CSE 471 PROJECT

Title of the Project: e-Tuition

Group Number: Team Square

Group Members:

Student ID	Student Name
18101184	Zarin Tahia Hossain
18101617	Aninda Sadman
18101443	Taslim Ahmed Taseeb

Introduction

It is a budget friendly online platform for finding private tutor. In this system both the teacher and the student have to buy a subscription for 1 year. The student can look for their suitable tutor in the system and they can send request to the teacher. If the teacher accepts the request of the teacher then they can communicate and start their class through message, video calling and screen sharing. At the end of the month the tutor receives his fees and the student has to pay for the tuition.

Motivation

The main idea of the system came from the daily struggles of many students and it will bring great conveniences to its user. Also, it has a great business value. This system is mainly targeted for the students of any background and also for the young teachers. During difficult time like covid-19 students cannot visit their private tutor, this system can be a great solution for that and also the tutor does not have to stay jobless during this pandemic.

System Description

Project Sponsor: Zarin Tahia - entrepreneur.

Business Need

This system is helpful for both the students and the people who is looking for teaching as a job. As it is fully an online platform so it will save valuable time for both the students and the teachers. Those who has difficulties to find their preferable private tutor can easily find them. Also, it is a very good income source for the young teachers. Furthermore, it reduces hassle.

Business Requirements

Students and teachers both are user here. This is a website where students from kindergarten to class 12 can get their private tutors. Also, services for the students who are going to sit for university admission tests are available. Moreover, who are taking preparation for the BCS and GRE test can also find private tutor for them.

1. Both teacher and student have to open an account in the website.
2. Both teacher and student have individual profile where they provide their information.
3. Students can choose teacher by looking the profile of the teachers and find their tutor according to their needs.
4. Then the student needs to request the teacher. Student can also send messages to that particular teacher so that they can discuss what they need like their preferable time for the class.
5. Teacher will be categorized like Student Tutor, Casual Tutor, Professional Teacher, Senior Teacher etc.
6. The credit for a subject for a level is fixed which will be mentioned in the tutor's profile.
7. After adding each other the teacher can take class through a video call and screen sharing.
8. The transfer of money can be done through online banking as well as Bkash and Rocket. At the end of the month the student has to pay and the teacher will get his or her salary.

Business Value

1. It saves travel cost of both the students and the teachers by 30%.
2. Tough time like Covid –19 student cannot visit their private tutor; this system is a solution for that as a result it will increase the income of the teachers.
3. During the time of registration, a fixed percentage of amount have to give to the owner of the system, so it will increase the income of the system.
4. The registration fee is fixed for certain time, After the time limit, they have to renew it.
5. Very useful for the jobless people, they can earn money from their home.
6. It reduces hassle by 70%
7. Saves time of both the teachers and students by 50%.

Special Constraints

1.The system needs to be delivered by January 2021 before new class starts.

2. Every teacher have to provide demo video in their profile.

3.The teacher need to verify their S.S.C./O levels, H.S.C/A levels, BSC certificate (if required) and necessary document for registration.

4.The students also have to provide necessary documents (his/her or parents NID, educational information, address, phone number) for authentication purpose.

Functional Requirements:

1.Students Account:

- The system will allow the students to create a profile by verifying student's identity with necessary information.
- Applicable for all levels of students.
- Capable to interact with tutors after confirmation via private chat or video conference.

2.Tutor Account:

- The system will allow the tutors to create a profile by verifying their identity with necessary information.
- The tutors will have tiers right from professional teachers to part-time student tutors based on experience and position
- Hourly fee is how we measure tutor's accountability
- Tutors can be browsed category-wise

3.Admin Management:

- Dedicated for skeptical verification during registration process of both tutors and students
- Based on user feedback, able to adjust features as per needed and make changes whenever possible
- Only developers, owner and 1 HR person can be admin

4. Business Management:

- For students, mere annual subscription is a must beforehand for access to our services
- For teachers, a percentage of their earnings will be our commission

Non-functional Requirements:

1.Operational:

- The system should be both a website and a cross-platform app
- Online teaching session among students and tutors from anywhere in the country

2.Performance:

- Our system is scalable to no limit to number of students or teachers
- The system is snappy and light-weighted for the convenience of all kinds of net users
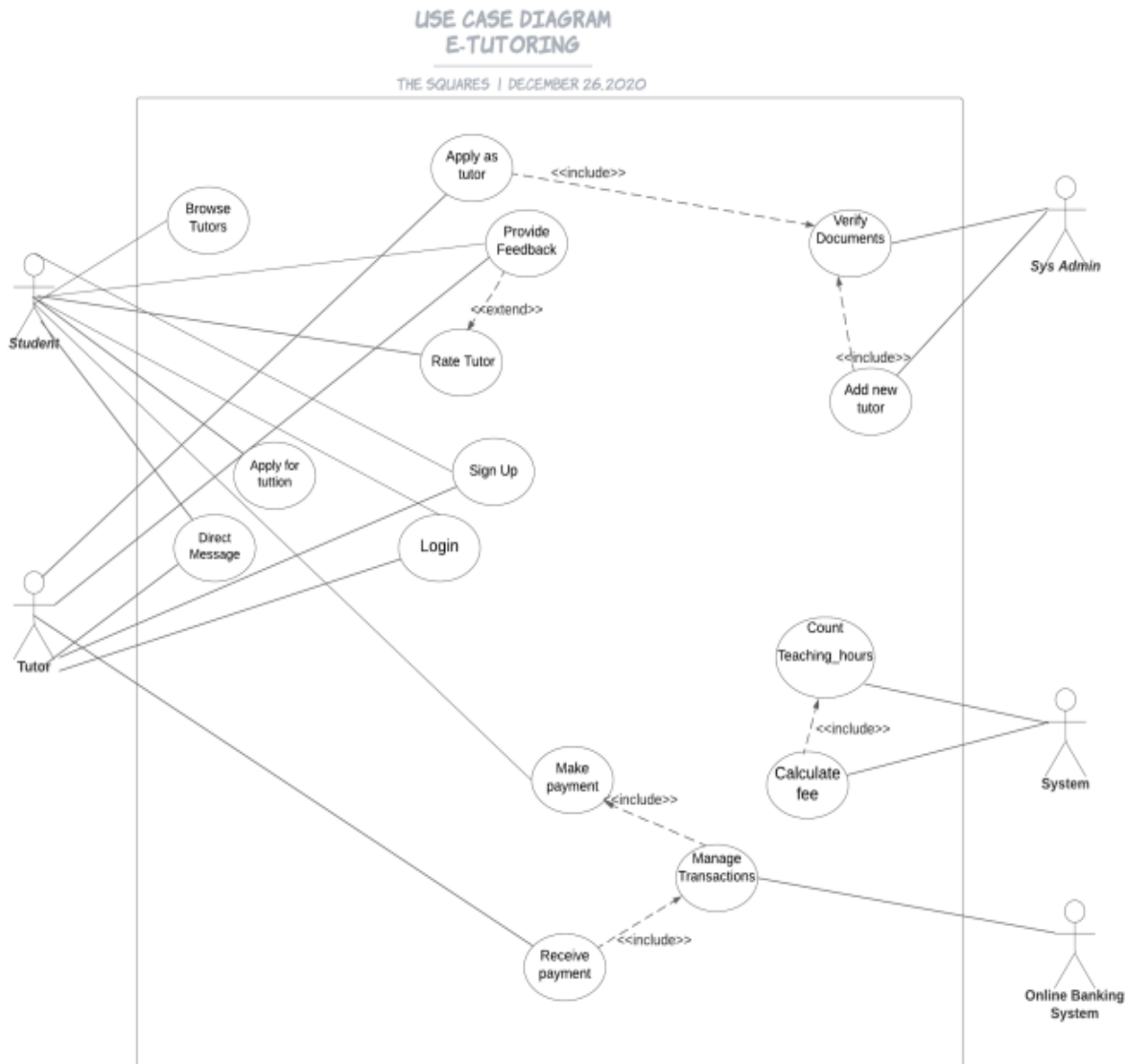
3. Security:

- Private info is not made public for any type of users
- The system includes all available safeguards.

4.Cultural and Political:

- Shield against data exploitation with the Privacy Policy Act

Use Case:

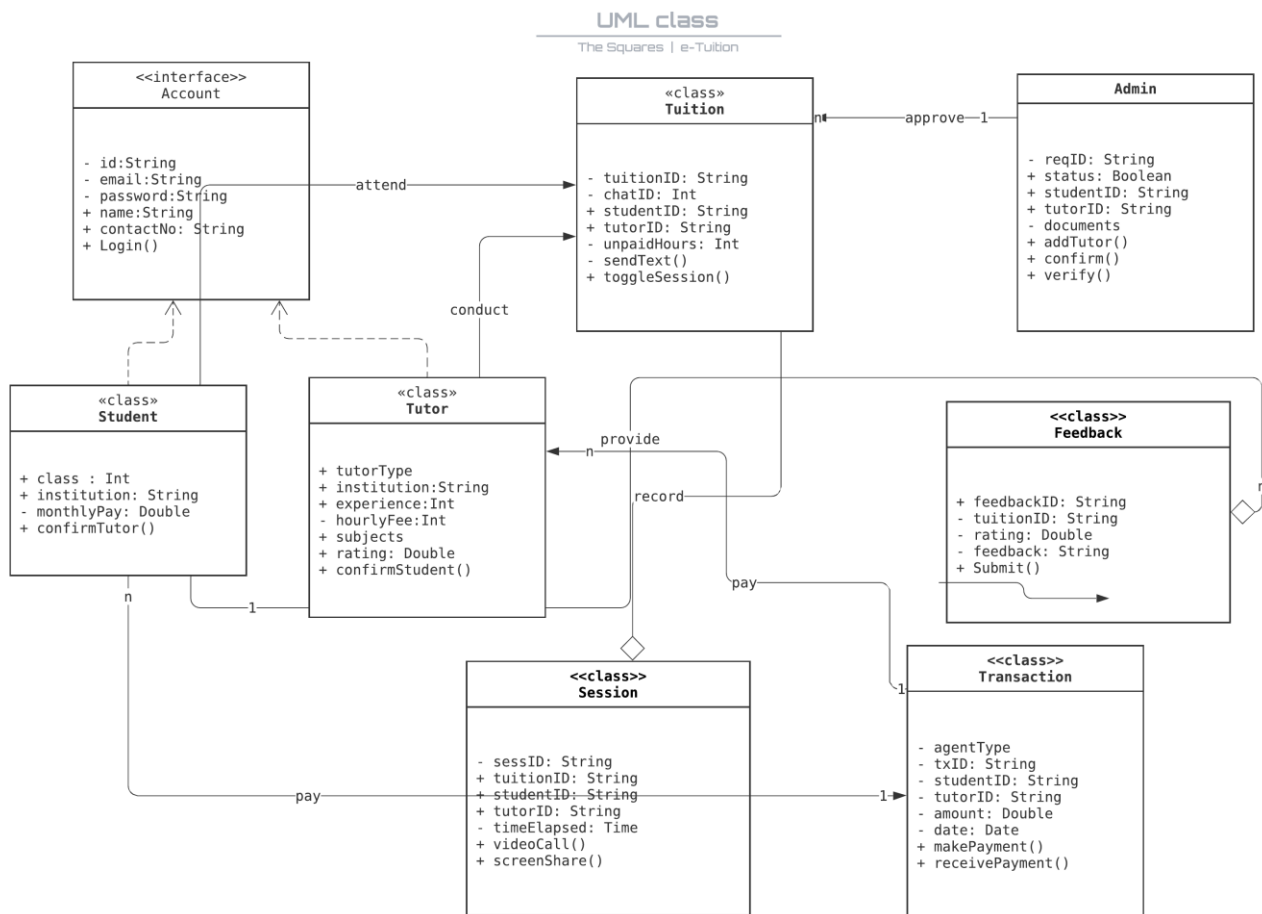USE CASE DIAGRAM
E-TUTORING

THE SQUARES | DECEMBER 26,2020

Description of Use Case:

In this use case diagram, student and tutor are the primary actors and Sys admin, System and online banking system are the secondary actor. The use cases of students are: sign up, log in, browse tutors, provide feedback, rate tutor, apply for tuition, direct message and make payment. The use cases of tutors are: login, sign up, direct message, apply as tutor, provide feedback about the system and receive payment. The use cases of Sys admin verify documents which were provided by the tutors and add them. The system will

count teaching hours and calculate fee based on that. Lastly, online banking system will handle the money transactions.
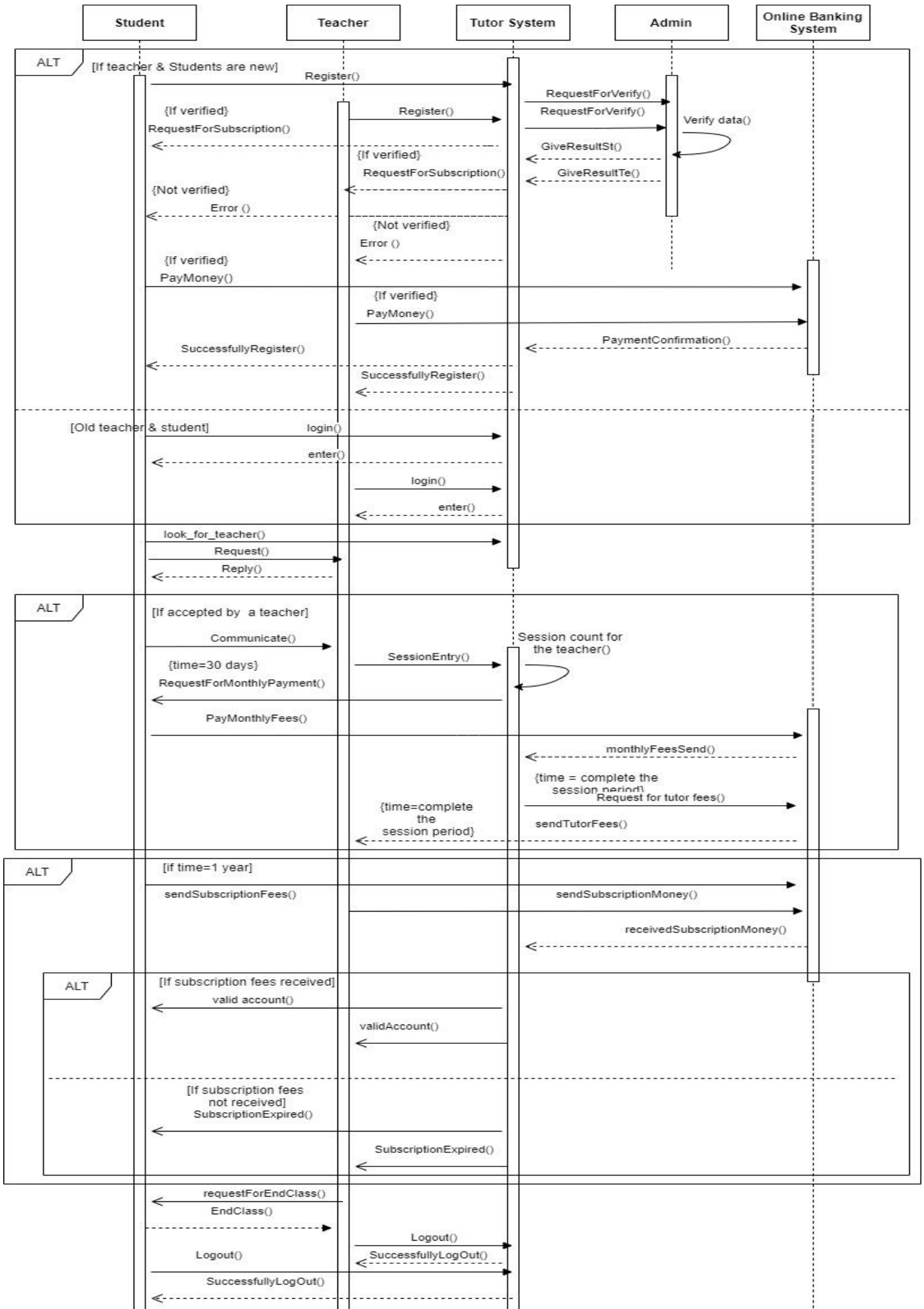
Class Diagram:

Description of Class diagram:

In our class diagram, we have an interface Account which will serve as blueprint for both Student and Tutor class. A student can also be a tutor like a Student Tutor. As for merging them, we have Admin class who will approve any tuition created for students with a designated tutor in a Tuition class. For keeping track, we have a Session class which has the features of video calling and screen sharing. In addition to that, in order to handle payment in both parties we have a Transaction class and a Feedback class for keeping tutors accountable. All of the relationships, methods and private variable are showcased properly with no margin of error here.

