

# **CSE300 Software Engineering**

# **Learning Management System**

**Group 8**

# **Requirement Document**

# 

[**1. Introduction**](#_kx09p3w6yfjk) **4**

[1.1 Purpose](#_kr2gvnjd5pte) 4

[1.2 Scope of the Project](#_kr2gvnjd5pte) 4

[1.3 Intended Audience](#_kr2gvnjd5pte) 5

[1.4 Definitions, Acronyms and Abbreviations](#_kr2gvnjd5pte) 5

[**2. Overall Description**](#_hk25oxfinxe2) **6**

[2.1 Product Perspective](#_eyrqax454yqw) 6

[2.2 Product Functions](#_eyrqax454yqw) 6

[2.3 Operating Environment](#_eyrqax454yqw) 6

[2.4 Design and Implementation constraints](#_eyrqax454yqw) 7

[2.5 Assumptions and Dependencies](#_eyrqax454yqw) 7

[**3. SPECIFIC REQUIREMENTS**](#_30vdvmyjtj9w) **8**

[3.1 FUNCTIONAL REQUIREMENTS](#_ruj93wc3bfsr) 8

[3.1.1 User Class 1 - The Student](#_s1xraq20itdj) 8

[3.1.1.1 Functional requirement 1.1](#_nk60x55xn5us) 8

[3.1.1.2 Functional requirement 1.2](#_72v1olm934s) 8

[3.1.1.3 Functional requirement 1.3](#_vsdiglhtrmt4) 8

[3.1.1.4 Functional requirement 1.4](#_nk60x55xn5us) 9

[3.1.1.5 Functional requirement 1.5](#_9vrspxeasxbn) 9

[3.1.1.6 Functional requirement 1.6](#_39oxvgxzav0) 9

[3.1.1.7 Functional requirement 1.7](#_nk60x55xn5us) 9

[3.1.1.8 Functional requirement 1.8](#_nk60x55xn5us) 10

[3.1.1.9 Functional requirement 1.9](#_nk60x55xn5us) 10

[3.1.1.10 Functional requirement 1.10](#_nk60x55xn5us) 10

[3.1.1.11 Functional requirement 1.11](#_7denwakmx7nz) 10

[3.1.1.12 Functional requirement 1.12](#_7cq4vbj2jfas) 11

[3.1.1.13 Functional requirement 1.13](#_5u7qjj1kvf9h) 11

[3.1.1.14 Functional requirement 1.14](#_ao5n2dg9xbmh) 11

[3.1.2 User Class 2 - The Faculty](#_k1g0zjdrmalj) 11

[3.1.2.1 Functional requirement 2.1](#_y3rehb4272wi) 11

[3.1.2.2 Functional requirement 2.2](#_y3rehb4272wi) 12

[3.1.2.3 Functional requirement 2.3](#_y3rehb4272wi) 12

[3.1.2.4 Functional requirement 2.4](#_y3rehb4272wi) 12

[3.1.2.5 Functional requirement 2.5](#_y3rehb4272wi) 12

[3.1.2.6 Functional requirement 2.6](#_y3rehb4272wi) 13

[3.1.2.7 Functional requirement 2.7](#_g77e7haqek0f) 13

[3.1.2.8 Functional requirement 2.8](#_pc89xfwjftln) 13

[3.1.2.9 Functional requirement 2.9](#_ju7ivkcejzr5) 13

[3.1.2.10 Functional requirement 2.10](#_xgkgni5zartt) 14

[3.1.2.11 Functional requirement 2.11](#_zihm45f8oqp8) 14

[3.1.2.12 Functional requirement 2.12](#_hqfugdyubwxx) 14

[**3.2 NON- FUNCTIONAL REQUIREMENTS**](#_l17360g8j851) **15**

[3.2.1 Non- Functional requirement : Security](#_7aq6keaamfy2) 15

[3.2.2 Non- Functional requirement : Accessibility](#_r3f0lxrtcopc) 15

[3.2.3 Non- Functional requirement : Availability](#_fzyf99rzfuue) 16

[3.2.4 Non- Functional requirement : Efficiency](#_w96wjelueob6) 16

[3.2.5 Non- Functional requirement : Reliability](#_d4kpxws16vzh) 16

[3.2.6 Non- Functional requirement: Serviceability](#_w50h0b2qbu7y) 17

[3.2.7 Non- Functional requirement : Maintainability](#_t2s6rxvvtwid) 17

[3.2.8 Non- Functional requirement : Portability](#_aqc5jlrio0s7) 18

[3.2.9 Non- Functional requirement : Performance](#_9zlsgtshqzhm) 18

[3.2.10 Non- Functional requirement : Usability](#_qwykof4spypj) 18

[3.2.11 Non- Functional requirement : Scalability](#_vc1d9lbgywh8) 18

[3.2.12 Non- Functional requirement : Data integrity](#_5tchtblb7oef) 19

[**4. Use Case Diagram**](#_jylul78e47c) **20**

[**5. ER Diagram**](#_u3cppcjz7o33) **20**

[5.1 Logical ER Diagram](#_96afruuioyhw) 20

[5.2 Physical ER Diagram](#_yqnqk5jpefcl) 20

[**6. Data-Flow Diagram**](#_iwg84o1qp1km) **21**

[6.1 Level 0](#_eok5pcr6tqnn) 21

[6.2 Level 1](#_xa7wb1e0fv6c) 21

[6.3 Level 2](#_kvfesgl5y3do) 21

[**7. Sequence Diagram**](#_hsfhn6os6fh4) **21**

[7.1 Admin](#_x7600pvl37et) 21

[7.2 Faculty](#_xr4zybbyx3mr) 21

[7.3 Student](#_wazbyhf1hcvy) 21

[**8. State Diagram**](#_psay98d9rn5z) **21**

# 1. Introduction

## 1.1 Purpose

The Software Requirements Specification (SRS) document will provide a detailed description of the requirements for the Learning Management System. This SRS will provide an understanding of what is to be expected from the newly introduced system which is to be constructed. The clear understanding of the system and its functionality will help us to build the correct software for the end user and will be used for the development of the future stages of the project. It also provides a common reference point for both the developer team and stakeholder community. This document may evolve over time as the changes are made in the software

## 1.2 Scope of the Project

The Learning Management System, “LMS” is a Web application which helps students and faculty belonging to a University to facilitate and manage the assignments and grades in a more organized way. This web application should be free and accessible from any browser on desktop. This software includes all posting assignments by the faculty for specific subjects, where students can also submit their assignments. Later these assignments can be graded by faculties and they can also add comments to it.

The objectives of the learning Management System is to simplify the management of all the data related to academics that take place in a University. It will be a one-stop solution for both students and faculty to check their work related status. Safety, ease of use and most importantly the efficiency of information display and retrieval are some benefits the development team are going to present with this system. This system should be user appropriate, easy to use, provide easy recovery of errors and have an overall end user high subjective satisfaction.

## 1.3 Intended Audience

The intended audience of this document are as follows:

1. Designers

The designers will use this document as a reference while designing the system to make sure they build the system according to the customer needs

1. Developers

The developers will use this document as a reference while building the system. This will assist them while developing to build the website as per the customers needs.

1. Testers

The testers will be able to look back to this document and make sure that the basic requirements of the customers work properly and are tested rigorously.

## 1.4 Definitions, Acronyms and Abbreviations

|  |  |
| --- | --- |
| DESC | Description |
| RAT | Rationale |
| DEP | Dependency |
| TAG | A unique identifier |
| GIST | A very short and simple description of the concept |
| SCALE | The measurement metric used by the requirement |
| MUST | Minimum requirement for a functionality to work |
| WISH | The task desired to be achieved |
| LMS | Learning Management System |

# 

# 2. Overall Description

## 2.1 Product Perspective

The Learning Management System (LMS) is an online website that can be used by the users (university/school members) to improve their mode of communication. This software contains easy to make submission portals and resource sharing. The faculties will be able to grade the assignments and students will be able to access these grades over the same platform

Different interfaces for students, faculties as well as administrators will be provided in the website and a relational database will be used. This website will be used over the internet.

