Business Rules of unversity system

1.Role and Permissions:

>Business rule:

· Roles determine what users can do in the system, and permissions are the specific actions allowed.

· Permissions must be assigned to roles, and roles are then assigned to users.

>Entities:

-Role

-Permission

-Role\_Permissions

>Attributes:

-Role:

.Role\_ID,

.role\_name

. description

-Permission:

.Permission\_ID,

.permission\_name,

.description

-Role\_permissions:

.Role\_ID (FK to Role),

. Permission\_ID (FK to Permission).

>Relationships:

· A **Role** can have multiple **Permissions**.

· A **Permission** can be assigned to multiple **Roles**.

· **Cardinality**: Many-to-Many (thurough role\_per)

2.Student and Courses

>Business rule:

· Students must enroll in courses to participate.

· Grades for courses are optional and can be updated when available.

>Entities:

-Student

-Course

-Enrollment

>Attributes:

-Student:

.Student\_ID,

.name,

.email,

.date\_of\_birth,

. level,

.cgpa,

.major\_department\_id,

. minor\_department\_id,

.role\_id

-Course:

.Course\_ID,

.name,

.credits,

.level,

.department\_id

-Enrollment

.Enrollment\_ID,

. Student\_ID (FK to Student),

. Course\_ID (FK to Course),

.enrollment\_date,

. grade

>Relationships:

· A **Student** can enroll in multiple **Courses**.

· A **Course** can have multiple **Students** enrolled.

· **Cardinality**: Many-to-Many (through Enrollment table).

3.Lecturer and Courses

>Business rule:

· Every course must have a supervising lecturer.

· A lecturer can oversee multiple courses but cannot supervise the same course as another lecturer.

>Entities:

-Lecurer

-Course

>Attributes:

-Lecurer:

.Lecturer\_ID,

.name,

.email,

.specialization,

.department\_id,

. role\_id

-Course:

.Course\_ID,

.name,

.credits,

.level,

.department\_id

>Relationships:

· Every course must have a supervising lecturer.

· A lecturer can oversee multiple courses but cannot supervise the same course as another lecturer.

4.Department and students

>Business rule:

Every student must have a major department, and having a minor department is optional.

>Entities:

-Deptartment

-Student

>Attributes:

-Deptartment:

.Department\_ID,

.name,

.faculty\_id

-Student

.student\_ID,

.name,

.email,

.major\_department\_id,

.minor\_department\_id,

.level

.cgpa,

>Relationships:

· A **Department** can have many **Students** as majors.

· A **Student** belongs to one **Major Department** and optionally one **Minor Department**.

· **Cardinality**: One-to-Many.

5.Department and courses

>Business rule:

Each course must belong to a specific department, which is responsible for its management.

>Entities:

-Deptartment

-Courses

>Attributes:

-Deptartment:

.Department\_ID,

.name,

.faculty\_id

-Course:

.Course\_ID,

.name,

.credits,

.level,

.department\_id

>Relationships:

· A **Department** can offer multiple **Courses**.

· A **Course** belongs to one **Department**.

· **Cardinality**: One-to-Many.

6.Student and Activites

>Business rule:

Students can join activities, and the system saves the dates they joined.

>Entities:

-Student

- Activites

-Student\_Activities

>Attributes:

-Student:

.Student\_ID,

.name,

.email,

.date\_of\_birth,

. level,

.cgpa,

.major\_department\_id,

. minor\_department\_id,

.role\_id

- Activites:

.Activity\_ID,

. activity\_name,

.description,

.created\_at

-Student\_Activities:

Student\_ID (FK to Student),

Activity\_ID (FK to Activity), participation\_date

>Relationships:

· A **Student** can participate in multiple **Activities**.

· An **Activity** can involve multiple **Students**.

· **Cardinality**: Many-to-Many (through Student\_Activities table).

7.Library and Books

>Business rule:

Libraries store and manage books. Each book is assigned to one library.

>Entities:

-Library

-Book

>Attributes:

-Library:

.Library\_ID,

.name,

.location

-Book:

.Book\_ID, title,

.author,

.publication\_year,

. genre,

.Library\_ID (FK

>Relationships:

· A **Library** can have multiple **Books**.

· A **Book** belongs to one **Library**.

· **Cardinality**: One-to-Many.

8.Student and Books(Borrowing)

>Business rule:

Borrowing records track when books are taken and returned.

>Entities:

-Student

-Book

-Borrow

>Attributes:

-Student:

.Student\_ID,

.name,

.email,

.date\_of\_birth,

. level,

.cgpa,

.major\_department\_id,

. minor\_department\_id,

.role\_id

-Book:

.Book\_ID, title,

.author,

.publication\_year,

. genre,

.Library\_ID (FK)

-Borrow:

.Borrow\_ID,

.Student\_ID (FK),

.Book\_ID (FK),

.borrow\_date,

.return\_date

>Relationships:

· A **Student** can borrow multiple **Books**.

· A **Book** can be borrowed by one **Student** at a time.

· **Cardinality**: One-to-Many (Borrow transaction).

9.Exams and courses

>Business rule:

Exams are scheduled for courses, and each exam is associated with a course.

>Entities:

-Exam

-Course

>Attributes:

-Exam:

.Exam\_ID,

.exam\_name,

.course\_id

.exam\_date,

.exam\_duration,

.exam\_type

-Course:

.Course\_ID,

.name,

.credits,

.level,

.department\_id

>Relationships:

* A **Course** can have multiple **Exams**.
* An **Exam** belongs to one **Course**.
* **Cardinality**: One-to-Many.

1. Exam Results and students

>Business rule:

Exams are scheduled for courses, and each exam is associated with a course.

>Entities:

-Student

-Exam

-Exam\_Result

>Attributes:

-Student:

.Student\_ID,

.name,

.email,

.date\_of\_birth,

. level,

.cgpa,

.major\_department\_id,

. minor\_department\_id,

.role\_id

-Exam:

Exam\_ID,

.exam\_name,

.course\_id.

.exam\_date,

.exam\_duration,

.exam\_type

-Exam\_Result

Exam\_Result\_ID,

.exam\_id,

.student\_id,

.grade,

>Relationships:

· A **Student** can have multiple **Exam Results**.

· An **Exam Result** belongs to one **Student**.

· **Cardinality**: One-to-Many.