

Designing a smart system for building student's schedule-An application at Neelain University



Mozamel M. Saeed ^{1,*}, Ahmed Fahim ^{1,2}

¹Department of Computer Science, Prince Sattam Bin Abdul-Aziz University, Al-Kharj, Saudi Arabia

²Faculty of Computers and Information, Suez University, Suez, Egypt

ARTICLE INFO

Article history:

Received 29 June 2020

Received in revised form

1 September 2020

Accepted 2 September 2020

Keywords:

Student's schedule

Process of deletion and addition

Building a smart system

Improving the educational process

University of Neelain

ABSTRACT

This paper aims to design a smart system for building student's schedules based on a set of limitations and restrictions that can mainly contribute to the formation of student's schedules. Where the process of schedule formation, addition, and deletion is done automatically, this is our contribution. This system helps to raise the student's abilities, improves his academic achievement, and raises his rate. Moreover, it allows the student to delete and add a course based on his average and the number of hours recorded by the system, and the number of hours he is allowed to register, and some other restrictions. In this research, the courses of the study plan are classified according to the scientific content (nature of the course) into four main groups; 1-preservation materials, 2-understanding materials, 3-higher understanding materials, 4-applied materials. The system should take the academic record of each student and analyzes it for identifying the group where the student has higher academic achievement in comparison with the rest. The process of deletion and addition will be done taking into account the priority of adding the courses of the lower levels and the courses that are prerequisite for other courses at higher levels, and the priority of deletion of courses that do not affect other courses at higher levels. VB Application is used in the practical application of the system, where it succeeds in building student's schedules for the computer department-university of Neelain, which contributed to enhancing the cumulative rate for the students, which reflect positively on the quality of the educational process.

© 2020 The Authors. Published by IASE. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

The institutions of higher education in Sudan have witnessed a significant expansion in the number of universities in both sectors (public and private). This situation requires the emergence of innovative methods of education based on modern techniques in order to meet the high demand in the higher education sector with the aim to achieve quantitative sufficiency and the quality of the various components of the educational process (Musa, 2018).

The study plan is the main foundation of the educational process in any academic institution. It helps in building the transitional processes between the knowledge group and the sciences of

specialization and adopting quality standards in terms of aligning the plan with the scientific specialization. At the university level, the study plan consists of a set of courses in different categories; 1-purely specialized which are known as specialized requirements that are often mandatory, 2-the general requirements which are common to students of the stage in that educational institution or university, and 3-the supporting courses that help enriching general knowledge and in preparing for the specialized courses.

The process of building student's schedule is considered one of the chronic problems facing the educational process in many educational institutions. It is repeated continuously at the beginning of each semester. The problem is created by the mechanism currently used during the building of the schedules that it does not take into account the skills and abilities of the students with regard to their current academic status (the general accumulative rate). The contribution of this research is that the proposed system builds the schedule automatically. The addition and deletion of courses

* Corresponding Author.

Email Address: m.musa@psau.edu.sa (M. M. Saeed)

<https://doi.org/10.21833/ijaas.2021.01.011>

Corresponding author's ORCID profile:

<https://orcid.org/0000-0002-5598-0734>

2313-626X/© 2020 The Authors. Published by IASE.

This is an open access article under the CC BY-NC-ND license

(<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

do not require an academic advisor. The system acts as an academic advisor for the student. The system takes into consideration all restrictions specified by the department and study plan while building and updating (add/delete course) the schedule. In the proposed system, the addition process is carried out in accordance with certain restrictions as follows: 1- no conflict in a time of courses in student's schedule, 2-prerequisite for course must be successfully passed, 3-exceeding the maximum burden allowed for registration according to the student's accumulative rate is not allowed, 4-the course should be within the study plan or optional or free courses available and accredited (if found) and provision of seats in the required division. On the other hand, the restrictions for the deletion process are as follows: 1-the number of school hours should not be less than the minimum school burden allowed for registration, which is 12 hours. If the course needed to be deleted, a requirement coinciding with another course at the same level, the student may not be allowed to delete it, except by deleting both courses together. The course needed to be deleted should not be a course from zero levels for the student.

2. Literature review

Computing has been used previously in solving the problems for about a while, where artificial intelligence and developmental algorithms have been used before (Burke and Landa, 2004), whereas their applications have been found in examinations and the other on studying halls. Conventional appointment classification and scheduling methods have focused on the act of categorizing as well as appointing systems.

Previous studies -as mentioned before- keenly saturated on finding solutions for the assignment of appointments for students and teachers as in Nasution (2015). The research aimed to achieve a transition from the conventional manual system to an online solution that copes with the requirements of the rapidly growing technological era. It helped both the student and the lecturer to agree on a time in which they both would be free for the student to receive academic consolation in an attempt to save the time and effort mostly made by the student.

Fanty et al. (1995) found more integrated solutions in which they would use a conversation-based solution as well as a calendar to find a suitable appointment time for the engagement party with the required end-user. This solution was based on three integral parts; a voice recognized to engage with the dialogue, a robust phrase spotting parser, and a dialogue logic module. The research concludes that the second part is most important for the solution, although it requires a specific set of patterns to operate.

Onuwa (2015) focused on the development and implementation of a system of timetable management specifically designed for students who utilize mobile phones. This research highlights a problem uneasy to solve analytically. However, this

research focuses on the development aspect more than the implementation in such a way that makes the timetable matter feasible for students specifically.

Osman (2013) achieved success in the development of a timetable for the entire university of Al-Jouf, utilizing generic algorithms. He proposes a method of genetic algorithms as it is suitable for the university environment. The resulting solution was broad and contained various sectors and professions engaged in the timetable while avoids any conflict of timing. He concludes that there is no fixed number of generations for the operation of input as it varies from one operation to another.

Pishva and Nishantha (2008) added the timetable management system to the distance education field, as well as its application to various structures and architecture of classrooms. They conclude that various technical challenges will be facing the development and implementation phases in particular due to their delicate and sensitive nature.

Chowdhary et al. (2014) introduced a timetable management system that automatically sets the time of each class and teacher based on the availability of resources (i.e., computer rooms, laboratories, and classrooms). The timetable can be viewed by students and teachers alike after inserting their own credentials. However, both parties are not permitted to edit the content of the timetable as it is finalized by the management. The research comes to the conclusion that such methodology reduces time and effort consumption caused by manually inserting data. Also, it is developed in such a way that no clashes or duplication of tasks for teachers or students alike occur.

