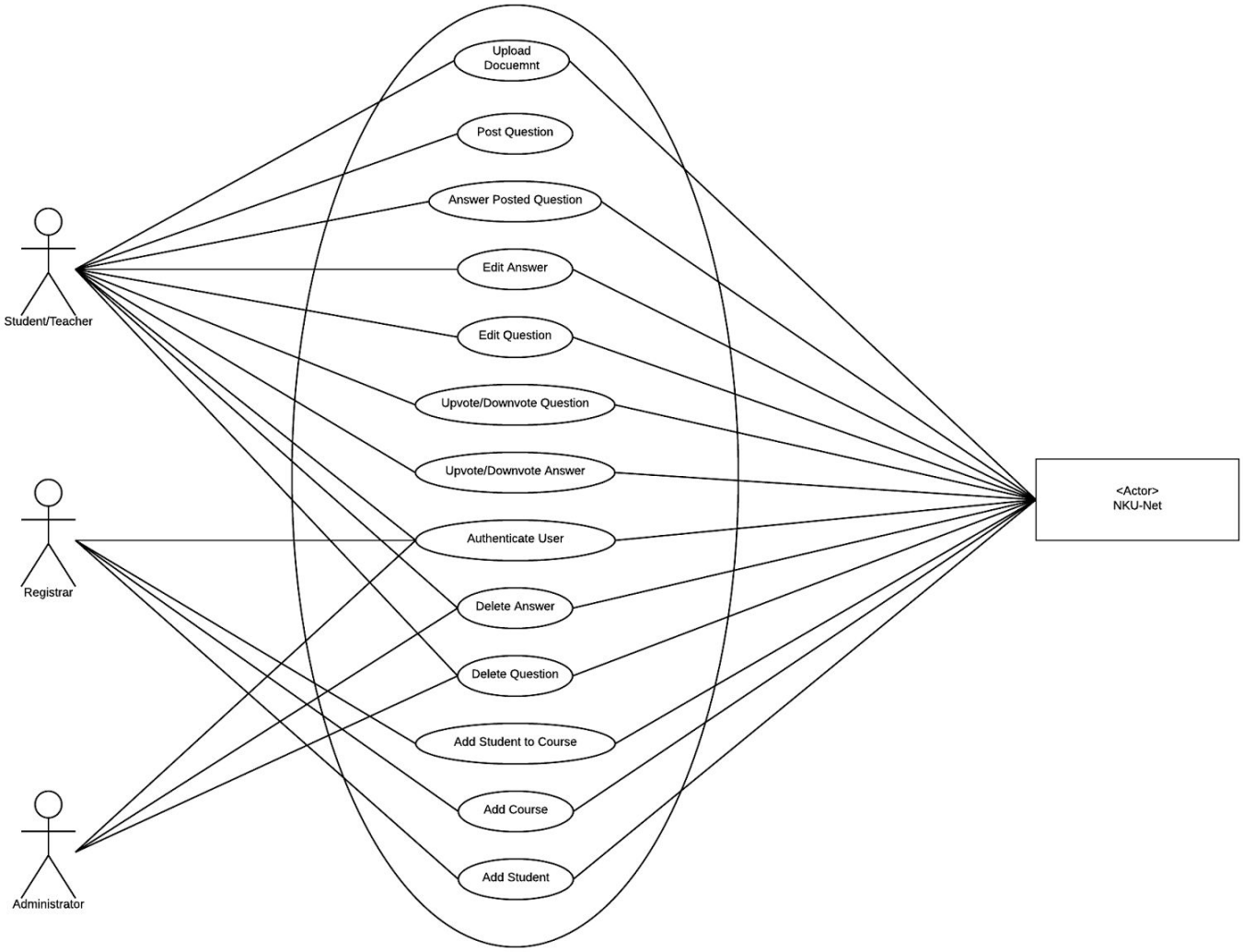
Hudson, Nathaniel • Jansen, Shane • Owens, Kirby • Zumwalde, Zach
Iteration 2 - Last Modified 4/16/17

Requirements Document

USE CASE DIAGRAM:



USE CASES

Use Case (UC1): Post Question.

Scope: Post Question System
Level: user-goal
Primary Actor: Student, Faculty Member
Stakeholders and Interests:
- Student: Wants to be able to post question on NKU-Net.
- Faculty Member: Wants to be able to post question on NKU-Net.
- Administrator: Wants to be able to post question on NKU-Net.
- School: Wants students/faculty to be able to post questions as quickly as possible.

Preconditions: Student or faculty member is logged into system upon being identified and authenticated.
Success guarantee (or Postconditions): Student or faculty member posts question onto NKU-Net.

Main Success Scenario (or Basic Flow):
1. Student/faculty member navigates to course forum.
2. Student/faculty member types out question title.
3. Student/faculty member types out question body.
4. Student/faculty member submits question to NKU-Net.

Student/faculty member repeats steps 1-4 until done posting questions.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, student/faculty member is logged off and connection is shut off.

Special Requirements:
- System posts question within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Student/faculty member can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).
Frequency of Occurrence: Could be nearly continuous.

Use Case (UC2): Answer Question.

