

**Web-Based Dynamic Registration System
For
Maple Leaf University
System Design and Implementation CS 5910-HY2
SUNY Old Westbury
Dr. Naresh Gupta**

Team Leader: Emiz Intriago: eintriag@oldwestbury.edu (631) 524-6448

Yesenia Fernandez: yferna16@oldwestbury.edu (516) 713-2249

Anibal Ruiz: aruiz23@oldwestbury.edu (516) 725-6020

Table of contents

Team Meeting Log	1
Problem Statement	2
System Required Specifications	2
Student	2
Explicitly Required	2
View	2
Search	3
Action	4
Explicitly Forbidden	4
Faculty	5
Explicitly Required	5
View	5
Actions	5
Search	6
Explicitly Forbidden	6
Admin	7
Explicitly Required	7
View	7
Search	7
Action	8
Explicitly Forbidden	9
Researcher	9
Explicitly Required	9
Explicitly Forbidden	9
Visitor	10
Explicitly Required	10
Explicitly Required	10
Use Case	11
Student	11
Successful log in case	11
Unsuccessful log in case	11
Successful case to RESET PASSWORD	12
Unsuccessful case to RESET PASSWORD	12
View Personal Information	13
View Registration	14
View Holds	14

View Current Schedule	15
View Semester Schedule	15
View Academics	16
View Graduation Audit	17
View Advisor	17
View Advisor (Current)	18
View Master schedule	18
View All Majors	19
View All Major Requirements	19
View All Minor	20
View All Minor Requirements	21
View Prerequisite	21
View All Courses	22
View All Semesters	23
Successful Registering Student to a Course	23
Successful Dropping a course as a Student	24
Unsuccessful Registration of a course (Exceeding Credit Limit)	24
Unsuccessful Registration of a course (No Prerequisites for the Course)	25
Unsuccessful Registration of a course (Time conflict with two courses)	25
Unsuccessful Registration of a course (Passing Time Limit)	26
Unsuccessful Registration of a course (Has a Hold)	27
Unsuccessful Registration of a course (Undergraduate registering to a Graduate Course)	27
Unsuccessful Registration of a course (Graduate registering to an undergraduate Course)	28
Faculty	28
Successful Login	29
Unsuccessful login	29
Successful login - a password Reset	30
View Class List	30
View Schedule	31
Grade Student	31
Grade student - time window	32
Take attendance	33
Take attendance - time window	33
View all Majors	34
View Major Requirements	35
View all Minors	35
View Minor requirements	36
View Students in class list	37
View all classes faculty is teaching	37
View all Courses	38
View all Departments	38

View buildings	39
View Academic calendar	39
View user Information	40
View Master schedule	40
View all semesters	41
View all Sections	42
View Advisors	42
Admin	42
Unsuccessful log in case	43
Successful case to RESET PASSWORD	43
Unsuccessful case to RESET PASSWORD	44
Administrators view all students case	45
Administrators view all faculty case	45
Administrators view all researchers	46
Administrators view all administrators	46
Administrator view all departments	47
Administrators view all majors	47
Administrators view all minors	48
Administrators view all courses	48
Successfully searching Prerequisites for a course	49
Unsuccessfully searching Prerequisites for a course (course does not exist)	49
Successfully searching a course - no prerequisites	50
Successfully editing a course	50
Unsuccessfully editing a course - time conflict - time and room	51
Unsuccessfully editing a course - schedule conflict	52
Successfully editing a course (professor)	53
Unsuccessfully editing a course - time conflict to another course - professor	53
View major requirements	54
View all major by department	54
View minor requirements	55
View all minor by department	55
View student grade and attendance	56
Add major to a student	56
View all buildings	57
Successfully add prerequisite to a course	58
Successfully remove a prerequisite from a course	58
Successfully remove a course	59
Successfully adding a course	59
Unsuccessfully adding a course - same course name	60
Successfully adding a section for a course	61
Unsuccessfully adding a section for a course - time conflict	61

Successfully adding an user	62
Unsuccessfully adding an user - same email for example	62
Successfully remove a user	63
unsuccessfully remove a user - faculty teaching a course	64
View locked users	64
Unlock users	64
Register a student to a course	65
View a student schedule	65
View a faculty schedule	66
View a faculty department	66
Successfully adding a department to faculty	67
View all advisors	67
View students holds	68
View student transcript	68
View student major	69
View user statistics	70
Researcher	70
Successful Login	70
Unsuccessful login	71
Successful login - a password Reset	71
View all Majors	72
View all Minors	72
View all Courses	73
View all buildings	73
View Academic calendar	74
View user Information	74
Edit user information	75
View Master schedule	75
View User statistics	76
View performance Statistic	77
Visitor	77
View Master schedule	77
View all Courses.	78
View all Majors and programs	78
View all Minors	79
View Academic calendar	79
Appendix A	79

Team Meeting Log

Team Members:

Date	Start Time	End Time	Notes	Yesenia Fernandez	Anibal Ruiz	Emiz Intriago
8/31/2022	3:50 pm	5:20 pm	Team Formed	✓	✓	✓
9/07/2022	3:50 pm	5:20 pm	Discussion of Project	✓	✓	✓
9/12/2022	3:50 pm	5:20 pm	Discussion of First Draft Outline	✓	✓	✓
9/14/2022	3:50 pm	5:20 pm	Discussion of Project	✓	✓	✓
9/15/2022	6:00 pm	7:00 pm	Team Discussion	✓	✓	✓
9/19/2022	3:50 pm	5:20 pm	Use cases and First Draft set-up	✓	✓	✓
9/27/2022	6:00 pm	8:30 pm	First Draft editing	✓	✓	✓
10/03/2022	3:50 pm	5:20 pm	Second Draft Review	✓	✓	✓
10/04/2022	6:30 pm	7:30 pm	Team Discussion	✓	✓	✓
10/11/2022	6:30 pm	7:30 pm	Uml and Use case Discussion	✓	✓	✓
10/17/2022	5:30 pm	6:30 pm	SRS and Databases	✓	✓	✓

10/18/2022	6:30pm	7:30pm	MySQL	✓	✓	✓
10/24/2022	5:30pm	6:45pm	Data Sheet and MySQL Database	✓	✓	✓
10/25/2022	6:30pm	7:30pm	Data Sheet	✓	✓	✓
11/01/2022	6:30pm	7:30pm	Website Development	✓	✓	✓
11/02/2022	3:50pm	5:20pm	Team Discussion	✓	✓	✓
11/08/2022	6:30pm	7:30pm	Final Editing of College Website	✓	✓	✓
11/09/2022	3:50pm	5:20pm	Team Discussion	✓	✓	✓
11/15/2022	6:30pm	7:30pm	Discussion of Course Catalog	✓	✓	✓
11/16/2022	3:17pm	4:30pm	Meeting with Professor and Team Discussion	✓	✓	✓
11/22/2022	6:30pm	7:30pm	Website Editing and Course Catalog	✓	✓	✓

11/23/2022	3:30pm	4:30pm	Meeting with Professor and Team Discussion	✓	✓	✓
11/28/2022	3:17pm	4:00pm	Team Discussion	✓	✓	✓
11/29/2022	6:00pm	7:00pm	User Manual Discussion	✓	✓	✓
11/30/2022	3:17pm	4:00pm	Meeting with Professor and Team Discussion	✓	✓	✓
12/5/2022	4:00pm	5:00pm	Team Discussion	✓	✓	✓
12/7/2022	4:00pm	5:00pm	Meeting with Professor and Team Discussion	✓	✓	✓
12/12/2022	3:45pm	4:20pm	Meeting with Professor and Team Discussion	✓	✓	✓

Problem Statement

The main goal of our project is to design a web-based dynamic registration system for Maple Leaf University that will provide the functional requirements for students, admin, faculty, visitor and researcher. On the homepage, users will view the menu to navigate through the website, view the academic calendar, view majors and minors, and view courses and their requirements.

The users of the system depending on the selected category will be given different tools to change and update information. In addition, the users are verified based on their email and password. The email and

passwords are checked and referenced to the database. If the email is referenced correctly but the password does not match then the login would not be able to authenticate. If the authentication fails 3 times consecutively then the account would be locked out.

System Required Specifications

Student

Explicitly Required

- ❖ Can be undergraduate/Graduate.
- ❖ Can be Full time/Part time.

View

- ❖ View personal information
- ❖ View master schedule for a given semester (Fall 2022 and Spring 2023)
- ❖ View the class schedule
- ❖ View courses offered by department
- ❖ View unofficial Transcript
- ❖ View semester grades
- ❖ View degree Audit
- ❖ View Major(s)
- ❖ View Minor(s)
- ❖ View Major Requirements
- ❖ View Minor Requirements
- ❖ View Holds
- ❖ View Advisor(s)

- ❖ View course Prerequisite(s)

Search

- ❖ Search the Master Schedule
- ❖ Search by CRN
- ❖ Search by Department name
- ❖ Search by Course name
- ❖ Search by section
- ❖ Search by professor name
- ❖ Search by Time_Slot
- ❖ Search by Room Number
- ❖ Search by available seats
- ❖ Search school catalog

Action

- ❖ Update personal information
- ❖ Add a major
- ❖ Add a minor
- ❖ Drop a major
- ❖ Drop a minor
- ❖ Change a major
- ❖ Change a minor
- ❖ add a course-section to their own schedule
- ❖ drop a course-section to their own schedule

Explicitly Forbidden

- ❖ Cannot modify ID or Email
- ❖ Cannot add course-sections with Holds (Finance, Health, Disciplinary, or Academic)
- ❖ Cannot add a course-section to schedule if prerequisite requirements aren't met
- ❖ Cannot go above or below the Credit Limit when full time (12 credits)
- ❖ Cannot go above or below the Credit Limit when partl time (4 credits)
- ❖ Cannot add Course-section with time_slot conflict in their schedule
- ❖ Cannot take a course-section that they have previously taken before and received a passing grade
- ❖ Cannot add a course-section if Seat Availability reaches max
- ❖ Cannot take a Graduate course-section if you are an undergraduate
- ❖ Cannot take an undergraduate course if you are a graduate
- ❖ Cannot take more than two majors
- ❖ Cannot take more than one major and one minor
- ❖ Cannot view anyone else's info

Faculty

Explicitly Required

- ❖ Can be full time or part time.
- ❖ Can belong to one or more departments.

