you have had the opportunity to complete any internships or work experience.

In this situation, taking on **“side projects”** - projects you complete outside of school classwork - is an excellent way to deepen your experience and show companies that you would be a great hire for their team.

### **Why it’s a great idea to take on side projects**

Side projects:

* Add experience and interest to your LinkedIn and resume
* Demonstrate to companies that you would be a great fit for an internship
* Give you something substantial to talk about when asked “what have you done” in a interview
* Provide recruiters with a sense of you as an individual, and makes you more memorable
* Make you much better prepared when you are hired for a software role
* Allow you to meet and talk with people who work in tech
* Let you advance your skills and focus on something that interests you!

### **What type of side projects can you take on?**

Successfully working on a side project will require both your time and interest, so it’s a good idea to choose something that genuinely appeals to you.

Some ideas for a side project might include:

* **Taking CodePath courses** which involve building hands-on projects.
* **Starting a blog** (or post on [Medium](https://medium.com/)) to share and journal concepts you are learning. Write about anything you find interesting as you learn.
* **Building a web application** that you can share and link to. Not just a site, but something storing or adapting data which other people can use.
* **Contributing in small ways to open-source projects online** to build up your real-world skills and learn from top engineers in the industry. Most open source projects don’t have a dedicated staff to support them, and rely on developers from all around the world. Check out these [tips on how to get started.](https://hackmd.io/@nesquena/BytxqCfjm?type=view#How-can-we-contribute-to-open-source-software-and-how-do-we-approach-or-get-started)
* **Developing a functional iOS or Android app** and executing it all the way to publication on the App Store or Play Store. (Bonus points if you can get some people using it!)
* **Attending hackathons** and building a prototype of any web or mobile project with a group. Hackathons are a great way to meet and talk to people from various companies. You can even try and meet one of the event organizers.
* If it’s possible at your university, **getting involved with research projects** can be a very rewarding experience. In particular, look for projects involving AI/Machine Learning, Ubiquitous Computing, Data Science/Visualization, Graphics, etc. (Whatever topic you are drawn to learning more about!)
* If you can, **try to attend software conferences and events** nearby. You can often find events on [meetup.com](https://meetup.com/), or even on your campus.

You can pick any of these approaches, or many others, to start building your out-of-class project portfolio. Make sure that you list and describe the most notable of your projects on your resume and your LinkedIn, and put the projects on GitHub with a clear README describing the project where feasible.

### **Is it more important to work on side projects, or interviewing skills?**

Be mindful that no one can do all of these things, and this is **not a checklist to complete**. Finding the time to do well in your classes, study for technical interviews, apply for internships and roles, *and also* work on side projects can feel like a near-impossible task. Not to mention the many other responsibilities you may have!

Ideally of course, you’d want to focus on both side projects and technical interviewing during your time in school.

We recommend choosing a side project that you’re passionate about, rather than one you’re just using to build your resume. It will be a lot more enjoyable, and you will learn a lot more.

You may find you dislike hackathons, or certain clubs or meetups. This is entirely normal; explore around and find what works for you.

The most important thing is to have patience. You don’t have to try to do everything at once, and you don’t need to check every box. This isn’t a race, it’s a long marathon, and **being an engineer is about life-long learning**. Take career development one step at a time.