College Database

Amron Berhanu
916320644
Github: amron98

Milestone/Version	Date
M2V1	5/7/2021
M1V2	4/23/2021
M1V1	3/9/2021

Table of Contents

Section I: Project Description

Colleges have been seeing a rise in enrollment since the turn of the millennium. A study from The National Center for Education Statistics shows that the overall enrollment rate of 18-to-24 year olds in the United States has increased from 35% in 2000 to 41% in 2018, and projects this trend to continue rising in the years to come. This means that Post Secondary institutions have to be accountable and maintain an exponentially increasing amount of data generated from their students, staff, and overall operations. This becomes an even bigger task if the institution only stores their records physically as paper files, as they have high storage costs and are easily prone to damage. Furthermore, Colleges that use computerized database management systems can also suffer from a lack of efficiency (which increases their storage costs) when poorly designed and are even prone to hacks that could cost them millions to resolve.

This database management system is designed to solve this problem, and allows a typical College to store valuable data pertaining to all of it's operations. It has been efficiently designed to scale with the expected rise in enrollment, whilst minimizing the storage costs and ensuring an industry-standard integrity that will deter potential hacks. Furthermore, an Institution using this database management system will be able to store GeoLocation data that can be used to help new students easily find their way around campus.

Section II: Use Cases

Use Case #1 Title:	**Enrolling in a Course without meeting Prerequisites**
Actors:	Jason (Student), Data Structures(Course)
Description:	Jason is a newly admitted freshman majoring in Computer Science. Since he has had programming experience before, he decided to challenge himself by taking the Data Structures course in his first semester. However, he was soon to find out that he was unable to do so since the course enforces a prerequisite of the Intro to Programming to be completed before attempting the course. And since he was attempting to concurrently enroll in both courses, the system prevented him from doing so.

Use Case Title:	**Locate a Classroom**
Actors:	Niko (Student)
Description:	Niko is a newly admitted transfer student majoring in Biology. He was not able to enroll in the classes he'd liked, so he settled with enrolling in classes that were crammed into a compact schedule with gaps as small 15 minutes between classes. He was worried that he would be late to some essential lectures during the first week, but fortunately, noticed that the school website has a feature that pinpoints a given campus location on a real-time map. Niko was then able to use this feature to locate all of his classes on time, and even used it to explore the campus during his free time.

Use Case Title:	**Receive Grant**
Actors:	Anna (Student) , Financial Aid (Grant), Scholarship(Grant)
Description:	Anna has decided to enroll in 15 units for the current semester. After registering for her courses, she incurred a hefty charge due to her new full-time status as well as additional fees related to her courses. However, she is a recipient of federal financial aid, and was also recently awarded a scholarship grant from the fellowship she is part of. She was later happy to find out that she received more money than she was charged, and received a refund amount that she used to buy herself a new laptop for her studies.

Use Case Title:	**Earn a Degree**
Actors:	Thomas (Student)
Description:	Thomas is a 5th year student majoring in Civil Engineering. He has completed all of his GE requirements, as well as his University requirements, and is currently enrolled in the last few classes required by his Bachelor's degree. Excited to be almost done with school, he applies for graduation in the Spring semester of that year, and was shortly approved since he was indeed only left with an amount of units that are considered acceptable, and as he was also in a good academic standing. A few months later, Thomas celebrated his tremendous achievement with his fellow classmates and family, and began preparing for his new job in the construction industry.

Use Case #3 Title:	**Evaluate a Course**
Actors:	Betty (Student), Prof. Potter (Instructor), Biology 100 (Course)
Description:	Betty has really enjoyed her semester and has learned a lot from her Biology 100 course. As she was preparing for her finals, she received a notice that the course evaluations were due within the next 10 days. Betty decided to go ahead and submit an evaluation for her Biology class and used that opportunity to praise her instructor's excellent teaching ability and the overall structure of the course.

Section III: Database Requirements (Business Rules)

1. **Students**
   a. A student shall have one unique student id
   b. A student shall have one first name
   c. A student shall have one last name
   d. A student shall have one email address
   e. A student shall have one phone number
   f. A student shall have zero or many addresses
   g. A student shall have many payment types
   h. A student shall have one academic level
   i. A student shall have one academic standing
   j. A student shall receive zero or many grants
   k. A student shall achieve zero or many academic honors
   l. A student shall have only one transcript
   m. A student shall enroll in one or many courses before the enrollment deadline
   n. A student shall major in one or many majors
   o. A student shall be charged zero or many fees

2. **Staff**
   a. A staff member shall have one unique staff id
   b. A staff member shall have one first name
   c. A staff member shall have one last name
   d. A staff member shall have one email address
   e. A staff member shall have one phone number
   f. A staff member may or may not be a finance member
   g. A staff member may or may not be a faculty member
   h. A staff member may or may not be an administration member
   i. A staff member shall have zero or many addresses
   j. A staff member shall have many payment types

3. **Faculty**
   a. A faculty member is a staff member
   b. A faculty member shall have one unique staff id
   c. A faculty member shall have one office phone number
   d. The faculty shall manage many courses offered to students
   e. A faculty member shall instruct zero or many course sections
   f. A faculty member shall chair at most one department

g. A faculty member shall work in only one department
h. A faculty member shall have an office in at most one room

4. **Administration**
    a. An administrative staff member is a staff member
    b. An administrative staff member shall have one unique staff id
    c. An administrative staff member shall have one office phone number
    d. An administrative staff member shall have one role
    e. An administrative staff member shall manage zero or many college coursework requirements
    f. An administrative staff member shall manage zero or many general education coursework requirements

5. **Finance**
    a. A finance staff member is a staff member
    b. A finance staff member shall have one unique staff id
    c. A finance staff member shall have one office phone number
    d. A finance staff member shall have one role
    e. A finance staff member shall manage at least one fee
    f. A finance staff member shall manage at least one grant

6. **Payment Types**
    a. A payment type shall have one unique staff id
    b. A payment type shall have one description
    c. A payment type shall have one address
    d. A payment type may or may not be a bank account
    e. A payment type may or may not be a credit card
    f. A payment type shall belong to many students
    g. A payment type shall belong to many staff

7. **Bank Account**
    a. A bank account is a payment type
    b. A bank account shall have one unique account number
    c. A bank account shall have one bank name
    d. A bank account shall have one routing number

