



**MILITARY TECHNICAL ACADEMY  
FACULTY OF INFORMATION TECHNOLOGY**

**DORMITORY MANAGEMENT SOFTWARE  
SOFTWARE REQUIREMENTS SPECIFICATION**

**Name: Nguyen Thi Trinh  
Class : ATBMTT-K50**

**Hà Nội, 11/2017**

## CATALOG

<b>I. Introduction .....</b>	<b>4</b>
<b>1. Purpose.....</b>	<b>4</b>
<b>2. Scope of Project.....</b>	<b>4</b>
<b>3. Glossary .....</b>	<b>4</b>
<b>4. References.....</b>	<b>4</b>
<b>5. Overview of Document .....</b>	<b>4</b>
<b>II. The Overall Description .....</b>	<b>5</b>
<b>III. Requirements Specification.....</b>	<b>16</b>
<b>1. Log in .....</b>	<b>16</b>
<b>2. Register at the dormitory.....</b>	<b>17</b>
<b>3. Extension Contract.....</b>	<b>18</b>
<b>4. Add student .....</b>	<b>19</b>
<b>5. Edit Student.....</b>	<b>19</b>
<b>6. Delete Student .....</b>	<b>20</b>
<b>7. Add Staff.....</b>	<b>20</b>
<b>8. Edit Staff.....</b>	<b>21</b>
<b>9. Delete Staff .....</b>	<b>22</b>
<b>10. Add Room.....</b>	<b>22</b>
<b>11. Edit Room.....</b>	<b>23</b>
<b>12. Delete Room .....</b>	<b>23</b>
<b>13. Add Dormitory .....</b>	<b>24</b>
<b>14. Edit Dormitory .....</b>	<b>24</b>
<b>15. Add Class.....</b>	<b>25</b>
<b>16. Edit Class.....</b>	<b>26</b>
<b>17. Add Faculty .....</b>	<b>26</b>
<b>18. Edit Faculty .....</b>	<b>27</b>
<b>19. Add Assessment of room.....</b>	<b>28</b>
<b>20. Add Compliment of Student.....</b>	<b>28</b>

**21. Add Discipline of student.....29**

**22. Add Bill.....29**

**23. Add Calendar.....30**

## **I. Introduction**

### **1. Purpose**

This Dormitory Management software requires new specification with the main objective is to provide a base for the foundation of the project. It will explain the purposes and features of the system, the interfaces of the system, what the system will do, the constraints under which it will be operated and how the system will react to external stimuli. Client's expectations are analyzed to produce specific unambiguous functions so that they can be used by development team with clear understanding to build a system as per user's need.

### **2. Scope of Project**

This software will be a Dormitory Management software for a university or a high school.

### **3. Glossary**

<b>Term</b>	<b>Definition</b>
<i>SRS</i>	<i>Software Requirements Specification</i>
DMS	Dormitory Management Software
User	Staff and student
Database	The existing membership database

