Group 9

ASSIGNMENT 4

Members:
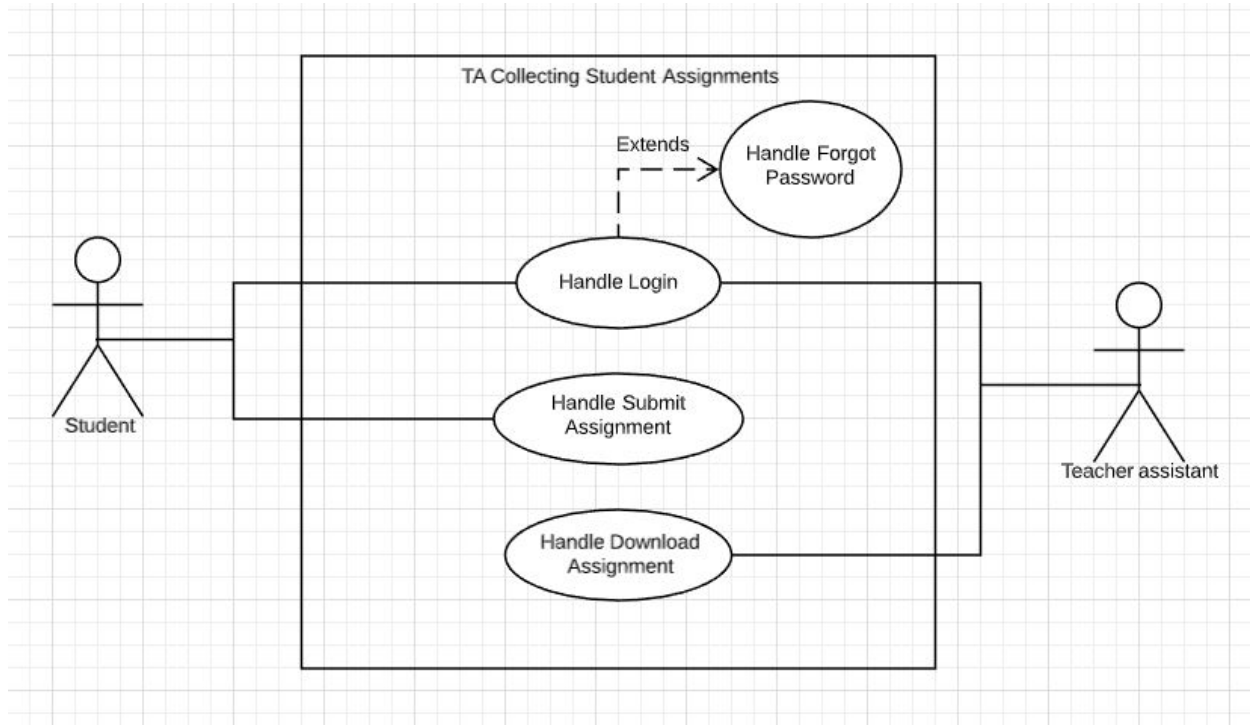**Collin Pounds**
Daniel Por
Isaac Shee