8. **Credit Card**
    a. A credit card is a payment type
    b. A credit card shall have one unique card number
    c. A credit card shall have one bank name
    d. A credit card shall have one security code (ccv)
    e. A credit card shall have one expiry month
    f. A credit card shall have one expiry year

9. Address

a. An address shall have one unique address id
b. An address shall have one street name
c. An address shall have one street number
d. An address shall have one apartment number
e. An address shall have one city
f. An address shall have one state
g. An address shall have one zip code
h. An address shall belong to zero or many students
i. An address shall belong to zero or many staff

10. Department

a. A department shall have one unique department id
b. A department shall have one department name
c. A department shall have one department phone number
d. A department shall be chaired by only one faculty member
e. A department shall have zero or many faculty members
f. A department shall offer zero or many courses
g. A department shall offer zero or many majors
h. A department may or may not have an office
i. A department shall manage zero or many core requirements

11. Majors

a. A major shall have one unique major id
b. A major shall have one major name
c. A major shall have one major type
d. A major shall provide zero or many degrees
e. A major shall be provided by at most one department
f. A major shall have zero or many students

12. Degree

a. A degree shall have one unique degree id
b. A degree shall have only and only one degree level
c. A degree shall have only and only one major
d. A degree shall have at least one GE requirement
e. A degree shall have at least one college requirement
f. A degree shall have at least one core requirement

13. Degree Levels

a. A degree level shall have one unique degree level id
b. A degree level shall have one level
c. A degree level shall be offered by zero or many degrees

14. Courses

a. A course shall have one unique course id
b. A course shall have one course title
c. A course shall have one number of units
d. A course shall have zero or many courses as prerequisites
e. A course shall have zero or many sections
f. A course shall be offered by one department
g. A course shall have zero or many enrolled students before the enrollment deadline
h. A course shall satisfy zero or many GE requirements
i. A course shall satisfy zero or many college requirements
j. A course shall satisfy zero or many course requirements
k. A course shall have zero or many student evaluations
l. A course shall be available in zero or many semesters
m. A course shall have zero or many course fees

15. Sections

a. A section shall belong to only and only one course
b. A section shall have one unique section id
c. A section shall have one section title
d. A section shall have one section type
e. A section shall may or may not be located in one classroom
f. A section shall be in session for zero or many timeslots
g. A section shall be in session for zero or many working days

16. Transcripts

a. A transcript shall have one unique transcript id
b. A transcript shall be belong to only one student
c. A transcript shall have one GPA
d. A transcript shall have one number of units taken
e. A transcript shall have one path to the transcript file

17. Academic Levels

a. An academic level shall have one academic level id
b. An academic level shall have one academic level name
c. An academic level shall belong to zero or many students

18. Academic Standing

a. An academic standing shall have one academic standing id
b. An academic standing shall have one academic standing name
c. An academic standing shall have one minimum GPA requirement
d. An academic standing shall belong to zero or many students

19. Honors (Strong)

a. An academic honor shall have one unique honor id
b. An academic honor shall have one title
c. An academic honor shall have one description
d. An academic honor shall be awarded to zero or many students

20. Grants

a. A grant shall have one unique grant id
b. A grant shall have one grant name
c. A grant shall have one grant amount
d. A grant shall have one grant provider
e. A grant shall be received by zero or many students
f. A grant shall be managed by zero or many finance staff members

21. Fees

a. A fee shall have one unique fee id
b. A fee shall have one fee name
c. A fee shall have one fee amount
d. A fee shall have one fee description
e. A fee shall be charged by zero or many courses
f. A fee shall be managed by zero or many finance staff members
g. A fee shall be charged to zero or many students

22. Semesters

a. A semester shall have one unique semester id
b. A semester shall have one semester name
c. A semester shall have one semester year
d. A semester shall have zero or many courses available

23. Evaluations

a. An evaluation shall have one unique evaluation id
b. An evaluation shall evaluate one course
c. An evaluation shall evaluate one faculty member
d. An evaluation shall have one path to the file of evaluation contents

24. Timeslots

a. A timeslot shall have one unique slot id
b. A timeslot shall have one start time
c. A timeslot shall have one end time
d. A timeslot shall be on zero or many working days
e. A timeslot shall belong to zero or many course sections

25. Working days

a. A working day shall have one day id
b. A working day shall have one day
c. A working day shall have zero or many timeslots
d. A working day shall belong to zero or many course sections

26. Buildings

a. A building shall have one unique building id
b. A building shall have one building name
c. A building shall have one location
d. A building shall contain zero or many rooms

27. Rooms

a. A room shall have one unique room id
b. A room shall have one room name
c. A room shall have one room number
d. A room shall have one capacity
e. A room shall be located in only and only one building
f. A room shall be a classroom to zero or many sections
g. A room shall be an office to zero or many faculty staff members

28. Locations

a. A location shall have one unique location id
b. A location shall have one latitude
c. A location shall have one longitude
d. A location shall locate at most one building

29. GE Requirements

a. A GE requirement shall have one unique requirement id
b. A GE requirement shall have one area of study
c. A GE requirement shall have one division level
d. A GE requirement shall have one units requirement
e. A GE requirement shall be satisfied by zero or many courses
f. A GE requirement shall be required by zero or many degrees
g. A GE requirement shall be managed by zero or many administration staff members

30. College Requirements

a. A college requirement shall have one unique requirement id
b. A college requirement shall have one description
c. A college requirement shall have one units requirement
d. A college requirement shall be satisfied by zero or many courses
e. A college requirement shall be satisfied by zero or many degrees

f. A college requirement shall be managed by zero or many administration staff members

31. Core Requirements

a. A core requirement shall have one unique requirement id
b. A core requirement shall have one area of study
c. A core requirement shall have one division level
d. A core requirement shall have one units requirement
e. A core requirement shall be satisfied by zero or many courses
f. A core requirement shall be satisfied by zero or many degrees
g. A core requirement shall be managed by zero or many departments

Section IV: Detailed List of Main Entities, Attributes and Keys

1. **Students (Strong)**
   - student_id: key, numeric
   - first_name: alphanumeric
   - last_name: alphanumeric
   - email: alphanumeric
   - phone: numeric

2. **Staff (Strong)**
   - staff_id: key, numeric
   - first_name: alphanumeric
   - last_name: alphanumeric
   - email: alphanumeric
   - phone: numeric

