Design Document

SCIS Curriculum Management

by

Gabrielle Moestar and Oscar Aparicio

Course Instructor: Masoud Sadjadi

Mentor: Tim Downey

Client: Eric Johnson

CIS 4911 – Senior Project

12/04/2012

COPYRIGHT AND TRADEMARK NOTICES

Contact Information

Florida International University - Modesto Madique Campus

School of Computer and Information Sciences

11200 SW 8th Street, Room ECS-258

Miami, FL 33199

Abstract

This project aims to address the shortcomings by creating, implementing, and developing a content management system that is sophisticated and user friendly enough for the least technologically savant user to allow the creation and maintenance of course information and requirements for the curriculum of user-created workflows to be used on the data fed in to it. This system uses pre-existing course information data and requirements collected over the years of the existing educational programs provided by the SCIS Department and curriculum committee.

This document serves to describe the design document. The document is divided in to three main chapters, accompanied by two additional chapters containing extra information and diagrams. The first chapter is the introduction briefly defining the problem at hand, along with the software process model used to design the methodologies for this system. The second chapter focuses on the detailed system design and its decompositions, the hardware and software mapping, persistent data management, and security/privacy. The third major chapter excessively elaborates on the detailed design with the use of static and dynamic models, and code specification. The two final chapters, Appendix and References, serve as the locations for more information referred to throughout the body of this document.

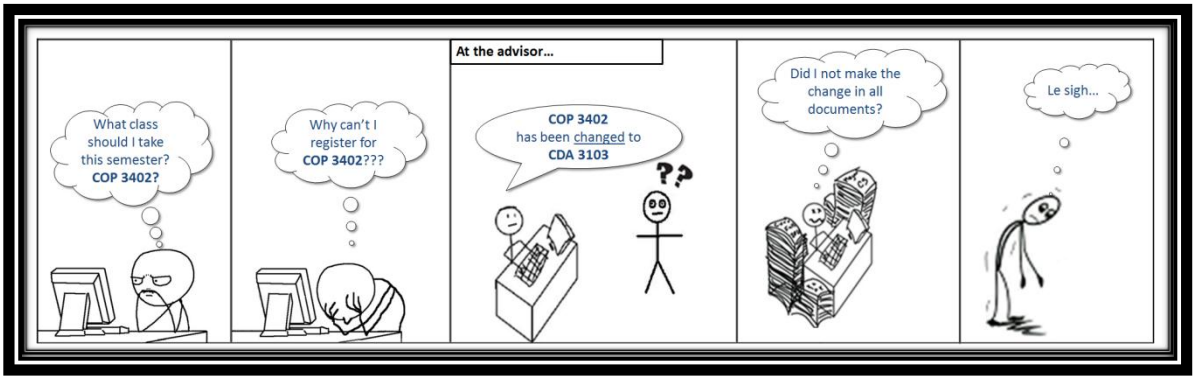
TABLE OF CONTENTS

1. Introduction

The SCIS Management system is a content management system designed to easily create and manipulate curriculum information with the help of a user-friendly web based application that will retrieve data from a database containing all course information and course requirements.

1.1 Problem Definition

The problem defined for this current system is the inability to easily create and maintain course information and requirements of the curriculum across documents consistent, which allows confusion for students.



This information can be obtained through different sources, an advisor, the departmental website, the catalog, and a major map. The data distribution through these documents appears to be inconsistent, and not in sync. (Undergraduate Programs)

The reason for this unfortunate inconvenience is initiated at the creation and maintenance of the curriculum by the curriculum committee.

1.2 Design Methodology

The SCIS Management system has been implemented with a relatively new methodology software development known as Agile Software Development.
Agile Software Development is based on iterative and incremental development to perform in a highly collaborative manner. It is a conceptual framework that encourages rapid and flexible response to change. This method promotes iteration which may not be functional enough for production, but the goal is to have an available release with minimal bugs at the end of each iteration.
In other words, a feature is developed and implemented until it's functional enough to present to the client, who in turn has to approve its current state. The project team proceeds to the next phase, and repeat the process until the product is finalized.

1.3 Definitions, Acronyms, and Abbreviations

SCIS – School of Information and Computing and Information Sciences

Subsystem - A group or library of well-related classes in a project

MVC – Model View Controller

1.4 Overview of Document

The remaining chapters of the document consist of two major chapters and two reference chapters.
Chapter 2 – System Design decomposes the entire system in to subsystems, mapping related hardware and software, data management, and security/privacy.
Chapter 3 – Detailed Design describes the behavior and structure of each subsystem with its static and dynamic models, and associated code.
The last two chapters, Chapter 4 – Glossary and Chapter 5 – Appendix, contain information serving as the locations for more information referred to throughout the body of this document

2. System Design

This chapter provides a high-level description of the system design. First, the decomposition of the system with its associated subsystems will be presented, followed by mapping hardware to each software. The third section excessively elaborates on the data management and its structure. Finally, user security and privacy are addressed.

2.1 Overview

SCIS Curriculum Management was build up on two architectural patterns: The MVC patterns, and the Data Repository pattern.

The MVC pattern separates the representation of data from the user's interaction with it. MVC enforces better organization and code reuse.

The Data Repository pattern is used since all the course information and requirements pertain to the central database created for SCIS Curriculum Management system. For relationships and data management refer to **Chapter 2.4.**

2.2 Subsystem Decomposition

The following section provides the subsystem decomposition with a brief description.

- Framework: provided by Yii Framework[1]
- Controller: This part of the system handles all entity controls
- Components: This subsystem deals with the versioned entities and its concrete clases
- Models: This subsystem handels all database features to connect to the system.

2.3 Hardware and Software Mapping

[1] http://www.yiiframework.com/

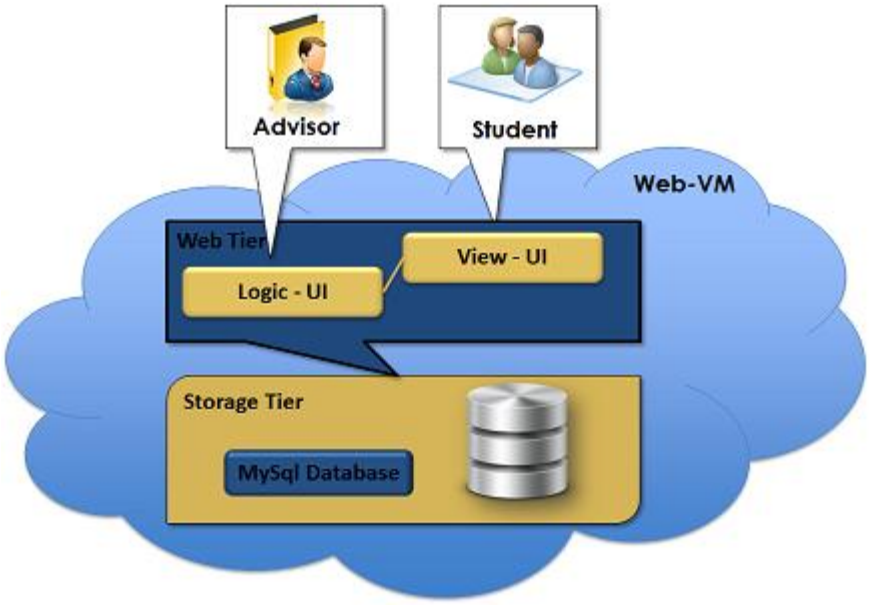


Figure 2-1 Hardware to Software tier mapping

2.4 Persistent Data Management

Appendix B visually depicts the relationships between the data and entities.

2.1 Security/Privacy

The database will be hosted on SCIS department servers, and therefore will be protected by the integrated firewalls on the network.

3. Detailed Design

This third chapter provides an in depth description of the system design through visual depiction of static models.

3.1 Overview

This section describes the behavior and structure of curriculum subsystem.