## **4. References**

### **5. Overview of Document**

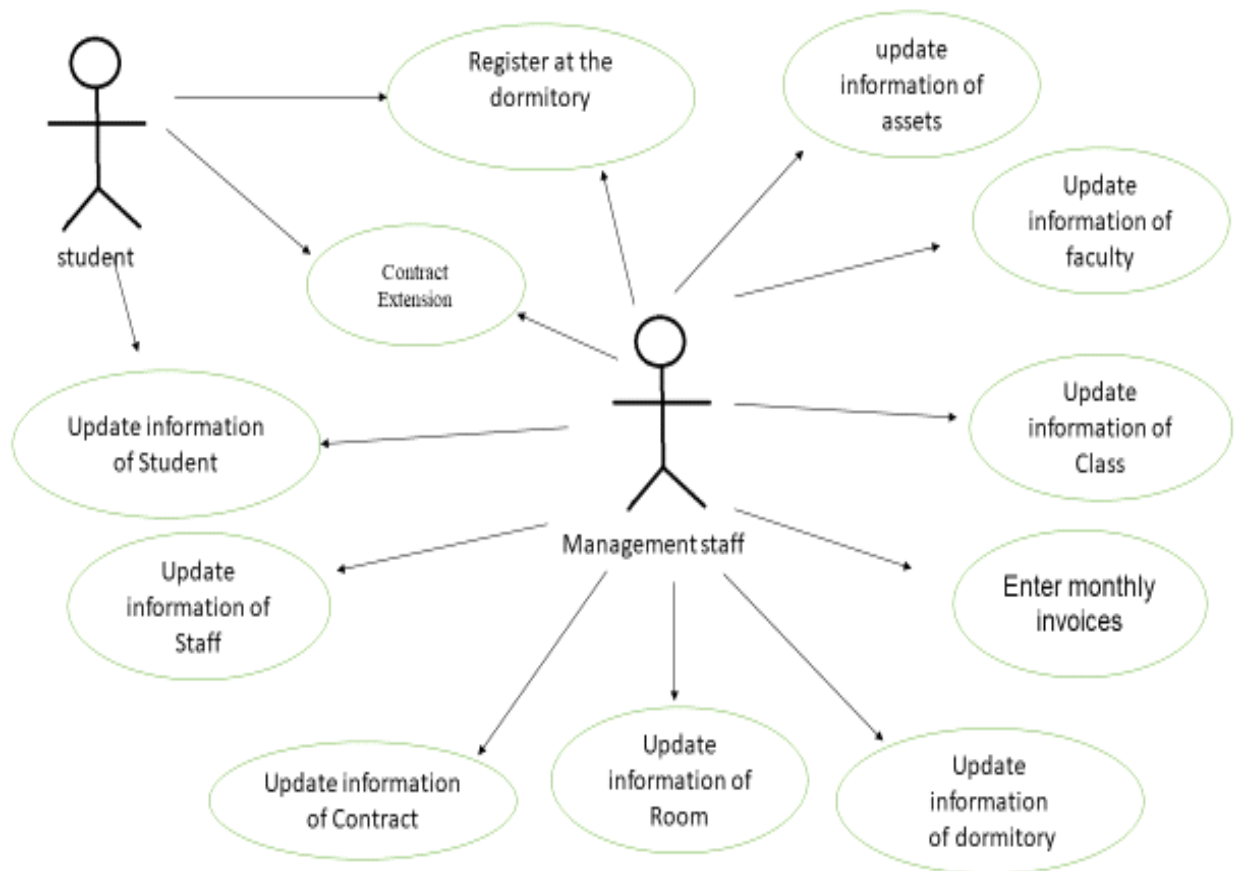
The next chapter, Overall Description section, of this document will give you an overview about the functionality of the product. It describes the informal requirements and is used to establish a context for the specific technical requirements in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes the details of the functionality of the product in technical terms .

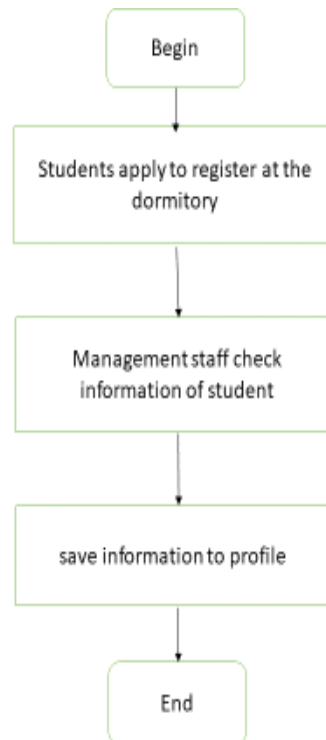
Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different languages.

## II. The Overall Description

### Use case



- **Use case : Register at the dormitory**
  - **Event trigger:** Students wishing to enter the dormitory must write a registration form.
  - **Business process model**



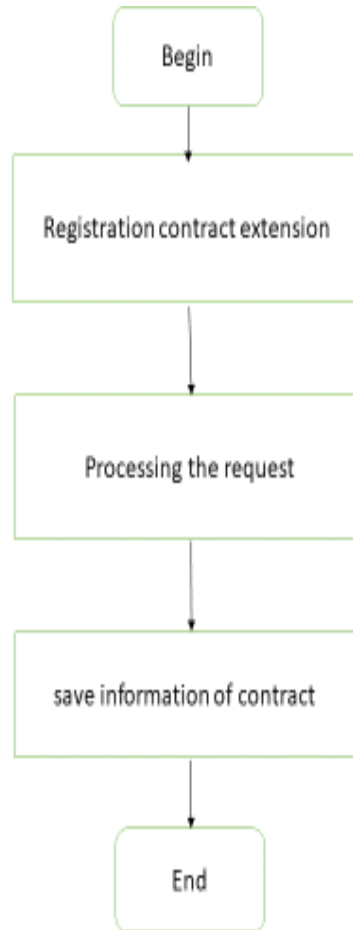
- **Description:**

Event code	Event	Actor	Content
(1)	Students apply to register at the dormitory	Student	Students write and submit applications in the dorm to management staff
(2)	Management staff check information of student	Management staff	The management staff confirms the student's registration information
(3)	Save information to profile	Management staff	Keep the student's application file for review

- **Use case: Contract extension**

- **Event trigger:** When the contract expires, the student wishes to continue to stay in the dormitory.