## 2.2 Product Functions

Learning Management System (LMS) will allow administrators to be able to create a class which will include one faculty member and multiple students.

The participating faculties will be able to create an assignment which will have a deadline and will allow students to submit even after that deadline has been met. The faculties will also be able to grade assignments that have been submitted already and provide comments related to these assignments. Faculties will also be able to share certain resources with the entire class.

The participating students will be able to view these assignments and submit them. They will also be able to view their grades and the comments given by the faculties. Students will also be able to view and download resources provided by the faculties.

## 2.3 Operating Environment

Server

* The system will need to be deployed on an online server
* Minimum storage capacity of 20 GB will be required which will be scaled up depending on the number of users

Client

* Any computer with a browser and internet connection will work.
* Minimum hardware required on client side is 1GB RAM with a display of resolution of 1280x720

## 2.4 Design and Implementation constraints

* Google Authentication for login will not be provided which is the primary method used for authentication in our university.
* The course registration will not be done on the platform and needs to be provided from outside the application into the database.
* The deletion of the submitted assignment will not be provided, so one a student has submitted a file. He/She can add a new file but will not be able to delete it.
* The students will only be able to see the courses details of the course they are enrolled in and cannot see the information about other courses.

# 

# 3. SPECIFIC REQUIREMENTS

## 3.1 FUNCTIONAL REQUIREMENTS

## 3.1.1 User Class 1 - The Student

### 3.1.1.1 Functional requirement 1.1

**ID:** FR1

**TITLE:** View Web Portal

**DESC:** A user should be able to view the web portal through any browser on the mobile phone.

**RAT:** In order for a user to view the web portal.

**DEP:** None

### 3.1.1.2 Functional requirement 1.2

**ID:** FR2

**TITLE:** User Login

**DESC:** A user should be able to login in the system by signing in with his/her assigned University Email Id.

**RAT:** In order for a user to login in the web portal.

**DEP:** FR1

### 3.1.1.3 Functional requirement 1.3

**ID :** FR3

**TITLE :** View student dashboard

**DESC :** A student is able to see their number of enrolled courses, number of due assignments and the total number of credits earned.

**RAT :** In order for students to view the dashboard.

**DEP :** FR2

### 3.1.1.4 Functional requirement 1.4

**ID :** FR4

**TITLE :** View all courses

**DESC :** A student is able to see all the list of their enrolled course names and course IDs.

**RAT :** In order for students to view list of courses

**DEP :** FR2, FR3

### 3.1.1.5 Functional requirement 1.5

**ID :** FR5

**TITLE :** View Course Assignments

**DESC :** This shows all the assignments, both submitted and pending of a particular course in which the student is enrolled. With its deadline if due, else the time of submission if the assignment is submitted.

**RAT :** In order for students to view a list of assignments.

**DEP :** FR4

### 3.1.1.6 Functional requirement 1.6

**ID :** FR6

**TITLE :** Submit Assignment

**DESC :** This allows Students to view assignment details and submit their assignment pdf on the web portal

**RAT :** In order for students to add their assignments.

**DEP :** FR5

### 3.1.1.7 Functional requirement 1.7

**ID :** FR7

**TITLE :** Modify Assignment

**DESC :** This allows students to modify their submitted assignment on the web-portal, which means they can re-submit the assignment(Change the file submitted).

**RAT :** In order for students to modify the assignment.

**DEP :** FR5

### 3.1.1.8 Functional requirement 1.8

**ID :** FR8

**TITLE :** View Grades

**DESC :** This allows students to view their grades for a specific assignment of a specific course.

**RAT :** In order for students to view grades.

**DEP :** FR4

### 3.1.1.9 Functional requirement 1.9

**ID :** FR9

**TITLE :** View faculty Comments

**DESC :** This function shows the comments posted by the faculty of a specific course in a specific assignment to the students.

**RAT :** In order for students to view faculty comments.

**DEP :** FR8, FR4

### 3.1.1.10 Functional requirement 1.10

**ID:** FR10

**TITLE:** View resources

**DESC:** This allows students to view the list of all resources uploaded by the faculty for a specific course.

**RAT:**  In order for students to view resources.

**DEP:** FR4

### 3.1.1.11 Functional requirement 1.11

**ID:** FR11

**TITLE:** Download Resources

**DESC:** This allows students to download the resources uploaded by the faculty for a specific course.

**RAT:** In order for students to download resources.

**DEP:** FR10

### 3.1.1.12 Functional requirement 1.12

**ID:** FR12

**TITLE:** View number of due assignments

**DESC:** This shows the number of pending assignments on the student dashboard, just to notify the student about the pending work.

**RAT:** In order for students to view the number of due assignments.

**DEP:** FR3

### 3.1.1.13 Functional requirement 1.13

**ID:** FR13

**TITLE:** View number of courses enrolled

**DESC:** This shows the total number of courses that the student is currently studying or enrolled in, which is displayed on the student dashboard.

**RAT:** In order for students to view the number of courses enrolled.

**DEP:** FR3

### 3.1.1.14 Functional requirement 1.14

**ID:** FR14

**TITLE:** View total credits earned

**DESC:** This shows the total number of credits taken by the student, which is displayed on the student dashboard.

**RAT:** In order for students to view the total credits earned.

**DEP:** FR3

## 3.1.2 User Class 2 - The Faculty

### 3.1.2.1 Functional requirement 2.1

**ID :** FR15

**TITLE :** View faculty dashboard

**DESC :** A faculty is able to see their dashboard with total number of enrolled students, number of ungraded assignments, number of courses assigned.

**RAT :** In order for faculty to view the dashboard.

**DEP :** FR2

### 3.1.2.2 Functional requirement 2.2

**ID :** FR16

**TITLE :** View Courses

**DESC :** It shows a list of all the courses with their names and ID assigned to that faculty, on the faculty dashboard.

**RAT :** In order for faculty to view the list of courses they take.

**DEP :** FR15

### 3.1.2.3 Functional requirement 2.3

**ID :** FR17

**TITLE :** View number of students enrolled

**DESC :** This shows the number of total students enrolled in each and every course taught by that faculty.

**RAT :** In order to show enrolled students in a course

**DEP :** FR15

### 3.1.2.4 Functional requirement 2.4

**ID :** FR18

**TITLE :** view posted assignments

**DESC :** This shows the list of all posted assignments and their details, of a particular course taught by the faculty.

**RAT :** In order for faculty to view posted assignments

**DEP :** FR2,FR16

### 3.1.2.5 Functional requirement 2.5

**ID :** FR19

**TITLE :** Add assignment

**DESC :** This allows faculty to upload new assignments for the students enrolled in that course along with the deadline and weightage of the assignment.

**RAT :** In order for faculty to post a new assignment.

**DEP :** FR16

### 3.1.2.6 Functional requirement 2.6

**ID :** FR20

**TITLE :** Edit assignment

**DESC :** This allows faculty to update the deadline, weightage and other details of the assignment for each course.

**RAT :** In order for faculty to modify the posted assignment.

**DEP :** FR16

### 3.1.2.7 Functional requirement 2.7

**ID :** FR21

**TITLE :** View submissions

**DESC :** This allows faculty to view all the submitted assignments of the students enrolled in that particular course.

**RAT :** In order for faculty to see all submitted the assignment

**DEP :** FR16

### 3.1.2.8 Functional requirement 2.8

**ID:** FR22

**TITLE:** Grade the submissions

**DESC:** The faculty is allowed to grade each and every submission submitted by the students for a particular course.

**RAT:** In order to evaluate the submitted assignment.

**DEP:** FR21

### 3.1.2.9 Functional requirement 2.9

**ID:** FR23

**TITLE:** Add Comments

**DESC:** This allows faculty to comment on the graded assignments for each student individually, on the web portal.

**RAT:** In order to add comments for each student.

**DEP:** FR21

### 3.1.2.10 Functional requirement 2.10

**ID:** FR24

**TITLE:** View resources

**DESC:** This allows faculty to view the list of all resources uploaded by him previously for a specific course.

**RAT:** In order to see the resources uploaded.

**DEP:** FR16

### 3.1.2.11 Functional requirement 2.11

**ID:** FR25

**TITLE:** Upload new resources

**DESC:** This allows faculty to upload new resources and documents related to the course for the enrolled students.

**RAT:** In order to add/upload a resource file.

**DEP:** FR24

### 3.1.2.12 Functional requirement 2.12

**ID:** FR26

**TITLE:** Change Password

**DESC:** This allows users to change their existing password and update their credentials to log in the web portal.

**RAT:** In order to update the password.

**DEP:** FR15

# 3.2 NON- FUNCTIONAL REQUIREMENTS

## 3.2.1 Non- Functional requirement : Security

**ID**: NR1

**TAG**: Admin/User Login Account

**GIST**: Security of accounts.

**SCALE**: If a user tries to log in to the web portal with an email account other than the registered email then the user should not be logged in. The user should be notified about log-in failure.

