

What should students  
learn?

# Who am I?

- I wrote my first code when I was 3 or 4 years old. It was Assembler for Atmega microcontrollers;
- 5 years of experience in commercial sector;
- Developer Advocate at Onix-Systems;

# IT directions at the moment

- Backend Development;
- Frontend Development;
- iOS Development;
- Android Development;
- Embedded Programming;

# Wow... wait a minute!

I forgot about qualities and skills

# Necessary qualities and skills

- Patience and ability to continuously learn;
- Logical thinking;
- Linux;
- Terminal;
- Git;

We can return to IT  
now

# Backend Development

- Backend consists of an application and database;
- Server-side languages: Ruby, Python, NodeJS;
- Databases: MySQL, PostgreSQL, MongoDB, Neo4j;

If you prefer to manipulate data, then your choice is backend development



# Frontend Development

- Frontend Development consists of HTML, CSS, JavaScript;
- Frontend Development changes constantly;
- HTML: Handlebars, Jade, Mustache and many others template engines;
- CSS: PostCSS, SASS, LESS are supersets of CSS. Bootstrap and Foundation are frameworks;
- and... we still not finished here;

# Frontend Development

- JavaScript: TypeScript is superset of JS. Babel transpiles the next generation JS to plain JS. React, Angular are frameworks;
- Module Bundlers: WebPack, Browserify, RollUp;
- Task runners: Gulp, Grunt;
- and... we've finished now :)

If you like to build things that other people can interact with, then your choice is frontend development

# iOS Development

- Are you ready to buy iPhone and iMac\MacBook?
- If not, are you ready to configure hackintosh?
- IDE: Xcode
- Languages: Objective-C and Swift;
- Frameworks: Cocoa Touch;

# Android Development

- Are you ready to buy Android device?
- IDE: Android Studio;
- Languages: Java and Kotlin;
- Frameworks: not sure if you need this, Android SDK is rich enough;

If you want to write  
applications for mobile,  
choose your ~~side~~ platform

# Embedded Programming

- Embedded programming is programming for microcontrollers;
- There is no common practice to start;
- You need to know a lot;
- Languages: Assembler and C;
- Microcontrollers: Tessel, Espruino, Atmega;

If you want to create software that interacts with hardware, then your choice is Embedded Programming



# Thanks



ghaiklor



ghaiklor



ghaiklor