Our proposed system focuses on building the academic courses schedule for each student based on course timelines, student's academic records, and the academic performance of each student. In our system, the study plan is categorized according to the scientific content (nature of the course) into four main categories; 1-preservation materials, 2-understanding materials, 3-higher understanding materials, 4-applied materials. This classification helps the proposed system to analyze the student's skills and academic performance, also it will help the student while selecting a course for addition or deletion, so the system acts as an academic advisor for the student.

3. The proposed system

The idea of modeling the proposed system for building student's schedules according to the study plan, considering that all students have completed all their academic procedures that are associated with the registration process. The process of building student's schedule is done through two stages; the first stage is to fill the schedule for each student. The second stage is the deletion and addition of courses if the student wants to change in his schedule. Each stage has requirements and procedures that are different from the other, despite the dependence of

stages on each other. When building the student's schedule, the system automatically records the first-level student's schedule with minimal restrictions since the student is new in the department. The schedule will contain the first five courses in the study plan. As for the students of the levels from the second to the last, the system will bring the courses that have been successfully passed (Academic Record). In order to extract the remaining courses for each student, those are calculated by subtracting the courses that have been successfully passed from the study plan. Verification should be made for the remaining courses that the student is entitled to register (These are the remaining courses that the student has passed their prerequisite courses, plus the remaining courses that do not have prerequisite). The courses without conflict in time and courses with conflict should be identified in two separate tables. The system will select only one course from each group of conflicting courses; the selected course will be for the lower level and is added to the group of courses without conflict. This combination of courses is the available courses for the student. The system will use this list of courses to build the initial schedule according to the hours allowed to be recorded for the student. The process of building student's schedule is shown in Fig. 1.

The second phase is concerned with the deletion and addition process; this process allows the student to change some courses in his schedule based on all available courses for him according to his academic register. The first week of each semester is always dedicated to the deletion and addition of courses, which is specified according to the academic calendar of the educational institution. This process contains two tracks. The first track represents the process of deletion, in which the course that is needed to be deleted is chosen from the student schedule. In this track, the number of registered hours in the student's schedule should not be less than the minimum units (12 hours) before or after the deletion process. Otherwise, the process is rejected since the system selects the optimal courses for the student. The main constraints should be observed during the deletion process are listed as follows:

- 1) If the units are less than the minimum, it means that the course cannot be deleted.
- 2) If the units are higher than the minimum units, it is important to make sure that the course that is needed to be deleted does not represent a prior requirement in the subsequent classrooms.
- 3) If the course is not a prerequisite to subsequent courses, it can be deleted from the student's schedule without a problem.
- 4) If the course is a prerequisite to other courses, it is necessary to confirm the deletion in all cases since deleting such course will affect the student in the following semesters, or to retract and let the schedule remain.

The second track represents the addition of courses, in which the course that is needed to be added is chosen from the list of available courses (courses without conflict and groups of conflict courses in teaching time). In this case, the condition is that the student must not exceed the maximum limit allowed for him according to his general grade; otherwise, the process is rejected. Since, in addition, the process the student may want to interchange between two conflicting courses in the same group, this process is allowed by performing deletion first then perform addition. Fig. 2 sketches the progress of the addition and deletion process.

4. Empirical study

The process of building student's schedule is carried out through the computer system, conducting all the logical procedures and restrictions that are associated with the deletion and addition, and then calculating the semester rate and global rate of each student, under which the allowable limit for hours is determined, by using the following equations:

$$\text{Semester rate} = \frac{\text{Total points in semester}}{\text{Total credit hours in semester}} \quad (1)$$

where,

$$\text{Total points in semester} = \sum \text{Course credit hours} * \text{Weight} \quad (2)$$

$$\text{Semester rate} = \frac{\text{Total points in semester}}{\text{Total registered credit hours in semester}} \quad (3)$$

$$\frac{\text{The general accumulative rate} = \frac{\text{Total points in all semesters}}{\text{Total registered credit hours in all semesters}}}{\text{Total registered credit hours in all semesters}} \quad (4)$$

Table 1 forms the bases for a course grade, semester grade, and general accumulative grade calculation for the student using the previous formulas. Semester rate and the general accumulative rate are ranging from 1 to 5, and this value is translated to its equivalent grade letter as in Table 1.

Table 1: Ladder of degrees

Degree	Grade	Weight	Points
95-100	A+	5.00	5.00
90- Less than 95	A	4.75	4.75
85- Less than 90	B+	4.50	4.50
80- Less than 85	B	4.00	4.00
75- Less than 80	C+	3.50	3.50
70- Less than 75	C	3.00	3.00
65- Less than 70	D+	2.50	2.50
60- Less than 65	D	2.00	2.00
Less than 60	F	1.00	1.00

The research automates the creation process of the student's schedule at Computer Department-Neelain University using VB application, through which a database is built, and interfaces are implemented. The main screen (Fig. 3) serves as an interface of the system and contains buttons for

transition and control the interaction with the proposed system.