Scope: Submit Answer System
Level: user-goal
Primary Actor: Student, Faculty Member
Stakeholders and Interests:
- Student: Wants to be able to post answer to question on NKU-Net.
- Faculty Member: Wants to be able to post answer to question on NKU-Net.
- Administrator: Wants to able to post answer to question on NKU-Net.
- School: Wants students/faculty members to post answers as quickly/easily as possible.

Preconditions: Student or faculty member is logged into system upon being identified and authenticated.
Success guarantee (or Postconditions): Student or faculty member posts answer to question on NKU-Net.

Main Success Scenario (or Basic Flow):
1. Student/faculty member selects course forum.
2. Student/faculty member selects question to answer.
3. Student/faculty member types out their answer to question.
4. Student/faculty member submits answer to question to NKU-Net.

Student/faculty member repeats steps 1-4 until done posting questions.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, student/faculty member is logged off and connection is shut off.

Special Requirements:
- System posts answer to question within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Student/faculty member can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC3): Edit Answer.

Scope: Edit Answer System
Level: user-goal
Primary Actor: Administrator
Stakeholders and Interests:
- Administrator: Wants to be able to edit answer on NKU-Net.
- School: Wants answers to be correct and helpful to students.

Preconditions: Administrator is logged into system upon being identified and authenticated with admin permissions.
Success guarantee (or Postconditions): Administrator's updated edit of the answer is uploaded to NKU-Net.

Main Success Scenario (or Basic Flow):
1. Administrator selects a class forum.
2. Administrator selects a question forum.
3. Administrator selects an answer that he/she wants to edit.
4. Administrator edits the answer to his/her liking.
5. Administrator saves the edit.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, Administrator is logged off and connection is shut off.

Special Requirements:
- System posts edited answer within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Administrator can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC4): Delete Answer.

Scope: Delete Answer System
Level: user-goal
Primary Actor: Administrator
Stakeholders and Interests:
- Administrator: Wants to be able to delete answer on NKU-Net.
- School: Wants inappropriate answers to be removed quickly.

Preconditions: Administrator is logged into system upon being identified and authenticated with admin permissions.
Success guarantee (or Postconditions): Administrator's chosen answer is deleted from NKU-Net.

Main Success Scenario (or Basic Flow):
1. Administrator selects a class forum.
2. Administrator selects a question forum.
3. Administrator selects an answer that he/she wants to edit
4. Administrator chooses to delete the answer.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, Administrator is logged off and connection is shut off.

Special Requirements:
- System deletes answer within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Administrator can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC5): Delete Question.

Scope: Delete Question System
Level: user-goal
Primary Actor: Administrator
Stakeholders and Interests:
- Administrator: Wants to be able to delete question on NKU-Net.
- School: Wants inappropriate questions to be deleted promptly.

Preconditions: Administrator is logged into system upon being identified and authenticated with admin permissions.
Success guarantee (or Postconditions): Administrator's chosen question is deleted from NKU-Net.

Main Success Scenario (or Basic Flow):
1. Administrator selects a class forum.
2. Administrator selects a question forum.
3. Administrator selects a question that he/she wants to edit
4. Administrator chooses to delete the question.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, Administrator is logged off and connection is shut off.

Special Requirements:
- System deletes question within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Administrator can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC6): Edit Question.

Scope: Edit Question System
Level: user-goal
Primary Actor: Administrator
Stakeholders and Interests:
- Administrator: Wants to be able to edit question on NKU-Net.
- School: Wants questions to be appropriate and helpful to students.

Preconditions: Administrator is logged into system upon being identified and authenticated with admin permissions.
Success guarantee (or Postconditions): Administrator's updated edit of the question is uploaded to NKU-Net.

Main Success Scenario (or Basic Flow):
1. Administrator selects a class forum.
2. Administrator selects a question that he/she wants to edit.
3. Administrator edits question to his/her liking.
4. Administrator saves question.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, Administrator is logged off and connection is shut off.

Special Requirements:
- System posts edited question within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Administrator can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC7): Upvote/Downvote Question.

Scope: Upvote/Downvote Question System
Level: user-goal
Primary Actor: Student, Faculty Member
Stakeholders and Interests:
- Student: Wants to receive credit/incentive for question.
- Faculty: Wants to encourage students into posting good questions.
- Administrator: Wants good posting techniques to be encouraged.
- School: Wants good posting techniquest to be encouraged.

Preconditions: Student/Faculty Member is logged into system upon being identified and authenticated.
Success guarantee (or Postconditions): Question's upvote/downvote information is updated in NKU-Net.

Main Success Scenario (or Basic Flow):
1. Student/Faculty Member selects a class forum.
2. Student/Faculty Member selects a question in forum.
3. Student/Faculty Member upvotes or downvotes a question.
4. System updates the vote count for the question.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, Administrator is logged off and connection is shut off.

Special Requirements:
- System upvotes/downvotes question within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Administrator can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC8): Upvote/Downvote Answer.

Scope: Upvote/Downvote Answer System
Level: user-goal
Primary Actor: Student/Faculty Member
Stakeholders and Interests:
- Student: Wants to be able to receive credit/incentive for answer.
- Faculty Member: Wants to encourage students into posting good answers.
- Administrator: Wants to encourage students into posting good answers.
- School: Wants answers that are the most helpful to be seen first.

