**Faculty of Engineering and Information Technology**

**School of Computer Science**

31927 – Application Development with .NET

**JULY 2024**

**ASSIGNMENT 1 – Code Design Report Template**

|  |  |
| --- | --- |
| **Due date** | Wednesday 22 July 2024, 11:00am |
| **Demonstration** | Required in the lab/tutorial session |
| **Weight** | 35% |
| **Groupwork** | Individual |
| **Submission** | Complete project folder zip |
| **Submit to** | Canvas |

**Summary**

This report template needs to be filled out to receive marks for the “Code Design” & “Database Design” section of Assignment 1. Please include as much justification & detail as you feel is necessary to receive full marks. For full marks screenshots are encouraged alongside code references. This **must be included in your zip file submission**.

|  |  |
| --- | --- |
| Student Name | Olivia Stewart |
| Student ID | 24848424 |
| Lab Time | 7 pm |
| Lab Tutor |  |

**Justification**

For each criterion, include a justification as to how you have fulfilled it. Include references to your code & screenshots of how you applied the given code or database design principle.

|  |  |
| --- | --- |
| **Criteria** | LINQ is used in at least 3 different classes. |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | I used linq in a number of cases in their standard form.        However I also made use of linq’s class library in order to create some of my custom classes. |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 1.5/1.5 |

|  |  |
| --- | --- |
| **Criteria** | Nullable Reference Type checking is enabled in all files |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | Nullable reference type checking is enabled in the project and null reference exceptions avoided. |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 1/1 |

|  |  |
| --- | --- |
| **Criteria** | Unit tests are written for all classes. These should test both success & error scenarios with mocks for the database queries. |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | Not really possible to test for “all classes” as I have a ton of classes. But I have made a number of unit tests    These mock database queries by mocking the repository which serves as a layer of abstraction over db.    To test repository I mock db context like so |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 3/3 |

|  |  |
| --- | --- |
| **Criteria** | Helpful comments (At least 1 per method) |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | I made comments on many methods and classes to explain in the project. However, the actual architecture is described in the readme. |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 0.5/0.5 |

|  |  |
| --- | --- |
| **Criteria** | Appropriate indenting and whitespace |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | All files have correct indenting and whitespace. The ide handles this automatically anyway |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 0.5/0.5 |

|  |  |
| --- | --- |
| **Criteria** | Consistent and appropriate C# naming convention used |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | I have followed the C# naming convention throughout all my files. |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 0.5/0.5 |

|  |  |
| --- | --- |
| **Criteria** | The unit of work pattern is used in all database classes |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | All access to the database in my program is done through unit of work and repository pattern.        This is facilitated through unit of work factory class which provides me with instances of unit of work. |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 3/3 |

|  |  |
| --- | --- |
| **Criteria** | Strong OOP principles used.   * At least 1 example of inheritance * At least 1 example of method overloading is used * At least 1 example of method overriding is used * At least 1 example of an extension method * At least 1 example of a generic   At least 1 example of a delegate |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | Repository service uses delegate, but I also used in many other places such as      Inheritance and method overloading is demonstrated in many places, for example views, which are inherited in order to render to console. |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 3/3 |

|  |  |
| --- | --- |
| **Criteria** | Use of Low coupling & high cohesion |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | Solution is split into relevant projects  That perform their functions.  Furthermore, most classes are encapsulated by an interface and using DI, provided into constructor. This allows for little coupling. |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 2/2 |

|  |  |
| --- | --- |
| **Criteria** | At least 3 tables in the database with foreign & primary keys setup correctly. |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | My implementation uses five tables. There is an image showing the primary keys setup correctly as well in sql server. |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 1/1 |

|  |  |
| --- | --- |
| **Criteria** | Minimum 10 Rows of Seed Data Exists In Each Table |
| **Justification**  Include code, screenshots & any written documentation on how you met this criteria. | I seeded by parsing a CSV file with 1000 entries. There should be a couple hundred entries for each table. |
| **Recommended Mark**  Based on your justification, what mark do you believe you deserve. | 1/1 |