Ain Shams University
Faculty of Engineering
Computer and Systems Engineering
CSE334s: Software Engineering

FACULTY MANAGEMENT SYSTEM

NAME	ID
Mohamed Hossam El-Din El-Sayed	1808581
Mohamed Hesham Alsaied Albendary	1808442
Ahmed Salah Abdelhak	1803820
Ahmed Reda Ibrahim Taha Elawam	1802745
Mohamed Ashraf Farouk Ibrahim	1805188
Mohamed Ahmed Mahmoud Asran	1808040

Contents

SOFTWARE SPECIFICATION DOCUMENT

Introduction

This faculty management system provides services to both students and academic advisors. It enables the students to register for new courses, view their grades and view descriptions for each course. It also enables the advisors to find students using their ID, view their names and GPAs, update course description, accept or reject students’ enrollments in new courses.

Source code can be found on this repo:
https://github.com/ASalah007/Faculty-Management-System

Feasibility Analysis

This software can be used in educational organizations and schools to facilitate the process of course registration for both students and academic advisors and manage students’ grades.

It can be implemented using an object-oriented programming language such as Java, and a relational database management system such as MySQL.

It can be integrated with other software to provide further functionalities, i.e., provide the history of the courses taken by the student and their grades to another piece of software to perform further operations.

System Requirements

1. Functional Requirements

    - The system enables the student to create an account if it is the first time using the software. After validating the entered data, the database is updated with the student's information
    - If the student already has an account, he can sign in by entering his e-mail and password. The system validates the entered data by checking the database and if everything is fine, the student is granted access to his account.
    - After signing in, the student is provided a list of available courses to register for. After registering a course, it will be in pending state waiting for the advisor to approve this registration and the student can cancel their registration at any time. Both the pending and approved courses of the student are stored in the system database.
    - If the academic advisor is granted an account by the administrator, he can sign into the system the same way as mentioned before for the student.
    - Both students and advisors have the option to log out of the system.
    - The academic advisor can search for the student using his ID. The student's name, GPA, and registered Courses are available to the advisor. Then, the advisor can either approve or reject each of these courses.

2. Non-Functional Requirements

- This software is implemented using Java programming language and MySQL database management system.
- It can be deployed on any desktop operating system as it is implemented using Java and MySQL.
- It can be integrated with other software to provide data about the students, their GPA, and the courses to do more operations on these data. The other software can access the system updated database.
- The faculty management system does very basic operations which are essential to every educational organization so it can be used as long as the organization adapts the credit-hour system, although integration with other software may be required in case more functionalities are needed.

Requirements Validation

Requirements \ Stakeholders	Developing Team	Student	Academic Advisor
Sign up	✓	✓	
Log in	✓	✓	✓
Course Registration	✓	✓	
Course Approval	✓		✓
Course Rejection	✓		✓
Viewing Average GPA	✓		✓
Viewing Grades	✓	✓	
Find Student by ID	✓		✓
Edit course description	✓		✓
Log out	✓	✓	✓

Table 1 – Source Traceability Matrix

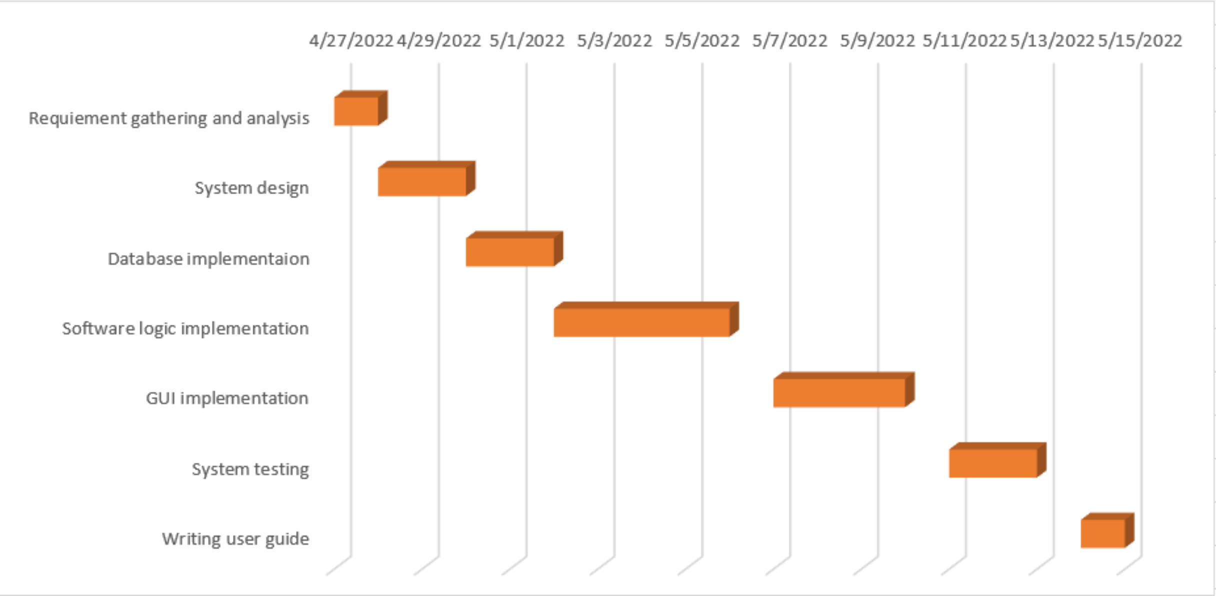
	Sign up	Log in	Course Registration	Course Approval	Course Rejection	Viewing Average GPA	Viewing Grades	Find Student by ID	Edit Course Description	Log out
Sign up										
Log in	D									
Course Registration	R	D								
Course Approval		D	D					D		
Course Rejection		D	D					D		
Viewing Average GPA		D						D		
Viewing Grades	R	D						R		
Find Student by ID		D								
Edit Course Description		D						D		
Log Out	R	D								

Table 2 – Requirement Traceability Matrix

Time Plan

Task	Start Date	Days to Complete
Requirement Gathering and Analysis	4/27/2022	1
System Design	4/28/2022	2
Database Implementation	4/30/2022	2
Software Logic Implementation	5/2/2022	4
GUI Implementation	5/7/2022	3
System Testing	5/11/2022	2
Writing User Guide	5/14/2022	1

Table 3 – Time Plan Table


4/27/2022
4/29/2022
5/1/2022
5/3/2022
5/5/2022
5/7/2022
5/9/2022
5/11/2022
5/13/2022
5/15/2022
Requiement gathering and analysis
System design
Database implementaion
Software logic implementation
GUI implementation
System testing
Writing user guide

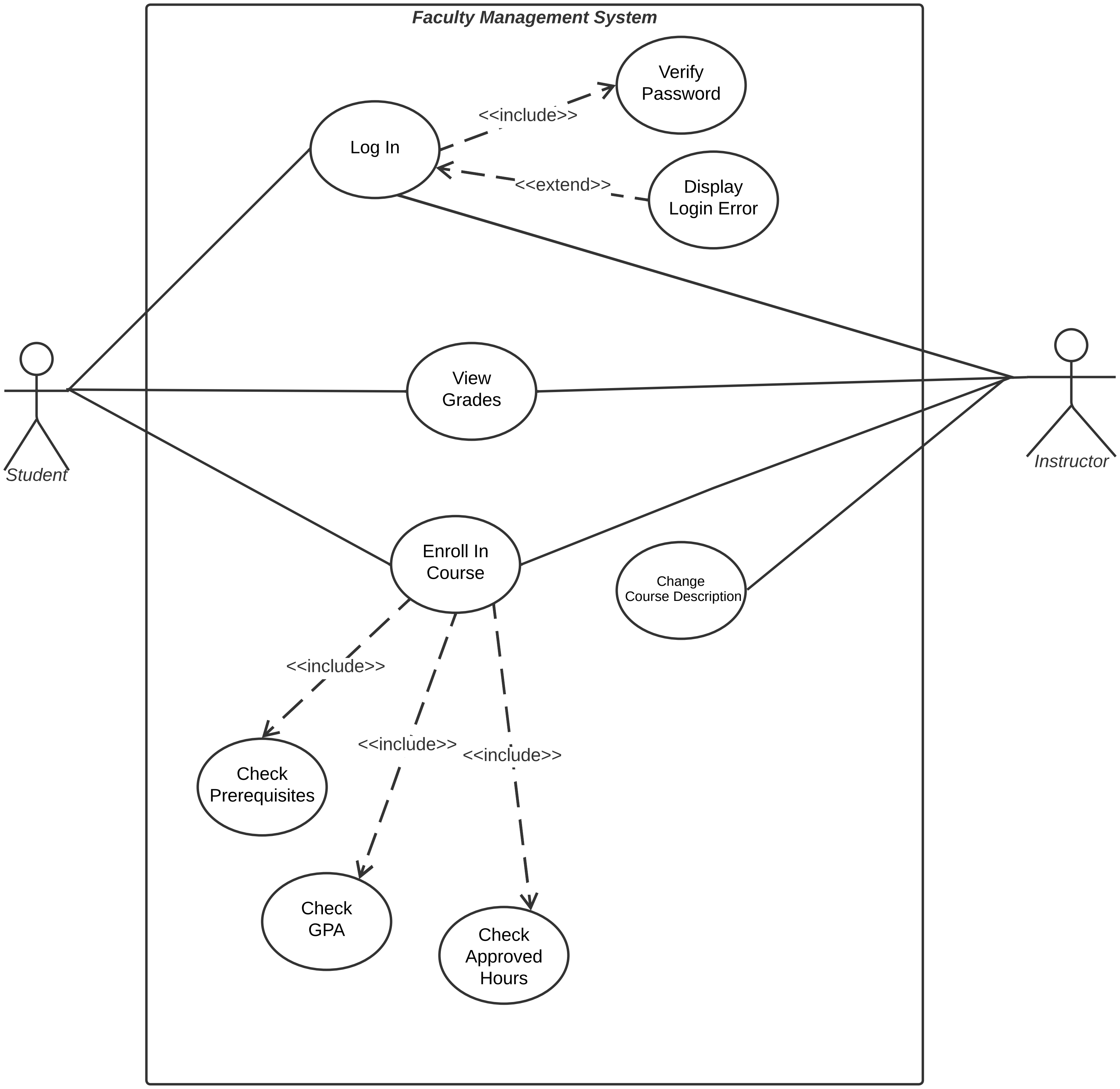
Conclusion

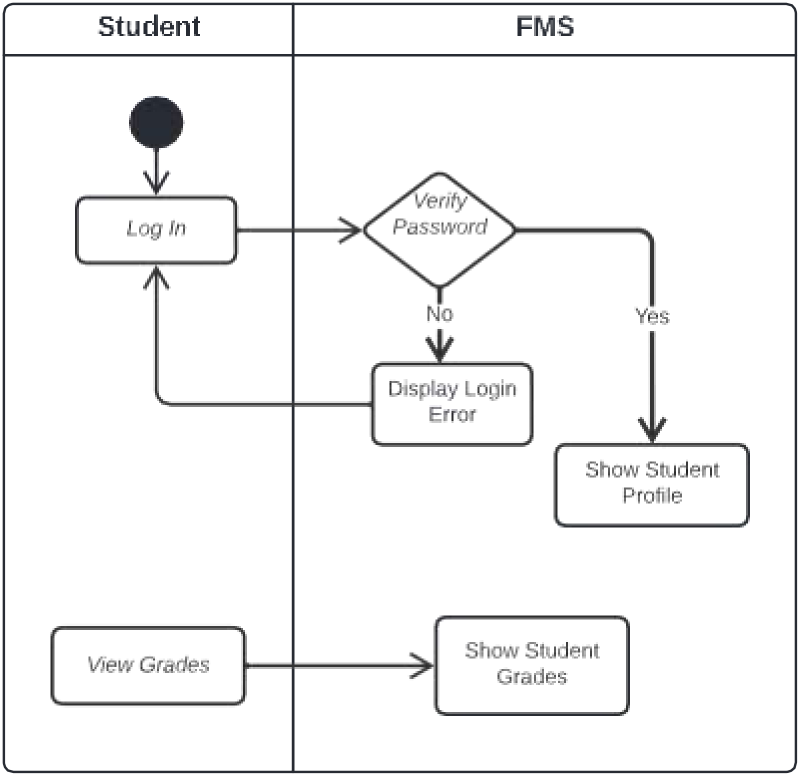
The system provides basic services to educational organizations to both students and academic advisors, it can be integrated with other software in order to provide further functionalities, and it can be deployed on any desktop operating system.