Sequence Diagram

| | Student | Teacher | Tutor System | Admin | Online Banking System |

ALT [If teacher & Students are new]

Register()

RequestForVerify()
RequestForVerify()

{If verified}
Register()
RequestForSubscription()

Verify data()

GiveResultSt()

{If verified}
RequestForSubscription()

GiveResultTe()

{Not verified}
Error ()

{Not verified}
Error ()

{If verified}
PayMoney()

{If verified}
PayMoney()

PaymentConfirmation()

SuccessfullyRegister()

SuccessfullyRegister()

[Old teacher & student]
login()

enter()

login()

enter()

look_for_teacher()
Request()
Reply()

ALT [If accepted by a teacher]

Communicate()

SessionEntry()

Session count for the teacher()

{time=30 days}
RequestForMonthlyPayment()

PayMonthlyFees()

monthlyFeesSend()

{time = complete the session period}
Request for tutor fees()

{time=complete the session period}
sendTutorFees()

ALT [if time=1 year]

sendSubscriptionFees()

sendSubscriptionMoney()

receivedSubscriptionMoney()

ALT [If subscription fees received]
valid account()

validAccount()

[If subscription fees not received]
SubscriptionExpired()

SubscriptionExpired()

requestForEndClass()
EndClass()

Logout()

Logout()
SuccessfullyLogOut()
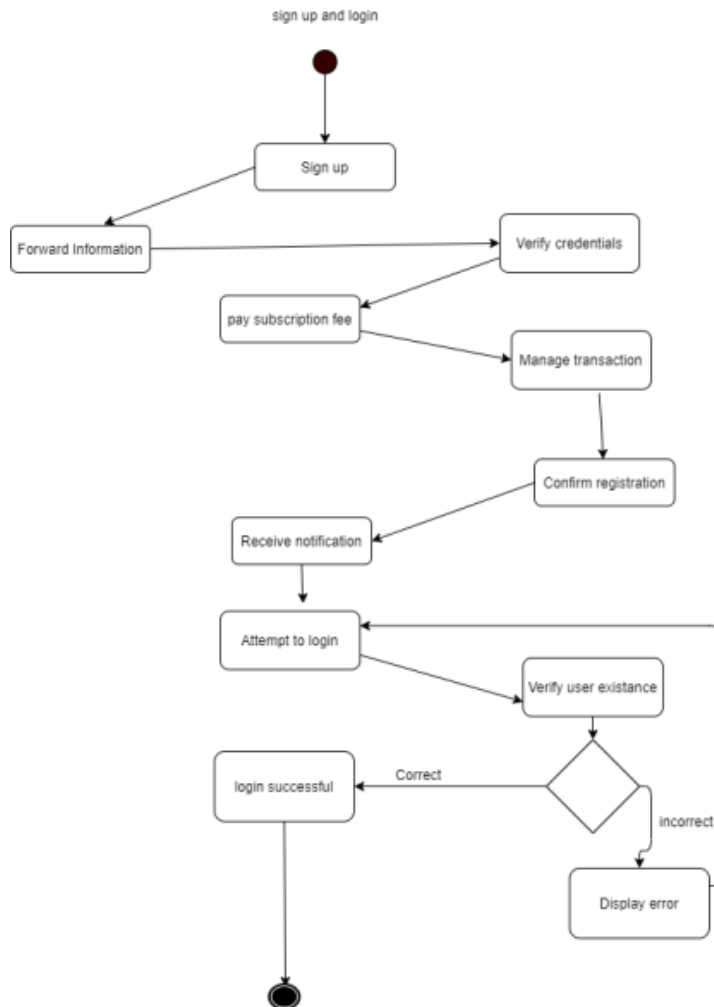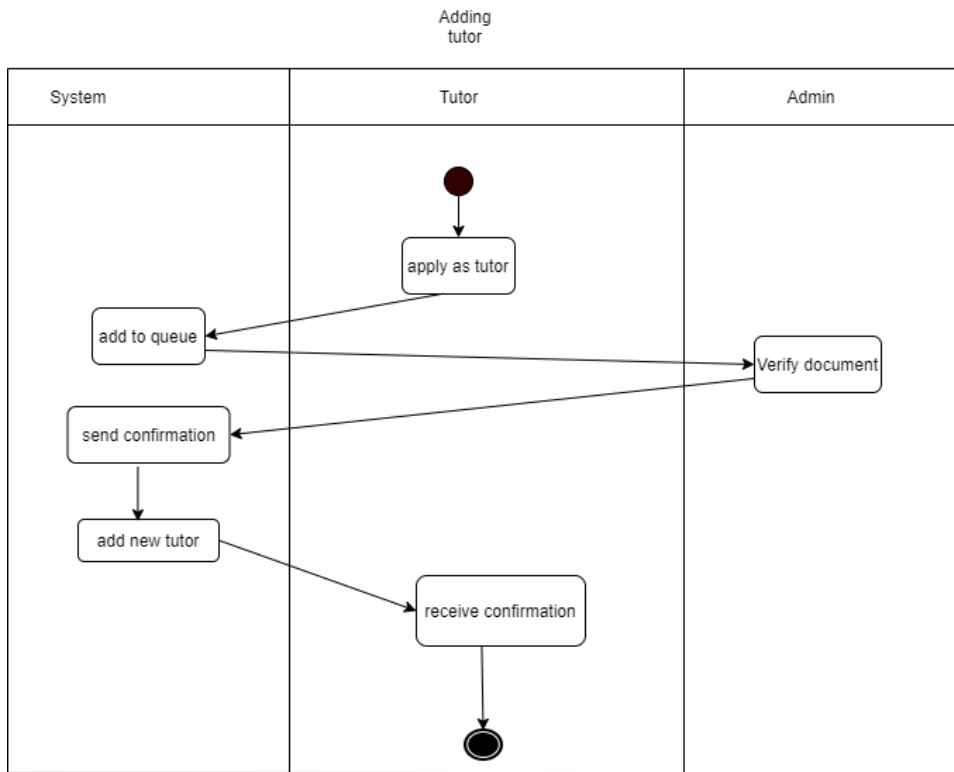
SuccessfullyLogOut()

