Computer Science Department

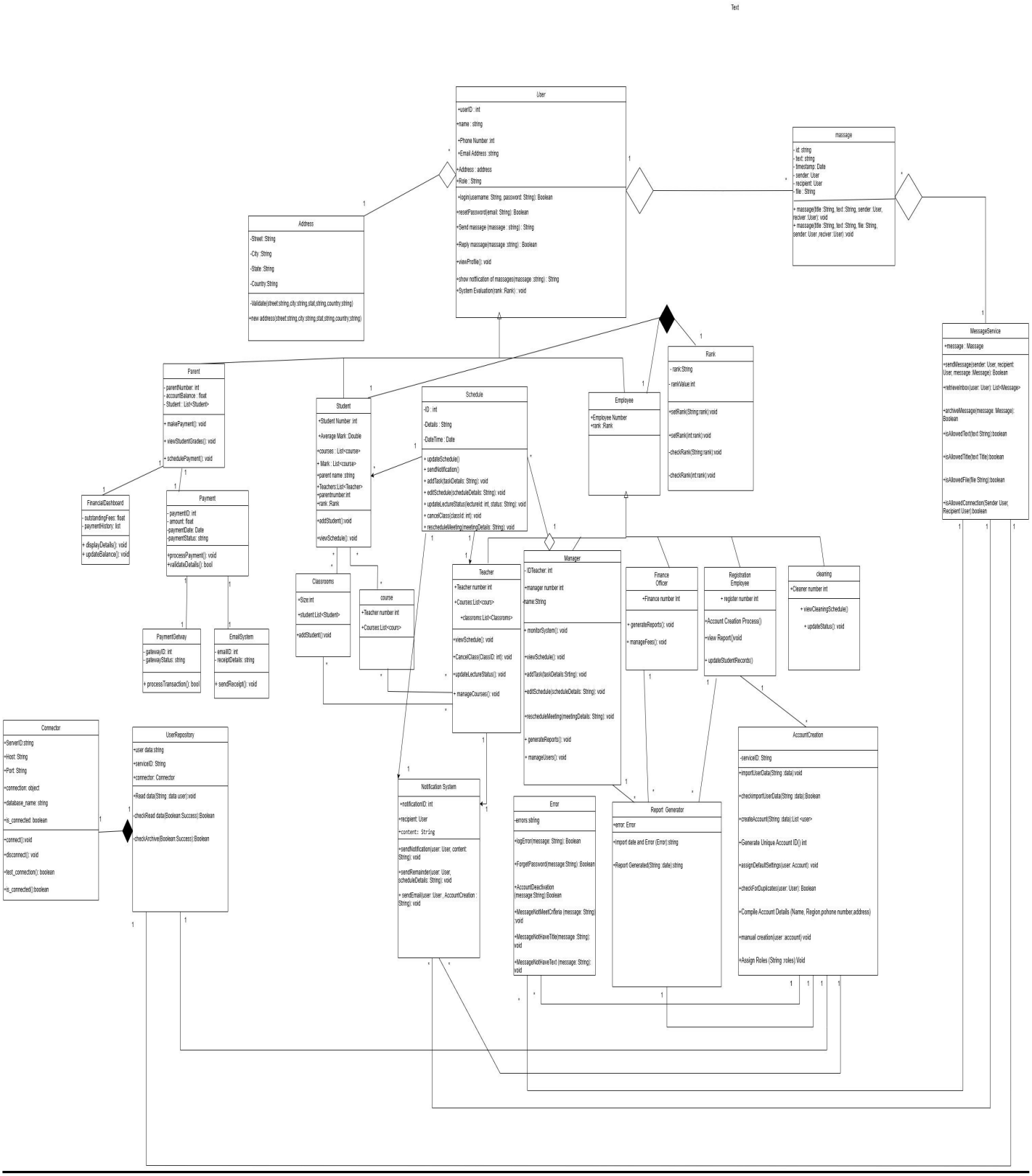
COMP433 – Group Assignment phase 4

<<Group Name: tech titans>>

Students Name	Student Number
Tariq Ladadweh	1221458
Ayman Aljamal	1220115
Ahmad Ewidat	1212596
Moath Mouadi	1210225
Mohamad Shalash	1220920

DETAILED ANALYSIS CLASS

Diagram



Describe each class:

1. Address

Purpose: Manages address details for users or entities.

2. User

Purpose: Represents the main user entity and handles authentication, communication, and system evaluation.

3. Message

Purpose: Manages communication and messaging between users.

4. Student

Purpose: Represents students and their academic and financial records.

5. Parent

Purpose: Represents parents overseeing their children’s education and payments.

6. Teacher

Purpose: Represents educators managing courses and students.

7. Classroom

Purpose: Manages the organization and composition of students in a class.

8. Schedule

Purpose: Organizes and manages events for students, teachers, and employees.

9. Employee

Purpose: Represents staff members and their assigned roles.

10. Rank

Purpose: Defines hierarchical levels within the organization.

11. Payment

Purpose: Handles payment transactions within the system.

12. NotificationSystem

Purpose: Handles sending notifications to users for various events, such as updates, payments, or system alerts.

13. Connector

Purpose: Manages system connections to external servers or databases.

14. UserRepository

Purpose: Centralized database for user information.

15. Error

Purpose: Handles error reporting and tracking.

16. ReportGenerator

Purpose: Generates reports for errors and system activities.

17. Manager

Purpose: Represents managers supervising employees and operations.

18. AccountCreation

Purpose: Manages the creation, updating, and deletion of user accounts.

19. Cleaner

Purpose: Represents personnel responsible for cleaning and maintenance.

20. PaymentGateway

Purpose: Manages secure online payment processing.

21. Finance

Purpose: Manages financial operations and reporting.

22. Registration

Purpose: Handles user or student registration in the system.

23. PaymentGateway

Purpose: Acts as an interface for secure payment processing, ensuring transactions are validated and completed.

24. Finance

Purpose: Handles the management of funds, including budgeting, expenses, and generating financial reports.

25.FinancialDashboard

Purpose: Provides a visual representation and management tool for financial data, such as balances, transactions, and reports, to monitor and analyze financial performance.

26.Course

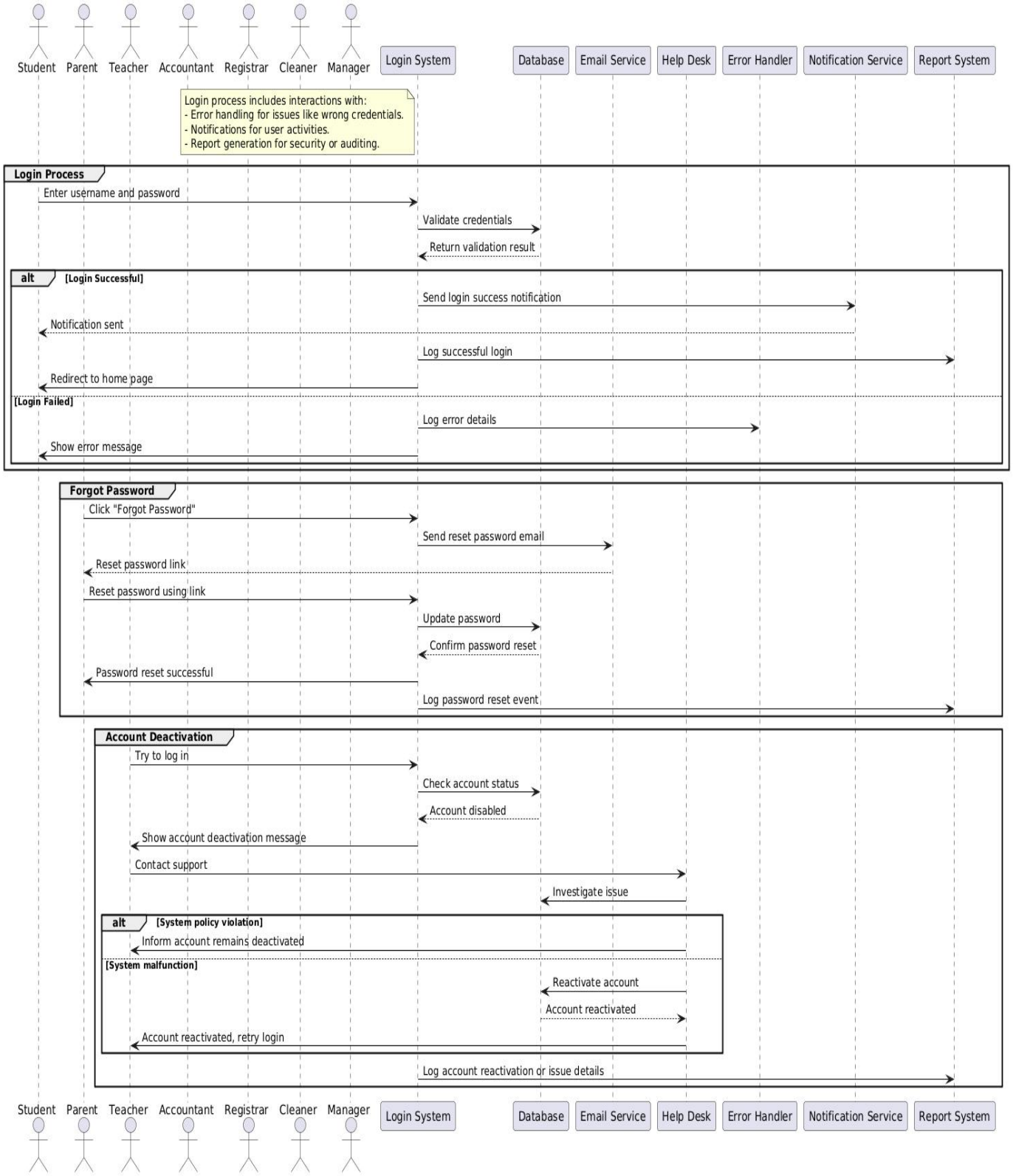
Purpose: Represents a unit of study or training provided within the system. It includes details such as course title, description, schedule, assigned teacher, and enrolled students. It serves as a central entity for managing educational content and activities.

27.MessageService

Purpose: Handles the management and delivery of messages within the system. This includes sending, receiving, forwarding, and managing messages between users or system components. It ensures reliable and efficient communication throughout the system.

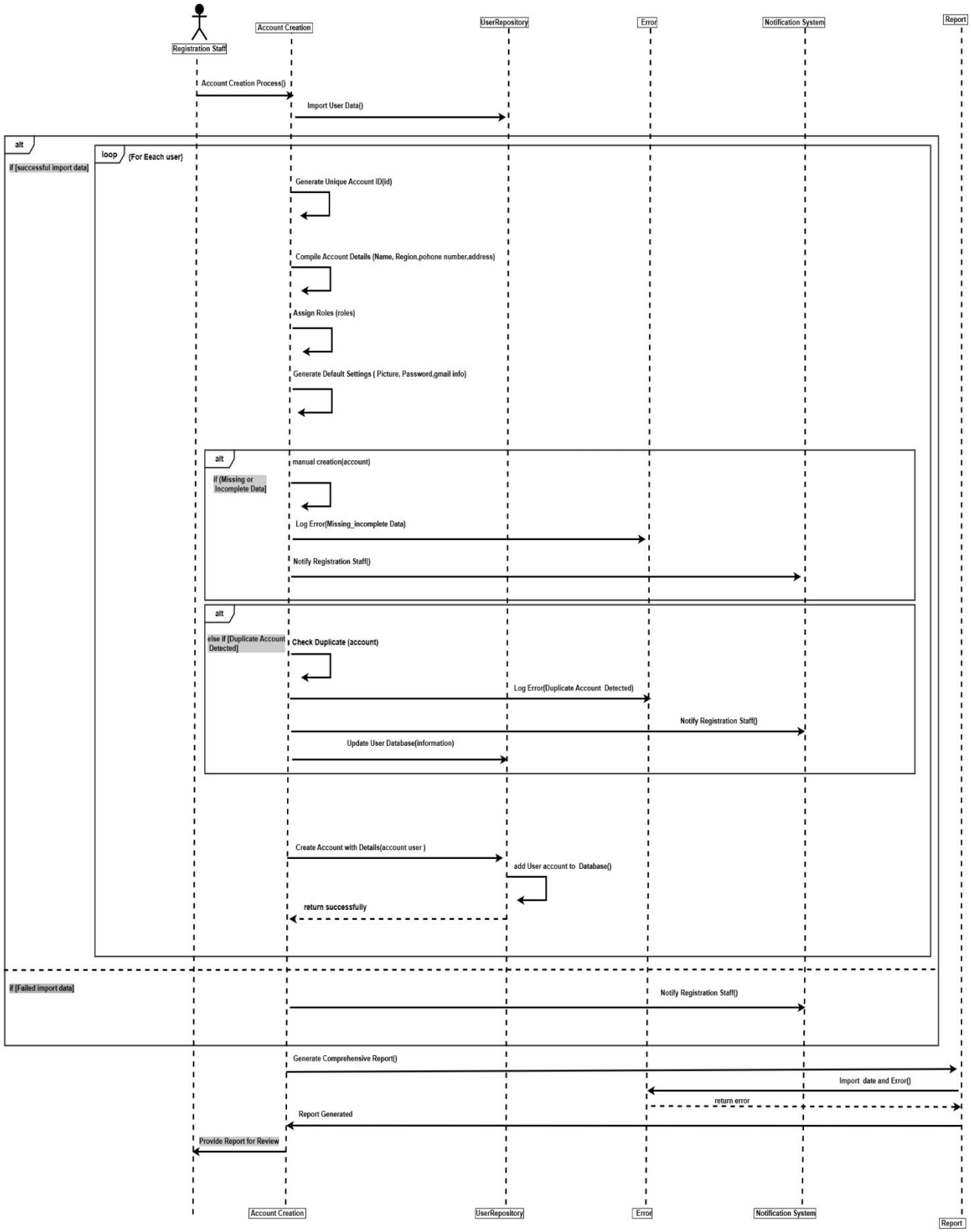
MoathMouadi 1210225

Login Sequence Diagram:



