Assignment Guidance and Front Sheet

This front sheet for assignments is designed to contain the brief, the submission instructions, and the actual student submission for any WMG assignment. As a result the sheet is completed by several people over time, and is therefore split up into sections explaining who completes what information and when. Yellow highlighted text indicates examples or further explanation of what is requested, and the highlight and instructions should be removed as you populate 'your' section.

This sheet is only to be used for components of assessment worth more than 3 CATS (e.g. for a 15 credit module, weighted more than 20%; or for a 10 credit module, weighted more than 30%).

To be completed by the student(s) prior to final submission:

Your actual submission should be written at the end of this cover sheet file, or attached with the cover sheet at the front if drafted in a separate file, program or application.

Student ID or IDs for group work	**e.g. 1234567**

To be completed (highlighted parts only) by the programme administration after approval and prior to issuing of the assessment; to be consulted by the student(s) so that you know how and when to submit:

Date set	24/02/2023
Submission date (excluding extensions)	28/04/2023
Submission guidance	Submit to Tabula
Marks return date (excluding extensions)	26/05/2023
Late submission policy	If work is submitted late, penalties will be applied at the rate of **5 marks per University working day** after the due date, up to a **maximum of 10 working days** late. After this period the mark for the work will be reduced to 0 (which is the maximum penalty). "Late" means **after the submission deadline time as well as the date** – work submitted after the given time even on the same day is counted as 1 day late. For **Postgraduate** students only, who started their **current course before 1 August 2019**, the daily penalty is **3 marks** rather than 5.
Resubmission policy	If you fail this assignment or module, please be aware that the University allows students to remedy such failure (within certain limits). Decisions to authorise such resubmissions are made by Exam Boards. Normally these will

	be issued at specific times of the year, depending on your programme of study. More information can be found from your programme office if you are concerned.

To be <u>completed</u> by the <u>module owner/tutor</u> prior to approval and issuing of the assessment; to be <u>consulted</u> by the <u>student(s)</u> so that you understand the assignment brief, its context within the module, and any specific criteria and advice from the tutor:

Module title & code	WM245 Programming Languages for Cyber Security
Module owner	
Module tutor	Dr Nikki Williams
Assessment type	Cyber Tool Report
Weighting of mark	60%

Assessment brief
Please see below

Word count	Described below
Module learning outcomes (numbered)	1. Compare different programming paradigms used to create software. 2. Reflect on how software vulnerabilities can be minimised during software creation. 3. Incorporate security features in small-scale programs. 4. Develop small-scale programs that employ the idioms of a programming paradigm in a conventional manner.
Learning outcomes assessed in this assessment (numbered)	1. Compare different programming paradigms used to create software. 3. Incorporate security features in small-scale programs. 4. Develop small-scale programs that employ the idioms of a programming paradigm in a conventional manner.
Marking guidelines	Generally indicated within specification
Academic guidance resources	All queries to be directed to the tutor's email, with responses posted via moodle or workshop sessions.

1 Context

During this module we are exploring different programming languages, and how your decisions, for example in terms of language choice, paradigm choice, and library choice can affect the security of the resulting program.

This assessment is your opportunity to demonstrate how you put this knowledge into practice, by developing a cyber tool.

2 The task

At a high level, you will create a cyber security tool through which you will showcase your awareness and understanding of programming languages, programming paradigms and the cyber security implications of design decisions. Your assignment will take the form of creating the tool, describing a cyber security tool within the report, and a video showing the tool working.

- The language and paradigm associated with the development and implementation of the tool should be appropriate for that tool.

It is expected that the tool will have a clear cyber security related purpose, and have a clearly defined scope. If you have any questions about the scale of your proposed cyber security tool you should speak to Dr Nikki Williams by 14th April. The cyber security tool provides a focus for your report, where you will demonstrate your knowledge and understanding of a number of programing related concepts.

You have relatively free rein to design and build your tool. This assignment is for you to demonstrate that you can select and apply a suitable language (C, Python, C#, Java etc) and a suitable paradigm (OOP, event driven, etc) for the tool (or component within the tool). The rationale behind your selection of language and paradigm will be presented in a report.

Code reuse is acceptable. Where it occurs, it must be justified in the report and properly attributed to the original author.

Code must be fully commented. Use Harvard citations within your comments where you are reusing material that originated elsewhere. The full reference for your comment citations should appear in the References section of your report.

Every learner should submit a 5-minute (max length) video of their tool working. This should step through the various stages of the tool. It is acceptable to include some elements where the output has been created in advance but the code should be shown to be working. In the video your voice must be heard and your face seen – this will be reviewed to help determine the mark for the cyber tool. Suitable software for video creation is MSTeams, but you are welcome to use alternatives.

The **word count** for this assessment is **1,500 words** +/-10%.

