

# Weighted Bidirectional Graph-based Academic Curricula Model to Support the Tutorial Competence

Luz A. Sánchez Gálvez, Mario Anzures García, Álvaro Campos Gregorio

Benemérita Universidad Autónoma de Puebla,  
Facultad de Ciencias de la Computación,  
México

{lgalvez, mario.anzures}@correo.buap.mx, alvarocampos9110@gmail.com

**Abstract.** In competency-based model, the professor is known as a tutor, who guides and orients the students throughout their professional instructor. Therefore, the tutor must identify which courses the students have approved and which courses should they undertake over the next period, so their academic instruction does not fall behind. Consequently, the curricula must be modeled to specify a complex structure composed of courses (which are represented as nodes), links (symbolized as edges) among the prerequisites of the course, as well as the semester number (represented by the weight of the node). This paper proposes a weighted bidirectional graph, which will represent the intrinsic and wired restrictions in the flow of subjects that students must follow and pass, establishing an organized context (which is regulated by the weight of each subject) within which they can be oriented and guided by the tutor. A system to monitor the students' academic progress is presented, and provide an interaction between the students and the tutor, where the weighted bidirectional graph is integrated. Such a system is a collaborative application; whose development is based on an ontological model. Therefore, this work integrates the graphs with artificial intelligence.

**Keywords.** Weighted bidirectional graph, academic curricula, model, tutorial competency, collaborative application.

## 1 Introduction

Currently, the role of the university professor has changed in the competency-based model; requiring him/her to become an instructor and guider committed to the integral education of the student.

In such a way, he/she is provided with a solid knowledge and a work discipline that will allow him/her to continue learning throughout his life, to face problems and challenges both professionally and personally. Therefore, the responsibility of the professor as a university instructor grows, in order to guarantee the graduate to have a competent professional profile according to the needs and current demands of society, the productive and work environment. Given this panorama, the professor should performance the triple competence: technical-pedagogical, tutorial and research [1]. Therefore, it is necessary to carry out a process of innovation of the academic curricula, which implies a substantial change from the way of generating it by analyzing the overall performance of the professional profile in your field of work.

In order to identify the Units of Competence (UC) [2] until its application in the development of a didactic methodology; obtaining as result a successful and competent professional with high qualification to develop and adapt to their work environment. Consequently, each UC that makes up the curricula corresponds to the essential competences of this profile.

This paper focuses on tutorial competence, so the curricula must be modeled to know in each moment the student academic situation, what subjects should course him/her, in particular, those corresponding to the next period, so, the professor can recommend the subjects that the student should cover the subsequent semester.

Therefore, an application that provides an adequate interaction between the tutor and students is required.

In such a way, they are in constant communication, collaboration, and coordination to reaching the common goal, which is to manage and control the monitoring of the student's educational path. Consequently, a collaborative application is ideal to support the tutorial competence of the professor, since it offers a shared space interface where the tutor can guide and orient to the student throughout their university instruction.

This kind of application is a computer-based system that supports groups of people who are engaged in a common task (or goal), providing an interface to a shared environment [3]. The development of such application is founded on ad-hoc, use of toolkits, components, and conceptual modeling. This paper focuses in the fourth option, so that it applies an ontological model to develop the collaborative application of the tutorial competence.

On the other hand, this guide and orientation is grounded on the student's critical path, which indicates the subjects that he/she must take during their academic stay at the university, in accordance with the curricula, its profile and academic progress. Thus, a graph [4, 5] is proposed for modelling the academic curricula, providing a structure consisting of a set of vertices  $\{v_1, v_2, v_n\}$  and a set of edges  $\{e_1, e_2, e_m\}$ .

In particular, a weighted bidirectional graph [5] is applied, in order to verify if an aggregated subject the academic projection can be prerequisite of other subjects. Which should be automatically inserted and ordered them, without need to create a new list and traverse all graph. Accordingly, Topological Order Algorithm (AOT) [4] is used to pass through all graph nodes; this algorithm is integrated into a collaborative application for monitoring student academic life plan.

