Group/Team Members:

Lu Liu

Samantha Hughson

Wurood Mandawi

Ziva Wernick

Project Proposal

1. Main Idea: What is the main purpose of your application? Who are the main users of your application, and what does it do for them?
   1. Clark Computer Science Department Database - For Students, Professors, and Administration
   2. The main purpose of this application would be to help the communication gap between the CS department and the Students. This would be a way for students to check grades, classes and locations, but also for Professors to see all the classes current students have taken within the CS department as well as manage the TAs of the department and other activities (such as lab hours of TAs for students) that need to be available for faculty and student knowledge.
2. User Stories: User stories are short narratives explaining a task that a user needs to conduct and how they will interact with a software system to accomplish their task.
   1. If a student wants to view a syllabus for a past course. First, they will need to login to their account. Go to course Info for past semesters, find the course they are looking for and click it. Then, they can click syllabi to view the syllabus for that course.
   2. A professor can use the grade listing to update the student’s grade for the semester.
   3. Administration can add courses to the listing for the next semester by adding new classes and their locations for students, professors, and TAs will be able to see what class is when and where.
   4. A TA can see what lab times they are in charge of by viewing the Lab schedule.
3. Use Cases: A list of the main things different users can do. If you are familiar with UML, you may create a UML Use Case diagram.

REACH TARGET SAFETY

* 1. Students can
     1. See Grades/ class info for each class the student is in
     2. See Class listings/locations
     3. See Course Info for future semesters/past semesters
        1. Including syllabus
     4. See Syllabus
        1. including required textbooks
     5. Course schedule layout (topics etc)
        1. Related readings and videos
        2. Assignments
     6. Submit Assignments
     7. Submit clicker ID that becomes a part of their profile that can be viewed by every professor they have that pulls that data
     8. Create their own submission page for group assignments (keeps track of assignment and the students in the group each individually get marked as assignment submitted)
     9. Professor info (staff directory)
        1. Office hours
        2. What office they are in
        3. Email
        4. Classes they teach (current and past)
  2. Professors can
     1. See Class info (where it is, what TAs they have)
     2. See Class rosters and connected grades
     3. Attach Syllabus to course info
     4. Create/Edit Syllabus including textbook requirement
     5. Create/edit course schedule layout
     6. Create assignments to be submitted
        1. Must include a hidden solution for TAs that is reasonably achieved by students
     7. Grade Assignments
        1. Submit assignment grade to be viewed by student
        2. Include percentage of course the assignment is worth (percents cannot total over 100, but can be under until a certain deadline)
     8. Edit self profile for staff directory
  3. Administration can
     1. See and approve TA’s hours added
     2. Lab schedule
     3. Enter class information
     4. See professor information
  4. TAs can
     1. enter hours for shifts
     2. See the lab/TA schedule for each class
     3. View Assignment solutions
     4. Class rosters?

1. Data Requirements: What data does your application need to store. You do not yet need to create an ER Diagram or Relational Model. However, you will need to create those models eventually. If it’s helpful to sketch out a preliminary ER model diagram, you may do so.
2. Students profile
   1. Name
   2. Year
   3. Email
   4. Current courses(they are in)
   5. Past courses(they are in)
   6. Clicker ID
   7. Student ID
   8. Is a ta?
   9. Be supervised?
3. Professor profile
   1. Name
   2. Email
   3. Department
   4. Office number
   5. Office hours
   6. Current courses(they teach)ID
   7. Past courses(they teach)ID
   8. Future courses(they teach)ID
   9. Is a Advisor? (Advisor)ID
4. Administration profile
   1. Name
   2. Email
5. TA profile
   1. Name
   2. Email
   3. Student ID
   4. Office hours
   5. Current course(they TA)
   6. Past courses(they TA)
   7. Professor currently under
6. Lab Times
   1. Class name
   2. Class ID
   3. Time reserved
   4. TA for time
7. Course Listing
   1. Name
   2. ID
   3. Semester taught
   4. Year taught
   5. Pre requisites
   6. Professor teaching
8. Class roster
   1. Class name
   2. Sections
   3. Class ID
   4. Student name
   5. Student ID
   6. Semester
   7. Year
9. Student Grades
   1. Class ID
   2. Semester
   3. Year
   4. Grade received
   5. Credits
10. Functionality Requirements: What functionality does your application need to do, which can include user-centric functionality from user stories and use cases, but could also include processing that is invisible to users.
    1. Student profile
       1. Student: update, view
       2. Administration: add, update, select, delete
    2. Professor profile
       1. Professors: update, view
       2. Administration: add, update, select, delete
       3. Students: view (some values)
    3. Administration profile
       1. Administration: add, update, select, delete
       2. Professors: view (some values)
    4. TA profile
       1. Administration: add, update, select, delete
       2. TA: Update
       3. Students: view (some values)
       4. Professors: view, update
    5. Lab Times
       1. Administration: add, update, select, delete
       2. Students: view (some values)
       3. TA: view, Update
    6. Course Listing
       1. Administration: add, update, select, delete
       2. Students: view
       3. Professors: view
    7. Class roster
       1. Administration: add, update, select, delete
       2. Professors: update
       3. Students: view
    8. Student Grades
       1. Administration: add, update, select, delete
       2. Professors: update
       3. Students: view their own

NOTES:

-Grades might be a reach due to security

-Lab Schedule (constraints might be hard)

By Monday:

Highlight the goals

By Wednesday:

Relational Schema