

# Development Team Project: Design Document

**Due:** Monday, 27 March 2023, 11:55 PM

In this unit, the focus is on the first objective/deliverable, the design proposal report. This assignment has two component parts, the team submission, and your individual peer assessment. All components must be submitted by end of unit 3. This accounts for 30% of your final module mark.

**Only one submission is required from each team. Nominate one member of your team to submit on the team's behalf. The word count is 600 words.**

## Full Brief

For this assessment, you are advised to position yourself and your team as IT Software Consultants and Developers.

You are required to develop an application that provides a **secure repository** for an organisation with domain-specific requirements. A domain refers to a group of users with similar application and hardware requirements. A domain may be characterised by having:

- periodic pressure on resources (CPU, storage, or the network).
- interactive response requirements between request and reply.
- substantial data download requirements.

Such requirements place specific demands on a system's operations, and this influences the way that it should be operated and managed.

Domains that you should consider for the purpose of this assignment include (see the Unit reading list for the appropriate links):

- the International Space Station (NASA, 2007).
- the Dutch Police Internet Forensics (Government of the Netherlands, n.d.).
- the Hadron Collider at CERN (The Computer Security Team, 2020).

You can read more about the computer requirements of these systems using the references given. You should choose a single domain to focus your development on, and your system should be tailored to the requirements of this domain. In all domains, a user will need to be able to upload, download and share data.

## Application requirements:

The agreed criteria for successful development are:

- The solution should ensure that all privacy and security regulations are met, including those specified by the General Data Protection Regulation (GDPR) (ICO, n.d.).
- Mechanisms should be deployed to minimise the attack surface of the solution.
- The organisation would like to see a monolithic version of the application because they have concerns about security, scalability and supportability, and they may wish to extend use to partner organisations on a worldwide basis.

**Development conditions:** Open-source libraries must be used in your development to ensure that the code may be run for its assessment.

Your full brief is to:

1. Create a comprehensive design proposal report, describing how you will meet the requirements, and the design and implementation of the solution.
2. You should create a working prototype of your design based on a monolithic approach.

### **Part 1: The Design Proposal Document**

Your team is expected to prepare and deliver a design proposal report of your intended development work for the organisation. Note that the associated grading criteria are highlighted in the requirements below, to be reviewed alongside the criteria grid (Module Resources).

- Your report should detail the system requirements and assumptions (such as local or remote access, magnitude of storage required, CPU capacity, etc), design decisions (such as encryption algorithms, approach to data storage, use of databases, use of frameworks) and approaches that you have adopted to create your secure software solution. Build up a rationale where appropriate supported by literature (**Knowledge and Understanding weighted at 25%**).
- It should list security challenges you have identified/ expect to encounter (such as those in the STRIDE model and/or those forming the OWASP principles). Highlight (briefly) what paradigm(s), pattern(s), theories and practices you intend to utilise on this project to address security, technical and business challenges. Justify your approaches supported by literature. (**Application and Understanding weighted at 25%**).
- You should produce a graphical design, based on UML, that illustrates your approach via a number of views which should include, at a minimum, sequence diagrams, class diagrams, and activity diagrams. AND
- You should also state any tools, libraries and models that you will use in your solution, justified by academic research. AND
- Ensure that your system considers the functionality that similar applications provide, such as user registration, user roles, and CRUD functionality: which should you include? You will need to explain your reasoning and justifications for any omissions during your demonstrations (**Criticality weighted at 25%**).

**Presentation and Structure of your work (weighted at 25%)** includes spelling, style, evidence of proofreading, correct use (and format) of citations and references.

It is recommended you use tables and bullet-point lists to stay within the word count limit.

Checklist for the assignment:

- Bulleted list of system requirements and assumptions, design decisions and approaches you will use based on background information and additional academic research (ensure you include any references you have used);
- Bulleted list of security risks/ vulnerabilities you have identified, including reference to frameworks used (e.g. STRIDE, OWASP) with potential mitigations and references;
- UML design of solution using multiple diagrams (e.g. class, sequence, activity) with references;
- Bulleted list of tools (including development and test tools), libraries and models you will use;
- Remember to use a spell checker and proofread your work before submission.

- **You should get your design outline reviewed and approved by your tutor BEFORE you submit the final version in this Unit.** You are invited to arrange a meeting between your team and the tutor to get feedback.

**Please note that appendices** should not be used to extend the core report as reports should stand alone, complete and concise, without the appendices. They should really only be used if required, and only for supplementary and/ or supporting information. One key part of the exercises in this module is the need to be able to express ideas succinctly, concisely and with necessary brevity.

### **Learning Outcomes**

- Identify and manage security risks as part of a software development project.
- Critically analyse development problems and determine appropriate methodologies, tools and techniques (including program design and development) to solve them.
- Systematically develop and implement the skills required to be effective member of a development team in a virtual professional environment, adopting real-life perspectives on team roles and organisation.