ERP

The Conservation of The Holy Quran Society

Khaled Tayem

khaled Al-kteefan

Ayman Al-Hariri

Nisreen Al-Motlaq

School of Computer Engineering and

Information Technology

German Jordanian University

Amman, Jordan

Email: [k.tayem@gju.edu.com](mailto:k.tayem@gju.edu.com)

**Abstract**—the abstract goes here.

**KEYWORDS**

Registration system.

1. **INTRODUCTION**

The Conservation of The Holy Quran Society is a society concerned in teaching The Holy Quran different aspects like tajweed and tafseer the holy Quran for all ages.

The society was established in 1991 under the supervision of the Ministry of Awqaf, Islamic Affairs and Holy Sites. The General Administration is located in Amman, The Society has forty-two branches, and the branches also have 1050 centers, these centers are distributed all over Jordan.

For the first glance it shows that there is lack of information to the management for better decision making, which effects the performance of the centers, the centers must have a system to help for fair distribution to resources which are mainly donations.

The student progress should be noted and saved for future review, fair Distribution of students to the Kingdom centers, I've seen some centers have more students that it can handle and other centers have few students.

1. **STATEMENT OF PROBLEM**

The society has a unique management system, it has a lot of similarity to a school system but it's unique in some aspects.

* First the society is a nonprofit organization, there is small fees taken from student but it's not mandatory.
* Second enrollment to classes depends on student age and the amount of memorization of the holy Quran.
* Third the society usually works on Saturdays and after 3:00 pm from Sunday to Thursday.
* Fourth most of the teachers are Volunteer's and some get paid a small fees for transportation and any expenses needed for working in the society.
* Fifth every center has a manager and accountant with full salary part time job.

The main problem of the society that there is no computerized system, our goal is to establish an ERP system for the Society that includes:

* Registration system for the students.
* Transportation system.
* Teachers system.
* Payments system.
* Donations system.