SuccessfullyLogOut()

Description of sequence diagram:

Firstly, a new student and teacher need to register to the system. They have to provide necessary document to the system. The admin will verify the documents. If the document is verified than both of them need to buy a subscription for 1 year. They will pay the money through online banking. When the system gets the confirmation from the online banking system than the teacher and student can successfully register to the system. After that they have to login to the system. A student can view various teachers' profile and choose someone who is suitable for him or her. Then the student will send request to the teacher if the requested teacher accept the request than the student and teacher can start their class through screen sharing, text messaging and video calling. After discussing they will select class time. The system will automatically record the teacher session time. After one month the student needs to pay the fees through online banking only if he or she is accepted by a teacher. If the teacher completed his or her required session than will get the monthly salary. After 1 year both the teacher and student have to buy the subscription again unless their account will be disable. Finally, after finishing their classes they can log out from the system.
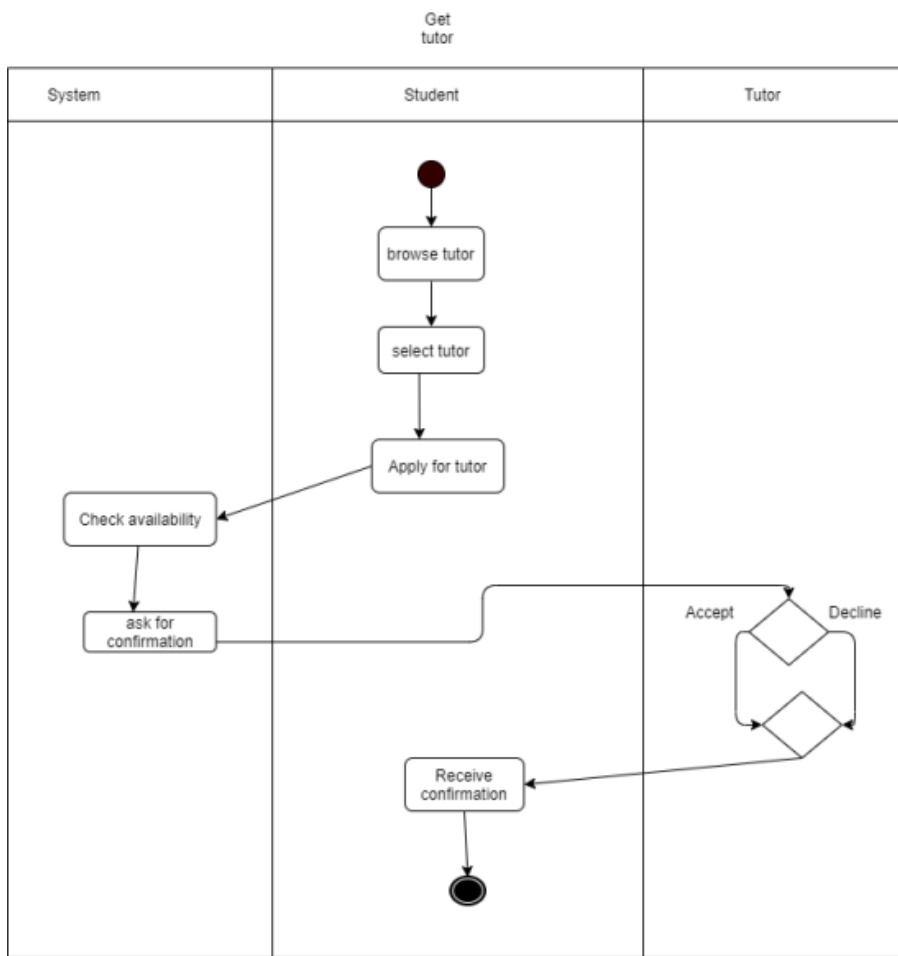
Activity diagrams:

sign up and login

Sign up

Forward Information

Verify credentials

pay subscription fee

Manage transaction

Confirm registration

Receive notification

Attempt to login

Verify user existance

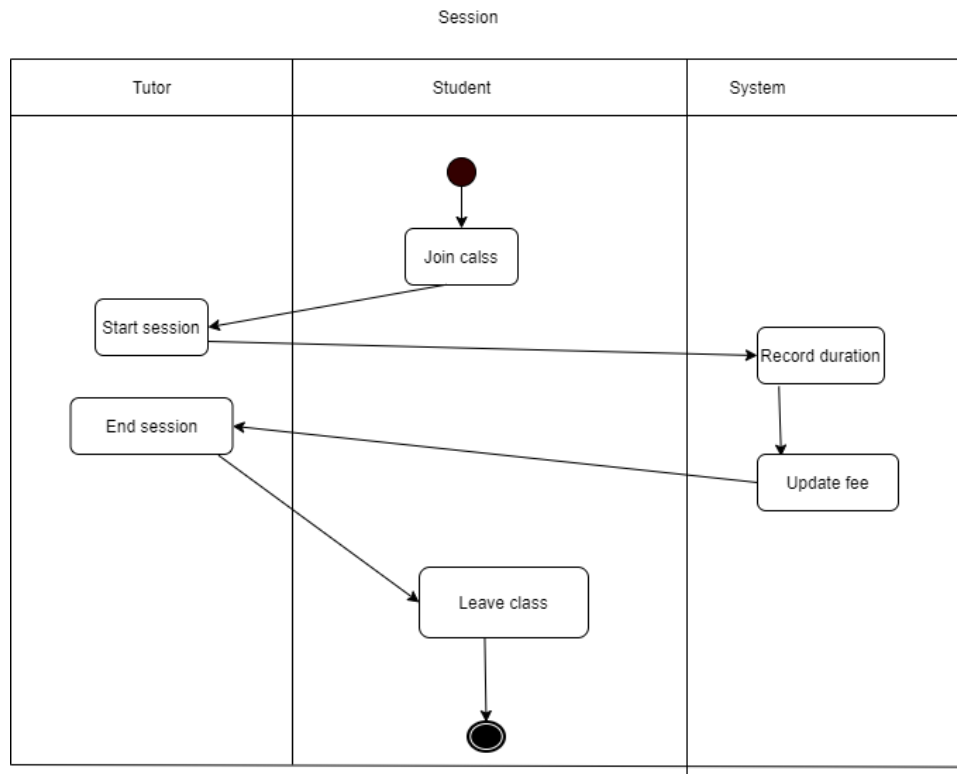login successful — Correct

incorrect

Display error

In the first diagram, teachers and students will sign up and login. Both need to pay subscription fee in order to use the platform. Their credentials will be verified and the registration will then be successful. After registration, users will be able to sign in using proper username and password.
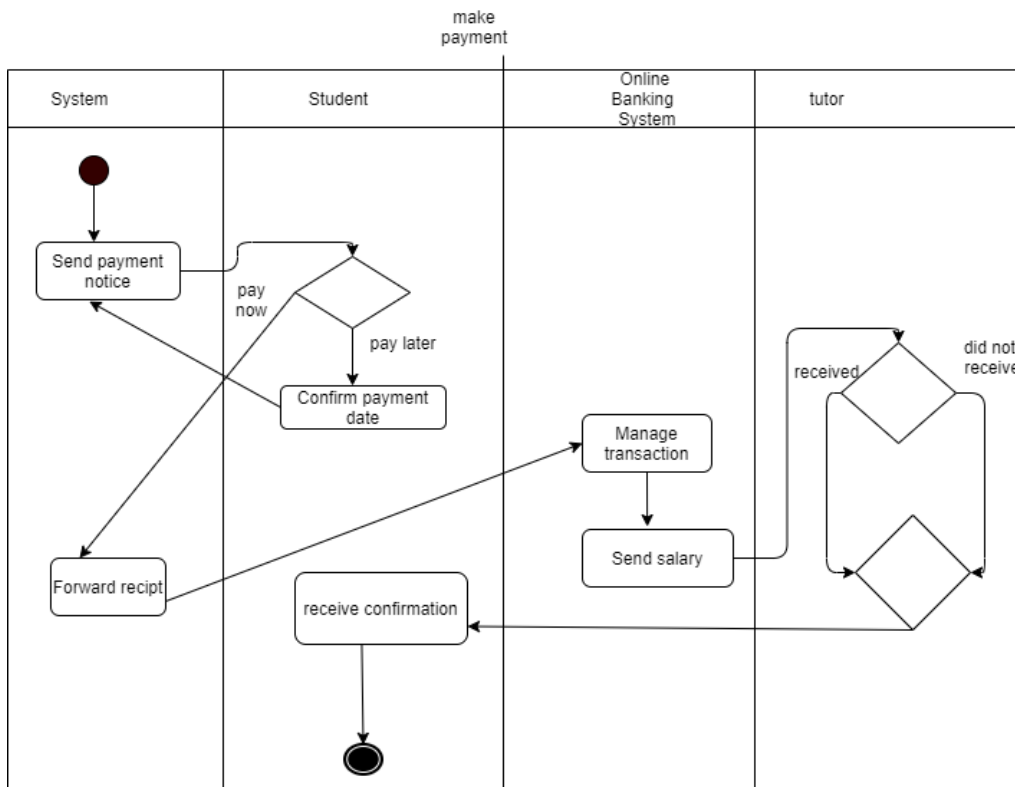
Adding tutor

In this diagram, if anyone wishes to join as a tutor then he/she will need to apply and provide credentials.