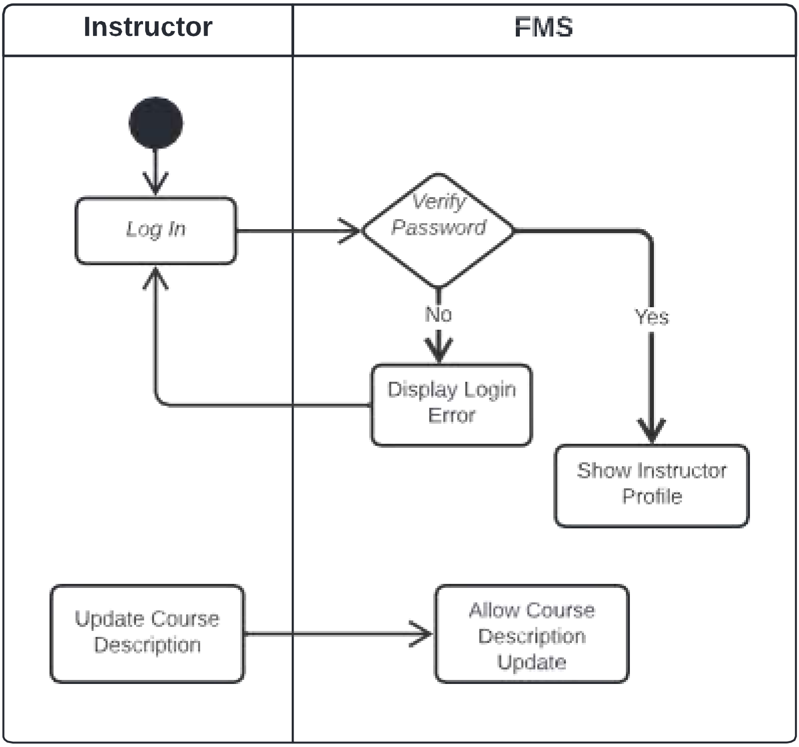
The software is implemented using Java and MySQL over a time span of 10 days.

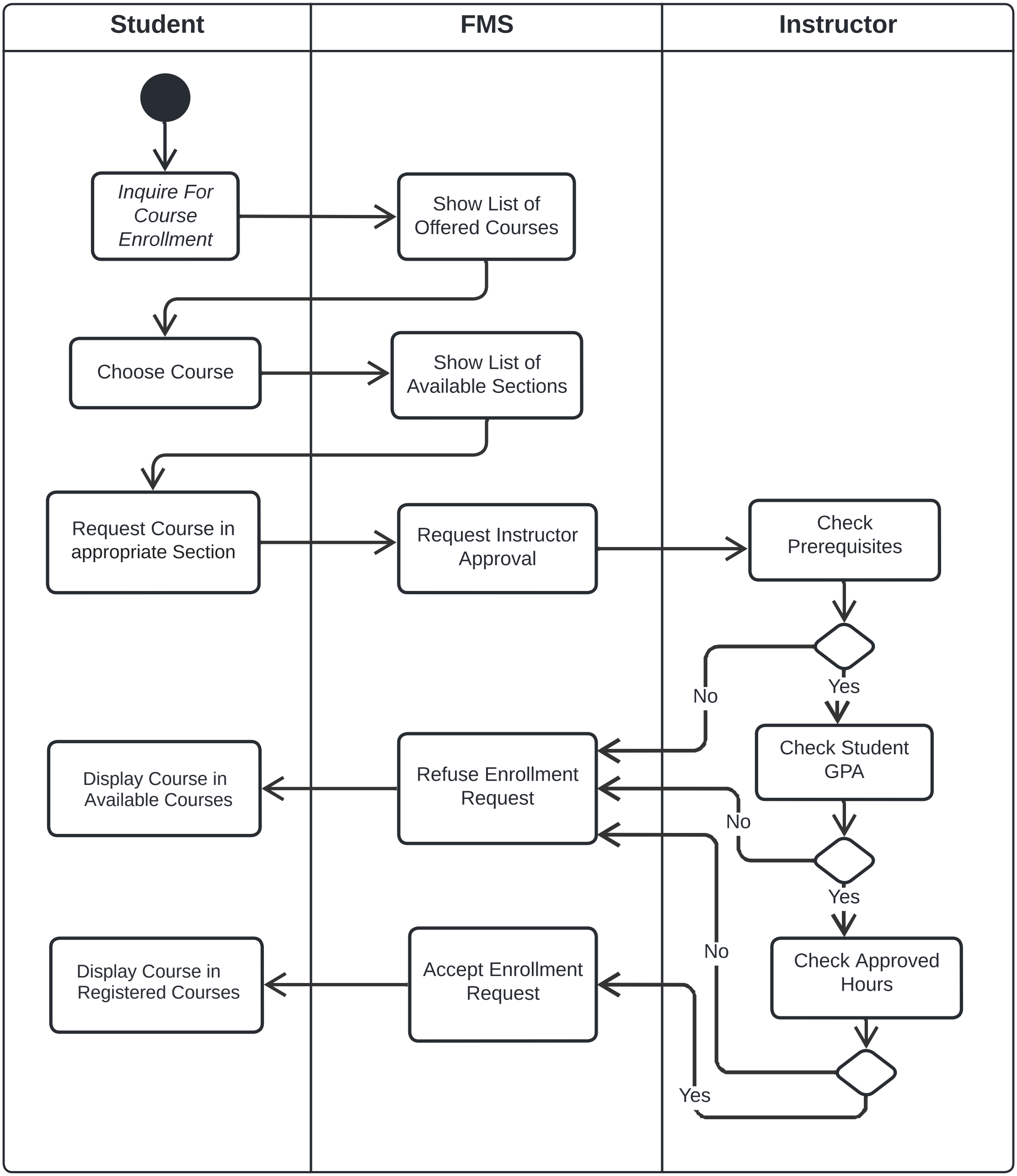
ANALYSIS AND DESIGN DOCUMENT

Use-Case


Faculty Management System
Verify Password
<<include>>
Log In
<<extend>>
Display Login Error
View Grades
Student
Instructor
Enroll In Course
Change Course Description
<<include>>
<<include>>
<<include>>
Check Prerequisites
Check GPA
Check Approved Hours

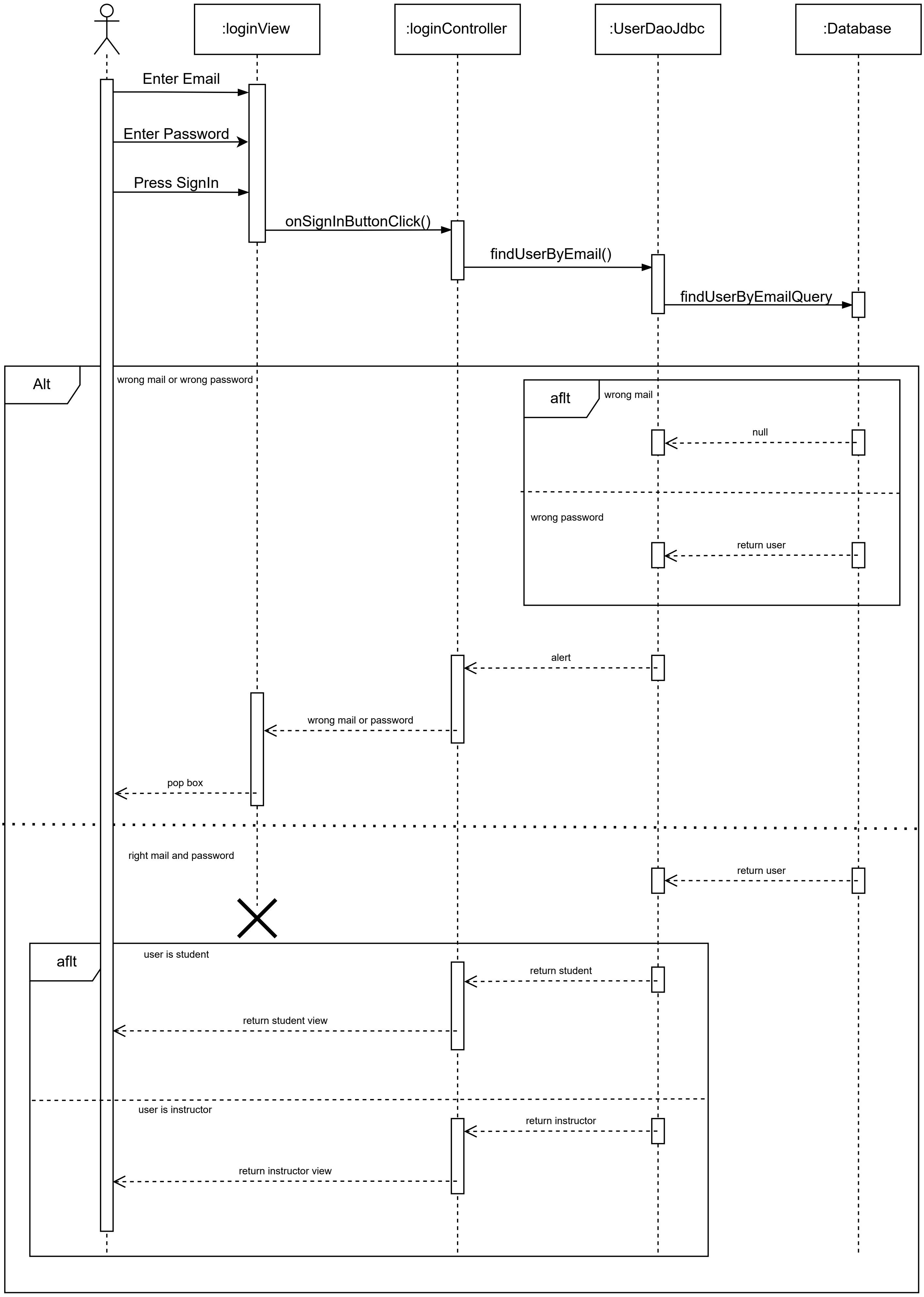

Student
FMS
Log In
Verify Password
No
Yes
Display Login Error
Show Student Profile
View Grades
Show Student Grades


Instructor
FMS
Log In
Verify Password
No
Yes
Display Login Error
Show Instructor Profile
Update Course Description
Allow Course Description Update

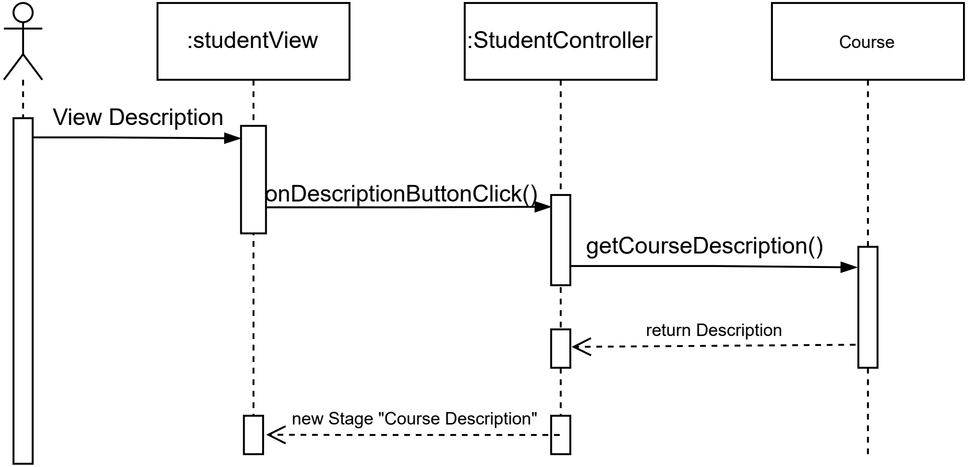

Student
FMS
Instructor
Inquire For Course Enrollment
Show List of Offered Courses
Choose Course
Show List of Available Sections
Request Course in appropriate Section
Request Instructor Approval
Check Prerequisites
No
Yes
Check Student GPA
Display Course in Available Courses
Refuse Enrollment Request
No
Yes
No
Check Approved Hours
Display Course in Registered Courses
Accept Enrollment Request
Yes

