

A Prototype Student Advising Expert System Supported with an Object-Oriented Database

M. Ayman Al Ahmar

Deputy Dean, College of Information Technology
Ajman University of Science and Technology (AUST)
United Arab Emirates (UAE)

Abstract— Using intelligent computer systems technology to support the academic advising process offers many advantages over the traditional student advising. The objective of this research is to develop a prototype student advising expert system that assists the students of Information Systems (IS) major in selecting their courses for each semester towards the academic degree. The system can also be used by academic advisors in their academic planning for students. The expert system is capable of advising students using prescriptive advising model and developmental advising model. The system is supported with an object-oriented database and provides a friendly graphical user interface. Academic advising cases tested using the system showed high matching (93%) between the automated advising provided by the expert system and the advising performed by human advisors. This proves that the developed prototype expert system is successful and promising.

Keywords- academic advising; expert system; object-oriented database.

I. INTRODUCTION

Student academic advising is an essential task in educational institutions. Traditionally a university student plans the courses semester-by-semester towards a degree through lengthy meetings with the human academic advisor. Advising meetings are usually held during the beginning of each academic semester. Since student advising is a time-consuming effort, there is a need for computerization of some parts of the advising process. Utilizing a computerized advising system, students can save the software consultation results and can then meet with the human advisor for further consultation (if there is still a need for the traditional face-to-face meeting). This hopefully will save valuable time for academic advisors and for students.

The objective of this research is to develop a prototype rule based Expert System (ES) for the academic advising of the students of the Information Systems (IS) Department, College of Information Technology, Ajman University of Science and Technology (AUST), UAE. The system is called "IS-Advisor" and helps students in course selection for each academic semester.

Literature reveals many endeavors in the field of automating academic advising activities including the application of expert systems [1-9]. There can never be a 'global' expert student advising system applicable to all academic institutions and departments because of the existence

of academic regulations and expert advising knowledge and reasoning specific to each academic unit.

As an example to illustrate this concept, AUST regulations allow each student to register from three to six courses per semester. However, the accumulated advising experience and grade statistics related to the IS department show that students who can be advised to register six courses without difficulty are students with AGPA 3.00 or above (out of 4.00), whereas students with AGPA greater than 2.00 and less than 2.25 are better advised to register 4 courses only the next semester to give them a chance to increase their AGPA.

The proposed ES (IS-Advisor) represents such specific advising knowledge and reasoning as rules in its knowledge base component and reasoning strategies in its inference engine component. Thus, the ES developed in this research is unique in its specific knowledge base and reasoning strategies and is intended to be of great help to the department of IS. Another particular feature of IS-Advisor is the Object-Oriented (OO) architecture of its database as will be addressed in subsequent sections.

II. MODELS OF ACADEMIC ADVISING

From the literature we select two models of academic advising adopted in the proposed expert system: Prescriptive advising model and developmental advising model. The prescriptive advising model is characterized by an advisor-student relationship in which students follow the prescriptive procedure of their advisors without assuming responsibility for decision making [10]. The developmental advising models rely on a shared responsibility between the student and the advisor in which the advisor directs the student to proper resources [11].

Literature studies show findings that support both models. For example, Fielstein L. in the research paper titled "Developmental versus prescriptive advising: Must it be one or the other?" stated that: "...intuitive students appeared to endorse the developmental approach to advising. On the other hand, the more 'thinking' students did not value a collaborative relationship and seemed more content with the criteria associated with prescriptive advising" [12]. In general advisors need to look at each student as an individual with individual characteristics.

III. EXPERT SYSTEMS

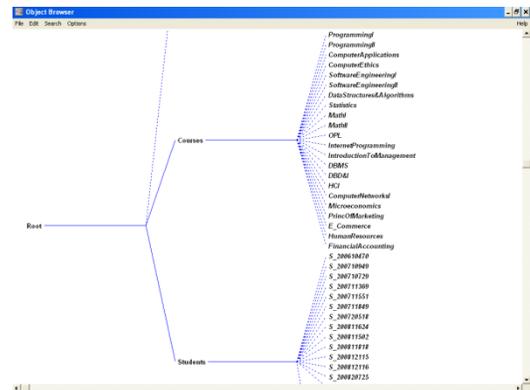
"Expert Systems (ES) are programs that attempt to emulate the behavior of human experts, usually confined to a specific field" [13]. Regarding the domain of academic advising, ES technology seems to be the most successful method of computerization because the dialogue between human advisor and the student can be conveniently emulated by the dialogue between the ES and the student, and the reasoning of the academic advisor can be successfully automated by the reasoning power of ES; particularly the rule-based ES. A rule based ES captures human knowledge using If-Then rules in a rule-based knowledge base. Academic advising process can be successfully modeled in computers as a rule-based expert system since most advising regulations are based on academic 'rules' such as "if you pass course A, then you can register course B" and so on. The proposed system in this research (IS-Advisor) is modeled as an ES with an object-oriented database, thus the main components of IS-Advisor are: The OO database, the rule-based knowledge base, the inference engine, and the user interface. The ES can explain its results by tracing the If-Then rules used to reach the conclusions through a component called the explanation subsystem. The following sections explain the details of the developed ES by explaining its components. The system is developed using Kappa-PC expert system shell [14]. Kappa-PC supports OO modeling which is adopted in this system since OO database allows each student and each course to be modeled as a single object. The ES knowledge was compiled from the university and college regulations, the long term experience of the author as an academic advisor and the Deputy Dean of the College of IT, and discussing the knowledge with students and advisors for their feedback.

