Athens Journal of Education - Volume 10, Issue 1, February 2023 – Pages 101-116

An Interactive Decision Support System for College Degree Planning

By Sobitha Samaranayake[*], Athula D. A. Gunawardena[±] & Robert R. Meyer[°]

Many students in the Unites States enter college without having decided on a focus for their studies, and thus are faced with choosing from a large number of potential majors and associated very complex sets of degree requirements which can include many courses in other areas of study. Academic advisors use academic planning tools to help students make decisions about class schedules, selecting an academic major or minor, planning for graduation, and many other academic related activities. There is a dearth of decision support systems for degree planning, mainly due to the complexity of degree requirements, and thus many existing academic planning tools utilize static documents or PDF files for displaying information pertaining to degree requirements and course prerequisites. This work considers the complexity of degree requirements and presents the design and implementation of an efficient interactive decision support system that helps students explore degree completion paths.

Keywords: degree planning, academic advising, academic decision making, decision support system, college degree planning

Introduction

Many universities employ direct communications between academic advisors and students as the primary advising system. Academic advisors are either faculty or professional advisors employed by an academic unit. During an advising session, advisors use academic planning tools to help students make decisions about class schedules, selecting an academic major or minor, planning for graduation, and many other academic related activities. Many existing academic planning tools utilize static documents or PDF files for displaying information pertaining to degree requirements and course prerequisites. Nevertheless, current students are digital natives who expect advising resources to be online and available in a user-friendly format. Due to the complexity of degree requirements and prerequisite dependencies, it is a challenge to develop and maintain systems that can analyze students' academic progress toward a degree.

Degree requirements vary in structure from one academic institution to another, and some of the requirements can be considerably complex. Most of the degree requirements are specified in terms of number of units, credits, or courses that must be taken to satisfy each requirement. Requirements may refer to additional attributes such as course level (lower-division vs. upper-division) or a student's

[*]Associate Professor, University of Wisconsin-Whitewater, USA.
[±]Professor, University of Wisconsin-Whitewater, USA.
[°]Professor Emeritus, University of Wisconsin-Whitewater, USA.

https://doi.org/10.30958/aje.10-1-6 doi=10.30958/aje.10-1-6

minimum grade point average (GPA). In addition, some of the courses may not be taken until a minimum number of units has been earned. Courses may only count once in the major or minor, either as a required course or as an elective, but not as both. There may be hidden prerequisites (i.e., prerequisites of a prerequisite course that may not be explicitly listed as a part of any other requirements) and other requirements such as selecting major/minor emphasis areas. Major/minor requirements are often defined in terms of a set of course requirements that covers specific subjects or areas of knowledge. Many courses specify prerequisites that are outlined using a list of courses, all of which or a subset of which must be completed successfully in order to satisfy the prerequisites. In addition, a few of the prerequisites may be tied to course grades to ensure students acquire the necessary knowledge for getting the maximum benefit from the next course. Since it is difficult to define a standard format for representing degree requirements, most of the existing academic planning tools use custom-made systems that are difficult to scale up.

This paper will focus on developing a Decision Support System (DSS) for degree planning. First, we identify possible degree requirement types and use a data structure that can represent such requirements. Then we create a DSS that helps students select a major along with the courses needed to satisfy additional degree requirements beyond those for the major. These additional requirements often include the selection and completion of a minor in a subarea and the completion of so-called general education (GE) requirements spread across a wide variety of topics. To demonstrate the potential of the decision support system, we describe an implementation based on degree requirements and majors/minors offered at University of Wisconsin-Whitewater (UWW), a mid-sized U.S. public institution with about 11,000 students, which has 574 pages of course information and degree requirements covering more than 150 possible majors and about 120 possible minors. A student generally has a choice of focus areas within the major and there are about 15,000 different course sequences that meet the Computer Science major requirements for a BS degree at UWW. In addition to deciding upon and meeting the requirements of both a major and minor, a student wishing to complete a degree is faced with the challenge of selecting courses that meet general education requirements that specify the minimum number of credits needed in a variety of additional subareas, such as Communication Skills, Calculation Skills, Quantitative and Technical Reasoning, Cultural Heritages, World of Ideas, Communities, Physical Health and Well-Being, and Racial/Ethnic Diversity.

Literature Review

It is estimated that 20 to 50 percent of students enter college in the United States as undecided or undeclared, without having decided on a focus for their studies, and more than 50 percent of students change their major at least once before graduation (Gordan, 2015). There are many reasons for the indecision of college students, including decision-making difficulties, gender differences, cultural differences, indecisive students, and types of career indecision (Soria & Stebletin, 2013). Many undecided students are skeptical about how their personal strengths and limitations relate to coursework required in particular majors. Furthermore, choosing a major may depend on personal interests, job market, program cost, or the complexity of graduation requirements (Pozzebon, Ashton, & Visser, 2014). Many students make initial choices based on their interests but change their majors because of changing career interests or academic interests (Bullock-yowell, McConnell, & Schedin, 2014). Wang and Orr (2019) used data analytics to inform decision-making in academic advising and supporting undecided students' academic success. Halasz and Bloom (2019) examined the resources students identified as most valuable and the factors most influential in their decision to transition out of majors. Streufert (2019) investigated the effects of alternative advising, such as coping with loss, managing anxiety, and restoring self-efficacy, and on renewing focus of undeclared students so that they stay focused and graduate on time. Marade and Brinthaupt (2018) examined reasons for students to change a college major. Iyer and Variawa (2019) used supervised Machine Learning classification algorithms to analyze the potential inclination of the undecided/undeclared first-year engineering students at the University of Toronto. Glaessgen et al. (2018) examined the challenges and experiences of first-generation undecided students transitioning to a new and unfamiliar academic environment. The relationship between academic major change and ten personality traits (the five broad and five narrow traits) was investigated in Foster (2017).