The paper is organized as follows. The background supporting the model is described in Section 2. The conceptual model based on weighted bidirectional graph is explained in Section 3. The collaborative application for the tutorial competence in Section 4. Finally, conclusions and future works are presented in Section 5.

## 2 Background of the Model

The competency-based model, the graph, the academic curricula, and the ontology to develop a collaborative application are the preliminaries that founded the development of the model for specifying a Student' Critical Route.

### 2.1 Competency-based Model

Competencies are understood as the capacities of a person, which are required to perform a function in a professional context, and that are reflected in their knowledge, abilities, skills and attitudes [2].

Competency-based model is an educational option that implies a new role of instruction, when this process becomes a generator of abilities that allow subjects to adapt to change, cognitive and socio-affective development, understanding and solution of increasingly complex situations, through the combination of theoretical, practical knowledge, experiences and behaviors [2]. For competency-based model to be effective, the use of significant didactic processes is necessary, techniques and evaluation instruments focused on providing feedback, and establishing levels of advancement, which clearly define the capacities that the student is expected to develop throughout their learning process.

The purpose of competency-based model is for the student to develop specific skills according to the curricular design, which must be related to the demands of society, its productive base, the desired professional profile. Consequently, it is required the instruction of the university graduates, who have professional competence [1], which is the set of knowledge and skills that allow us to accomplish a professional activity, in accordance with the demands of production and employment.

Therefore, a key factor will be the quality of teaching, falling on the responsibility of the professor, who will have to be competent enough to transmit and help the student to build their knowledge effectively. Thus, it should be considered that the professor plays a fundamental role as a central reference of all educational processes that take place both inside and outside the classroom. To perform this role, the professor must develop the triple competence: Technical-Pedagogical, Tutorial and Research.

In the Technical-Pedagogical competence, the professor requires having a mastery of the subject that will be taught and the teaching methodologies, which will allow them to transfer their knowledge to the students and motivate them to continue updating themselves during the development of their professional instruction and the performance of their duties in a job.

The Tutorial competence, allows the professor to orient and guide the students for them to achieve the appropriate professional profile to the current circumstances and needs of the environment where it will unfold, the professor must establish and follow an academic life plan according to the curricula based on the UC, in order to carry adequately out this competence.

The research competence provides guidelines and guidance to improve the technical-pedagogical and tutorial competence of a professor. Since, on the one hand, it allows them to update the knowledge of their discipline, as well as knowing and applying new ways of teaching. On the other hand, the professor acquires a general perspective of the academic aspects in which the student must be instructed.

## 2.2 Graph

A Graph is a non-linear data structure defined with the help of edges that expresses a set of points (nodes) and the relationships between these points  $G = (V, E)$ , and it can be expressed as a set of finite  $V$  and  $E$  elements [4].

Here,  $V$  elements are referred to as vertex or nodes,  $E$  elements are mentioned as edges. Each edge in  $E$  adjoins two different nodes in  $V$ . Nodes are shown by circles and edges are shown by lines (or arcs).

A directed graph is those, whose edges have a defined direction, which indicates some type of precedence relation (or dependency) between the nodes. Therefore, an acyclic directed graph is one that does not present cycles [4]. Undirected graphs have edges that do not have a direction.

The edges indicate a *two-way* relationship; so each one can be navigated in both directions. A bidirectional graph [4] is a graph in which each edge is given an independent orientation (direction, or arrow) at each end.

Consequently, this type of graph allows you to represent structures where it is necessary to move forward and backward along the same edge, without have that traverses all graph.

There are several algorithms for pass through a bidirectional graph [4]: Search first in depth, Transitive Closing, Shortest Path and Topological Order.

AOT processes the nodes of a bidirectional graph, so that each node is managed before those to which it points. The topological order of a graph  $G$  is of a linear order of all vertices, such that, if  $G$  contains an arc  $(U, V)$ , then  $U$  appears before  $V$  in the ordering [4]. Therefore, in this work. AOT is applied to determine the critical path of a student.