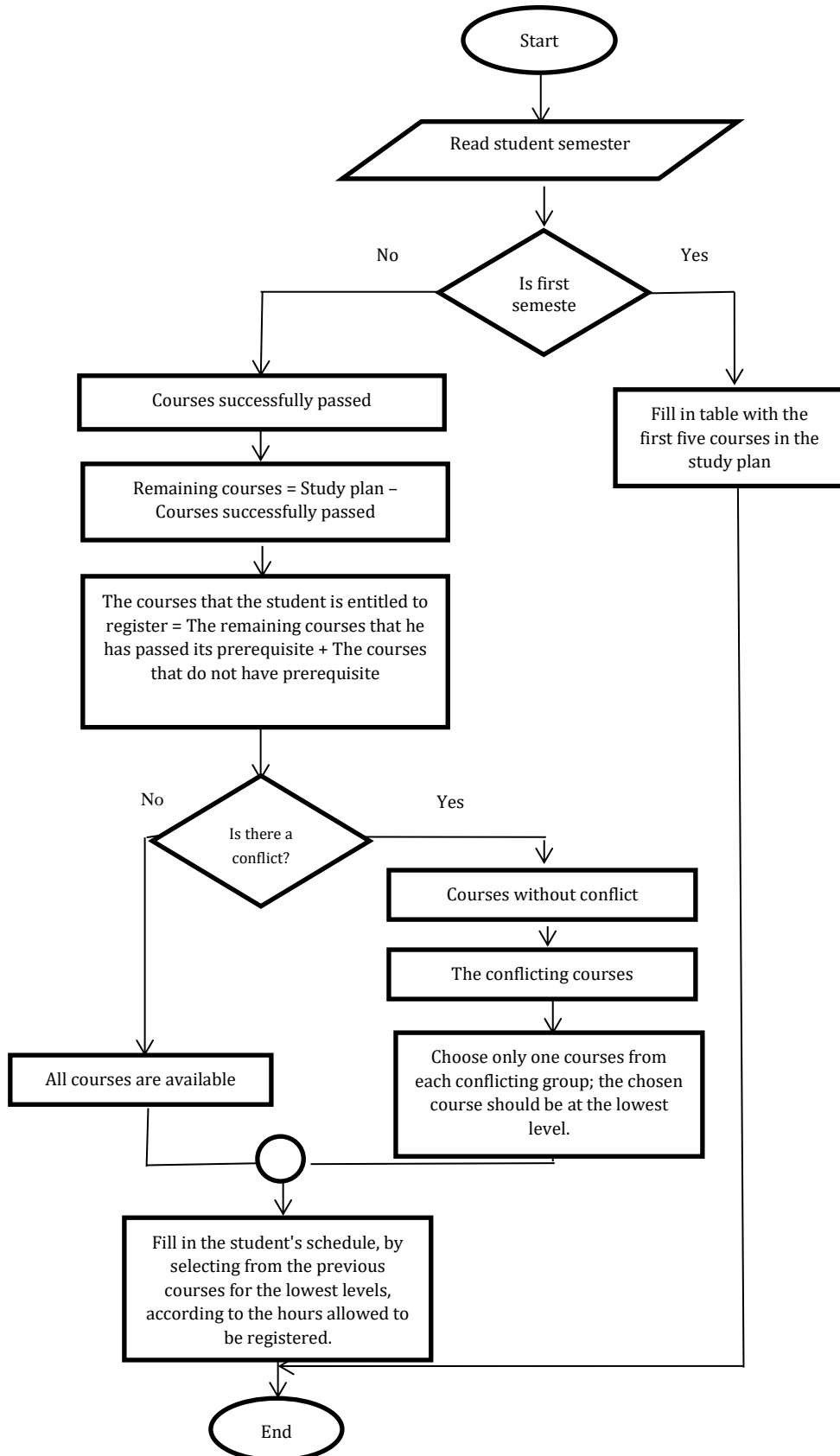


Fig. 1: Building student's schedule process

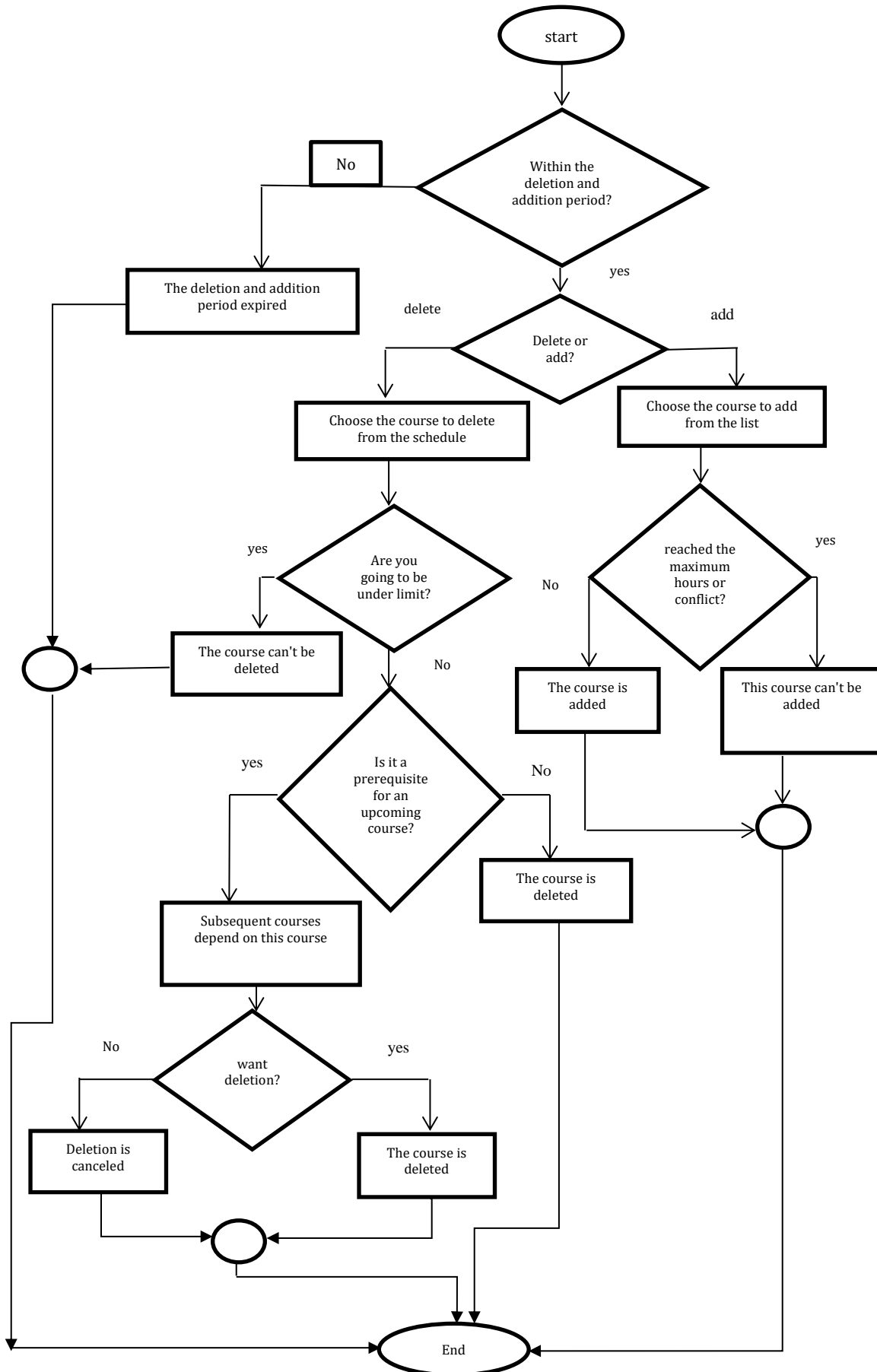


Fig. 2: The deletion and addition process

The system builds the student's schedules from the first to the last level. The system automatically builds the schedule for the student at the first-level

by selecting the first five courses from the study plan and adds them to the student's schedule. For the levels from the second to the last, the system

