Software Requirements Specification

For

Learning Management System

Jacob Rodriguez

Ahmet K Duran

Triet Minh Huynh

Murtaza Oklu

Version: 1.0 Date: April 24, 2018

Table of Contents

Table of Contents 2

Revision History 2

1. Introduction 3

1.1 Purpose 3

1.2 Glossary 3

1.3 Intended Audience and Document Overview 3

1.4 Product Scope 3

1.5 References 4

2. Overall Description 4

2.1 Product Perspective 4

2.2 Product Functions 4

2.3 User Classes and Characteristics 4

2.4 Operating Environment 5

2.5 Design and Implementation Constraints 5

2.6 User Documentation 6

2.7 Assumptions 6

3. External Interface Requirements 6

3.1 User Interfaces 6

3.2 Hardware Interfaces 6

3.3 Communications Interfaces 6

3.4 Functional Requirements 6

3.5 Other Nonfunctional Requirements 6

4. System Designs 7

4.1 Data Flow Diagram 7

4.2 UML Diagram 7

4.2.1 Use Case Diagram 7

4.2.2 Sequence Diagram 9

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For changes** | **Version** |
|  | | | |
| Jacob Rodriguez | 4/24 | First Revisions | Version 1 |
|  |  |  |  |

1. Introduction
   1. Purpose

The Purpose of this document is to present a detailed description of the Learning Management System. It will explain the purpose of the system and features of the system. It will also explain the interfaces of the system and what the system will do. It will explain the constraints it will operate under and how the system will react.

* 1. Glossary

|  |  |
| --- | --- |
| Term | Definition |
| Database | Collection of all information for this system. |
| Reader | Anyone visiting the site to read this article. |
| Software Requirement Specification | A document that completely describes all the functions the proposed system will do and any constraints. |
| Stakeholder | A person interested in the project who isn’t a developer. |
| User | Anyone who contributes to the system. |
| Administrator | Administrator who is responsible for the system. |
| GPA | Number representing the average value of the accumulated final grades earned in courses over time. |

* 1. Intended Audience and Document Overview

This document is intended for the following people:

* **Team Leader:** Team leaders are those who lead the entire project.
* **Implementers or Coders:** These are the individuals who implement the design stated by the developers using programming languages.
* **Testers:** This is the set of individuals who test the developed system and estimate the performance of the system.
* **Documentation Writers:** These people are the writers that prepare the user manuals and other necessary documentations.
* **Users:** these are the people that are using the system.
  1. Product Scope

This software will be a Learning Management System for a client who needs to have a student information system build. This system will be designed to store and receive student’s partial information in the current semester and other basic information. Basic information should include student’s name, student’s ID, registered courses in the current semester, each exam’s score in one course, GPA calculation in the current semester.

More specifically, the system main goal has two types of accessing modes, administrator and student user. Student information management system is managed by the administrator. The job of the administrator is to insert update and monitor the whole process. The student user would only view details of the student and can’t perform any changes to the information.

* 1. References

Schach, Stephen R. *Object-Oriented and Classical Software Engineering*. 8th ed., Langara College, 2017.

1. Overall Description
   1. Product Perspective

This system will be familiar to Blackboard or Moodle, which are typical learning management systems. Generally speaking, learning management system deals with all kinds of student details, academic related reports, college details, course details, curriculum, batch details and other resource related details too. There are many similar programs that offer similar settings but every school is different, which has different highlights for their students.

* 1. Product Functions

The learning management system is a system that supports learning process of teaching and learning in the school. Student can view their courses, exam grades, and their GPA. Administrator can manage the system from a dashboard.

* 1. User Classes and Characteristics

Physical Actors:

* **Student:** He/ she would only view details of the student. They can’t make changes. Multiple students can connect to the system. It’s convenient to think that every student represents every user that logs into the system as a student.
* **Administrator:** The administrator is the one that manages the student information management system. It’s their job to insert update and monitor the whole process. It’s convenient to think that every administrator represents every user that has administrator login information to login into the system.

System Actors:

* **Learning Management System (LMS):** The software system stores and retrieves student’s name, student’s ID, registered courses in the current semester, each exam’s score in one course, GPA calculation in the current semester.

