Service Oriented Architecture

# Assessment 1

**CA Value: 30%**

**Completion & Upload Date: 18th October 2024**

## Brief

You are required to design and implement *either* a console application, a desktop application (GUI) or a Web based application in C# which will access and use as fully as possible the facilities offered by *at least two* different third-party web service providers. The application should be “contextualized” in a specific application domain and NOT be simply a demo of web services. The web services consumed should be a minimum of two different services (REST/json).

**Note**: You are not required to create a database – you may keep test data in your code, or in a text/json/xml file. You are not required to develop login/logout functionality, but you can if you wish.

How you structure your project is up to you but make sure to display your knowledge of the following:

* Connecting to web services
* Classes and objects, including abstract classes and interfaces
* Arrays and arraylists
* Booleans and enums
* Reading data from a file

**Marks will be awarded as follows:**

1. Complete functionality 50%
2. Application Design and UI/UX 10%
3. Class structure & organisation 10%
4. Use of specified constructs (as listed above) 10%
5. Testing & general code quality 10%
6. Overall Impression/contextualization 10%

N.B. *Documentation may be included in the code or submitted in an MS-Word document not exceeding 3 pages.*

In your project read-me you should include links to any references or third-party tutorials you followed while creating your solution. Any sources (e.g. stack-overflow) you used when writing your code should be included in comments in the classes where they are used.

## Submission Checklist

Make sure to submit all the following to Moodle or your submission will not be marked:

|  |  |
| --- | --- |
| **GitHub Link**  Regular commits, relevant commit messages, minimum 2 branches  Must be either public, or have invited your lecturer as a collaborator |  |
| **Screencast**  5 minutes – talk through a running demo of your project including extra functionality, point out relevant code |  |
| **Coversheet**  Signed |  |
| **Zipped project**  Include everything needed to run the project, including DB configuration and seed data. |  |

## Academic Integrity

The assignment must be entirely the work of each student – in your own words. Students are not permitted to share any pseudocode or source code from their solution with any other student in the class. Students may not distribute the source code of their solution to any other student in any format (i.e., electronic, verbal, or hardcopy transmission). Any suspected plagiarism will be investigated, pursued, and reported to the Plagiarism Committee.

Generative artificial intelligence (AI) tools cannot be used in this assessment task. In this assessment, **you must not use** generative artificial intelligence (AI) (ChatGPT, ChatSonic, Bing Chat, Lex, DALL-E 2, or other tools) to generate any materials or content in relation to the assessment task.

The DkIT Academic Integrity Policy and Procedures, <https://www.dkit.ie/about-dkit/policies-and-guidelines/academic-policies.html>) states the following:

“Using generative artificial intelligence tools (e.g. ChatGPT) in an assessment unless explicitly permitted to do so and with proper acknowledgement, is a form of plagiarism”.

## Late Submissions

The institute’s standard policy on marking of late submissions will be applied:

* Submissions received up to 1 week late will have a 20% grade reduction applied.
* Submissions received up to 2 weeks late will have the above penalty applied and will be capped at 40%.
* Submissions received more than 2 weeks late cannot be graded.