ByteBoost Requirements Specification

Version 1.0

March 11, 2024

**Table of Contents**

[**BYTEBOOST REQUIREMENTS SPECIFICATION 1**](#_heading=h.19c6y18)

[**VERSION 1.0 1**](#_heading=h.3tbugp1)

**MARCH 11, 2024** [**1**](#_heading=h.28h4qwu)

[**1.**](#_heading=h.nmf14n) **EXECUTIVE SUMMARY 3**

[1.1](#_heading=h.37m2jsg) Project Overview 3

[1.2](#_heading=h.1mrcu09) Purpose and Scope of this Specification 3

[**2.**](#_heading=h.46r0co2) **PRODUCT/SERVICE DESCRIPTION 3**

[2.1](#_heading=h.2lwamvv) Product Context 3

[2.2](#_heading=h.111kx3o) User Characteristics 3

[2.3](#_heading=h.3l18frh) Assumptions 3

[2.4](#_heading=h.206ipza) Constraints 3

[2.5](#_heading=h.4k668n3) Dependencies 4

[**3.**](#_heading=h.2zbgiuw) **REQUIREMENTS 4**

[3.1](#_heading=h.1egqt2p) Functional Requirements 5

[3.2](#_heading=h.3ygebqi) Non-Functional Requirements 5

[*3.2.1*](#_heading=h.2dlolyb) *Product Requirements 5*

[**3.2.1.1**](#_heading=h.44sinio) **User Interface Requirements** 6

[**3.2.1.2**](#_heading=h.2jxsxqh) **Usability** 6

[**3.2.1.3**](#_heading=h.z337ya) **Efficiency** 6

[3.2.1.3.1](#_heading=h.3j2qqm3) Performance Requirements 6

[3.2.1.3.2](#_heading=h.1y810tw) Space Requirements 6

[**3.2.1.4**](#_heading=h.4i7ojhp) **Dependability** 6

[**3.2.1.5**](#_heading=h.2xcytpi) **Security** 7

[*3.2.2*](#_heading=h.sqyw64) *Organizational Requirements 7*

[**3.2.2.1**](#_heading=h.3whwml4) **Environmental Requirements** 7

[**3.2.2.2**](#_heading=h.2bn6wsx) **Operational Requirements** 7

[**3.2.2.3**](#_heading=h.qsh70q) **Development Requirements** 7

[*3.2.3*](#_heading=h.3cqmetx) *External Requirements 7*

[**3.2.3.1**](#_heading=h.1pxezwc) **Regulatory Requirements** 7

[**3.2.3.2**](#_heading=h.49x2ik5) **Ethical Requirements** 7

[**3.2.3.3**](#_heading=h.2p2csry) **Legislative Requirements** 7

[3.2.3.3.1](#_heading=h.147n2zr) Accounting Requirements 7

[3.2.3.3.2](#_heading=h.3o7alnk) Security Requirements 7

[3.3](#_heading=h.1rvwp1q) Domain Requirements 7

[**.**](#_heading=h.4bvk7pj)

# Executive Summary

## Project Overview

ByteBoost is a web-based learning platform targeting the students of Computer and Software Engineering primarily at Epoka University.

* Except curricular materials offered by Epoka University, this platform offers extracurricular materials such as: examinations, quizzes, video tutorials created and collected by our development team.
* Along with the lecturers’ guidance the application offers mentorship possibility working and collaboration with other volunteer’s students in Epoka with great achievements/high CGPA to help struggling students.

## Purpose and Scope of this Specification

**In scope**

* Aiding struggling students of Computer Engineering (CEN) and Software Engineering (SWE) to learn university courses in an easier way, thus progressing in their academic journey
* Boost the capabilities of the students autodidactic learning, an essential skill in the fields of CEN, SWE and engineering in general

**Out of Scope**

* Expansion of the platform to an even wider number of engineering fields
* Expansion of the platform to other educational institutions, other than Epoka University

# Product/Service Description

***2.1 Product Context***

* ByteBoost is an independent web-based learning platform tailored specifically for students of Computer and Software Engineering at Epoka University.
* It is designed to be self-contained, offering a comprehensive suite of tools and resources within the platform itself.
* While it primarily caters to the educational needs of Computer Engineering (CEN) and Software Engineering (SWE) students, it may be made to interface with existing learning management systems or educational platforms used within Epoka University to streamline certain processes.

***2.2 User Characteristics***

* **Student**
  + Typically, undergraduate or graduate students enrolled in Computer or Software Engineering programs.
  + Varying levels of technical expertise, ranging from novice to advanced users.
  + May have different learning preferences and habits, which is the idea behind the application.
* **Lecturers / Mentors**
  + Lecturers and Mentors at Epoka University
  + Experienced professionals in the field of Computer and Software Engineering.
  + High technical expertise.
  + May require administrative privileges for managing courses, assignments, and assessments.
* **System Administrator**
  + Administrative users responsible for maintaining the platform and providing technical support.
  + Moderate to high technical expertise, depending on their roles.
  + May not have in-depth knowledge of engineering concepts but need proficiency in using altering and maintaining the platform.

.

***2.3 Assumptions***

* Users have access to necessary hardware such as computers or mobile devices with internet connectivity.
* Users possess basic computer literacy skills to navigate and utilize the platform.
* Epoka University has adequate infrastructure support for hosting and maintaining the platform.
* Continuous availability of necessary resources like course materials and updates.

***2.4 Constraints***

* Access, Management, and Security:
  + Users' personal and academic information must be securely managed and protected.
* System Resource Constraints:
  + Limitations on server resources, such as disk space and bandwidth.
  + Scalability considerations to accommodate growing user base and content.
* Other Design Constraints:
  + Adherence to programming languages, frameworks, and development standards agreed upon by the development team.
  + Compatibility with existing Epoka University systems and infrastructure.

***2.5 Dependencies***

* Completion of backend development tasks before frontend features can be fully deployed.
* Continuous updates and maintenance to keep the platform synchronized with evolving curriculum requirements and educational standards.

# Requirements

## Functional Requirements

1. User Management:

* Ability to register as a new user (Student, Lecturer, Admin).
* User authentication (login/logout).
* Profile management (update profile information, change password).
* Student level users provide reviews after the end of each course.

2. Admin Management:

* Add, update, and delete accounts.
* Modify uploaded lectures and all their child uploaded files.
* Monitor all databases of the system.
* Have access to all users regardless of role.
* CRUD to all entries in every database of the system.

3. Course Management:

* Create, update, and delete courses.
* Assign Lecturers to courses.
* Set course enrollment limits.
* Manage course content (lessons, quizzes, mentor’s management, assignments).

4. Lesson Management:

* Create, update, and delete lessons within a course.
* Add multimedia content (text, images, videos) to lessons.
* Organize lessons into modules or sections.

5. Quiz Management:

* Create, update, and delete quizzes within a course.
* Define quiz questions (multiple choice, true/false, short answer, etc.).
* Set grading criteria and passing scores.

6. Assignment Management:

* Create, update, and delete assignments within a course.
* Specify assignment instructions, deadlines, and submission requirements.
* Grade assignments and provide feedback to students.

7. Enrollment Management:

* Allow students to enroll in courses.
* Set enrollment periods and restrictions.
* Handle course drops and refunds.

8. Progress Tracking:

* Track student progress within courses (completed lessons, quiz scores).
* Generate reports on student performance and course completion rates.
* Provide dashboards for students, Lecturers, and Admins to view progress.

