Web Development 1 (WEB1) - Course Information

# Course description

In this module we will study the basic technologies that allows us to build a web application; you will study front-end as well as back-end technologies. During this course students will build a web application from scratch applying a logical process.

# Course learning goals

The student will be able to:

* Understand the main technologies and tools involved in Web development.
* Provide insight in the design and development process of a web application.
* Get to know the following front-end technologies: HTML, CSS and JS.
* Get to know the following back-end technologies: PHP, MySql

Required knowledge:

* VCS (version control system)
* FIS2 (current IDE)
* EDB1 & EDB2
* PCS1 & PCS2

# Study material

This module uses internet resources. All resources has been placed in the end section of this document (resources section).

# Way of Working

The approach used in this course is *Learning-by-doing*. Therefore, this course is compound of 1 theory hour and 3 practical hours per week, at school. Besides, the student will have to spend a minimum of 8 hours (self-study) in the matter per week. The theory lectures explain what a certain technology is, the purpose of it and the essentials on how to use it. It is the student task to go deeper and study further in detail such technologies so that she is better prepared to tackle the issues that might appear during the practical assignment.

Furthermore, the student is expected to have the required knowledge previously stated to guarantee a smooth flow of the course and avoid falling behind and or into frustration.

The practical of this course is about building a web application which is called *The Project* (see “The Project” section). Since web development is mainly a collaborative process you will work in groups of 2 people. *The Project* has weekly deliverables (see course planning).

# Course Planning for Students

| week | lecture | Practical | reader | Deliverables to Tutor | | DEADLINE |
| --- | --- | --- | --- | --- | --- | --- |
| week 1 | **Introduction WEB1,**  **the process of building a website** | * Install all required tools (see w1 resources) * Create an Athena account to host your website * Work on Design Document | **Chapter 18;** | | Del A:   * Design Document | **Week2** (before theory lecture) |
| week 2 | **HTML** | * Structure your website using HTML | **Chapter1-9;** | | Del B:   * All HTML pages of your website. | **Week3** (before theory lecture) |
| week 3 | **CSS** | * Style your website applying CSS | **Chapter 15-17;** | | Del C:   * One CSS file containing your websites’ style and design. | **Week4** (before theory lecture) |
| week 4 | **JS** | * input validation * use of Ajax | **See w4 resources** | | Del D:   * Input validation * Ajax functionality | **Week5** (before theory lecture) |
| week 5 | **PHP** | * Template System * File Structure | **See w5 resources** | | Del E:   * Create a logical file structure * Show use of templating system | **Week6** (before theory lecture) |
| week 6 | **PHP** | * Authentication and Authorization. | **See w6 resources** | | Del F:   * Implement authentication and authorization | **Week7** (1 day before presentation) |
| week 7 | **Presentation** |  |  | |  |  |

# Grading

The grade of *The Project* is the final grade of WEB1. For each deliverable (A, B, …, F) you will get a grade between 1 and 10. Calculating the final grade for *The Project* occurs as described in the table below and the criteria underneath it.

There will be *Go* and *No-Go* sessions. This is determined by your teacher at any point in time during the course. A *No-Go* means your group cannot proceed any further and must leave the course receiving no final grade as consequence. If your group misses a deadline you get 0 points for that deliverable and your final grade will be decreased by 1 point. If your group misses more than one deadline you automatically get a *No-Go*.

|  |  |  |
| --- | --- | --- |
| Description | Related Deliverable(s) | Weight on final grade |
| Design Document | A | 20% |
| Front End development | AVG (B,C,D) | 50% |
| Back End Development | AVG (E,F) | 30% |

## Front-end criteria:

* HTML Tags properly used.
* One CSS file for the whole website
* Design harmony.
* In accordance with Design Document.
* Neat division of code (html/css/js) using comments.
* Use of AJAX.
* Website works properly.

## Back-end criteria:

* File structure
* Template system
* Authentication and Authorization

# THE PROJECT (building a web application)

You will work in groups of 2 people. Each group will gradually build a web application submitting deliverables weekly.

## Week 1: Deliverable A

In order to build a website you must set up your development environment first. Install the specific tools described on the table below. Since you are building the web application step by step some of the tools will be needed in later weeks but this is an appropriate moment to install it as you won’t have much time later on.

Under the Installation Guides directory on Sharepoint you find step by step guides on how to install and set up your environment. There are also guides with possible solutions to problems or issues you might encounter.

|  |  |  |
| --- | --- | --- |
| Tools | Purpose | Source |
| Athena server account (create) | Fontys’ Webhost for its students | <https://athena.fhict.nl/registration/> |
| Brackets | Text editor | <http://brackets.io/> |
| VCS | Version control | <https://git-scm.com/> |
| Bitbucket | Hosting, managing your code | <https://bitbucket.org/> |
| FileZilla | Transferring your project to Athena server | <https://filezilla-project.org/download.php?type=client> |
| NetBeans | IDE to develop the project | <https://netbeans.org/> |
| XAMPP | Local PHP development environment | <https://www.apachefriends.org/index.html> |

After installation of these tools you are ready to focus entirely on the process and design document of your web application. This document is usually the contract (what you promise) with the client (your teacher). But first brainstorm with your group mate about the topic your web application is going to be about.

