SCHOOL MANAGEMENT SYSTEM

Mato Ramic

Student Id: 918794492

Github Username: mato-r

Milestone/Version	Date
M2V1	04/19/22
M1V1	03/15/22

Table of Contents

Section I. Project Description

The aim of this project is to build a School Management System Database that will mimic current university systems such as UCLA, San Jose State University, San Francisco State University, etc. The project will help the school better manage the various school departments which consist of many faculty members, students, classes that are being offered in order to facilitate better structure and order. This management helps the school easily consider needs of a school department at any given point such as if a course needs to be added or removed, filling faculty roles where needed, removing students from course roster if a student has dropped a particular course, etc. In terms of managing courses and the various classes offered. For each class there will exist a roster list with seat availability, waitlist amount, class total, and waitlist max. This database will help facilitate the outputs of the current amount for each category. This allows schools to offer an accurate reading on the enrollment totals for classes which is important to have.
The project will help organize the staff across the entirety of the school with each of their responsibilities. A school contains many departments and each department will contain a variety of staff to fill the required areas of need for a given department. In the case of a heavily populated school, a solid database system is needed to account for each student with set information about said student.

As any student who enrolls in a school there should be a place where students can view and track their financial aid, tuition amount if any is offered for a given semester, and any payments made. Tuition amounts for a given semester with the system will need to change depending on a student's course load or currently enrolled courses. This in a sense will create an order that students will need to pay in order to stay enrolled in the

required courses. So this database system will offer means to track payments provided by enrolled students and also in return receipts will need to be created to be sent out to student's payments of their orders.

The School Management Database System as well will handle the tracking of devices that do connect to the school's wifi network. Different departments may have their own wifi network as schools will vary in size. This will monitor the connections that are being made and users that do connect need to provide their registered school email and password. Limiting students or staff to a set amount of connected devices to one specific network is important as it allows for the network to stay fast and reliable. This information will be tracked with this database and will aid in providing the school's expensive resource with more reliability. The goal is to help keep the school's various networks that it may have online safe, reliable, and fast for its students and staff. This creates the most productive environment possible for aspiring students and its well managed staff.

The School Management Database System will need to take into consideration school spending on supplies, department funding for course offerings, faculty member payrolls, campus facilities and its upkeep costs, etc. A school will need to offer food stands or places where students can spend their down time at. This is why this system will provide a solution for schools to track the food stand items that they do receive from the suppliers to supply the food stands. It will also aid in tracking sales so that schools can better gauge what needs to be ordered and what can be ordered less. Other areas of down time include a study room for example which can be offered for students to study or do various assignments. This database system will provide a means to also track room

availability as users need to schedule rooms and when finished the availability of said room should update to show that it is vacant for a certain time period. Book supply and renting of books will also be a point of interest for this database system. This makes tracking and managing school book supply and services easy. Book supply whether bought or rented for a set time period will be adjusted and reflect the current status of a particular book in the system. Hence, a structured overall system such as this one will improve efficiency of the school to better meet the needs of its students and faculty. The management system database will help better organize and manage the school's operation at all times by accounting for any changes or shifts in its funding, faculty, students, departments, facilities. This will increase the school's revenue and cash flow as operations continue to remain in good operation the school can continue to run. The better the school runs the more possibilities for course offerings, more school resources, and activities can be offered. This School Management Database System is perfect for any school wanting to improve its operations and have a better overall structure.

Section II. Use Cases

1. **Use Case**: Difficulty managing Departments

   Actor: School President (Jim), Department, Faculty, Employee

   Description: Jim is the School President of a university in San Francisco. He is stressed because he is unable to know whether a new department can be formed or if a department needs changes such as department employee recruitment or removal. Sometimes departments are not able to offer every course that may have been provided previously due to school funding discrepancies between different semesters. This means an Employee may not be needed to teach a specific course every semester as course availability changes. Jim is looking for a solution to better keep track of available funding in order to better recognize what courses can be offered for any given semester and how many employees are needed to consist within the faculty of the department for a successful semester.

   The School Management System Database will provide Jim with a means of keeping track of discrepancies between different semesters in order to identify what courses can be offered by the department. The system will aid Jim in knowing whether employees need to be added or removed depending on changes among the department. This results in a more structured and organized school when departments are provided with their needed resources.

2. **Use Case**: Enrollment Management of Students

Actor: School President (Mary), Director of Enrollment Management (John), Student (Kevin)

Description: Mary is the School President and is closely working together with the Director of Enrollment Management, John. A school needs a structured system when it comes to enrollment of students. A student, Kevin, is wanting to enroll into multiple courses. He does so by going to his school portal and enrolling into the courses he wants that are available. Any student that enrolls needs to be immediately marked into the system and the classes roster availability should change as well. If students that are enrolled into a given course decide to drop the course, the system should reflect this by increasing seat availability. Both Mary and John need a Management system that will track students who enroll into a course. Basic information should be listed such as Student Id, Full Name, Date of Enrollment for a course for each student that is enrolled into a course. A list of Majors and Minors offered at the given school will be needed to be listed as well.

The School Management System Database will provide Mary and John with a means to accurately track enrollment movement or changes. This will aid John, who is focusing on Enrollment, to know course information such as availability, basic information of enrolling students, course roster shifts. This information provides the school with knowledge of where changes need to be made to different departments based on enrollment patterns, potential future course offerings if more are needed, and credit awarding based on students completion of enrolled courses.

