

# Task 3: Website development

## Description

This part of the coursework requires you to develop the website using the design created in the previous deliverable.

You should choose one of the designs you developed, explaining the reasons for this choice based on your designs.

You should then develop the design and the functionality of the website using HTML, CSS, and optionally JavaScript.

We suggest the use of the Jekyll, that is a static site generator. It has the advantage of being lightweight, not needing a database, being included in GitHub Pages, and offering nevertheless a quite advanced template capability and SASS/SCSS support.

You are free to use more complex technologies you know of, both client-side (Angular, React, ...) server-side (nodejs, Python, PHP, Java, ...).

Please remember that the focus of this work is on website design, usability and interactivity. Although some server-side processing could be helpful, do not spend too much time developing sophisticated back end programs such as database systems. We are interested in how well your implementation fits your design.

*NOTE: You should avoid copyright materials, if you are using photos either take your own or use something copyright free. DO NOT just grab them from Google images.*

## Deliverable

The final deliverable is submitted electronically and is composed of 3 items: the publicly accessible website, its source code, and a report documenting the implementation.

### Part 1: Website

The site must:

- be accessible on the web (e.g., as GitHub Pages under github.io);
- include a minimum of 5 pages;
- be encoded in valid HTML + CSS (+ JavaScript, if needed);
- use appropriate formats for included media (images, videos, fonts, ...);
- reflect as much as possible the chosen design from coursework 2.

The site may:

- replace some complex functionality described in coursework 2 with a static (but still interactive) version: e.g., if a page should contain a working chat, the page may have the appearance and basic interactivity of the chat without a server side component, by showing just the user messages and fixed (or echo, or random) replies from the server;
- use placeholders for text (lorem ipsum...) and images (<https://placeholder.com/>, <https://placeimg.com/>, ...) inside internal pages (but don't use placeholders for the content of main pages or navigational items).

## Part 2: Website Source

The source must:

- be self-documenting, by expressive and coherent use of labels (ids, class names, variable names) and by using comments in the code;
- contain (in a README file in the main directory) the detailed description on how to deploy it;
- contain all the resources and the data needed by the website.

## Part 3: Report

This part requires you to prepare a 2000-word report that must:

- detail the technical aspects of the development
- discuss the process from design to development (i.e., any changes made) and delivery in detail
- include links to the published website and the source repository

Submit your **report** via QMPlus by midnight **Friday 8th December 2017**.

**Submission Instructions:** **coming soon...**

## Marking Scheme

This part of the coursework is marked in terms of the following percentages of the marks allocated for the prototyping part for each interface.

<i>Fit to design and requirements</i> How does your website fit to the requirements and the design identified in Deliverable 2?	Up to 20
<i>Design and Usability</i> How does your website look and perform, does it appeal to the target market, is it easy to navigate and understand, does it load quickly, does it use images appropriately and make use of title attributes and alt text etc.? How do you inform the users about how you process the forms and input provided by them?	Up to 30
<i>Well-structured and validated HTML and CSS</i> Is your code well-structured and easy to read, indented appropriately, have you added relevant meta tags, title tags etc. and does the HTML and the CSS validate to w3c specifications? Do you need to create and use separate CSS files for different types of users? Why? Why not? Have you used code from other sources? Is this really needed? Acknowledge which part of the code comes from which source and justify the reasons.	Up to 30
<i>Organisation and site structure</i> Have you organised and named all your files in an appropriate manner, made use of the title tag and have you commented your code appropriately so it is easy to see what is happening in your code for other developers?	Up to 20