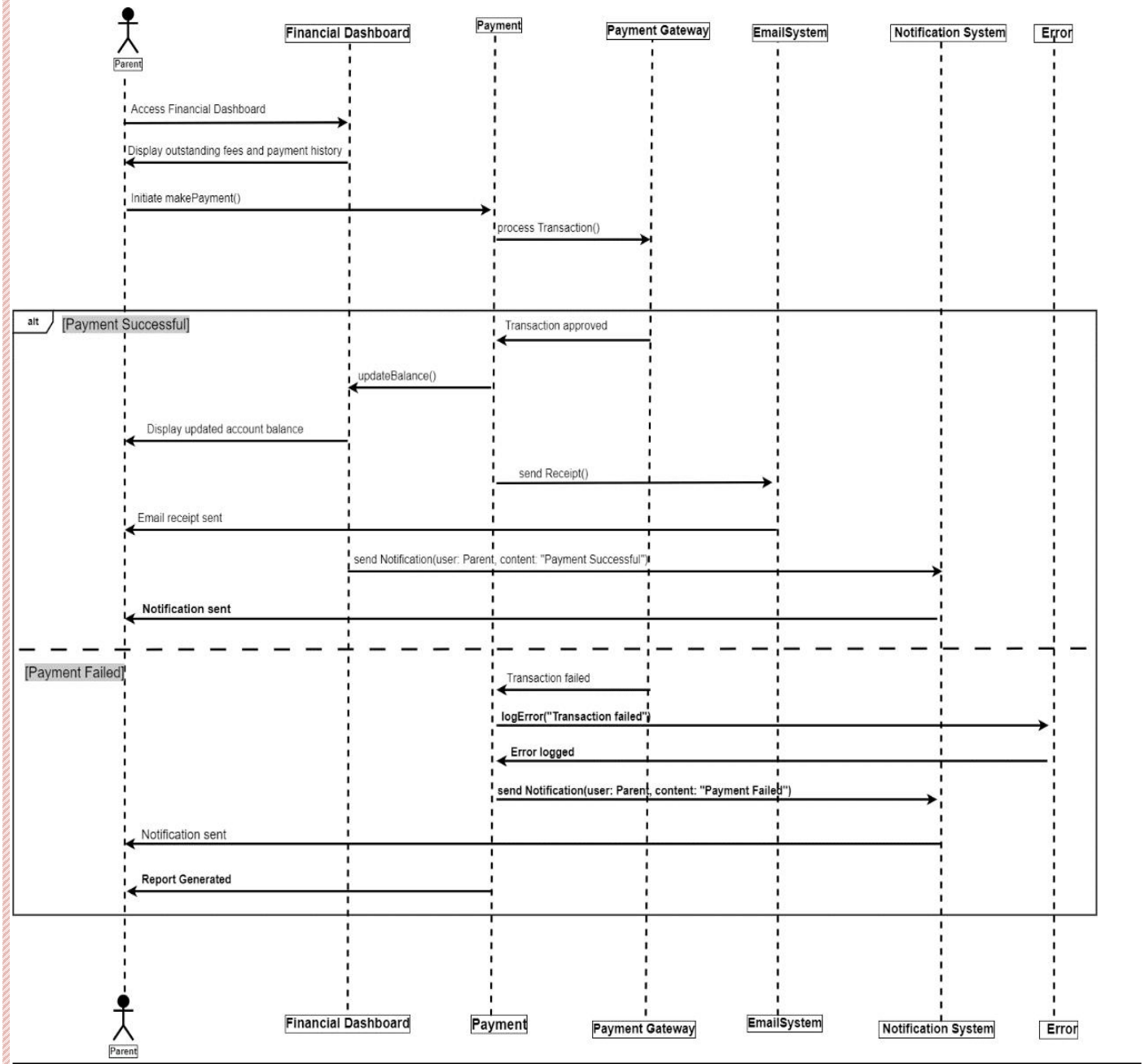
Ayman aljamal 1220115

Automatic Account Creation Sequence Diagram



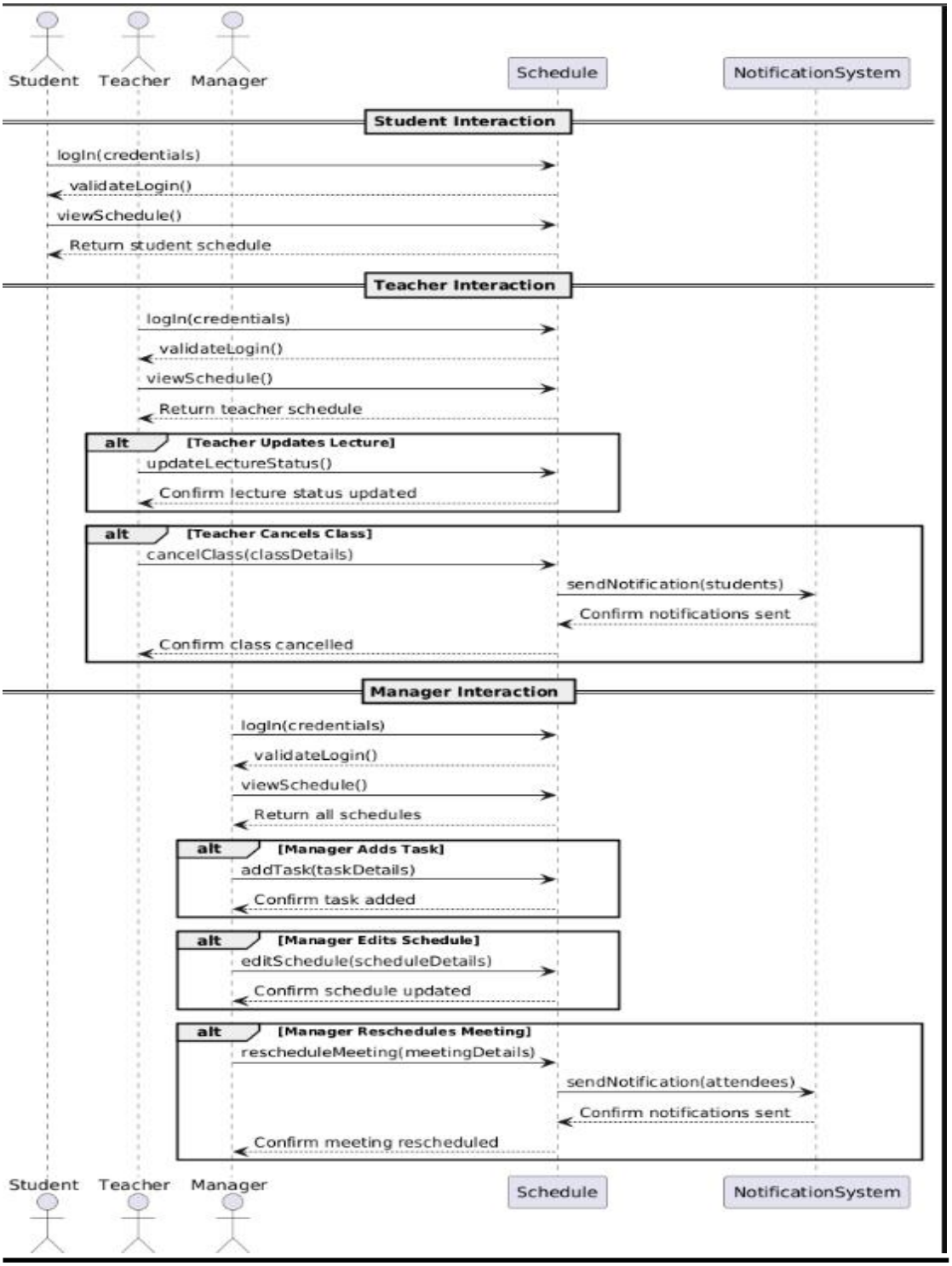
Mohamad Shalash-1220920

Financial Transaction Management Sequence Diagram :



Tariq Ladadweh-1221458

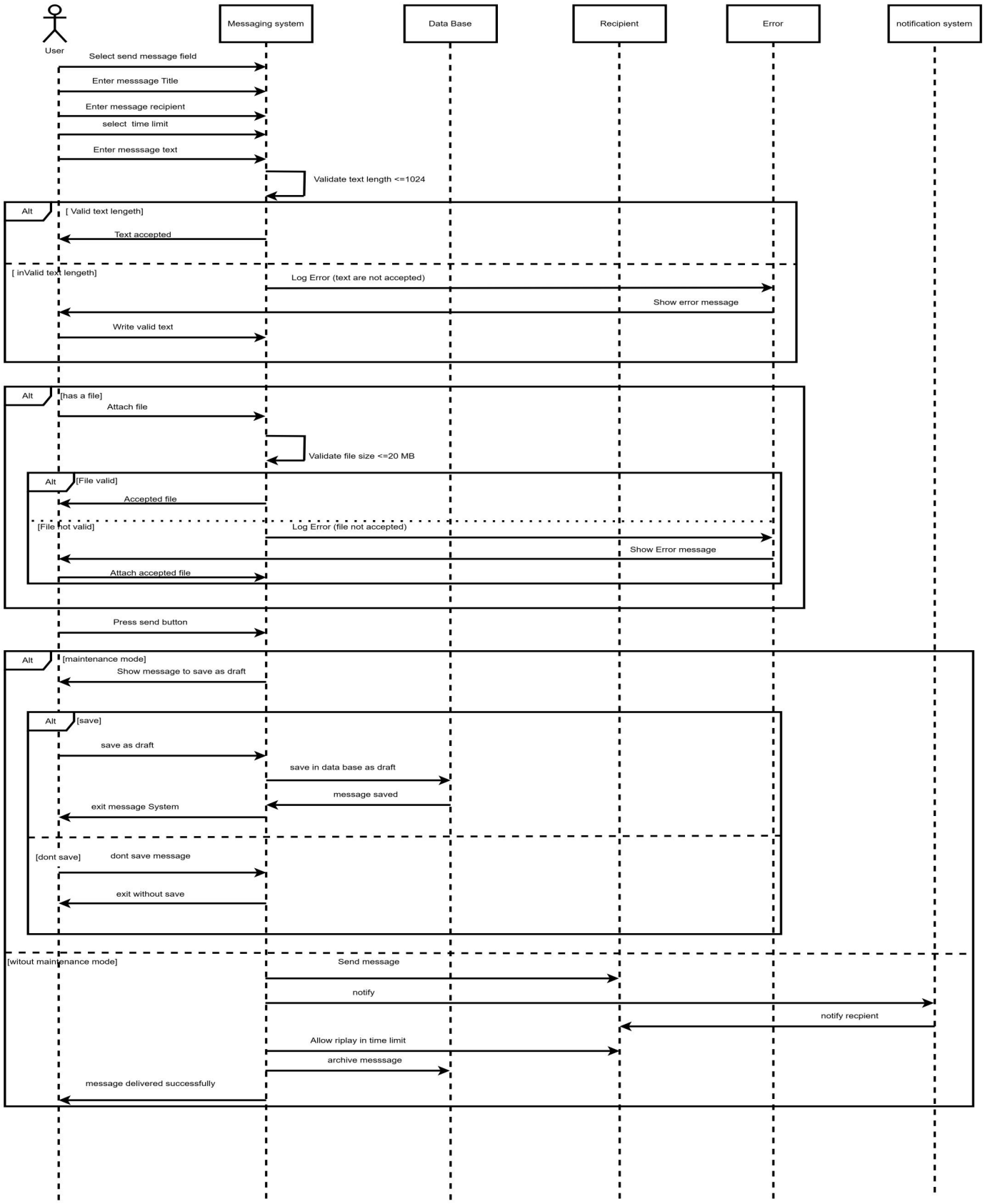
View and Manage Schedule Sequence Diagram:



e

AhmadEwidat -1212596

Send message Sequence Diagram:



Design Goals

specific system design

Reliability:

To ensure *high reliability and timeliness*, our system is built on a robust design that incorporates key principles such as redundancy, proactive monitoring, and fault tolerance. The system employs *redundant servers* to maintain continuous availability, ensuring that localized failures do not affect other components. Moreover, *real-time monitoring tools* continuously evaluate system performance, enabling the identification and resolution of potential issues before they impact users.

By implementing *proactive resolution mechanisms*, the system can handle failures gracefully. Automated failover processes seamlessly redirect traffic to backup servers, maintaining uninterrupted functionality. Additionally, robust data handling practices ensure secure and reliable processing of critical operations, with accountability guaranteed through verified timestamps.

This design delivers a dependable user experience, ensuring uninterrupted operations and consistent reliability, even in the face of unforeseen challenges.

