# Design Document (approx. 1000 words)

## Overview and Class Diagram (100-150 words)

Provide a brief *introduction* to the system and attach a **readable** picture of the *Class Diagram* that describes your system design.

## System Entities: Classes, Abstract Classes, and Interfaces (50-75 words per entity)

### Class: Entity1

Brief description of the purpose of the class, including attributes and most important methods identified from the system requirements.

### Class: Entity2

...

### Class: EntityN

### Interface: InterfaceXYZ

A brief description of the interface's purpose and the specified design contract. Explain why and how, e.g., *Entity1* implements the *InterfaceXYZ*.

## OOP Principles Applied (150-200 words)

Explain the Object-Oriented Programming (OOP) principles applied in the design and justify how they contribute to building a *modular* and *maintainable* software system.

### Encapsulation

Explanation of how encapsulation is achieved in the design.

### Inheritance

Explanation of how inheritance is utilised in the design (i.e., the classes that use it).

### Polymorphism

Explanation of how polymorphism is implemented in the design (i.e., the classes that use it).

## Relationships (150-200 words)

### Relationship: Association, Composition and Aggregation

Describe the associations between, e.g., objects of *Entity1* and *Entity2*, including relevant multiplicity.

Describe any composition or aggregation relationships between entities if present.

## Reflection on Design and Implementation (150-200 words)

Reflect on the design's alignment with system requirements and the actual system implementation:

* How well does the design meet the specified system requirements?
* How well does the implementation align with the design?
* Any improvements or modifications suggested based on the reflection?