Preconditions: Student/Faculty Member is logged into system upon being identified and authenticated.
Success guarantee (or Postconditions): Answer's upvote/downvote information is updated on NKU-Net.

Main Success Scenario (or Basic Flow):
1. Student/Faculty Member selects a class forum.
2. Student/Faculty Member selects a question forum.
3. Student/Faculty Member upvotes/downvotes the answer of their liking.
4. System updates the vote count for the answer.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, Administrator is logged off and connection is shut off.

Special Requirements:
- System updates votes within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Administrator can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC9): Authenticate User.

Scope: Authenticate System
Level: user-goal
Primary Actor: User
Stakeholders and Interests:
- Students: Wants to login as fast as possible.
- Faculty: Wants to login as fast as possible.
- Administrator: Wants login to be secure.
- School: Wants students to not be able to modify each other's data.

Preconditions: User already has an account created.
Success guarantee (or Postconditions): Administrator's updated edit of the answer is uploaded to NKU-Net.

Main Success Scenario (or Basic Flow):
1. User types in email address and password into login page.
2. System authenticates user.
3. System notifies user of successful login and redirects them to course page.

Extensions (or Alternative Flows):
*a. At any time, System fails:
    1. To ensure security of system, Administrator is logged off and connection is shut off.
2a. Authentication Fails
    1. System notifies user of authentication failure.

Special Requirements:
- System posts edited answer within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Administrator can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC10): Add Student.

Scope: Student Creation System
Level: registrar-goal
Primary Actor: Registrar
Stakeholders and Interests:
- Students: Wants to be able to access NKU-Net.
- Faculty: Wants proper students to be added to NKU-Net.
- Registrar: Wants to be able to add students as fast as possible.
- School: Wants student to be added quickly with little error.

Preconditions: User is already enrolled in school
Success guarantee (or Postconditions): Student is added and can access NKU-Net.
Main Success Scenario (or Basic Flow):
1. Registrar types in student's name and password.
2. System creates student's account.
3. System notifies success in creating user.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, Registrar is logged off and connection is shut off.

2a. Authentication Fails
1. System notifies Registrar of authentication failure.

Special Requirements:
- System adds student within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Registrar can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC11): Add Course.

Scope: Course Creation System
Level: registrar-goal
Primary Actor: Registrar
Stakeholders and Interests:
- Students: Wants to be able to access their course.
- Faculty: Wants their course to be added quickly and accurately.
- School: Wants students/faculty to be able to access their course.

Preconditions: Registrar is signed in and authenticated.
Success guarantee (or Postconditions): The course is added to the database and the course page is accessible.

Main Success Scenario (or Basic Flow):
1. Registrar types in course information.
2. System creates the course.
3. System notifies the Registrar of successful course creation.

Extensions (or Alternative Flows):
*a. At any time, System fails:
    1. To ensure security of system, Registrar is logged off and connection is shut off.
2a. Authentication Fails:
    1. System notifies Registrar of authentication failure.

Special Requirements:
- System creates course within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Registrar can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC12): Student/Faculty Uploads Document.

Scope: Document Upload System
Level: user-goal
Primary Actor: Student/Faculty
Stakeholders and Interests:
- Student/Faculty: Wants to be able to upload a document to NKU-Net.
- School: Wants documents to be shared among students for collaboration.

Preconditions: Student/Faculty is logged into system upon being identified and authenticated.
Success guarantee (or Postconditions): Student/Faculty's document is uploaded to NKU-Net.

Main Success Scenario (or Basic Flow):
1. Student/Faculty views their profile..
2. Student/Faculty click the upload button.
3. Student/Faculty chooses file(s) from their local machine.
4. Document(s) are uploaded to NKU-Net.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, Student/Faculty is logged off and connection is shut off.

Special Requirements:
- System uploads document within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Student/Faculty can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Use Case (UC13): Add Student To Course.

Scope: Student/Course Management System
Level: registrar-goal
Primary Actor: Registrar
Stakeholders and Interests:
- Students: Wants to be able to access their course.
- Faculty: Wants the proper students to be added to their course.
- Registrar: Wants to add students to courses as fast as possible.
- School: Wants students to be able to access their course.

Preconditions: Registrar is signed in and authenticated.
Success guarantee (or Postconditions): The student is added to the course and the student can access it.

Main Success Scenario (or Basic Flow):
1. Registrar selects student and course the student needs to be added to.
2. System adds student to course.
3. System informs registrar of successful addition.

Extensions (or Alternative Flows):
*a. At any time, System fails:
1. To ensure security of system, Registrar is logged off and connection is shut off.

2a. Authentication Fails
1. System notifies Registrar of authentication failure.

Special Requirements:
- System adds student to course within 30 seconds, 90% of the time.
- Compatible with all major Internet browsers.

Technology and Data Variations List:
*a. Registrar can use any major web browser (Firefox, Google Chrome, Safari, Microsoft Edge, etc.).

Frequency of Occurrence: Could be nearly continuous.

