

BSC. HONS IN BUSINESS INFORMATION TECHNOLOGY

SUBMITTED TO

Dr Ray Stoneham

University of Greenwich, UK.



SUBMITTED BY

Name: Md. Shahidul Islam

Id: 000989727

Date of submission: 13th April, 2017

TEAM: TECHBROSDIA

SITE URL:

TEAM MEMBERS AND THEIR ROLES

S/L	MEMBER NAME	ROLE
1	DELUAR JAHAN	DESIGNER & PROGRAMMER
2	MD.SHAHIDUL ISLAM (ME)	ANALYST & ARCHITECT
3	MD. JONAET RUBEL	DATABASE DESIGNER
4	MAHZABIN HOQUE	TESTER

REPOSITORY URL:

SCREENCAST URL:

Contents

CREDENTIALS:	1
EVALUATION OF PRODUCT AND PROCESS	2
Evaluation of product.....	2
Strength.....	3
Weakness	5
Development process:	6
EVALUATION OF TEAM	6
SELF EVALUATION.....	7
Appendix1: Requirement analysis	8
Requirements list:.....	8
Functional requirements.....	8
Non-functional	10
MOSCOW Prioritization	10
Prioritized requirements with the minimum usable subset:.....	14
Use case diagrams.....	27
Use case description	33
Appendix 2: System architecture.....	72
Initial class diagram.....	72
Final class diagram:.....	73
Sequence Diagram:	74
Activity diagram	85
Appendix 3: SCRUM artifacts	90
SCRUM Product Backlog:	90
Sprint backlog	97
Sprint 1:.....	97
Sprint 2:.....	99
Sprint 3:.....	102
Sprint 4:.....	105
Burn Down chart(Less detailed).....	107
Detailed burn down chart.....	108

CREDENTIALS:

Role	Full Name	User Name	Password
Administrator	Ray Stoneham	admin	admin
EC Manager	Siplu Sinha	siplu123	123456
EC Coordinator	Mustafizur Rahman	mustafizur123	123456
	Nazrana Haque	nazrana12	123456
	Nayeema Rahman	nayeema11	123456
Student	Lucy Riley	lucy123	123456
	Deluar Jahan	jahan11	123456
	Shahidul Islam	shahidul11	123456
	Junayet Rubel	junayet11	123456
	Mahzabin Hoque	mahzabin11	123456

EVALUATION OF PRODUCT AND PROCESS

Evaluation of product

Our target was to confirm the best. From my point of view, we successfully did that.



Strength

Functionality: The system is enriched with some extra features and functions with all necessary features and functionalities. **Our team believes that client says those they want but we have to deliver what they actually need.**

Usability: Metaphors are used through all over the application. Most important features and function are presented on top of the hierarchy. Meaningful error messages will let the user know about the error they made. We knew color talks. That's why effective and appropriate color were chosen used in the interfaces. Data presentation was one of our major goals. We used table, dropdown and other UI elements.

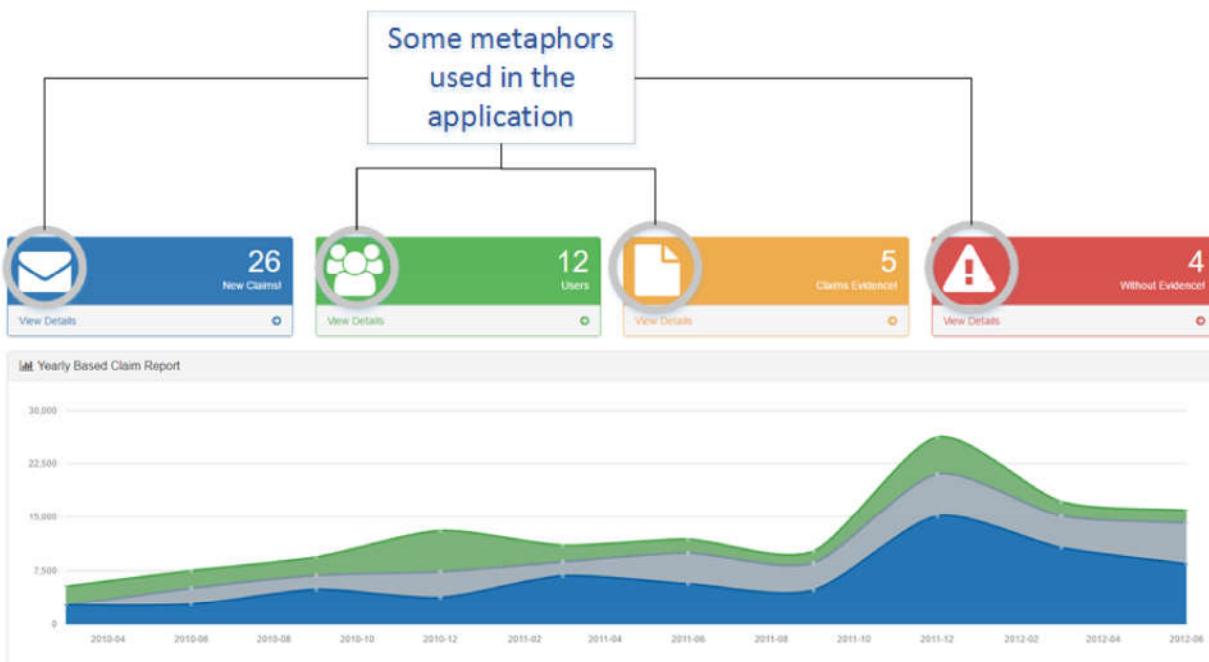


Figure 1: Some metaphors used in the application

Maintainability: We followed MVC pattern while developing the application. Again, clean code architecture was designed before designing and coding.

Efficiency: As the system can be defined as a part of a full university management system; every line of code need to be efficient enough. Our programmer tried his best to ensure efficiency.

Security: Security was the first concern of our team as different level of users can access the system. All passwords are encrypted to ensure security. Re-login or re-authentication is applied where there is matter of extreme decision making. Again, code is strong enough to protect SQL injection.

```

function createRandomPassword() {
    $chars = "abcdefghijklmnopqrstuvwxyz0123456789";
    srand((double) microtime() * 1000000);
    $i = 0;
    $pass = '';

    while ($i <= 11) {
        $num = rand() % 33;
        $tmp = substr($chars, $num, 1);
        $pass = $pass . $tmp;
        $i++;
    }

    return $pass;
}

```

Figure 2: Password will be encrypted.

Responsive Interfaces: The application layout is device responsive. From huge screen to minimum mobile screen user can easily brows.

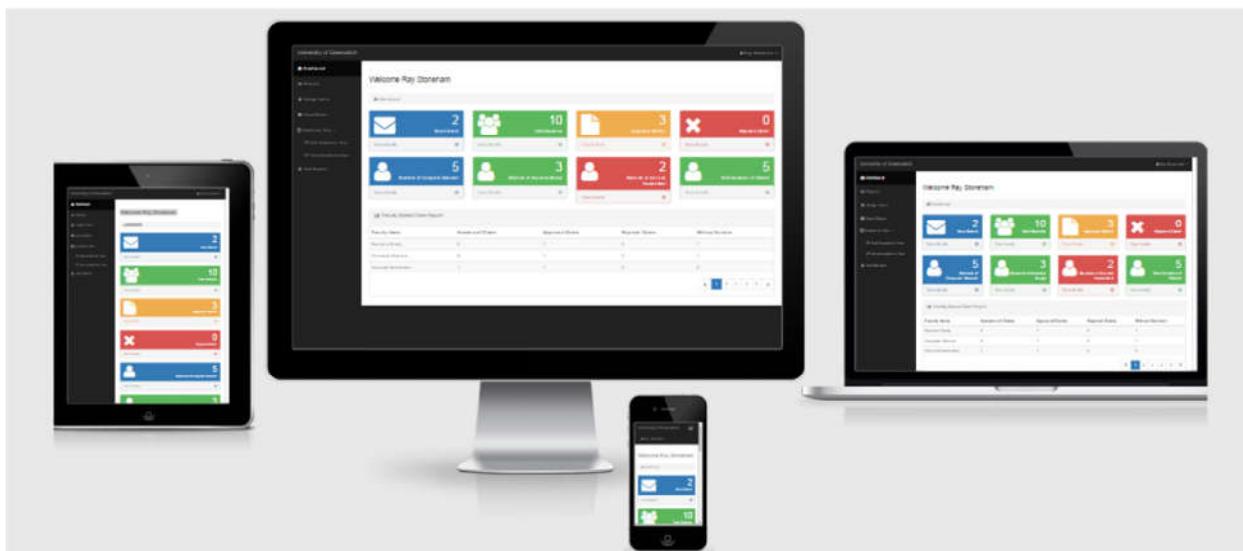


Figure 3: Responsive interfaces in different devices

Weakness

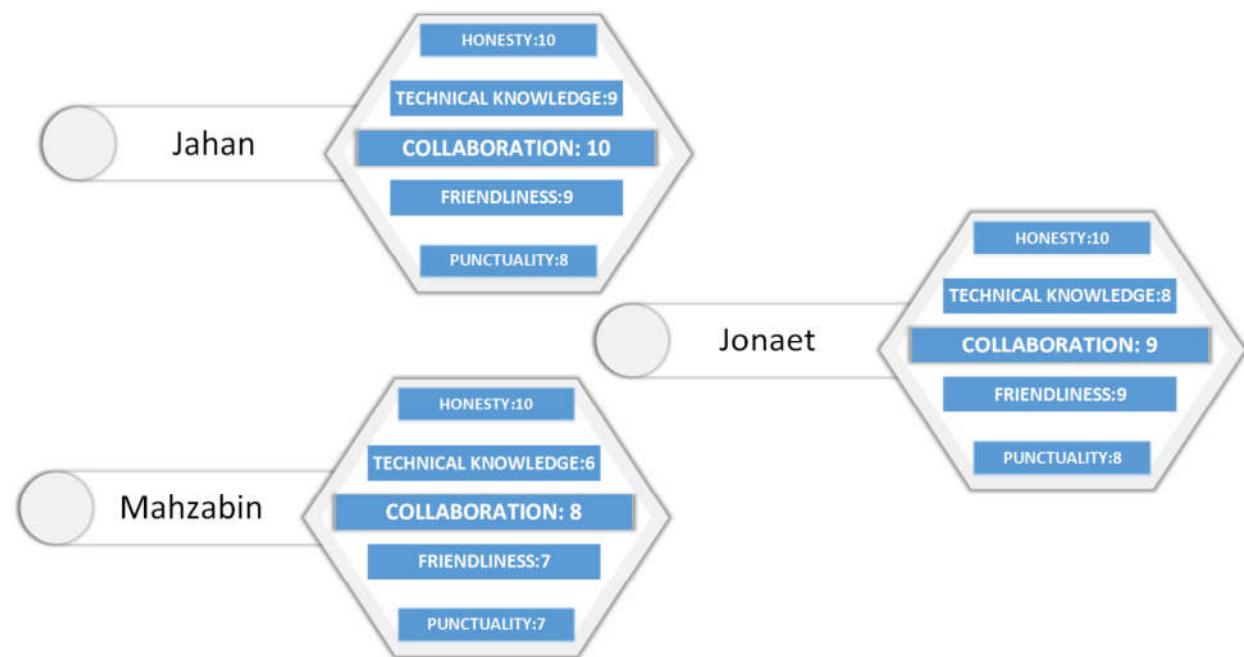
- No password recovery options.
- Once evidence are submitted, user can't delete them.

Development process:

Throughout the development we followed SCRUM methodology.

EVALUATION OF TEAM

The team members have been evaluated in five characteristics. Honesty, technical knowledge, collaboration, friendliness and punctuality. From my point of view, all members were good enough. Our programmer was tremendous. I will give him maximum marks in here. Then the position is for the database designer. Our tester was little bit of irregular when we started our work. But with days, she got herself on right track.

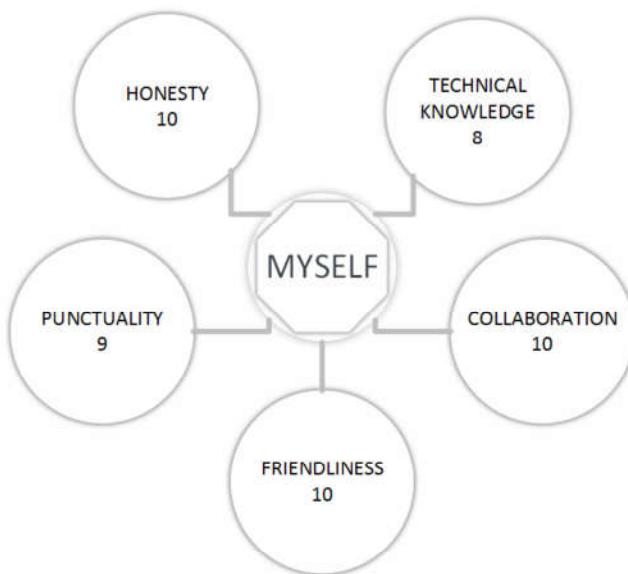


SELF EVALUATION

As an analyst and system architect, I tried to follow the best practices. Very first the system requirements (both functional and nonfunctional) were gathered by reading the scenario and the link given in the scenario. Then I applied MOSCOW prioritization. In the next step, I have prioritized the requirements with minimum usable subset. I have drawn use case diagrams and written use case descriptions.

Journey as an architect throughout the project was tremendous. I have drawn initial class diagram, final class diagrams, activity diagrams, sequence diagrams. I have followed MVC pattern here. I took help of my team members in some cases.

During this assignment, I have learnt so many new things. But the most important thing I have learnt is **team work**. For the very first time, I had to complete an assignment that requires team contribution. I thought I will not be that comfortable working as a team. But I have proved myself wrong. I have also evaluated myself in the same criteria as my team.



Appendix1: Requirement analysis

Requirements list:

Both functional and non-functional requirements were gathered.

Functional requirements

1. Users must have their individual dashboard according to roles.
2. Admin must can manage EC Manager.
3. Admin must can update his/her username and password.
4. Admin must can manage faculty.
5. Admin must can manage course under a faculty.
6. Admin must can manage subject for a course.
7. Admin must can manage assessment item for a subject with due date.
8. Admin must can set closure date for academic year.
9. Admin must manage EC coordinator.
10. Admin can assign EC coordinator to a faculty.
11. Admin must can manage student with their username and password.
12. Admin must can manage Claim Type.
13. Admin must can manage Claim circumstances.
14. Admin must can manage Evidence Type.
15. EC manager must can update his/her username and password.
16. EC manager must can view all claims.
17. EC coordinator must can update his/her username and password.
18. EC coordinator must can access his/her faculty students' claims only.

19. EC coordinator must can approve/ reject his/her faculty students claims those have evidence attached.
20. EC coordinator must can see the number of claims report for a specific year within his/her faculty.
21. EC manager must can see the claim percentage report of faculties for a particular academic year.
22. EC coordinator must can see number of students claiming a particular claim in specific academic year within his/her faculty.
23. EC coordinator must can see report on claims without due evidence uploaded within his/her faculty.
24. EC coordinator must can see undecided claims after 14 days of submission for his/her faculty.
25. Student must can submit multiple claims for a specific item of an assessment.
26. Student must can be able to see own claims and containing decisions.
27. Student must can submit evidence for his/her claim till final closure.
28. System must disable claim submitting option after due date for claiming.
29. System must send an email notification to due EC coordinator when a student from his/her faculty submit any claim.
30. System must send an email notification that contains EC coordinators decision to the specific student after his/her claim being processed.
31. ECC can create a specific date's claims report.
32. ECC can create a specific assessment items claims report.

Non-functional

- ❖ The user interface must be device responsive.
- ❖ Reports should be enriched with statistical analysis.
- ❖ All Password should be encrypted.

