|  |  |  |
| --- | --- | --- |
| **Use Case Description** | | |
| **System:** online examination system | | **Student Group ID:** |
| **Use Case name**: Update Course | | **UC ID:** 021 |
| **actor:** admin | | **Priority (H,M,L):** L |
| **Goal:**  update course info. | | |
| Input: name of the course , course number ,prerequisite course, number of credit hours. | | |
| **Pre-conditions:** the course that need toupdate is created previously | | |
| **Steps:** | | |
| **Actor** | **System** | |
| 1.admin enters course name that needs to update. Alt1  2.admin update the course number.  3.admin update the assigned . prerequisite course by writing the new course name and number. Alt1  4.admin update the credit hour to the creating course. Alt1 | 1.1system check if the chosen name available. EXP1.  3.1. system search about the assigned course then assign it to the creating course as prerequisite course. EXP2.  4.system assign the credit to the creating course. | |
| **Exceptional flows:**  EXP1: if the chosen name used previously display "chosen name used try another course try again". Goto 1.  EXP2: if the written prerequisite course name not found .display "prerequisite course not found, try again". Goto 3.  **Alternative flows:**  Alt1: actor can cancel Operation , exit use case. | | |
| **Post-conditions:** data base will be updated with the new changes | | |
| **Output:** send the changes info to admin email**.** | | |

|  |  |  |
| --- | --- | --- |
| **Use Case Description** | | |
| **System:** online examination system | | **Student Group ID:** |
| **Use Case name:** register admin | | **UC ID:** 018 |
| **actor:**  admin | | **Priority (H,M,L):** H |
| **Goal:** register in system's website | |  |
| * **Input:** user name , id , password and other info. | | |
| * **Steps:** | | |
| **Actor** | **System** | |
| 1. Enter Ticket Number , given by an admin. \*ALT1  **3. Actor enter** user name \*alt1  **5.** enter password . \*alt1  **7. enter email address. \*alt1**  **9. enter captcha litters . \*alt1**  **11. click submit button \*alt1** | 2. check if Ticket Number is valid, if valid display “valid” , and proceed to next step.  4. Sys check if the name chosen is available , display "available" . \*EXP1  6. if password more than 8 litters display "ok" \*EXP2 .  8. if email format is correct , display "Correct".\*EXP3  10. if captcha is correct , display "Correct" . \*EXP4  12.system perform the required process to complete the registration (update data base ,send varication email , etc) | |
| **Exceptional flows:**  EXP1: user name already used. Display "user name already used" Go to step 1 .  EXP2 : display" use more than 8 liters"  Go to step 3.  EXP3: email format not correct . display "email format not correct"  Go to step 5  EXP4: captcha wrong . display "try again " update captcha , go to 7  **Alternative flows:**  Alt1: actor can cancel Operation , exit use case. |  | |
| **Post-conditions:** *data base will be updated with user info ,* | | |
| * **Output:** confirmation and user information to client by email. | | |

\* Ticket Number : (to verify that actor is authorized to register as an admin)

|  |  |  |
| --- | --- | --- |
| **Use Case Description** | | |
| **System:** online examination system | | **Student Group ID:** |
| **Use Case name**: Delete/View Fac,Student | | **UC ID:** 023 |
| **actor:** admin | | **Priority (H,M,L):** L |
| **Goal:**  to add or update faculty or student to the system. | | |
| Input: ID of the faculty or student, delete or view selection. | | |
| **Pre-conditions:** the student or faculty that need toview or delete is created previously. | | |
| **Steps:** | | |
| **Actor** | **System** | |
| 1.admin choose between student or faculty. Alt1  2.admin choose between view or delete. Alt1  3.if view: admin enter the Fac,Std ID want to view. if delete: admin enter the ID want to delete. Alt1 | 1.1. system deal with the information as what chosen (student or faculty).  2.2. system deal with the information as what chosen (view or delete).  3.3. system search about the ID if the ID found and view chosen, then system display the information. If ID found and "delete" chosen, then system delete the information from database. EXP1. | |
| **Exceptional flows:**  EXP1.search ID not found . display "enter correct ID".goto 3.  **Alternative flows:**  Alt1: actor can cancel Operation , exit use case. | | |
| **Post-conditions:** data base will be updated with the new changes. | | |
| **Output:** send the changes info to admin email**.** | | |