Brief Use Cases:

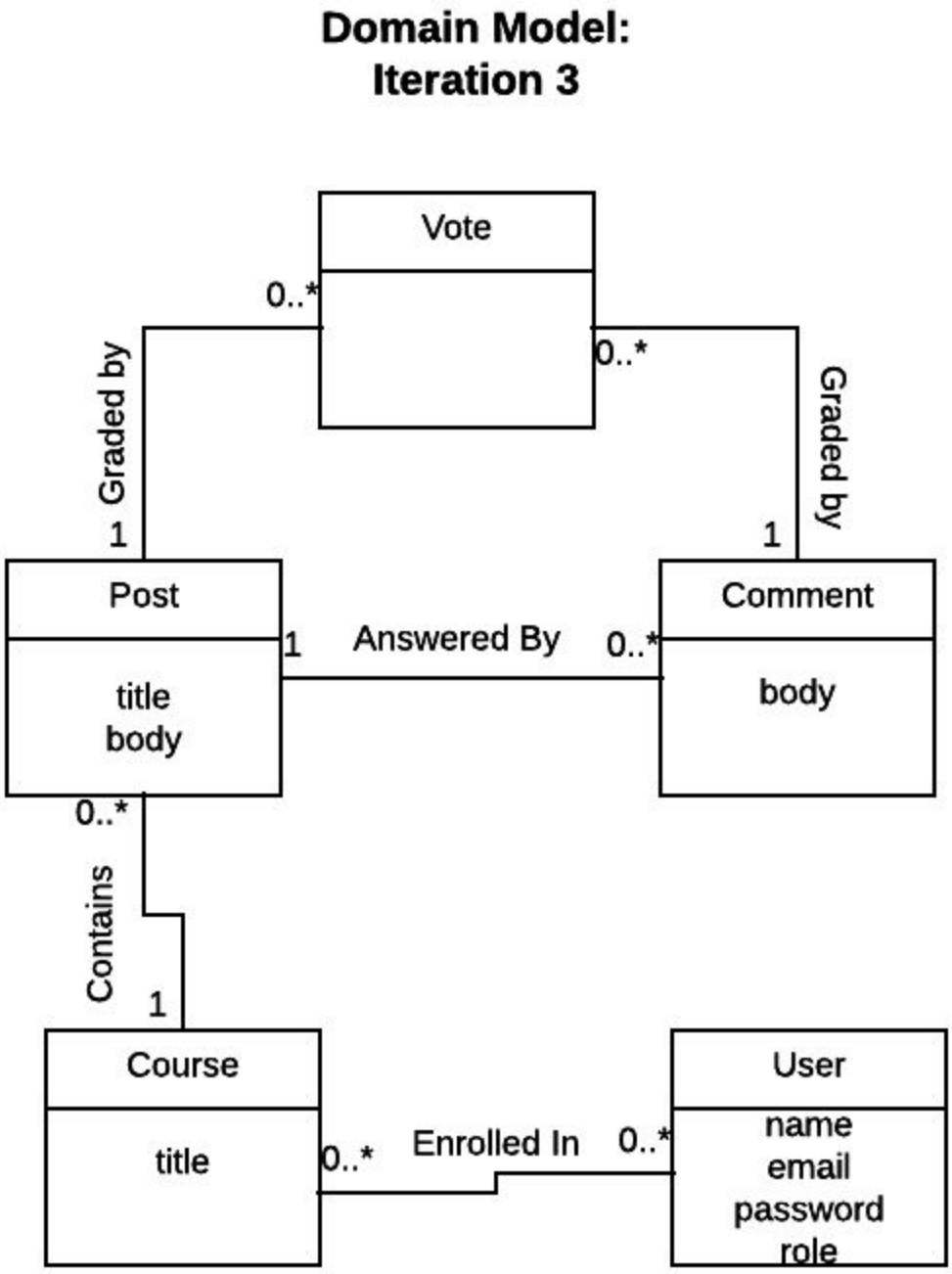
Upvote/Downvote Answer to Question: User views their course forum that has a list of questions. User selects a question to view and views question's answers. The user selects an answer and upvotes or downvotes it. The system increments/decrements the votes for the answer and updates the database. The system displays to the user the updated page.

Add Student to NKU-Net: Registrar/Administrator arrives on the update/create user page and fills out the required fields for the new student. The Registrar/Administrator submits the input and the system processes and validates the user input. If the information is unique then a new student is created by the system. If it is not unique then the system asks the Registrar/Administrator if they would like to update the student's information. The Registrar/Administrator says yes and the system updates the database. The system notifies the Registrar/Administrator of the successful update or creation of the student.