MOSCOW Prioritization

Must Have	<ol style="list-style-type: none">1. Users must have their individual dashboard according to roles.2. Admin must can manage EC Manager.3. Admin must can update his/her username and password.4. Admin must can manage faculty.5. Admin must can manage course under a faculty.6. Admin must can manage subject for a course.7. Admin must can manage assessment item for a subject with due date.8. Admin must can set closure date for academic year.9. Admin must manage EC coordinator.10. Admin can assign EC coordinator to a faculty.
-----------	---

	<p>11. Admin must can manage student with their username and password.</p> <p>12. Admin must can manage Claim Type.</p> <p>13. Admin must can manage Claim circumstances.</p> <p>14. Admin must can manage Evidence Type.</p> <p>15. EC manager must can update his/her username and password.</p> <p>16. EC manager must can view all claims.</p> <p>17. EC coordinator must can update his/her username and password.</p> <p>18. EC coordinator must can access his/her faculty students' claims only.</p> <p>19. EC coordinator must can approve/ reject his/her faculty students claims those have evidence attached.</p> <p>20. EC coordinator must can see the number of claims report for a specific year within his/her faculty.</p> <p>21. EC manager must can see the claim percentage report of faculties for a particular academic year.</p>
--	--

	<p>22. EC coordinator must can see number of students claiming a particular claim in specific academic year within his/her faculty.</p> <p>23. EC coordinator must can see report on claims without due evidence uploaded within his/her faculty.</p> <p>24. EC coordinator must can see undecided claims after 14 days of submission for his/her faculty.</p> <p>25. Student must can submit multiple claims for a specific item of an assessment.</p> <p>26. Student must can be able to see own claims and containing decisions.</p> <p>27. Student must can submit evidence for his/her claim till final closure.</p> <p>28. System must disable claim submitting option after due date for claiming.</p> <p>29. System must send an email notification to due EC coordinator when a student from his/her faculty submit any claim.</p> <p>30. System must send an email notification that contains EC coordinators decision to the</p>
--	---

	specific student after his/her claim being processed.
Should Have	<ol style="list-style-type: none">1. Reports should be enriched with statistical analysis.2. All Password should be encrypted.
Could Have	<ol style="list-style-type: none">1. ECC can create one particular dates' claims report.2. ECC can create one particular assessment items claims report.
Won't Have	<ol style="list-style-type: none">1. No social media sharing options.

Prioritized requirements with the minimum usable subset:

Requirement Identifier	Requirement Short Name	Requirement Description	Acceptance Criteria
FR01 (Functional)	Login	As an user(ECM,ECC, Admin,Student) , I want to login to the system.	<ul style="list-style-type: none"> • Same login window for every users.
FR02 (Functional)	Individual dashboard	As an user(ECM,ECC, Admin,Student) , I want to have my individual dashboard.	<ul style="list-style-type: none"> • Password needs to be encrypted.

FR03 (Functional)	ECM managing	As an Administrator , I want to manage ECM.	<ul style="list-style-type: none"> • Newly added ECM can only access his/her dashboard. • Second time authentication is needed in case of updating.
FR04 (Functional)	Faculty managing	As an Administrator , I want to Manage faculty.	<ul style="list-style-type: none"> • All faculty should be listed dynamically in a table/ drop down with update and delete button. • Second time authentication is needed in case of

			updating and deleting.
FR05 (Functional)	Course managing	As an Administrator , I want to manage course.	<ul style="list-style-type: none">• Second time authentication is needed in case of deleting.• Course should be listed in a dropdown/table.• Operation need to happen dynamically.• Searching option is needed.

FR06 (Functional)	Subject management	As an Administrator , I want to manage subject for a course.	<ul style="list-style-type: none"> • Second time authentication is needed in case of deleting. • Course should be listed in a dropdown/table. • Operation need to happen dynamically. • Searching option is needed.

FR07 (Functional)	Assessment item management	As an Administrator I want to manage assessment item.	<ul style="list-style-type: none"> • Second time authentication is needed in case of deleting. • Assessment item should be listed in a dropdown/table. • Operation need to happen dynamically. • Searching option is needed.
FR08 (Functional)	Closure date setting	As an Administrator I want to set	<ul style="list-style-type: none"> • System must check if the

		academic years closure date.	<p>date is before current date</p> <ul style="list-style-type: none"> • There must be a date chooser. • Operation need to happen dynamically.
FR09 (Functional)	ECC management	As an Administrator I want to manage ECC.	<ul style="list-style-type: none"> • Second time authentication is needed in case of deleting. • Operation need to happen dynamically. • Searching option is needed.

FR10 (Functional)	ECC assigning	As an Administrator I want to assign ECC to a faculty.	<ul style="list-style-type: none"> • System must take assigning date. • There must be a date chooser.
FR11 (Functional)	Student management	As an Administrator I want to manage student with username and password.	<ul style="list-style-type: none"> • Second time authentication is needed in case of deleting. • Operation need to happen dynamically. • Searching option is needed.

FR12 (Functional)	Claim viewing	As an ECM I want to View all claims.	<ul style="list-style-type: none"> Claims should be list in a table.
FR13 (Functional)	Claim accessing	As an ECC I want to access my faculty students' claims.	<ul style="list-style-type: none"> Claims should be list in a table. Searching option is needed.
FR14 (Functional)	Claim action	As an ECC I want to approve or reject my faculty students' claims.	<ul style="list-style-type: none"> Claims should be list in a table with approve, reject button. Searching option is needed.

FR15 (Functional)	Number of claims report	As an ECC I want to see report on number of claims in a specific year.	<ul style="list-style-type: none"> • A drop down box for selecting year. • PDF conversion option.
FR16 (Functional)	Claims percentage report	As an ECM I want to see report on faculties claim percentage for a year.	<ul style="list-style-type: none"> • A drop down box for selecting year. • PDF conversion option.
FR17 (Functional)	Particular claim report	As an ECC I want to see student number for a particular claim in a year within my faculty.	<ul style="list-style-type: none"> • PDF conversion option. • A drop down box for

			<p>selecting year.</p> <ul style="list-style-type: none">• A drop down box for selecting a particular claim.
FR18 (Functional)	Claims with no evidence	As an ECC I want to see report on claims containing no evidence within my faculty.	
FR19 (Functional)	Undecided claims	As an ECC I want to see all undecided claims. After 14 days within my faculty.	

FR20 (Functional)	Claim submission	As a Student I want to submit multiple claims for an assessment item.	
FR21 (Functional)	Own claims and decisions	As a Student I want to see my own claims and decisions.	
FR22 (Functional)	Evidence submitting	As a student I want to continue submitting evidence for my claim till closure date.	
FR23 (Functional)	Disable claim	As a System I want to disable new claims after due date for claiming.	

FR24 (Functional)	Sending email notification (to ECC)	As a System I want to send email notification to due ECC when any student submit any claim.	
FR25 (Functional)	Sending email notification (to student)	As a System I want to send email notification about ECCs' decision on a claim that being processed to due student.	
FR26 (Functional)	Specific date's claims report.	As an Administrator I want to create a specific date's claims report.	<ul style="list-style-type: none"> • A date chooser for selecting date.

FR27 (Functional)	Specific assessment items claims report	As an Administrator I want to create a specific assessment items claims report.	<ul style="list-style-type: none"> • A date chooser for selecting date.
NFR28 (Non-Functional)	Responsive interface	As a user I want suitable interface for all kind of devices.	
NFR29 (Non-Functional)	Statistical analysis	As a user I want to see reports with statistical analysis.	<ul style="list-style-type: none"> • Different kind of chart like pie chart, bar chart, line chart or others.

