Ye Mun

[yem7@pitt.edu](mailto:yem7@pitt.edu)

Stephen Wludarski

[swludarski@pitt.edu](mailto:swludarski@pitt.edu)

INFSCI 2710 Database Management

Dimitry Babichenko

April 28, 2023

ETD Submission Management System

Design Document

Table of Contents

[1. Introduction 3](#_Toc1302688626)

[2. E-R Model 5](#_Toc635170140)

[3. Class Model 6](#_Toc1920474887)

[4. Business Rules 7](#_Toc2144420939)

[5. Closing Thoughts 10](#_Toc1370400800)

1. **Introduction**

**Description:**

Submitting Electronic Theses and Dissertations (ETDs) is a graduation requirement in many graduate schools. At Duquesne University, our team has identified issues with the current process for managing ETD submission data. Specifically, the ETD coordinator faces challenges in tracking and updating ETD data locally. Although approved ETDs are published in the institutional repository and ProQuest Central, a local tracking database is still necessary to include basic information about students, committee members, school representatives, time stamps, and document information, including major metadata. This data management is particularly crucial in ensuring the safety of students' records, even in the event of institutional repository or ProQuest database corruption.

To address these challenges, our team has built a relational database using MySQL to streamline the process of ETD submissions. Our solution also utilizes Flask to create a RESTful API endpoint.[[1]](#footnote-1) In particular, our solution allows the ETD coordinator to have basic Create, Read, Update, and Delete (CRUD) database functions. With this approach, we aim to reduce the time spent entering the same data repeatedly, minimize errors, and improve overall efficiency. By creating a more efficient data management process, we will ensure that students' records are kept safe and secure.

This project will be beneficial and meaningful for Duquesne University and other institutions that face similar challenges in managing ETD submissions. By creating an efficient and streamlined process for tracking ETD submissions, we have reduced the time and effort required by the ETD coordinator to manage the data. In addition, the risk of errors or data loss will be greatly reduced, and students' records will be kept safe and secure. The CRUD functions enable the ETD manager to minimize errors, decrease data redundancy, centralize information that was often stored across multiple spreadsheets, and reduce staff time spent on maintaining ETD informational integrality.

