CS241 - Assignment 1 Semester 2 2024

Software Design Document: Student Academic Management System for Secondary School in Fiji

Group 8

Name	Student ID
Shamal Prasad	S11219545
Adrian Obadiah	S11198024
Gwen Mar	S11211732
Toni Nam	S11213524

TABLE OF CONTENT

TABLE OF CONTENT.....1
2.0 Introduction & Purpose.....3
2.1 Problem Domain.....3
2.2 Purpose.....3
2.3 Intended Users.....3
2.4 Organizational Context.....4
2.5 Social Significance.....4
2.6 Design Standards.....5
3. System Capabilities and Scope.....7
3.1. Stakeholders.....7
Internal Stakeholders:.....7
External Stakeholders.....8
3.2. Functional Requirements.....9
3.3. Non-Functional Requirements.....11
4. Design Risk Analysis (related to Design).....12
5. Data Modeling: Entity Relationship Diagram.....14
5.1 Registration And Login ERD.....14
5.2 Student Academic System ERD.....16
6. Application Architecture.....17
6.1. Structured Design Artifacts.....17
6.1.1. Structured Chart.....17
6.1.2. System Context Diagram/DFD Level 0 Diagram.....23
6.1.2.1. Primitive DFD diagram (Lowest level DFD) for a major module.....25
Diagram level 1 (Register User).....25
Diagram level 1 (Manage Student Profile).....26
Diagram level 1 (Create Course schedule).....27
Diagram level 1 (Create Student TimeTable).....28
Diagram level 1 (Course Instructor Mark and Comment Assignment).....29
Diagram level 1 (View all Assessment Mark).....30
Diagram level 1 (Create and Send Academic report).....31
Diagram level 1 (Save Student Attendance).....32
Diagram level 1 (View Attendance Record).....33
Diagram level 1 (Create Student Absence Notice to Parents).....34

Diagram level 1 (Post Assignment) 35
Diagram level 1 (Submit Assignment) 36
Diagram level 1 (Upcoming Assignment notice) 37
Diagram level 1 (Generate overall report) 38
Diagram level 1 (User Login) 39
6.2. Object-Oriented Design Artifacts 40
6.2.1. Use-Case Diagram and description 40
6.2.2. Domain Model Class diagram 47
6.2.3. A major Sequence Diagram (s) 48
6.2.4. Detailed/Design Class Diagram 51
7. User Interfaces 52
7.1 Admin Panel 52
7.2 Student User Interface 54
7.3 Teacher User Interface 57
8. Implementation Plan 58
9. Definitions and Acronyms 58
10. References 59

2.0 Introduction & Purpose

This document describes the design and development of an Online Student Academic Management System to address challenges faced by a Fiji secondary school in efficient and effective management of academic information and school-related activities and to satisfy the requirements of the learning environment and its users. The purpose of this document is to provide a comprehensive overview of the proposed software solutions, their objectives, and the design principles adopted in their development.

2.1 Problem Domain

Fiji secondary schools currently employ a manual approach to keep academic records, track academic performance, and monitor school-related activities. This approach has led to inaccurate academic performance tracking, communication failures, and record-keeping problems. The lack of a centralized system limits teachers' and parents' ability to monitor their child's or student's academic performance, create a well-organized system for scheduling classes and assignments, and ensure that educational standards are satisfied. This project seeks to close this gap by creating and deploying an online platform that makes academic information more accessible and improves user experience.

2.2 Purpose

The purpose of this design document is to provide a framework for developing an effective student academic management system. It outlines the architectural design, functional requirements, data management techniques, and user interface. This document is a reference for the development team, future stakeholders, and maintainers to ensure a consistent understanding of the system's scope and structure.

2.3 Intended Users

The users of the proposed system are:

a. **Students:** look to access their individualized schedules, view academic progress, track attendance, submit assignments, and correspond with teachers.
b. **Teachers:** access and modify student academic records, manage their designated courses and timetables, track students' attendance, post assignments, upload resources, and generate relevant reports.

c. **Administrator (Computer Science Teacher):** manages student profiles, and oversees student data, instructor assignments, class scheduling, and system configurations.
d. **Parents:** Track and monitor student's academic performance records and correspond with teachers

2.4 Organizational Context

A secondary school in Fiji is a medium-sized institution serving a diverse population of students, staff, and parents. This context emphasizes the need for a user-friendly interface with a variety of user configurations and preferences. To manage everyday school activities and increased student data, the system must handle vast amounts of data efficiently and effectively.

2.5 Social Significance

The design of the Student Academic Management System has many possessive social impacts such as:

1. **Improved Communication and Collaboration:** The system enables better communication between students, teachers, parents, and administrators. It cultivates a more engaged and informed school community.
2. **Enhanced access to education:** The system provides easier access to academic records, class schedules, and educational resources. This helps teachers in decision-making and encourages student progress.
3. **Digital Literacy**: Implementing an online system allows students, teachers, and parents to get more familiar with technology, encouraging digital literacy skills, which are becoming increasingly important in today's digital world.
4. **Individualized learning experiences:** Teachers can pinpoint each student's unique talents and weaknesses by examining their academic data. This allows teachers the ability to help their students effectively and for students to receive the help that they need.
5. **Increased productivity:** The system optimizes productivity by allowing administrators and teachers to spend less time maintaining records and more time focused on enhancing the student's learning experience.
6. **Inclusivity:** A well-designed system would cater to students with special needs by integrating features such as an easy-to-navigate interface and accessibility options.

7. **Sustainability:** Having an online academic system would minimize the environmental impacts that derive from using immense amounts of paper to store records and forms.
8. **Privacy and Security:** The system helps uphold data privacy standards by ensuring that student and family information is securely stored and handled by authorized users.

2.6 Design Standards

1. **User Interface [UI] Standards:**
    - Ensure the UI of the secondary school management system is responsive and can adapt to different screen sizes like mobile phones, tablets, and desktops for optimal user experience.
    - For consistency ensure that we maintain a consistent visual language throughout the application, this may include fonts, icons color patterns, and layout structure or design
    - Follow the standard WCAG guidelines (Web Content Accessibility Guidelines) to ensure the system is accessible to any secondary school across Fiji (including rural areas or schools that don't have regular access to high-speed internet).

2. **Database Design standards:**
    - Ensure that the database is normalized to the highest possible form (3NF, 4NF, or higher) to minimize anomalies, and data redundancy and improve data integrity.
    - The database should have appropriate indexing on frequently queried columns to optimize database performance. Analyzing query performance and adjusting indexing should also be taken into account.
    - It is essential that Secondary schools implement a backup and recovery plan for the school database system to protect against data loss
    - Regular testing of the backup and recovery procedures to ensure they are being stored correctly and efficiently.

3. **Coding standards:**
    - Maintain a standard code style throughout (Adhere to consistency). Observe recognized coding standards to improve and enhance readability and maintainability.
    - Make use of descriptive functions and variable names. Ensure that the names given to variables and functions have appropriately described their intended use for easier coding understanding.
    - Implement error handling by using try-except blocks to handle potential exceptions, provide informative error messages, and avoid common security vulnerabilities like SQL injections.

4. **Performance Standards:**
    - Optimize Database queries by utilizing database-specific optimization strategies and use proper index methods to avoid pointless data retrievals
    - Ensure storing of frequently used data in the memory to minimize database loads and enhance response times (usually cache data that is accessed often).
    - Reduce network traffic by avoiding pointless network requests and by using effective data transfer techniques.
    - Identify performance problems or bottlenecks and optimize code accordingly using necessary profiling tools
    - Ensure sufficient network infrastructure and hardware to support the system functionality.

5. **Documentation standards:**
    - Create comprehensive user guides to keep track of every feature and functionality of the system, provide concise and unambiguous documentation
    - Compose thorough documentation for developers: For developers, document API usage, algorithms, and code structure.
    - Keep your documentation current: As the system develops, update the documentation frequently.

6. **Security standards:**
    - The system should uphold data privacy standards, including access control and data encryption.
    - Implement strong encryption algorithms to protect sensitive data when in the transfer process or at rest. Also, use encryption protocols (HTTP) to secure communication between client and server.
    - Authentication and authorization should be implemented in multi-factor authentication(MFA) for enhanced security and role-based access control (RBAC) to grant appropriate access controls to different user roles. Update passwords and security policies regularly.
    - Ensure data privacy regulations are carried out correctly and responsibly for user data, provide clear privacy policies, and obtain user consent for data collection and usage.

7. **Testing standards**:
    - Thorough quality, unit, integration, and user testing are carried out to ensure system functionalities are free from errors and meet the client's needs.

3. System Capabilities and Scope

3.1. Stakeholders

Listed below are the stakeholders we have taken into account for our Student Academic System that we wish to implement in Fiji, We have broken it down into two categories:

Internal Stakeholders:

1. **Students**

   Have access to their assignments, class schedules, grades, and attendance records. They also may have access to educational materials like online textbooks, study materials, or supplementary resources.

2. **Teachers**

   Have access to enter assignments, attendance, and grades. They can also create class schedules, interact with parents and students, and access their respective students' records.

3. **School Administrators**

   The administrators are among the main stakeholders who will interact directly with the system. They will have the ability to design and edit lesson plans, manage user accounts, provide overall reports, and supervise the system's general operations to ensure they run smoothly and safely.

4. **Support staff (janitors /securities/technicians/ etc):** Should be able to log into the system and also provide technical assistance if needed, or help with maintenance.

External Stakeholders

i. **Parents**

   Parents can log into the system and monitor their child's progress, attendance, and behavior. They may also be able to interact with the system through inquiries or online messaging with administrators and teachers about concerns or questions about their children's education.

ii. **Alumni**

   They interact with the system through resource access, networking, and donations. They can also indirectly influence the system by offering assistance, mentorship, and advocating for the school.

iii. **Government**

   Through program evaluation, money, and tracking performance, the government engages with the secondary school system in Fiji by giving funds establishing guidelines, controlling activities, and enacting reforms. The government can use the system to monitor student performance and achievement data and also allocate funding to schools based on performance and other criteria.

iv. **Donors and community**

   The community and donors both contribute significantly to a school management system's success. Donors can provide resources and financial assistance, and the community can offer support and feedback on whether the system is a hit or miss.

3.2. Functional Requirements.

- **User Registration and Authentication**
  - Allow administrators, teachers, students, and parents to create accounts.
  - Implement role-based access control (admin, teacher, student, parent).
  - Provide login functionality with password recovery options.
  - Implement authentication using email or mobile number.
- **Student Information Management**
  - Store and manage detailed student records (name, date of birth, address, emergency contact information, etc.).
  - Allow modification of student profiles by authorized personnel (admin/teacher).
  - Maintain academic history (courses taken, grades, attendance).
  - Track disciplinary records or other notable incidents.
- **Class scheduling (timetable)**
  - Enable admin/teachers to create and manage class schedules.
  - Allow students to view their class schedules online.
  - Handle conflicts in scheduling for students and teachers.
- **Assignments and homework**
  - Allow teachers to post homework and assignments online.
  - Notify students of upcoming deadlines for assignments.
  - Allow students to submit assignments through the system.
  - Enable teachers to grade and provide feedback on assignments.
- **Attendance tracking.**
  - Allow teacher to ark daily attendance for each class
  - Generate reports for students (monthly/semester-wise)
  - Notify parents via email/SMS when a student is absent (BEST PART)
- **Marks and Grade Mgt.**
  - Allow teachers to enter and update marks for assessments, quizzes, and exams.
  - Automatically calculate final grades based on predefined grading criteria.
  - Allow students and parents to view grades and progress reports.
- **Admin Panel**
  - Provide full control to the admin to manage users, roles, and permissions.
  - Allow admins to view, edit, and delete records as needed.

- Provide analytics and reports for attendance, grades, and system usage.

- **Parental Access and Notifications**
  - Allow parents to view their child's academic progress, attendance, and grades.
  - Enable automated notifications for parents about attendance, grades, or important announcements.
  - Provide a messaging system for parents to communicate with teachers.
- **Reports and Analytics**
  - Generate student-specific and class-specific performance reports.
  - Create grade distribution reports for analysis.
  - Provide attendance trend reports (individual and overall class trends).
  - Implement export options (CSV, Excel, PDF) for reports.
- **Homework and Assignment Tracking**
  - Notify students about upcoming assignments and due dates.
  - Allow teachers to grade homework and provide feedback.
  - Maintain a history of assignments for reference.
- **Data Privacy and Security**
  - Ensure all sensitive student and school data is encrypted.
  - Implement audit logs to track user actions (admin actions, student/teacher updates).
  - Implement backup procedures to protect against data loss.
- **Mobile Accessibility**
  - Make the system accessible via mobile devices for students, parents, and teachers.
  - Implement responsive design for easy mobile usage.
- **Scalability and Performance**
  - Ensure the system can handle a growing number of students, classes, and data.
  - Optimize the system for performance to avoid delays, especially during peak times (exam results, registration periods).
- **Integration with Third-party Services**
  - Allow integration with email and SMS services for communication.
  - Integrate with existing school systems if required (library, transportation, etc.).
- **User Experience (UX)**
  - Ensure the system is easy to use with a clean and intuitive interface.
  - Provide a help section or user guide for all roles.

3.3. Non-Functional Requirements

- **Performance**
  - The system should be able to respond quickly to user's input and queries, especially during extremely busy times like registration periods or exam results.
  - The system should also be able to handle very large amounts of data like concurrent users and transactions without any performance interruptions.
- **Security**
  - **Data confidentiality and Integrity:** Students Sensitive data and school data should be protected from unauthorized access
  - Data stored in the system's database should be private, confidential, and protected from accidental or intentional modifications to maintain its accuracy and reliability.
  - **Data availability and access control:** Authorized users should be able to access the data when needed, ensuring that information is available on-site when required.
  - Access controls should also be strongly implemented to prevent unauthorized access to sensitive data (restrictions even to authorized personnel for very sensitive data like student passwords for accounts).
- **Reliability and availability**
  - The system should be highly available with little downtime to give users continuous access.
  - The system must also be fault tolerant and be able to bounce back from errors and carry on with minimal impact on the users.
  - The Systems should be backed up to avoid data loss or corruption (be reliable) and guarantee data accuracy and consistency.
- **Usability**
  - To reduce a lot of learning curves and increase user satisfaction, the system should be simple to use and intuitive for users of all skill levels.

- To ensure a good user experience the user interface should be visually appealing for users and consistent.

- **Accessibility**
  - The system must be usable by individuals with impairments to guarantee inclusion and adherence to accessibility guidelines.
- **Maintainability**
  - To make maintenance, updates, enhancements, and future improvements easier, the System should be constructed and designed in a modular fashion.
  - Developers and administrators should have access to clear and thorough documentation of the system. To make the system easy to test and debug, ensure that changes, modifications, or updates do not introduce new bugs and problems.
- **Cost-effectiveness**
  - When evaluating the initial and continuing costs the system should be economical to buy, install, and maintain for Secondary Schools in Fiji.

4. Design Risk Analysis.

Below is a list of risk analyses and their corresponding mitigation strategies for the Secondary High School Management System.

1. **Technical risks**
   - **Integration with Existing System**:
     - Risk: Integration with an existing or current Student Academic Management System for schools in Fiji may introduce data transfer or compatibility issues.
     - Mitigation: Conduct rigorous data mapping, and pilot testing, and implement data validation protocols with a detailed migration plan to ensure smooth integration.
   - **System Complexity**:
     - Risk: The system may be too complex for users, making it difficult to use, develop, test, or maintain.

- Mitigation: Adopt a user-centric design, and modular development approach, and provide user training, along with extensive usability testing.

- **Scalability Issues:**
  - Risk: The system may struggle with an increasing number of users, leading to performance issues.
  - Mitigation: Host the system on cloud services and use load balancers and scalable infrastructure to handle growth effectively.

2. **Project Management (Business) risks.**
   - **Cost Overrun:**
     - Risk: Development or maintenance costs may exceed the initial budget.
     - Mitigation strategy: Develop a detailed project plan with clear milestones, allocate a contingency budget, and monitor costs closely throughout the project.
   - **Implementation Delays**:
     - Risk: Unforeseen technical challenges could cause delays in project timelines.
     - Mitigation: Implement agile project management practices, with regular check-ins and progress reviews to adjust timelines as necessary.
3. **Operational Risks**
   i. Facility Shortage:
     - Risk: Some secondary schools in Fiji may lack the infrastructure to maintain the system.
     - Mitigation: Provide a cloud-based system to reduce the need for local infrastructure, offer offline functionality, or supply technical support to overcome resource limitations.

ii. User Adaptation:

- Risk: Teachers, administrators, or students may struggle to adapt to the new system.
- Mitigation: Provide user training, offer detailed user manuals, and conduct awareness campaigns to ease the transition.

4. Security Risks

i. Data Breaches:

- Risk: Unauthorized access could expose sensitive student information.
- Mitigation: Implement strong encryption, multi-factor authentication (MFA), and regular security audits to safeguard student data.

ii. Unauthorized Access:

- Risk: Weak access control mechanisms may result in data being accessed by unauthorized personnel.
- Mitigation: Use role-based access control (RBAC) and regularly audit permissions to ensure access is granted only to authorized individuals.

5. Compliance and Legal Risks

i. **Non-compliance with data privacy regulations**.

- Risk: The system may not fully comply with data privacy laws.
- Mitigation: Ensure the system adheres to data privacy laws (e.g., student data consent) and clearly communicate data collection and usage policies to all users.

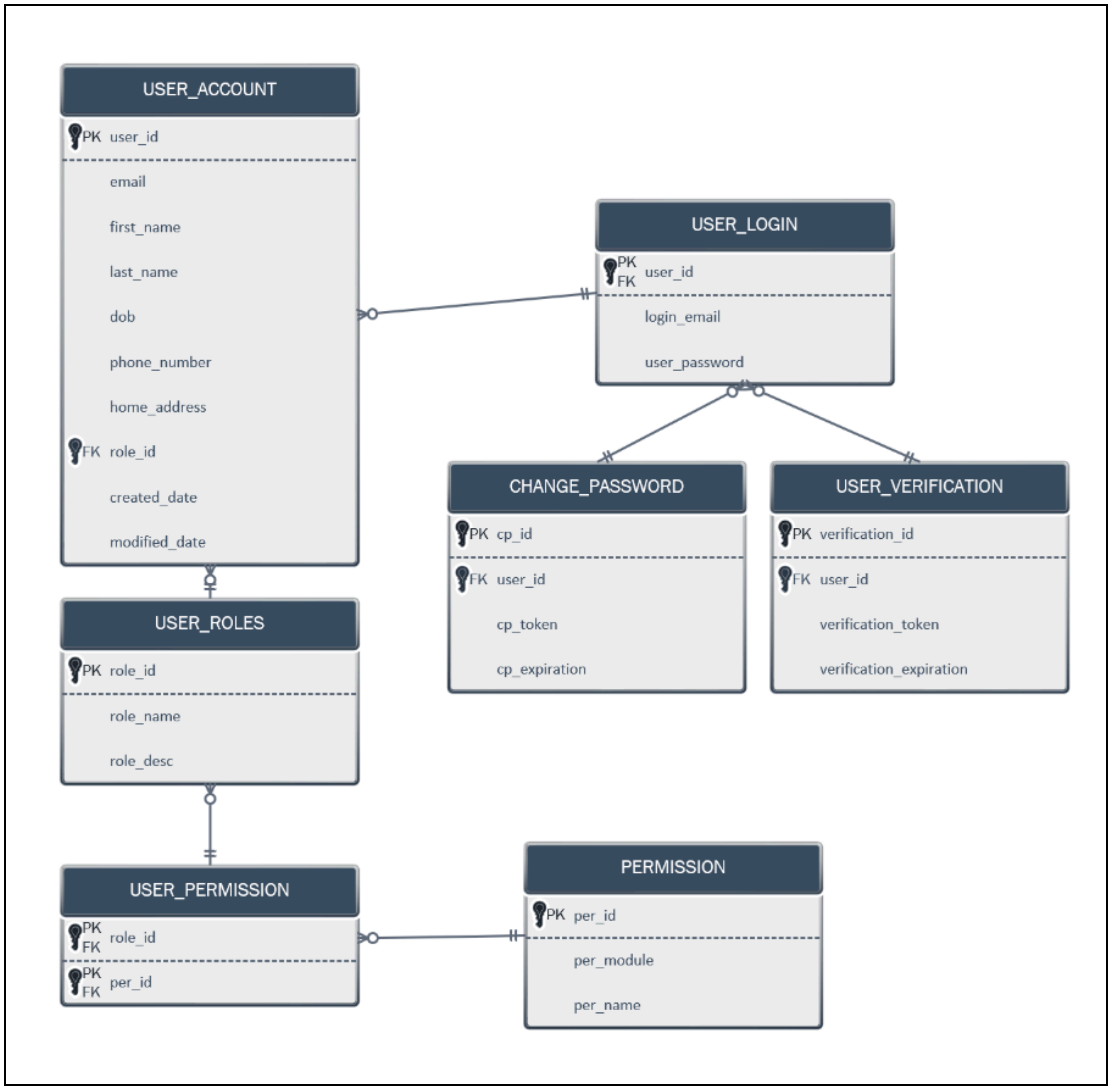
ii. **Failure to meet Accessibility standards**.

- Risk: The system might not meet WCAG guidelines, excluding some users.
- Mitigation: Regularly audit the system for accessibility compliance with the WCAG guidelines and make necessary adjustments to ensure accessibility for all users, including those with disabilities.

