Starting Your Career

A practical guide

Table of Contents

1. Introduction

If all goes well, in a few weeks you will successfully finish the HackYourFuture program -- a big achievement! However, you still have a long way to go in your career as a programmer.

The next step is to get working experience in a company. We will support you in your journey to employment, but you will have to do the hard work. This document will give you guidelines and a structure which will help you to find employment as a programmer.

HackYourFuture students have been very successful at finding employment after graduation. Over 30 students have graduated from HackYourFuture and even more than that have found employment in the tech industry through following our education. This guide gives an overview of all lessons learned and is based on our experience of helping HackYourFuture students find work.

In order to increase your chances of finding an internship there are three skills you will have to work on: job hunting skills, interview skills and coding skills.

Job hunting skills (chapter 1-3)

In the first chapter we will discuss some general guidelines with regards to starting your career as a programmer and finding that first internship.

The second chapter provides you some practical steps on how to prepare your CV and LinkedIn profile.

The third chapter details how you can search for internships, this will help you to increase the chances of finding employment as soon as possible after graduation.

Interview skills (chapter 4-5)

If you manage to secure an interview with a company, you need to know how to present yourself in a convincing way. Interviews have both a technical and non-technical component. We provide you with some tips and tools on how to prepare for the technical questions (chapter 4) as well as the non-technical questions (chapter 5).

Programming skills (chapter 6)

Quite some students stop coding after graduation, which reduces your chances of finding an internship significantly. The sixth chapter helps you to build your coding

skills post-HackYourFuture. We will discuss how you can improve the basic concepts taught in HackYourFuture, as well as how you can learn more advanced concepts.

Chapter 7 provides you with a checklist to give you some structure after graduation and to work on both technical skills (coding), prepare for technical and non-technical interviews and look for internships.

Finally, in chapter 8, we discuss how you can contribute to our community and continue to learn as an alumnus (graduate student).

Before we go any further:

If you want us to effectively help you on your job hunt, we will need you to have these things prepared:

1. **You have read this guide**
2. **You have a printable CV in .pdf format**
3. **You completed your Linkedin profile**
4. **You cleaned up your GitHub portfolio and added projects**
5. **(extra) An online CV with references to all of the above**
6. **You're constantly working to improve further**

2. Guidelines for finding an internship

This chapter describes some of the general guidelines that are important to consider in order to start your first internship after graduation of HackYourFuture.

1. You are responsible

When people graduate, in general we see 2 stereotypes of students:

Student 1: Keeps on improving after graduation. Continues coding 5 days of the week, works hard to find a company to join and prepares well for the interview process.

Student 2: Thinks that they deserve a job because they have a HYF diploma. Stops coding and waits for a job to come to him/her. Expects to get hired, even though their technical/interview skills might need extra work.

As HackYourFuture we will only introduce you to companies if you work like a student 1.

We have only three full-time employees and therefore limited time. We prefer to spend this time on students who put in the effort. Secondly, students who put in the effort have a much higher chance of getting hired.

Just to make things very clear: if you think that after graduation the hard work is over, and you stop developing yourself as a programmer, and expect us to just “arrange” you a job, we will not, and can not help you.

This guide will give you information on how to control the chance of finding an opportunity. If you put in the effort, HackYourFuture will continue to help you until you are employed. If you do not, it is up to you.

2. Internship > Job

Note that in this guide we will use the word *internship* instead of *job*. The main difference is that an internship is a short-term agreement (usually 3-6 months) in which you work for a company. The most common route to start your career as a programmer is through an internship. Most of our graduates have started with an internship.

Most graduates will start with an internship as most companies want to get to know you better first, before offering you a job. However, many internships result in a job offer. There are many examples of HYF students who started with an

internship and were then offered a contract by the same company. Employment can be both full- and part-time. To find a full-time position is more common and easier than part-time.