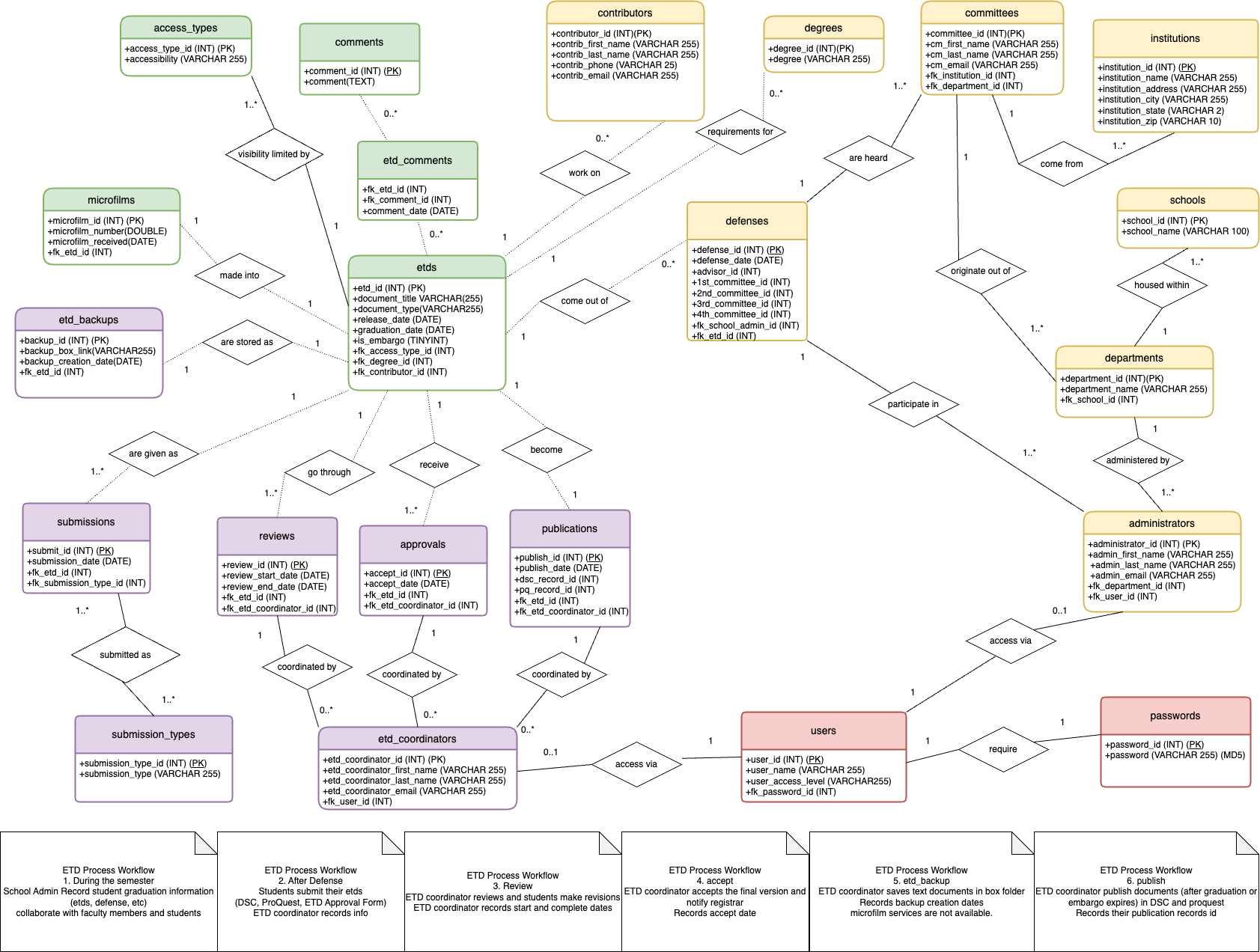
For example, by implementing a relational database, the ETD coordinator will be able to easily track students' progress and deadlines, ensuring that all necessary steps are completed on time. The database will also provide a centralized location for all ETD submission data, eliminating the need for the ETD coordinator to maintain multiple files and spreadsheets. Finally, the ability to set alerts for embargo expiration dates will ensure that these documents are processed on time, further improving the efficiency of the overall process.