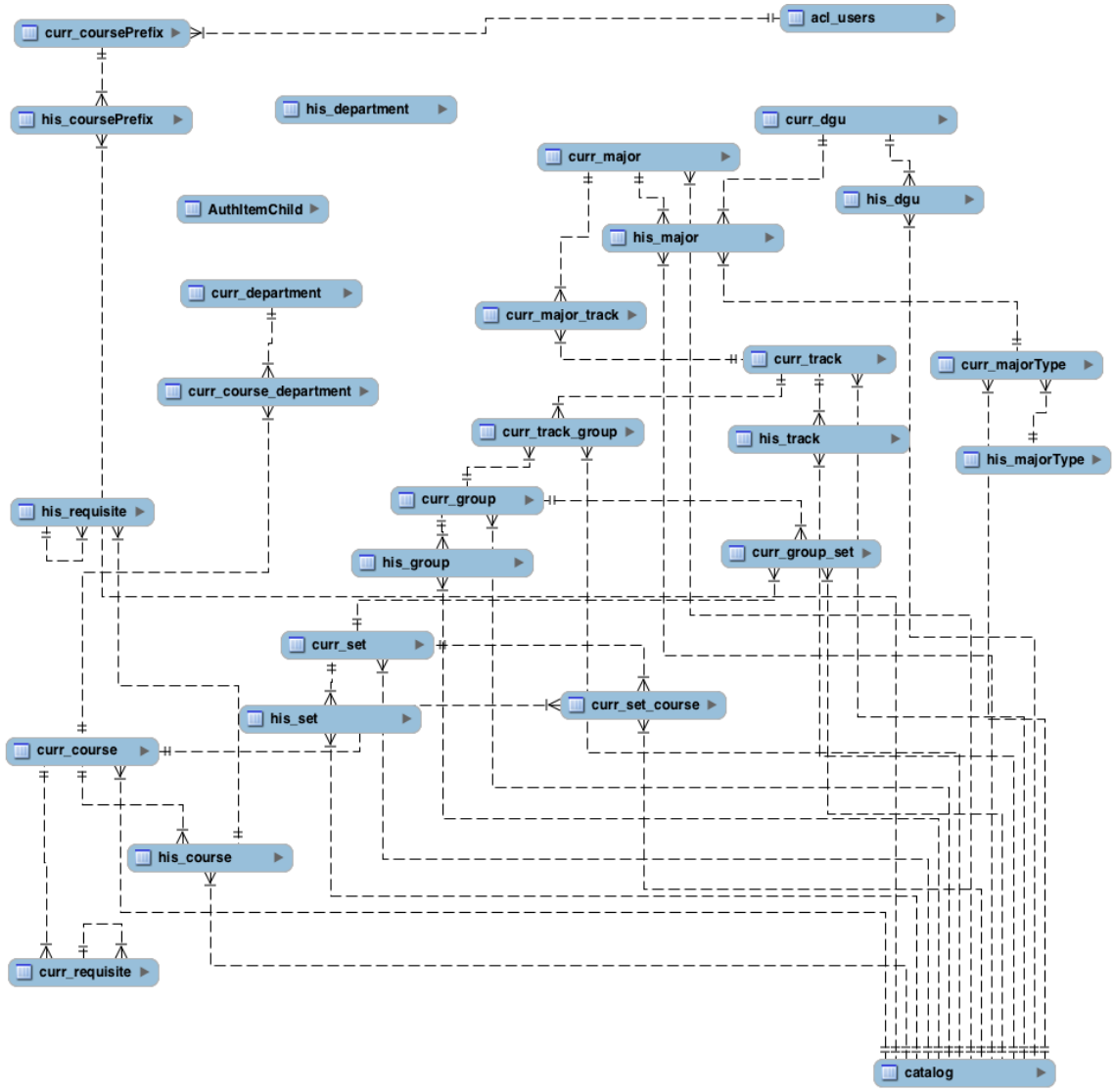


Figure 3-1 Database system structure

3.2 Static Model

The following is a detailed description of the curriculum components in the main subsystem

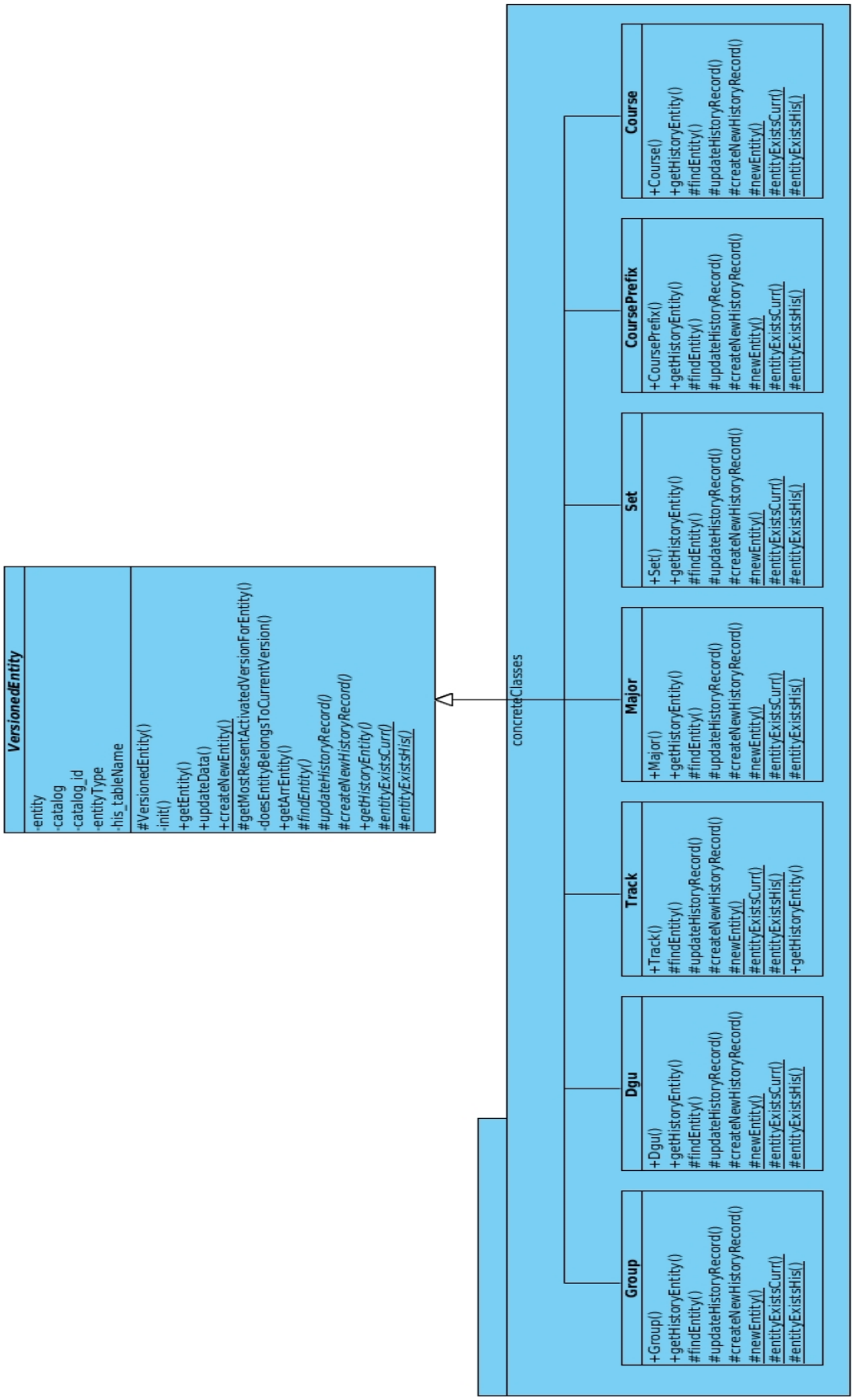


Figure 3-2 Detailed class diagram

3.3 Dynamic Model

This section presents state machine diagrams for the SCIS Curriculum Management's database behavior.