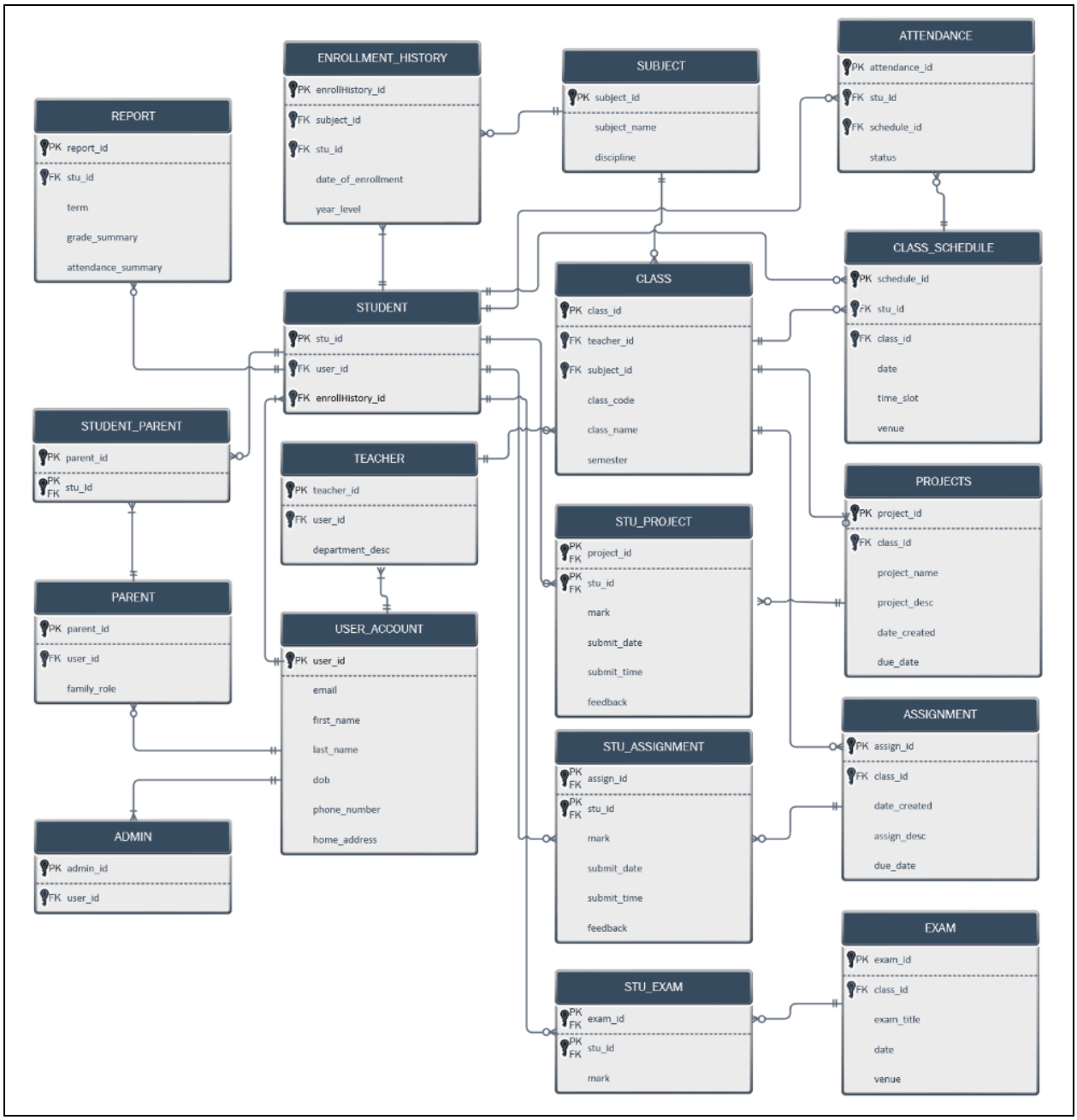
6. **External Risks**
   i. Political or Economic Changes:
      - Risk: Changes in the political or economic environment could affect funding for the system.
      - Mitigation: Engage government and donors early to secure funding commitments and maintain transparency regarding the system's benefits.
   ii. Resistance from Stakeholders:
      - Risk: Some stakeholders, especially those unfamiliar with digital systems, may resist adopting the new system.
      - Mitigation: Provide training sessions and demonstrate the benefits of the system to encourage stakeholder buy-in.

5. Data Modeling: Entity Relationship Diagram

5.1 Registration And Login ERD



5.2 Student Academic System ERD

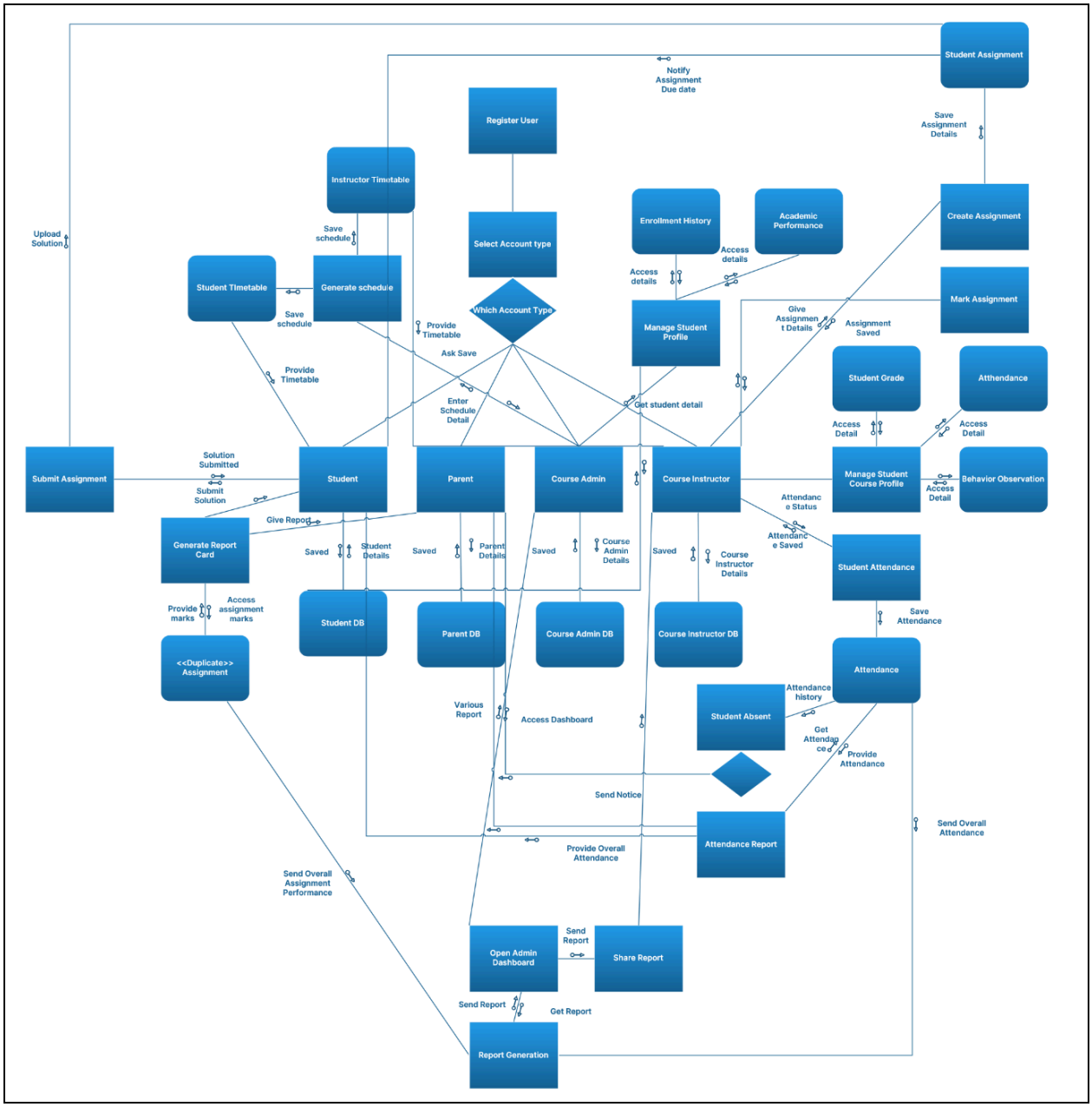


6. Application Architecture

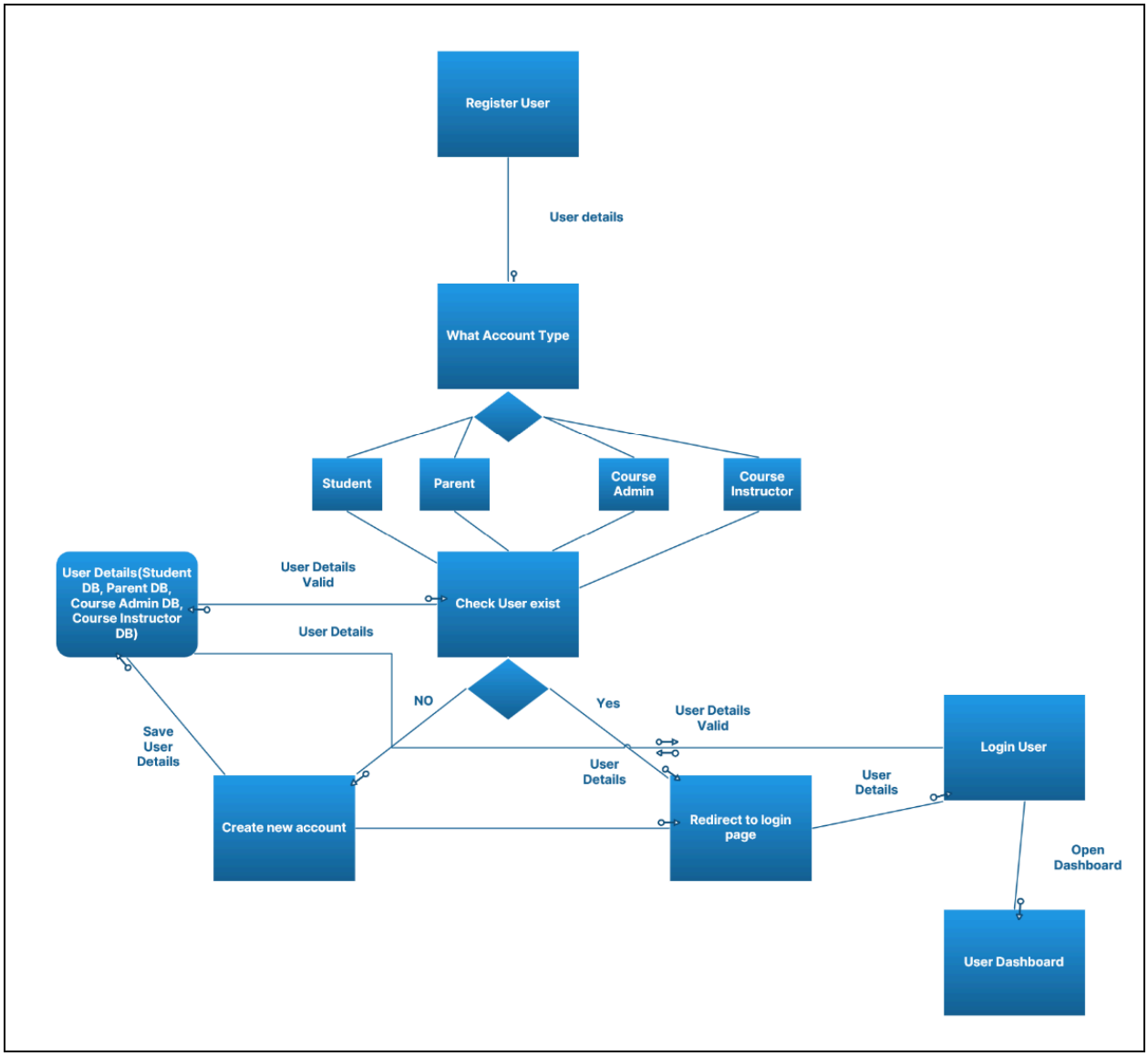
6.1. Structured Design Artifacts

6.1.1. Structured Chart

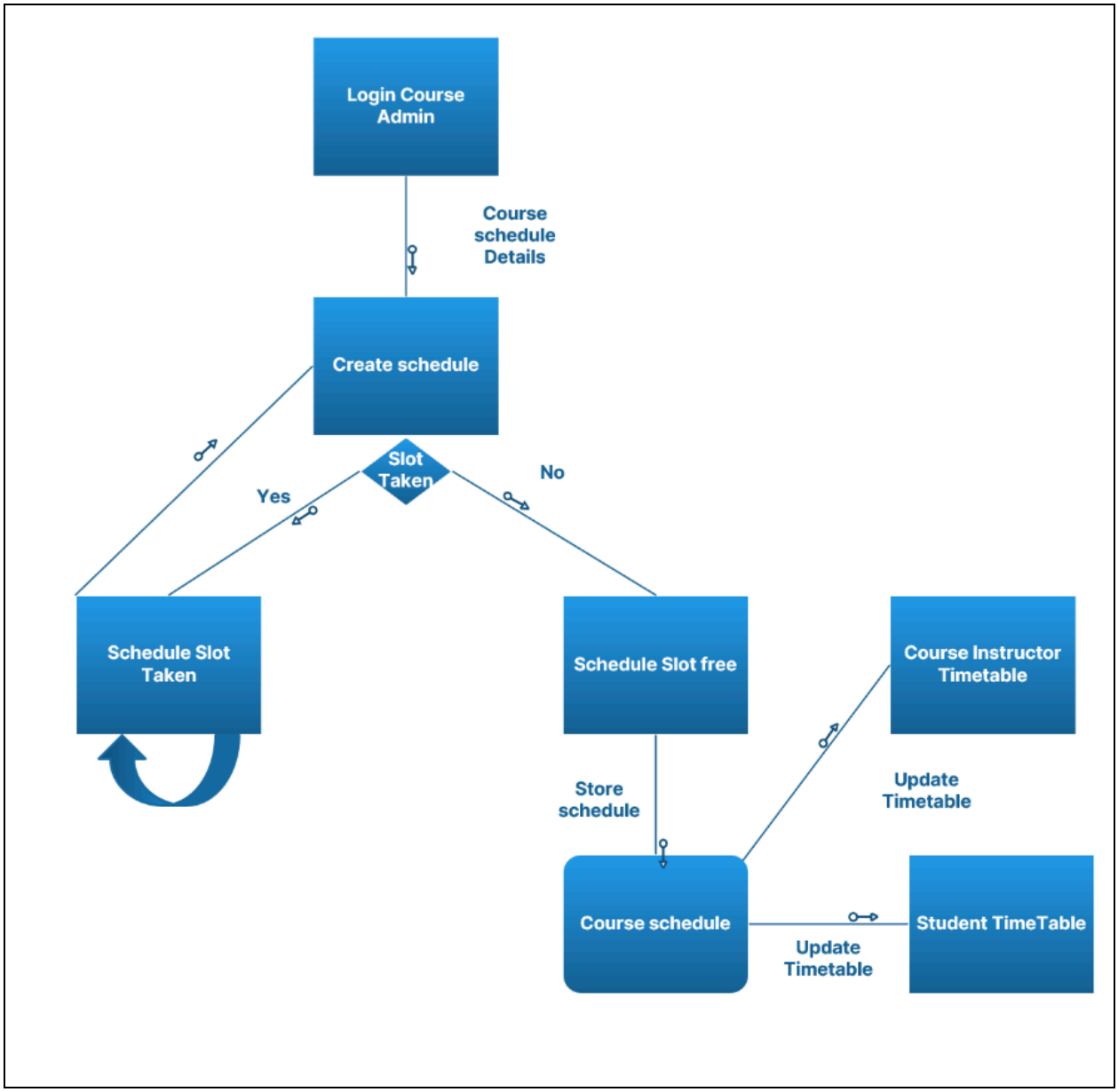
Main Module: Student Academic Management System



