



OLS Devs

Requirements Document

TutorBuddy

Team Members:

220032882, Ovayo Matiwane

220907730, Leah Mujeyi

220241406, Ntabozuko Sana

$\frac{90}{120}$

75%

Mel

Date Submitted: 9 May 2021

TABLE OF CONTENTS

1 SYSTEM OVERVIEW	3
1.1 Problem Description	3
1.2 Project Assumptions	3
2 FUNCTIONAL REQUIREMENTS	4
2.1 Business Use Case Model	4
2.2 Use Case Glossary.....	5
2.3 Use Case Narratives (User Stories)	7
2.3.1 Package A: Academic Package	7
2.3.2 Package B: Automated Package.....	10
2.3.3 Package C: Administration	12
3 NON-FUNCTIONAL REQUIREMENTS.....	15
3.1 Interface Requirements	15
3.2 Performance Requirements.....	15
3.3 Security Requirements	15
3.4 Operational Requirements.....	15
4 DATA REQUIREMENTS.....	16
4.1 Preliminary List of data and attributes	16
4.2 Domain Class Diagram.....	17
5 PRELIMINARY SCHEDULE OF RESPONSIBILITIES	18
6 RESEARCHING THE REQUIREMENTS.....	19
10 User Interview Questions	19

1 SYSTEM OVERVIEW

1.1 Problem Description

At Nelson Mandela University, students have to take a certain number of modules before a degree or certificate can be conferred. There are set times for lectures, practicals and tutorials but often students find themselves needing extra assistance to grasp the required concepts. To help the students cope with the course load, the TutorBuddy system will allow students to request additional help from senior students who have successfully completed the certain module with a mark of at least 70%.

The system will be accessible to all registered students at the university as well an administrator. The benefits offered by such a system are an insight of which modules are requested the most by students for tutors, this may indicate a need to change the way the module is offered by the lecturer. Senior students registered as tutors will also be able to get additional income.

The payment details between tutors and students are outside the system scope and are not captured by the system. The student pays the tutor directly.

