

Student Management System

By Ihor Melashchenko, Marvin Adorian Zanchi Dos Santos, and Emmanuel Abolade

Summary

Summary

Introduction

The Student Management System is designed to facilitate various functions for students at the university, including registration, amending personal details, requesting student ID card issuance, withdrawal, and topping up campus credit balances.

User Interface

The user interface of the application serves as the gateway for both students and university staff members. Upon launching the system, users are presented with an intuitive and user-friendly menu that guides them to their respective sections.

User Authentication

To access the system's functionalities, users must complete a secure login process using their unique credentials. For demonstration purposes, the system has been preloaded with 5 test student accounts and 5 test staff member accounts, ensuring a smooth user experience.

Student Functionalities

If the user is a student, they will have access to the following options:

1. **Register at the University:** Students can initiate the registration process, by providing the necessary information.
2. **Amend Details:** Students can update their personal information as needed.
3. **Request ID Card Issuance:** Students can request the issuance of a student ID card.
4. **Withdrawal:** Students can initiate the withdrawal process.
5. **Top Up Campus Credit:** Students can add funds to their campus credit balance.

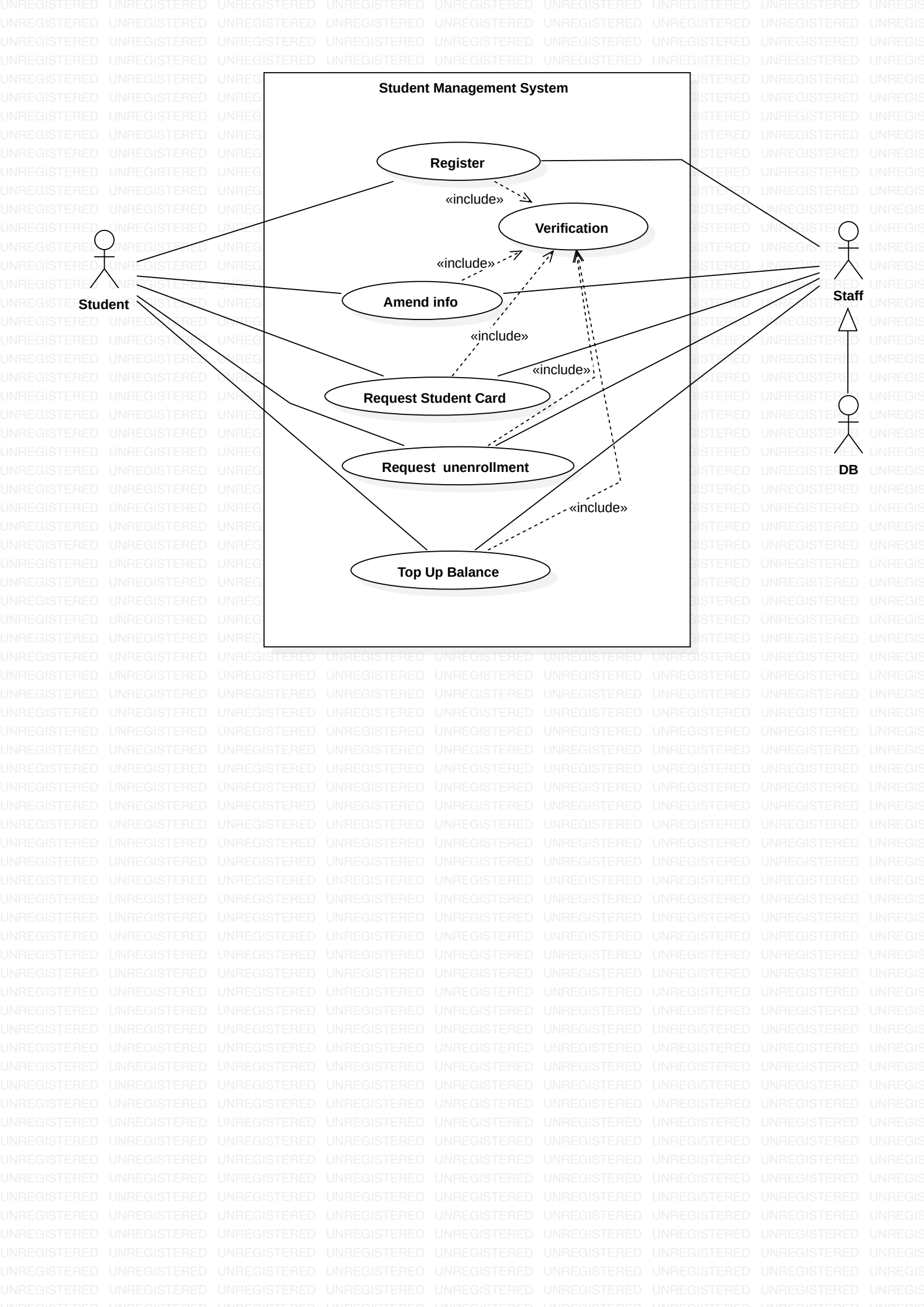
Staff Member Verification

For each option selected by a student, staff member verification is required. This verification process entails sending a request to a staff member, who can either accept or decline the student's request.

Database Updates

After a staff member accepts or declines a student's request, the system will update the database accordingly to reflect the decision.

Use Case Diagram



Scenarios

Scenarios

Name: Course Registration	UniquelD:001
Area: School	School
Actor(s): Student, Teacher, Admin DB	Student, System, Teacher, Admin DB.
Description:	Student wants to register a course so as to get enrolled.
Triggering Event:	Course Enrolment.
Trigger type: External Temporal	None.
Steps Performed (Main Path):	Information for Steps 1. Student logs into the system with their credentials. 2. Student selects "Course Registration" option. 3. Student selects a specific course. 4. The system checks if there are any prerequisites for the chosen course. 5. If prerequisites are met, the student confirms their course selection. 6. Student exits the system.
Preconditions:	Student has a valid credentials to access the system.
Postconditions:	The system updates the student's course list.
Assumptions:	Student may like to print out completed registration.
Requirements met:	Student registration was successful.
Outstanding issues:	Teacher gets notified and may disapprove of student.
Priority:	1
Risk:	Student will need to restart registration process if power goes off while using the system.