Use case diagrams

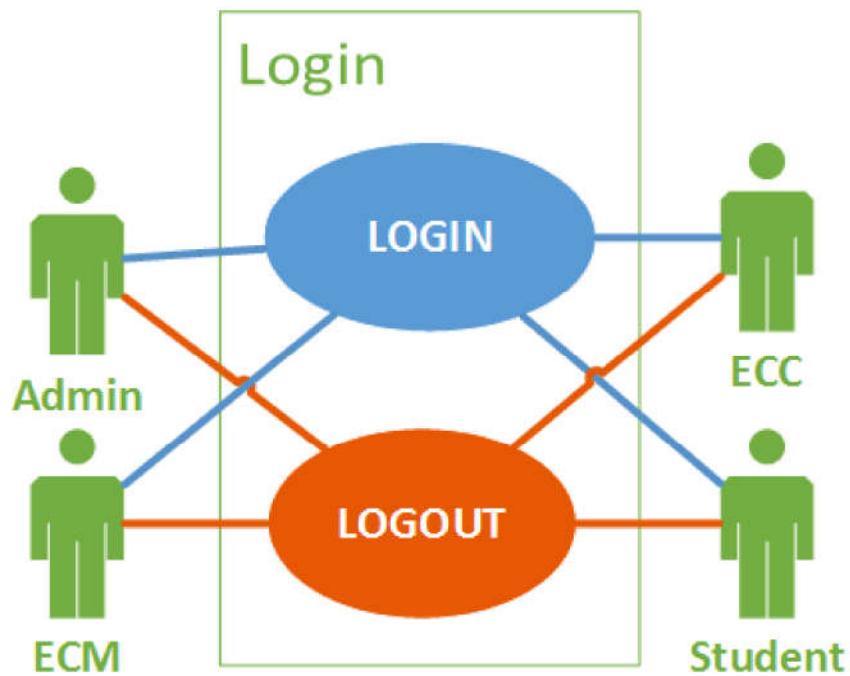


Figure 4: Login use case

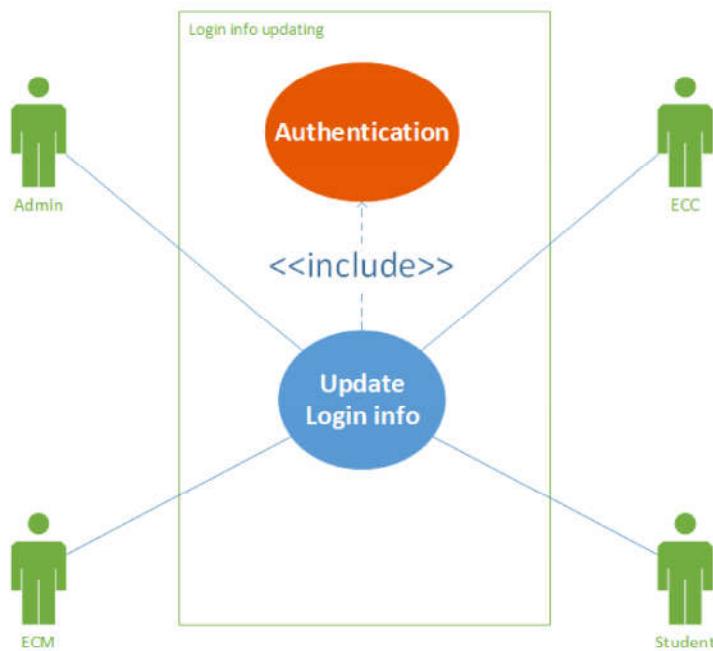


Figure 5: Login info updating

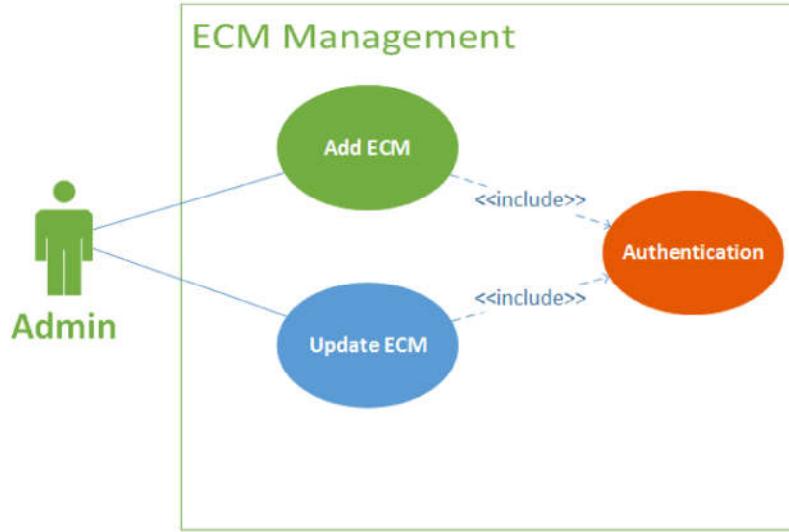


Figure 6: ECM Management

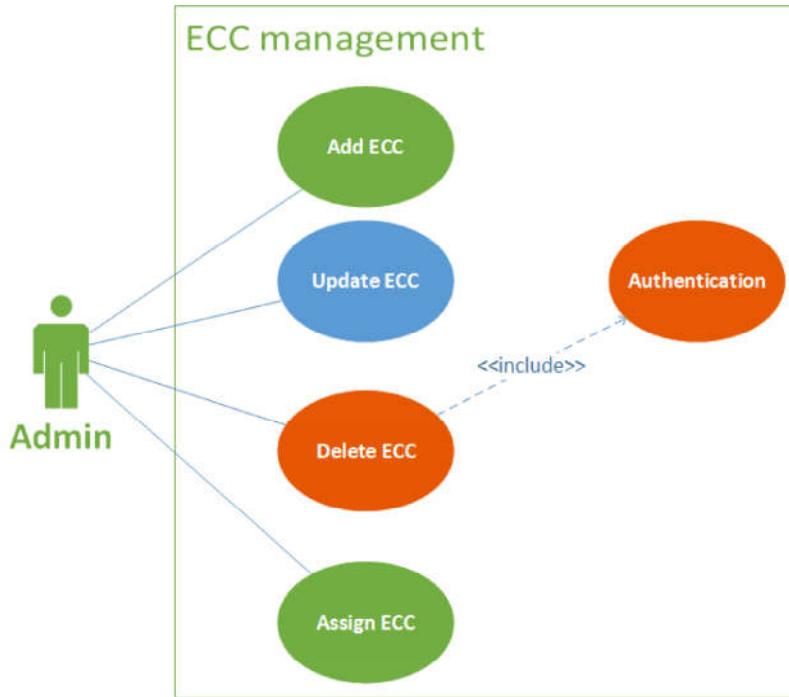


Figure 7: ECC Management

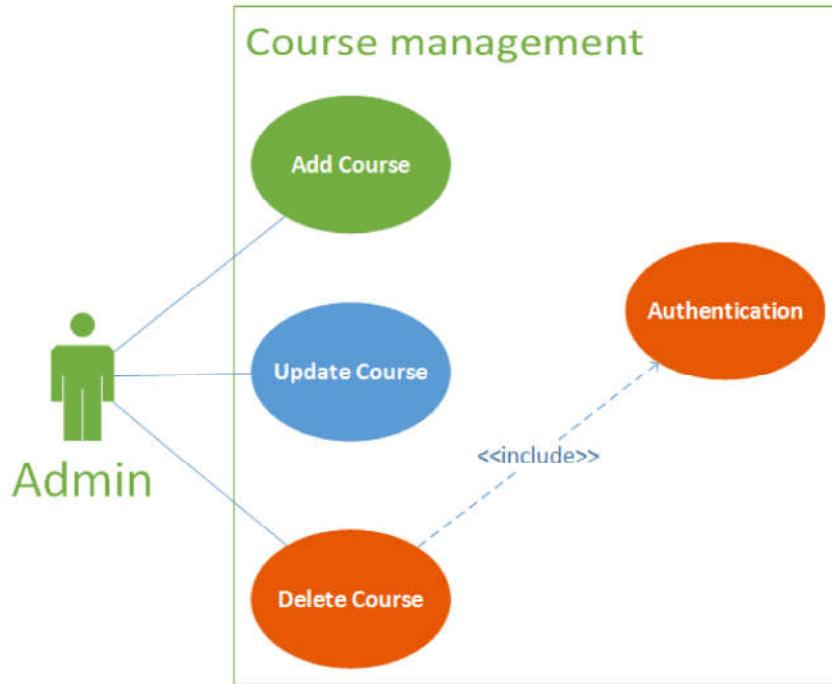


Figure 8: Course Management

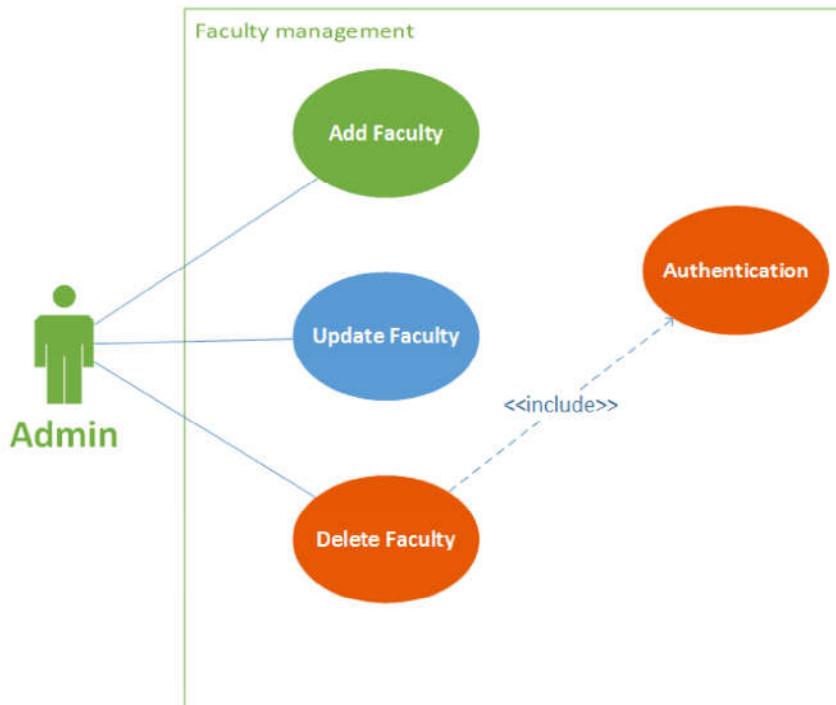


Figure 9: Faculty Management

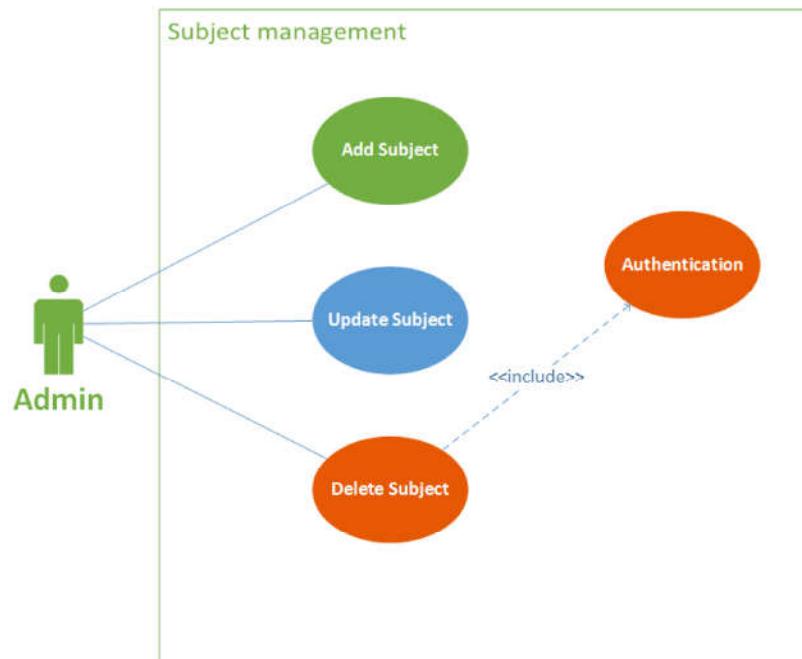


Figure 10: Subject Management

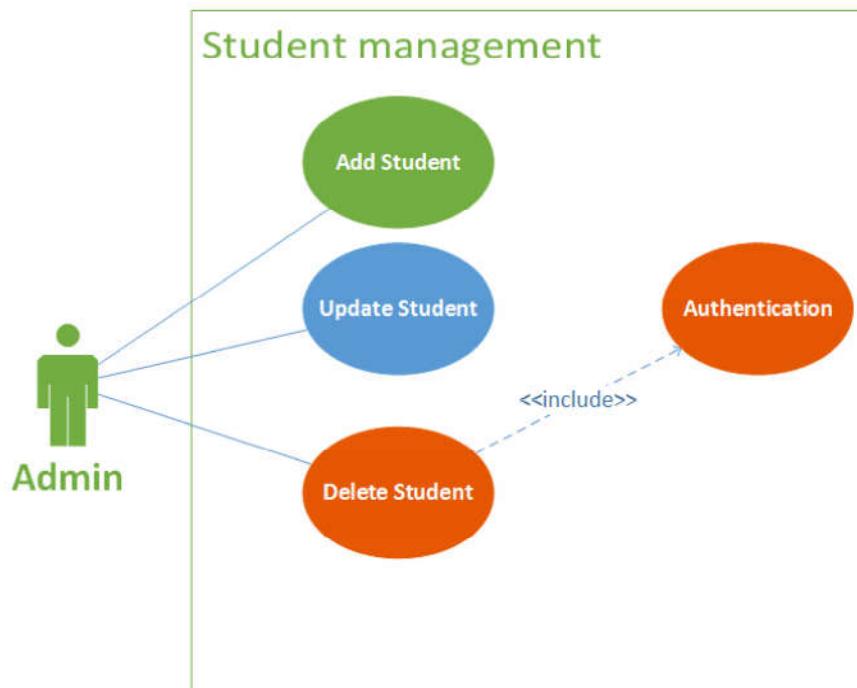


Figure 11: Student Management

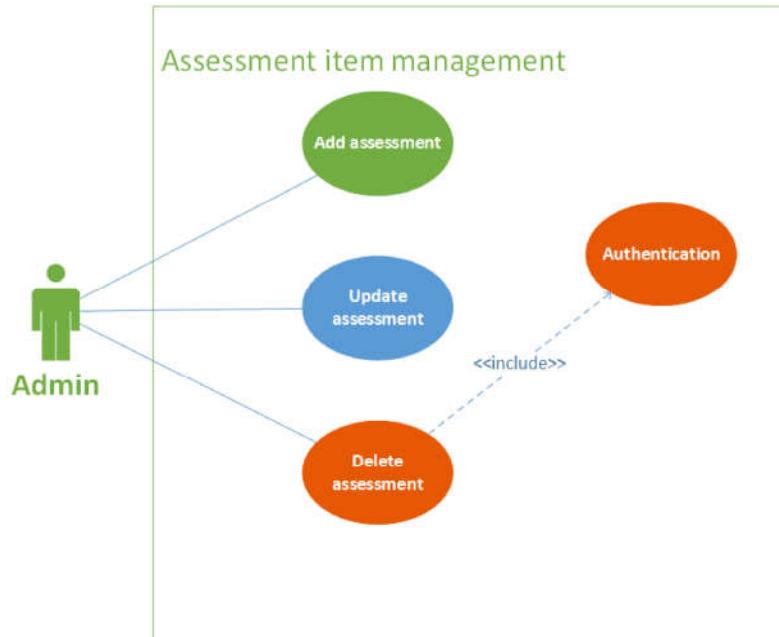


Figure 12: Assessment Management

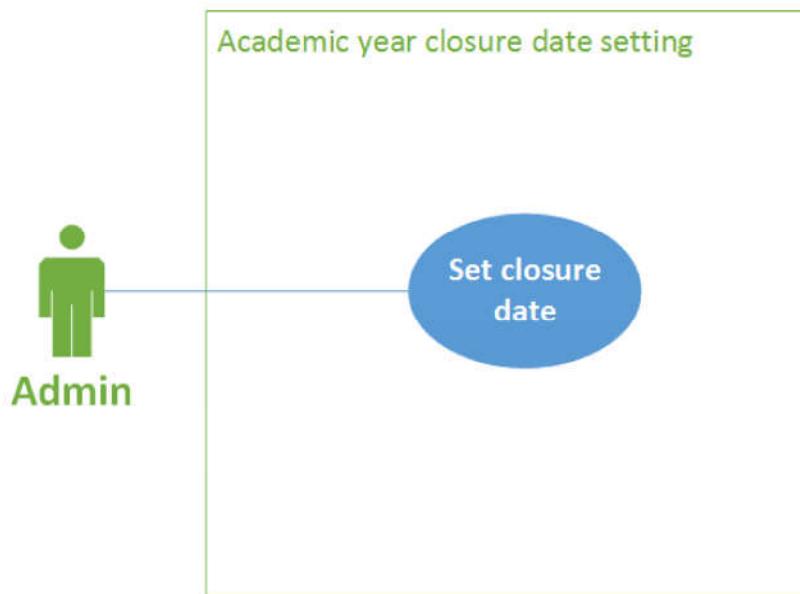


Figure 13: Academic year closure date setting

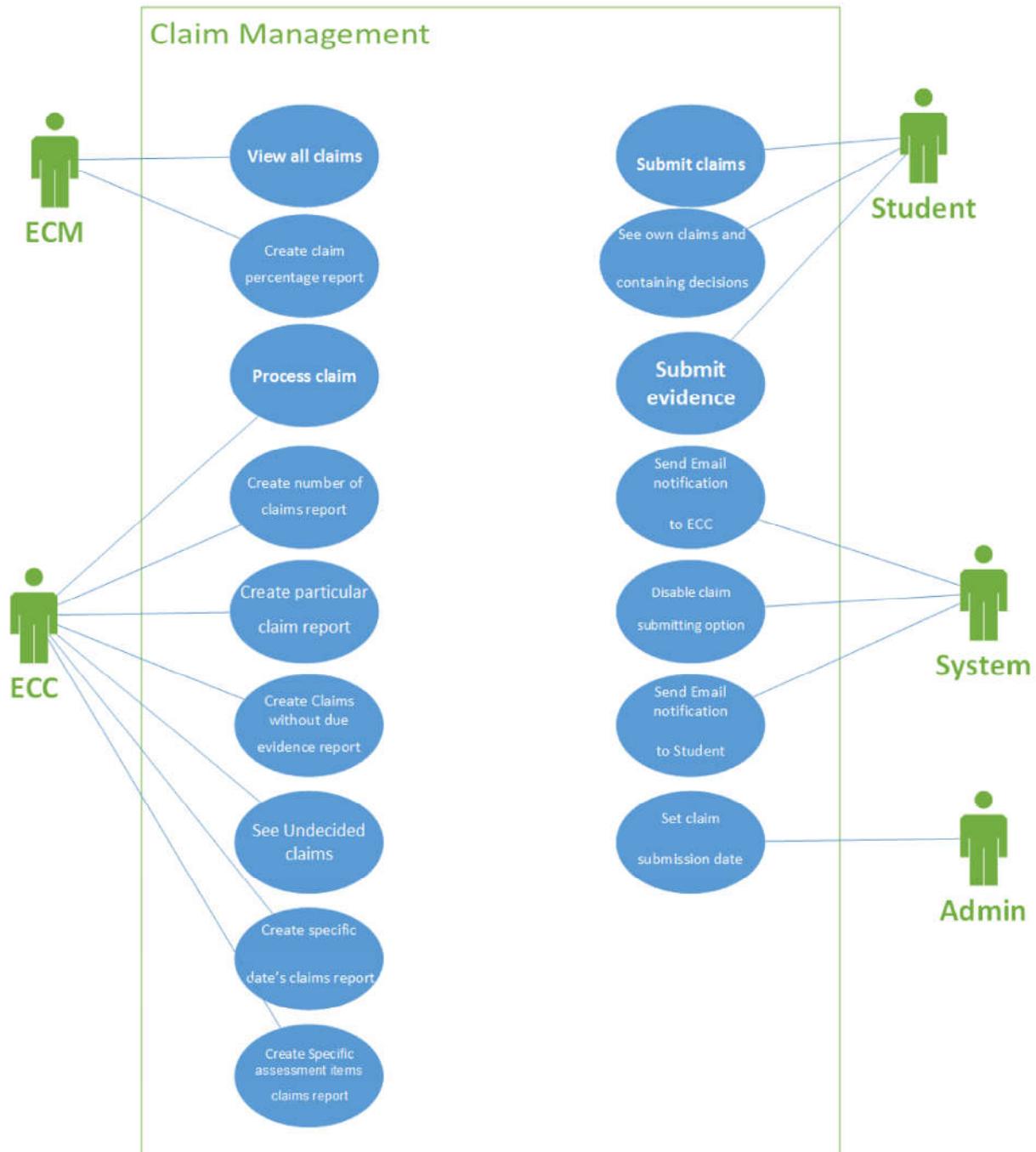


Figure 14: Claim Management

Use case description

USE CASE: 1	
Title	User login
Actor	Admin, ECM, ECC, Student
Scenario	User provides username and password. System check if login info is correct. If information is correct user is redirected to his/her due panel/ dashboard.
Pre-condition	None
Post-condition	None
Dependent Use case	None

USE CASE: 2	
Title	ECM adding
Actor	Admin
Scenario	Admin will switch to ECM panel. Then enter new ECM info and click add button.
Pre-condition	<ul style="list-style-type: none"> • Admin must be logged in. • Second time authentication. • Empty field validation.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown. • Previous ECM login will be restricted.
Dependent Use case	Use case 1

