

**CPSC 471
Spring 2019**

Select Electives

FINAL REPORT

**Kevin Ng
30029178
T01**

**Cristhian Sotelo-Plaza
30004060
T01**

**Katrina Chanco
30037408
T02**

Date: June 17, 2019

This report will go over the development of our web app, Select Electives. A database that will organize all elective based courses in several post-secondary institutes for students to look through. In the first section, we will go over an in-depth explanation of the problem at hand; the lack of resources to find out what electives are available for students to take, all compiled in a neat and organized fashion. We initially tackled this problem by creating an Entity Relational Diagram to plan out what entity we will keep track of and how each entity interacted with one another, then we structure out a relational diagram to base our database on. Afterwards, we drafted up some SQL queries, so we are able to grasp an idea of how we will have our users use the database, like adding and viewing courses. Our web app will be mainly built upon using JavaScript, along with some html and CSS. The end result is a fully functioning web app that provides two different accounts that can manipulate and bring up information from the database. Adding, removing and editing individual courses and listing multiple courses based on queries. We hope that it will provide potential use to be built and expanded upon in the future.

Introduction

As students who attend the University of Calgary, we face the issue of registering for courses to complete our degrees. Aside from the required list of courses, every faculty has a variety of optional courses that students are able to enroll in. Some of these require other courses to have been taken as a prerequisite, while others can be taken without needing a prerequisite course. Currently, the University of Calgary does not have an application that specifically allows someone to filter through the optional courses to determine which ones do not need a prerequisite, and which ones do. Our group wanted to create an application that provides users with an easy-to-read experience that focuses on available courses students can take that do not require a prerequisite course. This application would display all of these types of courses for users to browse through. More specifically, our group created a web application that utilizes a database to provide users with these features. Users from different schools can browse through the non-prerequisite courses of different faculties within their school, and search through a variety of courses that belong within those faculties. From there, descriptions of each course, comments for that course, and related textbooks are provided for the user's convenience. In addition, the user can add and remove courses to a Wish List that displays specific courses that the user would want to revisit later. There will be an Administration view, and a Student view for the web application that will have different features. The Administration Account will be able to edit, add and remove components of the web app. Student accounts can view information on the app, and interact with the Wish List. An Admin is also able to contact a Student from the same school and vice versa. All in all, our web application called, 'Select Electives' will provide the user with an interactive platform that displays elective courses which simplifies the selection process students go through when deciding what electives they would like to take.

Functionality and User Interaction

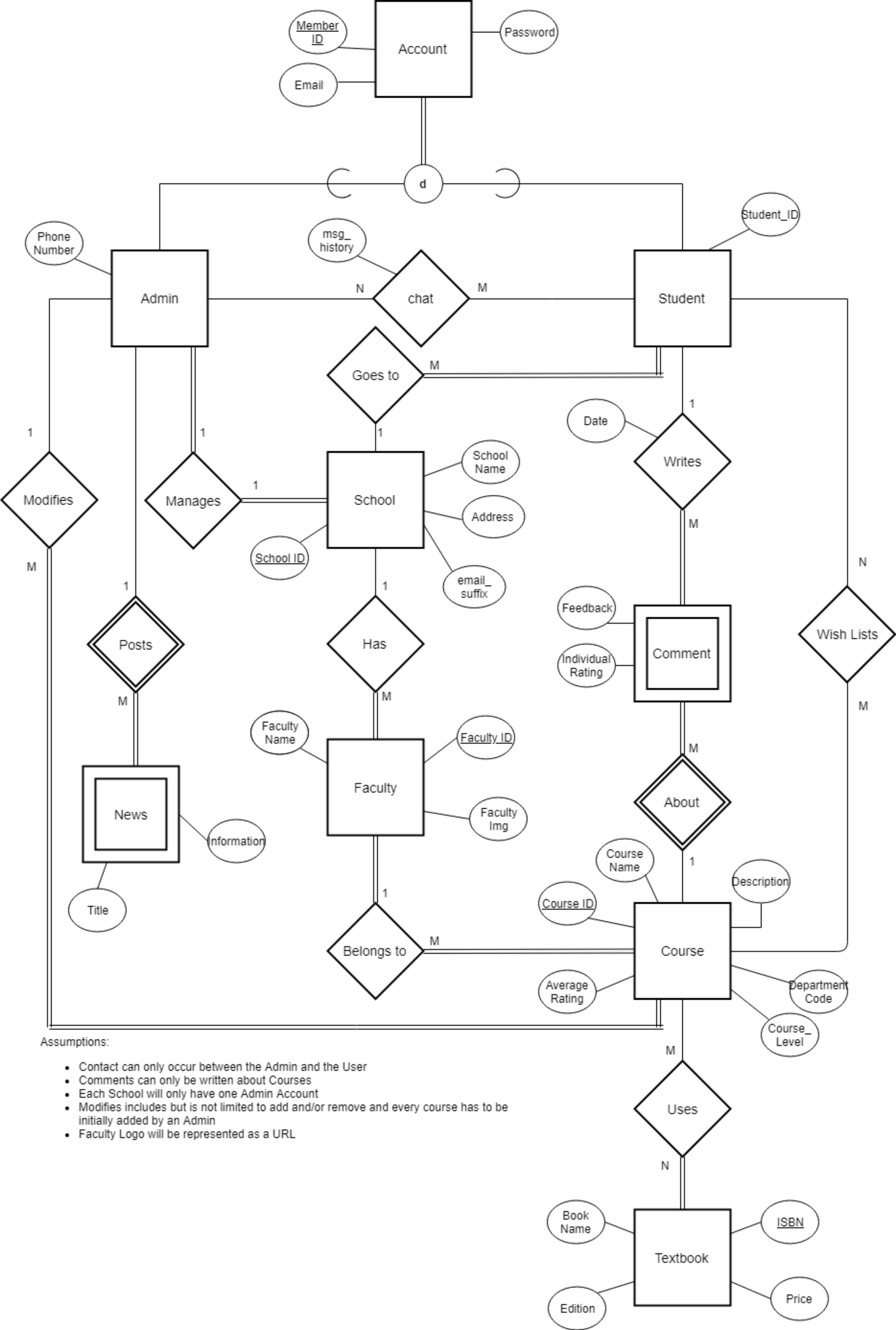
There will be two users for our system, a school admin and the students. The school admin is tasked with managing the majority of the database; adding, removing and editing courses and faculties within their school, posting any news and maintaining all the reports of the students. The student side have less interactions; the majority of what they do is viewing the information from the database and compiling any interest they have in a list, as well as being able to send reports to the admin if needed. Both the student and the admin are able to edit their Select Electives account settings when needed. Interactions between these two accounts occur in the messaging chat system between the two users, where all of the messages are logged and stored in the database. Any changes that the admin make from their end will be reflected in the student side of the web application as well. Altogether, the admin account and the student account create the interactive web application, ‘Select Electives’.