Sequence Diagram

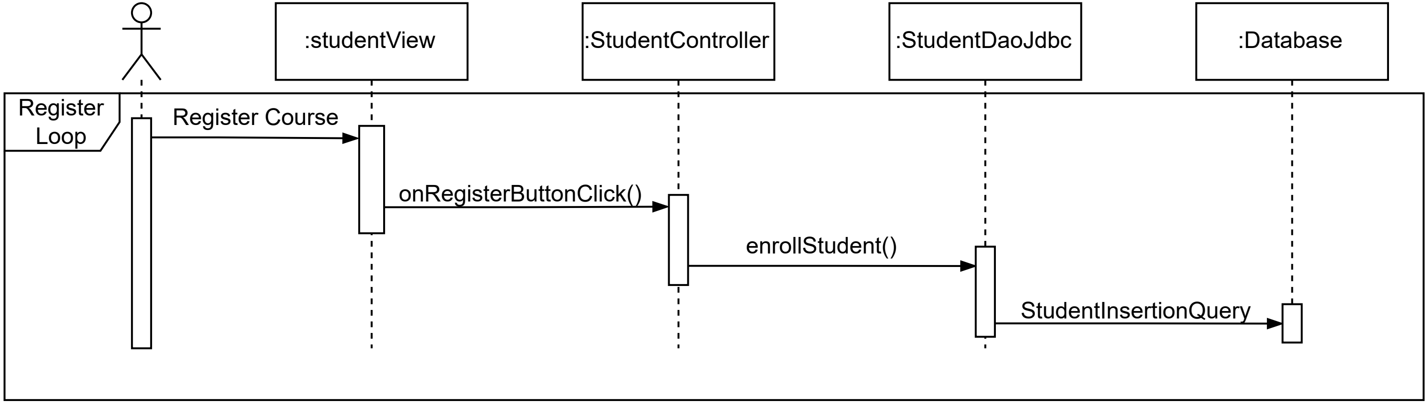
1. User Login



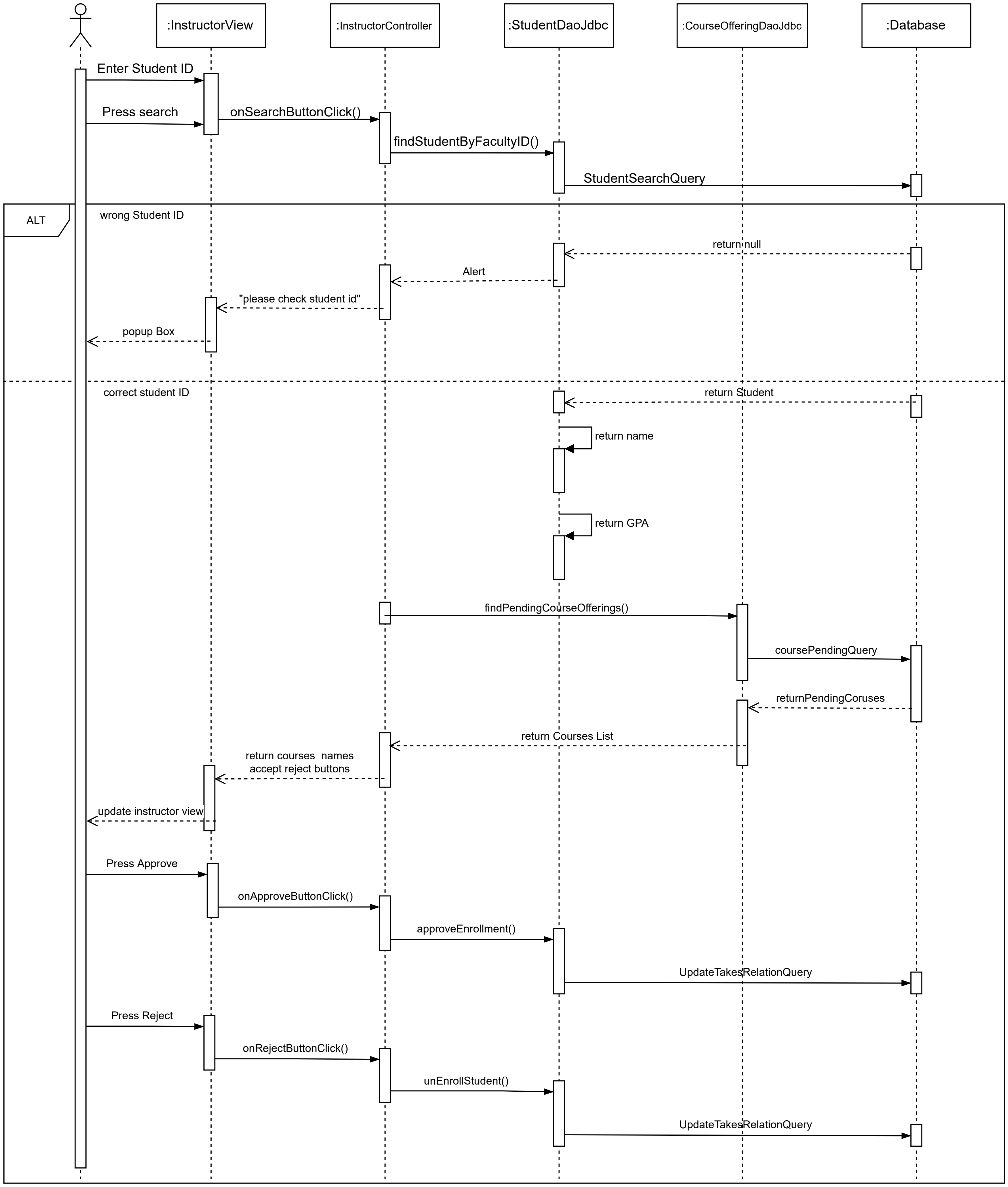
2. Student View Description



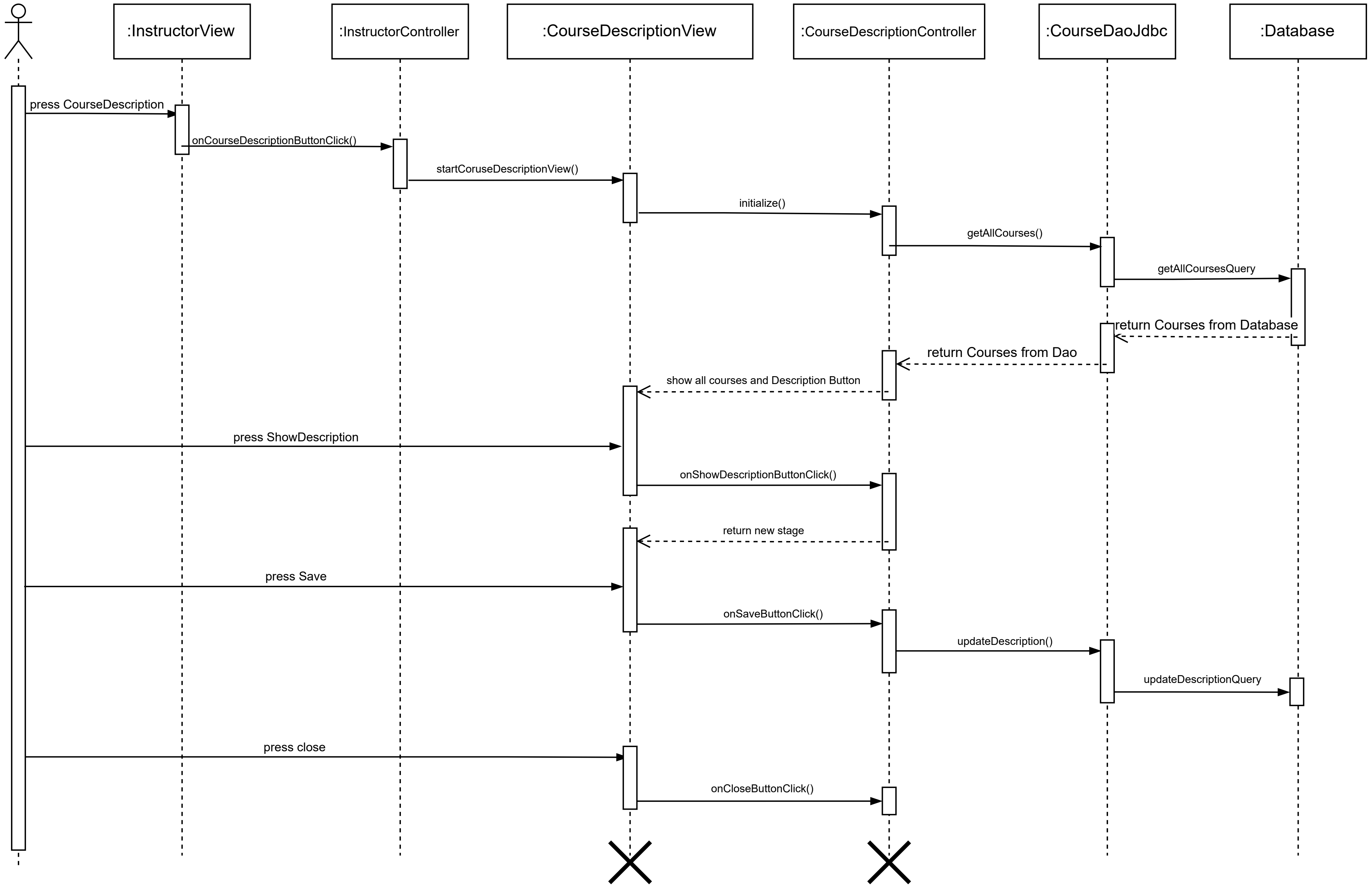
3. Student Register



4. Instructor

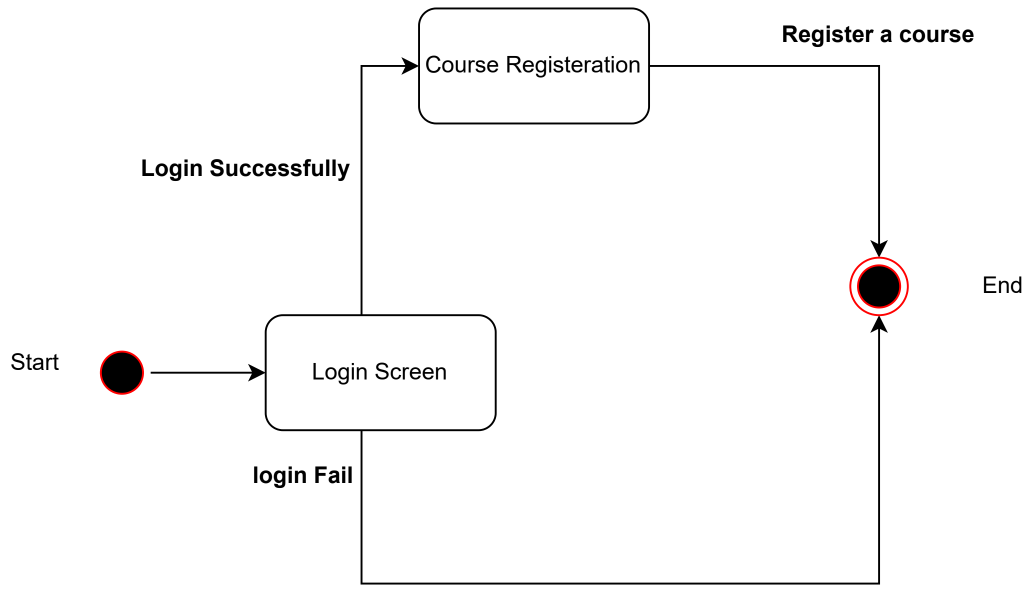


5. Update Description

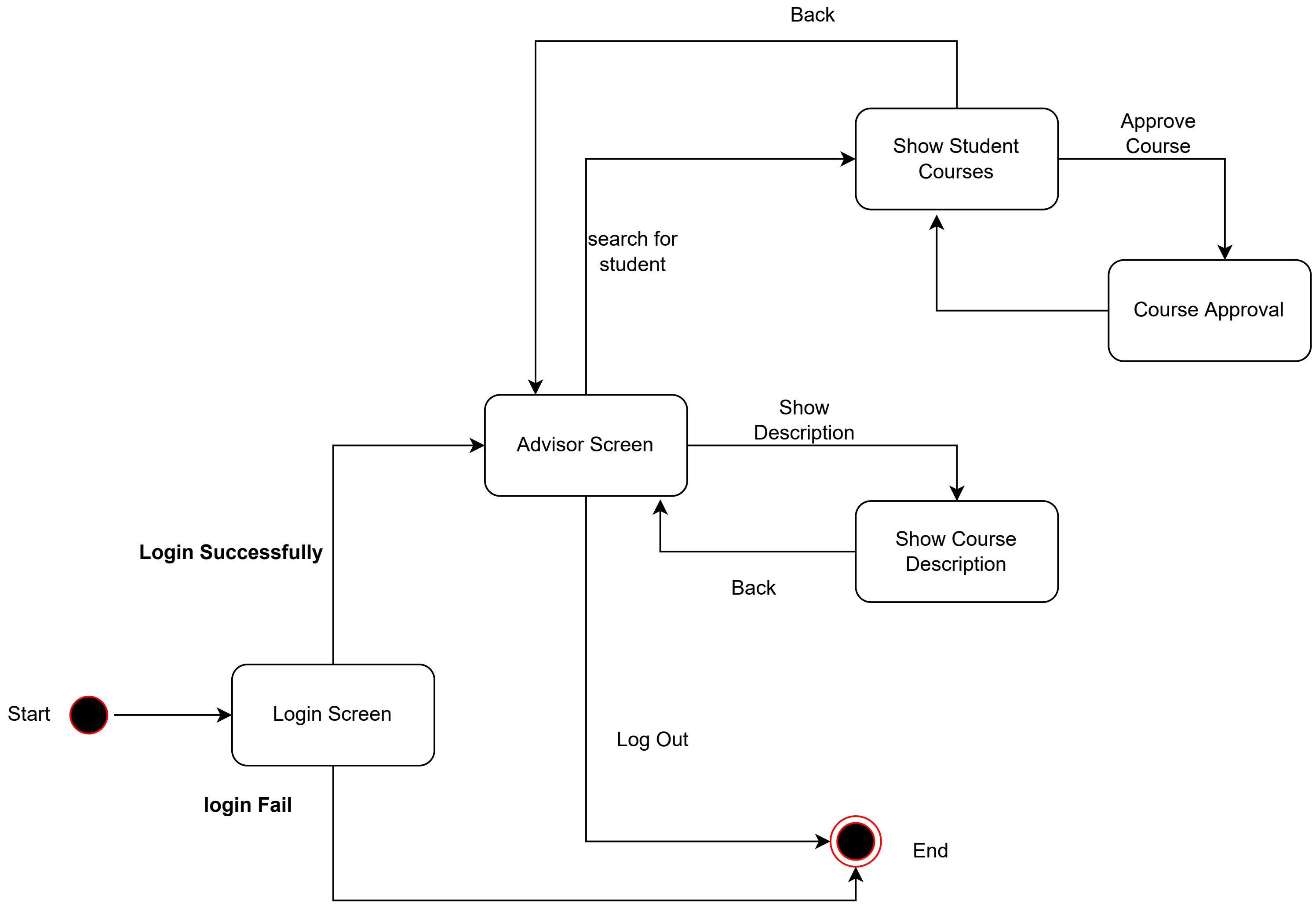


State Diagram

Student State Diagram



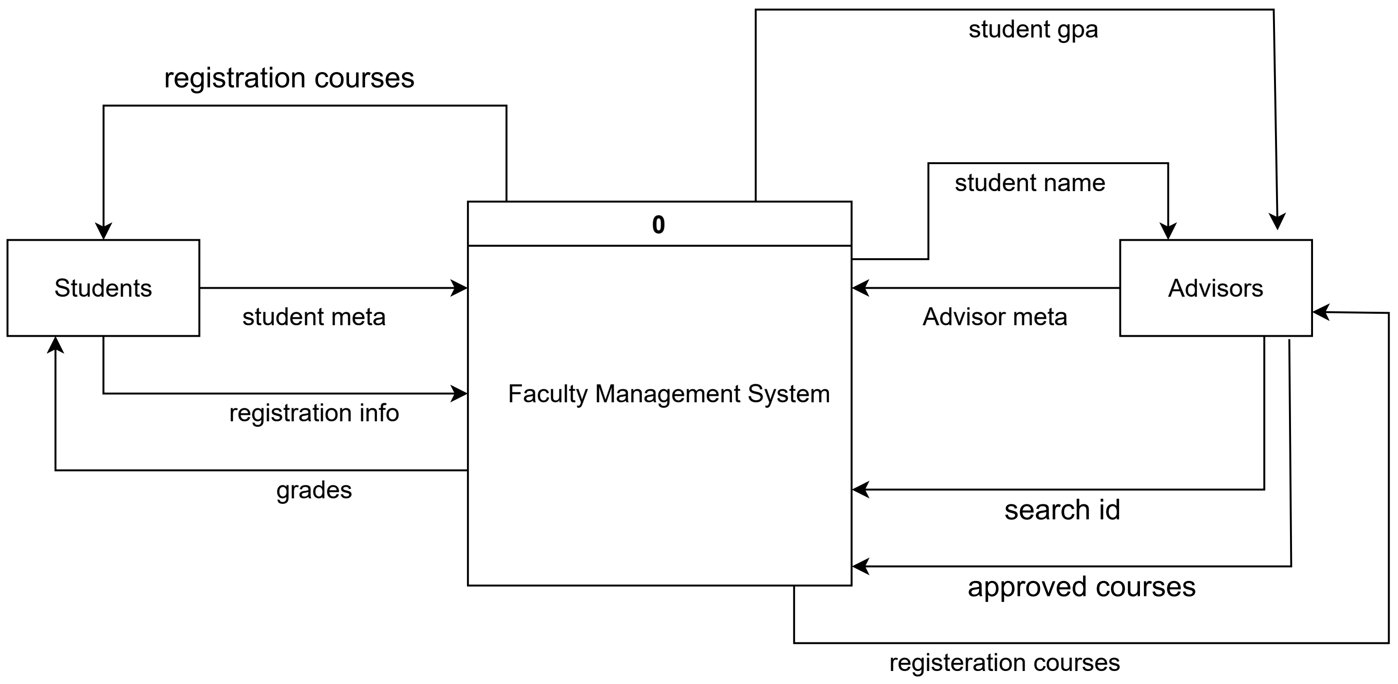
Instructor State Diagram

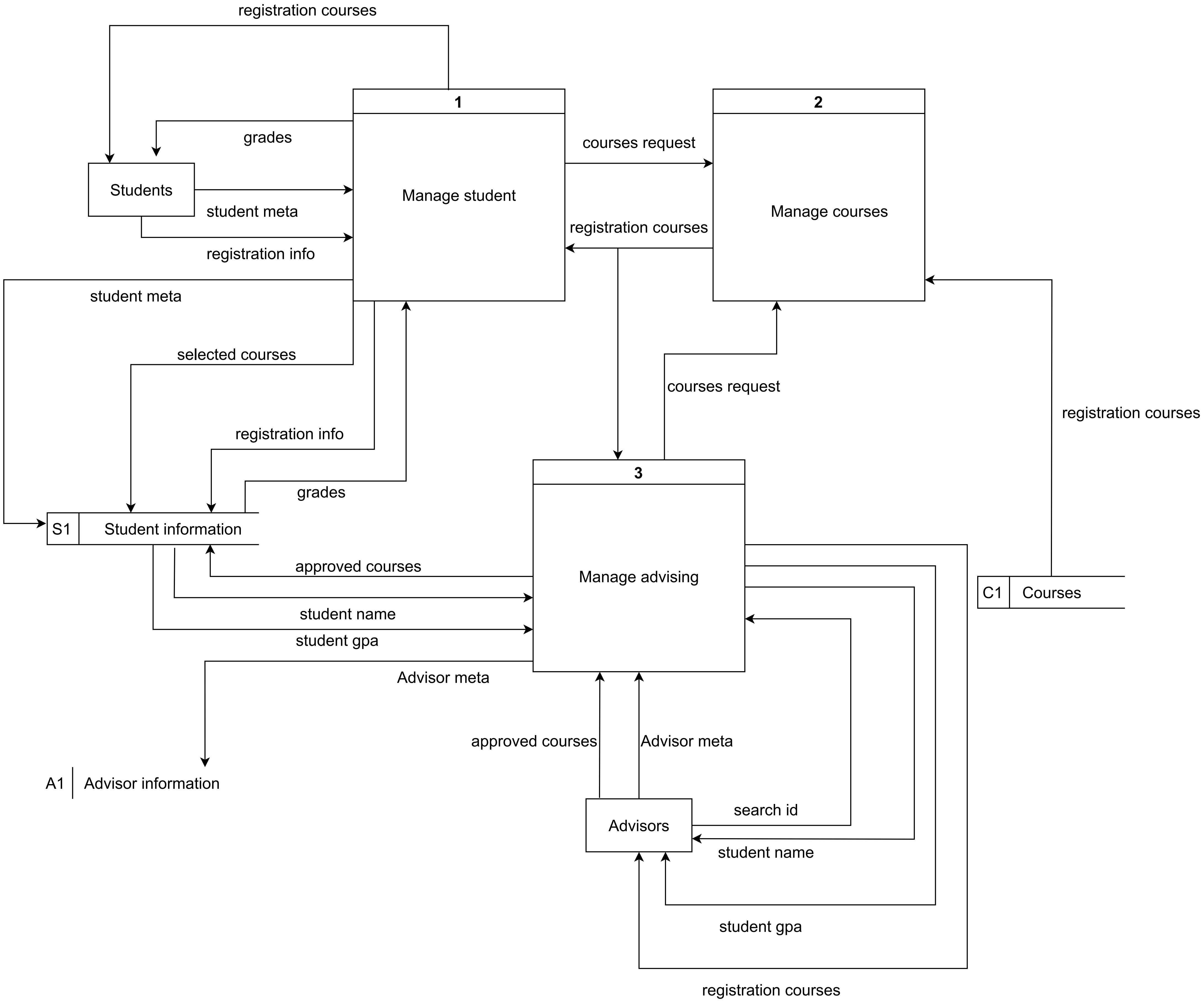


DFD

The system consists of three main processes

1. “Manage Student” process handles the student’s information. It takes the student’s metadata, his basic information such as his name, id, email, date of birth, etc., and stores it in the student’s database. It provides the student with a list of available registration courses and stores the selected ones for the student. It also provides the student with his grades.
2. “Manage Advising” process handles the academic advisor functionalities. It stores the advisor’s metadata, his e-mail, and password, in the database. It provides the instructor with the name, the GPA, and the list of registration courses for a student provided the student’s id. It stores the approved courses by the instructor for a certain student.
3. “Manage Courses” process accepts a request from the other processes to provide them with the required list of available registration courses.