1.2 Project Assumptions

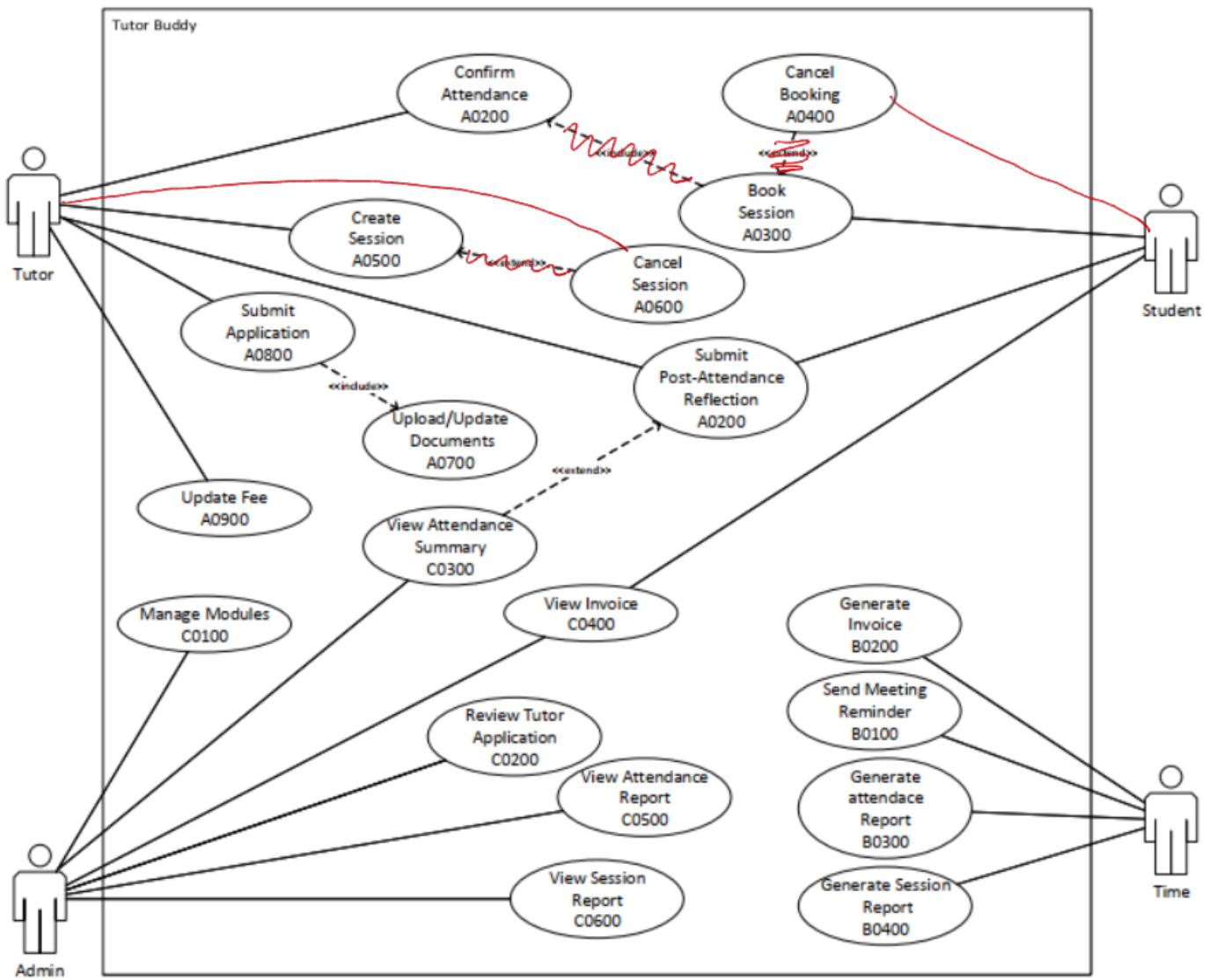
- All the system users are affiliated with the university.
- The university appoints one person to oversee the administration of the system.
- Nelson Mandela university students automatically have access to the system. That means by being a registered student, one can search for modules on the system and book the available sessions.
- The system is also affiliated with the university.
- In our use case model, a session is when a tutoring session has been scheduled - which may or may not happen, depending on whether the affected parties attend or cancel. An attendance is when a session actually takes place.

So this means the Student table is pre-populated.

Suggestion: Have a use case "Load Students" with an Admin that has access to this. It can be a text file that gets imported to the Student table.

2 FUNCTIONAL REQUIREMENTS

2.1 Business Use Case Model



2.2 Use Case Glossary

Package Name: Academic Package ✓		
Use Case Id	Use Case Name	Actors
A0100	Confirm Attendance	Student, Tutor
A0200	Submit Post-Attendance Reflection	Student, Tutor
A0300	Book Session	Student
A0400	Cancel Booking	Student
A0500	Create Session	Tutor
A0600	Cancel Session	Tutor
A0700	Upload/Update Documents	Tutor
A0800	Submit Application	Tutor
A0900	Update Fee	Tutor

Package Name: Automated Package ✓		
Use Case Id	Use Case Name	Actors
B0100	Send Meeting Reminder	Time
B0200	Generate Invoice	Time
B0300	Generate Attendance Report	Time
B0400	Generate Session Report	Time

Package Name: Administration Package ✓		
--	--	--

Use Case Id	Use Case Name	Actors
C0100	Manage Modules	Admin
C0200	Review Tutor Applications	Admin
Queries/Reports		
C0300	View Attendance Summary	Admin, Student, Tutor
C0400	View Invoice	Admin, Student
C0500	View Attendance Report	Admin
C0600	View Session Report	Admin