USE CASE: 3	
Title	ECM updating
Actor	Admin
Scenario	Admin will switch to ECM panel. Then select an ECM. Update info from text fields and click update button.
Pre-condition	<ul style="list-style-type: none"> • Admin must be logged in. • Second time authentication. • Empty selection check. • Empty field validation.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 4	
Title	Login info updating
Actor	Admin, ECM, ECC, Student
Scenario	User will login to the system. Navigate to his/her profile. Change login info.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Second time authentication. • Empty field validation.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 5	
Title	Faculty adding
Actor	Admin
Scenario	Admin will login to the system. Navigate to faculty panel. Add new faculty.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty field validation.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 6	
Title	Faculty updating
Actor	Admin
Scenario	Admin will login to the system. Navigate to faculty panel. Select a faculty. Update faculty.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty field validation. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 7	
Title	Faculty deleting
Actor	Admin
Scenario	Admin will login to the system. Navigate to faculty panel. Select a faculty. Delete faculty.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check. • Second time authentication.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 8	
Title	Course adding
Actor	Admin
Scenario	Admin will login to the system. Navigate to faculty panel. Select a faculty. Enter new Course info. Click Add button.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 9	
Title	Course updating
Actor	Admin
Scenario	<p>Admin will login to the system. Navigate to faculty panel. Select a faculty. Select course.</p> <p>Provide info. Click update button.</p>
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 9	
Title	Course deleting
Actor	Admin
Scenario	<p>Admin will login to the system. Navigate to faculty panel. Select a faculty. Select course.</p> <p>Click delete button.</p>
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check. • Second time authentication.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 10	
Title	Subject adding
Actor	Admin
Scenario	Admin will login to the system. Navigate to faculty panel. Select a faculty. Select a Course. Enter new subject info. Click Add button.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 11	
Title	Subject updating
Actor	Admin
Scenario	Admin will login to the system. Navigate to faculty panel. Select a faculty. Select Course. Select subject. Provide info. Click Update button.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 12	
Title	Subject deleting
Actor	Admin
Scenario	Admin will login to the system. Navigate to faculty panel. Select a faculty. Select Course. Select subject. Click delete button.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check. • Second time authentication.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 13	
Title	Assessment item adding with due date
Actor	Admin
Scenario	Admin will login to the system. Navigate to faculty panel. Select a faculty. Select a Course. Select a subject. Enter new assessment item info. Choose date. Click Add button.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check. • System checks if date is after current date.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 14	
Title	Assessment item updating
Actor	Admin
Scenario	Admin will login to the system. Navigate to faculty panel. Select a faculty. Select a Course. Select a subject. Select Assessment item. Provide info. Click Update button.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 15	
Title	Assessment item deleting
Actor	Admin
Scenario	Admin will login to the system. Navigate to faculty panel. Select a faculty. Select a Course. Select a subject. Select Assessment item. Click delete button.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check. • Second time authentication.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 16	
Title	Academic year closure date setting.
Actor	Admin
Scenario	Admin will login to the system. Select closure date from date chooser. Click Add button.
Pre-condition	<ul style="list-style-type: none"> • User must be logged in. • Empty selection check. • System checks if date is after current date.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 17	
Title	ECC adding
Actor	Admin
Scenario	Admin will login to the system. Then switch to ECC panel. Then enter new ECC info and click add button.
Pre-condition	<ul style="list-style-type: none"> • Admin must be logged in. • Empty field validation.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 18	
Title	ECC updating
Actor	Admin
Scenario	Admin will login to the system. Then switch to ECC panel. Then select an ECC. Update info from text fields and click update button.
Pre-condition	<ul style="list-style-type: none"> • Admin must be logged in. • Empty selection check. • Empty field validation.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 19	
Title	ECC deleting
Actor	Admin
Scenario	Admin will login to the system. Then switch to ECC panel. Then select/ search an ECC. Click Delete button.
Pre-condition	<ul style="list-style-type: none"> • Admin must be logged in. • Second time authentication. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 20	
Title	ECC assigning
Actor	Admin
Scenario	Admin will login to the system. Then switch to ECC panel. Then select/ search an ECC. Then select a faculty. Click assign button.
Pre-condition	<ul style="list-style-type: none"> • Admin must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 21	
Title	Student adding with username and password.
Actor	Admin
Scenario	Admin will login to the system. Then switch to Student panel. Then enter new Student info and click add button.
Pre-condition	<ul style="list-style-type: none"> • Admin must be logged in. • Empty field validation.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 22	
Title	Student updating
Actor	Admin
Scenario	Admin will login to the system. Then switch to Student panel. Then select/ search one student. Provide new information. Then click update button.
Pre-condition	<ul style="list-style-type: none"> • Admin must be logged in. • Empty selection check. • Empty field validation.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 23	
Title	Student deleting
Actor	Admin
Scenario	Admin will login to the system. Then switch to Student panel. Then select/ search one student. Click Delete button.
Pre-condition	<ul style="list-style-type: none"> • Admin must be logged in. • Second time authentication. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 24	
Title	Claim viewing
Actor	ECM
Scenario	ECM will login to the system. Then select a faculty. Then select a course. Then subject. Then assessment item. Click Claims button.
Pre-condition	<ul style="list-style-type: none"> • ECM must be logged in.
Post-condition	None
Dependent Use case	Use case 1

USE CASE: 25	
Title	Claim processing
Actor	ECC
Scenario	ECC will login to the system. Then Switch to claim panel. Click approve/reject button.
Pre-condition	<ul style="list-style-type: none"> • ECC must be logged in. • Only claimed those are attached with evidence will be process able.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 27	
Title	number of claims(monthly) report
Actor	ECC
Scenario	<p>ECC will login to the system. Then</p> <p>Switch to claim panel. Select number of claims report from a drop down/ table/.</p> <p>Select Year from a dropdown. Click Create report button.</p>
Pre-condition	<ul style="list-style-type: none"> • ECC must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 28	
Title	claim percentage report
Actor	ECM
Scenario	<p>ECM will login to the system. Then</p> <p>Switch to claim panel. Select claim percentage report from a drop down/table/. Select Year from a dropdown. Click Create report button.</p>
Pre-condition	<ul style="list-style-type: none"> • ECM must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 29	
Title	particular claim type report.
Actor	ECC
Scenario	ECC will login to the system. Then Switch to claim panel. Select particular claim type report from a drop down/ table/. Select Year from a dropdown. Click Create report button.
Pre-condition	<ul style="list-style-type: none"> • ECC must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 30	
Title	Claims without due evidence report.
Actor	ECC
Scenario	<p>ECC will login to the system. Then</p> <p>Switch to claim panel. Select Claims without due evidence report from a dropdown/ table/. Select Year from a dropdown. Click Create report button.</p>
Pre-condition	<ul style="list-style-type: none"> • ECC must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 31	
Title	Undecided claims
Actor	ECC
Scenario	ECC will login to the system. Then Switch to claim panel. Select Undecided claims from a drop down/ table/. Select Year from a dropdown. Undecided claims on table.
Pre-condition	<ul style="list-style-type: none"> • ECC must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 32	
Title	Claims submission for a specific item of an assessment.
Actor	Student
Scenario	<p>Student will login to the system. Then Switch to claim panel. Select Subject from a dropdown. Select assessment item from another dropdown. Provide claim details. Click submit button.</p>
Pre-condition	<ul style="list-style-type: none"> • Student must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1
Extensions	Student can be able to submit multiple claims for an assessment item.

USE CASE: 33	
Title	Own claims and containing decisions.
Actor	Student
Scenario	<p>Student will login to the system. Then Switch to claim panel. Select Subject from a dropdown. Select assessment item from another dropdown. See claim with their decisions in a table.</p>
Pre-condition	<ul style="list-style-type: none"> • Student must be logged in. • Empty selection check.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 34	
Title	Evidence submission
Actor	Student
Scenario	<p>Student will login to the system. Then Switch to claim panel. Select Subject from a dropdown. Select assessment item from another dropdown. Select a claim from table. Click brows evidence. Click submit.</p>
Pre-condition	<ul style="list-style-type: none"> • Student must be logged in. • Empty selection check. • Submitting date must not cross closure date.
Post-condition	<ul style="list-style-type: none"> • A success message will be shown.
Dependent Use case	Use case 1

USE CASE: 35	
Title	claim submitting option disabling.
Actor	System
Scenario	None
Pre-condition	<ul style="list-style-type: none"> • Current date must cross submission closure date.
Post-condition	None
Dependent Use case	None

USE CASE: 36	
Title	Email notification sending to ECC.
Actor	System
Scenario	None
Pre-condition	<ul style="list-style-type: none">• Student of due faculty must submit any claim.
Post-condition	None
Dependent Use case	USE CASE 32

USE CASE: 37	
Title	Email notification sending to student.
Actor	System
Scenario	None
Pre-condition	<ul style="list-style-type: none">• Due ECC must process due students claim.
Post-condition	None
Dependent Use case	USE CASE 32, 25

USE CASE: 38	
Title	Specific date's claims report.
Actor	ECC
Scenario	ECC will login to the system. Then Switch to claim panel. Select a date from date chooser. Click Create report.
Pre-condition	<ul style="list-style-type: none"> • ECC must be logged in.
Post-condition	None
Dependent Use case	USE CASE 1

USE CASE: 39	
Title	Specific assessment items claims report.
Actor	ECC
Scenario	ECC will login to the system. Then Switch to claim panel. Select Subject from a dropdown. Select assessment item from another dropdown. Click Create report.
Pre-condition	<ul style="list-style-type: none"> • ECC must be logged in.
Post-condition	None
Dependent Use case	USE CASE 1

Appendix 2: System architecture

Initial class diagram

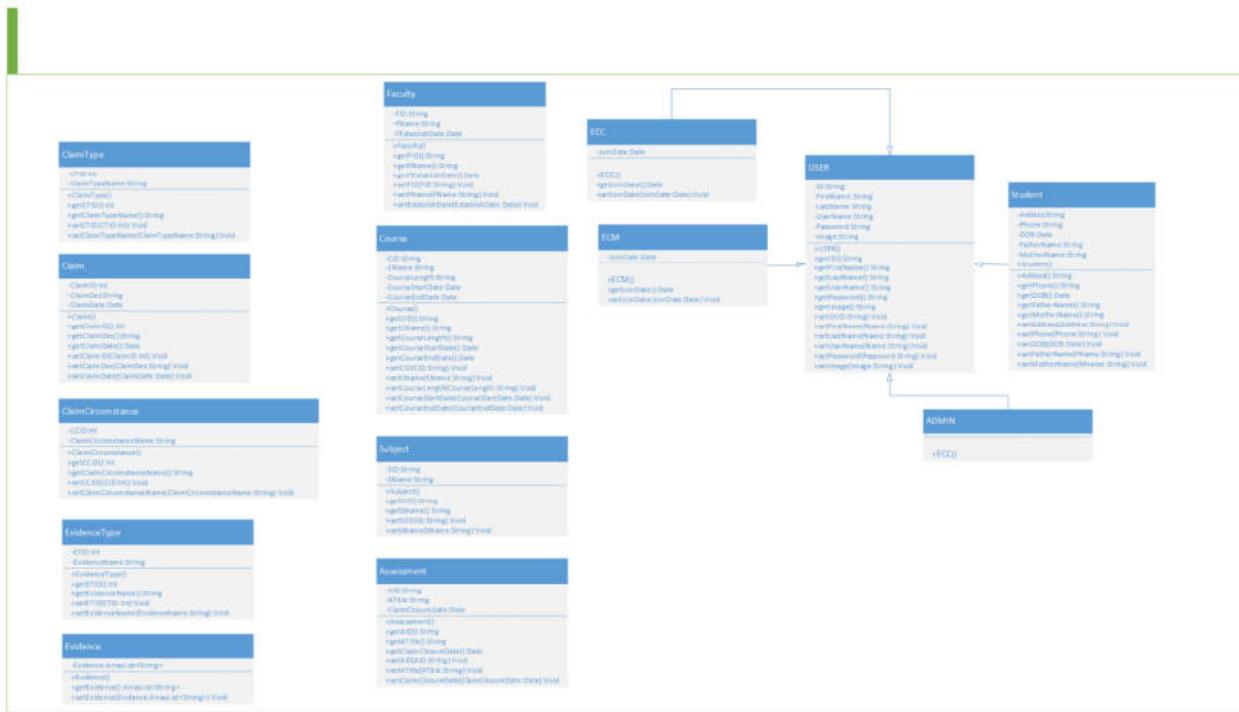


Figure 15: Initial class diagram

Final class diagram:

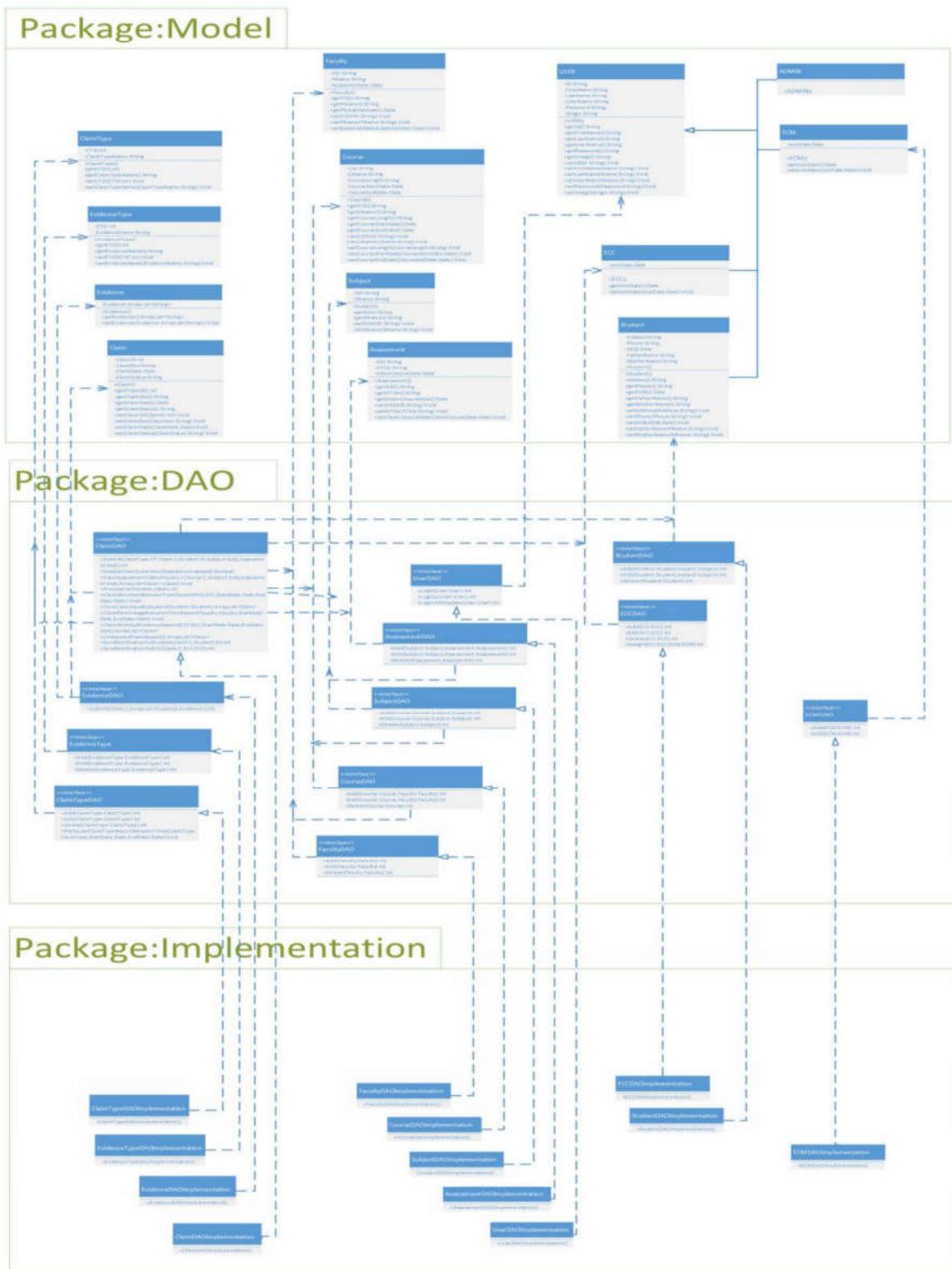


Figure 16: Final class diagram

Sequence Diagram:

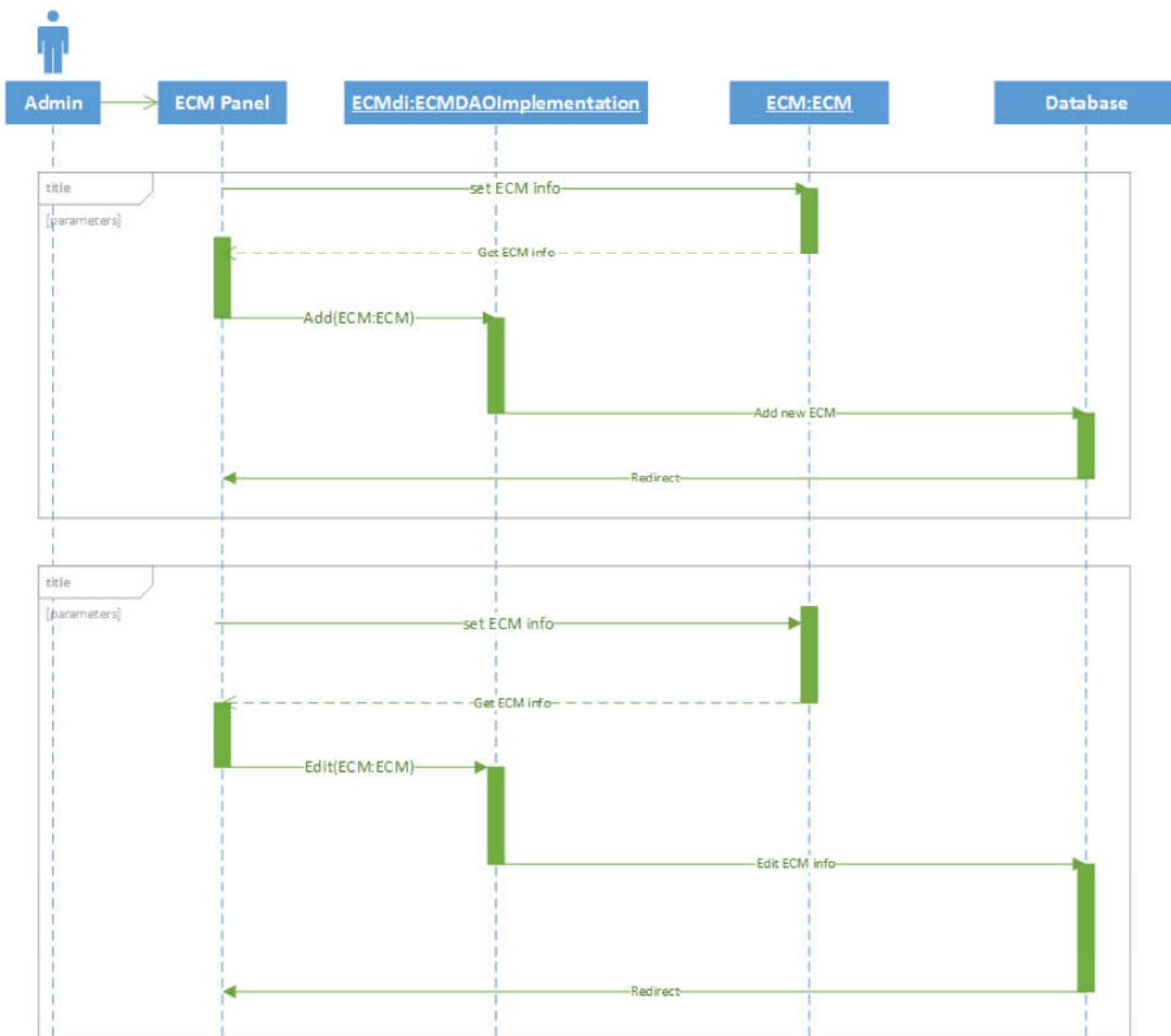


Figure 17: ECM Management

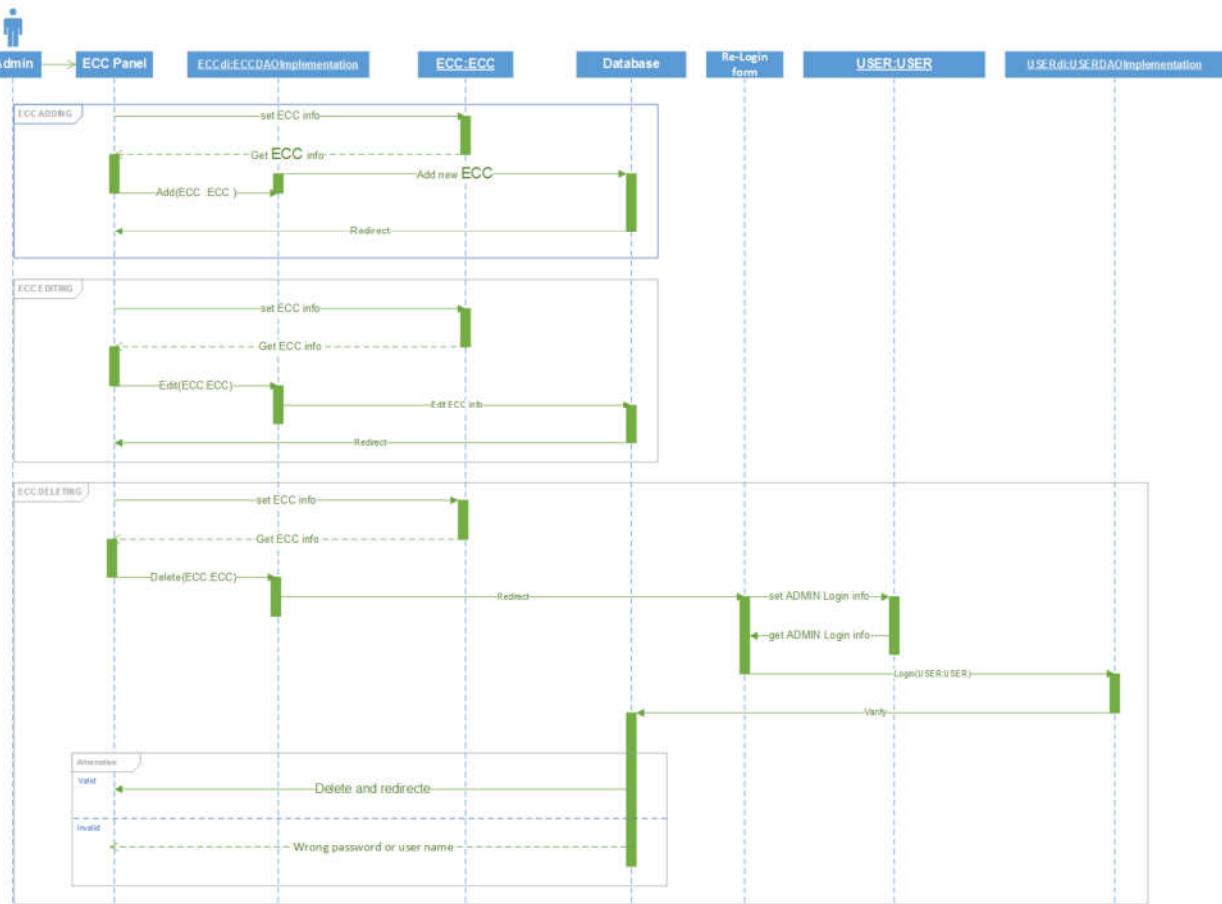


Figure 18: ECC Management

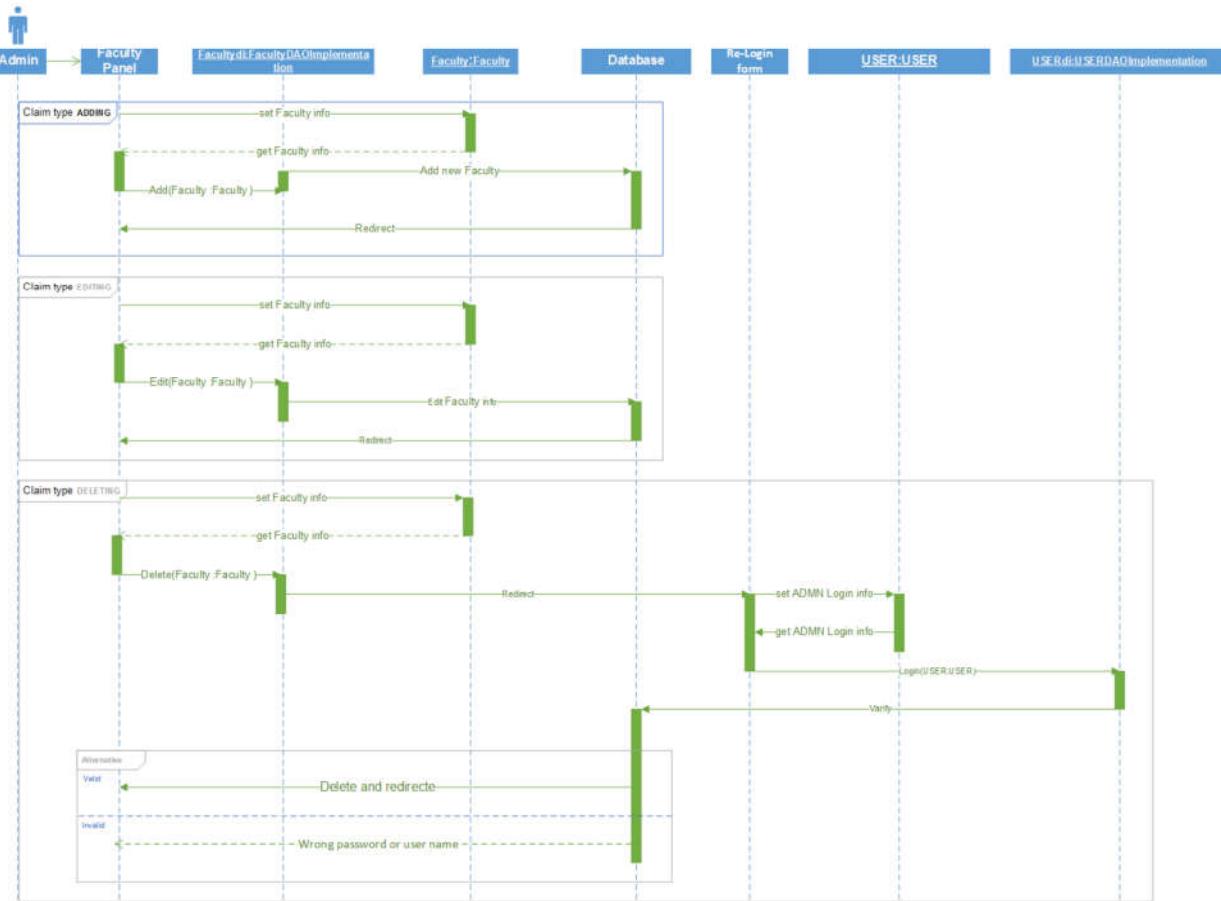


Figure 19: Faculty Management

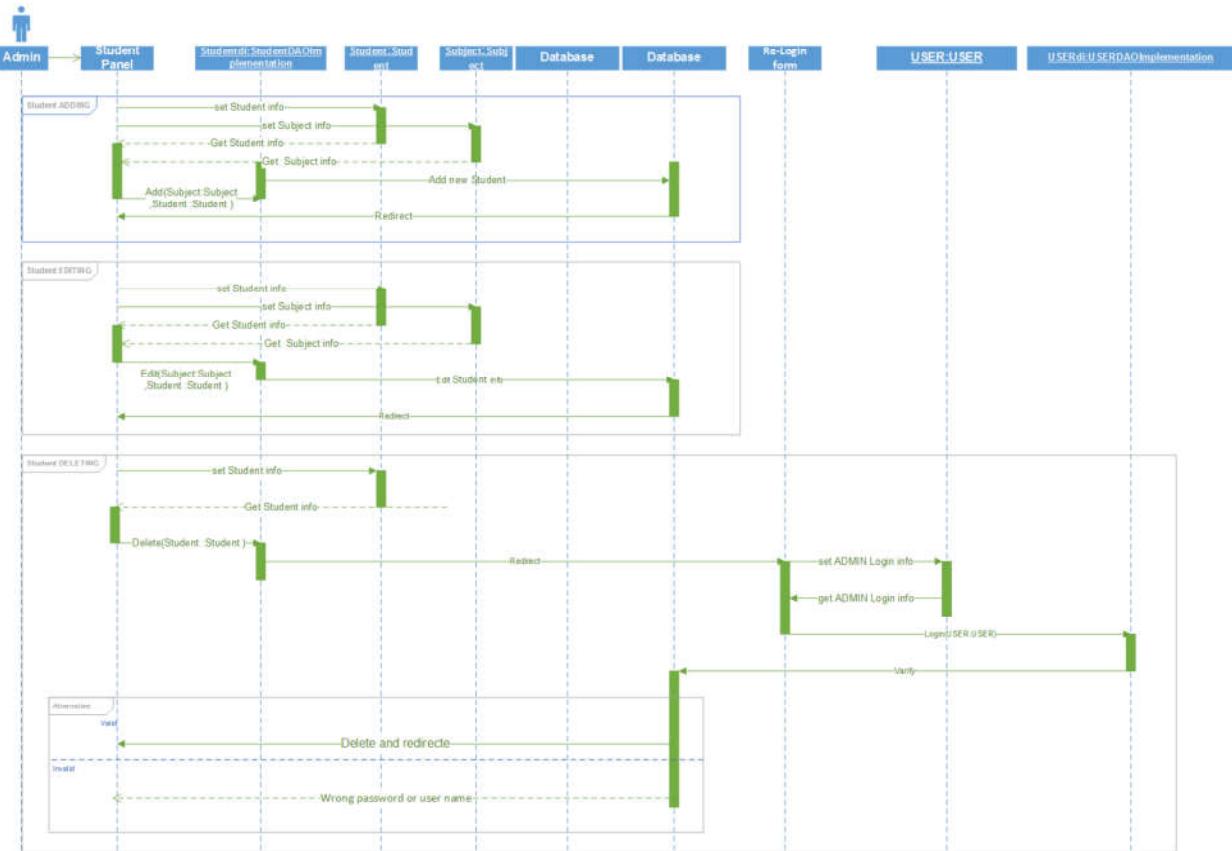


Figure 20: Student Management

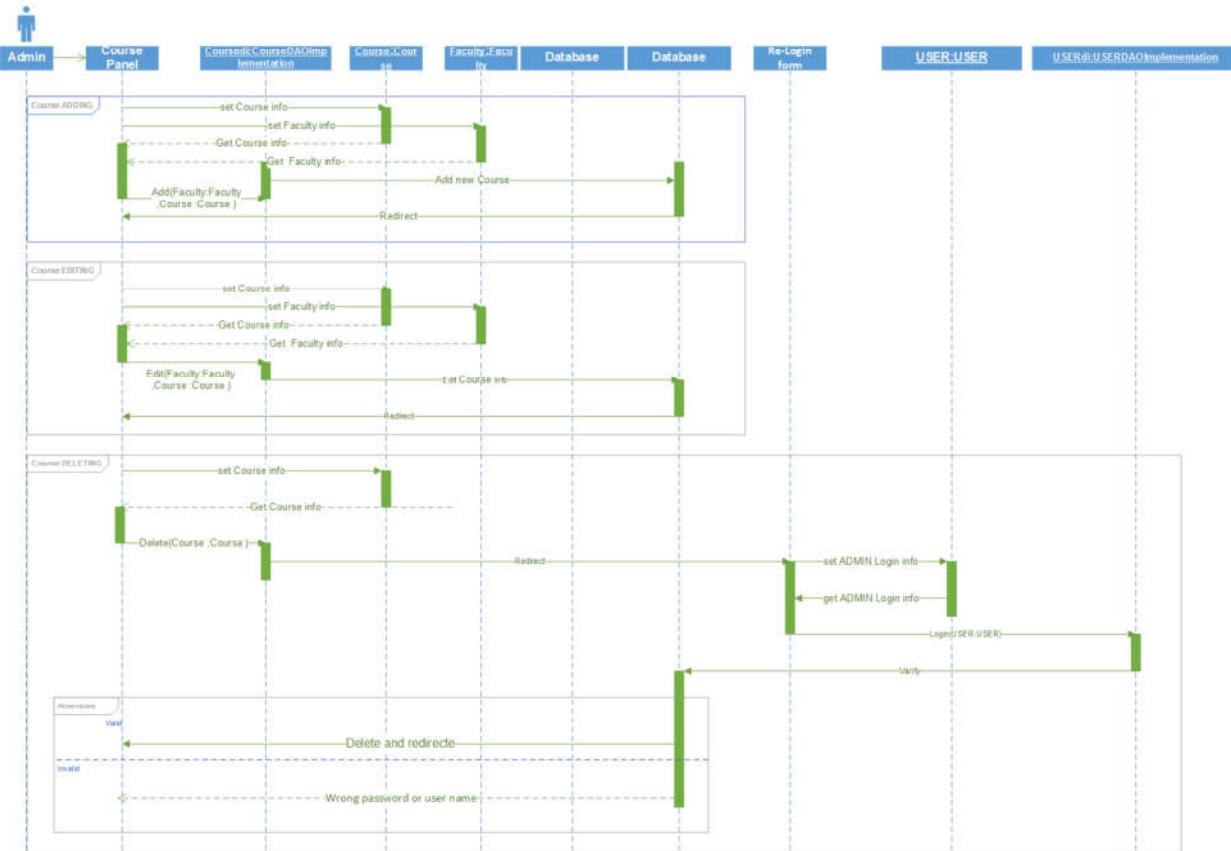


Figure 21: Course Management

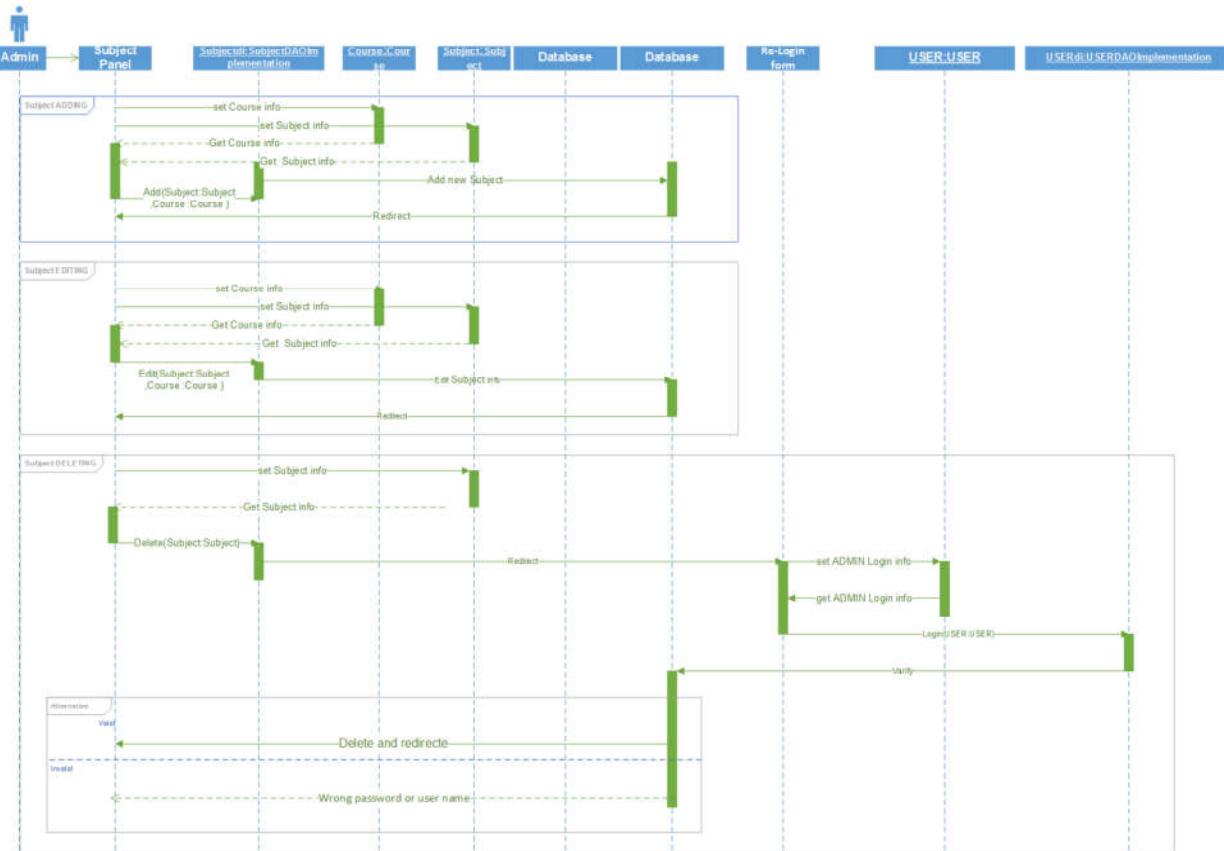


Figure 22: Subject Management

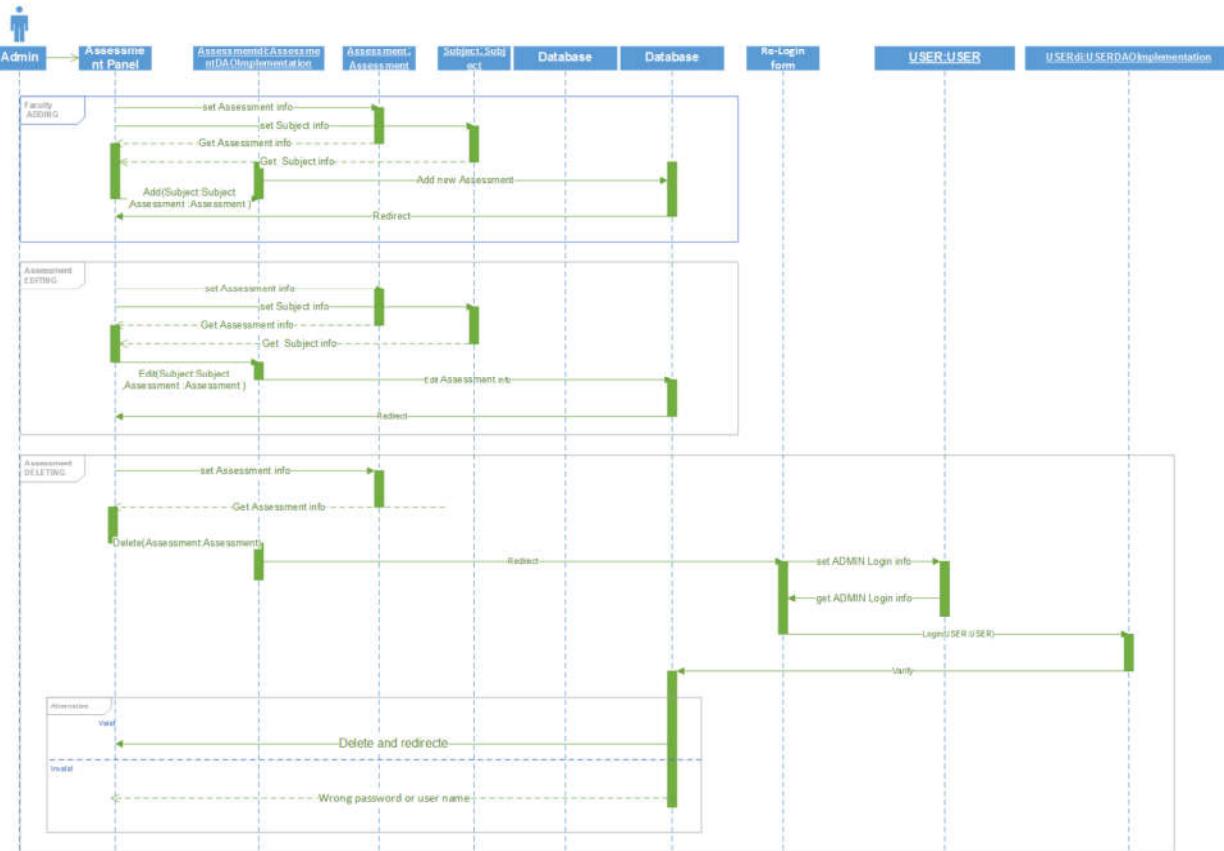


Figure 23: Assessment Management

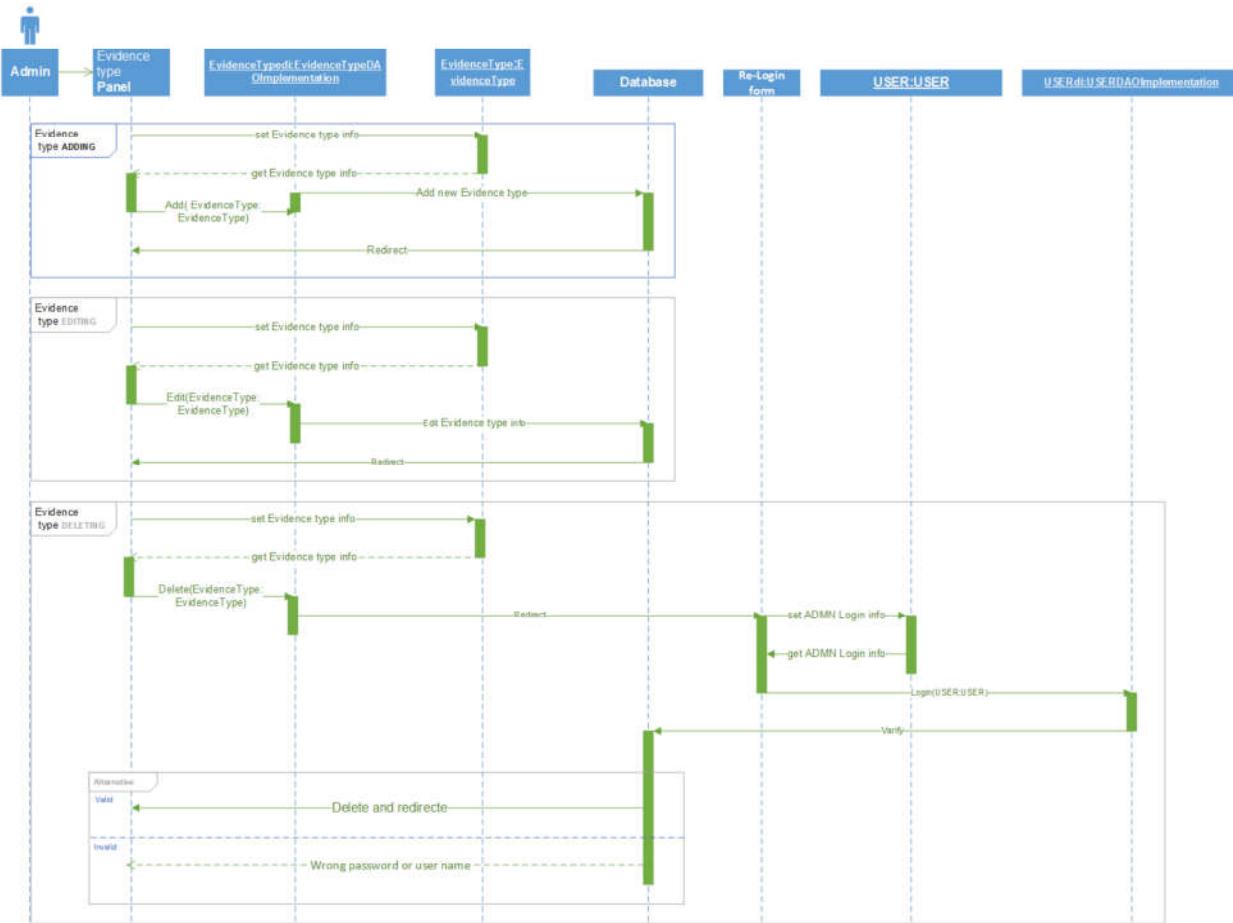


Figure 24: Evidence Type Management

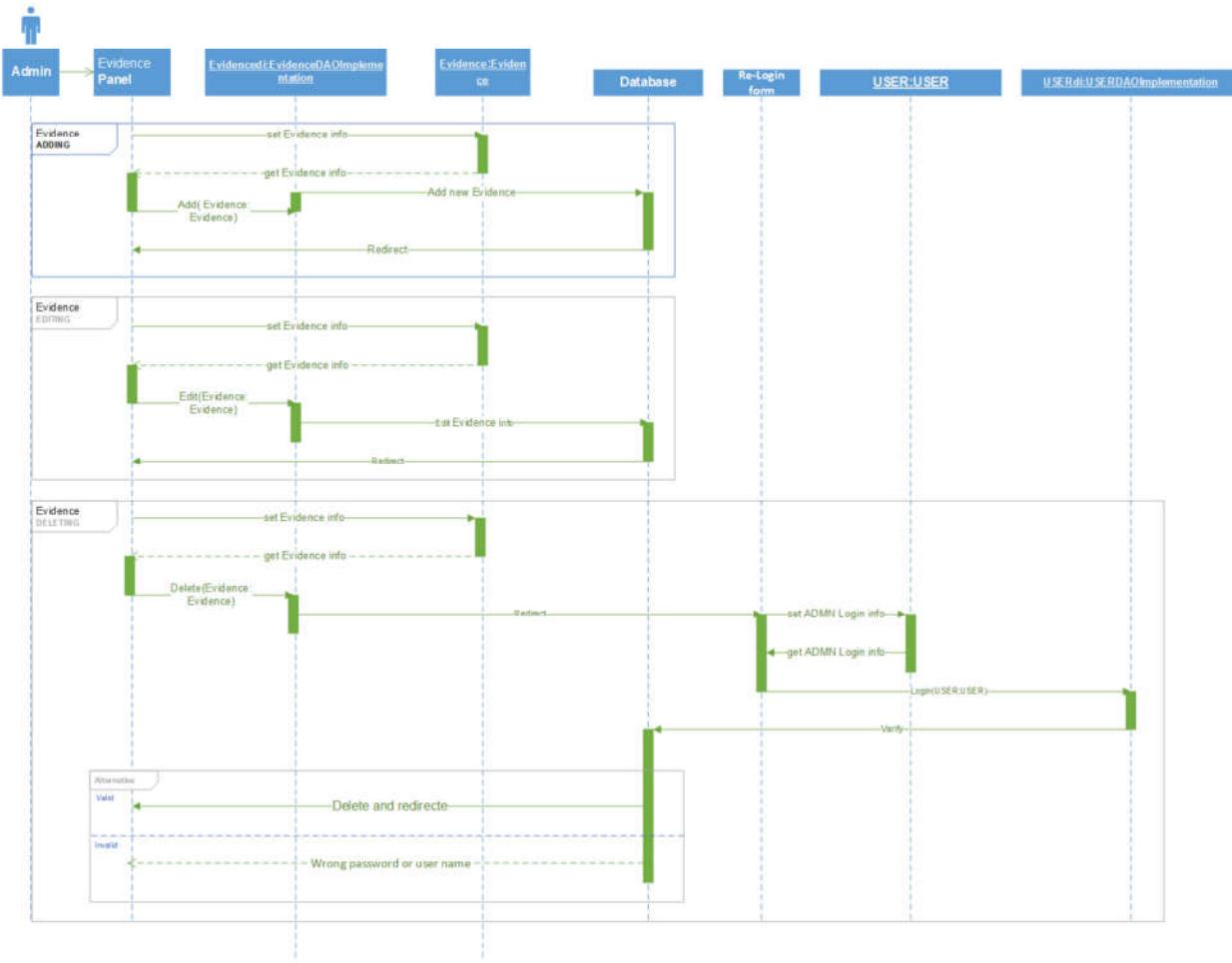


Figure 25: Evidence Management

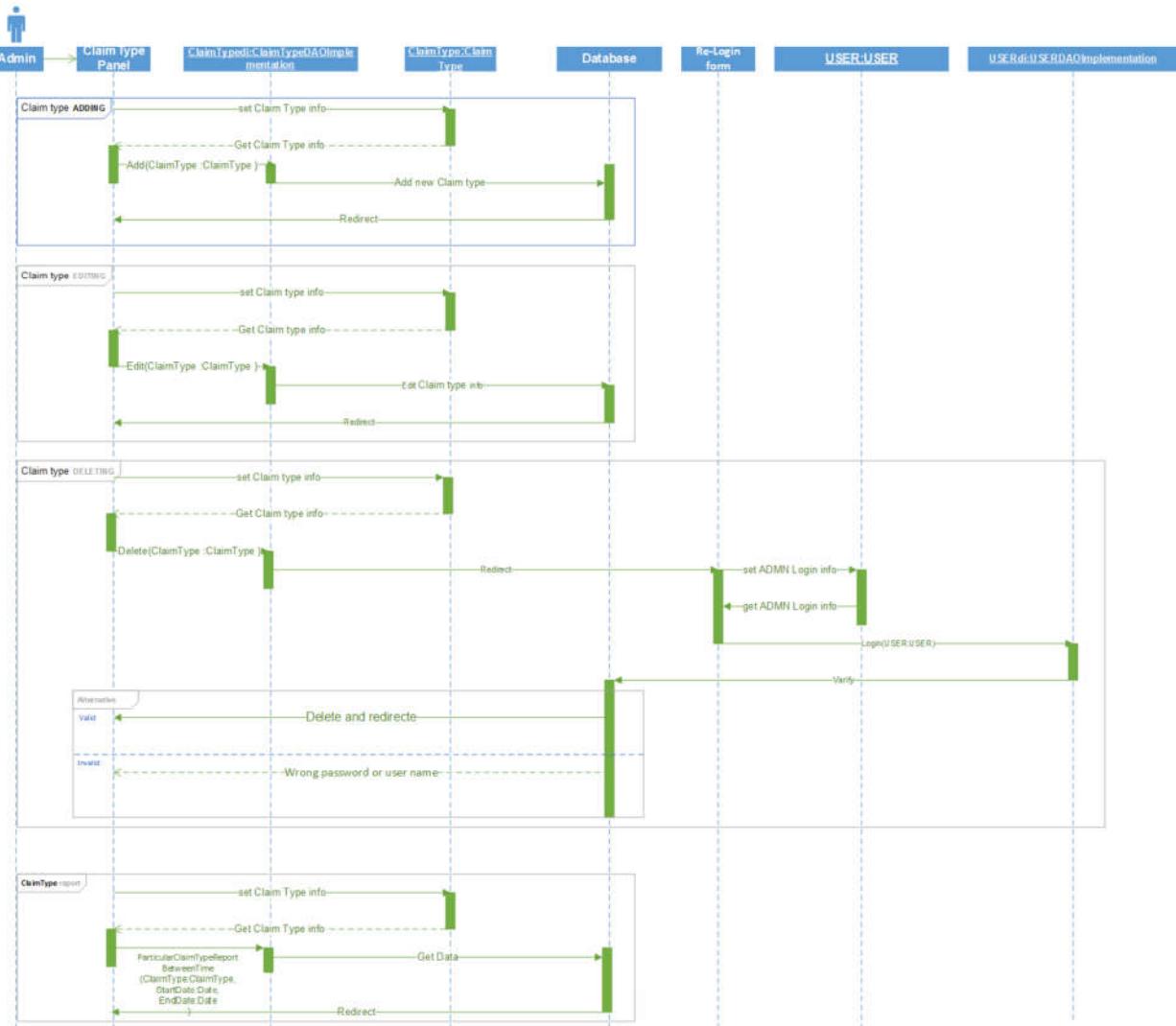


Figure 26: Claim Type Management



Figure 27: Claim Management

Activity diagram

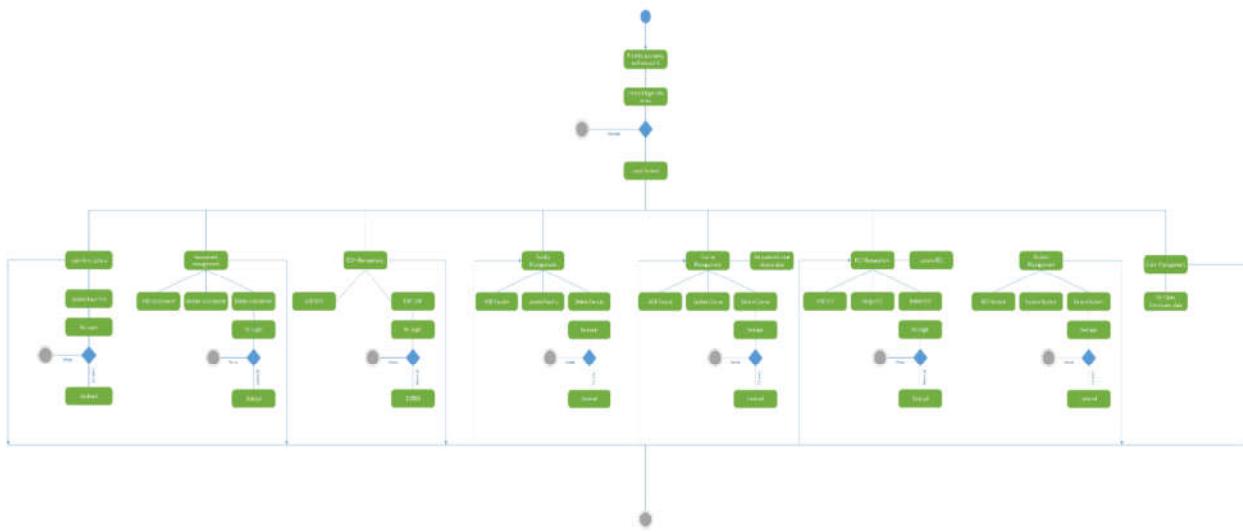


Figure 28: Admin Activity

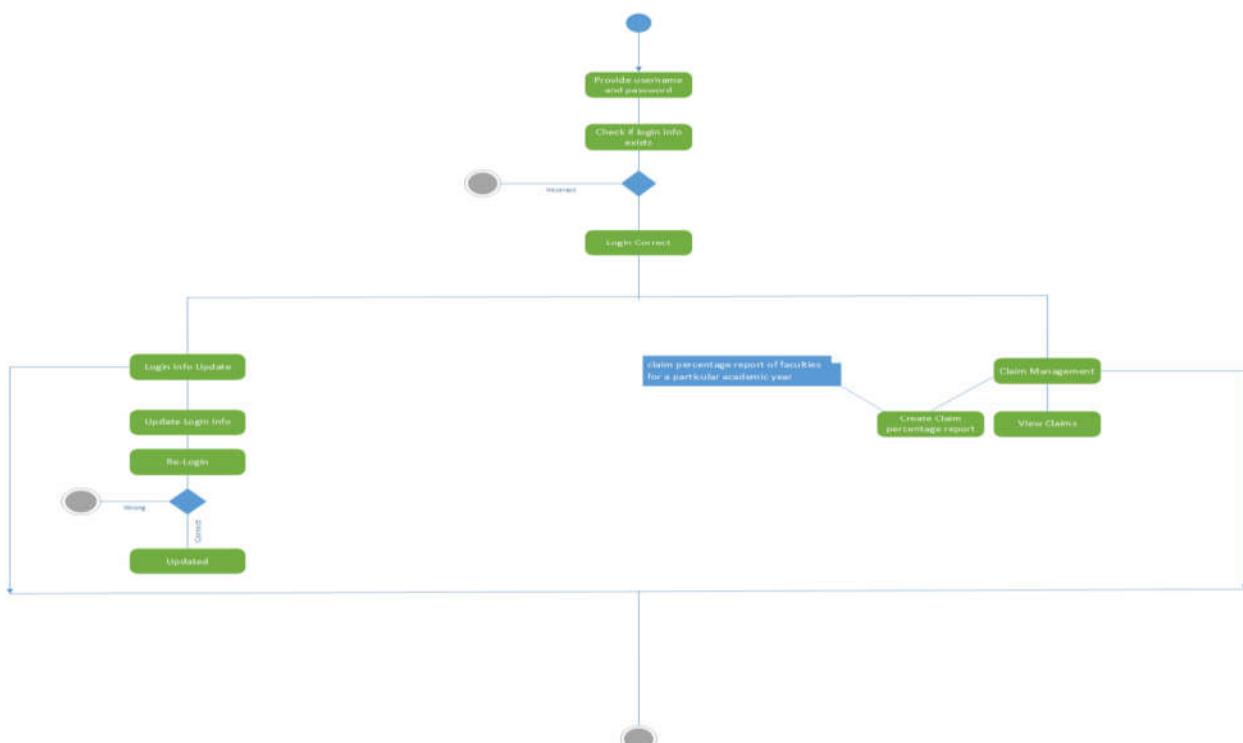


Figure 29: ECM activity

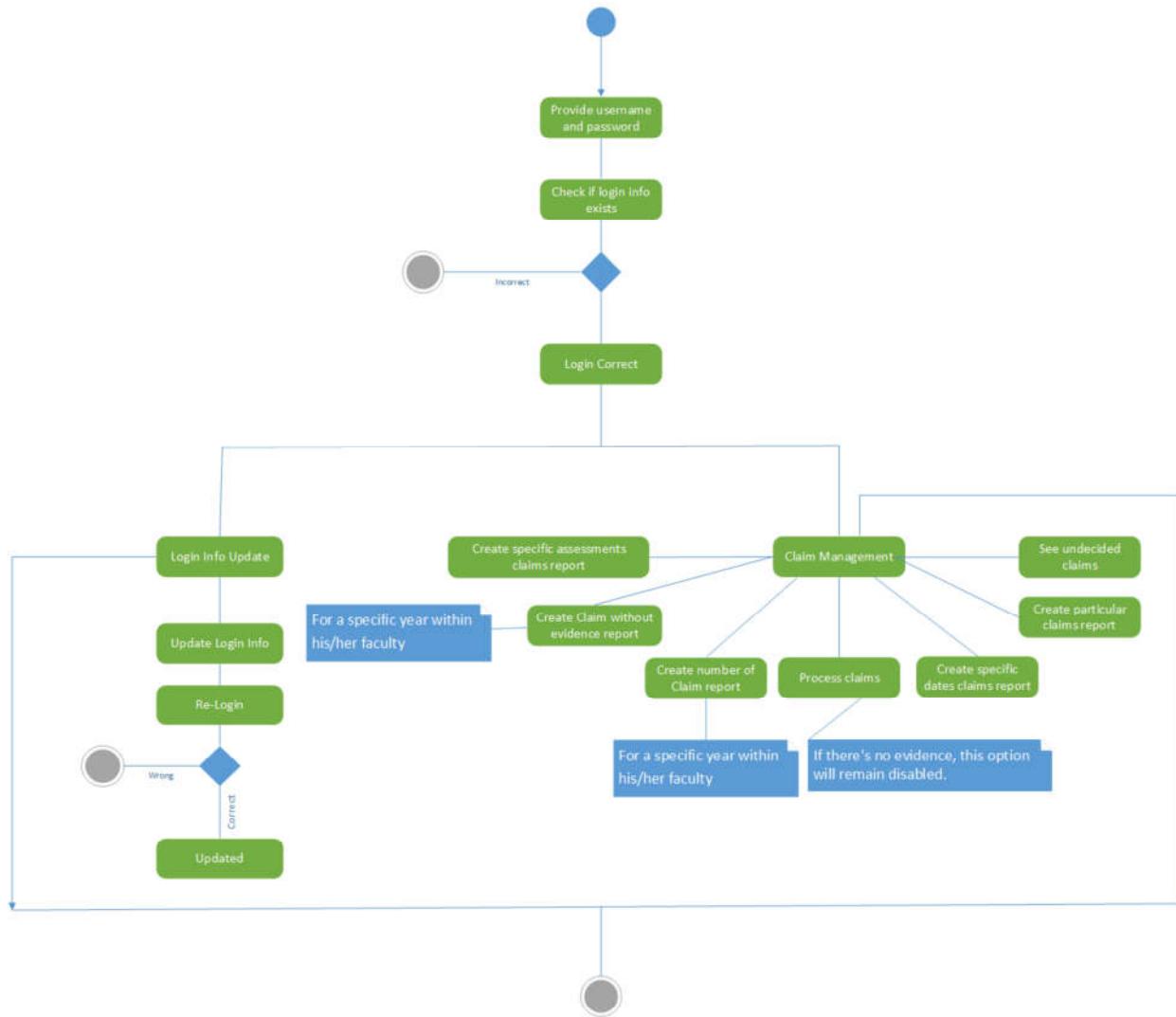


Figure 30: ECC activity

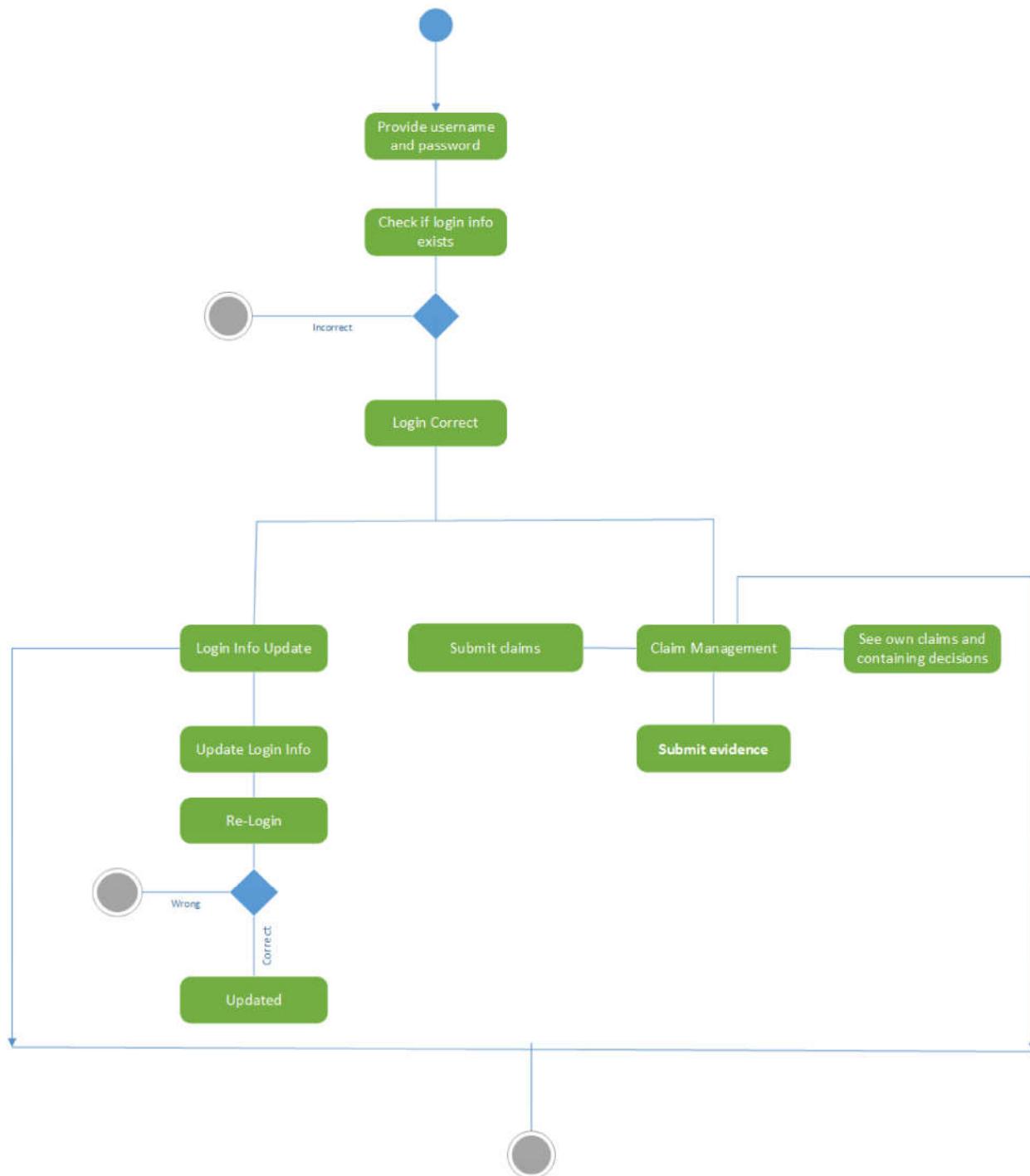


Figure 31: Student Activity

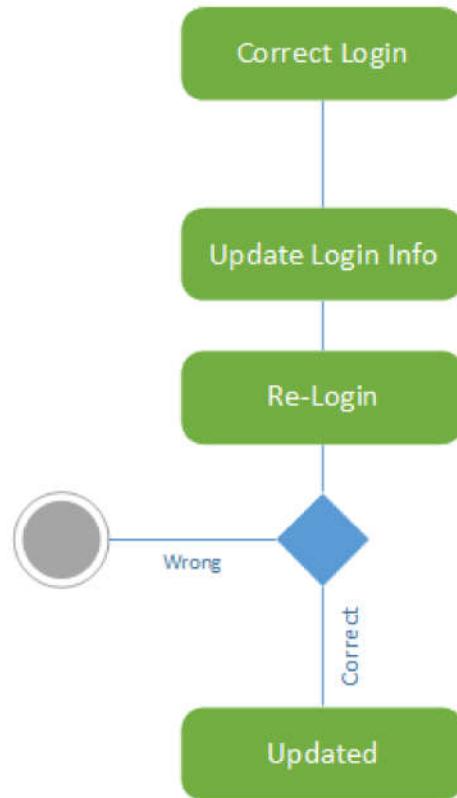


Figure 32: Login Info Update activity

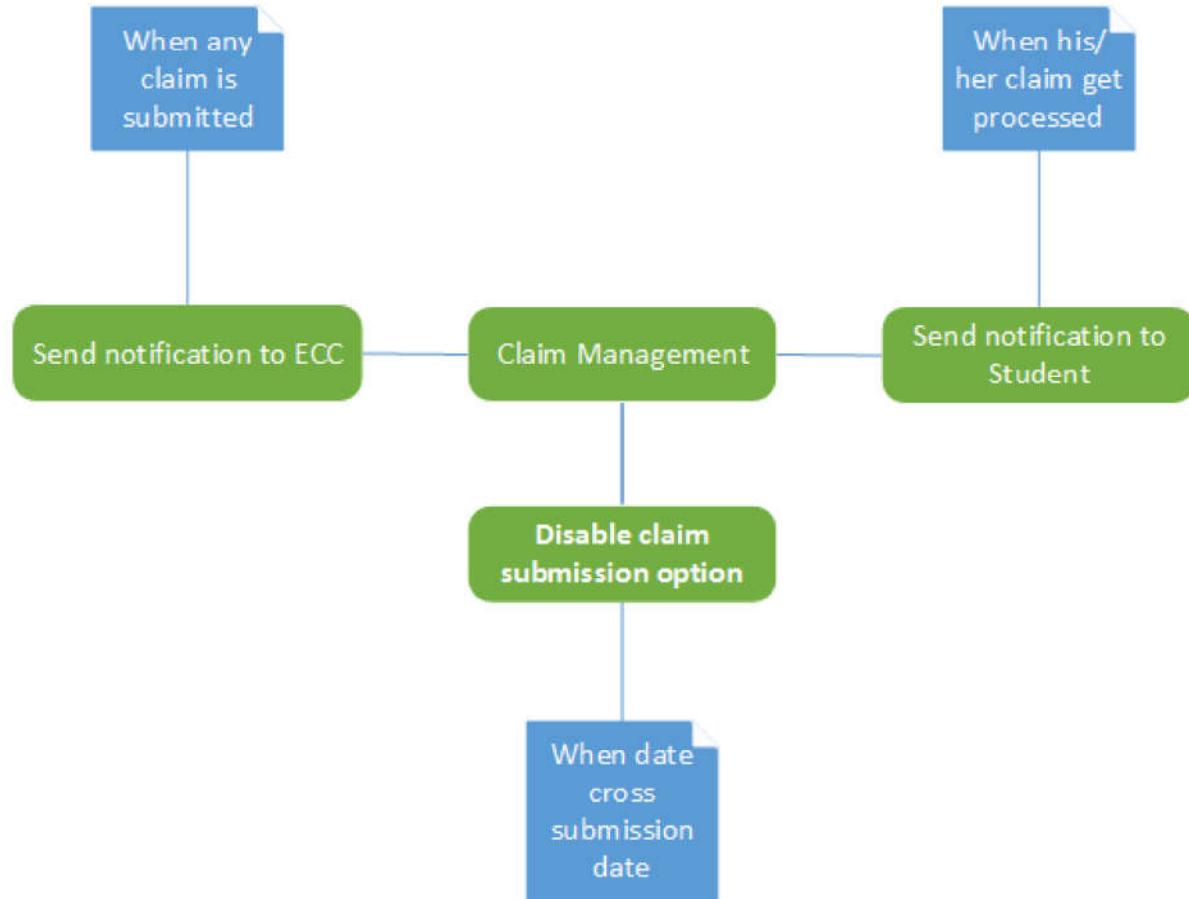


Figure 33: System Activity

Appendix 3: SCRUM artifacts

SCRUM Product Backlog:

ID	Story	Estimation	Priority
FR01	As an user(ECM,ECC, Admin,Student) , I want to login to the system.	16	High
FR02	As an user(ECM,ECC, Admin,Student) , I want to have my individual dashboard.	30	High
FR03	As an Administrator , I want to Manage ECM.	6	High

FR09	As an Administrator I want to Manage ECC.	5	High
FR04	As an Administrator , I want to manage Faculty.	5	High
FR05	As an Administrator , I want to manage course.	7	High
FR06	As an Administrator , I want to Manage subject.	8	High
FR07	As an Administrator I want to manage	8	High

	Assessment item.		
FR11	As an Administrator I want to manage student.	9	High
	As an ECM I want to View all claims.	3	High
FR13	As an ECC I want to access my faculty students claims.	3	High
FR15	As an ECC I want to See report on number of claims in a specific year.	3	High
FR14	As an ECC I want to approve or reject	3	High