General Design Goals

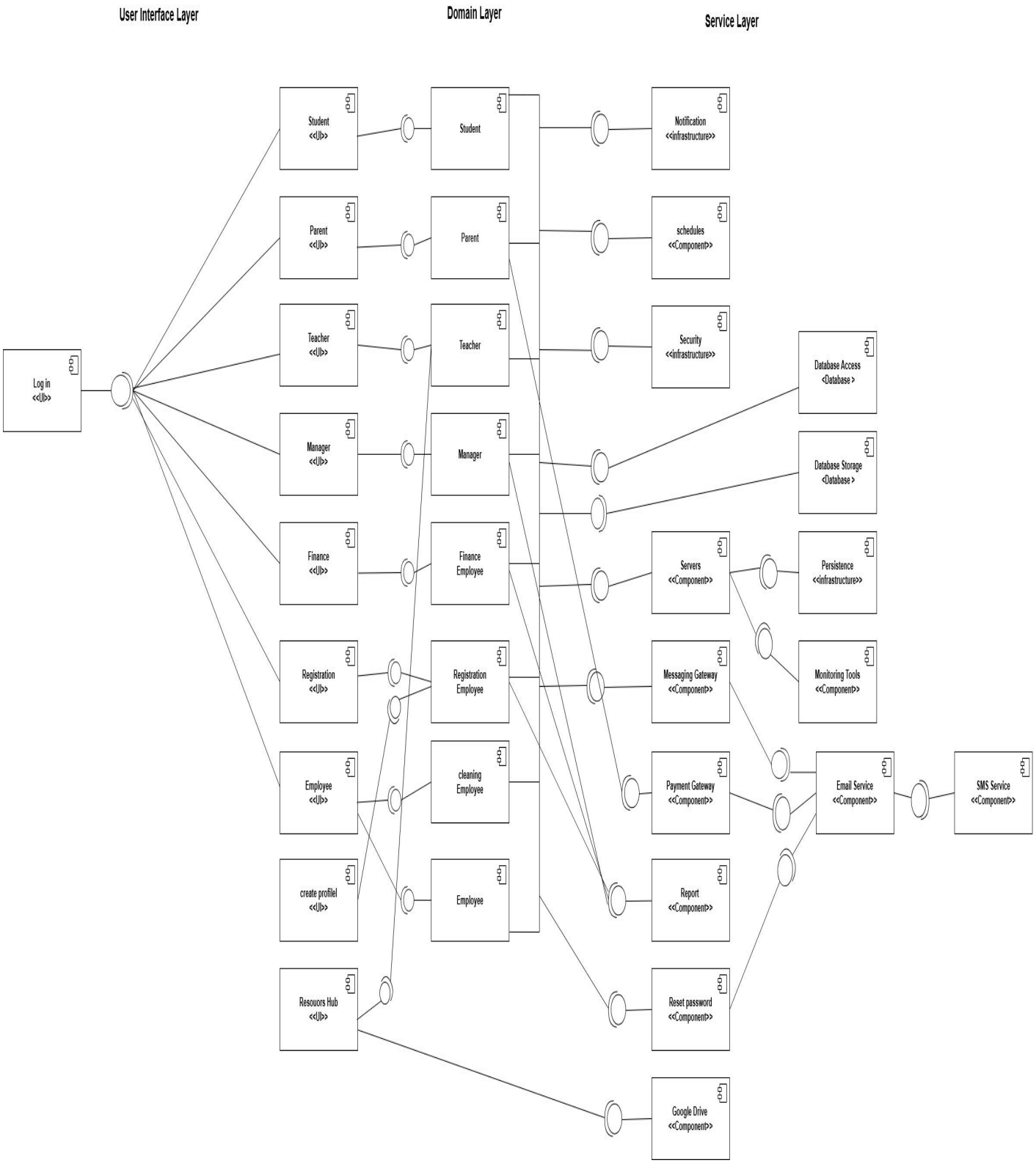
High Cohesion:

High cohesion is achieved when related functionalities are grouped within the same layer or component, ensuring a focused and modular design. For our school system, entities such as Students, Teachers, Parents, Projects, and Schedules are organized within their respective components. For example, the Teacher Component handles tasks related to lecture management, project assignments, and communication with students, while the Student Component focuses on project submissions, accessing schedules, and interacting with teachers. These entities are logically grouped within the same environment to perform interrelated tasks. Additionally, each component contains all the necessary and dependent classes and functions. This ensures that the system is modular, organized, and maintains a clear separation of responsibilities, allowing for ease of updates and debugging.

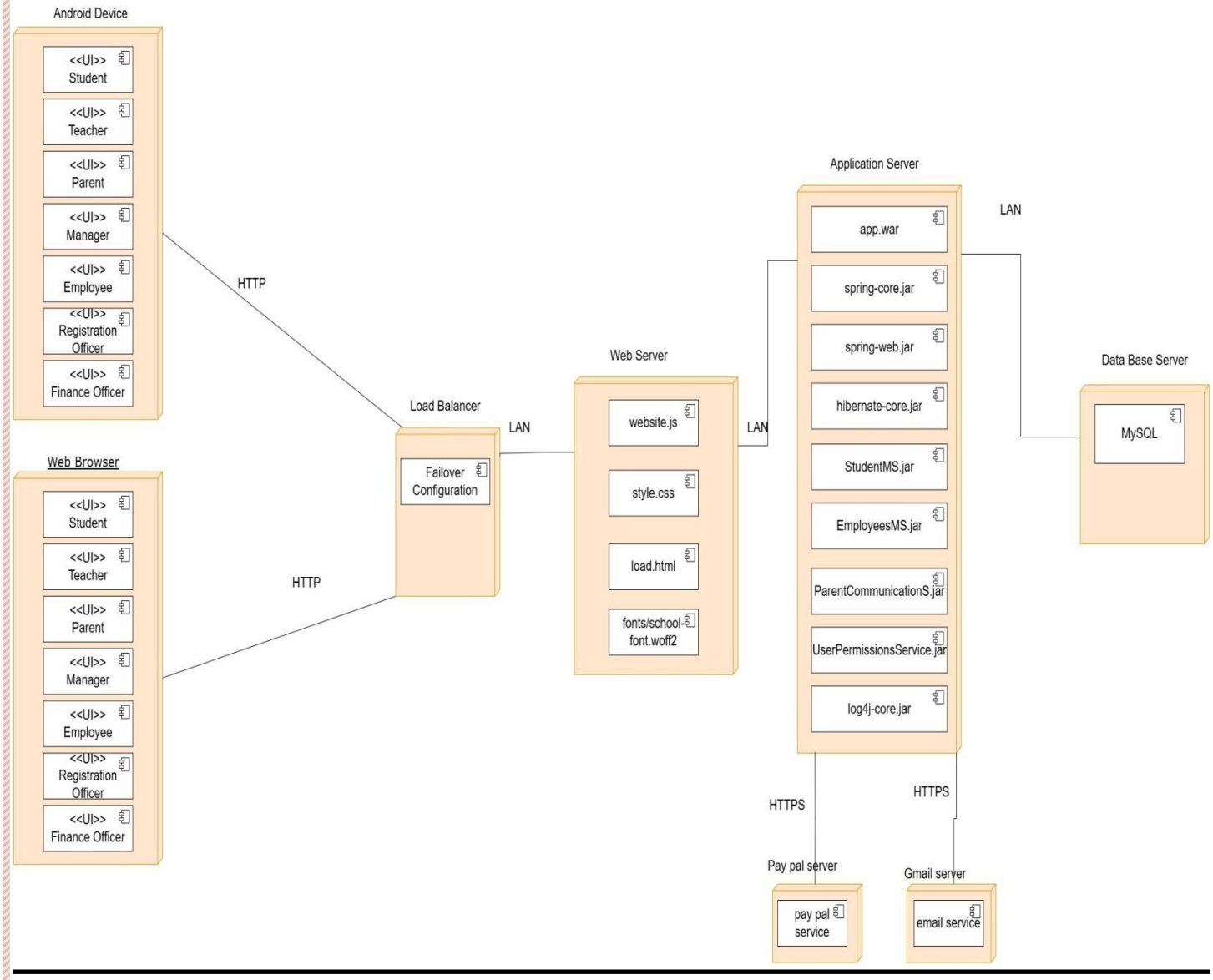
Flexibility:

Flexibility ensures that the system can adapt to changes, scale efficiently, and integrate new features or functionalities without disrupting the existing structure. In our school management system, we have adopted a three-layer architecture consisting of the User Interface Layer, the Service Layer and the Domain Layer. Each layer is designed independently, allowing for smooth integration of enhancements or adjustments. For example, the User Interface Layer can be updated to improve the user experience without affecting the underlying business logic(Domain Layer) or database(Service Layer). Similarly, adding new modules, such as advanced reporting or analytics, can be done seamlessly without impacting existing functionalities. This design approach ensures that the system remains adaptable to evolving requirements, supports future growth, and provides a long-term, sustainable solution.

UML Component diagram



UML Deployment diagram



COMP433 – Group Assignment phase 3

Moath Mouadi, 1210225

Scenario 1: User Login

Initial Assumption:

- The user has a registered account in the system and knows his/her own data

Normal:

In the beginning, the user opens the login page of the system and then he enters his username and password, and then the system verifies the validity of the values entered in the login process after verifying the values if they are correct, it directs or sends the user to the main page based on his identity, if the user is a student, it will send him to the main page related to the student, if the user is a teacher, it will send him to the teacher's main page, if the user is a manager, it will send him to the page related to the manager, and if a worker or administrative employee, it will send him to the page related to the manager....

Alternative

- If the user has forgotten their password, they must click on “Forgot Password” and follow certain procedures to verify that they are the account owner by retyping a new password.

Alternative

- If the user does not know how to do the registration process or their account is disabled under certain circumstances, the user can click on the “Help feature” and follow the existing procedures to help them resolve their issue.

Error

- If the user's input data is incorrect, the system will display a red-colored message to alert the user that there is an error in entering the “username or password”.

Other Activities:

- The system should be aware of the logins, whether they are true or false, to make the system more secure It sends a notification to the user that a failed registration process has occurred with the number of attempts, such as the Retaj website.

System State on Completion:

- It is normal for the user to log in and access the entire system, but it is not normal for the user to stay on the login page and try again

Moath Mouadi, 1210225

Use-Case: User Login

1.1 Brief Description

This user status describes secure access to the system for users by entering a usernameand password for authentication, ensuring users like students, parents, teachers can efficiently engage with the business services offered

1.2 Actor

1.2.1 students

1.2.2 parents

1.2.3 teachers

1.2.4 accountant

1.2.5 registrar

1.2.6 cleaners

1.2.7 Manager

2 .Preconditions (Entry condition)

2.1 The user must already have an existing account in the system

2.2 The system must be operational and accessible

3 .Flow of Events

3.1 Basic login flow process

3.1.1 The user is taken to the login page

3.1.2 The user enters the username and password in the appropriate place

3.1.3 The system validates the entered data which reduces the risk of unauthorized access

3.1.4 If the data entered is correct, the system will send or direct the user to the home page based on their role

3.1.5 The use case ends

4 . Error Flow

4.1.1 Forgot Password

4.1.2 If the user forgets their password, click on "Forgot Password?"

4.1.3 The system asks the user to enter their email address

4.1.4 The system sends a message to your email address to reset the password

4.1.5 The user resets their password and then returns to the registration page to try again

4.2 Account deactivation

4.2.1 If the user's account is disabled, they will see a message that the account is disabled, for details click on "Help "

4.2.2 The user will then contact the support team via email to find out why the account was disabled

4.2.3 The helpdesk investigates the cause of the disruption

4.2.4 If the cause is due to an error or system malfunction, the account will be

reactivated

4.2.5 However, if it violates the system's policies, the account remains disabled

and must create a new account

5 .Post-conditions (Exit condition)