IV. THE DEVELOPED EXPERT SYSTEM

A. The OO Database (OODB)

An important objective in database design is to develop an efficient database structure so that data can be stored, accessed, and modified easily. Much of the work in creating an effective database is in the modeling. It is the application domain that determines how the database should be modeled in order to be successful. The nature of university subjects' and students' records (the domain of this research) reveals that the OO model is the most appropriate database modeling method. OO structure allows each course and each student to be constructed as a different object, and the database modeled as a collection of these objects. This structure gives more flexibility to each object to have whatever features (i.e. attributes or fields) required to identify it while maintaining the integrity of the whole system. The database of IS-Advisor consists of the main classes: Courses and Students. Fig. 1 presents a portion of the object hierarchy of IS-Advisor which is the Kappa-PC's graphical representation of the OO database structure. Each study plan course in the database includes the following data: Title, ID, plan semester number (1 to 8), number of pre-requisite courses, List of pre-requisite courses (if any), pre-requisite hours (Some courses have a specified number of hours as their pre-requisite), type of course (There are three types of courses: Compulsory courses, major elective courses, and university elective courses), keywords describing course

contents (e.g. mathematics, programming, algorithm, management, marketing, etc.; these keywords are used to assist students in selecting courses based on their preferences as will be addressed later), course components (theory, lab, and/or tutorial), and course status (offered or not offered; note that fall -or odd- semester courses are offered in fall semester and spring -or even- semester courses are offered in spring semester). Each student object includes the following fields: ID, name, AGPA, passed compulsory courses, passed major elective courses, passed university elective courses, course grades semester-by-semester, earned credit hours, allowable courses, registered courses, course keyword preferences, and load preferences. Note that some data listed above are known and saved in the database (example: offered courses in a particular semester or AGPA of a student) and some data are inferred by the ES (example: lists of allowable and registered courses of a student). It is important to note that the proposed ES is intended to be used for course selection only, and based on courses selected by all students the timing of lectures will be determined manually by the timetabling committee in order to prevent the time conflict between courses. Thus the ES's recommended courses for students will be used as the input for the college timetabling committee. Therefore course timing is not a factor in the current version of the system and a component to automate the determination of lecture timings can be added to the system as a future work.



If: The student's passed hours are greater than or equal to 45 AND Computer Ethics is offered

Then: Add Computer Ethics to the student's allowable courses list.

Rule3:

If: The student passed Computer Applications AND The student's passed hours are greater than or equal to 40 AND Computer Networks I is offered

Then: Add Computer Networks I to the student's allowable courses list.

Rule4:

If: The number of courses in the student's recommended courses list is less than 3

Then: Show the message: Students should register minimum 3 courses and maximum 6 courses. If your case is an exception, please contact your academic advisor.

Student-preference rules are If-Then rules related to preferences input by the student like preferred courses and preferred number of courses that the student is willing to register in a particular semester. As an example of this rule category, consider the following rule:

Rule5:

If: The student's course preference keyword is Management

Then: Mark all allowable courses having Management as a course keyword.

C. The Inference Engine (IE)

Kappa-PC ES shell supports both rule-based reasoning (forward- and backward-chaining) as well as the micro-managing of the reasoning using classical programming techniques (particularly list processing). IS-Advisor's inference engine uses both If-Then rules processing and list processing techniques. The overall reasoning procedure of the IS-Advisor is unique and different than other academic advising expert systems available in literature since it is based on the accumulated academic advising knowledge within the IS department at AUST.

