**Assessment Task – Portfolio and Observation (Student Version)**

**Assessment Task 1: Develop a Student Management System Phase 1**

|  |  |  |  |
| --- | --- | --- | --- |
| **Learner information** | | | |
| **Learner name:** | Panupong Jangjun | **Learner ID:** | 100688999 |

| **Section A – Program/Course details** | | | |
| --- | --- | --- | --- |
| **Qualification code:** | ICT40120 | **Qualification title:** | Certificate IV in Information Technology (Programming) |
| **Subject Code:**  **Unit code:** | (CPRO3)  ICTPRG430  ICTPRG431  ICTPRG432  ICTDBS416 | **Subject Title:**  **Unit title:** | (C# Programming)   * Apply introductory object-oriented language skills * Apply query language in relational databases * Develop data-driven applications * Create basic relational databases |

| **Section B – Assessment task details** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Assessment number:** | 1 of 2 | | **Semester/Year:** | | 2/2024 |
| **Due date:** | Session 9 | **Duration of assessment:** | | 9 sessions | |
| **Assessment task results:** | This assessment task will be marked as: | | | | |
| Ungraded result: Satisfactory or Not Satisfactory  Other (e.g. points): | | | | |

| **Section C – Instructions to learners** |
| --- |
| **Task instructions:** |
| This assessment is comprised of 2 parts.   Part A: Software design & development Part B: Knowledge questions  In Part A, you are required to design and develop an application according to the requirements.  In Part B, you are required to answer the Knowledge questions  You will find detailed information for each part in the [supporting document](#_Supporting_document) section.  For all parts in this assessment   * You are required to correctly answer all questions and complete all tasks as per instructions and assessment criteria to a satisfactory level for each question/task of this assessment to be given a satisfactory result by the assessor. * Once you have completed all the questions, the assessment must be uploaded and submitted along with the signed assessment coversheet via Brightspace. * If the result is not satisfactory within the enrolment period as per Holmesglen assessment procedure, you will be requested to resubmit within seven days of receiving feedback. You are permitted two resubmissions per assessment task. * You must contribute to and abide by organisational standards including intellectual property and privacy laws. * You may use the internet for research purpose however the learner’s answer must be in their own words.   See supporting documentation for further instructions |