3. **Use Case**: Financials Management for Enrolled Students

Actor: Student (Ana), Student Portal

Description: Ana is a student who is a full time student for the next semester. Ana wants to know her various financial information such as the tuition amount for the semester, financial aid that is available on her account, and payment options. When a student is enrolling into courses tuition costs are adjusted based on the number of units being taken for the semester. A student should be able to access such information in the student portal. This is where sections such as financial aid, tuition cost, and payments can be viewed. To make payments students are rerouted to ePay where they have choices of payment plans if available or not. Payment can be done through sources like Visa/Debit or Paypal. When payments are made this amount should be deducted from students withstanding balances that are displayed inside their student portal. Also, a receipt should be given and sent to the student's email of any payments made. Reminders should be given to students who still need to make any outstanding payments prior to the start of semester sessions.

The School Management System Database will provide effective tracking of students financials such as financial aid being offered, payment options, and tuition fees for a given student. Ana should be able to accurately view her finances on her student portal and be confident that the amounts being displayed for a particular section such as tuition cost, owed amounts, and financial aid are all accurate. A database system that will provide tracking for this information will solve the schools financial system and provide Ana a structured student portal with an accurate financial section.

4. **Use Case**: Internet access and Control/Assignment

   Actor: Student (Juan), Devices, Wifi Network

   Description: The school provides students and employees with wifi access. A student named Juan wants to connect a device to the school's wifi network. To do so they must provide their registered school email. This is also the case for any employee as well a similar procedure is needed. When Juan successfully connects a device to the school's wifi network it is assigned an ip address which is stored by the database. To keep the school's wifi network stable people are limited to 5 devices. Juan decides to link 5 more devices but is noted by the system that only 5 devices can be connected to the school's wifi network at a time by a single user.

   The School Management System Database will allow the school to provide students, faculty, etc with a stable and regulated wifi network. This is important as it allows the school to continue operations and allows students with a reliable network to do assignments, tasks, etc.

5. **Use Case**: School's Food Management/Ordering

   Actor: School, Food Suppliers, Food Stand (Burger Shop), Customer (Tom)

   Description: Food Suppliers are 3rd party services that are not employees of the school but the school handles the orders that they acquire from the suppliers for the food stand items. The school wants a way to track sales of various food stands that are operated by the supplier's team. For example, when a food stand named burger shop sells an item to a

customer named Tom, this order should be tracked. This will allow the school to know whether or not changes in order quantity need to be raised or lowered. Lowering the order quantity when needed will save the school funds that can be redirected elsewhere.

The School Management System Database will provide a better way to manage and track these transactions, which will allow the school to always know the quantity of food/supplies to order and be prepared for every school day.

6. **Use Case**: Library Book Renting, Library Book Buying, updating Book Supply, Managing Book Returns, Check-In Study Rooms
   Actor: Librarian (Sarah), Student (Mia), Student (John)
   Description: Many schools have libraries on campus where students can buy or rent books. Also, study rooms may be available for students to check in. The Librarian of the school's library needs a proper way to manage the inflow and outflow of books that people are either buying or renting. Also, study rooms must be accurately booked and tracked so that students know whether or not there are available rooms. In the case of Mia, a student enrolled in a University, She would like to buy a book and also rent a book. The book buying process can be done simply through online checkout through the school store or bought in person. In terms of renting, they must be a currently enrolled student and a 30 day rental limit is applied. Mia provides her student id, once verified in the system that she is a currently enrolled student she is now allowed to rent the desired book. Book supply is updated regularly as books are either returned or in use. John is another currently enrolled student that wants to schedule a study room session. Study

rooms are limited to a max 2 hour window and a student id must be provided. Availability for study rooms is constantly updated to provide students with accurate representation of the current room status. If available by checking the system, John is able to book the room for a maximum of 2 hours. Once finished the room status is updated to empty.

The School Management System Database will provide proper structure to operate the school library with efficiency. Book supply will be tracked when being rented or bought which can help deter confusion for students when they are wanting to acquire books. Accurate study room availability will provide students with peace of mind knowing that their booked rooms won't be taken by those without appointment.

Section III. Database Requirements

1. **School**
   1.1. A school shall have at least one department
   1.2. A school shall have many enrolled students
   1.3. A school shall have at least one employee
   1.4. A school shall have at least one course
   1.5. A school shall have zero to many food stands
   1.6. A school shall be supplied by zero, one, or many suppliers
   1.7. A school shall offer many majors
   1.8. A school shall offer many minors

2. **Department**
   2.1. A department shall employ zero to many employees
   2.2. A department shall be managed by one and only one head of department
   2.3. A department shall offer zero to many courses
   2.4. A department shall belong to at least one school
   2.5. A department shall be connected to zero, one, or many wifi networks

3. **Head of Department**
   3.1. A head of department is a professor
   3.2. A head of department shall manage one and only one department

4. **Employee**
   4.1. An employee shall be employed to at most one department
   4.2. An employee shall work for at least one school
   4.3. An employee can supervise other employees
   4.4. An employee is a professor, security or janitor
   4.5. An employee shall be able to have at least one payment type

5. **Student**
   5.1. A student shall be enrolled to zero, one, or many schools
   5.2. A student shall be enrolled in at least one class
   5.3. A student shall have one and only one tuition amount per semester
   5.4. A student shall if offered have one financial aid amount per semester
   5.5. A student shall be able to have at least one payment type

6. **Semester**
   6.2. A semester shall have many enrollments

7. **Major**
   7.1. A major shall be associated with one and only one department
   7.2. A major shall or shall not be offered at a school

8. **Minor**

8.1. A minor shall be associated with one and only one department
8.2. A minor shall or shall not be offered at a school

9. Course
9.1. A course shall be offered by one and only one department
9.2. A course shall list at least one class

10. Class
10.1. A class shall be taught by at least one Employee
10.2. A class shall have zero, one or many students enrolled
10.3. A class shall be listed by one and only one course

11. Enrollment
11.1. An enrollment shall be linked to at least one order
11.2. An enrollment shall be belong to one semester at a time
11.3. An enrollment shall belong to one and only one class
11.4. An enrollment can have one student.

