Software Development

Foundations
Lesson 1

Agenda

01.   Introduction and icebreaker

02.   Course structure and LMS

03.   Software development

04.   Environment setup

05.   Markup languages

06.   HTML

Introduction & Ice breaker

Breakout rooms in groups of 3

5 minutes to find out what you have in common, the more interesting the better!

Choose 1 person from each group to share with the whole class

Some ideas of things to ask about:

- Pets
- Favourite food
- Hobbies
- Where you've travelled
- Favourite movies / music

Course structure

01. Foundations (3 weeks)

02. JavaScript (4 weeks)

03. Back-end development (6 weeks)

04. Full-stack integration (4 weeks)

05. Algorithm and structural foundations (4 weeks)

06. Front-end development (4 weeks)

07. Deployment (3 weeks)

08. Mobile development (4 weeks)

09. Capstone project (4 weeks)

LMS introduction

https://courses.circuitstream.com/d2l/home

- 5 assignments - **20%**
- Midterm - **30%**
- Capstone - **50%**

Need to get more than 70% overall to get a certificate

Expectations

- Show up to class
- Pay attention during explanations and follow along afterwards
- Complete homework and assignments
- Practice, practice, practice
- Make mistakes
- Overcome imposter syndrome
- Ask questions (during class or in Discord channel)
- Provide feedback (it's a gift!)

The bootcamp will be hard work, the most powerful predictor of success is a great attitude and a willingness to learn

Software development
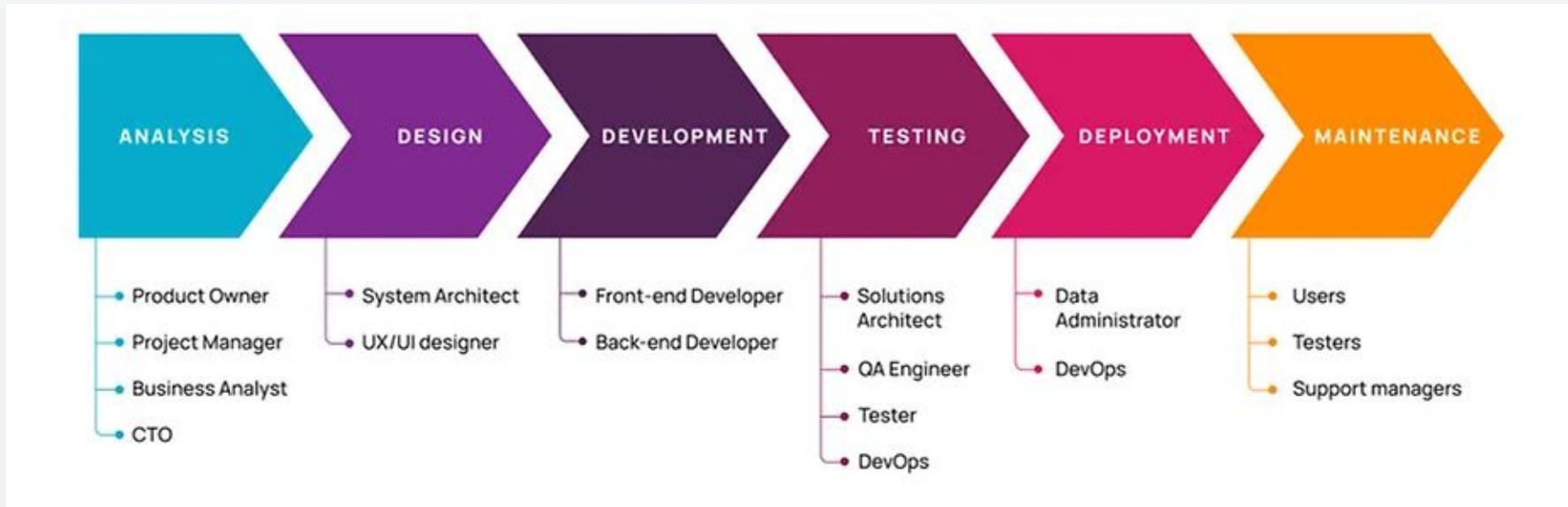
1. What is software?
2. What is Software Development?
3. What does a developer do?
4. What skills are needed?

Break

Agenda

01.   ~~Introduction and icebreaker~~

02.   ~~Course structure and LMS~~

03.   Software development

04.   Environment setup

05.   Markup languages

06.   HTML

Software development lifecycle

ANALYSIS | DESIGN | DEVELOPMENT | TESTING | DEPLOYMENT | MAINTENANCE

- Product Owner
- Project Manager
- Business Analyst
- CTO

- System Architect
- UX/UI designer

- Front-end Developer
- Back-end Developer

- Solutions Architect
- QA Engineer
- Tester
- DevOps

- Data Administrator
- DevOps

- Users
- Testers
- Support managers

Learning objectives

1. **Use Markdown to create a simple document**

2. **Setup a development environment**

3. **Recognize HTML and how it works**

Environment setup

- Install Visual Studio Code (or another text editor / IDE)
- Install a web browser (Chrome)
- Install Git
- Create a GitHub account
- Install GitHub Desktop

NOTE: If you are stuck somewhere in installation, please feel free to use premium support from LMS.

Markup languages

- Specifies the structure and formatting of a document
- Used to control the display of a document or enrich its content
- Examples
  - XML
  - HTML
  - LaTeX
  - Markdown

Markdown exercise (5-10 mins)

Basic syntax: https://www.markdownguide.org/basic-syntax/

Create a Markdown document with:

- One h1
- Two h2
- Two h3
- Emphasised text
- A table
- A link

NOTE: Live preview can be seen by Ctrl+Shift+v or Ctrl+K V for side-by-side view or little 🔲 icon on top-right corner.

HTML

- HyperText Markup Language
- Invented in 1989 as a language for simple structured documents
- What is hypertext?
- What are web standards?
- https://www.w3.org/standards/
- https://developer.mozilla.org/en-US/

HTML exercise (10 mins)

Install "HTML Preview" extension for VS Code if required.

Convert your Markdown document to HTML

HTML tags

- **<h1>**
- **<h2>**
- **<h3>**
- **<a>**
- **<p>**
- **<strong>**
- **<em>**
- **<img>**
- **
**
- **<table>**

For next class

Review the HTML elements reference page on MDN:

1. What do you think looks odd?
2. What makes perfect sense?
3. What looks useful?