3. **Faculty (Weak)**
   - staff_id: key, numeric
   - role: alphanumeric
   - office_phone: numeric

4. **Administration (Weak)**
   - staff_id: key, numeric
   - role: alphanumeric
   - office_phone: numeric

5. **Finance (Weak)**
   - staff_id: key, numeric
   - role: alphanumeric
   - office_phone: numeric

6. **ProcessFunds (Weak)**
   - fee: weak key, numeric
   - grant: weak key, numeric
   - processedBy: weak key, numeric

7. **PaymentTypes (Strong)**
   - type_id: key, numeric
   - address: weak key, numeric
   - description: alphanumeric

8. **StudentPaymentInfo (Weak)**
    - paymentInfo_id: key, numeric
    - student: weak key, numeric
    - paymentType: weak key, numeric

9. **StaffPaymentInfo (Weak)**
    - paymentInfo_id: key, numeric
    - staff: weak key, numeric
    - paymentType: weak key, numeric

10. **BankAccount (Weak)**
    - account_num: key, numeric
    - bank: alphanumeric
    - routing: numeric

11. **Credit Card (Weak)**
    - card_num: key, numeric
    - bank: alphanumeric
    - ccv: numeric
    - exp_date: composite, date

12. **Address (Strong)**
    - address_id: key, numeric
    - street_addr: composite, alphanumeric
    - city: alphanumeric
    - state: alphanumeric
    - zipcode: numeric

13. **StudentAddress (Weak)**
    - student_address_id: key, numeric
    - student: weak key, numeric
    - address: weak key, numeric

14. **StaffAddress (Weak)**
    - staff_address_id: key, numeric
    - staff: weak key, numeric
    - address: weak key, numeric

15. **Departments (Strong)**
    - dep_id: key, numeric
    - name: alphanumeric
    - phone: numeric

16. Departments (Strong)

- dep_id: key, numeric
- name: alphanumeric
- phone: numeric

17. MajorsRequirements (Weak)

- requirement_id: key, numeric
- requirement: weak key, numeric
- major: weak key, numeric
- department: weak key, numeric

18. Majors (Strong)

- major_id: key, numeric
- name: alphanumeric
- type: alphanumeric

19. MajorsByStudent (Weak)

- major: weak key, numeric
- student: weak key, numeric

20. Degree (Weak)

- degree_id: key, numeric
- major: weak key, numeric
- level: weak key, numeric

21. Degree Levels (Strong)

- degree_level_id: key, numeric
- level: alphanumeric

22. Courses (Strong)

- course_id: key, numeric
- title: alphanumeric
- units: numeric
- department: weak key, numeric

23. CourseAttributes (Weak)

- course: weak key, numeric
- GE: weak key, numeric
- core_req: weak key, numeric
- college_req: weak key, numeric

24. DegreeRequirements (Weak)

- degree: weak key, numeric

- GE: weak key, numeric
- core_req: weak key, numeric
- college_req: weak key, numeric

25. Enrollment (Weak)

- enrollment_id: key, numeric
- student: weak key, numeric
- course: weak key, numeric
- deadline: composite, timestamp

26. Sections (Weak)

- section_id: key, numeric
- title: alphanumeric
- type: numeric
- course: weak key, numeric
- instructor: weak key, numeric

27. ClassSchedule (Weak)

- schedule_id: key, numeric
- timeslot: weak key, numeric
- section: weak key, numeric
- day: weak key, numeric

28. Transcripts (Strong)

- transcript_id: key, numeric
- units_taken: derived, numeric
- GPA: derived, numeric
- student: weak key, numeric
- filepath: alphanumeric

29. Academic Levels (Strong)

- academic_level_id: key, numeric
- name: alphanumeric

30. Academic Standings (Strong)

- academic_standing_id: key, numeric
- standing: alphanumeric
- min_GPA: numeric

31. Honors (Strong)

- honor_id: key, numeric
- title: alphanumeric
- description: numeric

32. HonorsStudents (Weak)

- student: weak key, numeric
- honor: alphanumeric
- award_date: composite, date

33. Grants (Strong)

- grant_id: key, numeric
- name: alphanumeric
- amount: numeric
- provider: alphanumeric

34. GrantRecepients (Weak)

- recepient_id: key, numeric
- student: weak key, numeric
- grant: weak key, numeric
- award_date: composite, date

35. Fees (Strong)

- fee_id: key, numeric
- amount: numeric
- description: alphanumeric

36. CourseFees (Weak)

- course_fee_id: key, numeric
- course: weak key, numeric
- fee: weak key, numeric

37. StudentFees (Weak)

- student_fee_id: key, numeric
- student: weak key, numeric
- fee: weak key, numeric
- date: composite, timestamp

38. Semesters (Strong)

- semester_id: key, numeric
- semester: alphanumeric
- year: numeric

39. CoursesBySemester (Strong)

- evaluation_id: key, numeric
- course: weak key, numeric
- semester: weak key, numeric

40. Evaluations (Strong)

- evaluation_id: key, numeric
- course: weak key, numeric
- instructor: weak key, numeric
- filepath: alphanumeric

41. Timeslots (Strong)

- slot_id: key, numeric
- start: timestamp
- end: timestamp

42. WorkingDays (Strong)

- day_id: key, numeric
- day: alphanumeric

43. Buildings (Strong)

- building_id: key, numeric
- name: alphanumeric
- location_id: weak key, numeric

44. Rooms (Weak)

- room_id: key, numeric
- name: alphanumeric
- capacity: numeric
- room_no: numeric
- building_id: weak_key, numeric

45. Locations (Strong)

- location_id: key, numeric
- latitude: numeric
- longitude: numeric

46. GERequirements (Strong)

- requirement_id: key, numeric
- area: alphanumeric
- division: alphanumeric
- units_required: numeric

47. CollegeRequirements (Strong)

- requirement_id: key, numeric
- description: alphanumeric
- units_required: numeric