Below are the main goals of each of the actor list:

|  |  |
| --- | --- |
| LMS | Student |
| The student logs in with their student credentials. They are able to view their information such as exam’s scores, their registered courses, and their name. | |

|  |  |
| --- | --- |
| LMS | Administrator |
| The administrator logs in with their administrator credentials. They are able to view and update information of a student such as inputting their exam’s scores. | |

* 1. Operating Environment

The system will operate in the following environment for the student and the administrator:

* Apple Mac OS X
* Linux/Unix
* Microsoft Windows
  1. Design and Implementation Constraints

This system is created using Visual Basic .NET. The PC should have at least 4gb of RAM and CPU over 1.8 GHz to run the software with a good speed. Also, the program uses at least 130 gb of hard disk space to store the program libraries. For language support except from English language pack, there is also a Chinese, Czech, French, German, Italian, Japanese, Korean, Polish, Portuguese, Russian, Spanish and Turkish.

* 1. User Documentation
* A readme file to help the user with the installation of the software.
* A well-documented manual.
  1. Assumptions
* Administrator can edit a student’s partial information one at a time.

1. External Interface Requirements
   1. User Interfaces
   2. Hardware Interfaces

The user needs internet, a minimum CPU speed of 1.8 GHz with a dual core processor, 4 gb RAM and 130 gb of free hard disk space.

* 1. Communications Interfaces

In what way the communication is achieved is not important for the system and is therefore handled by the underlying operating systems.

* 1. Functional Requirements
* Students can view courses, exam scores, calculated GPA, and their name
* Administrators can modify course’s information.
* Administrators can posts student’s exam scores.
  1. Other Nonfunctional Requirements

Responsiveness: Less responsive time should be there so that the student and administrator should feel please wile using the system.

Availability: 24/7 availability should be there so the student can use it any time.

Scalability: the system should scale maximum number of users.

Security: Must be secure access of student’s confidential information. Administrators of the system will have full administrator rights.

Reliability: The system must be of good quality and consistently performs well.

1. System Designs

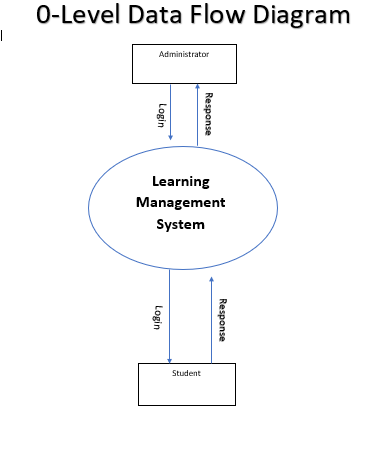
Design is an important representation of something is going to be created. Design is the best way to accurately translate a customer’s requirement into a product. Design creates a model that provide details about software data structure, architectural design, and interfaces.

There are two types of diagrams to represent the implementation of a system

* Data Flow Diagram
* UML Diagram
  1. Data Flow Diagram

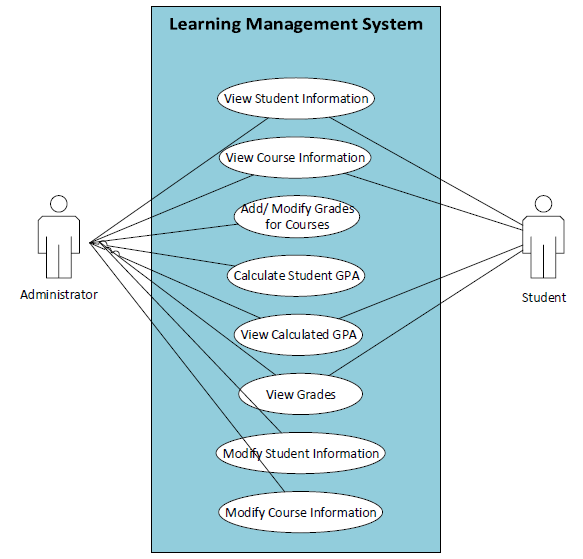
Data flow diagram is the preliminary step to create an overview of the system without going into great detail, which can be elaborated later.

**Level 0 Data Flow Diagram**

****

* 1. UML Diagram
     1. Use case Diagram

Use case diagram is a list of action or event steps defining the interactions between a role and a system to achieve a goal.



Administrator

* View Student Information: The administrator can view student information.
* View Course Information: The administrator can view course information.
* Add/ Modify Grades for Courses: The administrator can add grades for a student’s courses or modify the current grade.
* Calculate Student GPA: The administrator can calculate the student’s GPA by inputting grades in.
* View Calculated GPA: The administrator can view the calculated GPA of the student.
* View Grades: The administrator can view the grades of the student.
* Modify Student Information: The administrator can modify the student’s information if needed.
* Modify Course Information: The administrator can modify the course’s information.

Student

* View Student Information: The student can view their information.
* View Course Information: The student can view course information.
* View Calculated GPA: The student can view their calculated GPA.
* View Grades: The student can view their grades.
  + 1. Sequence Diagram