Table of Contents

Use Case Diagram

TA Collecting Student Assignments

Extends — Handle Forgot Password

Handle Login

Handle Submit Assignment

Handle Download Assignment

Student

Teacher assistant

Use Case Description

Primary Path

1. **Student** logs into website.
2. **Student** submits assignment to website.
3. **Teachers Assistant** logs into website
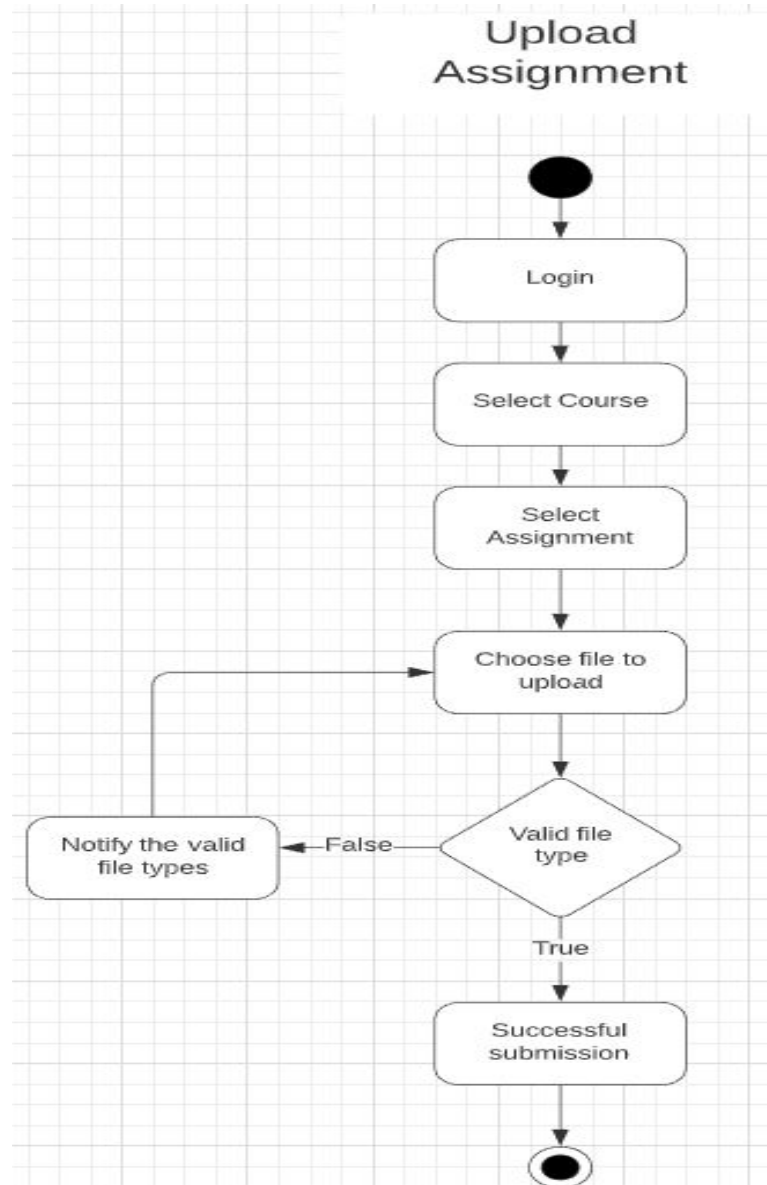4. **Teachers Assistant** downloads student submitted assignment.

Alternative Path

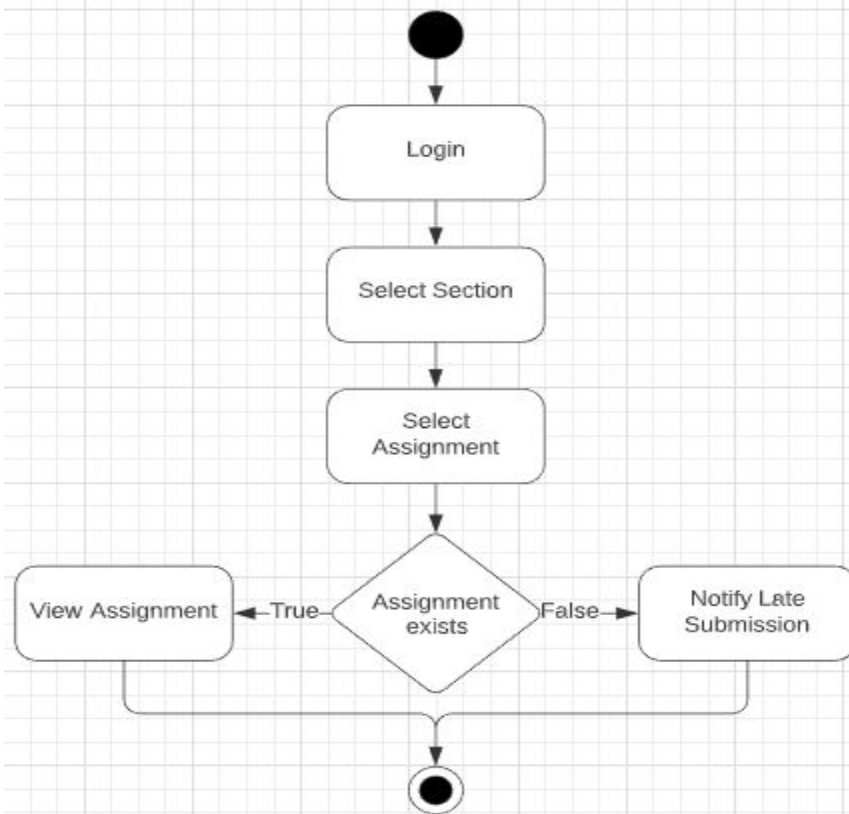1a. **Student** clicks forgot password and is taken to password recovery page