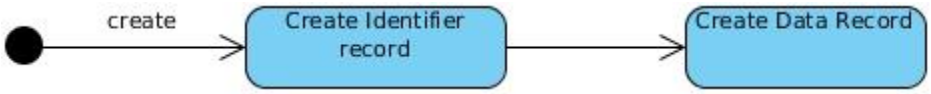


Figure 3-3 CreateEntity

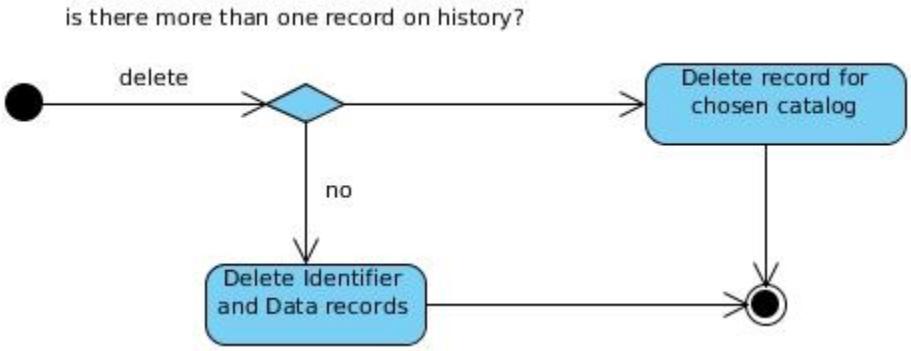


Figure 3-4 UpdateEntity

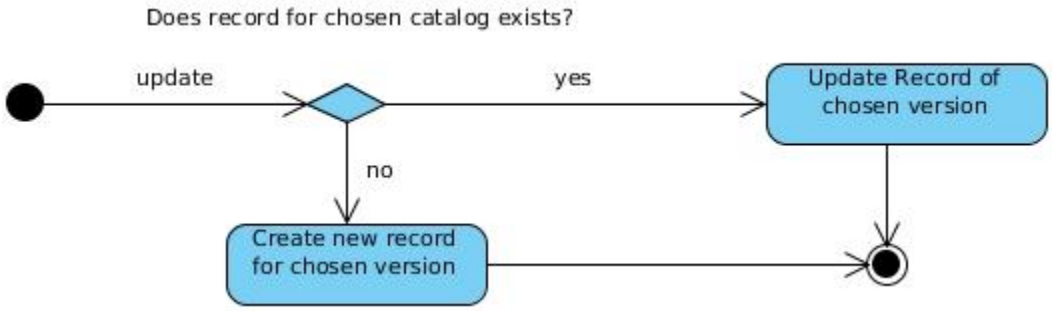


Figure 3-5 DeleteEntity

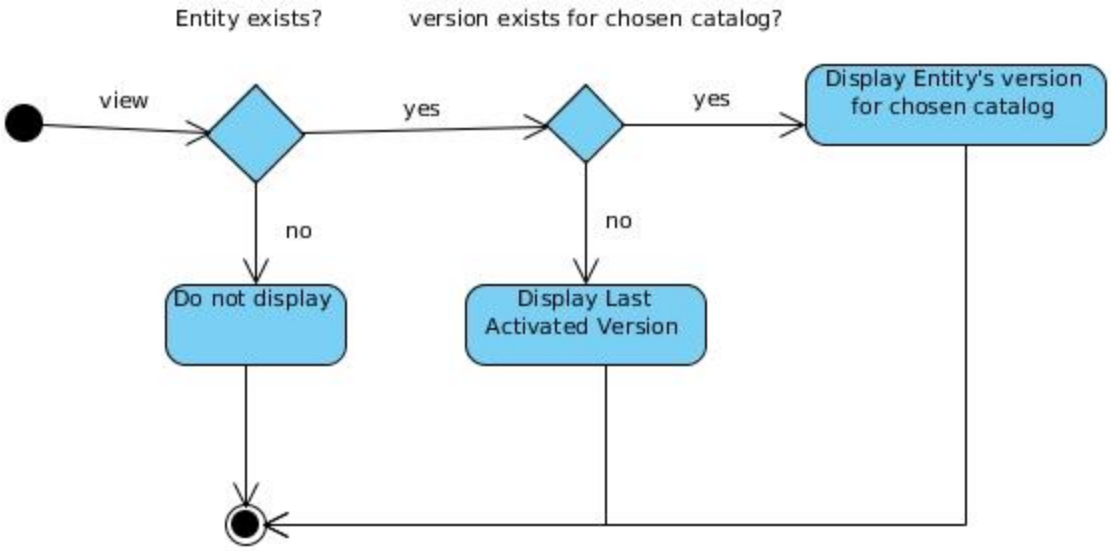


Figure 3-6 ViewEntity

3.4 Code Specification

Disclaimer: Refer to comments throughout the code as well as its attached document.

4. Glossary

- **Class Diagram**: A diagram that describes the structure of a system by showing the system's classes, attributes, and their relationship to other classes.
- **Dynamic Model**: A diagram containing a state machine depicting the main control object in a given subsystem.
- **Sequence Diagram**: A diagram that shows how objects operate with one another and their workflow.
- **Static Model**: A diagram providing a detailed description of the structure of a given subsystem.
- **Subsystem**: A group or library of well-related classes in a project.
- **Use Case**: A description of a potential series of interactions between software and an actor.