9. Accessibility and Usability:

* Design an intuitive user interface for easy navigation and interaction.
* Support multiple devices (desktops, tablets, mobile phones).

10. Security:

* Implement secure authentication and authorization mechanisms.
* Protect user data and ensure compliance with data privacy regulations.
* Regularly update and patch system vulnerabilities.

11. Scalability and Performance:

* Design the system to handle a large number of concurrent users.
* Optimize database queries and server-side processing for efficient performance.
* Plan for scalability to accommodate future growth in users and content.

12. Integration:

* Integrate with external systems such as Learning Management Systems (LMS), content

repositories, or payment gateways.

* Support standards for interoperability with other e-learning platforms (e.g., SCORM,

LTI).

13. Feedback and Improvement:

* Collect feedback from users (Students) to identify areas for

improvement.

* Regularly update the system based on user feedback and emerging technologies.
* At the successful end of any course the user will be able to review the program.

## Non-Functional Requirements

### Product Requirements

#### **User Interface Requirements**

* The user interface should be intuitive and user-friendly, with clear navigation and layout.
* Screen formats should be organized logically to facilitate easy access to different features and content.
* Error messages should be descriptive and provide guidance on resolving issues.
* Function keys or shortcuts should be provided for frequently used actions to enhance user productivity.

#### **Usability**

Learnability:

* User documentation should be comprehensive and easily accessible.
* Context-sensitive help should be available to guide users through common tasks.
* The platform should be designed for easy learning, catering to users of varying technical backgrounds.

#### **Efficiency**

##### Performance Requirements

* The platform should support a specified number of simultaneous users without performance degradation.
* Response times for common tasks should meet defined thresholds (e.g., 95% of transactions completed within 3 seconds).
* The system should handle peak workload conditions efficiently without significant performance impact.

##### Space Requirements

* The system should be scalable to accommodate increasing data and user load.
* Disk space requirements should be monitored and managed to ensure optimal performance and storage capacity.

#### **Dependability**

Availability:

* The platform should be available during specified hours of operation to meet user needs.
* A minimum level of availability should be maintained to prevent disruptions to user access.
* Scheduled maintenance should be communicated to users in advance to minimize downtime.

Reliability:

* The system should have a defined mean time between failures (MTBF) to ensure reliable operation.

Monitoring:

* Continuous monitoring should be implemented to detect failures and errors promptly.
* Error detection mechanisms should be in place to identify and address issues proactively.

Maintenance:

* The system should be designed for ease of maintenance, with modular components and clear interfaces.

#### **Security**

* The platform should implement encryption mechanisms to protect sensitive data during transmission and storage.
* Activity logging should be maintained to track user actions and system events for auditing purposes.
* Authentication and authorization mechanisms should be robust to prevent unauthorized access to system resources.
* Data integrity checks should be performed to ensure that stored data remains accurate and unaltered.
  + 1. **Organizational Requirements**

**3.2.2.1 Environmental Requirements**

* The platform should comply with environmental regulations related to energy consumption and emissions.
* Consideration should be given to the environmental impact of hardware and infrastructure used to host the platform.

**3.2.2.2 Operational Requirements**

* Operational procedures should be documented to guide system administrators and support staff.
* Disaster recovery plans should be in place to mitigate the impact of unexpected outages or disasters.

**3.2.2.3 Development Requirements**

* Development processes should adhere to industry best practices and standards to ensure quality and reliability.
* Version control and change management procedures should be followed to track and manage code changes effectively.
  + 1. **External Requirements**

**3.2.3.1 Regulatory Requirements**

* The platform should comply with relevant regulatory standards and requirements governing data privacy and security.
* Audit trails should be maintained to satisfy regulatory requirements for data tracking and accountability.

**3.2.3.2 Ethical Requirements**

* The platform should uphold ethical principles related to user privacy, fairness, and transparency in data processing and decision-making.

**3.2.3.3 Legislative Requirements**

* Legislative requirements regarding data protection, accessibility, and intellectual property rights should be observed and implemented in the platform's design and operation.

## Domain Requirements

* Domain-specific requirements related to computer and software engineering education should be considered, such as curriculum alignment, industry relevance, and accreditation standards.

1. Software Design / Diagrams
   1. Requirements Analysis
      1. User Scenarios

4.1.1.1 User Scenarios List

|  |  |  |
| --- | --- | --- |
| Nr | Name | Description |
| US\_01 | User logs in | Users: students, lecturers and admins log in using username and password |
| US\_02 | Change password | Users: patients, receptionist change their password |
| US\_03 | Add a new student user  from SignUp form | User creates a student account from the from |
| US\_04 | Add a new lecturer user from SignUp form | User creates a lecturer account from the from |
| US\_05 | Students’ list | User: lecturer view all the students enrolled on a course. User: admin view all the students registered |
| US\_06 | Lecturers’ list | User: admin views all the lecturers registered |
| US\_7 | Update user: student | Student updates personal information of his own profile. User: admin can edit every registered student information |
| US\_8 | Update user: lecturer | Lecturer updates personal information of his own profile. User: admin can edit every registered lecturer information |
| US\_9 | Delete user | Admin can delete an account of an existing student or lecturer |
| US\_10 | User: lecturer can create courses | Lecturers can create a new course and edit its content. The course list appears in student’s dashboard. |
| US\_11 | View profile | Student, lecturer and admin can view their profiles |

|  |  |  |
| --- | --- | --- |
| US\_12 | View student’s progress | Lecturer can view all of their student’s progress for each enrolled course from his/her dashboard. |
| US\_13 | Student contacts lecturer and vice versa | A chat-system granted by the backend will ensure the communication between user: student and lecturer. |
| US\_14 | View statistics | The lecturer can view statistics of feedback generated by the system. |
| US\_15 | Leave feedback | Students can leave feedback about the course efficiency. |
| US\_16 | User logs out | Users: lecturer, student and admin can log out from their accounts |
| US\_17 | Course Enrollment Approval | The admin has the ability to manage course enrollment requests by students. |
| US\_18 | Reset Password | Users who have forgotten their password can reset it using a secure process. |
| US\_19 | Download Course Materials | Students can download course materials provided by lecturers. |
| US\_20 | Update Course Content | Lecturers are responsible for keeping course content up-to-date. They upload new materials or edit existing ones and submit the updates for student access, ensuring the course reflects the latest information. |
| US\_21 | Enroll in Course | Students have the ability to enroll in courses through the platform. |
| US\_22 | Access Course Material | Students can access all available course materials for courses they are enrolled in. |
| US\_23 | Submit Assignment | Students submit assignments through the platform. They upload their work, submit it for grading. |
| US\_24 | View Course Announcements | Students keep informed about their courses through announcements posted by lecturers. |
| US\_25 | Receive Graded Assignments | Students receive graded assignments with feedback from lecturers. |

4.1.1.2 User Scenarios Extended

**US\_01 User logs in:**

1. User navigates to the login page of the platform.
2. User enters their username and password.
3. The system validates the credentials.
4. Upon successful validation, the user is redirected to their respective dashboard based on their role (student, lecturer, or admin).
5. If the validation fails, an error message is displayed, and the user is prompted to try again.

**US\_02 Change password:**

1. User accesses the account settings or profile section of the platform.
2. User selects the option to change their password.
3. The system prompts the user to enter their current password and then the new desired password.
4. After confirming the new password, the system validates the changes and updates the password accordingly.
5. Upon successful password change, the user receives a confirmation message.