There has been an interest in developing interactive and visualization tools for academic curricula and advising. Marques, Ding, and Hsu (2001) presented a design and development of a web based academic advising system. Gutiérrez et al. (2018) presented a design and implementation of a Learning Analytics Dashboard for Advisers, LADA, to support the decision-making process of academic advisers through comparative and predictive analysis. Moreno, Bischof, and Hoover (2012) presented an interactive visualization tool for exploring course dependencies between courses. Dechter (2007, 2009) introduced an integer linear programing model for finding academic plans that would satisfy a given set of graduation requirements and other constraints in the shortest possible time. Kowalski and Ealy (1991) used artificial intelligence to design an expert system for the advisement of two-year community college students.

Prerequisite visualization tools are extremely useful for preparing academic plans. Zucker (2009) presented a curriculum visualization tool for developing and arranging the flow of courses for a particular program. Aldrich (2014) used the overall topology of the courses at Benedictine University to propose a directed acyclic graph for representing prerequisite relations where each edge represents a logical relationship such as all of or one of. Chen and Siyuan (2017) presented an interactive course selection scheme with prerequisite hierarchy. Their work includes visualization of all of, one of, or either-or logical relationships of courses offered at University of British Columbia. Samaranayake and Gunawardena (2020) introduced a graphical data visualization tool that enables students and advisors to easily understand course prerequisite structure and to readily determine paths that lead to the satisfaction of degree requirements.

There has been an interest in designing DSS for course planning. Siddiqui, Raza, and Tariq (2018) introduced a web-based group DSS for academic term preparation at a business college of a large Middle Eastern university. Roushan et al. (2014) presented a DSS for course planning. Miranda, Rey, and Robles (2012) developed a web-based DSS for course and classroom scheduling. Oladokun and Oyewole (2015) presented a DSS for university admission seekers. Al-Qaheri, Hasan, and Al-Husain (2011) presented a DSS for course scheduling. Most of these decision support systems are designed for course scheduling.

Students considering academic degrees in institutions outside the United States (US) often have less flexibility in selecting elective courses in those institutions, so they often feel challenged when addressing course plan flexibility in a US college or university. Thus, in addition to cultural and social adjustments, many international students are additionally stressed by the need to adjust to a new academic environment (Mesidor & Sly, 2016; Rienties et al., 2012). The process of registering for classes is often different from experiences that the international students have had at academic institutions in their country of origin. Students may also struggle with choosing a major. Some students may want to complete their degree requirements early. International students may come to the new university or college with a predetermined academic plan, but they are often not well informed about the US curriculum and may want to change their major after they are exposed to different areas of study and new career opportunities. The DSS helps such students discover their own preferences for courses of study and empowers them to visualize degree paths based on their interests and skills.

The complexity of degree requirements, prerequisite dependencies, and user preferences make the automated degree planning problem an inherently hard combinatorial optimization problem. Due to its complexity, the present commercial degree planning systems have avoided automation and limited their features to semester by semester drag and drop course selections. Integer programming models to generate degree plans with simplified requirements have been proposed in Dechter (2007, 2009). Although these models are useful for calculating lower bounds for comparison, they are intractable for practical systems which deal with complex degree planning problems with various constraints and are expected to provide fast solutions. This work considers the complexity of degree requirements and presents the design and implementation of an efficient DSS that helps students explore degree completion paths.

Method

Most college degree requirements are specified in terms of number of units, credits, or courses that must be taken to satisfy each requirement. First, we define a suitable data structure for evaluating degree requirements. A typical degree requirement belongs to one of the following categories:

- Type A: complete k courses from a set of p courses where $1 \leq k \leq p$

- Type B: complete at least m courses/units, but no more than n courses/ units from a set of p courses where $0 \leq m \leq n \leq p$
- Type C: complete k units from a set of p courses where $1 \leq k \leq p$
- Type D: combination of Type A, Type B, and/or Type C requirements

Type A, Type B, and Type C degree requirements are relatively easy to implement but Type D requirements are often complex and difficult to implement. There may be other requirements, such as GPA requirements, minimum number of credits/units needed to complete, internships, capstone projects, etc. Samaranayake and Gunawardena (2020) introduced a generic requirement type, named basic requirement, that is able to represent most of the college degree requirements. A basic requirement is a 7-tuple (A, T, C_{min}, C_{max}, U_{min}, U_{max}, δ), where A is a set of objects, T is the type of requirement (select number of objects, select number of units, etc.), C_{min} is the lower bound of courses, C_{max} is the upper bound of the courses, U_{min} is the lower bound of the units, U_{max} is the upper bound of the units, and $\delta: A \rightarrow \{1, 0\}$ is a function such that $\delta(A) = 1$ if A is a credit-bearing set of objects and $\delta(A) = 0$ otherwise. Any Type A, Type B, or Type C degree requirement can be represented using a combination of basic requirements. In this work, we use the basic requirements model to represent any degree requirement found in a college catalog.

Exploring Major and Minor Paths

When choosing a major or a minor, a student would normally have completed some courses that may count toward satisfying requirements for some of the majors or minors. In this section we define an efficient process for mapping completed courses to prospective majors or minors.

Let S be the set of all courses, M be the set of all majors, and N be the set of all minors offered by a degree-granting institution. A major path is a minimal set of courses that satisfies all the requirements of a college major and a minor path is a minimal set of courses that satisfies all the requirements of a college minor. It is possible to generate all possible major paths for the set of all the majors, M, offered at a given institution.

There are about 15,000 different course sequences that meet the Computer Science major requirements for a BS degree at UWW. Hence, there could be millions of possible major paths so it would be a daunting task to create and maintain such a collection of possible major paths. Instead, for a given student we first identify a subset M^C of M consisting of all the possible majors that would allow the student to complete the degree requirements in a timely manner, based on already completed courses.

