Software Requirements Specification

for

DegreeOveriview

Version 1.0 approved

Prepared by Bohui WU, Xuan WANG, Fuhao RUAN, Yu WU

Lily

03/15/2021

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Project Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Features 3

2.3 User Classes and Characteristics 4

2.4 Operating Environment 4

2.5 Design and Implementation Constraints 4

2.6 User Documentation 5

2.7 Assumptions and Dependencies 5

3. System Features 5

4. External Interface Requirements 5

4.1 User Interfaces 5

4.2 Hardware Interfaces 5

4.3 Software Interfaces 5

4.4 Communications Interfaces 6

5. Other Nonfunctional Requirements 6

5.1 Performance Requirements 6

5.2 Safety Requirements 6

5.3 Security Requirements 6

5.4 Software Quality Attributes 6

6. Other Requirements 6

Appendix A: Glossary 6

Appendix B: Analysis Models 6

Appendix C: Issues List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Bohui WU,  Xuan WANG,  Yu WU,  Fuhao RUAN | 03/16/2021 | The first version. | 1.0 |
|  |  |  |  |

# Introduction

## Purpose

The software requirement specification (SRS) is for the project “DegreeOverview” from Lily. The current version of SRS is 1.0, released on March 16, 2021. This document will give a high-level overview of DegreeOverview in section 1, system features in section 2, external interface requirements in section 3, and other non-functional requirements and other requirements in sections 5 and 6. This document is designed to help developers determine the major features of the product.

## Document Conventions

This document uses the font “Arial” as the font for heading 1, heading 2, and the contents. The font sizes are 18, 14, and 11, respectively.

In this document, all dates will be written using the date and time notation in the United States, which is MM/dd/yyyy.

In this document, **bold text** indicates the content is of great significance, and the reader should pay more attention to it. *Italic text* is used when company names and product names are mentioned.

## Intended Audience and Reading Suggestions

The intended readers of the document and their reading suggestions are as follows:

* Product managers: please read from section 1 to section 3;
* System architects: please read section 2 and section 3;
* Software developers: please read the entire document from section 1 to section 6;
* Testers: please read from section 2 to section 6.

## Project Scope

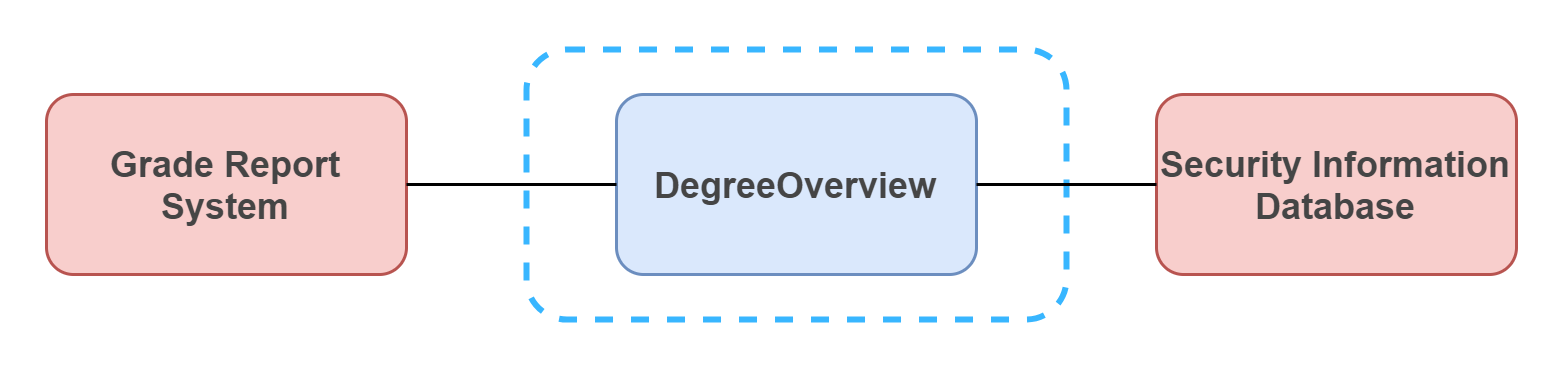
DegreeOverview is a course definition system that aims to help course designers better plan and design university courses, and enables both the lecturers and students to understand and visualize the relationships between courses and their intended learning outcome.

