**Learning Management System**

**Contributors: Jabari Fowler, Rayan Jangda, Eldrin Preston, Armando Sanchez, Ismael Moreno**

**CS 3321**

**Professor Singh**

**12/8/2020**

## **1. Introduction**

This collaborative project aims to create a learning management system (L.M.S.) for the use of storing student information, gradebooks and accounts for university students.

## **1.1 Project Overview**

The aim of this project is to implement and recreate a platform like Blackboard or CANVAS which is a learning management system. The collaborators will make this product easy to access and easy to use.

## **1.2 Project Deliverables**

The Primary deliverables for the client is to provide a SPMP which is this document, a requirement analysis document, system design document, test manual and source code.

## **1.3 Evolution of the Software Project Management Plan**

This programming team went about developing the final software product using the waterfall life cycle model.

## **2.1 Process Model**

The programming team decided on the waterfall life cycle model as we wanted to go through this project in a linear fashion. This model fit the team’s criteria in the best manner. (Figure 1)

## **2.1.1 Project Planning**

Project planning includes description of project tasks, activities and functions, dependencies, resource requirements and a detailed schedule. This activity results in the software project management plan for the L.M.S. (Figure 2)

**2.1.2 Requirements Analysis**

The requirements analysis activity takes the problem statement and reviews it in terms of consistency, completeness, and feasibility. The main part of the requirements states that there are two accessing modes which are Administrator and User. (Figure 3 & 4)

## 

## **2.1.3 System Design Description (SDD)**

The purpose of the system design activity is to devise a system architecture that maps the analysis model to the chosen target environment. The major part of the system design phase is the design of subsystems. The system design activity also refines the use cases from the analysis model and describes in terms of interaction diagrams how the objects interact in each specific use case.

## **2.1.4 Analysis Review**

Review of software project management plan, requirements analysis, and design. The meetings will take place on Dec. 9 from 4:00 PM - 5:30 PM in Zoom meeting. The Analysis Review consists of a set of presentations given by members of the L.M.S. project.

## **2.1.5 Functional Prototype Demonstration**

This activity involves successful execution of a functional prototype of the L.M.S. using Netbean. The functional prototype of the L.M.S will be presented during the internal review Dec 9, 2020.

## **2.1.7 Object Design Phase**

The object design phase specifies the requirements established by the user requirements. New classes are added to the analysis object model if necessary by the system architecture. Attributes and methods for each object are fully added as necessary.

## **2.1.8 Implementation**

The focus of this activity is on coding the individual objects described in the object design document.

## **2.1.9 Unit Testing**

During unit testing, test suites are designed and executed for objects or collections of objects in each subsystem. Unit testing enables the individual subsystems to be tested independent from the status of the other subsystems. The result of this activity is part of the test process that describes how to operate the L.M.S. and how to interpret the input data.

## **2.1.10 System Integration**

During this activity an integration strategy is devised, that specifies the order in which the subsystems of the L.M.S. are integrated and tested with respect to the use cases defined in the analysis model.

## **2.1.11 System Testing**

**Structural Testing:** This activity tests the major data paths in the complete L.M.S.

**Functional Testing:** Tests the major functionality (use cases) with the complete L.M.S. The basis for the functional testing activity is the test manual which is revised according to the results of the system testing phase.

**2.1.12 Manual Integration**

During this activity, the project deliverables are revised. As a result, a complete set of documents consisting of the software project management plan, requirements analysis document, software design document, test manual and source code is made available on the project home page. The system documentation will also be printed on Github.

## **2.2 Organizational Structure**

The team decided to use the democartic style to make decisions smoother for the overall team. To back this decision of the democratic approach was the disregard for either of our ego in completion of the project.

The clients of the Learning Management System project are:

K. Singh

The project managers are:

Armando Daniel Sanchez, Eldrin Preston, Ismael Moreno, Jabari Fowler, Rayan Jangda

## **2.3 Meeting Times**

There is a weekly project meeting for each group. The initial project meeting times are:

|  |  |  |  |
| --- | --- | --- | --- |
| **Group** | **Day** | **Time** | **Location** |
| L.M.S. | Monday | 8:00 p.m. | Zoom |
| L.M.S. | Wednesday | 8:30 p.m | Zoom |
| L.M.S. | Thursday | 8:30 p.m. | Zoom |
| L.M.S. | Tuesday | 8:00 p.m. | Zoom |
| L.M.S. | Thursday | 8:30 p.m. | Zoom |
| L.M.S. | Tuesday | 8:30 p.m. | Zoom |
|  |  |  |  |

## 

## 

## 

## 

## 

## **2.4 Project Responsibilities Tables**

Table 1 describes the group assignments.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Team Assignments** | | | | | |
|  | **Design** | **Maintenance** |  |  |  |
| Requirements Analysis | Rayan Jangda | Armando Sanchez |  |  |  |
| System Design | Jabari Fowler | Ismael Moreno |  |  |  |
| Object Design | Eldrin Preston | Jabari Fowler |  |  |  |
| Implementation | Ismael Moreno | Rayan Jangda |  |  |  |
| Testing | Armando Sanchez | Eldrin Preston |  |  |  |

## 

## 

## **3.1 Assumptions, Dependencies and Constraints**

In this project each member is dependent on each contributor’s independent work to bring this project together as a whole. One of the constraints is that you can only access the software product locally.