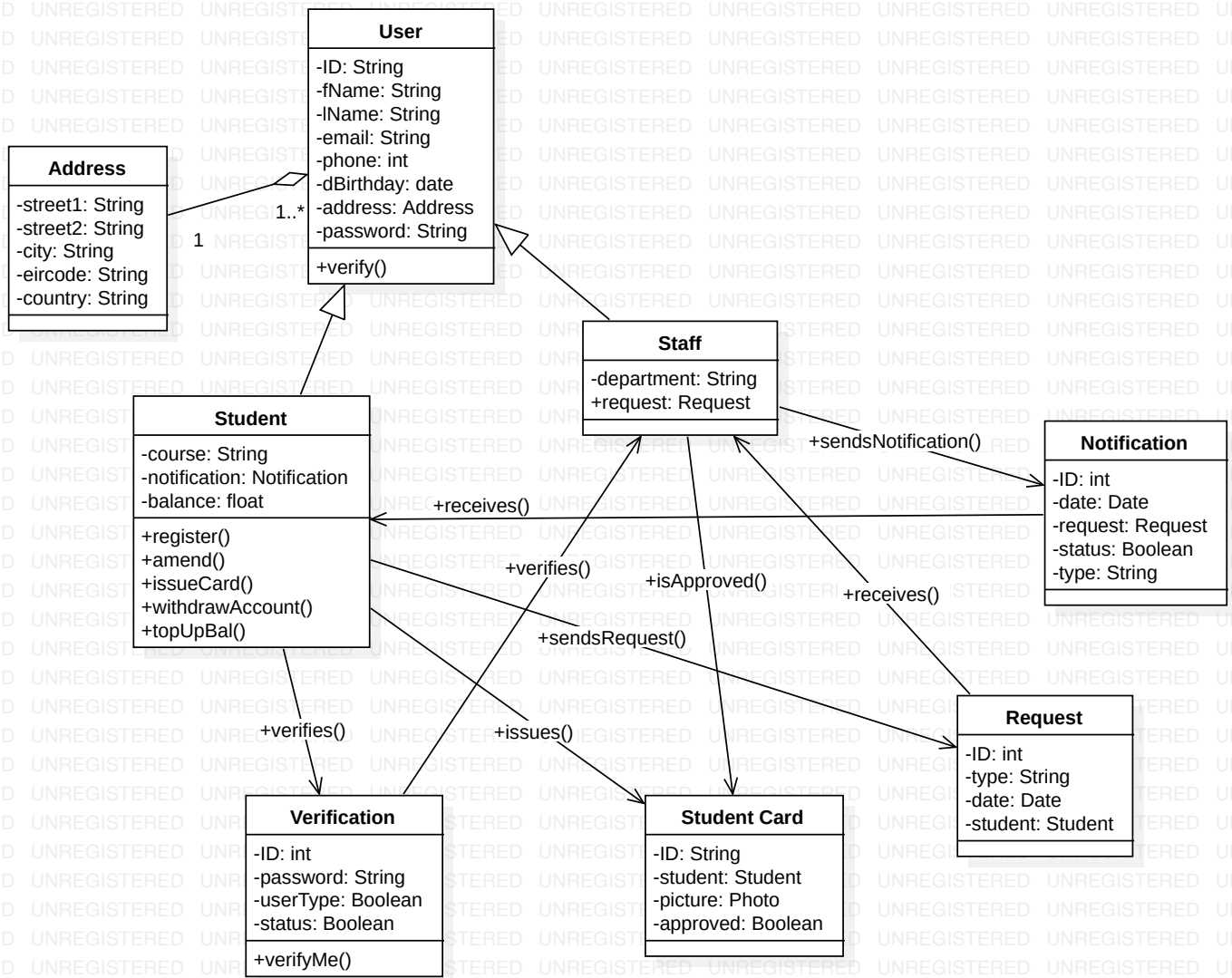
Name: Amend Information	UniqueID:002
Area: School	School
Actor(s): Student, Teacher, Admin DB	Student, System, Admin DB.
Description:	A student needs to update their personal information (e.g., address, contact information).
Triggering Event:	Personal information requires to an update.
Trigger type: External Temporal	None.
Steps Performed (Main Path) 1. 2. 3. 4. 5. 6.	Information for Steps Student logs into the system with their credentials. Student selects "Personal Information" option. Student selects the information they want to amend (e.g., address, phone number). Student makes the necessary changes and saves the updated information. System displays current student information. Student exits the system.
Preconditions:	Student has a valid credentials to access the system.
Postconditions:	The system verifies and updates the student's information. Teacher is notified.
Assumptions:	Student may like to print out completed updated record.
Requirements met:	Student Details Amendment was successful.
Outstanding issues:	Students must remember their new password if that was what was changed.
Priority:	2
Risk:	If verification fails, student may not be able to update their information.

Name: Request Student ID Card	UniqueID:003
Area: School	School
Actor(s): Student, Teacher, Admin DB	Student, System, Admin DB.
Description:	A student needs a replacement/New student ID card.
Triggering Event:	ID card is needed for identification and Account Top up.
Trigger type: External Temporal	None.
Steps Performed (Main Path)	Information for Steps 1. Student logs into the system with their credentials. 2. Student selects "request student card" option. 3. Student provides a reason for the replacement request. 4. System retrieves the student information. 5. Student confirms the card details (name, photo, course etc.) 6. System confirms success for student ID card request. 7. The system generates a request for a new ID card and informs the administrative staff. 8. Student exits the system.
Preconditions:	Student has a valid credentials to access the system.
Postconditions:	Administrative staff verifies the request and processes the new ID card for collection.
Assumptions:	Student may need to upload a passport size photo to complete the process if not available in the student data.
Requirements met:	Student card request was successful.
Outstanding issues:	Student Id card requires collection at the school admin office.
Priority:	3
Risk:	Request may fail if criteria for upload are not met.