Worked by: Collin, Isaac, Daniel

Activity Diagrams

1. **Student upload assignment activity**
2. **TA viewing assignment**
3. **Instructor add/delete student**

Upload Assignment

```
            ●
            │
            ▼
        ┌────────┐
        │ Login  │
        └────────┘
            │
            ▼
        ┌──────────────┐
        │ Select Course│
        └──────────────┘
            │
            ▼
        ┌──────────────┐
        │    Select    │
        │  Assignment  │
        └──────────────┘
            │
            ▼
        ┌──────────────┐
   ┌───▶│ Choose file  │
   │    │  to upload   │
   │    └──────────────┘
   │        │
   │        ▼
┌──────────────┐        ◇
│ Notify the   │◀─False─ Valid file
│ valid file   │         type
│ types        │          ◇
└──────────────┘          │
                        True
                          │
                          ▼
                  ┌──────────────┐
                  │  Successful  │
                  │  submission  │
                  └──────────────┘
                          │
                          ▼
                          ◉
```

TA view assignment

```
                    ●
                    │
                    ▼
              ┌───────────┐
              │   Login   │
              └───────────┘
                    │
                    ▼
              ┌───────────┐
              │  Select   │
              │  Section  │
              └───────────┘
                    │
                    ▼
              ┌───────────┐
              │  Select   │
              │ Assignment│
              └───────────┘
                    │
                    ▼
                  ╱─────╲
  ┌───────────┐ ╱Assignment╲  ┌────────────┐
  │   View    │◀─True─│ exists │─False─▶│ Notify Late│
  │Assignment │ ╲      ╱       │Submission  │
  └───────────┘  ╲─────╱       └────────────┘
       │                             │
       └──────────────┬──────────────┘
                      ▼
                    ◉
```