**US\_03 Add a new student user from SignUp form:**

1. User accesses the signup page or registration form provided on the platform.
2. User fills out the required information such as username, email, password, and student-specific details.
3. After submitting the form, the system validates the information and creates a new student account.
4. The newly created student receives a confirmation email with login instructions.
5. Admins may have access to review and approve new student registrations before they are fully activated.

**US\_04 Add a new lecturer user from SignUp form:**

1. Similar to adding a student user, the admin or authorized personnel accesses the signup page or registration form.
2. They fill out the required information for creating a new lecturer account, including username, email, password, and lecturer-specific details.
3. Upon submission, the system verifies the information and creates the new lecturer account.
4. The new lecturer receives a confirmation email with login details and instructions.
5. Admins may have the option to review and approve new lecturer registrations before they are fully activated.

**US\_05 Students’ list:**

1. For lecturers: Lecturers navigate to the course management section or dashboard.
2. They select the specific course they want to view the enrolled students for.
3. The system displays a list of students enrolled in that course, along with their relevant details such as name, email, and progress.
4. For admins: Admins access the user management section or dashboard.
5. They can filter or search for students and view a comprehensive list of all registered students on the platform.

**US\_06 Lecturers’ list:**

1. Admins access the user management section or dashboard.
2. They navigate to the section specifically for managing lecturers.
3. The system presents a list of all registered lecturers, including their details such as name, email, and courses taught.
4. Admins may have additional options to sort, filter, or perform actions such as editing or deleting lecturer accounts.

**US\_07 Update user: student:**

1. Students log in to their accounts and navigate to the profile or account settings section.
2. They select the option to edit their personal information.
3. The system displays a form populated with their existing details.
4. Students make the necessary changes (e.g., contact information, profile picture).
5. Upon submission, the system validates the changes and updates the student's profile.
6. For admins updating student information, they can access the user management section, search for the specific student, and edit their details accordingly.

**US\_08 Update user: lecturer:**

1. Similar to updating student information, lecturers log in and navigate to their profile or account settings.
2. They choose to edit their personal information.
3. The system presents a form with pre-filled details.
4. Lecturers modify the necessary information (e.g., contact details, qualifications).
5. After submitting the changes, the system validates and updates the lecturer's profile.
6. Admins, when updating lecturer information, follow a similar process in the user management section.

**US\_09 Delete user:**

1. Admins access the user management section or dashboard.
2. They search for the specific user (student or lecturer) they want to delete.
3. The system prompts for confirmation before proceeding with the deletion.
4. Upon confirmation, the user's account is permanently removed from the system.
5. Depending on system settings, associated data (e.g., course enrollments, feedback) may also be handled (e.g., archived or deleted).

**US\_10 User: lecturer can create courses:**

1. Lecturers log in to their accounts and navigate to the course management section.
2. They select the option to create a new course.
3. The system presents a form where lecturers enter details about the course (e.g., title, description, schedule).
4. Lecturers add relevant course materials such as syllabus, lecture slides, and assignments.
5. After completing the course creation, lecturers submit the information.
6. The newly created course is added to the platform, and students can now enroll in it.

**US\_11 View profile:**

1. Users (students, lecturers, admins) log in to their respective accounts.
2. They navigate to the profile or account settings section.
3. The system displays their profile information, including username, email, and any additional details.
4. Users can review their profile and make edits if necessary.
5. For admins, they may have access to view and edit profiles of all users on the platform.

**US\_12 View student’s progress:**

1. Lecturers log in to their accounts and access the course management section.
2. They select the specific course they want to view the students' progress for.
3. The system displays a dashboard or report showing each student enrolled in the course.
4. Lecturers can view individual student progress, including grades, assignments completed, and participation.
5. For each student, detailed information such as quiz scores, assignment submissions, and attendance records may be provided.
6. This allows lecturers to assess student performance and provide targeted support or interventions as needed.

**US\_13 Student contacts lecturer and vice versa:**

1. Students log in to their accounts and navigate to the messaging or communication section.
2. They initiate a new conversation and select the lecturer they want to contact.
3. Students compose their message, including the subject and content, and then send it.
4. Lecturers receive a notification about the new message and can respond directly within the platform.
5. Both students and lecturers can engage in a conversation thread, facilitating communication for queries, clarifications, or discussions related to coursework.

**US\_14 View statistics:**

1. Lecturers log in to their accounts and access the statistics or analytics section.
2. They can view various statistics related to course feedback, performance, and engagement.
3. The system presents visualizations such as charts, graphs, or tables to represent the data.
4. Lecturers can analyze feedback trends, student performance distributions, and other relevant metrics.
5. Additionally, lecturers may have options to filter or customize the statistics based on specific parameters or time periods.

**US\_15 Leave feedback:**

1. Students log in to their accounts and navigate to the course feedback or evaluation section.
2. They select the specific course they want to provide feedback for.
3. Students complete the feedback form, rating various aspects of the course (e.g., content, instructor effectiveness, course materials).
4. Additionally, students may have the opportunity to provide written comments or suggestions for improvement.
5. After submitting the feedback, the system records the responses and may provide a confirmation message.
6. Lecturers can access the aggregated feedback data to gain insights and make improvements to their courses.

**US\_16 User logs out:**

1. Users (students, lecturers, admins) navigate to the logout or sign-out option within the platform.
2. They click on the logout button, signaling their intent to end the current session.
3. The system clears the user's session data and logs them out of the platform.
4. Users are redirected to the login page or a confirmation screen confirming their successful logout.
5. Upon logout, users' access to platform features and data is terminated until they log in again.

**US\_17 Course Enrollment Approval**

1. Admin logs into their account and navigates to the enrollment approval section.
2. They review the list of pending enrollment requests from students.
3. For each request, the admin checks enrollment criteria such as prerequisites and class capacity.
4. The admin selects to approve or deny the enrollment request.
5. Upon approval or denial, an automatic notification is sent to the student.
6. The system updates the course roster and enrollment records accordingly.

**US\_18 - Reset Password:**

1. User selects the “Forgot Password” link on the login page.
2. They enter their registered email address and submit the request.
3. The system sends a password reset email with a secure link to the user’s email.
4. The user clicks on the reset link and is redirected to a password reset page.
5. They enter a new password, confirm it, and submit the changes.
6. The system updates the user’s credentials and displays a success message.

**US\_19 - Download Course Materials:**

1. Student logs into their account and selects the desired course.
2. They navigate to the course materials section.
3. Student browses the available materials and selects the ones to download.
4. They click the download link for each selected material.
5. The system initiates the download and saves the files to the student’s device.
6. After the download, the student can access the materials offline as needed.

**US\_20 - Update Course Content:**

1. Lecturer logs into their account and goes to the course management area.
2. They select the course they want to update.
3. Lecturer uploads new materials or edits existing content within the course.
4. They review the changes and submit them for student access.
5. The system updates the course content and notifies enrolled students of the changes.
6. Lecturer can view the updated course to ensure all materials are displayed correctly.

**US\_21 - Enroll in Course**

1. Student logs into their account.
2. Navigates to the course catalog.
3. Selects a course and views the details.
4. Clicks the enroll button within the enrollment period.
5. Receives a confirmation message upon successful enrollment.

**US\_22 - Access Course Material**

1. Student logs into their account and goes to their dashboard.
2. Selects an enrolled course.
3. Accesses available course materials, such as lecture notes or required reading documents.

**US\_23 - Submit Assignment:**

1. Student logs into their account and goes to the course where the assignment is posted.
2. Student clicks on the assignments section and selects the relevant assignment.
3. Reads the instructions and uploads the completed assignment file.
4. Submits the assignment by clicking the 'Submit' button.
5. System confirms receipt with a timestamp and sends a confirmation email to the student.
6. The student can check the status of the assignment review at any time.

