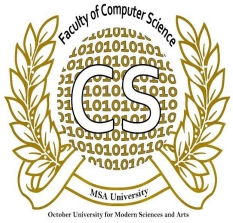

Software Design Document for project ————

Hossam Mohamed,Adham Ahmed,Kirolos Ehab ,Karim , Menna Ramzy
Supervised by: Dr. ayman ezzat

November 16, 2021

1 Introduction

1.1 Purpose

this software design document describes the architecture and system design of a simple school management system

1.2 Scope

this software will provide various benefits for the user it will make it easier to access Increases Productivity,Best Student-Teacher Collaboration Saves Natural Resources,Access from Anywhere, Increase in Student Enrollment Ratio,Transparency with Parents Increases,Reduction in the Cost of Communication and Reduces Workload.

2 System Overview

Give a general description of the functionality, context and design of your project. Provide any background information if necessary.

3 System Architecture

3.1 Architectural Design

Develop a modular program structure and explain the relationships between the modules to achieve the complete functionality of the system. This is a high level overview of how responsibilities of the system were partitioned and then assigned to subsystems. Identify each high level subsystem and the roles or responsibilities assigned to it. Describe how these subsystems collaborate with each other in order

to achieve the desired functionality. Don t go into too much detail about the individual subsystems. The main purpose is to gain a general understanding of how and why the system was decomposed, and how the individual parts work together. Provide a diagram showing the major subsystems and data repositories and their interconnections. Describe the diagram if required.

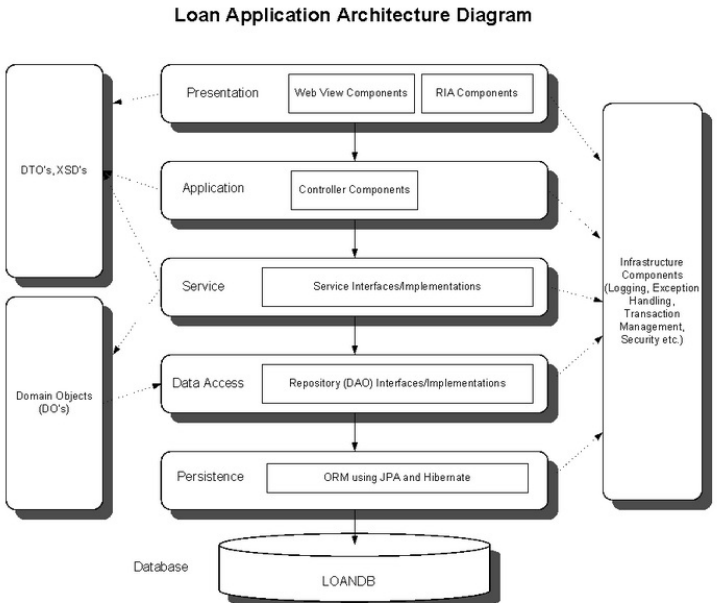

Figure 1: Architectural Design

3.2 Decomposition Description

Provide a decomposition of the subsystems in the architectural design. Supplement with text as needed. You may choose to give a functional description or an object oriented description. For a functional description, put top level data flow diagram (DFD) and structural decomposition diagrams. For an OO description, put subsystem model, object diagrams, generalization hierarchy diagram(s) (if any), aggregation hierarchy diagram(s) (if any), interface specifications, and sequence diagrams here.

3.3 Design Rationale

Discuss the rationale for selecting the architecture described in 3.1 including critical issues and trade/offs that were considered. You may discuss other architectures that were considered, provided that you explain why you didn't choose them.

4 Data Design

4.1 Data Description

Explain how the information domain of your system is transformed into data structures. Describe how the major data or system entities are stored, processed and organized. List any databases or data storage items.

4.2 Data Dictionary

Alphabetically list the system entities or major data along with their types and descriptions. If you provided a functional description in Section 3.2, list all the functions and function parameters. If you provided an OO description, list the objects and its attributes, methods and method parameters.

5 Component Design

In this section, we take a closer look at what each component does in a more systematic way. If you gave a functional description in section 3.2, provide a summary of your algorithm for each function listed in 3.2 in procedural description language (PDL) or pseudo-code. If you gave an OO description, summarize each object member function for all the objects listed in 3.2 in PDL or pseudo code. Describe any local data when necessary.

6 Humnan Interface Design

6.1 Overview of User Interface

Describe the functionality of the system from the user s perspective. Explain how the user will be able to use your system to complete al the expected features and the feedback information that will be displayed for the user.

6.2 Screen Images

Display screenshots showing the interface from the users perspective. These can be hand drawn or you can use an automated drawing tool. Just make them as accurate as possible.

6.3 Screen Objects and Actions

A discussion of screen objects and actions associated with those objects.

7 APPENDICES

This section is optional. Appendices may be included, either directly or by reference, to provide supporting details that could aid in the understanding of the Software Design Document.

8 References

References