View

- ❖ Can view everything about a student.
 - View holds
 - View major

- view minor
 - View transcripts
 - View degree audit
 - View advisees
 - View their classlist
-
- ❖ View the semester class schedule for a given semester.
 - ❖ View the classlist for a course-section that he/she is teaching.
 - ❖ View buildings.

Actions

- ❖ Update personal information
- ❖ Faculty can assign final grades.
- ❖ Faculty can assign midterm grades,
- ❖ Keep attendance: a student can be only marked the day of class.

Search

- ❖ Search the Master Schedule
- ❖ Search by CRN
- ❖ Search by Department name
- ❖ Search by Course name
- ❖ Search by section
- ❖ Search by professor name
- ❖ Search by Time_Slot
- ❖ Search by Room Number
- ❖ Search by available seats

Explicitly Forbidden

- ❖ Cannot change final grade if the time window is closed.
- ❖ Cannot change midterm if the time window is closed.
- ❖ Cannot modify anything about a student's information.
- ❖ Cannot teach courses in departments other than their own.
- ❖ Cannot teach two course-sections at the same time.
- ❖ Cannot teach more than two course-sections if full time.
- ❖ Cannot teach more than one course if part time.
- ❖ Can not add or drop students from their courses.
- ❖ Cannot take attendance before or after the class.

Admin

Explicitly Required

- ❖ Can be only Full time

View

- ❖ View all users
- ❖ View all users information
- ❖ View the master schedule
- ❖ View major
- ❖ View minor
- ❖ View major requirements
- ❖ View minor requirements
- ❖ View departments
- ❖ View buildings

- ❖ View rooms
- ❖ View student class schedule
- ❖ View faculty class schedule
- ❖ View the class list that the faculty is teaching.
- ❖ View course prerequisites
- ❖ View professors in a department
- ❖ View courses in the department

Search

- ❖ Search any user by ID
- ❖ Search the master schedule
- ❖ Search by CRN
- ❖ Search by Department name
- ❖ Search by Course name
- ❖ Search by faculty name
- ❖ Search by Time_Slot
- ❖ Search by Room Number
- ❖ Search by available seats
- ❖ Search course prerequisites

Action

- ❖ Can create users
- ❖ Can update users
- ❖ Can Delete users
- ❖ Can create master schedule

- ❖ Can update master schedule
- ❖ Can add student to course-section
- ❖ Can Drop a student from the course
- ❖ Change a grade if requested by faculty
- ❖ Can Update/change student holds
- ❖ Create a new course
- ❖ Can delete a course
- ❖ Can Add Department faculty
- ❖ Can Delete Department faculty
- ❖ Can Update Department faculty
- ❖ Can Add course sections
- ❖ Can Delete course sections
- ❖ Can Update course sections
- ❖ Change minor requirements
- ❖ Change major requirements
- ❖ Create minor
- ❖ Create major

Explicitly Forbidden

- ❖ Two course-section cannot be held in the same time_slot and in the same room
- ❖ Faculty can only teach in their own department
- ❖ Cannot make students go above or below the Credit Limit when part time (8 credits)
- ❖ Cannot make students go above or below the Credit Limit when full time (12 credits)
- ❖ Cannot add course-sections to a student schedule if they have Holds (Finance, Health, Disciplinary, or Academic)

- ❖ Cannot add a course-section to student schedule if prerequisite requirements aren't met

Researcher

Explicitly Required

- ❖ All information that researchers view will be anonymous.
- ❖ View performance statistics.

Explicitly Forbidden

- ❖ Researchers cannot access faculty information.
- ❖ Researchers cannot access student information.
- ❖ Researchers cannot access administrator information.

Visitor

- ❖ Can browse the homepage

Explicitly Required

- ❖ View the academic calendar.
- ❖ View the homepage.
- ❖ View the courses.
- ❖ View the university catalog.
- ❖ Can search the master schedule.

Explicitly Required

- ❖ Visitors cannot register for courses if they are not students.

Use Case

Student

Successful log in case

Use case	Login (successful login)
Actor	Student
Initial condition	1. The student attempts to login into the system
Flow of events	<ol style="list-style-type: none"> 1. The student logs into the system with their email and password 2. The student successfully logs in 3. The student logs out of the system
Exit condition	1. The student logs out and exits the system

Unsuccessful log in case

Use case	Login(Unsuccessful Login)
Actor	Student
Initial Condition	1. The student attempts to log in into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student attempt to login into the system 2. The student's email or password is incorrect 3. The student does not login into the system
Exit Condition	1. The student returns to home page

Successful case to RESET PASSWORD

Use case	Login(Successful Login - password reset)
Actor	Student
Initial Condition	<ol style="list-style-type: none"> 1. The student will attempt to login into the system from the login page but fails three times
Flow of Events	<ol style="list-style-type: none"> 1. The student attempts to login into the system 2. Student's information is not valid (repeated 3 times) 3. The student is locked out of her/his account and clicks on reset password button. 4. Student gets a message to validate their email address that is associated with their account in order to generate a code that will enable them to reset their password 5. The student uses the code to reset the password 6. The student creates a new password and uses that new password to login 7. The student can login into the system successfully
Exit Condition	<ol style="list-style-type: none"> 1. The student logs out and exits the system

Unsuccessful case to RESET PASSWORD

Use case	Login(Unsuccessful Login - password reset)
Actor	Student

Initial Condition	<ol style="list-style-type: none"> 1. The student attempts to login into the system but fails to do it three times
Flow of Events	<ol style="list-style-type: none"> 1. The student attempts to login into the system 2. Student's information is not valid (repeated 3 times) 3. The student is locked out of her/his account and clicks on reset password button. 4. Student gets a message to validate their email address that is associated with their account in order to generate a code that will enable them to reset their password 5. The student enters the incorrect email address 6. The student is locked out of their account 7. The student does not login into the system
Exit Condition	<ol style="list-style-type: none"> 1. The student returns to the homepage

View Personal Information

Use case	View Personal Information
Actor	Student
Initial Condition	<ol style="list-style-type: none"> 1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student goes to menu tab and clicks on Students information 3. The student views their personal information that involves their full name, date of birth, address, email, phone number, emergency contact, a picture that

	<p>appears on their photo ID, and additional information</p> <p>4. The student will log out of the system</p>
Exit Condition	<p>1. The student logs out and exits the system</p>

View Registration

Use case	View Registration
Actor	Student
Initial Condition	<p>1. The student will login into the system</p>
Flow of Events	<p>1. The student logs into the system</p> <p>2. The student goes to menu tab and clicks on registration</p> <p>3. The student views their registration information that includes registration status, holds, add/drop classes, master schedule, and so on.</p> <p>4. The student will log out of the system</p>
Exit Condition	<p>1. The student logs out and exits the system</p>

View Holds

Use case	View Registration (Holds)
Actor	Student
Initial Condition	<p>1. The student will login into the system</p>
Flow of Events	<p>1. The student logs into the system</p> <p>2. The student goes to menu tab and selects Registration</p> <p>3. The student goes to view holds link</p>

	<ol style="list-style-type: none"> The student views the holds they have in their account, whether it being academically, financially, health, disciplinary or none (good standing) The student will log out of the system
Exit Condition	<ol style="list-style-type: none"> The student logs out and exits the system

View Current Schedule

Use case	View student information (Current Schedule)
Actor	Student
Initial Condition	<ol style="list-style-type: none"> The student will login into the system
Flow of Events	<ol style="list-style-type: none"> The student logs into the system The student goes to menu tab and clicks on registration option for students The student views their registration information The student clicks on schedule by day & time link The student views their current schedule for the semester The student will log out of the system
Exit Condition	<ol style="list-style-type: none"> The student logs out and exits the system

View Semester Schedule

Use case	View Schedule (Semester)
Actor	Student
Initial Condition	<ol style="list-style-type: none"> The students will login into the system

Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student goes to menu tab clicks on registration option 3. The student views their registration information 4. The student goes to detail schedule link 5. The student views their current schedule for the semester 6. The student selects the current term (fall/spring/winter/summer session) for the semester 7. The student views their current schedule for the semester 8. The student will log out the system
Exit Condition	<ol style="list-style-type: none"> 1. The student logs out and exits the system

View Academics

Use Case	View Academics
Actor	Student
Initial Condition	<ol style="list-style-type: none"> 1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student goes to the menu tab and selects academic option 3. The student has the option to view grades, degree audit, advisor and so on 4. The student logs out of the system
Exit Condition	<ol style="list-style-type: none"> 1. The student logs out and exits the system

