

FINAL-YEAR PROJECT: SOFTWARE SUBMISSION REQUIREMENTS

All students developing software applications as part of their final-year projects must submit a complete, runnable copy of their source code and all supporting files required to execute and test the application.

1. Hosting Requirements

Students are not required to purchase web hosting, a domain name or a cloud server. All applications must be capable of running locally on a compatible computer, without any hosting services.

2. Submission Format

Students must compress their complete project code and supporting files into a single ZIP file, upload the file to their student OneDrive account, and include the sharing link in the appendix of their final-year project report.

The ZIP file must contain:

- Complete source code and application files.
- Database files or database setup scripts, where applicable.
- Dependency files and other required resources.
- A README file containing installation and execution instructions.
- Sample data and test login credentials, where applicable.

Students must ensure that the OneDrive sharing link is accessible to the examiner and that the file can be downloaded without unnecessary access restrictions.

3. Supported Technologies

Students may use approved programming languages, frameworks and database technologies, including:

- HTML, CSS and JavaScript.
- JavaScript with Node.js.
- Python with Flask or Django.
- PHP and MySQL/MariaDB.
- SQLite or other approved database management systems.

Students may use local development environments, such as XAMPP, where appropriate to their chosen technologies.

Regardless of the technology selected, students must provide all necessary files and instructions for running their applications locally.

4. Database Requirements

All database components required to run the application must be included in the submission.

Where applicable, students must provide:

- The database file or an SQL export.
- Database creation and initialisation scripts.
- Instructions for configuring the database connection.
- Sample records required for testing the application's functionality.

Students must ensure that the database can be accessed and used on a compatible computer other than their own development computer.

5. Execution Instructions

Every submission must include a README file explaining how to install, configure and run the application.

The README must specify:

- Required software, frameworks and versions, where applicable.
- Installation and dependency requirements.
- Database setup and configuration procedures.
- Commands or steps for starting the application.
- The local address used to access the application, where applicable.
- Test login credentials for the relevant user roles, where applicable.

Instructions must be sufficiently clear for an examiner to run the application without relying on undocumented configurations or assistance from the student.

6. Functionality Requirements

The submitted application must be capable of running on a compatible computer other than the student's development computer by following the instructions provided.

Students must ensure that all required source files, dependencies, database components and supporting resources are included.

Applications must not depend on files, database credentials, absolute file paths or undocumented settings that are available only on the student's computer.

Any external services or internet connectivity required by the application must be clearly documented.

7. Testing and Assessment

The department reserves the right to download, extract, install, run and test the submitted application to verify its functionality.

Students are required to demonstrate their applications during the final viva and explain their source code, system architecture, database design, implementation and major functionalities.

A project that cannot be executed because of missing source files, dependencies, database components or inadequate setup instructions may be considered incomplete, subject to the department's assessment regulations.

Students are responsible for ensuring that their submitted code and supporting files are complete, accessible and ready for assessment.