5.1Login successfully: User access to the system

5.2 Login Failed: The user remains on the registration page and can try again

6 .Special Requirements

6.1 Password Encryption

6.2 Record the number of valid and failed logins for security monitoring

7 .Extension Points

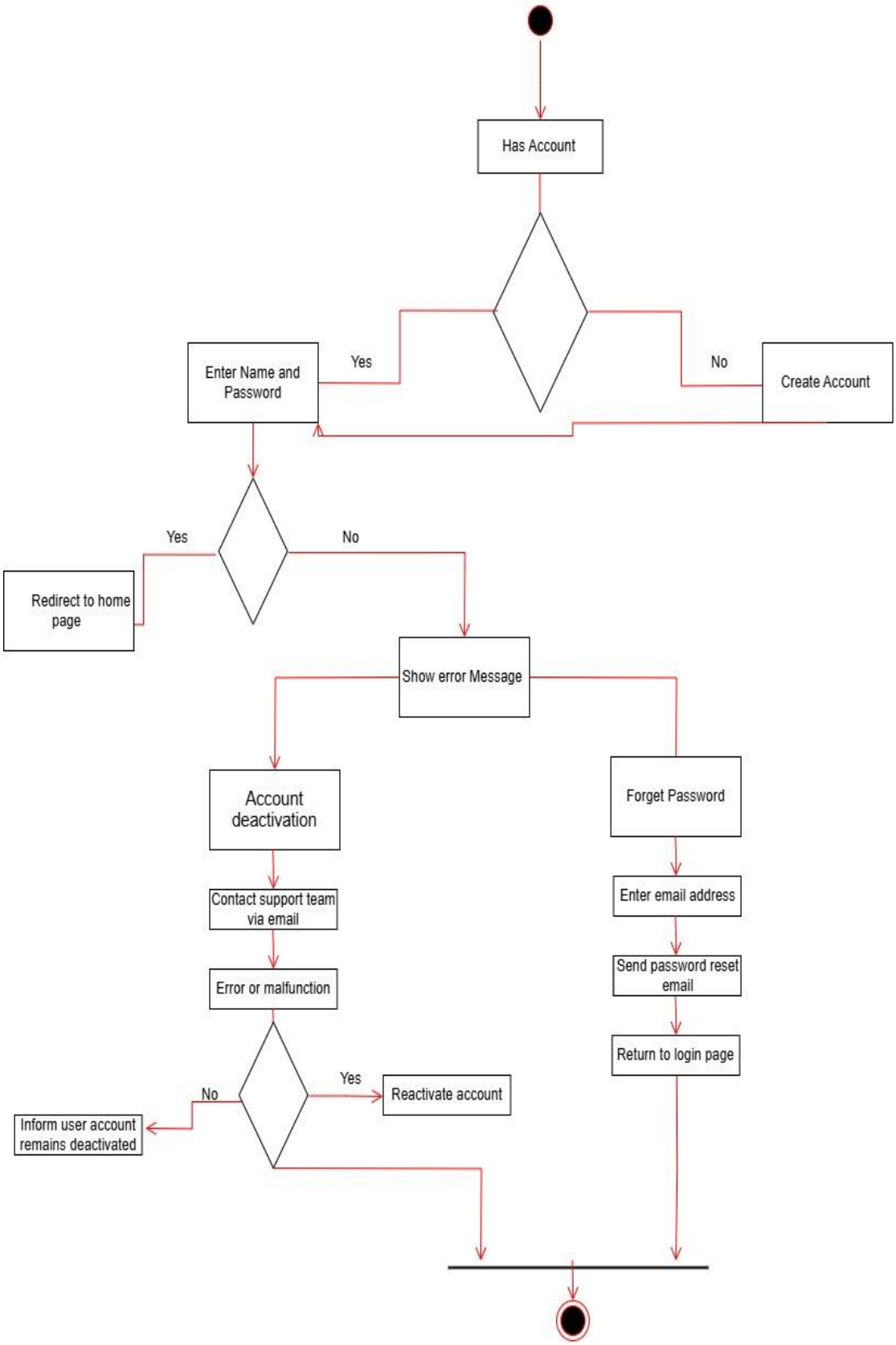
7.1 Perform an authentication process after logging in to make sure the user is human

7.2 Ensures users can log out of other devices, enhancing control over access points

7.3 Sign in with your facial recognition after memorizing your username and password

User Login Activity Diagram

Moath Mouadi, 1210225



Ayman Aljamal,1220115

Scenario 2: Automatic Account Creation

Initial Assumption:

The school has collected complete and accurate data for all targeted users, including students, parents, teachers, general staff, financial staff, registration staff, and the principal, regardless of their employment status (part-time, permanent, or non-permanent). Based on this data, the system automatically creates user accounts.

Normal Flow:

1. The system imports user data from the school's registration , ensuring that all relevant information for users such as students, teachers, parents, and staff… is complete and accurate.
2. For each user (whether a student, teacher, general staff member, financial staff, registration staff, or Manager), the system automatically performs the following actions:
   2.1 The system generates a unique account ID for each user, ensuring no duplicates.
   2.2 The account includes relevant details, such as the user's name, region, and, for students, the associated parent's information.
   2.3 The system assigns specific roles to each user based on the imported data, such as student, parent, teacher, general staff (janitors, guides, assistants), financial staff, registration staff, or Manager.

   2.4 The system generates default account settings for each user, such as a default profile picture and a temporary password. These can be modified by the user upon their first login.
3. The system updates the user database to reflect all created accounts, ensuring consistency, completeness, and that all required information is accurately recorded.
4. The process concludes when all valid user accounts are created successfully, with all necessary data captured and stored in the system.

Alternative Flow:

1. If the user already has an existing account, the system will retain the old account, update it with the new information, and save the updated data in the database.
2. If any required information is missing during account creation, the system will place the account in manual creation mode for further processing.

Error Flow:

- If the data retrieval process fails, or if the data provided is incomplete or incompatible with the operating system, the system will notify the responsible registration personnel about the error, allowing them to correct it and log the issue for reporting purposes.
- If the system detects invalid or duplicate information, it will flag the data and log it for inclusion in a report.
- If the system detects the required information is missing, it will the data log it for inclusion in a report.

Other Activities:

- Assigning user roles (students, parents, teachers, general staff, financial staff, registration staff, and the Manager) and permissions during account creation.
- Logging errors in a report during the data retrieval process for account creation.

- Create default data for secondary fields, such as contact details, address, and photo. Assigning missing user accounts to manually created accounts.
- Verifying that all required fields are completed correctly before account creation.
- Ensuring that all data is consistent and aligned with system requirements during account updates.
- Maintaining a backup of all account creation logs for auditing purposes.
- Notifying responsible personnel of any discrepancies detected in the account creation process.

System State on Completion:

- The system will have successfully created all user accounts, ensuring data integrity and accuracy.
- All database information will be preserved without loss, and any modifications to existing accounts will be reflected with updated data.
- A comprehensive report will be generated, detailing:
  - All errors encountered during the account creation process.
  - Accounts that were processed manually due to missing or incorrect data.
  - Accounts that existed previously and were updated with new information.
  - All duplicate accounts data
  - Accounts that were not created due to missing important data
- The system will ensure that all accounts are accurately reflected in the database.

Ayman Aljamal,1220115

Use-Case: Automatic Account Creation

1. Brief Description

This case describes how the school's automatic account creation mechanism generates user accounts for all registered individual students' students, staff, and parents based on data retrieved from the school's database. It automatically assigns roles and default settings, then logs or flags any errors (such as missing data or duplicates) for the Registration Staff to handle.

1.2 Actor

1.2.1 registration staff

2. Preconditions (Entry condition)

2.1 The **school's database** is online and accessible to the account creation system.
2.2 The **registration staff** has the necessary credentials/permissions to run or schedule the automatic account creation process.
2.3 The system's **configuration** allows it to connect to and read from the school's database without errors.

3.Flow of Events
3.1 Basic Flow - Automatic Account Creation

3.1.1 The Registration Staff start (or schedules) the automatic account creation process through the system's interface.

3.1.2 The system **imports user data** (students, staff, parents, etc.) from the school's database.

3.1.3 The system **verifies** that the imported data is properly structured and attempts to **detect** any immediate format errors or missing fields.

3.1.4 If data is **incorrect** or cannot be parsed at all, the system **notifies** the Registration Staff

3.1 **For each user entry:**
3.2.1 The system **generates a unique account ID**.
3.2.2 The system **creates an account** with all relevant information (name, role, …).
3.2.3 The system **assigns roles** (student, teacher, parent, staff, registration staff, or manager) based on imported data and **applies default account settings**.
3.3 The system **checks for errors** across the newly created accounts:
3.3.1 If **no errors** (duplicates, missing info) are found, the system proceeds.
3.3.2 If **errors** are found, the system follows the relevant alternative flow (duplicates or missing data).
3.3.3 The system **updates** the user database with the **newly created accounts**, ensuring all required account information is accurately recorded.
3.3.4 Once all valid user accounts have been created, the **use case ends** successfully.

5. Alternative Flows

5.1 Failed Import

5.2.1 If the system fails to import data due to connectivity issues or major structural errors in the database:

5.2.2 The system logs the failure in a comprehensive error report.

5.2.3 The system notifies the Registration Staff of the failure and provides details for troubleshooting.

5.2 Duplicate Accounts Detected

5.2.1 If the system detects that a user's entry already exists or there is a duplicate record:

5.2.2 The system skips the creation of a new account for the duplicate entry.
5.2.3 The system logs the duplicate record in a report.
5.2.4 The system notifies the Registration Staff to review and resolve the duplicate issue.

5.3 Missing Data

5.3.1 If the system identifies missing or invalid critical information (e.g., name, ID, or role):
5.3.2 The system moves the user's account to a Manual Creation Queue (partial account creation).
5.3.3 The system logs the missing data issue in a report.
5.3.4 The system notifies the Registration Staff to complete or correct the user's information.

6. Postconditions (Exit Conditions)

1. All valid user accounts are created, with correct data (name, ID, and role) assigned and stored in the system.
2. Errors or incomplete data are logged and flagged for review, with details on issues like missing or incorrect information, duplicates.
3. Once all accounts are processed (either created or flagged), the system exits the account creation process, and the Registration Staff is aware of any remaining manual tasks.

6.Special Requirements
6.1 Ensure account data integrity and prevent duplication.
6.2 Implement secure storage of account credentials and encryption of temporary passwords.

6.3 In the absence of missing data

6.4 Within creating an account within the existing data and creating a temporary password that must be changed

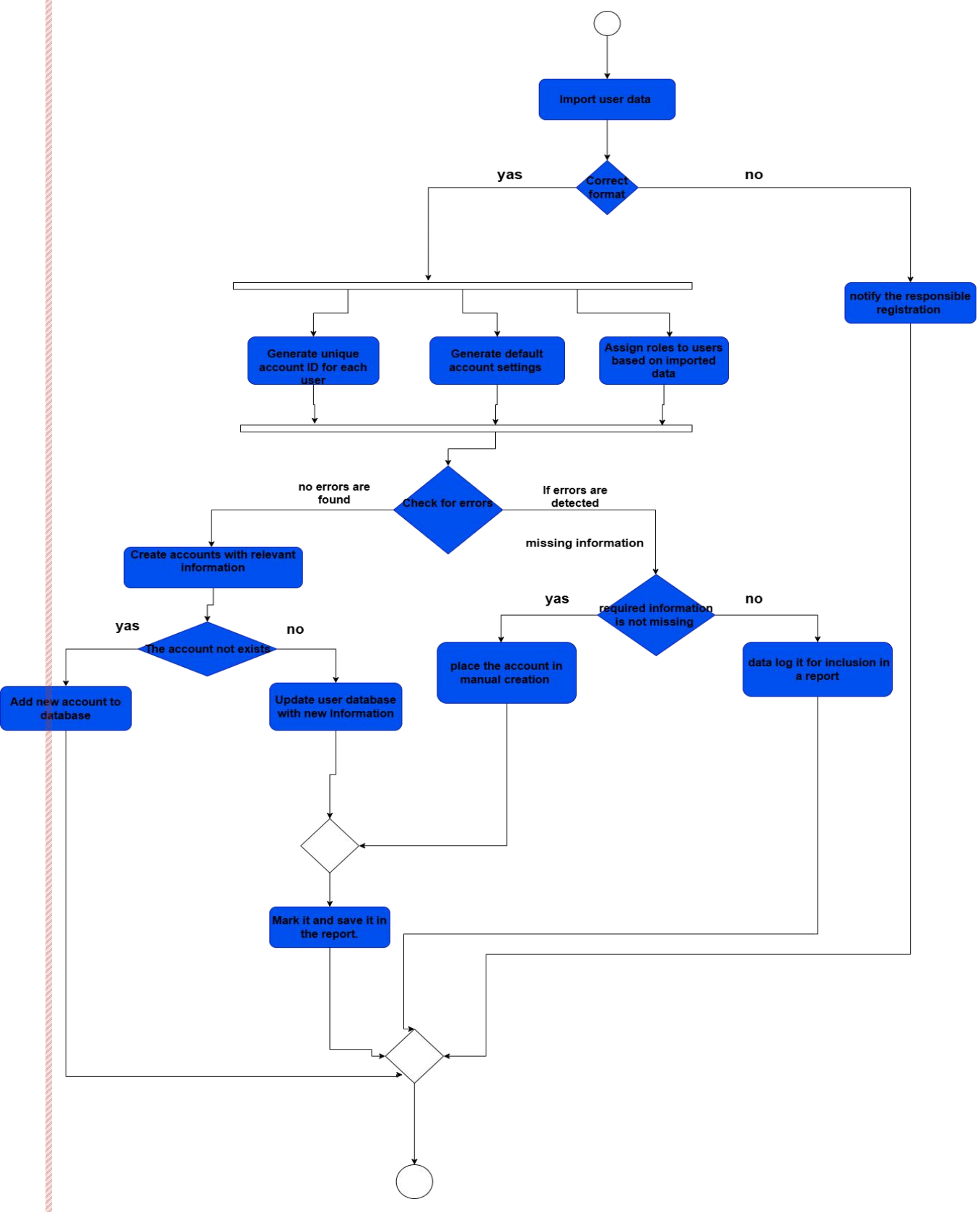
6.5 Within the preservation of accounts after their creation at the end of each account creation process

7.Extension Points

- Once the automatic account creation process is finalized, the system performs a comprehensive verification to confirm that all valid user accounts have been successfully created. This includes ensuring that all essential data (name, ID, and role) has been accurately assigned and securely stored in the system database.
- For any detected issues, such as incomplete or invalid data, the system generates detailed error logs. These logs are categorized and flagged for review by the registration staff, highlighting specific issues (missing fields, duplicate entries, or inconsistencies) associated with each affected account.
- The extension process allows for seamless integration with follow-up actions, such as manual account corrections, additional data verification, or reprocessing of flagged entries to ensure the system remains consistent and operational.