There are three main steps performed in the process of determining the recommended courses for a particular IS student. In Step 1 all courses that are offered and can be registered by the student are stored in a list called Allowable Courses. Step 2 performs the ranking process for the courses contained in Allowable Courses list. The courses are ranked in a descending order as following: (1) Courses that are pre-requisite for subsequent courses (have the highest priority), (2) Courses matching student preferences (in case preferences are given), (3) Courses officially in the current student's registration semester (fall or spring) according to the study plan, (4) Courses whose pre-requisites were passed in the previous semester (in order not to leave a long time gap between a course and its pre-requisite), and (5) Remaining 'equal' allowable courses (if any) are displayed to the user in order to rank them as preferred. The list resulted from this step

is called Ordered Allowable Courses. Step 3 is the filtering step that generates the ordered list of Recommended Courses based on the contents of the list Ordered Allowable Courses. This step follows one of the two advising models: Perspective advising (option 'One-Step Advising' in Fig. 4) or developmental advising (option 'Student's Preferences' in Fig. 4). In 'One-Step Advising' option the list of Recommended Courses is generated as following: (a) Students with AGPA greater than or equal to 3.00 are given the courses ranked from 1 to 6 (from the Ordered Allowable Courses list). (b) Students with AGPA greater than 2.24 and less than 3.00 are given the courses ranked from 1 to 5. (c) Students with AGPA greater than or equal to 2.00 and less than 2.25 are given the courses ranked from 1 to 4. Note that if the remaining number of courses for a student towards graduation is less than the number of courses that can be suggested by the system, then the students is recommended to take the remaining courses only. In "Students' Preferences" option the student is asked to select the number of courses he/she is willing to register (3 to 6 courses) and course keyword preferences. Consequently the list Recommended Courses is prepared as explained in 'One-Step Advising' option above however here level 2 of ranking (courses matching student's preferences) is activated and the number of courses is equal to the number of courses selected by the student (if possible). In addition, more system messages are given here during the user-system interaction in order to guide the student to consider a 'more' suitable course selection.

D. The User Interface and Sample Consultation

Interactions between the users and the system are supported through a friendly graphical user interface running under Windows environment. Fig. 2 shows the main screen of the system where various options are displayed. The Button "Offered Courses" presents all currently offered courses and the button "Study Plan Structure" displays a semester-by-semester structure plan of the IS program. The user can enter the user manual and get more help on using the system by selecting the "Help" button, or exit the system by clicking "Exit". The main option here is "Academic Advising" from which the user is directed to a screen asking for the student ID number (Fig. 3).

After a welcoming message and displaying the currently available student transcript data, the student is given two options as shown in Fig. 4. These two options work as explained while discussing the inference engine above. Fig. 5 shows the result of selecting the option One-Step Advising for a particular student whose AGPA is between 2.25 and 2.99. Note that the Recommended Courses list generated here is ranked by priority from 1 (highest priority) to 5 (lowest priority). The student can click here on the option "Explain!" and get the explanation screen shown in Fig. 6. This explanation screen (related to this 'consultation 1' example) gives the academic reasons for suggesting these courses and for ranking them this way.

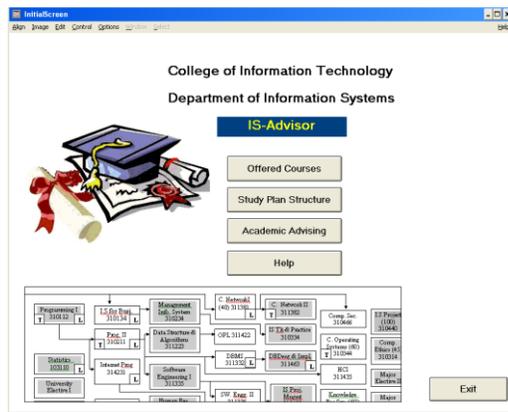


Figure 2. The main screen of IS-Advisor.

Please enter your student ID number and click OK

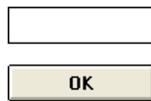


Figure 3. Entering student ID number.

Start Academic Advising

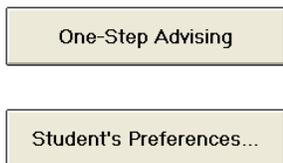


Figure 4. Academic advising models.

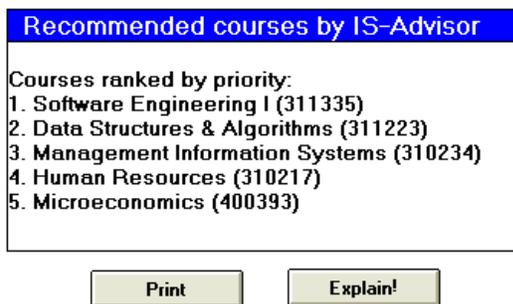


Figure 5. Recommended courses by the system.

