ER Diagram

To design the ER model for the database system described, we'll start by identifying the main entities and their relationships based on the given requirements. Here's the ER model:

Entities:
1. Student
2. Course
3. Major
4. Lecture Period

Relationships:
1. Enrolls_In (between Student and Course)
2. Completed (between Student and Course)
3. Requires (between Major and Course)
4. Prerequisite (between Course and Course)
5. Has_Lecture_Period (between Course and Lecture Period)

Attributes:

Student:
- student_id (Primary Key)
- name
- email
- other relevant student information

Course:
- course_code (Primary Key)
- course_name
- credits
- other relevant course information

Major:
- major_id (Primary Key)
- major_name
- other relevant major information

Lecture Period:
- lecture_period_id (Primary Key)
- day_of_week
- start_time

- end_time
- other relevant lecture period information

Relationship Cardinality and Constraints:
- A student can enroll in multiple courses (M: N relationship between Student and Course).
- A student completes multiple courses (M: N relationship between Student and Course).
- Each major requires multiple courses (M: N relationship between Major and Course).
- Each course can have multiple prerequisites (M: N relationship between Course and Course).
- Each course has multiple lecture periods, and each lecture period can belong to multiple courses (M: N relationship between Course and Lecture Period).

Assumptions:
1. Each course has a unique course code.
2. Each student has a unique student ID.
3. Each major has a unique major ID.
4. Lecture periods are defined by their unique IDs.
5. Prerequisites are stored as relationships between courses, rather than explicit attributes of courses.
6. Other attributes and constraints such as grades in completed courses or specific major requirements are omitted for simplicity.

Shortcomings and Design Decisions:
- The model does not explicitly handle historical data going back 10 years for student results in all courses. This could be addressed by adding a 'semester' attribute to the Completed relationship to specify when the course was completed.
- The model does not include tables for faculty, departments, or semesters, which may be necessary for a comprehensive university database system.
- The model assumes a simple structure for courses and majors; additional complexity may need to be introduced for programs with multiple concentrations or specializations.