

I can help you becoming a full stack developer:

Presentation:

Become a full stack developer with my help! I have more than 10 years experience as a teacher and 3 as a software developer!

This course aims to teach you not just the most used tools the market is using right now (2022), but also: fundamentals of computer science (which are very important for a complete developer) and market best practices and organization methodologies.

All of that with 34 projects where you are going to learn by DOING!

And, during the course, if I fell like you are prepared I can help you getting a job (not guaranteed, but I will do my best).

Study program:

1. **Software development fundamentals** - 30 lessons

1. GitHub - 3 lessons
2. Html & CSS: intro * - 4 lessons
3. Introduction to JS and programming logic * - 4 lessons
4. JS: DOM, events and WebStorage * - 5 lessons
5. HTML & CSS: Forms, flexbox, responsive * - 6 lessons
6. JS ES6 & unit tests * - 3 lessons
7. HoF * - 5 lessons

2. **FrontEnd**: - 21 lessons

1. JS & async tests * - 3 lessons
2. Intro to React * - 2 lessons
3. Components, state, events and forms * - 2 lessons
4. Life cycles * - 2 lessons
5. Agile methodologies * - 1 lesson
6. Automated tests: RTS * - 3 lessons
7. Redux * - 5 lessons
8. Project
9. Context API & Hooks * - 3 lessons
10. Project

3. **BackEnd**: - 41 lessons

1. Docker Intro * - 3 lessons
2. SQL Intro * - 4 lessons
3. SQL Advanced * - 3 lessons
4. Creating APIs * - 5 lessons
5. Architecture * - 3 lessons
6. ORMs & authentication * - 4 lessons
7. Deployment * - 2 lessons
8. Typescript ** - 3 lessons
9. OOP e Solid * - 5 lessons
10. Project
11. MongoDB Intro * - 5 lessons
12. MongoDB + API + OOP * - 2 lessons
13. Project
14. VPS + CI/CD + DNS - 2 lessons

4. **Computer science**: - 20 lessons

1. Intro to python ** - 3 lessons
2. Design Patterns * - 3 lessons
3. Web Scraping ** - 3 lessons

4. Algorithms * - 3 lessons
5. Data structure I: Arrays, Lists, Queues, Stacks * - 4 lessons
6. Data structure II: HashMaps, Set * - 2 lessons

* A project is included at the end of this subject ** This subject can be skipped if you want to do BackEnd and CS classes using C# - if you choose to skip these lessons the total price of the course will be reduced accordingly

Study program - detailed:

1. Software development fundamentals - 30 lessons

1. GitHub - 3 lessons

1. Understanding Git and how it works
2. Basic commands
3. Internet - How it works?

2. Html & CSS: intro * - 4 lessons

1. Page basic structure
2. CSS basics
3. CSS: selectors and positioning
4. Semantic HTML
5. Project - Lessons Learned

3. Introduction to JS and programming logic * - 4 lessons

1. First steps into JS
2. Array e for loop
3. Programming logic and algorithmics
4. Objects and functions
5. Project - playground functions

4. JS: DOM, events and WebStorage * - 5 lessons

1. JS - DOM and selectors
2. Working with elements
3. Events
4. Web Storage
5. Browser tips and tricks
6. Project - Pixel Art + Bonus projects (optional)

5. HTML & CSS: Forms, flexbox, responsive * - 6 lessons

1. Forms
2. Libraries and frameworks CSS
3. Flexbox - part I
4. Flexbox - part II
5. Responsible CSS - Mobile first
6. Project - Hogwarts school of magic

6. JS ES6 & unit tests * - 3 lessons

1. let, const, arrow functions and template literals
2. Exceptions and objects
3. Jest
4. Project - unit tests

7. HoF * - 5 lessons

1. HoF introduction
2. forEach, find, some, every, sort
3. map and filter
4. reduce
5. spread operator, rest operator, destructuring
6. Project - Zoo functions

2. **FrontEnd**: - 21 lessons

1. JS & async tests * - 3 lessons
 1. Asynchronous programming and callbacks
 2. Fetch API + async / await
 3. Jest - async testing
 4. Project - Shopping cart
2. Intro to React * - 2 lessons
 1. React basics
 2. React Components
 3. Project - Solar system
3. Components, state, events and forms * - 2 lessons
 1. State & events
 2. Forms on React
 3. Project - Triumphy
4. Life cycles * - 2 lessons
 1. Components life cycle
 2. React Router
 3. Project - YourTunes
5. Agile methodologies * - 1 lesson
 1. Scrum & Kanban
 2. Project - Online Store
6. Automated tests: RTL * - 3 lessons
 1. RTL Basics
 2. Mocks & Inputs
 3. Testing React Router
 4. Project - React Testing
7. Redux * - 5 lessons
 1. Global State
 2. Redux with React
 3. Practice Redux
 4. Async Actions
 5. Tests with Redux & React
 6. Project - Digital Wallet
8. Project - Trivia game
9. Context API & Hooks * - 3 lessons
 1. Context API
 2. useState & useContext
 3. useEffect & Custom hooks
 4. Project - Starwars Datatable
10. Project - Recipe App

3. **BackEnd**: - 41 lessons

1. Docker Intro * - 3 lessons
 1. Basic about containers and docker
 2. Manipulating docker images
 3. Orquestration with docker compose
 4. Project - To Do List with docker
2. SQL Intro * - 4 lessons
 1. Intro - Relational Databases and DB modelling
 2. Querying a database
 3. Filtering data
 4. Manipulating tables
 5. Project - All for one
3. SQL Advanced * - 3 lessons
 1. Most used functions on SQL
 2. JOIN's - 'easy pizzy'
 3. Advanced modelling of Databases
 4. Project - One for all
4. Creating APIs * - 5 lessons
 1. Intro - Framework to create a server
 2. API Rest with a framework
 3. Integration tests
 4. Middlwwares