Name: Top Up Balance	UniquelD:004
Area: School	School
Actor(s): Student, Teacher, Admin DB	Student, System, Admin DB.
Description:	A student wants to add funds to their student account balance.
Triggering Event:	Fund availability for use in the school.
Trigger type: External Temporal	None.
Steps Performed (Main Path) 1. 2. 3. 4. 5. 6. 7. 8.	Information for Steps Student logs into the system with their credentials. Student selects "Top up" option. System displays current balance. Student adds an amount to top up. Student provides payment information (e.g., credit card details). System validates and update the top up balance. Student confirms the new balance. Student exits the system.
Preconditions:	Student has a valid credentials to access the system.
Postconditions:	Student receives transaction details notification.
Assumptions:	Card may be declined if credit card credentials are incorrect.
Requirements met:	Student tops up was successful.
Outstanding issues:	There should be money in student credit card account to complete the top up balance.
Priority:	4
Risk:	Request may fail if criteria for topping up are not met.

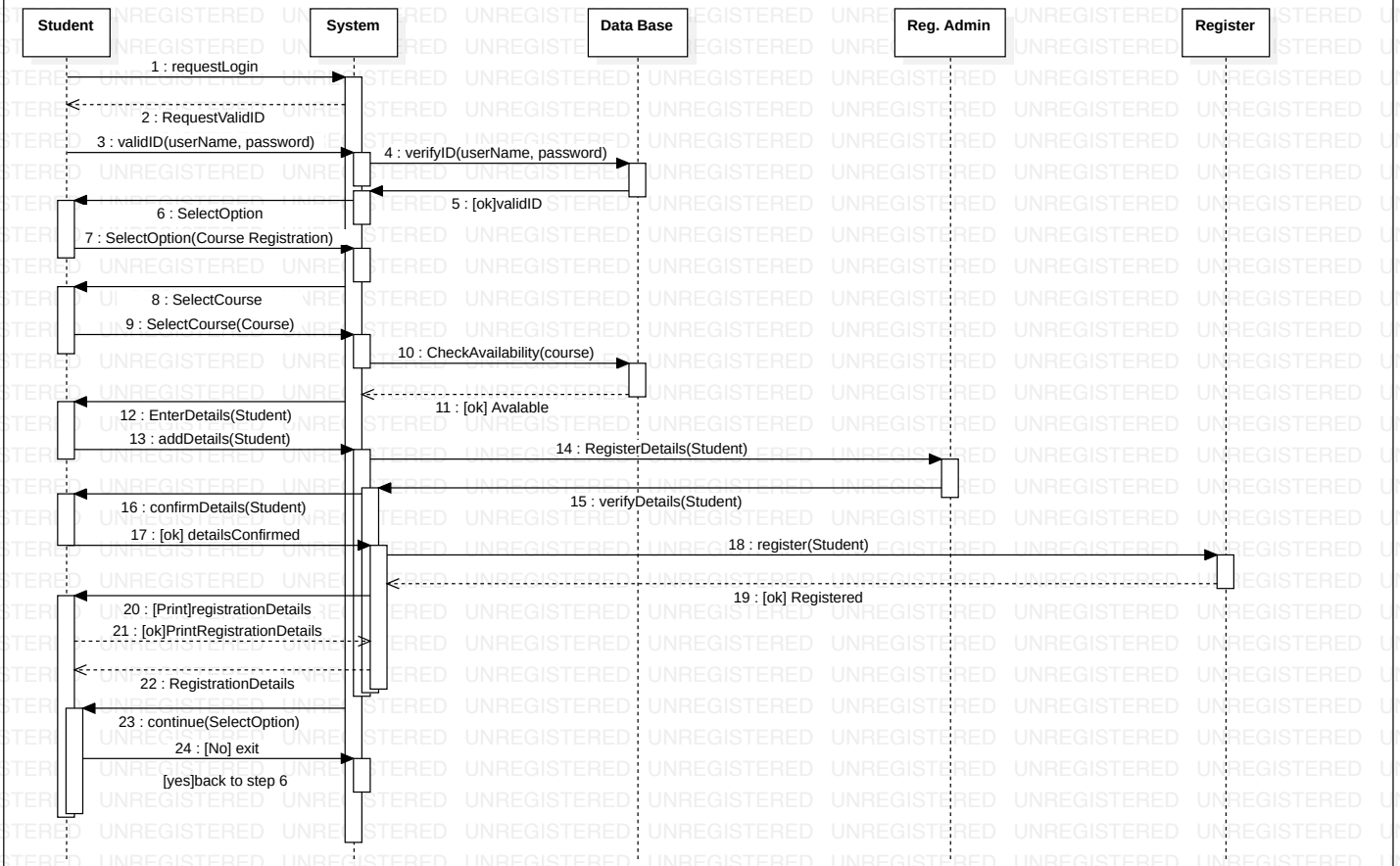
Name: Request withdraw	UniqueID:005
Area: School	School
Actor(s): Student, Teacher, Admin DB	Student, System, Admin DB.
Description:	A student wants to withdraw from a course or school.
Triggering Event:	Student is unable to cope or has specific reasons.
Trigger type: External Temporal	None.
Steps Performed (Main Path)	Information for Steps 1. Student logs into the system with their credentials. 2. Student selects "Withdrawal" or "Drop Course" option. 3. System displays courses. 4. Student selects the course they wish to withdraw from. 5. System checks if the withdrawal is within the allowed timeframe. 6. If allowed, the system processes the withdrawal and updates the student's course list. 7. Student confirms the withdrawal request. 8. Student exits the system.
Preconditions:	Student has a valid credentials to access the system.
Postconditions:	Student receives notification of the withdrawal.
Assumptions:	Selecting and confirming the wrong course might create a problem.
Requirements met:	Student successfully withdrew from course.
Outstanding issues:	Teacher needs to be notified.
Priority:	5
Risk:	Request may fail if criteria if withdrew course are not met.

