*Process MeNtOR 3.o*

*Uni-SEP*

Course Management System

**Design Document**

|  |  |
| --- | --- |
| Version: |  |
| Print Date: |  |
| Release Date: | 4/6/2018 |
| Release State: |  |
| Approval State: |  |
| Approved by: |  |
| Prepared by: | Graeme Brabers, Chivaughn Charles and Wieke Harmsen |
| Reviewed by: |  |
| Path Name: |  |
| File Name: | SDDfinal.docx |
| Document No: |  |

# Document Change Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Authors** | **Summary of Changes** |
| 1 | 2/2/2018 | Chivaughn Charles | Component Diagram and Deployment Diagram |
| 2 | 4/2/2018 | Chivaughn Charles | Updated Component Diagram and Deployment Diagram, Introduction |
| 3 | 6/2/2018 | Wieke Harmsen, Graeme Brabers | Lay-out, general review, Class Diagram, |
|  |  |  |  |

# Document Sign-Off

|  |  |  |
| --- | --- | --- |
| **Name (Position)** | **Signature** | **Date** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Contents

1 Introduction 4

1.1 Overview 4

1.2 Resources - References 4

2 Major Design Decisions 5

3 Architecture 6

4 Detailed Class Diagrams 9

4.1 UML Class Diagrams 9

# Introduction

## Overview

This SDD describes the design of a course management system as a collection of design entities, presented in several design views. We will first look at major design decisions, which are how the system will perform its functions. After that we had to implement the system with all the required operations. This document contains two UML diagrams that show the architecture of the system, a component diagram and a deployment diagram. We will also provide a detailed UML class diagram.

## Resources – References

Kontogiannis, K. (2018). CS2212B Introduction to Software Engineering, lecture slides week 5. Retrieved from <https://www.owl.uwo.ca>

Brabers, G., Charles, C. and Harmsen, W. (2018). Software Requirements System for a Course Management System.

# Major Design Decisions

Most of the design decision made in this project were based on the sample code that was provided to us, but some design decisions are also made by ourselves. An important design decision is that we decided to use the Singleton design pattern for the implementation of the server. Furthermore we used text files to represent our databases, what simplified our program. One of the requirements of this program was that every user should have access to only his own operations, so we decided to implement this by giving every operation its own class.

# Architecture

Below you can find our component and deployment diagrams together with their documentation.