## **3.2 Risks**

**Risk:** Portion of the data once deleted may be lost if not saved after applying the data.

**Contingency:** Add a database that will allow for storage of information to be held.

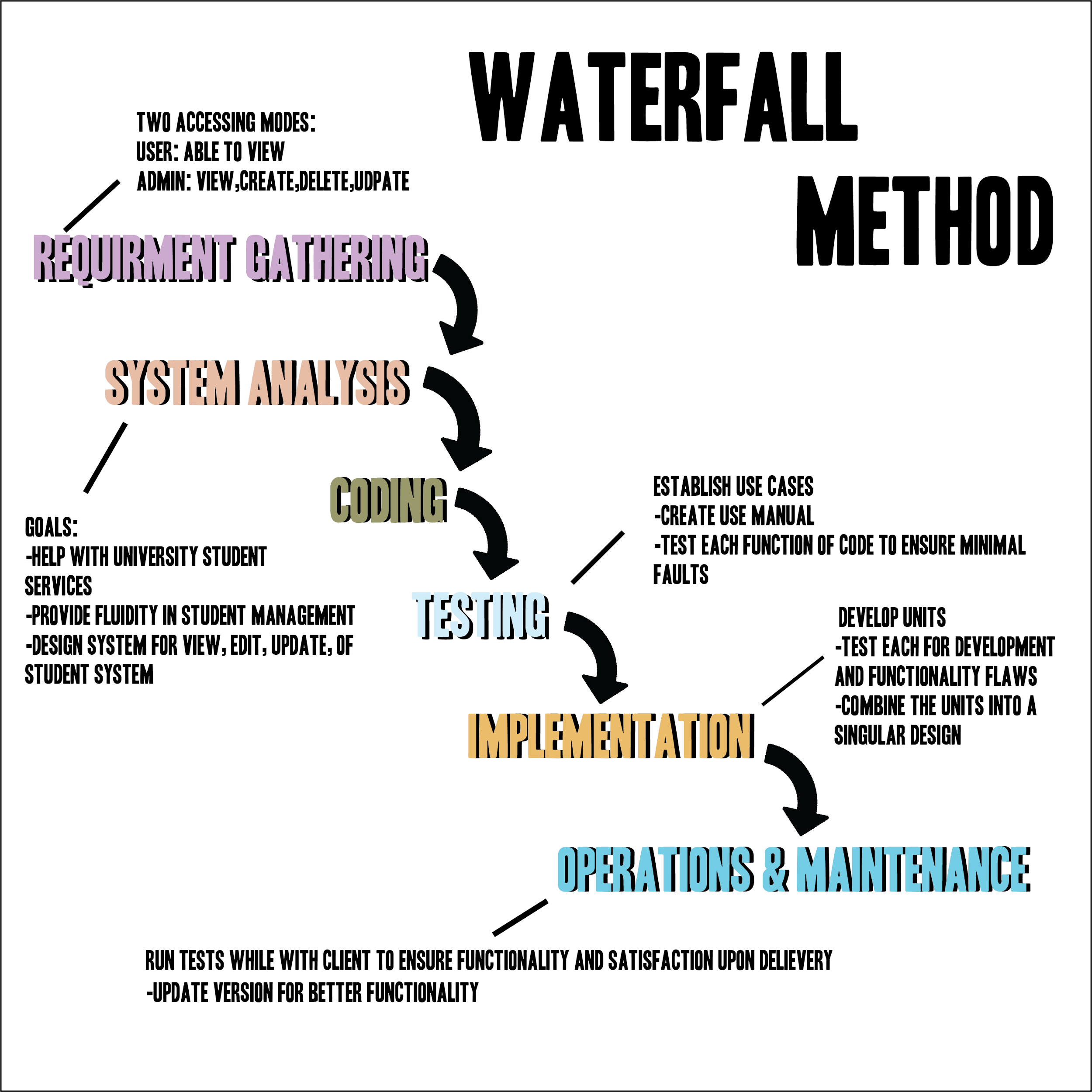
**Risk:**Passwords are not mandatory to have special characters in it .