Project Design Diagram Discussion

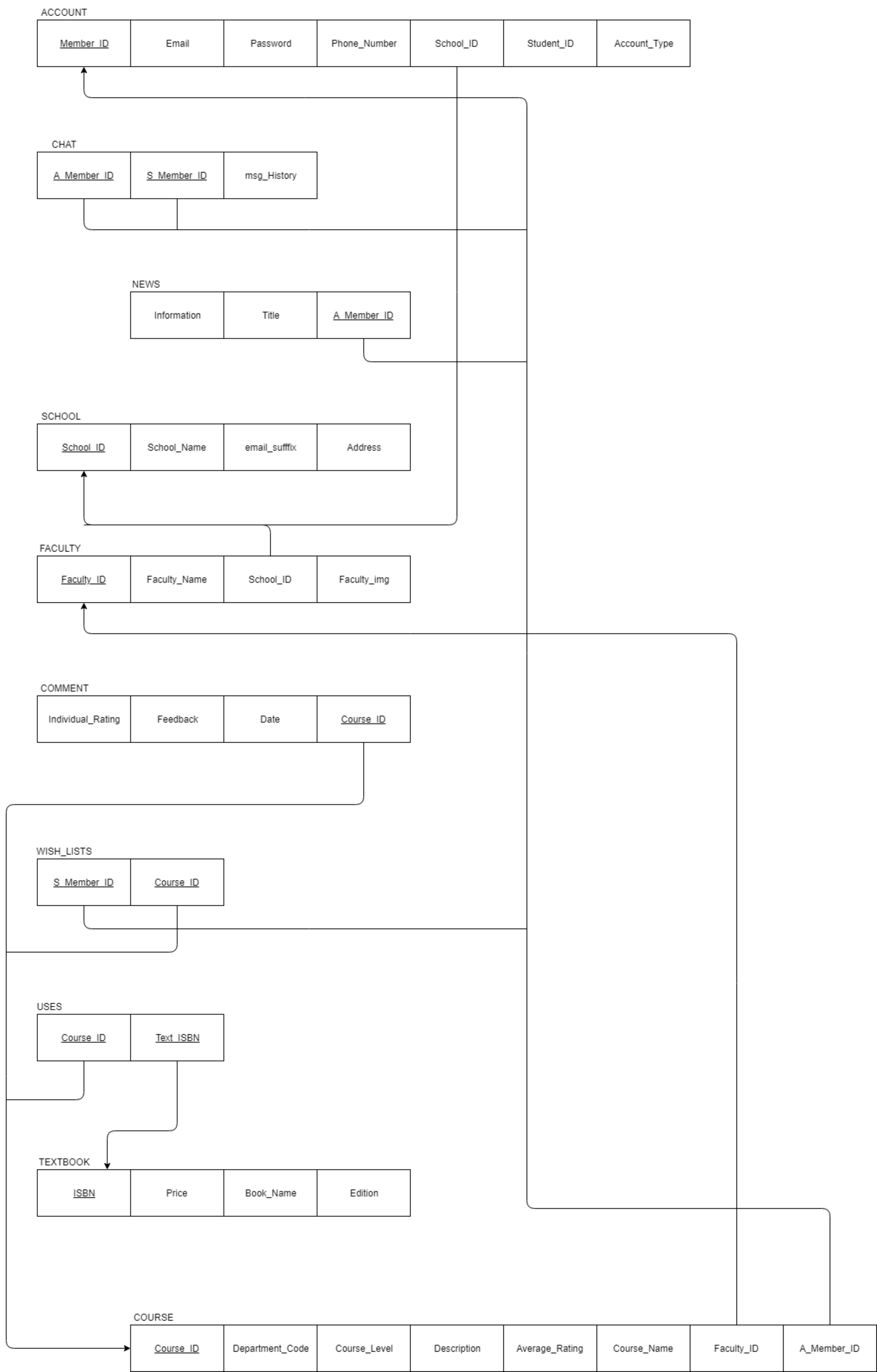
The following diagrams (seen on the following pages) were utilized when creating the project ‘Select Electives’:

- Entity Relationship Diagram (ERD)
- Relation Model Diagram
- HIPO & DFD Diagrams

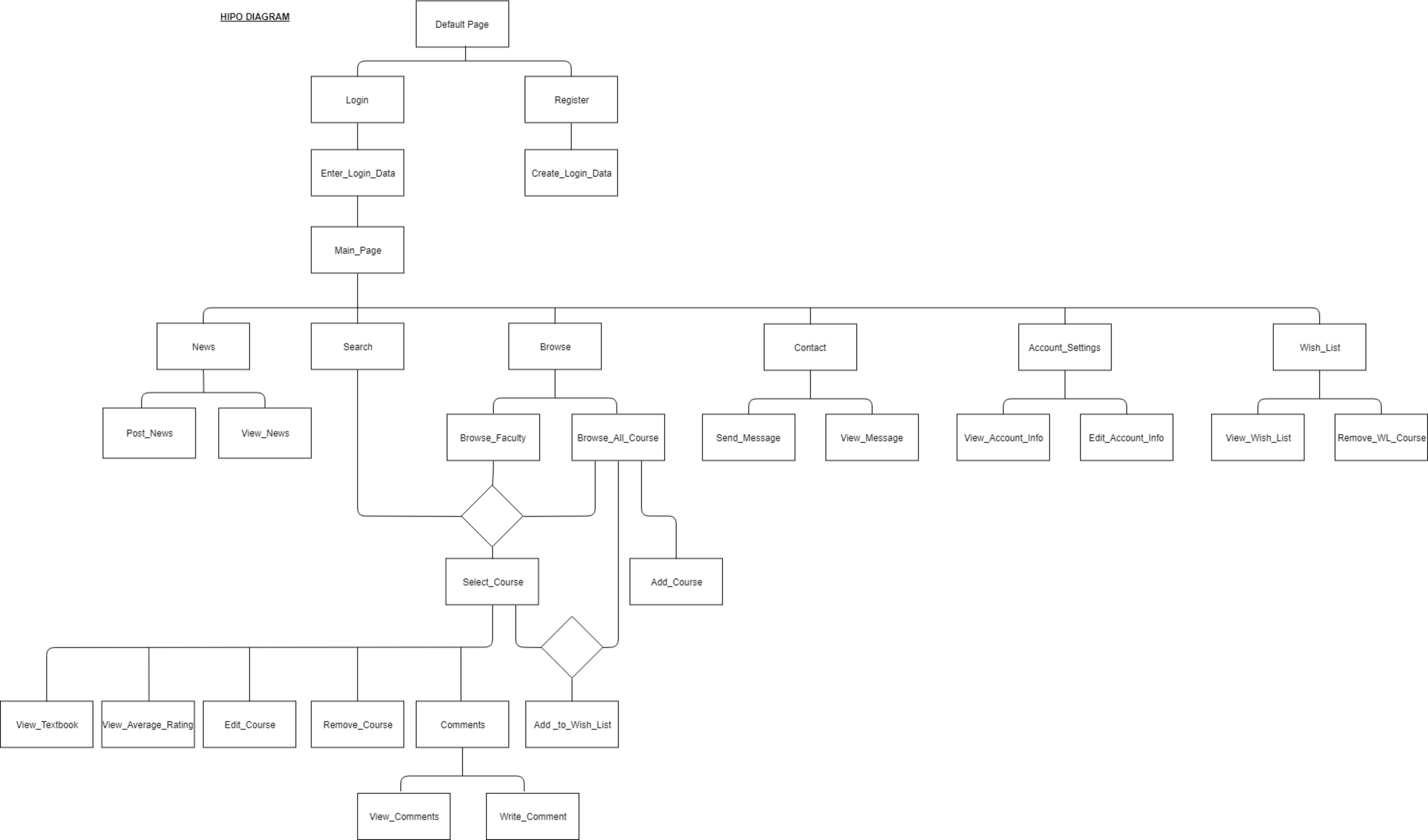
During this process, significant decisions were made to transition from design to implementation. In particular, the chat, wishlist, and uses_textbook relationships needed to be standalone entities in order for the database to communicate the information properly. This led to those relationships to be many-to-many for this feature to occur. Many assumptions were necessary for the diagrams to make sense with the implementation decisions that were made. These assumptions may not have reflected in the diagrams as well as they did in the implementation portion, but they are important when fully executing the web app. The assumptions made are as follows. The chat would occur only between the admin and the users. Comments made for a course would be related to said course. Every school would have one admin account, meaning there would only be one admin per school. The relationship, ‘modifies’ would encompass adding, removing and editing/updating features of the web application. With these assumptions in mind, the implementation of Select Electives can proceed.



