Assessment Task 1-3

Qualification national code and title ICT50220 - Diploma in Information Technology (Advanced Programming)
Unit/s national code/s and title/s ICTPRG554 - Manage data persistence using noSQL data stores
ICTPRG603 - Develop advanced mobile multi-touch applications

Assessment type (☑):

☐ Questioning (Oral/Written)
☐ Practical Demonstration
☐ 3rd Party Report
X Other – Portfolio

Assessment Resources:

College to supply:
- Computers running Windows 11
- Visual Studio
- Internet access
- Notes and links to online resources

Assessment Instructions:

This is part one of the Portfolio assessment, a group of exercises testing the requirements of the cluster. The Portfolio is due in week 18, although students are encouraged to submit earlier so there is more time for resits.

This part covers using web APIs and list views.

Assessment Instrument:

Setup

Create a new Xamarin.Forms C# project in Visual Studio. The name of the project should include your name and the name of the exercise above.

Brief

Your task is to create a currency converter app. The app should use a number pad similar to a calculator app and allow the user to enter an amount in Australian dollars, which will be converted into another currency. The currency to convert to should be selected from options in a scrolling list view.

The API

The app should use the following web API…

https://openexchangerates.org/

The site has documentation but the free plan is limited in what you can get. This is the API call you will need to use.

```
https://openexchangerates.org/api/latest.json?
app_id=744c0e65d9ae400eae78bcbf1151ff54
```

This will give you all the currency data as compared to the US dollar (USD) as JSON. You will need to extract the information you want by creating a class that matches the API data and then deserialising it into an object (or objects) of that class. You may need to do some research into JSON and may also find it useful to serialise an object of your class in order to compare its JSON with what you get from the API. If they match, you should be good.

Important Note

This is a free API and, since we're not paying for it, it likely has limits and may temporarily block you if you use it too much. As such, it's a good idea to minimise your use of it as much as possible during development and testing. One way of doing this is to copy the result it gets back to you and manually store it in a string. So, you could copy the API call you want to use into your browser, copy the result and store it in your code. You can then use the stored API result to test parts of your app without needing to call the API itself.

Don't forget to undo this before you hand it in - the final app should use the up to date information from the API call.

Testing

Test that the app fails gracefully under the following scenarios…

- If there is no internet connection.
- If you make a mistake in the URL. (Say, https://openex**chng**erates.org...) This is the same thing that would happen if the website moved.
- If you make a mistake in the API call. (Say "https://openexchangerates.org/**blargle**") This is the same thing that would happen if the API changed.

Finally, you should make sure the app is working by using Google to convert the same amounts and check. Be aware that the API Google uses and the API we're using may update at different intervals and there might be slight differences. As long as the numbers are in the right ballpark, it should be okay.

Submission

Compress the solution folder and submit the compressed file via Blackboard under "Assessment" in the sidebar.