**MUST**: 100% of the time.

**ID**: NR2

**TAG**: User Login Account

**GIST**: Security of accounts.

**SCALE**: The password can be changed by the user only if he is logged in into the system. The user is not allowed to change email Id.

**MUST**: 100% of the time.

## 3.2.2 Non- Functional requirement : Accessibility

**ID**: NR3

**TITLE**: Application accessibility

**DESC**: The application should only be accessible by the registered user. No one else can able to login.

**MUST**: 100% of the time.

## 3.2.3 Non- Functional requirement : Availability

**ID**: NR4

**TAG**: System Availability

**GIST**: The availability of the system when it is used

**SCALE**: The average system availability without taking into consideration the network failure.

**MUST**: More than 98% of the time.

**WISH**: 100% of the time

**ID**: NR5

**TAG**: System Availability

**GIST**: Internet Connection Requirement

**SCALE**: The web application should be connected to the Internet. In order for the application to communicate with the database.

**MUST**: 100% of the time

## 3.2.4 Non- Functional requirement : Efficiency

**ID**: NR6

**TITLE**: Application efficiency

**GIST**: Rate of assignments entry

**SCALE**: The initial system shall be able to handle the entry of assignments by students at a minimum rate of 10 per second.

**MUST**: 100% of the time.

## 3.2.5 Non- Functional requirement : Reliability

**ID**: NR7

**TITLE**: Application reliability

**GIST**: Access without failure

**SCALE**: Students/Faculties can access their resources/assignments without failure.

**MUST**: 98% of the time.

**WISH**: 100% of the time

**ID**: NR8

**TITLE**: Application reliability

**GIST**: Upload Rate

**SCALE**: The system should give enough time for upload to complete and should not cancel the upload process quickly for students and faculties.

**MUST**: 98% of the time.

**WISH**: 100% of the time

## 3.2.6 Non- Functional requirement: Serviceability

**ID**: NR9

**TITLE**: Application serviceability

**GIST**: Email is sent when new assignment or grades are uploaded

**SCALE**: When the faculty’s automated emails are edited/replaced, there’s no need to recompile any code.

**MUST**: 98% of the time.

## 3.2.7 Non- Functional requirement : Maintainability

**ID**: NR10

**TITLE**: Application extensibility

**GIST**: In order for future functions to be implemented easily to the application.

**SCALE**: The application should be easy to extend. The code should be written in a way that favors the implementation of new functions.

**MUST**: 100% of the time

**ID**: NR11

**TITLE**: Application testability

**GIST**: In order to test the application.

**SCALE**: Test environments should be built for the application to allow testing of the applications' different functions.

**MUST**: 100% of the time

## 3.2.8 Non- Functional requirement : Portability

**ID**: NR12

**TITLE**: Application portability

**GIST**: The adaptable platform for the application to run on.

**SCALE**: If hosted, the application should be able to run in any browser.

**MUST**: 100% of the time

## 3.2.9 Non- Functional requirement : Performance

**ID**: NR13

**TAG**: Response Time

**GIST**: The fastness of the changing between different web pages

**SCALE**: The response time of the different output webpage after performing some actions.

**MUST**: No more than 5 seconds 100% of the time.

**WISH**: No more than 2 seconds 100% of the time.

## 3.2.10 Non- Functional requirement : Usability

**ID**: NR14

**TAG**: User Interface

**GIST**: Every feature should be clearly visible to the user.

**SCALE**: The website’s interface has to be user-friendly and easy to use. Any new user can be able to use this system easily without any difficulties.

**MUST**: 100% of the time.

## 3.2.11 Non- Functional requirement : Scalability

**ID**: NR15

**TITLE**: Application scalability

**GIST**: Addition of users

**SCALE**: The system shall be scalable to support unlimited growth in the number of users enrolled

**MUST**: 100% of the time.

## 3.2.12 Non- Functional requirement : Data integrity

**ID**: NR16

**TITLE**: Application Data integrity

**GIST**: Database updates

**SCALE**: The system shall maintain data integrity by keeping backups of all updates to the database for every records transaction.