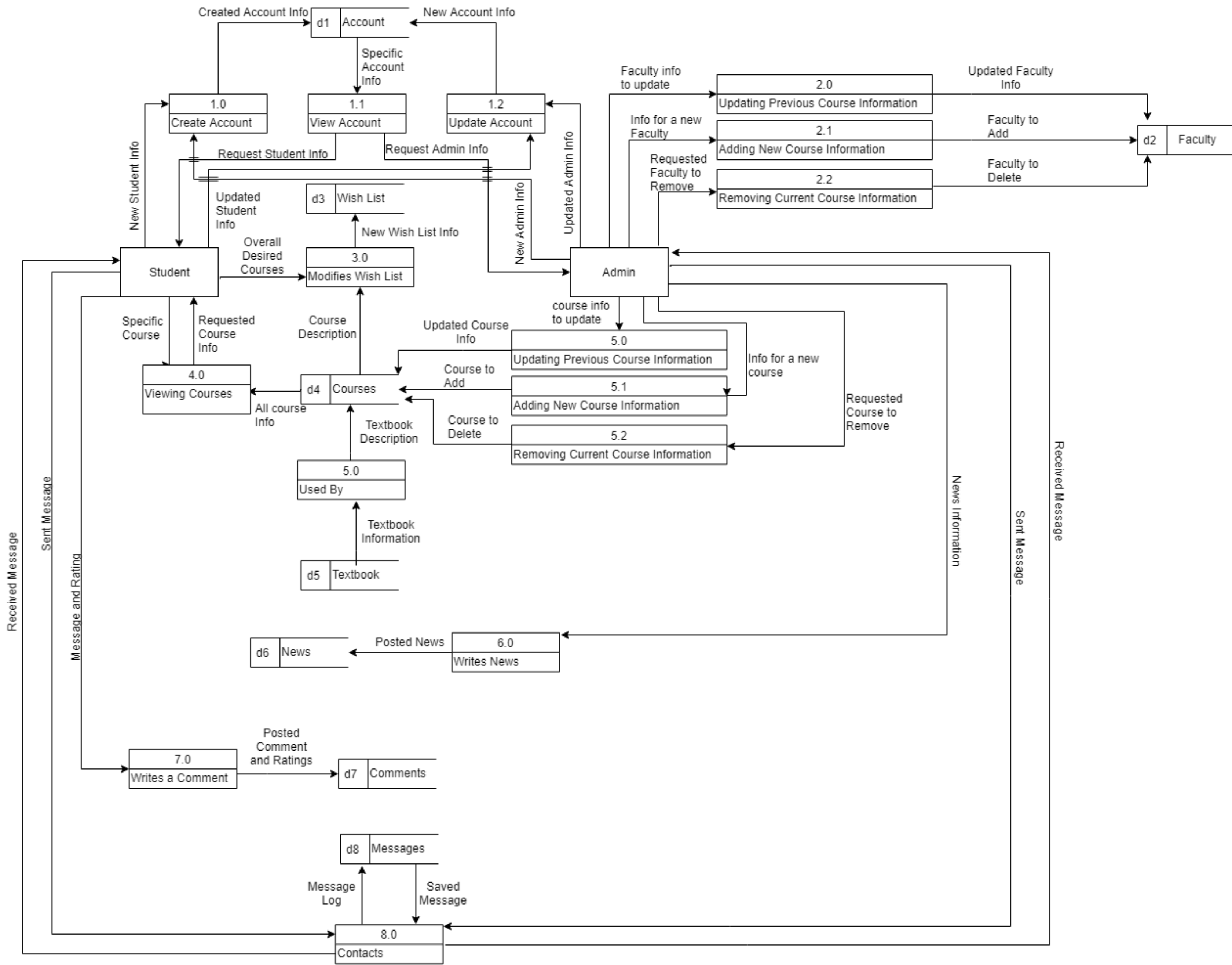
Note: Lines were connected onto single tracks to simplify the visual appearance. Every connection begins at the box with no arrow head and points to the box with an arrow head at its end. Every connection is made between two boxes.



HIPO DIAGRAM



DFD DIAGRAM



DBMS

Our selected DBMS was pgAdmin4. This management tool is used for Postgres, an open source database, and was the platform that met the needs of our group for this project. The application has an easy to use interface that catered to both novice and experienced users like ourselves. This tool, pgAdmin4, is free to install, and provides users with a customizable display where components are easily accessible and not hard to follow. There are scripts available to guide users for the creation of the tables and SQL queries, including a feature that checks a query for any errors that may be found in the format of said query. A user can execute their query within pgAdmin4 to see if the returned value was what they expected it to be. Also, the user can create, edit and delete information from the database directly in pgAdmin4. This management tool provides users with a table display of the data within the database. Considering all these features together, pgAdmin4 has made our database implementation process smooth and efficient.