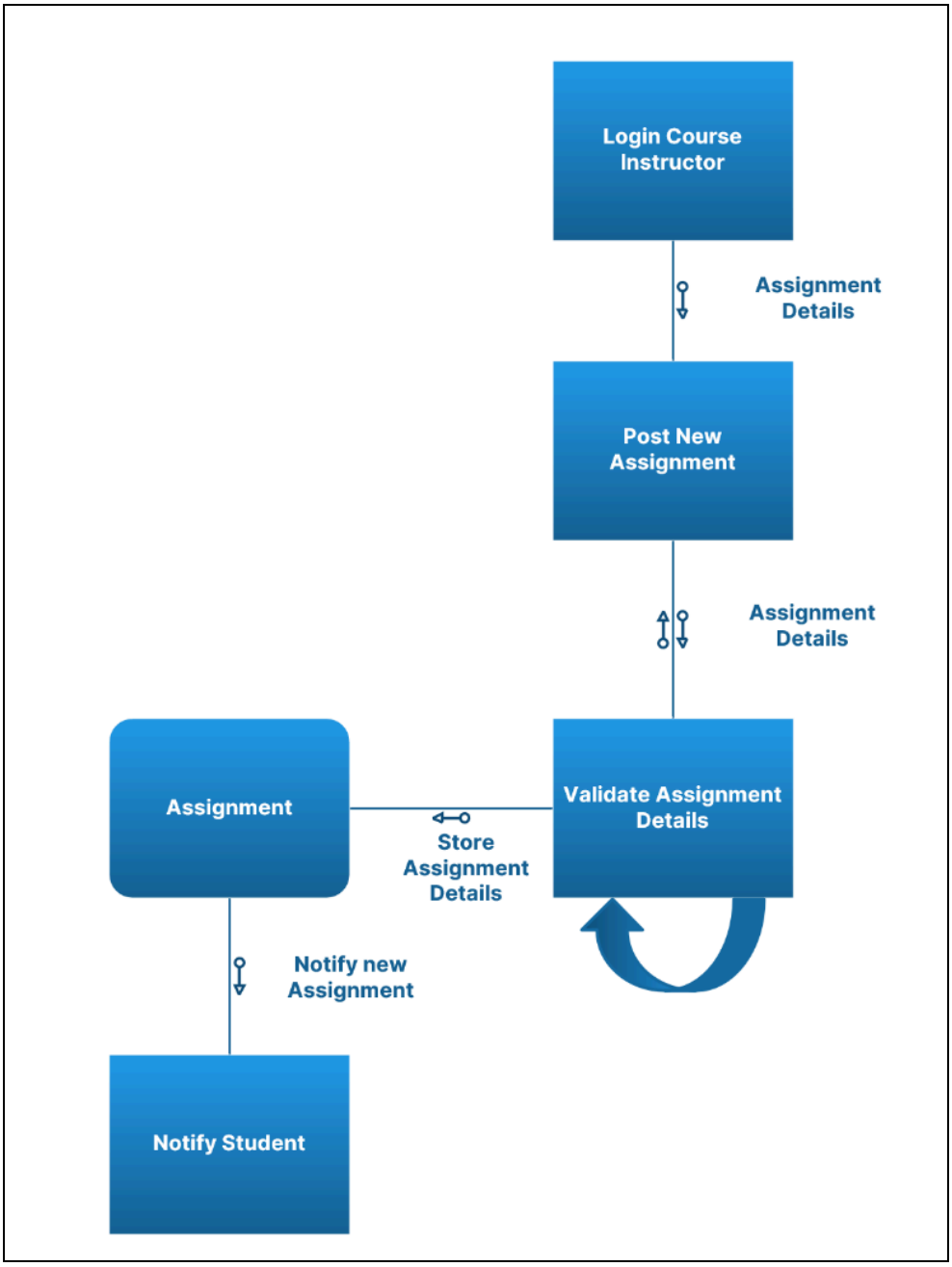
Sub Module: Register and Login



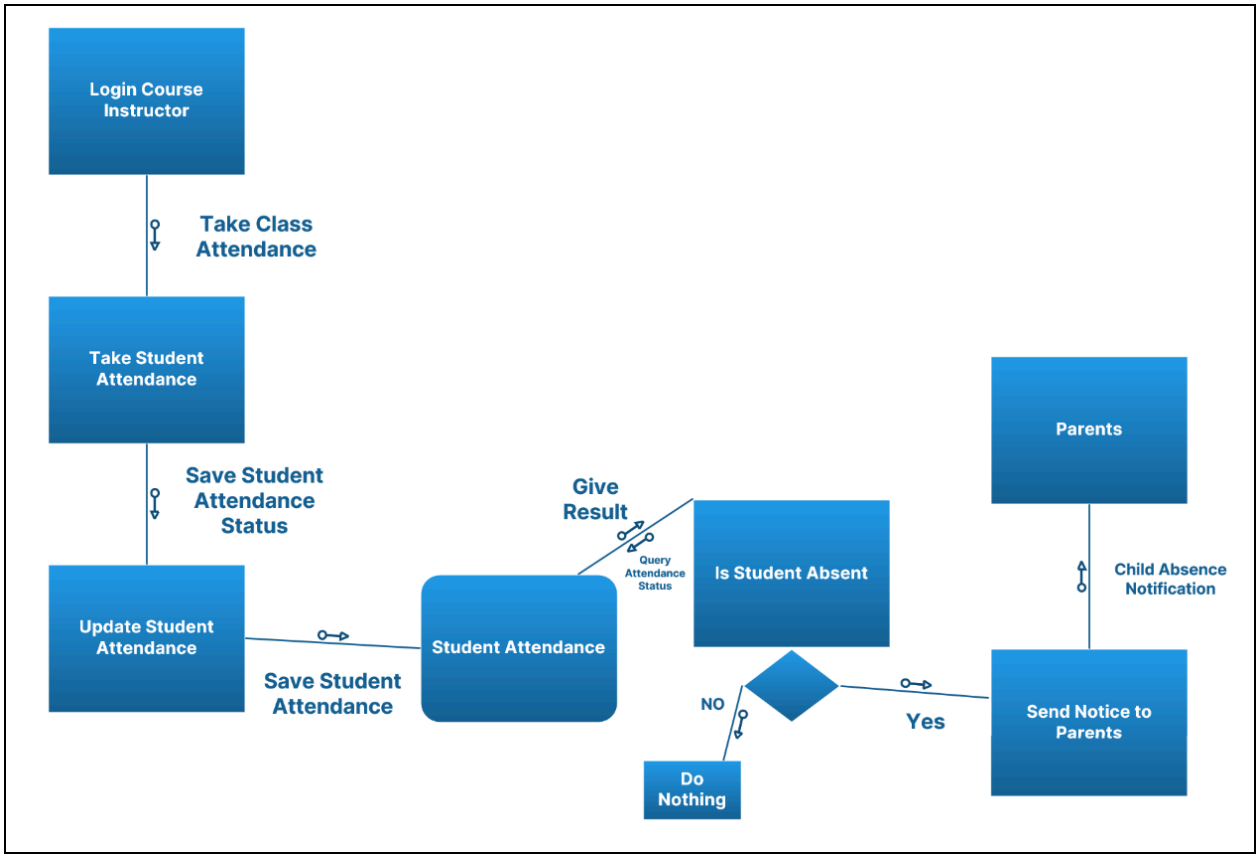
Sub Module: Create Course schedule



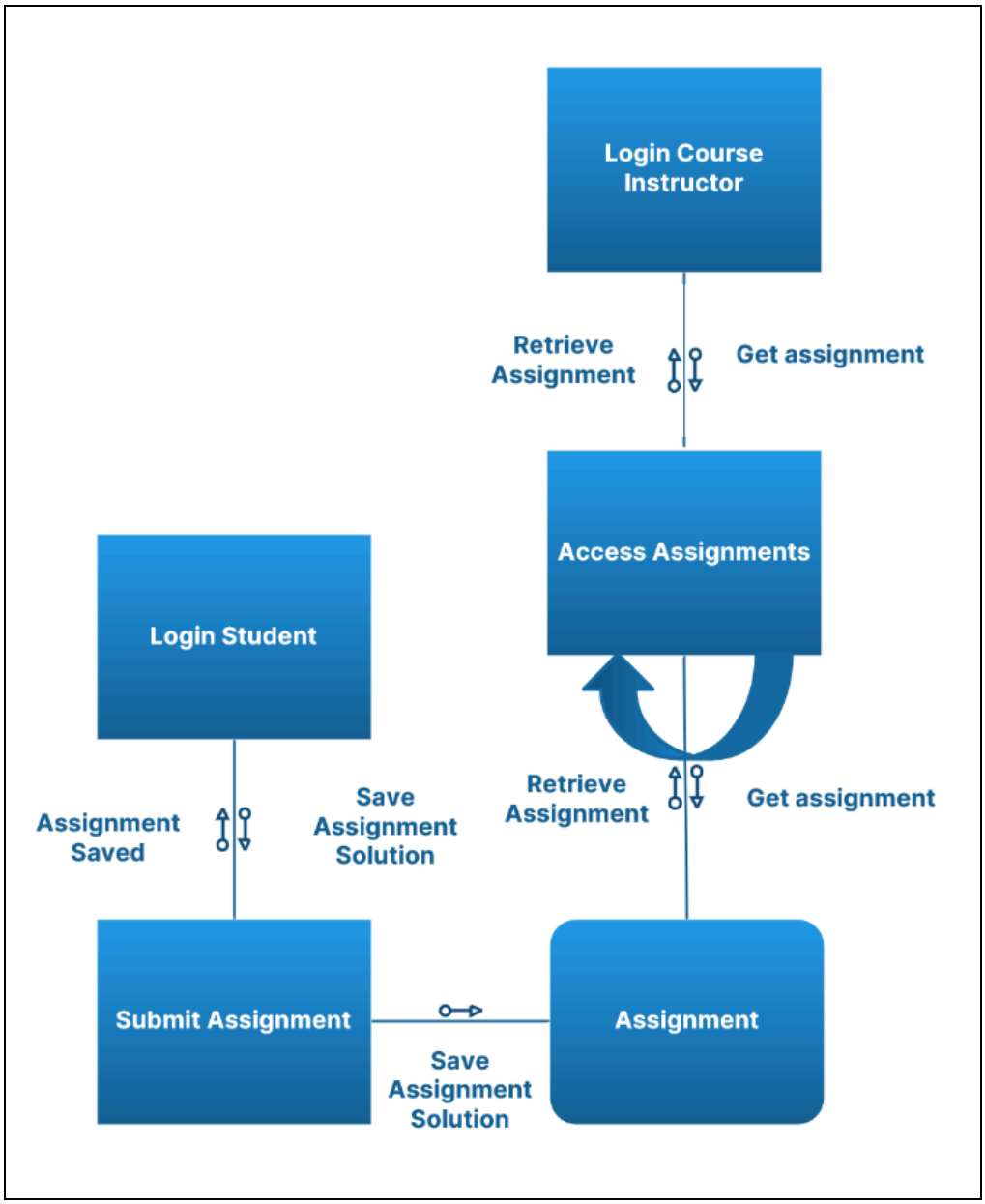
Sub Module: Post new Assignment



Sub Module: Take Student Attendance



Sub Module: Student Submit Assignment



6.1.2. System Context Diagram/DFD Level 0 Diagram

Context Diagram

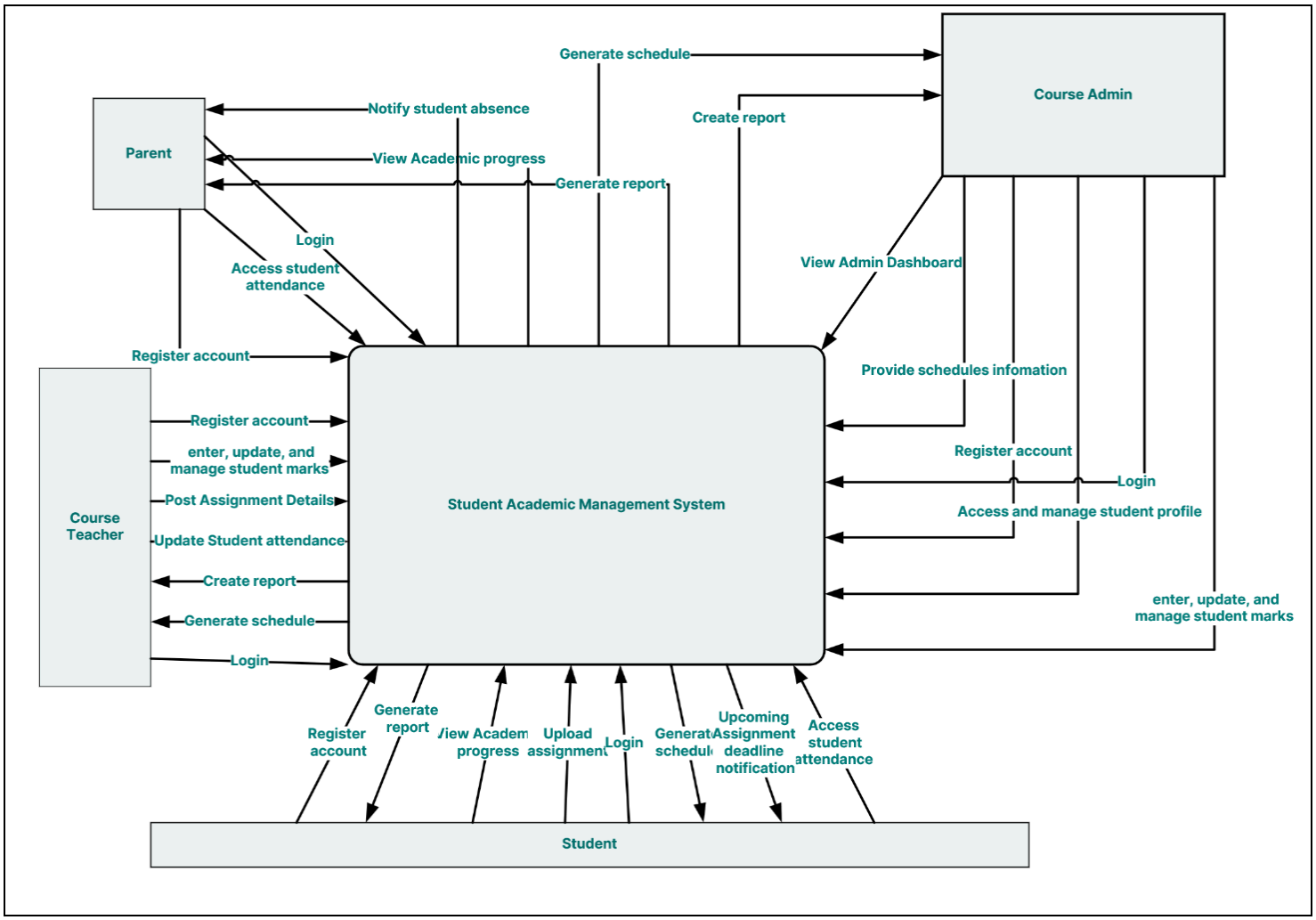
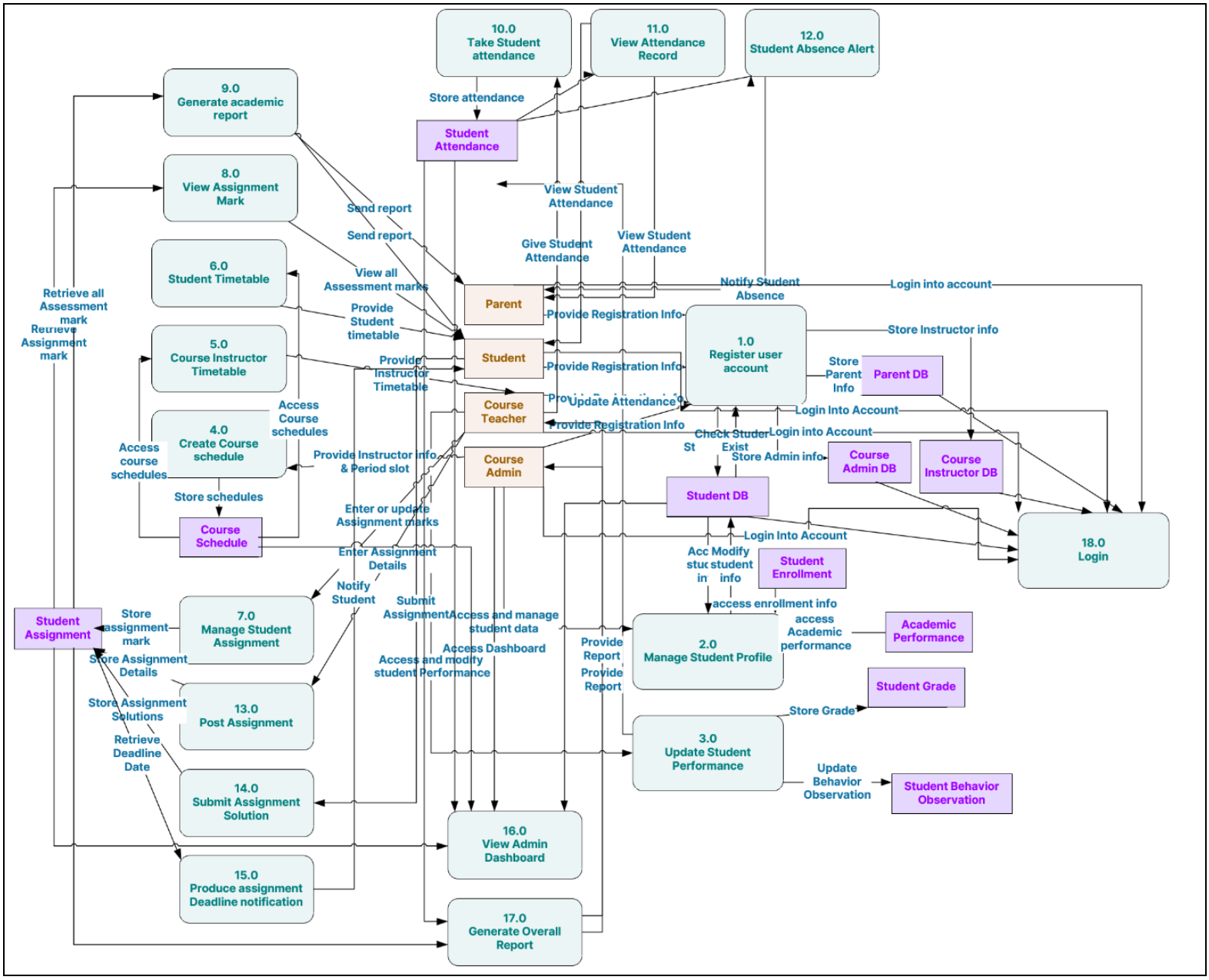


Diagram level 0



6.1.2.1. Primitive DFD diagram (Lowest level DFD) for a major module.

Diagram level 1 (Register User)

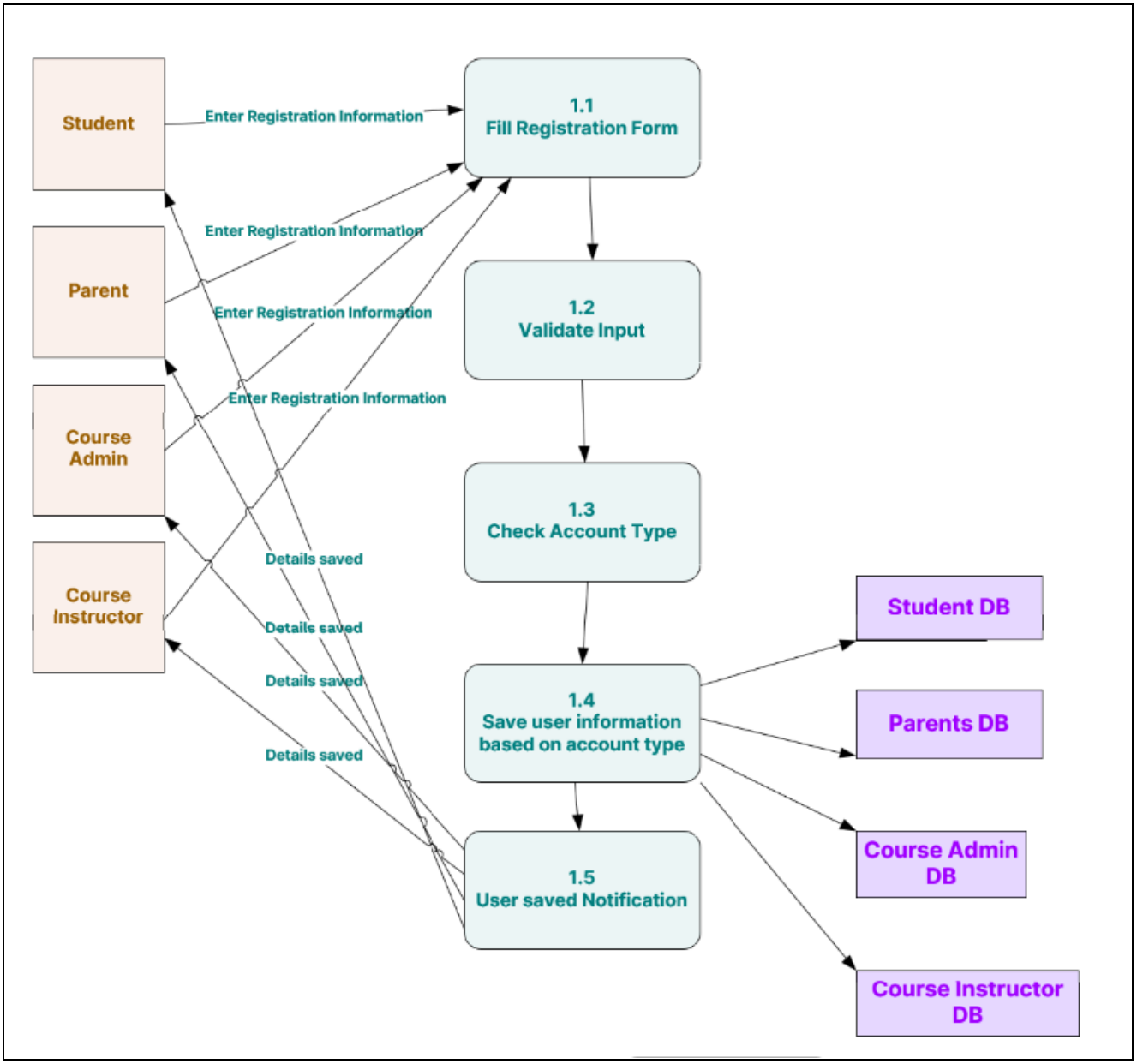


Diagram level 1 (Manage Student Profile)

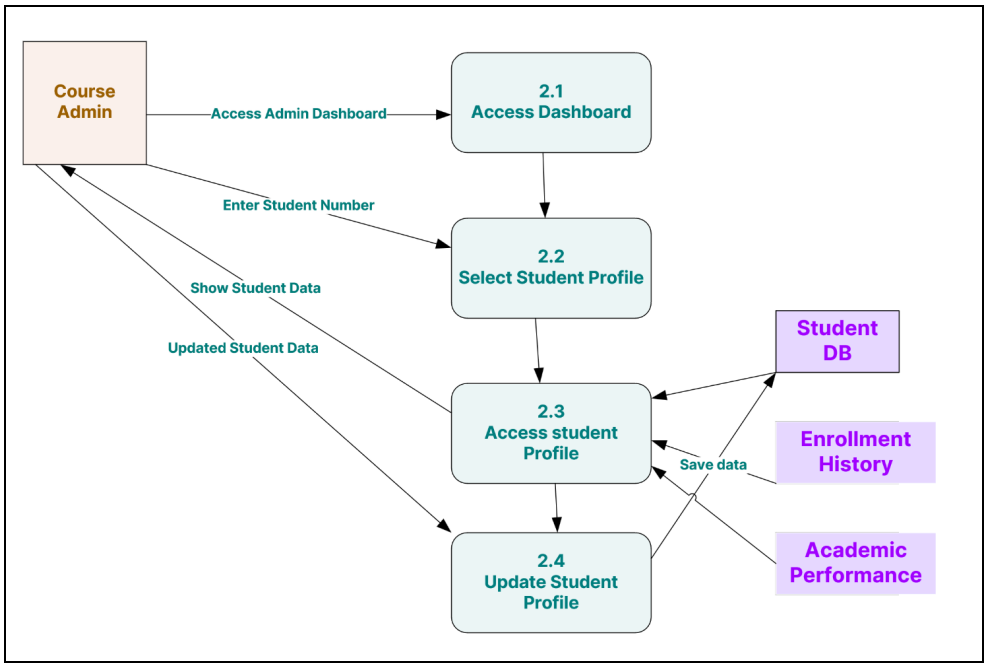


Diagram level 1 (Manage Student Course Profile)

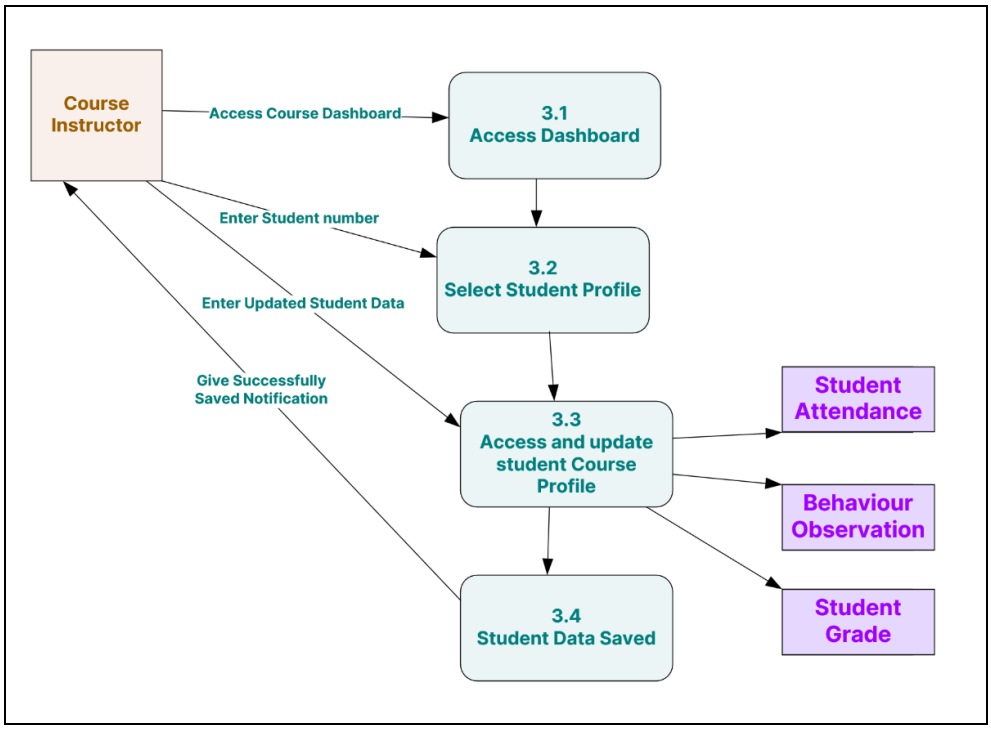


Diagram level 1 (Create Course schedule)

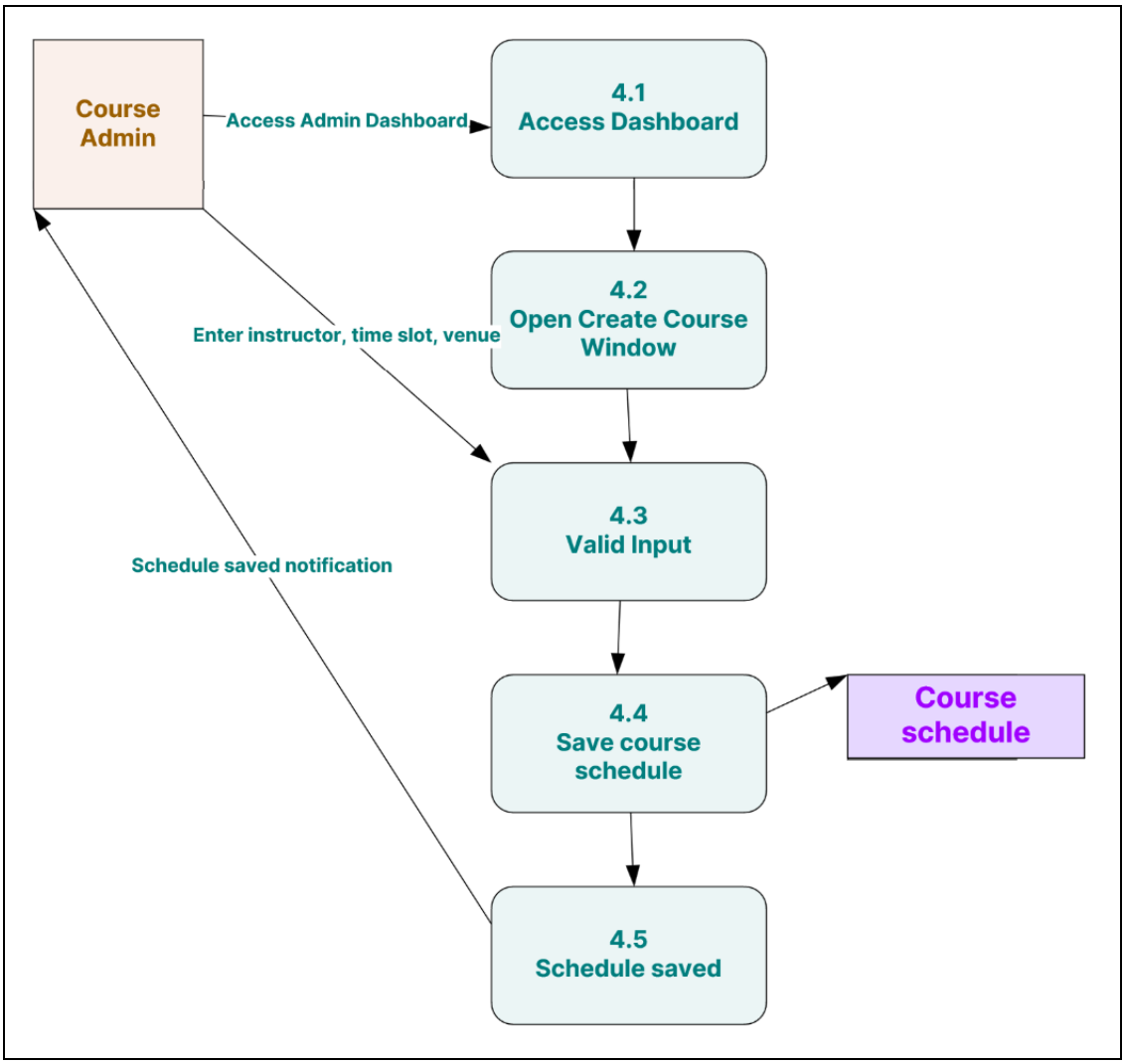


Diagram level 1 (Create Course Instructor TimeTable)

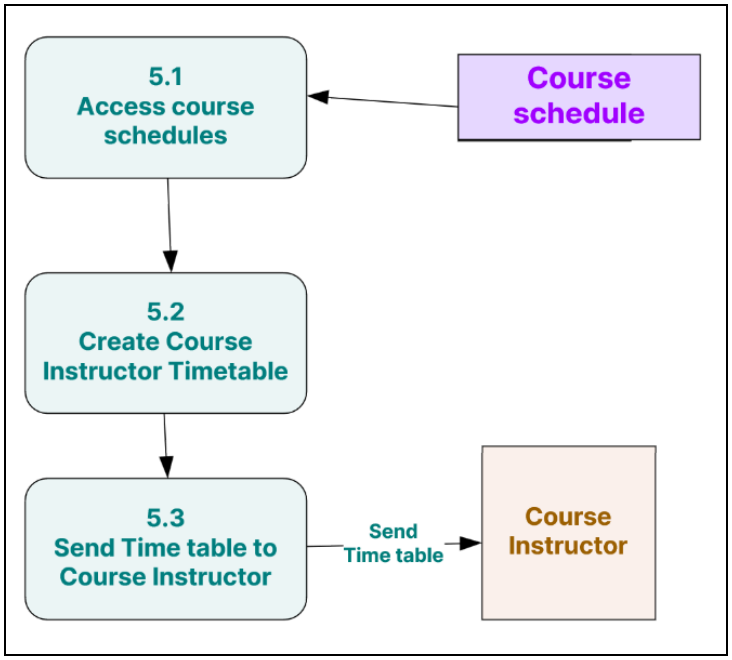


Diagram level 1 (Create Student TimeTable)