| **Section D – Conditions for assessment** | |
| --- | --- |
| **Conditions:**  Learner to complete and attach Assessment Cover Sheet to the completed Assessment Task. | |
| * This is an **individual task.** * You must meet all criteria listed in the marking guide to be satisfactory in this task. * You must submit all required working files, documentation, and any other assets that you feel may be required in a zipped file, including the completed and signed coversheet. The assessment must be completed and submitted electronically to Brightspace by the due date. If this is not possible, you must contact your assessor to gain written approval for an alternative arrangement for submitting the assessment. * If not successful within the enrolment period as per Holmesglen assessment procedure, you will be requested to resubmit within 7 days of receiving feedback. You will have the opportunity to resubmit if any part of the assessment is deemed unsatisfactory (you are permitted TWO (2) resubmission per assessment task).   Resubmissions must be submitted by the resubmission due date provided by your teacher.   * This task is open book. You may use the internet for research purposes only. All answers must be in your own words. Where a quote is used, you must cite the information source. * If you feel you require special allowance or adjustment to this task, please discuss with your assessor within one week of commencing this assessment. Any change to assessment arrangements must be reviewed by the Education Manager and approved by the Head of Department. * You can appeal an assessment decision according to the Holmesglen Assessment Complaints and Appeals Procedure. * You are expected to dedicate time to develop this assessment task both in and out of the classroom. * Learners must contribute to and abide by organisational standards including intellectual property, privacy laws, and plagiarism and academic honesty. Further information is detailed at: <https://holmesglen.edu.au/Students/Student-Resources/> | |
| **Equipment/resources learners must supply:**  **Learners opting to BYOD laptop or intending to learn remotely will require access to:** | **Equipment/resources to be provided by the RTO:**  **This is a blended learning course and as such a remote learner will access their own computer equipment as per the specification provided. The Institute will provide the following:** |
| A PC/laptop with the following minimum specification:   * Quad Core CPU * 8GB of RAM * CPU with minimum 2GHz processor or faster * 200GB of Storage * Access to internet connection (ADSL or cable connection desirable) * Headset with microphone & webcam   Applications:   * Office 365 (including Word, Excel, PowerPoint…) – free to download through office.com login with your Holmesglen email * WebEx - free to download * 7Zip or an equivalent compression utility - free to download * OneDrive or google drive account for storage * Microsoft Visual Studio Community 2022 – free to download * MS SQL Server (already embedded in MS VS) | A Mac or PC/laptop with the following minimum specification:   * Quad Core CPU * 8GB of RAM * CPU with minimum 2GHz processor or faster * 200GB of Storage * Access to internet connection   Applications:   * Office 365 (including Word, Excel, PowerPoint…) * WebEx * 7Zip * OneDrive * Microsoft Visual Studio Community 2022 * MS SQL Server (already embedded in MS VS) |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Section E – Marking Guide** | | | | | | |
| **Assessment number:** | | 1 of 2 | **Assessment title:** | | Develop a Student Management System Phase 1 | |
| **Learner ID:** | | 100688999 | **Learner name:** | | Panupong Jangjun | |
| **Subject Code:**  **Unit code:** | (CPRO3)  ICTPRG430  ICTPRG431  ICTPRG432  ICTDBS416 | | | **Subject Title:**  **Unit title:** | | (C# Programming)   * Apply introductory object-oriented language skills * Apply query language in relational databases * Develop data-driven applications * Create basic relational databases |
| **Date:** | | Click here to enter a date. | | | | |

# Assessment Submission Cover Sheet (VET)

|  |  |
| --- | --- |
| **Student Declaration – Must be signed before submission** |  |
| By submitting this assessment task and signing the below, I acknowledge and agree that:  • This completed assessment task is my own work.  • I understand the serious nature of plagiarism and I am aware of the penalties that exist for breaching this.  • I have kept a copy of this assessment task.  • The assessor may provide a copy of this assessment task to another member of the Institute for validation and/or benchmarking purposes. | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Student ID:** | **100688999** | **Student name:** | **Panupong Jangjun** |

|  |  |
| --- | --- |
| **Submission or observation date:** |  |
| **Student signature**  For electronic submissions: By typing your name in the student signature field, you are accepting the above declaration. | **Panupong Jangjun** |

| Section F – Feedback to Student | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Has the student successfully completed this assessment task?** | | | | | **Yes** | **No** |
|  |  |
| **Additional Assessor comments (as appropriate):** | | | | | | |
| **Resubmission allowed:** | **Yes** | **No** | **Resubmission due date:** |  | | |
| **Assessor name:** |  | | | | | |
| **Assessor signature:** |  | | | | | |
| **Assessed date:** |  | | | | | |

# Supporting document

# Task: Develop a Student Management System Phase 1

# The Project Brief

## You’ve been tasked with designing and developing a Student Management System application for educational institutions. The application should allow administrators to manage student records, including adding new students, updating existing student information, deleting student records, and viewing student details. The system should be developed using C#, following object-oriented programming principles, and should implement CRUD operations with an SQL server database.