**MUST**: 100% of the time.

3.2.13 Non- Functional requirement : Adaptability

**ID**: NR17

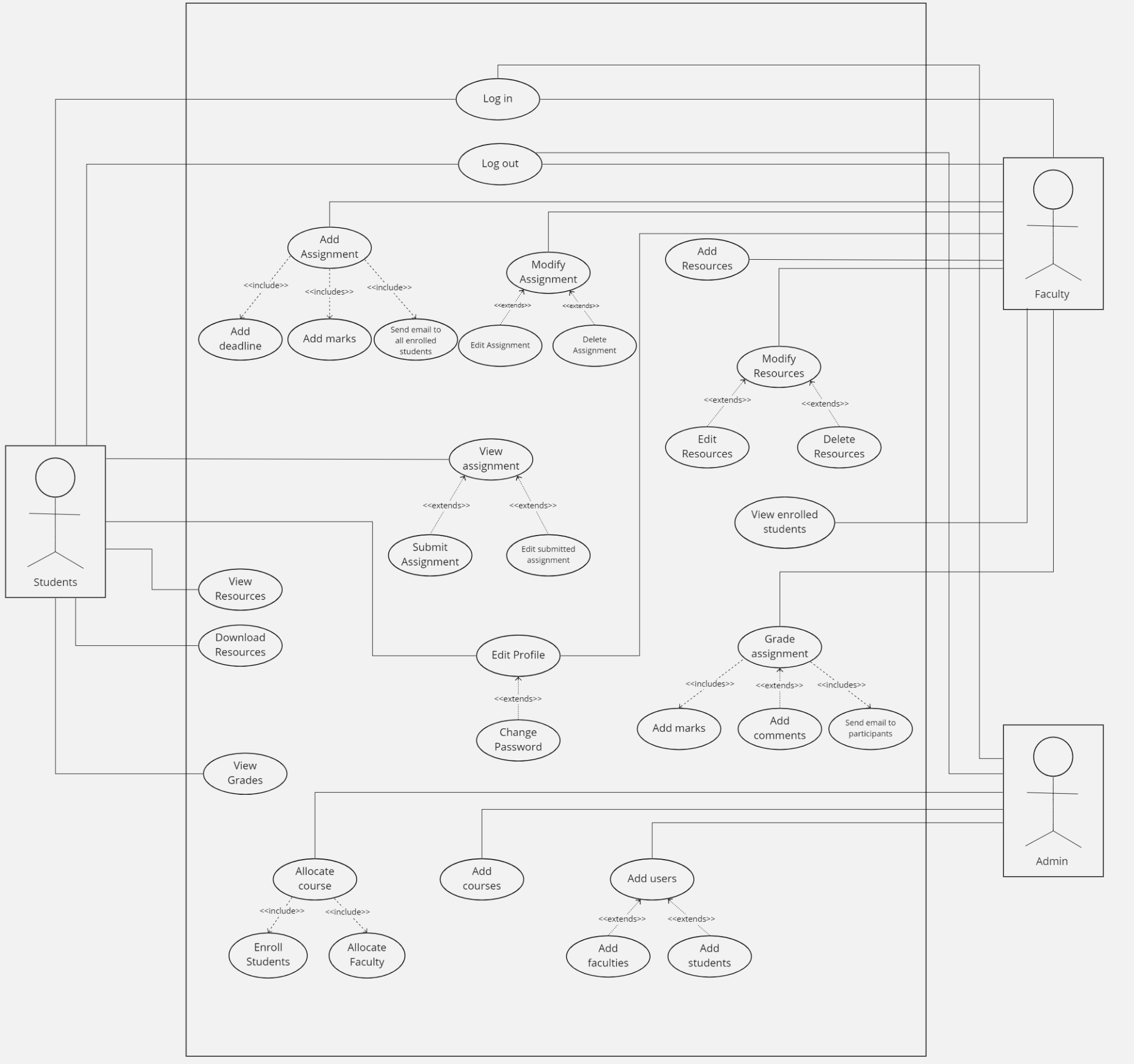
**TITLE**: Application adaptability

**GIST**: Change device

**SCALE**: The system should be responsive on any device