Instructor add/delete student
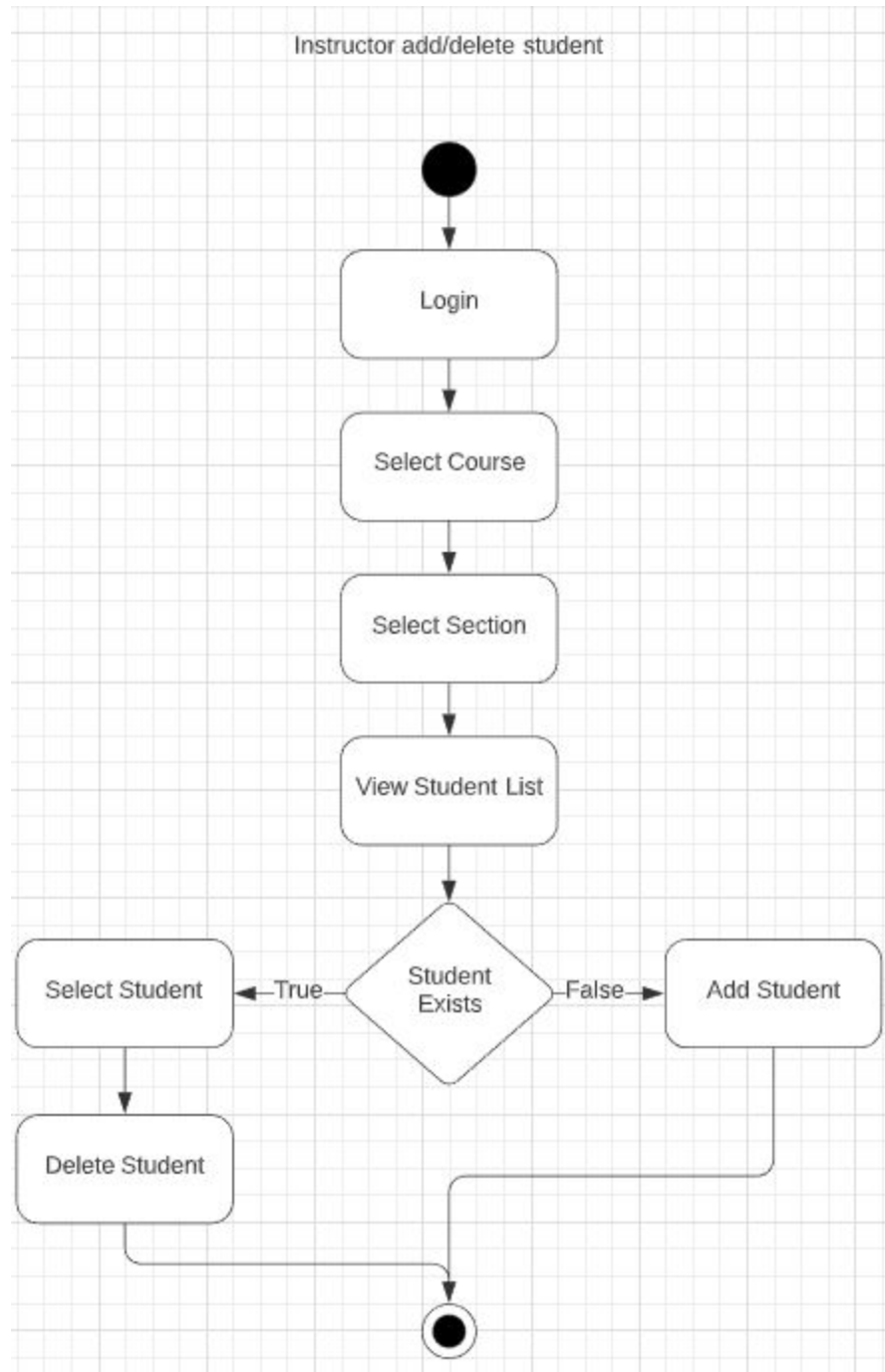
Worked by: Isaac, Daniel, Collin

Class Diagram

Create/Edit/Remove
Courses & Sections

Instructor

- instructorID: string
- instructorPassword: string
- Name: String

+ createSection(): void
+ createCourse(): void
+ editSection(sectionID: string):
sectionData
+ editCourse(courseID: string):
courseData
+ deleteSection(sectionID: string): void
+ deleteCourse(courseID: string): void

1

Course

+ courseID: string
+ courseName: string
+ courseNumber: string
+ startDate: date
+ endDate: date

1..*

+ getCourseNumber(courseID):int
+ setCourseID(): void
+ getCourseName(courseID): string
+ setCourseName(string): void
+ getStartDate(courseID): datetime
+ setStartDate(datetime): void
+ getEndDate(courseID): datetime
+ setEndDate(datetime): void

1

1..*

Section

+ sectionID: string
+Classroom: string

+ getSectionID(courseID): int
+ setSectionID(sectionID): void
+ getClassroom(courseID, sectionID): string
+ setClassroom(classroom, courseID, sectionID): void

1..*

1

Student

- studentID: integer
- studentName: string
- studentEmail: string

+ joinCourse(courseID: string): void
+ leaveCourse(courseID: string): void
+ joinSection(sectionID: string): void
+ leaveSection(sectionID: string): void