## Development requirements

Here are the requirements for the "Student Management System Phase 1" program:

|  |  |
| --- | --- |
| 1 | User can use a command line-based console interface (CLI) to operate the system. |
| 2 | User can add new student records, update existing student information, delete student records, and view student details. |
| 3 | The system should store student data in an SQL server database and implement CRUD operations. |
| 4 | The student table should include the following fields (Student Id, Full name, Phone, Email, Date of birth, Enrolment date,  Enrolment certificate, Total score):   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **StudentId** | **FullName** | **Phone** | **Email** | **DoB** | **EnrolmentDate** | **EnrolmentCert** | **TotalScore** | |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  | |
| 5 | The system should include mathematical calculations in the CLI list to help perform:   * Average of total scores * High and low scores * Calculations of student ages |
| 6 | Optional requirements:   * Keep Program Running Until User Decides to Exit: The system should continuously run, presenting the main menu after each operation, until the user chooses to exit. * Help Menu: Add a help menu option that provides instructions on how to use the system. |

You are required to:

* Design and develop a software program that meets the above requirements.
* Perform and document unit and system tests for the software program.
* Fix issues and bugs if exist to make the system error-free.

You are also required to follow the questions in the Part A and Part B below to finish all the tasks and answer all the questions.

## Development environment

* Visual studio
* SQL Server Management Studio

## Organisational guidelines

* C# Coding Conventions - <https://docs.microsoft.com/en-us/dotnet/csharp/fundamentals/coding-style/coding-conventions>
* Software test principles and guidelines. See attached file below.



**Please use the check list below to check if you have got everything ready before submitting the assessment for review.**

|  |  |
| --- | --- |
| **Submission check list** | |
| Assessment document with all questions answered |  |
| All source codes |  |
| The database file/SQL query |  |
| The testing report document |  |

## 

## Part A – Software design & development

In this part, you are required to design and develop the application using a selected programming language (e.g. C#).