12. Financial Aid
12.1. A financial aid amount shall if offered be linked to one student

13. Tuition
13.1. A tuition amount shall be linked to one and only one student
13.2. A tuition amount shall contain one enrollment

14. Order
14.1. An order shall be created for one order list

15. Order List
15.1. An order list shall be linked with one order at a time

16. Payment Type
16.1. A payment type is a bank account, paypal, or cryptocurrency
16.2. A bank account payment type is a credit, or debit card
16.3. A cryptocurrency payment type is Bitcoin or Ethereum

17. Payment
17.1. A payment shall have been made by one and only one payment type
17.2. A payment shall produce only one receipt
17.3. A payment shall be linked to only one order

18. Reciept
18.1. A receipt shall be produced by only one payment at a time

19. Wifi Network

19.1. A wifi network shall be connected to zero to many devices
19.2. A wifi network shall be connected to zero, one, or many departments

20. Device

20.1. A device shall be provided one ip address when connecting to a wifi network
20.2. A device shall be connected to at most one wifi network at a time
20.3. A device shall be registered to one user

21. User

21.1. A user shall be linked to at least one device but at most five devices
21.2. A user is a registered employee or student
21.3. A user shall be able to buy many books
21.4. A user shall be able to rent many books
21.5. A user shall be able to book one study room at a time

22. Food Stand

22.1. A food stand shall receive food stand items from at least one supplier
22.2. A food stand shall serve many food stand items
22.3. A food stand only accepts bank account payment type
22.4. A food stand shall be operated by at least one employee

23. Supplier

23.1. A supplier shall supply a food stand with many food stand items
23.2. A supplier shall create zero to many orders
23.3. A supplier shall supply zero to many schools

24. Food Stand Item

24.2. A food stand item can be served at many food stands

25. Book

25.1. A book shall be bought by only one user at a time
25.2. A book shall be rented by only one user at a time for up to 30 days

26. Study Room

26.1. A study room shall be booked by one user for up to two hours at a time

Section IV. Detailed List of Main Entities, Attributes and Key

1. **School (Strong)**
    - school_id: key, composite
    - name: composite, alphanumeric
    - address: alphanumeric, composite, multivalue
        I. street
        II. city
        III. zip
        IV. state
        V. country
    - phone: numeric, multivalue, composite
        I. country_code
        II. area_code
        III. phone_number

2. **Department (Strong)**
    - department_id: key, numeric
    - name: composite, alphanumeric
    - phone: numeric, multivalue, composite
        I. country_code
        II. area_code
        III. phone_number

3. **Head of Department (Weak)**
    - head_of_department_id: key, numeric
    - department_id: key, numeric
    - name: composite, alphanumeric
    - phone: numeric, multivalue, composite
        I. country_code
        II. area_code
        III. phone_number

4. **Employee (Weak)**
    - employee_id: key, numeric
    - department_id: key, numeric
    - user_id: key, numeric
    - name: composite, alphanumeric
    - ssn: numeric
    - dob: multivalue, timestamp
    - date_of_hire: multivalue, timestamp
    - phone: numeric, multivalue, composite
        I. country_code

II. area_code
III. phone_number

5. **Student (Strong)**
    - student_id: key, alphanumeric
    - user_id: key, numeric
    - name: composite, alphanumeric
    - dob: multivalue, timestamp
    - phone: numeric, multivalue, composite
        I. country_code
        II. area_code
        III. phone_number

6. **Semester (Strong)**
    - semester_id: key, numeric
    - name: composite, alphanumeric
    - start_date: date, composite
    - end_date: date, composite

7. **Major (Strong)**
    - major_id: key, numeric
    - department_id: key, numeric
    - name: composite, alphanumeric

8. **Minor (Strong)**
    - minor_id: key, numeric
    - department_id: key, numeric
    - name: composite, alphanumeric

9. **Course (Strong)**
    - course_id: key, numeric
    - school_id: key, numeric
    - department_id: key, numeric
    - name: alphanumeric
    - course_number: numeric
    - description: alphanumeric

10. **Class (Weak)**
    - class_id: key, numeric
    - course_id: key, numeric
    - employee_id: key, numeric
    - class_limit: numeric
    - waitlist_limit: numeric
    - class_total: numeric
    - waitlist_total: numeric

11. Enrollment (Weak)

- enrollment_id: key, numeric
- class_id: key, numeric
- student_id: key, numeric
- semester_id: key, numeric
- reciept_id: key, numeric
- date: multivalue, timestamp

12. Financial Aid (Weak)

- financial_id: key, numeric
- student_id: key, numeric
- date: multivalue, timestamp
- amount: numeric
- type: composite, alphanumeric

13. Tuition (Weak)

- tuition_id: key, numeric
- student_id: key, numeric
- enrollment_id: key, numeric
- date: multivalue, timestamp
- amount: numeric

