



JOHN EDMONDSON HIGH SCHOOL

Assessment Notification

Faculty: Industrial Arts Course SOFTWARE DESIGN and DEVELOPMENT Year: 12

Assessment Task: HSC ASSESSMENT TASK 3

Assessment Weighting: 30% Due: Term 3 Week 1 Date: 26/07/2024 by 8:20am

Task Type: Hand in Task In Class Task Practical Task

Outcomes assessed (NESA)

H5.1 applies project management techniques to maximise the productivity of the software development

H5.2 creates and justifies the need for the various types of documentation required for a software solution

H6.1 assesses the skills required in the software development cycle

H6.3 uses and describes a collaborative approach during the software development cycle

H6.4 develops and evaluates effective user interfaces, in consultation with appropriate people

H4.1 identifies needs to which software solutions are appropriate

H4.2 applies appropriate development methods to

Task Description/Overview

The goal of this course is to assess the design and development of software solutions, covering various development approaches and techniques.

Detailed Assessment Task Description

Submission MUST be uploaded to CANVAS by 8.20am 26/07/2024.

Submission is to be in the form of a PDF of Word Document.

Assessment Criteria (refer to attached assessment documentation for more detail)

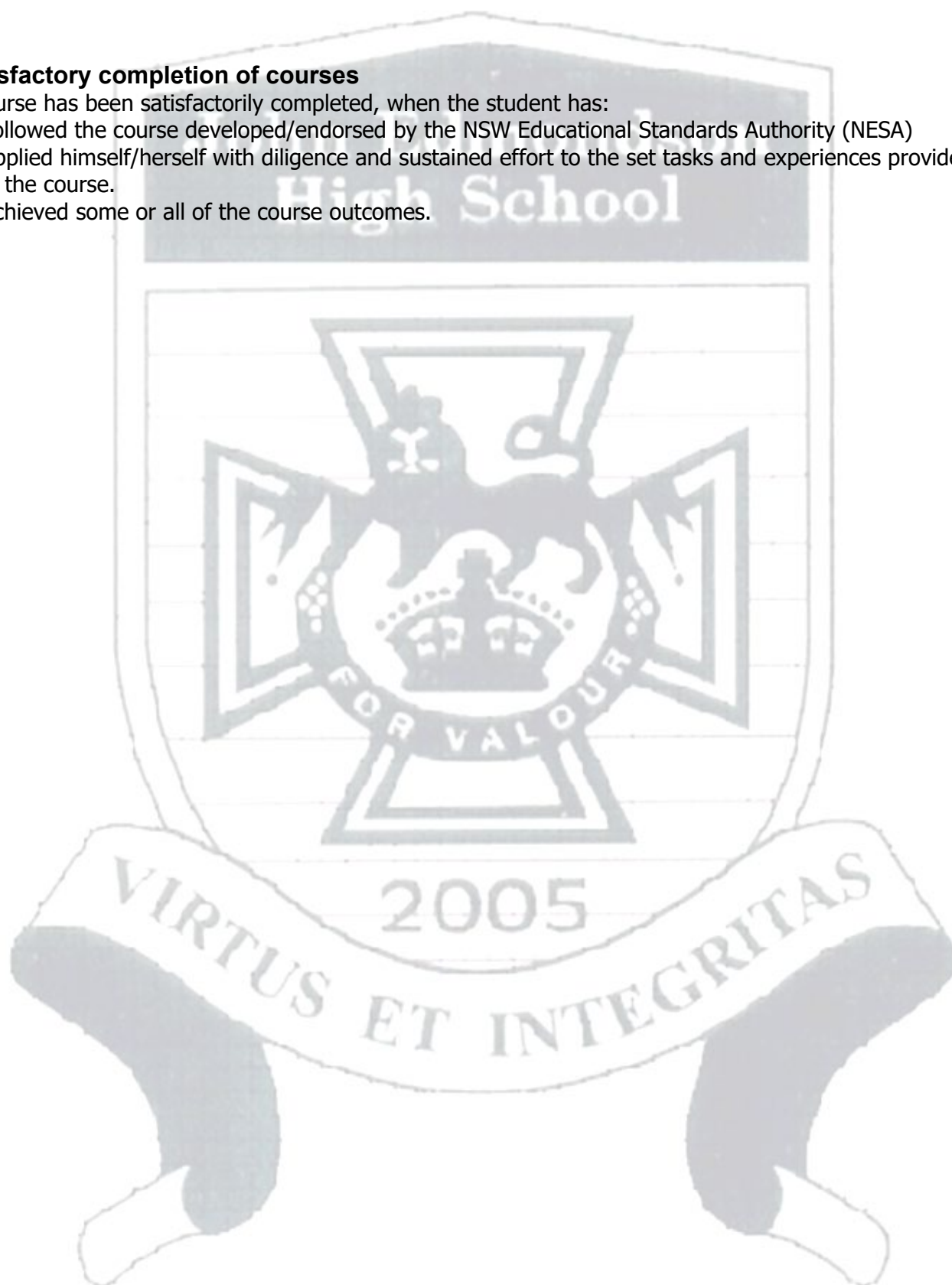
Grade	Description	Mark Range
Outstanding (O)	The student has an extensive knowledge and understanding of the content and can readily apply this knowledge. In addition, the student has achieved a very high level of competence in the processes and skills and can apply these skills to new situations.	90-100
High (H)	The student has a thorough knowledge and understanding of the content and a high level of competence in the processes and skills. In addition, the student is able to apply this knowledge and these skills to most situations.	80-89
Sound (S)	The student has a sound knowledge and understanding of the content and has achieved a good level of competence in the processes and skills.	60-79
Basic (B)	The student has a basic knowledge and understanding of the content and has achieved a basic level of competence in the processes and skills.	30-59
Limited (L)	The student has an elementary knowledge and	0-29