After verification by the admin, he/she will be added and will receive confirmation.

Get tutor

In the third diagram, if anyone wishes to get a tutor then will browse and select and apply for that tutor. Then system will check availability and ask for confirmation from the tutor but it will be up to him/her to accept the proposal or not.
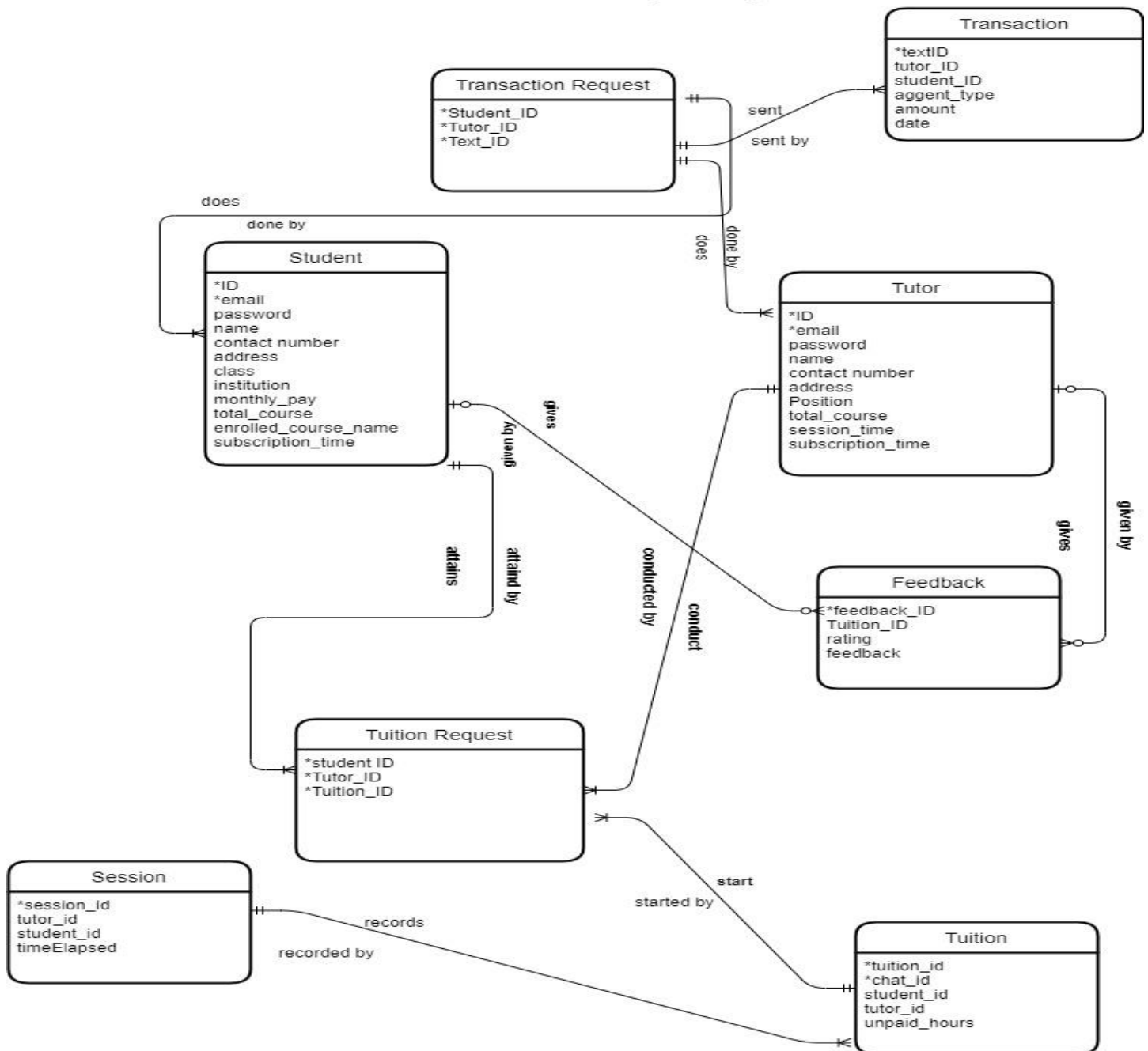

In the fourth diagram, students will join class and teachers will start the sessions. The system will record the duration and update fee. After teacher ends the sessions, the teacher and students will leave the class.