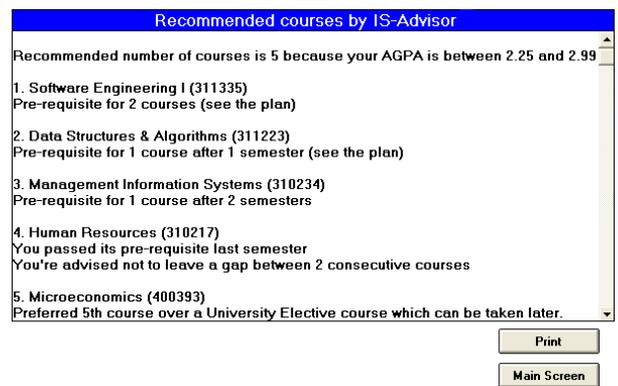


Figure 6. A Sample Explanation Screen (consultation 1).

As a second consultation (consultation 2) example for a student with AGPA greater than 2.99, the option 'Student's Preferences' on the screen of Fig. 4 results in various query screens as shown in Fig. 7, and Fig. 8.

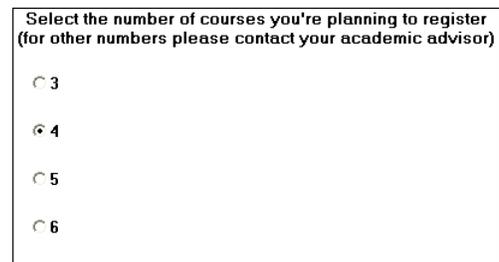


Figure 7. Specifying the number of courses.

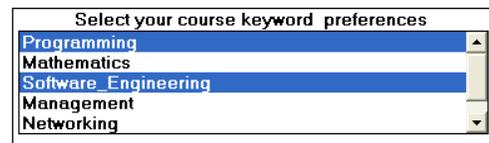


Figure 8. Selecting course keyword preferences.

The result of this consultation is displayed in Fig. 9 below.

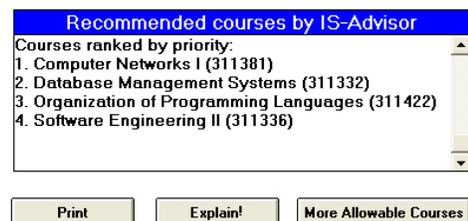


Figure 9. Recommended courses by the system.

The options Explain! And More Allowable Courses give the screens of Fig. 10 and Fig. 11 respectively. Note that for this particular sample advising consultation, the student's preferred number of courses is 4 (see Fig. 7 and Fig. 10). Since the student's AGPA is above 2.99 in this example, the student can register up to 6 courses, this fact is given to the student in Fig. 10 and clicking on 'More Allowable Courses' will suggest to the student the list of allowable courses (Fig. 11). In case the student agrees to increase the number of courses, he/she can increase his preferred number of courses in the screen of Fig. 7.

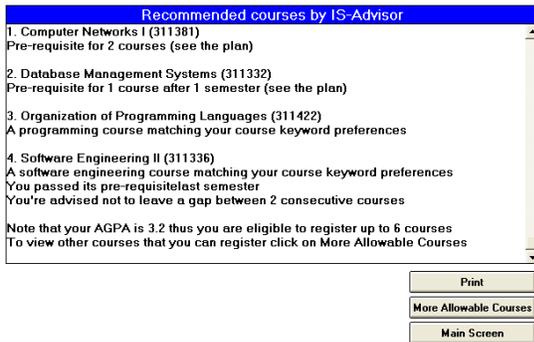


Figure 10. A Sample Explanation Screen (consultation 2).



Figure 11. A Sample List of More Allowable Courses.

V. SYSTEM TESTING

The system was tested by comparing its results with randomly selected actual (i.e. human advisor guided) student registration files for three academic semesters. The results of comparing 130 registration processes show that 93% of the actual cases matched with the system's recommendations. This number shows that the prototype system is successful and going in the correct direction. The reasons for the 7% unmatched results include:

- Warned students: Warned students are students with AGPA less than 2.00 and the regulations require them to repeat some low grade courses to increase their AGPA. Such category of students is outside the scope of the current version of the prototype system. (Planned to be modeled in a modified version).
- Exceptional cases: The current version of IS-Advisor follows strictly the registration rules; however, there are few exceptions which are performed under some special conditions in human-guided advising situations and not counted for in IS-Advisor. To give an example: Some students (because of official medical reports) request to register less than 3 courses in a semester. Such exceptional cases cannot be handled by the current version of the

system, and such students are directed by the system to contact their academic advisors (see Fig. 7).