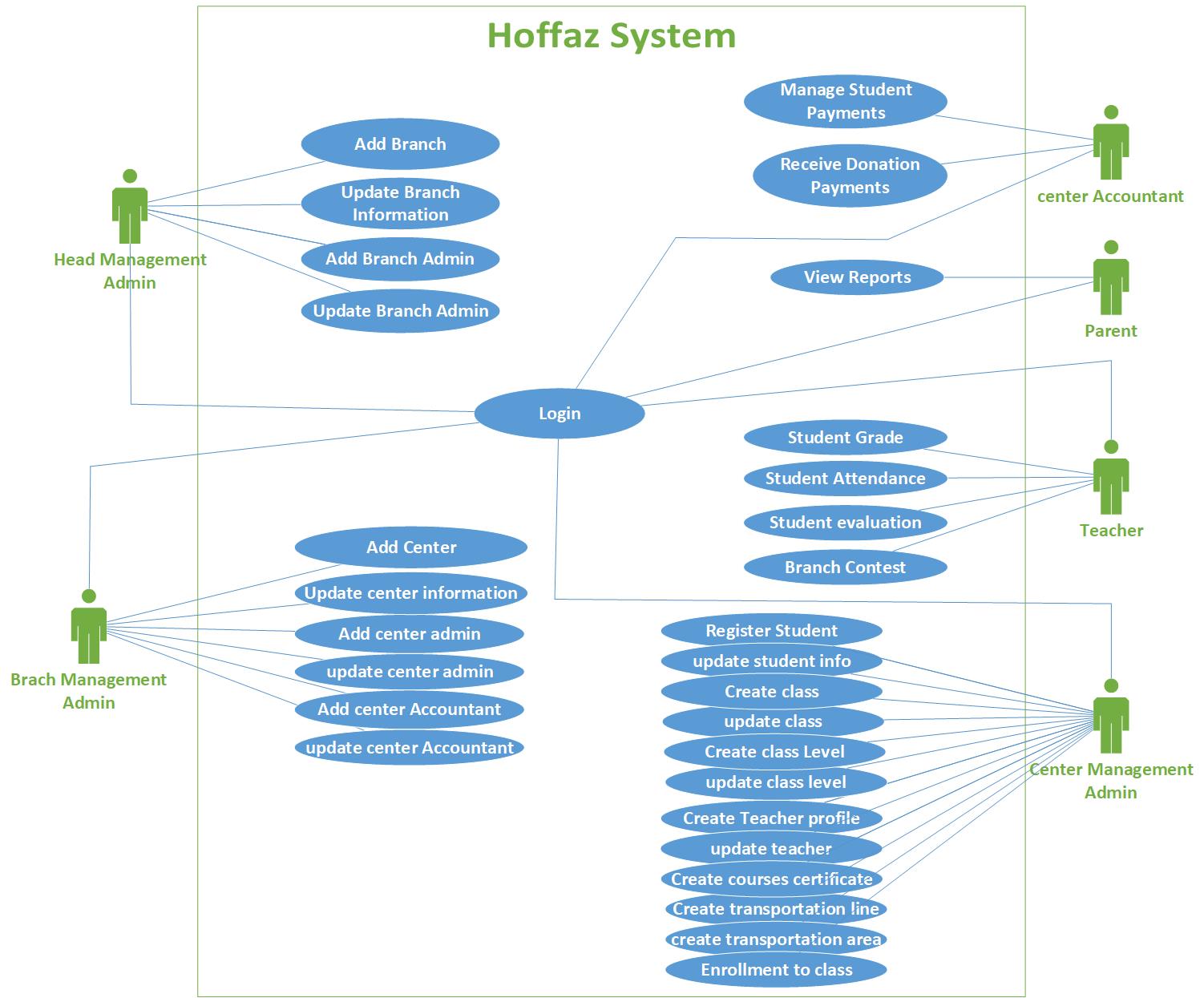
1. **SYSTEM REQUIREMENTS**

The Conservation of The Holy Quran Society has three levels of administrations as following:

* Head level managed by the Head admin
* Branch level managed by the Branch admin
* Center level managed by the center admin

The functional requirements as supplied have been reorganized as follow Divided by actor:

1. Head management actor: the highest management where he's Responsible for branch management.
   * Add branch.
   * Update branch information.
   * Add branch admin.
   * Update branch admin information.
2. Branch management actor: responsible for center management.
   * Add center.
   * Update center information.
   * Add center admin.
   * Update center admin information.
3. Center management actor: The person responsible for student registration, course creation, create teacher profile.
   * Register a student.
   * Update student information.
   * Create class (student age is the criteria for enrollment).
   * Update class.
   * Create class level (student amount of memorization of Quran is the criteria for enrollment).
   * Update class level.
   * Create courses and certificates.
   * Create transportation line.
   * Record student attendance.
   * Student enrollment to class.
4. Center accountant actor: manage and insert all incomes from students and donors.
   * Manage student payments.
   * Receive donations.
5. Teacher actor: responsible for student grading and evaluation, student attendance and nominate for the branch contest.
   * Student grades.
   * Student attendance.
   * Student evaluation.
   * Branch contest.
6. Parent: person who follows the progress of the student.
   * View reports.
7. **USE CASE DIAGRAM AND DESCRIPTION**