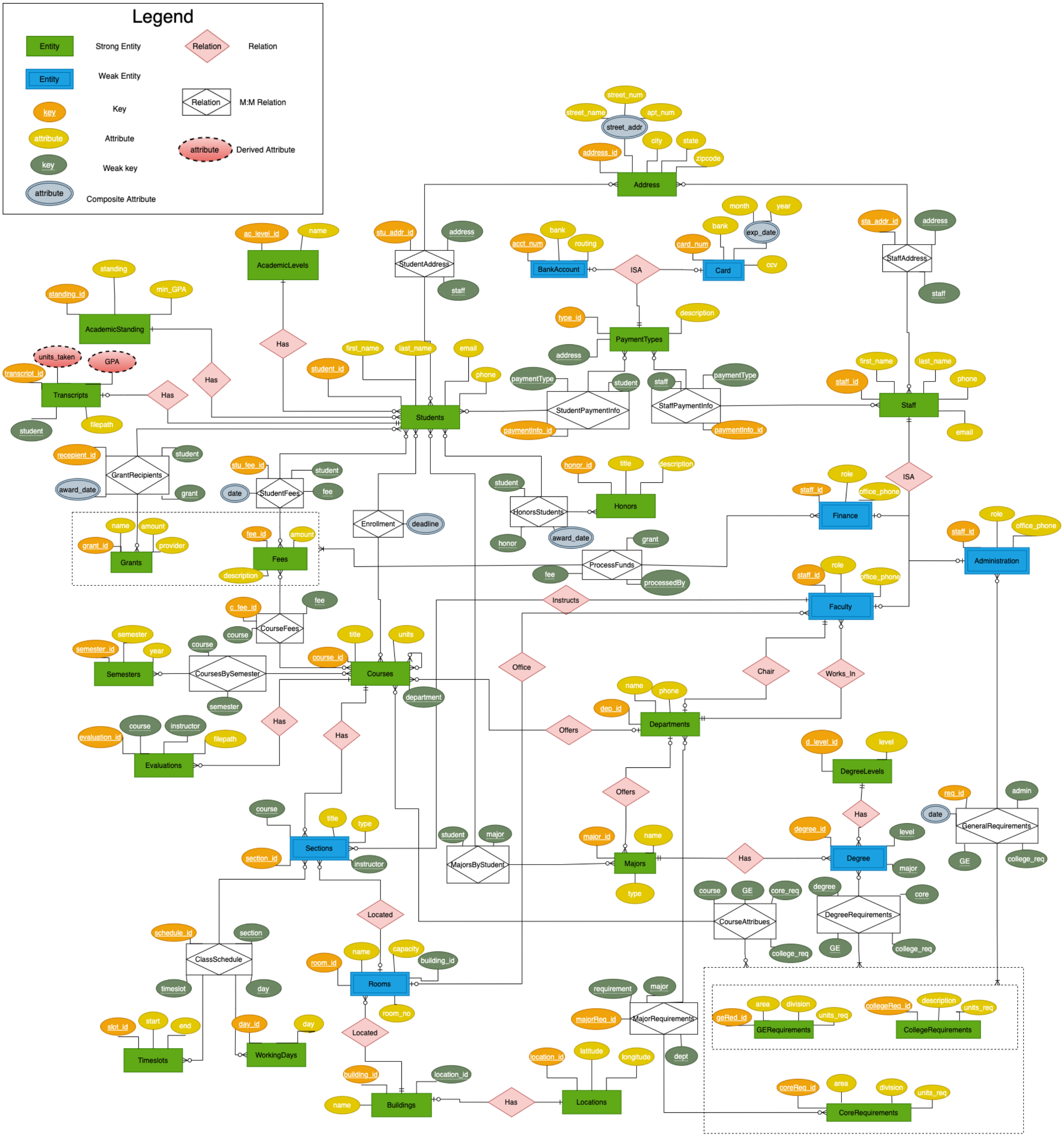
48. CoreRequirements (Strong)

- requirement_id: key, numeric
- area: alphanumeric
- division: alphanumeric
- units_required: numeric

49. GeneralRequirements (Strong)

- requirement_id: key, numeric
- GE: weak key, numeric
- college_req: weak key, numeric
- admin: weak key, numeric
- date: composite, date