2.3 Use Case Narratives (User Stories)

2.3.1 Package A: Academic Package

Use Case ID	Use Case Name
A0100	Confirm Attendance
Primary Business Actors	Other participating Actors
Student, Tutor	
Description	Users confirm that they have/are attending a session.
Pre-Conditions	A session needs to have been created by the tutor and the student must have booked to attend the session.
Triggers	<ul style="list-style-type: none"> Time Under the user portal, a button pops up next to the scheduled session under the list of scheduled sessions, user clicks on the "confirm attendance" button.
Post-Conditions	User attendance to a session is confirmed and registered.
Basic Flow of Events	<ol style="list-style-type: none"> When the time scheduled, a "Confirm Attendance" button is activated next to the listed schedule. User clicks on the button. Attendance is confirmed For student, fee is added to the database The scheduled session is marked as attended The scheduled listing is removed from the list of booked or scheduled sessions on the user portal.
Alternate Flow of Events	<ol style="list-style-type: none"> Student does not click on the button for the entire duration of the session. Student is charged a "no cancellation fee". The scheduled session is marked as unattended. Basic flow 6.
Alternate Flow of Events	<ol style="list-style-type: none"> Tutor Does not click on the button for the entire duration of the session. Scheduled session is marked as unattended. Tutor is removed as a tutor for that module, they will need to re-apply. Basic flow 6.

Use Case ID	Use Case Name
A0200	Submit Post-Attendance Reflection
Primary Business Actors	Other participating Actors
Student, Tutor	
Description	Users submit a reflection on the work covered during a session and a rating of how helpful it was for the student.
Pre-Conditions	Actors must have confirmed attendance to the session.
Triggers	<ul style="list-style-type: none"> At the end of a session, "Reflection" button is activated next to the session listing. Clicking on the button triggers the use case. This use case is optional for all users.
Post-Conditions	Users submit a report on the session in the form of a form and it is stored in the database.
Basic Flow of Events	<ol style="list-style-type: none"> User clicks on the "Reflection" button. <i>trigger</i> A form pops up with general questions. User clicks on the ratings appearing on the form. User clicks on submit form. Reflection is stored and reflection button is deactivated.

	6. System displays the user portal.
Alternate Flow of Events	4. User does not click the submit button and leaves page. 5. Reflection button remains activated.

Use Case ID	Use Case Name
A0300	Book Session
Primary Business Actors	Other participating Actors
Student	Tutor
Description	The user books for a session that they want to attend.
Pre-Conditions	A session must be created by the tutor with the necessary details
Triggers	<ul style="list-style-type: none"> • User requires tutoring for a certain module. • User searches for the module to check for any scheduled sessions. <i>not triggers</i> • Scheduled sessions for the modules are listed. • User clicks on search module button under the user portal
Post-Conditions	A student books to attend a scheduled session.
Basic Flow of Events	<ol style="list-style-type: none"> 1. User clicks on search module button under the user portal. 2. A search pop-up menu is presented by the system. 3. User types in the module code or module name. 4. User clicks on Search. 5. List of possible modules appear. 6. User locates the module. 7. User clicks on the module. 8. Available scheduled sessions appear. 9. Student selects one. 10. Scheduled session details appear in a list on student portal. 11. Student has booked a session. 12. User is returned to portal.
Alternate Flow of Events	<ol style="list-style-type: none"> 6. The module does not appear. 7. Message is written to the user that the module is not on the database. 8. Basic flow 12

Use Case ID	Use Case Name
A0400	Cancel Booking
Primary Business Actors	Other participating Actors
Student	Tutor
Description	The user cancels a booking that they had made.
Pre-Conditions	The user must have booked for a session and the session must not have passed.
Triggers	<ul style="list-style-type: none"> • User wants to cancel a booked session. • Under the user portal, in the list of booked sessions, there is a Cancel button next to the session listed.
Post-Conditions	The booking is canceled and the user is removed from the list of attendees.
Basic Flow of Events	<ol style="list-style-type: none"> 1. User clicks on Cancel Booking 2. Schedule listing is removed from user portal