View Graduation Audit

Use case	View Graduation audit
Actor	Student
Initial Condition	1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student goes to menu tab and selects academics option 3. The student goes to degree audit link 4. The student chooses the undergraduate option for their degree audit 5. The student views all the credits they have received 6. The student views the credits they still need to graduate 7. The student will log out of the system
Exit Condition	1. The system logs out and exits the system

View Advisor

Use case	View advisor
Actor	Student
Initial Condition	1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student goes to menu tab and selects academic 3. The student clicks on support & services link 4. The student goes to advising center option 5. The student goes to meet the advisors option 6. The student can view all advisors

	7. The student will log out of the system
Exit Condition	1. The student logs out and exits the system

View Advisor (Current)

Use case	View Current advisor
Actor	Student
Initial Condition	1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student goes to menu tab and selects academic 3. The student clicks the support & services link 4. The student goes to advising center option 5. The student goes to find my advisor tab 6. The student will view their current faculty advisor 7. The student will log out of the system
Exit Condition	1. The student logs out and exits the system

View Master schedule

Use case	View Master Schedule
Actor	Student
Initial Condition	1. The student will login the system
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system 2. The student will select view courses 3. The student views master schedule 4. The student will logout the system

Exit condition	1. The student will logout and exit the system.
----------------	---

View All Majors

Use case	View All majors
Actor	Student
Initial Condition	1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student goes to menu tab and select academics option in menu 3. The student goes to academic information section 4. The student goes to majors & programs link 5. The student views all the majors available at the university, of all types and of different degrees 6. The student will log out of the system
Exit Condition	1. The student logs out and exits the system

View All Major Requirements

Use case	View all Major (Requirements)
Actor	Student
Initial Condition	1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student will go to the menu tab select academics option 3. The student goes to majors & programs link

	<ol style="list-style-type: none"> 4. The student views all the majors available at the university, of all types and of different degrees 5. The student goes to the major that they are interested in 6. The student views all the major requirements they need in order to graduate 7. The student will log out of the system
Exit Condition	<ol style="list-style-type: none"> 1. The student logs out and exits the system

View All Minor

Use case	View All Minor
Actor	Student
Initial Condition	<ol style="list-style-type: none"> 1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student goes to menu tab and clicks academics option 3. The student goes to academic information section 4. The student goes to majors & programs link 5. The student goes to minor option and hits apply 6. The student views all the minors available to them at the university of many different types 7. The student will log out of the system
Exit Condition	<ol style="list-style-type: none"> 1. The student logs out and exits the system

View All Minor Requirements

Use case	View All Minor (Requirements)
Actor	Student
Initial Condition	1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student will go to the menu tab and select majors & programs link 3. The student goes to minor option and hits apply 4. The student views all the minors available at the university, of all types and of different degrees 5. The student goes to the minor that they are interested in 6. The student views all the minor requirements they need in order to graduate 7. The student will log out of the system
Exit Condition	1. The student logs out and exits the system

View Prerequisite

Use case	View Prerequisite
Actor	The student
Initial Condition	1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student goes to the menu tab and selects academic information section 3. The student goes to majors & programs link 4. The student views all the majors available to

	<p>them at the university of many different types</p> <ol style="list-style-type: none"> 5. The student goes to the major that they are interested in 6. The student goes to core requirements section 7. The students view all the courses that they must take for their major 8. The student will log out of the system
Exit Condition	<ol style="list-style-type: none"> 1. The students logs out and exits the system

View All Courses

Use case	View all courses
Actor	Student
Initial Condition	<ol style="list-style-type: none"> 1. The student will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The student logs into the system 2. The student goes to the menu tab and selects academics option 3. The student goes to academic information section in academics 4. The student goes to college catalog 5. The student goes to undergraduate catalog in pdf form 6. The student views all the courses for the different majors and minors that the student is interested in 7. The student will log out of the system
Exit Condition	<ol style="list-style-type: none"> 1. The students logs out and exits the system

View All Semesters

Use case	View all semesters
Actor	Student
Initial Condition	1. The student will login the system
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system. 2. The student will select view semesters. 3. The student views all semesters. 4. The student will logout the system
Exit condition	1. The student will logout and exit the system.

Successful Registering Student to a Course

Use case	Successful registration to a course
Actor	Student
Initial Condition	1. The student will login the system
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system. 2. The student will select view courses. 3. The student views all courses. 4. The student selects a course, 5. The student selects a section. 6. The student inputs CRN number to register for a course 7. Prompt pops up signifying that the student registers for the course. 8. The student will logout the system
Exit condition	1. The student will logout and exit the system.

Successful Dropping a course as a Student

Use case	Drop a course
Actor	Student
Initial Condition	1. The student will login to the system.
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system. 2. The student will select the view semester schedule. 3. The student will select a course they choose to drop. 4. The student drops the course. 5. The student will logout the system.
Exit condition	1. The system will out and exit the system.

Unsuccessful Registration of a course (Exceeding Credit Limit)

Use case	Unsuccessful Registration of a course - credit limit
Actor	Student
Initial Condition	1. The student will login into the system.
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system. 2. The student goes to the student Info/Registration menu, 3. The student goes to student registration. 4. The student selects the semester. 5. The student selects the department 6. The student selects a course 7. The student views a prompted message that he/she cannot register because they exceed credit limitations.

	8. The student will logout the system.
Exit condition	1. The student will logout and exit the system.

Unsuccessful Registration of a course (No Prerequisites for the Course)

Use case	Unsuccessful Registration of a course - NO prerequisites
Actor	Student
Initial Condition	1. The student will login into the system.
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system. 2. The student goes to menu tab and selects Registration 3. The student goes to student registration. 4. The student inputs the CRN number to add course-section 5. The student views a prompted message that he/she cannot register because they do not meet course-section prerequisites. 6. The student will logout the system.
Exit condition	1. The student will logout and exit the system.

Unsuccessful Registration of a course (Time conflict with two courses)

Use case	Unsuccessful Registration of a course - time conflict with two courses.
Actor	Student

Initial Condition	1. The student will login into the system.
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system. 2. The student goes to menu tab and selects Registration 3. The student goes to student registration. 4. The student inputs the CRN number to add course-section 5. The student views a prompted message that he/she cannot register because a time conflict with another course-section. 6. The student will logout the system.
Exit condition	1. The student will logout and exit the system.

Unsuccessful Registration of a course (Passing Time Limit)

Use case	Unsuccessful Registration of a course - passing time limit
Actor	Student
Initial Condition	1. The student will login into the system.
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system. 2. The student goes to menu tab and selects Registration 3. The student goes to student registration. 4. The student inputs the CRN number to add course-section 5. The student views a prompted message that he/she cannot register because the time limit is passed. 6. The student will logout the system.
Exit condition	1. The student will logout and exit the system.

Unsuccessful Registration of a course (Has a Hold)

Use case	Unsuccessful Registration of a course - has a hold
Actor	Student
Initial Condition	1. The student will login into the system.
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system. 2. The student goes to menu tab and selects Registration 3. The student goes to student registration. 4. The student inputs the CRN number to add course-section 5. The student views a prompted message that he/she cannot register because the student has a hold. 6. The student will logout the system.
Exit condition	1. The student will logout and exit the system.

Unsuccessful Registration of a course (Undergraduate registering to a Graduate Course)

Use case	Unsuccessful Registration of a course (undergraduate registering to a graduate Course)
Actor	Student
Initial Condition	1. The student will login into the system.
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system. 2. The student goes to menu tab and selects Registration 3. The student goes to student registration. 4. The student inputs the CRN number to

	<p>add course-section</p> <p>5. The student views a prompted message that he/she cannot register for a graduate course while being undergraduate</p> <p>The student will logout the system.</p>
Exit condition	<p>1. The student will logout and exit the system.</p>

Unsuccessful Registration of a course (Graduate registering to an undergraduate Course)

Use case	Unsuccessful Registration of a course (Graduate registering to an undergraduate Course)
Actor	Student
Initial Condition	<p>1. The student will login into the system.</p>
Flow of events	<p>1. The student will login to the system.</p> <p>2. The student goes to menu tab and selects Registration</p> <p>3. The student goes to student registration.</p> <p>4. The student inputs the CRN number to add course-section</p> <p>5. The student views a prompted message that he/she cannot register for an undergraduate course while being a graduate</p> <p>6. The student will logout the system.</p>
Exit condition	<p>1. The student will logout and exit the system.</p>

Faculty

Successful Login

Use case	Login (a successful login)
Actors	Faculty
Initial condition	1. The faculty attempts to log into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty log into the system using email and password. 2. The faculty is logged in successfully. 3. The faculty log off the system
Exit Condition	1. The faculty log out and exit the system.

Unsuccessful login

Use case	Login (an unsuccessful login)
Actors	Faculty
Initial condition	1. The faculty attempts to log into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty attempts to login into the system. 2. The faculty email or password are incorrect 3. The faculty does not login into the system.
Exit condition	1. The faculty does not login and returns to the home page. Then exits the system.

Successful login - a password Reset

Use case	Login. Unsuccessful login - password reset
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty attempts to login to the system using email and password but fails to do it 3 times.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty attempts to login to the system. 2. Faculty's info is not valid. (repeated 3 times) 3. Faculty account is locked out. Reset password button. 4. Faculty gets a message to a validated email associated to the account with a generated code to reset password. 5. Faculty use the code to reset passwords. 6. The faculty enter a new password and use that password to login. 7. The faculty will be able to login to the system successfully.
Exit condition	<ol style="list-style-type: none"> 1. The faculty log out and exit the system.