1

1

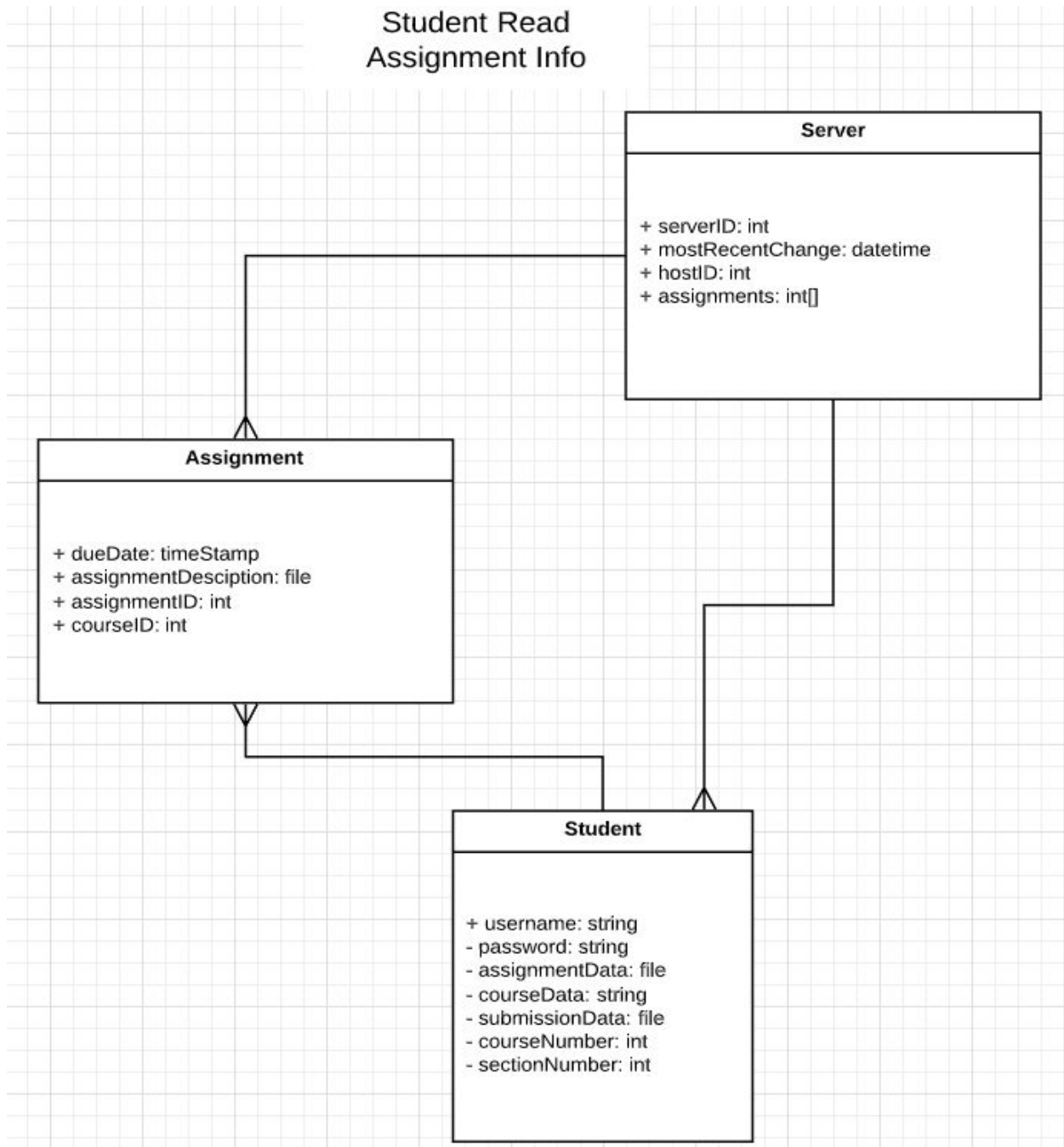**Description: This class diagram reflects the classes involved in creating/editing/removing