There are several differences between jobs and internships (summarized in the table below):

- There are plenty of jobs available. However, to start with a job as a *junior developer* can be challenging as you are competing with more experienced programmers (with 2+ years of experience). For most junior positions some degree of working experience as a programmer is required. For an internship experience is often not required. Therefore, your chances of finding an internship are generally higher.
- If you get hired as an intern, you are not expected to add value to the company from day 1. If you get hired as a junior developer, you are required to be more productive and knowledgeable about programming. The expectations are usually higher.
- Some HackYourFuture students receive public benefits of different kinds ("integrationsydelse", "dagpenge", etc.). That's not a bad thing, quite frankly it's fairly common. Most municipalities and job centers will be happy to see you move into the job market via an internship, as many internships lead to a paid position afterwards. Once you are offered a job, you will receive a salary and stop receiving benefits. The salary depends on the company.
- There are many "grey areas" in the Danish system regarding employment, so if you have questions you can always try to ask us or your case worker for advice.

Generally speaking, we recommend you to aim for an internship at the beginning. The internship then gives your room to learn and gain experience after which you are more likely to land a junior developer role. A classic career path for many newly educated developers is an internship first, after which they are offered a job. Think about it from the company's side: Hiring is very expensive. Being able to "test" a new worker for some weeks before offering them a paid position and contract is a way for them to secure themselves against bad investments.

When we introduce you to an employer for an internship, we will always make sure that their intention is **to hire you after the internship** (if you do well during the internship of course). In practice, many of the internships lead to a job offer**.**

Important!
If you have a case worker/sagsbehandler at the municipality (*Kommune)* it is very important to have a good relationship with them throughout the HackYourFuture course. Especially in smaller municipalities they are not always convinced an internship is the best path as they want you to get out of the public subsidies as soon as possible. Nurturing a good relationship with your case worker can help.

We are more than happy to talk to your k*ommune* or case worker to help

convince them that your internship has a very good chance of leading to a job and salary.

	Internship	**Job**
Difficulty to get interview	Moderate	Hard
Add value	Not from day 1	From day 1
Experience required	Not usually required	Yes (in most cases)
Compensation	Depends on the company, usually is low compared to a salary (job)	Yes (salary)
Kommune/case worker	Usually they are supportive, but need reassurance	Happy :)

3. Be positive and remain calm

It might take some time after graduation before you actually find an internship. It might also happen that you are interviewed for a position and get rejected. It is normal. In both cases, we advise you to remain calm, positive and continue to work on what you *can control* (your coding skills and interview skills).

Some of our best students have taken quite some time to find a good opportunity as well. For instance, Zuhair from class06, applied at dozens of companies but was not hired. He helped us to build our new website and kept improving his technical skills and knowledge of the Danish job market. Finally, he was hired by WizKids A/S - a dream job for him.

4. Choose a specialization

Before you start to think about internships it is important to ask yourself: what type of work would I want to do? There are basically three options available:

- Mostly front-end work (HTML, CSS, JS)
- Mostly back-end work (Node.js, Databases)
- Both (full stack)

This is important as it will influence your CV (chapter 2), which internships to look for (chapter 3) and how to continue coding (chapter 6). You probably already have an idea of what you prefer, but reach out to us if you don't. We can help you to find your specialization.

5. Paying it forward. What about recruitment fees?

HackYourFuture is a **not-for-profit organization**. Over 90% of our budget comes from philanthropy (Den A.P. Møllerske Støttefond is our main funder) and the remaining 10% comes from small grants and partnerships.

We currently don't charge companies any money for the talent we provide them with, but at some point in the future this will have to change. We will always make sure this does NOT affect your application process, or internship compensation and salary. Our priority is always to find you a position first.

HackYourFuture will do its best to help you find an internship after the program. When we do, we will inform companies about any fees we might require and the companies will have agreed to pay this form of compensation.

The educational fee structure looks like this (we will negotiate the exact number depending on the size and situation of the company):

Internship: **a small monthly fee**
Full time position: **one-time educational fee**

If you are hired after an internship the company will only pay the difference to the full educational fee. So the company never pays more than the educational fee. Make sense?

