COM661 Full Stack Strategies and Development

SEMESTER 1, 2025-2026

Assignment 1

"Individual Full Stack Application Development – Back End"

Over the course of the module, we construct the ***Biz Directory*** - a sample full stack application built around a database containing information on a range of businesses and a collection of user-contributed reviews. This assignment tests your understanding of this material by having you build the **back-end** of your own full stack application, based around a dataset of your choice. You can obtain your dataset from any source, but you should modify its structure (if required) to provide the most appropriate fit for your planned functionality.

Your application can be in any field but must be driven by a back-end **Python Flask** API that interfaces with a **MongoDB** database and must provide opportunities to demonstrate all of the main database operations (create, retrieve, update, delete).

The primary focus should be on **usability** and **functionality**. The application should fulfil a well-defined need and should provide a range of API endpoints that could potentially support different front-end applications.

Learning Outcomes Assessed

This assignment addresses all module learning outcomes.

Marking Criteria

Submissions will be marked according to the following criteria

- Database structure and appropriateness to the application
- Range and complexity of database queries
- Provision of API endpoints as RESTful resources with appropriate use of HTTP verbs and status codes
- Use of individual user accounts and appropriate authentication
- Evidence of automated API testing
- Code structure and presentation

- Provision of appropriate documentation

This assignment is worth **50%** of the total mark for the module

Effort

This assignment has a total notional workload of **20 hours**.

Submission

The deadline for submission is 12pm (Noon) on **Monday 3rd November 2025**. All work should be submitted to the **CW1 Submission** link in the **Assessment** area of Blackboard and should consist of the following elements:

- A Zip file containing all code files for the back-end API.
- A Zip file containing a JSON file for each MongoDB collection in the database. JSON files should be generated by running the **mongoexport** command for each collection.
- A video submission that (i) introduces your application area and dataset, (ii) demonstrates your API in use with **Postman**, and (iii) provides a walkthrough of your code. The maximum length of the video is **5 minutes**. Please note that videos with a duration longer than 5 minutes will be penalised according to the following scheme.

5 minutes + 10%	no penalty
5 minutes + 10% - 19%	reduction in total mark by 5%
5 minutes + >20% - 29%	reduction in total mark by 10%
5 minutes + >30% - 39%	reduction in total mark by 15%
5 minutes + >40% - 49%	reduction in total mark by 20%
5 minutes + >=50%	maximum total mark achievable is 40%

- Separate PDF files containing
    - a complete code listing of the back-end API
    - a summary of all API endpoints
    - a printout of the result of running an automated collection of tests on the API through Postman
    - a printout of the API documentation generated through Postman
- A completed self-evaluation sheet on which you provide a critical appraisal of your submission against a range of marking criteria.

A video demonstrating the preparation of a submission pack for the sample ***Biz Directory API*** application will be made available on Blackboard.

Please note: The **source code files** and **video** are **compulsory submission elements**. Without either of these the submission will be deemed incomplete and cannot be marked.

Marking Scheme

You should be able to judge the level of your submission by reference to the following criteria. Note that the ***Biz Directory*** example referred to here, relates to the state of the application at the end of **Practical BE08.** Please note that a detailed rubric and marking guide is provided at the end of this document.

Classification	Criteria
1st Class (70%+)	The application will be fit-for-purpose and usable for the intended task. Functionality will be significantly in excess of that demonstrated by the ***Biz Directory API*** example with database operations beyond the standard CRUD functionality. The application will be robust with appropriate validation and authentication and will be free from bugs.
2:1 (60-69%)	The application will have a level of functionality above that of the ***Biz Directory API*** example but there may be minor functionality deficiencies that prevent it from being truly fit for the intended purpose. The application should be free from major bugs that prevent elements of functionality from running properly.
2:2 (50-59%)	The application will be similar to the ***Biz Directory API*** example in terms of working functionality and data structure but will suffer from bugs or performance issues that prevent it from being a usable application. There will be at least some evidence that the developer has attempted to extend the basic functionality.
3rd (40-49%)	The application will be heavily influenced by the ***Biz Directory API*** example, with extended re-use of code without modification or addition. There will have been no visible attempt to enhance the functionality and there may be significant bugs that prevent it from working properly.