**Contingency:** Make special characters necessary for password to be approved.

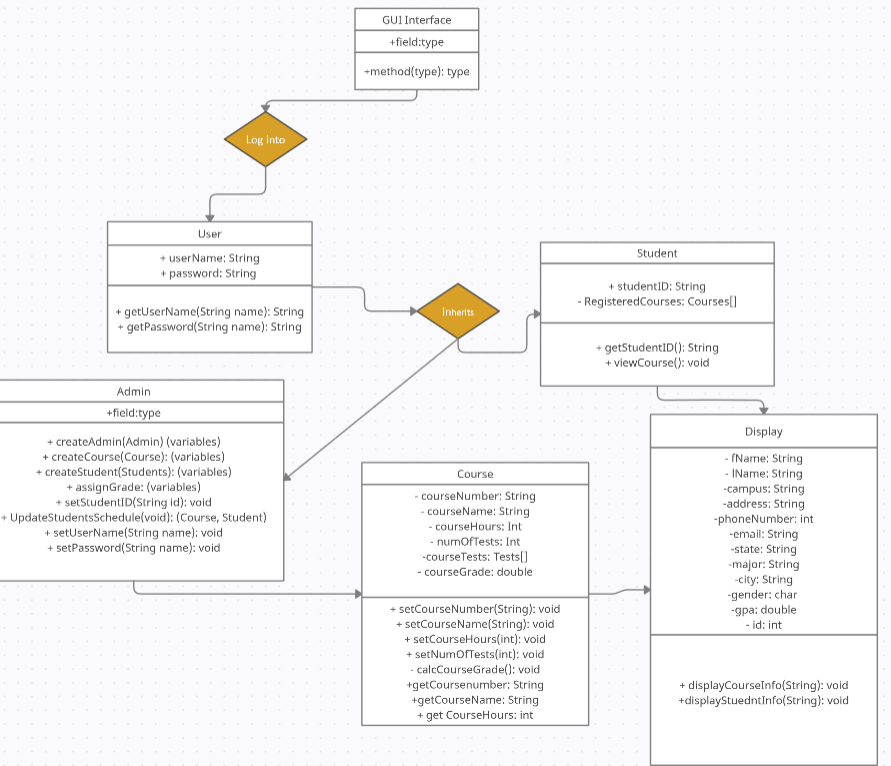
**Risk:** Some data may be entered incorrectly for the users data because administrator error.

**Contingency:** Allow users to edit more simple tasks such as personal information along with course work they have taken.

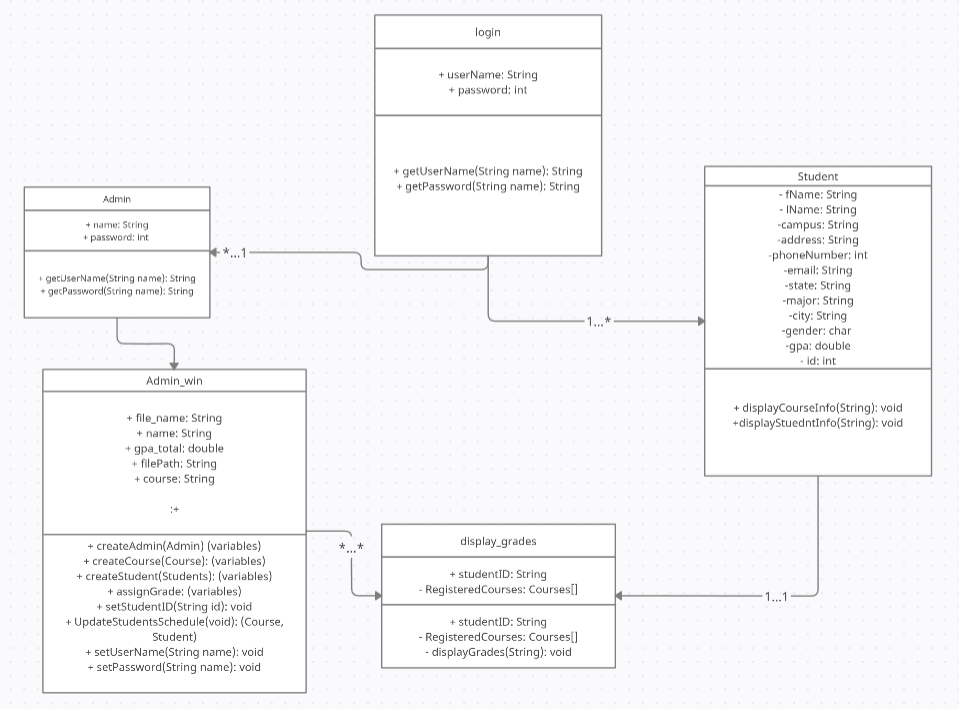
**Appendix**



*Figure 1 - Water Fall Life Cycle Model*



*Figure 2- UML Diagram*

**

*Figure 3- Class Diagram*

*Figure 4- Use Case*