Assessment 2: Software Project

Start Assignment

Due 2 Dec by 23:59   **Points** 100   **Submitting** a file upload   **Attempts** 0
Allowed attempts 2

IT5016D: Software Development Fundamentals

Assessment 2: Software Project

This assessment represents 50% of your final grade.

Assessment Overview

Think about the Software Project like a piece of art, only you're creating a *Software Project* based on your own technical research and emerging practice.

Learning Outcomes

LO1: Develop program logic to a professional working brief.

LO2: Apply the stages of software development to develop a simple code solution.

Conditions

- It is recommended that you spend between **20 and 25 hours** on this assignment. It is better to add content as you find it, rather than at the end.
- All course materials, and any other resources, can be used to complete this assessment.
- The work you submit must be your own work. It is an individual assessment.
- You can ask your facilitator to clarify the instructions, and/or for advice, but they cannot do or solve the assessment tasks – you must carry out the tasks yourself!

Assessment Instructions

This assessment gives hands-on experience of the language of the Software Development Fundamentals.

1. Please read the scenario below and apply code solutions to develop a Help Desk ticketing system prototype.

2. Record the stages of the Software Development Life Cycle (SDLC) and include with the submission of your Software Project.

It is valuable to record your experiences during the Software Development Lifecycle (SLDC), which is why we ask you to submit this with your Software Project.   These experiences will help you to professionally communicate and contextualize your practice as a professional and build confidence in your technical skills and practice.

Scenario

Help Desk Ticketing System Prototype

Client Requirements

The client would like a Help Desk ticketing system prototype developed.

The Help Desk ticketing system should handle tickets from **internal customers only**.

Tickets will be requested for assistance **from** the Help Desk **by** staff members of the organisation.

Requirements of the Help Desk Ticketing System

Tickets:

- Tickets can be submitted by providing all of the following information:
- Staff ID,
- Ticket creator name,
- Contact email
- Description of the issue

Internal Tickets' ticket number should be assigned automatically using the **counter static field plus 2000.**

All information must be provided as input while submitting the ticket.

Responding to tickets:

- If the ticket's description of the issue contains the words *"Password Change",* the new password should be generated following the rule,
- T*he first two characters of the staffID, followed by the first three characters of the ticket creator name.*

Hint: (can be useful to consider: split(), join(), string operations)

- There should be an option, after the ticket has been submitted, to **respond** to a ticket by providing a feedback response.

Default response can be set as "Not Yet Provided".

Statistics:

- There should be a way to keep track of:
- The number of tickets submitted
- The number of resolved tickets
- The Number of open tickets
- A way to display those statistics to the console.

- If the **staff member has submitted** the "Password change" request, after the new password is generated and the ticket's response has been updated, the ticket should close, with the number of closed tickets increased and the number of open tickets reduced by 1. Ticket's status should be changed to "Closed".

- Once a member of the **IT department provides the response to a ticket**, the ticket should close, with the number of closed tickets increased and the number of open tickets reduced by 1. Ticket's status should be changed to "Closed".

- There should be an option for the **IT department to reopen the ticket**. At this point the number of open tickets should be increased and the number of closed tickets should be reduced by 1. Ticket's status should be changed to "Reopened"

Displaying the ticket:

- There should be a way to display the ticket information:
- Ticket number,
- Name of the ticket's creator,
- StaffID,
- Email address,
- Description of the issue,

- Response from the IT department and ticket status (open, closed or reopened).

The output format is shown in the examples at the end of Technical Requirements section.

Technical Requirements

The senior developer has provided you with the following technical requirements for the project.

- The Ticket class should contain common ticket information in the Ticket class.
- The Ticket class should also have method allowing the staff to submit ticket and the IT team to respond to the tickets, specifically resolve, reopen and provide a response to the ticket.
- The Ticket class should contain a method for resolving password change requests. As well as calling the method that would generate the new password, it should set up a response for the ticket and change the ticket status to closed.
- There should be a method to print information for all the ticket objects.

    Hint: research and use List<Ticket>

- The TicketStats method in Ticket class should contain information on ticket statistics and shall be able to return the statistics information.
- The main class, containing the Main method.
- Create at least one instance of submitting tickets and include at least one ticket with the request of "Password change".
- After the tickets are created, display ticket statistics.
- Resolve some of the tickets, then display the ticket information and ticket statistics. o Reopen some of the resolved tickets, then display the ticket information and ticket statistics.

