# Assessment Task Part C

|  |  |
| --- | --- |
| **Assessment Title:** | Testing |
| **Assessment Instructions:** | In this assessment students must confirm the application meets initial specifications.  This assessment task:   * involves two sub-tasks (Parts **C1 - Test procedures** and **C2 - Test outcomes**). * is a written and practical task in which the student must demonstrate that the completed program meets design requirements. * is open book * may be conducted in class, at a workplace or online depending on the student group   Students must:   * produce a written testing procedure * use referencing when using information from other sources * complete all tasks to a satisfactory level to receive a satisfactory result |
| **Duration of the Assessment:** | 2 hrs |
| **Required Knowledge** | * introductory level knowledge of programming language * debugging techniques * application testing methods |
| **Resources required for this Assessment:** | |
| **Supplied by Institute/workplace** | For class-based or workplace-based students:   * computer * word processing software * internet access |
| **Supplied by student** | For remote/online students:   * computer * word processing software * internet access |

# Task C1: Test Procedure

This task requires the student to document a test procedure which confirms the correct functioning of the completed computer program. The test procedure must confirm that the program meets the **Design Requirements** specified in the **Resources** section of this document (and also in your summary of those requirements in the Project Brief Checklist in Task A2 above). The purpose of testing in this case is to demonstrate to all stakeholders that the program as a whole meets requirements and is fit for purpose, above and beyond "breaking" errors.

The procedure must consist of:

* a written description of test conditions (eg. the type and structure of test data that must be available for testing).
* a written series of steps to follow to complete the test and confirm the correct functioning of the program.

The test procedure must operate on the model of "If I run the program with these inputs, I expect to see these outputs." It must cover both "success" cases (valid input) and "failure" cases (invalid or incomplete input).

*It may be possible in some circumstances for the student to write and run a formal code test using a tool such as Python's unittest module, but this is not required (as the work involved may be greater and more challenging than that of the original program being tested).*

**Product:** Written test procedure, which must be uploaded to SuniConnect when complete.

# Task C2: Test Outcomes

When the test procedure has been documented, the student must run the test procedure and record:

* the outcome of the test
* any problems identified
* any code updates or changes made because of the test

Repeat this cycle of testing and updating until the program satisfies all design requirements.

This process will be observed by the assessor and recorded using the Observation Checklist below.

**Product:** Written log of testing outcomes and any required code changes.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Task C2: Observation Checklist** | | | | | |
| ICTPRG301 Apply introductory programming techniques | | | | | |
| **Student’s Name:** |  | | | **Student ID:** |  |
| **Student Instructions:** | You will be observed by an assessor completing the following task/s. During the task/s you may be asked oral questions by the assessor to confirm your understanding.  Observations will be recorded by the assessor as **S** if the task/s has been performed to a satisfactory skill level or **NS** if the task/s have NOT been performed satisfactorily.  You must achieve a satisfactory result for the whole of the task. | | | | |
| **Description:** | This checklist records the outcome of the student’s tests to confirm that the program meets Design Requirements. | | | | |
| **Tasks to be observed** | | **1** | **Comments on performance and/or oral question responses (Optional):** | | |
| 1: Perform test(s) devised in Task C1. | | S  NS |  | | |
| 2: Record the outcome of the test. | | S  NS |  | | |
| 3: Record any problems identified. | | S  NS |  | | |
| 4: Record any code updates or changes made as a result of the test. | | S  NS |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Assessor Report** | | | |
|  | | | |
|  | | | |
|  | | | |
|  | | | |
| **Assessment Outcome:** | **SATISFACTORY** | | |
| **NOT SATISFACTORY** | Is resubmission required?Yes  No | |
| **Resubmission:** | Competency development strategies discussed with student? | | |
| Agreed due date for resubmission: / / | | |
| **Assessor Name:** |  | | |
| **Assessor Signature:** | A black background with a black square  Description automatically generated with medium confidence | | **Date:**  / / |