5. Appendix

5.1 Appendix A – Use case diagrams

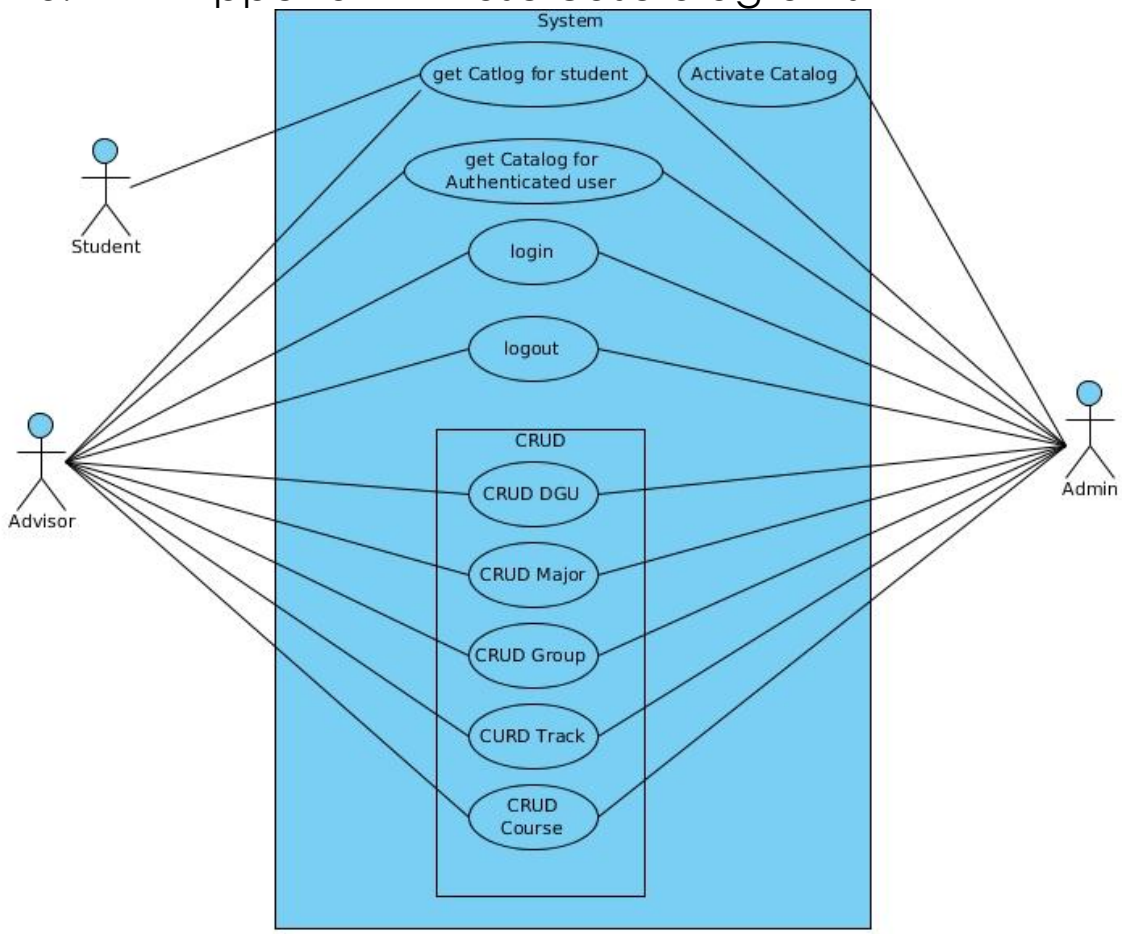


Figure 5-1 Use Case Diagram

5.2 Appendix B – ER-diagrams

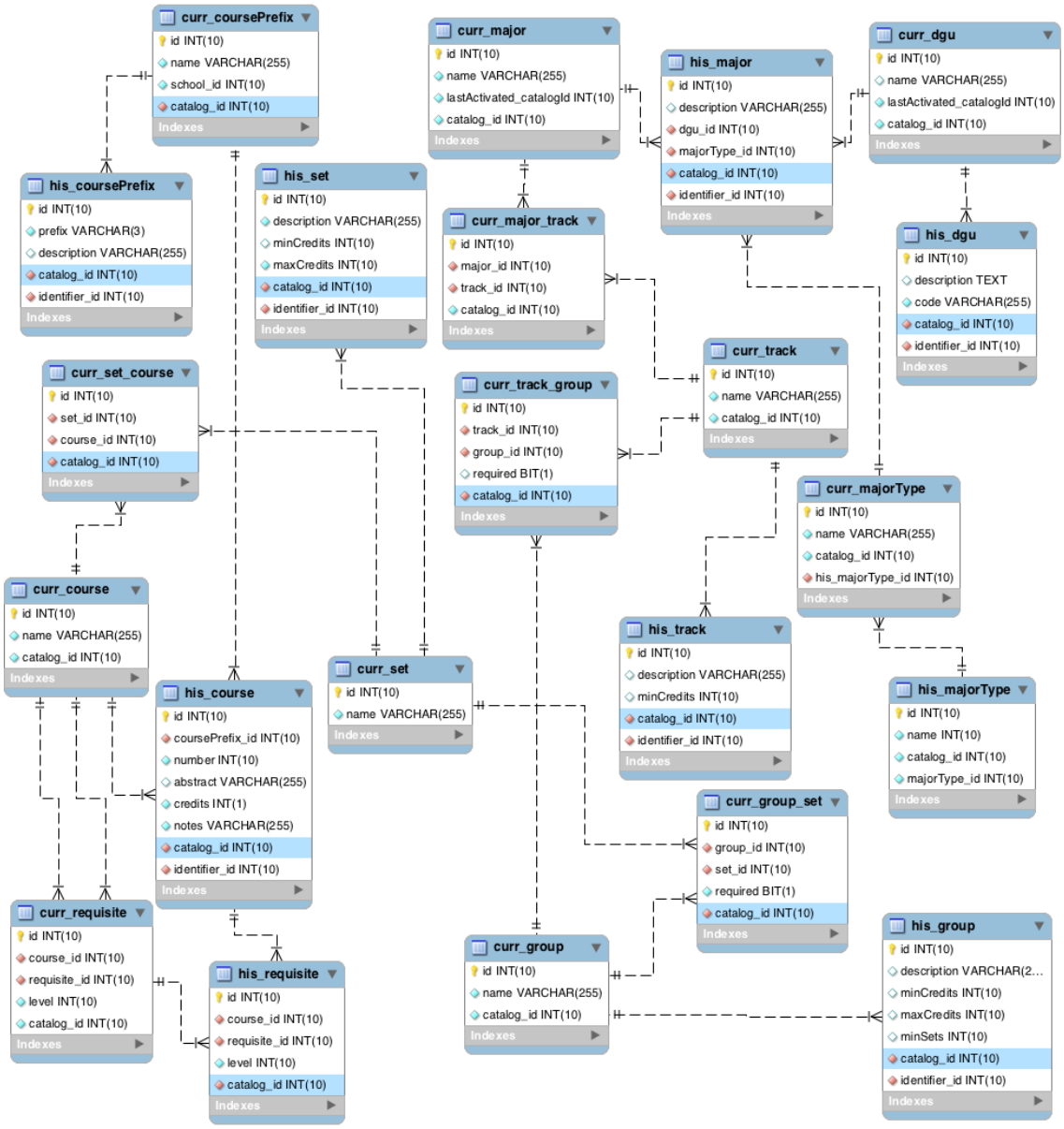


Figure 5-2 Detailed overview of Class Diagram

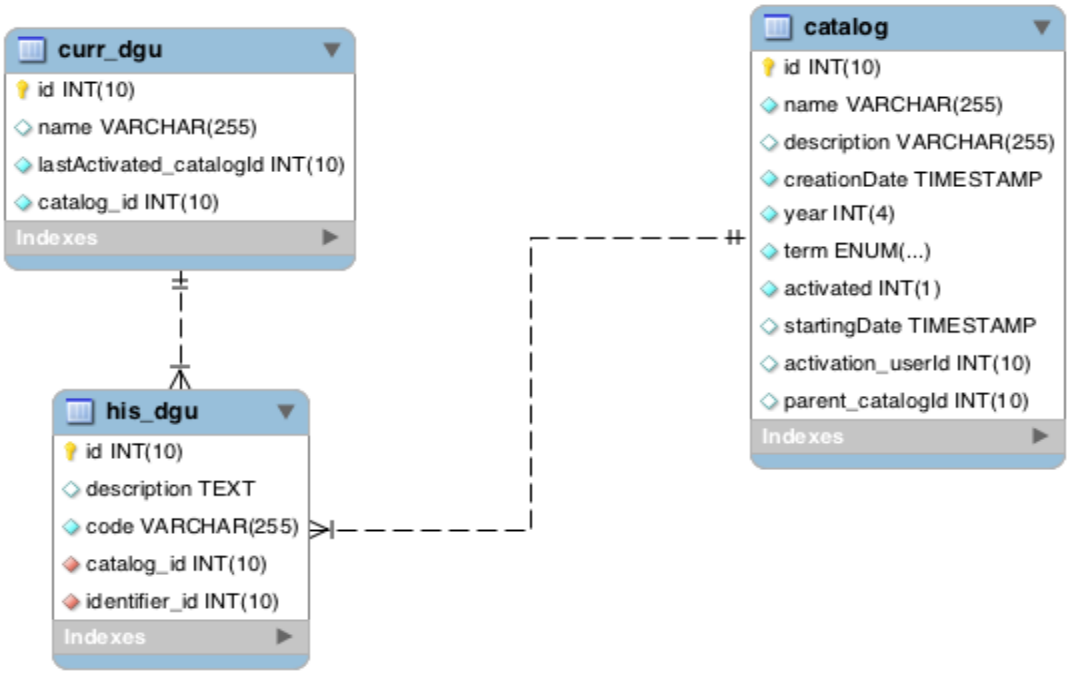


