

Frontend

how to

September 30, 2021



uocode

Contents



Introduction	2
Program overview	3
Where do I begin?	6
How to succeed?	7
Epilogue: What now?	8

Introduction



Welcome on board!

By joining this short program you've made your first step on the road to becoming a frontend developer. If you're feeling a little lost or overwhelmed, that's normal. This document will help you understand the program a little better.

What challenges can you expect to encounter? How will your studies be structured (and will they)? What will you learn and accomplish? We'll cover a little bit of everything.

Why frontend?

So, you've been thinking about starting a career in IT and have explored various options. Your choice fell on frontend development. Why so? Many people appreciate the fact that it makes their work more tangible: you can see what you've just created. One of the issues people often have with working in IT is that you work with 'invisible' things - you cannot see algorithms and lines of code. When you code for the backend, you may be working on a program that does not have much of an interface, even if it does quite powerful things. If that level of abstraction is not for you, then you were right to start with frontend!

Why this program?

Why is this program designed to fit into 5 months? Is that enough time to master new skills? **Yes, if you're up to it.** The main purpose of this program is to familiarize you with the key frontend development principles and give you sufficient skills to start working on real-life projects.

You will probably still make mistakes (and gain new knowledge in the process), but life is a never ending learning process and the only way to avoid mistakes is to stand still and do nothing. With this program, you will gain enough knowledge to decide which way to move forward.

Program overview



Why Challenge Based Learning?

You will learn by overcoming challenges (more on this in our CBL guide). Why challenges and not just goals? A goal, by definition, does not specify the level of complexity. It refers to a result that you achieve, but it is not necessarily something you need to work hard for. A challenge, on the other hand, always entails something that you will find, well, challenging. It is something that is never perceived as easy, something that always requires you to work hard and overcome potential obstacles. From this comparison, we can see that challenges promote growth, whereas simple goals might not always do so.

Setting goals in the form of challenges gives you additional motivation to work hard. It also gives you a more pronounced feeling of accomplishment. Each challenge is set in such a way that, to overcome it, you need to master a set of skills that will be useful in making you a successful developer. This allows you to feel how your skillset grows with every new accomplished task.

Challenge Based Learning is a great framework for problem-solving in any sphere, not just in a learning environment. As you get used to it while working on different challenges, you will notice how it can be useful in everyday situations and in the workplace in particular. You develop adaptability and learn to find relevant information quickly and efficiently.

What's peer-to-peer learning?

There's only so far you can go on your own. In a traditional school setting, teachers are the ones who provide support to the learner. So, what should you do when there are no teachers? Keep studying! In fact, the teacher-student model, though still widely used, is not the most effective in the modern learning environment. One teacher is usually attached to a rather large group of students, so they can rarely offer personalized involvement.

What can be done to mitigate this problem? Peer-to-peer (p2p) learning, of course! In this way, each student can also be a teacher for their peers. According to p2p, you always change your role. If you know something, you can share it with others. And when others know something that you don't, they will share that knowledge with you. An important sign that you've truly understood a concept is when you can explain it to someone else, and peer-to-peer learning gives you the perfect opportunity to check this. In an environment that requires students to share their understanding of different concepts with others, students are more likely to develop a deep understanding of each topic.

Peer-to-peer is not about letting another student use your solution. That would be of no use to either party. Rather, it's about offering more support, brainstorming problems together, sharing ideas and insights.

Besides, p2p helps the students to develop communication and networking skills, enhances their confidence, and prompts them to ask the right questions and respect the input of others. All these soft skills boost the future professional life of students, as they are all required in any work environment.



Our methodology

Overall, this program is based on a combination of CBL and p2p, as discussed above. Students work towards overcoming specific challenges, and they support each other in the process of learning.

Together, these two approaches create an environment where each individual works towards quality results that can bring real value, as opposed to simply ticking 'done' on tasks without understanding their intended purpose.

While it may be easy to overlook at first glance, the reflection process quickly proves to be one of the most important aspects of learning. This is where you can think about your achievements and areas of potential growth. By evaluating your own progress (with honesty!), you take control of your career and life in general, instead of just going with the flow and wondering why you're never lucky.

General structure of the program

Before you begin the program, you need to complete the **Warm-up** challenge. It is designed to prepare you for studying with ucode in general. It will help you navigate through the communication channels, the learning management system (LMS), and the peculiarities of remote studying.

The program is divided into five sessions: four longer studying sessions and one short session to wrap everything up and get ready to take your first steps in the new profession.

The first three sessions are, in turn, divided into Sprints and short challenges. Sprints contain short tasks that are designed to help you develop the necessary skills and get you acquainted with the technology. They have progressive complexity, so you begin with shorter and simpler tasks and move on to the more challenging ones until you are ready to work on the challenge that follows.