analyzes the current status of the student (the academic record), where the courses are classified according to the student rate in each individual passed the course, where the creation process depends on this analysis. The system finds the courses successfully passed by the student, then subtracts these courses from the study plan. The resulting courses from this operation will be referred to as remaining courses, and then the system excludes courses where its prerequisite course did not successfully pass or did not study by the student; the result of this operation is called the available courses. The available courses list may contain some conflicting courses. So, the system divides this list into two lists; the first one is courses without conflict, and the other is courses with conflict. There is some courses conflict in time X and others in time Y, and so on, so the system selects only one course from each group to be available for the student, the system selects the lowest level course from each group of conflicting courses. After that, the system selects the lowest level courses from the final available courses and adds them -one by one- to the student's schedule, such that the registered hours do not exceed the allowed hours to register for the student according to his general rate. In this stage, the student has an optimal schedule. The system is very stable. If we run the system many times, the result doesn't change. Each student gets the same schedule in each run. The building schedule is shown in Fig. 1. However, the student has the opportunity to make some changes in his schedule by adding or/and removing courses according to the rules discussed before. So the system will keep -for each student- the list of available course and the conflict courses, so the student can directly select from them one course to be added to his schedule if the selected course is already in the schedule or in conflict with an existing course in the schedule it will not be added, so the system avoids and prevent the problems of conflicting courses and duplicate of any courses. This process is known as the deletion and addition process, and it is shown in Fig. 2. When the student wants to delete one course from his schedule, he selects the course to delete, and the system applies rules of the deletion process

discussed before, as shown in Fig. 2. When the student tries to delete a course represents a prerequisite for an upcoming course -that may affect the student in the advanced levels and delay graduation- the system displays an informative message for the student and allows him to cancel the process. Figs. 3 to 22 show the most important operations in the proposed system.

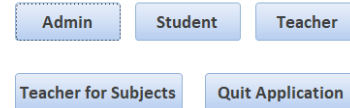


Fig. 3: The main application screen

Through the screen in Fig. 4, the student's rate is calculated for all students first. Based on student's rate, the system determines the allowed hours for each student, then updates the semester's number for all students, each according to the year of admission to the university, and finally constructs the schedules for all students, each according to his average and the courses that he successfully passed. Moreover, it is possible to undo the updating of the semester or the schedules that have been built.

Fig. 5 shows the Schedule that has been built for a student who is in his first semester at the university. The schedule contains the first five courses in the study plan. Fig. 6 shows the student's screen through which the student can review his current schedule and the credit hours for each course, and the timetable for each course. Fig. 7 shows a student's screen that explains to him the points he obtained in each course. It is noted that the grade weight (g.w) of these courses is zero, which means that their grades (stgrade) have not yet been monitored. Fig. 8 shows a student's screen that explains to him the points he obtained in each course. It is noted that the student has failed in course No (5) named calculus and his grade weight is one in this course. Fig. 9 shows a student's screen that explains to the student the courses that he passed successfully and the credit hours for each course. Note that the student fails in course number 5; it is not listed here in Fig. 9.

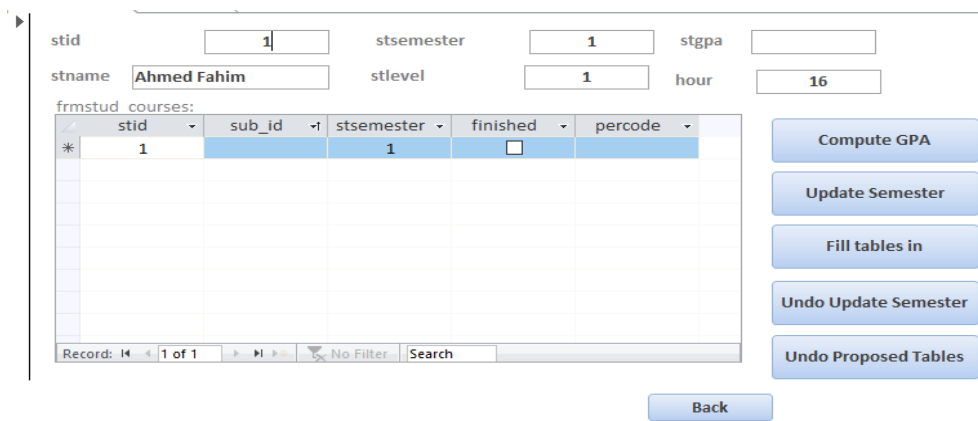


Fig. 4: System administrator's screen

stid stsemester stgpa
 stname stlevel hour

frmstud courses:

stid	sub_id	stsemester	finished	percode
1	1	1	<input type="checkbox"/>	1
1	2	1	<input type="checkbox"/>	2
1	3	1	<input type="checkbox"/>	1
1	4	1	<input type="checkbox"/>	2
1	5	1	<input type="checkbox"/>	2
*	1	1	<input type="checkbox"/>	

Record: 14 | 1 of 5 | No Filter | Search

Fig. 5: New student's schedule

stid hour_allowed lastaddremove
 stname reg hours

Table Allowed Contradiction Average mark for groups GPA Results Finished courses Remaining courses

sub_id	sub_name	day	stime	endtime	base	c_h
1	Reading skills	1	8	10	0	3
2	writing skills	1	10	12	0	3
3	Listening Skills	2	8	10	0	3
4	Computer Basics	2	10	12	1	3
5	Calculus	3	10	12	1	3

Fig. 6: Reviewing a student's schedule

stid hour_allowed lastaddremove
 stname reg hours

Table Allowed Contradiction Average mark for groups GPA Results Finished courses Remaining courses

sub_id	sub_name	finished	note	stgrade	g_w
1	Reading skills	<input type="checkbox"/>		No mark yet	0
2	writing skills	<input type="checkbox"/>		No mark yet	0
3	Listening Skills	<input type="checkbox"/>		No mark yet	0
4	Computer Basics	<input type="checkbox"/>		No mark yet	0
5	Calculus	<input type="checkbox"/>		No mark yet	0

Fig. 7: A student's schedule with grades not monitored

stid hour_allowed lastaddremove
 stname reg hours

Table Allowed Contradiction Average mark for groups GPA Results Finished courses Remaining courses

sub_id	sub_name	finished	note	stgrade	g_w
1	Reading skills	<input checked="" type="checkbox"/>		C+	3.5
2	writing skills	<input checked="" type="checkbox"/>		C	3
3	Listening Skills	<input checked="" type="checkbox"/>		B	4
4	Computer Basics	<input checked="" type="checkbox"/>		A	4.75
5	Calculus	<input type="checkbox"/>		F	1