Figure 5-3 Bird's eye view - DGU class diagram

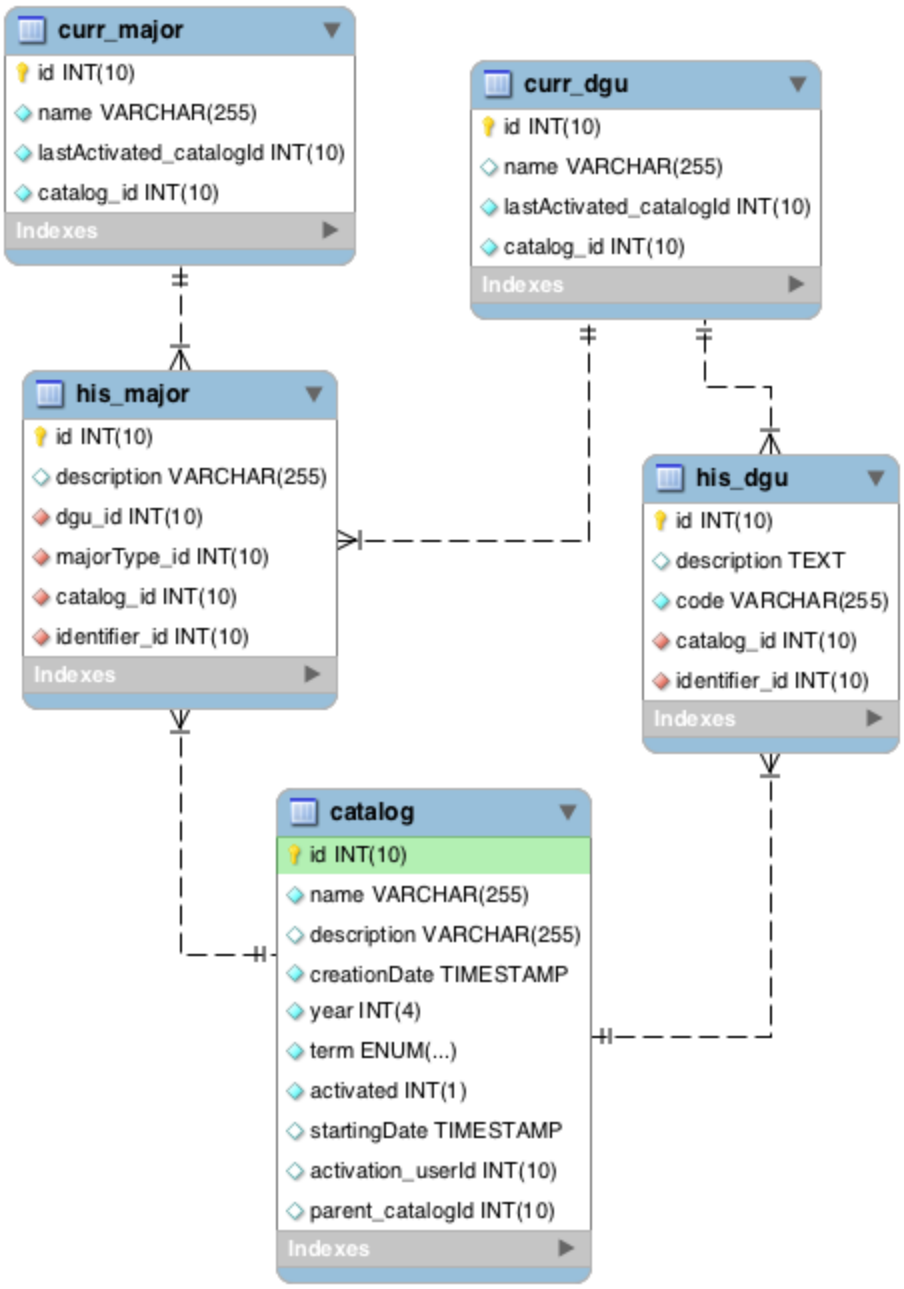


Figure 5-4 Birds-eye view DGU_MAJOR relation

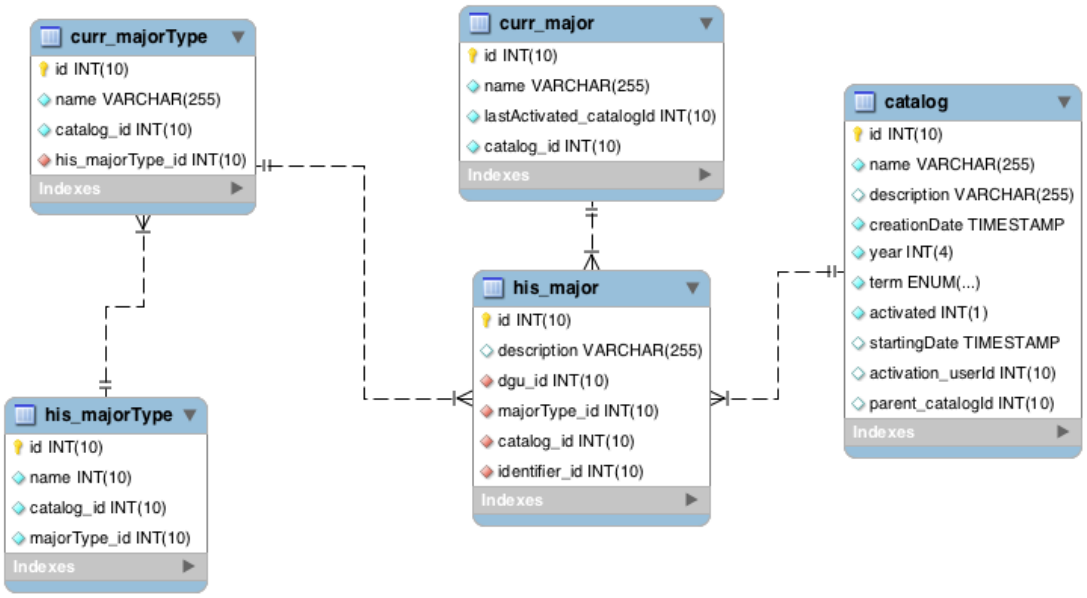


Figure 5-5 Birds eye view major

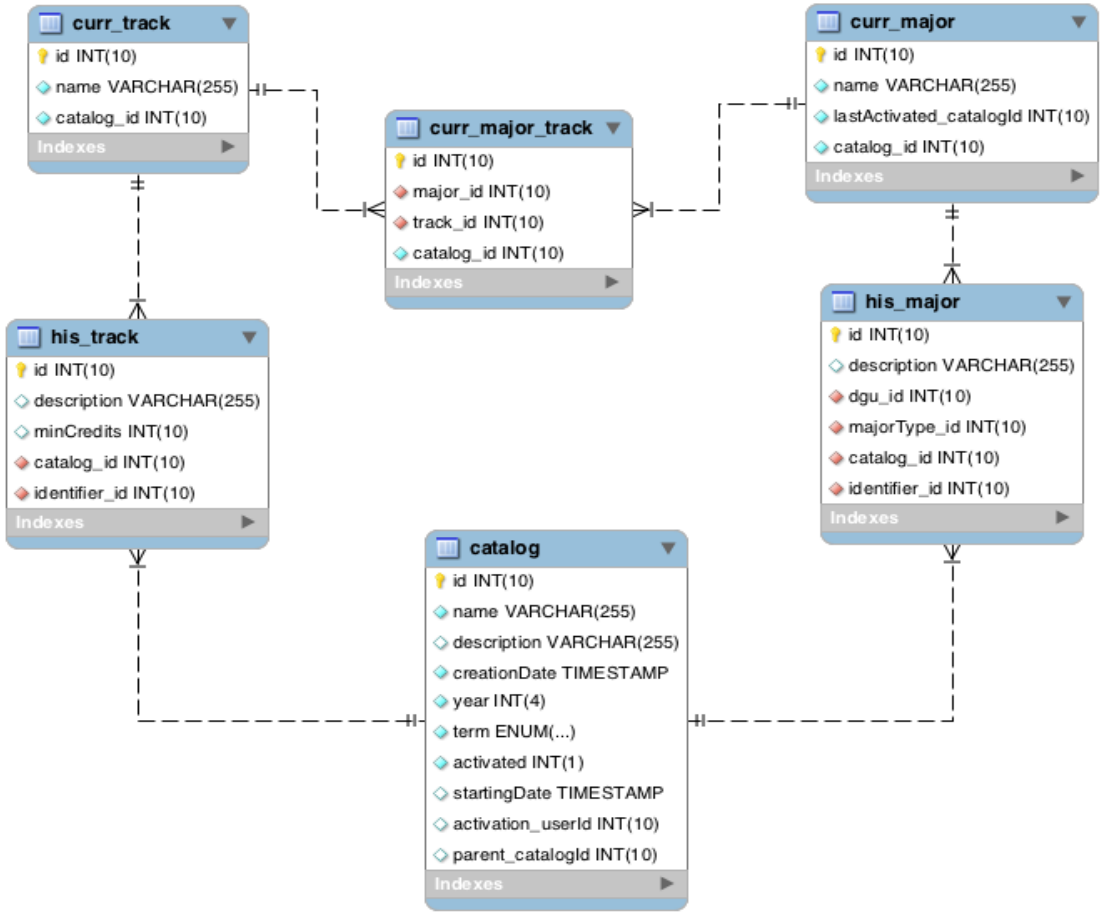