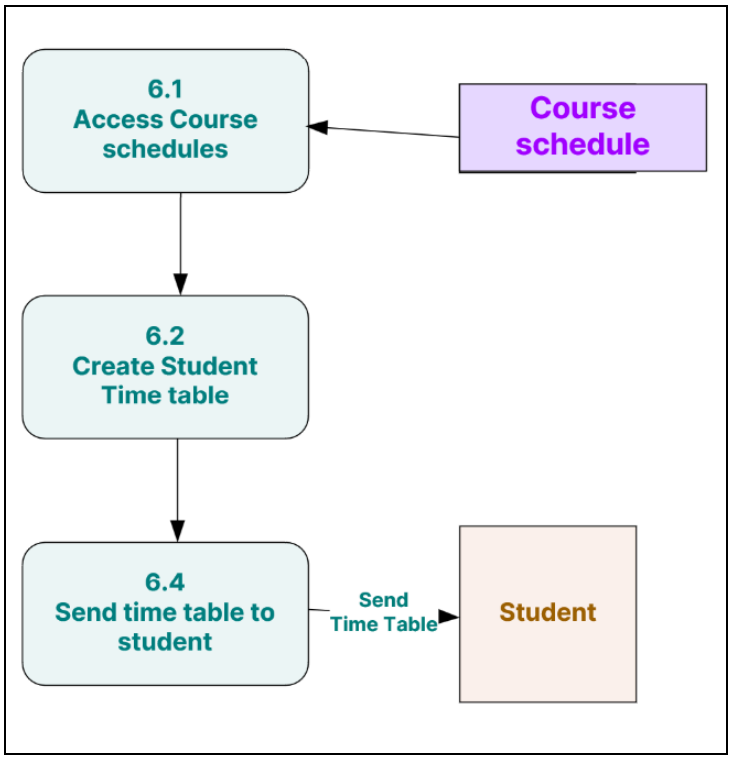


Diagram level 1 (Course Instructor Mark and Comment Assignment)

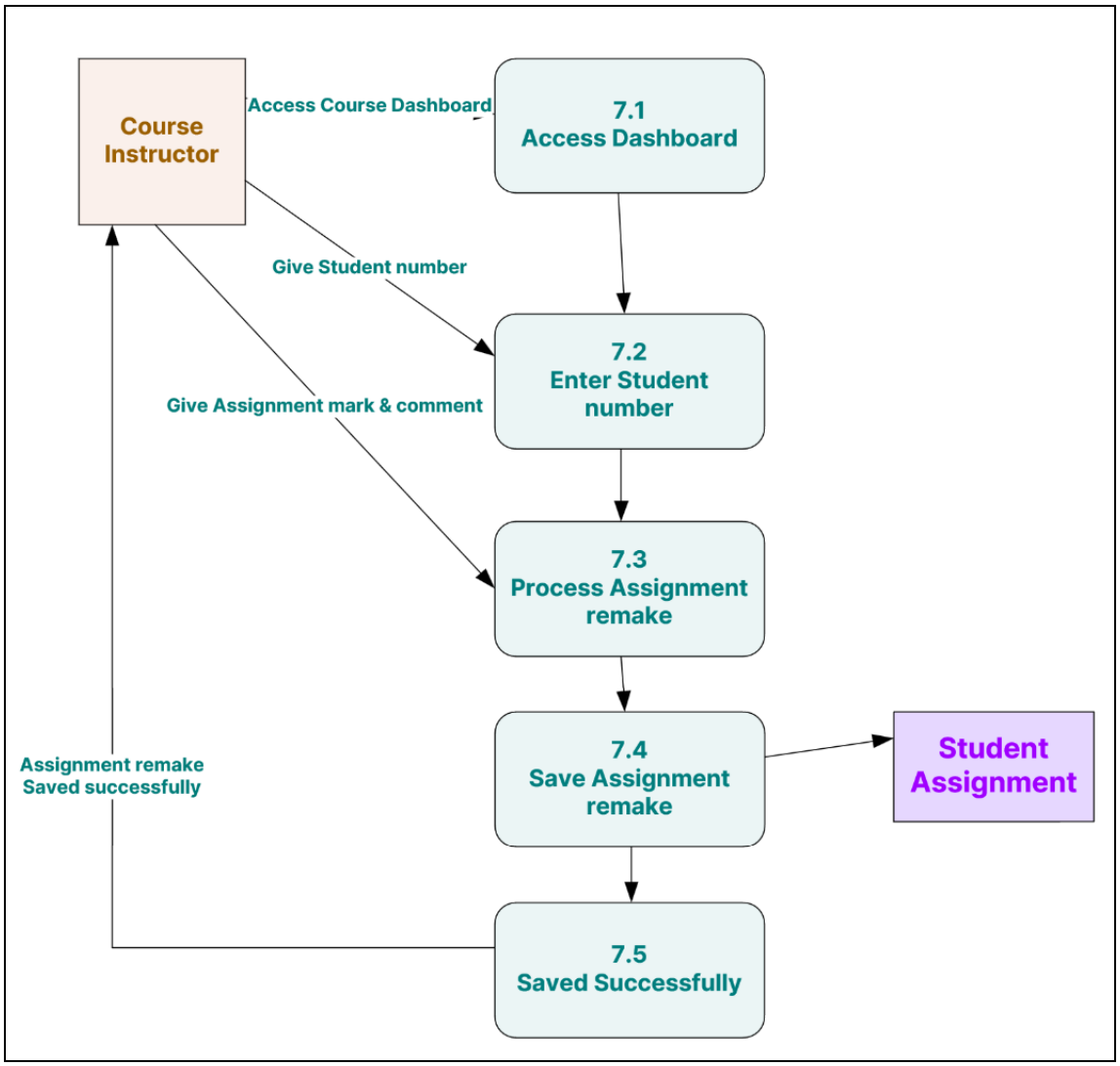


Diagram level 1 (View all Assessment Mark)

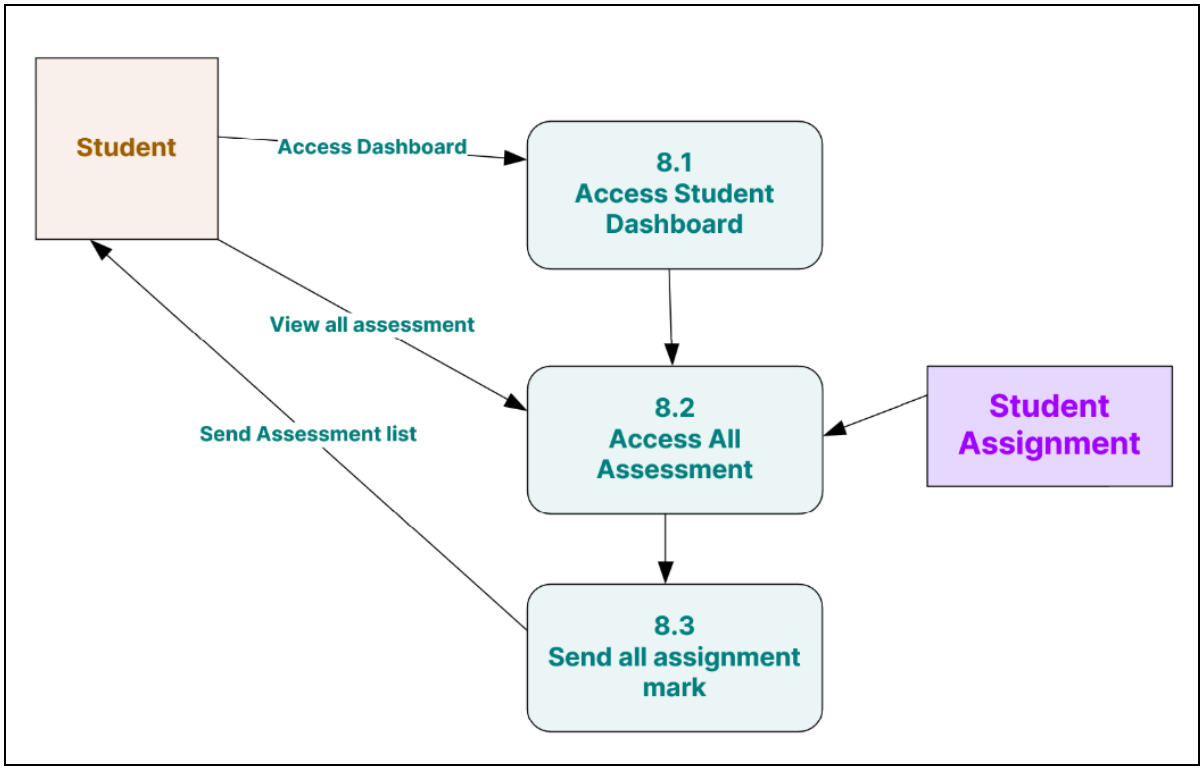


Diagram level 1 (Create and Send Academic report)

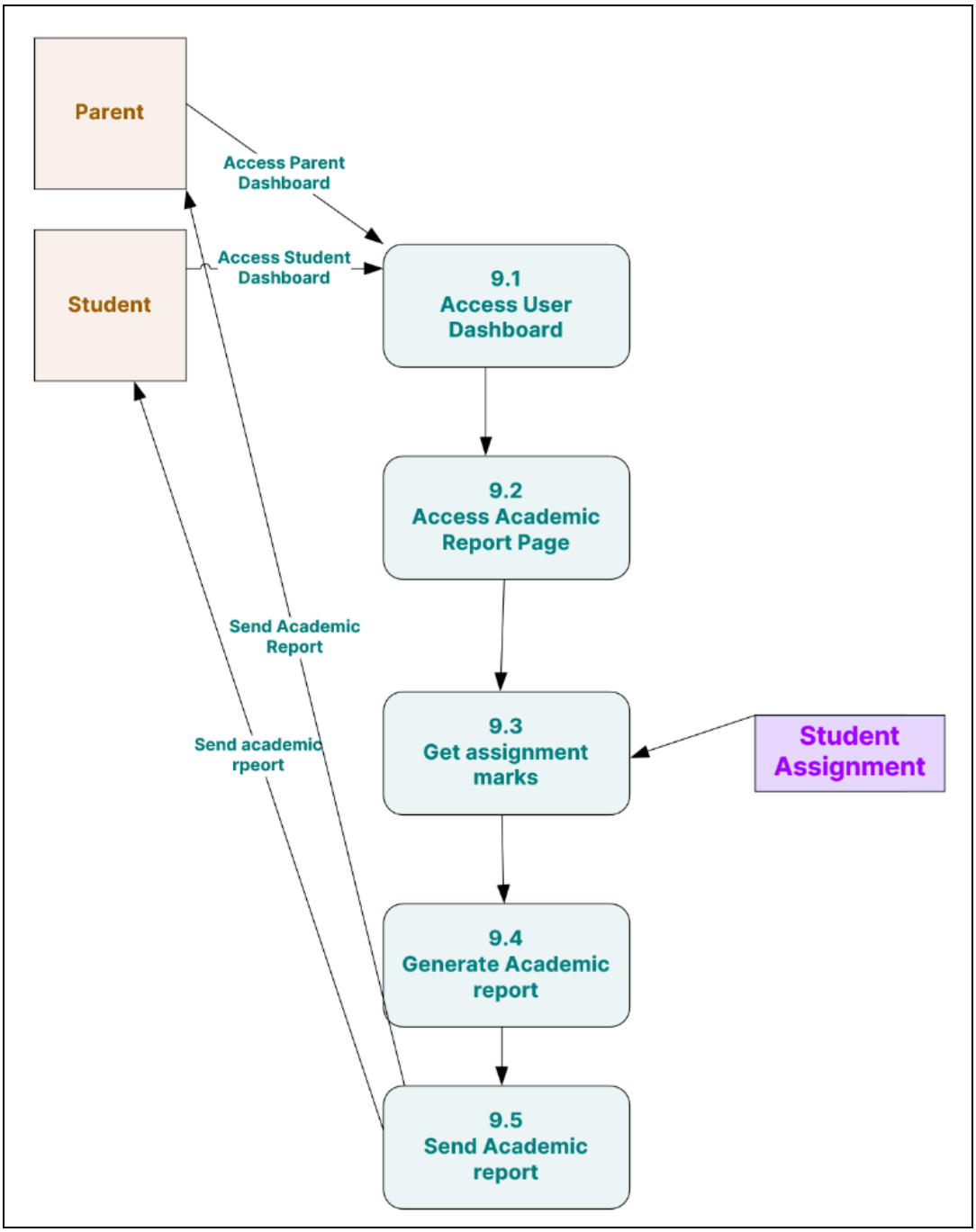


Diagram level 1 (Save Student Attendance)

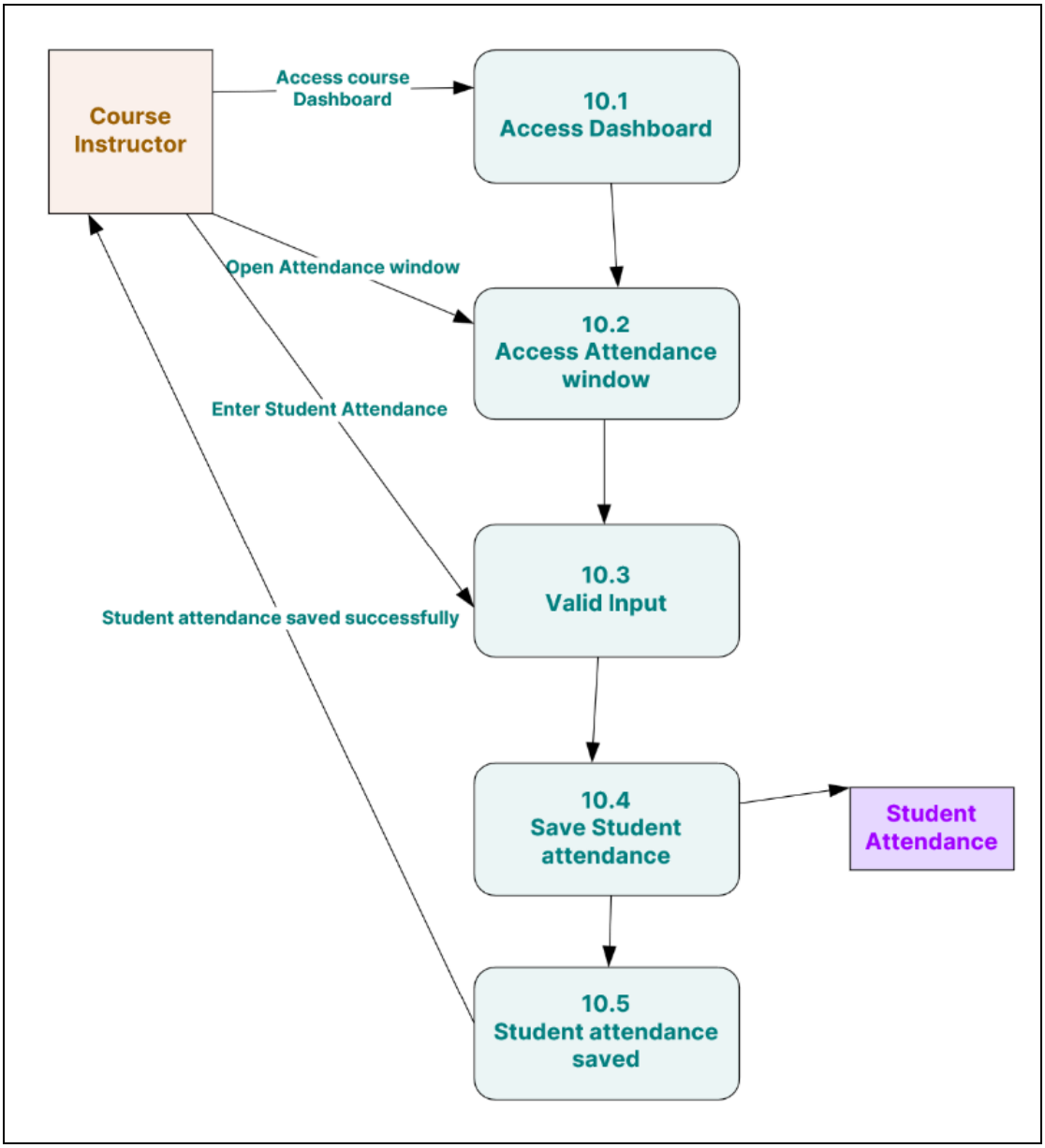


Diagram level 1 (View Attendance Record)

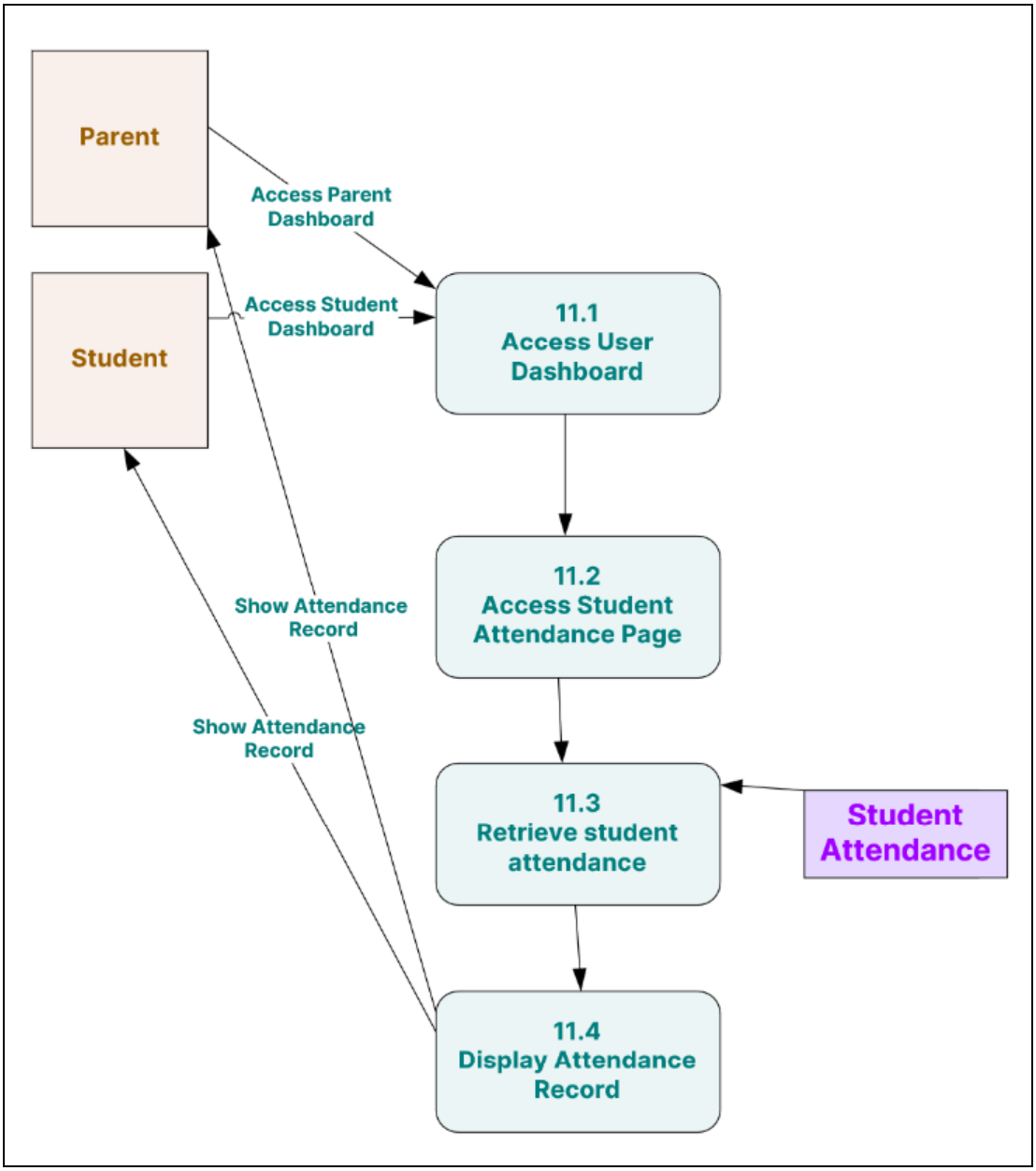


Diagram level 1 (Create Student Absence Notice to Parents)

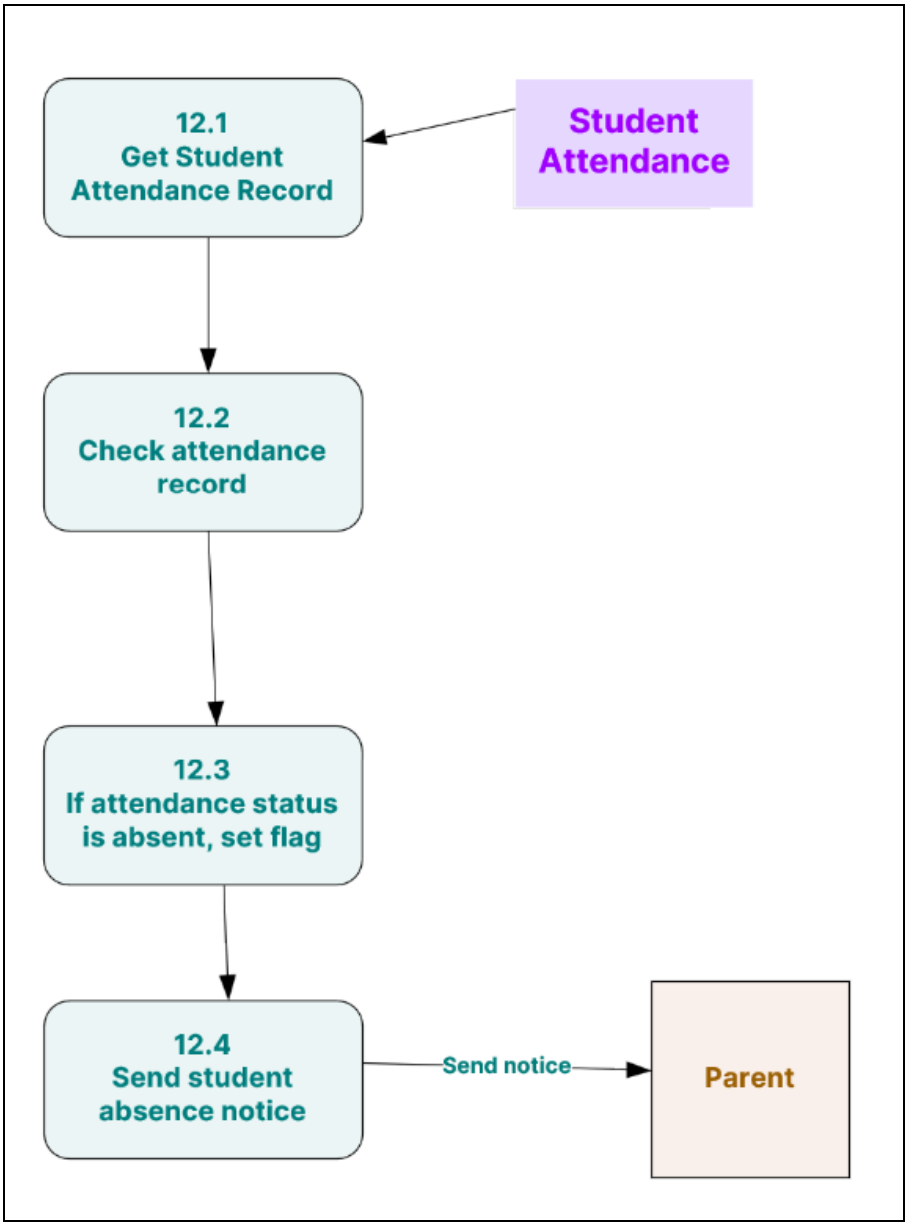


Diagram level 1 (Post Assignment)