courses and sections as an Instructor.**
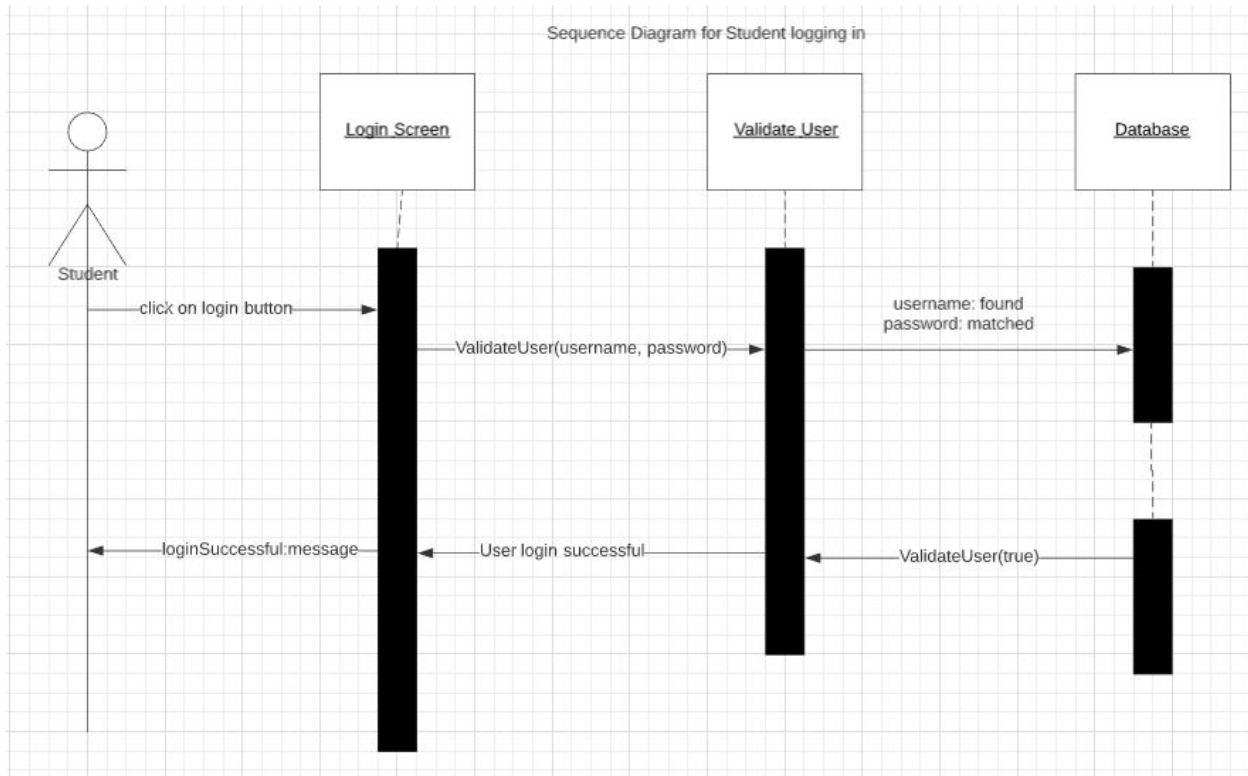
Worked by: Collin, Isaac, Daniel