|  |  |  |
| --- | --- | --- |
| **Use Case Description** | | |
| **System:** online examination system | | **Student Group ID:** |
| **Use Case name**: Create Course | | **UC ID:** 020 |
| **actor:** admin | | **Priority (H,M,L):** M |
| **Goal:**  create course by name and number**.** | | |
| Input: name of the course , course number ,prerequisite course, number of credit hours . | | |
| **Pre-conditions:** the course that need tocreate is not created previously | | |
| **Steps:** | | |
| **Actor** | **System** | |
| 1.admin enters course name. Alt1  2.admin enters the course number. Alt1  3.admin assign prerequisite course by writing its name and number. Alt1  4.admin enter the credit hour to the creating course. Alt1 | 1.1system check if the chosen name available. EXP1.  3.1. system search about the assigned course then assign it to the creating course as prerequisite course. EXP2.  4.system assign the credit to the creating course. | |
| **Exceptional flows:**  EXP1: if the chosen name used previously display "chosen name used try another course try again". Goto 1.  EXP2: if the written prerequisite course name not found .display "prerequisite course not found, try again". Goto 3.  **Alternative flows:**  Alt1: actor can cancel Operation , exit use case. | | |
| **Post-conditions:** data base will be updated with the changes | | |
| **Output:** send the changes info to admin email**.** | | |

|  |  |  |
| --- | --- | --- |
| **Use Case Description** | | |
| **System:** online examination system | | **Student Group ID:** |
| **Use Case name**: Assign/Unassign Course to faculty**.** | | **UC ID:** 019 |
| **actor:** admin | | **Priority (H,M,L):** L |
| **Stakeholders:** administration | |  |
| **Goal:** assign or unassign course to faculty | | |
| Input: name of the course , course number , faculity name | | |
| **Pre-conditions:** the course that need toassign or unassign created previously | | |
| **Steps:** | | |
| **Actor** | **System** | |
| 1.admin enter the course name . Alt1  2.admin choose assign course or unassign course . Alt1  3.admin enter the faculty member ID. Alt1 | 1.1. System check if the course available display "available" . \*EXP1  3.1. system doesn't found the faculty. \*EXP2 | |
| **Exceptional flows:**  EXP1: not found course or wrong spell. Display "course is not found, create course first or try again" goto 1.  EXP2: enter the faculty ID again display "enter faculty ID correctly". Goto 3.  **Alternative flows:**  Alt1: actor can cancel Operation , exit use case. | | |
| **Post-conditions:** data base will be updated with the changes | | |
| **Output:** send the changes info to faculty email**.** | | |

|  |  |  |
| --- | --- | --- |
| **Use Case Description** | | |
| **System:** online examination system | | **Student Group ID:** |
| **Use Case name**: Add/Update Fac,Student | | **UC ID:** 022 |
| **actor:** admin | | **Priority (H,M,L):** L |
| **Goal:**  to add or update faculty or student to the system. | | |
| Input: ID of the new faculty or student and | | |
| **Pre-conditions:** the student or faculty that need toupdate is created previously. | | |
| **Steps:** | | |
| **Actor** | **System** | |
| 1.admin choose between student or faculty. Alt1  2.admin choose between add or update. Alt1  3.if update: admin enter the Fac,STD ID need to update. if add: admin enter the new faculty or student info.  Alt1 | 1.1. system deal with the information as what chosen (student or faculty).  2.2. system deal with the information as what chosen (add or update).  3.3. system search about the ID if the ID found and update chosen, then you can update EXP1. If ID not found and "add" chosen, then you can add. EXP2.  3.4.system generate default user and password for the new member (student or faculty). | |
| **Exceptional flows:**  EXP1.search ID not found . display "enter correct ID".goto 3.  EXP2. Entered a used ID display "ID used, try another ID".goto3.  **Alternative flows:**  Alt1: actor can cancel Operation , exit use case. | | |
| **Post-conditions:** data base will be updated with the new changes | | |
| **Output:** send the changes info to admin email**.** | | |

|  |  |  |
| --- | --- | --- |
| **Use Case Description** | | |
| **System:** online examination system | | **Student Group ID:** |
| **Use Case name**: Delete Course | | **UC ID:** 024 |
| **actor:** admin | | **Priority (H,M,L):** L |
| **Goal:**  delete course. | | |
| Input: name of the course . | | |
| **Pre-conditions:** the course is created previously | | |
| **Steps:** | | |
| **Actor** | **System** | |
| 1.admin enters course name that needs to be deleted. Alt1  2.admin conform. Alt1 | 1.1system check if the chosen name available. EXP1.  2.1. system delete the course. EXP2. | |
| **Exceptional flows:**  EXP1: if the course not exist display "course not available ". Goto 1.  EXP2: if the system cant delete the course .display "error while deleting". Goto 1.  **Alternative flows:**  Alt1: actor can cancel Operation , exit use case. | | |
| **Post-conditions:** course will be deleted | | |
| **Output:** send the changes info to admin email**.** | | |