	understanding in a few areas of the content and still required further work to achieve competence in the processes and skills.	
--	--	--

Satisfactory completion of courses

A course has been satisfactorily completed, when the student has:

- Followed the course developed/endorsed by the NSW Educational Standards Authority (NESA)
- Applied himself/herself with diligence and sustained effort to the set tasks and experiences provided in the course.
- Achieved some or all of the course outcomes.



HSC Software Design and Development

HSC Assessment Task - 3

High School

Due Date: Term 3, Week 1, Friday - (26/07/2024) by 8:20am

Submission MUST be uploaded to CANVAS as a PDF or Word document by 8.20am.

Only submit on CANVAS, no hardcopy required.

Assessment Outcomes

A student:

H5.1 applies project management techniques to maximise the productivity of the software development.
H5.2 creates and justifies the need for the various types of documentation required for a software solution.

H6.1 assesses the skills required in the software development cycle.

H6.3 uses and describes a collaborative approach during the software development cycle.

H6.4 develops and evaluates effective user interfaces, in consultation with appropriate people.

H4.1 identifies needs to which software solutions are appropriate.

H4.2 applies appropriate development methods to solve the problem.

Context

The goal of this course is to assess the design and development of software solutions, covering various development approaches and techniques.

Project management ensures projects meet their objectives on time. In commercial software teams, project managers oversee the overall design and development process. They need strong communication, time-management, and adaptability skills to motivate teams, resolve conflicts, and focus resources on achieving project goals.

The Software Development Life Cycle (SDLC) offers a structured framework for project development. Software development is iterative, with evolving requirements and new ideas emerging during the process. Effective processes and techniques should be flexible enough to accommodate these changes.

Scenario

Design and create a system for retail shop frontend of your choice (an example would be the register at Michelle's Patisserie or a similar retail outlet). You need to describe the system modelling tools you use for your program.

Sales system: Your system must be able to sell the products of your shop.

- This sale system is required to perform the transaction for each individual sale. It must be able to add and subtract items from the sale as per a regular POS system.
 - **Company Name / Theme:** You are to create a franchise identity including a logo, themed colours, etc. to establish brand identity.
 - **Daily Sales:** Your System must also tally up the total number of a product sold, the total amount of money made on each product, and the total amount of money made in total.
 - **Additional Modules** - You should also expand on your system by including additional functionality including stock or employee management. Be creative!
-

Phase 1 – Defining and Understanding the Problem

To understand the depth and complexity of the problem, we commence by creating system modelling tools.