SQL Queries

```
db.query("SELECT * FROM public.account INNER JOIN public.school ON  
public.account.school_no=public.school.school_id AND public.account.email= $1 AND  
public.account.password= $2",[user,userpass])
```

```
db.query("SELECT * FROM public.account INNER JOIN public.school ON  
public.account.school_no=public.school.school_id AND public.account.member_id=  
$1",[req.session.user])
```

```
db.query("SELECT max(member_id) FROM public.account", [])
```

```
db.query("SELECT max(school_id) FROM public.school", [])
```

```
db.query("SELECT school_id FROM public.school WHERE email_suffix = $1", [emailSuff])
```

```
db.query("SELECT email FROM public.account WHERE email = $1", [req.body.email])
```

```
db.query("INSERT INTO public.account(member_id, email, password, acc_type, student_id,  
school_no, name) VALUES ($1, $2, $3, $4, $5, $6, $7)",  
[nextAccID, req.body.email, req.body.password, req.body.accType, req.body.sID, emailSuffSchoolID, req.body.name])
```

```
db.query("INSERT INTO public.school(school_id, school_name, email_suffix, address)  
VALUES ($1, $2, $3, $4)", [nextSchoolID, req.body.school, emailSuff, req.body.address])
```



```
db.query("INSERT INTO public.account(member_id, email, password, acc_type, phone_no, school_no, name) VALUES ($1, $2, $3, $4, $5, $6, $7)", [nextAccID, req.body.email, req.body.password, req.body.accType, req.body.sID, nextSchoolID, req.body.name])
```

```
db.query("SELECT MAX(faculty_id) FROM public.faculty;", [])
```

```
db.query("SELECT faculty_id FROM public.faculty WHERE faculty_name= $1 AND school_no= $2", [req.body.faculty, req.body.school])
```

```
db.query("INSERT INTO public.faculty(faculty_id, faculty_name, faculty_img, school_no) VALUES ($1, $2, $3, $4)", [nextFacID, req.body.faculty, req.body.img, req.body.school])
```

```
db.query("SELECT MAX(course_id) FROM public.course;", [])
```

```
db.query("SELECT course_id FROM public.course WHERE dept_code= $1 AND course_level= $2 AND admin_id= $3", [req.body.dept, req.body.level, req.body.admin])
```

```
db.query("SELECT faculty_id FROM public.faculty WHERE faculty_name= $1", [req.body.faculty])
```

```
db.query("INSERT INTO public.course(course_id, dept_code, course_level, description, avg_rating, course_name, faculty_no, admin_id) VALUES ($1, $2, $3, $4, $5, $6, $7, $8)", [nextCourseID, req.body.dept, req.body.level, req.body.description, 0, req.body.name, facultyID, req.body.admin])
```

```
db.query("INSERT INTO public.textbook(\"ISBN\", price, book_name, edition) VALUES ($1, $2, $3, $4)", [req.body.isbn1, req.body.price1, req.body.text1, req.body.edition1])
```

```
db.query("INSERT INTO public.uses_textbook(\"text_ISBN\", course_no) VALUES ($1, $2)", [req.body.isbn1, nextCourseID])
```

```
db.query("INSERT INTO public.textbook(\"ISBN\", price, book_name, edition) VALUES ($1, $2, $3, $4)", [req.body.isbn2, req.body.price2, req.body.text2, req.body.edition2])
```

```
db.query("INSERT INTO public.uses_textbook(\"text_ISBN\", course_no) VALUES ($1, $2)", [req.body.isbn2, nextCourseID])
```

```
db.query("SELECT * FROM public.faculty WHERE school_no = $1", [schoolID])
```

```
db.query("SELECT * FROM public.faculty WHERE faculty_id = $1", [facultyID])
```

```
db.query("SELECT * FROM public.faculty WHERE faculty_name = $1 AND school_no = $2", [req.body.faculty, req.body.schoolID])
```

```

db.query("UPDATE public.faculty SET faculty_name= $1, faculty_img= $2 WHERE
faculty_id= $3", [req.body.faculty, req.body.img, facultyID]

db.query("UPDATE public.faculty SET faculty_name= $1, faculty_img= $2 WHERE
faculty_id= $3", [req.body.faculty, req.body.img, facultyID]

db.query("DELETE FROM public.faculty WHERE faculty_id= $1", [req.body.facultyID]

db.query("SELECT A.member_id, A.name, A.student_id FROM public.account AS A INNER
JOIN public.chat AS C ON A.member_id = C.student_id WHERE C.admin_id = $1",
[req.body.user]

db.query("UPDATE public.chat SET msg_history = $1 WHERE student_id= $2 AND
admin_id= $3", [req.body.chatLog, req.body.chatID, req.body.user]

db.query("SELECT * FROM public.chat WHERE student_id = $1", [req.body.user]

db.query("UPDATE public.chat AS CH SET msg_history = $1, admin_id = A.member_id
FROM public.account AS A WHERE CH.student_id= $2 AND A.acc_type = 'admin' AND
A.school_no = $3", [req.body.chatLog, req.body.user, req.body.school]

db.query("SELECT * FROM public.account WHERE acc_type='admin' AND school_no = $1",
[req.body.school]

db.query("INSERT INTO public.chat(student_id, admin_id, msg_history) VALUES ($1, $2,
$3)", [req.body.user, result.rows[0].member_id, req.body.chatLog]

db.query("UPDATE public.chat SET admin_id=null WHERE student_id= $1 AND admin_id=
$2", [req.body.student, req.body.admin]

db.query("SELECT msg_history FROM public.chat WHERE student_id= $1 AND admin_id=
$2", [req.body.chatID, req.body.user]

db.query("SELECT C.msg_history FROM public.chat AS C INNER JOIN public.account AS A
ON C.admin_id = A.member_id WHERE C.student_id= $1 AND A.school_no= $2",
[req.body.user, req.body.school]

db.query("SELECT * FROM public.account WHERE member_id = $1 AND password =
$2", [user, userpass]

db.query("SELECT * FROM public.account WHERE email = $1", [req.body.email]

db.query("UPDATE public.account SET email= $1, password= $2, phone_no= $3, student_id=
$4, name= $5 WHERE member_id= $6",
[req.body.email, userpass, req.body.phone, req.body.studID, req.body.name, req.body.id]

