API Development Assessment

# Introduction

This is the assessment for the API Development week. This assessment has been designed in relation to the SFIA framework, with the aim of assessing your ability to work towards SFIA level 2. This document details the content of the assessment and the deliverables required on its completion. It also details the scheme in which we will be marking this assessment; the marking scheme has been created in tandem with SFIA level 2 skills.

## Marking scheme

Below are the skills that we will be evaluating for this assessment. These skills are as described in the SFIA 7 framework; please see below if you wish to have more information:

https://www.sfia-online.org/en/framework

The skills this assessment will assess are the following:

Programming/Software Development – marking scheme

Below is the list of criteria that will be assessed from your deliverable:

|  |  |  |
| --- | --- | --- |
| **SFIA Skill** | **Rating** | **Details** |
| Designs, codes, verifies, tests, documents, amends and refactors simple programs/scripts. | 1 | Software is missing functionality in major areas for creating, reading, updating and deleting. No tests implemented and designs documented were not implemented. Best practices not adhered to in the project. |
| 2 | Software functionality is working in areas but is not a fully working product. Tests were not implemented; code was commented in small areas. Parts of the project adhered to best practices but not consistently throughout the software. |
| 3 | Software is functional in all major areas but still has small bugs and/or errors. Tests were beginning to be implemented, with basic functions being tested. Best practices were adhered to for most areas of the project. |
| 4 | Software is fully functional and has been tested in all relevant areas. Best practices were consistently adhered to throughout the project. |
| 5 | Software is fully functional and has been tested in all areas, with best practices and refactoring adhered to and implemented throughout the project. Software implements concepts outside of the brief specified at a good level. |
| Applies agreed standards and tools, to achieve a well-engineered result. | 1 | Tools and workflows discussed in software documentation around design not implemented to a quality capacity. |
| 2 | Tools and workflows are referred to within the project, but the implementation is minimal. |
| 3 | Tools and workflows discussed in documentation are implemented throughout the project at a high level, but more exploration could have occurred. |
| 4 | Software produced is in line with the documentation with changes made where needed, with references to refactoring and commenting apparent in key areas. |
| 5 | Software fully implements designs with justifications as to why changes were made where needed. Refactoring of code and commenting is rife throughout all relevant areas of the software. |

Systems integration and build – marking scheme

Below is the list of criteria that will be assessed from your deliverable:

|  |  |  |
| --- | --- | --- |
| **SFIA Skill** | **Rating** | **Details** |
| Produces software builds from software source code | 1 | VCS implementation was non-existent. Build server was not installed and therefore no builds of software were created. |
| 2 | VCS implementation was attempted but structure was poor and/or content in VCS was irrelevant. Build server was installed but software did not build successfully from repository. |
| 3 | VCS was implemented and code was stored in a structured manner. Some of the content in the VCS could have been omitted but the majority was relevant. Build server installed and successfully built software manually. |
| 4 | VCS was implemented and code was stored in a structured manner with branches. All VCS content was relevant, no unnecessary files. Build server installed and successfully built software after a push. |
| 5 | VCS was implemented and code was stored in a structured manner with branches, git ignore file and scripted hooks. All files relevant to the repo, nothing unnecessary. Build server installed and successfully built software after a push with artefact produced for successful builds. |
| Conducts tests as defined in an integration test specification, records the details of any failures. Analyses and reports on integration test activities and results. Identifies and reports issues and risks. | 1 | No tests written for back-end or front-end. No logs or reports about application produced. |
| 2 | Basic tests written for back-end, no implementation of front-end tests (or vice versa). Results produced not acknowledged in any way. |
| 3 | Basic tests written for both the back-end and front-end. Results are tabulated and logged but no further follow up. |
| 4 | Majority of tests for relevant features within the app are created in both front and back-end of application. Results of tests logged and discussed in report for test coverage. |
| 5 | Tests for all relevant CRUD features within the app are created in both the front and back-end of application. Results of tests logged and discussed in report for test coverage. Further addition of code quality checking is also implemented. |

Testing – marking scheme

Below is the list of criteria that will be assessed from your deliverable:

|  |  |  |
| --- | --- | --- |
| **SFIA Skill** | **Rating** | **Details** |
| Designs test cases and creates test scripts and supporting data | 1 | No tests written. |
| 2 | Basic tests written for back-end. Results produced not acknowledged in any way. |
| 3 | Adequate tests written for the back-end. |
| 4 | Majority of tests for relevant features within the app are created. |
| 5 | Tests for all relevant CRUD features within the app are created. Further addition of code quality checking is also implemented via a tool like SonarQube. |
| Analyses and reports test activities and results | 1 | No tests written. No logs or reports about application produced. |
| 2 | Basic tests written for back-end. Results produced not acknowledged in any way. |
| 3 | Results of tests are tabulated and logged but no further follow up. |
| 4 | Results of tests logged and discussed in report for test coverage. |
| 5 | Results of tests logged and discussed in report for test coverage. Further addition of code quality checking is also implemented, with the results provided. |