Class Diagram



Sequence Diagrams

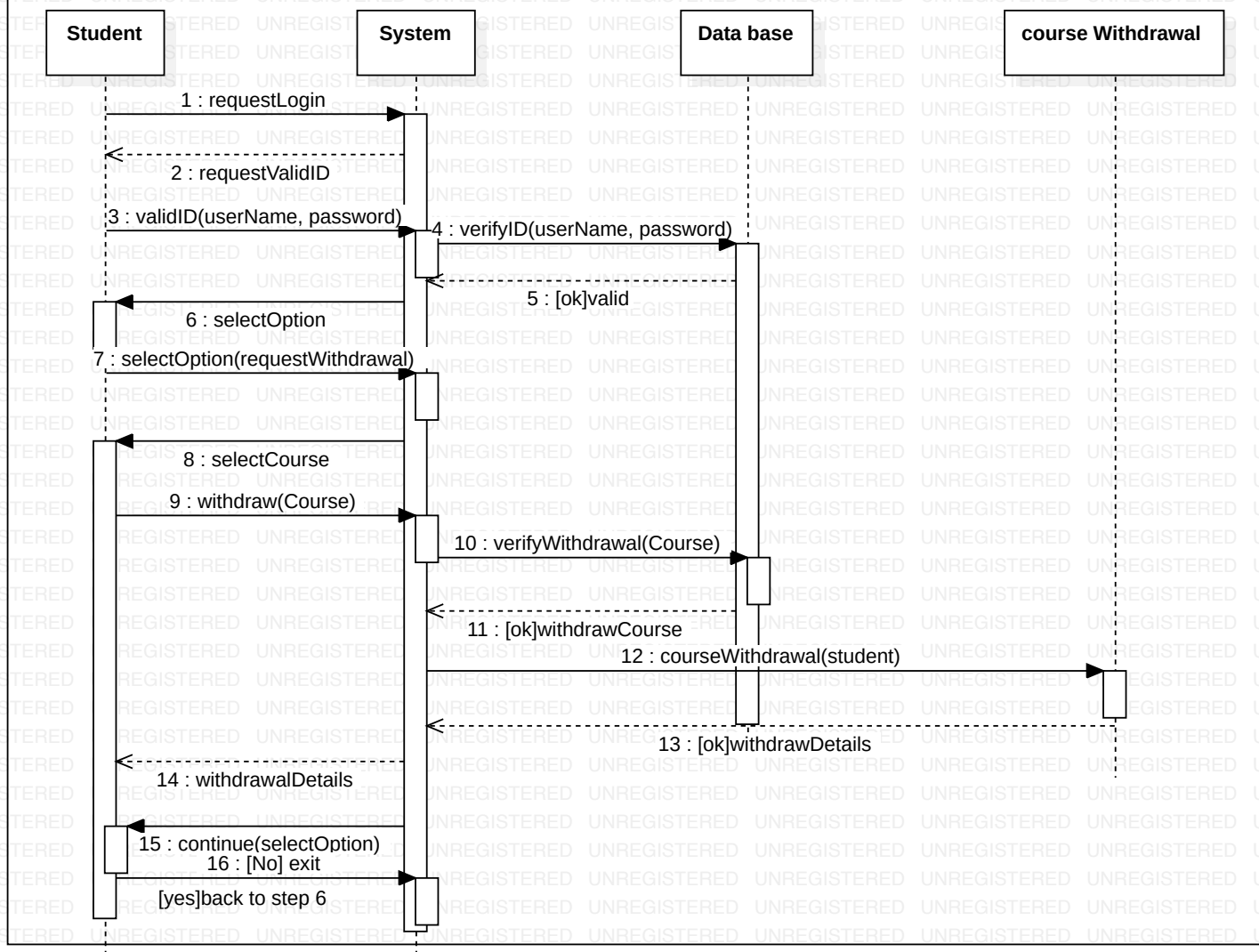
sd Register Course



Message Analysis Table – Register Course

Sequence Num	Caller	Callee	Msg Name	Msg Type	Msg Param	Msg Constrain
1	Student	System	Request Login	Synchronous	Valid ID	-
2	System	Student	Request Valid ID	Synchronous	-	-
3	Student	System	Enter ID	Synchronous	Username, Password	Unique Student number
4	System	Database	Verify ID	Synchronous	Credentials	-
5	Data Base	System	Valid	Synchronous	-	Valid Credentials
6	System	Student	Select Option	Synchronous	-	-
7	Student	System	Select Course Registration	Synchronous	Course Registration	-
8	System	Student	Choose Course	Synchronous	Course	-
9	Student	System	Select Course	Synchronous	Course	-
10	System	Data Base	Check Availability	Synchronous	Course	=Available
11	Data Base	System	[Ok]Availability	Synchronous	Yes	-
12	System	Student	Enter Student Details	Synchronous	Student Details	-
13	Student	System	Add Student Details	Synchronous	Student Details	-
14	System	Reg. Admin	Register Student	Synchronous	Register	-
15	Reg. Admin	System	Verify student Details	Synchronous	Verify Details	-
16	System	Student	Confirm student Details	Synchronous	Student Details	-
17	Student	System	Yes/No	Synchronous	-	-
18	System	Register	Add Student Registration	Synchronous	-	-
19	Register	System	[ok]Registered	Synchronous	-	-
20	System	Student	Print Reg. Details	Synchronous	Reg. details	-
21	Student	System	[ok]Print Registration	Synchronous	-	-
22	System	Student	Registration Details	Synchronous	-	-
23	System	Student	Select Option	Synchronous	-	-
24	Student	System	[No]Exit	-	-	-