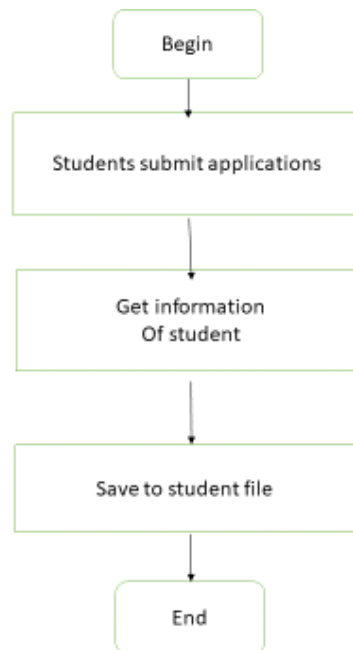
- **Business process model**



- **Description:**

Event code	Event	Actor	Content
(1)	Registration contract extension	Student	Students write and submit a contract extension
(2)	Processing the request	Management staff	Management staff handle the requests of the students
(3)	Save information of contract	Management staff	Store new contract information on student documents

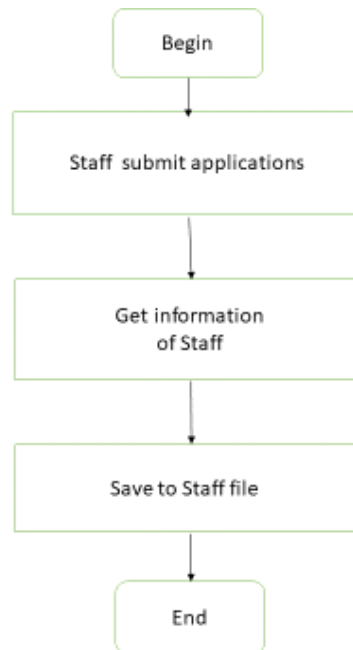
- **Use case: Update information of Student**
  - **Event trigger:** When students submit applications to the school or when there is a change of information, the administrator will have to update the student file.
  - **Business process model**



- **Description:**
  - + Step 1: Get information of student
    - Input: Student documents.
    - Output: Information of students need to be stored.
  - + Step 2: Update to student file
    - Input: Information of students need to be stored.
    - Output: student file.
- **Use case: Update information of staff**



- **Event trigger** when there are new employees or when there is a change in student's information, Management Staff will update the information of staff.
- **Business process model**



- **Description:**

- + Step 1: Get information of staff

- Input: Student documents.

- Output: Information of students need to be stored

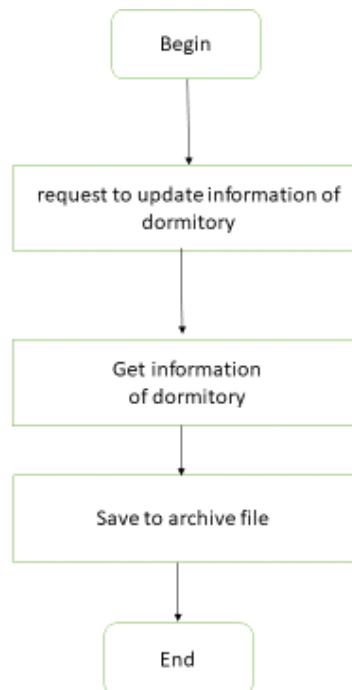
- + Step 2: Update to staff file

- Input: Information of students need to be stored

- Output: student file

- **Use case: Update information of dormitory**

- **Event trigger:** When there is a change in information of the dormitory, management Staff will update the information of dormitory.
- **Business process model**



- **Description:**

- + Step 1: Get information of dormitory

- Input: A request to enter information of dormitory

- Output: Information of dormitory need to be stored.

- + Step 2: Update to staff file

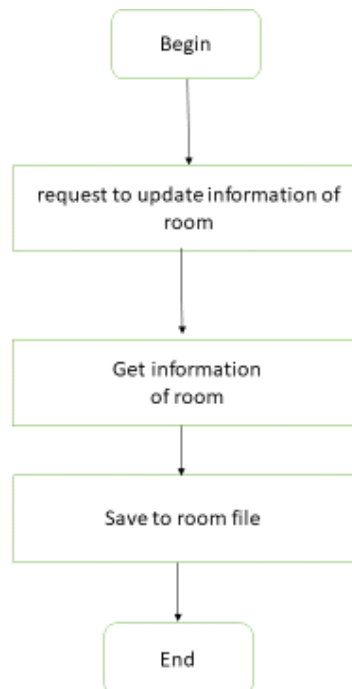
- Input: Information of dormitory need to be stored

- Output: Dormitory file

- **Use case: Update information of room**

- **Event trigger:** When there is a change in information of the rooms, management Staff will update the information of rooms.