|  |  |
| --- | --- |
| **A.1.** Read the project brief for this application, review and clarify development requirements with your client (your assessor). And list all the requirements using bullet points. | |
| Answer:   * User can use CLI to operate the system * All the data should store in an SQL server database the user can add new data, update the existing data, delete data, and view the data. * Must have the exact name and data type on database as below      * The program be able to calculate average of total score, high and low score, and student ages * Optional feature (not require) - Keep program running until user decide to exit. - Help menu option to provide the instruction. | |
| **A.2.** Plan and determine the class hierarchy (inheritance tree) to satisfy user requirements. You are required to use the following notation to represent the inheritance relationship.  A diagram of a structure  Description automatically generated | |
| Answer: | |
| **A.3.** Plan and determine the member fields and methods for **each** class documented in question 2 above. You are required to use the following notation to represent a class.  A white rectangular object with black text  Description automatically generated | |
| Answer: | |
| **A.4.** In terms of object-oriented programming, explain “inheritance”, “polymorphism” and “overriding methods”. You may explain the three terms by using an example. | |
| Answer:   |  |  | | --- | --- | | **Terms** | **Your answers** | | inheritance | Inheritance is an OOP concept that allows other class to inherit (child) the methods and properties of another class (parent) | | polymorphism | polymorphism is an OOP concept that allows multiple methods in the same class to have a same name but different parameter to make the code more flexible | | overriding methods | overriding methods is an OOP concept that allows the child class to change the statement of the parent method that is already defined  for perform overriding methods have to add the “Virtual” keyword in the parent class on the method that want to override,  Add the “Override” keywords in the child class on the method that you proving a new statement. | | |
| **A.5** Implement all classes according to the design above using an OOP programming language (e.g. C#). You need to make sure:   * Inheritance and polymorphism (overriding methods) have been correctly implemented. * Code conventions are followed * Codes are fully documented using comments   Provide a **screenshot** of the codes for each class definition. | |
| Answer:   |  |  | | --- | --- | | Class 1 |  | | Class 2 |  | | Class 3 |  | | Class 4 |  | | Class 5 |  | | Class 6 |  | | Class 7 |  | | Class 8 |  |   I have followed code conventions  I have fully documented using comments | |
| **A.6** Implement two versions of constructors for all classes with member fields.   * One constructor accepts zero parameter and initialize each member fields to a reasonable default value. * The other constructor accepts several parameters for initializing all member fields.   Provide a screenshot of the two constructors for each class. | |
| Answer:   |  |  |  | | --- | --- | --- | | Class Name | Constructor #1 – without parameter  Screenshot | Constructor #2 – with parameter  Screenshot | | Program |  |  | | Display |  |  | | MainMenu |  |  | | StudentMenu |  |  | | CalculationMenu |  |  | | Class Name | Constructor #1 – without parameter  Screenshot | Constructor #2 – with parameter  Screenshot | | MenagementMenu |  |  | | GeneralMethod |  |  | | Database |  |  | | |
| **A.7** Implement the console user interface according to the requirements. Provide a screenshot of the codes.  You need to make sure:   * Code conventions are followed * Codes are fully documented using comments | |
| Answer:              I have followed code conventions  I have fully documented using comments | |
| **A.8** Implement the database table for storing student data. Provide a screenshot of the table structure in the SQL server. | |
| Answer: | |
| **A.9** Implement the **Add Student** method and provide a screenshot of the code and a screenshot showing a student record being added to the database. | |
| Answer:   * Screenshot of code:      * Screenshot of added student record: | |
| **A.10** Implement the **Update Student** method and provide a screenshot of the code and a screenshot showing a student record being updated in the database. | |
| Answer:   * Screenshot of code:      * Screenshot of updated student record: | |
| **A.11** Implement the **Delete** **Student** method and provide a screenshot of the code and a screenshot showing a student record being deleted from the database. | |
| Answer:   * Screenshot of code:      * Screenshot of deleted student record: | |
| **A.12** Implement the **View Student** method and provide a screenshot of the code and a screenshot showing student details being retrieved from the database. | |
| Answer:   * Screenshot of code:      * Screenshot of retrieved student record: | |
| **A.13** Implement the **average** **score** calculation and provide a screenshot of the code and a screenshot showing the average score calculation in the console. | |
| Answer:   * Screenshot of code:      * Screenshot of average score calculation: | |
| **A.14** Implement the **high score** calculation and provide a screenshot of the code and a screenshot showing the high score calculation in the console. | |
| See the detail in A.15 | |
| **A.15** Implement the **low score** calculation and provide a screenshot of the code and a screenshot showing the low score calculation in the console. | |
| Answer:   * Screenshot of code:      * Screenshot of age calculation: | |
| **A.16** Implement the **age** calculation and provide a screenshot of the code and a screenshot showing the age calculation for a student in the console. | |
| Answer:   * Screenshot of code:      * Screenshot of age calculation: | |
| **A.17:** Develop a **unit test** for one class and list at least 3 requirements with bullet points. |
| Answer:  *Provide screenshots of the unit test requirements:*  Requirement 1: *e.g., Validate that the input data is not null or empty.*  Requirement 2: *e.g., Ensure that the system calculates the average scores correctly.*  Requirement 3: *e.g., Check that the input data falls within the acceptable range of values, (ages between 16 to 100 y.o.).* |
| **A.18:** Develop a system test for the application and list at least 3 requirements with bullet points. |
| Answer:  *Provide screenshots of the system test requirements:*  Requirement 1: *The system test should validate that the command line interface correctly displays the menu options.*  Requirement 2: *The system test should verify that the system correctly performs CRUD operations for student records.*  Requirement 3: *The system test should check that the system calculates the average, high, and low scores correctly.* |
| **A.19:** Produce and save the test report **using provided template**. Write the submitted test report file name below. |
| Answer:  Test report file name: |