sd Request Course-withdrawal



Message Analysis Table –Request Withdrawal

Sequence Num	Caller	Callee	Msg Name	Msg Type	Msg Param	Msg Constrain
1	Student	System	Request Login	Synchronous	Valid ID	-
2	System	Student	Request Valid ID	Synchronous	-	-
3	Student	System	Enter ID	Synchronous	Username, Password	Unique Student number
4	System	Data base	Verify ID	Synchronous	Credentials	-
5	Data Base	System	Valid	Synchronous	-	Valid Credentials
6	System	Student	Select Option	Synchronous	-	-
7	Student	System	Select Request Withdrawal	Synchronous	-	-
8	System	Student	Select Course	Synchronous	-	-
9	Student	System	Withdraw Course	Synchronous	Course	-
10	System	Data Base	Verify Course Withdrawal	Synchronous	-	-
11	Data Base	System	[ok]Withdraw completed	Synchronous	-	-
12	System	Course Withdrawal	Course Withdrawal Details	Synchronous	-	-
13	Course Withdrawal	System	[ok] withdrawal Details	Synchronous	Course withdrawal Details	-
14	System	Student	Withdrawal Details	Synchronous	-	
16	System	Student	Select Option	Synchronous	-	-
17	Student	System	[No]Exit	-	-	-

sd Top-up Balance

Student

System

Data Base

TopUpBalance

1 : requestLogin

2 : requestValidID

3 : validID(userName, password)

4 : verifyID(userName, password)

5 : [ok]valid

6 : selectOption

7 : selectOption(topUpBalance)

8 : selectAmount(amount)

9 : addedAmount(amount)

10 : enterCardDetails

11 : cardDetail(num, expdate, cvc)

12 : verifyCardDetails

13 : [ok]cardVerified

14 : confirmAmount(amount)

15 : [ok]amountconfirmed

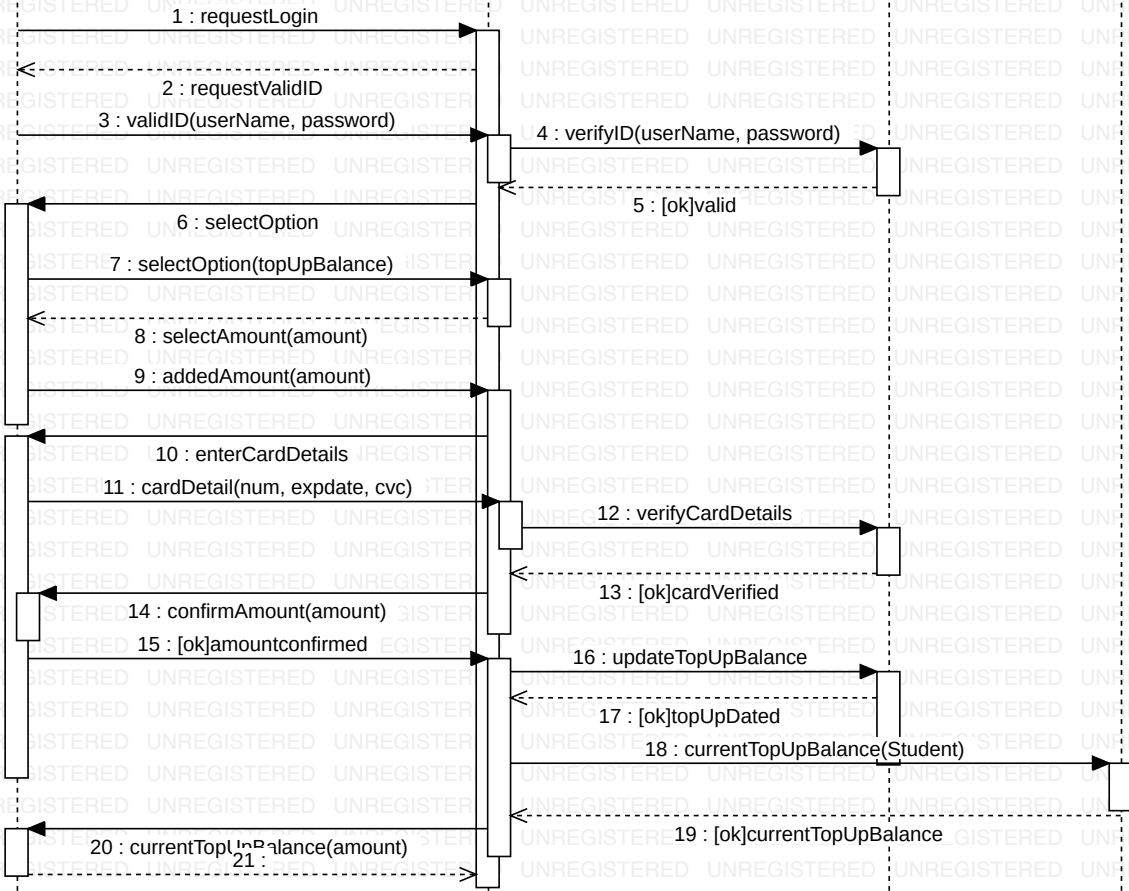
16 : updateTopUpBalance

17 : [ok]topUpDated

18 : currentTopUpBalance(Student)

20 : currentTopUpBalance(amount)