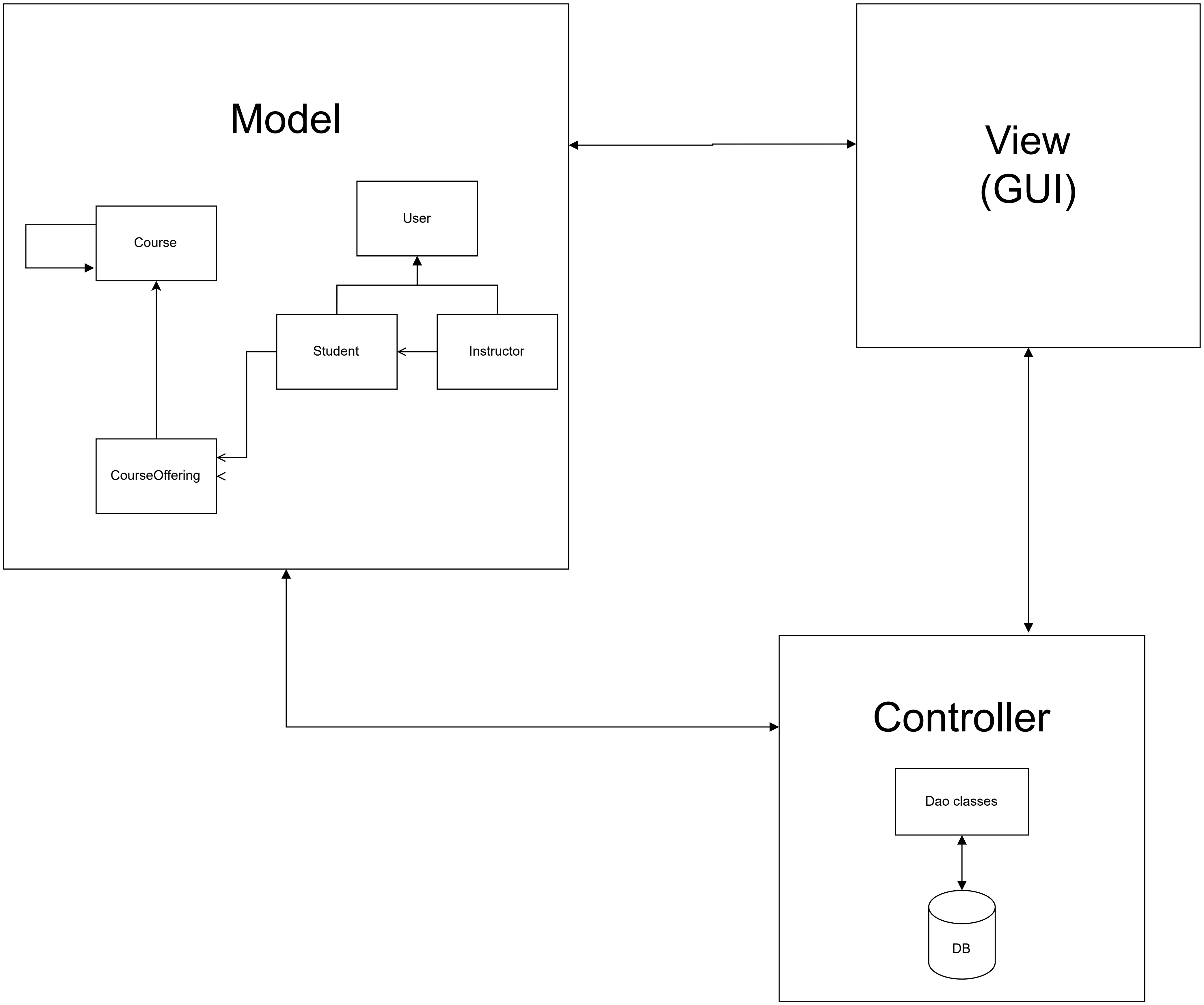




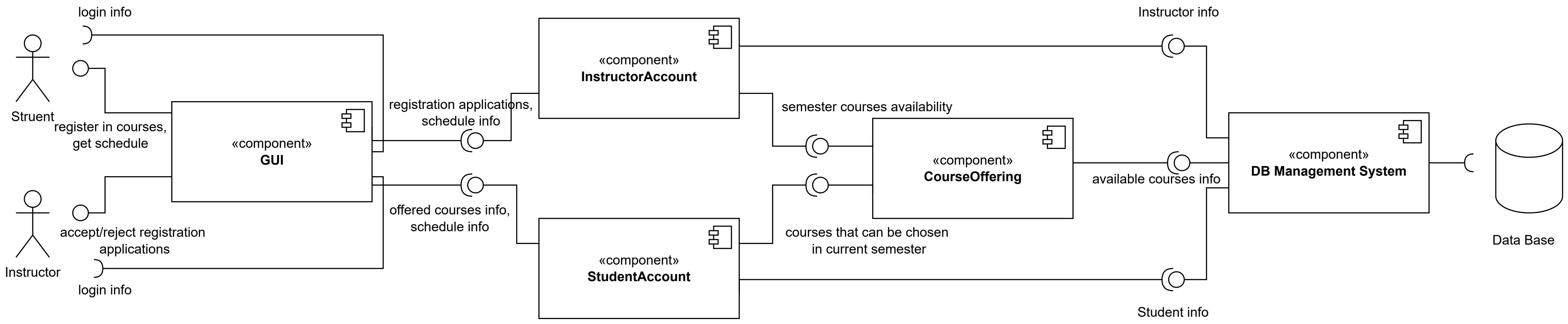
System Architecture

System is designed using the MVC architecture.

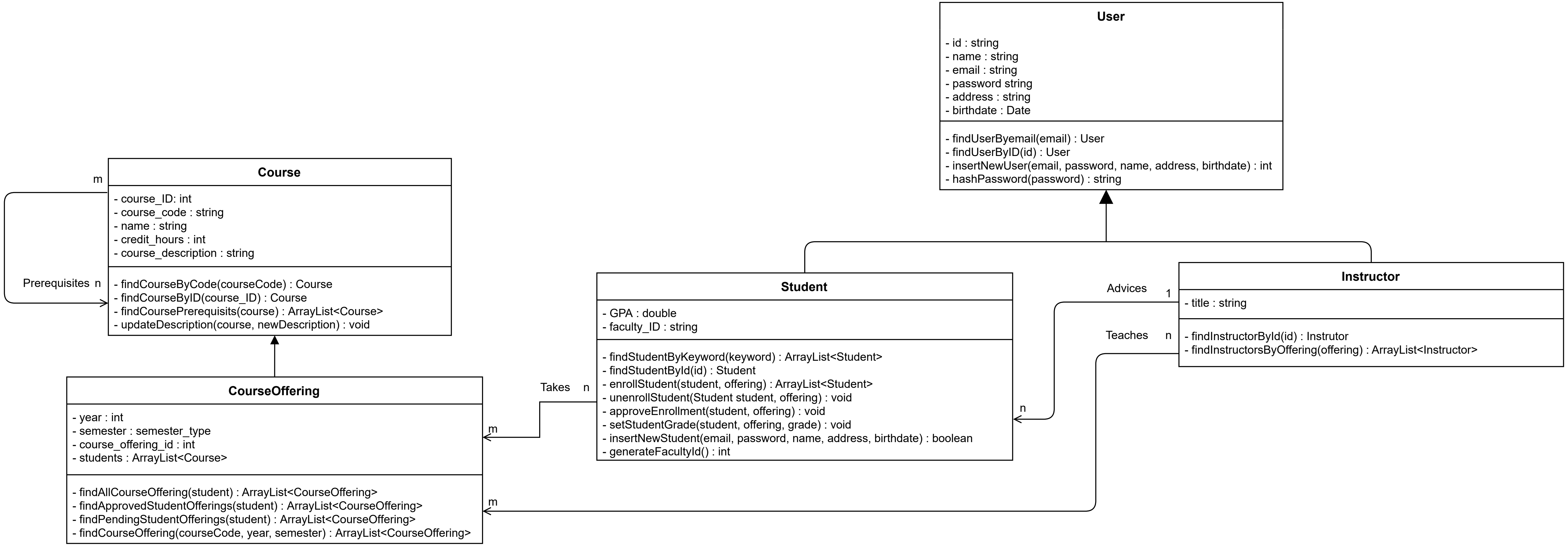
- Model contains the class diagram of the software. For details of the class diagrams, please refer to Class Diagram section.
- View contains the implementation of the GUI using JavaFX
- Controller contains data access objects "DAO" which handle communication with the database using JDBC API



Component Diagram



Class Diagram


User
- id : string
- name : string
- email : string
- password string
- address : string
- birthdate : Date
- findUserByemail(email) : User
- findUserByID(id) : User
- insertNewUser(email, password, name, address, birthdate) : int
- hashPassword(password) : string
Course
- course_ID: int
- course_code : string
- name : string
- credit_hours : int
- course_description : string
- findCourseByCode(courseCode) : Course
- findCourseByID(course_ID) : Course
- findCoursePrerequisits(course) : ArrayList<Course>
- updateDescription(course, newDescription) : void
m
Prerequisites
n
Student
- GPA : double
- faculty_ID : string
- findStudentByKeyword(keyword) : ArrayList<Student>
- findStudentById(id) : Student
- enrollStudent(student, offering) : ArrayList<Student>
- unenrollStudent(Student student, offering) : void
- approveEnrollment(student, offering) : void
- setStudentGrade(student, offering, grade) : void
- insertNewStudent(email, password, name, address, birthdate) : boolean
- generateFacultyId() : int
Instructor
- title : string
- findInstructorById(id) : Instrutor
- findInstructorsByOffering(offering) : ArrayList<Instructor>
Advices
1
n
Teaches
n
m
Takes
n
m
CourseOffering
- year : int
- semester : semester_type
- course_offering_id : int
- students : ArrayList<Course>
- findAllCourseOffering(student) : ArrayList<CourseOffering>
- findApprovedStudentOfferings(student) : ArrayList<CourseOffering>
- findPendingStudentOfferings(student) : ArrayList<CourseOffering>
- findCourseOffering(courseCode, year, semester) : ArrayList<CourseOffering>

END-USER GUIDE

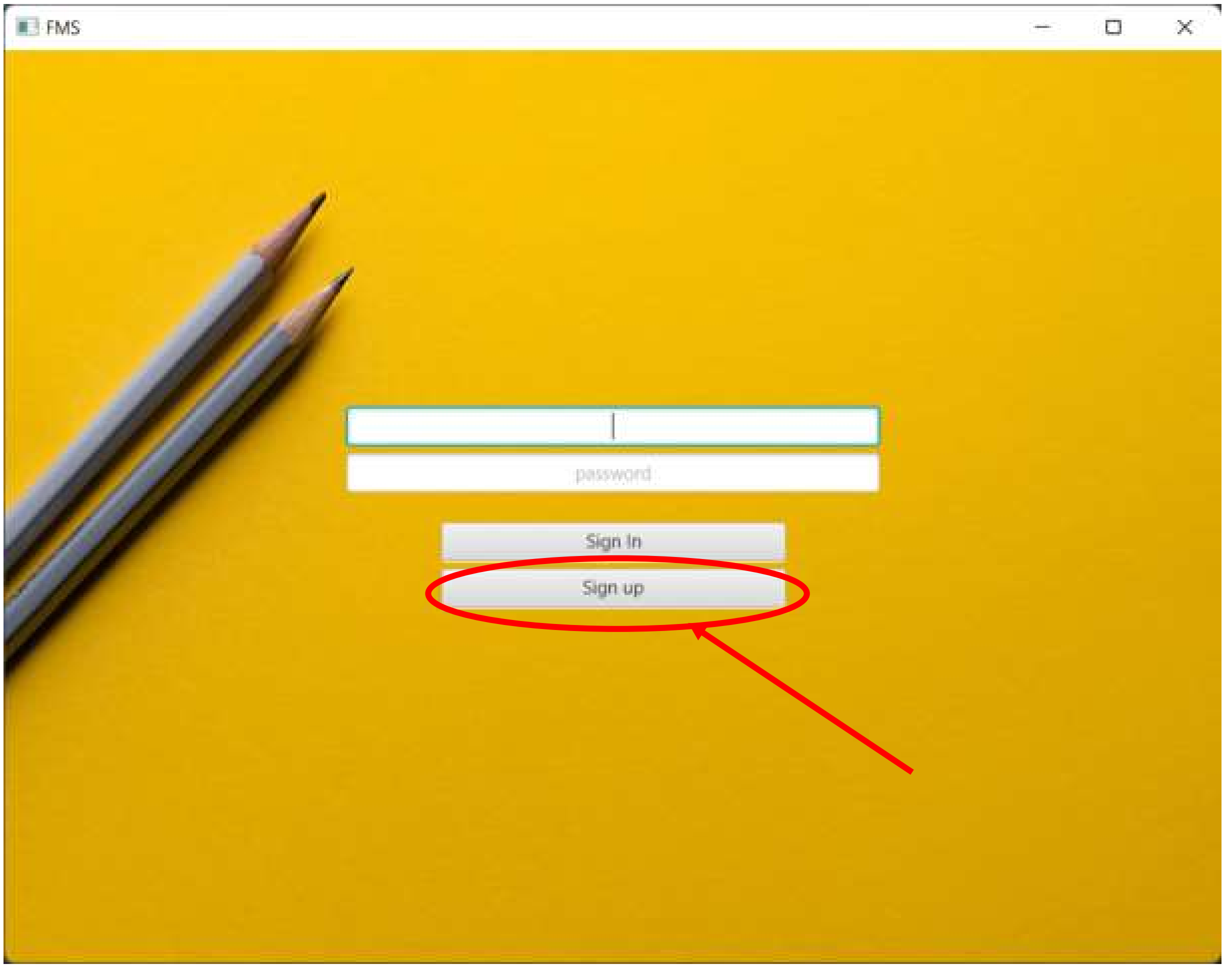
Introduction

Faculty management system (FMS) is a desktop application that helps both students and advisors, it allows students to enroll in new courses, view their grades and see course descriptions. It also allows advisors to search for a student, accept or reject student enrollment, see their total GPA and update course description. In the next pages, each operation will be described in details.