Automatic Account Creation Activity Diagram

Ayman Aljamal,1220115



Mohamad Shalash – 1220920

Scenario 3: Financial Transaction Management

Initial Assumption:

- For the Financial Transaction Management scenario, it is assumed that the parent has valid login credentials and access to an active account with accurate financial records displayed on their dashboard. The outstanding fees or dues are correctly reflected, and the parent has a valid payment method, such as a credit card or bank account, supported by the system. The system and payment gateway are fully operational, ensuring seamless processing. Additionally, communication channels, like the parent's registered email, are functional for sending payment confirmations and receipts. Robust security protocols are in place to safeguard sensitive financial data, and the payment method has sufficient funds to complete the transaction successfully.

Normal Flow:

- Login in the system:
  - The parent logs into the system using valid credentials.
- Go to the Payment Section:
  - The parent scans through the system and clicks on the "Financial Dashboard" link to find the outstanding fees.
- Paying Method:
  - The parent clicks on the "Pay Now" button and automatically goes to the payment page.
- Entering Payment Information:
  - The parent supplies the necessary information to the credit card including card number, the date of expiry of the card and the CVV.
- Payment Wrapper:
  - The parent inspects the payment details and presses 'Confirm' in order for the payment to commence.
- System Processes Payment:
  - The system successfully and safely carries out the payment transaction through the payment gateway.
- Confirmation Message:
  - After completing the payment, the page displays a congratulatory message confirming that the transaction was successful and the new balance together with the financial dashboard is also updated.
- Printing of Receipts:
  - The system prints the receipts and emails them to the parent's email account that is provided in the system.

Alternate Flows :

Alternate Flow 1 (Partial Payment) :

- The parent chooses the "Pay Partially" option rather than opting for full payment.
- They fill in the partial amount and click on the confirm button for the payment to be made.
- The system reconfigures the new outstanding amount and refreshes the dashboard.

Alternate Flow 2 (Schedule Payment):

- The parent clicks to make a payment through a scheduled option.
- Then they create the schedule and confirm the payment specifics.
- The system sets up the payment to be made automatically through notification or deduction.

Error Flows:

Error Flow 1 (Invalid Payment Details):

Incorrect card details are handed by the parent.

The system gives out an error message “Incorrect Card Information”.

The parents can try once more or select a different method.

Error Flow 2 (Payment Gateway Issue):

The payment gateway is presently not reachable.

The system sends out a failure message “Payment Failed. Please Try Again”

The parent is suggested to wait and try to pay the payment once more.

Other Activities:

1- **Email Notification Generation**
   - Triggering the parent payment confirmation email with a receipt and transaction information is enabled systemotorically.
2- **Ledger Update**
   - The Parents ledger in the Financial Records Database gets a ledger update automatically and reflects any payment made along with balance after the payment.
3- **Parent’s Payment History Update**
   - The finished transaction is now attached into parents payment history in “Financial Dashboard” of the parent.
4- **Budget Adjustment for School Accounts**
   - The system makes an entry in the financial record of the school accounts showing the crediting of the payment made.
5- **Scheduled Payment Check**
   - If the parent has a scheduled payment, the system re-schedules the next payment and due date with the latest transaction.
6- **Notification to Financial Officer**
   - To the financial officer who are in charge of transactions, record summary in the form of reports, emails, or printed copies are sent.

System State on Completion:

1- **Updated Outstanding Balance**
   - If available, the new outstanding balance for the parent is shown in the financial board that has been altered.
2- **Receipt Archive**
   - The system incorporates a receipt generated in the course of the transaction for storage in the databank.
3- **Transaction Record**
   - The payment is documented as a successful transaction providing information such as:
     - ❖ Transaction ID
     - ❖ Date and time
     - ❖ Amount paid

- ❖ Method of payment

4- Security Lock on Transaction Details

- In order to enhance data security only the authorized personnel are allowed to view the transaction details.

5- Ready State for Next Transaction

- The system remains in a passive state waiting for additional steps by the parent and/or other users.

Mohamad Shalash - 1220920

Use-Case: Financial Transaction Management

1.1 Brief Description

This business use case describes the process of carrying out financial transactions so that parents can check their dues and make payments in a safe manner through the system. The parent goes to their Financial Dashboard, chooses the appropriate payment option, fills in the payment details, and presses the Pay button. Once the secure payment is done successfully, the system manages the rest of the processes including changes to the outstanding balance, maintains the transactions history, and – most importantly – sends the receipt through email.

1.2 Actor

1.2.1 Primary Actor:

- Parents – who make most of the transactions and are required to start and complete the payment process.

%.2.2 Supporting Actors:

- FinancialDashboard: The interface displaying outstanding fees, payment options, and history.
- Payment: The subsystem that captures, validates, and processes payments. PaymentGateway: The entity that handles secure payments to ensure safety.
- EmailSystem: Handles the sending of email receipts upon successful transactions.
- NotificationSystem: Notifies parents of transaction statuses such as payment failure.
- Error: Manages and displays error messages when validation or processing issues occur.

2. Preconditions (Entry Condition)

2.1 The parents should have signed in to their system accounts and they should be active.
2.2 The system should be able to show all outstanding fees or dues in the financial dashboard.
2.3 At all times, the payment gateway must be available.
2.4 All systems mean of communication such as email should be in working order.

3. Flow of Events

3.1 Basic Flow - Payment Processing

3.1.1 The first step in the process is logging into an account where there is an option titled "Financial Dashboard".

3.1.2 In this step, the system presents cookies and outstanding fees, payment history and different payment alternatives.

3.1.3 The parent proceeds to the payment page by selecting the "Pay Now" button.

3.1.4 The parent fills in the card details bearing the card number, the expiry date, and the CVV.

3.1.5 It is now the turn of the parent to order the system to submit the payment by clicking on the button labelled "Submit".

3.1.6 Thus, the steps logically follow each other, and the parent has a seamless experience in making the payments where the system captures the payment details, verifies and processes the transaction through a payment gateway.

3.1.7 A confirmation receipt appears, and the system adjusts the parent's balance owed and records the overall transaction.

3.1.8 The last step is finished now, receiving an email receipt from the system. The parent has been dispatched the receipt by the system and the email purports the receipt.

3.1.9 The use case ends.

4. Alternative Flows

4.1 Partial Payment

- ❖ The parent opts to chem the "Pay Partially" tab.
- ❖ In this case, there is no need for the parent to settle the whole amount in the first instance.
- ❖ The system performs Partial Payment, reason being that a subtractive charge needs to be made, and the remainder which needs to be paid is set accordingly.

4.2 Scheduled Payment

- ❖ The scheduled payment of the amount fixed on specific dates is arranged by the parent.
- ❖ After the schedule is created, the system properly keeps and displays the related upcoming payments (in terms of due dates) on the financial dashboard.

4.3 Payment Details Error

- ❖ If a payment is scheduled and the parent provides the CVV or the date of the card which is already outdated, the payment would be turned down as the details provided are not adequate.

- ❖ An error message is displayed requiring the parent to provide further information so that they can start making the necessary transactions.

4.4 Payment Gateway Failure

- ❖ The system stays inactive at the start, notifying the parent regarding the transaction failure after payment, as the payment gateway is down or the transaction fails.

5. Post-conditions (Exit Condition)

5.1 The outstanding balance in the financial dashboard is updated.
5.2 A receipt is created and sent via email to the parent as proof of payment.
5.3 The particulars of the transaction are recorded in the system in the financial records.

6. Special Requirements

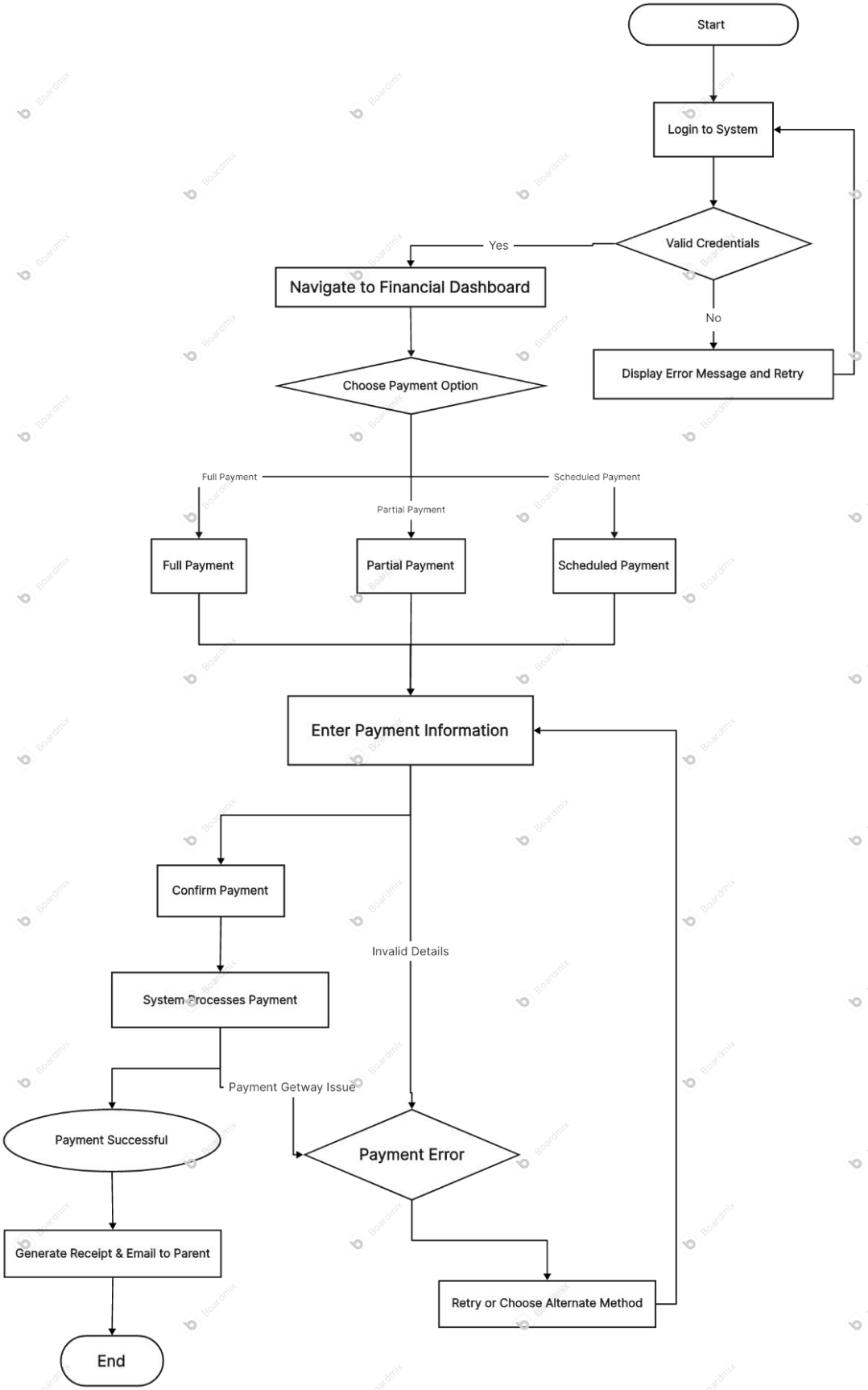
6.1 The payment system must comply with PCI DSS requirements of securing the transactions.
6.2 The system should provide end to end security for sending information regarding the payment.
6.3 The user interface must be self-explanatory so that parents can follow the process of making the payments easily.
6.4 The receipt will include completion of the transaction and the transaction id, date of the transaction, amount and the balance.

7. Extension Points

7.1 Adding More Payment Methods: Other methods of payment like mobile wallets or bank transfer may be integrated to the system.
7.2 Notifications Preferences: The parent could opt to receive SMS notifications for payment confirmation, in addition to email receipts.
7.3 Multi-currency Support: The system may be expanded to include various and more currencies for foreign end users.

Financial Transaction Management Activity Diagram

Mohamad Shalash – 1220920



Tariq Ladadweh 1221458

Scenario: Schedule Management

Initial Assumption:

- The schedule management module is fully operational, and users have credentials to access their accounts.
- Schedules for classes, meetings, and tasks are preloaded into the system and ready for updates or modifications.

Normal Flow:

1. **Student Interaction:**
    - The student logs into the system to view their weekly schedule, which includes class timings, teacher names, and room numbers.
    - They can check for any changes such as rescheduled or canceled classes.
2. **Teacher Interaction:**
    - The teacher logs in to review their class schedules and update the status of lectures (e.g., completed, canceled, or rescheduled).
3. **Manager Interaction:**
    - The manager logs in to monitor and manage schedules for tasks, meetings, and events.
    - They can add, modify, or remove scheduled items as required.

Each interaction above represents a distinct user flow that caters to specific roles.

Alternative Flows:

1. **Student:**
    - A student receives reminders for upcoming classes or changes to their schedule via notifications.
2. **Teacher:**
    - A teacher cancels a class and informs the system, prompting it to notify affected students and update schedules automatically.
3. **Manager:**
    - The manager reschedules a meeting, which triggers notifications to all attendees with the updated details.