	my faculty students claims.		
FR16	As an ECM I want to see report on faculties claim percentage for a year.	4	Medium
FR17	As an ECC I want to see student number for a particular claim in a year within my faculty.	5	Medium
FR18	As an ECC I want to see report on claims containing no evidence within my faculty.	8	Medium
FR19	As an ECC I want to see all undecided	5	Medium

	claims after 14 days within my faculty.		
FR20	As a Student I want to submit multiple claims for an assessment item.	7	Medium
FR21	As a Student I want to see my own claims and decisions.	5	High
FR22	As a student I want to continue submitting evidence for my claim till closure date.	7	High
FR23	As a System I want to disable new claims after due date for claiming.	4	Medium

FR24	As a System I want to send email notification to due ECC when any student submit any claim.	3	High
FR25	As a System I want to send email notification about ECCs' decision on a claim that being processed to due student.	3	High
FR26	As an Administrator I want to create a specific date's claims report.	7	low
FR27	As an Administrator I	6	Medium

	want to create a specific assessment items claims report.		
Total		170	

Sprint backlog

Sprint 1:

S	User Story	Task	Day	Day	Day	Day	Day
			1	2	3	4	5
P	As an user(ECM,ECC, Admin,Student), I want to login to the system.	Design database	6	5	3	0	0
		Design login page	1	0	0	0	0
		Develop login feature	1	0	0	0	0
		Automate test	1	0	0	0	0
R	As an user(ECM,ECC, Admin,Student), I want my individual dashboard.	Write test plan	3	2	0	0	0
		Design dashboard	5	4	4	3	0
		Integrate with login page	1	0	0	0	0
		Test	4	1	0	0	0
I	As an user(ECM,ECC, Admin,Student), I want my individual dashboard.	Write test plan	3	2	0	0	0
		Design dashboard	5	4	4	3	0
		Integrate with login page	1	0	0	0	0
		Test	4	1	0	0	0
N	As an user(ECM,ECC, Admin,Student), I want my individual dashboard.	Write test plan	3	2	0	0	0
		Design dashboard	5	4	4	3	0
		Integrate with login page	1	0	0	0	0
		Test	4	1	0	0	0

T O N E	As an Administrator , I want to manage ECM.						
		Write test plan	2	0	0	0	0
		Develop features	1	0	0	0	0
		Meeting with Shiplu sir	0	0	2	0	0
		Test	2	0	0	0	0