14. Order (Strong)

- order_id: key, numeric
- date: multivalue, timestamp
- amount: numeric

15. Order List (Weak)

- order_list_id: key, numeric
- order_id: key, numeric
- item: composite, alphanumeric

16. Payment Type (Strong)

- payment_type_id: key, numeric
- student_id: key, numeric
- employee_id: key, numeric
- payment_type_name: composite, alphanumeric

17. Payment (Weak)

- payment_id: key, numeric
- order_id: key, numeric
- reciept_id: key, numeric
- student_id: key, numeric
- employee_id: key, numeric
- date: multivalue, timestamp
- amount: numeric

- balance: numeric, derived

18. Reciept (Weak)

- reciept_id: key, numeric
- order_id: key, numeric
- payment_id: key, numeric
- student_id: key, numeric
- employee_id: key, numeric
- date_of_payment: multivalue, timestamp
- amount: numeric

19. Wifi Network (Strong)

- network_id: key, numeric
- SSID: alphanumeric
- password: alphanumeric
- router_brand: alphanumeric
- name: alphanumeric

20. Device (Strong)

- device_id: key, numeric
- user_id: key, numeric
- name: composite, alphanumeric
- ip: composite, alphanumeric
- connected_date: multivalue, timestamp

21. User (Strong)

- user_id: key, numeric
- email: composite, alphanumeric
- password: composite, alphanumeric

22. Food Stand (Strong)

- food_stand_id: key, numeric
- employee_id: key, numeric
- stand_name: alphanumeric
- phone: numeric, multivalue, composite
    I. country_code
    II. area_code
    III. phone_number

23. Supplier (Strong)

- supplier_id: key, numeric
- supplier_name: composite, alphanumeric
- phone: numeric, multivalue, composite

I. country_code
II. area_code
III. phone_number

24. Food Stand Item (Weak)

- food_stand_item_id: key, numeric
- food_stand_id: key, numeric
- item_name: composite, alphanumeric
- item_cost: numeric

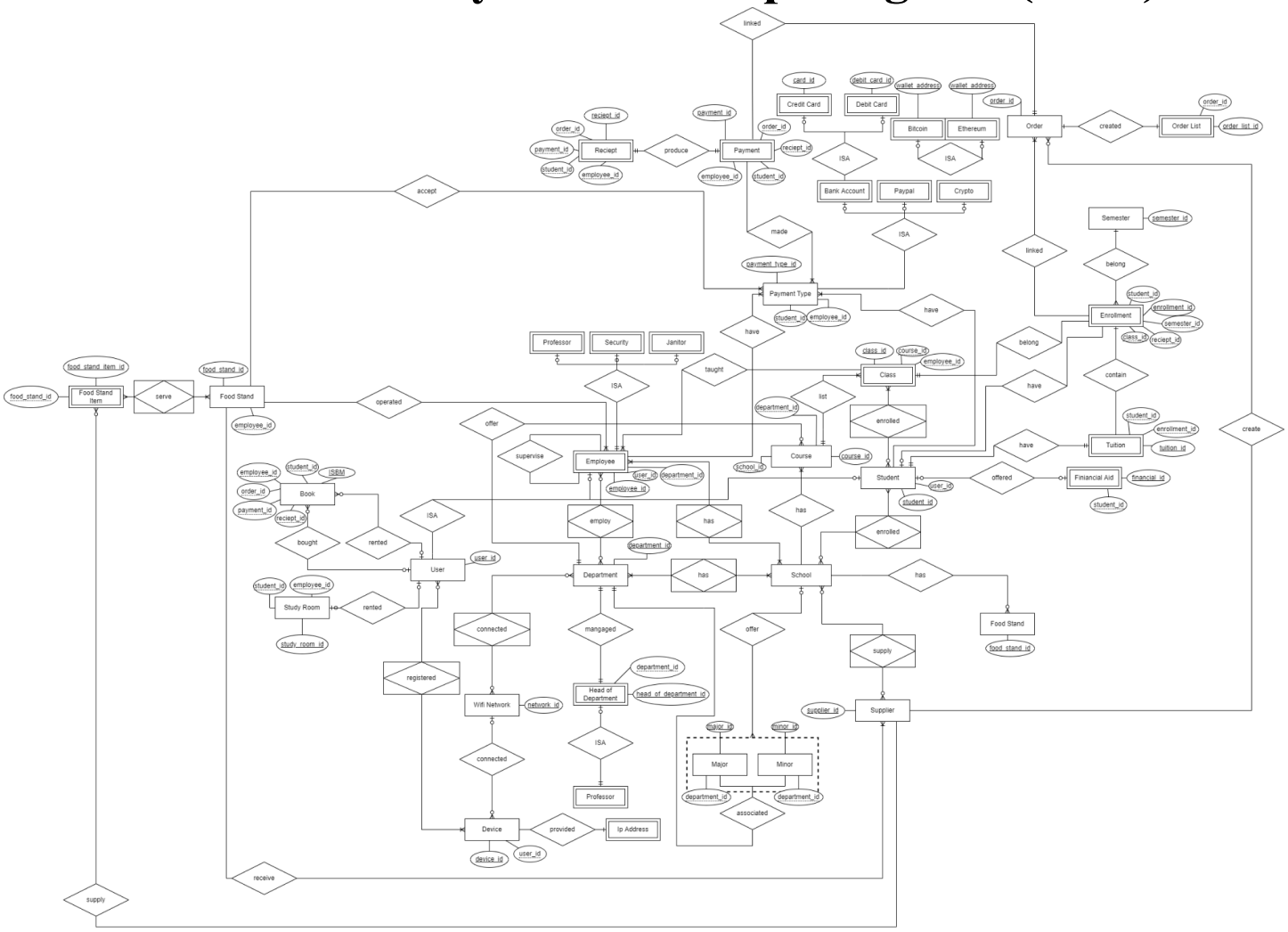
25. Book (Strong)