- **Business process model**



- **Description:**

- + Step 1: Get information of room

- Input: A request to enter information of room.

- Output: Information of room need to be stored.

- + Step 2: Update to staff file

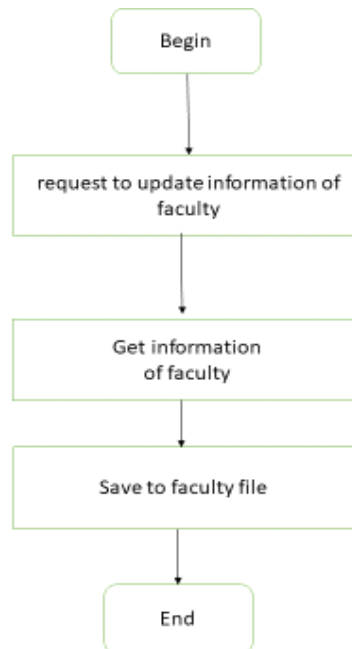
- Input: Information of room need to be stored.

- Output: Room file.

- **Use case: Update information of faculty**

- **Event trigger:** When there is a change in information of the faculty, management Staff will update the information of faculty.

- **Business process model**



- **Description:**

- + Step 1: Get information of faculty

- Input: A request to enter information of faculty.

- Output: Information of faculty need to be stored.

- + Step 2: Update to faculty file

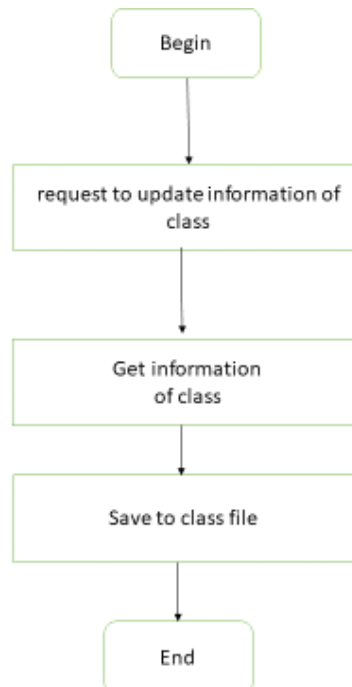
- Input: Information of faculty need to be stored.

- Output: Faculty file.

- **Use case: Update information of class**

- **Event trigger:** When there is a change in information of the class, management Staff will update the information of faculty.

- **Business process model**



- **Description:**

- + Step 1: Get information of class

- Input: A request to enter information of class.

- Output: Information of class need to be stored.

- + Step 2: Update to class file

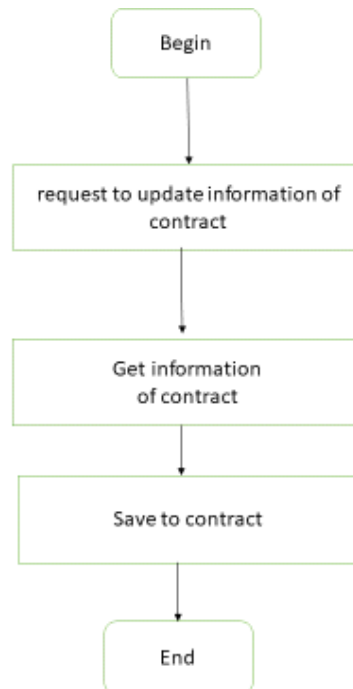
- Input: Information of class need to be stored.

- Output: Class file.

- **Use case: Update information of contract**

- **Event trigger:** when students are in school or when there is a change in information of the class, management Staff will update the information of faculty.

- **Business process model**



- **Description:**

- + Step 1: Get information of contract

- Input: A request to enter information of contract

- Output: Information of contract need to be stored.