On the other hand, the graphs have been used to model real-world systems, such as the Internet (each node represents a router and each edge represents a connection between routers), airline connections (each node is an airport and each edge is a flight), a city road network (each node represents an intersection and each edge represents a block), a wireframe drawing in computer graphics, or the solution of mathematical problems [5, 6, 7, 8].

## 2.3 Academic Curricula

Academics curricula are a complex structure constituted by course (its semester number) and prerequisite mappings between courses, which has been modelled in order to modify and improve it. This modeled has been represented through:

- Tables and spreadsheets that are not inherently easy to summarize or analyze [8, 9, 10, 11].
- Multidimensional scaling to generate a two-dimensional map of the top forty-two elective courses [12, 13].
- Concept maps were being used to improve a science curriculum [14].
- A digraph used during advising and recruitment events to explain the curriculum [15].
- A digraph focused on the student academic path [16].



elements of the interface). A Graph is a non-linear data structure defined with the help of edges

### 3 Model of the Curricula

In this paper, a curricula model to help to the tutor in the tracking academic of the student was developed. The model is based on graph theory, in particular on weighted bidirectional graph, a topic under discrete mathematics. A bidirectional graph can be defined with different step counts for access between nodes. This depends on the relationships identified in the problem handled. Thus, the bidirectional graph model can work properly provided that all necessary nodes and inter-node relationships are defined in full; as well as the opportunity to go back and forth to the edges; simplifying the processing of the curricula.

This study was carried out by focusing on the solution of subject and subject prerequisites, and the relationships between them. By using these elements can be built a graph structure to specify the curricula and support the tutorial competence carried out by tutor: Consequently, an AOT is applied to pass through the weighted bidirectional graph, in such a way that it is possible to generate the critical path of the student and its academic projection when necessary. In order to facilitate, the guide and orientation process that a tutor should perform in favor (or benefit) of his/her students.

In this work, a weighted bidirectional graph to represent the curricula of any university career, has been designed. Considering that:

1. Each subject can be studied as long as the established prerequisites have been passed, that is, the predecessor subjects.
2. The number of subjects to study is six in the fall and spring periods, while in summer it is three.
3. There are subjects without prerequisite that may be taken in a given semester.

The nodes of the weighted bidirectional graph correspond to each of the subjects of the curricula, the edges represent the sequence of the same, and the weight characterize the semester number.

The proposed AOT to find the critical path that the student will follow in its academic stay consists

of the following steps, which will be explicated in the next subsections.

#### 3.1 Graph Model

Start with a graph that represents the curricula subjects and the empty list. This list will contain the subjects for creating the academic projection. The curricula of the Engineering on Information Technologies (EIT) of the Department of Computer Science (DCS), is modelled by a weighted bidirectional graph (see Figure 2).

