

Advanced Web Technologies SET09103

Coursework Specification #2

1 Task

The objective is to demonstrate your understanding of server-side web development and mastery of the Python Flask micro-framework. You will achieve this by completing a personal project in which you design, implement, and evaluate a web application on a topic of your choice.

This is a personal project so the web-app that you will build for this second coursework is completely up to you. However it is advisable to discuss your idea with the module leader in advance to ensure that it has sufficient scope to enable you to achieve a decent grade. Some ideas for suitable projects include a simple Twitter clone or a Bookmarking/URL shortening web-app. You should carefully consider the nature of the problem domain, and design a URL hierarchy and user interface that provides your users with a good experience. You should also research similar web-apps that you can use as a benchmark against which to measure the functionality of your own.

The coursework should be fun, so use your imagination, and give your creativity a free rein. Invention and originality will be rewarded by the marking scheme. I hope you enjoy working on it.

2 Submission & Deliverables

Your coursework deliverables comprise the following:

1. Source code.
2. Report.

The method of submission for both deliverables is via a single Git repository. Both your report and source code must be committed to a Git repository in "sourcecode" and "report" folders respectively.

- All sourcecode and your report must be placed in a Git repository.
- Your Git repository must be named lastname_firstname_set09103_cw2.
- Your repository must be pushed to a hosting service, e.g. Bitbucket, Github, or the school's internal Git server.
- Email the Git clone URL for your repository to s.wells@napier.ac.uk at least one week before the assignment deadline. This should be the SSH clone URL (the one that starts with either git@github or git@bitbucket).
- If your repository is private then you must add the user siwells as a collaborator so that your work can be retrieved.
- Your Git repository must contain only the files for your web-app, e.g. the sourcecode and report. It is your responsibility to ensure that you have placed all of the source code necessary to run your web-app in your repository. Note that this does not mean you should include either Python or Flask or any of the other tools that are already installed on the module's development server.

If you are in any doubt about any of the requirements for the coursework or any aspect of the submission procedure then please contact the module coordinator for further guidance.

2.1 Sourcecode

You may use additional Python modules, Flask plugins, Javascript Libraries, or miscellaneous software in the production of your project. However you must be able to justify why you used additional software rather than writing a solution yourself. Clear documentation must be supplied in an appendix of your report for anything you use that is not within our default learning environment. This documentation must clearly explain the process of installing and using your choice of additional software.

2.2 Report

Your report must be no longer than 6 pages in length (excluding appendices) and written using the Napier report template:

http://github.com/edinburgh-napier/aux_latex_cw_template

It is recommended that you use \LaTeX to typeset your report, but so long as your report is according to the template format this will not be enforced. The format of the submitted report must be PDF and must include the following sections:

Title of your web-app.

Introduction Describing your web-app and its core features. This section should include at least one screenshot of your web-apps main page.

Design Explaining how and why your web-app is structured the way it is. This section should include a navigation map for the URL hierarchy of your web-app. Where appropriate to the project, e.g. when users can supply data to the web-app, you should address and report upon any issues regarding security and privacy that will arise as a result

Enhancements Describing the features that you would add or improve.

Critical Evaluation This is about *what* you built and should explain the features of the web-app that you feel work well, or work poorly, and why.

Personal Evaluation This is about how *you* performed. You should reflect upon what you learned, the challenges you faced, the methods you used to overcome challenges, and how you feel you performed.

References If you have used additional resources then these should be cited. Please acknowledge all sources of help and material. If there is anything included that was not written by you for this module, then you should indicate that within this section.

Appendices (Optional) Appendices may be used to include supplemental data, for example test data, screenshots, or documentation, but these must be referenced from within the main body of your report.

2.3 Demonstration

It is school policy that all coursework must be demonstrated and that without a demonstration your submission will not be marked. Demonstrations will be held during regular lab sessions and all students will have the opportunity to sign-up for a demo slot. The module coordinator will contact the class closer to the deadline to organise demo slots.