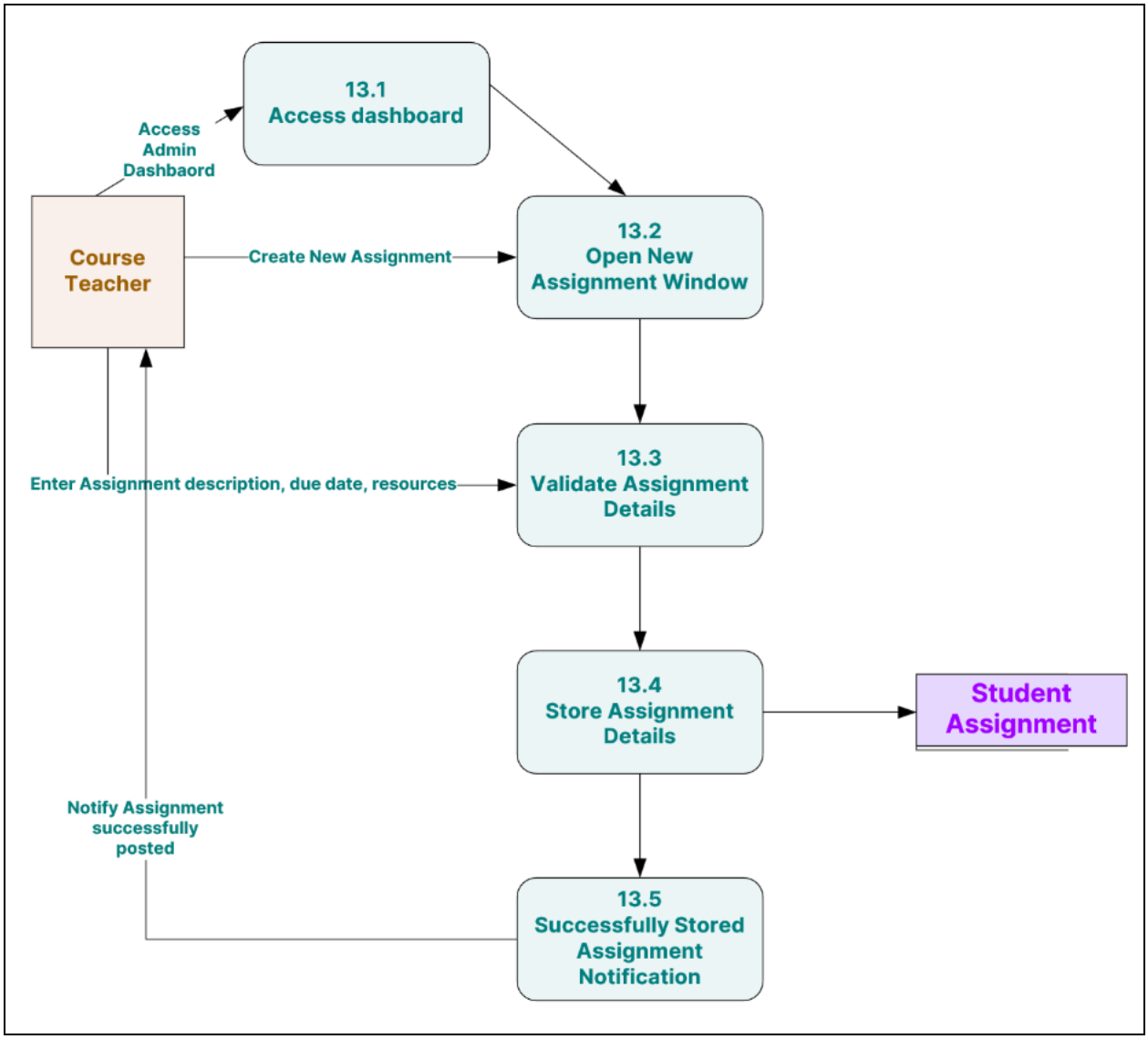


Diagram level 1 (Submit Assignment)

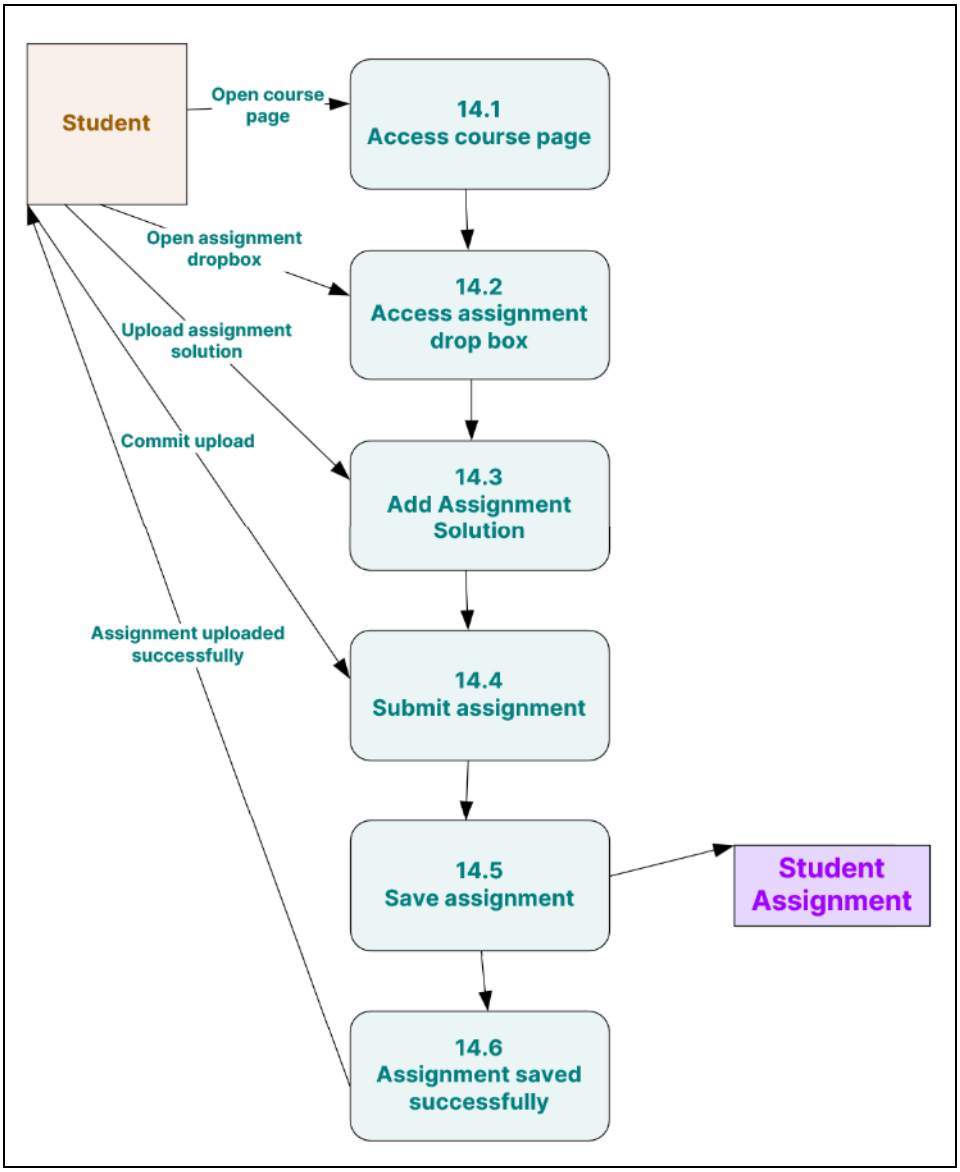


Diagram level 1 (Upcoming Assignment notice)

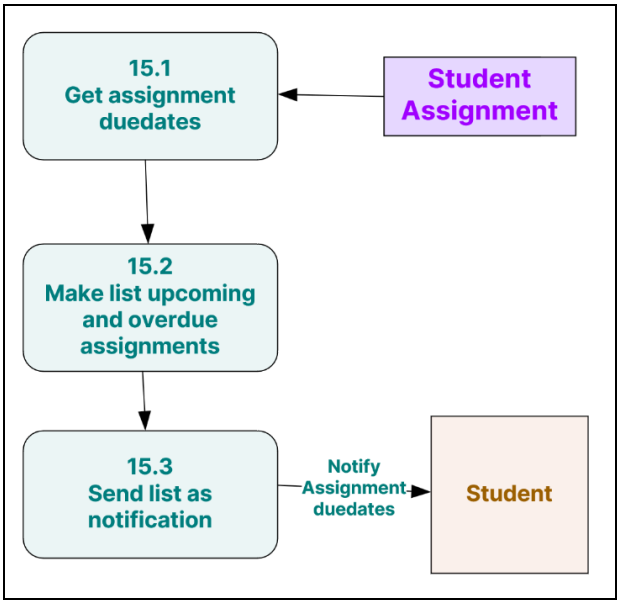


Diagram level 1 (View Admin Dashboard)

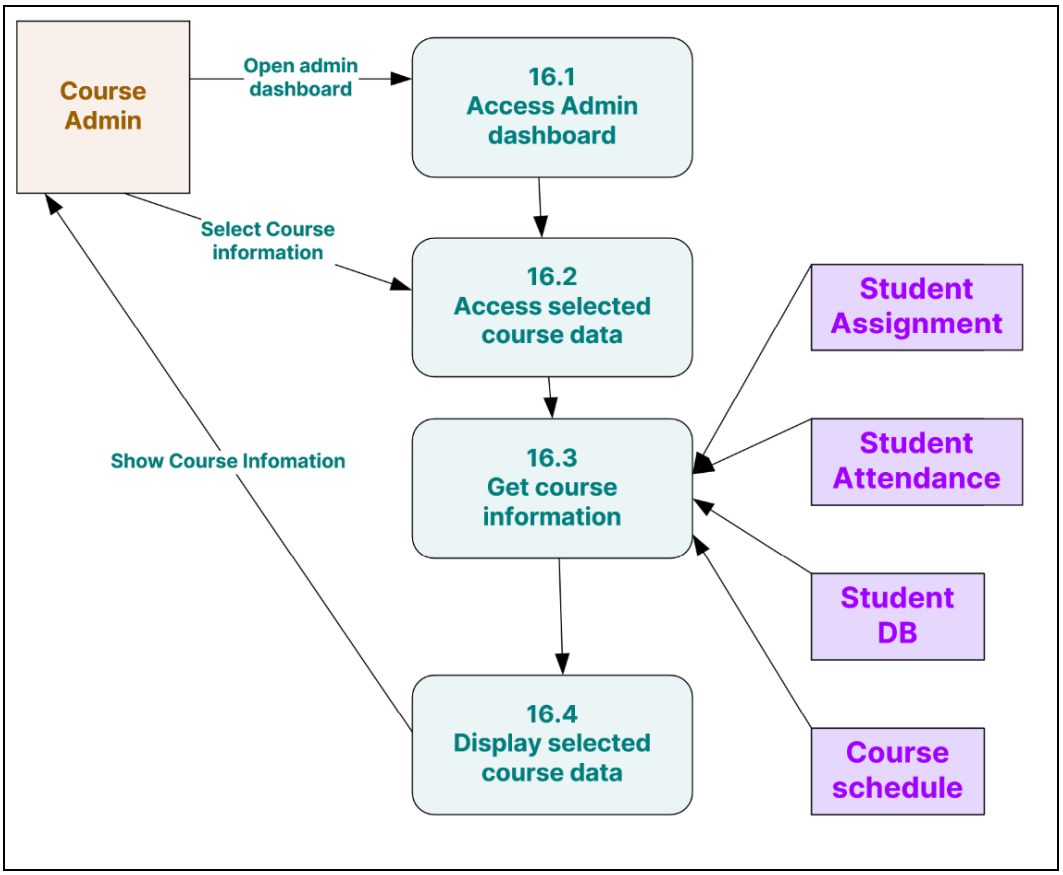


Diagram level 1 (Generate overall report)

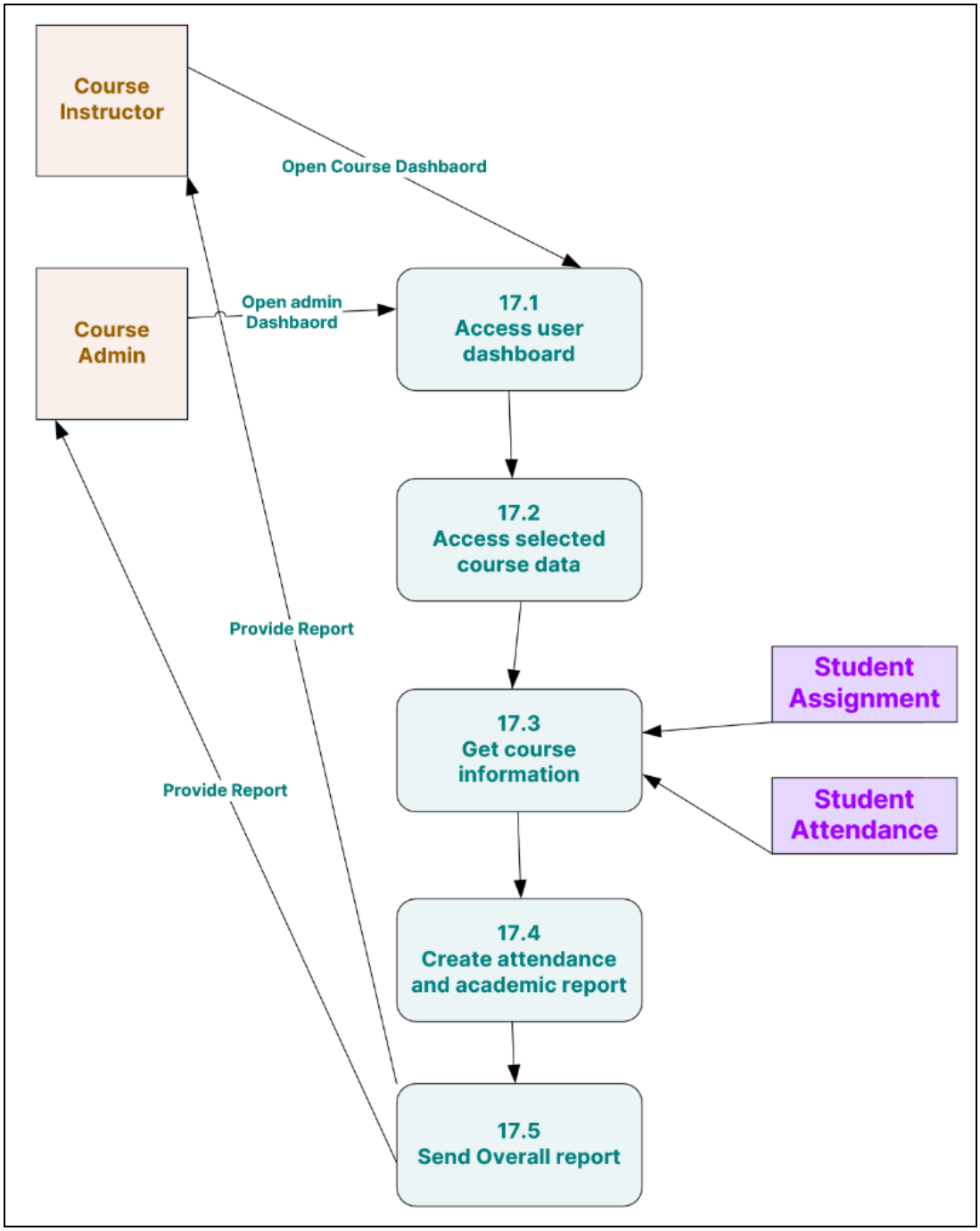
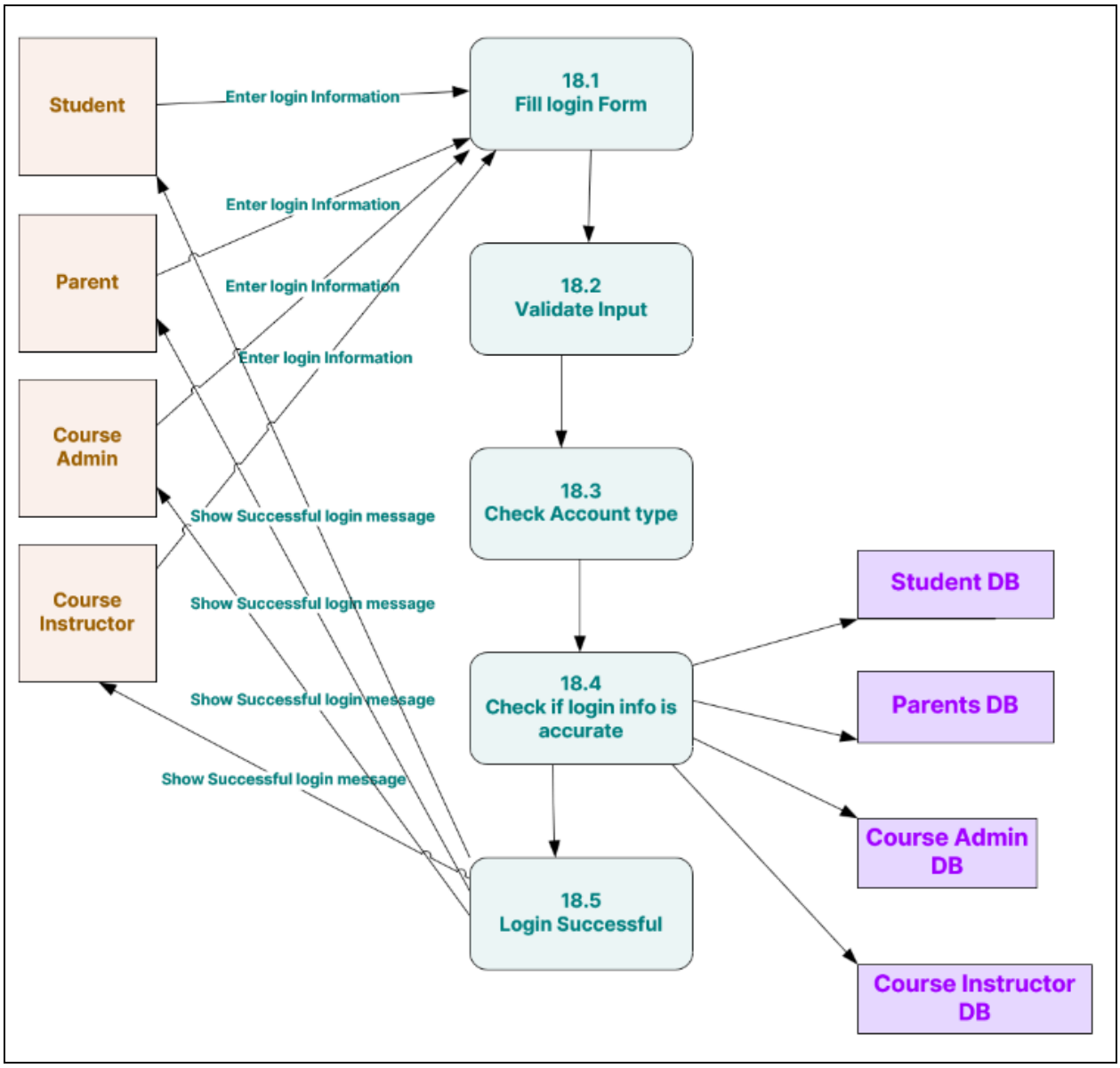
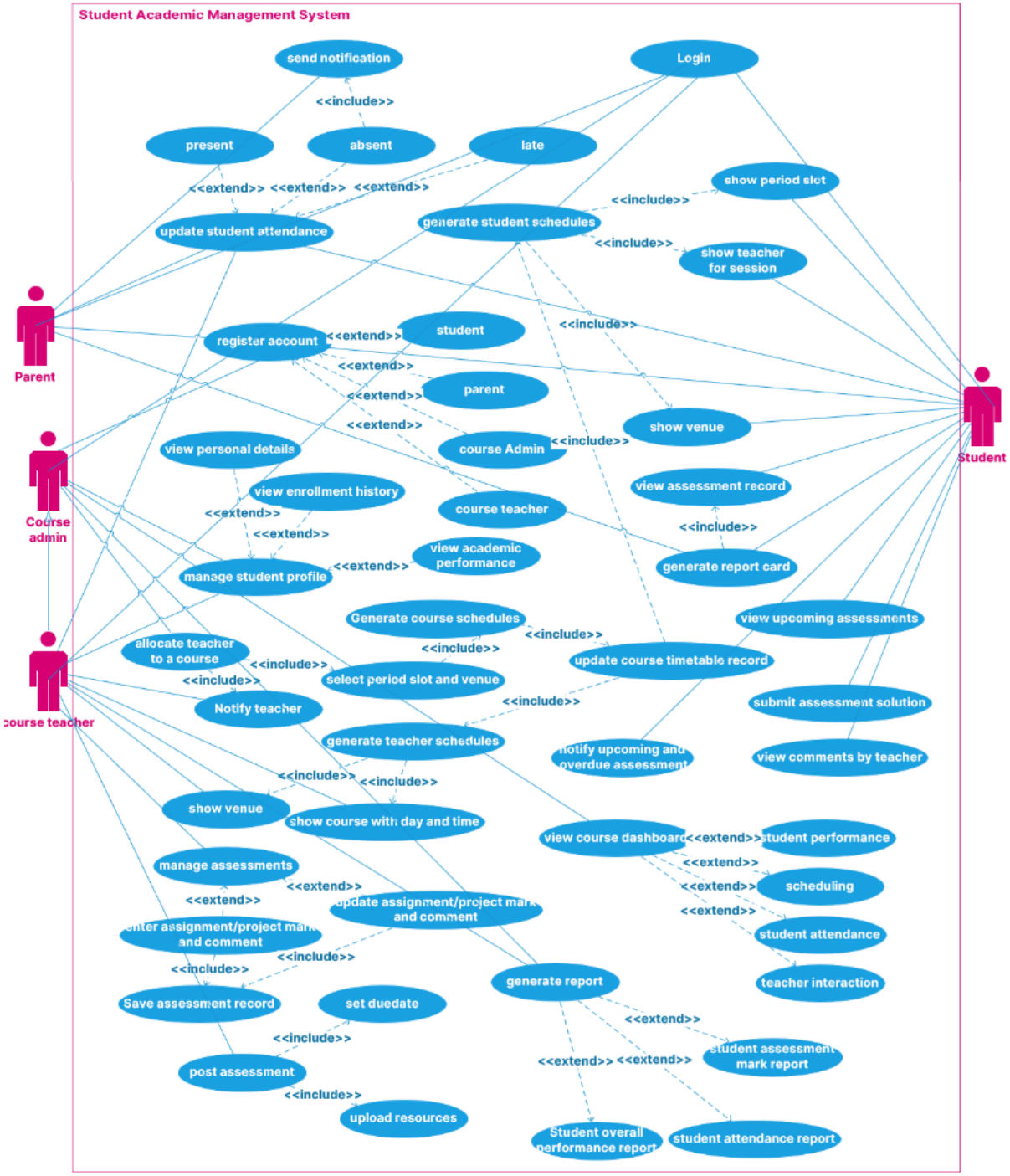


Diagram level 1 (User Login)



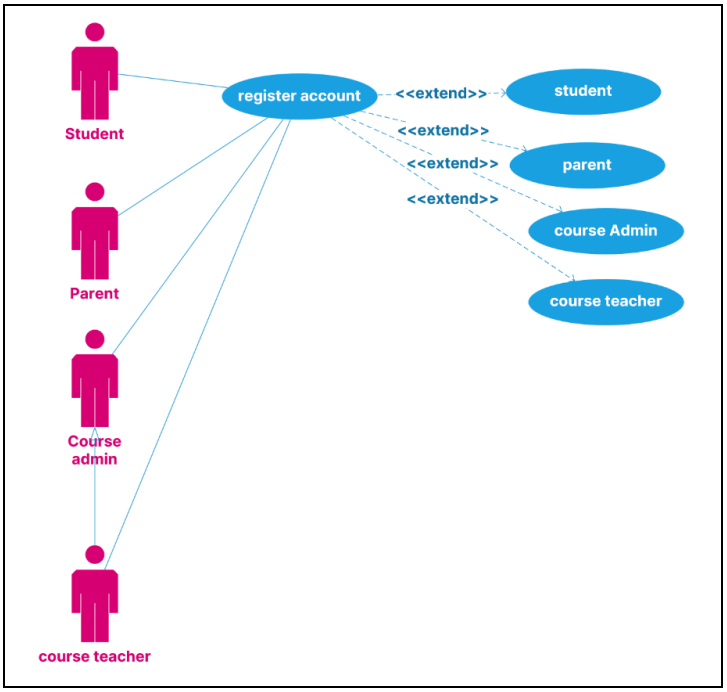
6.2. Object-Oriented Design Artifacts

6.2.1. Use-Case Diagram and Description



Use case: Register users with various levels of access

Diagram:



Brief Description:

The users register themselves with the Student Academic Management System based on whether they are a student, parent of a student, course administrator, or teacher who will teach the course. After registering with the system the various users can authenticate themselves into the system to access various services of the system depending on the account type.