Figure 5-6 Birds eye view major_track relation

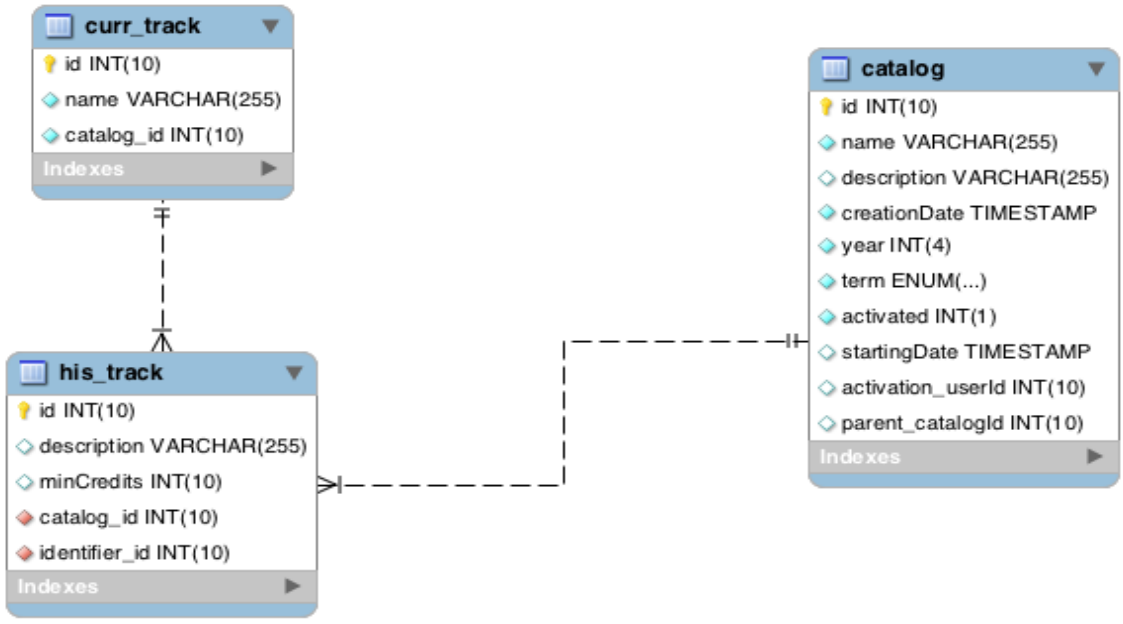


Figure 5-7 Birds eye view track

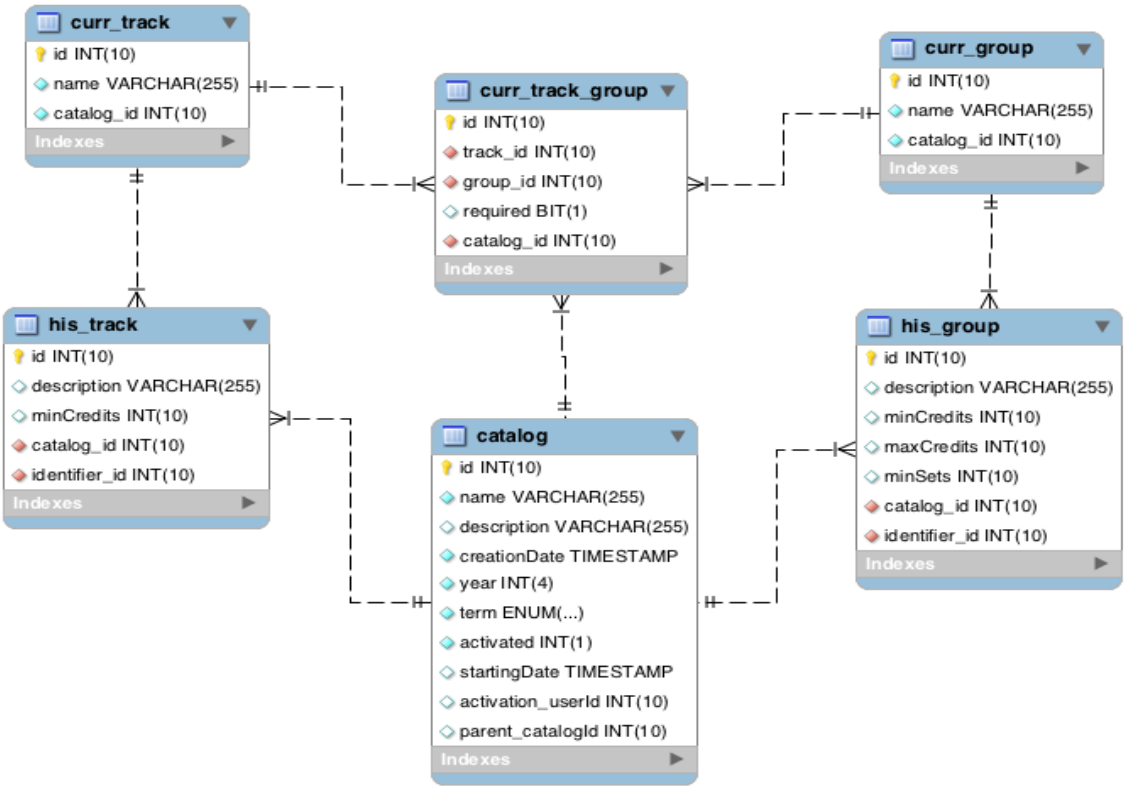


Figure 5-8 Bird's eye view track_group relation

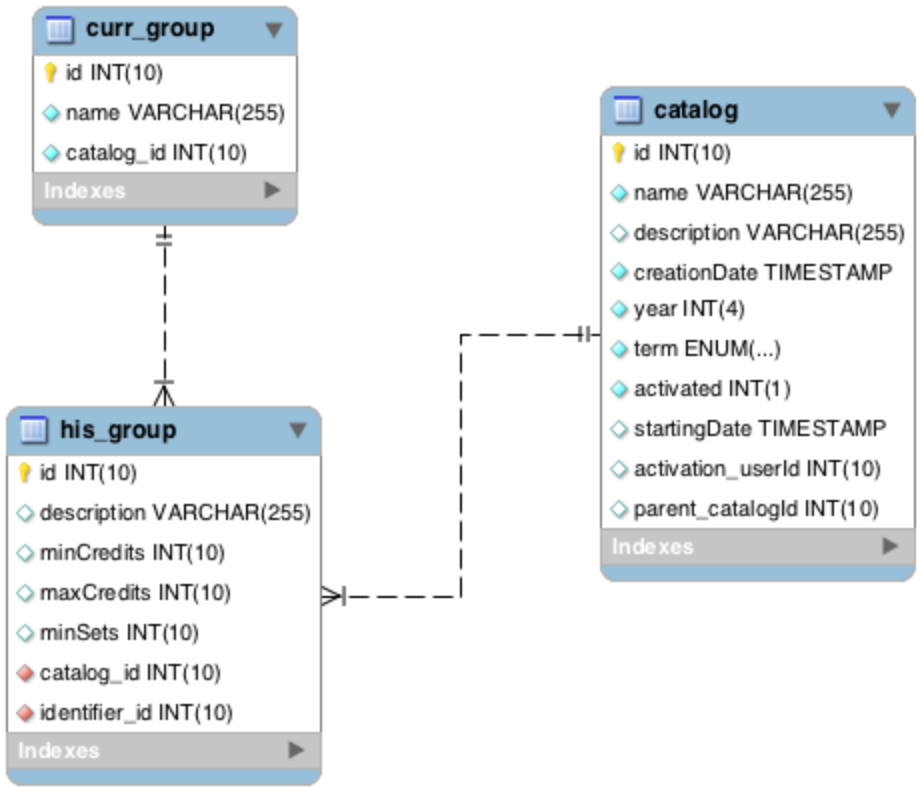


