CST2550 Reset Coursework

Ahmed Eissa, Andrew Lewis-Smith

January 27, 2025

Individual Coursework: Library Management System

1. Scenario

You have been hired as a software consultant by a local community library that needs a digital system to manage their collection. The library has been using paper records and now needs to transition to a digital system. They require a solution that allows efficient searching and management of their resources.

2. Requirements

2.1 Core Requirements

- Design and implement a Library Management System in C# .NET
- Create a data structure to store and efficiently search library resources
- Implement a SQL database to persist data
- Provide an intuitive user interface (console, WinForms, or .NET Core Web API)
- Perform time complexity analysis of your chosen data structure and algorithms

2.2 Functional Requirements

- Store details of library resources (books, journals, media) including:
  - Unique ID.
  - Title.
  - Author/Creator.
  - Publication Year.
  - Genre/Category.
  - Available Status.
- Allow resources to be added, removed, and updated.
- Implement at least two different search algorithms (e.g., by title, author, genre).
- Track borrowing status and due dates.
- Generate simple reports (e.g., overdue items, resources by category).

3. Technical Requirements

- Use C# .NET (Console, WinForms, or Web API)
- Implement custom data structures appropriate for the task (no STL data structures)
- Create and use a SQL Server database with Entity Framework
- Include comprehensive validation
- Implement unit tests
- Document time complexity analysis of key operations

4. Deliverables

- Source code (C# and SQL)
- Database creation script
- README with compilation and usage instructions
- Report (max 5 pages excluding references, cover, and contents)
- Video demonstration (max 6 minutes)

5. Report Requirements

The report should include:

- Introduction and description of the project
- Justification of selected data structures and algorithms
- Time complexity analysis with pseudo-code
- Testing approach and test cases
- Conclusion with limitations and reflection
- References (Harvard style)

6. Marking Criteria

- Data structure design and implementation (30%)
- Algorithm analysis and implementation (25%)
- Database integration (15%)
- User interface and functionality (15%)
- Code quality and testing (10%)
- Report quality (5%)

7. Submission

Submit all deliverables via either GitHub or MS Azure DevOps repository. The repository should be public and include all required materials.

When you finish your work, you need to invite us to your repo so we can look at your work and grant you the marks. So, you have to make everything **public.**

- For London Students:
  - Ahmed Eissa (a.eissa@mdx.ac.uk)
  - Andrew Lewis-Smith (A.Lewis-Smith@mdx.ac.uk)
- For MRU Students:
  - Karel Veerabudren (k.veerabudren@mdx.ac.mu)
  - Aisha Idoo (a.idoo@mdx.ac.mu)

The repo should contain the following:

- A tracked history of code commit.
- All the files mentioned in the “Deliverables” section above.

All **MUST** be submitted by 5:30pm on Tuesday, 01st July 2025.

DO NOT make any changes to anything in the repo following your submission or after the deadline – it will result in the submission being regarded as late and marks will be capped at 40%.

8. Academic Misconduct

We can now detect the use of AI in coding work, thanks to Originality.AI (https://originality.ai). They offer a tool that identifies whether the work has been generated by any AI platform, such as ChatGPT. This tool is intended for assistance purposes only and should not be used to complete entire assignments. Any work found to be copied from another source or created using AI will be considered academic misconduct and reported accordingly.

This is an individual assignment, and you are required to complete it independently. Do not copy work from another source without proper citation. Any material or ideas sourced from online content, textbooks, etc., should be appropriately referenced. Please familiarize yourself with the university's academic integrity and misconduct policy:

https://www.mdx.ac.uk/about-us/policies/university-regulations.

9. Extenuating Circumstances

There may be difficult circumstances in your life that affect your ability to meet an assessment deadline or affect your performance in an assessment. These are known as extenuating circumstances or 'ECs'. Extenuating circumstances are exceptional, seriously adverse, and outside of your control. Please see link for further information and guidelines:

https://unihub.mdx.ac.uk/your-study/assessment-and-regulations/extenuating-circumstances

Good luck.