RASI Driving School Management System

CSD 4010 Software Development

by

Vishnu Chakravarthi Nallaiah Raj

M00606188

**Contents**

|  |  |  |
| --- | --- | --- |
| S.no | Title | Page No. |
| I | Purpose of the Project | 3 |
| II | Product Description | 3 |
| III | Functional Requirements | 5 |
| IV | Quality Requirements | 6 |
| V | Stakeholders | 6 |
| VI | Use Case Diagram | 7 |
| VII | Prioritization | 8 |
| VIII | Use Case Template | 9 |
| IX | Key Business Process | 11 |
| X | Activity Diagram Model | 13 |
| XI | Acceptance Test | 16 |

1. Purpose of the Project

Mr. Shaji and his family are running a driving school called RASI driving School for the past six years. There are only a few driving schools in Vallioor which cannot manage the number of students willing to learn driving. None of the schools in that area don’t have any computerized system to maintain the student's record and driving classes. The students are registered by the old traditional way of using paper and pen which causes more time and resource. They need separate employees to feed the data to the system. For the past 2 years, the RASI driving school has successfully increased the number of students from its previous years. The situation becomes worse when there are more than 100 students in a single day. They also face the following problems in the existing system.

* The employees need to check a pile of documents to find a student’s record.
* The records of the students and employees are stored in files which are insecure.
* There are no backup files for any of the records.
* Since the class records of students are stored in notebooks, it is unable to analyze the classes and performance of the student.
* The scheduling of classes is incorrect due to the absence of a perfect system.
* Moreover, the students can't view their own records and progress.

1. Product Description

The management needs a new computerized system mainly to register new students and maintain their records securely and to schedule the instructors to the newly registered students. The system has totally three types of login such as

* Admin Login
* Trainer Login
* Student Login

1. Admin Login and Activities:

Admin login is provided to the admin and the office staffs. They can register new users. The users who want to book an appointment through phone can call the admin and they can book an appointment for them. If the user doesn’t know to use the software system, they can visit the office and register their details through admin. They can also schedule the training session for the students. Admin can view all the details of students and trainers. The system should also allow the admin to delete the student details.

1. Trainer Login and Activities:

Trainer login helps the trainers to take the attendance of the students and to update their progress to the system. They can also update their status to the system so that the admin can reschedule the class if the trainer is absent. The system should allow the trainers to view only their student's detail. Hence, if there is any mishandling of the student’s information, the admin knows where to go.

1. Student Login and Activities:

The student login is the major work in the system as they are the main stakeholders. The students should first fill their registration form and wait for their form to be approved by the admin. They can also opt their type of driving class as either “Test driving class” or “Normal driving class”. They can choose whether they need a heavy driving class too in the registration form. The students once approved by the admin can login to the system and can view their information, class schedule and their progress.

1. System Boundaries:

This system contains some boundaries such as

* Some people don’t like to register through the software so they follow the old way of coming to the office.
* The system doesn’t allow the customers to pay online.
* Some elderly customers may not know use the online registration system.
* The system should be advertised if not it will not be known to many people.
* Many students may not have a computer system in their home which makes them contact the driving school for their registration.
* However, the system has more advantages, it requires more budget than the old traditional way.

1. **Functional Requirements**
2. Registration of New Student:

* FR1: Get the details of new students
* FR2: Check the age and type of class opted.

Fit Criterion: The student should be above 17 years to attend the class. If the student wants “Test Class”, then he/she should be above 18 years of age. If he/she want “Heavy Licence Class”, then they should be above 21.

* FR3: Get the students choice of time for classes.
* FR4: Save the form and profile.
* FR5: Delete the form and profile.
* FR6: Send the form for approval of admin.