Error Flow:

1. Missing or incorrect schedule data triggers an error flag, notifying relevant personnel to resolve the issue.
2. Unauthorized attempts to access or alter schedules are logged and blocked.
3. Synchronization issues with external systems send alerts to the IT team for prompt resolution.

System State on Completion:

- All users have accurate and updated schedules.

- Notifications of changes are delivered successfully.
- Logs of actions and updates are maintained for auditing purposes.

Use Case: View and Manage Schedule

1. Brief Description:

This use case outlines how students, teachers, and managers interact with the system to view, update, and manage their individual schedules for classes, meetings, and tasks.

2. Actors:

- **Primary:** Students, Teachers, Managers

3. Preconditions (Entry Condition):

1. Users must log in with valid credentials.
2. The schedule database must be up-to-date.

4. Flow of Events:

Basic Flow:

1. **Students:**
    - Log in to access their weekly schedules, including details such as timings, teacher names, and room numbers.
2. **Teachers:**
    - Log in to view their schedules.
    - Update lecture statuses and information.
3. **Managers:**
    - Log in to view their schedules, manage schedules, add or edit tasks, and ensure all schedules are accurate.

Alternative Flows:

1. **Manager Rescheduling:**
    - A manager reschedules a meeting and the system sends notifications to attendees.
2. **Teacher Cancellation:**
    - A teacher cancels a class, and the system updates affected schedules while notifying students.
3. **Student Reminders:**
    - The system sends reminders to students about schedule changes or upcoming classes.

5. Post-Conditions (Exit Condition):

1. Schedules are updated and valid for all users.
2. Notifications about changes are sent successfully.
3. Records are logged for auditing purposes.

6. Special Requirements:

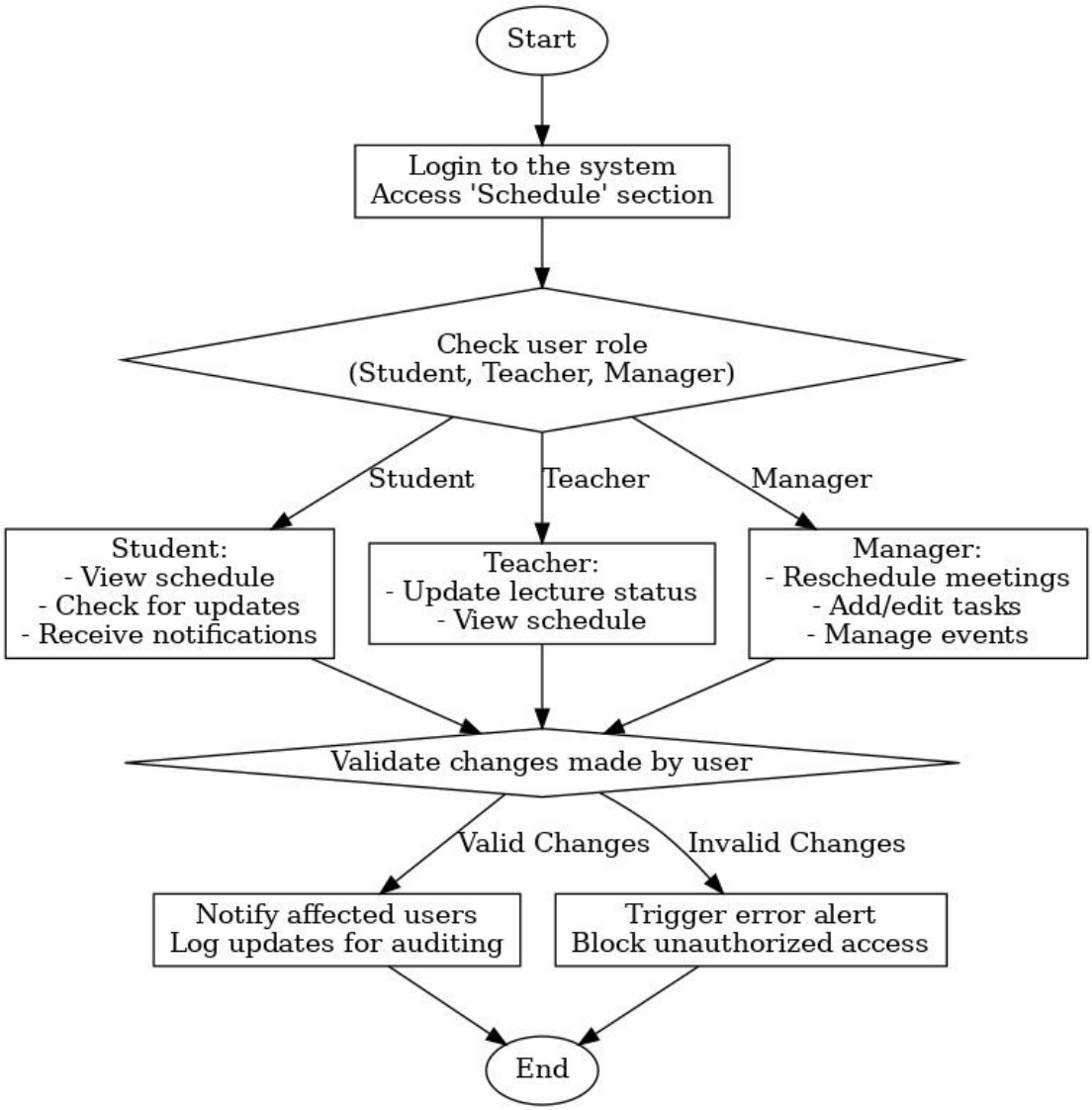
1. The system must enforce role-based access control to prevent unauthorized changes.
2. Notifications should be timely and reliable for affected users.
3. The interface should support easy and intuitive schedule updates.

7. Extension Points:

1. Integration with messaging platforms for instant schedule updates.
2. Enhanced visualization tools for managing complex schedules.

View and manage schedule Activity Diagram

Tariq Ladadweh 1221458



Ahmad ewidat ,1212596

Scenario : Sending messages

Initial Assumption:

The system allows the all actors to send a message containing a mandatory title of no more than 24 characters, a mandatory text of no more than 1024 words, and an optional file of no more than 20 MB to the specified department, student, or their parents.

Normal Flow:

After logging into the actor's account, the actor clicks on the send message box, then writes the message title, then chooses the recipient , then writes the text, and the system checks the length of the text (up to 1024 words), then clicks on the send button and the message appears in the sender's outbox and in the recipient's inbox, allowing the recipient to reply to the message in the replies box.

Alternative Flow:

The actor attaches a file in addition to the text. The system verifies the file size (up to 20 MB), then sends the message normally and the recipient can download the file.

The actor determines the time period for responding to the message or completely prevents responses. The recipient can respond within the specified time period, and after the period ends, he is not allowed to respond to the message.

An actor attempts to send a message during a scheduled system maintenance period The system displays a notification: "The messaging service is currently undergoing. maintenance. You can save your message as a draft and send it once the system is operational." so the actor saves the message as a draft and receives a confirmation.Once the system maintenance is completed, the actor logs back in, retrieves the saved draft, and sends the message.

The actor returns or exits the system, and the message is saved as a draft or ignored via a confirmation message that appears to the actor.

Error Flow:

The actor attaches a file larger than 20 MB, the system prevents him from attaching the file and displays a red message indicating the size of the file that can be attached, so he chooses a file within the allowed size.
The actor writes a text message larger than 1024 words, the system displays a red error message indicating that the allowed text size is 1024 words, so he modifies the text.
The message is sent without specifying the recipient to which the message is sent, so an error message is displayed in red, with the commitment to specifying the recipient to which it is sent.
A blank message is sent or contains a file without text, so an error message is displayed in red asking you to write text.

Other Activities:

Specify user permissions to send messages where The system allows the teacher to send messages to his students or their parents, allows students to send messages to their teachers, allows their parents to send messages to the manager or their children's teachers, and allows the manager to send messages to all students. It also allows staff, registration staff and finance staff to send messages to the school manager.

Messages are stored as drafts if the system crashes or if the user exits without sending the message.

Replies are linked to the original message so that they appear below the message as replies when you search for the original message.

Send a notification to the user when the message arrives.

The message is automatically archived by the system to allow for inbox and outbox deletion at the start of a new semester.

System State on Completion:

The message was successfully sent by the sender and appeared in the recipient's inbox and appeared as a notification on the phone. The recipient can respond to it through the replies box within the specified time period for responses.

Ahmad Ewidat,1212596

Use-Case: **Sending messages**

1.Description

1.1 Brief Description

After logging in and selecting the Send Message box, the address is written, then the recipient is selected from among the permitted recipients. The message must contain an address and text that does not exceed 1024 words in length, and a file up to 20 MB in size can be attached. There is also a place to reply to messages, and the message will appear in the sender's outbox and the recipient's inbox, and a notification will be sent to the phone that the message has arrived.

1.2 Actors

1.2.1 student

1.2.2 teacher

1.2.3 manager

1.2.4 employee

1.2.5 parent

1.2.6 manager

1.2.7 Registration officer

1.2.8 Finance Officer

2.Pre-Conditions.

2.1.The Actor must have successfully logged into his account in the system

2.2 The system should be able to check the text length and attachment size if there is

any file attached to the message

2.3 Each user determines who can send messages

2.4 Specify user permissions to send messages where The system allows the teacher to send messages to his students or their parents, allows students to send messages to their teachers, allows their parents to send messages to the manager or their children's teachers, and allows the manager to send messages to all students. It also allows staff, registration staff and finance staff to send messages to the school manager.

3.**Flow of Events**

3.**1Basic Flow**

3.1.1 The actor logs into their account in the system.

3.1.2 The actor clicks on the "Send Message" box.

3.1.3 The actor writes the message title.

3.1.4 The actor chooses the recipient from among those allowed

3.1.5 The actor writes the message text.

3.1.6 The system checks the length of the text (up to 1024 words).

3.1.7 The actor clicks on the "Send" button.

3.1.8 The message appears in the sender's "Outbox."

3.1.9 The message appears in the recipient's "Inbox."

3.1.10 The recipient can reply to the message in the reply box.

4. **Alternative Flows**

4.1 File Attachment:

4.1.1 The actor attaches a file along with the text.

4.1.2 The system verifies the file size (up to 20 MB).

4.1.3 The message is sent normally, and the recipient can download the attached file.

4.2 Response Time Period:

4.2.1 The actor specifies a time period for responding to the message or completely prevents responses.

4.2.2 The recipient can respond within the specified time period.

4.2.3 After the period ends, the recipient is not allowed to respond to the message.

4.3 System Maintenance:

4.3.1 The actor attempts to send a message during a scheduled system maintenance period.

4.3.2 The system displays a notification: "The messaging service is currently undergoing maintenance. You can save your message as a draft and send it once the system is operational."

4.3.3 The actor saves the message as a draft and receives confirmation.

4.3.4 Once the system maintenance is completed, the teacher logs back in, retrieves the saved draft, and sends the message.

4.4 Recipient Selection:

4.4.1 if the actor is teacher, After selecting the class the teacher selects the recipients (student, group of students, their parents, or both).

4.4.2 The message is sent to the selected users.

4.5 Exiting the System or Saving as Draft:

4.5.1 The actor returns or exits the system:

4.5.2 The system prompts with a confirmation message, asking if the actor wants to save the message as a draft or ignore it.

4.5.3 The actor confirms to save it as a draft or ignores the message.

5**. Post-conditions (Exit condition)**

5.1 The message is successfully delivered to the selected recipients, and it appears in their inbox.

5.2 If a file was attached, the recipient can download the file.

5.3 If the actor decided to save the message as a draft, the message is saved in the system, and

The teacher can retrieve it later.