Entity Relationship Diagram

Student Read
Assignment Info

Server

+ serverID: int
+ mostRecentChange: datetime
+ hostID: int
+ assignments: int[]

Assignment

+ dueDate: timeStamp
+ assignmentDesciption: file
+ assignmentID: int
+ courseID: int

Student

+ username: string
- password: string
- assignmentData: file
- courseData: string
- submissionData: file
- courseNumber: int
- sectionNumber: int

Description: This Entity relationship diagram is in Crow's foot notation. It shows the relationship and the related data for the student submitting an assignment.
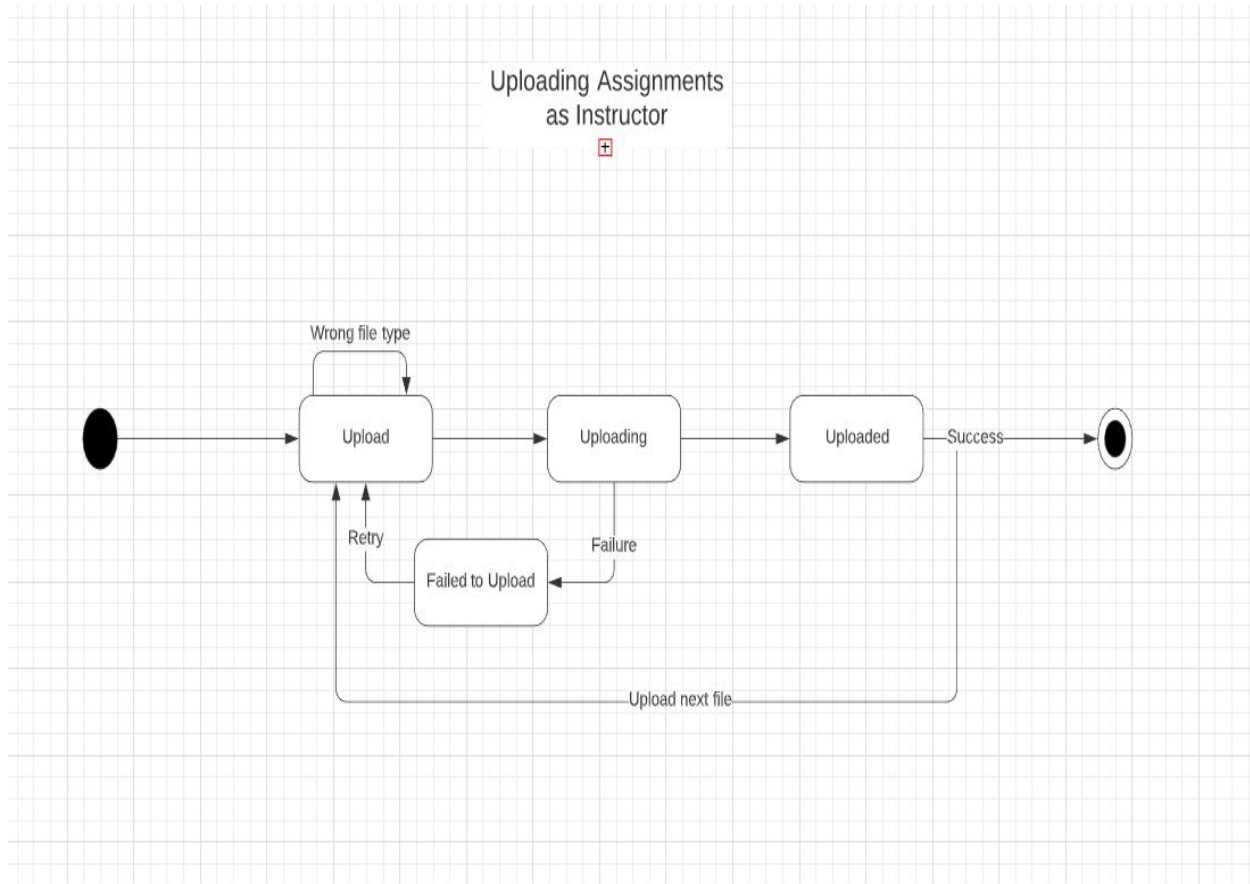
Worked by: Collin, isaac, Daniel

Sequence Diagram

Sequence Diagram for Student logging in

Login Screen   Validate User   Database

Student

—click on login button—→

—ValidateUser(username, password)—→

username: found
password: matched

←—loginSuccessful:message— ←—User login successful— ←—ValidateUser(true)—

Description: The sequence diagram shows the sequence and order of events taking place in relation to the student logging in. From the student clicking the login button, to the database validating the credentials and returning whether the credentials match or not.

Worked by: Collin, Isaac, Daniel

State Machine Diagram

Uploading Assignments as Instructor

Wrong file type

Upload → Uploading → Uploaded → Success

Retry — Failure — Failed to Upload

Upload next file

Description: This diagram is to demonstrate the states the system would go through when uploading an assignment (as an Instructor). There are 4 major states that are Upload, Uploading, Failed to Upload and Uploaded with potential to fail during the Upload -> Uploading stage. After uploading successfully, the user can upload a new file or the system state would reach the end.

Worked by: Daniel, Collin, Isaac