VI. CONCLUSION AND FUTURE WORK

In this research a prototype expert system with an object-oriented database for student academic advising has been designed and developed. By implementing prescriptive advising model and developmental advising model, the system provides the students and advisors with a useful tool for quick and easy course selection and evaluation of various alternatives. The system has a graphical user interface and simple menus; information is displayed in a way that is familiar for both advisors and students. The present state of the system was discussed and illustrated with sample consultations. The system is successful and efficient. System testing revealed that 93% of academic advising test cases show an agreement between the system advising in course selection and human advising. Enriching the system by adding more data and knowledge rules is a continuous process. Many parts of the system can be improved further and some issues deserve future work, among them:

1. Currently the system operates as a stand-alone system. It would be better to connect IS-Advisor with the university's student information system. This will automate the process of importing students' data.
2. Advising of students with exceptional cases and warned students is outside the scope of the current system. This feature can be added in future developments of the system.
3. The system can be improved so that it automates the determination of lecture timings for courses based on the courses recommended for all students so that time conflict between lectures is prevented.

REFERENCES

- [1] T. Feghali, I. Zbib, and S. Hallal, "A web-based decision support tool for academic advising", *Educational Technology & Society*, Vol. 14, No. 1, pp. 82-94, 2011.
- [2] A. N. Nambiar and A. K. Dutta, "Expert system for student advising using JESS", *International Conference on Educational and Information Technology (ICEIT)*, China, September 17-19, 2010.
- [3] F. Albaloooshi and S. Shatnawi, "HE-Advisor: A multidisciplinary web-based higher education advisory system", *Global Journal of Computer Science & Technology*, Vol. 10, No. 7, pp. 37-49, September 2010.
- [4] R. Zucker, "ViCurriAS: A curriculum visualization tool for faculty, advisors, and students", *Journal of Computing Sciences in Colleges*, Vol. 25, No. 2, pp. 138-145, December 2009.
- [5] D. Pokrajac and M. Rasamny, "Interactive virtual expert system for advising (InVESTa)", *36th Annual ASEE/IEEE Frontiers in Education Conference*, San Diego, CA, USA, October 27-31, 2006.
- [6] K. Kowalski, "On-line advising with JavaScript rule-based system", *Proceedings of Society for Information Technology & Teacher Education International Conference*, Chesapeake, VA, USA, pp. 2922-2927, 2004.
- [7] R. M. Siegfried, A. M. Wittenstein, and T. Sharma, "An automated advising system for course selection and scheduling", *Journal of Computing Sciences in Colleges*, Vol. 18, No. 3, pp. 17-25, February 2003.
- [8] A. M. Wittenstein and T. Sharma, "FROSH2: An expert system for freshman advisement", *Proceeding of the National Conference on Undergraduate Research (NCUR)*, University of Wisconsin, Whitewater, Wisconsin, USA, April 25-27, 2002.

- [9] M. Patankar, "A rule-based expert system approach to academic advising", *Innovations in Education and Training International*, Vol. 35, No. 1, pp. 49-58, February 1998.
- [10] B. B. Crookston, "A developmental view of academic advising as teaching", *Journal of College Student Personnel*, Vol. 13, No. 1, pp. 12-17, 1972.
- [11] C. M. Chando, "Predicting advising style preference from student characteristics", Doctoral dissertation, University of Memphis, UAS, 1997.
- [12] L. L. Fielstein, "Developmental versus prescriptive advising: Must it be one or the other?", *The National Academic Advising Association (NACADA) Journal*, Vol. 14, No. 2, pp. 76-79, 1994.
- [13] R. J. Schalkoff, "Intelligent systems: Principles, paradigms and pragmatics", Jones & Bartlett Publishers, 2009.
- [14] Intellicorp, *Kappa-PC 2.4 ES shell manuals*, Intellicorp, Inc., USA, 1997.

AUTHORS PROFILE

Dr. Ayman Al Ahmar is Assistant Professor and the Deputy Dean of the College of Information Technology, Ajman University of Science and Technology, UAE. He received his B.Sc. (1994), M.Sc. (1997), and Ph.D. (2001) degrees from Middle East Technical University (METU), Ankara, Turkey. His current research interests include Artificial Intelligence, Software Engineering, and Engineering Information Systems. He is a member of IEEE and IEEE Computer Society.