5.4 If the actor exited without saving the message as a draft, the system discards the message.

6. Special Requirements

6.1 Ensuring that the message reaches the specified recipients and that the data is intact

6.2 Ensuring that the file size is up to 20 MB and the text length is up to 1024 words until the sending process is completed

6.3 The system's ability to archive messages so that message boxes are cleaned at the beginning of each semester

6.4 The recipient's inability to respond to the message outside the specified period

7.Extension Points

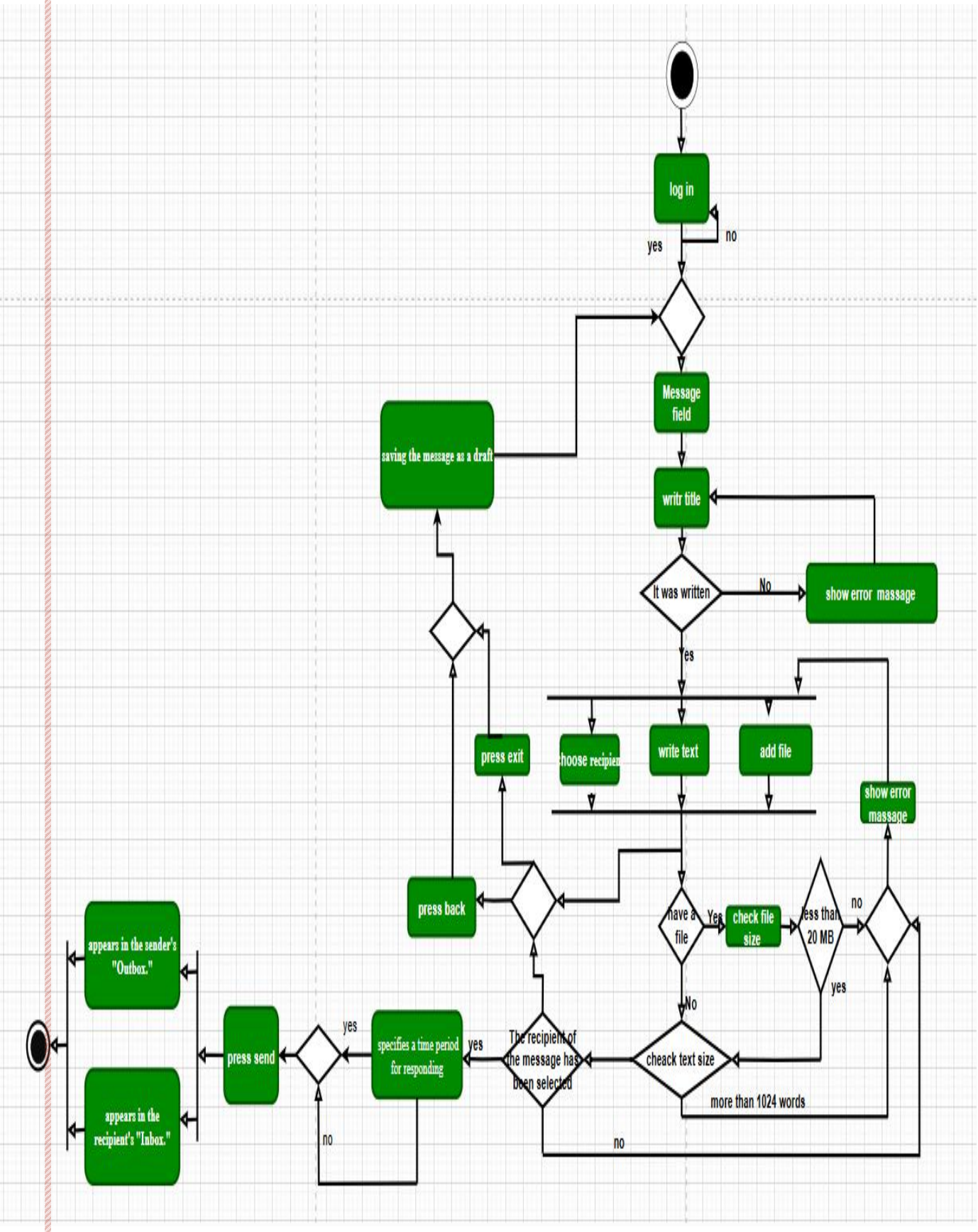
7.1 Increase the allowed file size to be larger than the current size (20 MB)

7.2 The system can allow actor to set limits on responses based on other criteria, such as setting specific hours for responses or restricting responses after a certain number of answers

7.3 The system can be improved to allow the teacher to send the same message to multiple classes at once without having to manually select each class.

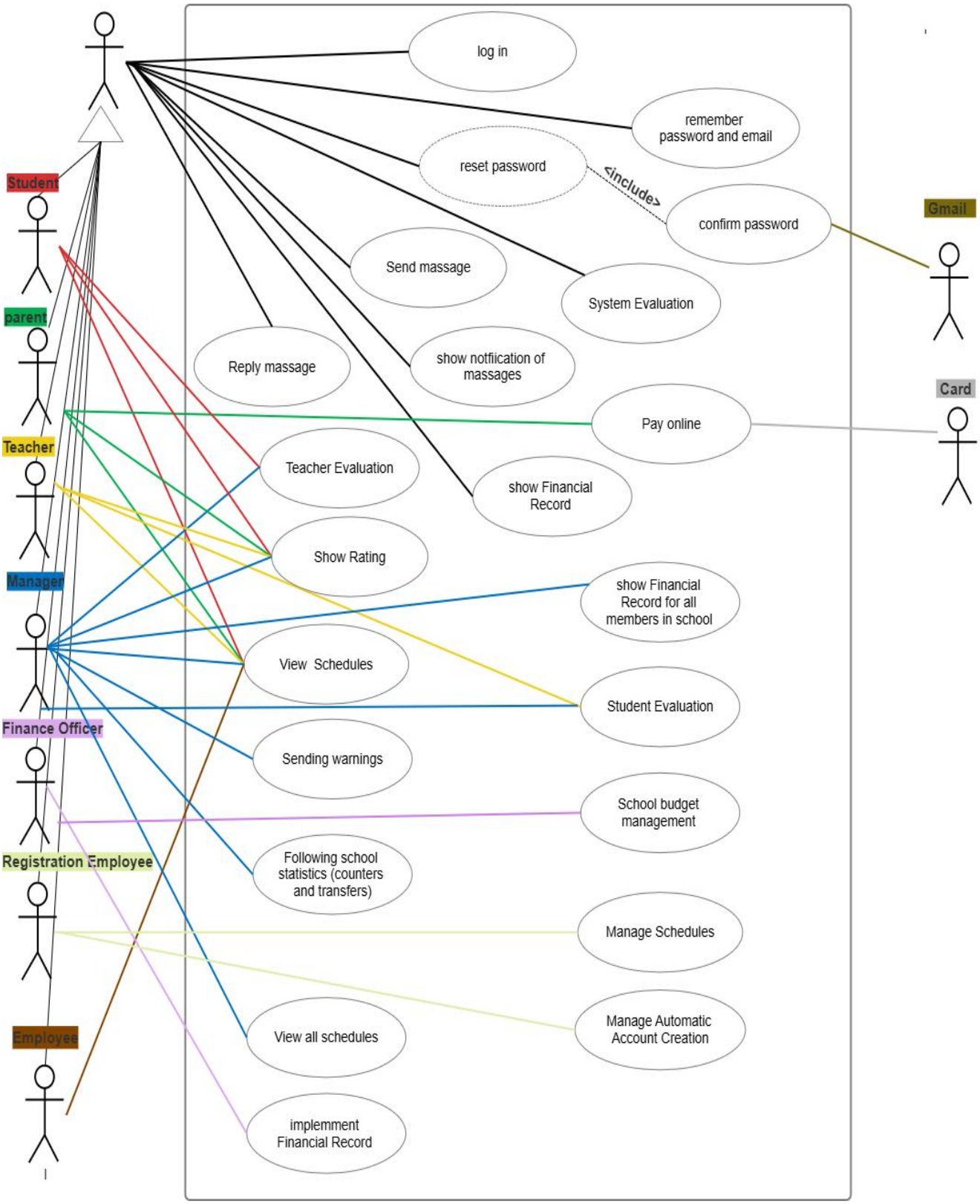
7.4 The system can be expanded to allow the actor to schedule messages to be sent at a later time or on a specific date, which is useful for advance planning.

7.5 The system can be expanded to integrate with an external email service, allowing the actor to send messages via email in addition to the internal messaging system



Activity diagram: Sending messages

Group Task



Actors and Use-Case Diagram Analysis & Modeling

Actors' roles and semantic descriptions.

Student
Role: Takes such steps as accessing schedules, interacting with the teachers, handing in assignments and rating the teachers.
Semantic Description: Depicts the core users who are in the process of receiving education as well as other related services.

Parent
Role: Changes schedules, provides the finance needed for children, communicates with teachers and with the manager.
Semantic Description: Safeguards the interests of children both in terms of academics and finances with respect to the school

Teacher
Role: Modifies the data for the course, makes messages, makes assignments, processes students and schedules.
Semantic Description: A person who provides the content of the education and supervises how the students are performing.

Manager
Role: Manages all activities, executes side of finance, caters for scheduling and communication.
Semantic Description: A person who carries out the routine duties of an administrator and communicates with the users.

Finance Officer
Role: Keeps the books of the accounts, applies milestones like discounts or bonuses, administer salaries and budgets.
Semantic Description: Looks after and modifies the accounts of the school.

Registration Employee
Role: Revisions the particulars of the users, distributes the user's codes, and handles user account removals.
Semantic Description: Performs administrative tasks such as managing information of users.

Employee
Role: Executes typical jobs given by the manager which may include cleaning, escorting guests, or offering assistance in the school.
Semantic Description: Aids the normal functions in the school and keeps the work progress uninterrupted.

COMP433 – Group Assignment phase 2

User Requirements

1. The system shall implement a login mechanism that requires users to authenticate themselves using a username and password combination. This method will ensure secure access to the system and protect user data.
2. The system shall automatically create accounts for all users registered in the school, including accounts for students, parents, and teachers.
3. The system shall provide a weekly timetable for students, teachers, staff, and the principal.
4. The system shall implement a financial record module to manage and display financial information for each user.
5. The system must provide the finance officer with full access to all financial records of all users, ensuring the ability to view, manage financial data.
6. The system shall provide a built-in messaging platform to facilitate communication among teachers, students, parents, and administrators.
7. The system should give the manager special powers to manage operations, access comprehensive financial records, assign tasks, and communicate with all users.
8. The system should be able to provide ratings for teachers, students and staff and show them to other users.

System Requirements

1.1 The system shall provide a login mechanism that supports multiple types of users, including students, parents,employees, and teachers.
1.2 The password shall exclude numbers, special characters, and symbols.

1.3 The username shall support multiple types of identification: parents shall use their ID, students shall use their student ID, and teachers shall use their teacher ID,employee shall use their employee ID.

1.4 The system shall provide a checkbox to allow users to opt to remember their username and password or not.

1.5 The system shall prevent users from being logged in on multiple devices or web pages simultaneously.

1.6 The system shall provide an option for users to reset their password. The password reset process shall be confirmed via email.

2.1 The ID for each user shall be unique and shall be assigned automatically by the system.

2.2 The accounts automatically created by the system should include relevant information for each user, such as name and region. The student account shall also include the parent's information.

2.3 The system shall automatically delete the account of any user who leaves the school, but the deletion shall be confirmed by the registration employee.

2.4 The user shall be able to update their information, but cannot change their ID. The registration employee shall have the ability to change the user’s ID.

2. 5The system shall assign roles to users during account creation, including :general employees (janitors, guides, assistants), financial staff, registration staff, students, parents, teachers, and the principal.

3.1 The system should display to the student the course name, its schedule, the method of teaching (whether in-person, online, or canceled), the teacher's name, room number, and the name and contact number of the social guide.

3.2 The system should display to the teacher the course name, its schedule, the classroom number, the number of students, and provide a field to update the lecture status.

3.3 The system should display to the principal their weekly schedule, including tasks they can add themselves (such as meetings) and pre-programmed annual tasks. If the principal has teaching duties, these shall also be included.

3.4 The system should display the employee's schedule, which includes their assigned tasks and work locations as designated by the manager.

3.5 The system should display to the parent their child's schedule, along with the teacher's information, office hours for visitors, scheduled meetings, and the telephone numbers for the social guide, manager, and school.

4.1 The system shall provide the manager with access to a comprehensive financial record dashboard. This dashboard shall display the financial records of all users in the system, including students, employees, teachers, and the manager's own financial information. Additionally, the system shall generate and display financial reports, summarizing key financial data and transactions

4.2 The system shall display the salary details of all employees, including teachers and managers. This feature shall include a history of salaries, bonuses, and discounts applied.

4.3 The system shall display detailed financial information for students , including installments, outstanding debts, grants, and payment history related to installments. This information must be organized clearly and made accessible to the respective parent and student through their dedicated interfaces.

4.4 The system shall display the salary history for all employees, including details of salaries from previous years, the current balance, financial transactions, bonuses, and deductions.

4.5 The system shall require email confirmation to access to financial records. Users must verify their identity through a confirmation email sent to their registered email address before proceeding.

4.6 The system shall provide an option for users to make payments using a credit card. This feature must support secure authentication using a user ID and password to ensure safe and verified transactions.

4.7 The system shall automatically create bank accounts for all employees and students. The student or parents should be able to use their bank account linked with their ID for making payments. Additionally, employees should be able to withdraw their salaries directly through their respective bank accounts .

5.1 The system shall allow the finance officer to add discounts or bonuses to employees' records, with the ability to request approval from the manager before applying the changes.

5.2 The system shall allow the financial officer to monitor and manage the salaries of all employees, including the ability to deposit and withdraw salaries within the available budget.

5.3 The financial manager shall be responsible for managing the school's budget, overseeing expenditures on activities and costs such as supplies, equipment, and other operational expenses.
5.4 The financial manager shall be responsible for managing the school's grants, which are distributed based on the financial and academic status of the students

5.5 The finance officer shall have the ability to send emails or announcements to the school community regarding grants and their details, as well as donations to the school, whether for projects, students, or school clubs such as scouts. Additionally, the finance officer shall be able to send messages to individuals or institutions regarding donations.

6.1 The system should provide the ability to send text messages, as well as attachments such as Word files, images, etc. With the possibility of not replying to messages or replying only for a certain period of time. The maximum limit for text messages is 1024 words and the maximum file size is 20 MB.

6.2 The teacher can send a message or upload assignments to students or to a specific student, as well as to parents.

6.3 Students can send a message to their teacher privately, turn in assignments, or send a message to the teacher and class.

6.4 The manager can send a message to any user in the system.

6.5 The system should provide message archiving at the beginning of each new semester.

6.6 The system should send notifications to the mobile phone, provide the ability to access the message through the phone or through the system, and respond to the message only through the system.

