.NET 2020 Exam

Name: Nikolai Svanåsbakken
Student number: 183554
Project title: Oiran

Requirements:

As the project works towards a remote database, access to the internet is required. The login details for the database are hardcoded into the DotNet.DataAccess-project's DotNetContext.cs-file, and the DotNet.DBApplication.WebApi-project's Startup.cs-file.
As the database is located on the school's network, access to this is required.

Known Issues:

Starting the API directly from within Visual Studio will open up a browser web page with a URL containing "/swagger" at the end. I could not figure out how to stop this from happening.

The list on the left-hand side of *master detail* did not show the database entries in the intended way, so I had to make do with overriding the VideoGame objects ToString()-method to get what I wanted. The result is ultimately more-or-less identical.

The project title isn't mentioned anywhere other than on the *main* page. I could not figure out how to edit the title bar.

The Application:

The application is a simple one where you are able to perform CRUD operations on a database that contains information about video games.
It contain four different pages:

The *main* page is nothing more than a welcome screen to serve as a starting point for the application. It serves no other practical purpose.
The *master detail* page loads in all video game data from the database asynchronously, and shows a list of the items with their main title and subtitle on the left hand side, and a more detailed view on the right hand side. There is a button that will navigate to *view content* and bring with it the selected object.
The *add content* page, as the name suggests, is where you can add an entry to the database.

The *view content* page should load a selected entry from the *master detail* page to allow for it being updated or deleted. This page is navigated to from *master detail* with the selected object, which is then used to fill in the text boxes. At this point the user may freely edit the object or delete it.

Side notes:

One of the NuGets used in this project, *Microsoft.Toolkit.Uwp.UI.Controls*, is necessary for the function of this application as it was automatically included on project creation. I am unsure if it's allowed or not as the requirements state that, apart from a select few exceptions, only NuGets that are "by Windows" are allowed. This NuGet is "by Windows.Toolkit". It is my opinion that it is allowed as it is still made "by Windows", but I am aware of the possibility that it isn't. If it isn't, I am sorry.