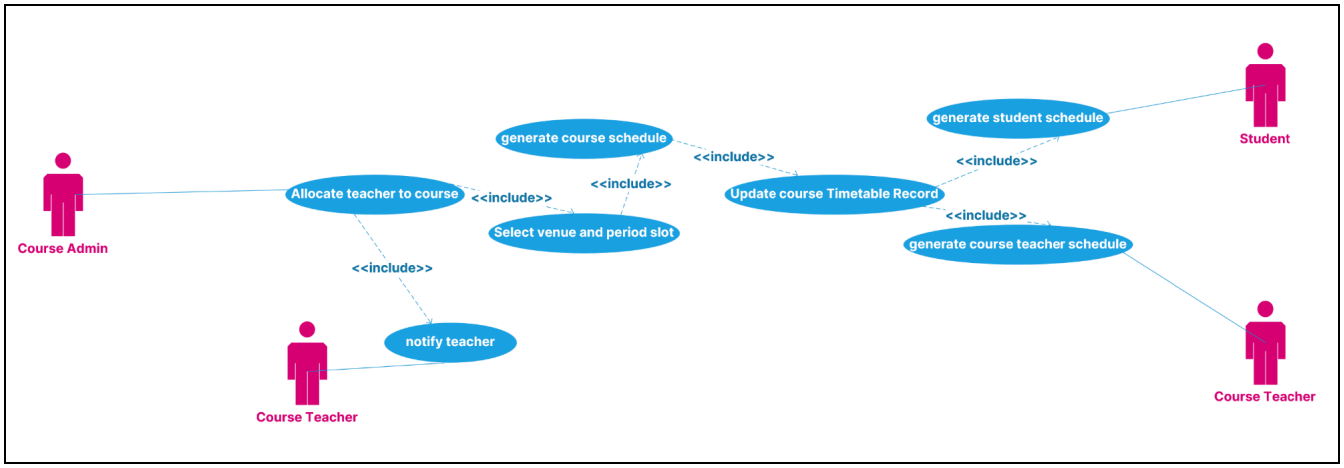
Step-by-step description:

1. The user, whether a student, parent of a student, course admin, or course teacher enters data that the system will ask them based on the type of account being created.
2. After filling in the user registration form, the user will press the register button and the website will validate the input. If any errors, a warning message will be shown to the user.
3. Once the user data is valid, the website will send the information to the server where it will be stored in a database.
4. After which the user can login with the details he/she has just registered with.

5. The user will now have access to the home page based on their account type.
6. After using the services on the web page the user can now choose to log out.

Use Case: Course Admin Creates course schedule

Diagram:



Brief Description:

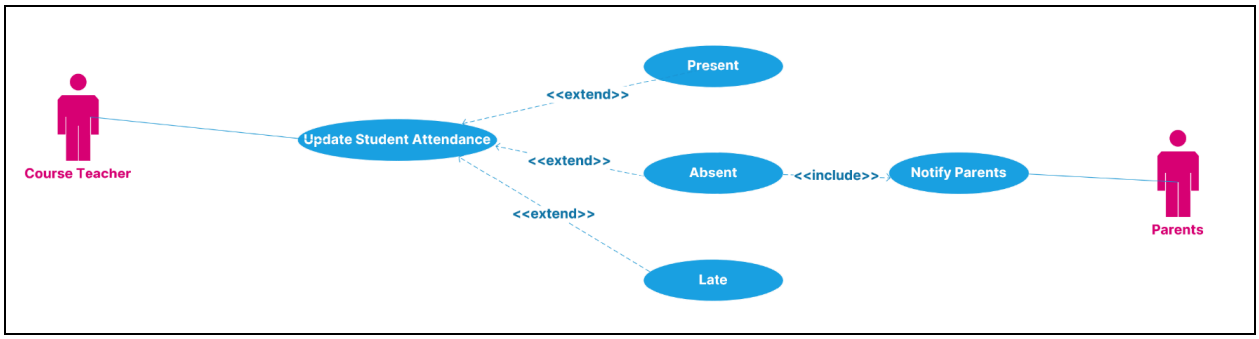
The course admin uses the Student Academic Management System to create the schedule for the course by selecting the course teacher, venue of the classes, and the time slot the class should take place. After saving the schedule, the course teacher and student timetables will be updated accordingly.

Step-by-step description:

1. Course admin logs into the system.
2. Course admin proceeds to open a create schedule window from the admin dashboard and fill in all the necessary details.
3. He then saves the schedule where the system checks if there aren't any conflicts. If there's a conflict the system will throw an error, else it will save the information.
4. A successful save message will pop up to show user information has been saved.
5. The user can now logout or continue using the system.

Use Case: Course Teacher/Instructor Updates student attendance

Diagram:



Brief Description:

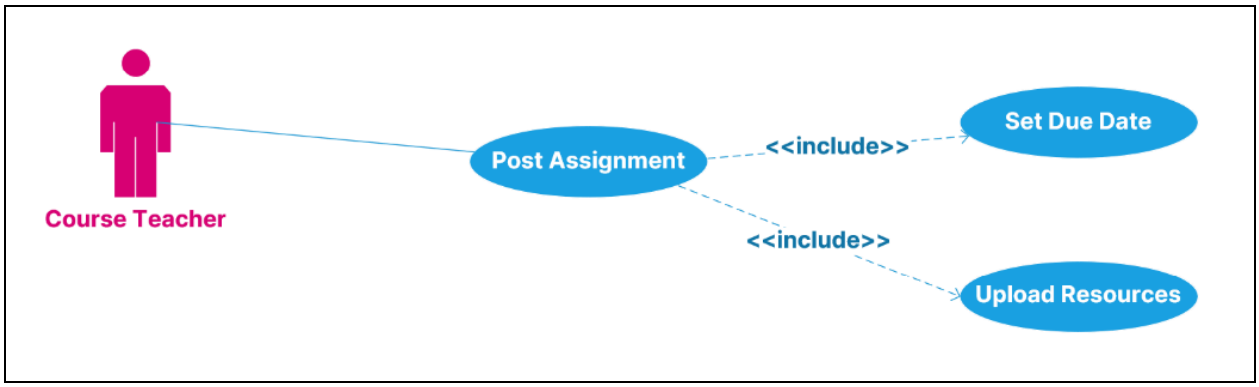
The course teacher uses the Student Academic Management System to save students' attendance for a class by marking them present, late, or absent. In case a student is absent, a notification is sent to their parent's account to notify them that their child has been absent for a class.

Step-by-step description:

1. The course teacher logs into the system.
2. Enter the attendance status for all students through a window in their dashboard.
3. Attendance status can either be present, late or absent.
4. In case a student's status is marked absent, a notification will be sent to their parents where they can view it from their dashboard.
5. The information will be stored and a message of success will be shown.
6. The user can choose to log off or continue using the system.

Use Case: Course Teacher/Instructor Post Assignment

Diagram:



Brief Description:

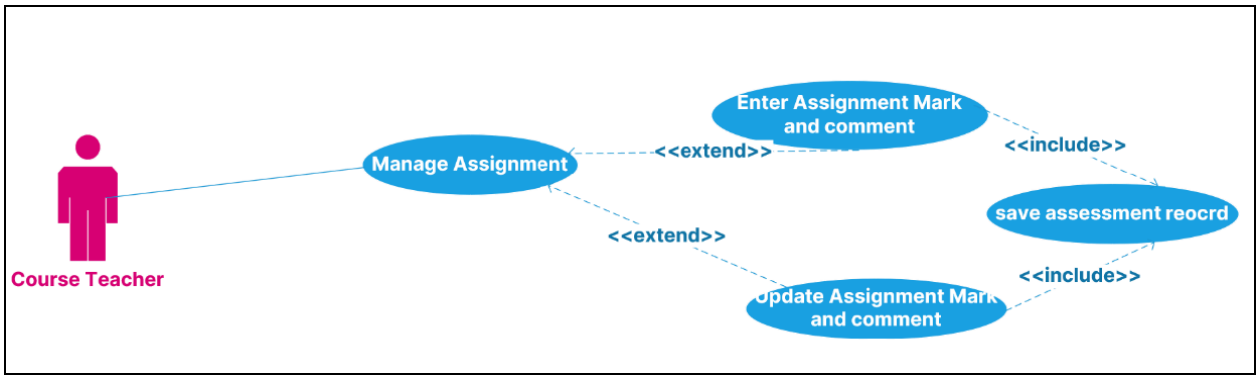
The Course Teacher Uses the Student Academic Management System to create new assignments and post them for students to complete and submit.

Step-by-step description:

1. The course teacher logs into the system.
2. Create a new assignment from the dashboard by opening a new assignment creation window.
3. Enter all the necessary details and save the assignment.
4. The system will validate the assignment information and save it if everything is alright or throw an error if there are any issues.
5. If saved successfully, a message indicating that the assignment is saved and successfully posted will appear.
6. The user can now choose to logout.

Use Case: Course Teacher/Instructor Mark Assignment

Diagram:



Brief Description:

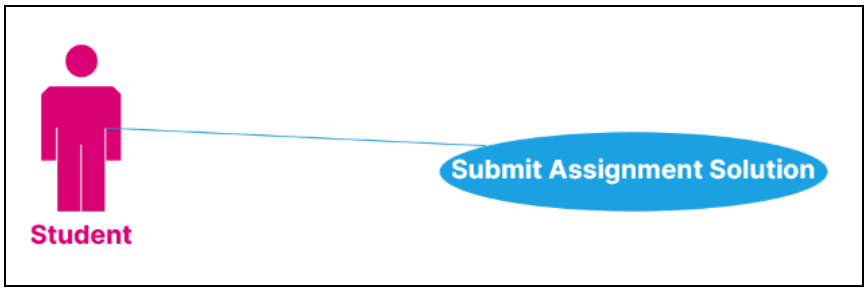
The Course Teacher Uses the Student Academic Management System to mark and enter marks for student submission of their solutions.

Step-by-step description:

1. The course teacher logs into the system.
2. The course teacher accesses a student's submission box and accesses their solution.
3. After accessing and marking the assignment solution, the course teacher can upload the marks.
4. After uploading the mark the user can now log off or continue using the system.

Use Case: Course Teacher/Instructor Mark Assignment

Diagram:



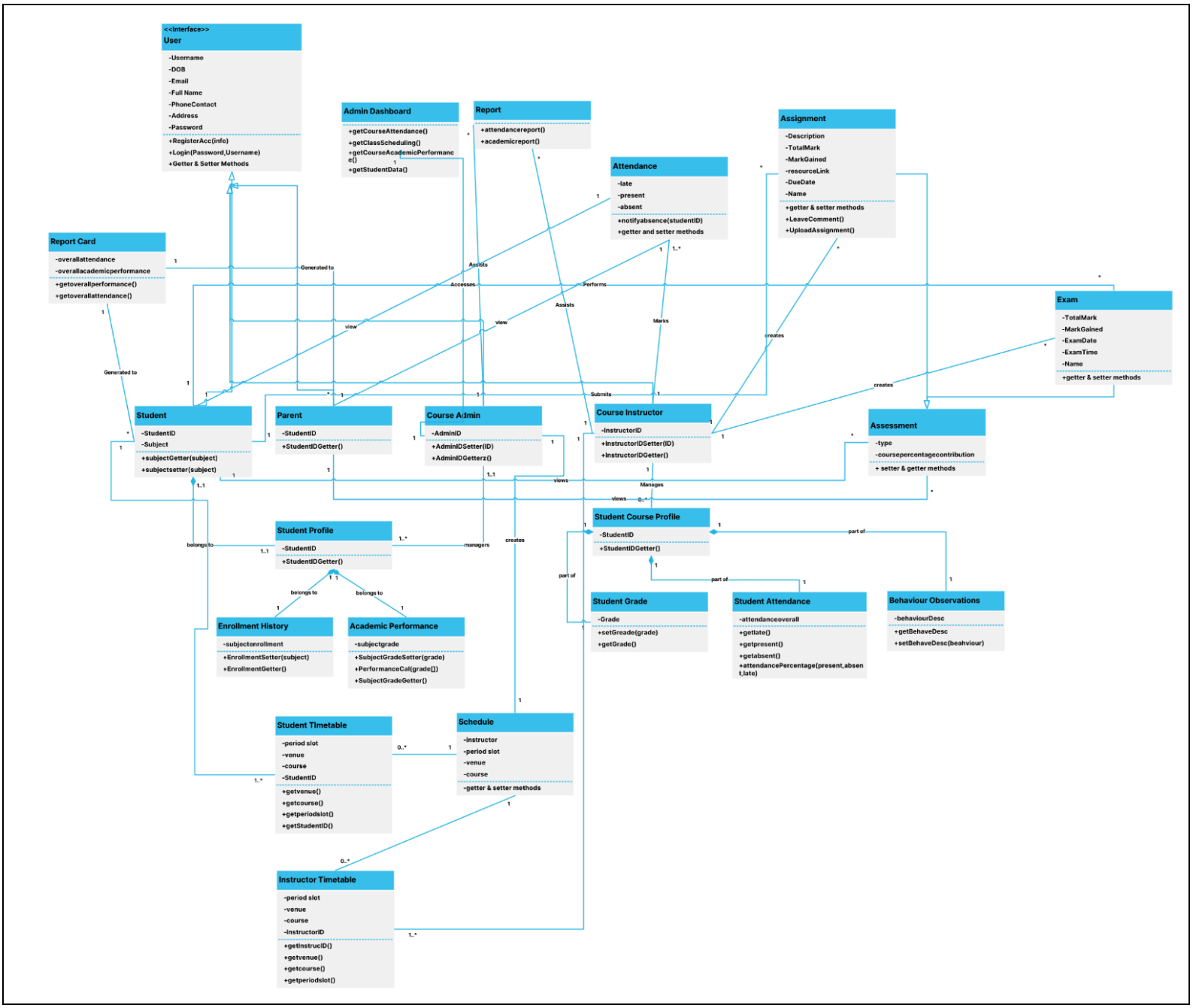
Brief Description:

A student uses the Student Academic Management System to submit their assignment solution through a submission dropbox.

Step-by-step description:

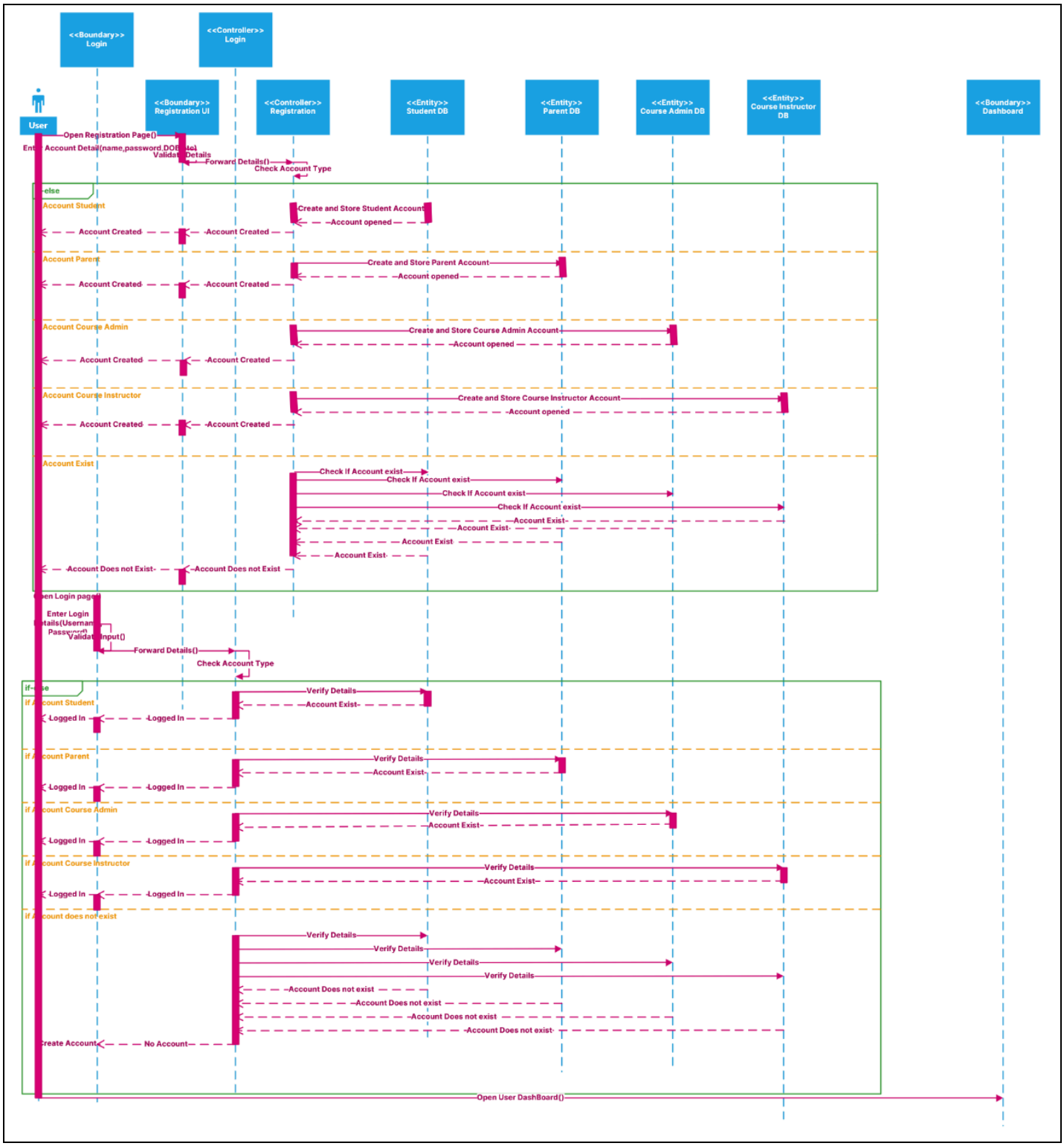
1. The student logs into the system.
2. The student accesses their assignment dropbox.
3. Saves the solution and uploads it.
4. A message to indicate the solution is saved is shown.
5. Students can now log out or continue using the system.