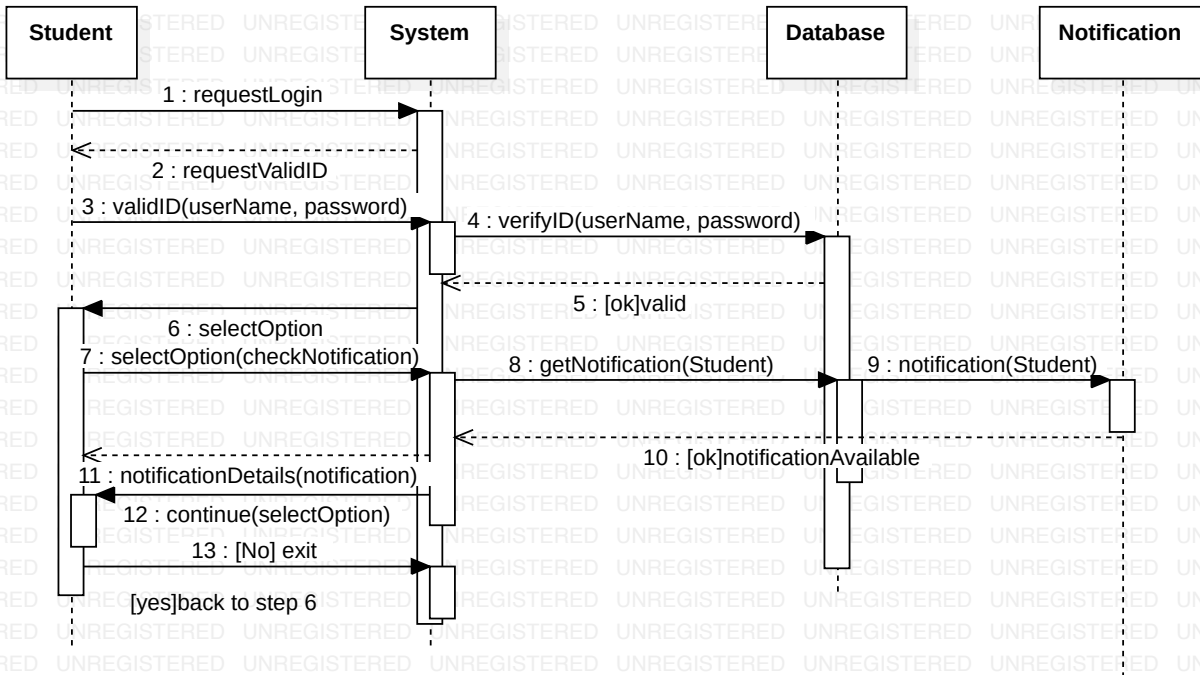
21 : [ok]currentTopUpBalance



Message Analysis Table –Top up balance

Sequence Num	Caller	Callee	Msg Name	Msg Type	Msg Param	Msg Constrain
1	Student	System	Request Login	Synchronous	Valid ID	-
2	System	Student	Request Valid ID	Synchronous	-	-
3	Student	System	Enter ID	Synchronous	Username, Password	Unique Student number
4	System	Data base	Verify ID	Synchronous	Credentials	-
5	Data Base	System	Valid	Synchronous	-	Valid Credentials
6	System	Student	Select Option	Synchronous	-	-
7	Student	System	Select Top-up Balance option	Synchronous	-	-
8	System	Student	Select amount	Synchronous	-	-
9	Student	System	Add amount	Synchronous	-	Number
10	System	Student	Enter card details	Synchronous	-	-
11	Student	System	Card details	Synchronous	Num, expiry date and cvc	
12	System	Data Base	Verify card details	Synchronous	-	-
13	Data Base	System	[ok]card verified	Synchronous	-	-
14	System	Student	Confirm Amount	Synchronous	Amount	-
15	Student	System	ok	Synchronous	-	-
16	System	Data base	Update TopUp Balance	Synchronous	Top Up Balance	-
17	Data Base	System	[ok]TopUp Updated	Synchronous	-	-
18	System	TopUp Balance	Current TopUp Balance	Synchronous	-	-
19	TopUp Balance	System	[ok]	Synchronous	-	-
20	System	Student	Current TopUp Balance	Synchronous	-	-
21	System	Student	Select Option	Synchronous	-	-
22	Student	System	[No]Exit	-	-	-

