

# Assignment 3: Requirements Analysis

Levi Olson

## 1. System users

- Instructor - Associate Professor Sean P. Goggins
- TA - Caleb Heinzman
- Students

## 2. User Requirements

- Instructor
  - The instructor shall be able to add/modify/remove assignments as they see fit.
  - The instructor shall be able to view any submitted assignment.
  - The instructor shall be able to transfer or drop students between sections as they see fit.
  - The instructor shall be able to modify a grade for any given assignment
- TA's
  - A TA shall be able to collect and view any given assignment.
  - A TA shall be able to add/modify a grade for any given assignment.
  - A TA shall be able to create comments and notes attached to any given assignment.
- Students
  - A student shall be able to submit their programming assignment.
  - A student shall be able to view all of the necessary requirements for any given assignment.
  - A student shall be able to view the due date of an assignment
  - A student shall be able to view the grade given to their assignments as well as any associated comments with that assignment.

## 3. Relevant Data & Constraints for Each Activity

\*Note: all attributes for each entity are repeatedly listed, the bolded attributes are the ones that pertain to the certain activity.

- Instructor: add/modify/remove assignments
  - Data Entities:
    - Assignment
      - Attributes: **name, grade, date submitted, comments, due date, late due date, requirements**
- Instructor: transfer or drop students between sections
  - Data Entities:
    - Students
      - Attributes: **name, grade**

- Sections
    - Attributes: **number of students**, time slot, room number, **capacity**
    - Constraint: A student can not be transferred to a section that is already full. A check must be made comparing the section number of students attribute to the capacity attribute.
- TA/Instructor: Collect and view any given assignment
  - Data Entities:
    - Assignment
      - Attributes: **name, grade, date submitted, comments, due date, late due date, requirements**
    - Student
      - Attributes: **name**, grade
- TA/Instructor: add/modify a grade for any given assignment
  - Data Entities:
    - Assignment
      - Attributes: **name, grade**, date submitted, comments, **due date, late due date**, requirements
    - Constraint: An assignment can not be changed passed a given date (either a date unique to that assignment or the last day to submit grades)
    - Constraint: If an assignment is submitted past the due date apply a late penalty as needed
- TA: create comments and notes
  - Data Entities:
    - Assignment
      - Attributes: **name**, grade, date submitted, **comments**, due date, late due date, requirements
- Student: submit their programming assignment
  - Data Entities:
    - Student
      - Attributes: **name**, grade
    - Assignment
      - Attributes: **name**, grade, **date submitted**, comments, due date, **late due date**, requirements
    - Constraint: A student shall not be able to submit a programming assignment after all of the other assignments have been graded (after the late due date).
- Student: view requirements for any given assignment
  - Data Entities:
    - Assignment
      - Attributes: **name**, grade, date submitted, comments, due date, late due date, **requirements**
- Student: view due date for any given assignment

- Data Entities
  - Assignment
    - Attributes: **name**, grade, date submitted, comments, **due date**, **late due date**, requirements
- Student: view grade and comments for any given assignment
  - Data Entities:
    - Assignment
      - Attributes: **name**, **grade**, date submitted, **comments**, due date, late due date, requirements

#### 4. System Constraints & Requirements

- The system will be online, so all users will need to be connected to the internet to be able to utilise it. Furthermore the system should be able to handle simultaneous uploads, therefore it will need to have a significant amount of processing power. The system would store assignments on a remote server, which needs to be fairly large to store all of the assignments. Also, this server needs to be running 24/7 so students can submit and access assignments at whatever time is most convenient. The system will utilize a login system where each user has a unique username and password.