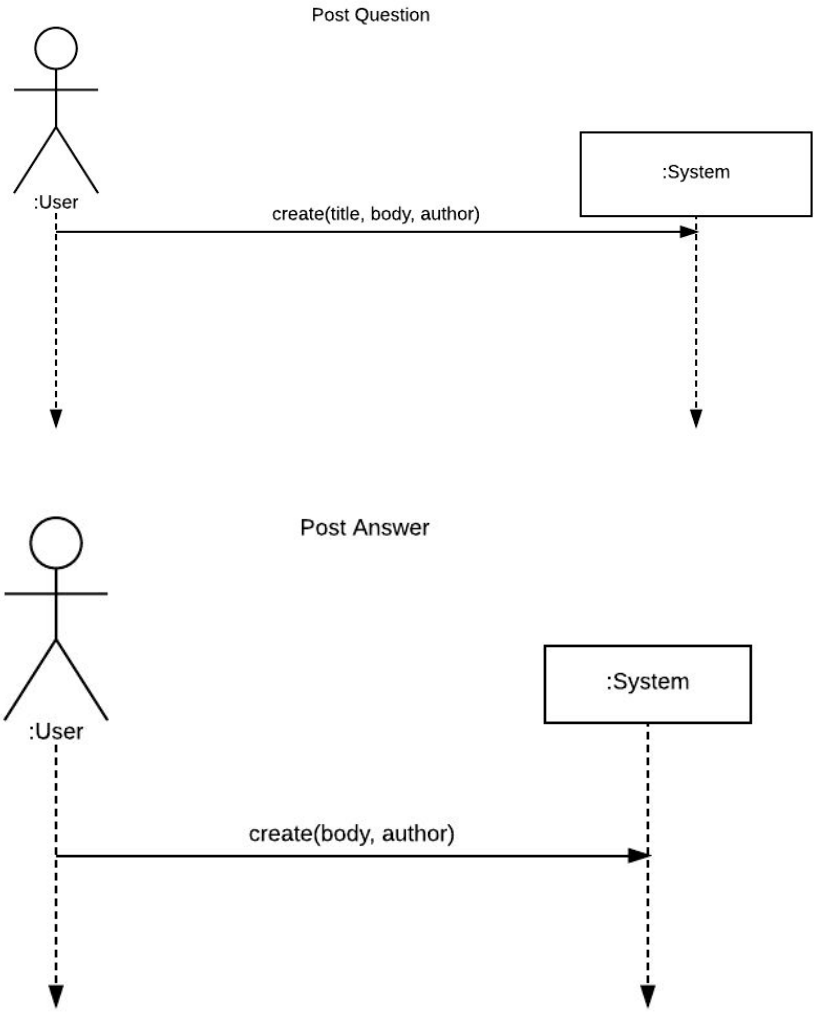
Add Faculty Member to NKU-Net: Registrar/Administrator arrives on the update user page and fills out the required fields for the new faculty member. The Registrar/Administrator submits the input and the system processes and validates the user input. If the information is unique then a new faculty member is created by the system. If it is not unique then the system asks the Registrar/Administrator if they would like to update the faculty member's information. The Registrar/Administrator says yes and the system updates the database. The system notifies the Registrar/Administrator of the successful update or creation of the faculty member.

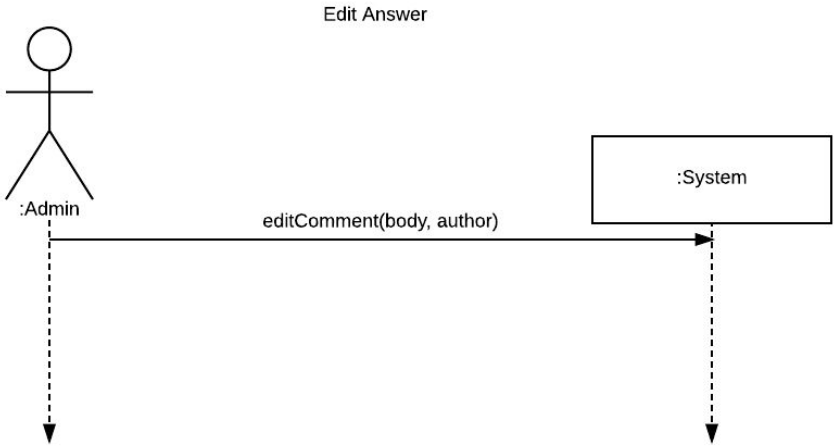
Update Profile Page: The user arrives on their profile page and edits fields that they would like to change. The user saves the changes and the system validates and processes user input. The changes are made in the database and the system displays to the user the updated page.

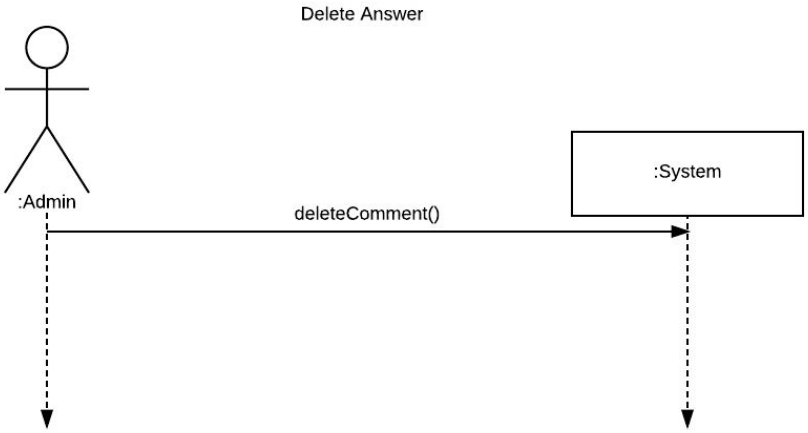
DOMAIN MODEL:

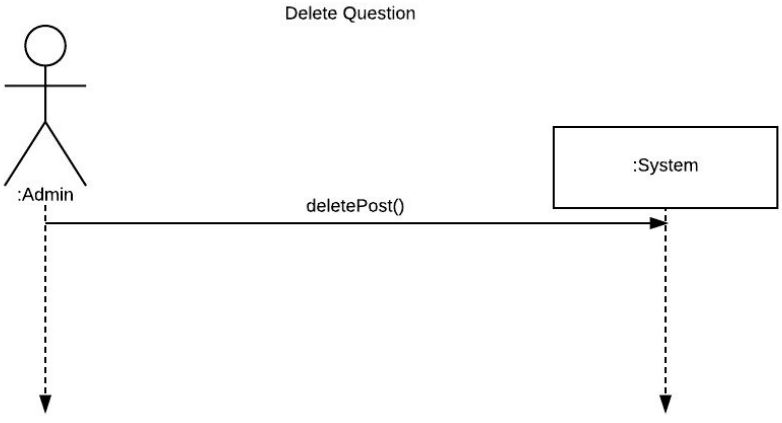


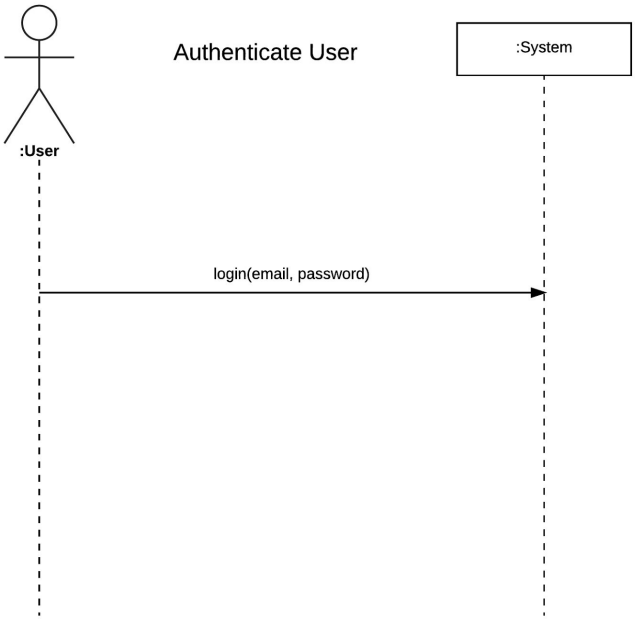
SYSTEM SEQUENCE DIAGRAMS:

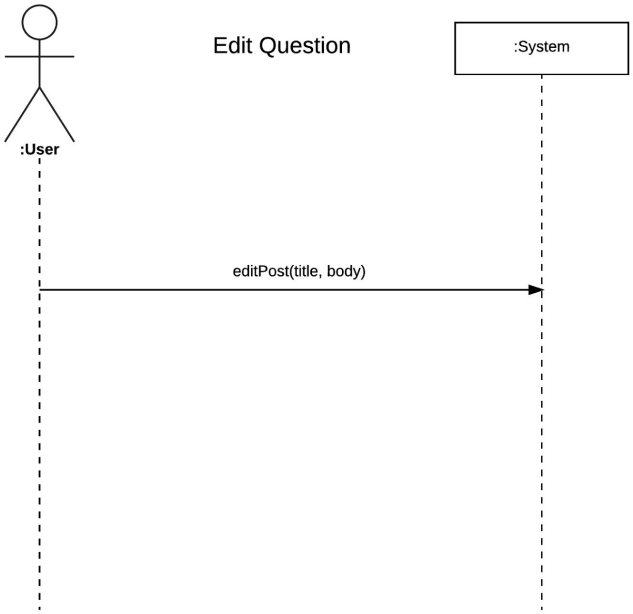


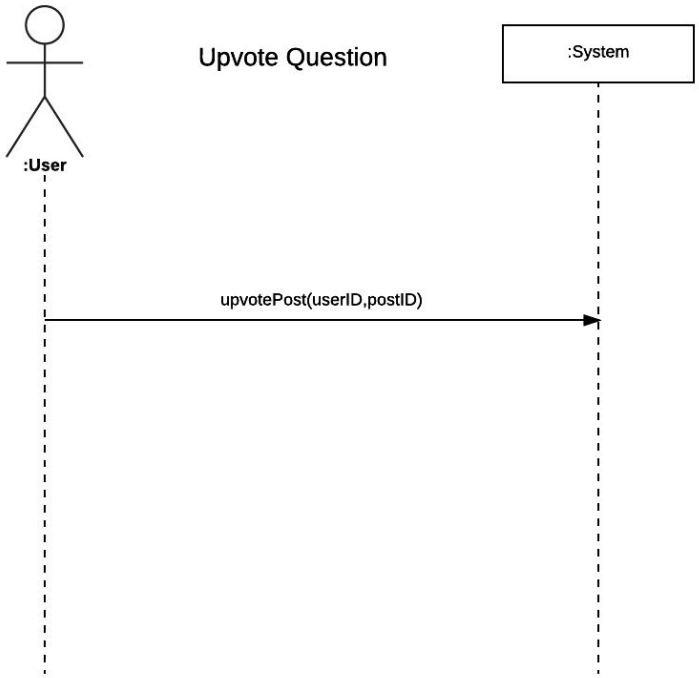

Edit Answer
:Admin
:System
editComment(body, author)