Section V: Entity Relationship Diagram

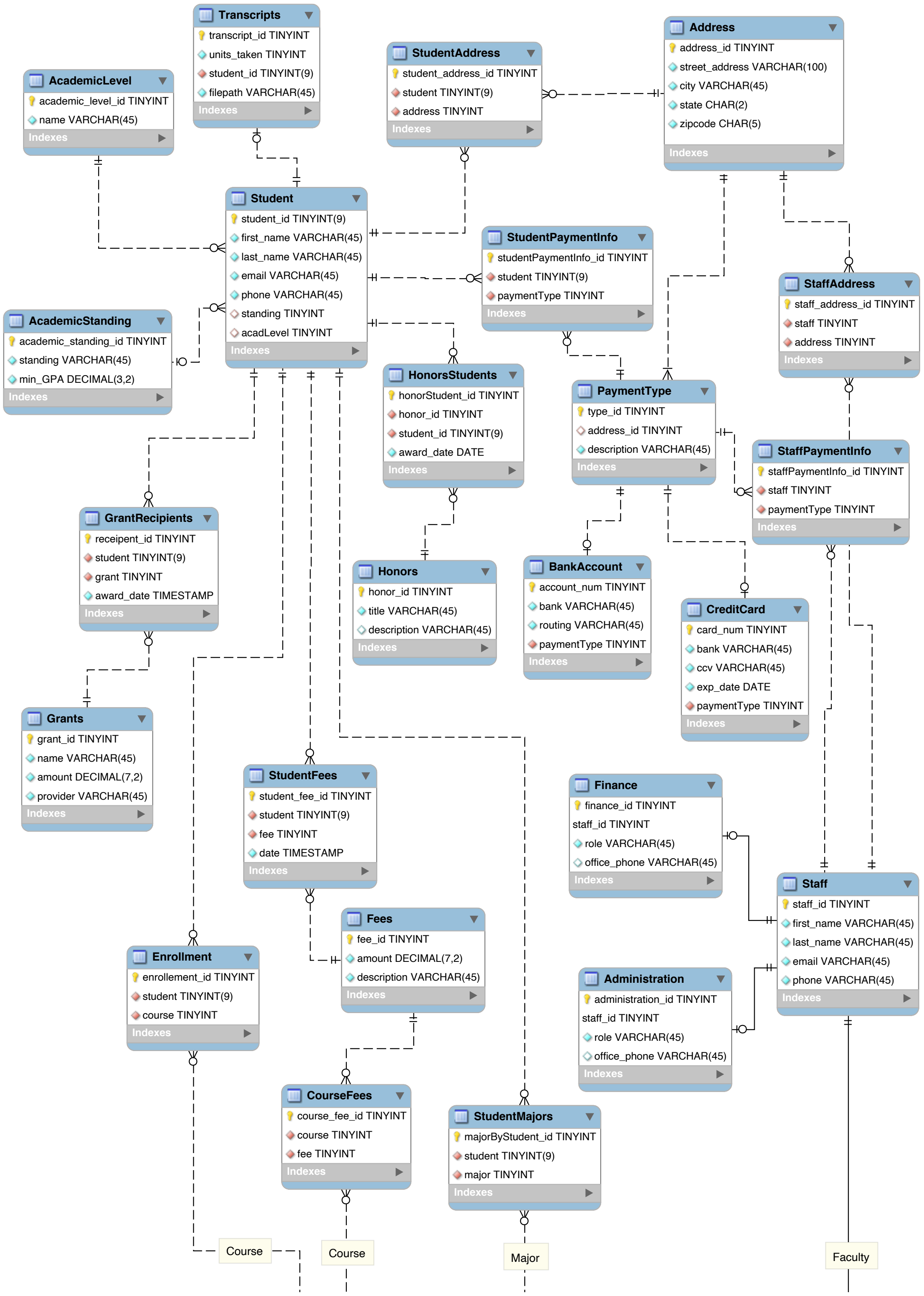


Section VI: Testing Table

Rule	Entity A	Relation	Entity B	Cardinality	Pass /Fail	Error Description
1	Students	Has	AcademicLevels	M:1	Pass	None
2	Students	Has	AcademicStanding	M:1	Pass	None
3	Students	Has	Address	M:M	Pass	None
4	Students	Recieves	Grants	M:M	Pass	None
5	Students	Has	PaymentTypes	M:M	Pass	None
6	Students	Award	Honors	M:M	Pass	None
7	Students	Major	Majors	M:M	Pass	None
8	Students	Enroll	Courses	M:M	Pass	None
9	Students	Has	Fees	M:M	Pass	None
10	Students	Has	Transcripts	M:1	Fail	While multiple students could have the same GPA and units, they cannot have the same info that would be in a transcript
11	Staff	ISA	Administration	1:1	Pass	None
12	Staff	ISA	Faculty	1:1	Pass	None
13	Staff	ISA	Finance	1:1	Pass	None
14	Staff	Has	Address	M:M	Pass	None
15	Staff	Has	PaymentTypes	M:M	Pass	None
16	Staff	Office	Rooms	M:1	Fail	Total Participation rule broken
16	BankAccount	ISA	PaymentTypes	1:1	Pass	None
17	Card	ISA	PaymentTypes	1:1	Pass	None
18	Faculty	Instruct	Course	1:M	Fail	Total Participation rule broken
19	Faculty	Chair	Departments	1:1	Fail	Constraint was reversed. A Faculty member might not chair a Department
20	Faculty	Works_In	Departments	M:1	Pass	None

Rule	Entity A	Relation	Entity B	Cardinality	Pass /Fail	Error Description
21	Finance	Manages	Fees	M:M	Fail	Total Participation rule broken
22	Finance	Manages	Grants	M:M	Fail	Total Participation rule broken
23	Administration	Manages	GERequirements	M:M	Fail	Total Participation rule broken
24	Administration	Manages	CollegeRequirement	M:M	Fail	Total Participation rule broken
25	Departments	Manages	CoreRequirements	M:M	Pass	None
26	Departments	Offers	Majors	1:M	Pass	None
27	Departments	Office	Rooms	1:1	Fail	Total Participation rule broken
28	Departments	Offers	Courses	1:M	Pass	None
29	Courses	Has	Fees	M:M	Pass	None
30	Courses	Has	Sections	1:M	Pass	None
31	Courses	Has	Evaluations	1:M	Pass	None
32	Courses	Satisfies	GERequirements	M:M	Pass	None
33	Courses	Satisfies	CollegeRequirements	M:M	Pass	None
34	Courses	Satisfies	CoreRequirements	M:M	Pass	None
35	Sections	Located	Rooms	M:1	Fail	Constraint fails. A section might exist without a room, ex - online learning
36	Sections	Scheduled	Timeslots	M:M	Pass	None
37	Sections	Scheduled	WorkingDays	M:M	Pass	None
38	Rooms	Located	Buildings	M:1	Pass	None
39	Buildings	Located	Locations	1:1	Pass	None
40	Timeslots	Scheduled	WorkingDays	M:M	Pass	None
41	Majors	Has	Degree	M:M	Pass	None
42	Degree	Has	DegreeLevels	M:M	Pass	None
43	Degree	Has	GERequirements	M:M	Pass	None
44	Degree	Has	CollegeRequirements	M:M	Pass	None
45	Degree	Has	CoreRequirements	M:M	Pass	None