```

```
db.query("UPDATE public.account SET email= $1, password= $2, phone_no= $3, student_id= $4, name= $5 WHERE member_id= $6",  
[req.body.email,userpass,req.body.phone,req.body.studID,req.body.name,req.body.id]  
db.query("SELECT * FROM public.course WHERE public.course.admin_id = $1", [admin_id]
```

```
db.query("SELECT * FROM public.course, public.account WHERE  
public.account.acc_type='admin' AND public.account.school_no= $1 AND  
public.course.admin_id=public.account.member_id", [school_no]
```

```
db.query("DELETE FROM public.course WHERE admin_id= $1 AND course_id = $2",  
[admin_id, course_id]
```

```
db.query("SELECT * FROM public.wishlist WHERE student_no = $1 AND course_no = $2",  
[req.body.user, req.body.course_id]
```

```
db.query("INSERT INTO public.wishlist(student_no, course_no) VALUES ($1, $2)",  
[req.body.user, req.body.course_id]
```

```
db.query("SELECT * FROM public.wishlist INNER JOIN public.course ON  
public.wishlist.course_no = public.course.course_id WHERE public.wishlist.student_no= $1",  
[req.body.user]
```

```
db.query("DELETE FROM public.wishlist WHERE student_no= $1 AND course_no = $2",  
[req.body.user, req.body.course]
```

```
db.query("SELECT * FROM public.course INNER JOIN public.faculty ON  
public.course.faculty_no = public.faculty.faculty_id AND public.course.faculty_no = $1 AND  
public.faculty.school_no= $2", [req.body.faculty_no,req.body.school]
```

```
db.query("SELECT * FROM public.course WHERE course_id= $1", [req.body.course]
```

```
db.query("SELECT * FROM public.textbook, public.uses_textbook WHERE text_isbn = isbn  
AND course_no = $1 ", [req.body.course_id]
```

```
db.query("SELECT * FROM public.comment WHERE course_no= $1", [req.body.course]
```

```
db.query("INSERT INTO public.comment(student_rating, feedback, date, course_no) VALUES  
($1, $2, $3, $4)", [req.body.rating,req.body.comment,req.body.date,req.body.course]
```

```
db.query("UPDATE public.course SET avg_rating = student_rating FROM public.comment  
WHERE student_rating IN (SELECT AVG(student_rating) FROM public.comment WHERE  
course_no = $1 AND student_rating > 0) AND course_id = $2",  
[req.body.course_id,req.body.course_id]
```

```
db.query("SELECT * FROM public.course INNER JOIN public.faculty ON  
public.course.faculty_no = public.faculty.faculty_id WHERE public.course.course_id = $1",  
[req.body.course_id])
```

```
db.query("SELECT * FROM public.textbook, public.uses_textbook WHERE  
public.textbook.isbn = public.uses_textbook.text_isbn AND public.uses_textbook.course_no =  
$1", [req.body.course_id])
```

```
db.query("SELECT * FROM public.course INNER JOIN public.faculty ON  
public.course.faculty_no = public.faculty.faculty_id WHERE public.course.course_id = $1 AND  
public.faculty.faculty_name = $2 ", [req.body.course_id, req.body.faculty])
```

```
db.query("SELECT * FROM public.course WHERE public.course.course_id = $1",  
[req.body.course_id])
```

```
db.query("SELECT * FROM public.course WHERE public.course.dept_code = $1 AND  
public.course.course_level = $2", [req.body.dept, req.body.level])
```

```
db.query("UPDATE public.course SET dept_code=$1, course_level=$2, description=$3,  
course_name=$4, faculty_no=$5 WHERE course_id = $6", [req.body.dept, newLevel,  
req.body.description, req.body.name, newFacID, req.body.course_id])
```

```
db.query("SELECT * FROM public.textbook WHERE isbn= $1", [req.body.isbn])
```

```
db.query("INSERT INTO public.textbook (isbn, price, book_name, edition)  
VALUES($1,$2,$3,$4)", [req.body.isbn, req.body.price, req.body.name, req.body.edition])
```

```
db.query("INSERT INTO public.uses_textbook (text_isbn, course_no) VALUES($1,$2)",  
[req.body.isbn, req.body.course])
```

```
db.query("UPDATE public.textbook SET price= $1, book_name= $2, edition= $3 WHERE  
isbn= $4", [req.body.price, req.body.name, req.body.edition, req.body.isbn])
```

```
db.query("INSERT INTO public.uses_textbook (text_isbn, course_no) SELECT $1, $2  
WHERE NOT EXISTS (SELECT * FROM public.uses_textbook WHERE text_isbn= $3)",  
[req.body.isbn, req.body.course, req.body.isbn])
```

```
db.query("SELECT * FROM public.news AS N, public.account AS A WHERE A.school_no =  
$1 AND N.admin_id = A.member_id AND A.acc_type = 'admin'", [req.body.school_no])
```

```
db.query("SELECT * FROM public.course AS C, public.account AS A WHERE A.school_no =  
$1 AND C.admin_id = A.member_id AND A.acc_type = 'admin'", [req.body.school])
```

```
db.query("INSERT INTO public.news (info, title, admin_id) VALUES($1,$2,$3)",  
[req.body.info, req.body.title, req.body.admin_id])
```

References

Websites:

University of Calgary, “First-Year Options” <https://www.ucalgary.ca/degreeguide/options>
PGAdmin, “pgAdmin 4 4.8 Documentation” <https://www.pgadmin.org/docs/pgadmin4/dev/>

Textbooks:

Elmasri, R. and Navathe, S. (2017). *Fundamentals of database systems*. 1272 Seiten: [Verlag nicht
ermittelbar].

Select Electives

User Manual

Table of Contents

Pg. 1 - Login and Registration

Pg. 2 - Main Page

Pg. 3 - Viewing and Editing Account Info

Pg. 5 - Browsing and Viewing Courses

Pg. 7 - Viewing, Adding and Editing Individual Courses

Pg. 11 - Adding, Editing and Removing Faculties

Pg. 13 - Admin and Student Chat

Pg. 15 - Posting and Viewing News

Pg. 16 - Sample Data Used in the Manual

Login and Registration



SELECT ~~pre reqs~~ ELECTIVES

Select Electives is a web app that provides a database for post secondary electives, classes without any pre-requisite. Providing a way for students to get descriptive information about certain electives, they will be able to rate and comment on each course, as well as saving a wishlist of potential classes they would like to take for future uses.