- ISBM: key, alphanumeric
- student_id: key, numeric
- employee_id: key, numeric
- order_id: key, numeric
- payment_id: key, numeric
- reciept_id: key, numeric
- title: composite, alphanumeric
- author: composite, alphanumeric
- bought_date: multivalue, timestamp
- rent_date: multivalue, timestamp
- start_time: multivalue, timestamp
- end_time: multivalue, timestamp

26. Study Room (Strong)

- study_room_id: key, numeric
- student_id: key, numeric
- employee_id: key, numeric
- room_number: alphanumeric
- booked_room_date: multivalue, timestamp
- start_time: multivalue, timestamp
- end_time: multivalue, timestamp

Section V. Entity Relationship Diagram (ERD)



Section VI. Testing Table

Rule	Entity A	Relation	Entity B	Cardinality	Pass/Fail	Error Description
1	School	Have	Department	M-to-M	Pass	None
2	School	Enrolled	Students	M-to-M	Pass	None
3	School	Have	Employee	M-to-M	Pass	None
4	School	Have	Course	M-to-N	Pass	None
5	School	Have	Food Stands	M-to-N	Pass	None
6	School	Supplied	Supplier	M-to-M	Pass	None
7	School	Offer	Major	M-to-1	Pass	None
8	School	Offer	Minor	M-to-1	Pass	None
9	Department	Employ	Employees	M-to-1	Fail	An employee can work for more than one Department
10	Department	Managed	Head of Department	1-to-1	Pass	None
11	Department	Offer	Courses	M-to-1	Pass	None
12	Department	Belong	School	M-to-M	Pass	None
13	Department	Connected	Wifi Networks	M-to-M	Pass	None
14	Head of Department	ISA	Professor	1-to-1	Pass	None

Rule	Entity A	Relation	Entity B	Cardinality	Pass/Fail	Error Description
15	Head of Department	Manage	Department	1-to-1	Pass	None
16	Employee	Employed	Department	1-to-M	Fail	An employee may teach for more than one department
17	Employee	Work	School	M-to-M	Pass	None
18	Employee	Supervise	Employee	1-to-M	Pass	None
19	Employee	ISA	Professor	1-to-1	Pass	None
20	Employee	ISA	Security	1-to-1	Pass	None
21	Employee	ISA	Janitor	1-to-1	Pass	None
22	Employee	Have	Payment Type	M-to-N	Pass	None
23	Student	Enrolled	School	M-to-M	Pass	None
24	Student	Enrolled	Class	M-to-M	Pass	None
25	Student	Have	Tuition	1-to-1	Pass	None
26	Student	Offered	Financial Aid	1-to-1	Pass	None
27	Student	Have	Payment Type	M-to-N	Pass	None
28	Semester	Have	Enrollment	M-to-1	Pass	None
29	Major	Associated	Department	1-to-N	Pass	None

Rule	Entity A	Relation	Entity B	Cardinality	Pass/Fail	Error Description
30	Major	Offered	School	1-to-M	Pass	None
31	Minor	Associated	Department	1-to-N	Pass	None
32	Minor	Associated	School	1-to-M	Pass	None
33	Course	Offered	Department	1-to-M	Pass	None
34	Course	List	Class	M-to-1	Pass	None
35	Class	Taught	Employee	M-to-N	Pass	None
36	Class	Have	Students	M-to-M	Pass	None
37	Class	Listed	Course	1-to-M	Pass	None
38	Enrollment	Linked	Order	M-to-N	Pass	None
39	Enrollment	Belong	Semester	1-to-M	Pass	None
40	Enrollment	Belong	Class	1-to-N	Pass	None
41	Enrollment	Have	Student	1-to-N	Pass	None
42	Financial Aid	Offered	Student	1-to-1	Pass	None
43	Tuition	Linked	Student	1-to-1	Pass	None
44	Tuition	Contain	Enrollment	1-to-N	Pass	None
45	Order	Created	Order List	1-to-1	Pass	None
46	Order List	Linked	Order	1-to-1	Pass	None
47	Payment Type	ISA	Bank Account	1-to-1	Pass	None

Rule	Entity A	Relation	Entity B	Cardinality	Pass/Fail	Error Description
48	Payment Type	ISA	Paypal	1-to-1	Pass	None
49	Payment Type	ISA	Cryptocurrency	1-to-1	Pass	None
50	Bank Account Payment Type	ISA	Credit Card	1-to-1	Pass	None
51	Bank Account Payment Type	ISA	Debit Card	1-to-1	Pass	None
52	Cryptocurrency Payment Type	ISA	Bitcoin	1-to-1	Pass	None
53	Cryptocurrency Payment Type	ISA	Ethereum	1-to-1	Pass	None
54	Payment	Made By	Payment Type	1-to-N	Fail	A payment if offered can be split into multiple payment types
55	Payment	Produce	Receipt	1-to-1	Pass	None
56	Payment	Linked	Order	1-to-N	Pass	None
57	Receipt	Produced	Payment	1-to-1	Pass	None