View Class-List

Use case	View class list
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty login into the system.

	<ol style="list-style-type: none"> 2. The faculty goes to the menu tab and selects information. 3. The faculty view the class list next to the class that the faculty would like to appear. 4. The faculty will log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty log out and exit the system.

View course-section Schedule

Use case	View course-section schedule
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty login to the system using email and password.
Flow of event of use case	<ol style="list-style-type: none"> 1. The faculty login to the system. 2. The faculty goes to menu tab and select information 3. The faculty view the course-section schedule for the semester. 4. The faculty logs out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and then exit the system.

Grade a Student

Use case	Grade student
Actors	Faculty
Initial condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty login into the system.

	<ol style="list-style-type: none"> 2. The faculty goes to menu tab and select information 3. The faculty will view the class list next to the course-section that the faculty would like to appear. 4. The faculty will grade a student by selecting the grade with a button next to the student name. 5. The faculty will press the button and assign the grade. 6. The faculty will log out the system
Exit condition	<ol style="list-style-type: none"> 1. The faculty will logout and exit the system.

Grade student - time window

Use case	Grade student - time window
Actors	Faculty
Initial condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty login into the system. 2. The faculty goes to menu tab and select information 3. The faculty will view the class list next to the course-section that the faculty would like to appear. 4. The faculty will grade a student by selecting the garde with a button next to the student name. 5. The faculty will press the button and assign the grade.

	<ol style="list-style-type: none"> 6. A message will appear stating that they cannot grade the student with the time window starts and ends. 7. The faculty will log out the system
Exit condition	<ol style="list-style-type: none"> 1. The faculty will logout and exit the system.

Take attendance

Use case	Attendance
Actors	Faculty
Initial condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty login into the system. 2. The faculty goes to menu tab information for faculty. 3. The faculty will view the course-section list next to the class that the faculty would like to appear. 4. The faculty will enter the attendance site. 5. The faculty will press the button and take attendance, present or absent. 6. The faculty will log out the system
Exit condition	<ol style="list-style-type: none"> 1. The faculty will logout and exit the system.

Take attendance - time window

Use case	Attendance - time window
Actors	Faculty

Initial condition	1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty login into the system. 2. The faculty goes to menu tab information for faculty. 3. The faculty will view the class list next to the course-section that the faculty would like to appear. 4. The faculty will enter the attendance site. 5. The faculty will press the button and take attendance, present or absent. 6. The faculty won't be able to take attendance for a class that is not offered that day. It will say that the attendance window has passed. 7. The faculty will log out the system
Exit condition	1. The faculty will logout and exit the system.

View all Majors

Use case	View All majors
Actor	Faculty
Initial Condition	1. The faculty will login into the system
Flow of Events	<ol style="list-style-type: none"> 1. The faculty logs into the system 2. The faculty goes to menu tab and select academics option in menu 3. The faculty goes to academic information section 4. The faculty goes to majors & programs link

	<ol style="list-style-type: none"> The faculty views all the majors available at the university, of all types and of different degrees The faculty will log out of the system
Exit Condition	<ol style="list-style-type: none"> The faculty logs out and exits the system

View Major Requirements

Use case	View all Major requirements
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> The faculty will login into the system. The faculty goes to view major requirements. The faculty view all the major requirements. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> The faculty will log out and exit the system.

View all Minors

Use case	View all Minors
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> The faculty will login into the system using email and password.

Flow of events of use case	<ol style="list-style-type: none"> 2. The faculty will login into the system. 3. The faculty goes to view all minors menu tab. 4. The faculty view all the minors. 5. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

View Minor requirements

Use case	View Minor requirements
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty will login into the system. 2. The faculty goes to view minor requirements. 3. The faculty view the minor requirements. 4. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

View Students in class list

Use case	View students in class list
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty will login into the system. 2. The faculty goes to view faculty information. 3. The faculty will view the students in the class list. 4. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

View all classes faculty is teaching

Use case	View all classes faculty is teaching.
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty will login into the system. 2. The faculty goes to view the faculty

	<p>information menu tab.</p> <ol style="list-style-type: none"> 3. The faculty view all classes the faculty is teaching. 4. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

View all Courses

Use case	View all courses
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty will login into the syste using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty will login into the system. 2. The faculty goes to view all courses menu tab under courses. 3. The faculty view all the courses. 4. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

View all Departments

Use case	View all departments
----------	----------------------

Actors	Faculty
Initial Condition	1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty will login into the system. 2. The faculty goes to view all departments menu tabs under departments. 3. The faculty view all the departments.. 4. The faculty log out the system.
Exit condition	1. The faculty will log out and exit the system.

View buildings

Use case	View all buildings
Actor	Student
Initial Condition	1. The student will login the system
Flow of events	<ol style="list-style-type: none"> 1. The student will login to the system. 2. The student will select view buildings. 3. The student views all buildings. 4. The student will logout the system
Exit condition	1. The student logs out and exits the system.

View Academic calendar

Use case	View academic calendar
Actors	Faculty
Initial Condition	1. The Faculty will start and enter the system using email and password.

Flow of events of use case	<ol style="list-style-type: none"> 1. The Faculty will start the system. 2. The Faculty will go to the view academic calendar menu tab. 3. The Faculty will view the academic calendar. 4. The Faculty will exit the system.
Exit condition	<ol style="list-style-type: none"> 1. The Faculty will exit the system.

View user Information

Use case	View user information
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 2. The faculty will login into the system. 3. The faculty goes to view user information menu tab under user information. 4. The faculty view the user information. 5. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

View Master schedule

Use case	View Master schedule
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 2. The faculty will login into the system.

	<ol style="list-style-type: none"> 3. The faculty goes to view the master schedule menu tab under courses. 4. The faculty view the master schedule by semester. 5. The faculty will select the semester. 6. The faculty will view the master schedule of that semester. 7. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

View all semesters

Use case	View all semesters
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty will login into the system. 2. The faculty goes to view all semesters menu tab under semesters. 3. The faculty view the semesters information. 4. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

View all Sections

Use case	View all Sections
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty will login into the system. 2. The faculty goes to view all courses menu tabs under courses. 3. The faculty will select the section link 4. The faculty will view all the sections for a particular course. 5. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

View Advisors

Use case	View advisors
Actors	Faculty
Initial Condition	<ol style="list-style-type: none"> 1. The faculty will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty will login into the system. 2. The faculty goes to view all advisors menu tabs under advisors. 3. The faculty view all the advisors. 4. The faculty log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

Admin

Successful log in case

Use case	Login (successful login)
Actor	Admin
Initial condition	1. Admin attempts to login into the system
Flow of events	<ol style="list-style-type: none"> 1. Admin logs into the system with their email and password 2. Admin successfully logs in 3. Admin logs out of the system
Exit condition	1. Admin logs out of the system

Unsuccessful log in case

Use case	Login (Unsuccessful Login)
Actor	The Administrator
Initial Condition	1. The Admin attempts to log in into the system but was unsuccessful 3 times
Flow of Events	<ol style="list-style-type: none"> 1. Admin attempt to login into the system 2. Admin's email or password is incorrect 3. The Admin does not login into the system
Exit Condition	4. The Admin does not login and return to home page.

Successful case to RESET PASSWORD

Use case	Login(Successful Login - password reset)
Actor	Admin

Initial Condition	1. Admin attempts to login into the system but fails to do it three times
Flow of Events	<ol style="list-style-type: none"> 1. Admin attempts to login into the system 2. Student's information is not valid (repeated 3 times) 3. The student is locked out of her/his account. Reset password button 4. Admin gets a message to validate their email address that is associated with their account in order to generate a code that will enable them to reset their password 5. The admin uses the code to reset the password 6. The admin creates a new password and uses that new password to login 7. The admin can login into the system successfully
Exit Condition	1. Admin logs out and exits the system

Unsuccessful case to RESET PASSWORD

Use case	Login(Unsuccessful Login - password reset)
Actor	Admin
Initial Condition	1. Admin attempts to login into the system but fails to do it three times
Flow of Events	<ol style="list-style-type: none"> 1. Admin attempts to login into the system 2. Student's information is not valid (repeated 3 times) 3. Admin is locked out of her/his account. Reset password button

	<ol style="list-style-type: none"> 4. Admin gets a message to validate their email address that is associated with their account in order to generate a code that will enable them to reset their password 8. Admin enters the incorrect email address 9. Admin is locked out of their account 10. Admin does not login into the system
Exit Condition	<ol style="list-style-type: none"> 1. The admin exits the system

View All Students

Use case	Viewing Student
Actor	Administrator
Initial Condition	<ol style="list-style-type: none"> 1. Admin will login into the system
Flow of events use case	<ol style="list-style-type: none"> 1. Admin will login into the system 2. Admin will select form menu tab informations and select students 3. Admin can now view list of all students full name with student id number, what year they are in, full/ part times and if they are a graduate or undergraduate 4. Admin will log out of the system
Exit condition	<ol style="list-style-type: none"> 1. Admin log out and exit the system

View All Faculty

Use case	Viewing faculty
Actor	Administrator
Initial Condition	<ol style="list-style-type: none"> 1. Admin will login into the system

Flow of events use case	<ol style="list-style-type: none"> 1. Admin will login into the system 2. Admin will select form menu tab informations and select faculty 3. Admin can now view list of faculty names with information such as department they are from and courses they teach 4. Admin will log out of the system
Exit condition	<ol style="list-style-type: none"> 1. Admin log out and exit the system

View All Researchers

Use case	View user information
Actors	Admin
Initial Condition	<ol style="list-style-type: none"> 1. The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The admin will login into the system. 2. The admin goes to view the user information menu tab under researcher information and view all reachers information. 3. The admin views the user information. 4. The admin log out of the system.
Exit condition	<ol style="list-style-type: none"> 1. The admin log out of the system.