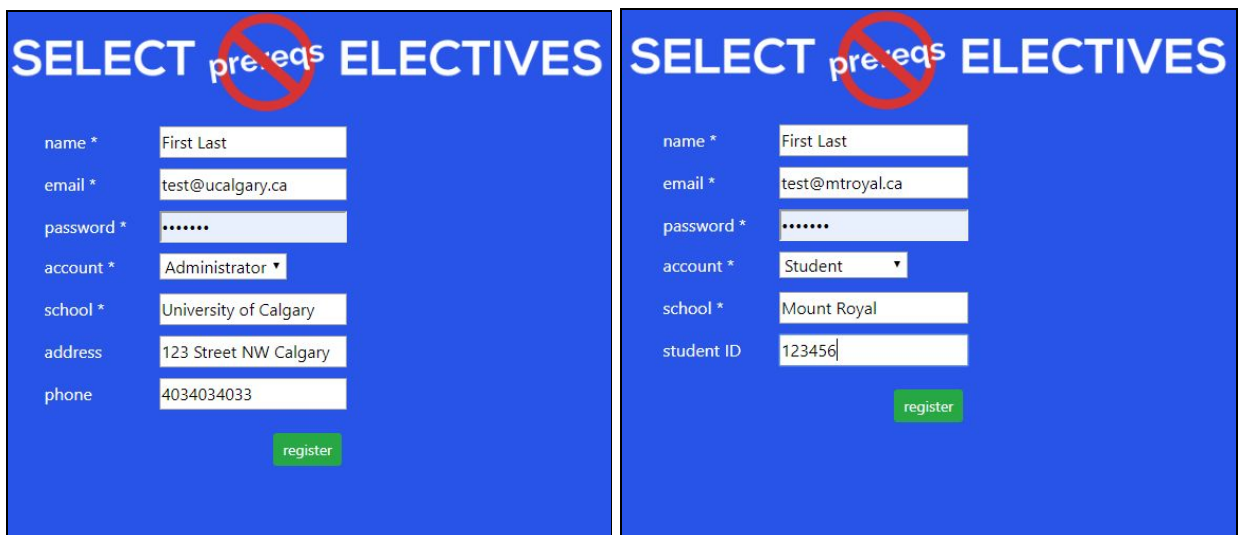
randomemail@ucalgary.ca

.....

login

register

Just a simple Login page, showcases a short blurb of our webapp and provides a login as well as a registration.



SELECT ~~pre reqs~~ ELECTIVES

name * First Last

email * test@ucalgary.ca

password *

account * Administrator

school * University of Calgary

address 123 Street NW Calgary

phone 4034034033

register

SELECT ~~pre reqs~~ ELECTIVES

name * First Last

email * test@mtroyal.ca

password *

account * Student

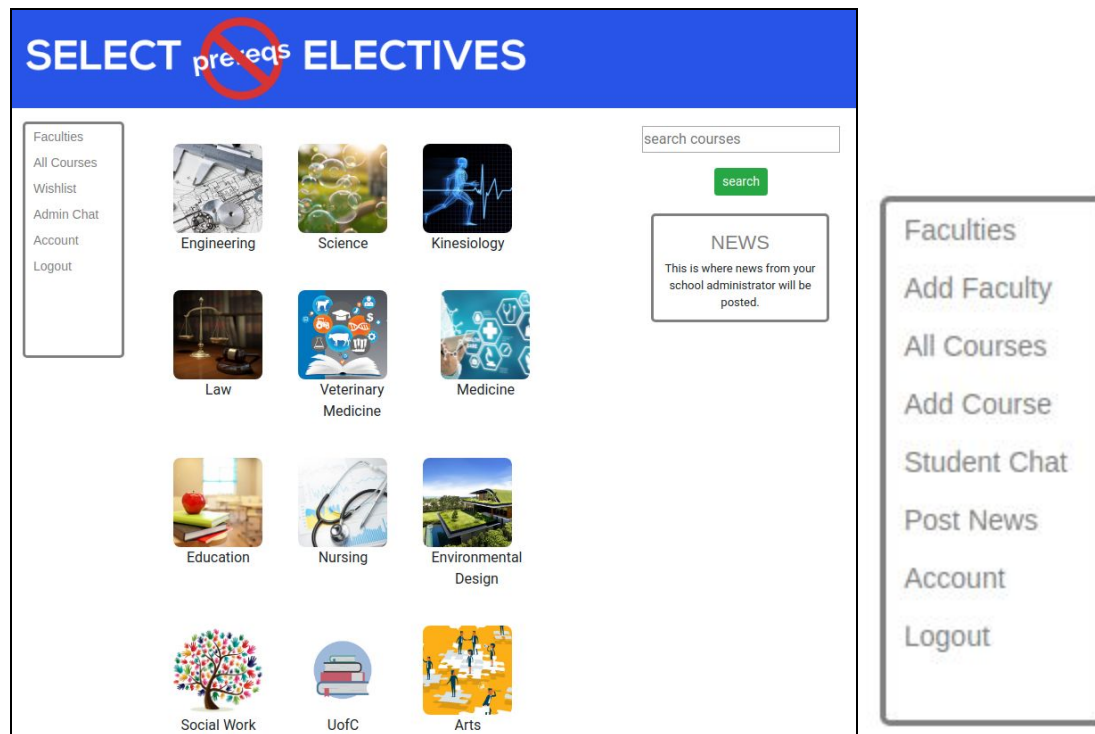
school * Mount Royal

student ID 123456

register

Student and Admin registration. Students have a feature to add their student ID to prove legitimacy, while an administrator can provide their school's phone number and address since there is only one admin account per institution.

Main Page



After logging in, you will initially be brought into your school's faculty page. On the left side of the main page, there is a menu bar which allows you to load up any of the information. The image on the left shows for what it would look like on the student side, while the image on the right is the menu for the Admin. The middle portion is the preloaded information of the selected page, the list of faculties is the default for when a person logs in. Then on the right side of the page, there is any news posted by the admin. This will appear for the students, but not on the admin side. Along with the news there is also the search bar to go directly to a course instead of browsing a large selection for it.

Viewing and Editing Account Info

Faculties

All Courses

Wishlist

Admin Chat

Account

Logout

name *

Random

email *

randomemail@ucalgary.ca

school name *

University of Calgary

student ID

177013

old password *

new password

save changes

Viewing Account Info on Student side

Faculties

Add Faculty

All Courses

Add Course

Student Chat

Post News

Account

Logout

name *

Cristhian

email *

cristhian.soteloplaz@ucalgary.ca

school name *

University of Calgary

phone

9876534

old password *

new password

save changes

Viewing account info on Admin side

name *

email *

school name *

student ID

old password *

new password

account information successfully updated

[save changes](#)

When viewing your account info, you can also change it as long as you have the current (old) password. If the password matches, then saving changes will work and your new account information will be changed and stored.

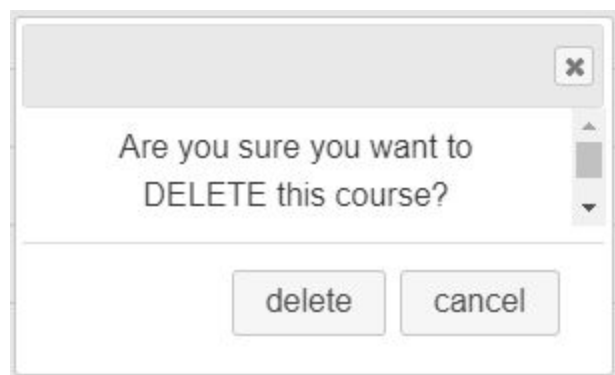
Browsing and Viewing Courses

[Faculties](#)
[Add Faculty](#)
[All Courses](#)
[Add Course](#)
[Student Chat](#)
[Post News](#)
[Account](#)
[Logout](#)

List of All Courses Offered in School

CPSC 217: Intro to cpsc	
EDUC 207: How Children Learn to Read	
EDUC 211: Academic Writing in Education	
EDUC 309: Early Childhood Development	
ART 231: Arts Fundamentals: 2D	
ART 233: Arts Fundamentals: 3D	
ART 235: Intro Photog & Digital Imaging	
NURS 385: Disc & Profession Nursing	
NURS 388: Supporting Health	
NURS 389: Integrate Nurse Role & Practice	
EVDS 401: Introduction to EVDS	
SOWK 201: Introduction to Social Work	
SOWK 302: Research Context Theme	

Clicking the all courses button will bring you to a list of every course in the database based on the account's school. There is not any filtering in this mode. From any of the list views, an admin will have a trash can button on the end of each row, this will enable the admin to remove the selected course. There is also a confirmation popup in case of any accidental clicks.