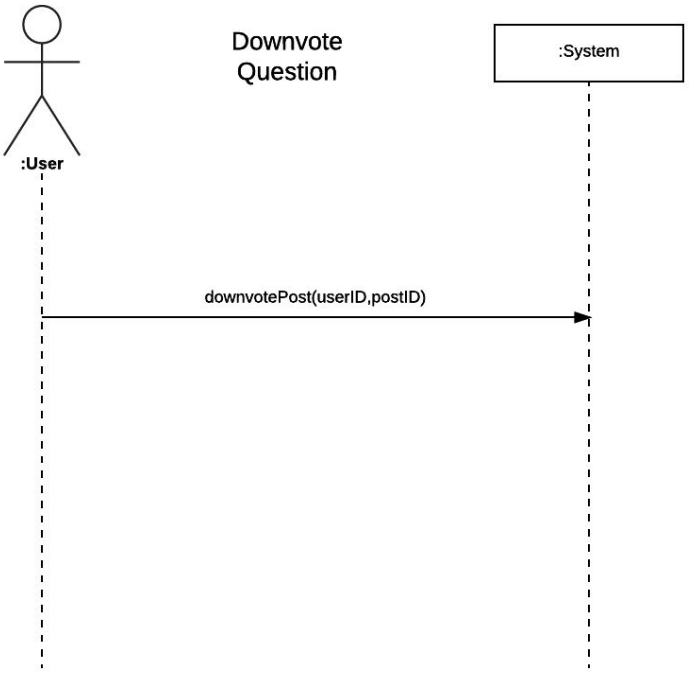

Delete Answer
:Admin
:System
deleteComment()

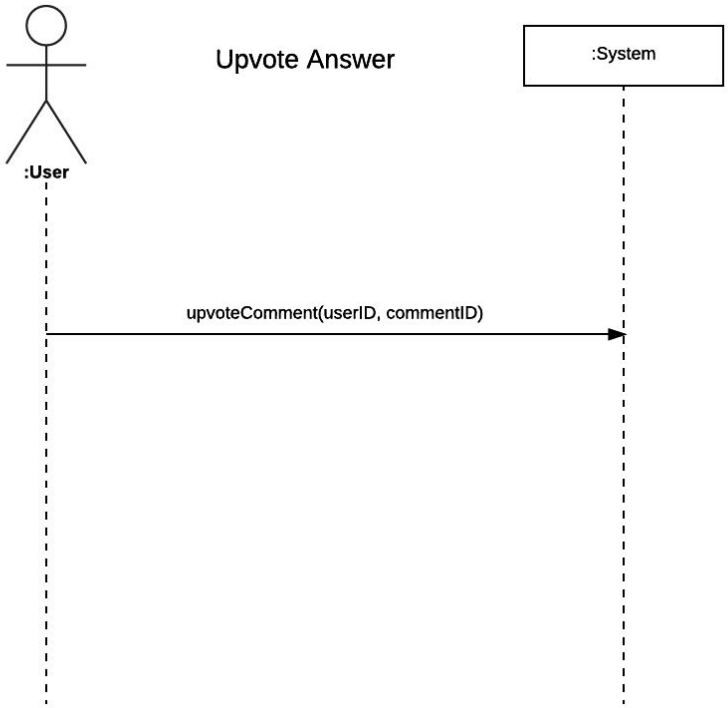

Delete Question
:Admin
:System
deletePost()

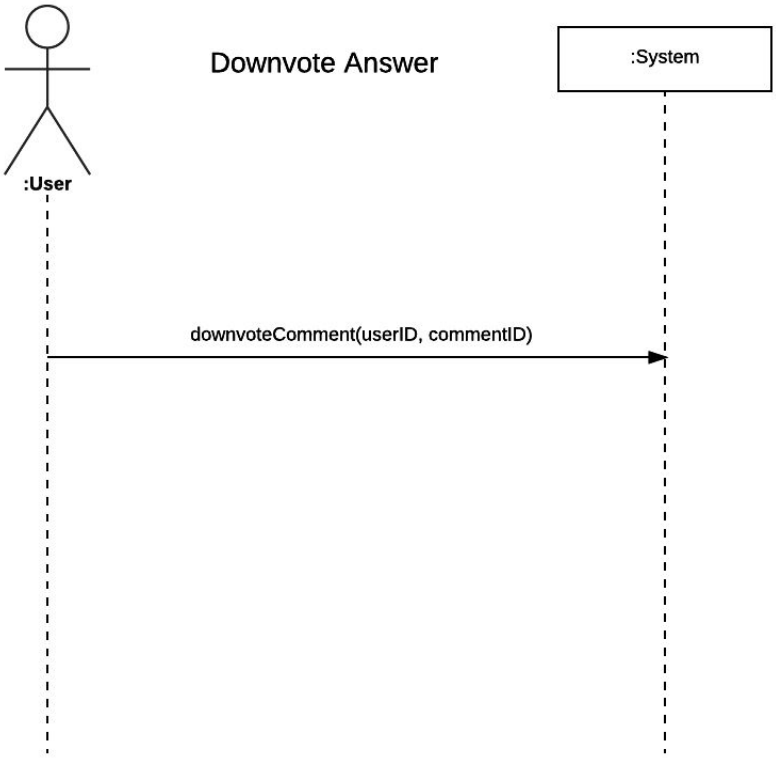

Authenticate User
:System
:User
login(email, password)