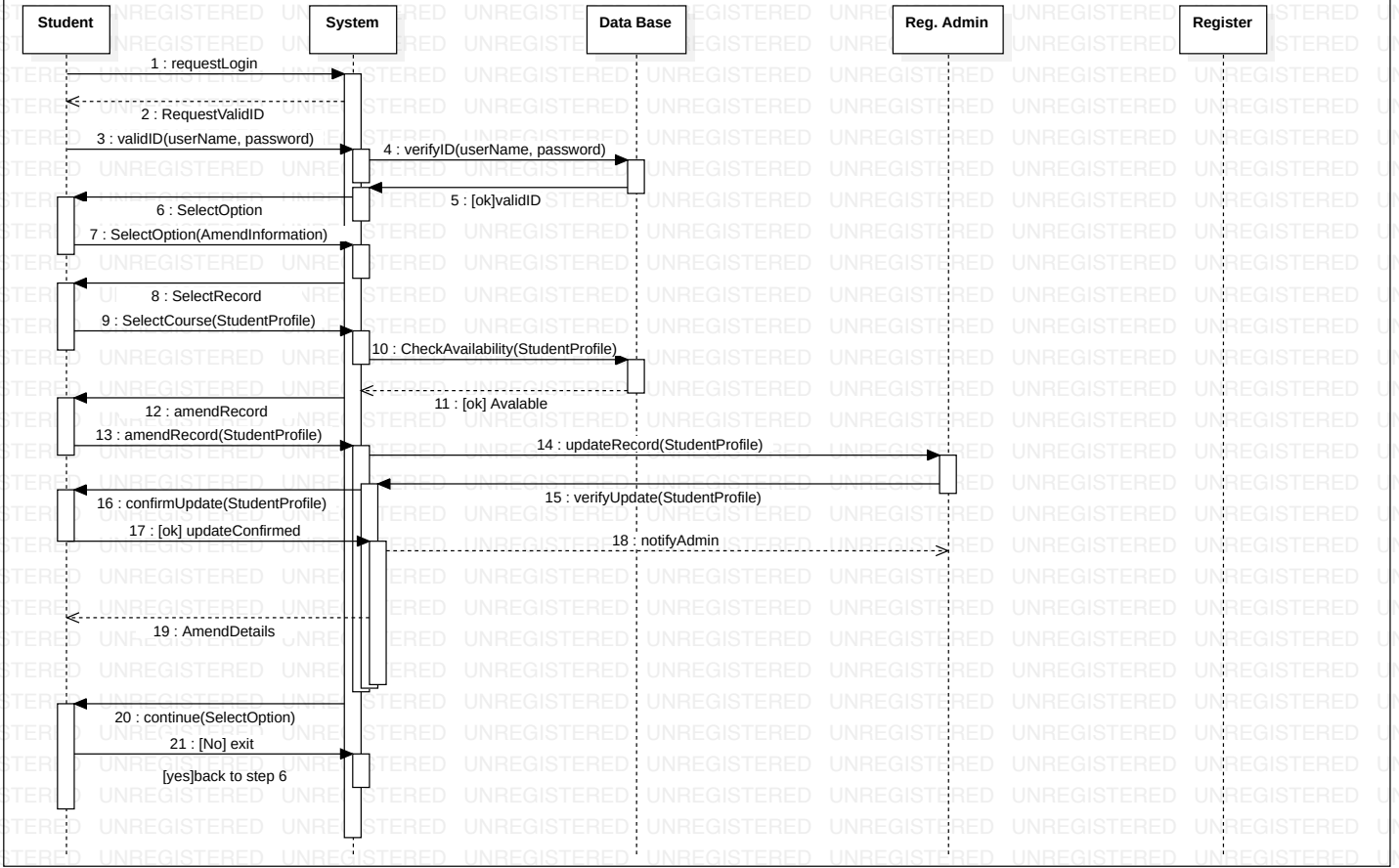
sd Check Notification



Message Analysis Table – Check notification

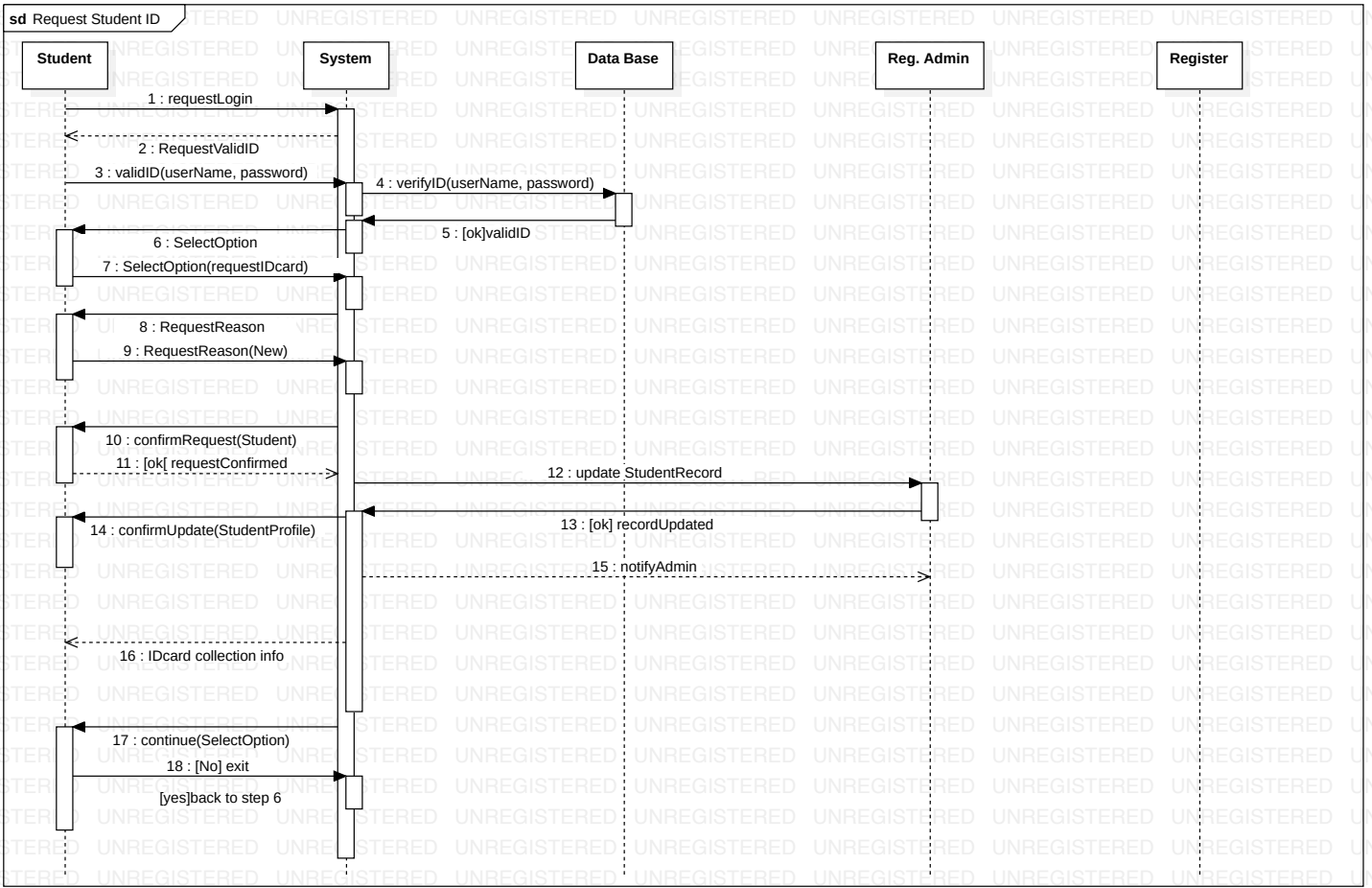
Sequence Num	Caller	Callee	Msg Name	Msg Type	Msg Param	Msg Constrain
1	Student	System	Request Login	Synchronous	Valid ID	-
2	System	Student	Request Valid ID	Synchronous	-	-
3	Student	System	Enter ID	Synchronous	Username, Password	Unique Student number
4	System	Data base	Verify ID	Synchronous	Credentials	-
5	Data Base	System	Valid	Synchronous	-	Valid Credentials
6	System	Student	Select Option	Synchronous	-	-
7	Student	System	Select check Notification	Synchronous	-	-
8	System	Data Base	Get Notification	Synchronous	-	-
9	Data Base	Notification	Student's Notification	Synchronous	-	-
10	Notification	System	[ok] notification available	Synchronous	Notification	-
11	System	Student	Notification Details	Synchronous	-	-
12	System	Student	Select Option	Synchronous	-	-
13	Student	System	[No]Exit	-	-	-