6.2.2. Domain Model Class Diagram

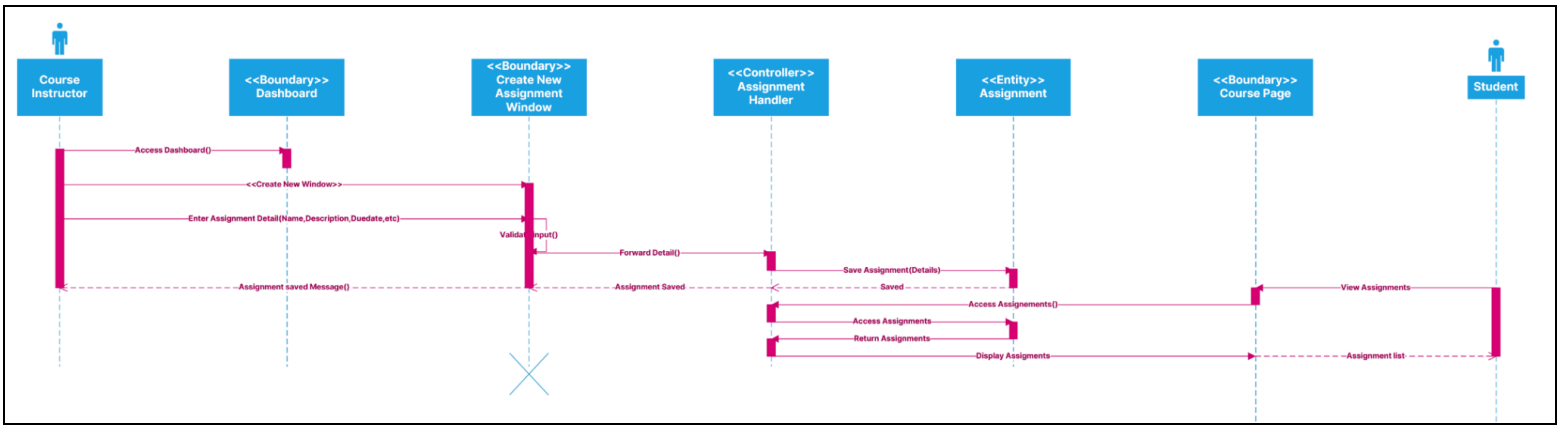


6.2.3. A major Sequence Diagram (s)

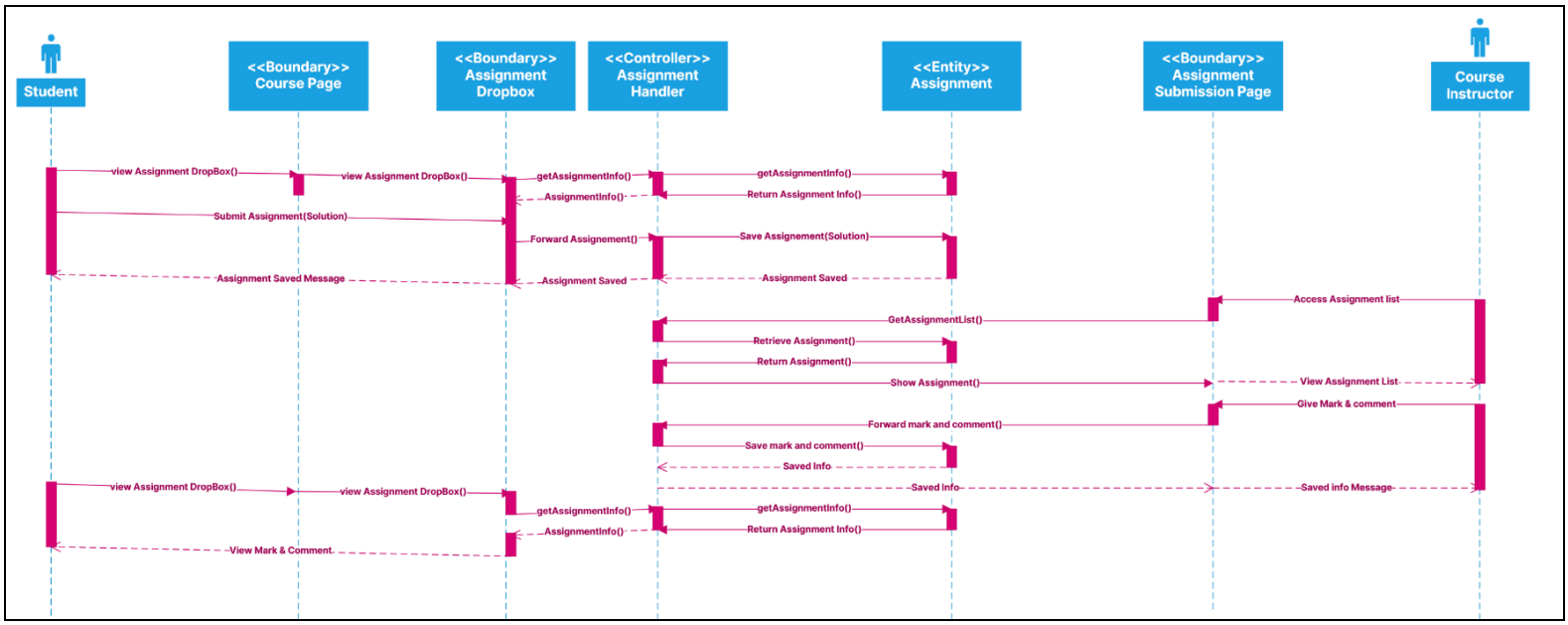
Registration And Login



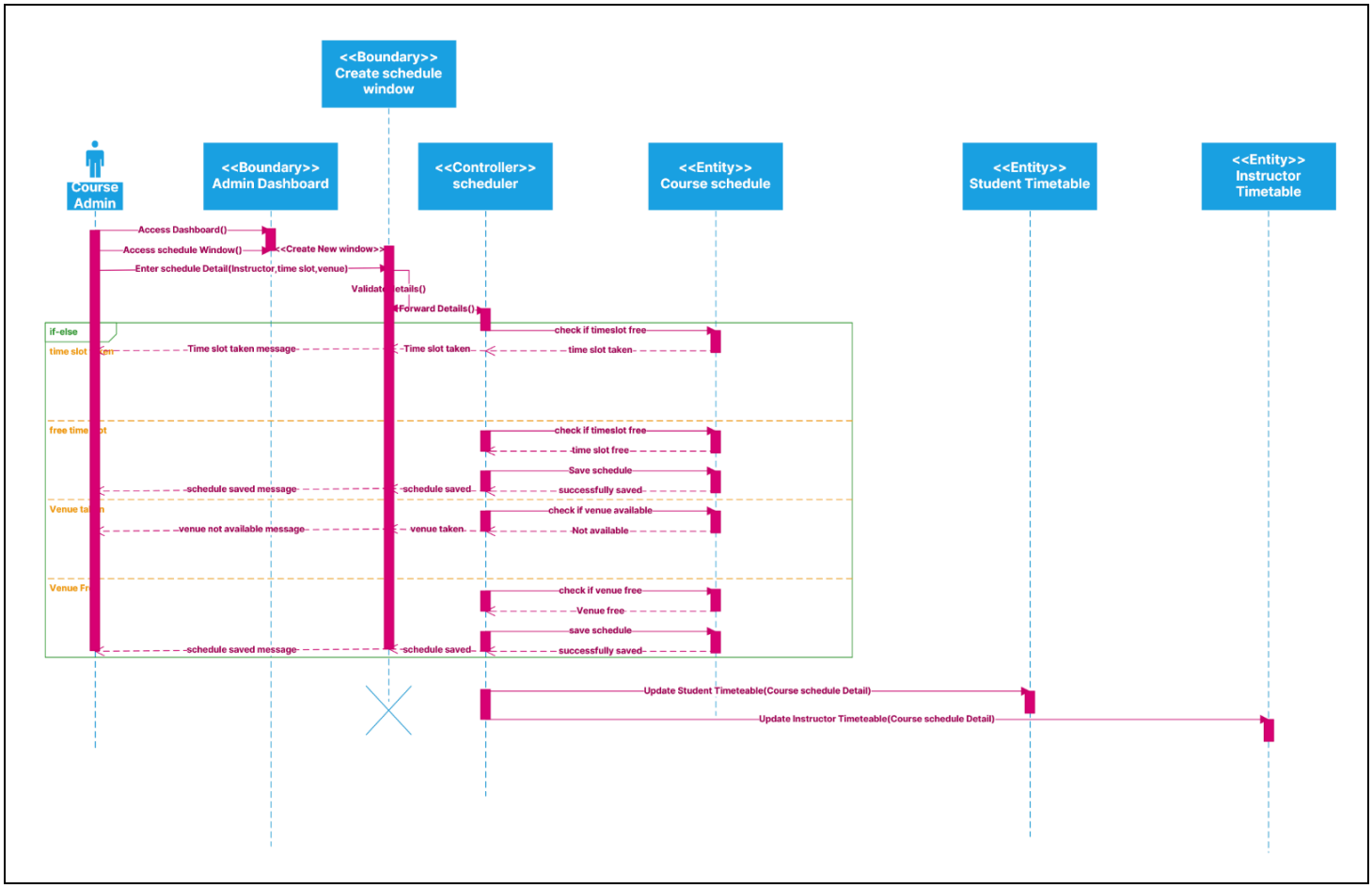
Create Assignment



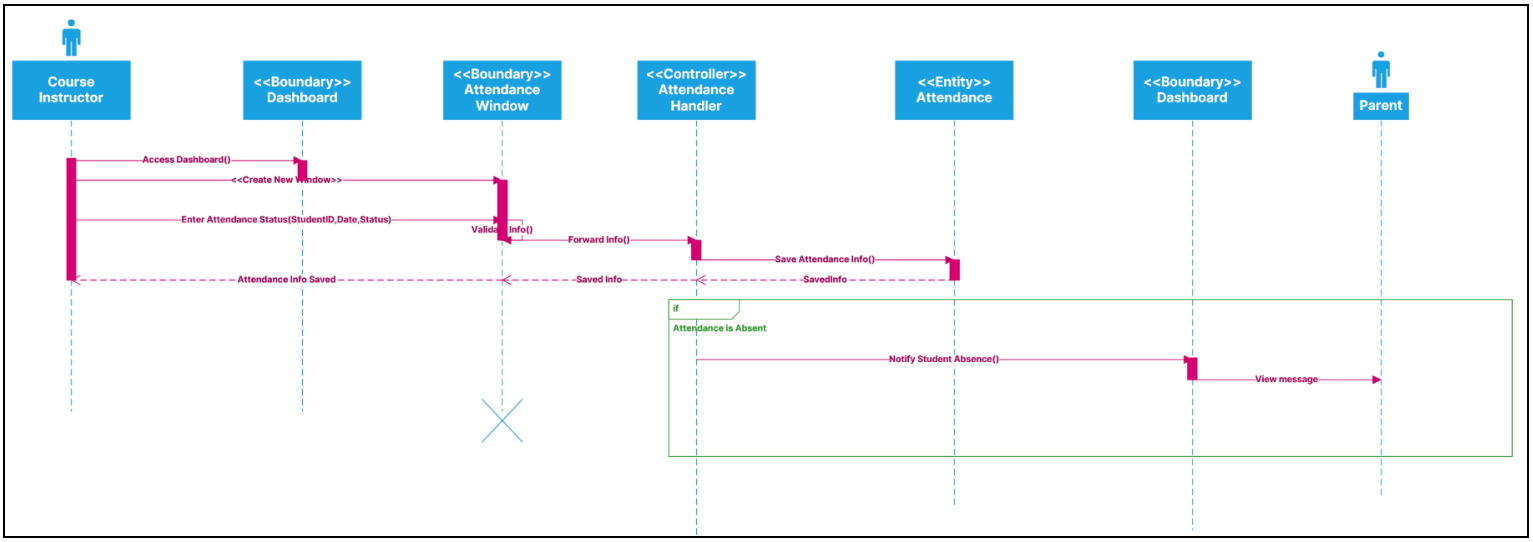
Submit And Mark Assignment



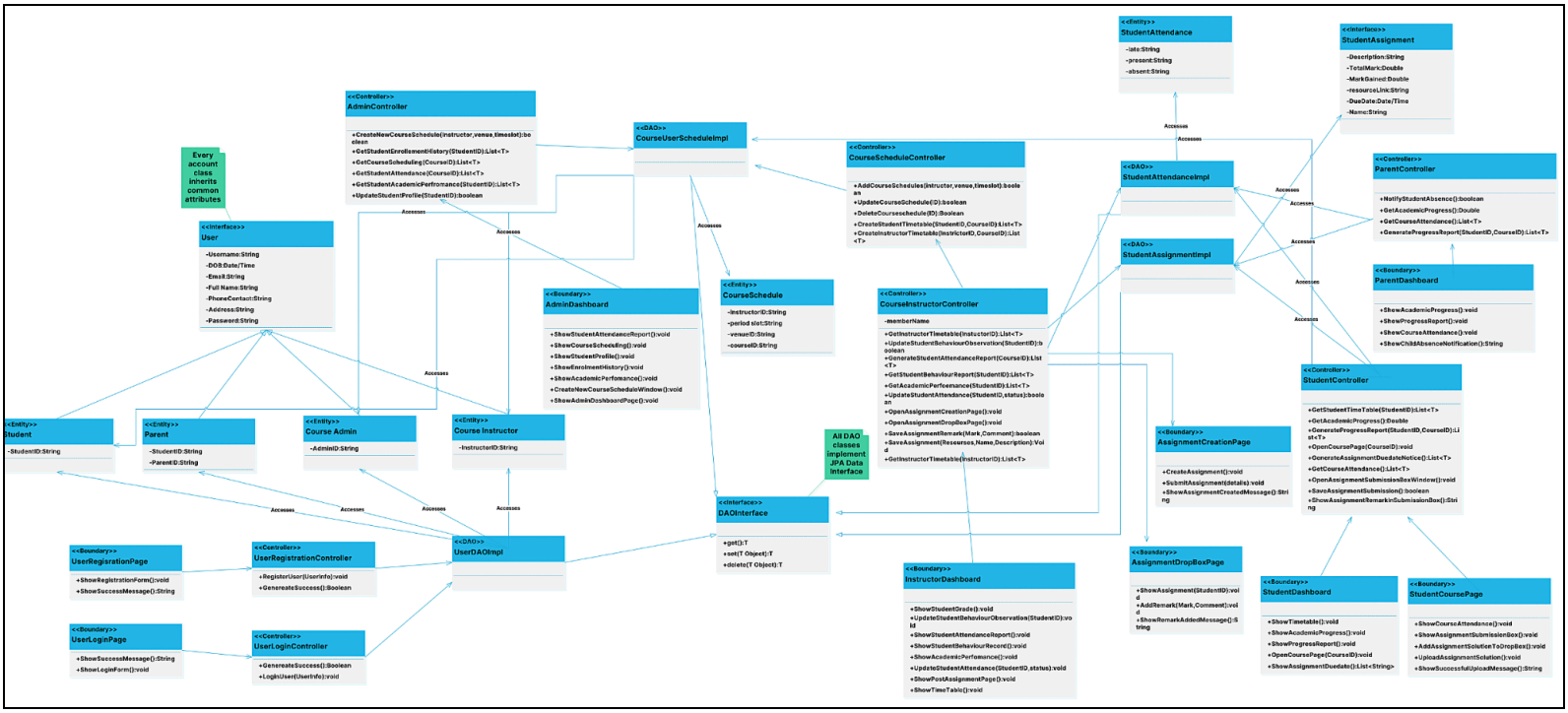
Create Course Schedule



Take Student Attendance

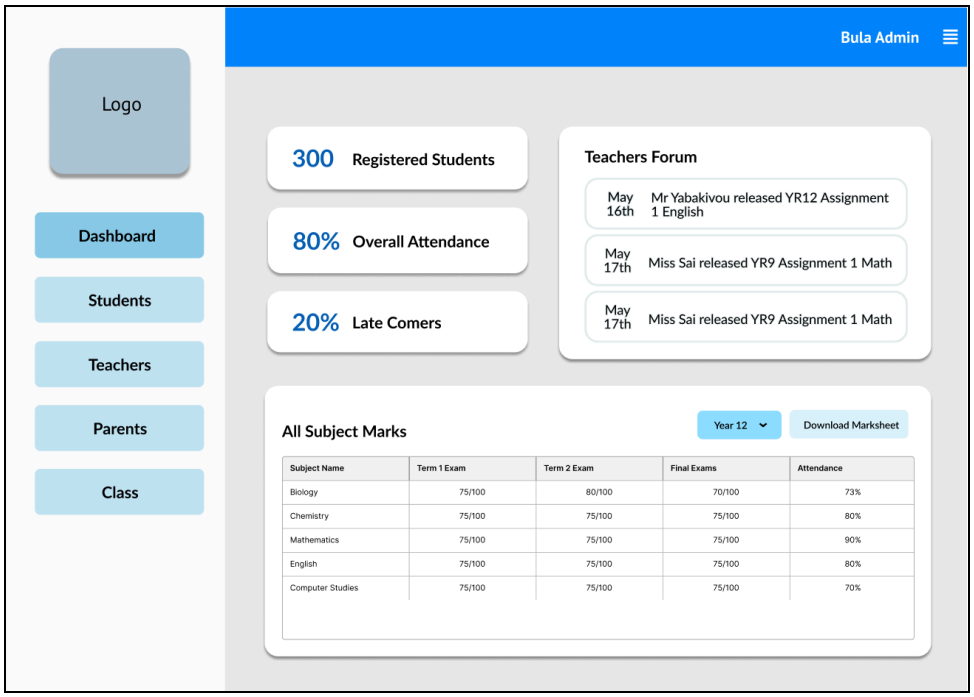


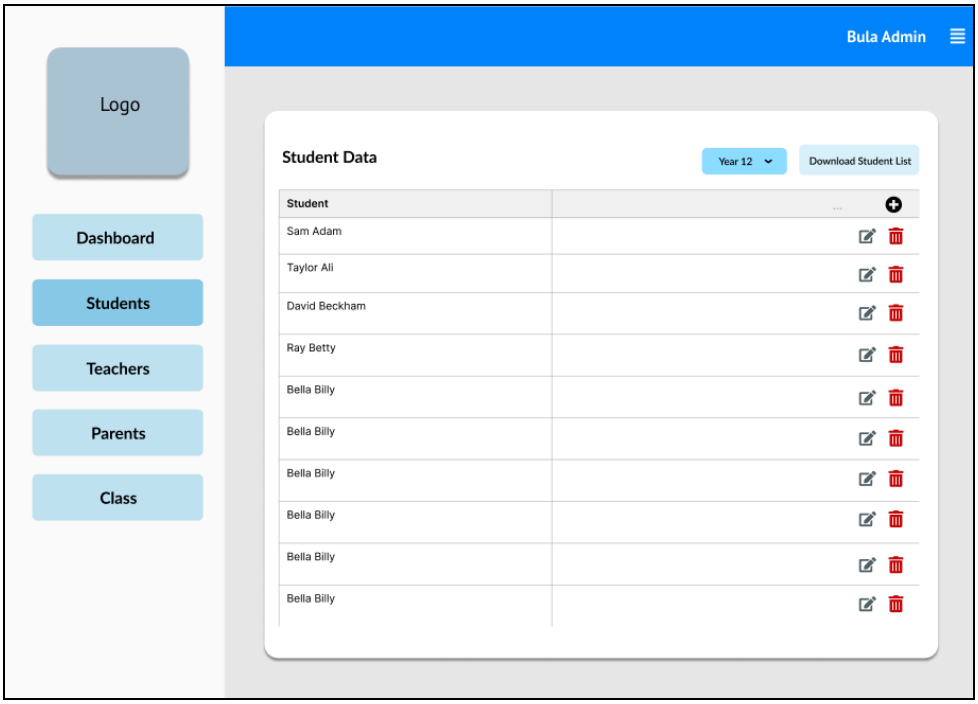
6.2.4. Detailed/Design Class Diagram

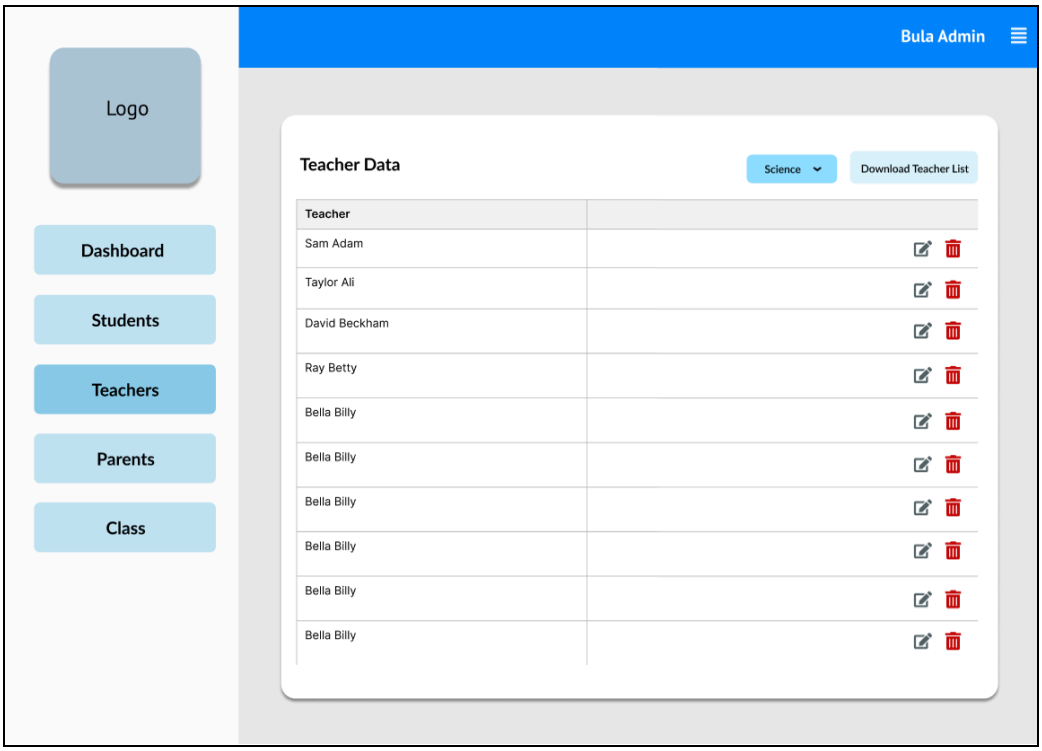


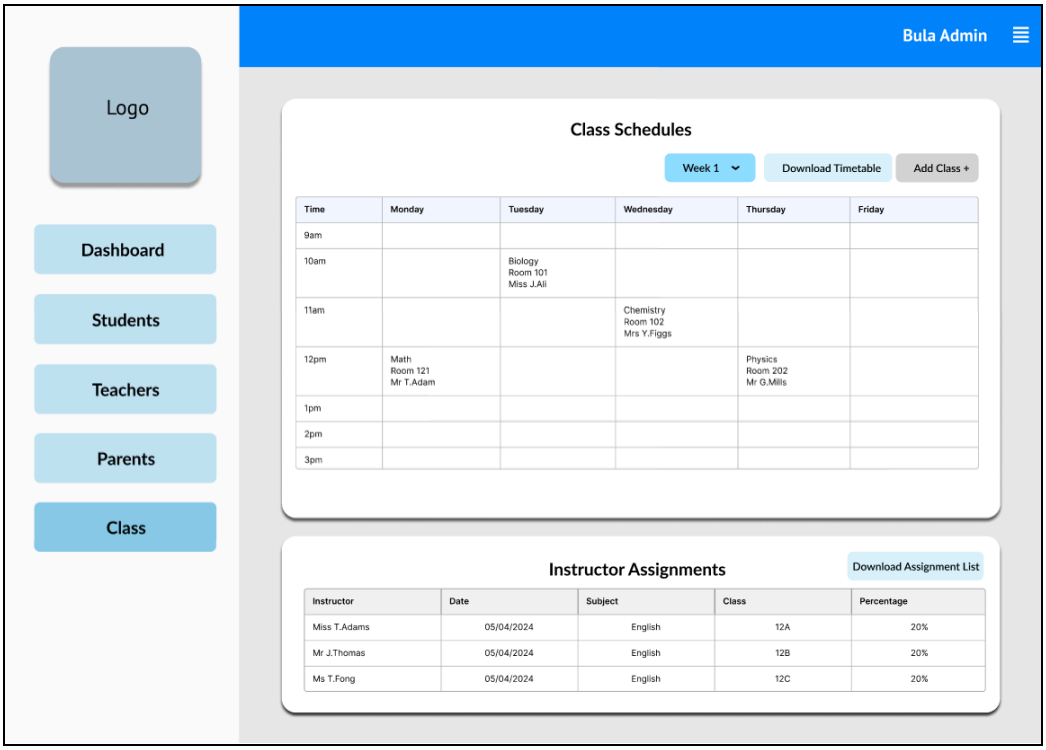
7. User Interfaces

7.1 Admin Panel

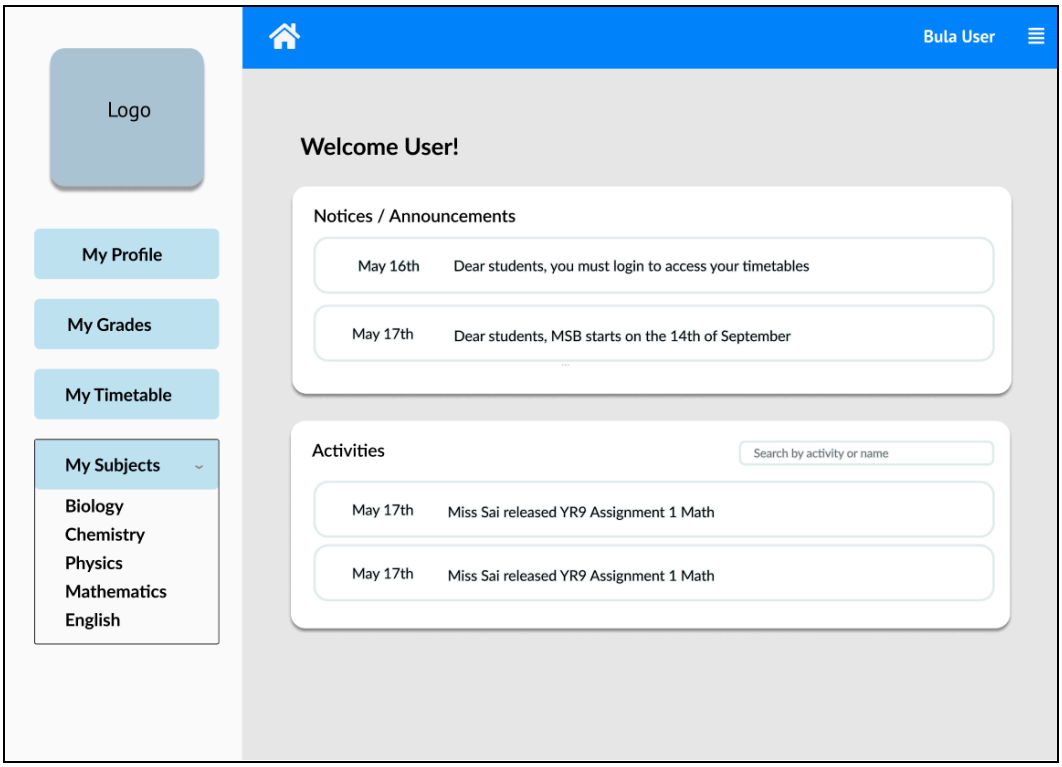


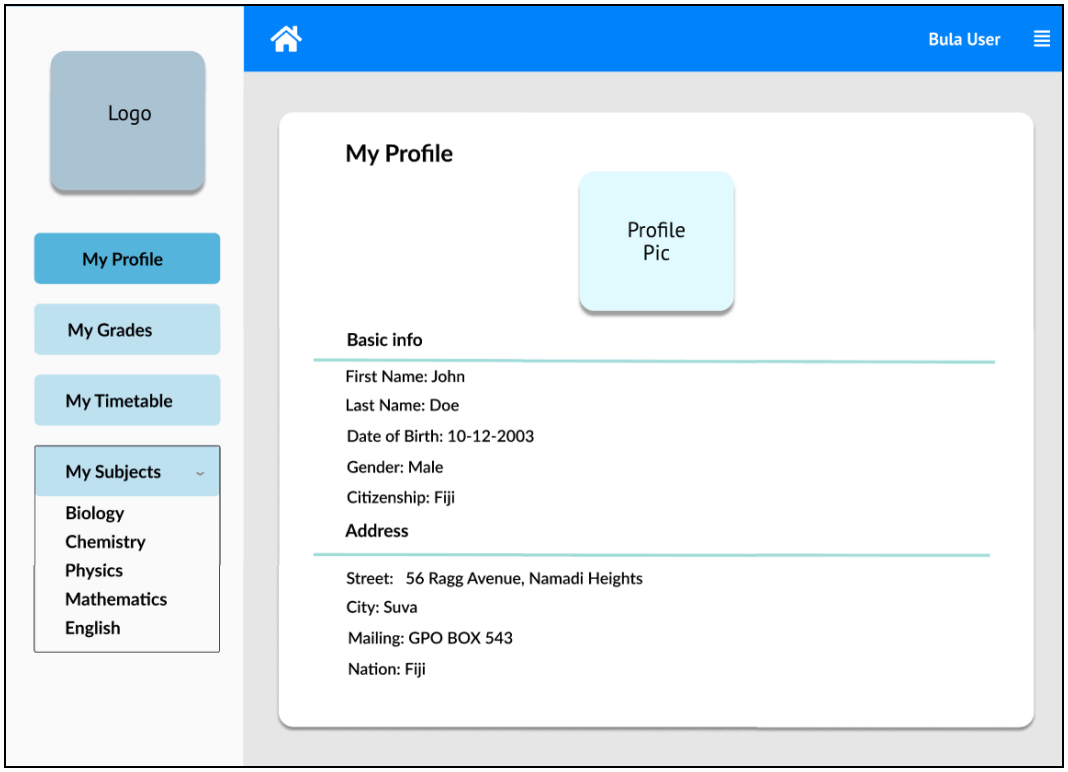


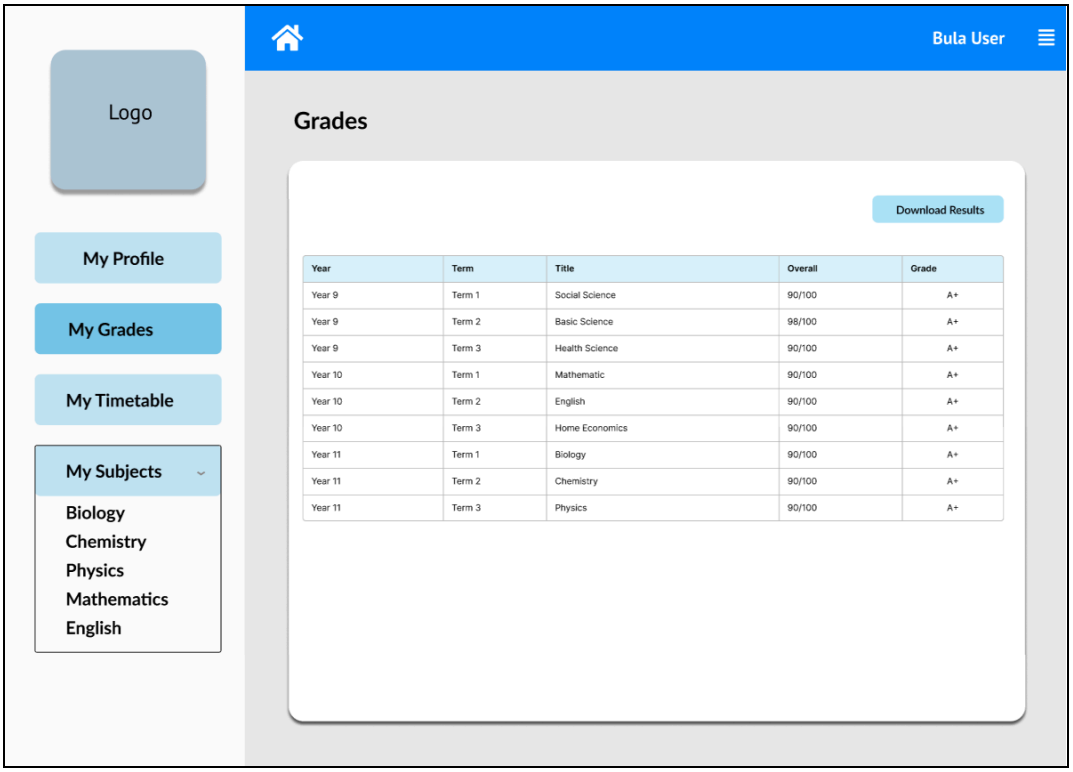

Bula Admin
Logo
Dashboard
Students
Teachers
Parents
Class
Teacher Data
Science
Download Teacher List
Teacher
Sam Adam
Taylor Ali
David Beckham
Ray Betty
Bella Billy
Bella Billy
Bella Billy
Bella Billy
Bella Billy
Bella Billy

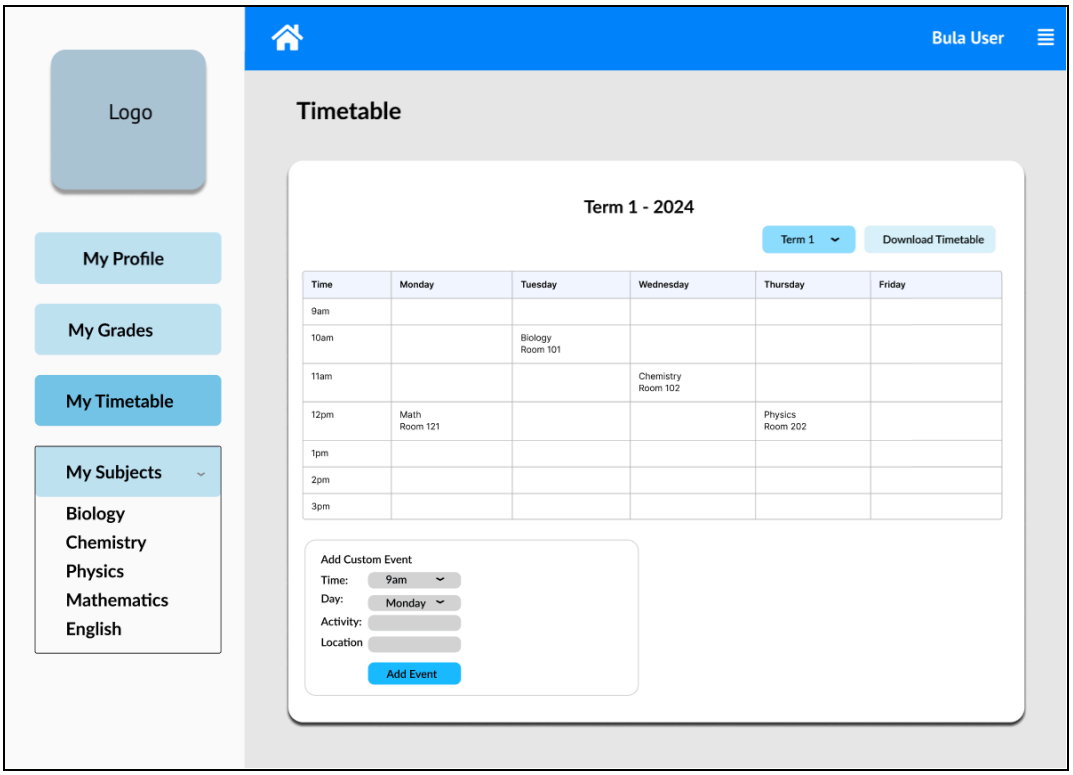

Bula Admin
Logo
Dashboard
Students
Teachers
Parents
Class
Class Schedules
Week 1
Download Timetable
Add Class +
Time
Monday
Tuesday
Wednesday
Thursday
Friday
9am
10am
Biology
Room 101
Miss J.Ali
11am
Chemistry
Room 102
Mrs Y.Figgs
12pm
Math
Room 121
Mr T.Adam
Physics
Room 202
Mr G.Mills
1pm
2pm
3pm
Instructor Assignments
Download Assignment List
Instructor
Date
Subject
Class
Percentage
Miss T.Adams
05/04/2024
English
12A
20%
Mr J.Thomas
05/04/2024
English
12B
20%
Ms T.Fong
05/04/2024
English
12C
20%

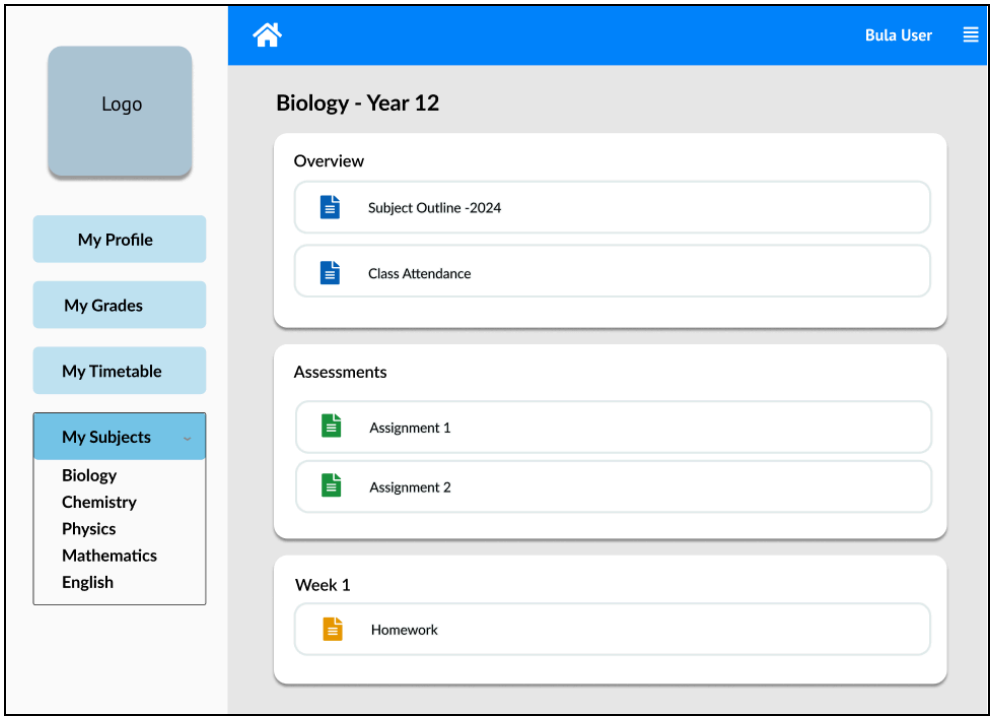
7.2 Student User Interface



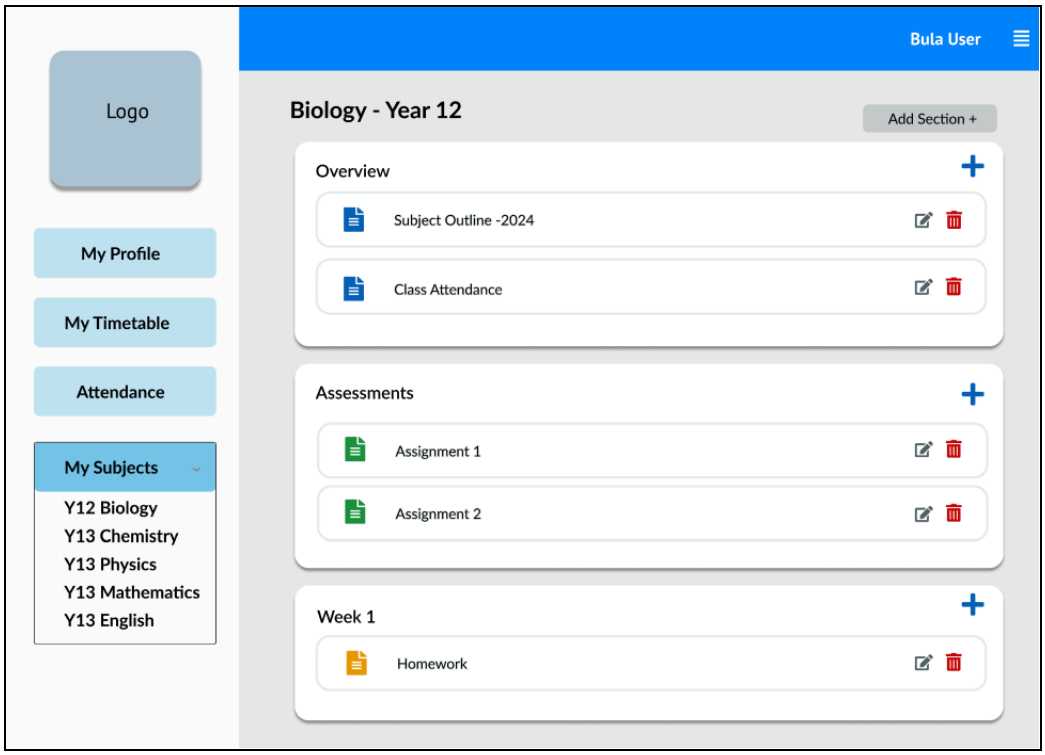



Logo
My Profile
My Grades
My Timetable
My Subjects
Biology
Chemistry
Physics
Mathematics
English
Bula User
Grades
Download Results
Year | Term | Title | Overall | Grade
Year 9 | Term 1 | Social Science | 90/100 | A+
Year 9 | Term 2 | Basic Science | 98/100 | A+
Year 9 | Term 3 | Health Science | 90/100 | A+
Year 10 | Term 1 | Mathematic | 90/100 | A+
Year 10 | Term 2 | English | 90/100 | A+
Year 10 | Term 3 | Home Economics | 90/100 | A+
Year 11 | Term 1 | Biology | 90/100 | A+
Year 11 | Term 2 | Chemistry | 90/100 | A+
Year 11 | Term 3 | Physics | 90/100 | A+


Logo
My Profile
My Grades
My Timetable
My Subjects
Biology
Chemistry
Physics
Mathematics
English
Bula User
Timetable
Term 1 - 2024
Term 1
Download Timetable
Time | Monday | Tuesday | Wednesday | Thursday | Friday
9am
10am
Biology Room 101
11am
Chemistry Room 102
12pm
Math Room 121
Physics Room 202
1pm
2pm
3pm
Add Custom Event
Time: 9am
Day: Monday
Activity:
Location
Add Event



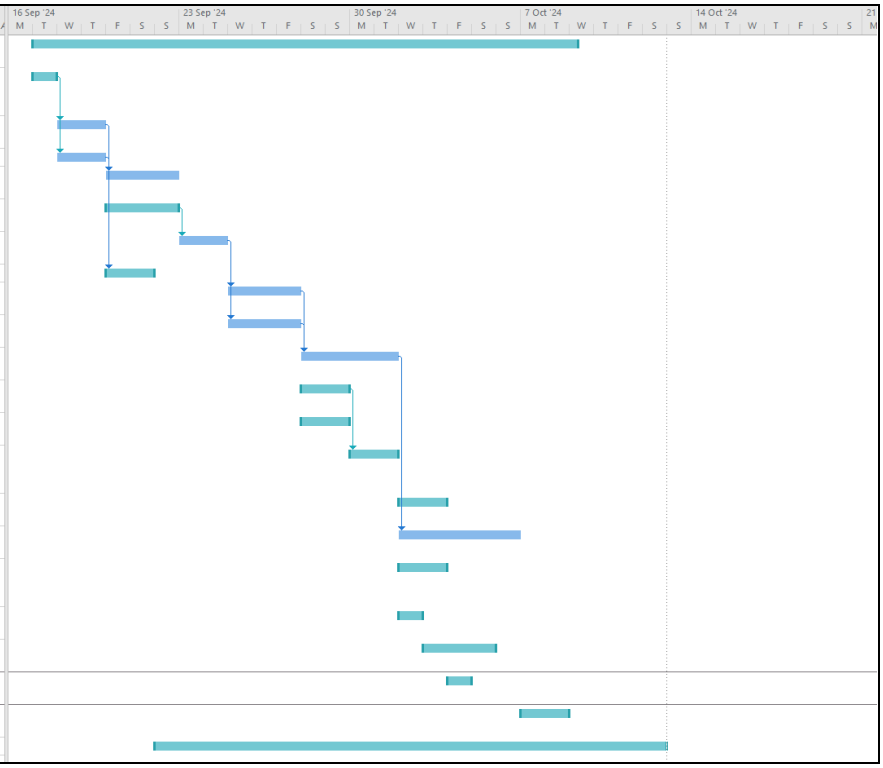
7.3 Teacher User Interface



8. Implementation Plan

GANTT CHART

	Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1		Database Creation and modification	22.13 days	Tue 17/09/24	Wed 9/10/24		
2		Registration & Login Page Creation and Implementation	1 day	Tue 17/09/24	Tue 17/09/24		
3		User Profile Creation	2 days	Wed 18/09/24	Thu 19/09/24	2	
4		Testing	2 days	Wed 18/09/2	Thu 19/09/24	2	
5		Admin DashBoard Creation	3 days	Fri 20/09/24	Sun 22/09/24	3,4	
