System Design Document

IS / HCC 636: Systems Analysis and Design
Department of Information Systems
University of Maryland, Baltimore County

Introduction

Generic Corporation has approved your Problem Analysis and Requirements Document and is interested in pursuing the project into system design. You are now tasked with creating a design and implementation plan for the software requirements that address your client's problem. Your system design document will consist of some of the following elements.

1. Introduction
2. Data Flow Diagrams
3. Entity Relationship Diagrams
4. Use Case Analysis (Group Choice)
5. Sequence Diagram (Group Choice)
6. Class Diagram (Group Choice)
7. State Diagram (Group Choice)
8. Discussion of Coupling and Cohesion
9. User Interface Mockup (Group Choice)
10. Interface Structure Diagram
11. Structure Chart or Control Flow Diagram (Group Choice)
12. Updated Project Plan (Bonus)

Each aspect of this project is explained in the following sections of this document, along with *recommended word counts for deliverables*. Once again, everything is negotiable because that's simply how things work in the real world. If you have a strong reason to emphasize one aspect of this report at the expense of another, please contact your instructor. Risk taking and negotiation are encouraged, but please understand that the descriptions below outline

reasonable deliverables for this assignment. Unjustifiable deviations from the recommendations in each section may result in loss of all points associated with that section. In particular, anyone reading your report must be able to clearly identify answers to the question at the end of each section.

Six aspects of this report are labeled as "Group Choice," and your group must do two of these. If your group chooses to do more, then the two best sections will count towards your final grade for this project and any mistakes made in the remaining completed sections will not count against your grade. The Updated Project Plan is entirely optional. Groups completing this section will receive up to 5 bonus points.

1 Introduction

Every software artifact has an appropriate context. You must set the context for your system design document by providing a brief introduction. The introduction should include a discussion of the purpose of the system design document, the structure of the system design document, and a summary of the most important overall lessons learned during the process of design and conclusions that will affect implementation. What are the goals and outcomes of your system design document?

Introduction: 400–700 words

2 Data Flow Diagrams

Data Flow Diagrams depict how a software system uses and produces data. In particular, they demonstrate which processes need access to what data and show how that data will be accessed. Your group must create a Level 0 DFD along with three child diagrams. Your three child diagrams can be children of any parent. Thus, you could choose to create three child diagrams where each is a child of a different process in your Level 0 DFD. You could also choose to create three child diagrams in a hierarchy where each child diagram explores a process in the previous child diagram. You could also choose to do anything in between, so long as you have three complete child diagrams. Each child diagram should be accompanied with some explanatory text describing how this DFD satisfies your software requirements and highlighting potential implementation concerns. What data does your system require? When and where does your system require this data? What data does your system produce? Why?

Data Flow Diagrams: 4 diagrams and 1,250–1,500 words.

3 Entity Relationship Diagrams

Entity Relationship Diagrams represent the structure of data in a software system, including the relationships between data elements. Your group must create a single ERD representing all data elements present on your primitive DFDs. This diagram must be accompanied by explanatory text detailing how it relates to both the DFDs (i.e., justifying entities, attributes, and relationships based on the processes) or the software requirements (i.e., justifying the same based on stakeholder needs). You may use any valid notation for your ERD, but it must be legible. This may require spreading the ERD across multiple pages. What is the structure of the data used in your system?

Entity Relationship Diagrams: 1 diagram and 1,000–1,500 words.

Use Case Analysis (Group Choice)

Use Cases provide a structured narrative of a system behavior from the viewpoint of a particular stakeholder or user. If you choose, your group can create a use case diagram depicting the complete interactions that three users might have with your system. For each of the three users included in your diagram, you must depict all of their interactions with the system. You must then describe how an interested, non-expert stakeholder should interpret this diagram. What would a sample interaction with this system look like?

Although this is listed as a "Group Choice" option, your group may not choose to do this section if they previously created a Use Case analysis in your Problem Analysis and Requirements Document.

Use Case Analysis: 1 diagram and 400-700 words.

4 Sequence Diagram (Group Choice)

Sequence diagrams detail the precise ordering of events for a particular use case. If your group chooses to do this, you must identify the most complex interaction in your system and diagram it. This diagram must include some explanatory text describing how this sequence relates to the Use Case diagram. Which uses of your system involve complex sequences of action? How does your design make these action sequences understandable?

Your group cannot choose to create a sequence diagram without first creating a Use Case Analysis.

Sequence Diagram: 1 diagram and 400–700 words.

5 Class Diagram (Group Choice)

Class diagrams are tools used in Object Oriented software systems to show how related data attributes and methods are encapsulated into cohesive objects. Class diagrams also show the relationship between classes, including inheritance relationships of various types. Your group may choose to create a class diagram for all relevant classes in your software system. This diagram must be accompanied by a short paragraph highlighting non-trivial classes and relationships. Which classes will you implement to support the stakeholders' requirements? Can you justify this design as supporting cohesion and limiting coupling?