Use Case ID	Use Case Name
A0500	Create Session
Primary Business Actors	Other participating Actors

Tutor	
Description	The user schedules a tutoring session with the relevant details – date, time, and any other information.
Pre-Conditions	User must be a tutor for the module.
Triggers	<ul style="list-style-type: none"> Under the user portal, a button to create session is listed above the list of sessions created. Clicking on the button triggers this use case.
Post-Conditions	A session is created and students are allowed booking for it.
Basic Flow of Events	<ol style="list-style-type: none"> User Clicks on Create Session A form pops up User fills in the form with session details User clicks OK Session is listed under user portal and can be found when searching for it by students User is returned to portal

Use Case ID	Use Case Name	
A0600	Cancel Session	
Primary Business Actors		Other participating Actors
Tutor		Student
Description	The user cancels a session that they had previously scheduled.	
Pre-Conditions	<ul style="list-style-type: none"> A session must have been scheduled by the user. The starting time for the session must not have passed. 	
Triggers	<ul style="list-style-type: none"> Under the user portal, a list of sessions scheduled by the user appears. Next to each listing there is a Cancel button. Clicking on this button triggers this use case. 	
Post-Conditions	The scheduled session is cancelled and all students that had booked to attend the session are notified.	
Basic Flow of Events	<ol style="list-style-type: none"> User clicks on Cancel Session. System displays a confirm button. User confirms cancelation. Students that booked for the session are automatically emailed. Session is canceled. Session is marked as canceled under portal until after the scheduled time passes. 	

Use Case ID	Use Case Name	
A0700	Upload/Update Documents	
Primary Business Actors		Other participating Actors
Tutor		
Description	The user uploads the required documents to be a tutor for that certain module – transcript and identification.	
Pre-Conditions	The user must have applied to be a tutor for one or more modules, or they are adding modules to the current ones. This means they have a pending application.	
Triggers	<ul style="list-style-type: none"> Under the user portal, next to the pending application listing, there is a “Upload documents” button – clicking on this button triggers this use case. 	
Post-Conditions	The user documents are uploaded/ updated.	
Basic Flow of Events	<ol style="list-style-type: none"> User clicks on upload documents Upload window pops up 	

	<ol style="list-style-type: none"> 3. User is prompted to browse for documents. 4. User finds document and uploads it. 5. User selects Okay. 6. Document is uploaded by the system. 7. Success message is displayed. 8. System returns to user portal. 	✓
--	--	---

Use Case ID	Use Case Name	
A0800	Submit Application	
Primary Business Actors		Other participating Actors
Tutor		Admin
Description	The user applies to become a tutor for a certain module/s.	
Pre-Conditions	The student must qualify to tutor those modules, that means they must have completed it already and have received a mark of 70% or more for the module.	
Triggers	Under the student portal, student clicks on the “Apply to tutor” button.	
Post-Conditions	The student is accepted to be a tutor and can schedule tutoring sessions. Alternatively they are rejected.	
Basic Flow of Events	<ol style="list-style-type: none"> 1. Students clicks on Apply to Tutor Button. 2. Application form pops up. 3. Student fills out form. 4. Students clicks on submit. 5. Success message is displayed. 6. System returns to user portal. 	

Use Case ID	Use Case Name	
A0900	Update Fee	
Primary Business Actors		Other participating Actors
Tutor		
Description	The tutor changes the hourly fee they charge for sessions.	
Pre-Conditions	The user needs to be a tutor already.	
Triggers	Under the user portal, the tutor clicks on “Update fee”.	
Post-Conditions	The fee is updated and the tutor’s profile on the database is updated.	
Basic Flow of Events	<ol style="list-style-type: none"> 1. User clicks on trigger button. 2. A form pops up displaying current fee. 3. Students enters new fee and clicks on OK. 4. Success message is displayed. 5. System returns to user portal. 	