1. Management of Student Application:

* FR7: Approve the form
* FR8: Decline the form
* FR9: Assign a driving instructor
* FR10: Inform the student, if the driving instructor is not free
* FR11: Put the student in waiting list, if he wants to continue
* FR12: Check for any other time, if he doesn’t want to wait
* FR13: Reject the application, if he doesn’t want to continue

1. Student Progress Management:

* FR14: Record the attendance
* FR15: Test the driving skills
* FR16: Record the progress by giving mark
* FR17: Attend more classes, if student progress is weak
* FR18: Allow to attend the License test

1. Fee Management:

* FR19: Inform fee balance
* FR20: Issue License card, if fee dues are paid

1. **Quality Requirements**
2. Simplicity:

* QR1: The system should be simple

Fit Criterion: The system should be simple and not much complicated. It should not take much time to register a new user. It should be easily understandable by the admin, trainers, and students.

1. Availability:

* QR2: The system should be available in all standard web browsers

Fit criteria: The system should be available on all web browsers like Internet Explorer, Safari, Chrome, etc.

1. Reliability:

* QR3: The system should be reliable.

Fit criteria: The system should produce and give the correct database for all the users. It should not show wrong information.

1. Security:

* QR4: The administrators can view all the details of the students and trainer which requires authentication.
* QR5: Trainers can view the details of the student under his training alone. He is restricted from seeing other students detail. He also requires authentication.
* QR6: Students can only view their profile. They also need login credentials to enter their profile.

1. **Stakeholders**

The main stakeholders of the system are the students, the staffs, the admin and the trainer.

1. Students:

The students are the main participants of the system. They are going to use the system to register themselves and to get an appointment from the driving school admin.

1. Staff:

Staffs are the people like receptionist, office staffs. They help the student to fill the registration form and appoint them to the driving instructors.

1. Admin:

Admin is the people who are running the driving school. They approve the application of the students.

1. Trainers:

The trainer takes the classes for the students and updates the progress of students to the system by testing the driving skill of the students.

1. **Use Case Diagram**

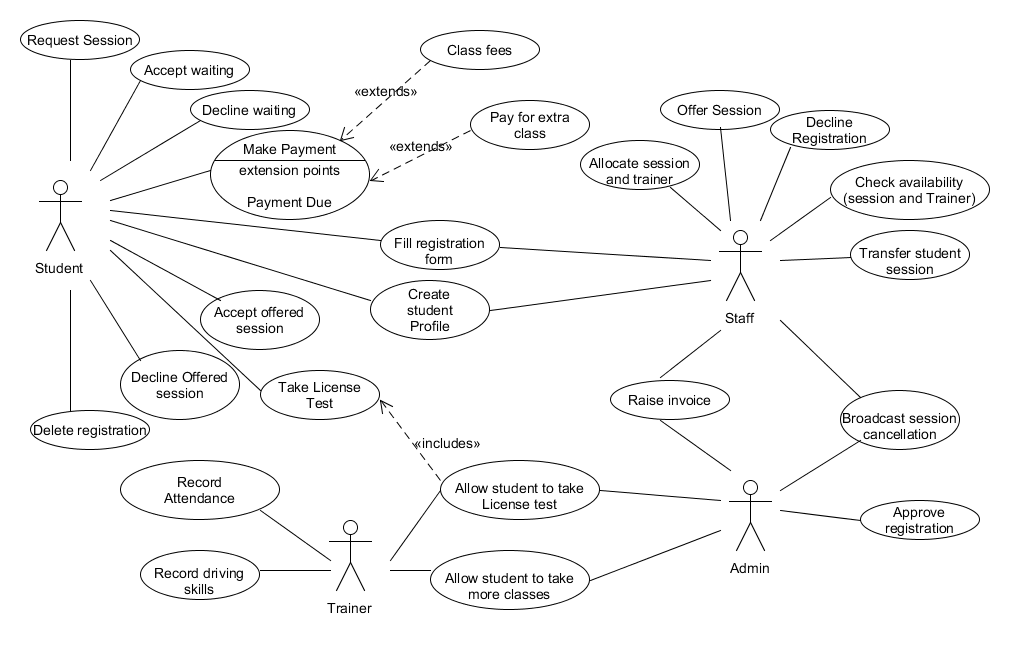


Figure 1: Use case diagram

In the above use-case diagram, the actor, the student registers his interest to the system and can create his profile. Also, if he makes a phone call or comes to the office, the staff creates a profile and register his interest. The student should be above 17 years of age to attend the classes. The student can request session through the registration form. If the requested session is full, then the staff puts him in waiting list and informs the student. The student can be in waiting list or can make a new request for another session. The staff gives an appointment if there is a vacancy or the staff may give some other session timing. If the student accepts the timing, the admin approves his application. Otherwise, the student can decline the application request.