When you find an internship or job yourself, we ask you to inform the company about the fees, because they are very important for our community and the long-term survival of HYF. Fees are used to train the next generation of HackYourFuture students, who come from similar circumstances as yourself.

If you are on track to find a job by yourself, let one of our staff know in advance. We can communicate with your potential employer without reducing your chances of getting the job. You can also provide them with our contact details in order for the company to contact us.

If you have had refugee status in Denmark less than 2 years, then the company hiring you is eligible for a state-funded bonus.

2. Preparing your CV and LinkedIn

The goal of a CV is to show employers that you have the required skills to fulfill the job you are applying for.

Check out this video by Kevin to warm up for making your own CV!

During the project module you will be asked by us to start preparing/updating your CV and LinkedIn profile. Please don't take this lightly. You are learning a lot at HackYourFuture - but if you can't communicate this skill with a good CV then it reduces your chances of finding a job!

Please follow the following instructions to prepare your CV:

1. Before you start work on writing your CV, review some successful examples from our CV database. If you want us to give detailed feedback on a CV, please use the same template and send it to us as an editable Google doc. This way we can provide in-text comments on draft versions, using Track & Changes as well as more general comments. Otherwise the feedback becomes tedious and will take much more time. Once finalized, convert it to .pdf.

2. We suggest to keep the CVs basic and clean. Date of birth, marital status, country of origin are not advised. Address is only advised if you live somewhat close to the city/company.

3. Write a short introduction on top of your CV about what type of person you are, what work you are passionate about, and what type of work you are looking for. This so-called summary-statement should not be longer than 2-3 sentences. Keep it simple.

4. Add a professional picture: Before taking your photo, make sure you use a background that isn't distracting, dress for the job you want, smile and have a friendly expression. Also, be sure that your face takes up the majority of the image. No selfies!

5. Put the most important and relevant accomplishments first. It is also good practice to start with your work experience followed by education afterwards. Some of you might have limited work experience. Often, participation in the HackYourFuture course can be featured on top. It might be useful to spend a few lines of text on describing the final project you worked on.

6. When mentioning previous jobs, describe clearly what you did in the company, what your responsibilities were, and mention successes (if any). Work experience not related to web development should be listed on the CV if the skills used are

relevant (project management, customer interaction, leadership, etc.).

7. As a recent HYF graduate, it is very likely that you don't have a lot of experience with programming. That's ok. A good way to show what you can do is by writing about the best projects you've been working on, including a Github link to your (clean!) code and include links to your website and LinkedIn. Make sure to write what those projects are about (shortly) and what your role was in the team, what technologies you used and maybe even what you have learned.
8. Make sure to only list skills/competencies (soft and technical) you actually possess. More than 10 skills is overkill.

9. Be sure your CV has no spelling mistakes! Make sure somebody who has a high proficiency in English writing and reading checks for mistakes (we all make them).

10. We believe a good CV is a maximum of two pages (ideally one). Recruiters usually review hundreds of CVs every week so you need to get their attention quickly. A clean and structured format, a professional picture and the right insights into your previous experience and education usually go a long way.

NovoRésumé Partnership
You can find a lot of easy-to-use templates here: NovoResumé

We have an agreement with NovoRésumé who kindly offer premium accounts to all HYF students who would like one. (Make an account, then send Christopher the email address you used and he can unlock the premium version for you).

For more info on how to write a solid programmer cv check this article that has lots of great ideas!

Updating your Linkedin profile:

We know what you might be thinking: Why Linkedin? Can't I just *not* have a Linkedin? The answer isn't always simple, but look at it this way: Linkedin can be a really powerful extra platform for your job hunt. Not all jobs are advertised publicly, so Linkedin can actually help companies find YOU, instead of the other way around.

It also is the place where you can gather a professional network (and make it visible to recruiters). At HackYourFuture you will meet a minimum of 15 professional developers personally during your classes, more than 50 others are connected via our Slack channel. Connect with them!

On top of all that Linkedin is also a great place to read up on companies and get an insight into who works there. Check out Zoey's profile for a good example on how to structure your content and Welcome Group organises workshops on the topic once a month - check their events here.