mobile, tablet, laptop etc

**MUST**: 100% of the time.

# 4. Use Case Diagram



# 5. ER Diagram

## 5.1 Logical ER Diagram

## 

## 5.2 Physical ER Diagram

# 6. Data-Flow Diagram

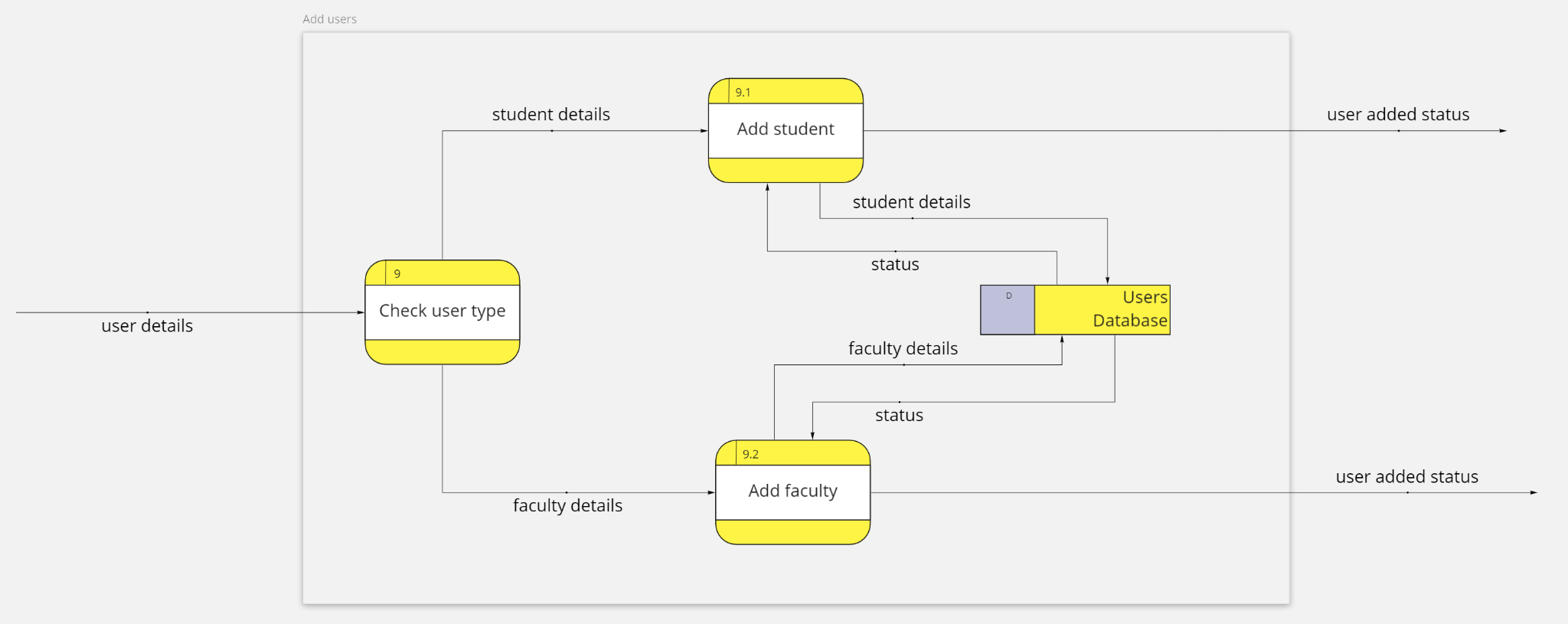
## 6.1 Level 0

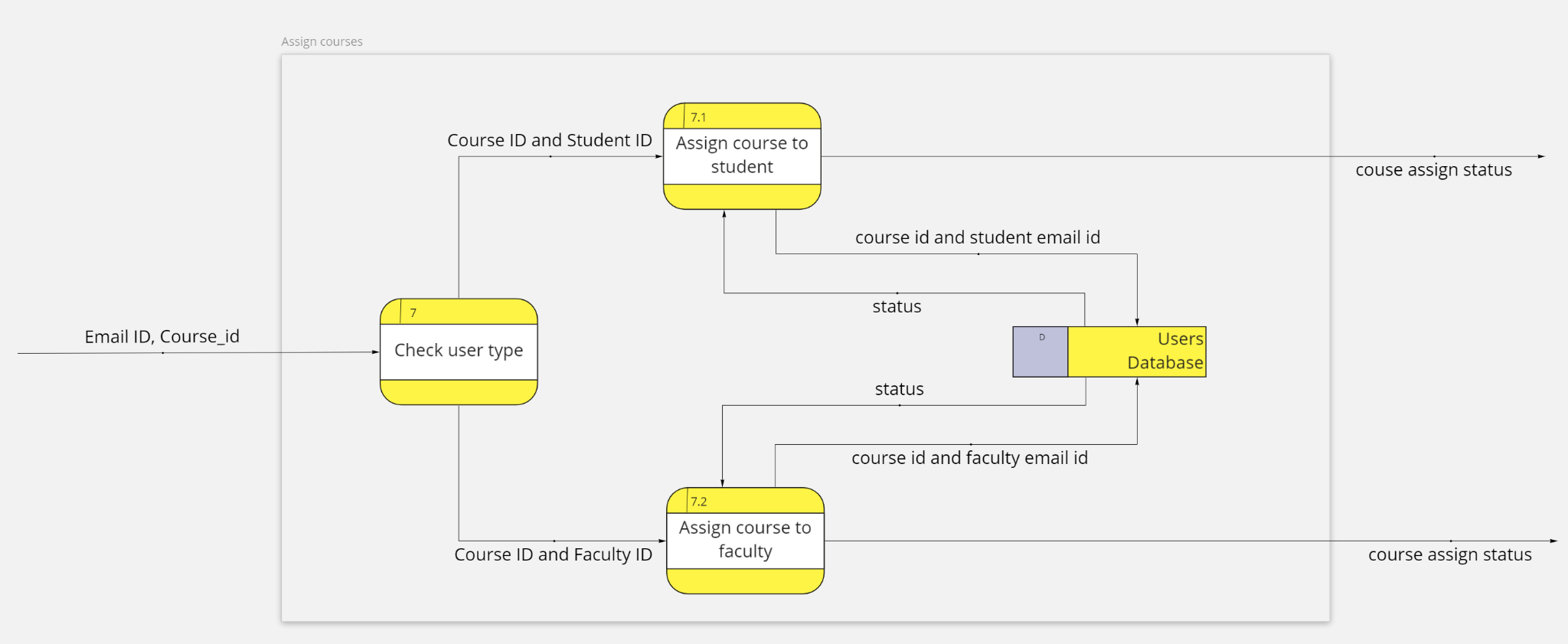