|  |
| --- |
|  |

| **Skills to be observed for part A by assessor** | | **1. Date:** | | **2. Date:** | | **Comment** |
| --- | --- | --- | --- | --- | --- | --- |
|  | |  | |
| **Satisfactory** | | **Satisfactory** | |
| **Yes** | **No** | **Yes** | **No** |
| **A.20** | Learner correctly implemented polymorphism for code extensibility. |  |  |  |  |  |
| **A.21** | Learner applied documentation conventions and documented the application according to organisational documentation conventions. |  |  |  |  |  |
| **A.22** | Learner applied code conventions to organisational requirements. |  |  |  |  |  |
| **A.23** | Learner communicated relationships between ideas and information, in a style appropriate to the audience and purpose, and selects the vocabulary, grammatical structures and conventions appropriate to the text, in relation to coding, recording outcomes, and documenting activities. |  |  |  |  |  |
| **A.24** | Learner selected from, and flexibly applies, mathematical and problem-solving strategies and techniques, in a programming context. |  |  |  |  |  |
| **A.25** | Learner recognised and followed, explicit and implicit standard and meets expectations associated with own role when developing code that is compliant with standards and guidelines. |  |  |  |  |  |
| **A.26** | Learner used a formal decision-making process, identifying and evaluating several choices against a limited set of criteria, when selecting language data types, operators and expressions. |  |  |  |  |  |
| **A.27** | Learner evaluated the effectiveness of decisions, in terms of how well they meet the stated design specifications. |  |  |  |  |  |
| **A.28** | Learner used analytical processes to decide on a course of action when debugging. |  |  |  |  |  |
| **A.29** | Learner utilised features within applications to develop software programs. |  |  |  |  |  |
| **A.30** | Learner recognised, and used language and symbols, when applying the coding syntax. |  |  |  |  |  |
| **A.31** | Learner actively identified systems, devices, and applications with the potential to meet current and future needs regarding programming. |  |  |  |  |  |

## Part B – Knowledge Questions

|  |
| --- |
| **B.1:** What do we mean by data modelling? What tools are used in data modelling?  (you may visit this [article](https://www.simplilearn.com/top-powerful-data-modeling-tools-software-article#:~:text=Data%20modeling%20involves%20devising%20a,of%20an%20optimal%20data%20framework.) to support with your answer) |
| Answer: |
| **B.2:** What are the steps you should undertake when designing a database? (see this [article](https://support.microsoft.com/en-au/office/database-design-basics-eb2159cf-1e30-401a-8084-bd4f9c9ca1f5) for more information) |
| Answer: |
| **B.3:** What is a Database Management System. Give at least two examples of popular DBMS’s. |
| Answer: |
| **B.4:** What are the principles of Structured Query Language (SQL) |
| Answer: |
| **B.5:** What do primary and foreign keys mean in a database table? |
| Answer: |
| **B.6:** Explain the difference between based on purpose, commands and impact:   * Data definition language (DDL), * Data manipulation language (DML) and * Data control language (DCL) |
| Answer:   |  |  |  |  | | --- | --- | --- | --- | | **Compare based on:** | **DDL** | **DML** | **DCL** | | **Purpose** |  |  |  | | **Commands** |  |  |  | | **Impact** |  |  |  | |
| **B.7:** Identify organisational guidelines and frameworks related to testing. List at least 3 guidelines. You can refer to the following Test Principles and Guidelines for your answer, (*double click to open*): |
| Answer:  1.  2.  3. |
|  |

| **Skills to be observed for this Part B by assessor** | | **1. Date:** | | **2. Date:** | | | **Comment** |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | | |
| **Satisfactory** | | **Satisfactory** | | |
| **Yes** | **No** | **Yes** | | **No** |  |
| **B.8** | Learner can discuss why is it important to gather client requirements for the database? And able to give three examples of information gathering methods. |  |  |  |  | |  |
| **B.9** | Learner can identify the necessary systems and protocols for utilizing query language in relational databases. |  |  |  |  | |  |
| **B.10** | Learner can use correct naming conventions for database design. |  |  |  |  | |  |
| **B.11** | Learner can discuss how you would implement security on a database. |  |  |  |  | |  |
| **B.12** | Learner analysed and identified test data using multiple test-case design techniques. |  |  |  |  | |  |
| **B.13** | Learner can discuss the debugging and testing approaches and techniques in software development. |  |  |  |  | |  |
| **B.14** | Learner took responsibility for planning, sequencing and prioritising tasks and own workload. |  |  |  |  | |  |