Web Development

This document contains some information extracted from the Curriculum for Web Development developers and highlights general characteristics, objectives and tools/projects adopted or suggested for every module.

Modules

Additional Modules

L. Language Course

M. Mentoring

S. Soft Skills

1. Basic Digital Literacy (2 weeks)

The Basic Digital Literacy module lays the foundation for the learners to be able to use common development tools and identify the technological basics of the internet, as well as introduces them to common workflows developers utilise in their day-to-day jobs.

Learning objectives

After this module the learner will…

- ➔ use the linux bash terminal to create, view and manipulate files and folders
- ➔ be able to use tools such as git and GitHub to work on and maintain projects
- ➔ author documents using a basic markup language such as markdown
- ➔ understand the concept of programming and programming languages

Main topics

ID	topic	content	exemplary tasks
1.1	Course Intro	Information about course | technical setup of the laptops	Set up the computer | getting to know the class and the staff
1.2	Web Dev	Linux basic | desktop environment | web development tools and languages	Introduction to basic concepts of development
1.3	Viewing & Navigating	Introduction to the linux terminal | relative and absolute paths | documentation resources	Use the terminal to move around the filesystem and view folders and files
1.4	Creating & Manipulating	Simple bash commands | file and folder manipulation | markdown authoring	Create a document via shell and edit the content of a markdown document
1.5	Installing	Package management in Linux | Node package managers and runners	Installing and updating packages with apt, npm and run scripts with npx
1.6	Versioning	Version Control Systems (VCS) | common basic workflows with git | working with remote repositories	Create first repository and commit markdown documents
1.7	Publishing	Internet protocols | WWW | GitHub | advanced markdown	Publish a repository on GitHub
1.8	Collaborating	Sharing code with others | resolving conflicts | asking for reviews	Change and share documents with peers and make a pull request

2. UI Basics (9 weeks)

The UI basics module provides the participants with a deep understanding of how to create user interfaces for websites and applications as well as employ various development tools to support their efforts.

Learning objectives

After this module the learner will …

- ➔ Plan and create user interfaces for various screen sizes and use cases
- ➔ Design simple wireframes and implement them using HTML and CSS
- ➔ Adopt common development workflows using different build tools and layout models
- ➔ Create a landing page using Bootstrap Framework and publish it on GitHub Pages

Main topics

ID	topic	content	exemplary tasks
2.1	Boilerplate	Basic concept of HTML | document structure | code editors | doctype	Create a basic structure for a webpage
2.2	Content	HTML primary elements |images | hyperlinks | basic styling using CSS | CSS classes and pseudo-classes | debugging	Extend basic web page structures, use graphical elements and style with CSS
2.3	Box-Model	CSS design principles | block level semantic | modelling boxes | positioning	Design and implement a webpage using semantic elements
2.4	UI/UX	user interface design | styling texts | icons | using colors	Develop a webpage based on a user interface design
2.5	Data	organizing data in tables | styling tables | form and dynamic data | inputs	Represent data in a web page and create a registration form
2.6	Responsivity	Mobile first | media queries | responsive design principles	Develop a responsive and mobile first web page
2.7	Layout	Layout models, flexbox and CSS grid, design tools, use cases	Design a layout in Figma and implement it
2.8	Interactions	Shapes | transition | animations with CSS	Create CSS animation effects
2.9	Framework	Bootstrap | SASS | NPM | GitHub Pages	Create a landing page and publish it to GitHub pages
2.11	Publishing	GitHub Pages using package	Publishing through the terminal with git
2.10	Project week	Frontend web development	Create your web portfolio

3. Programming Basics (9 weeks)

The Programming Basics module is designed to train the participants to approach various logical problems and develop problem solving skills, as well as teach them the basics of programming and debugging from the ground up.

Learning objectives

After this module the learner will …

- ➔ Identify common logical problem patterns and solve them by writing algorithms
- ➔ Write programs using JavaScript
- ➔ Understand programming concepts and design patterns
- ➔ Identify and debug compilation, runtime and logical errors in JavaScript

