- ☆ System must be easy to use , It provides friendly user interface
- ☆ Must be speed
- ☆ Don’t take time after failure
- ☆ Don’t take many mega bytes from memory

1.Student information :

1.1 Enter student information

User should enter required data :

- Full name
- ID
- Level
- Address
- Phone number

- ☆ Administration staff can also get these information from student by email
- ☆ Administration staff only who can enter student information
- ➢ 1.1.1 in case of incorrect or incomplete information “show invalid data”

  Send email to student to correct your information

- ➢ 1.1.2 in case of correct information
  Show Success operation

1.*2* Update student information

- ☆ Administration staff can update :
  - ▪ Full name
  - ▪ Address

    - Phone number

- ➢ 1.2.1 in case of incorrect or incomplete information “show unvalid data”

  Send email to student to correct your information
- ➢ 1.2.2 in case of correct information Show Success operation
- ☆ This section authorize to Administration staff only

---

2. Courses Details :

2.1 Display courses information

Display to user courses details , it will show them all current details

- ➢ **2.1.1**

  if user click on (display courses details) , it will show them all current details (current courses, Subsequent courses, the numbers oh hours of each course, name of each course).

---

3. Student services :

- ☆ Must that each student show your details only

3.1 Display student information

Display to student your personal details

- ➢ 3.1.1 If students click on (display personal details), it will show them their full name, level, ID, address and phone number
- ☆ This section authorize to student only and must be login to system

3.2 Display semester details :

Display to student your current semester details

- ➢ 3.2.1 if student click on (display current details) , it will show them all current details (current courses, current GPA, current level), full name and ID.

3.3 case report :

Display to student your case report details about all list of courses in all levels that they toke and full name, level and ID

- ➢ 3.3.1 If student click on (display student information) , it will show them their information about all list of courses (course name, grades, GPA) in all levels that they toke and full name, level and ID

---

4. Administration staff operations

4.1 staff signing in :

✰ Must that this section don’t displayed to anybody , administration staff only who show and modify it.

This operation has 5 tasks that can be done with the administrator only

The admin-user should enter the information of the staff and make for each user an individual username and password. Also there is another function that is done by only the admin user that is the updating of the staff user information.

- ➢ 4.1.1 **signing in :** Before any staff member can access the app, the app asks him to enter his name and password that is previously chosen by admin for each staff member, after entering them the app should valid them if they are valid the app is opened with the interface that is specialized for this user, if it's invalid the system return an "Error message" .

4.2 Add courses : User enter the information that is needed to add this course , then after entering data of course the app should assure that this course is not

- ➢ 4.2.1 **Add course :** When a member of staff wants to add a course to the system he must first open the app then choose **("Add courses")** then the application should connect to the previously-added courses then it should ask the user to enter the information that is needed to add this course like (course name, credit hours, doctors that teach this course, the level in which it should be taught), then after entering these data the app should assure that this course is not

4.3 Helping student about which courses is available for him/her :

When a student wants to know about the courses that is available to him/her, he should ask the staff member that manages this student account to help him/her.

- ➢ 4.3.1 **Available courses for a student :** When a student wants to know about the courses that is available to him/her, he should ask the staff member that manages this student account to help him/her, then the staff member should choose **("Available courses for a student")**, then he asks the student for his/her name, then enter this name in the input box, then the system should valid the entered name?? If it's found on the students' names so the app should return for the user the available courses, if not then the app should return an **"Error message"**.

4.4 Registration of courses for student : When a student wants to register his term courses he should ask the staff member that manages his account to help him/her , then he asks the student for his/her name, then the app should open the registration page which have a list of the available courses for this student, then the student chooses the desired courses that he/she wants to enroll in

- 4.4.1 **Student registration** : When a student wants to register his term courses he should ask the staff member that manages his account to help him/her, first the user should choose **("Student registration")**, then he asks the student for his/her name, then enter this name in the input box, then the system should valid the entered name?? If it not found then the app should return an **"Error message"**, but if it's found, then the app should open the registration page which have a list of the available courses for this student, then the student chooses the desired courses that he/she wants to enroll in

---

4.5 Insert the students' degrees : insert degrees in, after the app gets the response from the user it should open a table which contains all the students of this level (rows) and the courses for each students (columns) then the user should assign the degrees of each student in this table

- 4.5.1 **Insert the students' degrees :** When the user chooses **("Insert Term Degrees")** then the app should ask him to enter which level students he want to insert degrees in, after the app gets the response from the user it should open a table which contains all the students of this level (rows) and the courses for each students (columns) then the user should assign the degrees of each student in this table then he can choose **(“print and save”)** if he wants to print and save this table, or he can choose **(“save and exit”)** if he wants to save without printing the table.

4.6 Calculate the student GPA :

GPA that is calculated according to his degrees in each course

- 4.6.1 **Calculate the student GPA :** Task 3.6 is a subtask from task 3.5, as each student should have a GPA that is calculated according to his degrees in each course from the assigned degrees by the user and a column called student GPA is added after the user finishes assigning the degrees of each student

- ☆ System must be usable
- ☆ System must be Available
- ☆ System must be Extensible
- ☆ System must be reliable