Figure 5-9 Birds eye view group

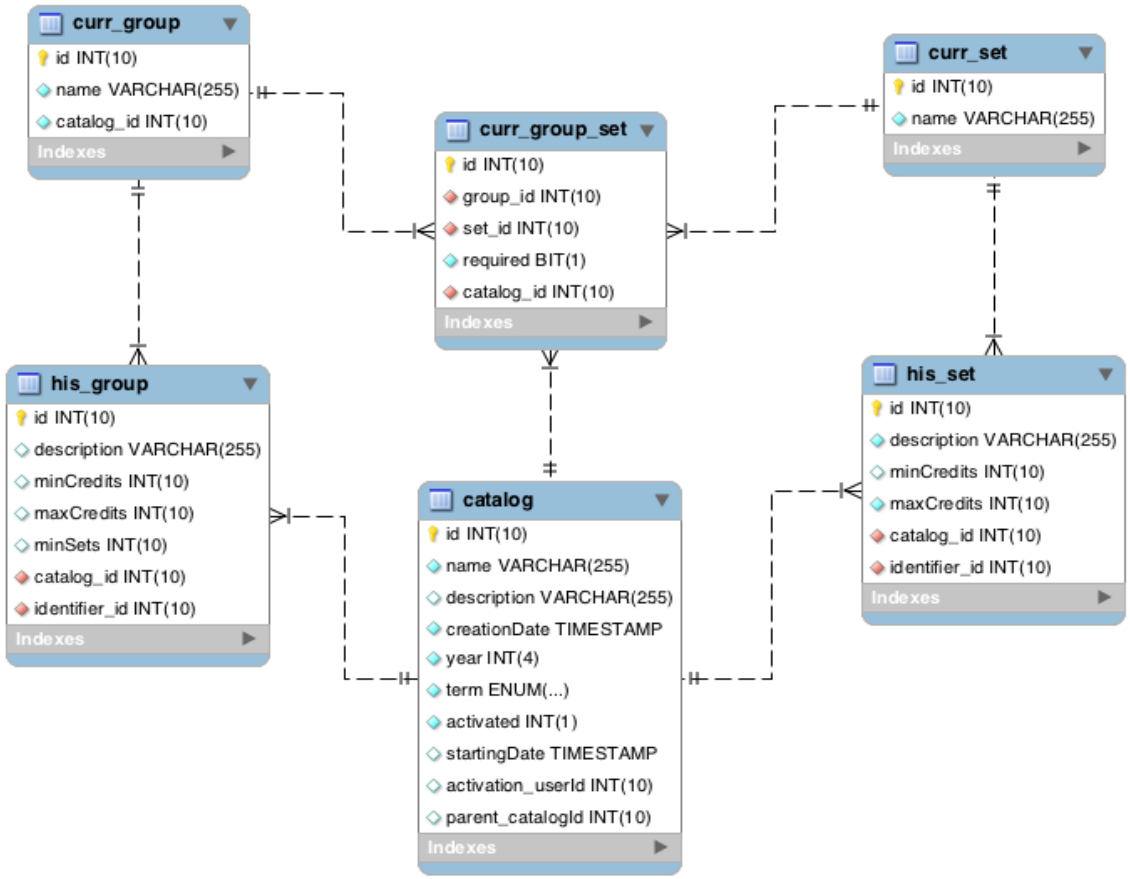


Figure 5-10 Birds eye view group_set relation

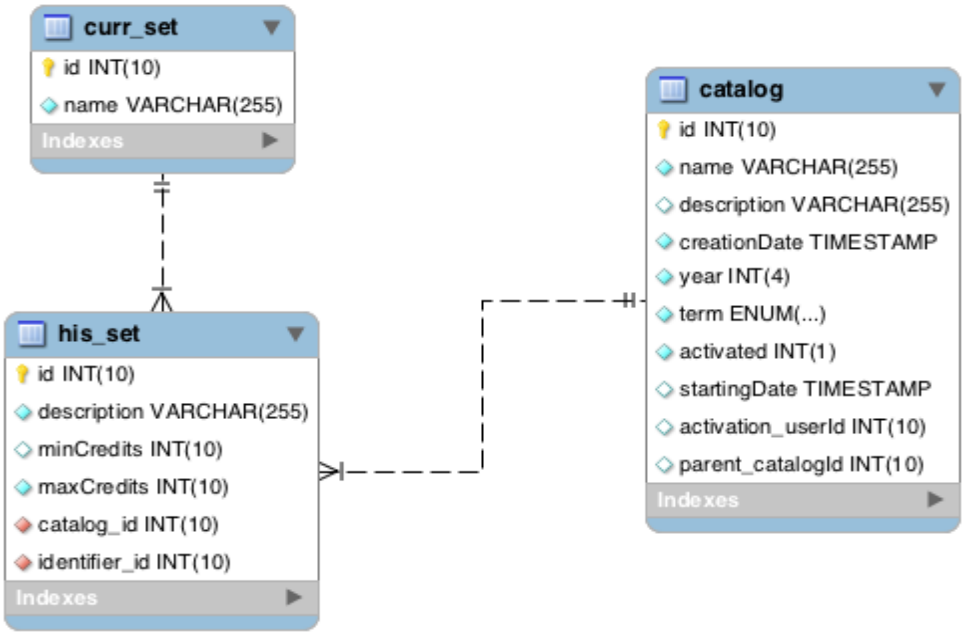


Figure 5-11 Birds eye view set

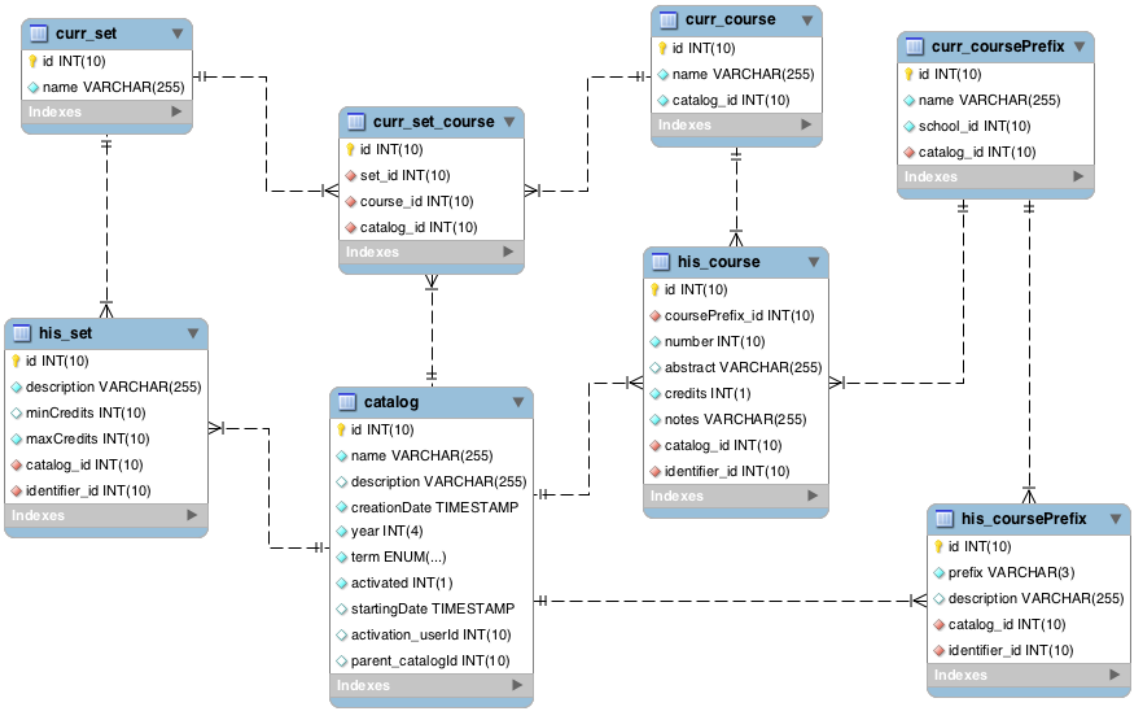


Figure 5-12 Birds eye view set_course relation