2.3.2 Package B: Automated Package

Use Case ID	Use Case Name	
B0100	Send Meeting Reminder	
Primary Business Actors		Other participating Actors
Time		Student, Tutor

Description	Thirty minutes before a session is to start, the tutor and student receive a notification reminding them about the session
Pre-Conditions	<ul style="list-style-type: none"> There needs to be a meeting that is booked/scheduled between a tutor and a student or students
Triggers	Time
Post-Conditions	All the users meant to attend the session will receive a notification reminding them about the meeting
Basic Flow of Events	<ol style="list-style-type: none"> Trigger Occurs Users are notified about their scheduled session

Use Case ID	Use Case Name
B0200	Generate Invoice
Primary Business Actors	
Time	
	Admin, student
Description	At the end of each month, an invoice detailing how much each student owes for the months lessons is generated.
Pre-Conditions	<ul style="list-style-type: none"> There needs to be at least one session that would have occurred in the month
Triggers	Time
Post-Conditions	Invoice is sent to student. Invoice is generated and made available
Basic Flow of Events	<ol style="list-style-type: none"> Trigger occurs Total owed is calculated Invoice is generated Invoice is made available for viewing

Use Case ID	Use Case Name
B0300	Generate attendance Report
Primary Business Actors	
Time	
	Student, Tutor, Admin
Description	At the end of each month, the attendance report showing when a session occurred and how many students were present in the session is generated.
Pre-Conditions	Have at least one session occurred where a tutor and student both confirmed attendance
Triggers	Time
Post-Conditions	An attendance Report is made available for viewing
Basic Flow of Events	<ol style="list-style-type: none"> Trigger occurs Attendance record is generated and made available for viewing

Use Case ID	Use Case Name
B0400	Generate Session Report
Primary Business Actors	
Time	
	Admin
Description	At the end of each month, a report showing how many sessions were booked for each module is generated and made available for viewing by admin
Pre-Conditions	There needs to be at least one session that was booked for the particular module

Triggers	Time
Post-Conditions	A session report is generated
Basic Flow of Events	<ol style="list-style-type: none"> 1. Trigger occurs 2. Session report is generated and made available for viewing

2.3.3 Package C: Administration

Use Case ID	Use Case Name
C0100	Manage modules
Primary Business Actors	Other participating Actors
Admin	
Description	The university may at any given point add or remove modules offered in courses. The admin will have to make sure the modules available on the tutor buddy are up to date
Pre-Conditions	User must be employed as a staff member by the university Have the need to maintain a module
Triggers	<ul style="list-style-type: none"> • A new module may need to be added • An existing module may need to be removed
Post-Conditions	A module has been added or removed
Basic Flow of Events	<ol style="list-style-type: none"> 1. User logs in using university login details 2. The home page is displayed 3. Users taps modules 4. The user has the option to add or remove a module 5. The user taps add module 6. User enters relevant module information on pop-up 7. The user taps add 8. User taps confirm the details of the module to be added are correct 9. The module is added 10. Returns to home page
Alternate Flow of events	<ol style="list-style-type: none"> 5. User taps delete module 6. User is asked to enter module code 7. Module is displayed and user is asked to confirm that they would like to delete the module 8. The module is deleted 9. Basic flow 10

Use Case ID	Use Case Name
C0200	Review Tutor Application
Primary Business Actors	Other participating Actors
Admin	Tutor

Description Admin approves the tutor applications for the module

Pre-Conditions	<ul style="list-style-type: none"> • Must be admin • Must have a tutor request to approve for a particular module
Triggers	The administrator has a request to approve
Post-Conditions	A tutor is either approved or rejected
Basic Flow of Events	<ol style="list-style-type: none"> 1. From the home page, review applications is clicked 2. List of new tutor applications is shown 3. Admin selects tutor to review 4. Admin clicks check documentation 5. Admin clicks approve tutor 6. Tutor receives feedback email
Alternate flow of events	<ol style="list-style-type: none"> 1. Admin clicks reject tutor 2. Basic flow 6