sd Amend Information



Message Analysis Table – Amend Information

Sequence Num	Caller	Callee	Msg Name	Msg Type	Msg Param	Msg Constrain
1	Student	System	Request Login	Synchronous	Valid ID	-
2	System	Student	Request Valid ID	Synchronous	-	-
3	Student	System	Enter ID	Synchronous	Username, Password	Unique Student number
4	System	Database	Verify ID	Synchronous	Credentials	-
5	Data Base	System	Valid	Synchronous	-	Valid Credentials
6	System	Student	Select Option	Synchronous	-	-
7	Student	System	Select Amend Information	Synchronous	-	-
8	System	Student	Select Record	Synchronous	-	-
9	Student	System	Select Student Profile	Synchronous	Student Profile	-
10	System	Data Base	Check Student Profile Availability	Synchronous	Student Profile	=Available
11	Data Base	System	[Ok]Availability	Synchronous	Yes	-
12	System	Student	Amend Record	Synchronous	Student Details	-
13	Student	System	Amend Student Details	Synchronous	Student Details	-
14	System	Data base	Update student Record	Synchronous	Student Details	-
15	Reg. Admin	System	Verify updated student Details	Synchronous	Verify Details	-
16	System	Student	Confirm updated student Details	Synchronous	Student Details	-
17	Student	System	Yes/No	Synchronous	-	-
18	System	Reg. Admin	Student Update notification	Synchronous	-	-
19	System	Student	Amended details	Synchronous	-	-
20	System	Student	Select Option	Synchronous	-	-
21	Student	System	[No]Exit	-	-	-



Message Analysis Table –Request Student_ID_Card

Sequence Num	Caller	Callee	Msg Name	Msg Type	Msg Param	Msg Constrain
1	Student	System	Request Login	Synchronous	Valid ID	-
2	System	Student	Request Valid ID	Synchronous	-	-
3	Student	System	Enter ID	Synchronous	Username, Password	Unique Student number
4	System	Data base	Verify ID	Synchronous	Credentials	-
5	Data Base	System	Valid	Synchronous	-	Valid Credentials
6	System	Student	Select Option	Synchronous	-	-
7	Student	System	Select Request ID card	Synchronous	-	-
8	System	Student	Request Reason	Synchronous	-	-
9	Student	System	New	Synchronous	Student ID card	-
10	System	Student	Confirm Request	Synchronous	-	-
11	Student	System	[ok] confirmed	Synchronous	-	-
12	System	Data Base	Update Student Record	Synchronous	-	-
13	Data Base	System	[ok]record Updated	Synchronous	-	-
14	System	Reg. Admin	Request Notification	Synchronous	Student ID card	Post collection
15	System	Student	ID card collection info	Synchronous	-	-
16	System	Student	Select Option	Synchronous	-	-
17	Student	System	[No]Exit	-	-	-

Test Cases

Test Cases

Name:	TC-01: Registration/enrolment test
Requirement:	User can make course registration/enrolment.
Preconditions:	User must be logged in to the system with valid credentials.
Steps:	<ol style="list-style-type: none">1. User enters credentials to log in.2. User selects the “Course Registration” option.3. User selects the specific course.4. User confirms selected course.5. User gets confirmation of the chosen course.6. User returns to the home page/exits the system.
Expected results:	<ol style="list-style-type: none">1. User successful is registered/enrolled.2. Notification of completed registration is received.

Name:	TC-02: Amends account details test.
Requirement:	User can amend account details.
Preconditions:	User must be logged in to the system with valid credentials.
Steps:	<ol style="list-style-type: none">1. User enters credentials to log in.2. User selects the “Amend details” option.3. User selects the specific field to amend.4. User type new details information.5. User clicks the save button.6. User gets an alert notification of successfully amended information.7. User returns to the home page/exits the system.
Expected results:	<ol style="list-style-type: none">1. User successfully amended detail.2. Notification of completed amended information is received.

Name:	TC-03: Request student ID card test.
Requirement:	User can make requests to order a student ID card.
Preconditions:	User must be logged in to the system with valid credentials.
Steps:	<ol style="list-style-type: none"> 1. User enters credentials to log in. 2. User selects the "Print Student Card" option. 3. User selects delivery/collection address. 4. User confirms delivery/collection address. 5. User gets alert confirmation of the chosen delivery/collection method. 6. User returns to the home page/exits the system.
Expected results:	<ol style="list-style-type: none"> 1. User successfully order the Student ID Card 2. Notification of completed Student ID Card request is received.

Name:	TC-04: Top-up balance test.
Requirement:	User can make a top-up balance.
Preconditions:	User must be logged in to the system with valid credentials.
Steps:	<ol style="list-style-type: none"> 1. User enters credentials to log in. 2. User selects the "Top Up" option. 3. User selects the "Add money" option. 4. User types the amount to Top up. 5. User type credit card details information. 6. User clicks the Send button. 7. User gets an alert notification of successfully Top up the balance. 8. User returns to the home page/exits the system.
Expected results:	<ol style="list-style-type: none"> 1. User successfully Top Up Balance. 2. Notification of completed Top Up transaction is received.

Name:	TC-05: Course withdrawal test.
Requirement:	User can request for course withdrawal.
Preconditions:	User must be logged in to the system with valid credentials.
Steps:	<ol style="list-style-type: none"> 1. User enters credentials to log in. 2. User selects the “Withdraw Course” option. 3. User selects the course to withdraw. 4. User click the save button. 5. User gets alert confirmation of successful withdrawal course. 6. User returns to the home page/exits the system.
Expected results:	<ol style="list-style-type: none"> 3. User successfully withdraw the course. 4. Notification of completed withdrew of the chosen course is received.