Student Name : \_Cathal Henchy - 20091405\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Demo Video : \_<https://www.youtube.com/watch?v=WgkLDYHRNuY>\_

Github : \_<https://github.com/cathalohinse/aiteanna-speisiula>\_\_

Heroku : \_<https://aiteanna-speisiula.herokuapp.com/>\_\_\_\_\_\_\_\_\_

Glitch : \_<https://aiteanna-speisiula.glitch.me/>\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | User Accounts | POI Model | POI Features | Deployment | Git |
| Baseline | This has been achieved successfully. | This has been achieved successfully. The POI has a name, and a location. | This has been achieved successfully. The user can create a POI, and can view all POIs (both new & existing) on the ‘report’ view. | This ran in the local port (<http://LAPTOP-S80VBEOH:3000>) successfully. | This was committed to GIT using a GUI (Sourcetree). A very descriptive & interactive Readme.md file was also written. |
| Good | This has been achieved successfully. | This was achieved successfully using Cloudinary. | All POIs (both new & existing), can be updated, and deleted. | This was connected successfully to a mongodb cloud atlas cluster. | Multiple pre-releases, and working releases were published on GitHub, and tags also applied on Sourcetree. |
| Excellent | A User Settings screen is accessible, the User Account can be deleted, and there is basic admin. through the functionality of changing the User’s Settings. | A Category model was successfully created. This is also an attribute of the POI model. | New Categories (Province & County) can be created, and they (along with any pre-existing ones) can be deleted. | This was deployed successfully to Glitch. | Branches for Developing the code were created and successfully merged back to the master branch. A development branch was made for the Delete-user functionality and merged successfully to the master branch. |
| Outstanding | *This was not attempted.* | *This was not attempted.* | *This was not attempted.* | This was deployed successfully to Heroku. | The Release branch was created successfully. |

**Additional Notes:** Please refer to the GitFlow excel file. Only after I had completed this project, did I research Git Flow properly. Having felt that I completely understood the process, I wished to simulate feature & release branches - whereby the features are worked upon individually on their own branches, and occasionally merged with develop for backup and for collaboration etc., and whereby release branches are created when a deadline etc. approaches and a working release is required. The develop branch can continue working, whereas the release branch is created so that the desired features/updates etc. can be tidied up and ironed out and made ready for release. Once satisfied that all required work is done, the release branch is merged to the master branch (the release itself) and it is also merged to the develop branch, so that it can be incorporated into the existing work taking place concurrently to this release. I simulated this in sourcetree exactly as described, with both Feature1 and Feature2 being developed concurrently, and Release1 being developed concurrently with Feature2. However, as I didn’t push all features and releases to github with each commit (they weren’t ticked by default whenever I made a push, which I didn’t notice until after), I felt the need to take screenshots, of my Sourcetree work, as it wouldn’t have been captured in Github. These screenshots are contained in the excel file. Most changes made here were token changes, and therefore didn’t necessitate any redeployment.