Fig. 8: A student's schedule with grades monitored

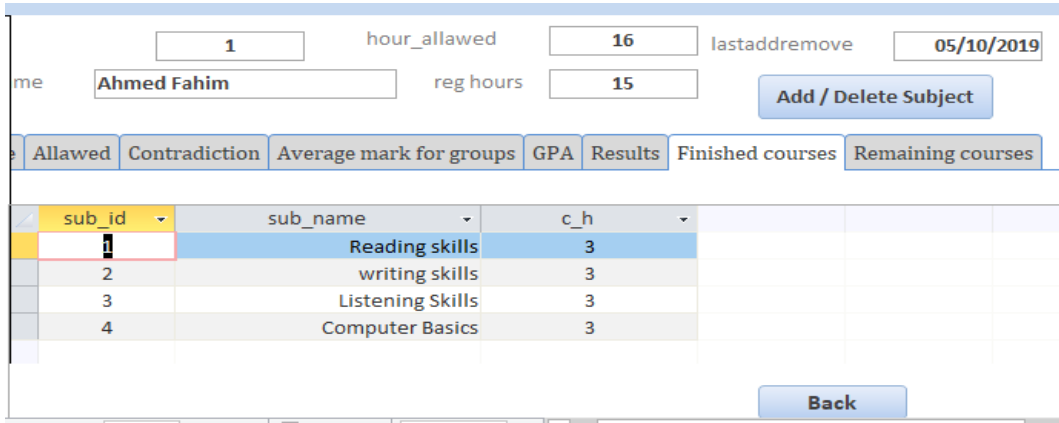


Fig. 9: The courses passed by the student

Fig. 10 shows the student's screen, which explains to him the remaining courses according to his study plan and the prerequisite (prev) of the course, if any. It is noted that course No (5) is a prerequisite for course No (6) where the student failed passing it, as shown in Fig. 8.

Fig. 11 shows the system Administrator's screen, which explains the process of building the schedules, depending on the student's rate and hours allowed him to register. Fig. 11 shows the schedule built for the student in the second semester. It should be noted that course No (6) is not added to his schedule

because it depends on course No (5), where the student failed passing it during the previous semester.

Fig. 12 shows the student's screen, which explains to him his course schedule that was built automatically and the time for the lectures. Note that the student rate allows him 17 credit hours to study in this semester, but the system gives him 16 credit hours, so the student may delete 2 credit hours course and add 3 credit hours course since there is no one credit hour course available if he wants to take all the 17 hours allowed for him to register.

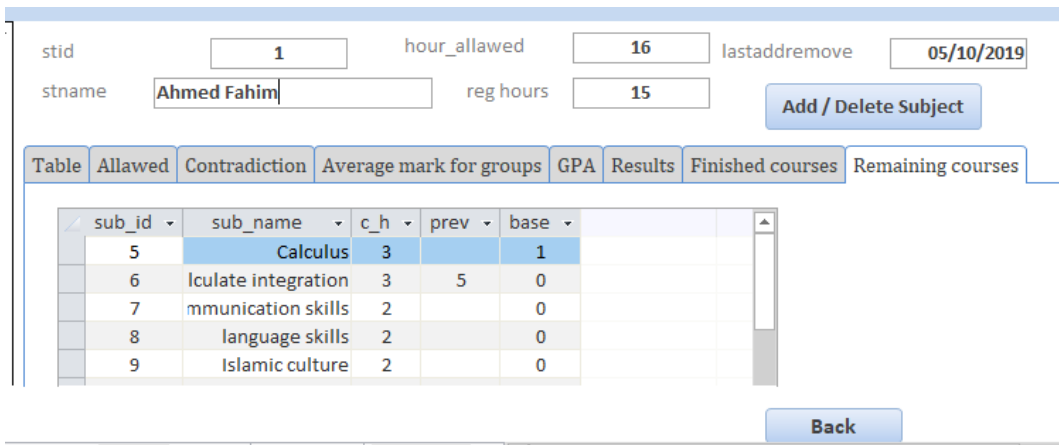


Fig. 10: The remaining courses for the student and the Prior courses

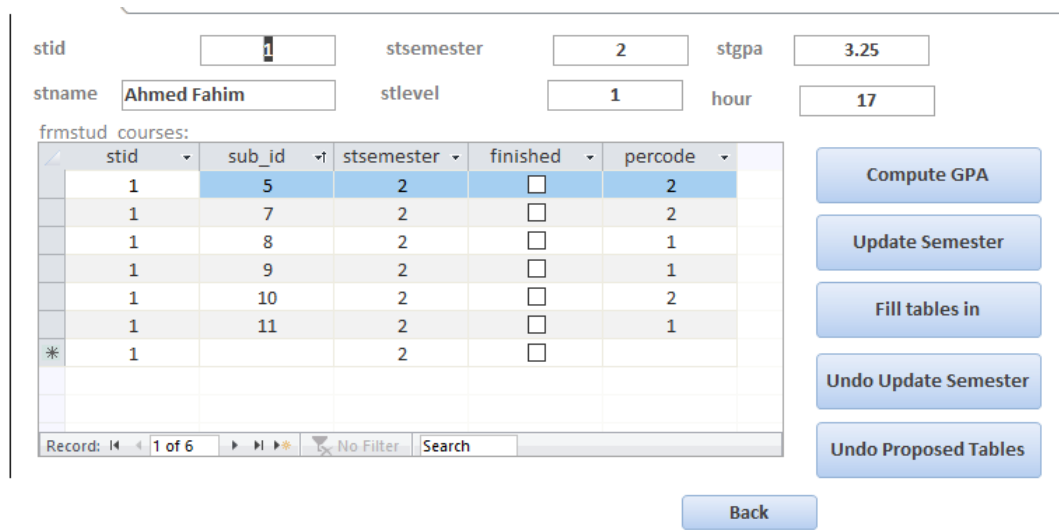


Fig. 11: Mechanism of building schedules

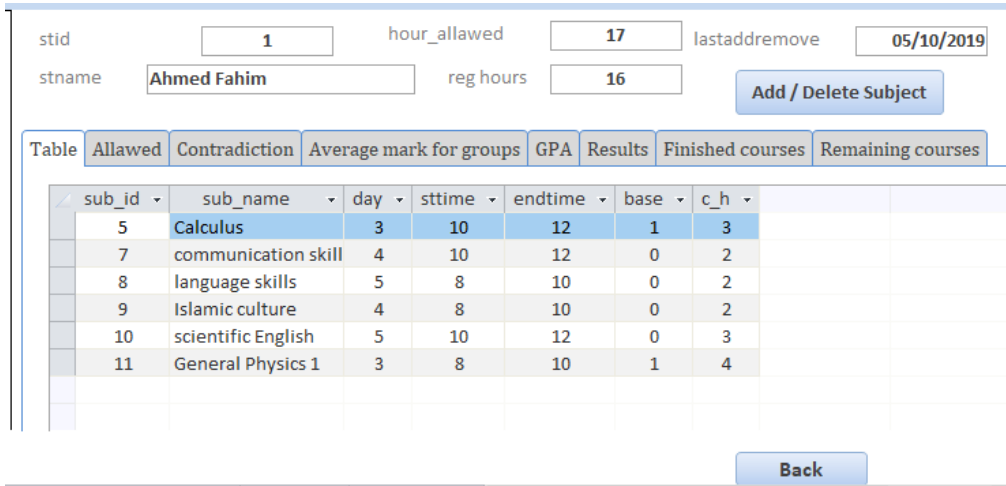


Fig. 12: The built student's schedule

Fig. 13 shows the student's screen, which shows him the courses from which he can choose to study, as there is no conflict in the time of lectures. It also shows a classification of courses according to the