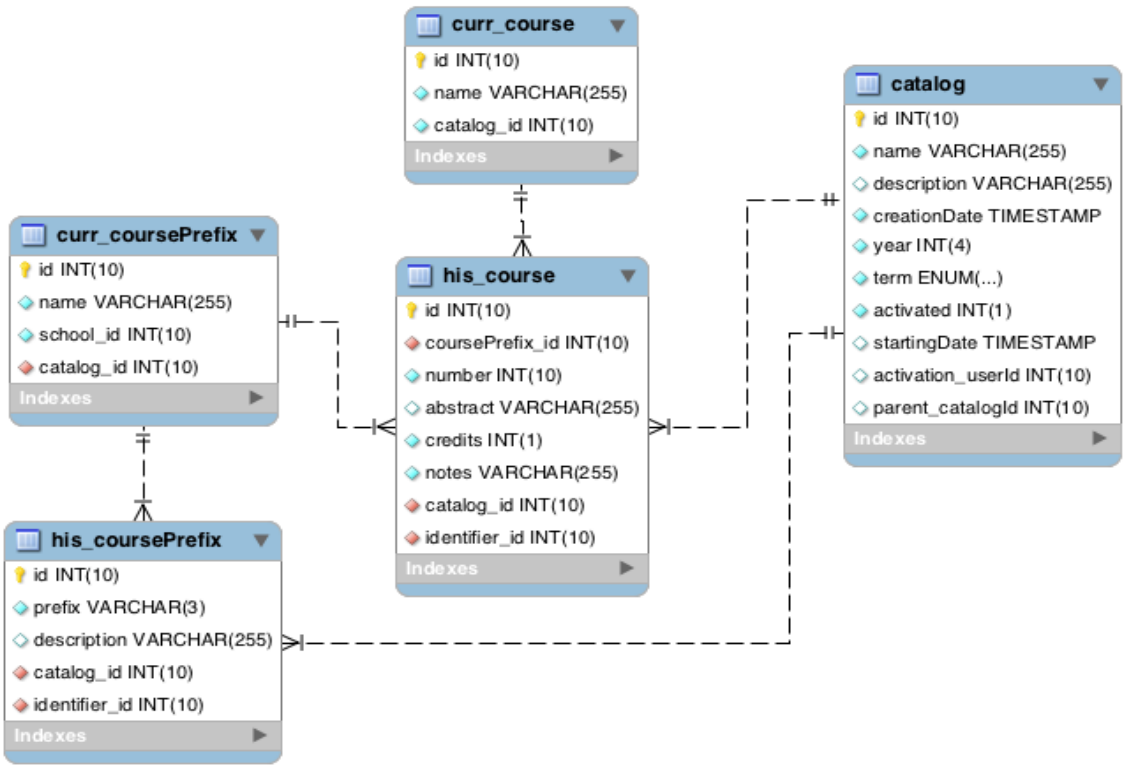


Figure 5-13 Birds eye view course

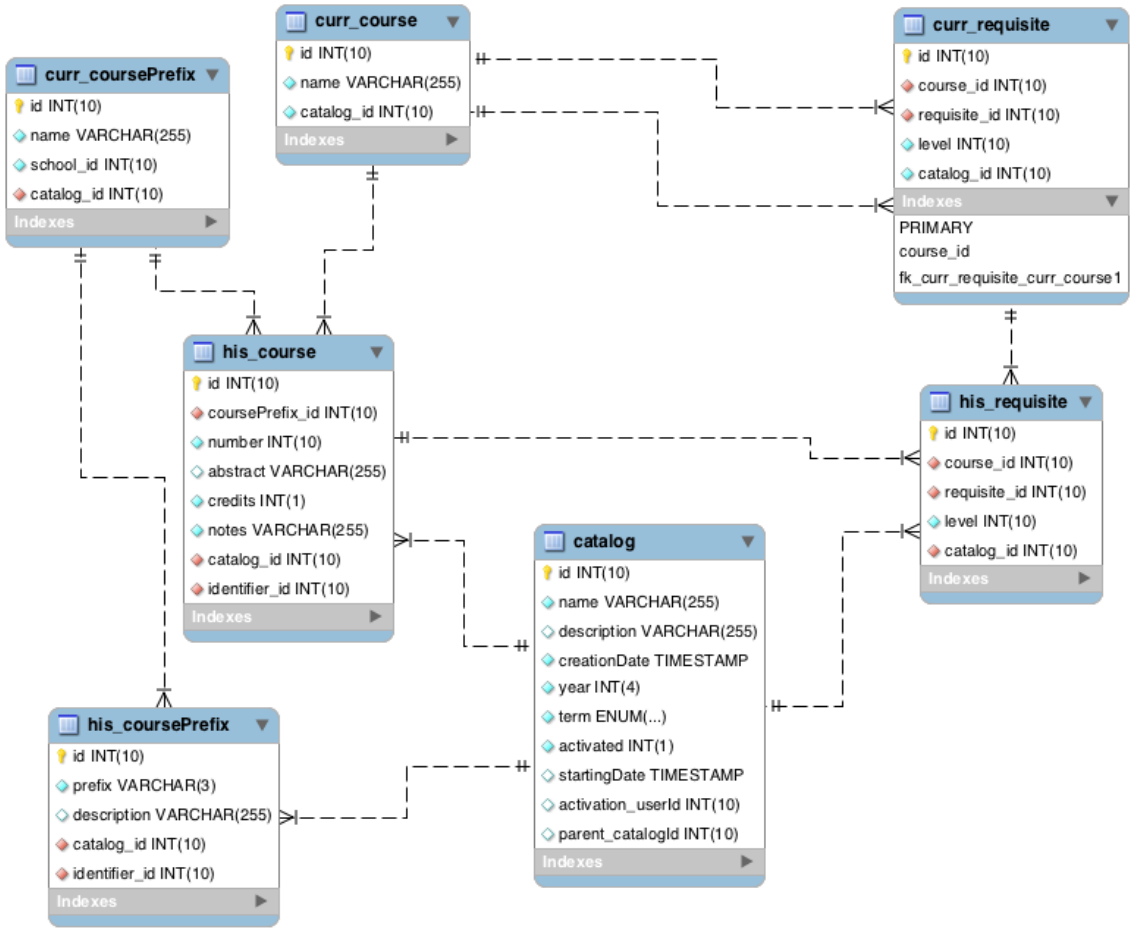


Figure 5-14 birds eye view course_requisite relation

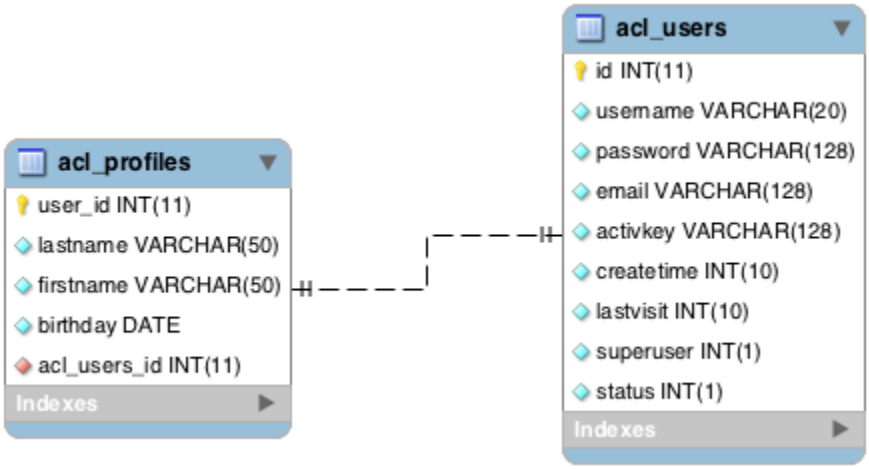


Figure 5-15 ACL-user relation