

Web Programming

1. **Course number and name:** 020PWBES3 Web Programming
2. **Credits and contact hours:** 4 ECTS credits, 2x1:15 contact hours
3. **Instructor's or course coordinator's name:** Youssef El Bakouny
4. **Text book:**

Full-Stack JavaScript Development: Develop, Test and Deploy with MongoDB, Express, Angular and Node on AWS. Eric Bush, Maura van der Linden. Red Sky 2016.

 - a. **Other supplemental materials:**

Full Stack Web Development Specialization. Jogesh K. Muppala, David Rossiter. Massive Online Open Courses (MOOCs) on Coursera.
5. **Specific course information**
 - a. **Catalog description:**

This course covers the development of web applications on both the front-end (client-side) and the back-end (server-side). It is, in fact, a hands-on web programming course where a MongoDB, Express, Angular and Node (MEAN) web application is gradually designed and implemented as the course progresses. The course first introduces the basic languages used for web development, namely HTML, CSS and JavaScript. Several interactive web pages are then implemented using these languages. These first implementations essentially demonstrate how time consuming web development can be when all components are implemented from scratch. This naturally leads to the introduction of the Twitter Bootstrap web framework and the quick implementation of several web pages using this framework. Afterwards, the Angular framework along with its underlying Model View Controller (MVC) design pattern is explained. An Angular Single Page Application (SPA) is then implemented. At this stage, the front-end has been fully implemented while the back-end is still mocked using a simulated JSON-Server. In the final part of the course, the mocked back-end is replaced by a fully functional REST API implemented using Node.js, the Express framework and the MongoDB database.
 - b. **Prerequisites:**
 - c. **Required:** Elective for CCE students
6. **Specific goals for the course**
 - a. **Specific outcomes of instruction:**
 - Implement an interactive web page using HTML, CSS and JavaScript.
 - Design and implement a user-friendly web interface using the Twitter Bootstrap framework.
 - Design and implement a modular web application front-end using the Angular framework.

- Construct a JSON document database using MongoDB.
- Design and implement, for the web application's back-end, a REST API using Node.js, the Express framework and MongoDB.
- Write high quality code in accordance with the best practices of Web development.

b. KPI:

KPI	c3	e3	i2	k2	k3
Covered	x	x		x	x
Assessed	x	x	x	x	x

7. Brief list of topics to be covered

- Course introduction. (1 lecture)
- HTML & CSS (2 lectures)
- The JavaScript programming language and the Document Object Model (DOM) (3 lectures)
- The Twitter Bootstrap UI framework: the grid system and responsive web design, the navigation bar, forms and buttons, badges, labels, alerts, progress bars, images, media objects, thumbnails, tables, panels, tabs, wells, pills, accordion, collapse, scrollspy, affix, modals, tooltips, popovers, carousel (5 lectures)
- Web tools: the Less and Sass CSS preprocessors, a quick introduction to Node.js followed by an explanation of the Bower package manager as well as the Grunt and Gulp task runners. The Yeoman web scaffolding tool is also briefly exposed (3 lectures)
- The Angular framework: the Model View Controller (MVC) design pattern, modules, controllers, scope, forms, directives, filters, factory, service, dependency injection, templates, routing, Single Page Applications (SPAs). This series of lectures is concluded with an explanation of Representational State Transfer (REST) APIs along with the RESTful services provided by Angular. The back-end's REST API is then mocked using JSON-server and the fully functional front-end of the web application is then connected, using Angular's RESTful services, to this mocked back-end (7 lectures)
- The Node.js JavaScript runtime: the npm package manager, Node modules and the efficient use of the non-blocking IO model (1 lecture)
- The Express web framework for Node.js: building a server-side REST API, the Express router and the Express Generator (2 lectures)
- Document databases: MongoDB and the Mongoose Object-Document Mapper (ODM) (2 lectures)
- Putting the pieces together to finalize the MEAN web application: implementing a fully functional REST API using Node.js, Express and MongoDB and connecting the fully functional Angular front-end to it (2 lectures)