Let $V = \{(s, m) \mid s \in S, m \in M\}$ be the set of all ordered pairs that connect courses to majors. In many universities, a single department offers majority of the courses satisfying a given major. Therefore, V is a relatively small subset of $S \times M$.

Let $M_i^R = \{R_{i1}, R_{i2}, \ldots, R_{ir}\}$ be the set of requirements for the major $M_i \in M$, where each requirement in R_{ij} is a predicate defined on a subset of S.

Let $C = \{c_1, c_2, \dots, c_n\}$ be the set of courses completed (taken/waived) by a particular student and $V_c = \{(s, m) \mid s \in C, m \in M\}$ be the subset of V that defines a mapping between the set of completed courses C and the set of majors M.

Let M^c consists of majors accepting some or all courses in C and N^c consists of minors accepting some or all courses in C.

Using the set V_c, we can find an ordered list of majors $M^{c_i} = \{M_{i0}, M_{i1}, M_{i2}, \dots, M_{ij}\}$ where the course c_i satisfies one or more requirements of each M_{ik} in M^{c_i}. Then $M^C = \bigcup_{i=1}^{n} M^{c_i}$. Hence, we can easily identify the set M^C for a given set of completed courses C. The set M^C consists of all the possible majors for which the student has completed at least one course, thus the set M^C helps students find the major or majors that require the least number of units to complete. Although there could be majors/minors for which the student has taken no courses but which require fewer units for completion, since the student has some course experience in an area, that area is a suitable area for consideration.

Requirements for each major $M_i \in M^C$ can be expressed using a set of basic requirements and then we can use completed courses to evaluate completion levels of requirements for majors in M^C. We use the same process for exploring a set of possible minors, N^c of N, for identifying possible major/minor combinations for each major $M_i \in M^C$.

Upon exploring possible major/minor combinations, we can identify a set of courses needed to complete the remaining degree requirements while minimizing the total number of remaining credits needed for each major/minor combination. Such a process allows us to produce the shortest path in terms of credits needed for graduation.

Data Visualization Method

Major requirements, prerequisite conditions, and course rotations must be taken into consideration when planning courses for the completion of the degree. In general, prerequisites are completed/waived/transferred courses or test scores that must be completed before taking a specific course, and some of the prerequisites are tied to course grades and courses from other disciplines. Table 1 displays an example of the prerequisite conditions for a sample set of seven courses.

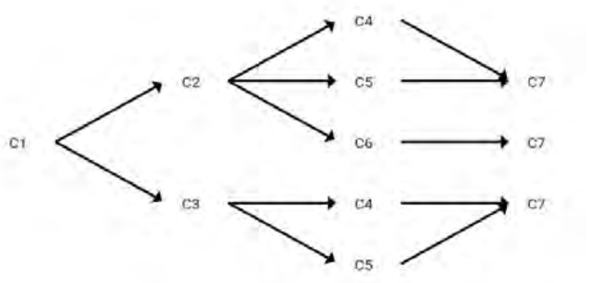
Table 1. Prerequisite Conditions for a Sample Set of Courses

Courses	**Prerequisites**
$C2, C3$	$C1$ with a grade of C or better
$C4, C5$	$C2$ with a grade of C or better or C*3* with a grade of B or better
$C6$	$C2$ with a grade of C or better
$C7$	$C6$ or (C4 and $C5$), with a grade of C or better

Suppose a student wishes to select a major path that includes the course $C7$. In order to check if course prerequisites for $C7$ are satisfied or to find the shortest path for satisfying the prerequisites, it would be extremely helpful if the prerequisite structure can be visualized using a directed graph.

We use an adjacency matrix of a directed graph $D(V, E)$to represent course prerequisite structure (CPS) where nodes (V) represent courses and edges (E) represent prerequisite relationships. Table 1 contains information needed to define an adjacency matrix of the directed graphs for the sample set of courses. Figure 1 shows a directed graph depicting the prerequisite structure described in Table 1.

Figure 1. Prerequisite Conditions for a Sample Set of Courses

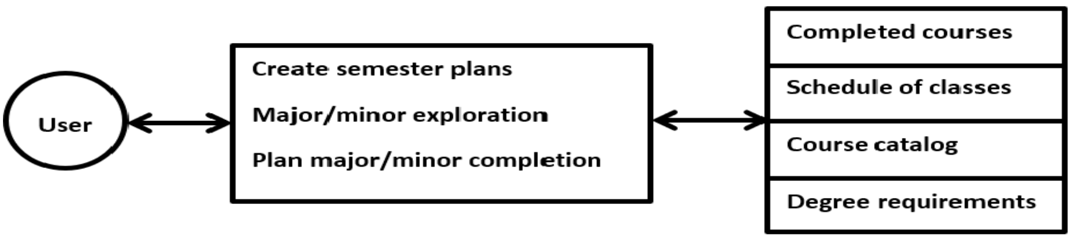


Existing degree planning tools show CPS using static data structures similar to Table 1. We utilize novel visualization tools introduced in Samaranayake and Gunawardena (2020) to visualize CPS for degree paths as a directed graph and then dynamically update the prerequisite structure using completed courses.

Implementation

The current implementation of the DSS is based on degree requirements and majors/minors offered at University of Wisconsin-Whitewater. Degree requirements are often specified in terms of course offerings. Therefore, each requirement is stored in a database using an appropriate format suitable for our algorithms. In order to speed up the process, we use the basic requirement type for storing each type of basic requirement. We use a relational database to store degree requirements and course information. A set of completed/waived courses is needed to explore major/minor combinations. Figure 2 shows the architecture diagram of the DSS system.

Figure 2. Architecture Diagram of the DSS System



Exploring Major and Minor Paths

The UW-Whitewater database consists of 154 majors, 121 minors, and 3300 courses. Students must complete at least 120 units. Some of the majors require an approved minor. Courses counted toward the major cannot be counted for a minor. Students must satisfy general education requirements. In addition, some majors require students to complete a separate mathematics requirement.