The final diagram is about the tutors' payment. The student will receive a payment notice from the system. If he/she decides to pay later, must confirm the payment date. If the decision is to pay now, the receipt will be forwarded to online banking system which will send salary to the tutor and the tutor will give student a confirmation about whether or not he/she has received the payment.
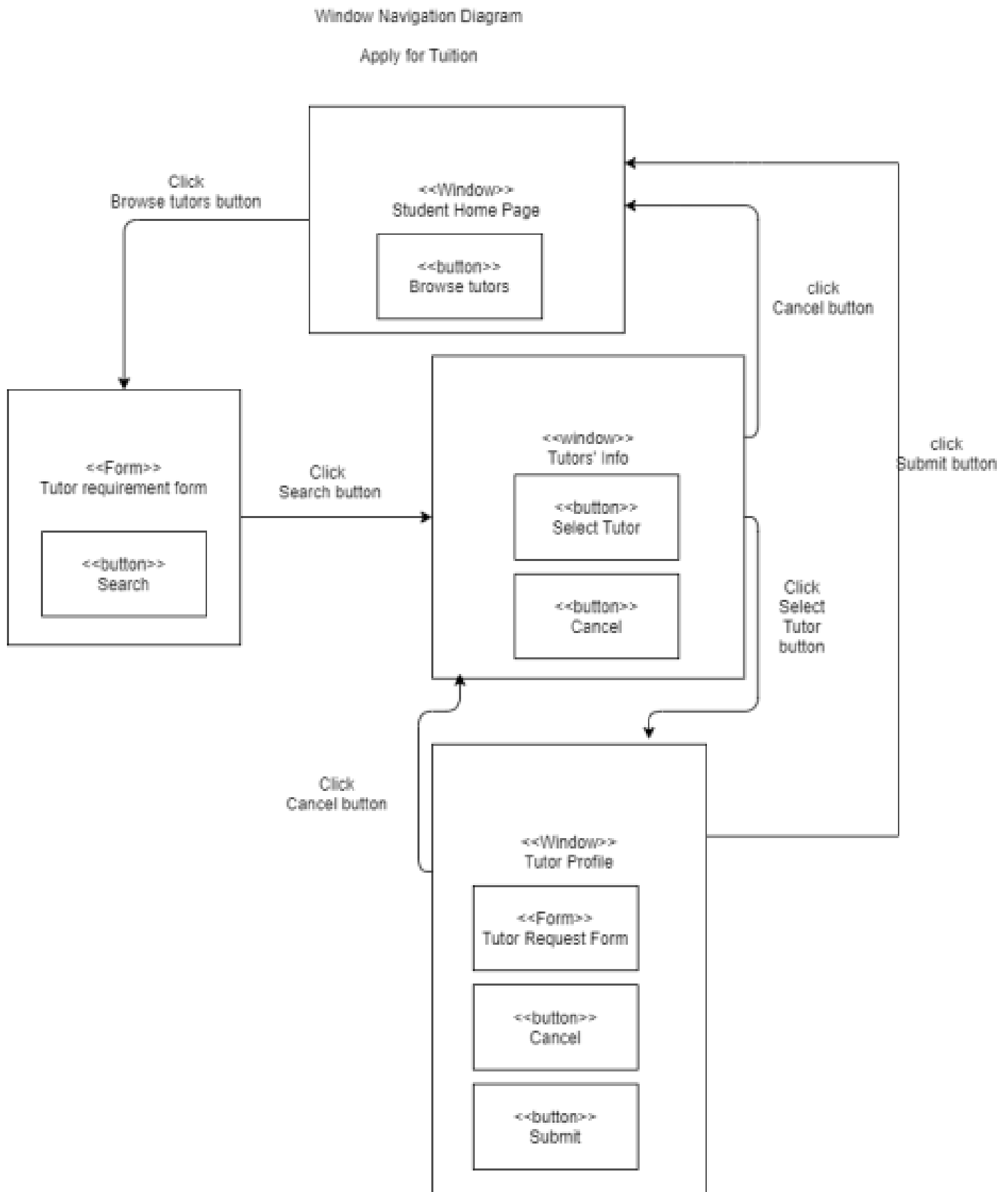
Entity Relationship Diagram:

e-Tuition (ERD)

ERD Description:

In the ER diagram there is two account one for student and one for teacher account. As both of them have to subscribe before creating an account so I create relationship with the accounts and transaction entity. As the accounts and transaction entity have many to many relationships so decomposed it to many to one relationship to a new entity called transaction request. Any account can give feedback regarding the system, tuition, student and teacher which will be recorded. One student and teacher can create a tuition which time, unpaid hours, student, teacher id will be recorded and from the tuition, the duration will be recorded in the session.

Window diagram:

Window Navigation Diagram

Apply for Tuition

Click
Browse tutors button

<<Window>>
Student Home Page

<<button>>
Browse tutors

click
Cancel button

<<Form>>
Tutor requirement form

Click
Search button

<<button>>
Search

<<window>>
Tutors' Info

<<button>>
Select Tutor

<<button>>
Cancel

click
Submit button

Click
Select
Tutor
button

Click
Cancel button

<<Window>>
Tutor Profile

<<Form>>
Tutor Request Form

<<button>>
Cancel

<<button>>
Submit

WND description:

For window navigation diagram, we used the use case" Apply for Tuition". This activity belongs to the student. After login, the student will see a homepage where he will find the browse tutor button. After clicking that button, he will fill up the tutor requirement form which will contain his preferences like university, department, gender etc. After filling the preferences, he will click the search button which will navigate to a window called "Tutors' Info" where he will find a queue containing the information of the tutors that match his requirements. Then he will select a tutor and read his profile from "Tutor Profile" window. If he is satisfied, then he will fill up the tutor request form mentioning that he is interested and also give the preferred schedule and click the submit button which will take him back to the home page. If the student is not interested, he will click cancel button to go back to the "Tutors' Info" window.

Conclusion

Our e-tuition is a platform for student of any background, class, level. Anyone can learn here and can be a student. This platform also helps a tutor to earn by just staying at home. It is user friendly and very convenient to use. Now-a-days finding an appropriate home tutor become a very difficult task. In this system every teacher has to provide demo video and there will be rating for the teacher which will help the student to find their preferable tutor. Furthermore, in normal system a tutor may have to struggle with their fess, but in this system a tutor will get the right amount of fees on the right time. We want users to feel comfortable while using this as we will provide proper security to their information.