View all Administrators

Use case	View all administrators
Actors	Administators
Initial Condition	<ol style="list-style-type: none"> 1. The admin will login into the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login into the system

	<ol style="list-style-type: none"> Admin will go to user information select tab menu for administrators User will be able to see admin name and the needed information Admin logs out of system
Exit Condition	<ol style="list-style-type: none"> Admin logs out of system

View all departments

Use case	View all administrators
Actors	Administrators
Initial Condition	<ol style="list-style-type: none"> The admin will login into the system.
Flow of events	<ol style="list-style-type: none"> The admin will login into the system Admin will select the button called departments User will be able to see admin name and the needed information Admin logs out of system
Exit Condition	<ol style="list-style-type: none"> Admin logs out of system

View all majors

Use case	View all Majors
Actors	Administrators
Initial Condition	<ol style="list-style-type: none"> The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> The admin will login into the system. The admin goes to view all major menu selection. The admin can view all the majors.

	4. The faculty log out the system.
Exit condition	1. The faculty will log out and exit the system.

View all Minors

Use case	View all Minors
Actors	Administrators
Initial Condition	2. The admin will login into the system.
Flow of events of use case	5. The admin will login into the system. 6. The admin goes to view all minor menu selection. 7. The admin can view all the minor. 8. The faculty log out the system.
Exit condition	2. The faculty will log out and exit the system.

View all courses

Use case	View all Minors
Actors	Administrators
Initial Condition	1. The admin will login into the system.
Flow of events of use case	1. The admin will login into the system. 2. The admin will select course catalog 3. Admin can view what coures are available and their description 4. Admin logs out the system.
Exit condition	1. The admin will log out and exit the

	system.
--	---------

Successfully searching Prerequisites for a course

Use case	Successfully searching Prerequisites for a course
Actor	Admin
Initial condition	1. The admin will login the system successfully.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system successfully. 2. The admin goes to course menu tab. 3. The admin select view all courses 4. The admin looks for a course. 5. The admin selects the course prerequisite 6. The admin views all prerequisites. 7. The admin logout the system
Exit condition	1. The admin will log out and exit the system.

Unsuccessfully searching Prerequisites for a course (course does not exist)

Use case	Unsuccessfully searching Prerequisites for a course - course does not exist
Actor	Admin
Initial condition	1. The admin will login the system successfully.
Flow of events	1. The admin will login the system successfully.

	<ol style="list-style-type: none"> 2. The admin goes to the course menu tab. 3. The admin select view all courses 4. The admin looks for a course. 5. Course is not found 6. The admin logout the system
Exit condition	<ol style="list-style-type: none"> 1. The admin will log out and exit the system.

Successfully searching a course - no prerequisites

Use case	Successfully searching a course - no prerequisite
Actor	Admin
Initial condition	<ol style="list-style-type: none"> 1. The admin will login the system successfully.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system successfully. 2. The admin goes to the course menu tab. 3. The admin select view all courses 4. The admin looks for a course. 5. The admin selects the course prerequisite 6. The admin views a prompted message that there are no prerequisites for this course. 7. The admin logout the system
Exit condition	<ol style="list-style-type: none"> 1. The admin will log out and exit the system.

Successfully editing a course

Use case	Editing a course
Actor	Admin

Initial condition	1. The admin will login the system successfully.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system successfully. 2. The admin goes to the course menu tab. 3. The admin select view all courses 4. The admin will select the edit course. 5. The admin will select the edit section. 6. The admin will select edit time, room, professor and the seat amount. 7. The admin edits the course. 8. The admin logout the system
Exit condition	1. The admin will log out and exit the system.

Unsuccessfully editing a course - time conflict - time and room

Use case	Unsuccessfully Editing a course - time conflict
Actor	Admin
Initial condition	1. The admin will login the system successfully.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system successfully. 2. The admin goes to the course menu tab. 3. The admin select view all courses 4. The admin will select the edit course. 5. The admin will select the edit section. 6. The admin will select edit time, room, professor and the seat amount. 7. The admin will view a prompted message

	<p>that there is a time conflict with another course.</p> <p>8. The admin logout the system</p>
Exit condition	<p>1. The admin will log out and exit the system.</p>

Unsuccessfully editing a course - schedule conflict

Use case	Unsuccessfully Editing a course - Schedule conflict
Actor	Admin
Initial condition	<p>1. The admin will login the system successfully.</p>
Flow of events	<p>1. The admin will login the system successfully.</p> <p>2. The admin goes to the course menu tab.</p> <p>3. The admin select view all courses</p> <p>4. The admin will select the edit course.</p> <p>5. The admin will select the edit section.</p> <p>6. The admin will select edit time, room, professor and the seat amount.</p> <p>7. The admin will view a prompted message that there is a time conflict with another course.</p> <p>8. The course time and room changes are not saved.</p> <p>9. The admin logout the system</p>
Exit condition	<p>2. The admin will log out and exit the system.</p>

Successfully editing a course (professor)

Use case	Successfully Editing a course - professor
Actor	Admin
Initial condition	<ol style="list-style-type: none"> 1. The admin will login the system successfully.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system successfully. 2. The admin goes to the course menu tab. 3. The admin select view all courses 4. The admin will select the edit course. 5. The admin will select the edit section. 6. The admin will select an edit professor. 7. The admin will select a professor. 8. The course professor will be edited. 9. The admin logout the system
Exit condition	<ol style="list-style-type: none"> 1. The admin will log out and exit the system.

Unsuccessfully editing a course - time conflict to another course - professor

Use case	Unsuccessfully Editing a course - professor conflict
Actor	Admin
Initial condition	<ol style="list-style-type: none"> 1. The admin will login the system successfully.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system successfully.

	<ol style="list-style-type: none"> The admin goes to the course menu tab. The admin select view all courses The admin will select the edit course. The admin will select the edit section. The admin will select an edit professor. The admin views a prompted message that the course is conflicted with a course in the professor schedule. The admin will not edit the course. The admin logout the system
Exit condition	<ol style="list-style-type: none"> The admin will log out and exit the system.

View major requirements

Use case	View all Major requirements
Actors	Admin
Initial Condition	<ol style="list-style-type: none"> The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> The admin will login into the system. The admin goes to view major requirements. The admin views all the major requirements. The admin logs out of the system.
Exit condition	<ol style="list-style-type: none"> The faculty will log out and exit the system.

View all major by department

Use case	View all Major requirements
----------	-----------------------------

Actors	Faculty
Initial Condition	1. The faculty will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The faculty will login into the system. 2. The faculty goes to view major requirements by department. 3. The faculty view all the major requirements. 4. The faculty log out the system.
Exit condition	1. The admin will log out and exit the system.

View minor requirements

Use case	View all Minor requirements
Actors	admin
Initial Condition	1. The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The admin will login into the system. 2. The admin goes to view minor requirements. 3. The admin views all the minor requirements. 4. The admin logs out the system.
Exit condition	1. The admin will log out and exit the system.

View all minor by department

Use case	View all Minor requirements
Actors	Faculty

Initial Condition	1. The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The admin will login into the system. 2. The admin goes to view Minor requirements by department. 3. The admin views all the Minor requirements. 4. The admin logs out the system.
Exit condition	2. The admin will log out and exit the system.

View student grade and attendance

Use case	Admin view student grade and attendance
Actors	Admin
Initial condition	2. The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 7. The admin login into the system. 8. The admin will select view majors. 9. The admin will select view courses. 10. The admin will select view courses sections. 11. The admin selects the view class list. 12. The admin will select the student's grade. 13. The admin views the student grade and attendance. 14. The admin will log out the system
Exit condition	2. The admin logout and exit the system.

Add major to a student

Use case	Add a major to a student
----------	--------------------------

Actors	Admin
Initial Condition	1. The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 2. The admin will login into the system. 3. The admin will check to see if student needs any prerequisites to add a course 4. Admin will check to see if there is seat available in the course 5. Admin will check to see if there is a schedule conflict with other courses that the student will be taking 6. Admin will add student to needed course 7. The admin logs out the system.
Exit condition	1. The admin will log out and exit the system.

View all buildings

Use case	View all buildings
Actors	Admin
Initial Condition	1. The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The admin will login into the system. 2. The fadmin goes to view all buildings menu tabs under buildings. 3. The admin views all the buildings.. 4. The admin logs out the system.
Exit condition	1. The admin will log out and exit the system.

Successfully add prerequisite to a course

Use case	Add prerequisites to a course
Actors	Admin
Initial Condition	1. The admin will login the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system. 2. The admin will select the view of all courses. 3. The admin will select edit courses. 4. The admin will select the edit section. 5. The admin will select view prerequisites. 6. The admin will select add prerequisites. 7. The admin will choose a department. 8. The admin will choose a course in that department. 9. The admin will add the course as a prerequisite. 10. The prerequisite is added. 11. The admin will log out.
Exit condition	1. The admin will logout and exit the system.