First, completed courses (Figure 3) are mapped to majors to identify the set M^C. Then, requirements for each major $M_i \in M^C$ are evaluated using the completed courses to produce the report in Figure 4. The output includes a list of possible majors. For each such major, it also includes a list of courses satisfying its requirements and the number of credits needed to complete the major. The first semester of the course history includes a list of courses completed, transferred, or waived by the end of the first semester.

Furthermore, the DSS provides a mechanism for displaying progress of the requirements for each possible major and minor. Figure 5 includes a list of requirements for the computer science major, general emphasis (BS).

Figure 3. Sample Report of Completed Courses

Course History:

Term	Subject	Cat	Title		Grade	Units Taken	Points	Type
Fall 18	AMERIND	102	INTRO AMER INDIAN STUDIES	GHDV	BC	3.00	7.500	TR
Fall 18	ANTHROPL	9999S	ANTHROPL ELECTIVES - GS	GEGS	C	3.00	6.000	TR
Fall 18	BIOLOGY	141	INTRODUCTORY BIOLOGY I	GEGL	BC	5.00	12.500	TR
Fall 18	BIOLOGY	257	INTRODUCTION TO ECOLOGY		A	3.00	12.000	TR
Fall 18	COMM	999	COMM ELECTIVES		AB	3.00	10.500	TR
Fall 18	COMM	9999H	COMM ELECTIVE - GH	GEGH	A	3.00	12.000	TR
Fall 18	ENGLISH	90W	FUNDAMENTALS OF ENGL - WAIVER		T	0.00	0.000	TR
Fall 18	ENGLISH	101W	FRESHMAN ENGLISH - WAIVER		T	0.00	0.000	TR
Fall 18	ENGLISH	102	WRITING/READING/RESEARCH		A	3.00	12.000	TR
Fall 18	ENGLISH	274	CREATIVE WRITNG	GEGH	A	3.00	12.000	TR
Fall 18	GENED	120	HISTORICAL PERSPECTIVES	GEGH	A	3.00	12.000	TR
Fall 18	GEOGRPY	9999L	GEOGRPY ELECTIVES - GL	GEGL	AB	4.00	14.000	TR
Fall 18	GEOLGY	999	GEOLGY ELECTIVES		A	3.00	12.000	TR
Fall 18	HISTRY	124	AMERICAN HISTORY TO 1877	GHDV	A	3.00	12.000	TR
Fall 18	HISTRY	154	WESTERN CIVILIZATION	GEGH	A	3.00	12.000	TR
Fall 18	MATH	40W	PRE - ALGEBRA - WAIVER		T	0.00	0.000	TR
Fall 18	MATH	41W	BEGINNING ALGEBRA - WAIVER		T	0.00	0.000	TR
Fall 18	MATH	141W	INTERMEDIATE ALGEBRA - WAIVER		T	0.00	0.000	TR
Fall 18	MATH	254	CALC/ANALYTIC GEOMETRY II		B	5.00	15.000	TR
Fall 18	PEGNRL	160	BEGINNING TENNIS	GEGP	A	1.00	4.000	TR
Fall 18	PHILSPHY	241	INTRO TO PHILOSOPHY	GEGH	A	3.00	12.000	TR
Fall 18	PSYCH	211	INTRODUCTORY PSYCHOLOGY	GEGS	D	3.00	3.000	TR
Fall 18	BIOLOGY	142	INTRODUCTORY BIOLOGY II	GEGL	C+	5.00	11.650	EN
Fall 18	COMPSCI	172	INTRODUCTION TO JAVA	GEGM	A-	3.00	11.010	EN
Fall 18	MATH	280	DISCRETE MATHEMATICS		C-	3.00	5.010	EN
Fall 18	MATH	342	APPLIED STATISTICS		C-	3.00	5.010	EN
Spring 19	COMPSCI	220	INTERMEDIATE JAVA		W	3.00	0.000	EN
Spring 19	COMPSCI	271	ASSEMBLY PROGRAMMING		F	3.00	0.000	EN
Repeat Code: EXPG - First Attempt Exclude from GPA								
Spring 19	ENGLISH	370	ADVANCED COMPOSITION		B	3.00	9.000	EN
Spring 19	MATH	355	MATRICES/LINEAR ALGEBRA		B-	3.00	8.010	EN
Fall 19	COMPSCI	220	INTERMEDIATE JAVA		C	3.00	6.000	EN
Fall 19	COMPSCI	271	ASSEMBLY PROGRAMMING		F	3.00	0.000	EN
Repeat Code: CRED - Repeated for Credit								
Fall 19	COMPSCI	381	JAVASCRIPT AND DHTML		B+	3.00	9.990	EN
Fall 19	COMPSCI	382	SERVER-SIDE SCRIPTING		F	3.00	0.000	EN
Fall 20	COMPSCI	223	DATA STRUCTURES		W	3.00	0.000	EN
Fall 20	COMPSCI	353	CYBERSECURITY LAW		B+	3.00	9.990	EN
Spring 21	COMPSCI	223	DATA STRUCTURES			3.00	0.000	IP