Section VII: Database Model/EER



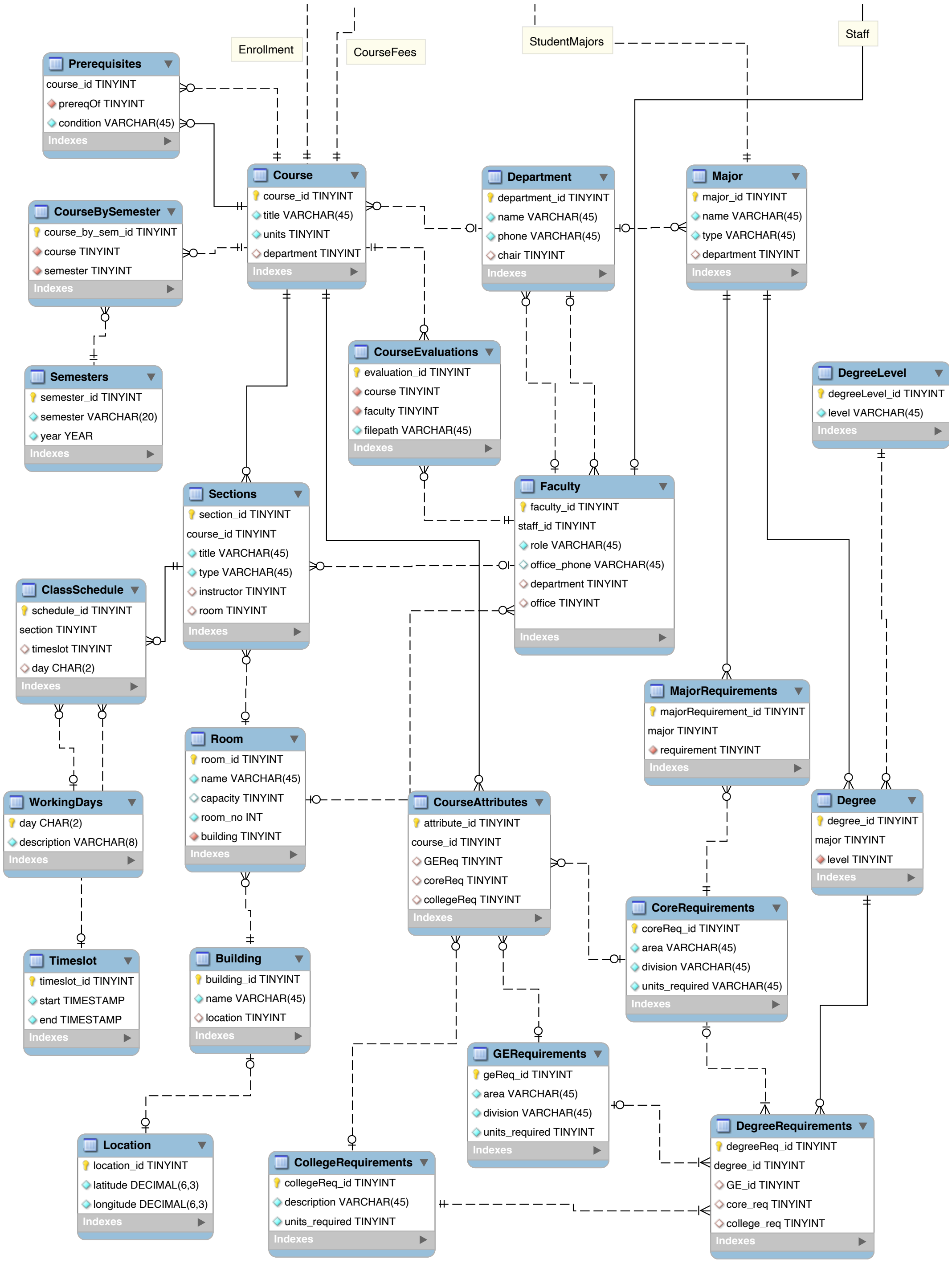

Enrollment
CourseFees
StudentMajors
Staff
Prerequisites
course_id TINYINT
prereqOf TINYINT
condition VARCHAR(45)
Indexes
Course
course_id TINYINT
title VARCHAR(45)
units TINYINT
department TINYINT
Indexes
Department
department_id TINYINT
name VARCHAR(45)
phone VARCHAR(45)
chair TINYINT
Indexes
Major
major_id TINYINT
name VARCHAR(45)
type VARCHAR(45)
department TINYINT
Indexes
CourseBySemester
course_by_sem_id TINYINT
course TINYINT
semester TINYINT
Indexes
Semesters
semester_id TINYINT
semester VARCHAR(20)
year YEAR
Indexes
CourseEvaluations
evaluation_id TINYINT
course TINYINT
faculty TINYINT
filepath VARCHAR(45)
Indexes
DegreeLevel
degreeLevel_id TINYINT
level VARCHAR(45)
Indexes
Sections
section_id TINYINT
course_id TINYINT
title VARCHAR(45)
type VARCHAR(45)
instructor TINYINT
room TINYINT
Indexes
Faculty
faculty_id TINYINT
staff_id TINYINT
role VARCHAR(45)
office_phone VARCHAR(45)
department TINYINT
office TINYINT
Indexes
ClassSchedule
schedule_id TINYINT
section TINYINT
timeslot TINYINT
day CHAR(2)
Indexes
MajorRequirements
majorRequirement_id TINYINT
major TINYINT
requirement TINYINT
Indexes
Room
room_id TINYINT
name VARCHAR(45)
capacity TINYINT
room_no INT
building TINYINT
Indexes
WorkingDays
day CHAR(2)
description VARCHAR(8)
Indexes
CourseAttributes
attribute_id TINYINT
course_id TINYINT
GEReq TINYINT
coreReq TINYINT
collegeReq TINYINT
Indexes
Degree
degree_id TINYINT
major TINYINT
level TINYINT
Indexes
CoreRequirements
coreReq_id TINYINT
area VARCHAR(45)
division VARCHAR(45)
units_required VARCHAR(45)
Indexes
Timeslot
timeslot_id TINYINT
start TIMESTAMP
end TIMESTAMP
Indexes
Building
building_id TINYINT
name VARCHAR(45)
location TINYINT
Indexes
GERequirements
geReq_id TINYINT
area VARCHAR(45)
division VARCHAR(45)
units_required TINYINT
Indexes
Location
location_id TINYINT
latitude DECIMAL(6,3)
longitude DECIMAL(6,3)
Indexes
CollegeRequirements
collegeReq_id TINYINT
description VARCHAR(45)
units_required TINYINT
Indexes
DegreeRequirements
degreeReq_id TINYINT
degree_id TINYINT
GE_id TINYINT
core_req TINYINT
college_req TINYINT
Indexes