Fail (<40%)	The application will suffer from major bugs that prevent it from being properly examined. There will have been no attempt to enhance functionality from the ***Biz Directory API*** example

Feedback

An **optional interim submission** during Week 4 will consist of a short (one page max) summary of your data set, application domain and implementation plan. This submission is for **feedback only** (i.e., no marks are awarded for this) and feedback on this will be provided through Blackboard within one week.

Individual feedback on each final submission will be provided in written form under the categories presented in the marking criteria above.

Feedback on final submissions will be returned within 20 working days after submission in accordance with Faculty policy.

Plagiarism Statement

By making a submission, you will be deemed to have made the following declaration of ownership. Source: http://www.ulster.ac.uk/academicservices/student/plagiarism.pdf.

"I declare that this is all my own work. Any material I have referred to has been accurately referenced and any contribution of Artificial Intelligence technology has been fully acknowledged. I understand the importance of academic integrity and have read and understood the University's General Regulation: Student Academic Integrity and the Academic Misconduct Procedure. I understand that I must not upload my work before, during or after submission to any unapproved plagiarism detectors or answer sharing platforms, or equivalent, and that only University-approved platforms should be used."

Coursework 1 – ASSESSMENT RUBRIC/MARKING PROFORMA

Criteria (100%)	0-29% fail	30-39% fail	40-49% 3rd	50-59% 2.2	60-69% 2.1	70-79% 1st	80-100% High 1st
Choice of Dataset 15%	None or data from the module material used.	Insufficient data to demonstrate full functionality.	Basic dataset which largely follows the structure of that provided in the demonstration application	Usable dataset but missing some desirable feature (e.g., sub-document collections, range of different data types).	All desired features present, but dataset is limited in size or not fully populated	Comprehensive dataset with a large collection, good range of types, and populated sub-documents.	Comprehensive dataset as for 70-79% but additional merit (e.g., from an original or unusual source, containing non text/numeric data, multiple sub-documents, etc.)
Database Functionality 25%	None or minimal application database activity implemented.	Basic CRUD operations incomplete.	All CRUD operations implemented on top-level collections.	All CRUD operations implemented on top-level and sub-document collections.	Use of searching and matching within CRUD operations.	Evidence of complex queries such as multi-valued search, aggregation pipeline, geo-location queries, etc.	Comprehensive set of queries demonstrated, exceeding the complexity demonstrated in the module material. Evidence of additional research.
API Structure 25%	None or minimal attempt at provision of endpoints.	Less than the basic CRUD operations supported.	Basic CRUD operations supported.	Support for full set of CRUD operations on documents and sub-documents.	Some additional endpoints to support increased functionality (e.g., pagination, filtering, etc.)	Comprehensive back-end provision at a level greater than that provided by the demonstration application.	An API with a level of functionality well in advance of the demonstration application and which is immediately usable for the stated purpose.
Usability 25%	No or minimal functionality	Some functionality, but not all CRUD operations supported and/or essential error trapping not present	Basic functionality on a par with the demonstration application	Some additional functionality over the demonstration application	All user input checked and appropriate HTTP status codes returned	Comprehensive range of functionality that could support multiple front-end applications	Validation includes type and range checking, Provision of informative exception messages.
Submission Package 10%	No supporting documents provided.	Multiple supporting documents missing.	One supporting document missing.	All elements submitted, but multiple deficiencies (e.g., coverage of POSTMAN testing and documentation, lack of detail in the video).	All elements submitted but one exhibits some deficiency.	All elements submitted to a good standard.	Evidence of exceptional attention to detail in testing, documentation and/or video presentation.