- + Step 2: Update to contract file

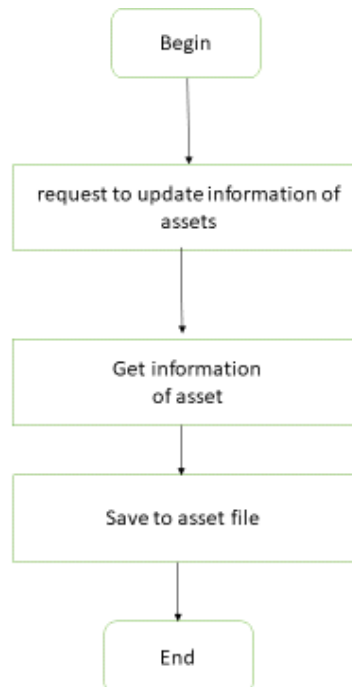
- Input: Information of contract need to be stored.

- Output: contract file.

- **Use case: Update information of asset**

- **Event trigger:** When the material is damaged or newly added to the common property.

- **Business process model**



- **Description:**

- + Step 1: Get information of asset

- Input: A request to enter information of asset

- Output: Information of asset need to be stored.

- + Step 2: Update to asset file

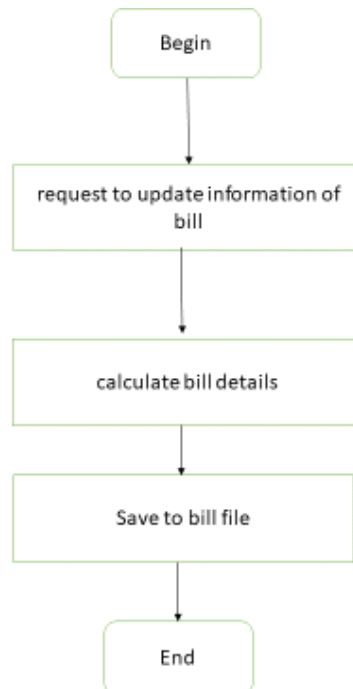
- Input: Information of asset need to be stored.

- Output: asset file.

- **Use case: Enter monthly Bill**

- **Event trigger:** Every month students have to pay rent, utility bills and incurred bills, management staff have to enter the monthly bill for the student to pay

- **Business process model**



- **Description:**

- + Step 1: Get information of bill

- Input: A request to enter information of bill.

- Output: Information of bill need to be stored.

- + Step 2: Update to asset file

- Input: Information of bill need to be stored.

- Output: bill file.

### III. Requirements Specification

#### 1. Log in

<b>Use Case Name</b>	Log in
<b>Trigger</b>	User want to access the system.
<b>Precondition</b>	The Editor has accessed the Article Manager main screen.
<b>Basic Path</b>	<ul style="list-style-type: none"> <li>- At the main user interface, select the login function</li> <li>- The system displaying the login interface will ask the user</li> </ul>



	<p>to enter a username and password.</p> <ul style="list-style-type: none"> <li>• - User enters username, password, chooses agree to login</li> <li>• - System receives information, check username, password of user</li> <li>• - If it is valid, the system accepts login and shows successful login message</li> </ul>
<b>Alternative Paths</b>	<p>1. In step 2, if the user does not want to continue, select cancel</p> <p>1. In step 4, if the login information is incorrect, the system refuses to log in and shows the message</p>
<b>Postcondition</b>	The user logs on to the system
<b>Actor</b>	Student and Staff

## 2. Register at the dormitory

<b>Use Case Name</b>	Register at the dormitory
<b>Trigger</b>	Students who want to enter the dormitory must write a registration form.
<b>Precondition</b>	Students wishing to enter the dormitory.
<b>Basic Path</b>	<p>1. At the main interface, the user chooses the function of registration in the dormitory.</p> <p>2. System displays interface registration</p> <p>3. User inputs information.</p> <p>4. System will receive information, check whether the information is valid</p> <p>5. If valid, the system accepts and informs successful registration</p> <p>6. The user selects the suitable printing function for the contract.</p>
<b>Alternative Paths</b>	<p>1. In step 2, if the user does not want to continue, select Cancel</p> <p>2. In step 4, the system shows that the input information is invalid, the system refused to log in, displays the message</p>

<b>Postcondition</b>	One student is enrolled in the dormitory list
<b>Actor</b>	Student

### 3. Extension Contract

Use Case Name	Extension Contract
<b>Trigger</b>	When the contract expires, the student wishes to continue staying in the dormitory.
<b>Precondition</b>	The contract has expired.
<b>Basic Path</b>	<ol style="list-style-type: none"> <li>1. From the main user interface, select the contract renewal function</li> <li>2. System displays interface extension contract.</li> <li>3. User enters the student code to extend the contract.</li> <li>4. The system receives information, verifies whether the information entered by the user is correct.</li> <li>5. If the information is valid, the system displays the options for renewal such as: 1 term extension, 1 year extension</li> <li>6. User selects one of the options.</li> <li>7. Successful acceptance and notification system</li> </ol>
<b>Alternative Paths</b>	<ol style="list-style-type: none"> <li>1. In step 2,3,5 If the user does not want to continue, select Cancel</li> <li>2. In step 4, the system for checking user's information entered is not in the database; The system displays error messages.</li> </ol>
<b>Postcondition</b>	One student is allowed to renew the contract
<b>Actor</b>	Student