- A** Define the need of your system. What do you require your system to perform and what are the environment boundaries for your system.
 - B** Criteria for Success. A self-assessment checklist.
 - C** Create a System Flowchart, DFD, and Structure Chart for your system.
-

Phase 2 – Planning and Designing Software Solutions

- a. Create annotated screen designs for your GUI (use can use adobe XD, or can create a webpage or Canva for you GUI design)
 - b. Create an IPO, Flowchart and Pseudocode for any 3 of your subroutines.
-

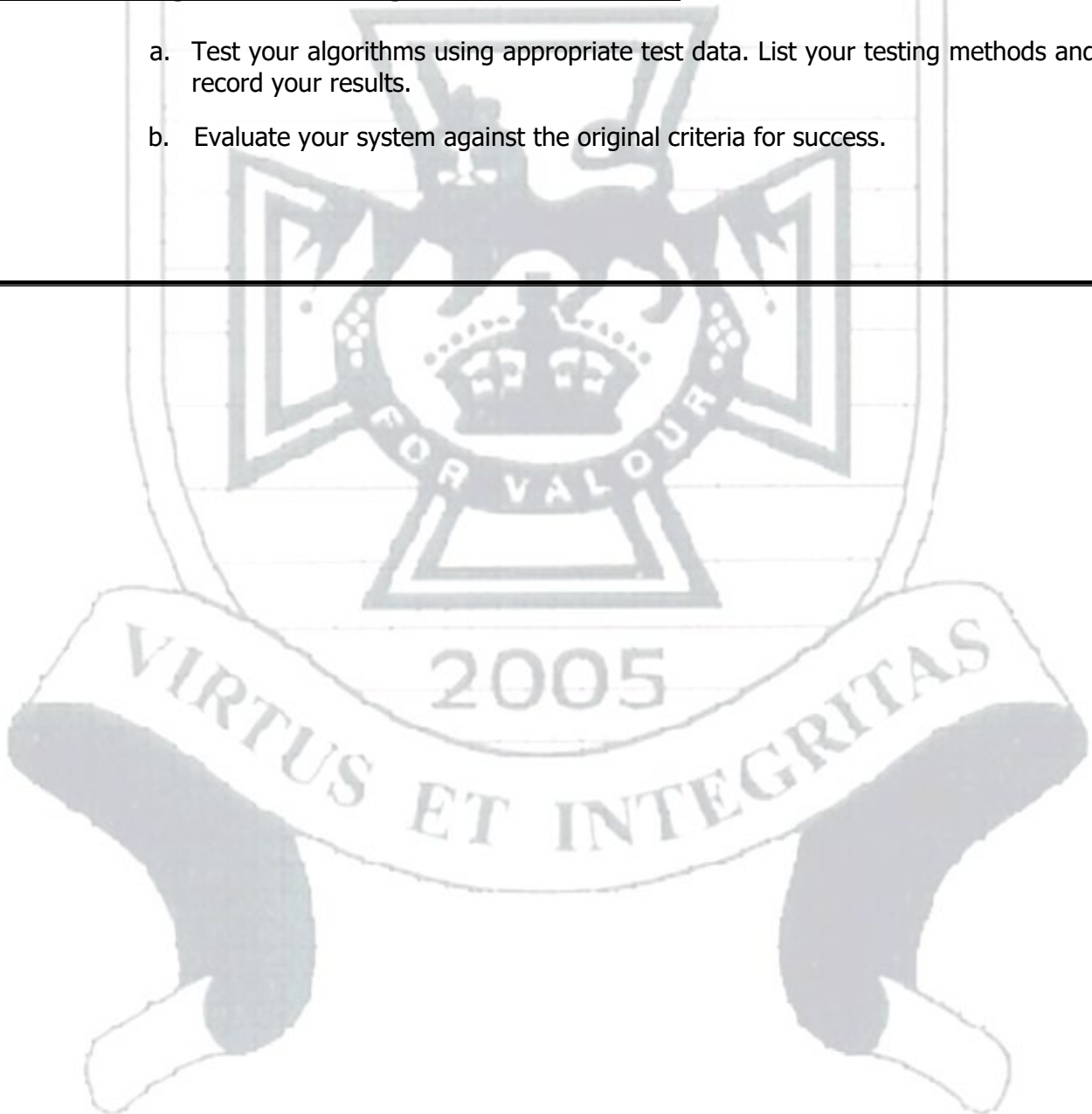
Phase 3 – Implementing Software Solutions

- a. User Documentation. Create a user manual for your software program. Can be in any form that is easy for the end user to access and understand.
- b. Provide a lexical analysis for one of your algorithms with the token list provided to you in the class.