On the student's side, a + sign will be shown to indicate if they want to add the course to the student's wishlist.

CPSC 217: Intro to cpsc

List of All Courses Offered in Faculty of Science

CPSC 217: Intro to cpssc	+
PHYS 303: Quantum Mysteries & Paradoxes	+
SCIE 331: Scientific Explorations	+
GLGY 305: Introduction to Dinosaurs	+
PHYS 303: Quantum Mysteries & Paradoxes	+

From the Faculty Page on the student side, if you click one of the faculties, it will bring up a list of courses from that one particular faculty.

[Faculties](#)
[All Courses](#)
[Wishlist](#)
[Admin Chat](#)
[Account](#)
[Logout](#)

List of Courses of interest

PHYS 303: Quantum Mysteries & Paradoxes	🗑
CPSC 217: Intro to cpssc	🗑

Clicking on Wishlist will pull up all courses the student added to their wishlist, making it easier to navigate to courses that they were interested in when browsing. Pressing the trash can will remove the course from the wishlist.

Viewing, Adding and Editing Individual Courses

Faculty *	<input type="text" value="Science"/>
Department code *	<input type="text" value="PHYS"/>
Course level *	<input type="text" value="303"/>
Course name *	<input type="text" value="Quantum Mysteries & Paradoxes"/>
Description	<div><p>Aims to explain basic quantum phenomena for students outside the physical sciences. Topics covered may include wave-particle duality, quantum interference, as well as the paradoxes of entanglement and quantum non-locality. Applications such as quantum cryptography and quantum teleportation are discussed, as are the philosophical interpretations of the quantum picture of the world.</p></div>
Textbook 1	
ISBN	<input type="text" value="9780306819643"/>
name	<input type="text" value=", The Quantum Universe"/>
edition	<input type="text" value="13"/>
Textbook 2	
ISBN	<input type="text"/>
name	<input type="text"/>
edition	<input type="text"/>
course successfully added	
<input type="button" value="add course"/>	

On the admin side, when you click on the add course option from the menu, you will be able to add a course by filling out the necessary fields and up to two textbooks.

Aims to explain basic quantum phenomena for students outside the physical sciences. Topics covered may include wave-particle duality, quantum interference, as well as the paradoxes of entanglement and quantum non-locality. Applications such as quantum cryptography and quantum teleportation are discussed, as are the philosophical interpretations of the quantum picture of the world.

Textbook 1

ISBN	<input type="text" value="9234902523"/>
name	<input type="text" value="Quantum Computing"/>
edition	<input type="text" value="2"/>
price	<input type="text" value="80"/>

Textbook 2

ISBN	<input type="text"/>
name	<input type="text"/>
edition	<input type="text"/>
price	<input type="text"/>

On the admin side, when you click on a pre-existing course, you can then get an edit field to change any of the information, including changing textbooks. Click save changes to update the information to the Database.

PHYS 303: Quantum Mysteries & Paradoxes

Aims to explain basic quantum phenomena for students outside the physical sciences. Topics covered may include wave-particle duality, quantum interference, as well as the paradoxes of entanglement and quantum non-locality. Applications such as quantum cryptography and quantum teleportation are discussed, as are the philosophical interpretations of the quantum picture of the world.

COMMENTS



comment

PHYS 303: Quantum Mysteries & Paradoxes

Aims to explain basic quantum phenomena for students outside the physical sciences. Topics covered may include wave-particle duality, quantum interference, as well as the paradoxes of entanglement and quantum non-locality. Applications such as quantum cryptography and quantum teleportation are discussed, as are the philosophical interpretations of the quantum picture of the world.

COMMENTS



06/16/2019 8:38:25 PM

Random: Easiest 4.0, would recommend for that easy boost. Wasn't as much schrodinger cat as I expected.

Rating: 5

comment

Once the course is added you can view the page through any browsing method. On the course page you can view the in depth description, as well as any comments and ratings on the course. By clicking the comment button, a student can add their own comments.

PHYS 303: Quantum Mysteries & Paradoxes

Aims to explain basic quantum phenomena for students outside the physical sciences. Topics covered may include wave-particle duality, quantum interference, as well as the paradoxes of entanglement and quantum non-locality. Applications such as quantum cryptography and quantum teleportation are discussed, as are the philosophical interpretations of the quantum picture of the world.

COMMENTS

06/16/2019 8:38:25 PM

Random: Easiest 4.0, would recommend for that easy boost. Wasn't as much schrodinger cat as I expected.

Rating: 5

Comment

If math is not your forte, it can be pretty difficult.

Rating 3 ▼

submit

PHYS 303: Quantum Mysteries & Paradoxes

Aims to explain basic quantum phenomena for students outside the physical sciences. Topics covered may include wave-particle duality, quantum interference, as well as the paradoxes of entanglement and quantum non-locality. Applications such as quantum cryptography and quantum teleportation are discussed, as are the philosophical interpretations of the quantum picture of the world.

COMMENTS

06/16/2019 8:38:25 PM

Random: Easiest 4.0, would recommend for that easy boost. Wasn't as much schrodinger cat as I expected.

Rating: 5

06/16/2019 8:58:20 PM

Random: If math is not your forte, it can be pretty difficult.

Rating: 3

comment

Note: Random is the account name, in an actual account, it would use a person's real name.

Adding, Editing and Removing Faculties

Faculties

Add Faculty

Sorry, no faculties found :(

Starting from a new school, with a blank faculties list, you can click add Faculties to bring up the page to add new faculties.

Faculties

Add Faculty

All Courses

Add Course

Student Chat

Post News

Account

Logout

Faculty name *

Science

Image URL *

<https://d2ai0ibaxpbki1.cloudfront.net/v2/images/collections/video-mu>

faculty successfully added

add faculty

Faculties

Add Faculty

All Courses


Add Course

Student Chat

Post News

Account

Logout



Science

Clicking on the faculty gives you the ability to edit or remove the faculty

Faculty name *

Science

Image URL *

https://d2ai0ibaxpbki1.cloudfront.net/v2/images/collections/video-mu

remove

save changes

You can edit the information or click remove to just delete the faculty.