## Component Diagram

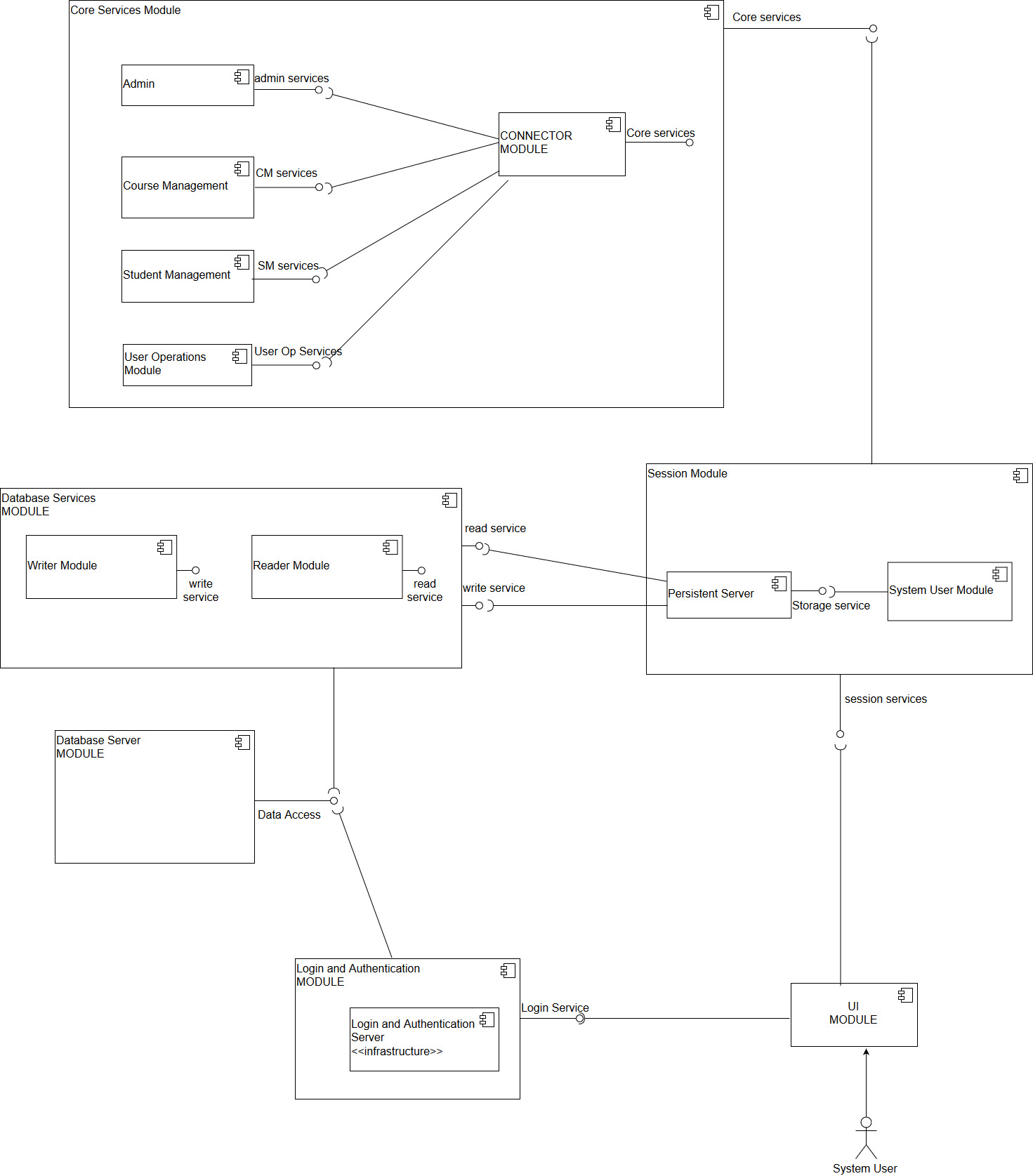


Figure 3.1.1 Component Diagram

**Documentation for Figure 3.1.1**

*CM\_Services*

CreateCourse Factory

Create Course Object

Attach evaluation entities for course

Attach evaluation strategies for course

Attach Instructor to Course

*Add Mark*

Final

Midtem

Projects

*Modify Mark*

Final

Midtem

Projects

Calculate Final Mark

*SM\_Services*

Enroll student in course

Print record for student

Declare status of enrollment

Select and change notification strategy

*Admin\_Services*

Start System

Stop System

Create Instructor

Create Student

Read Student record

Write Student Record

*Write Course Record*

Write students enrolled

Write students allowed to enroll

Write evaluation entities

Write evaluation strategies

Write student records for a course

*Read Course Record*

Read students enrolled

Read students allowed to enroll

Read evaluation entities

Read evaluation strategies

*Login and Authentication Services*

Login

Authentication

*Data access*

Provides access to contents of database

*Database Services*

Read to Database

*Storage service*

Provides a simple interface between the System User Module and the database services

*Session services*

Provides a simple interface between the UI Module and Session Module

## Deployment Diagram

This is our deployment diagram.

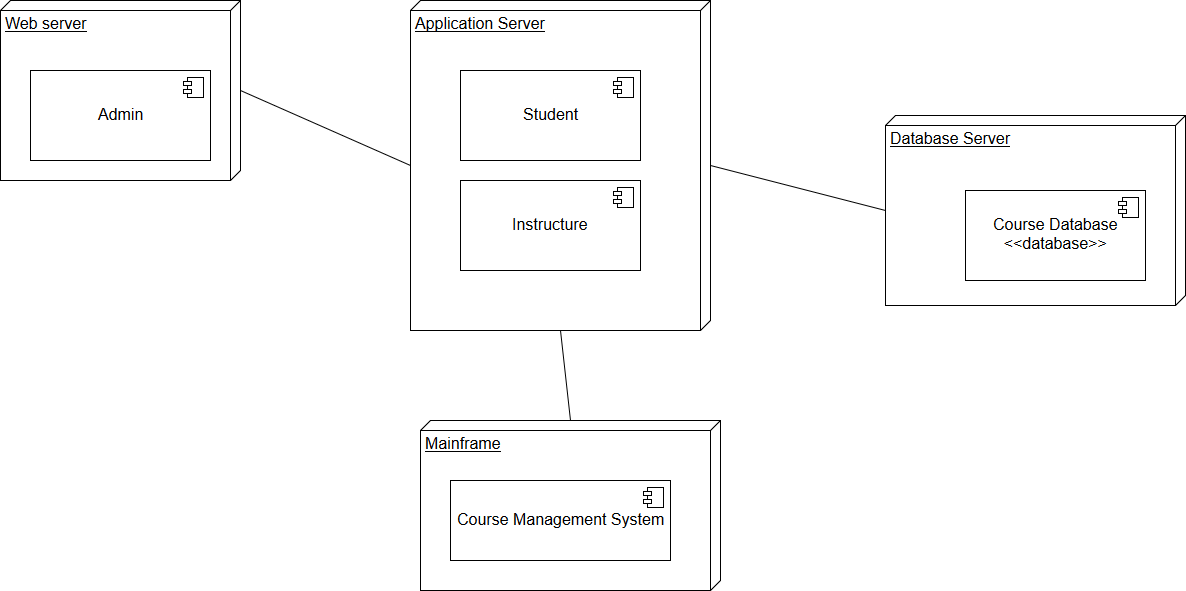


Figure 3.2 Deployment Diagram

# Detailed Class Diagrams

## UML Class Diagrams

On the next page you can find our class diagram. Because it is so big, the text becomes a bit unreadable, that is the reason why we also added this picture as attachment to this document.