## References

Software Development Workshop III Project V2

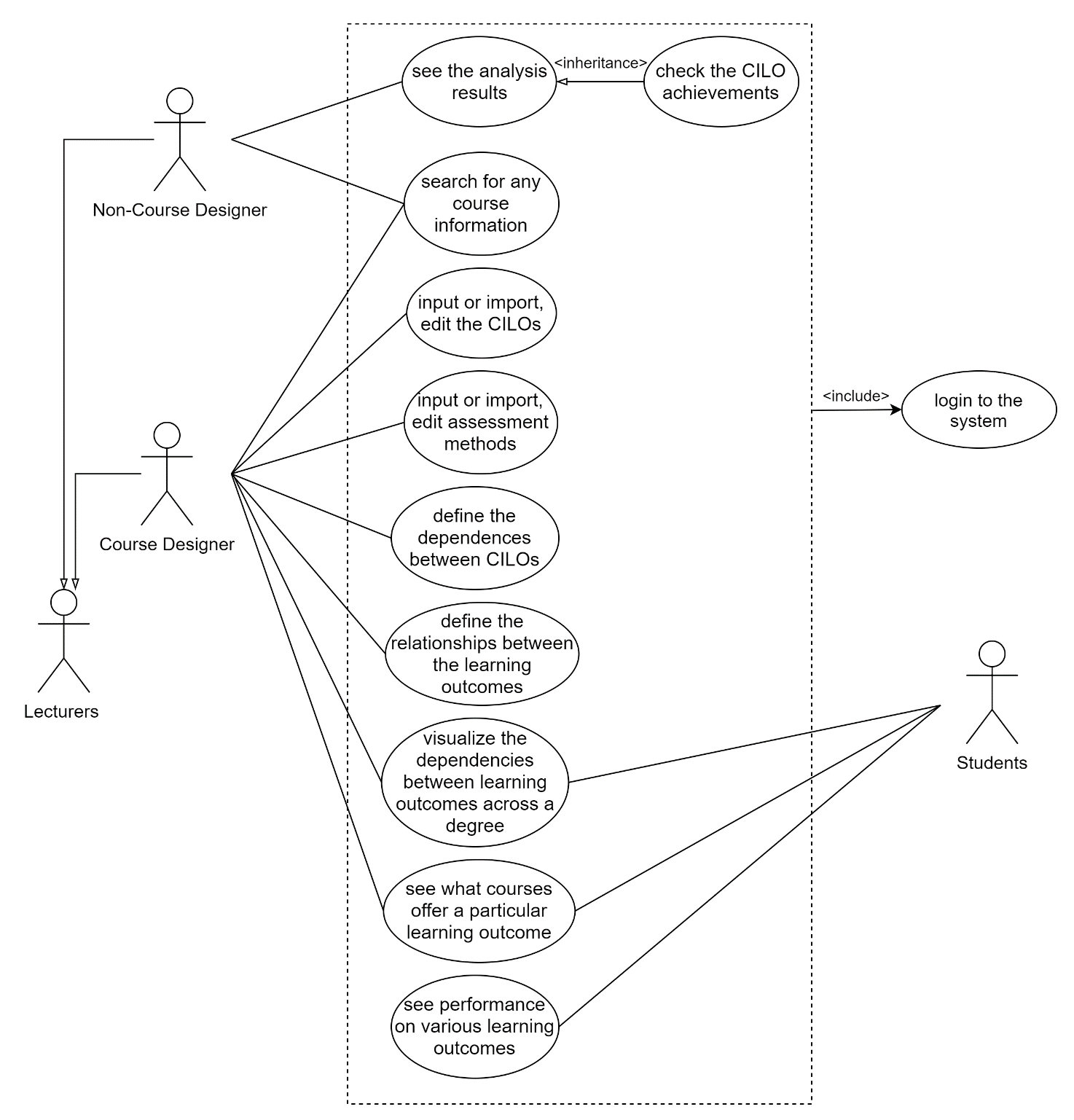
# Overall Description

## Product Perspective



DegreeOverview is a software designed to deploy to a server running Windows Server. However, to function properly, some of the data may come from other independent systems within the campus. In this case, DegreeOverview needs to be connected to the university’s Security Information Database and the Grade Report System.

## Product Features



The following description is directly quoted from the *Software Development Workshop III Project V2* after correcting some grammatical mistakes.

**For Lecturers (Course Designer)**

* Search for information about any course available in the system.
* Input or import, edit the CILOs of a course (refer to the syllabus sample document)
* Input or import, edit assessment methods of a course (refer to the syllabus sample document).
* Define the dependencies between CILOs of courses (CILOs of a course can depend on some CILOs of its prerequisite courses).
* Define the relationships between the learning outcomes (CILOs) and the course assessment and their percentages (refer to the syllabus document.
* Visualize the dependencies between learning outcomes across a degree.
* See what courses offer a particular learning outcome.

**For Normal Lectures (Non-Course Designer)**

* Search for information about any course available in the system.
* See the analysis results (not a MUST function).
* The lecturer can check the CILO achievements for a course (e.g., students’ average performance in one year on a CILO, comparing average performance for different years on a CILO).

**For Students**

* Visualize the dependencies between learning outcomes across a degree.
* See what courses offer a particular learning outcome.
* See his or her performance on various learning outcomes.