**US\_24 - View Course Announcements:**

1. Student logs into their account and selects an enrolled course.
2. Clicks on the 'Announcements' section to view recent updates or information.
3. Reads through the list of announcements posted by the lecturer.
4. Student may acknowledge receipt or post queries related to the announcements if the feature is enabled.

**US\_25 - Receive Graded Assignments:**

1. Student logs into their account and goes to the 'Grades' section.
2. Selects the assignment for which feedback is expected.
3. Reviews the grade, lecturer's comments, and any annotations on the submitted work.
4. May ask for further clarification or submit a regrade request if the option is available.

**4.1.1.2 User Scenarios Extended**

|  |  |
| --- | --- |
| Name | User logs in |
| Summary | User enters their username and password to access his account |
| Actor | Student / Lecturer / Admin |
| Description | The system validates the credentials and the user gains access on his account after typing his correct username and password. |
| Precondition | User must have an account and should  choose one of the roles (student / lecturer / admin) before logging in. |
| Alternatives | If the validation fails, an error message is displayed, and the user is prompted to try again. |
| Post Condition | User is redirected to their respective dashboard based on their role (student, lecturer, or admin). |

***US\_01 – User logs in***

|  |  |
| --- | --- |
| Name | Change password |
| Summary | User accesses the account settings and selects the option to change their password. |
| Actor | Student / Lecturer / Admin |
| Description | The system prompts the user to enter their current password and then the new desired password. |
| Precondition | The user should be logged in. The old password  should be typed correctly and both fields with the  new password should match. Also, the new  password should be different from the old one  and it should follow the validation rules. |
| Alternatives | If the fields are not filled correctly (validation rules  not passed successfully), then information cannot  be saved. The user is allowed to try again. |
| Post Condition | After confirming the new password, the system validates the changes and updates the password accordingly. Upon successful password change, the user receives a confirmation message. |

***US\_02 – Change password***

|  |  |
| --- | --- |
| Name | Add a new student user from SignUp form |
| Summary | User accesses the signup page or registration form provided on the platform. |
| Actor | Student / Admin |
| Description | Student fills out the required information such as username, email, password, and student-specific details. |
| Precondition | The user should fill all the details and new student’s unique information should not match  with any other student in the database, so one  student cannot have two accounts. |
| Alternatives | If the system detects invalid or incomplete information, it prompts the user to correct the data and resubmit the form. The user is allowed to try again. |
| Post Condition | The newly created student receives a confirmation email with login instructions. Admins may have access to review and approve new student registrations before they are fully activated. |

***US\_03 – Add a new student user from SignUp form***

|  |  |
| --- | --- |
| Name | Add a new lecturer user from SignUp form |
| Summary | User accesses the signup page or registration form provided on the platform. |
| Actor | Lecturer / Admin |
| Description | Lecturer fills out the required information such as username, email, password, and lecturer-specific details. |
| Precondition | The user should fill all the details and new lecturer’s unique information should not match  with any other lecturer in the database, so one  lecturer cannot have two accounts. |
| Alternatives | If the system detects invalid or incomplete information, it prompts the user to correct the data and resubmit the form. The user is allowed to try again. |
| Post Condition | Upon submission, the system verifies the information and creates the new lecturer account.  The new lecturer receives a confirmation email with login details and instructions. Admins may have the option to review and approve new lecturer registrations before they are fully activated. |

***US\_04 – Add a new lecturer user from SignUp form***

|  |  |
| --- | --- |
| Name | Student’s list |
| Summary | Access the list of students. |
| Actor | Lecturer / Admin |
| Description | Lecturers navigate to the course management section or dashboard. They select the specific course they want to view the enrolled students for. The system displays a list of students enrolled in that course, along with their relevant details such as name, email, and progress. |
| Precondition | To access the list of students you should be logged in as the lecturer or as admin. |
| Alternatives | Admin can view the whole list of the students registered on the platform, whereas the lecturer can view only the list of his students. |
| Post Condition | A table with the list of the students is shown. |

***US\_05 – Student’s list***

|  |  |
| --- | --- |
| Name | Lecturers’ list |
| Summary | Access the list of lecturers. |
| Actor | Admin |
| Description | Admins access the user management section or dashboard. They navigate to the section specifically for managing lecturers to access the list of all lecturers on the platform. |
| Precondition | To access the list of lecturers you should be logged in as admin. Students and lecturers have no access on the list. |
| Alternatives | Admins may have additional options to sort, filter, or perform actions such as editing or deleting lecturers’ accounts. |
| Post Condition | A table with the list of the lecturers is shown. |

***US\_06 – Lecturers’ list***

|  |  |
| --- | --- |
| Name | Update user: student |
| Summary | Students edit their personal information. |
| Actor | Student / Admin |
| Description | Students log in to their accounts and navigate to the profile or account settings section. They select the option to edit their personal information. The system displays a form populated with their existing details. Students make the necessary changes (e.g., contact information, profile picture). For admins updating student information, they can access the user management section, search for the specific student, and edit their details accordingly. |
| Precondition | To modify student user information you should be logged in as either student or admin. |
| Alternatives | Admins may have additional options to perform actions such as editing or deleting students’ accounts. |
| Post Condition | Student’s personal information has been updated. |

***US\_07 – Update user: student***

|  |  |
| --- | --- |
| Name | Update user: lecturer |
| Summary | Lecturers edit their personal information. |
| Actor | Lecturer / Admin |
| Description | Lecturers log in to their accounts and navigate to the profile or account settings section. They select the option to edit their personal information. The system displays a form populated with their existing details. Lecturers make the necessary changes (e.g., contact information, profile picture). For admins updating lecturer information, they can access the user management section, search for the specific lecturer, and edit their details accordingly. |
| Precondition | To modify student user information you should be logged in as either lecturer or admin. |
| Alternatives | Admins may have additional options to perform actions such as editing or deleting lecturers’ accounts. |
| Post Condition | Lecturer’s personal information has been updated. |

***US\_08 – Update user: lecturer***

|  |  |
| --- | --- |
| Name | Delete user |
| Summary | Admin can delete an existing student or lecturer account. |
| Actor | Admin |
| Description | Admins access the user management section or dashboard. They search for the specific user (student or lecturer) they want to delete. The system prompts for confirmation before proceeding with the deletion. |
| Precondition | To delete user account you should be logged in as admin. The system prompts for confirmation before proceeding with the deletion. |
| Alternatives | Depending on system settings, associated data (e.g., course enrollments, feedback) may also be handled (e.g., archived or deleted). |
| Post Condition | The user (student or lecturer) is deleted from the database. |

***US\_09 – Delete user***

|  |  |
| --- | --- |
| Name | User: lecturer can create courses |
| Summary | Lecturers can create courses and add relevant course materials. |
| Actor | Lecturer |
| Description | Lecturers log in to their accounts and navigate to the course management section. They select the option to create a new course. The system presents a form where lecturers enter details about the course (e.g., title, description, schedule). Lecturers add relevant course materials such as syllabus, lecture slides, and assignments |
| Precondition | To create courses you should be logged in as lecturer. |
| Alternatives | After completing the course creation, lecturers submit the information. |
| Post Condition | The newly created course is added to the platform, and students can now enroll in it. |