Rule	Entity A	Relation	Entity B	Cardinality	Pass/Fail	Error Description
58	Wifi Network	Connected	Device	M-to-1	Pass	None
60	Wifi Network	Connected	Department	M-to-M	Pass	None
61	Device	Provided	Ip Address	1-to-N	Pass	None
62	Device	Connected	Wifi Network	1-to-M	Pass	None
63	Device	Registered	User	1-to-M	Fail	More than one user can register on the same device
64	User	Linked	Device	M-to-1	Fail	More than one user can be linked to the same device
65	User	ISA	Employee	1-to-1	Pass	None
66	User	ISA	Student	1-to-1	Pass	None
67	User	Buy	Book	M-to-1	Pass	None
68	User	Rent	Book	M-to-1	Pass	None
69	User	Book	Study Room	1-to-1	Pass	None
70	Food Stand	Receive	Supplier	M-to-N	Pass	None
71	Food Stand	Serve	Food Stand Item	M-to-M	Pass	None

Rule	Entity A	Relation	Entity B	Cardinality	Pass/Fail	Error Description
72	Food Stand	Accepts	Bank Account Payment Type	1-to-N	Fail	A Food Stand may accept other forms of payment.
73	Food Stand	Operated	Employee	M-to-N	Pass	None
74	Supplier	Supply	Food Stand Item	M-to-N	Pass	None
75	Supplier	Create	Order	M-to-N	Pass	None
76	Supplier	Supply	School	M-to-M	Pass	None
77	Food Stand Item	Served	Food Stand	M-to-M	Pass	None
78	Book	Bought	User	1-to-M	Pass	None
79	Book	Rented	User	1-to-M	Pass	None
80	Study Room	Booked	User	1-to-1	Pass	None