## User Classes and Characteristics

There are three classes of users, lecturers (course designers), regular lecturers (non-course designers), and students. They all come from the same university. The division of their classes is based on their role in the university. For the subset of functions allowed, please refer to section 2.2. They all have access to the Internet and are equipped with basic knowledge about how to use a browser on a computer.

## Operating Environment

**Back-end**

* Operating system: Windows 7 or above.

**Front-end**

* Chrome 89.0.4389.82 or above.

## Design and Implementation Constraints

**Policies**

* TO BE DISCUSSED

**Language**

* Object Orient

**Hardware limitations**

* TO BE DISCUSSED

**Interfaces to other applications**

* Security information database;
* Grade report system;
* Web server;
* Software’s database.

**Specific technologies, tools**

* TO BE DISCUSSED

**Communications protocols**

* HTTP/HTTPS

**Design conventions or programming standards**

* TO BE DISCUSSED

## User Documentation

No user manual or online help is provided.

## Assumptions and Dependencies

The server should be connected to the network where it will be accessible by the end-users. The users must be able to connect to the network that the server can be reached.

# System Features

TO BE DISCUSSED.

# External Interface Requirements

## User Interfaces

TO BE DISCUSSED.

## Hardware Interfaces

The software does not interact directly with any physical devices.

## Software Interfaces

**Database**

* The software requires a relational database to store all the information enclosed within the software. In this case, MySQL is preferred.

**Grade Report System**

* TO BE DISCUSSED.

**Security Information System**

* TO BE DISCUSSED.

**Web Server**

* TO BE DISCUSSED.

## Communications Interfaces

Users can access the front-end via HTTP or HTTPS.

# Other Nonfunctional Requirements

## Performance Requirements

TO BE DISCUSSED.

## Safety Requirements

TO BE DISCUSSED.

## Security Requirements

TO BE DISCUSSED.

## Software Quality Attributes

TO BE DISCUSSED.

# Other Requirements

TO BE DISCUSSED.

Appendix A: Glossary

HTTP: Hypertext Transfer Protocol

HTTPS: Hypertext Transfer Protocol Secure

Appendix B: Analysis Models

TO BE ADDED.

Appendix C: Issues List

None.