student's abilities and rate in this type of course, which helps the student to determine one or more courses that can be added, or when he wants to make changes in his schedule.

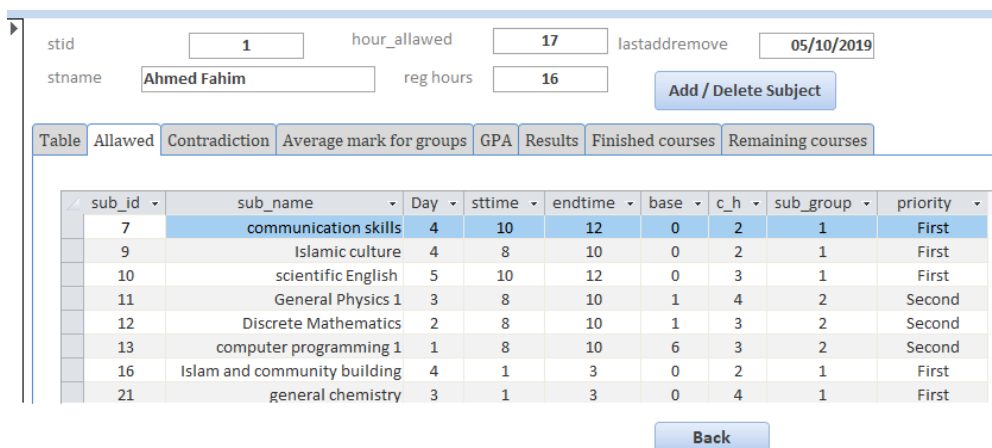


Fig. 13: Courses available for addition according to the student's potentials

Fig. 14 shows the student's screen, which helps him to determine the courses that he can add from among the conflicting courses. It also exposes to him groups of conflicting courses at the time of the lecture. Then he can choose from each group only one course to add to his schedule. Note that two courses are on Tuesday from 10 to 12 am, and the other three courses are on Thursday from 8 to 10 am.

Figs. 13 and 14 are the most important thing in the system. To search for a course to add to the schedule, you spend a lot of time and require a lot of information, from the study plan, a timetable for courses in the current semester, and the academic history for the student. This information exists in different documents, and you need to note the time for each course, and if it has a pre-required course, and so on. Here the system acts as an academic supervisor for the student.

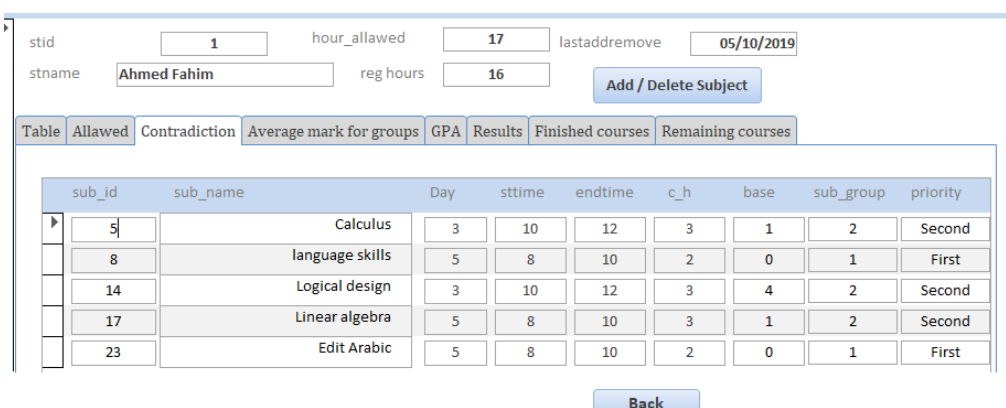


Fig. 14: The conflicting courses

Fig. 15 shows the student the percentage of grades in each type of course. This percentage is used in determining the course classification according to the student's abilities, which gives priority to courses for the student when making the

addition. Fig. 15 analyzes the student's obtained grades in passed courses all over the previous semesters and arranges the group of courses in which the student performs well first.

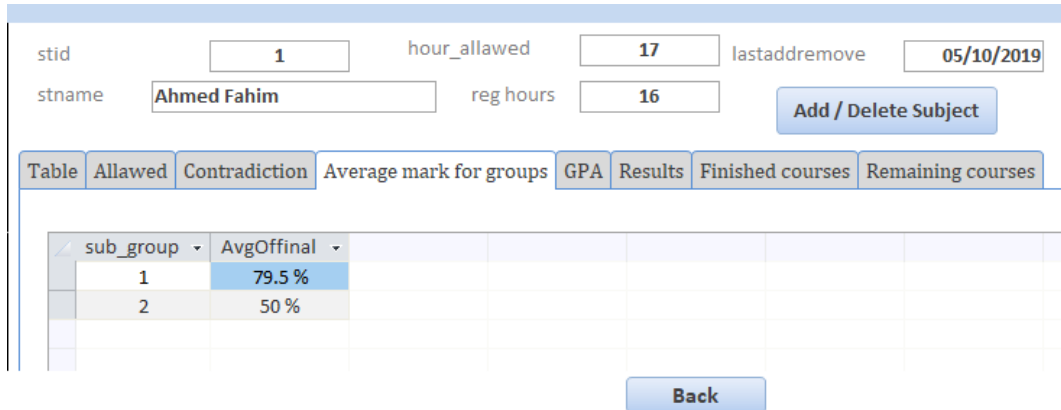


Fig. 15: Rating ratio - Courses classification

Fig. 16 shows the student's average in the previous semesters, the total points obtained, and credit hours. Note that, semester rate is 3.25

calculated using Eq. 3, so the student's grade for this semester is C from Table 1.