## 

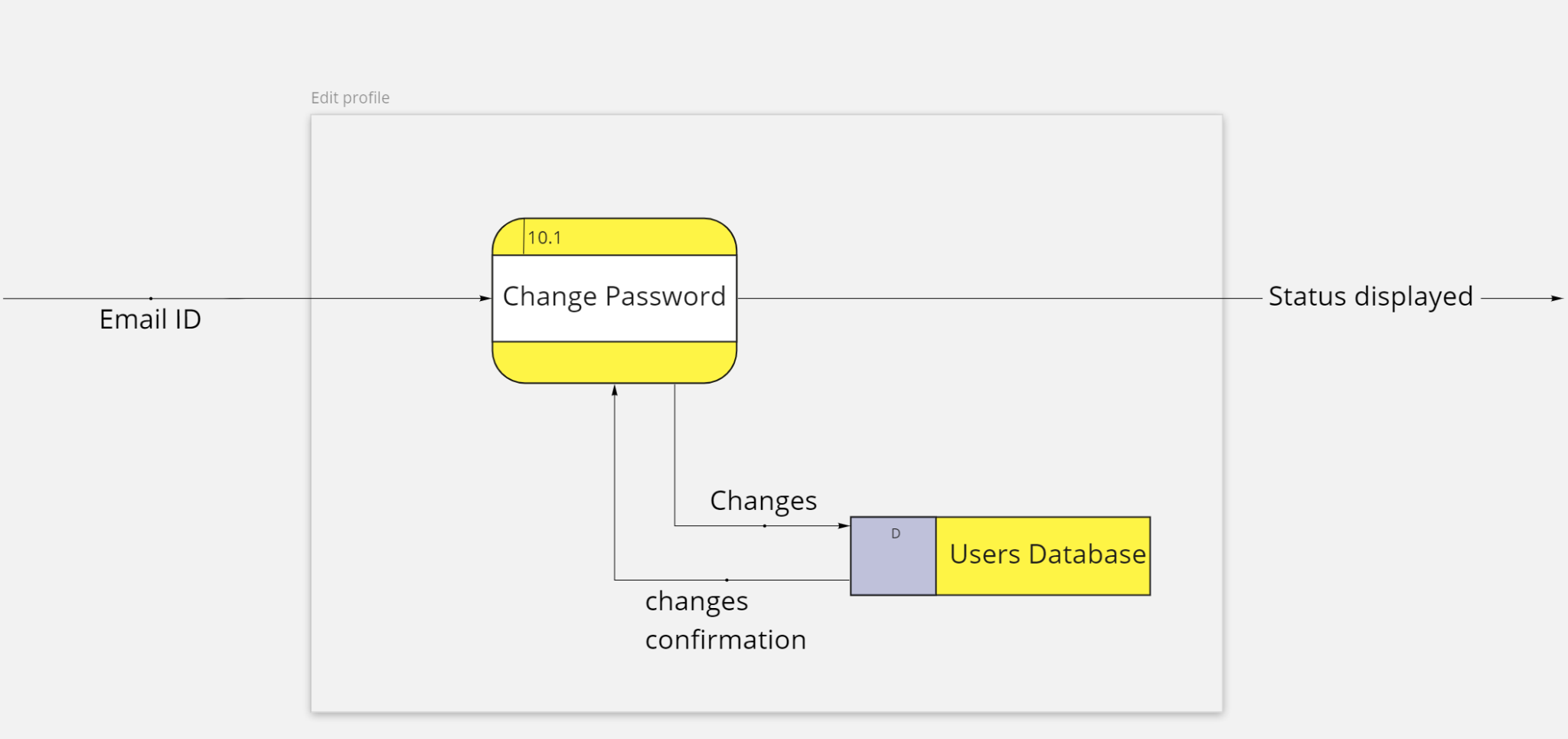
## 6.2 Level 1

## 

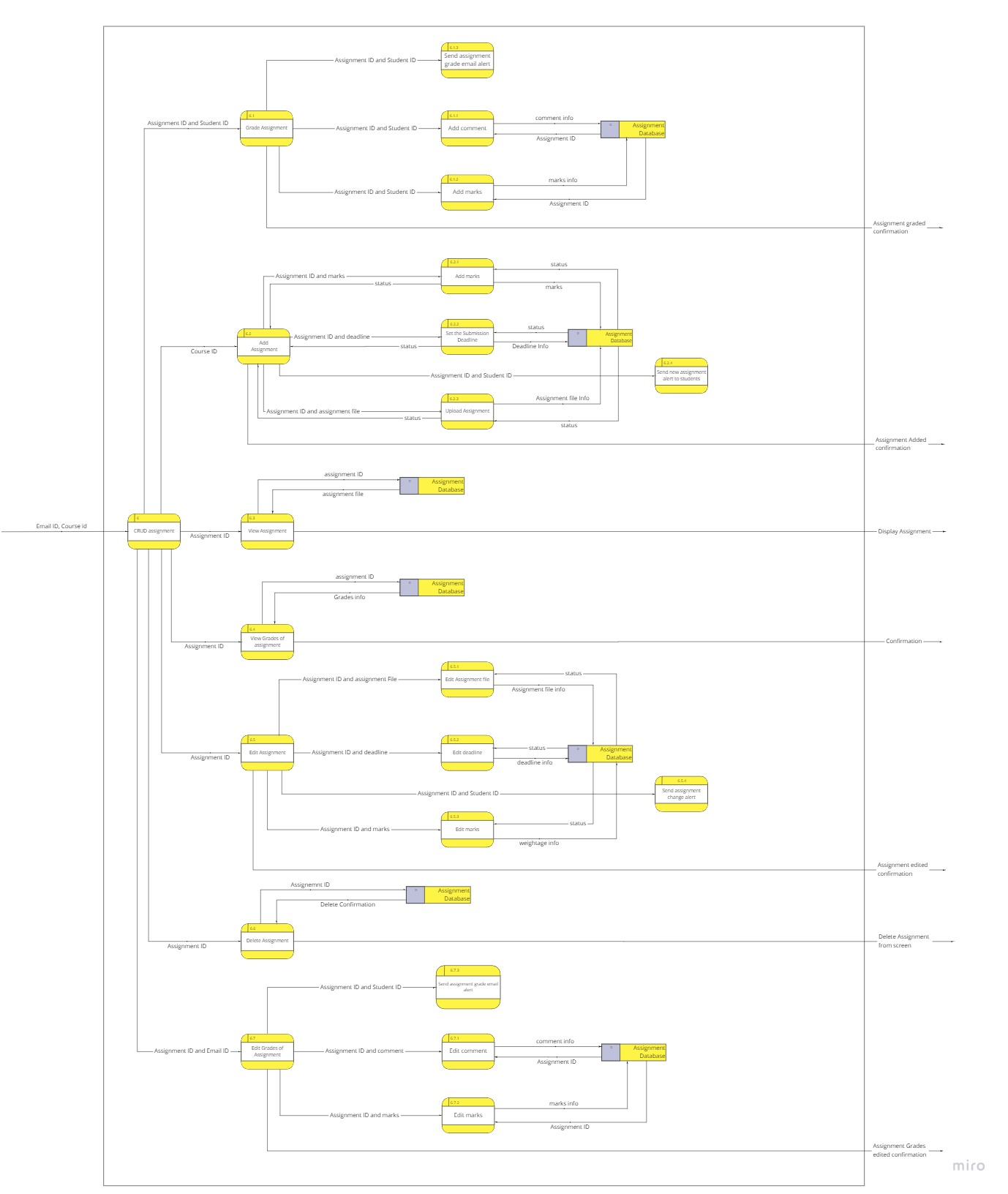
## 6.3 Level 2

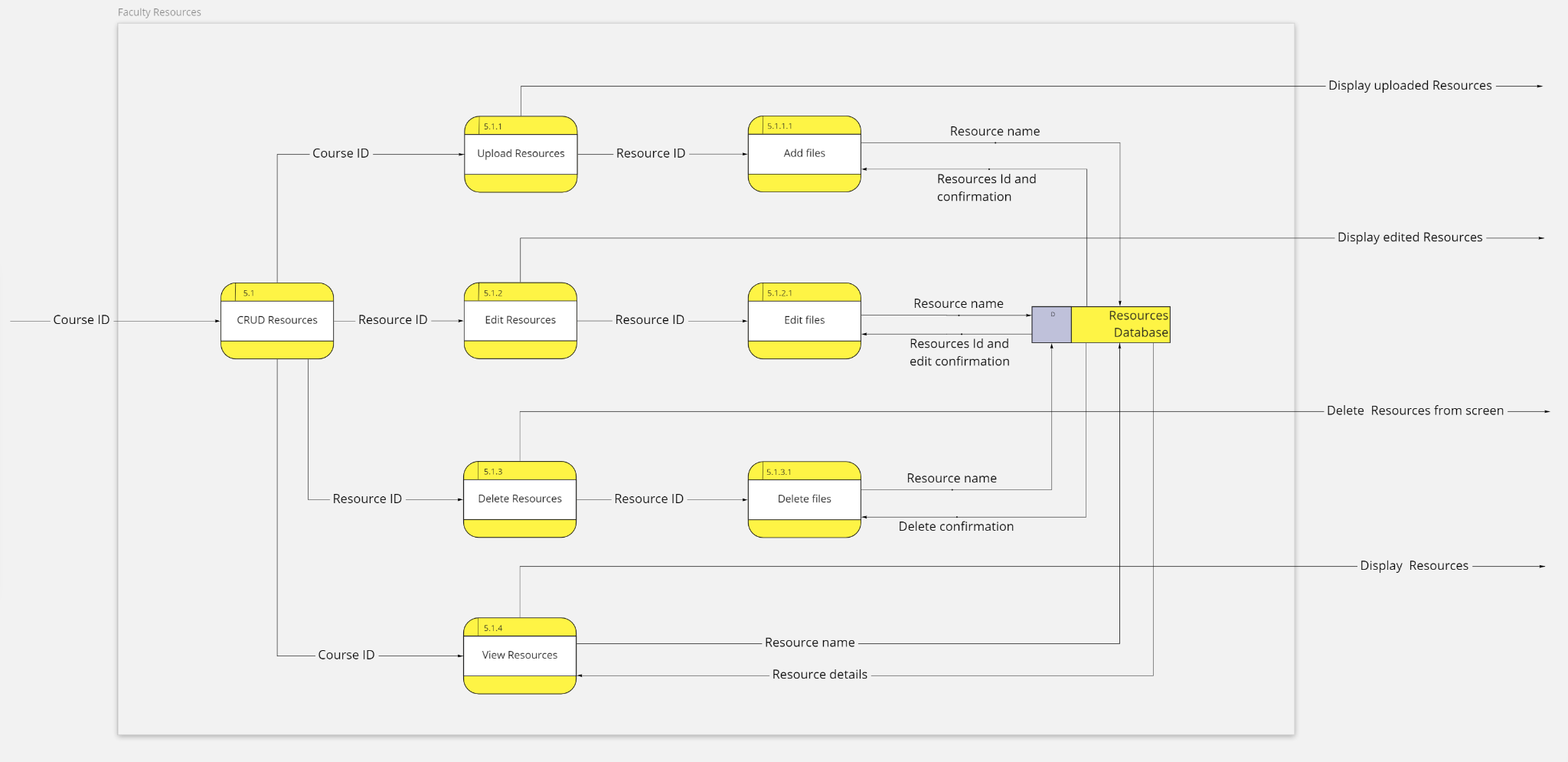
6.3.1 Add users

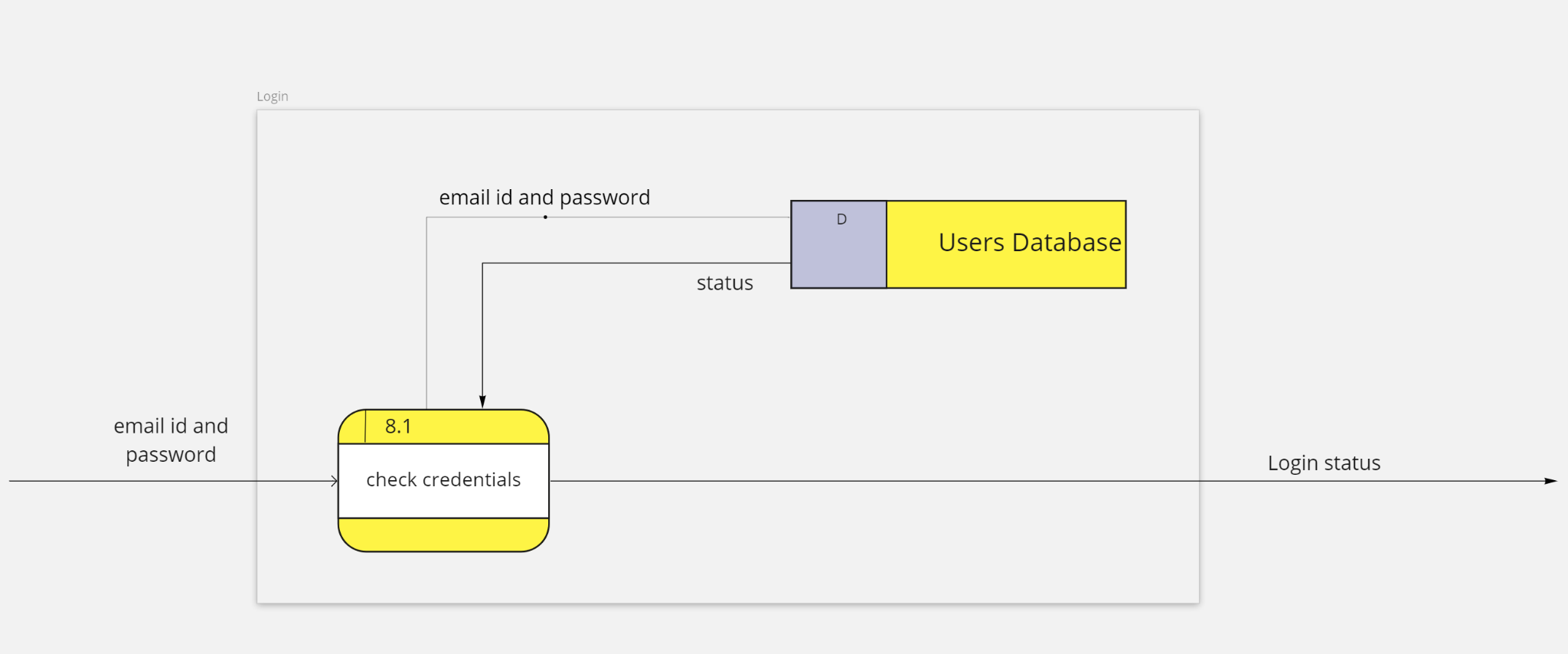