Use Case ID	Use Case Name	
C0600	View Session Report	
Primary Business Actors		Other participating Actors
Admin		Time
Description		The admin will be able to see how many session were booked throughout the month and
Pre-Conditions		User must be logged in with valid university credentials Booking report needs to have been generated
Triggers		User selects view then session booking report
Post-Conditions		The system displays a report of all the sessions booked for specified period
Basic Flow of Events		<ol style="list-style-type: none"> 1. From the homepage user clicks view drop down menu 2. User selects “session report” from the menu 3. User clicks desired month 4. Booking report for that month is displayed

Use Case ID	Use Case Name	
C0300	View Attendance Summary	
Primary Business Actors		Other participating Actors
Admin, Student, Tutor		
Description		After each session, the student and tutor upload a summary of what was covered in that session that can then be viewed by admin.
Pre-Conditions		The user must be logged in with valid university credentials. There has to be a session that has occurred.
Triggers		User clicks on view session summary.
Post-Conditions		A summary of the session is displayed.
Basic Flow of Events		<ol style="list-style-type: none"> 1. From the portal page user clicks sessions. 2. User selects the day and time the session occurred or session ID. 3. Session is displayed. 4. User clicks on the session. 5. User clicks view session summary. 6. Session summary is displayed.
Alternate flow of events		<ol style="list-style-type: none"> 3. User asked to pick different session as the one selected has no results.

4. Basic flow of events 2

Use Case ID	Use Case Name
C0500	View Attendance Report
Primary Business Actors	Other participating Actors
Admin	Time
Description	At the end of each month, the admin can view the history of sessions that were attended that month
Pre-Conditions	The user must have valid log in credentials from the university User must be logged in Attendance report needs to have been generated
Triggers	<ul style="list-style-type: none"> The user selects "View" then "Attendance Report"
Post-Conditions	A comprehensive view of the attendance is displayed
Basic Flow of Events	<ol style="list-style-type: none"> From the portal screen user clicks view dropdown menu User clicks "attendance report" The user has the option of viewing the attendance for current month or past months User clicks desired month The system displays the attendance report for that month

Use Case ID	Use Case Name
C0400	View Invoice
Primary Business Actors	Other participating Actors
Admin, Student	Time
Description	The admin will be able to view each student's invoice which will show the student's name, how many sessions they attended as well as how much they owe.
Pre-Conditions	<ul style="list-style-type: none"> Users must be affiliated with the university There needs to have been at least one session that occurred
Triggers	Clicking 'view invoice'
Post-Conditions	System displays invoice on screen
Basic Flow of Events	<ol style="list-style-type: none"> Once the portal page is displayed User clicks the view invoice button User has option to pick from current invoice or past invoices User clicks current invoice Invoice is displayed

3 NON-FUNCTIONAL REQUIREMENTS

3.1 Interface Requirements

The following interface requirements will be ensured in the TutorBuddy web application:

- The application must be adaptive to any device being used to access the website – this includes laptops, tablets, smartphones etc.
- The interface must provide enough information to provide ease of access to the users.
- There are general functionalities that most websites follow, i.e. the general menu at the top of the page, the website must stick to these as much as possible to avoid confusing the users.
- A general theme will be chosen for the website and we will try to stick to it for all the webpages as much as possible.

3.2 Performance Requirements

The following are the performance requirements that the system must meet:

- The system must not be dependent on the hardware properties of the user – i.e. it should be able to run on any device that connects to the internet.
- The system should take as little time as possible to load pages and perform commands.
- The system should use as little pictures as possible to avoid heavy data requirements when loading.
- The page-to-page lag time of the system must be optimized so that it is always at its lowest to keep users happy.