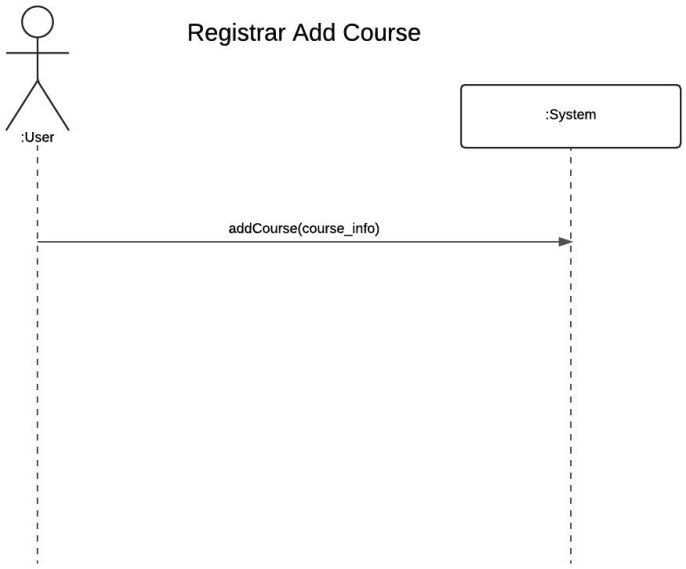

Edit Question
:System
:User
editPost(title, body)

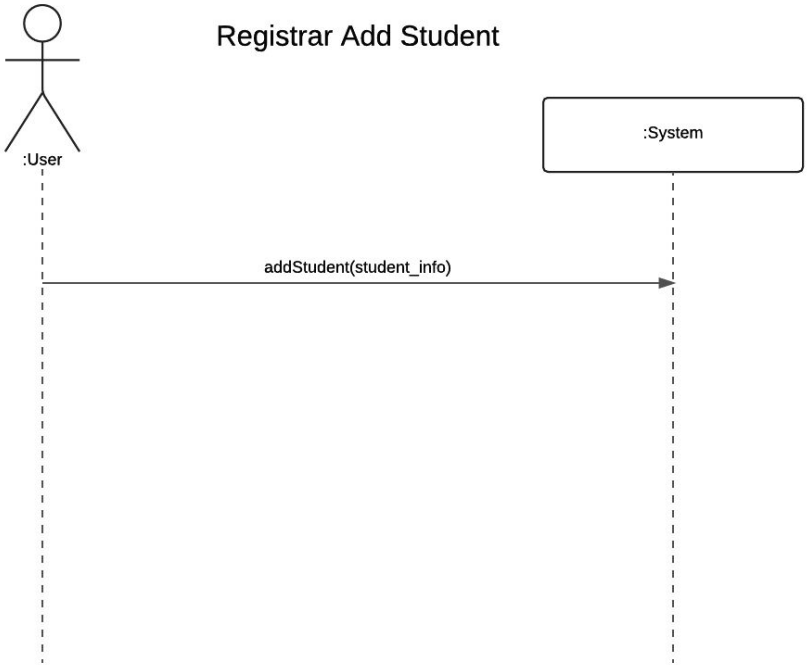

Upvote Question
:System
:User
upvotePost(userID,postID)

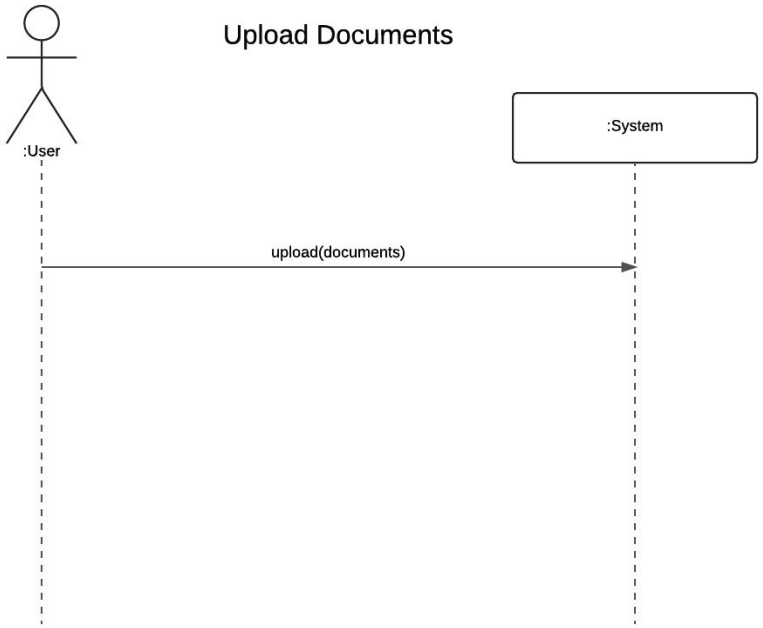

Downvote
Question
:System
:User
downvotePost(userID,postID)


Upvote Answer
:System
:User
upvoteComment(userID, commentID)


Downvote Answer
:System
:User
downvoteComment(userID, commentID)


Registrar Add Course
:User
:System
addCourse(course_info)


Registrar Add Student
:User
:System
addStudent(student_info)


Upload Documents
:User
:System
upload(documents)