Section VII: Database Model/EER

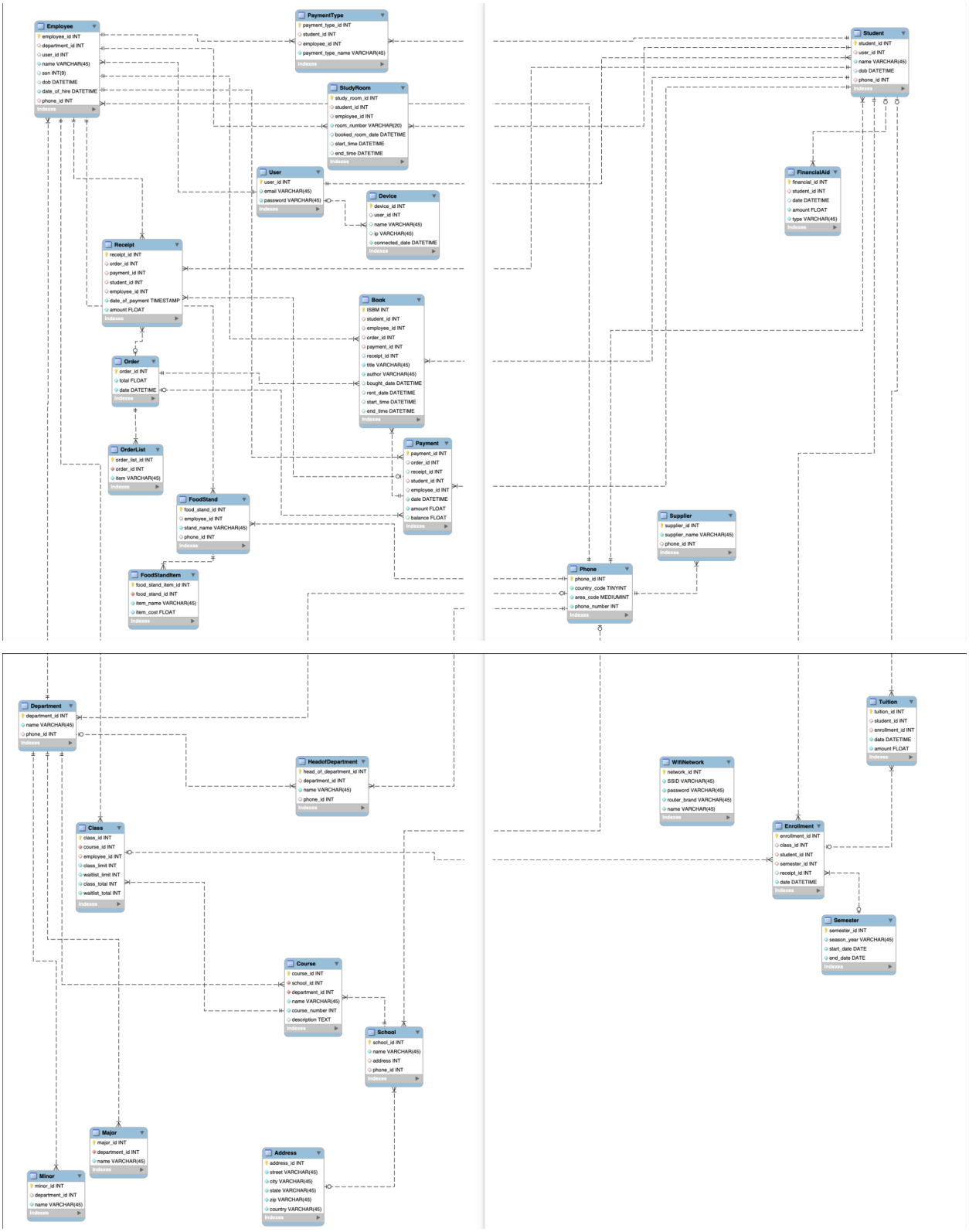


Table	FK	ON DELETE	ON UPDATE	Comment
Book	fk_student_id_book	SET NULL	NO ACTION	If the student is deleted then the book is available for others so this should be set to null. This would be the case if they stopped renting it for example or their time period ended.
Book	fk_employee_id_book	SET NULL	NO ACTION	If the employee is deleted then the book is available for others so this should be set to null. This would be the case if they stopped renting it for example or their time period ended.
Book	fk_order_id_book	SET NULL	NO ACTION	If the order is deleted then the book is available for others so this should be set to null.
Book	fk_payment_id_book	CASCADE	NO ACTION	If the payment gets deleted the book was mostly purchased so it can also be removed from available inventory.
Class	fk_course_i	CASCADE	NO	If a course is deleted then the

Table	FK	ON DELETE	ON UPDATE	Comment
	d_class		ACTION	classes listed within the course shall also be deleted as it is no longer offered at the school.
Class	fk_employee_id_class	SET NULL	NO ACTION	If an employee is removed from the class such as the teacher then the position is empty until the department fills the position. The class shall remain offered for students even while the teacher role is currently empty for the time being.
Course	fk_school_id_course	CASCADE	NO ACTION	If the school is no longer in operation, then the course is no longer able to be offered and shall be removed upon the deletion of school.
Course	fk_department_id_course	CASCADE	NO ACTION	If the department is removed then the courses that were also offered under the department shall be removed from the system.

Table	FK	ON DELETE	ON UPDATE	Comment
Department	fk_phone_id_department	SET NULL	NO ACTION	The department shall still exist if phone is deleted
Device	fk_user_id_device	CASCADE	NO ACTION	If the user_id is deleted all devices linked to that user shall also be deleted from the system.
Employee	fk_department_id_employee	CASCADE	NO ACTION	If the department is removed then the employees that were working under that department shall be removed from that department. They may have been working at more than one department so we don't want to remove the employee itself from the system just from that department's employee list.
Employee	fk_user_id_employee	SET NULL	NO ACTION	If the user id is removed the employee shall still exist as the user id is created when the employee connects to the wifi network it shall be empty for the time

Table	FK	ON DELETE	ON UPDATE	Comment
				being.
Employee	fk_phone_id_employee	SET NULL	NO ACTION	The employee shall still exist if the phone is deleted from the system.
Enrollment	fk_class_id_enrollment	CASCADE	NO ACTION	If the class is removed from the system, then the student's enrollment to the specific class is also removed from the system.
Enrollment	fk_student_id_enrollment	CASCADE	NO ACTION	If the student is no longer in attendance at the school they are removed from the system so there is no need to keep enrollment information on said student.
Enrollment	fk_semester_id_enrollment	CASCADE	NO ACTION	If the semester is no longer in progress or is not going to be offered by the school then the enrollment information is not needed for that particular semester.
FinancialAid	fk_student_i	CASCADE	NO	If the student is no longer

Table	FK	ON DELETE	ON UPDATE	Comment
	d_financial_aid		ACTION	attending the school then the financial aid no longer needs to be accounted for by the school.
FoodStand	fk_employee_id_food_stand	SET NULL	NO ACTION	The food stand shall still remain if the employee position is empty. As the food stand needs to simply fill the employee role with a new employee.
FoodStand	fk_phone_id_food_stand	SET NULL	NO ACTION	The food stand shall still exist if the phone is deleted from the system.
FoodStandItem	fk_food_stand_id_food_stand_item	CASCADE	NO ACTION	If a food stand is no longer operating then the food stand items will no longer be sold from that food stand. The food stand item depends on the food stand to be sold.
HeadofDepartment	fk_department_id_head_of_department	CASCADE	NO ACTION	If the department is removed from the school, the head of department shall also be relieved

Table	FK	ON DELETE	ON UPDATE	Comment
				as well from his role as head of department.
HeadofDepartment	fk_phone_id_head_of_department	SET NULL	NO ACTION	The head of department shall still exist if the phone is deleted.
Major	fk_department_id_major	CASCADE	NO ACTION	If a department is deleted from the school then the majors that were listed under the department shall also be removed. As the school can only offer the majors that it has departments for.
Minor	fk_department_id_minor	CASCADE	NO ACTION	If a department is deleted from the school then the minors that were listed under the department shall also be removed. As the school can only offer the minors that it has departments for.
Order List	fk_order_id_order_list	CASCADE	NO ACTION	If the order is deleted then the order list shall be deleted as well since there is no longer that order to be fulfilled.

Table	FK	ON DELETE	ON UPDATE	Comment
Payment	fk_order_id_payment	SET NULL	NO ACTION	If the order is not existing then it can be set to null as an order hasn't been made yet.
Payment	fk_student_id_payment	SET NULL	NO ACTION	If the student has been deleted from the system we can assume they are no longer enrolled, we do however still want to keep the payment registered in the system.
Payment	fk_employee_id_payment	SET NULL	NO ACTION	If the employee has been deleted from the system we can assume they are no longer working for the school, we do however still want to keep the payment registered in the system.
PaymentType	fk_student_id_payment_type	SET NULL	NO ACTION	If a student is no longer at the school and is removed from the system. The available payment types the school accepts will still be offered for future students.