3.3 Security Requirements

The following security requirements will be ensured in the TutorBuddy web application:

- Users are automatically registered with their student or staff numbers – this means they have to use passwords to log onto the system, this ensures personal information stored on the system is secured.
- Distinguishing between the three users; student, tutor and administrator allows for protection against malicious tampering to the users of the system.

3.4 Operational Requirements

The following operational requirements apply for users to have ease of access to the system

- The system is web-based, this means that any device that can connect to the internet will be able to use the system, this eliminates a lot of limitations.
- A user requires internet connectivity in order to be able to access the system.
- The system uses a relational database management system and we will queried using SQL.
- The system will be hosted at one of the public hosting companies i.e. Afrihost, MWeb, AWS etc.

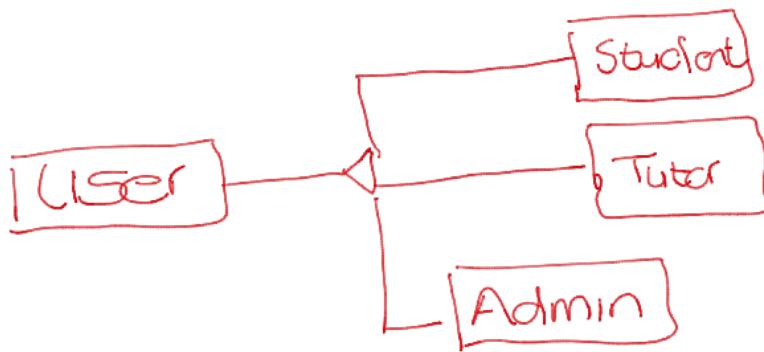
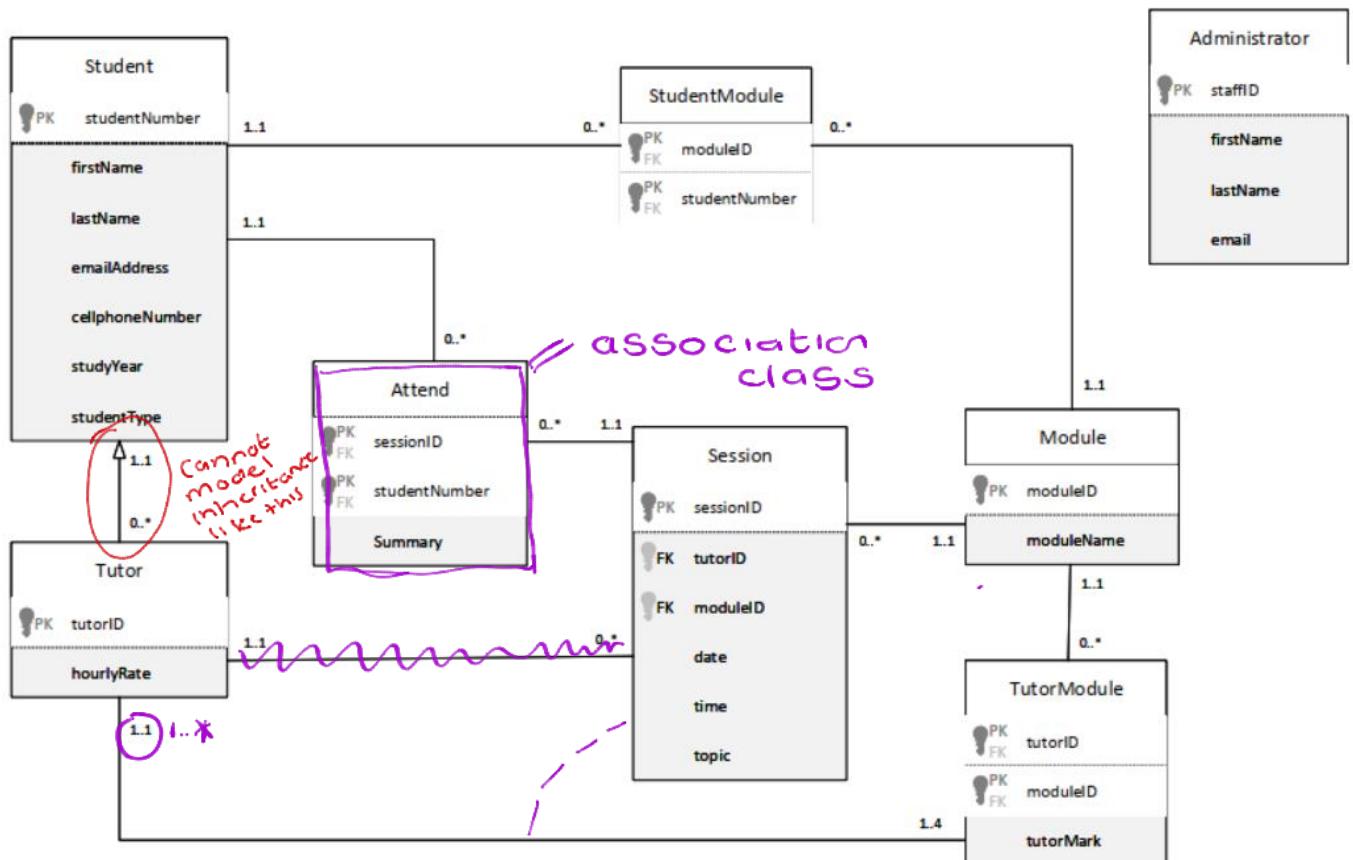
4 DATA REQUIREMENTS

