# Data Modeling Assignment

## Problem Statement

Consider an application such as Blackboard, an online learning management system (LMS) that provides tools for **faculty** and **students** to create and share a learning experience. **Faculty** can author **courses** that contain **learning modules** broken up into **lessons**. **Modules** and **lessons** can be rearranged into a different order based on the **calendar schedule**. An LMS should provide a set of rich **content widgets** to build each of the various **topics** in a particular **lesson**. Widgets come in variety of **types**: youtube videos, slides, text documents, raw HTML, evaluations, and many more. **Evaluation widgets** can be a simple essay assignment, a submission assignment, or an exam. **Exams** are used to evaluate the **student**'s progress as they answer the various types of **questions** in an exam, such as essay questions, multiple choice questions, fill in the blank questions, and many more types of questions. Based on the **popularity** of courses, **the registrar's office** creates several **sections** for a course for a given **semester**. There are 5 types of semesters: fall, spring, full summer, summer 1 and summer 2. Some of the less popular courses are only taught in particular semester in a given academic year. **Students** enroll in different sections for a course. When registering for a course, students can see a section's **seat capacity**, and the **faculty** assigned to teach the section for a particular course. Undergrad students tend to enroll for many more courses than graduate students. The registrar's office keeps track of **student progress** in the **enrollment**, such as the **final grade**, **letter grade**, and **student feedback**. From time to time, the University asks everyone to verify their **personal profile information** such as **username**, **password**, **first name**, **last name**, **emails**, **phones**, and **addresses**. **Users** can provide multiple emails, addresses and phones. **Faculty** additionally need to update their **benefits**, **tenure status**, **parking**, and **bank account info**. **Students** have to verify their **financial aid info**, **work-study**, and **scholarship**.
Students with scholarships are always keeping an eye on their **gpa** that it does not drop below a certain threshold. Students can see their final grades for a particular course they were enrolled in. **Grades** are neatly broken by the various **assessments** such as assignments and exams. Even down to the points they lost on a particular question based on a **rubric** that keeps track on how much each question was worth on an exam or how much a particular part of an assignment was worth. Students often go to **office hours** to review an evaluation with their **instructor** or **teaching assistant**. They go question by question reviewing their answers and where they might have gone wrong.

- Nouns that are candidate classes or attributes
  User (username, password, first name, last name, emails, phones, addresses),
  registrar's office,
  Faculty (benefits, tenure states, parking, bank account info),
  Students (type: Undergrad, Grad, financial aid info, work-study, scholarship, gpa),
  Courses (popularity),
  Learning modules,
  Lessons (topics),
  Calendar schedule,
  Widgets (types: youtube videos, slides, text documents, raw HTML, evaluations, and many more),
  Evaluation widgets (types: a simple essay assignment, a submission assignment, or an exam),
  Exam (question types: essay questions, multiple choice questions, fill in the blank questions, and many more types of questions),
  Section (semesters: fall, spring, full summer, summer 1 and summer 2, academic year, seat capacity),
  Grades
  Assignment
  Rubric
  Office hours (instructor/teaching assistant)

- List verbs as candidate relations between classes
  - Faculty author courses
  - Courses contain learning modules
  - Learning modules broken up into lessons
  - Modules and lessons can be rearranged into a different order based on the calendar schedule
  - Widgets build topic
  - Evaluation is a widget
  - Evaluation widgets can be essay assignment, submission assignment, or exam
  - Exam evaluate student
  - Questions in exam
  - Registrar's office creates sections based on popularity of courses
  - Sections for a course for a semester
  - Students enroll in sections

    - Section has seat capacity, assigned faculty
    - Registrar's office tracks student in enrollment
    - Grades broken by assessments
    - Assignments and exams are assessments
    - Rubric tracks points of questions in exams or parts in assignments
    - Students go to office hours
    - Office hour has instructor or teaching assistant

- Generalization/specialization (inheritance, if applicable, explain) - show parts of your diagram that specifically illustrates the use of inheritance
  As is shown in the naive UML,
    - the registrar's office, faculty and student extend the abstract class user;
    - youtube videos, slides, text documents, raw HTML and evaluations extends the abstract class widget;
    - essay assignment and submission assignment extend the abstract class assignment;
    - assignment and exam extend the abstract class assessment.

  And all the subclasses possess attributes belong to their superclass and their own.

- Associations, aggregation and/or composition, e.g., empty or filled in diamonds (1 to * or 1 to 1..*, if applicable, explain) - capture any lifecycle dependencies between classes using aggregation or composition.
    - Association is like relationship between faculty and section, faculties are assigned to teach sections for courses.
    - Aggregation is presented by lesson consists of widgets.
    - Composition is the relationship between course and learning modules, if the course is removed, then there shouldn't be any modules exist for this course.

- Classes vs. attributes analysis

  As is shown in naive diagram, I suggest office hour as a attribute of class faculty. However, I soon realize a office hour involves not only instructors, but also students who becoming a TA or making reservation to meet instructors at office hour. Meanwhile office hour should have information about where it will be held. In this case, I change the "office hour" from attribute to a class.

- Naive diagram link: https://drive.google.com/file/d/1PrxcuF-3YBh8GbdnAPaUfWIYbQKdfKwJ/view?
- Final diagram link: https://drive.google.com/file/d/1ZXTgYXE6y25NaufTluurjVOJuMwD3jUz/view?
- Reify process
    - In naive diagram, I used "students enroll in sections" to describe their relationship. This indicated a weak relationship. After referring to the course material, I added a class enrollment. By review the problem statement, I also added some attributes to it. These attributes are for the registrar's office to track student progress.
    - Deleted class calendar schedule. In this schema, I cannot foresee any information required from calendar schedule.  Lesson can be rearrange to the corresponding time frame through an attribute schedule, which also meets the requirement in problem statement.