1. **E-R Model**

**Diagram

Description automatically generated**

1. **Class Model**



1. **Business Rules**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Entity 1** | **Entity 2** | **Cardinality on Entity 1 side** | **Cardinality on Entity 2 side** | **Nature of Entity Relationship** | **Business Rule(s)** |
| ETDs | Comments | 0..\* | 0..\* | Optional | An ETD may have zero or many Comments.  A Comment may be associated with zero or many ETDs. |
| ETDs | Access Types | 1 | 1..\* | Required | An ETD must have exactly one Access Type associated with it. An Access Type may have one or many ETDs associated with it. |
| ETDs | Microfilms | 1 | 1 | Optional | An ETD may only have one Microfilm associated with it. A Microfilm may only have one ETD associated with it. |
| ETDs | ETD\_Backups | 1 | 1 | Optional | An ETD must only have one ETD\_Backup. An ETD\_Backup must only have one ETD associated with it. |
| ETDs | Reviews | 1 | 1..\* | Optional | An ETD must only have one Reviews timeline. A Reviews timeline might be associated with one or many ETDs. |
| ETDs | Approvals | 1 | 1..\* | Optional | An ETD must only have one Approvals timeline. An Approvals timeline may be associated with one or many ETDs. |
| ETDs | Publications | 1 | 1 | Optional | An ETD must only have one Publications timetable. A Publications timeline must only have one ETD. |
| ETDs | Submissions | 1 | 1..\* | Optional | An ETD must only have one Submission. A Submission might be associated with one or more ETD. |
| ETDs | Contributors | 1 | 0..\* | Optional | An ETD can only be associated with one Contributor. A Contributor might be associated with zero to many ETDs. |
| ETDs | Degrees | 1 | 0..\* | Optional | An ETD can only be associated with one Degree. A Degree might be associated with zero or many ETDs. |
| ETDs | Defenses | 1 | 0..\* | Optional | An ETD can only be associated with one Defense. A Defense might be associated with zero or many ETDs. |
| ETD\_Coordinators | Reviews | 0..\* | 1 | Required | An ETD\_Coordinator might be associated with zero to many Reviews. Reviews must be associated with only one ETD\_Coordinator. |
| ETD\_Coordinators | Approvals | 0..\* | 1 | Required | An ETD\_Coordinator might be associated with zero to many Approvals. Approvalss must be associated with only one ETD\_Coordinator. |
| ETD\_Coordinators | Publications | 0..\* | 1 | Required | An ETD\_Coordinator might be associated with zero to many Publications. Publications must be associated with only one ETD\_Coordinator. |
| Submissions | Submission\_Types | 1..\* | 1..\* | Required | A Submission timeline can be associated with one or more Submission\_Type. A Submission\_Type can be associated with one or many Submission timeline. |
| Users | ETD\_Coordinators | 1 | 0..1 | Required | A User profile must only be associated with one ETD\_Coordinator profile. An ETD\_Coordinator may be associated with only zero or one User account. |
| Users | Administrators | 1 | 0..1 | Required | A User profile must only be associated with one Administrator. An Administrator may be associated with only zero or one User account. |
| Users | Passwords | 1 | 1 | Required | A User must only have one Password. A Password can only be associated with one User. |
| Administrators | Departments | 1..\* | 1 | Required | An Administrator can be a part of one or many Departments. A Department must only have one Administrator. |
| Administrators | Defenses | 1..\* | 1 | Required | An Administrator can serve on many Defenses. A Defense can only have one Administrator serving on it. |
| Departments | Schools | 1 | 1..\* | Required | A Department can only belong to one School. A School can have many Departments. |
| Committees | Institutions | 1 | 1..\* | Required | A Committee member can only be from one Institution. An Institution can have one or more Committee member. |
| Committees | Defenses | 1..\* | 1 | Required | A Committee member can serve on one or more Defenses. A Defense can only be associated with one Committee. |
| Committees | Departments | 1 | 1..\* | Required | A Committee member must belong to one Department. A Department can have one or many Committee members. |

1. **Closing Thoughts**

Throughout the process of conceptualizing and implementing the ETD Submission Management System, our team discovered both expected and unexpected roadblocks as well as our own implemented strategies that helped streamline processes during our development. The team anticipated that the integration of our SQL database into a broader network of programs, which would allow it Create, Read, Update, and Delete (CRUD) functionality would be a challenge given that no one on the team had prior experience with the process. We were surprised, however, at how important clarifying the business rules for ETDs at Duquesne University was to quickly and efficiently implement the database. While it originally seemed very apparent from the data, we had received that tracked ETDs, the process of designing and talking out the logic that governs the process enabled us to develop a more streamlined and usable product for ETD Coordinators.

Looking ahead, our team plans to further improve and iterate on our current implementation with the specific goal of creating a Graphic User Interface (GUI), which will further enable ETD Coordinators to more effectively carry out their work. Though this current version of the project does not yet have a GUI, we feel that if we had a little more time to design, test, as well as having an easy area to host the project would have more easily enabled us to implement that portion of the project. Yet, we feel that even without a fully implemented GUI our team has effectively provided a way for ETD Coordinators to better manage the work at Duquesne University.

1. https://github.com/imyem7/etd\_db/blob/main/final-documentation/API\_documentation.md [↑](#footnote-ref-1)