Figure 4. Initial Portion of Sample Output of Possible Majors

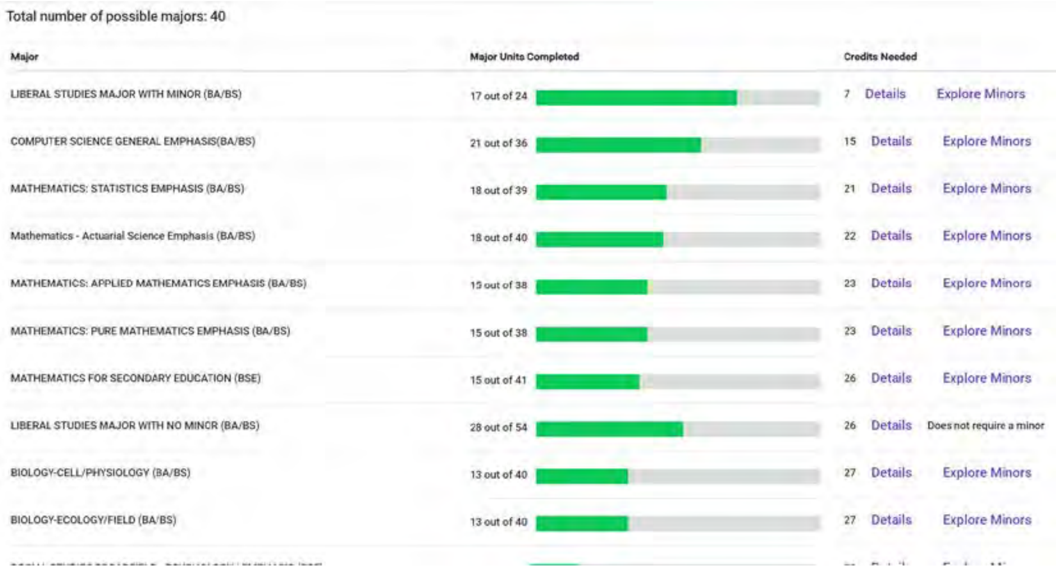
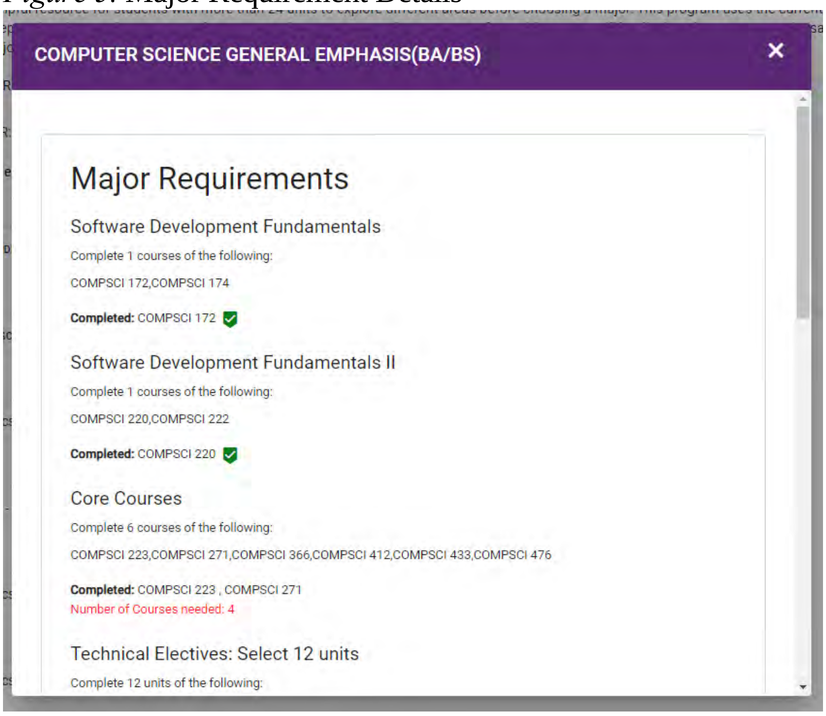
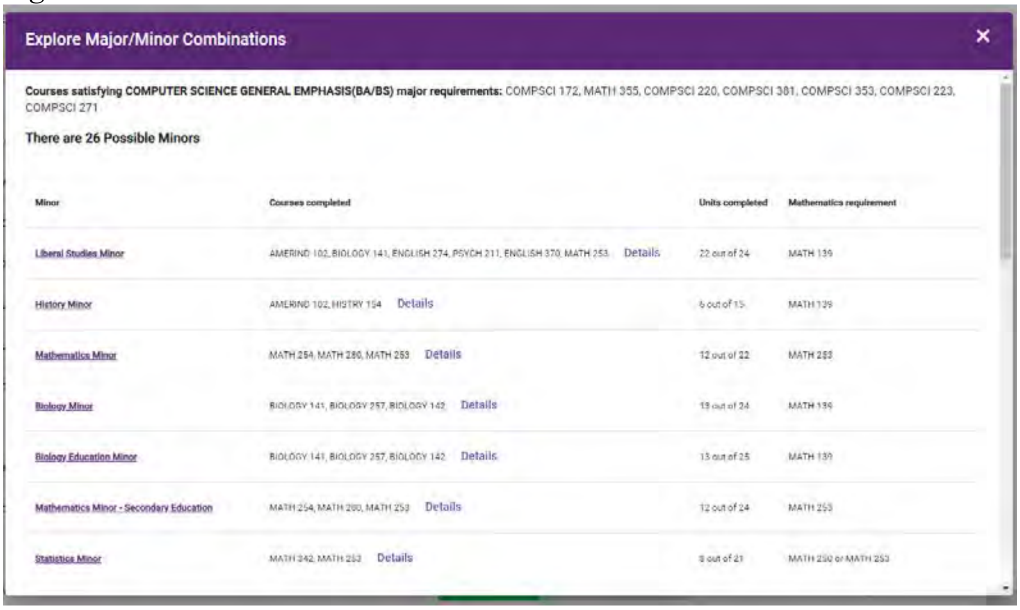


Figure 5. Major Requirement Details



In this sample execution, there are 40 possible majors in the set M^C. The DSS also provides a mechanism to explore minors for those majors that require an approved minor. Figure 6 includes a list of possible minors if the student chooses computer science general emphasis (BS) as the major. There are 26 possible major/ minor combinations for the selected major.