Table	FK	ON DELETE	ON UPDATE	Comment
StudentAddress	student	ON CASCADE	ON CASCADE	If a student is deleted then their address information should be deleted as well
StudentAddress	address	ON CASCADE	ON CASCADE	If an address is deleted then its respective entry in the StudentAddress table needs to be removed
Transcripts	student_id	NO ACTION	ON CASCADE	Transcripts are important college records and should not be removed if the student is deleted
Student	standing	SET NULL	ON CASCADE	A newly admitted student may not yet have an academic standing
Student	acadLevel	SET NULL	ON CASCADE	A student may temporarily not have an assigned academic level as they advance through the next level
StudentPaymentInfo	student	ON CASCADE	ON CASCADE	If a student is deleted then their payment information should be deleted as well
StudentPaymentInfo	paymentType	ON CASCADE	ON CASCADE	If a payment type is no longer supported by the college then all respective payment information should be deleted as well
StaffAddress	staff	ON CASCADE	ON CASCADE	If a staff member is deleted then their address information should be deleted as well
StaffAddress	address	ON CASCADE	ON CASCADE	If an address is deleted then its respective entry in the StaffAddress table needs to be removed
HonorsStudents	**honor_id**	ON CASCADE	ON CASCADE	If an honor is deleted the information related to it should be deleted as well
HonorsStudents	**student_id**	ON CASCADE	ON CASCADE	If a student is deleted then the information related to it in the HonorStudent should be deleted as well
StaffPaymentInfo	staff	ON CASCADE	ON CASCADE	If a staff member is deleted then their payment information should be deleted as well
StaffPaymentInfo	paymentType	ON CASCADE	ON CASCADE	If a payment type is no longer supported by the college then all respective payment information should be deleted as well
BankAccount	paymentType	ON CASCADE	ON CASCADE	Since a bank account is a payment type, if a payment type is deleted then respective bank account should be deleted
CreditCard	paymentType	ON CASCADE	ON CASCADE	Since a credit card is a payment type, if a payment type is deleted then respective credit card should be deleted
GrantRecipients	**student**	ON CASCADE	ON CASCADE	If a student is deleted then the information related to it in the GrantRecipient should be deleted as well
GrantRecipients	**grant**	ON CASCADE	ON CASCADE	If a grant is deleted then the information related to it should be deleted as well

Table	FK	ON DELETE	ON UPDATE	Comment
StudentFees	student	ON CASCADE	ON CASCADE	If a student is deleted then the information related to it in the GrantRecipient should be deleted as well
StudentFees	fee	ON CASCADE	ON CASCADE	If a fee is deleted then then the charges made to students with that fee should be deleted as well
Finance	staff_id	ON CASCADE	ON CASCADE	Since a finance member is a staff member, if the staff member is deleted then respective finance member should be deleted
Enrollment	student	ON CASCADE	ON CASCADE	If a student is deleted then their enrollment information related should be deleted as well
Enrollment	course	ON CASCADE	ON CASCADE	If a course is deleted then its enrollment information related should be deleted as well
CourseFees	course	ON CASCADE	ON CASCADE	If a course is deleted then all fees related to it should be deleted as well
CourseFees	fee	ON CASCADE	ON CASCADE	If a certain fee is no longer supported by the college then courses cannot charge that fee
StudentMajors	student	ON CASCADE	ON CASCADE	If a student is deleted then the information related to it in the StudentMajors should be deleted as well
StudentMajors	major	ON CASCADE	ON CASCADE	If a major is deleted then the all students cannot be in that major anymore
Prerequisites	prereqOf	ON CASCADE	ON CASCADE	If a course is deleted then it's prerequisites should be removed as well
Course	department	SET NULL	ON CASCADE	A course may not necessarily be offered by a specific department yet
Major	department	SET NULL	ON CASCADE	A major may not necessarily be offered by a specific department yet
CourseEvaluations	course	NO ACTION	ON CASCADE	CourseEvaluations are important college records and should not be removed if the course is deleted
CourseEvaluations	faculty	NO ACTION	ON CASCADE	CourseEvaluations are important college records and should not be removed if the student is deleted
CourseBySemester	course	ON CASCADE	ON CASCADE	If a course is deleted then it cannot be offered in any semester
Sections	course_id	ON CASCADE	ON CASCADE	Since a section belongs to a course, if a course is deleted then its sections should be deleted as well

Table	FK	ON DELETE	ON UPDATE	Comment
Sections	instructor	SET NULL	ON CASCADE	A section may temporarily not have an instructor
Sections	room	SET NULL	ON CASCADE	A section may temporarily not have a classroom
Faculty	staff_id	ON CASCADE	ON CASCADE	Since a faculty member is a staff member, if the staff member is deleted then its respective faculty member should be deleted
Faculty	department	SET NULL	ON CASCADE	If a department is removed from the college then its faculty members should not be removed, they may change departments
Faculty	office	SET NULL	ON CASCADE	If a faculty member is no longer assigned an office then it should be set to null until assigned a new office
Administration	staff_id	ON CASCADE	ON CASCADE	Since a finance member is a staff member, if the staff member is deleted then respective finance member should be deleted
ClassSchedule	timeslot	SET NULL	ON CASCADE	If a timeslot is deleted, then it should be set to null in the class schedule until it is assigned a new one
ClassSchedule	section	ON CASCADE	ON CASCADE	If a section is deleted then its class schedule should be deleted as well
ClassSchedule	day	SET NULL	ON CASCADE	If a work day is deleted, then it should be set to null in the class schedule until it is assigned a new one
Room	building	ON CASCADE	ON CASCADE	If a building is deleted then all rooms that is contains should be deleted as well
MajorRequirements	major	ON CASCADE	ON CASCADE	If a major is deleted then its major requirements should be deleted as well
MajorRequirements	requirement	ON CASCADE	ON CASCADE	If a core requirement is no longer offered in the college then majors should not be able to require it
Building	location	SET NULL	ON CASCADE	If a location is deleted then the building located there should have a location set to null
Degree	major	ON CASCADE	ON CASCADE	If a major is no longer offered then all degrees of that major should be deleted as well
Degree	level	ON CASCADE	ON CASCADE	If a degree level is no longer offered then all degrees of that level should be deleted as well
CourseAttributes	course_id	ON CASCADE	ON CASCADE	If a course is deleted then its course attributes should be deleted as well

Table	FK	ON DELETE	ON UPDATE	Comment
CourseAttributes	GEReq	SET NULL	ON CASCADE	If a GE requirement is deleted then a course satisfying it should set it to null in its attributes
CourseAttributes	coreReq	SET NULL	ON CASCADE	If a core requirement is deleted then a course satisfying it should set it to null in its attributes
CourseAttributes	collegeReq	SET NULL	ON CASCADE	If a college requirement is deleted then a course satisfying it should set it to null in its attributes
DegreeRequirements	degree_id	ON CASCADE	ON CASCADE	If a degree is deleted then its degree requirements should be deleted as well
DegreeRequirements	GEReq	SET NULL	ON CASCADE	If a GE requirement is deleted then a degree requiring it should set it to null that requirement
DegreeRequirements	coreReq	SET NULL	ON CASCADE	If a core requirement is deleted then a degree requiring it should set it to null that requirement
DegreeRequirements	collegeReq	SET NULL	ON CASCADE	If a college requirement is deleted then a degree requiring it should set it to null that requirement
Department	chair	SET NULL	ON CASCADE	A department may temporarily not have a department chair, either upon creation or if the current chair is no longer serving