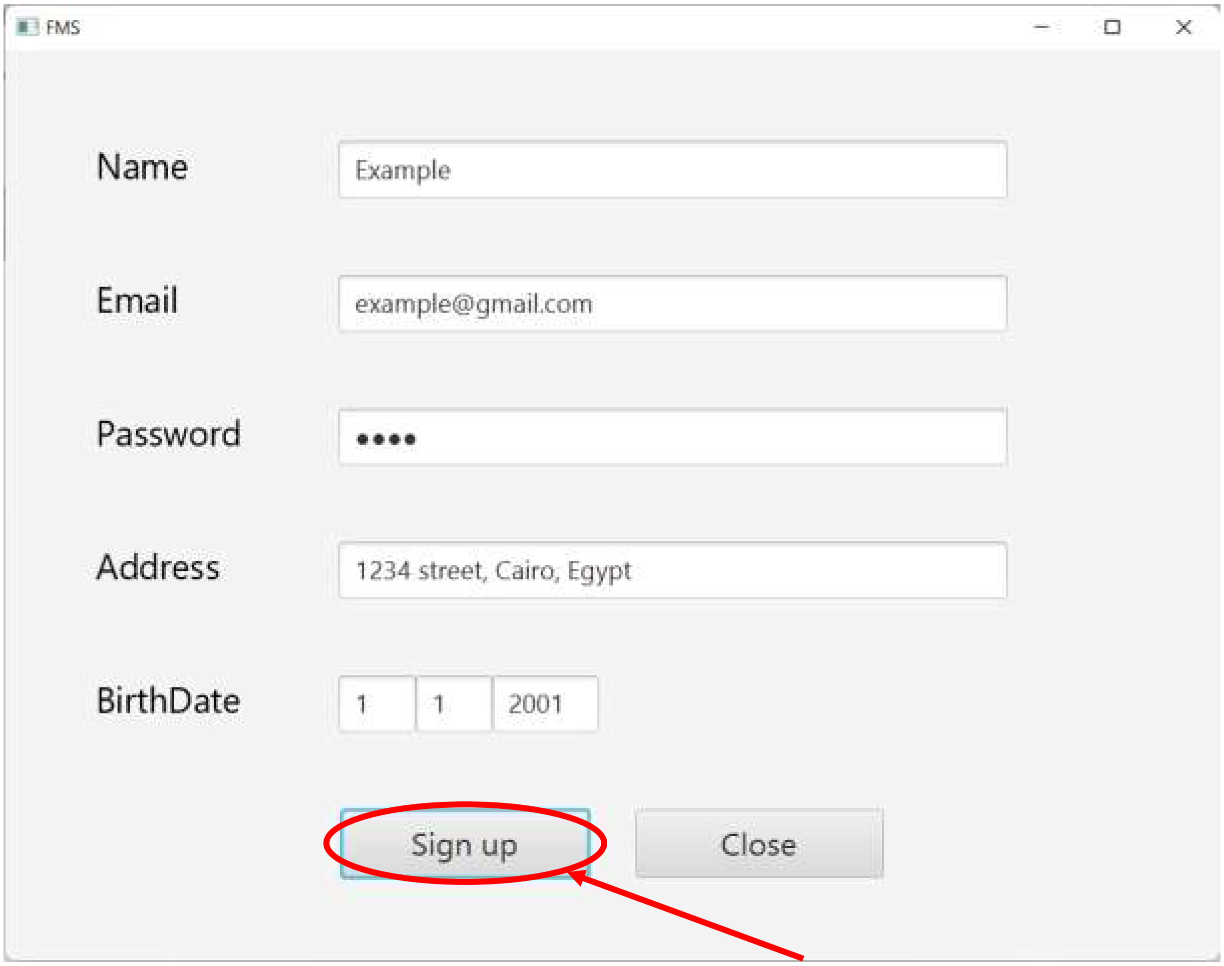
Operations for students

1. Signing up

If you have just started our software and don't have an account, no worries! You can create a new one by pressing "Sign up" button as shown below



A window for creating your account appears in which you enter your data as shown below, after you have entered your data, press Sign up and the software will validate entered data and if everything is correct, congratulations! Your account has been created successfully and you can sign in using that account


FMS

Name: Example

Email: example@gmail.com

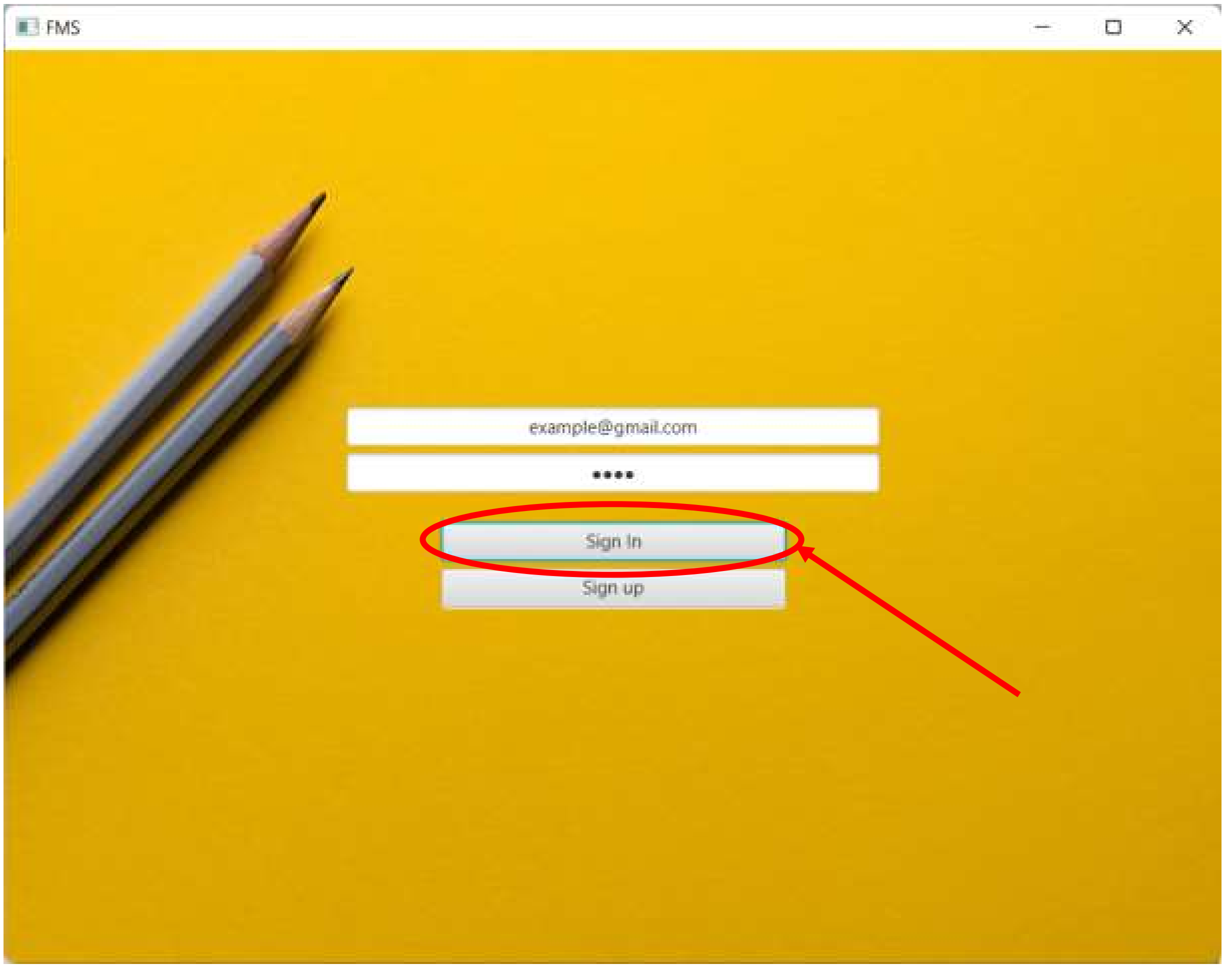
Password: ●●●●

Address: 1234 street, Cairo, Egypt

BirthDate: 1 1 2001

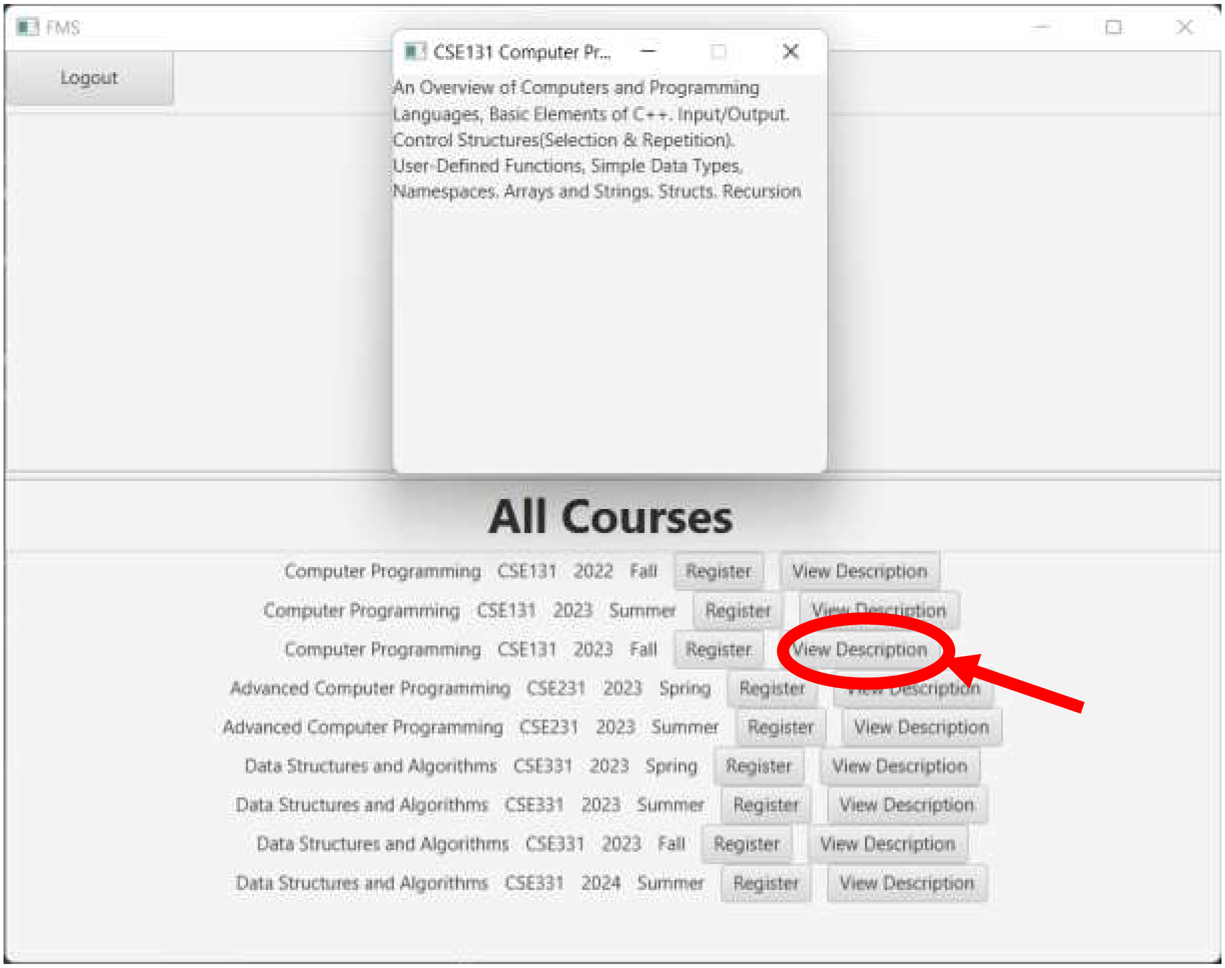
2. Signing in

After you have created an account, enter your email and password then press sign in



3. View description

After you have signed in, you can click “View Description” button next to any course to view its description



An Overview of Computers and Programming Languages, Basic Elements of C++. Input/Output. Control Structures(Selection & Repetition). User-Defined Functions, Simple Data Types, Namespaces. Arrays and Strings. Structs. Recursion

All Courses

Computer Programming CSE131 2022 Fall

Computer Programming CSE131 2023 Summer

Computer Programming CSE131 2023 Fall

Advanced Computer Programming CSE231 2023 Spring

Advanced Computer Programming CSE231 2023 Summer

Data Structures and Algorithms CSE331 2023 Spring

Data Structures and Algorithms CSE331 2023 Summer

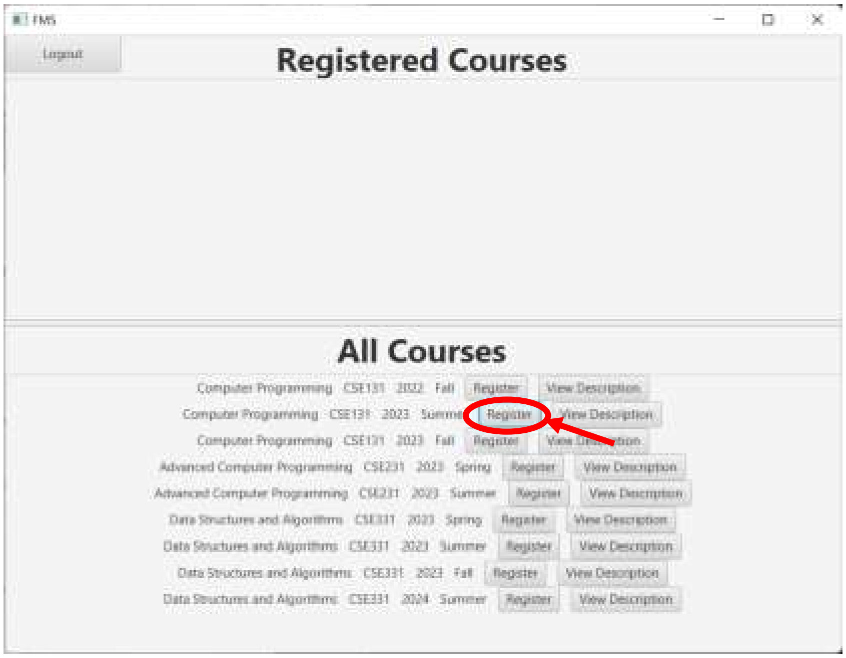
Data Structures and Algorithms CSE331 2023 Fall

Data Structures and Algorithms CSE331 2024 Summer

4. Registering courses

while signed in, you can register for courses you want then wait for your advisor feedback.