	As an Administrator , I want to manage ECC.						
		Write test plan	2	0	0	0	0
		Develop features	1	0	0	0	0
		Test	2	0	0	0	0

Sprint 2:

	User Story	Task	Day 1	Day 2	Day 3	Day 4	Day 5
S	As an Administrator , I want to manage Faculty.	Write test plan	2	0	0	0	0
		Develop feature	2	0	0	0	0
		Test	1	0	0	0	0
P	As an Administrator , I want to manage course.	Write test plan	2	0	0	0	0
		Develop feature	2	2	0	0	0
		Test	1	0	0	0	0
R	As an Administrator , I	Write test plan	2	0	0	0	0
		Develop features	1	1	0	0	0

N	want to Manage subject.	Meeting with Shiplu sir	0	0	2	0	0
		Test	2	0	0	0	0
	As an Administrator I want to manage assessment item.						
		Write test plan	2	0	0	0	0
		Develop features	1	0	2	0	0
		Test	2	1	0	0	0
T	As an Administrator I want to manage student.	Write test plan	0	2	0	0	0
		Develop features	0	3	1	0	0
		Test	0	0	3	0	0
T	As an ECM I want to View all claims.	Write test plan	0	1	0	0	0
		Develop features	0	1	0	0	0
		Test	0	1	0	0	0

W						
O	As an ECC I want to access my faculty students claims.	Write test plan	0	1	0	0
		Develop features	0	1	0	0
		Test	0	1	0	0
	As an ECC I want to See report on number of claims in a specific year.	Write test plan	0	0	1	0
		Develop features	0	0	1	0
		Test	0	0	1	0
	As an ECC I want to approve or reject my faculty students claims.	Write test plan	0	0	1	0
		Develop features	0	0	1	0
		Test	0	0	1	0

Sprint 3:

	User Story	Task	Day	Day	Day	Day	Day
			1	2	3	4	5
S	As an ECM I want to see report on faculties claim percentage for a year.	Write test plan	1	0	0	0	0
		Develop feature	2	0	0	0	0
		Test	1	0	0	0	0
P	As an ECC I want to see student number for a particular claim in a year within my faculty.	Write test plan	2	0	0	0	0
		Develop feature	2	0	0	0	0
		Test	1	0	0	0	0
R	As an ECC I want to see report on	Write test plan	2	0	0	0	0
		Develop features	1	1	0	0	0
I		Write test plan	2	0	0	0	0
		Develop features	1	1	0	0	0
N		Write test plan	2	0	0	0	0
		Develop features	1	1	0	0	0

T	claims containing no evidence within my faculty.	Meeting with Shiplu sir	0	2	0	0	0
		Test	0	2	0	0	0
As an ECC I want to see all undecided claims After 14 days within my faculty.		Write test plan	0	2	0	0	0