Once the student was appointed to the class, his attendance is recorded by the trainer. The trainer also updates the test result of the student to the system so that the admin can see his progress and can decide whether he can attend the “License Test” or not. The administrator can allow the student to take more classes if he is not well trained. The student can take the “License Test” once he gets permission from the admin or the trainer. The admin informs the students whether there is a session on the day or not through the system.

1. **Prioritisation**

|  |  |  |
| --- | --- | --- |
| Prioritisation | Use Case | Comment |
| High | Create profile and fill registration form  Check availability of both session  and Trainer  Request another session  Approve registration  Accept waiting  Decline waiting  Offer Session  Accept offered session  Decline offered session  Transfer student session  Allow student to take more classes | These are the essential use cases which are the reason for the need of a new system. |
| Medium | Decline registration(admin)  Record Attendance  Record Driving Skills  Allow student to take License test | These use cases are not going to take much time for the users |
| Low | Delete Registration(Student)  Make payment  Raise invoice  Broadcast session cancellation | The payment mechanism is not given more priority in this case study |

Table 1: Prioritisation of use case

Rationale:

The use cases are categorized into high, medium and low priorities based on their need in the system. In the table below, the use cases which are listed on the high priority segment has a lot of functions to do and needs more concentration. Without these, there is no need to design the system. For example, creating the student profile is the most important thing for which the whole system is going to work. After creating the profile, the either student or admin fills the registration form. So, following this all the other functions are performed.

The medium priority listings are next to the major functions which is not going to take much effort and function. Declining registration form is done by the admin if he doesn’t want to register the student. Recording attendance and driving skills are done by the trainer, in which he is not going to save or send anything. He is just updating which is not highly prioritized. Payment mechanism and broadcasting session cancellation are given low priority as a payment mechanism are not concentrated in this version of the software.

1. **Use Case Templates**
2. Use Case Template 1:

|  |  |  |
| --- | --- | --- |
| Identifier and Name | | UC1: Create profile and fill registration form |
| Initiator | | Student or Staff |
| Goal | | To create a profile for student |
| Assumption | | none |
| Main Success Scenario: | | |
| 1 | The student opens the website or makes a call to the office staff | |
| 2 | The student enters his personal details or the staff enters the details of the student | |
| 3 | The student or the staff saves the profile | |
| 4 | The system asks the actor to fill the registration form | |
| 5 | The actor fills the registration form | |
| 6 | The system asks for student’s date of birth | |
| 7 | The actor enters his session preference | |
| 8 | The actor saves the registration form | |
| 9 | The system calculates the student’s age | |
| 10 | The system checks the type of class chosen | |
| 11 | The system validates the other details | |
| 12 | The form is saved for approval | |
| 13 | The system produces a success message | |
| Extensions: | | |
| 4a | Uploads related documents | |
| 9a | The system rejects the form if the age is below 17 | |

Table 2: Use case template 1 – Creating profile and registering a student

In this use case of creating a profile and filling the application form, the process is initiated by the student. The student can request his session but he can’t view the availability of session. The system also calculates his age by getting his date of birth. So, the student can’t register if he/she is not more than 17.

1. Use Case Template 2:

|  |  |  |
| --- | --- | --- |
| Identifier and Name | | UC2: Check availability of both session and trainer |
| Initiator | | Staff |
| Goal | | To check the availability of session and trainer |
| Assumption | | The student has opted for a session |
| Main Success Scenario: | | |
| 1 | The staff receives the registration form from student | |
| 2 | The staff checks whether there is an availability in the session which the student prefer | |
| 3 | Inform student if there is no availability | |
| 4 | Get reply from student | |
| 5 | Assign the session to the student | |
| 6 | The staff informs the student to pay the fees | |
| 7 | Update the student details to the trainer | |
| 8 | Send registration form to the admin for approval | |
| Extensions: | | |
| 3a | Suggest available sessions to the student | |
| 5a | Put the student in waiting list if he still wants the session in which there is no availability | |

Table 3: Use case template 2 – Checking and assigning session to the student

The staff assigns the trainer and session to the student. The staff informs the student if there is no availability in the requested session and give some suggestion. The staff put the student in waiting list if there is no availability. They also inform the student to pay the fees and update their information to the assigned trainer.

