

# Project 2016

## Data Representation and Querying

Due: 01/12/2016

The following document contains the instructions for Project 2016 for Data Representation and Querying. The project will be worth 50% of your mark for this module. You must use GitHub to manage the development of your project.

You are required to develop a single-page web application(SPA) [7] written in the programming language Python [1] using the Flask framework [6]. You must devise an idea for a web application, write the software, write documentation explaining how the application works, and write a short user guide for it.

### GitHub

GitHub must be used to manage the development of the software. You might also try using GitHub Issues [2]. GitHub issues allow collaborators to track project progress through bug reports, milestones and labels.

You will have the opportunity during the weekly timetabled computer labs to work on the project, and to ask the lecturer for advice. The lecturer will add issues and comments to your repository to record these interactions. It is likely that these interactions will heavily affect the marking of the project.

### Technology

You can use any combination of the following technologies in your project. However, note that the main part of the application must be written in Python. Any other technologies including libraries and frameworks you wish to use must be agreed to by the lecturer.

**Languages** — Python, HTML, JavaScript, CSS, Typescript, Less, Sass

**Libraries** — Anulgar.js, jQuery, React, Bootstrap, Skeleton, Ember.js.

**Frameworks** — Flask, Pyramid.

**Databases** — Any SQL RDBMS, CouchDB, MongoDB, Neo4j, Redis.

## Submission

Your GitHub repository, along with the Issues tracker and all other GitHub facilities, will form the main submission of the project.

## Quality

A good project submission will demonstrate a well-managed, steady effort to create a useful and easy-to-use web application. The code base will be well thought-out.

## Team-work

Students may work on this project as a team, subject to approval from the lecturer. Note that team submissions must reflect a team effort, and may be marked to a higher standard. Team members will still, however, receive an individual mark which will mainly be determined by the individual's contributions to the team repository. Thus, it is important that team members ensure they make sufficient contributions to the repository in their own name.

## Student Conduct

You should familiarise yourself with GMIT's code of student conduct [3] and the policy on plagiarism [4]. In particular, note two things. First, students are expected to treat other students and staff politely and with courtesy. Second, it is assumed that all work you submit is being presented as your own work, unless referenced otherwise.

## Marking scheme

The following marking scheme will be used to mark the project. Students should note, however, that in certain circumstances the examiner's overall impression of the project may influence marks in each individual component.

---

40%	Programming	Well-written code; logical directory structure; extensive commenting; extensive git commits.
40%	Architecture	Good separation of concerns; clear API.
20%	User Experience	Nice flow through the application; clear user interface.

---

## Expected standard

Please note that this is a level 7 module. You should be aware that the standard required for submissions at level 8 (fourth year) is higher than at level 7 (third year), which in turn is higher than at level 6 (first and second year). Significant effort is made to ensure that the standard is fair and consistent across third level institutes, both nationally and internationally. The standard we set for modules in computing is informed by Quality and Qualifications Ireland's Award Standard for Computing [5]. Below is a particularly relevant selection of the learning outcomes contained in that document.

### Level 8 (Year 4)

*The learner will be able to:*

- *describe the limitations of some current computing theories.*
- *evaluate information through online research.*
- *model and design complex computer-based systems in a way that demonstrates comprehension of the trade-off involved in design choices.*
- *demonstrate mastery of a complex and specialised area of skills and tools;*
- *manage one's own learning and development, including time management and organisational skills.*
- *manage a computer-based project throughout all stages of the life-cycle.*
- *apply quality concepts to products and processes of own work.*

**Level 7 (Year 3)**

*The learner will be able to:*

- *integrate concepts learned across a variety of subject areas.*
- *identify relevant material on a given topic from available information sources.*
- *succinctly present rational and reasoned arguments to a range of audiences.*
- *develop innovative solutions to pragmatic situations.*
- *recognise the suitability of a given solution to a problem.*
- *apply knowledge learned in new situations.*

**Level 6 (Years 1 and 2)**

*The learner will be able to:*

- *describe, recognise and apply best practices in computing.*
- *apply knowledge in a practical setting under supervision.*
- *demonstrate the capacity to learn new knowledge and skills.*
- *use troubleshooting strategies and techniques in correcting a variety of computer hardware and software problems.*
- *implement computer based systems solutions to well-defined problems.*

**References**

- [1] Python Software Foundation. Welcome to python.org.  
<https://www.python.org/>.
- [2] GitHub. Mastering issues.  
<https://guides.github.com/features/issues/>.
- [3] GMIT. Code of student conduct.  
<http://www.gmit.ie/sites/default/files/public/about/docs/code-student-conduct-2016-17.pdf>.

- [4] GMIT. Policy on plagiarism.  
<http://www.gmit.ie/sites/default/files/public/general/docs/10-no2-plagiariam-3.5.pdf>.
- [5] Quality and Qualifications Ireland. Award standards – computing.  
<http://www.qqi.ie/Publications/Computing%20-%20QQI%20Awards%20Standards.pdf>.
- [6] Armin Ronacher. Flask.  
<http://flask.pocoo.org/>.
- [7] StackOverflow. Why do we need a single page application?