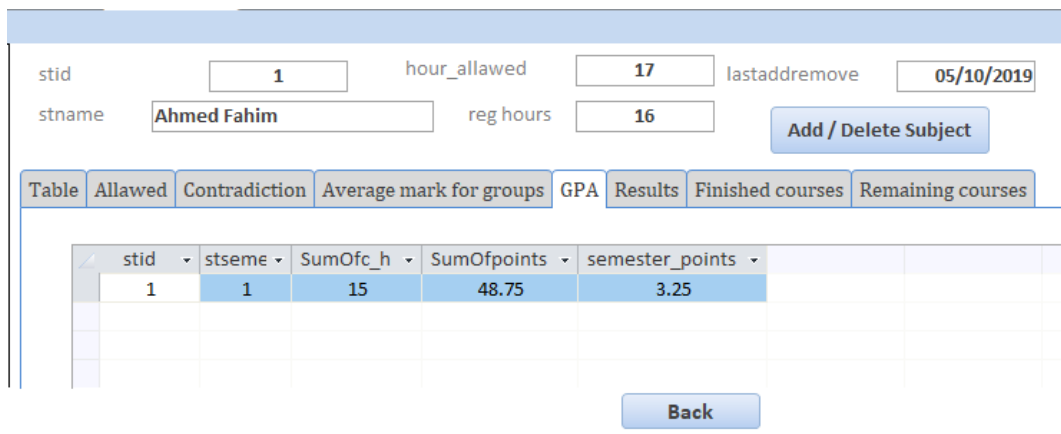


Fig. 16: Student's average

Fig. 17 shows that the grades have not been monitored until the appointment time. Fig. 17 represents the student's score for the current semester. Fig. 18 shows a message, which states that

the deletion and addition period is over. Fig. 19 shows the courses from which the student can choose for the addition, according to the limit allowed for him.

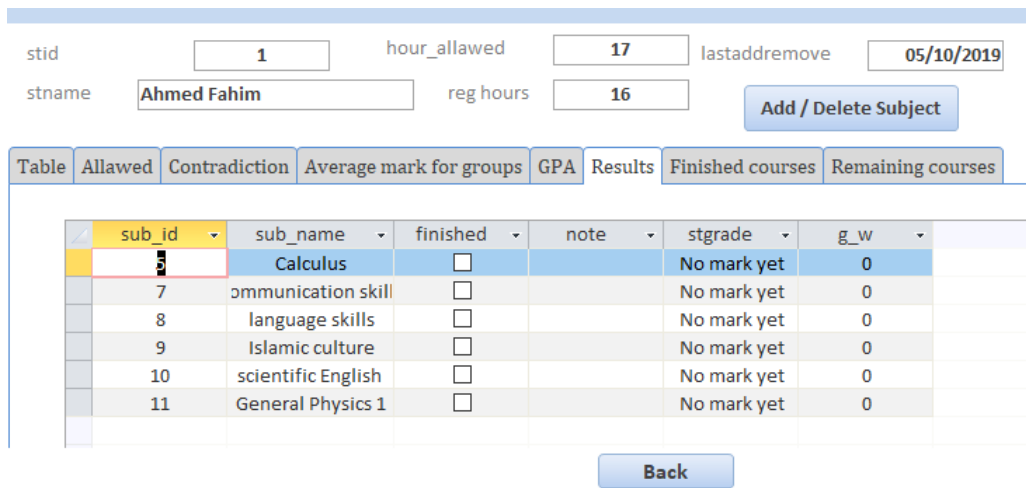


Fig. 17: Student's result

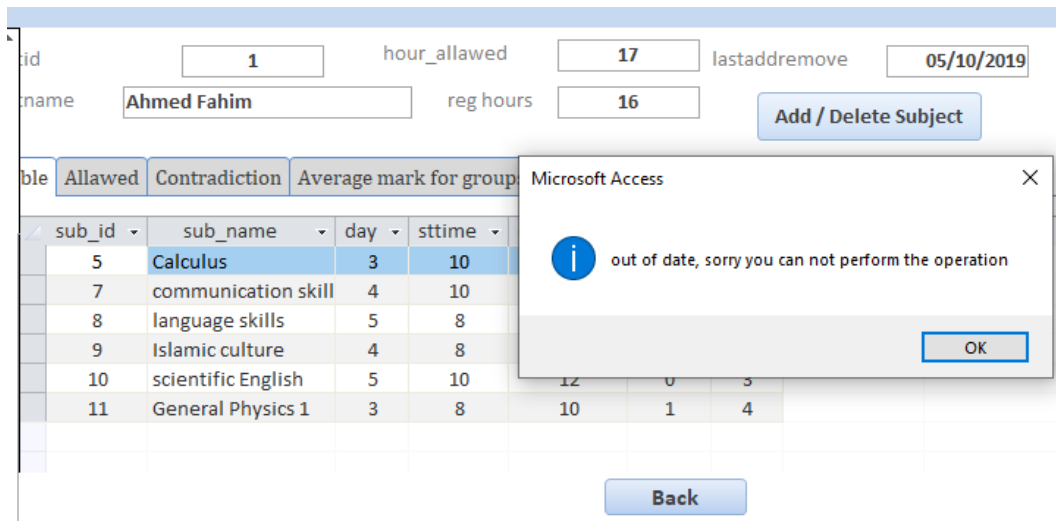


Fig. 18: Deletion and addition period

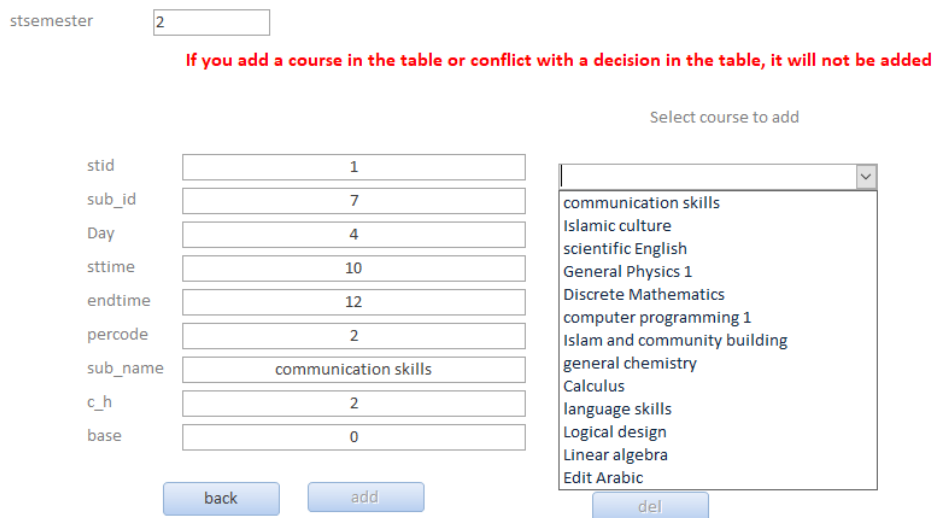


Fig. 19: Select a course for addition from the list

Fig. 20 shows a message, which states that the student has reached the maximum limit and cannot add the course. Fig. 21 shows the courses in the student's schedule from which he can delete. Fig. 22 shows a message, which states that the course you want to delete is a prerequisite for an upcoming

course. It is preferred not to delete, and the student has the right to cancel the deletion process. Fig. 23 shows a message, which states that the student has decided to cancel the deletion process that he was making on the previous screen.