2.4 Important Dates

- Clone URL to module coordinator: At least one week before the deadline.
- Submission deadline: 11:45PM on Sunday 25th November.
- Demos: During the regular lab session on Monday 26th November & Monday 3rd December.
- Return of work & feedback: Under ideal circumstances you will be emailed written feedback within three working weeks of the submission deadline. However you will also receive verbal feedback both during your demonstration and during contact time throughout the trimester. Verbal feedback is as important, sometimes more so, than written feedback, and should neither be discounted nor disregarded.

2.5 Assessment Criteria & Grade Guide

This coursework is worth 60% of your overall grade for this module. The mark breakdown for this coursework is as follows:

- 40-49%** This grade band indicates work that is acceptable, that demonstrates basic skills in the core concepts covered by the module, but where there is plenty of room for improvement. To achieve a mark in this band you must have developed your own working web-app with multiple routes allowing the user some, but not extensive interaction. It may be based directly on an extension of the practical work covered in class and your report must adequately describe your work.
- 50-59%** A grade in this band constitutes good work. A submission in this mark band will indicate that you have developed a web-app that is less ambitious in its functionality but will offer the user suitable ways of interacting. Your report will be well written and will reference the material you have used.
- 60-69%** Work in this grade band is at a very-good level. To achieve a mark in this band you will have developed a web-app with very good functionality, for example, offering the user multiple URLs and some evidence of appropriate utilisation of: routing, requests, redirects, responses, custom error code handling, static files, templates, sessions, logging, testing, data storage, encryption, and data transports. You will have evaluated the data that you store with respect to security and privacy issues. Your report will address all the necessary sections effectively, be very well written, clearly presented, and will reference material you have used.
- 70-100%** This grade band indicates work at an excellent (70+), exceptional (80+), or exemplary (90+) level. A submission in this mark band will consist of an application that has extended the lab work covered in class to offer an excellent level of functionality, both in terms of the number of features and their quality of implementation. To attract a grade at this level a submission must also have been effectively evaluated. Your design and code will be excellent making good use of Flask features and demonstrating exemplary design, a solid application, and excellent URL structure. You will have evaluated the data that you store with respect to security and privacy issues and will have effectively protected any data that can be deemed sensitive, personal or private. Your report with the sections detailed above will be comprehensive, very well written, well presented, and will correctly reference all the material you have used. This is likely to include textbooks, online forums, library & language documentation, tutorials, and some of the suggested reading for the module.

You should think strategically about how to approach this assignment. The grade guide is cumulative, i.e. to get a higher grade, you must also have achieved the functionality required to attain a lower grade. A lower-risk strategy is to identify the core features that you think will attract a pass mark then aim to complete those features as soon as possible. You can then iterate over your *working* solution to improve those features.



1. Module number	SET09103
2. Module title	Advanced Web Technologies
3. Module leader	Dr Simon Wells
4. Tutor with responsibility for this Assessment	Dr Simon Wells
5. Assessment	See Attached coursework description
6. Weighting	60%
7. Size and/or time limits for assessment	See Attached coursework description
8. Deadline of submission	Your attention is drawn to the penalties for late submissions 11:45PM on Sunday 25th November
9. Arrangements for submission	See Attached coursework description
10. Assessment Regulations	All assessments are subject to the University Regulations
11. The requirements for the assessment	See Attached coursework description
12. Special instructions	None
13. Return of work and feedback	You will be emailed written feedback within three working weeks of the submission deadline. However you will also receive verbal feedback during your demonstration.
14. Assessment criteria	See Attached coursework description for details. This assessment covers: LO1: Understand the role of HTTP and related protocols in the design and efficient exploitation of robust and scalable services and APIs for the Internet and Web. LO2: Evaluate the sensitivity of data gathered by your Web app and select appropriate tools and techniques to ensure its security and privacy. LO3: Demonstrate effective use of client side scripting languages and libraries at an advanced level to produce a compelling user experience. LO4: Demonstrate competence at an advanced level in the design, development, and evaluation of web applications and services using server-side languages, libraries, and tools.