****

|  |  |
| --- | --- |
| **Use case Name** | **Login use case** |
| **Actors** | All actors |
| **Pre-condition** |  |
| **Main flow** | * enter user name and password * submit by press add button * Validate user name & password |
| **Post condition** | Display form depend on the actor type |
| **Alternatives** | * invalid user name or password   + stay in the same screen and notify the actor to the error   + save failed attempt for login (user, terminal IP, date) |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **View reports** |
| **Actors** | Parents |
| **Pre-condition** | Parent login |
| **Main flow** | * Select /search student * submit by press select button |
| **Post condition** | Display student info |
| **Alternatives** | * student not found   + stay in the same screen and notify the actor |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **Add Branch** |
| **Actors** | Head Admin |
| **Pre-condition** | Head Admin login |
| **Main flow** | * Select Add Branch * Add the information about the branch(Name, Location, Phone) * submit by press add button * Validate All data entered * Validate Branch name not exist |
| **Post condition** | * Add branch to the database * Display message insure adding branch to the system |
| **Alternatives** | * missing data when validate   + stay in the same screen and notify the actor to the missing data * Branch name exist when validate   + stay in the same screen and notify the actor to the existed name |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **update Branch** |
| **Actors** | Head Admin |
| **Pre-condition** | * Head Admin login * Exist branch in data base |
| **Main flow** | * Select update Branch info by Actor * select/search a branch to update * show the info of branch * edit the info by Actor * submit by press update button * validate all data entered * Validate branch name not exist. |
| **Post condition** | * Update branch in the database * Save old values for auditing. * Display message insure the info updated |
| **Alternatives** | * missing data   + stay in the same screen and notify the actor to the missing data * Branch name exist   + stay in the same screen and notify the actors |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **Add Branch Admin** |
| **Actors** | Head Admin |
| **Pre-condition** | * Head Admin login * Exist branches |
| **Main flow** | * Select add Branch admin * add the information about the admin (first name, Last name, second name, third name, phone, address, branch name, nationality number, WhatsApp number, work-status) * submit by press add button * validate the following data not empty( first name, Last name, second name, third name, phone, address, branch name, nationality number, WhatsApp number, work-status) * validate nationality number not exist |
| **Post condition** | * Add branch in the database * Display message insure adding admin |
| **Alternatives** | * missing data   + stay in the same screen and notify the actor to the missing data * Branch nationality number exist   + notify the actors |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **update Branch Admin** |
| **Actors** | Head Admin |
| **Pre-condition** | Head Admin login |
| **Main flow** | * Select update Branch Admin by Actor * select/search a branch Admin to update * show the info of branch Admin * edit the info by Actor * submit by press update button * validate the following data not empty( first name, Last name, second name, third name, phone, address, branch name, nationality number, WhatsApp number, work-status) * validate nationality number not exist |
| **Post condition** | * Add branch in the database * Save old values for auditing. * Display message insure adding admin |
| **Alternatives** | * missing data   + stay in the same screen and notify the actor to the missing data * Branch nationality number exist   + notify the actors |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **Add center** |
| **Actors** | Branch Admin |
| **Pre-condition** | * Head Admin login * Exist branches |
| **Main flow** | * Select add center * add the information about the center (name, location ,phone, branch name) * validate the following data All data entered and center name not exist |
| **Post condition** | * Add center in the database * Display message insure adding admin |
| **Alternatives** | * missing data   + stay in the same screen and notify the actor to the missing data * center name exist   + stay in the same screen and notify the actors to the existed name |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **update center info** |
| **Actors** | Branch Admin |
| **Pre-condition** | branch Admin login  Exist center in the database |
| **Main flow** | * Select update center information by actors * select/search a center to update * show the info by actors * show the info of the center * edit the info by actors * press update button by actor * validate all data entered * validate center name not exist |
| **Post condition** | * Update center in the database * Save old values for auditing. * Display message insure the update center done |
| **Alternatives** | * missing data   + stay in the same screen and notify the actor to the missing data * center name exist   + stay in the same screen and notify the actors to the existed name |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **Add Center Admin** |
| **Actors** | Branch Admin |
| **Pre-condition** | * Branch Admin login * Exist Centers |
| **Main flow** | * Select add Center admin * add the information about the admin (first name, Last name, second name, third name, phone, address, center name, nationality number, WhatsApp number, work-status) * submit by press add button * validate the following data not empty( first name, Last name, second name, third name phone, address, center name, nationality number, WhatsApp number, work-status) * validate nationality number not exist |
| **Post condition** | * Add Center Admin to the database * Display message insure adding admin |
| **Alternatives** | * missing data   + stay in the same screen and notify the actor to the missing data * nationality number exist   + notify the actors |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | update Center Admin |