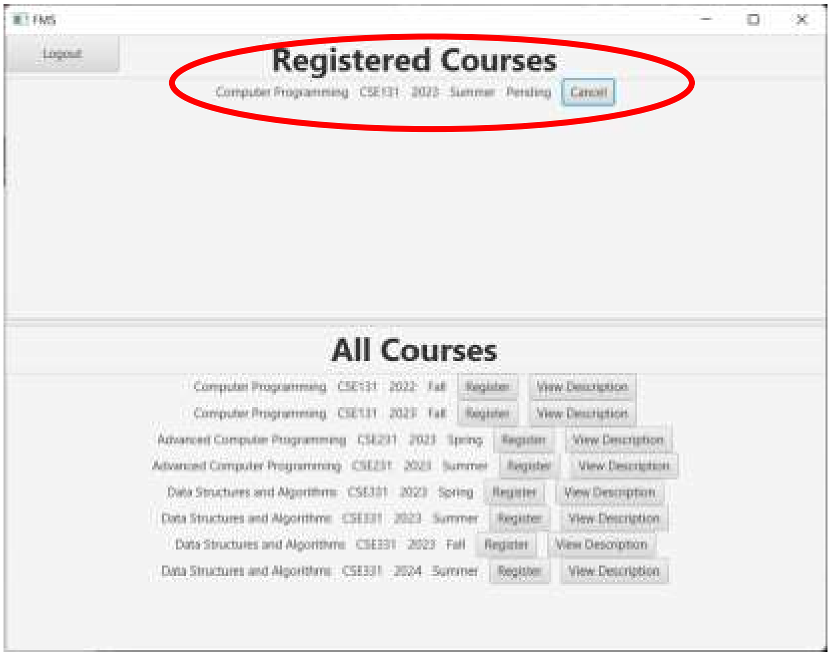
You can cancel registration at any time. You can also register for any future semester



Registered Courses

All Courses

Computer Programming CSE131 2022 Fall Register View Description
Computer Programming CSE131 2023 Summe Register View Description
Computer Programming CSE131 2023 Fall Register View Description
Advanced Computer Programming CSE231 2023 Spring Register View Description
Advanced Computer Programming CSE231 2023 Summer Register View Description
Data Structures and Algorithms CSE331 2023 Spring Register View Description
Data Structures and Algorithms CSE331 2023 Summer Register View Description
Data Structures and Algorithms CSE331 2023 Fall Register View Description
Data Structures and Algorithms CSE331 2024 Summer Register View Description



Registered Courses

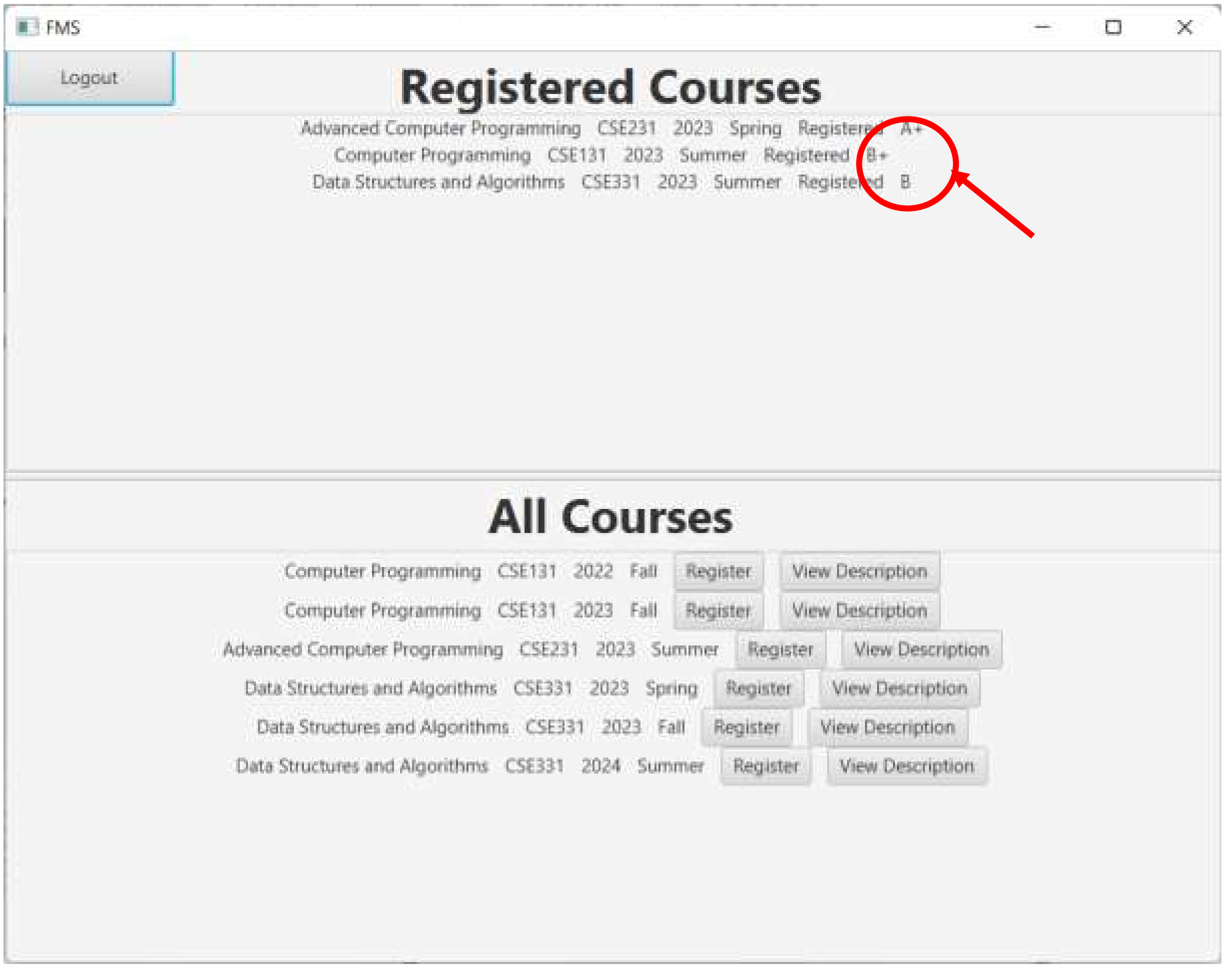
Computer Programming CSE131 2023 Summer Pending Cancel

All Courses

Computer Programming CSE131 2022 Fall Register View Description
Computer Programming CSE131 2023 Fall Register View Description
Advanced Computer Programming CSE231 2023 Spring Register View Description
Advanced Computer Programming CSE231 2023 Summer Register View Description
Data Structures and Algorithms CSE331 2023 Spring Register View Description
Data Structures and Algorithms CSE331 2023 Summer Register View Description
Data Structures and Algorithms CSE331 2023 Fall Register View Description
Data Structures and Algorithms CSE331 2024 Summer Register View Description

5. View grades

After your advisor has accepted your registration and you have taken the final exam, your grades will be shown in Registered Courses section



Registered Courses

Advanced Computer Programming CSE231 2023 Spring Registered A+
Computer Programming CSE131 2023 Summer Registered B+
Data Structures and Algorithms CSE331 2023 Summer Registered B

All Courses

Computer Programming CSE131 2022 Fall
Computer Programming CSE131 2023 Fall
Advanced Computer Programming CSE231 2023 Summer
Data Structures and Algorithms CSE331 2023 Spring
Data Structures and Algorithms CSE331 2023 Fall
Data Structures and Algorithms CSE331 2024 Summer

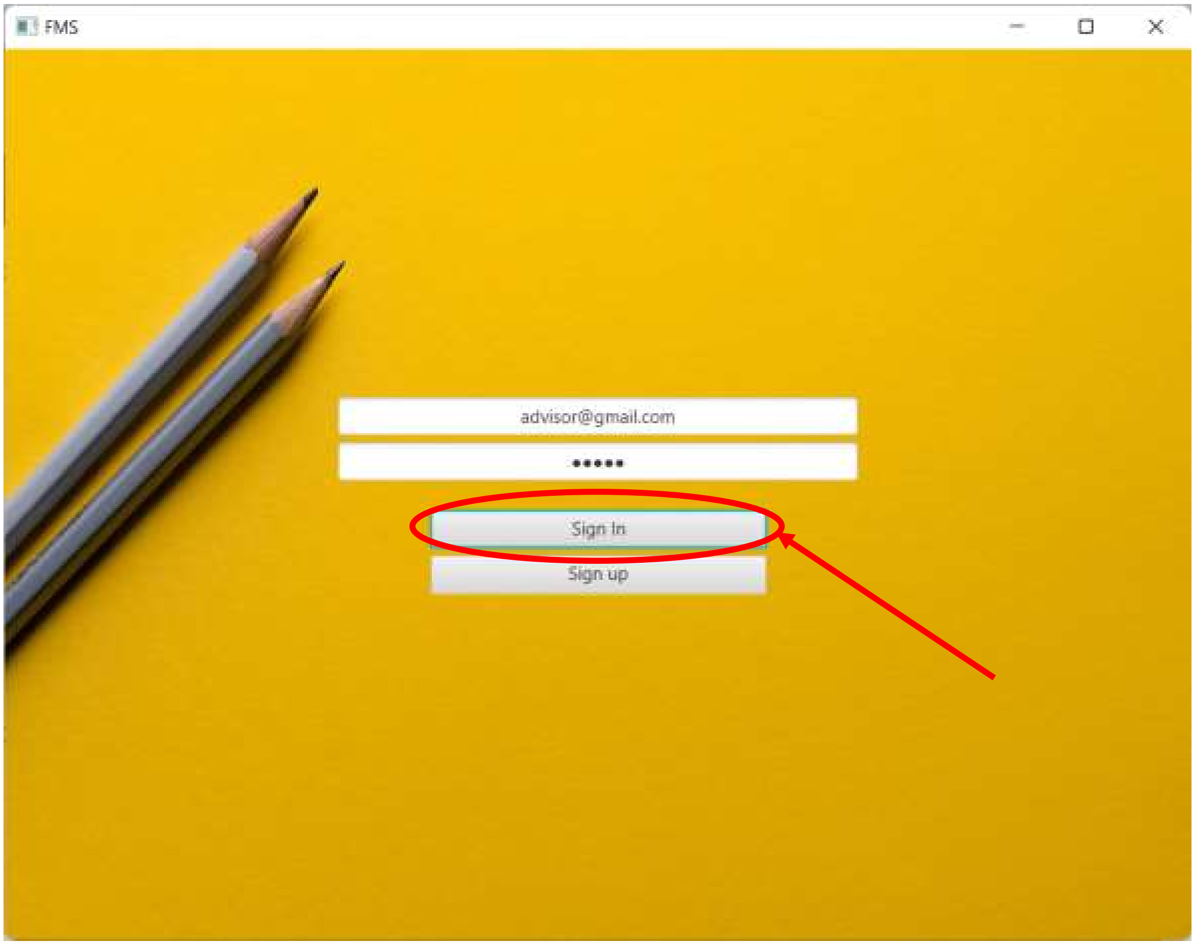
Operation for advisors

1. Signing up

To create a new advisor account, please contact the database administrator to create a new account for you as it's not possible to directly create a new account from the software for security reasons