Figure 6. Initial Portion of List of Possible Minors



Semester Planning

The DSS system helps students select courses for the next semester by visualizing the available courses, based on the CPS. Figure 7 shows the CPS for computer science major general emphasis at UWW, prior to completing any of the courses in the major. Nodes with a stack of courses represent prerequisite courses where only one of the courses is needed to be taken to satisfy the prerequisite. If two or more arrows are pointing to the same child node, then each of the prerequisite relationships must be satisfied for the course list attached to the child node to be available. CPS is extremely useful for identifying any bottleneck conditions that may prolong the graduation date. For example, *Compsci* 223 and *Compsci* 271 are prerequisite courses for many of the 300-level or higher computer science courses. Hence, their prerequisites must be completed as soon as possible to minimize the time to complete the degree.

Figure 7. Course Prerequisite Structure for Computer Science General Emphasis

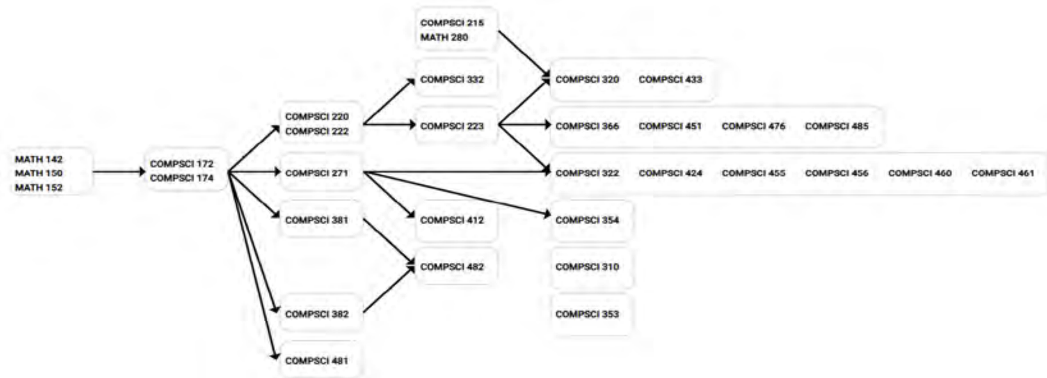
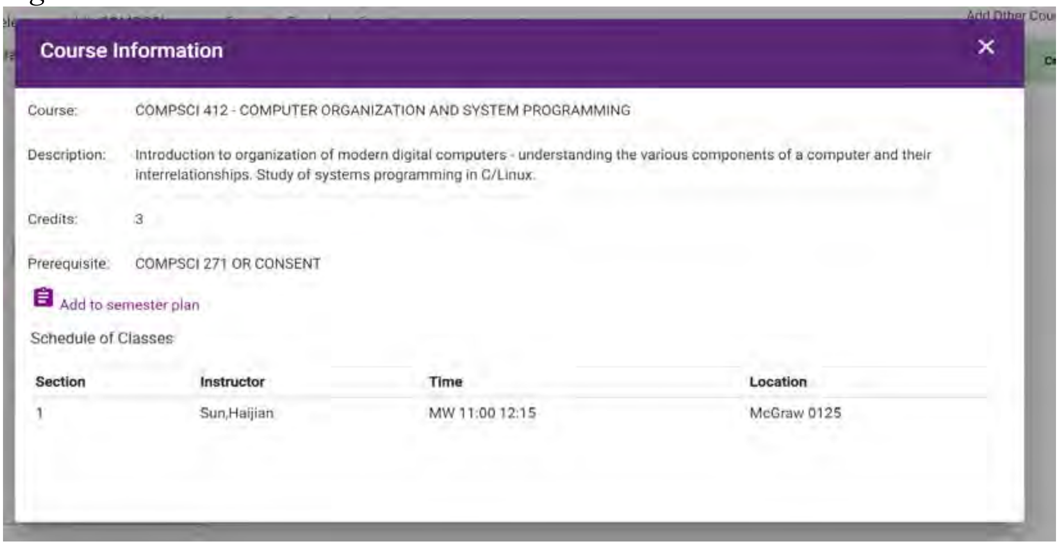


Figure 8 shows the semester plan page of the DSS. The semester planning page consists of a course prerequisite structure and a list of degree requirements for a selected major. The list of degree requirements includes an indication of whether each requirement has been satisfied and courses credited towards satisfying each requirement. The CPS is updated dynamically to narrow down the major path choices, based on the completed and planned courses. Course grades are displayed where * represents grades for the courses that are in progress and the courses shown in purple are the courses planned for the next semester. Green arrows point to courses that are available to take in the next semester, based on the completed courses. The courses shown in orange are the courses whose prerequisites are satisfied and available for planning the next semester.

Figure 8. Semester Plan Page

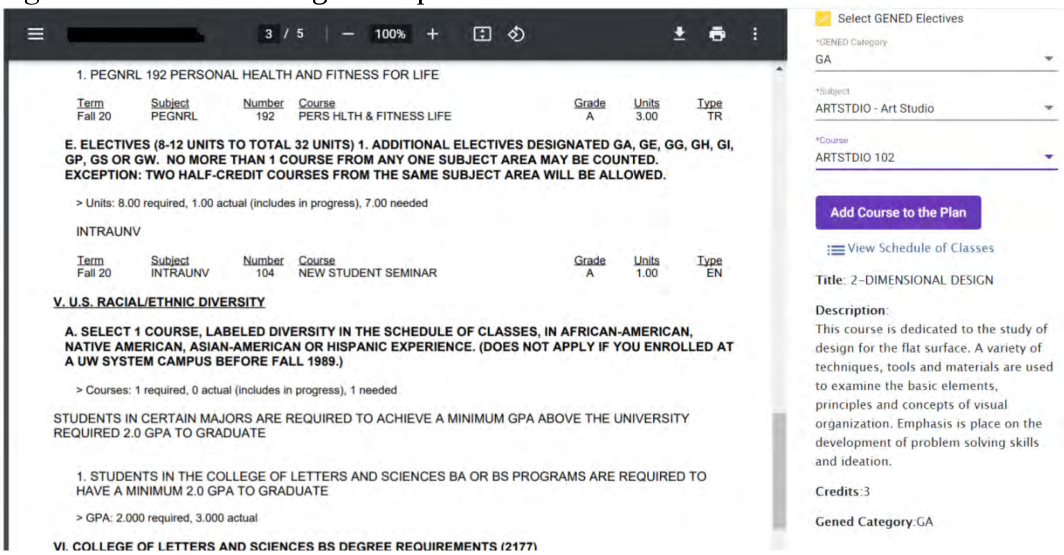
A dialog box is linked to each course shown in orange for students to view course information and course schedule. Students may use the CPS to select any of the courses shown in orange and add to the semester plan. Figure 9 shows the course information dialog box.