6		Create Course Module Creation	3 days	Fri 20/09/24	Sun 22/09/24		
7		Course Schedule Module Creation	2 days	Mon 23/09/24	Tue 24/09/24	6	
8		HomePage Creation	2 days	Fri 20/09/24	Sat 21/09/24	4	
9		Student Timetable Module Creation	3 days	Wed 25/09/24	Fri 27/09/24	7	
10		Instructor Timetable Mdule Creation	3 days	Wed 25/09/24	Fri 27/09/24	7	
11		Course Page Creation	4 days	Sat 28/09/24	Tue 1/10/24	10,9	
12		Assignment Page Creation	2 days	Sat 28/09/24	Sun 29/09/24		
13		Resource Upload Module Creation	2 days	Sat 28/09/24	Sun 29/09/24		
14		Assignment Dropbox Module Creation	2 days	Mon 30/09/24	Tue 1/10/24	12	
15		Course Marksheet Module Creation	2 days	Wed 2/10/24	Thu 3/10/24		
16		Instructor Panel creation	5 days	Wed 2/10/24	Sun 6/10/24	11	
17		Instructor attendance Fill Module	2 days	Wed 2/10/24	Thu 3/10/24		
18		Add Mark to Assignements	1 day	Wed 2/10/24	Wed 2/10/24		
19		Admin Report Generation Module	3 days	Thu 3/10/24	Sat 5/10/24		
20		Absence Notification	1 day	Fri 4/10/24	Fri 4/10/24		
21		Report Card Generation	2 days	Mon 7/10/24	Tue 8/10/24		
22		Componet Testing	21 days	Sun 22/09/2	Sat 12/10/24		



9. Definitions and Acronyms.

Term/Acronym	Definition
WCAG	Web Content Accessibility Guideline
MFA	Multi-factor authentication(MFA)
RBAC	Role-based access control
DFD	Data flow diagram
API	An application programming interface is a way for two or more computer programs or components to communicate with each other.
SQL injection	a code injection technique used to attack data-driven applications, in which malicious SQL statements are inserted into an entry field for execution
Stakeholders	A stakeholder is a person, group, or organization with a vested interest, or stake, in the decision-making and activities of a business, organization, or project.
HTTPS	Hypertext Transfer Protocol Secure

10. References

Rediker Software. (n.d.). *Benefits of student information system*. Rediker. https://www.rediker.com/benefits-of-student-information-system#:~:text=Benefits%20for%20Students,interface%20by%20teachers%20and%20students

Edsembli. (n.d.). *10 benefits of a student management system*. Edsembli. https://www.edsembli.com/10-benefits-of-a-student-management-system/

Assignment 1 Mark Allocation Sheet

After having discussed it as a group, we recommend the following mark allocation to each group member based on contribution or lack of it throughout the assignment.

Group number __8_____

Project manager _____Toni Nam_____

Member Student ID	Percentage Contribution	Responsibilities
S11219545	100%	Use Case Diagram Structured Chart Domain Class Diagram Design Class Diagram Data Flow Diagram Sequence Diagram
S11213524	100%	Background Readings Information Procurement Introduction ERD diagram User Interfaces Use Case Diagram
S11211732	100%	Design standards Stakeholders Non-Functional Requirements Design Risk Analysis
S11198024	100%	Sequence Diagram Design Risk Analysis Functional ERD diagram User Interfaces

Signed

Member Name	ID	Signature
Shamal Prasad	S11219545	SPrasad

Toni Nam	S11213524	THN
Gwen Mar	S11211732	GMar
Adrian Obadiah	S11198024	A.Obadiah