3. How to search for internships

There are two ways to find your first internship as a programmer:

Option 1: Finding an internship through the HYF partner network (we connect you to a company)
Option 2: You find an internships yourself

HackYourFuture will support you as good as we can in your task of finding a job. We have a network of partner companies and we will do our best to investigate openings and make an introduction for you.

However, we strongly advise you to be involved in this process as well (option 2). Helping HYF's staff to find a position that suits you well will increase your chances of finding the job YOU want.

We have only so much time to search for internships, so if you want to increase your chances, you will have to put in effort. This is also a great opportunity for you to gain job-hunting skills which will come in handy for the years to come.

Here are a few tips to use when looking for jobs:

- Linkedin.com, thehub.dk, Indeed.com, angel.co/jobs and whoishiring.io are great sites to get you started!
- Look for jobs that have been posted recently (not older than 2-3 weeks), and that list skills and technologies similar to those you gained throughout you HackYourFuture education.
- Look for companies with medium (6+) to big sized programming teams. You can do this by looking for the company's LinkedIn and exploring how many and what type of employees they list on their page.
- Take a close look at their company profile. Do they have an international team? Is the company's profile one you would find yourself comfortable working in?

If you find opportunities you are very interested in, communicate with the HackYourFuture staff via Slack to discuss next steps. We will assess the job postings and contact the right person in the organizations you selected to make an introduction to the HYF program.

Should I contact companies myself?

Of course you are free to contact companies yourself. However, sometimes there can be a better response-rate when HackYourFuture makes the introduction, as companies feel more obliged to answer to an organisation than an individual. Also, when you apply to a position yourself you often have to draft a motivation letter (in perfect English!) and compete with other job applicants, often with much more experience. Finally, remember we would still like to negotiate an *education fee* with the company so we can train the next batch of students!

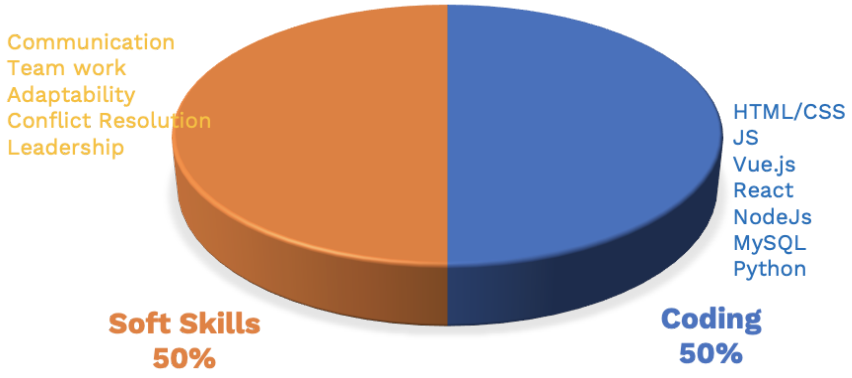
4. Non-technical interviews

Interviews will usually consist of both a non-technical and a technical component. Often the non-technical part is neglected. In essence, in this part of the interview, the company wants to get to know what type of person you are, and if you fit into the company culture.

Why is it important to prepare for the non-technical interview?

1. This is often at least 50% of the interview.
2. With some practice you can easily improve your performance in the non-technical part of the interview! With only a few hours of serious practice most of you can make sure to do very well with non-technical questions. This is much more difficult with technical questions!

In this section we will give you a few resources and tips to start practicing.



Resources & Tips
This document provides you with the Top 10 **non-technical** Interview Questions and Sample Answers.

And here is another document we have created with the 29 most common interview questions, including “good” answers.

Both documents will give you a general overview of the general structure of a good answer. Please read them carefully.

It is important that you think of the following:

1. Provide Specific Evidence
You have to provide credible evidence to support your answer. Give at least one, but preferably two, arguments to support your answer. So, if you are asked about

your strengths, provide the interviewer with at least two strengths.