John Edmondson
High School

Phase 4 – Testing and Evaluating Software Solutions

- a. Test your algorithms using appropriate test data. List your testing methods and record your results.
- b. Evaluate your system against the original criteria for success.



Marking Criteria

Outcome	N	E	D	C	B	A	Mark
Section 1 – Defining and Understanding the Problem							
Define the need of your system		States a problem with little or no detail.	Provides a brief explanation of the problem. Briefly outlines how and why the project is being undertaken. Provides a brief explanation of how a solution may be developed	Provides a sound explanation of the problem. Briefly outlines how and why the project is being undertaken. Provides a brief explanation of how a solution might be developed in detail.			/5
Criteria for Success		Provides some criteria for success	Provides sound criteria for success	Provides an exemplar Criteria for success clearly outlining requirements for a successful project			/5
Section 2 – Planning and Designing the Solution							
System Models		Demonstrates a minimal understanding of the use of system modelling tools	Demonstrates a sound understanding of the use of system modelling tools	Demonstrates an outstanding understanding of the use of system modelling tools			/20
Screen Designs		Some drawings or something.	Satisfactory screen designs with comments	Detailed screen designs with annotated comments			/10
IPO and Algorithms		Attempt algorithms for project	Provide 3 mostly correct algorithms for project.	Provide 3 accurate, well written algorithms			/20
Section 3 – Implementing Software Solutions							
User Documentation		Provides minimal information for the user.	Provides a well-designed, informative user manual	Provides an excellently planned out and explicitly detailed user manual			/10
Lexical analysis		Provide minimal information about Lexical analysis	Provide a well- designed, informative lexical analysis with correct tokens for your lexicons.	Provide a excellently planned, with explicit detailed lexical analysis with correct tokens for your lexicons.			/15
Section 4 – Testing and Evaluating Software Solution							
Testing		Engages in some testing of software solution	Tests software solution using a number of testing methods. Clearly documents results	Engages in a wide range of testing processes. Explicitly and clearly records findings, analyzing results.			/10
Evaluation		Evaluates software solution.	Evaluates software solution using identified criteria for success.	Evaluates software solution using identified criteria for success. Justifies any changes or variations to solution.			/5
Total Mark:							/100

Feedback

.....

.....

.....

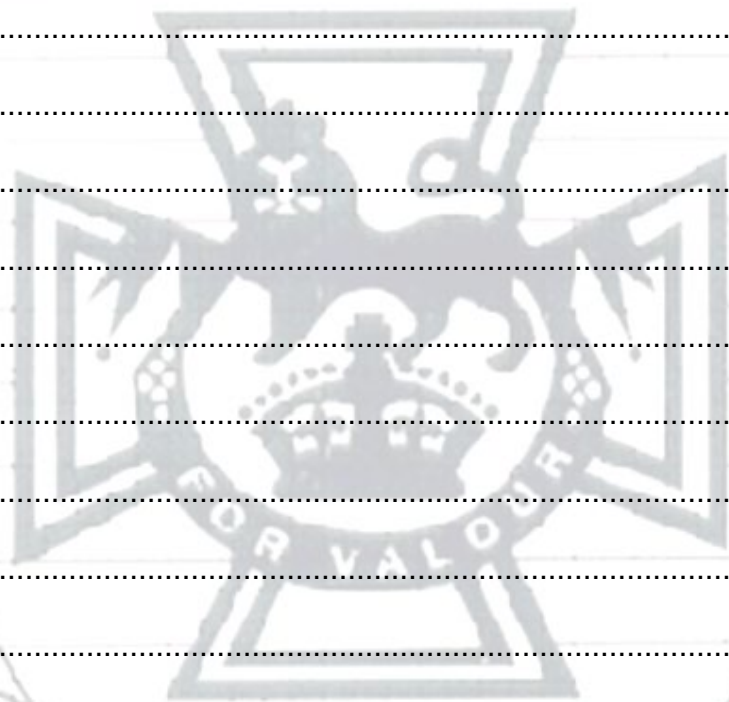
.....

.....

.....

.....

John Edmondson
High School



2005

VIRTUS ET INTEGRITAS

John Edmondson
High School



2005

VIRTUS ET INTEGRITAS