| **Actors** | Branch Admin |
| **Pre-condition** | Branch Admin login |
| **Main flow** | * Select update Center Admin by Actor * select/search a Center Admin to update * show the info of Center Admin * edit the info by Actor * submit by press update button * validate the following data not empty( first name, Last name, second name, third name, phone, address, Center name, nationality number, WhatsApp number, work-status) * validate nationality number not exist |
| **Post condition** | * Update Center Admin info in the database * Save old values for auditing. * Display message insure updating admin |
| **Alternatives** | * missing data   + stay in the same screen and notify the actor to the missing data * nationality number exist   + notify the actors |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **Add Center Accountant** |
| **Actors** | Branch Admin |
| **Pre-condition** | * Branch Admin login * Exist Centers |
| **Main flow** | * Select add Center Accountant * add the information about the admin (first name, Last name, second name, third name, phone, address, center name, nationality number, WhatsApp number, work-status) * submit by press add button * validate the following data not empty( first name, Last name, second name, third name, phone, address, center name, nationality number, WhatsApp number, work-status) * validate nationality number not exist |
| **Post condition** | * Add Center Accountant to the database * Display message insure adding admin |
| **Alternatives** | * missing data   + stay in the same screen and notify the actor to the missing data * nationality number exist   + notify the actors |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **update Center Accountant** |
| **Actors** | Branch Admin |
| **Pre-condition** | Branch Admin login |
| **Main flow** | * Select update Center Accountant by Actor * select/search a Center Accountant to update * show the info of Center Accountant * edit the info by Actor * submit by press update button * validate the following data not empty( first name, Last name, second name, third name, phone, address, Center name, nationality number, WhatsApp number, work-status) * validate nationality number not exist |
| **Post condition** | * Update Center Accountant info in the database * Save old values for auditing. * Display message insure updating admin |
| **Alternatives** | * missing data   + stay in the same screen and notify the actor to the missing data * nationality number exist   + notify the actors |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **Manage student payment** |
| **Actors** | Center Accountant |
| **Pre-condition** | Center Accountant login |
| **Main flow** | * Select student payment by Center Accountant * select /search student * enter the amount of payment * submit by press add button * validate the amount |
| **Post condition** | * Add payment to the database * Display message insure payment |
| **Alternatives** | * invalid amount   + stay in the same screen and notify the actor to the error |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case Name** | **Receive Donations Payments** |
| **Actors** | Center Accountant |
| **Pre-condition** | Center Accountant login |
| **Main flow** | * Select Donations by Center Accountant * enter the amount of Donation * submit by press add button * validate the amount |
| **Post condition** | * Add Donation to the database * Display message insure payment |
| **Alternatives** | * invalid amount   + stay in the same screen and notify the actor to the error |
| **Exceptions** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Register Student** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. * Class created. * Teacher exists. |
| **Main Flow** | 1. Select Register new student form. 2. Add student information (First name, Second name, third name, last name, birthdate, nationality, nationality number, phone, what sup, address, transportation, trip name, stop name, class name, year, semester) 3. Submit the form by actor. 4. Validate data not empty First name, Second name, third name, last name, birthdate, nationality, nationality number, phone, address, transportation, class name. 5. validate nationality number not exist |
| **Post condition** | * Add student to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Student national number exist   + notify the actors * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Update Student information** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. |
| **Main Flow** | 1. Select update student form. 2. update student information (phone, WhatsApp, address, transportation, trip name, stop name) 3. Submit the form by actor. 4. Validate data not empty First name, Second name, third name, last name, birthdate, nationality, nationality number, phone, address, transportation. |
| **Post condition** | * Add student to the Database. * Save old values for auditing. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Create class** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. |
| **Main Flow** | 1. Select create class form. 2. Add class information (class name, working days, age range, Description). 3. Submit the form by actor. 4. Validate data not empty class name, working days, age range, Description. 5. Validate class name not exist. |
| **Post condition** | * Add class to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Class name exist.   + Stay in the same screen and notify the actor that class name exist. * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Update class** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. |
| **Main Flow** | 1. Select update class form. 2. update class information(working days, age range) 3. Submit the form by actor. 4. Validate data not empty age range, Description. |
| **Post condition** | * Update class to the Database. * Save old values for auditing. * Display message successful updated transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Create class level** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. * Class created. |