Class Diagram: 1 diagram and 400–700 words.

6 State Diagram (Group Choice)

State diagrams, also called state machine diagrams, highlight the lifecycle of information or data encapsulated in an object over that object's lifecycle. They are particularly valuable for objects that support asynchronous or concurrent access. Your group may choose to create a state diagram for the objects in your class that deserve additional explication or otherwise fit the intended use of state diagrams. This diagram must be accompanied by textual description of the diagram, including justification for its creation. What objects involve particularly complicated data access over their life cycle? How does your design mitigate these complications?

Your group cannot choose to create a state diagram without first creating a class diagram.
State Diagram: 1 diagram and 700–1,000 words.

7 Discussion of Coupling and Cohesion

Coupling and cohesion are critical aspects of any software system. Ensuring that related data and functions are grouped appropriately and avoiding unnecessary relationships between data and functions are two of the most important goals in building maintainable, successful software systems. Your group must directly address these concerns by describing the approach you've taken to support cohesion and limit coupling. What aspects of your system are coupled together and why? What steps have you taken to ensure related data or functions are grouped together?

Discussion of Coupling and Cohesion: 700–1,000 words.

8 User Interface Mockup (Group Choice)

User interface mockups or wireframes are common design artifacts that help stakeholders evaluate all or part of a system under construction. Your group must create mockups of two separate but related screens for your systems. You may use digital mockup tools or create them on paper and provide scans or images of the interface. Your group must describe the goals of the mockup, including particular elements of the design where you plan to solicit stakeholder feedback. What presentation and layout of this system will allow users to most easily and effectively use the system to achieve their goals?

User Interface Mockup: 2 diagrams and 600-800 words.

9 Interface Structure Diagram

Interface structure diagrams pictorially describe all user interactions with the system. They support stakeholder discussion of the interface with minimal investment in prototypes or UI development. They can be used to ensure that there are no dead ends in the interface and that all interactions have a consistent interface. Your group must create an interface structure diagram and describe a common use of the system while referencing this diagram. How are you informing the user of their options at each stage of the interface? Do they know the consequences of leaving? Have you avoided all dead ends?

Interface Structure Diagram: 1 diagram and 1,250–1,500 words.

10 Structure Chart or Control Flow Diagram (Group Choice)

Structure charts are used to show the thread of control for part or all of a software system as control passes from one module to another. They emphasize the relationship between processes and can easily display the relationship between modules as a part of a higher-level software architecture.

Control flow diagrams are used to show which conditions lead to the passing of control from one software module to another. They are similar to structure charts, but they are intended to be used at a lower level. For example, they are not great at showing the software architecture, but they may represent the actual control flow as implemented in source code.

Your group may choose to create either a structure chart or a control flow diagram. You may not implement both, but you may simply choose not to select this option as one of your two group choice options. If you do choose to implement one of these, you must provide both the diagram and a textual explanation of why it is valuable for your project. What thread of control

concerns prompted you to create either a structure chart of control flow diagram? How does your chosen diagram help you address this concern?

Structure Chart or Control Flow Diagram: 1 diagram and 400–700 words.

Updated Project Plan (Bonus)

Your group may optionally include an updated project plan, identifying remaining tasks and their dependencies, estimating a schedule, and including a staffing summary. Generic Corporation's executives recognize that this project plan will be updated throughout the project, and they expect to see a more detailed and accurate plan than provided in the software requirements document. Where does the project stand after the design has been conlcluded? What does an updated breakdown of remaining tasks and an associated schedule look like?

Project Plan: 600–800 words

Grading Summary

This project will be graded on a score from 0 to 100. Each section of the document will be worth points as described in Table 1.

Table 1: System Requirements Document Grading Breakdown

Section	**Point Value**
Introduction	10 points
Data Flow Diagrams	20 points
Entity Relationship Diagrams	10 points
Group Choice #1	10 points
Group Choice #2	10 points
Discussion of Coupling and Cohesion (Non-Prototyping Option)	10 points
Interface Structure Diagram (Non-Prototyping Option)	10 points
Updated Project Plan (Bonus)	5 bonus points
Total	**80**, with 5 bonus points possible

Please note that the values in Table 1 do not add up to 100 points. The remaining 20 points are reserved for the quality of writing and cohesiveness of the document. Communication is absolutely critical in systems analysis and design. It is at least as important as any of the individual sections. *Your document must read as if it were written with a single, clear voice.* Grammar errors, missing references, non sequiturs, and other communications errors will definitely affect this portion of your grade. Where possible, communications errors will not affect the individual sections.