#### 4. Add student

<b>Use Case Name</b>	<b>Add student</b>
<b>Trigger</b>	when a student enrolls in a dormitory
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the main interface, the user selects the Add Student function  2. User inputs information of students he wants to add.  3. The system receives and checks the information entered by the user.  4. If the information is valid, the system will inform you of the success, showing the newly updated student list.
<b>Alternative Paths</b>	In step 2 If the user does not want to continue, select Cancel.  In step 4, the system check info entered doesn't exist in the data base, it will display error message.
<b>Postcondition</b>	Management staff updates the list of student successfully.
<b>Actor</b>	Management Staff

#### 5. Edit Student

<b>Use Case Name</b>	<b>Sdit information of student</b>
<b>Trigger</b>	When student information changes
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the main interface, the user selects the Edit Student function  2. User selects student to edit his info.  3. User inputs information of students want to Edit.  4. The system receives and checks the information entered by the user.  5. If the information is valid, the system will announce the

	correction of the information successfully and display the new list of students.
<b>Alternative Paths</b>	In step 2,3 If the user does not want to continue, select Cancel  In step 4, If the system checks the information is not valid; The system displays error messages.
<b>Postcondition</b>	Management staff updates the list of student successfully
<b>Actor</b>	Management Staff

## 6. Delete Student

<b>Use Case Name</b>	<b>Delete student</b>
<b>Trigger</b>	when the student cancels the contract
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the main interface, the user selects the Delete Student function.  2. User selects student to delete and select delete student.  3. Information processing system recieved.  4. If the student is deleted, the system displays the message.
<b>Alternative Paths</b>	In step 2,If the user does not want to continue, select Cancel
<b>Postcondition</b>	Management staff updates the list of student sucessfully
<b>Actor</b>	Management Staff

## 7. Add Staff

<b>Use Case Name</b>	<b>Add information of Staff</b>
<b>Trigger</b>	when there are new staff.
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the main interface, the user selects the Add staff function

	<p>2. User input information of staff he wants to add.</p> <p>3. The system receives and checks the information entered by the user.</p> <p>4. If the information is valid, the system will inform you of the success, showing the newly updated staff list.</p>
<b>Alternative Paths</b>	<p>In step 2, If the user does not want to continue, select Cancel</p> <p>In step 4, If the system checks the information is not valid; The system displays error messages.</p>
<b>Postcondition</b>	Management staff updates the list of staff successfully
<b>Actor</b>	Management Staff

## 8. Edit Staff

<b>Use Case Name</b>	<b>Update information of student</b>
<b>Trigger</b>	When information of staff changes
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	<p>1. From the main interface, the user selects the Update Staff function</p> <p>2. User selects staff whose information to be edited.</p> <p>3. User input information staffs want to update.</p> <p>4. The system receives and checks the information entered by the user.</p> <p>5. If the information is valid, the system will announce the success of correction of the information , displaying the new list of staffs.</p>
<b>Alternative Paths</b>	<p>In step 2,3 If the user does not want to continue, select Cancel</p> <p>In step 4 If the system checks the information is not valid; The system displays error messages.</p>
<b>Postcondition</b>	Management staff updates the list of student successfully
<b>Actor</b>	Management Staff

## 9. Delete Staff

<b>Use Case Name</b>	<b>Delete information of Staff</b>
<b>Trigger</b>	when the staff cancels the contract
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the main interface, the user selects the Delete staff function. 2. User selects staff whose info he wants to delete and select delete it. 3. Information processing system received. 4. If the staff is deleted, the system display the message.
<b>Alternative Paths</b>	In step 2, If the user does not want to continue, select Cancel
<b>Postcondition</b>	Management staff updates the list of staff successfully
<b>Actor</b>	Management Staff

## 10. Add Room

<b>Use Case Name</b>	<b>Add room</b>
<b>Trigger</b>	When new rooms are added.
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the dorm management interface, the user selects the Add room function 2. User inputs information of room want to add. 3. The system receives and checks the information entered by the user. 4. If the information is valid, the system will inform you of the success, showing the newly updated room list.
<b>Alternative Paths</b>	In step 2, If the user does not want to continue, select Cancel In step 4, If the system checks the information is not valid; The

	system displays error messages.
<b>Postcondition</b>	Management staff updates room list successfully
<b>Actor</b>	Management Staff