The Design Document is entirely based on chapter 18 of the book *HTML & CSS design and build websites*, John Ducket. After reading and understanding the concepts on this chapter create a document and name it following this pattern: *Design\_lastNameMember1\_lastNameMember2.* Remember this is a web application therefore you should consider in the design document things such as registration, logging in, posting, etc.

Your design document must consist of the following sections:

1. Web application topic
2. Target Audience
3. Sitemap
4. Wireframes
5. Visual Design

Sent the Design document to your teacher by e-mail.

## Week 2: Deliverable B

Create the layout and HTML structure according to your Design Document. You must provide all .html pages of your web application.

All the pages have to be uploaded to Athena server, use FileZilla for this purpose. FileZilla needs certain account data to connect to Athena server, find these details on your Athena account (under FTP settings) you created previous week.

After successful connection upload all your .html pages.

## Week 3: Deliverable C

Create the style of your website according to your Design Document. The CSS file must be external and put into a separate directory on Athena server.

## Week 4: Deliverable D

Add AJAX functionality to your website.

Use JavaScript to enhance user interaction (experience).

Validate your input in your form (front end validation); if you don’t have a form create one and update your design document.

Update your code and upload to Athena server.

## Week5: Deliverable E

From this week on we are also dealing with back-end development. You need, therefore, to setup a PHP local development environment so that you keep your production version of your web application untouched on Athena. All features and functionality should be implemented and test first locally using XAMPP and only then you can push your files to Athena.

Setting up your PHP development Environment:

You have installed all necessary tools in week one. Now do the following:

1. Set-up XAMPP & Netbeans to make debugging of your PHP scripts possible (for specific steps refer to Installation Guides directory on Sharepoint)
2. Make a simple PHP sample file in which you can demonstrate that your environment is set-up correctly and you can place a breakpoint for debugging purpose.

As you can see your project is escalating. In order to guarantee maintenance is crucial to have your files well organized. Create a logical file structure for your project.

Templating is one great advantage you can take of PHP. Create one PHP file (index.php) with the HTML structure/site layout then load the correct content based on the query string in the URL. For example index.php?page=home should show the content of the homepage.

## Week 6: Deliverable F

This week is about authentication and authorization. Implement registration functionality on the web application as well as Log in and log out functionality.

Make at least one page not accessible when the visitor is not logged in.

# RESOURCES:

## Week 1: PROCESS & DESIGN

|  |  |
| --- | --- |
| **Topic** | **Source** |
| Process & Design of a website | Chapter 18; *HTML & CSS design and build websites*, John Ducket |
| Installation of tools, setting up environment | [Click here](https://portal.fhict.nl/es/WEB1/Forms/AllItems.aspx?RootFolder=%2Fes%2FWEB1%2FInstallation%20Guides&FolderCTID=0x012000515219E6DDEEF4479D2C543BEA1F0BCD&View=%7B3F79EEF8-D29D-4D6E-859F-B7F06682D1C9%7D&InitialTabId=Ribbon%2EDocument&VisibilityContext=WSSTabPersistence) |

## Week 2: HTML

|  |  |
| --- | --- |
| **Topic** | **Source** |
| HTML tutorials, examples, etc. | <http://www.w3schools.com/html/default.asp> |
| World Wide Web Consortium (HTML standard) | <https://www.w3.org/standards/webdesign/htmlcss> |

## Week 3: CSS

|  |  |
| --- | --- |
| **Topic** | **Source** |
| CSS tutorials, examples, etc. | <http://www.w3schools.com/css/default.asp> |
| World Wide Web Consortium (CSS standard) | <https://www.w3.org/standards/webdesign/htmlcss> |

## Week 4: Js

|  |  |
| --- | --- |
| **Topic** | **Source** |
| JS theory | <https://developer.mozilla.org/en-US/docs/Web/JavaScript> |
| JS tutorials, examples, etc | <http://www.w3schools.com/js/> |

## Week 5: PHP

|  |  |
| --- | --- |
| **Topic** | **Source** |
| PHP official site | [http://php.net](http://php.net/) |
| PHP Arrays | <http://w3schools.com/php/php_ref_array.asp> |
| PHP Superglobal variables | <http://www.w3schools.com/php/php_superglobals.asp> |

## Week6: PHP

|  |  |
| --- | --- |
| **Topic** | **Source** |
| PHP Cookies | <http://php.net/manual/en/features.cookies.php> |
| PHP Session | <http://php.net/manual/en/book.session.php> |
| PHP PDO class | <http://php.net/manual/en/class.pdo.php> |
| PHP PDOStatement class | <http://php.net/manual/en/class.pdostatement.php> |
| PHP PDOException class | <http://php.net/manual/en/class.pdoexception.php> |
| PHP File uploading | <http://php.net/manual/en/features.file-upload.php> |
| PHP File system | <http://php.net/manual/en/book.filesystem.php> |