1. Use Case Template 3:

|  |  |  |
| --- | --- | --- |
| Identifier and Name | | UC3: Request another session |
| Initiator | | Student |
| Goal | | To change the session |
| Start and Stop point | | The existing student wants to change his session |
| Main Success Scenario: | | |
| 1 | The student makes a request to the staff | |
| 2 | The staff contacts trainer and check for vacancies in any other session | |
| 3 | Staff gets reply from trainer | |
| 4 | Give suggestion of available session to the student | |
| 5 | Get the reply from the student | |
| 6 | Assign the session to the student | |
| Extensions: | | |
| 6a | Ask him to continue the same session if there is no availability | |
| 6b | Put him on waiting list if he still wants to change the session | |

Table 4: Use case template 3 – Requesting another session by student

In this case, the student learning class needs a transfer of session. So, he makes a request to the admin of the driving school. The staff looks for any other vacancy in some other session. Assigns him new session if there is a vacancy. Otherwise put him back in the old session. But he will be given chance before the new students on the waiting list.

1. **Key Business Process**

* Registering new student
* Assigning session and trainer
* Tracking the student’s progress
* Maintaining student’s records

1. Registering New User:

The student can register himself or can call the staff to register. The student should enter his valid details to the system. The system calculates the age of the student by getting his birth date. It allows him to register if and only if he is above 17. The student can request for the session he wishes for in the registration form. He can create his own profile which helps him to track his records.

Key process:

* Creating profile
* Registering student

Rules:

* The student should be above 17.
* The student should have valid certificates.

1. Assigning session and trainer:

The staff and the admin assign session and trainer for the student. The staff checks for the vacancy in the session which the student asked for. If there is a vacancy, then the staff sends the application of student to admin for confirmation. The staff suggests the student about another session which has space if the session requested by the student is full. If the student is okay with the suggested session, then the staff forward the application. If the student wants the same session he asked, then the staff put him in waiting list. If the prospective student needs a change in session, then the staff should repeat the same process.

Key Process:

* Check availability of session and trainer
* Assign session to the student
* Put student in waiting list if the session is full

1. Tracking the Student Progress:

Once the student is assigned to the trainer, the trainer starts monitoring the student. He should take the driving classes for a period of 35 days. The trainer first notes his attendance. He also monitors his driving skills. On the end of 35 days, the trainer assesses the student by keeping some tests and update his scores to the system. The admin analyses the progress of the student and allows him to take the License test. If the progress of the student is not up to the mark, then the admin asks the student to take extra classes.

Key Process:

* Record student activities
* Analyze student progress

Rules:

* Student should attend 35-day class
* Student should pass the driving test

1. **Activity Diagram**

Process P1: Registering new student

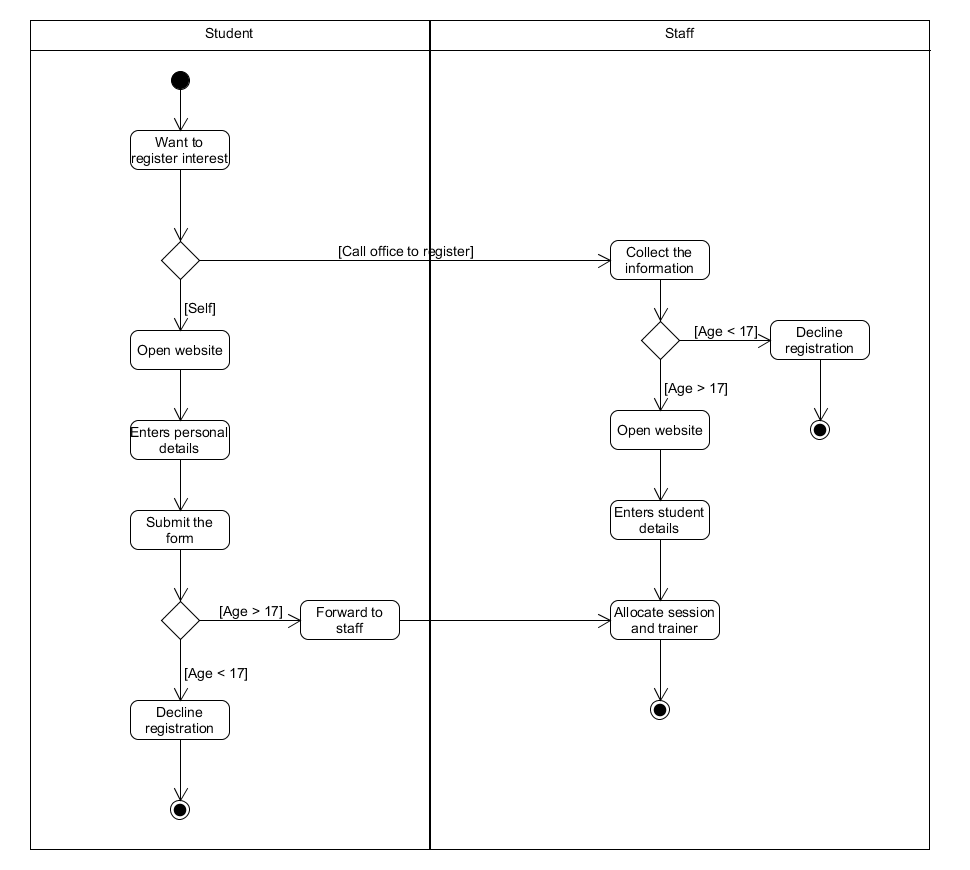


Figure 2: Activity diagram 1 – Registering new student

The student is the initiator of this process. On the left side, the student register by himself. The student opens the web page and enter his details. He just submits the form for approval. The system checks his age and forwards the form to the staff for verification if he is above 17. Decline the request if he is below 17.

On the right, the student makes a call to the office. The staffs in the office help the student to register. The staff gets the details of the student through phone and enters the details into the form on the web page if the student is above 17 else the request is declined. The form is forwarded to the next process.