The **first** session (session 0) helps you navigate through the basics of programming. Whichever programming path you choose, there's no escaping the Terminal. You might not need to work in it all the time, but you will nonetheless need to interact with it one way or another. To avoid embarrassing moments in the future, you'll get acquainted with it in your very first (zero!) sprint. When you master the basics of working in the Terminal, you will start learning HTML and CSS - the main junior-level frontend developer instruments. You will wrap this session up by creating a portfolio website using the skills gained while working on the first two Sprints.

Keep in mind: this is not just a task to test your skills. Strive to build a website that you can use as a real portfolio.

The **second** session (session 1) starts your journey in the world of JavaScript. You will spend this time getting acquainted with the main principles of working with the DOM and the ways JS can help you with it. This session ends with another major project: an online store catalog, which you will build using your newly gained knowledge.

The **third** session (session 2) brings more JS practice as well as new peaks to reach: working with APIs and introduction to React. First, working on the Sprint, you will get to practice more JS tricks and learn to use APIs. Then comes the bigger challenge. You'll delve into both React and APIs by writing your own version of StackOverflow using their



API. Don't be afraid to learn new concepts: that's the basis of Challenge Based Learning. If you only do what you know, you'll never gain new skills.

The **fourth** session (session 3) is dedicated to your creative project. You'll get together with one or two of your peers to form a small team and work on a game. For this, you will research React in more detail and learn to use or even to create your own game engine. It may seem overwhelming: you need to get from a basic concept to a working product in about one month! But don't let it freeze you. To accomplish something, you need to start working on it. Once you take the first steps, the project will seem less and less intimidating with each new development stage. You'll be amazed at what you can achieve with proper motivation and coordinated teamwork.

The **fifth** (and final) session (session 4) is all about getting you ready to enter the job market. You'll revisit the portfolio you created during the **first** session. In addition to updating your portfolio, you will also prepare a resume, which will be added to your web page. Don't be disappointed if you don't like your 'old' project - you've grown and gained new skills. This is your chance to optimize and improve your code as well as adding your new projects to the website.

Where do I begin?



Short answer: from the beginning.

That means going through all sections of the **Warm-Up** challenge, logging into your LMS account, and setting up remote work if you need to.

This program is designed to take you from the very basics of programming to the more advanced and nuanced aspects of frontend development in particular.

So, take a deep breath, drink some water, gather your courage (if you need to), and open **Sprint00**. Go through the Engage and Investigate sections and get ready to tackle the first tasks in Act. If you take your time to research all guiding questions and complete the guiding activities, you should be ready to start completing tasks. If you still have questions, immerse yourself in the peer-to-peer experience: ask if your neighbors understood the questions you found challenging. Use the corresponding slack channels to ask your remote peers.

Although the first three sessions require only individual work on each task, you can (and should!) communicate with each other on every stage and discuss the challenges and their solutions. **That is the power of peer-to-peer learning.** Don't shut down while working on a task. Your peers might give you some valuable advice (they may point out something you have overlooked or haven't thought of yet) and feedback (perhaps they can give you suggestions on how to improve your solution and make it more elegant). Similarly, you can share your knowledge and experience with other students, which is also a gratifying experience.

How to succeed?



Some basic recommendations:

- carefully go through all guiding questions and complete all guiding activities
- don't be afraid to ask questions and communicate with your peers
- but with any question you have, google first, then ask others
- learn to appreciate opinions given by other people
- if you know the answer to a peer's question, offer your advice! It will be appreciated
- face each challenge with a clear head, open mind, and good mood
- allow yourself to make mistakes - that's the only way to learn something new
- if you begin working on a challenge, don't set it aside for too long, or you might forget all the best solution ideas
- don't postpone all tasks till the last minute
- show up for all assessments on time

Something to keep in mind

It's up to you how you organize your learning schedule. However, for most people, regular studying in shorter periods of time brings better results than longer, but more spaced out, studying sessions. It's optimal if you can dedicate a couple of hours every day to studying. This way you will gradually learn new skills and your brain will have time to process new information (but not enough time to forget it without proper practice).

Avoid starting a studying session if you know you're already tired and won't be able to focus properly. It's best to take your time, have the rest you need, and get back to work full of energy.

You won't have strict deadlines and no one will be pushing you to meet a specific deadline for each task, but it's in your best interest not to fall behind the group - this way you'll be able to benefit from the power of p2p the most. Pay attention to the estimated time for each challenge.

Epilogue: What now?



You are now ready to take on a new adventure - `the frontend program`! It may be difficult at times, but we promise it will all be worth it.

Remember that you can always revisit this document (and other useful materials) if you have any questions - it's natural that your memory won't store all the information you've just read.

You've got a lot of work ahead, so get started!