Subsequently, you need to provide examples and details to back up your arguments. For instance:

I work hard (argument 1) → In seven months I graduated from HYF without any background in programming (evidence)

I like to work in teams (argument 2) → In my previous career I was in charge of a team (evidence)

2. Adjust your answer to the company

To improve your answers it is useful to adjust it to the specific context in which you have been asked the question and adjust it to the realities of the company you are applying to. For instance if you are asked why you are a good developer and your are applying to a company that works on highly advanced technology, you could use this:

I think I am a good developer because I constantly keep on learning new things. Even when I was a student at HYF I tried to provide multiple answers for a homework assignment [give an example]. I also built my own project in ... (e.g. Vue.js) [explain project].

I think that fits well with the DNA of this company. From conversations with your employees and your website I have learned you are constantly trying to incorporate new technologies and frameworks [give an example of a new framework the company uses and explain why you would like to learn it].

To give such an answer, you need to research the company you are interviewing for. You want to know at least the following:

- who works there (including those that interview you),
- what technologies are used,
- which clients is the company working for,
- what kind of working environment do they have (informal/formal, international/Danish, more technical or more commercially oriented, etc.)

3. Body Language

Even if you give the perfect answer, if you body language is weak you will have a difficult time to impress the interviewers. Here you can find some tips on how to improve your body language. Here is a helpful video.

4. *Ask questions yourself*

It is important that you prepare some questions yourself as well. Having some

questions ready shows that:

a) You prepared for the interview which is a sign of motivation

b) It re-establishes the balance. An interview is not a one-way street in which you need to prove your value. They also need to show their value to you. By asking questions you create more of an equal dynamic.

Practice
After reading and watching the content above, start practicing. The way you practice is the following:

- Spend about 30 minutes a day answering three questions, for one week in total.
- Record your answer to three random questions and send it to our staff
- We will provide you with feedback.

Once you graduate we always try to organize practice interviews for you with both a general interviewer to test your non-technical skills and a technical interviewer. If you want this practice, be sure to be proactive about helping us setting it up! Schedule a suitable time with your classmates

5. Technical Interviews

In every interview you will be asked technical questions about programming. The type of questions will depend on what role you are applying for (front-end/backend), and whether it's for an internship or for a junior developer role.

In addition to the interview, you will often get an assignment in which you will have to build a certain small program that serves a certain purpose, with the aim to see the quality of your code. We have collected some tests from relevant Danish companies here.

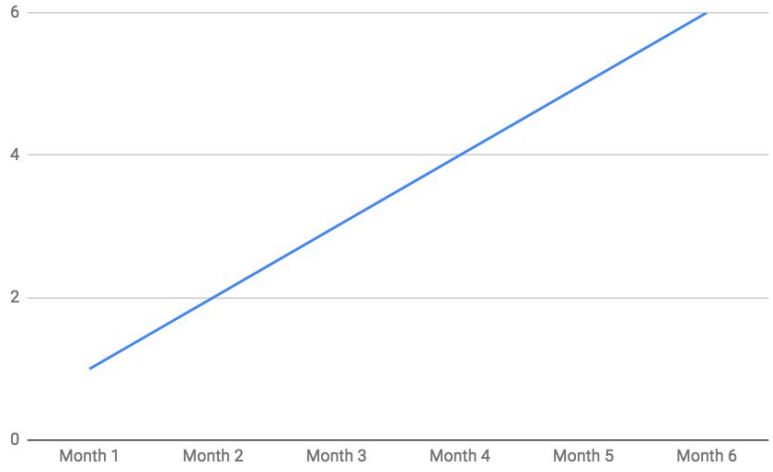
You can find a massive list of JavaScript questions that will be very helpful to be able to answer here:

1. https://www.guru99.com/javascript-interview-questions-answers.html
(includes a small test)
2. https://www.toptal.com/javascript/interview-questions
3. https://github.com/ganqqwerty/123-Essential-JavaScript-Interview-Questions
4. https://www.edureka.co/blog/interview-questions/javascript-interview-questions/
5. https://www.geeksforgeeks.org/commonly-asked-javascript-interview-questions-set-1/
6. https://github.com/h5bp/Front-end-Developer-Interview-Questions
7. https://github.com/MaximAbramchuck/awesome-interview-questions#javascript
8. https://github.com/yangshun/front-end-interview-handbook/blob/master/questions/javascript-questions.md

In addition to this, make sure to go through your graduation project again. You should be able to explain your code in detail. If you are asked about the project, this is the part of the technical interview that you can control.