Process P2: Assigning session and trainer

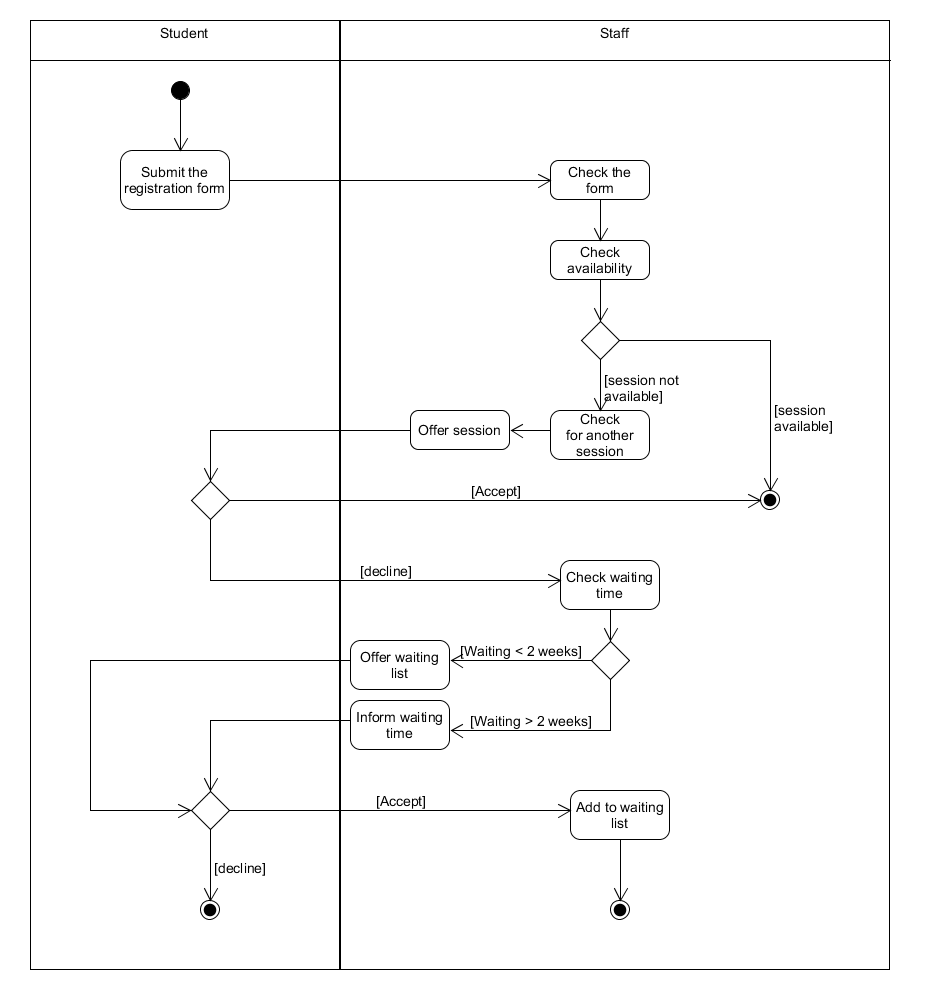


Figure 3: Activity diagram 2 – Assigning session and trainer

The student submits the form to the system. The staff verifies the information entered by the student and check the availability of session the student opted. If there is no availability, the staff suggest another session. If the student accepts the offered session, then his form is sent for approval. The student needs the same session he requested and if there is no vacancy, put the student on the waiting list. The student can stay if he accepts the waiting or he/she can decline the registration.