- **The example output is provided below:**

Displaying Ticket Statistics

Tickets Created: 3

Tickets Resolved: 1

Tickets To Solve: 2

Printing Tickets:

Ticket Number: 2001

Ticket Creator: Inna

Staff ID: INNAM

Email Address: inna@whitecliffe.co.nz

Description: My monitor stopped working

Response: Not Yet Provided

Ticket Status: Open

Ticket Number: 2002

Ticket Creator: Maria

Staff ID: MARIAH

Email Address: maria@whitecliffe.co.nz

Description: Request for a videocamera to conduct webinars

Response: Not Yet Provided

Ticket Status: Open

Ticket Number: 2003

Ticket Creator: John

Staff ID: JOHNS

Email Address: john@whitecliffe.co.nz

Description: Password change

Response: New password generated: JOJoh

Ticket Status: Closed

Displaying Ticket Statistics

Tickets Created: 3

Tickets Resolved: 2

Tickets To Solve: 1

Printing Tickets:

Ticket Number: 2001

Ticket Creator: Inna

Staff ID: INNAM

Email Address: inna@whitecliffe.co.nz

Description: My monitor stopped working

Response: The monitor has been replaced.

Ticket Status: Closed

Ticket Number: 2002

Ticket Creator: Maria

Staff ID: MARIAH

Email Address: maria@whitecliffe.co.nz

Description: Request for a videocamera to conduct webinars

Response: Not Yet Provided

Ticket Status: Open

Submission Checklist

Task	Evidence Required
1	*Software_Project* Python Code file
2	*ReadMe* file with instructions
3	Word/PDF file with the details of Software Development Lifecycle stages during the development of the project
4	Instructions required to display your software project

Success Criteria

The assessment will be marked according to the rubric below. Please take time to read it and ask for clarification if necessary.

You are required to obtain a minimum of 50% of the total available marks to be successful in this assessment.

A maximum of two attempts are allowed to complete this assessment. The maximum percentage to be awarded on a second assessment attempt is 50%.

Submission instructions

When you have completed the assessment:

- Verify that all items have been completed using the submission checklist provided
- Navigate to the assignment item in Canvas (Dashboard > Select *Course* > *Assignment* tab > **Assessment 2: Software Project**).
- Please ensure your document is named "*IT5016D_Assessment 2_<StudentID>*".
- Upload your assessment document using upload facility in Canvas
- Read the declaration and submit the assessment

Declaration

By submitting your work, you are indicating that you agree to the following declaration:

"The work presented in this assessment is to the best of my knowledge original, except as acknowledged in the text, and the material has not already been submitted, either in whole or in part, for any academic award at this or any other tertiary institution. I promise not to share this assessment in part or whole with any other student at Whitecliffe or outside this campus."

| IT5016D Assessment 2 Software Project |

Criteria	Ratings				Pts
Software Development Life Cycle (SDLC)	**20 to >16.0 Pts Appropriate stages of the software development life cycle were followed and includes a description of each stage (requirements analysis and solution design, solution development, testing).**	**16 to >13.0 Pts Some stages of the software development life cycle were followed and includes a description of some of these stages.**	**13 to >10.0 Pts General indication the appropriate stages of the software development life cycle were followed with little description of any of the stages.**	**10 to >0 Pts Little or no attempt to follow the appropriate stages of the software development life cycle and/or describe any of the stages.**	20 pts
Solution Development: Code to re-set password	**20 to >16.0 Pts Code to reset the password correctly reset the password using String Operations.**	**16 to >13.0 Pts Code to reset the password shows correct use of String Operations to some extent and produces a password following the client requirements specifications.**	**13 to >10.0 Pts Code to reset the password shows correct use of String Operations but does not produce a password following the client requirements specifications.**	**10 to >0 Pts The code to reset the password does not show the correct use of String Operations and does not produce a password following the client requirements specifications.**	20 pts

Criteria	Ratings				Pts
Solution Development: Statistics	**20 to >16.0 Pts** Object Oriented concepts are used correctly in the solution development.	**16 to >13.0 Pts** Object oriented concepts are used in the solution but are not implemented in the most efficient way.	**13 to >10.0 Pts** Minimum Object-oriented concepts are used in the solution development.	**10 to >0 Pts** Object-oriented concepts are not used appropriately.	20 pts
Solution Development: Technical Requirements	**20 to >16.0 Pts** Evidence of technical requirement analysis and solution design stages match the specifications of the brief and more than one 'Classes' are used in the solution development.	**16 to >13.0 Pts** Evidence of technical requirement analysis and solution design stages match the specifications of the brief.	**13 to >10.0 Pts** Evidence of technical requirement analysis and solution design stages partially match the specifications of the brief.	**10 to >0 Pts** Evidence of technical requirement analysis and solution design stages is not provided and/or the evidence does not match the specifications of the brief.	20 pts
Solution Development: Compile and run	**20 to >15.0 Pts** The solution created compiles and runs as expected.	**15 to >10.0 Pts** The solution created compiles but does not run as expected.		**10 to >0 Pts** The solution created does not compile.	20 pts
				Total points: 100	