6. Coding Skills

In the last 7 months the improvement of your coding skills would look a bit like this in a graph:



Now that you have graduated, but do not have a job yet, it is VERY important, that you keep this line going higher and higher.

In our experience, many students after graduation either reduce the number of hours of coding or even stop coding altogether. If you do not code, and don't improve yourself, this line will actually start going down as you will probably will forget important knowledge. Finding employment will be very hard.

Everyone deserves a small break after the intensive months at HYF, but remember that you're not in the goal yet! So as long as you do not have a job your task is to focus on improving yourself as a programmer so that your chances of finding a job increase.

This can be divided along two categories:

1. Understanding the Fundamentals

The HackYourFuture program moves fast, very fast. Often students tell us they did not have ample time to fully grasp the concepts that they were taught. Now that you have graduated there is no excuse not to revisit those basics. This might seem boring, but it is very necessary. If you have the ability to turn your weaknesses into a strength, very little stands in your way. It is advised to focus on the weaknesses within your specialization. So if you want to be a frontender, studying MySQL, instead of React for instance, into detail is probably not the most effective use of your time.

How to do this?
1. Go over the HYF curriculum checklist and see what parts you are lacking knowledge about
2. Re-do the exercises, without looking at your old code. Identify your weakness(es).
3. Come up with exercises yourself for the concepts you are trying to learn

2. Learning more advanced concepts / Specialize

After improving the fundamentals, it's time to look further into learning new concepts. It's most important to choose these topics strategically, taking your specialization in consideration. If you are aiming for a front-end position for example, it's a great moment to dive really deep in understanding html/css better, and diving even deeper into understanding advanced React concepts. This will help you convince your future employer in an interview, as you will have more in-depth knowledge. It's important not just to understand these new concepts, but to actually also use these concepts in real life code.

How to do this?

1. Build an extra feature to your project

You can contact the teachers that were involved in your graduation project and work with them to determine which feature could be added. Perhaps you have an idea yourself already.

2. *Build personal projects*

Personal projects can help to spark the interest of recruiters and hiring managers. Personal projects show you are passionate about programming. Also, it helps you to improve your coding skills after HackYourFuture. A personal project can be anything programming related, whether it be a website, a small JavaScript game, a mobile application, etc. We have collected 4 project tutorials to help you demonstrate self-initiative and genuine interest in this field.

Project 1: Build a Quotes Machine | 1 week | Link
Project 2: Build a Sound Machine | 1 week | Link
Project 3: Build a JavaScript Calculator | 1 week | Link
Project 4: Build the Simon Game | 1 week | Link

Each project should take a maximum of 20 hours. If necessary, you can link yourself up to a mentor and ask them to have a look at your code at the end of the project. After that you have to make improvements based on the feedback.

Important! It is necessary that you keep us in the loop about your progress. We

expect you to build at least a few new projects over the 2 months following your graduation. You can do the ones we suggested, or even better: work on personal projects you are passionate about!

Include your best ones on your resume as soon as they are finished.