### 11.Edit Room

<b>Use Case Name</b>	<b>Edit information of room</b>
<b>Trigger</b>	When information of room changes
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	<p>1. From the dorm management interface, the user selects room, then select Edit function</p> <p>2. User input information of room want to edit.</p> <p>3. The system receives and checks the information entered by the user.</p> <p>4. If the information is valid, the system will announce the success of the correction of the information, displaying the new room list.</p>
<b>Alternative Paths</b>	<p>In step 2,3 If the user does not want to continue, select Cancel</p> <p>In step 4, If the system checks the information is not valid; The system displays error messages.</p>
<b>Postcondition</b>	Management staff updates the room list successfully
<b>Actor</b>	Management Staff

### 12. Delete Room

<b>Use Case Name</b>	<b>Delete room</b>
<b>Trigger</b>	One room can not be used anymore
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the main interface, the user selects the Delete room

	<p>function.</p> <p>2. User selects room which he wants to delete.</p> <p>3. Information processing system received.</p> <p>4. If the student is deleted, the system displays the message.</p>
<b>Alternative Paths</b>	In step 2, If the user does not want to continue, select Cancel
<b>Postcondition</b>	Management staff updates the list of room successfully
<b>Actor</b>	Management Staff

### 13.Add Dormitory

<b>Use Case Name</b>	<b>Add Dormitory</b>
<b>Trigger</b>	When new dormitories are added.
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	<p>1. From the dorm management interface, the user selects the Add dormitory function</p> <p>2. User inputs information of dormitory he wants to add.</p> <p>3. The system receives and checks the information entered by the user.</p> <p>4. If the information is valid, the system will inform you of the success, showing the newly updated dormitory list.</p>
<b>Alternative Paths</b>	<p>In step 2, If the user does not want to continue, select Cancel</p> <p>In step 4, If the system checks the information is not valid; The system displays error messages.</p>
<b>Post condition</b>	Management staff updates dormitory list successfully
<b>Actor</b>	Management Staff

### 14.Edit Dormitory

<b>Use Case Name</b>	<b>Edit information of dormitory</b>
----------------------	--------------------------------------



<b>Trigger</b>	When information of dormitory changes
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	<p>1. From the dorm management interface, the user selects dormitory, then selects Edit function</p> <p>2. User inputs information of dormitory he wants to edit.</p> <p>3. The system receives and checks the information entered by the user.</p> <p>4. If the information is valid, the system will announce the correction of the information successfully and display the new dormitory list.</p>
<b>Alternative Paths</b>	<p>In step 2,3 If the user does not want to continue, select Cancel</p> <p>In step 4, If the system checks the information is not valid; The system displays error messages.</p>
<b>Postcondition</b>	Management staff updates the dormitory list successfully
<b>Actor</b>	Management Staff

### 15.Add Class

<b>Use Case Name</b>	<b>Add class</b>
<b>Trigger</b>	When new classes are added.
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	<p>1. From the student management interface, the user selects the Add class function</p> <p>2. User inputs information of class he wants to add.</p> <p>3. The system receives and checks the information entered by the user.</p> <p>4. If the information is valid, the system will inform you of the success, showing the newly updated class list.</p>
<b>Alternative Paths</b>	<p>In step 2,If the user does not want to continue, select Cancel</p> <p>In step 4, If the system checks the information is not valid; The</p>

	system displays error messages.
<b>Post condition</b>	Management staff updates class list successfully.
<b>Actor</b>	Management Staff

### 16.Edit Class

<b>Use Case Name</b>	<b>Edit information of class</b>
<b>Trigger</b>	When information of class changes
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	<ol style="list-style-type: none"> <li>1. From the student management interface, the user selects class, then select Edit function</li> <li>2. User inputs information of class.</li> <li>3. The system receives and checks the information entered by the user.</li> <li>4. If the information is valid, the system will announce the correction of the information successfully and display the new class list.</li> </ol>
<b>Alternative Paths</b>	<p>In step 2,3 If the user does not want to continue, select Cancel</p> <p>In step 4, If the system checks the information is not valid; The system displays error messages.</p>
<b>Postcondition</b>	Management staff updates the class list successfully.
<b>Actor</b>	Management Staff

### 17.Add Faculty

<b>Use Case Name</b>	<b>Add faculty</b>
<b>Trigger</b>	When new faculties are added.
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	<ol style="list-style-type: none"> <li>1. From the student management interface, the user selects the Add Faculty function</li> </ol>