Faculty name *

Science

Image URL *

https://d2ai0ibaxpbki1.cloudfront.net/v2/images/c

remove

save changes

remove faculty?

remove

cancel

Admin and Student Chat

Faculties
All Courses
Wishlist
Admin Chat
Account
Logout

Message input field: Hello, where can I find registration dates? send

Faculties
All Courses
Wishlist
Admin Chat
Account
Logout

Kevin Student: Hello, where can I find registration dates?

Message input field: send

Clicking the Admin Chat button as a student, you will immediately be brought up to a typical chat page, where sending the message will bring it to the admin of the school.



On the Admin side when you click student chat, you will be given a list of students that have sent a message, you can also delete messages to reduce clogging, a student resending a message will bring back a deleted log with all previous messages included.



After clicking on the message, you can chat on it just like on the student side.

Posting and Viewing News

[Faculties](#)
[Add Faculty](#)
[All Courses](#)
[Add Course](#)
[Student Chat](#)
[Post News](#)
[Account](#)
[Logout](#)

News Title

YOGA UPDATE

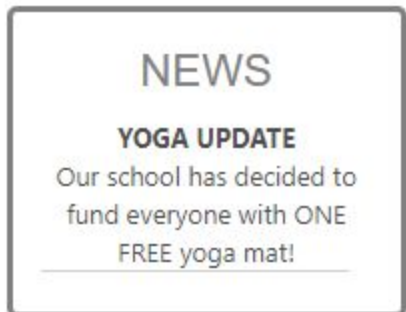
News Information

Our school has decided to fund everyone with ONE FREE yoga mat!

news posted successfully

post news

Clicking on the post news button as an admin will allow you to post any important announcement that you want on the side of the page for every student to view. Once you have successfully posted a news, it will appear like the image below for students.



Appendix - Sample Data Used in the Manual

account

	member_id [PK] integer	email text	password text	acc_type text	phone_no text	student_id text	school_no integer	name text
1	4	anotherone@ucalgary...	yeah	student	[null]	1654322	1	Another One
2	6	kevin@ucalgary.ca	password	student	[null]	177013	1	Kevin Student
3	7	royalstud@mtroyal.ca	royal	student	[null]	5677654	3	Royal Stud
4	1	cristhian.soteloplaz@...	cpsc471	admin	9876534		1	Cristhian
5	8	saitstudent@edu.sait...	sait	student	[null]	000	2	saitstudent
6	5	randomemail@ucalgar...	12345	student		177013	1	Random
7	2	kat@edu.sait.ca	kat	admin	1234566789		2	Katrina Chanco
8	3	kevin.ng@mtroyal.ca	password	admin	4034034034		3	Kevin

chat

	student_id integer	admin_id integer	msg_history text[]
1	6	1	{"Kevin Student: Hello, where can I find registration dates?","..."}
2	4	1	{hello,"whats up"}

comment

	student_rating integer	feedback text	date text	course_no integer
1	5	Random: Easiest 4.0, woul...	06/16/2019 8:38:25 PM	29
2	3	Random: If math is not you...	06/16/2019 8:58:20 PM	29
3	1	saitstudent: Trash course I...	06/17/2019 11:19:47 AM	34
4	1	Random: Big Oof. That is al...	06/17/2019 12:24:58 PM	29

course

	course_id [PK] integer	dept_code text	course_level integer	description text	avg_rating double precision	course_name text	faculty_no integer	admin_id integer	
1	3	EDUC	207	Understand t...	0	How Children Le...	7	1	▲
2	4	EDUC	211	Understand h...	0	Academic Writin...	7	1	
3	6	ART	231	Understand t...	0	Arts Fundament...	8	1	
4	7	ART	233	Understand t...	0	Arts Fundament...	8	1	
5	8	ART	235	Learn to take ...	0	Intro Photog & D...	8	1	
6	9	NURS	385	Discuss abou...	0	Disc & Professio...	9	1	
7	10	NURS	388	Learn about h...	0	Supporting Health	9	1	
8	11	NURS	389	Practice nursi...	0	Integrate Nurse ...	9	1	
9	12	EVDS	401	Basic introdu...	0	Introduction to E...	10	1	
10	13	SOWK	201	Basic introdu...	0	Introduction to S...	11	1	
11	14	SOWK	302	Learn how to ...	0	Research Conte...	11	1	
12	15	SOWK	303	Get some indi...	0	Practice Individu...	11	1	
13	16	UNIV	401	Interdisciplina...	0	Topic Interdiscip...	12	1	
14	17	UNIV	501	Understandin...	0	Experimental Le...	12	1	
15	18	MDSC	408	Yep, somethi...	0	Computation for...	6	1	
16	19	MDSC	507	Figure out the...	0	Spec Prob in Me...	6	1	
17	20	VETM	300	Learn some c...	0	Clinical Skills I	5	1	▼

faculty

	faculty_id [PK] integer	faculty_name text	faculty_img text	school_no integer
1	1	Engineering	https://www.n...	1
2	2	Science	https://d2ai0i...	1
3	3	Kinesiology	https://pbs.tw...	1
4	4	Law	http://village.p...	1
5	5	Veterinary Medi...	https://blogs....	1
6	6	Medicine	https://www.i...	1
7	7	Education	https://cdn20....	1
8	9	Nursing	https://demed...	1
9	10	Environmental D...	https://ariahe...	1
10	11	Social Work	https://blogs....	1
11	12	UofC	https://www.fl...	1
12	8	Arts	https://sloanr...	1
13	13	Yoga	https://www.e...	2
14	14	Science	https://d2ai0i...	3
15	15	Arts	https://sloanr...	3
16	16	Nursing	https://demed...	3

news

	info text	title text	admin_id integer
1	Remember to register ea...	Registration FALL 2019	1
2	THE NEWS POSTING WO...	EXCITING NEWS	1
3	Our school has decided t...	YOGA UPDATE	2

school

	school_id [PK] integer	school_name text	email_suffix text	address text
1	1	University of Calgary	@ucalgary.ca	2500 University Drive NW, Calgary, Alberta, C...
2	2	SAIT	@edu.sait.ca	1301 16 Ave NW, Calgary, Alberta, CANADA ...
3	3	Mount Royal University	@mtroyal.ca	4825 Mount Royal Gate SW, Calgary, Alberta,...

textbook

	isbn [PK] text	price text	book_name text	edition text
1	102312	120	Quantum Mysterie...	3
2	420420	420	Big Arts boy	3

uses_textbook

	text_isbn text	course_no integer
1	102312	35
2	420420	36

wishlist

	student_no integer	course_no integer
1	5	29
2	5	1
3	4	4
4	4	5
5	4	30
6	4	1
7	4	3