***US\_10 – User: lecturer can create courses***

|  |  |
| --- | --- |
| Name | View profile |
| Summary | Users(students, lecturers, admins) log in to their respective accounts and navigate to their profile/account section. |
| Actor | Student / Lecturer / Admin |
| Description | Users log in to their respective accounts. They navigate to the profile or account settings section. The system displays their profile information, including username, email, and any additional details. Users can review their profile and make edits if necessary. |
| Precondition | To view your profile, you must be logged in your user account. |
| Alternatives | For admins, they may have access to view and edit profiles of all users on the platform. |
| Post Condition | The user views his personal information. |

***US\_11 – View profile***

|  |  |
| --- | --- |
| Name | View student’s progress |
| Summary | Lecturers log in to their accounts and view each students’ progress. |
| Actor | Lecturer / Admin |
| Description | Lecturers log in to their accounts and access the course management section. They select the specific course they want to view the students' progress for. The system displays a dashboard or report showing each student enrolled in the course. Lecturers can view individual student progress, including grades, assignments completed, and participation. For each student, detailed information such as quiz scores, assignment submissions, and attendance records may be provided. This allows lecturers to assess student performance and provide targeted support or interventions as needed. |
| Precondition | To view students’ progress, you must be logged in as lecturer and student should be enrolled in your course. |
| Alternatives | The student can also see his own progress. Admin can also see the platform students’ progress. |
| Post Condition | Lecturers assess student performance and provide targeted support or interventions as needed. |

***US\_12 – View student’s progress***

|  |  |
| --- | --- |
| Name | Student contacts lecturer and vice versa |
| Summary | Students and lecturers communicate through the messaging system. |
| Actor | Student / Lecturer / Admin |
| Description | Students log in to their accounts and access the messaging or communication section. They initiate a new conversation, selecting the lecturer they want to contact. Students compose a message, including subject and content, and send it. Lecturers receive a notification about the new message and can respond within the platform. Both students and lecturers can engage in conversation threads for queries, clarifications, or coursework discussions. |
| Precondition | Students must be logged in. Lecturers must be registered on the platform. |
| Alternatives | Lecturers can also initiate contact with students. Admin may oversee communication if necessary. |
| Post Condition | Communication between students and lecturers is facilitated effectively. |

***US\_13 – Student contacts lecturer and vice versa***

|  |  |
| --- | --- |
| Name | View statistics |
| Summary | Lecturers access and analyze course-related statistics. |
| Actor | Lecturer / Admin |
| Description | Lecturers log in to their accounts and navigate to the statistics or analytics section. They can access various statistics concerning course feedback, performance, and engagement. The system presents data through visualizations like charts, graphs, or tables. Lecturers can analyze feedback trends, student performance distributions, and other relevant metrics. Additionally, they may have options to filter or customize statistics based on specific parameters or time periods. |
| Precondition | Lecturers must be logged in. |
| Alternatives | Admin may also access and analyze statistics. |
| Post Condition | Lecturers gain insights into course performance and engagement. |

***US\_14 – View statistics***

|  |  |
| --- | --- |
| Name | Leave feedback |
| Summary | Students provide feedback on courses, which lecturers can access for improvement. |
| Actor | Student / Lecturer / Admin |
| Description | Students log in to their accounts and access the course feedback or evaluation section. They select the specific course they want to provide feedback for. Students complete a feedback form, rating various aspects of the course such as content, instructor effectiveness, and course materials. Additionally, they may provide written comments or suggestions for improvement. After submission, the system records the responses and may display a confirmation message. Lecturers can access aggregated feedback data to gain insights and make improvements to their courses. |
| Precondition | Students must be logged in. |
| Alternatives | Admin may review feedback data as well. |
| Post Condition | Lecturers gain valuable insights from student feedback to enhance course quality. |

***US\_15– Leave feedback***

|  |  |
| --- | --- |
| Name | User logs out |
| Summary | Users end their current session and log out of the platform. |
| Actor | Student / Lecturer / Admin |
| Description | Users, including students, lecturers, and admins, navigate to the logout or sign-out option within the platform. They click on the logout button, indicating their intention to end the current session. The system clears the user's session data and logs them out of the platform. Users are then redirected to the login page or a confirmation screen confirming their successful logout. Upon logout, users lose access to platform features and data until they log in again. |
| Precondition | Users must be logged in. |
| Alternatives | - |
| Post Condition | Users successfully end their session and are logged out of the platform. |

***US\_16– User logs out***

|  |  |
| --- | --- |
| Name | Course Enrollment Approval |
| Summary | Admins manage and decide on student enrollment requests. |
| Actor | Admin |
| Description | Admins log into their accounts and proceed to the enrollment approval section to review pending requests from students. Each request is examined against enrollment criteria such as prerequisites and class capacity. Based on this, admins approve or deny the enrollment. Notifications of the decision are sent out to students, and the course roster is updated accordingly in the system. |
| Precondition | Admins must be logged into their account. |
| Alternatives | - |
| Post Condition | Enrollment requests are processed, and students are notified of the outcome. |

***US\_17– Course Enrollment Approval***

|  |  |
| --- | --- |
| Name | Reset Password |
| Summary | Users can reset their forgotten passwords using a secure process. |
| Actor | Student / Lecturer / Admin |
| Description | Users click the “Forgot Password” link on the login page, enter their registered email, and submit a reset request. The system sends a password reset email with a secure link. Users click on this link and are redirected to reset their password, which the system confirms with a success message. |
| Precondition | Users must have a registered email with the platform. |
| Alternatives | - |
| Post Condition | Users have reset their passwords and can log in with the new credentials. |

***US\_18– Reset Password***

|  |  |
| --- | --- |
| Name | Download Course Materials |
| Summary | Students can download materials for offline access. |
| Actor | Student |
| Description | Students log into their account, select a course, and navigate to the materials section. They select materials to download for offline viewing. The system confirms the download and saves the files. |
| Precondition | Students must be enrolled in the course and logged in. |
| Alternatives | - |
| Post Condition | Students have the course materials available for offline use. |

***US\_19– Download Course Materials***

|  |  |
| --- | --- |
| Name | Update Course Content |
| Summary | Lecturers update and manage course content for students. |
| Actor | Lecturer |
| Description | Lecturers log into their account, go to the course management area, and select a course to update. They upload or edit content, submit changes, and the system updates the course. Enrolled students are notified of the changes. |
| Precondition | Lecturers must be logged in and assigned to the course. |
| Alternatives | Admins may assist in content management. |
| Post Condition | Course content is updated. |

***US\_20– Update Course Content***

|  |  |
| --- | --- |
| Name | Enroll in Course |
| Summary | Students enroll in courses during the enrollment period. |
| Actor | Student |
| Description | Students log into their account, navigate to the course catalog, select a course, and click the enroll button. They receive confirmation upon successful enrollment. |
| Precondition | The enrollment period must be open, and students must be logged in. |
| Alternatives | Admin approval might be required for enrollment. |
| Post Condition | Students are enrolled in the course and can access course materials. |

***US\_21– Enroll in Course***

|  |  |
| --- | --- |
| Name | Access Course Materials |
| Summary | Students access available learning resources for their enrolled courses. |
| Actor | Student |
| Description | Students log into the platform, navigate to their dashboard, and select an enrolled course. Within the course, they access a repository of course materials such as lecture notes, required readings, and other supplementary documents provided by the lecturer. The materials are available for viewing or downloading, enabling students to engage with the course content outside of scheduled classes. |
| Precondition | Students must be enrolled in the course and logged in to access materials. |
| Alternatives | Lecturers and admins may have the ability to view all course materials across different courses for monitoring and quality assurance purposes. |
| Post Condition | Students successfully access and utilize course materials to supplement their learning experience. |