4.1 Preliminary List of data and attributes

Entity	Entity Attributes
Administrator	staffID<pk> firstName lastName email
Student	studentNumber<pk> firstName lastName emailAddress cellphoneNumber studyYear studentType
Tutor	Extend Student tutorID<pk> HourlyRate
Module	moduleID<pk> moduleName
Session	sessionID<pk> tutorID<fk> moduleID<fk> date time topic
Attend	sessionID<pk><fk> studentNumber<pk><fk> summary
StudentModule	moduleID<pk><fk> studentNumber<pk><fk>
TutorModule	tutorID<pk><fk> moduleID<pk><fk>

refer to class diagram

4.2 Domain Class Diagram



5 Preliminary Schedule of Responsibilities

Use Case ID & Name	Team Member Responsible	
	Student Number	Name
A0100 – Confirm Attendance A0200 – Submit Post-attendance Reflection A0300 – Book Session A0400 – Cancel Booking C0400 – View Invoice C0300 – View Attendance Summary	220032882	Ovayo Matiwane
B0100 – Send Meeting Reminder B0200 – Generate Invoice B0300 – Generate Attendance Report B0400 – Generate Session Report C0100 – Manage Modules C0200 – Review Tutor Applications C0300 – View Attendance Summary C0400 – View Invoice C0500 – View Attendance Report C0600 – View Session Report	220907730	Leah Mujeyi
A0500 – Create Session A0600 – Cancel Session A0700 – Upload/Update Documents A0800 – Submit Application A0900 – Update Fee C0300 – New Attendance Summary	220241406	Ntabozuko Sana

C400 , C300 should not be shared

6 Researching the Requirements

10 User Interview Questions

Good

Do you have any modules that you require additional academic support for?

2. Would you be willing to use a system that connects you to a tutor for additional academic support?
3. What kind of information should be made available on the system?
4. Do you prefer to search for a tutor amongst a list of tutors, or the system should assign one automatically upon registration?
5. Do you think a TutorBuddy system will aid to academic success?
6. What kind of computing device(s) do you use?
7. Do you prefer to manage your own session bookings?
8. Would you like the option to edit previously uploaded documents?
9. Have you, in the past or currently, used a similar system?
10. Do you prefer that the user information be kept on the system upon successful completion of a module?