6.3.2 Assign courses

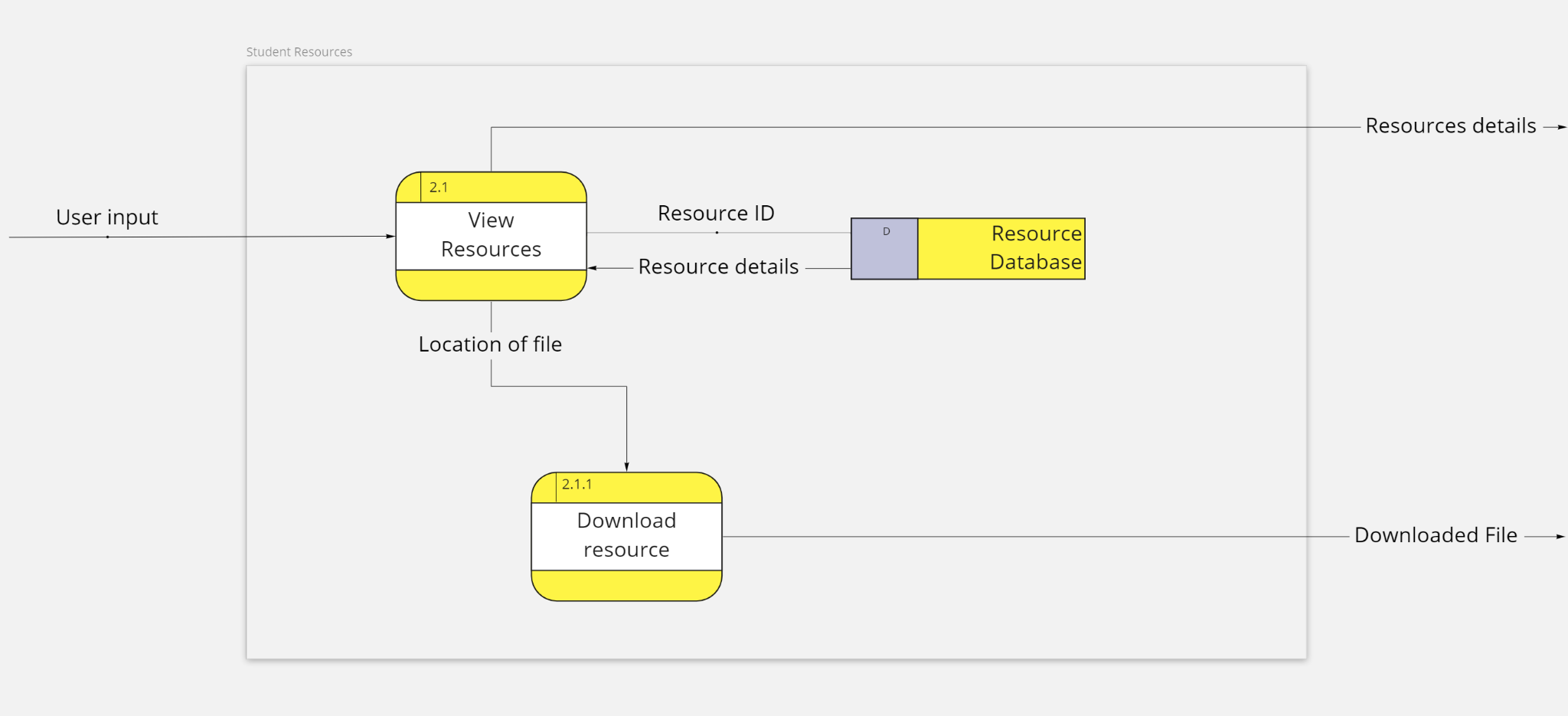
6.3.3 Edit Profile

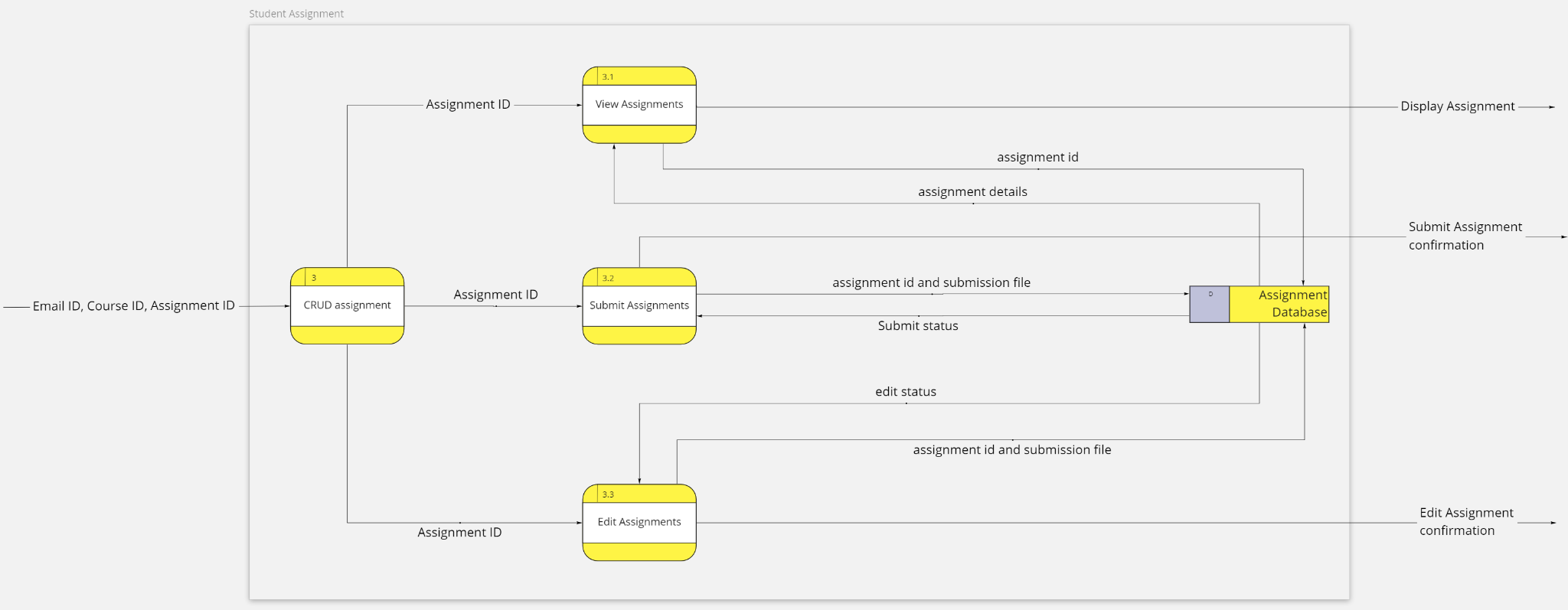
6.3.4 Faculty Assignment/grades

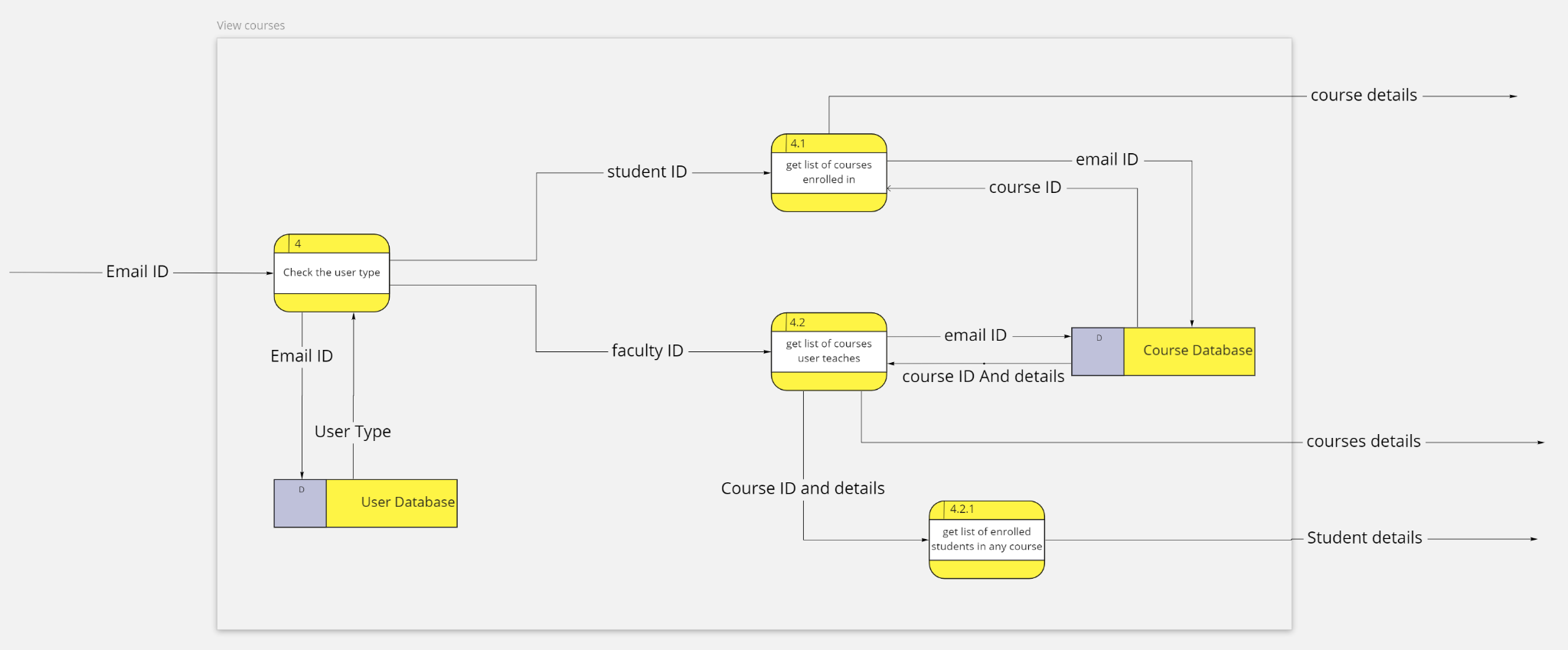


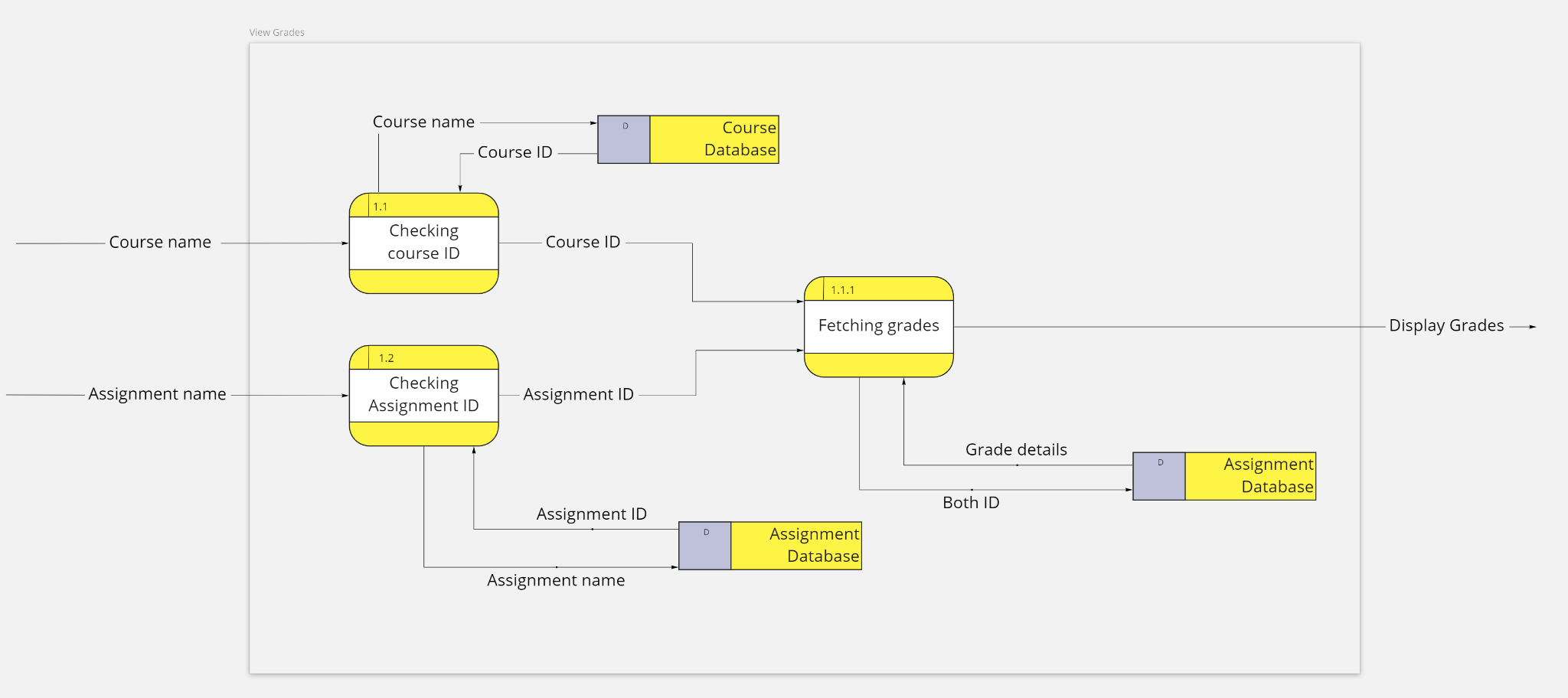
6.3.5 Faculty Resources