Successfully remove a prerequisite from a course

Use case	View all buildings
Actors	Admin
Initial Condition	8. The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 9. The admin will login into the system. 10. The admin goes to view all buildings menu tabs under buildings.

	11. The admin views all the buildings.. 12. The admin logs out the system.
Exit condition	2. The admin will log out and exit the system.

Successfully remove a course

Use case	Successfully remove a course
Actors	Admin
Initial Condition	1. The admin will login the system.
Flow of events	1. The admin will login the system. 2. The admin will select the view of all courses. 3. The admin will select remove course 4. The admin will remove all prerequisites, enrollments, sections, and the course is removed. 5. The admin will log out.
Exit condition	1. The admin will logout and exit the system.

Successfully adding a course

Use case	Successfully add a course
Actors	Admin
Initial Condition	1. The admin will login the system.
Flow of events	1. The admin will login the system. 2. The admin will select the view of all courses.

	<ol style="list-style-type: none"> 3. The admin will select the add course. 4. The admin will select the department where the course belongs. 5. The admin will choose the credits. 6. The admin will save changes and create the course. 7. The admin will log out.
Exit condition	<ol style="list-style-type: none"> 1. The admin will logout and exit the system.

Unsuccessfully adding a course - same course name

Use case	Successfully add a course
Actors	Admin
Initial Condition	<ol style="list-style-type: none"> 1. The admin will login the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system. 2. The admin will select the view of all courses. 3. The admin will select the add course. 4. The admin will select the department where the course belongs. 5. The admin will create a name for the course. 6. The admin views a prompted message that the course name has been used before. 7. The admin will log out.
Exit condition	<ol style="list-style-type: none"> 1. The admin will logout and exit the system.

Successfully adding a section for a course

Use case	Successfully add a section for a course
Actors	Admin
Initial Condition	1. The admin will login the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system. 2. The admin will select the view of all courses. 3. The admin will select the add a section. 4. The admin will select the room and time. 5. The admin will choose a professor for that section. 6. The admin selects room and time again to not have conflict with the professor schedule. 7. The admin will select the seat amount. 8. The admin will save changes and create the section. 9. The admin will log out.
Exit condition	1. The admin will logout and exit the system.

Unsuccessfully adding a section for a course - time conflict

Use case	Unsuccessfully add a section for a course - time conflict.
Actors	Admin
Initial Condition	1. The admin will login the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system. 2. The admin will select the view of all

	<p>courses.</p> <ol style="list-style-type: none"> The admin will select the add a section. The admin will select the room and time. The admin views a prompted message that there is conflict with room or time. The admin will log out.
Exit condition	<ol style="list-style-type: none"> The admin will logout and exit the system.

Successfully adding an user

Use case	Successfully adding an user.
Actors	Admin
Initial Condition	<ol style="list-style-type: none"> The admin will login the system.
Flow of events	<ol style="list-style-type: none"> The admin will login the system. The admin will select the view of all users. The admin will select the add an user.. The admin will type and email The admin type password. The admin type first and last name. The admin selects the date of birth. The admin will type all user address information. The admin will select if the user is Student (graduate/undergraduate), Faculty, Researcher. The admin creates the new user. The admin will log out.
Exit condition	<ol style="list-style-type: none"> The admin will logout and exit the system.

Unsuccessfully adding an user - same email for example

Use case	Unsuccessfully adding an user - same email
----------	--

Actors	Admin
Initial Condition	1. The admin will login the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system. 2. The admin will select the view of all users. 3. The admin will select the add an user.. 4. The admin will type and email 5. The admin type password. 6. The admin type first and last name. 7. The admin selects the date of birth. 8. The admin will type all user address information. 9. The admin will select if the user is Student (graduate/undergraduate), Faculty, Researcher. 10. The admin views a prompted message that the email already exists. 11. The admi will log out.
Exit condition	1. The admin will logout and exit the system.

Successfully remove a user

Use case	Successfully removing an user1
Actors	Admin
Initial Condition	1. The admin will login the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system. 2. The admin will select the view of all users. 3. The admin will select remove. 4. The admit removes the user. 5. The admin will log out.
Exit condition	1. The admin will logout and exit the system.

unsuccessfully remove a user - faculty teaching a course

Use case	unsuccessfully removing an user - faculting teaching a course
Actors	Admin
Initial Condition	1. The admin will login the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system. 2. The admin will select the view of all users. 3. The admin will select remove. 4. The admit views a prompted message that user is faculty and he/she is teaching a course. 5. The admi will log out.
Exit condition	1. The admin will logout and exit the system.

View locked users

Use case	Admin view locked users.
Actors	Admin
Initial Condition	1. The admin will login the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system. 2. The admin selects locked accounts. 3. The admin will select the view of all locked accounts. 4. The admin will view all locked accounts. 5. The admin will log out.
Exit condition	1. The admin will logout and exit the system.

Unlock users

Use case	Admin unlocked users.
Actors	Admin

Initial Condition	1. The admin will login the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system. 2. The admin selects locked accounts. 3. The admin will select the view of all locked accounts. 4. The admin selects the unlocked account. 5. The account will be unlocked and removed from the list. 6. The admin will log out.
Exit condition	1. The admin will logout and exit the system.

Register a student to a course

Use case	Admin registers a student to a course
Actors	Admin
Initial Condition	1. The admin will login the system.
Flow of events	<ol style="list-style-type: none"> 1. The admin will login the system. 2. The admin goes to the course menu. 3. The admin selects the view course. 4. The admin selects view enrollment on a class. 5. The admin adds student to the class selecting the CRN. 6. The admin will log out.
Exit condition	1. The admin will logout and exit the system.

View a student schedule

Use case	Admin View student schedule
Actors	Admin
Initial Condition	2. The admin login to the system.
Flow of event of use case	5. The admin login to the system.

	<ol style="list-style-type: none"> 6. The admin goes to menu tab information for students.. 7. The admin selects the student schedule. 8. The admin views the student schedule. 9. The admin logs out the system.
Exit condition	<ol style="list-style-type: none"> 2. The admin will log out and then exit the system.

View a faculty schedule

Use case	Admin View faculty schedule
Actors	Admin
Initial Condition	<ol style="list-style-type: none"> 1. The admin login to the system.
Flow of event of use case	<ol style="list-style-type: none"> 1. The admin login to the system. 2. The admin goes to menu tab information for faculty.. 3. The admin selects the faculty schedule. 4. The admin views the faculty schedule. 5. The admin logs out the system.
Exit condition	<ol style="list-style-type: none"> 1. The admin will log out and then exit the system.

View a faculty department

Use case	Admin View faculty department
Actors	Admin
Initial Condition	<ol style="list-style-type: none"> 1. The admin login to the system.
Flow of event of use case	<ol style="list-style-type: none"> 1. The admin login to the system. 2. The admin goes to menu tab information

	for faculty. 3. The admin selects department info. 4. The admin views the faculty department. 5. The admin logs out the system.
Exit condition	1. The admin will log out and then exit the system.

Successfully adding a department to faculty

Use case	Successfully adding a department to a faculty.
Actors	Admin
Initial Condition	1. The admin login to the system.
Flow of event of use case	1. The admin login to the system. 2. The admin goes to menu tab information for faculty.. 3. The admin selects view departments. 4. The admin selects to add a faculty to the department. 5. The faculty is now added to that department. 6. The admin logs out the system.
Exit condition	1. The admin will log out and then exit the system.

View all advisors

Use case	View advisors
Actors	Admin

Initial Condition	2. The admin will login into the system.
Flow of events of use case	5. The admin will login into the system. 6. The admin goes to view all advisors menu tab under advisors. 7. The admin views all the advisors. 8. The admin logs out of the system.
Exit condition	2. The faculty will log out and exit the system.

View students holds

Use case	View Student holds
Actors	Admin
Initial Condition	1. The admin will login into the system.
Flow of events of use case	1. The admin will login into the system. 2. The admin goes to view all student menu tabs under advisors. 3. The admin selects views the student holds. 4. The admin views all the students holds. 5. The admin logs out the system.
Exit condition	1. The admin will log out and exit the system.

View student transcript

Use case	View Student transcript
Actors	Admin
Initial Condition	1. The admin will login into the system.

Flow of events of use case	<ol style="list-style-type: none"> 1. The admin will login into the system. 2. The admin goes to view all student menu tabs under advisors. 3. The admin selects views on the student transcript. 6. The admin views all the students' transcripts. 7. The admin logs out the system.
Exit condition	<ol style="list-style-type: none"> 2. The admin will log out and exit the system.

View student major

Use case	View Student major
Actors	Admin
Initial Condition	<ol style="list-style-type: none"> 1. The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The admin will login into the system. 2. The admin goes to view all student menu tabs under advisors. 3. The admin selects views the students major. 4. The admin views all the students major. 5. The admin logs out the system.
Exit condition	<ol style="list-style-type: none"> 1. The admin will log out and exit the system.

View user statistics

Use case	View Student statistics
Actors	Admin
Initial Condition	1. The admin will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The admin will login into the system. 2. The admin goes to view all student menu tabs under advisors. 3. The admin selects views the student statistics. 4. The admin views all the students' statistics. 5. The admin logs out the system.
Exit condition	1. The admin will log out and exit the system.