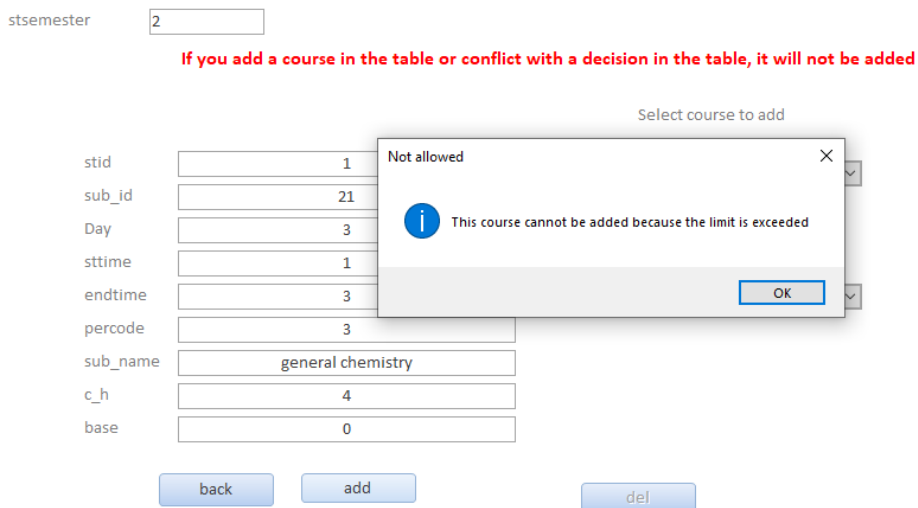


Fig. 20: A course addition process after reaching the maximum

stsemester

If you add a course in the table or conflict with a decision in the table, it will not be added

stid	<input type="text" value="1"/>	Select course to add	<input type="text"/>
sub_id	<input type="text" value="7"/>		
Day	<input type="text" value="4"/>		
sttime	<input type="text" value="10"/>	Select course to delete	<input type="text"/>
endtime	<input type="text" value="12"/>		
percode	<input type="text" value="2"/>		
sub_name	<input type="text" value="communication skills"/>		
c_h	<input type="text" value="2"/>		
base	<input type="text" value="0"/>		

back
add
del

Fig. 21: Select a course for deletion

stsemester

If you add a course in the table or conflict with a decision in the table, it will not be added

stid	<input type="text" value="1"/>	Select course to add	<input type="text"/>
sub_id	<input type="text" value="11"/>		
Day	<input type="text" value="3"/>		
sttime	<input type="text" value="8"/>	Select course to delete	<input type="text" value="General Physics 1"/>
endtime	<input type="text" value="10"/>		
percode	<input type="text" value="1"/>		
sub_name	<input type="text" value="General Physics 1"/>		
c_h	<input type="text" value="4"/>		
base	<input type="text" value="1"/>		

back
add
del

Advise ✕

there is/are 1 cours/es depends on this cours

OK
Cancel

Fig. 22: Course deletion process

stsemester

If you add a course in the table or conflict with a decision in the table, it will not be added

stid	<input type="text" value="1"/>	Select course to add	<input type="text"/>
sub_id	<input type="text" value="11"/>		
Day	<input type="text" value="3"/>		
sttime	<input type="text" value="8"/>	Select course to delete	<input type="text" value="General Physics 1"/>
endtime	<input type="text" value="10"/>		
percode	<input type="text" value="1"/>		
sub_name	<input type="text" value="General Physics 1"/>		
c_h	<input type="text" value="4"/>		
base	<input type="text" value="1"/>		

back
add
del

information ✕

i you canceled the operation

OK

Fig. 23: Cancellation of the deletion process

5. Conclusion

An integrated computer system was provided for the process of building students' schedules, based on the classification of courses in the study plan. The procedures for deletion and addition should be carried out through these classifications and in accordance with the abilities and capabilities of the students.

The paper explains the high efficiency of the proposed system that applied to the students of Computer Department-Neelain University in Sudan. The paper has contributed directly to improving the overall level of students, which has reflected positively on the quality of the educational process and led to a decrease in the rate of stumbles among students.

Compliance with ethical standards

Conflict of interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

Burke EK and Landa Silva JD (2004). The design of memetic algorithms for scheduling and timetabling problems. In: Hart

WE, Krasnogor N, and Smith JE (Eds.), Studies in fuzziness and soft computing: Vol. 166. Recent advances in memetic algorithms and related search technologies: 289-312. Springer, Berlin, Germany.
https://doi.org/10.1007/3-540-32363-5_13

Chowdhary A, Kakde P, Dhoke S, Ingle S, Rushiya R, and Gawandeg D (2014). Timetable generation system. International Journal of Computer Science and Mobile Computing, 3(2): 410-414.

Fanty M, Sutton S, Novick DG, and Cole R (1995). Automated appointment scheduling. In ESCA Workshop on Spoken Dialogue Systems, Vigs. Denmark: 144-147.

Musa MMS (2018). Designing and implementation of a flexible institution system according to new standards to ensure the quality of educational process in higher education institutions. International Journal of Computer Science and Technology 9(3): 51-59.

Nasution AI (2015). Developing smart appointment and scheduling system for innovation in smart universities. International Journal of Computer Science and Information Technology Research, 3(4):112-118.

Onuwa NI (2015). Design and implementation of mobile based students timetable management system. Higher National Diploma Thesis, Akanu Ibiam Federal Polytechnic, Amoncha, Nigeria.

Osman A (2013). Building Al Jouf University timetable using genetic algorithm. Available online at:
<https://www.researchgate.net/publication/307633722>

Pishva D and Nishantha GGD (2008). Smart classrooms for distance education and their adoption to multiple classroom architecture. Journal of Networks, 3(5): 54-64.
<https://doi.org/10.4304/jnw.3.5.54-64>