| **Main Flow** | 1. Select create class level form. 2. Add class level information(class name, class level, Description) 3. Submit the form by actor. 4. Validate data not empty class name, class level, Description. 5. Validate class level not exist. |
| **Post condition** | * Add class to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Class level exist.   + Stay in the same screen and notify the actor that class level exist. * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **update class level** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. * Class created. |
| **Main Flow** | 1. Select update class level form. 2. update class level information(Description) |
| **Post condition** | * Add class to the Database. * Save old values for auditing. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Enrollment to class** |
| **Actors** | Teacher |
| **Pre-condition** | * Center admin login. * Class created. * Teacher exists. |
| **Main Flow** | 1. Select Register new student form. 2. Select student. 3. Add student class (class name, year, semester) 4. Submit the form by actor. 5. Validate data not empty First name, Second name, third name, last name, birthdate, nationality, nationality number, phone, address, transportation, class name. |
| **Post condition** | * Save Enroll student to the class transaction to Database. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Create teacher profile** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. * Create courses and certificate. |
| **Main Flow** | 1. Select create teacher profile form. 2. Add teacher information (First name, Second name, third name, last name, birthdate, nationality, nationality number, phone, WhatsApp, address, The amount of memorization of the Koran, courses and certificates, date of issue, certificate number ) 3. Submit the form by actor. 4. Validate data not empty First name, Second name, third name, last name, birthdate, nationality, nationality number, phone, address, the amount of memorization of the Koran, courses and certificates. 5. validate nationality number not exist |
| **Post condition** | * Add teacher profile to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Student national number exist   + notify the actors * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **update teacher profile** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. * Create courses and certificate. |
| **Main Flow** | 1. Select update teacher profile form. 2. update teacher information (phone, WhatsApp, address, The amount of memorization of the Koran, courses and certificates ) 3. Submit the form by actor. 4. Validate data not empty First name, Second name, third name, last name, birthdate, nationality, nationality number, phone, address, the amount of memorization of the Koran, courses and certificates. |
| **Post condition** | * Update teacher profile to the Database. * Save old values for auditing. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Create courses and certificates** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. |
| **Main Flow** | 1. Select Create courses certificate form. 2. Add courses or certificate information (name, place of issue) 3. Submit the form by actor. 4. Validate data not empty name, place of issue. |
| **Post condition** | * Insert courses and certificates to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Create transportation line** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. |
| **Main Flow** | 1. Select create transportation line. 2. Add transportation line information (name, Description) 3. Submit the form by actor. 4. Validate data not empty name, Description. |
| **Post condition** | * Add transportation line to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Create transportation area** |
| **Actors** | Center Management Admin |
| **Pre-condition** | * Center admin login. * Transportation line created. |
| **Main Flow** | 1. Select create transportation area form. 2. Add transportation line information (name, Description, Transportation line) 3. Submit the form by actor. 4. Validate data not empty name, Description, Transportation line. |
| **Post condition** | * Add transportation line to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Student Grade** |
| **Actors** | Teacher |
| **Pre-condition** | * Teacher login. * Enrolment to class. |
| **Main Flow** | 1. Select Student Grade form. 2. Add Student Grade. 3. Submit the form by actor. 4. Validate data not empty grade. |
| **Post condition** | * Add student grade line to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Student attendance** |
| **Actors** | Teacher |
| **Pre-condition** | * Teacher login. * Enrolment to class. |
| **Main Flow** | 1. Select Student attendance form. 2. Select class date. 3. Check for student absence 4. Submit the form by actor. 5. Validate data not empty date. |
| **Post condition** | * Add student attendance to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Student evaluation** |
| **Actors** | Teacher |
| **Pre-condition** | * Teacher login. * Enrolment to class. |
| **Main Flow** | 1. Select Student evaluation form. 2. Add Student evaluation for (). 3. Evaluation (Excellent, very good, good, Average, poor) 4. Submit the form by actor. 5. Validate data not empty grade. |
| **Post condition** | * Add student grade line to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |

|  |  |
| --- | --- |
| **Use case name** | **Branch Contest** |
| **Actors** | Teacher |
| **Pre-condition** | * Teacher login. * Enrolment to class. |
| **Main Flow** | 1. Select Branch Contest form. 2. Select student eligible for contest. 3. Select contest type (5, 10, 15, 20, 25, and 30) versus of the holy Quran. 4. Select date of contest. 5. Enter the contest grade. 6. Submit the form by actor. 7. Validate data not empty date. |
| **Post condition** | * Add student attendance to the Database. * Display message successful insert transaction. |
| **Alternatives** | * Missing data when validate.   + Stay in the same screen and notify the actor with the missing data. |
| **Exception** | Uncompleted use case show error message. |