7. Checklist

Name: **Class:** **Specialization:**

Weeks	Checklist	Notes
Week 0: Project Presentation		
Week 1-2: Project 1	❏ Make Pull Request for Personal Project 1 and publish it ❏ Ask Teacher for Feedback ❏ Implement Feedback	
Week 3-4: Project 2	❏ Make Pull Request for Personal Project 2 and publish it ❏ Ask Teacher for Feedback ❏ Implement Feedback	
Week 5-6: Project 3	❏ Make Pull Request for Personal Project 3 and publish it ❏ Ask Teacher for Feedback ❏ Implement Feedback	
Week 7-8: Project 4	❏ Make Pull Request for Personal Project 4 and publish it ❏ Ask Teacher for Feedback ❏ Implement Feedback	
Week X: Career Training	❏ Attend career training ❏ Attend mock interview	

8. Life as a HYF Alumnus

After your graduation you will join the **#alumni** channel in Slack, this will be the main information source for everything related to alumni matters in HYF. However, there are several ways to stay directly involved with HackYourFuture, detailed below. This is really important because HackYourFuture is only possible because of volunteers. If you want us to help new students just like we have supported you, we can use your help!

Homework support *(+/- 3 hours. Flexible)*
Every week students have to hand in homework assignments. These assignments are key to measure their progress and to put into practice what they have learned. To have an extra check on how the students are doing there are three tests inside the curriculum. (after Javascript 2, Javascript 3 and React).

As a homework assistant you will check and provide feedback on the homework of the students and optionally revise their tests. Feedback should be provided within a week after homework has been submitted. Giving feedback in time is crucial as the next assignment builds on the knowledge gained throughout the previous homework assignment. This is a good example of detailed and helpful feedback.

1:1 Mentor *(1 hour Sunday, 2-4 hours throughout the week . Flexible)*
Some students need additional support throughout the course as the speed of our program is a challenge for them. That's exactly what you will give them as a 1:1 mentor. You will help students by:

1) Assessing which concepts they are struggling with
2) Recapping these concepts by explaining and using examples of code
3) *Supporting* them with their homework (as opposed to making it *for* them)

It is important to make clear agreement beforehand about the extent to which you will provide support and when. You can meet the student you are mentoring on the Sunday but due to the information overload of the class, most of the support will have to occur throughout the week. We ask you to organize at least one, but preferably two meetings or video calls a week of at least an hour.

Teaching Assistant (TA) *(4 hours Sunday + 1 hour preparation + ~2 hours homework* support)

As a TA you provide support to the leading teacher. He/she might ask you to prepare presentation material to explain some of the concepts covered in class or an exercise to practice the theory. In some cases you are also asked to teach a

small part of the class. In addition, you will help the students when they are working on an exercise and get stuck. You can provide them with an additional explanation, and steer them in the right direction. In this way we reduce the risk of students falling behind during class. Optionally, you can also help out with checking the homework.

Teacher *(4 hours Sunday, ~3 hours preparation + ~3 hours homework support)*

During the week you will help providing feedback on the homework in GitHub and prepare your lecture. You have the freedom to structure your own lecture though the topics and guidelines for every week are provided by us in line with the curriculum. In practice this will mean that you have to study the concepts taught in the module and make a lesson plan containing both theory and practice. For reference, you can look at previous classes on our YouTube channel. Finally, if possible you will also help out with checking the homework.

To get used to our way of operating, you will start as a TA or homework support (or both). In principle we require you to participate in at least one module before you can start teaching or become a coding mentor. To gain experience you will be paired up with a more experienced teacher.

Together we can take a look at our teaching schedule. We check when the modules in which you want to contribute take place, and make a plan for the next steps.

Workshops / Masterclasses
From time to time we organize workshops or alumni masterclasses in which we go into detail with a specific topic. We have organized masterclasses on .NET development, Web Bluetooth/Web USB, Web Content Management & CMS Systems, Docker and many more. If you have a suggestion for a topic, please let us know!

Currently, we are especially looking for:
- Homework Assistants
- 1:1 Mentors
- Teachers who can help out in the final Project Module (7 weeks including the final presentation)

We are always open to other suggestions as well. If you have a great idea, please let us know. **Some examples**:

a) Some alumni have opened doors at companies in their network or at the company they work for. This has led to many internship positions and jobs for other students!

b) Many alumni have helped out to further develop the curriculum. If you see any

room for improvement - let us know!

c) Some alumni organize social events and study groups to keep our community alive and thriving!

Let’s go! :)