3 Deliverables

1) The source code for the tool developed. This should be submitted as a separate file to the report.
2) A 5-minute (max) video of your tool working.
3) A 1,500 word report (+/-10%) to include the following:
    a. An introduction providing factual information about what the tool is, the purpose of the tool you have designed, who the target audience is, and the language(s) used an paradigms(s) used.

b. A discussion of decisions you made, justifying your choices, and exploring the alternative options you had. Make sure you cover at least two distinct design decisions made during the development process. Your choice of decisions to discuss should allow you to demonstrate your knowledge of a variety of topics (so do not pick two decisions of a similar type), and to demonstrate your understand how the decisions you made could impact the end program. It is possible that towards the end of the assignment, you may have decided in retrospect that your initial approach was not the most suitable. In this case, you should discuss how you would modify your approach if you were to complete the task again.
c. A discussion of how you developed your program with cyber security in mind – this is where you can highlight security features included, the security benefits afforded due to the language or paradigm choices you have made, or specific choices of how to implement a particular concept, or the choice of more secure methods or functions. Make sure you cover at least two distinct cyber security related features or decisions, and these should not duplicate the discussion provided for point b.

The report should include references to support the points being made about the programming languages, paradigms, and concepts. References should be in Harvard referencing format.

4 Style and substance

Notes on style and substance:

Style represents the effort taken by the author to make things accessible to the reader; substance is the concept being expressed. Poor style will hide good substance. Good style cannot hide poor substance.
Style includes (but is not limited to):

- physical layout, typography, use of white space, pagination, headers and footers, consistency, colour,
- logical layout, structure via (sub-)sections, section numbering, section headings, organisation of sentences and paragraphs, sequencing of material,
- language, lucidity, economy (padding / waffle removal), appropriate assumptions,
- supporting material as necessary (diagrams, tables),
- conventions, referencing

Substance is the collection of cyber-security and programming concepts you are seeking to communicate to the reader. These must be correct, relevant and necessary to the problem being addressed.

5 Miscellaneous Important Constraints

a) All activity must be conducted legally and ethically.
b) Use conventional Harvard academic references.
c) You are advised to use diagrams wherever they help your explanation. Original sources of diagrams must be referenced where they are not of your own creation.
d) The overall word limit is 1,500 words +/-10%.
e) The text visible on the printed page must be consistent with the text accessible to pdf document text analysis tools. Inconsistency will be treated as academic misconduct.
f) Your name should not appear on any page.
g) This is an individual assignment.

Criteria	Weighting	Criteria				
		Excellent 1st (>70%)	**Very Good 2:1 (60-69%)**	**Good 2:2 (50-59%)**	**Satisfactory 3rd (40-49%)**	**Poor Minor Fail (0-39%)**
Knowledge of programming languages, paradigms and concepts applied to the tool development, including options and justifications for decisions made.	40	Demonstrated a comprehensive knowledge and understanding of the subject and application to the topic.	Demonstrated an extensive knowledge and understanding of the subject and the application to the topic.	Demonstrated a good knowledge and understanding of the subject with some application to the topic.	Demonstrated a satisfactory knowledge and understanding of the subject with little application to the topic.	Below satisfactory attainment.
Discussion of how cyber security was or would be taken into consideration through the tool development and deployment,	30	Demonstrated a comprehensive knowledge and understanding of the subject with detailed specific security examples to illustrate points made.	Demonstrated an extensive knowledge and understanding of cyber security with specific security examples to illustrate points made.	Demonstrated a good knowledge and understanding of cyber security with some relevance to the tool developed.	Demonstrated a satisfactory knowledge and understanding of cyber security with limited relevance to the tool developed.	Below satisfactory attainment.
The cyber security tool – performs the expected task(s), demonstrates programming skills, and is suitably commented.	20	Very high-quality work demonstrating excellent knowledge and understanding, accuracy, relevance, and tackling a complex cyber task.	High quality work demonstrating good knowledge and understanding, accuracy, relevance, and tackling a cyber task.	Competent work, demonstrating reasonable knowledge and understanding, accuracy, relevance, and tackling a cyber task.	Work of limited quality, demonstrating some relevant knowledge and understanding, and limited application to a cyber task.	Below satisfactory attainment.

Criteria	Weighting	Criteria				
		Excellent 1st (>70%)	**Very Good 2:1 (60-69%)**	**Good 2:2 (50-59%)**	**Satisfactory 3rd (40-49%)**	**Poor Minor Fail (0-39%)**
Presentation of the code and report is clear, logically structured, and includes information to aid understanding.	10	The work is extremely well presented and shows the topic exploration with an extremely high degree of clarity. Text and graphics are clear and unnecessary jargon is avoided. Accurately uses Harvard referencing style. Accurate spelling and grammar.	The work is mostly well presented and shows the topic exploration with a degree of clarity. Text and graphics are clear though some unnecessary jargon is used. Harvard referencing style is used but lacks some accuracy. Has some spelling and grammar errors.	The work is adequately presented and shows the topic. Some of the text and graphics are unclear, and some inappropriate jargon is used. Harvard referencing style is used but lacks accuracy. Has spelling and grammar errors.	The work is poorly presented and does not clearly show the topic. Text and/or graphics are unclear, and a significant amount of inappropriate jargon is used. Harvard referencing style is used inaccurately and there are spelling and grammar errors.	Below satisfactory attainment.