	<p>2. User inputs information of faculty he wants to add.</p> <p>3. The system receives and checks the information entered by the user.</p> <p>4. If the information is valid, the system will inform you of the success, showing the newly updated faculty list.</p>
<b>Alternative Paths</b>	<p>In step 2, If the user does not want to continue, select Cancel</p> <p>In step 4, If the system checks the information is not valid; The system displays error messages.</p>
<b>Post condition</b>	Management staff updates faculty list successfully.
<b>Actor</b>	Management Staff

### 18.Edit Faculty

<b>Use Case Name</b>	<b>Edit information of faculty</b>
<b>Trigger</b>	When information of faculty changes
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	<p>1. From the student management interface, the user selects room, then select Edit function</p> <p>2. User inputs information of faculty he wants to edit.</p> <p>3. The system receives and checks the information entered by the user.</p> <p>4. If the information is valid, the system will announce that the correction of the information is successful, displaying the new faculty list.</p>
<b>Alternative Paths</b>	<p>In step 2,3 If the user does not want to continue, select Cancel</p> <p>In step 4, If the system checks the information is not valid; The system displays error messages.</p>
<b>Postcondition</b>	Management staff updates the faculty list successfully.
<b>Actor</b>	Management Staff

### 19.Add Assessment of room

<b>Use Case Name</b>	<b>Add Assessment for room</b>
<b>Trigger</b>	At the end of school year, managers conduct assessments of all aspects
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the dorm management interface, the user selects room, then selects Add Assessment function  2. User inputs information of Assessment he wants to add to room.  3. The system receives and checks the information entered by the user.  4. If the information is valid, the system will inform you of the success
<b>Alternative Paths</b>	In step 2, If the user does not want to continue, select Cancel  In step 4, If the system checks the information is not valid; The system displays error messages.
<b>Postcondition</b>	Management staff updates assessment to room successfully.
<b>Actor</b>	Management Staff

### 20.Add Compliment of Student

<b>Use Case Name</b>	<b>Add compliment of student</b>
<b>Trigger</b>	when a student is rewarded.
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the student management interface, the user selects student, then selects Add compliment function  2. User inputs information of compliment he wants to add for room  3. The system receives and checks the information entered by the user.

	4. If the information is valid, the system will inform you of the success
<b>Alternative Paths</b>	In step 2, If the user does not want to continue, select Cancel  In step 4, If the system checks the information is not valid; The system will display error messages.
<b>Postcondition</b>	Management staff updates compliment for student successfully.
<b>Actor</b>	Management Staff

### 21.Add Discipline of student

<b>Use Case Name</b>	<b>Add Discipline of student</b>
<b>Trigger</b>	when a student violates the rules of the dorm
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the student management interface, the user selects student, then selects Add discipline function  2. User inputs information  3. The system receives and checks the information entered by the user.  4. If the information is valid, the system will inform you of the success
<b>Alternative Paths</b>	In step 2, If the user does not want to continue, select Cancel  In step 4, If the system checks the information is not valid; The system will displays error messages.
<b>Postcondition</b>	Management staff updates compliment for student successfully
<b>Actor</b>	Management Staff

### 22.Add Bill

<b>Use Case Name</b>	<b>Add Bill for room</b>
<b>Trigger</b>	At the end of each month, the manager enters the bill of the room

<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the student management interface, the user selects student, then selects Add bill function  2. User inputs information  3. The system receives and checks the information entered by the user.  4. If the information is valid, the system will inform you of the success
<b>Alternative Paths</b>	In step 2, If the user does not want to continue, select Cancel  In step 4, If the system checks the information is not valid; The system will display error messages.
<b>Postcondition</b>	Management staff updates discipline for student successfully
<b>Actor</b>	Management Staff

### 23.Add Calendar

<b>Use Case Name</b>	<b>Add Calendar for staff</b>
<b>Trigger</b>	when the staff assigns a dormitory calendar
<b>Precondition</b>	- Successful login with manager access
<b>Basic Path</b>	1. From the staff management interface, the user selects, then selects Add bill function  2. User inputs information  3. The system receives and checks the information entered by the user.  4. If the information is valid, the system will inform you of the success
<b>Alternative Paths</b>	In step 2, If the user does not want to continue, select Cancel  In step 4, If the system checks the information is not valid; The system will display error messages.
<b>Postcondition</b>	Management staff updates calendar successfully.

<b>Actor</b>	Management Staff
--------------	------------------