- 5. BackEnd Framework + Sql
- 6. Project - Talker Manager
- 5. Architecture * - 3 lessons
 - 1. Model layer
 - 2. Services Layer
 - 3. Controller layer
 - 4. Project - Store Manager
- 6. ORMs & authentication * - 4 lessons
 - 1. ORMs basics
 - 2. Associations
 - 3. JWT
 - 4. API integrated tests
 - 5. Project - Blog API
- 7. Deployment * - 2 lessons
 - 1. Deploy basics
 - 2. Deploy Docker & Heroku
 - 3. Project - Stranger Things
- 8. Typescript ** - 3 lessons
 - 1. TS basics
 - 2. Static typing & generics
 - 3. TS with backend framework
 - 4. Project - Smith
- 9. OOP e Solid * - 5 lessons
 - 1. Intro to OOP
 - 2. Inheritance and interfaces
 - 3. Polimorphism
 - 4. SOLID: S, O and D
 - 5. SOLID: L and I
 - 6. Project - Dragons
- 10. Project - Futebol club
- 11. MongoDB Intro * - 5 lessons
 - 1. NoSql and Mongo intro
 - 2. Filter operations
 - 3. Query operations
 - 4. Simple updates
 - 5. Complex updates - arrays
 - 6. Project - Commerce
- 12. MongoDB + API + OOP * - 2 lessons
 - 1. MSC architecture - model
 - 2. MSC architecture - Service & controller
 - 3. Project - Car Shop
- 13. Project - Delivery App
- 14. VPS + CI/CD + DNS - 2 lessons
 - 1. Connecting to a virtual private server
 - 2. CI/CD and pipelines

4. Computer science: - 20 lessons

- 1. Intro to python ** - 3 lessons
 - 1. Python basics
 - 2. Input / output
 - 3. Tests
 - 4. Project - Job Insights
- 2. Design Patterns * - 3 lessons
 - 1. OOP with a programming language
 - 2. Patterns: Iterator, Adapter, Strategy
 - 3. Patterns: Decorator, Observer, Factory
 - 4. Project - Storage
- 3. Web Scraping ** - 3 lessons
 - 1. Web structure, and security
 - 2. Data scraping
 - 3. Other scraping tools
 - 4. Project - Tech news
- 4. Algorithms * - 3 lessons
 - 1. Algorithms complexity
 - 2. Recursivity and other strategies
 - 3. Ordenations and search algorithms
 - 4. Project - Algorithms

5. Data structure I: Arrays, Lists, Queues, Stacks * - 4 lessons
 1. Computer architecture
 2. Arrays
 3. Nodes and concatenate lists
 4. Queues and Stacks
 5. Project - We are not Google!
 6. Data structure II: HashMaps, Set * - 2 lessons
 1. Hashmap & Dict
 2. Set
 3. Project - restaurant orders
-

Course routine:

I offer 3 possibilities: 1. 1 class / week 2. 2 classes / week 3. 3 classes / week (weekend included)

A normal class day consists of: 1. I send you the material to be studied in ADVANCE to our class: 1h30 - 2h30 of studies 2. I give you a class: 1h or 1h30 (to be agreed) 3. You do the exercises or project: 2h - 3h exercises / 1 week project 4. Both projects and exercises have automated correction, if you need personalized help you can pay for single monitoring sessions

Prices:

1. Price / lesson: \$90 / 1h or 120 / 1h30min
2. Price for the material ~~5.000~~ - free
3. Single monitoring session (to solve doubts about exercises): \$50 / 1h

Optionals:

1. CV and Pitch consulting: \$150 / 1 session (1h30min)
2. Job interview simulation: \$300 / 1 process (3 interviews: 1 of 30min, and 2 of 1h)

Full course price:

1. 1h lesson: $90 * 112 \text{ lessons} = \10.080
 2. 1h30 lesson: $120 * 112 \text{ lessons} = \13.440
-

Details:

* All this course codes MUST be written in english, this is to help you, because it's the standard of the market!

Lesson + project quantities:

1. 112 lessons
2. 30 simple projects + 4 complex projects

Default durations:

1. Simple project: 1 week
2. Complex project: 2 week

Course duration:

1. 1 class / week: 2 years and 4 months lessons + 9 months projects = 3 years and 1 month
2. 2 classes / week: 1 year and 2 months + 9 months projects = 2 years and 2 month
3. 3 classes / week: 9 months + 9 months projects = 1 year and 6 months

* Plus eventual extra time to finish projects ** The durations can be even more reduced if you are capable, and I have the time, to do more than 3 lessons per week OR if you are capable of having 2 subjects on the same class (this last option WON'T change the total price of the course)

Payment conditions:

1. Monthly: All the classes of a month are paid at the end of that month - no discount
 2. Monthly ahead: All the classes of a month are paid at the begging of that month - 5% discount (\$9.576 / 12.768)
 3. 50 - 50: 50% paid at the beggining of the course, 50% paid at the end - 7.5% discount (\$9.324 / 12.432)
 4. Paid ahead: The whole value is paid ahead - 10% discount (\$9.072 / 12.096)
-

I can help you get a job:

1. If you are performing well during our class and I fell like you are ready to enter the market before the end of the course I can help you getting a job since I have contacts with many companies (not guaranteed, but I will do my best).
2. If needed I can help you creating a CV and a pitch to help you with job interview (see prices section).
3. If you get a job before the end of the course we can either reduce the number of lessons / week and increased the time to deliver each project OR we can transform the rest of the course on mentoring sessions so that I help you with your day to day activities at work (if you have choose 50 - 50 payment you MUST pay the rest of the course even if you get a job).