Process P3: Tracking student progress

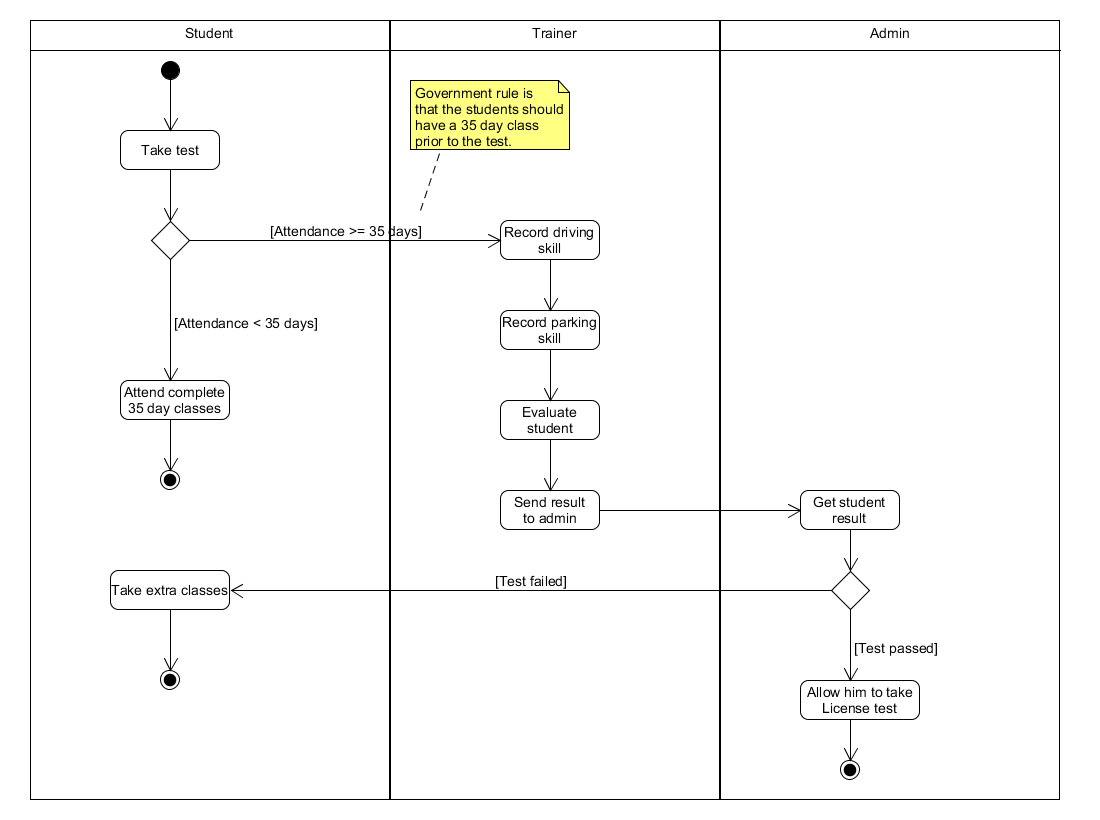


Figure 4: Activity diagram 3 – Tracking student progress

The student wants to take the test. The trainer checks his attendance and allows him to take the test if his presence is more than 35 days. The trainer watches and records his driving skills and evaluate. The trainer once evaluated the student, send the result to the admin. The admin allows the student to take License test if his skills are good. Else, the student is asked to take extra classes.

1. **Acceptance Test**
2. Use case: Create profile and fill registration form
3. Test UC1-T1: Test that, when the student chooses create profile, the system prompts for student name, age, and sex.
4. Test UC1-T2: Test that, when student saves his profile, the system prompts to fill the registration form
5. Test UC1-T3: Test that, when the student submits the form, the system displays error message if the age of the student is less than 7.
6. Test UC1-T4: Test that, when the student submits the form successfully, the system displays a success message
7. Use case: Check availability of both session and Trainer
8. Test UC2-T1: Test that, when student submit the registration form, the system prompts for session preference.
9. Test UC2-T2: Test that, when the staff checks for available session, the system displays the available spaces
10. Test UC2-T3: Test that, when the staff selects the session, the system prompts for confirmation
11. Test UC2-T4: Test that, when staff confirm the session, the system assigns the session to the student.
12. Use case: Request another session
13. Test UC3-T1: Test that, when the student request for another session, the system prompts for his preference.
14. Test UC3-T2: Test that, when student submits his request, the system prompts for confirmation.
15. Test UC3-T3: Test that, when student confirm his selection, the system displays a success message
16. Use case: Approve and decline registration
17. Test UC4-T1: Test that, when the admin selects a student detail, the system shows all the detail of the student
18. Test UC4-T2: Test that, when the admin approves the student registration, the system sends a message to the student.
19. Test UC4-T3: Test that, when the admin rejects the registration, the system sends a message to the student about rejection
20. Use case: Accept and decline waiting
21. Test UC5-T1: Test that, when the student accepts or decline the waiting, the system prompts for a confirmation.
22. Test UC5-T2: Test that, when the student confirms, the system produces a success message and send the student status to the staff.
23. Use case: Accept offered session
24. Test UC6-T1: Test that, when the student accepts the offered session, the system prompts for a confirmation.
25. Test UC6-T2: Test that, when the student confirms his selection, the system displays a success message.
26. Use case: Decline offered session
27. Test UC7-T1: Test that, when the student declines the offered session, the system prompts for a confirmation.
28. Test UC7-T2: Test that, when the student confirms his selection, the system prompts for his preference.
29. Test UC7-T3: Test that, when the student enters his preference, the system displays success message.
30. Use case: Transfer student session
31. Test UC8-T1: Test that, when the student request for transferring session, the system prompts for a reason.
32. Test UC8-T2: Test that, when the student submits his request, the system displays a success message.
33. Use case: Raise Invoice
34. Test UC9-T1: Test that, when the admin raises an invoice, the system prompts for quote.
35. Test UC9-T2: Test that, when the admin submits his request, the system displays a success message and update the student

10. Use case: Broadcast Session Cancellation

1. Test UC10-T1: Test that, when the admin broadcasts a message, the system prompts for confirmation
2. Test UC10-T2: Test that, when the admin confirms, the system produces a success message.