2. Signing in

After you have obtained a new account, enter your email and password as shown then press "Sign in"

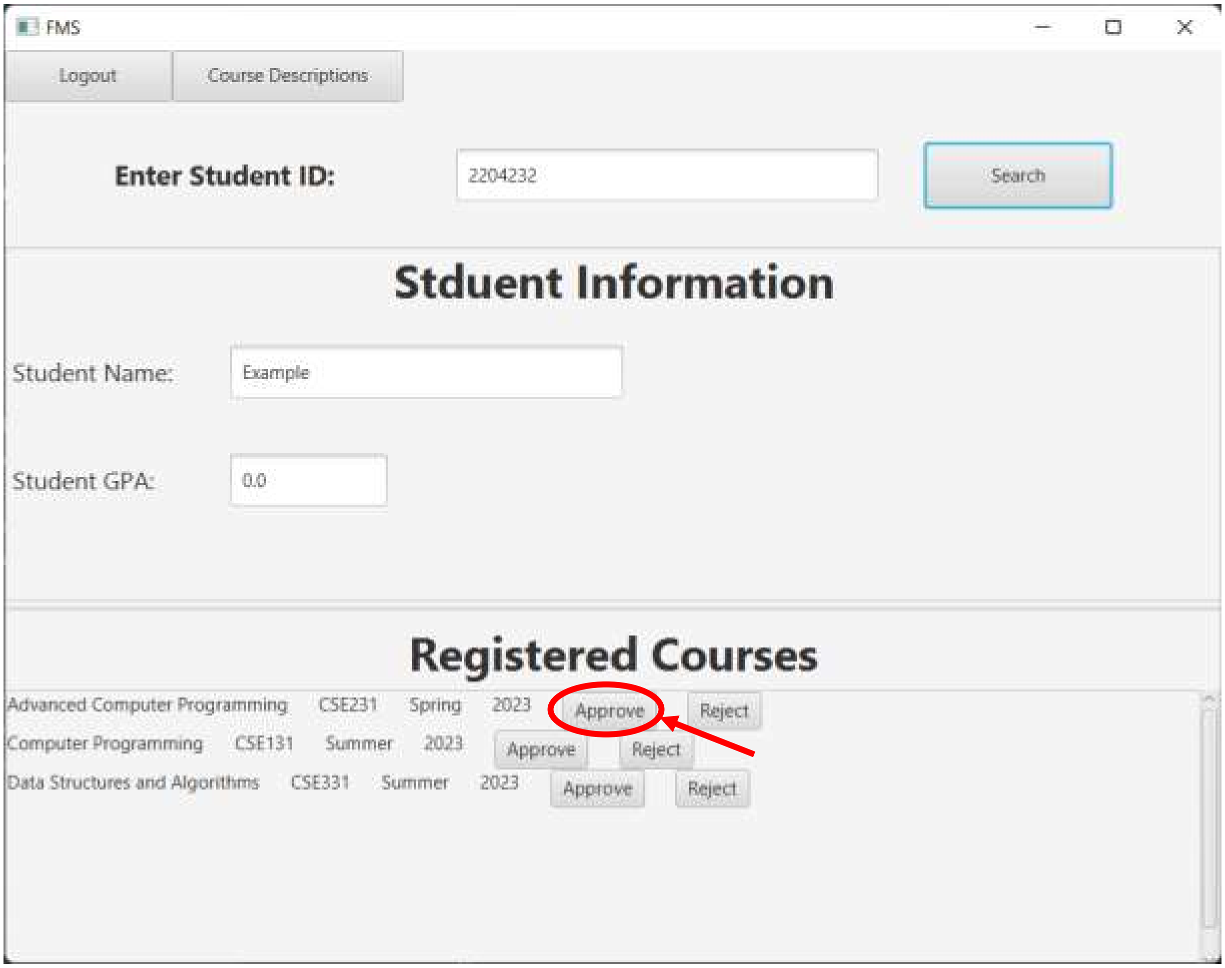


advisor@gmail.com

•••••

3. Approving or rejecting registrations

After signing in, you can search for a student faculty ID and approve/decline their pending requests



Enter Student ID: 2204232

Stduent Information

Student Name: Example

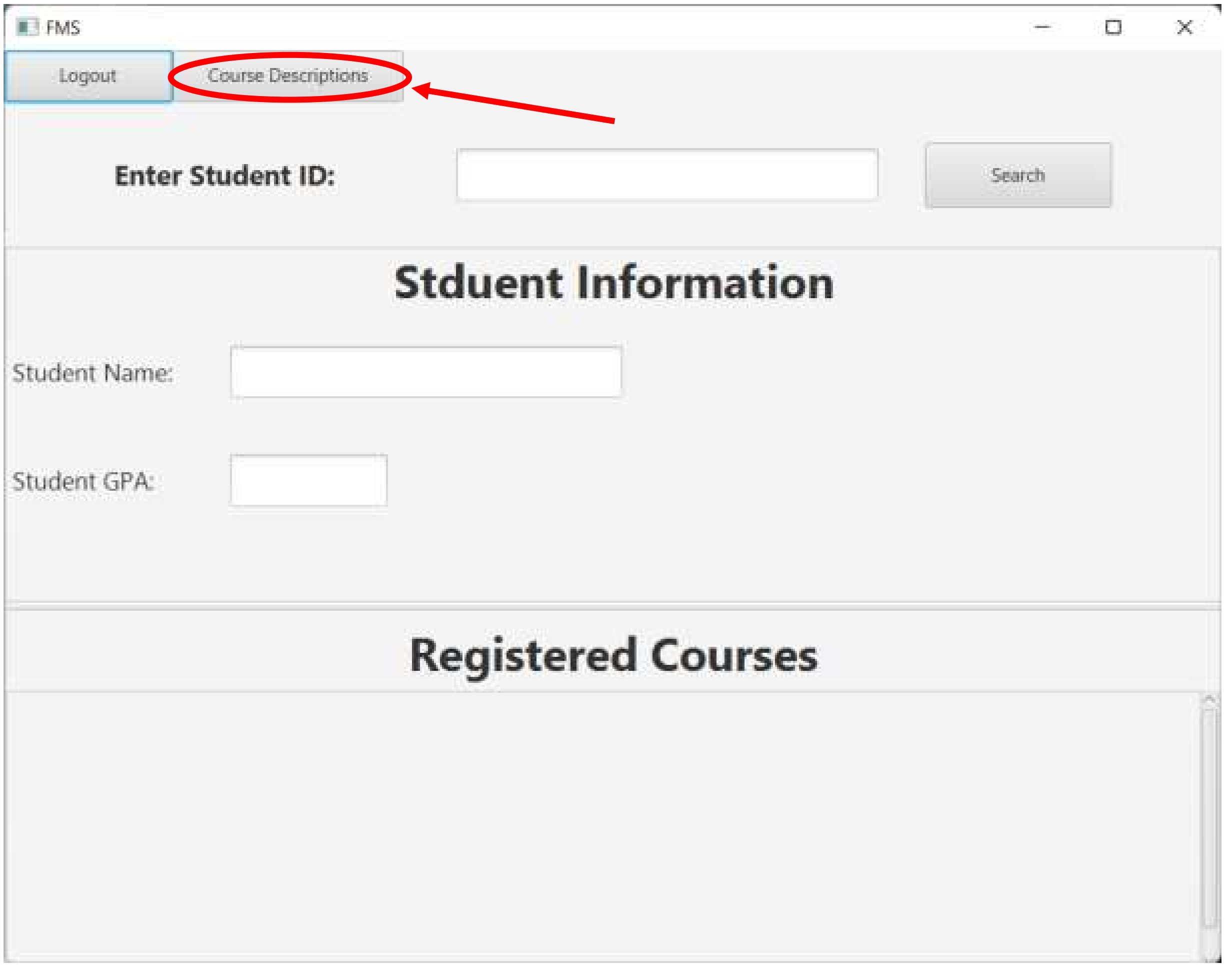
Student GPA: 0.0

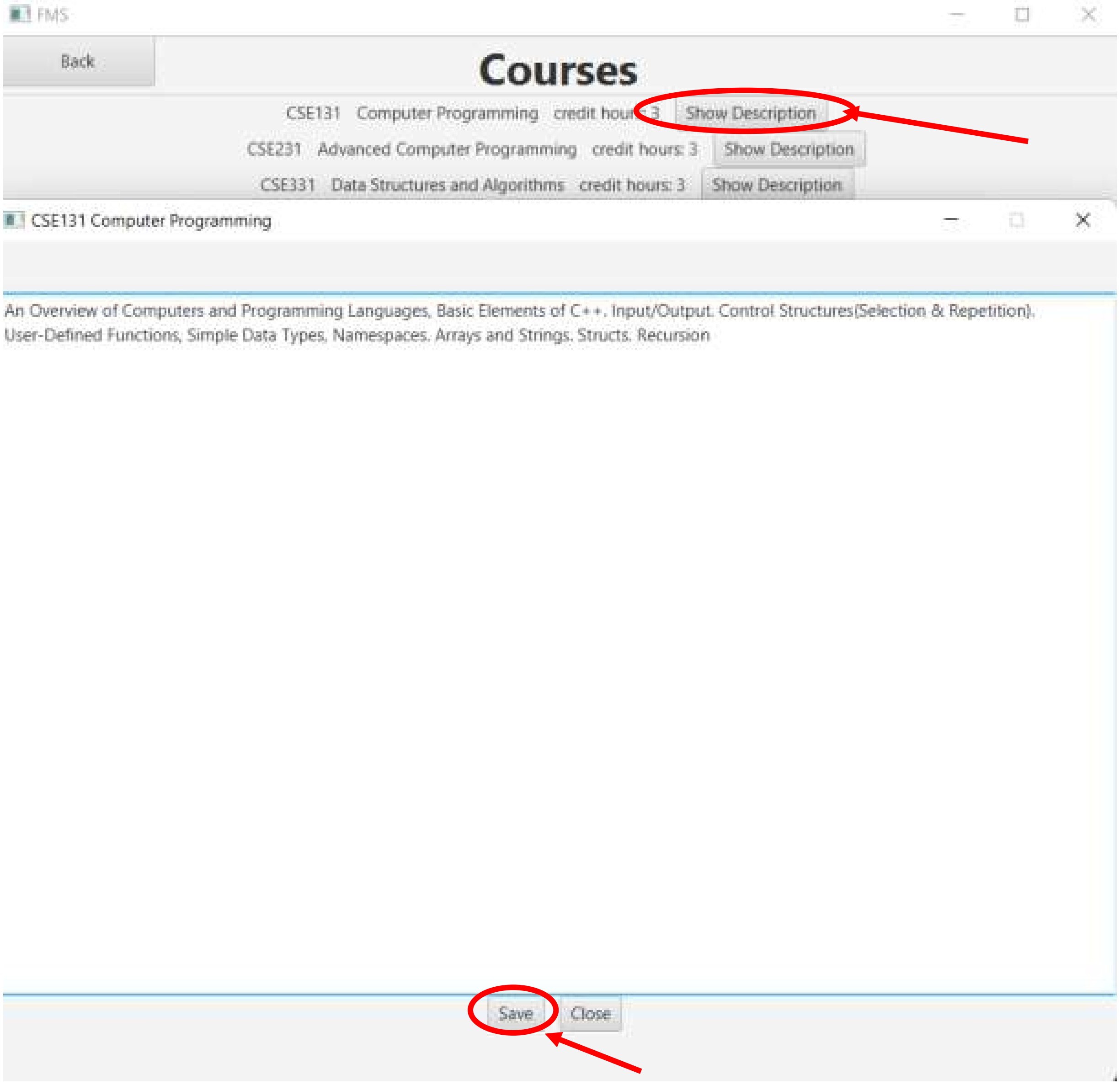
Registered Courses

```
Advanced Computer Programming    CSE231    Spring    2023
Computer Programming    CSE131    Summer    2023
Data Structures and Algorithms    CSE331    Summer    2023
```

4. Changing course descriptions

You can change course description as shown below





CSE131 Computer Programming credit hours: 3

CSE231 Advanced Computer Programming credit hours: 3

CSE331 Data Structures and Algorithms credit hours: 3

An Overview of Computers and Programming Languages, Basic Elements of C++. Input/Output. Control Structures(Selection & Repetition). User-Defined Functions, Simple Data Types, Namespaces. Arrays and Strings. Structs. Recursion