6.7 The system must give the ability to search for messages by name or filter messages by sections (teacher's name and subject name,).

7.1The system should provide the manager with access to the schedules of all employees, including their free time, quota times, public holidays, quota information, and status.

7.2 The system should provide the manager with reports on students' financial records.

7.3The system must provide the manager with reports on the number of current students, new entrants, and transfers.

7.4 The system should provide the manager with reports on the financial records of employees as well as the number of current employees in the system.

7.5The system should provide the manager with expense reports in terms of curricular and extracurricular activities or the purchase of new materials for the school.

7.6The system should provide the manager with the authority to permanently or temporarily expel a student, or give the student a warning.

8.1The system should provide a feature for students to rate teachers in terms of their teaching performance and communication methods.

8.2 The system should provide a feature for teachers to assess students in terms of skills and ethics.

8.3 The system should provide a feature for teachers where students are evaluated in terms of academic performance, their grades are entered, and the results are issued in the form of certificates.

8.4 The system must provide access for the manager to evaluate teachers and assess their achievements.

8.5 The system should allow users to evaluate the system in terms of performance, efficiency, and speed.

Feature/User Req.	Function Point	Total Function Points (* 12)	# of Days
UR1	3	36	5
UR2	4	48	6
UR3	3	36	5
UR4	5	60	8
UR5	3	36	5
UR6	3	36	5
UR7	2	24	3
UR8	2	24	3
Total	25	300	40

The development team of 5 developers will work full-time **(8 hours/day)** for **2 weeks (10 working days)**. Developers 1, 2, and 3 earn **$1300/month**, while Developers 4 and 5 earn **$1500/month**. Since 10 days equal **0.5 months**, the salaries for Developers 1, 2, and 3 total **$1950**, and for Developers 4 and 5 total **$1500**, resulting in a total cost of **$3450** for the 2 weeks.

Adding a profit margin of **30% to 50%**, the total project cost is estimated to range from **$4485 (30% profit)** to **$5175 (50% profit)**.

Min offer: 4500$

Max offer: 5200$

COMP433 – Group Assignment phase 1

Introduction

After a thorough review of the current system at Modern Schools, we have developed a clear vision for creating a comprehensive school management platform designed to meet the needs of all stakeholders, including students, teachers, parents, and administrators. This platform will provide efficient and user-friendly tools to address the challenges faced in daily operations, offering an integrated solution to modernize and streamline the school's processes.

The existing system heavily relies on manual operations and face-to-face interactions, creating several critical challenges. Unpredictable geopolitical and weather conditions often disrupt regular attendance, making full in-person education unreliable. This issue became even more evident during the COVID-19 pandemic, where the abrupt transition to e-learning without prior preparation caused significant setbacks in education.

Administrative tasks like student registration and record management are also inefficient due to manual processes, leading to delays and errors. Similarly, managing financial operations, including tuition payments and employee payroll, is cumbersome and error-prone. Teachers face difficulties in tracking and updating grades and assignments, while students and parents lack real-time access to progress updates, affecting overall academic performance.

Teachers struggle to coordinate with departments like registration and finance, while parents have limited channels to stay updated on their children’s progress. This lack of efficient communication reduces parental engagement, negatively impacting student success. Additionally, the administration lacks a structured and fair system for evaluating employee performance, which limits its ability to assess productivity and foster motivation.

To address these issues, we propose developing a {hybrid school management platform} consisting of a {web-based application} and a {mobile application}. The web platform will cater to administrators, teachers, and parents for detailed management and tracking, while the mobile app will provide on-the-go access for students and parents.

Key features of the platform include

Our proposed system introduces a range of innovative features designed to address the challenges faced by Modern Schools. {Automated registration and student management}will simplify the enrollment process and enable efficient attendance tracking through digital records. {Integrated financial management} tools will streamline payroll, tuition payments, and other financial operations, reducing errors and delays. Teachers will benefit from {centralized grade and assignment tracking}, allowing them to update and share information in real time, enhancing academic transparency. {Enhanced communication channels}, such as built-in messaging systems, will facilitate seamless interactions among teachers, parents, and administrators, improving collaboration. Additionally, the platform will offer {e-learning support} through integration with tools like Zoom, enabling smooth transitions between in-person and remote learning. Parents will have access to {dedicated portals} to monitor their children's academic and behavioral progress, fostering greater engagement. Finally, the system will include {employee evaluation tools}, empowering administrators to assess and track staff performance using data-driven insights, ensuring fairness and boosting productivity.

This proposed solution will not only address the limitations of the current system but also provide Modern Schools with a scalable and future-ready platform. By combining innovation with usability, this system ensures that the school operates efficiently under all circumstances, empowering its stakeholders and enhancing overall educational quality.

Registration Department

Automated Registration Process

Our system will provide an intuitive, user-friendly interface for online student and parent registration, simplifying the enrollment process and reducing paperwork.

Once registered, automated email confirmations will ensure seamless communication.

Usability: The interface is designed to be easy to navigate for users with varying levels of technical skills, minimizing the need for extensive training.

Class Allocation System

An intelligent algorithm will assign students to classes based on their preferences, availability, and the teacher's workload. This ensures optimal distribution of students across classes.

Efficiency: The AI-driven system will eliminate overlaps by dynamically organizing weekly schedules according to teacher work patterns and class capacities.

Schedule Generation

The dynamic scheduling tool will generate class schedules based on room availability, teacher availability, and student preferences, ensuring no conflicts in class times.

Performance: The scheduling system will provide real-time updates, so all scheduling information remains accessible and up to date at any time.

→AI Implementation Details

AI will be used to allocate students into classes by analyzing factors such as class sizes, subject preferences, and teacher schedules. Machine learning algorithms will continuously improve the class allocation process, optimizing it based on evolving data

Student Certificates

The system will streamline the process of issuing school certificates at the end of each semester, including certificates for students excelling in various fields, making it easier for the administration to handle student records.

Registration of Students and Teachers

The system will include a comprehensive registration process, storing personal information, academic history, and evaluations for both students and teachers. This ensures complete and accurate records that can be accessed anytime.

Transportation Management

For transferring students and teachers, the system will generate comprehensive files containing personal details, academic performance, and recommendations from previous institutions.

Efficiency: AI will assist by offering feedback on the student's academic performance and improvement in different subjects.

Accounting Department

Student Fee Management

A centralized dashboard for tracking registration fees, discounts, and additional expenses such as books and supplies. Parents can view and manage payments through their portal.

→A secure online payment gateway will be integrated into the platform to facilitate smooth transactions.

Teacher Salary Management

An automated payroll system that calculates salaries based on attendance and hours worked, ensuring timely payments and accurate records.
→ Payroll data will be connected to attendance records, with automatic calculations and report generation for administrative purposes.

Event Cost Tracking

A feature to plan and budget for school events, including tracking expenses and generating reports for future reference.
→ Administrators will input estimated costs, and the system will track actual expenditures against the planned budget, providing real-time financial insights.

Financial Grants

A system for contacting third-party organizations regarding financial grants for underprivileged students, with grants distributed based on specific student assessments.
→ Students can apply for grants through a form-based submission, and the system will assess eligibility using predefined criteria.

Management

Real-Time Reporting

A comprehensive dashboard for school administrators that provides real-time insights into student registration, attendance, and financial data, enabling informed decision-making .

→The dashboard will pull data from various school systems and provide quick access to critical information, ensuring that administrators can make timely, data-driven decisions.

Communication Tools

A built-in messaging system for sending announcements and notifications to parents, teachers, and students. This feature includes bulk messaging capabilities for efficiency.

→The messaging system will allow administrators to send messages individually or in bulk, with options for text, email, and app notifications.

Compliance Tracking

Tools to monitor and report on compliance with educational regulations and standards, ensuring that the school meets all legal requirements,

→ The system will track regulatory deadlines and store compliance data securely, with access restricted to authorized personnel only. Notifications will be sent when compliance actions are due.

External Communication

A dedicated feature for the principal to communicate with external parties via the school's official email address.

**→The principal will have access to a secure email interface that connects directly with the school's official communication channels, ensuring all external correspondence is handled professionally and securely.

Teacher

Resource Repository

A digital space for teachers to store and share educational materials, lesson plans, and multimedia resources, fostering collaboration and innovation in teaching methods .

→The repository will allow teachers to upload, organize, and share resources securely, promoting collaborative teaching and enhancing the quality of education.

Performance Tracking

An interface for teachers to monitor their performance metrics, including attendance records and feedback from students and parents .

→The performance tracking system will collect data from various sources, such as attendance and feedback surveys, and display it in a simple, user-friendly interface that encourages continuous improvement.

Professional Development Tools

Access to online training resources and workshops to enhance teaching skills, along with a system for tracking completed professional development activities.

→ Teachers will have access to a library of online courses and workshops, which they can complete at their own pace. The system will automatically track their progress and issue certificates for completed activities.

Parent

Student Progress Monitoring

Parents can access a detailed view of their child's academic performance, including grades, attendance, and teacher comments. This transparency fosters engagement in their child's education.

→ Parents will have a personalized portal to monitor their child's grades and attendance in real-time. Notifications will be sent for any updates or changes, ensuring parents stay informed and engaged.

Financial Overview

A clear breakdown of all financial transactions, including fees paid, outstanding balances, and applicable discounts.

→ The financial dashboard will display an itemized list of all payments, discounts, and outstanding balances. Parents can easily track their payments and view any pending charges or discounts.

Communication

A log of all communications from teachers and administrators, ensuring parents are kept informed about important updates and their child's progress.

→A communication log will be integrated into the parent portal, allowing parents to review all messages and notifications from teachers and school staff. It will be easy to track all correspondence for better communication and transparency.

Staff Management System

Task Assignment and Tracking

A feature for assigning tasks to staff members, allowing them to track their workload and mark tasks as complete.

→ A user-friendly task management interface will be implemented, enabling staff to easily view and manage their assigned tasks. Staff members will be able to mark tasks as complete, ensuring progress is tracked in real-time.

Payroll Overview

Staff can access their salary information, deductions, and any pending payments, promoting transparency and trust.

→ The payroll system will allow staff members to view a detailed breakdown of their salaries, including deductions and pending payments, through a secure portal. Only authorized users will have access to this sensitive information to ensure privacy and security.

Leave Management System

A tool for staff to request leave, with automated approval workflows that streamline the process for management.

The leave management system will allow staff to submit leave requests through a digital form. The system will automatically route the requests for approval based on preset criteria, and staff will be notified of the approval status.

Student

Class Schedule

The system allows students to view their weekly class schedule, including the names of teachers, subject titles, and class timings. Additionally, students can see when their teachers are available for consultations .

→A simple, child-friendly interface will display all relevant information in an easy-to-read format. The system will allow students to quickly access their schedule and teacher availability with minimal effort.

Delivery of Assignments

Students can view their assigned tasks and track their progress. Once assignments are submitted, students will be able to see feedback and grades after teachers have completed the grading process.

→Students will submit their assignments through the platform, and feedback will be automatically provided when grading is complete. All tasks will be visible in a user-friendly layout for easy access.

Communication

Students can communicate directly with their teachers via an internal messaging system for queries or feedback.

→A built-in messaging system will allow students to send and receive messages from teachers and administrators. The communication interface will be simple to navigate, ensuring smooth and efficient interactions.

Non-Functional Requirements

Usability

The system should provide an intuitive and easy-to-use interface across all categories (students, teachers, administrators). Tasks such as delivering assignments, tracking progress, and communicating with teachers should be straightforward and accessible .

Measurable Criteria

-The user interface should reduce the average task completion time by at least 20% compared to current systems.

%95 -of users should report satisfaction with the ease of navigation based on post-usage surveys.

-The system will feature customizable views, allowing users to personalize their experience.

Security

Given the sensitive nature of the data (financial, academic records, and personal information), security measures must protect all stakeholders. Only authorized users (teachers, administrators, students, and parents) will have access to relevant data .

Measurable Criteria

-The system will employ end-to-end encryption for all sensitive data transmission.

-Multi-factor authentication will be required for access to administrative features.

-Access control lists will be implemented, ensuring that users can only access data pertinent to their role.

Regular security audits will be conducted, with the system achieving compliance with the highest security standards.

Maintainability

The system should be built in a modular fashion, allowing easy updates and integration of new features as educational needs evolve. The software architecture should be flexible to accommodate future growth, including new types of users, data, and functionality .

Measurable Criteria

-The system’s modularity should allow for 80% of feature updates to be completed without downtime.

%90 -of reported system bugs should be resolved within 24 hours

-The system should be able to scale efficiently to handle 1,000 concurrent users with minimal performance degradation.

Software development process

The Prototyping + Incremental Development (Agile) model involves developing the software in smaller, manageable chunks called sprints typically lasting two weeks. In this model, the process begins with creating prototypes—early versions of the system designed to demonstrate its basic features. These prototypes are not fully functional systems but serve as a tool to help the client better understand the idea, design, and potential user experience of the system.

Each sprint focuses on developing a specific set of features or functionalities. After completing a sprint, feedback from the client is collected to evaluate whether the delivered features align with their expectations and needs. Based on this feedback, adjustments and improvements are made for the next sprint.

This process is repeated in incremental phases, ensuring that the system gradually evolves, and the client remains involved throughout the development process. It allows for flexibility and continuous refinement, helping to address any issues early on and making sure the final product meets the client's requirements and expectations.