		Develop features	0	2	0	0	0
		Test	0	1	0	0	0
As a Student I want to submit multiple claims for an assessment item.		Write test plan	0	2	0	0	0
		Develop features	0	3	0	0	0
		Test	0	2	0	0	0
As a Student I want to see my		Write test plan	0	1	1	0	0
		Develop features	0	0	2	0	0
		Test	0	0	1	0	0

H R E E	own claims and decisions.						
	As a student I want to continue submitting evidence for my claim till closure date.	Write test plan	0	0	2	0	0
		Develop features	0	0	3	0	0
		Test	0	0	2	0	0
	As a System I want to disable new claims after due date for claiming.	Write test plan	0	0	1	1	0
		Develop features	0	0	0	1	0
		Test	0	0	0	1	0

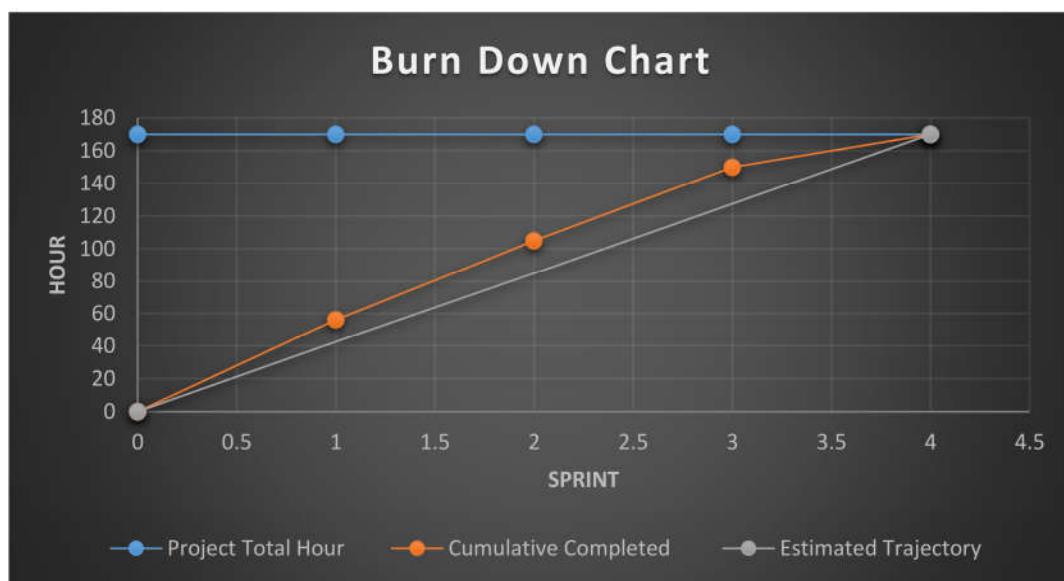
Sprint 4:

S	User Story	Task	Day	Day	Day	Day	Day
			1	2	3	4	5
P	As a System I want to send email notification to due ECC when any student submit any claim.	Write test plan	1	0	0	0	0
		Develop feature	1	0	0	0	0
		Test	1	0	0	0	0
I	As a System I want to send email notification about ECCs' decision on a claim that being processed to due student.	Write test plan	1	0	0	0	0
		Develop feature	1	0	0	0	0
		Test	1	0	0	0	0
N		Write test plan	0	2	0	0	0
T							

F O U R	As an Administrator I want to create a specific date's claims report.	Develop features	0	2	0	0	0
		Meeting with Shiplu sir	0	2	0	0	0
		Test	0	2	0	0	0
	As an Administrator I want to create a specific assessment items claims report.	Write test plan	0	0	2	0	0
		Develop features	0	0	2	0	0
		Test	0	0	2	0	0

Burn Down chart(Less detailed)

Sprint	Project Total Hour	Completed	Cumulative Completed	Estimated Trajectory
0	170	0	0	0
1	170	56	56	#N/A
2	170	49	105	#N/A
3	170	45	150	#N/A
4	170	20	170	170



Detailed burn down chart

Day	Project Total Hour	Completed	Cumulative Completed	Estimated Trajectory
Day0	170	0	0	0
Day1	170	32	32	#N/A
Day2	170	12	44	#N/A
Day3	170	9	53	#N/A
Day4	170	3	56	#N/A
Day5	170	0	56	#N/A
Day1	170	20	76	#N/A
Day2	170	15	91	#N/A
Day3	170	14	105	#N/A
Day4	170	0	105	#N/A
Day5	170	0	105	#N/A
Day1	170	12	117	#N/A
Day2	170	18	135	#N/A
Day3	170	12	147	#N/A
Day4	170	3	150	#N/A
Day5	170	0	150	#N/A
Day1	170	6	156	#N/A
Day2	170	8	164	#N/A
Day3	170	6	170	#N/A
Day4	170	0	170	#N/A
Day5	170	0	170	170