6.3.6 Login 

6.3.7 Student resources

6.3.8 Student Assignment

6.3.9 View Courses

6.3.10 View Grades

# 7. Sequence Diagram

## 7.1 Admin

## 

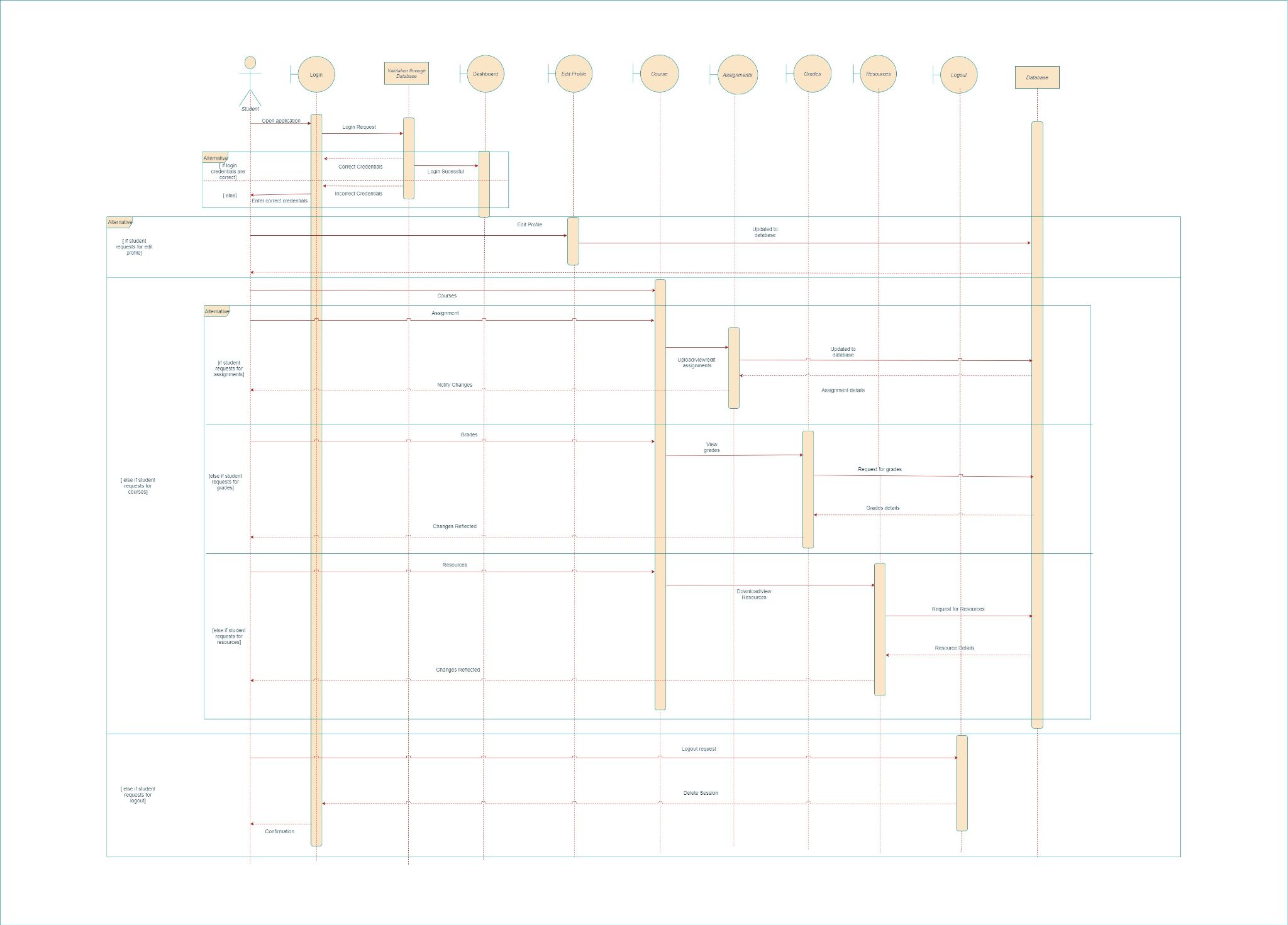
## 

## 7.2 Faculty

## 

## 

## 7.3 Student



# 8. State Diagram