Figure 9. Course Information



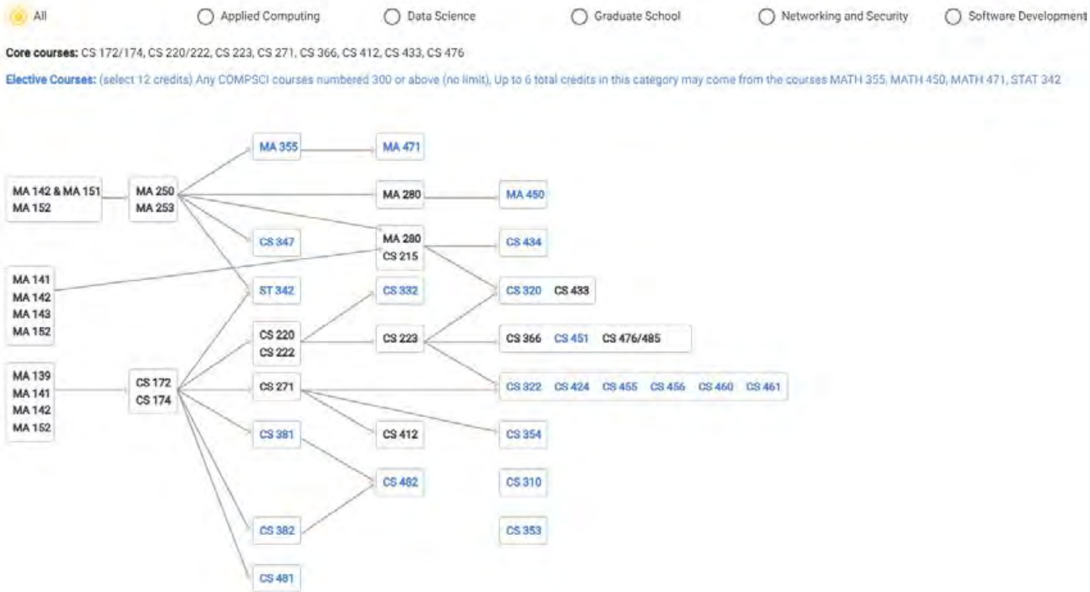
The DSS allows students to view the official academic progress report adjacent to course information (Figure 10) and then plan courses interactively, based on the degree requirements yet to be satisfied (requirements in bold).

Figure 10. Academic Progress Report



There are many advantages inherent in the use of data visualization tools. Figure 12 illustrates the ability to visualize the complete course structure for the computer science comprehensive emphasis. Courses appearing in black are the required courses and their prerequisites, and the courses appearing in blue are the elective courses from which students must select four courses.

Figure 11. Course Prerequisite Structure for Computer Science Comprehensive Emphasis



There are many ways of choosing the required four courses out of twenty-two possible electives. The DSS system uses a dynamic directed graph to help students select elective courses based on their interests and career goals. The visualization in Figure 12 also helps students select the four elective courses needed for a degree in software engineering.

Students who are interested in pursuing a career in applied computing, data science, network and security, or software engineering may select any of the radio buttons at the top (Figure 11) to view a course structure graph that helps them select electives based on their career choices.

Figure 12. Dynamic Directed Graph Depicting Career-Based Elective Courses

Conclusion

This work describes the basic requirement model for representing degree requirements and presents the design and implementation of a DSS that helps students explore possible major/minor combinations and create semester plans.

The current implementation is based on the UW-Whitewater course catalog of 154 majors, 121 minors, and 3,300 courses. Since requirements vary from one major/minor to another, a typical degree mapping application uses separate files for processing individual major/minor requirements. As such, it is a daunting task to create and maintain such an application and there are no such applications for mapping courses to multiple majors/minors. The data structure we propose in this work is capable of using a single application for processing every major/minor requirement. Hence, this application eliminates the painstaking what-if analysis of static data for exploring possible majors/minors. Furthermore, this application helps students quickly compare major/minor combinations. Although the implementation is based on the UW-Whitewater course catalog, the system can easily be extended to course catalogs at other universities. Many universities use degree requirements that can be represented by the proposed data structure and we are in the process of applying the DDS system at a few other universities.

There is a dearth of data visualization tools for displaying college degree-planning information, mainly due to complexity of degree requirements. The new data structure used in this research is extremely useful for creating data visualization tools. We have already developed a dependency evaluation and visualization tool using a version of the aforementioned data structure. We plan to expand the current DSS to create an interactive degree audit system and degree personalization system that helps students create degree plans that align well with their professional interests.

Future Work

The main purpose of this work is to introduce a DSS system that uses a new data model to reduce the complexity of mapping completed courses to requirements. We are in the process of introducing an interest-aligned degree planning and career path selection system that uses the new data structure and dynamic directed graphs to guide students to select degree paths that align well with their interests, personalities, and aptitudes. The features of the enhanced DSS system are based on Holland's theory which classifies people using six types of traits: Realistic, Investigative, Artistic, Social, Enterprising, and Conventional (RIASEC).