Researcher

Successful Login

Use case	Login (a successful login)
Actors	Researcher
Initial condition	1. The researcher attempts to log into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The researcher logs into the system using email and password. 2. The researcher is logged in successfully. 3. The researcher log off the system
Exit Condition	1. The researcher logs out and exits the

	system.
--	---------

Unsuccessful login

Use case	Login (an unsuccessful login)
Actors	Researcher
Initial condition	1. The Researcher attempts to log into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Researcher attempts to login into the system. 2. The Researcher email or password are not correct. 3. The Researcher does not login into the system.
Exit condition	1. The Researcher will exit the system.

Successful login - a password Reset

Use case	Login. Unsuccessful login - password reset
Actors	Researcher
Initial Condition	1. The Researcher attempts to login to the system using email and password but fails to do it 3 times.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Researcher attempts to login to the system. 2. Researcher's info is not valid. (repeated 3 times) 3. Researcher account is locked out. Reset password button.

	<ol style="list-style-type: none"> 4. Researcher gets a message to a validated email associated to the account with a generated code to reset password. 5. Researcher use the code to reset passwords. 6. The Researcher enter a new password and use that password to login. 7. The Researcher will be able to login to the system successfully.
Exit condition	<ol style="list-style-type: none"> 1. The Researcher log out and exit the system.

View all Majors

Use case	View all Majors
Actors	Researcher
Initial Condition	<ol style="list-style-type: none"> 1. The Researcher will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Researcher will login into the system. 2. The Researcher goes to view all major menu tabs. 3. The Researcher view all the majors. 4. The Researcher log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The Researcher will log out and exit the system.

View all Minors

Use case	View all Minors
Actors	Researcher

Initial Condition	1. The Researcher will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Researcher will login into the system. 2. The Researcher goes to view all minors menu tab. 3. The Researcher view all the minors. 4. The Researcher log out the system.
Exit condition	1. The Researcher will log out and exit the system.

View all Courses

Use case	View all courses
Actors	Researcher
Initial Condition	1. The Researcher will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Researcher will login into the system. 2. The Researcher goes to view all courses menu tab under courses. 3. The Researcher views all the courses. 4. The Researcher log out of the system.
Exit condition	1. The Researcher will log out and exit the system.

View all buildings

Use case	View all buildings
Actors	Researcher
Initial Condition	1. The Researcher will login into the system

	using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Researcher will login into the system. 2. The Researcher goes to view all buildings menu tab under buildings. 3. The Researcher view all the buildings.. 4. The Researcher log out the system.
Exit condition	<ol style="list-style-type: none"> 1. The Researcher will log out and exit the system.

View Academic calendar

Use case	View academic calendar
Actors	Researcher
Initial Condition	<ol style="list-style-type: none"> 2. The Researcher will start and enter the system
Flow of events of use case	<ol style="list-style-type: none"> 5. The Researcher will start the system. 6. The Researcher will go to the view academic calendar menu tab. 7. The Researcher will view the academic calendar. 8. The Researcher will exit the system.
Exit condition	<ol style="list-style-type: none"> 2. The Researcher will exit the system.

View User Information

Use case	View user information
Actors	Researcher
Initial Condition	<ol style="list-style-type: none"> 2. The Researcher will login into the system using email and password.

Flow of events of use case	<ol style="list-style-type: none"> 5. The Researcher will login into the system. 6. The Researcher goes to view the user information menu tab under user information. 7. The Researcher views the user information. 8. The Researcher log out of the system.
Exit condition	<ol style="list-style-type: none"> 1. The Researcher will log out and exit the system.

Edit user information

Use case	Edit user information.
Actors	Researcher
Initial Condition	<ol style="list-style-type: none"> 1. The Researcher will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Researcher will login into the system. 2. The Researcher goes to edit the user information menu tab 3. The Researcher will edit the user information 4. The Researcher will click the update button to save changes. 5. The Researcher logs out of the system.
Exit condition	<ol style="list-style-type: none"> 1. The faculty will log out and exit the system.

View Master schedule

Use case	View Master schedule
----------	----------------------

Actors	Researcher
Initial Condition	1. The Researcher will login into the system using email and password.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Researcher will login into the system. 2. The Researcher goes to view the master schedule menu tab under courses. 3. The Researcher views the master schedule by semester. 4. The Researcher will select the semester. 5. The Researcher will view the master schedule for that semester. 6. The Researcher logs out of the system.
Exit condition	1. The Researcher will log out and exit the system.

View User statistics

Use case	View User statistics
Actors	Researcher
Initial Condition	1. The researcher will sign into the system using email and password.
Flow of events	<ol style="list-style-type: none"> 1. The researcher will login in the system. 2. The researcher will go to the view User Statistics menu tab. 3. The researcher will view all the anonymous data. 4. The researcher will logout the system
Exit condition	1. The researcher will logout and exit the system.

View performance Statistic

Use case	View performance Statistic
Actors	Researcher
Initial Condition	1. The researcher will sign into the system using email and password.
Flow of events	<ol style="list-style-type: none"> 1. The researcher will login in the system. 2. The researcher will go to the performance statistics menu tab. 3. The researcher will view all the anonymous data. 4. The researcher will logout the system
Exit condition	1. The researcher will logout and exit the system.

Visitor

View Master schedule

Use case	View Master schedule
Actors	Visitor
Initial Condition	1. The Visitor will start the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Visitor will start the system. 2. The Visitor goes to view the master schedule menu tab. 3. The Visitor views the master schedule by semester. 4. The Visitor will select the semester. 5. The Visitor will view the master schedule of that semester.

	6. The Visitor exits the system.
Exit condition	1. The Visitor will exit the system.

View all Courses.

Use case	View all courses
Actors	Visitor
Initial Condition	1. The Visitor will login into the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Visitor will login into the system. 2. The Visitor goes to view all courses menu tab under courses. 3. The Visitor views all the courses. 4. The Visitor log out of the system.
Exit condition	1. The Visitor will log out and exit the system.

View all Majors and programs

Use case	View all Majors and programs
Actors	Visitor
Initial Condition	1. The Visitor will enter the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Visitor will enter the system. 2. The Visitor goes to view all major menu tabs. 3. The Visitor views all the majors. 4. The Visitor exits the system.
Exit condition	1. The Visitor will exit the system.

View all Minors

Use case	View all Minors
Actors	Visitor
Initial Condition	1. The Visitor will enter the system.
Flow of events of use case	<ol style="list-style-type: none"> 1. The Visitor will enter the system. 2. The Visitor goes to view all minors menu tab. 3. The Visitor views all the minors. 4. The Visitor exits the system.
Exit condition	1. The Visitor will exit the system.

View Academic calendar

Use case	View academic calendar
Actors	Visitor
Initial Condition	3. The visitor will start and enter the system.
Flow ef events of use case	<ol style="list-style-type: none"> 9. The visitor will start the system. 10. The visitor will go to the view academic calendar menu tab. 11. The visitor will view the academic calendar. 12. The visitor will exit the system.
Exit condition	3. The visitor will exit the system.

Entities and Attributes

- ❖ **User (Supertype)**
 - **UserID(pk)**
 - **FirstName**
 - **LastName**
 - **DOB**
 - **Street_address**
 - **City**
 - **state**
 - **zipCode**
 - **UserType**

- ❖ **Login**
 - **UserID(fk)**
 - **Email**
 - **Password**
 - **UserType**

- ❖ **Student (Subtype)**
 - **StudentID(PK)**
 - **Year**
 - **StudentType**

- ❖ **GraduateStudent**
 - **StudentID(PK)**
 - **DepartmentID(FK)**
 - **MS/Phd**
 - **Year**
 - **StudentType**

- ❖ **UndergraduateStudent**
 - **StudentID(PK)**
 - **Year**
 - **StudentType**

- ❖ **FulltimeUnderGraduateStudent**
 - **StudentID(PK,FK)**

- **MaxCreditsAllowedPerSemester**
- **CreditsTaking**
- **StudentStanding(Senior, Junior, etc)**
- **Thesis(0/1)**

❖ **PartTimeUnderGraduateStudent**

- **StudentID(PK,FK)**
- **Year**
- **MaxCreditsAllowedPerSemester**
- **CreditsTaking**
- **StudentStanding(Senior, Junior, etc)**

❖ **FulltimeGraduateStudent**

- **StudentID(PK,FK)**
- **Year**
- **creditsEarned**
- **MS/Phd**
- **Thesis (0/1)**

❖ **PartTimeGraduateStudent**

- **StudentID(PK,FK)**
- **Year**
- **creditsEarned**
- **MS/Phd**
- **Thesis (0/1)**

❖ **Faculty (Subtype)**

- **FacultyID(PK)**
- **FacultyType**
- **Specialty**
- **Rank**

❖ **FulltimeFaculty (Subtype of Faculty)**

- **FacultyID(PK,FK)**
- **MaxCourseLoad(2)**
- **OfficeID(FK)**

- ❖ **PartTimeFaculty (Subtype of Faculty)**
 - **FacultyID(PK,FK)**
 - **MaxcourseLoad(1)**
 - **OfficeID(FK)**

- ❖ **Administrator (Subtype of User)**
 - **AdminID(PK)**
 - **Priority(0/1)**

- ❖ **Researcher (Subtype of User)**
 - **ResearcherID(PK)**
 - **Statues(0/1)**

- ❖ **Department**
 - **DepartmentID(PK)**
 - **DepartmentName**
 - **PhoneNumber**
 - **DepartmentEmail**
 - **BuildingID(FK)**
 - **RoomID(FK)**
 - **Faculty_Chair_ID**

- ❖ **Department/Faculty**
 - **DepartmentID(FK)**
 - **FacultyID(FK)**
 - **DateOfAppointment**
 - **PercentageTimeCommitment**