***US\_22– Access Course Materials***

|  |  |
| --- | --- |
| Name | Submit Assignment |
| Summary | Students upload and submit assignments for grading. |
| Actor | Student |
| Description | Students log into their account, select a course, and navigate to the assignment they need to submit. They upload their work, submit the assignment, and receive a confirmation email. |
| Precondition | Students must be enrolled in the course and logged in. |
| Alternatives | - |
| Post Condition | Assignments are submitted and await grading by the lecturer. |

***US\_23– Submit Assignment***

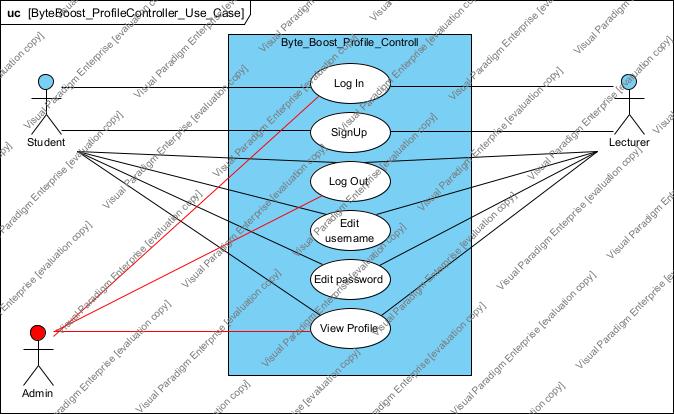
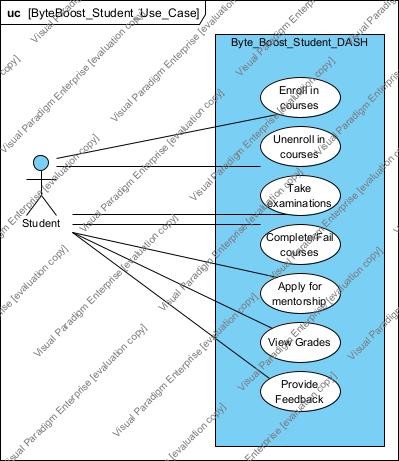
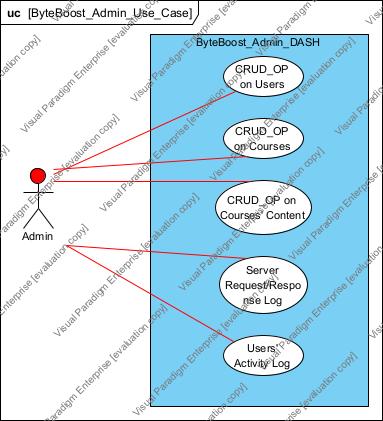
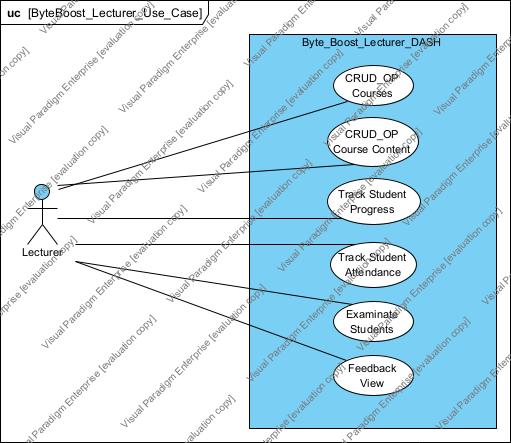
|  |  |
| --- | --- |
| Name | View Course Announcements |
| Summary | Students access announcements to stay informed about their courses. |
| Actor | Student |
| Description | Students log into their account, select a course, and click on the 'Announcements' section to read updates from lecturers. |
| Precondition | Students must be logged in and enrolled in the course. |
| Alternatives | Lecturers and admins can also post announcements. |
| Post Condition | Students are updated with the latest course information. |

***US\_24– View Course Announcements***

|  |  |
| --- | --- |
| Name | Receive Graded Assignments |
| Summary | Students review grades and feedback on submitted assignments. |
| Actor | Student |
| Description | Students log into their account and go to the 'Grades' section. They select an assignment to view the grade and feedback from the lecturer. |
| Precondition | Students must be logged in and have submitted assignments. |
| Alternatives | - |
| Post Condition | Students receive feedback and understand their performance. |

***US\_25– Receive Graded Assignments***

**4.2 Behavioral Diagrams**

**4.2.1 Use Case Diagrams**

**4.2.2 Activity Diagrams**

**A diagram of a username and password

Description automatically generated**

**AD\_01-***US\_01* ***– User logs in***

**A diagram of a computer program

Description automatically generated**

**AD\_02 -** *US\_02* ***– Change password***

**A diagram of a diagram

Description automatically generated**

**AD\_03 -** *US\_03* ***– Add a new student user from SignUp form***

**A diagram of a diagram

Description automatically generated**

**AD\_04 -** *US\_04* ***– Add a new lecturer user from SignUp form***

**A diagram of a user navigation

Description automatically generated**

**AD\_05 -** *US\_05* ***– Student’s list***

**A diagram of a software flowchart

Description automatically generated**

**AD\_06 -** *US\_06* ***– Lecturers’ list***

**A diagram of a student

Description automatically generated**

**AD\_07 -** *US\_07* ***– Update user: student***

**A diagram of a software development

Description automatically generated with medium confidence**

**AD\_08 -** *US\_08* ***– Update user: lecturer***

**A diagram of a system

Description automatically generated**

**AD\_09 -** *US\_09* ***– Delete user***

**A diagram of a course

Description automatically generated**

**AD\_10 -** *US\_10* ***– User: lecturer can create courses***

**A diagram of a program

Description automatically generated**

**AD\_11 -** *US\_11* ***– View profile***

**A diagram of a flowchart

Description automatically generated**

**AD\_12 -** *US\_12* ***– View student’s progress***

**A diagram of a company

Description automatically generated**

**AD\_13 -***US\_13* ***– Student contacts lecturer and vice versa***

**A diagram of a company

Description automatically generated**

**AD\_14 -** *US\_14* ***– View statistics***

**A diagram of a course

Description automatically generated**

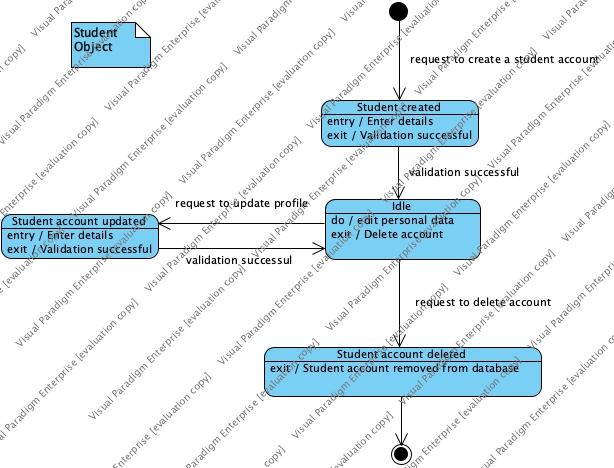
**AD\_15 -** *US\_15* ***– Leave feedback***