Main topics

ID	topic	content	exemplary tasks
3.1	Language	Intro to ECMAscript | variables | data types | statements | expressions |standard built in objects |Math	Create a simple program to manipulate strings and add numbers
3.2	Program	Flow control | decisions | block scope | multiple choice | numerical repetition	Implement the FizzBuzz algorithm with JavaScript
3.3	Function I	Routines and subroutines | functions | functional scope | return statements | parameters	Write a program to calculate the average of a list of numbers
3.4	Data Structure	Objects | object scope | destructuring | advanced array methods | logical thinking	Write a program to analyze and filter a set of data
3.5	Function II	Callbacks | higher order functions | closures | recursive functions	Write a program that executes a sequence of tasks
3.6	Classes	Classes | prototypes | instances | inheritance | subclasses	Create a program that accepts user data and returns info
3.7	Project week	Individual project week and presentation	Create a calculator using object oriented JavaScript

4. Single Page Application (9 weeks)

The Single Page Application module is where participants gain first-hand experience with common architecture patterns facilitated by JavaScript front end frameworks. From starting up their own apps to integrating a state store in their client side applications, participants gain a perspective on modern and practical implementations in web development.

Learning objectives

After this module the learner will …

- ➔ Work with the DOM API to manipulate HTML documents using JavaScript
- ➔ Understand the basics of working with asynchronous code
- ➔ Develop and deploy single page applications using React
- ➔ Understand and employ reusable components to create web applications

Main Topics

ID	topic	content	exemplary tasks
4.1	DOM	HTML DOM | JavaScript in the browser | querying and manipulating| events	Create a dynamic web page with HTML and JavaScript
4.2	Modules	JavaScript modules | import and exporting | bundling | NPM workflow	Use external modules in a web application
4.3	Asynchronous programming	Promises | Fetch API | CORS | APIs | JSON | LocalStorage	Create a forecast application using data from an API
4.4	Boilerplate	Intro to React | Bootstrapping application | rendering | dev tools	Bootstrap a React app
4.5	Components	React components | JSX templating | Data flow | lifecycles | styling	Create a markdown editor using React
4.6	Router	Routing in a SPA | browser history API | matching | navigation | route parameters	Create a dynamic routing application
4.7	Store	State management concepts | Context API | advanced implementation	Create an authentication app
4.8	Deployment	Serverless infrastructure comparison	Deploy React apps
4.9	Workshop	In-depth hands on workshop	Create an app with technologies learned
4.10	Project	Individual projects week and presentation	Develop a fullstack app using MERN stack

5. Backend (9 weeks)

The Backend module focuses on common database technologies and Node frameworks to create a transactional RESTful API service for storing and serving JSON data. It introduces the participants to a common implementation architecture for web applications which relies solely on a remote server for all aspects of the model view controller pattern (MVC).

Learning objectives

After this module the learner will …

- ➔ Design, create and maintain backend application using Node.js and Express
- ➔ Understand and use databases to store, manipulate and retrieve data
- ➔ Understand the role of RESTful API servers in the current web development landscape
- ➔ Deploy server and databases to third party service providers

Main topics

ID	topic	content	exemplary tasks
5.1	Server	Intro to server side| MVC | Node built-in modules | Express | routing | handling errors | deployment	Create a server app using Node and Express
5.2	Database basics	Intro to databases | key concept of data structures | popular types of databases: NoSQL vs. SQL | CRUD	Create a database and perform common operations on it
5.3	Database advanced	ODM and ORM | Mongoose | Model | schema design | relations	Extend the database and connect it with the server
5.4	Security	Making data consistent and secure | Validation | Encryption | Authentication | JSON Web Token	Add authentication and improve security of the backend
5.5	Workshop	In-depth multi-track workshop	Create an app with technologies learned in the workshop
5.6	Project	Individual projects week and presentation	Develop a music app using iTunes or Spotify API

6. Final Project (7 weeks)

The final project demonstrates the learners ability to apply and assimilate the various skills acquired throughout the course. Furthermore the student will learn to structure larger applications and experience all phases of a software project. The main focus of this module is to work in a team and collaboratively create and improve a software application.

Learning objectives

After this module the learner will …

- ➔ Have the self-esteem from planning creating and presenting a full software project
- ➔ Be able to work in teams and split work into smaller packages
- ➔ Have created and structured a code base larger than all previous examples

Main topics

ID	topic	content
6.1	Project Management	Form a team | brainstorm about project purpose | identify potential pitfalls
6.2	Agile methods	Daily standups | split work into stories | sprint planning
6.3	project design	Mockups | wireframes | design | accessibility
6.4	project implementation	Working on project | code review | pair programming | create project presentation | documentation of project | deployment of project
6.5	project presentation	Slidedeck | project demonstration | presentation skills
6.6	Evaluation	Detailed feedback by teachers and senior developers
6.7	Final Event	Feedback Session | Celebrating Success | Handing out DCI Certificate