References

Al-Qaheri, H., Hasan, M. K., & Al-Husain, R. (2011). A Decision Support System for a Three-Stage University Course Scheduler with an Application to College of Business Administration, Kuwait University. *International Journal of Data Analysis and Information Systems*, *3*(Dec), 95-110.

Aldrich, P. R. (2014). *The Curriculum Prerequisite Network: A Tool for Visualizing and Analyzing Academic Curricula*. arXiv Preprint arXiv: 1408.5340.

Bullock-yowell, E., McConnell, A. E., & Schedin, E. A. (2014) Decided and Undecided Students: Career Self-Efficacy, Negative Thinking, and Decision-Making Difficulties. *NACADA Journal*, *34*(1), 22-34.

Chen, J., & Siyuan, H. (2017). *UBCourse Vis*. Retrieved from: https://www.cs.ubc.ca/~tmm /courses/547-17F/projects/jiahong-siyuan/report.pdf. [Accessed 17 November 2021]

Dechter, A. (2007). Model Based Student Academic Planning. *International Journal of Applied Management and Technology*, *5*(1), 87-104.

Dechter, A. (2009). Facilitating Timely Completion of a College Degree with Optimization Technology. *Association for the Advancement of computing in Educational Journal*, *17*(3), 215-229.

Foster, N. A. (2017). Change of Academic Major: The Influence of Broad and Narrow Personality Traits. *College Student Journal*, *51*(3), 363-379.

Glaessgen, T. A., MacGregor, C., Cornelius-White, J. H., Hornberger, R. S., & Baumann, D. M. (2018). First-Generation Students with Undecided Majors: A Qualitative Study of University Reacculturation. *NACADA Journal*, *38*(1), 22-35.

Gordan, V. N., & Steele, G. E. (2015). *The Undecided College Student: An Academic and Career Advising Challenge*. 4th Edition. Charles C. Thomas Publisher.

Gutiérrez, F., Seipp, K., Ochoa, X., Chiluiza, K., De Laet, T., & Verbert, K. (2018). LADA: A Learning Analytics Dashboard for Academic Advising. *Computers in Human Behavior, 107*(Jun), 105826.

Halasz, H. M., & Bloom, J. L. (2019). Major Adjustment: Undergraduates' Transition Experiences when Leaving Selective Degree Programs. *NACADA Journal*, *39*(1), 77-88.

Iyer, S., Variawa, C. (2019). Using Machine Learning as a Tool to Help Guide Undeclared/Undecided First-Year Engineering Students Towards a Discipline. In *Proceedings of the Canadian Engineering Education Association (CEFA-ACEG19) Conference.*

Kowalski, K., & Ealy, D. (1991). Schedule Advisement Expert System. *Computers in Human Behavior, 17*(4), 259-265.

Marade, A. A., & Brinthaupt, T. M. (2018). Good and Bad Reasons for Changing a College Major: A Comparison of Student and Faculty Views*. Education, 138*(4), 323-336.

Marques, O., Ding, X., & Hsu, S. (2001). Design and Development of a Web-Based Academic Advising System. In *Proceedings of the 31st Annual Frontiers in Education Conference. Impact on Engineering and Science Education*.

Mesidor, J. K., & Sly, K. F. (2016). Factors that Contribute to the Adjustment of International Students. *Journal of International Students*, *6*(1), 262-282.

Miranda, J., Ray, P. A., & Robles, J. M. (2012). udpSkeduler: A Web Architecture Based Decision Support System for Course and Classroom Scheduling. *Decision Support Systems, 52*(2), 505-513.

Moreno, G. A., Bischof, W. F., & Hoover, H. J. (2012). Interactive Visualization of Dependencies. *Computers and Education, 58*(4), 1296-1307.

Oladokun, V. O., & Oyewole, D. I. (2015). A Fuzzy Inference Based Decision Support System for Solving the University-Course Admission Choice Problem. *International Journal of Computer Applications*, *112*(3), 1-7.

Pozzebon, J. A., Ashton, M. C., & Visser, B. A. (2014). Major Changes, Personality, Ability, and Congruence in the Prediction of Academic Outcomes. *Journal of Career Assessments*, *22*(1), 75-88.

Rienties, B., Beausaert, S., Grohnert, T., Niemantsverdriet, S., & Kommers, P. (2012). Understanding Academic Performance of International Students: The Role of Ethnicity, Academic and Social Integration. *Higher Education*, *63*(6), 685-700.

Roushan, T., Chaki, D., Hasdak, O., Chowdhury, M., Annajiat, A., Rahman, M., et al. (2014). University Course Advising: Overcoming the Challenges Using Decision Support System. In *16th International Conference on Computer and Information Technology (ICCIT)*, 13-18.

Samaranayake, S., & Gunawardena, A. D. (2020). Dependency Evaluation and Visualization Tool for Systems Represented by a Directed Acyclic Graph. *International Journal of Advanced Computer Science and Applications, 11*(7), 1-7.

Siddiqui, A. W., Raza, S. A., & Tariq, Z. M. (2018). A Web-Based Group Decision Support System for Academic Term Preparation. *Decision Support Systems, 114*(Oct), 1-17.

Soria, K. M., & Stebletin, M. (2013). Major Decisions: Motivations for Selecting a Major, Satisfaction, and Belonging. *NCADA Journal*, *33*(2), 29-43.

Streufert, B. (2019). Advising Alternatives: A Case Study. *NACADA Review*, *1*(1), 14-29.

Wang, R., & Orr, J. E. (2019). Use of Data Analytics in Supporting the Advising of Undecided Students. *Journal of College Student Retention: Research, Theory & Practice*, *22*(Oct), 76-84.

Zucker, R. (2009). ViCurriAS: A Curriculum Visualization Tool for Faculty, Advisors, and Students. *Journal of Computing Sciences in Colleges*, *25*(2), 138-145.