**A diagram of a login page

Description automatically generated**

**AD\_16 -** *US\_16* ***– User logs out***

**4.2.3 State Diagrams**

****

**SD\_01 - Student**

**A diagram of a project

Description automatically generated**

**SD\_02 – Lecturer**

**A diagram of course completion

Description automatically generated**

**SD\_03 - CourseA diagram of a course

Description automatically generated**

**SD\_04 - FeedbackA diagram of a student assignment

Description automatically generated**

**SD\_05 – Grade/Assignments**

**4.2.4 Sequence Diagrams**

**A diagram of a login program

Description automatically generated**

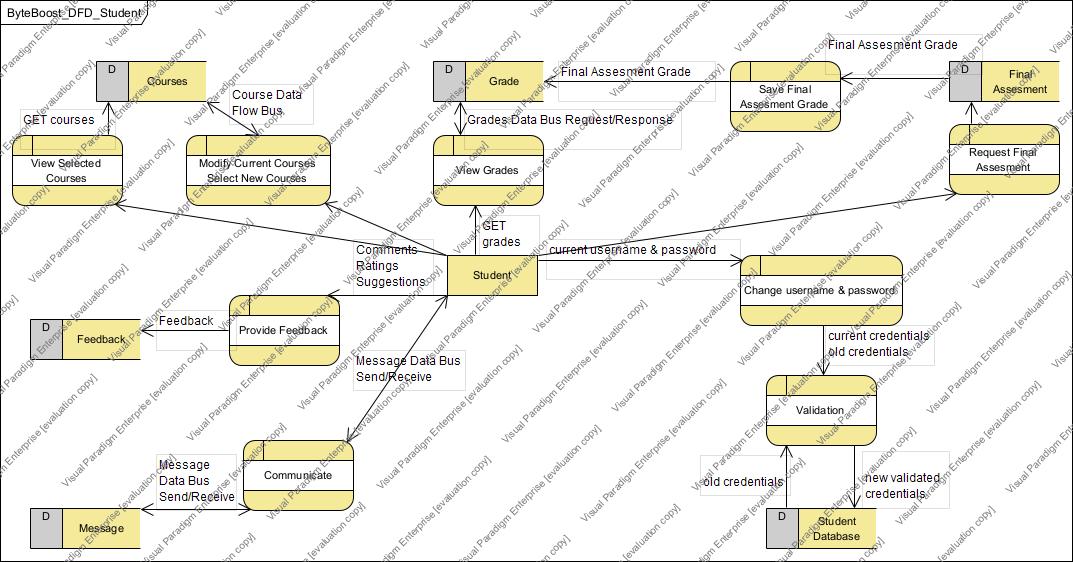