- ❖ **Building**
 - **BuildingID**
 - **BuildingName**
 - **Usage (Student center, Administration Building, Academic building, Heating plant, Receiving Building)**
- ❖ **Room**
 - **RoomID(PK) ((NAB))<---ID name example**
 - **BuildingID(FK)**

- **RoomNumber**
- **RoomType**

❖ **Classroom**

- **RoomNumber(FK)**
- **BuildingID(FK)**
- **ClassroomCapacity**

❖ **Office**

- **RoomNumber(FK)**
- **BuildingID(FK)**
- **OfficeCapacity**

❖ **Lab**

- **RoomNumber(FK)**
- **BuildingID(FK)**
- **LabCapacity**

❖ **Semester**

- **SemesterID(PK)**
- **SemesterName**
- **SemesterStart**
- **SemesterEnd**
- **SemesterYear**

❖ **Enrollment**

- **CRN(PK,FK)**
- **StudentID**
- **Grade**
- **DateOfRegistration**

❖ **Attendance**

- **StudentID(PK,FK)**
- **CourseID(FK)**

- **CRN(FK)**
- **AttendanceDate**
- **Present(1)/Absent (0)**

❖ **Course-Section**

- **CRN(PK)**
- **CourseID (FK)**
- **SectionNumber**
- **FacultyID(FK)**
- **BuildingID(FK)**
- **RoomID(FK)**
- **Time_SlotID(FK)**
- **SemesterID(FK)**
- **SectionAvailability**
- **SemesterYear**

❖ **ClassList**

- CRN(PK)**
- FacultyID(FK)**
- StudentID(FK)**
- MajorID(FK)**
- RoomID(FK)**
- SemesterID(FK)**

❖ **Course_Prerequisite**

- **CourseID(PK, FK)**
- **PrerequisiteId(PK, FK)**
- **GradeRequired**

❖ **Major Requirement**

- **MajorID(FK)**
- **CourseID**

➤ **GradeRequirement**

❖ **Student/Holds**

- **StudentId(PK,FK)**
- **HoldID(FK)**
- **DateofHold**

❖ **MinorRequirements**

- **MinorID(FK)**
- **CourseID(FK)**
- **GradeRequirement**

❖ **StudentMinor**

-**StudentID(PK)**

-**MinorID(FK)**

-**DateofDecision**

❖ **Minor**

- **MinorID(PK)**
- **DepartmentID(FK)**
- **DateofOrigin**
- **CreditsRequired**

❖ **StudentMajor**

- **StudentID(PK)**

- **MajorID(FK)**

- **DateofDecision**

❖ **Major**

- **MajorID**
- **MajorName**

- **DepartmentID(FK)**
- **DateofOrigin**
- **CreditsRequired**

❖ **Advisor**

-FacultyID

-StudentID

❖ **Holds**

- **HoldID**
- **HoldType**
- **HoldDescription**

❖ **TimeSlot**

-TimeSlotID (PK)

-Day

-Period

❖ **TimeSlot_Day**

-TimeSlotID

-DayID

❖ **Day**

-DayID

-Weekday

❖ **TimeSlot_Period**

-PeriodID (PK)

-TimeSlotID (FK)

❖ **Period**

➤ PeriodNumber(PK)

➤ Starttime

➤ Endtime

❖ **Student_History**

➤ StudentID(fk)

➤ CRN(fk)

➤ CourseID(fk)

➤ GradeReceived

➤ SemesterID(fk)

❖ **Faculty_History**

➤ FacultyID(pk,fk)

➤ CRN(fk)

➤ CourseID

➤ SemesterYearID

➤ CRN(PK,FK)

➤ FacultyID(FK)

➤ DepartmentID(FK)

➤ RoomID(FK)

➤ SemesterID(FK)

Relationships and Attributes

Student

- A student can apply to one to four courses.

- A student can have zero to 4 holds.
- A student can have one and only one ID.
- A student can have one or more advisors.
- A Student Can Be Part Time and have 8 credits
- A Student Can Be Full-time and have 12 credits

Faculty

- A Faculty Can Teach Many Courses.
- A faculty can teach many students.
- A Faculty Can Have One And Only One Department.
- A faculty can have one and only one Id.
- A faculty can have one and only one Major.
- A Faculty Can Be Either Part Time Or Full Time.

Department

- A Department Can Have One Or More Faculty.
- A Department Can Have One Or More Majors.
- A Department Can Have One Or More Minors.
- A Department Can Have One Or More Courses.

Student and hold

- A student can have 1-4 holds.
- A hold can have none or many students.

Student and Advisor

- An advisor can advise 1 or more students.
- A student can have 1 or more advisors.

student and section

- A section can have 1 to 15 students.

Student and major

- A Student Can Have 1 Or 2 Majors.

Student and minor

- A student can have 0 or 1 minor.

Department and faculty

- Faculty Member Can work in 1 or more departments
- A Department can have many faculty working.

Building and Room

- A Building Can Have Many Rooms.
- A room can only belong to one building.

Major and prerequisite

- A major can have 1 or more prerequisites.

- A prerequisite can belong to one or more majors.

Minor and prerequisite

- A Minor Can Have 1 Or More Prerequisites.
- Aprerequisitecanbelongtooneormoreminors.

TimeSlot and Period

- A Time Slot can have 1 or more periods.
- A periods can have 0 or more Times Slots

Schema:

User(User_ID, First_Name, Last_Name, DOB, Street_Address, City, State, ZipCode, User_Type)

Login (User_ID, User_Email, Password, User_Type)

Student (Student_ID, Student_Type, Year)

UnderGraduateStudent (Student_ID, Year, StudentType)

FullTimeUnderGraduateStudent (StudentID, MaxCreditsAllowedPerSemester, CreditsTaking, StudentStanding, Thesis, MS/Phd)

PartTimeUnderGraduateStudent (StudentID, MaxCreditsAllowedPerSemester, CreditsTaking, StudentStanding, Thesis, MS/Phd)

GraduateStudent (StudentID, MS/Phd, DepartmentID, Student_Type)

FullTimeGraduateStudent (StudentID, creditsEarned, Year, MS/Phd, Thesis)

PartTimeGraduateStudent (StudentID, creditsEarned, StudentProgram, Year, Thesis)

Researcher (ResearcherID, Statues)

Faculty (FacultyID, DepartmentID, FacultyType, Specialty)

Administration (AdminID, Priority)

Part-Time_Faculty (FacultyID, MaxcourseLoad, OfficeID)

Full-Time_Faculty (FacultyID, MaxcourseLoad, OfficeID)

Holds (HoldID, HoldType, HoldDescription)

Department (DepartmentID, Department_Name, Phone_Number, Department_Email, Building_ID, Room_ID, Faculty_Chair_ID)

Major (MajorID, MajorName, DepartmentID, DateOfOrigin)

Minor (MinorID, MinorName, DepartmentID, DateOfOrigin)

Course-Section(CRN, CourseID, SectionNumber, FacultyID, TimeSlotID, SectionAvailability, SemesterYear)

Course_Prerequisite(PrerequisiteID, CourseID, GradeRequired)

TimeSlot(TimeslotID, Day, Period)

Timeslot_Day(TimeslotID, Day_ID)

Timeslot_Period(TimeSlotID, PeriodID)

Day(Day_Id, weekDay)

Period(PeriodID, StartTime, EndTime)

Semester(SemesterID, SemesterName, SemesterStart, SemesterEnd, SemesterYear)

Building(BuildingID, BuildingName, BuildingUse)

Room(RoomID, BuildingID, RoomNumber, RoomType)

Classroom (RoomNumber, BuildingID, ClassroomCapacity)

Office (RoomNumber, BuildingID, OfficeCapacity)

Lab(RoomNumber, BuildingID, LabCapacity)

Student_History(StudentID, CRN, CourseID, SemesterID, GradeReceived)

Faculty_History(FacultyID, CRN, CourseID, SemesterID)

ClassList (CRN, FacultyID, StudentID, MajorID, RoomID, SemesterID)

Attendance (StudentID, CRN, CourseID, AttendanceDate, Present/Absent)

Schema for relationships:

The *many to many relationships - needed to be break down to - 1 to many* so we got:

Advisor (FacultyID, StudentID, DateofAssignment)

Enrollment(CRN, StudentID, Grade, DateofRegistration)

StudentHolds (StudentID, DateofHold, HoldID)

Department/Faculty (DepartmentID, FacultyID, PercentageTimeCommitment, dateOfAppointment)

StudentMajor (MajorID, StudentID, DateofDecision)

StudentMinor(MinorID, StudentID, DateofDecision)

MajorRequirements(MajorID, CourseID, GradeRequirement)

MinorRequirements(MinorID, CourseID, GradeRequirement)

Appendix A

Team Leader:

Emiz Intriago

Responsibilities:

Anibal Ruiz:

Required and Forbidden rules for faculty.

Required and Forbidden rules for Visitor and Researcher.

Use cases for Faculty.

Help with use cases for visitors and researchers.

Yessenia Fernandez:

Meeting Log

Required and Forbidden rules for Students.

Required and Forbidden rules for Visitor and Researcher.

Use cases for Students.

Help with use cases for visitors and researchers.

Emiz Intriago:

Required and Forbidden rules for Administrator.

Required and Forbidden rules for Visitor and Researcher.

Use cases for Administrator.

Help with use cases for visitors and researchers.