Section VIII: Forward Engineering

Please refer to databasemodel.sql located in milestones/Milestone2/files

Section IX: Inserting Data

Please refer to databasemodel.sql located in milestones/Milestone2/files

Section X: Testing

Please refer to databasemodel.sql located in milestones/Milestone2/files

Section XI: Testing Table

Entity	SQL Query	Pass/Fail	Error Description	Possible Solution
AcademicLevel	DELETE	Pass	None	None
AcademicLevel	UPDATE	Pass	None	None
AcademicStanding	DELETE	Pass	None	None
AcademicStanding	UPDATE	Pass	None	None
Student	DELETE	Fail	Cannot delete or update a parent row	Change the student FK constraint in the Transcripts table from NO ACTION to SET NULL
Student	UPDATE	Pass	None	None
Staff	DELETE	Fail	Cannot delete or update a parent row	Change the faculty FK constraint in the CourseEvaluation table from NO ACTION to SET NULL
Staff	UPDATE	Pass	None	None
Department	DELETE	Pass	None	None
Department	UPDATE	Pass	None	None
Location	DELETE	Pass	None	None
Location	UPDATE	Pass	None	None
Building	DELETE	Pass	None	None
Building	UPDATE	Pass	None	None
Room	DELETE	Pass	None	None
Room	UPDATE	Pass	None	None
Faculty	DELETE	Pass	None	None
Faculty	UPDATE	Fail	Unknown column ‘name’	Wrong attribute specified, use ‘role’ instead
Administration	DELETE	Pass	None	None
Administration	UPDATE	Pass	None	None
Finance	DELETE	Pass	None	None
Finance	UPDATE	Pass	None	None

Entity	SQL Query	Pass/Fail	Error Description	Possible Solution
Address	DELETE	Pass	None	None
Address	UPDATE	Pass	None	None
PaymentType	DELETE	Pass	None	None
PaymentType	UPDATE	Pass	None	None
StudentPaymentInfo	DELETE	Pass	None	None
StudentPaymentInfo	UPDATE	Pass	None	None
StaffPaymentInfo	DELETE	Pass	None	None
StaffPaymentInfo	UPDATE	Pass	None	None
BankAccount	DELETE	Pass	None	None
BankAccount	UPDATE	Fail	Cannot add or update a child row	The ‘paymentType’ FK being updated did not exist. Update with existing records only
CreditCard	DELETE	Pass	None	None
CreditCard	UPDATE	Pass	None	None
StudentAddress	DELETE	Pass	None	None
StudentAddress	UPDATE	Pass	None	None
StaffAddress	DELETE	Pass	None	None
StaffAddress	UPDATE	Pass	None	None
Major	DELETE	Pass	None	None
Major	UPDATE	Pass	None	None
CoreRequirement	DELETE	Pass	None	None
CoreRequirement	UPDATE	Pass	None	None
MajorRequirement	DELETE	Pass	None	None
MajorRequirement	UPDATE	Pass	None	None
StudentMajors	DELETE	Pass	None	None
StudentMajors	UPDATE	Pass	None	None
DegreeLevel	DELETE	Pass	None	None

Entity	SQL Query	Pass/Fail	Error Description	Possible Solution
DegreeLevel	UPDATE	Pass	None	None
Degree	DELETE	Pass	None	None
Degree	UPDATE	Fail	Cannot add or update a child row	The 'major' FK being updated did not exist. Update with existing records only
Course	DELETE	Pass	None	None
Course	UPDATE	Pass	None	None
GERequirement	DELETE	Pass	None	None
GERequirement	UPDATE	Fail	Duplicate entry for key	Unknown. The specified key does not exist in the table
CollegeRequirement	DELETE	Pass	None	None
CollegeRequirement	UPDATE	Pass	None	None
CourseAttribute	DELETE	Pass	None	None
CourseAttribute	UPDATE	Pass	None	None
DegreeRequirement	DELETE	Pass	None	None
DegreeRequirement	UPDATE	Pass	None	None
Semester	DELETE	Pass	None	None
Semester	UPDATE	Pass	None	None
Enrollment	DELETE	Pass	None	None
Enrollment	UPDATE	Pass	None	None
Sections	DELETE	Pass	None	None
Sections	UPDATE	Pass	None	None
Timeslot	DELETE	Pass	None	None
Timeslot	UPDATE	Pass	None	None
WorkingDays	DELETE	Pass	None	None
WorkingDays	UPDATE	FAIL	Data too long for column	Update the description to VARCHAR(9) since 'Wednesday' > 8
ClassSchedule	DELETE	Pass	None	None

Entity	SQL Query	Pass/Fail	Error Description	Possible Solution
ClassSchedule	UPDATE	Pass	None	None
Transcript	DELETE	Pass	None	None
Transcript	UPDATE	FAIL	Out of range value	Change the units_taken from TINYINT to INT
Honors	DELETE	Pass	None	None
Honors	UPDATE	Pass	None	None
HonorStudents	DELETE	Pass	None	None
HonorStudents	UPDATE	Pass	None	None
Grants	DELETE	Pass	None	None
Grants	UPDATE	Pass	None	None
GrantRecipients	DELETE	Pass	None	None
GrantRecipients	UPDATE	Pass	None	None
Fees	DELETE	Pass	None	None
Fees	UPDATE	Pass	None	None
CourseFees	DELETE	Pass	None	None
CourseFees	UPDATE	Pass	None	None
StudentFees	DELETE	Pass	None	None
StudentFees	UPDATE	Pass	None	None
CourseBySemester	DELETE	Pass	None	None
CourseBySemester	UPDATE	Pass	None	None
CourseEvaluation	DELETE	Pass	None	None
CourseEvaluation	UPDATE	Pass	None	None
Prerequisites	DELETE	Pass	None	None
Prerequisites	UPDATE	Pass	None	None
Trigger: addCourse	INSERT	FAIL	None	False positive. Prerequisite data for courses need to be inserted before enrollment
Trigger: updateStudentFee	INSERT	FAIL	Subquery returns more than 1 row	SELECT sum(fee) instead of SELECT fee

Entity	SQL Query	Pass/Fail	Error Description	Possible Solution
Procedure: numStudentsInCourse	SELECT	Pass	None	None
Function: studentBalance	SELECT	Pass	None	None