**A screenshot of a computer program

Description automatically generated**

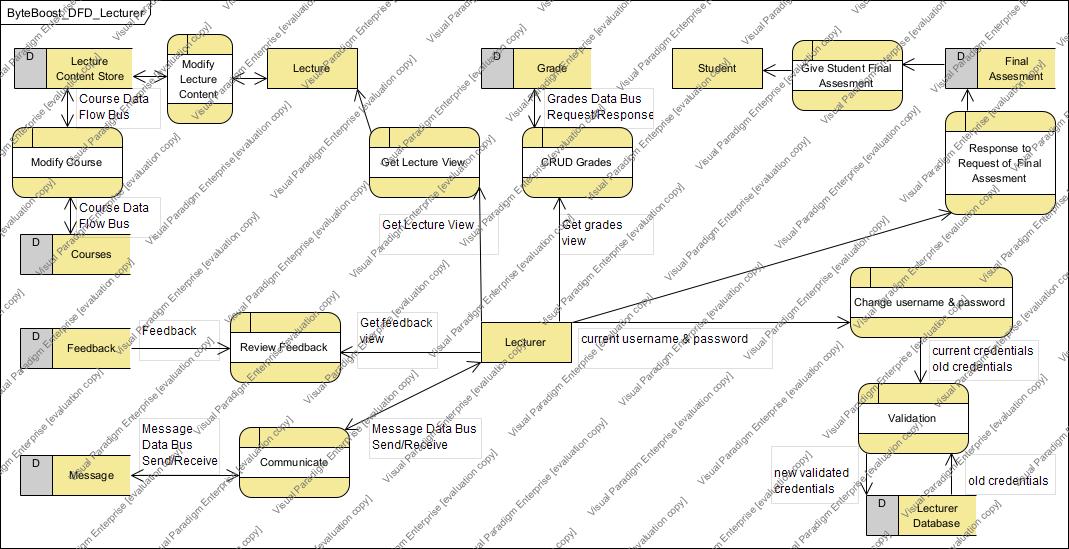
**A screenshot of a computer program

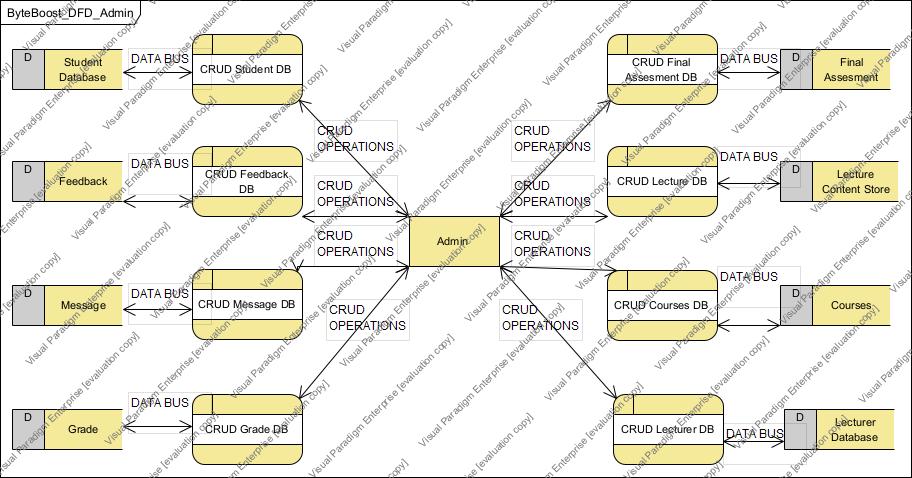
Description automatically generated**

**4.3 DFD - Data Flow Diagram**

**4.3.1 Student DFD**

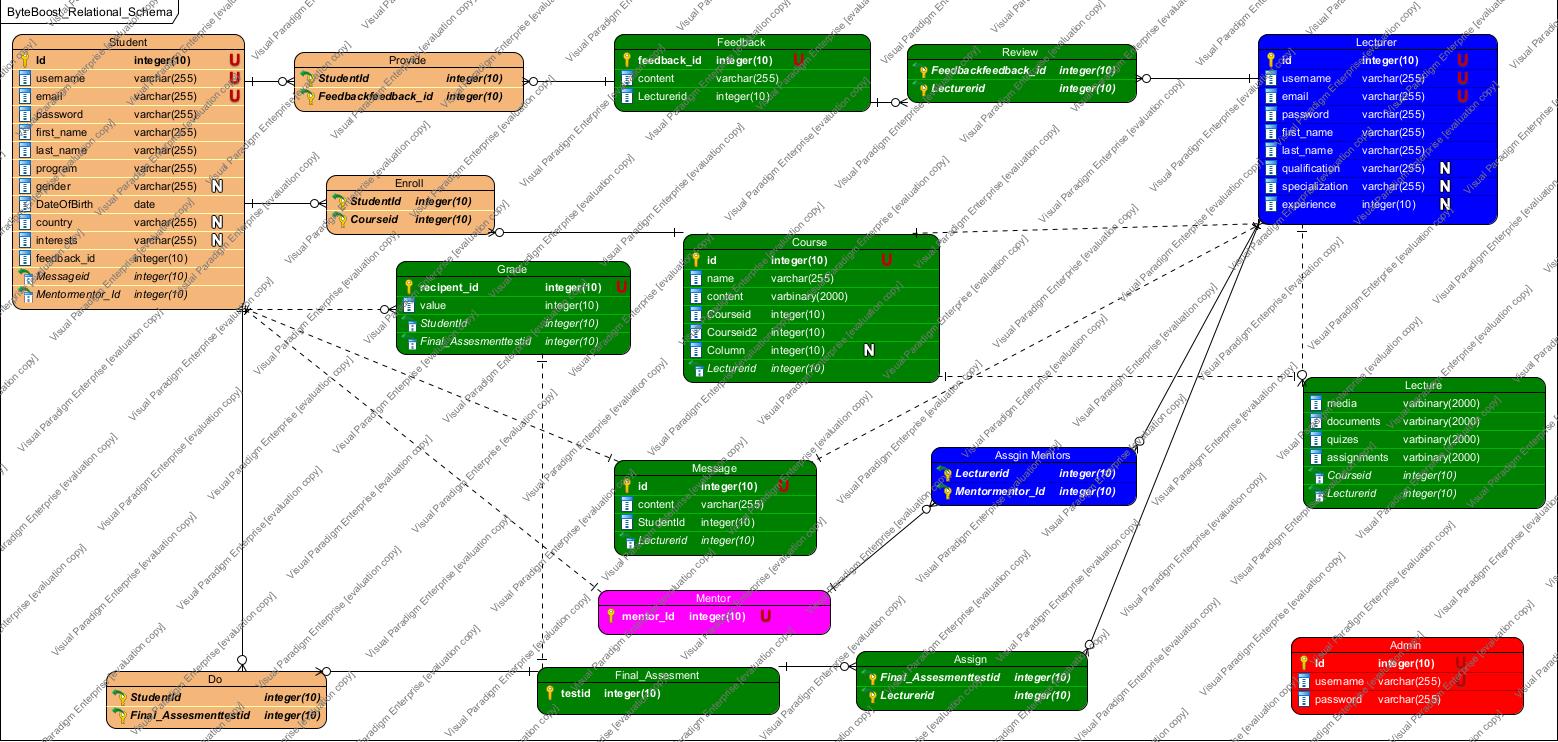
**4.3.2 Lecturer DFD**



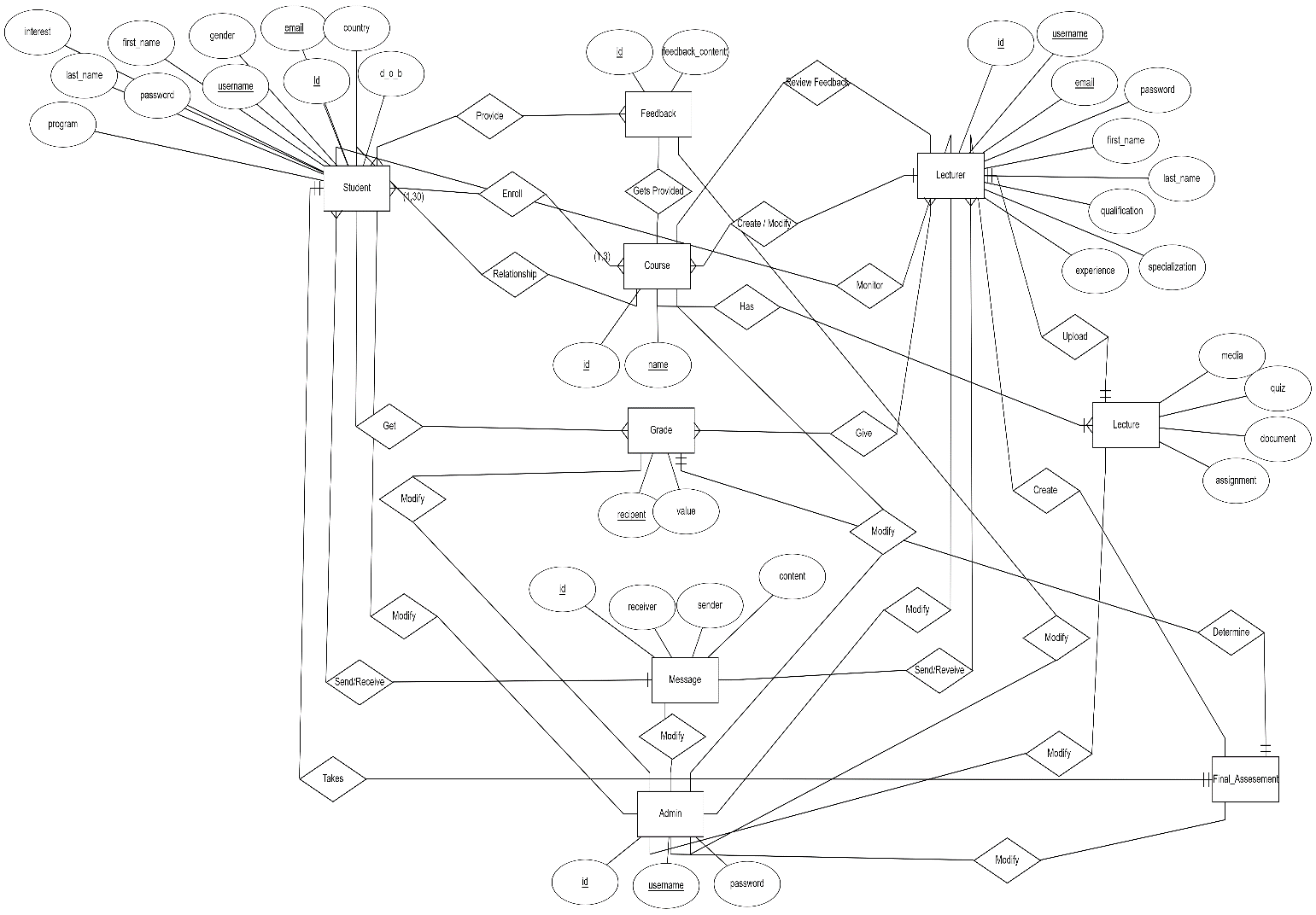
**4.3.3 Admin DFD**

**4.4 Entity Relation**

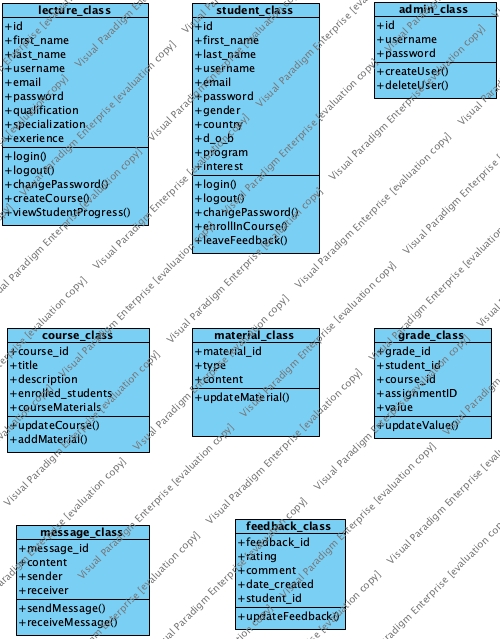
**4.4.1 Relational Schema**

****

**4.4.2 Entity Relation Diagram**

****

**4.5 Structural Diagrams**

**4.5.1 Class Diagram  
**

**4.5.2 Object Diagrams**

**4.5.3 Component Diagrams**

**4.5.4 Deployment Diagram**