Table	FK	ON DELETE	ON UPDATE	Comment
PaymentType	fk_employee_id_payment_type	SET NULL	NO ACTION	If an employee is no longer at the school and is removed from the system. The available payment types the school accepts will still be offered for future employees.
Receipt	fk_order_id_receipt	RESTRICT	NO ACTION	A receipt is created every time someone has paid for an order to an extent or fully. So orders should not be deleted to keep on file for financial reasons.
Receipt	fk_payment_id_receipt	RESTRICT	NO ACTION	The payment should be kept on hand for reference for financial reasons.
Receipt	fk_student_id_receipt	SET NULL	NO ACTION	If the student is no longer in the system the receipt shall remain and will still exist.
Receipt	fk_employee_id_receipt	SET NULL	NO ACTION	IIf the employee is no longer in the system the receipt shall

Table	FK	ON DELETE	ON UPDATE	Comment
				remain and will still exist.
School	fk_address_school	SET NULL	NO ACTION	The school shall still remain if the school address is deleted from the system.
School	fk_phone_id_school	SET NULL	NO ACTION	The school shall still remain if the phone is deleted.
Student	fk_user_id_student	SET NULL	NO ACTION	If the user id is deleted the student shall still exist as the user id gets created when connecting to the wifi network. So it shall remain empty until they connect again.
Student	fk_phone_id_student	SET NULL	NO ACTION	The student shall still remain even if the phone is deleted from the system.
StudyRoom	fk_student_id_study_room	SET NULL	NO ACTION	The study room shall still exist even if it is not currently occupied by a student. This just means that the room is vacant for a set time slot and can be filled.

Table	FK	ON DELETE	ON UPDATE	Comment
StudyRoom	fk_employee_id_study_room	SET NULL	NO ACTION	The study room shall still exist even if it is not currently occupied by an employee. This just means that the room is vacant for a set time slot and can be filled.
Supplier	fk_phone_id_supplier	SET NULL	NO ACTION	The supplier shall still exist if the phone is deleted from the system.
Tuition	fk_student_id_tuition	CASCADE	NO ACTION	If the student is no longer attending the school and in the system the tuition shall be removed from the system as they are no longer in attendance enrolled in classes.
Tuition	fk_enrollment_id_tuition	CASCADE	NO ACTION	If the enrollment of a particular student is no longer in the system then that means that said student is also not enrolled for the semester. So the tuition shall be removed if

Table	FK	ON DELETE	ON UPDATE	Comment
				they are not enrolled for the present semester.

Section VIII: Forward Engineering

File for this section added to github repository

Section IX: Inserting Data

File for this section added to github repository

Section X: Testing

File for this section added to github repository

Section XI: Testing Table

Entity	SQLQuery	Pass/Fail	Error Description	Possible Solution
Address	Delete	Pass	None	None
Address	Update	Pass	None	None
Book	Delete	Pass	None	None
Book	Update	Pass	None	None
Class	Delete	Pass	None	None
Class	Update	Pass	None	None
Course	Delete	Pass	None	None
Course	Update	Pass	None	None
Department	Delete	Pass	None	None
Department	Update	Pass	None	None
Device	Delete	Pass	None	None
Device	Update	Pass	None	None
Employee	Delete	Pass	None	None
Employee	Update	Pass	None	None
Enrollment	Delete	Pass	None	None
Enrollment	Update	Pass	None	None
FinancialAid	Delete	Pass	None	None
FinancialAid	Update	Pass	None	None

Entity	SQLQuery	Pass/Fail	Error Description	Possible Solution
FoodStand	Delete	Pass	None	None
FoodStand	Update	Pass	None	None
FoodStandItem	Delete	Pass	None	None
FoodStandItem	Update	Pass	None	None
HeadofDepartment	Delete	Pass	None	None
HeadofDepartment	Update	Pass	None	None
Major	Delete	Pass	None	None
Major	Update	Pass	None	None
Minor	Delete	Pass	None	None
Minor	Update	Pass	None	None
Order	Delete	Pass	None	None
Order	Update	Pass	None	None
OrderList	Delete	Pass	None	None
OrderList	Update	Fail	Order List Key not found	Key was misspelled, change to order_list_id
Payment	Delete	Fail	Cannot delete or update a parent row:	Change foreign key of Receipt table:

Entity	SQLQuery	Pass/Fail	Error Description	Possible Solution
			a foreign key constraint fails	payment_id from ON DELETE: RESTRICT to CASCADE or SET NULL to allow deletion of payments
Payment	Update	Pass	None	None
PaymentType	Delete	Pass	None	None
PaymentType	Update	Fail	Payment Type Key not found	Key was misspelled, change to payment_type_id
Phone	Delete	Pass	None	None
Phone	Update	Pass	None	None
Receipt	Delete	Pass	None	None
Receipt	Update	Pass	None	None
School	Delete	Pass	None	None
School	Update	Pass	None	None
Semester	Delete	Pass	None	None

Entity	SQLQuery	Pass/Fail	Error Description	Possible Solution
Semester	Update	Pass	None	None
Student	Delete	Pass	None	None
Student	Update	Pass	None	None
StudyRoom	Delete	Pass	None	None
StudyRoom	Update	Pass	None	None
Supplier	Delete	Pass	None	None
Supplier	Update	Pass	None	None
Tuition	Delete	Pass	None	None
Tuition	Update	Pass	None	None
User	Delete	Pass	None	None
User	Update	Pass	None	None
WifiNetwork	Delete	Pass	None	None
WifiNetwork	Update	Pass	None	None