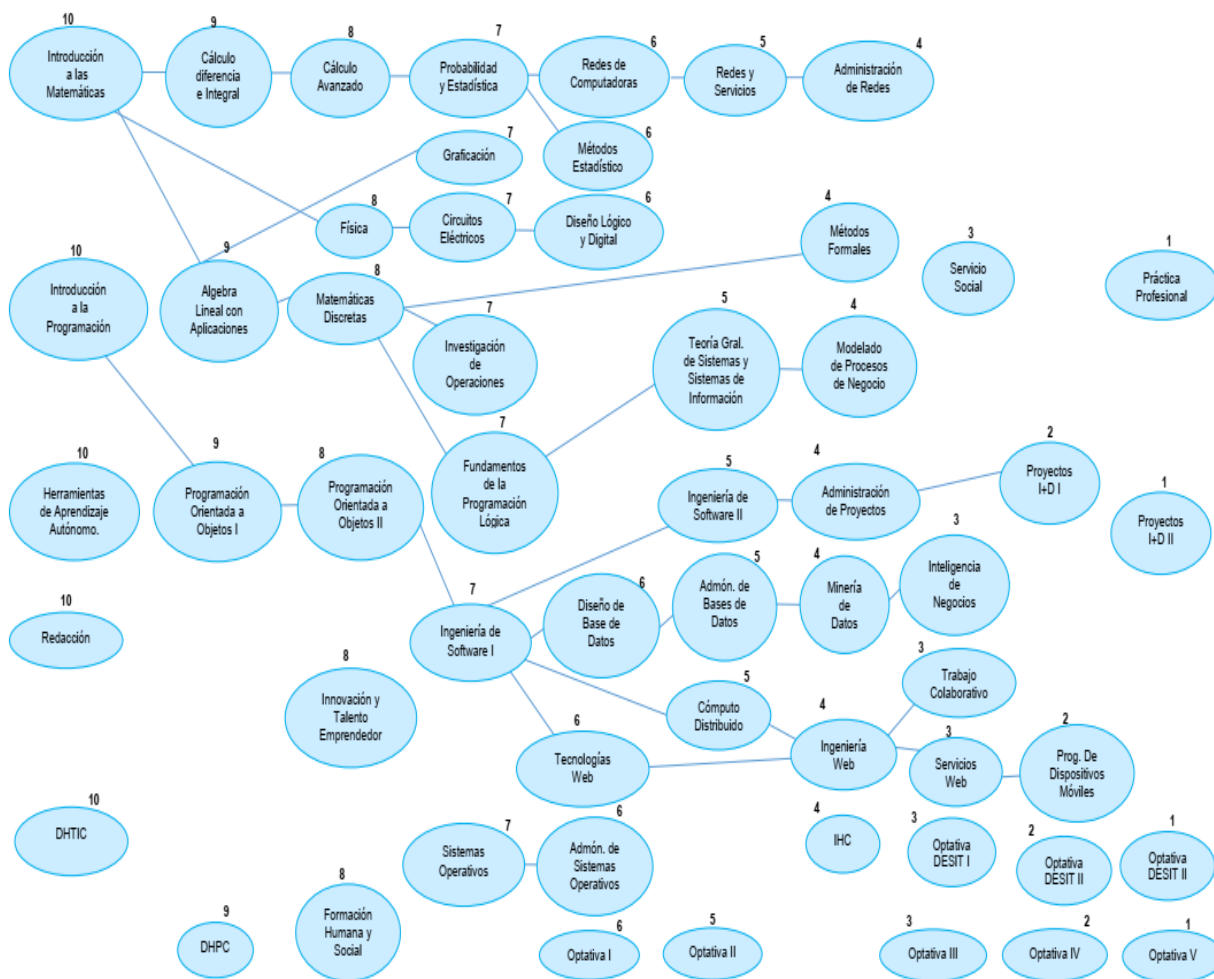
The nodes represent subjects, arcs characterize the subjects' prerequisites, and weights symbolize semester number. In such a way, once one subject with prerequisites has been identified, it determines whether yet the prerequisites subject has been approved. Therefore, these should be removed from the list, which contains the subjects that must be taken.

#### 3.2 Grade Zero Nodes

Those nodes without incident edges are called of grade zero. Consequently, these nodes are selected from the graph, and added to the list. The order in which they enter to the list, it will depend on the weight (it indicates the semester that belongs to the subject) of the node, that is, greater weight corresponds to lower period. Consequently, the weight is ranked of 9 to 1 (see Figure 1), 9 specifies semester first, 8 stipulates semester second, so forth. In this way, it is possible to collocate the grade zero node in the semester corresponding and control that the subjects' prediction list, is the appropriated. In other words, the students course the subjects without prerequisite in the suitable semester, to they have the necessary knowledge on each semester. For these reasons, the labor of tutor to orient and guide to the students is simplified.

#### 3.3 Partial Academic Projection

Only, the nodes (subjects), which will be studied, in a period, they will be selected and kept in the list. Six in the case of spring and fall, or three in summer. Subsequently, these nodes will be removed from the graph. In this process step, the weighted bidirectional graph is used by the



**Fig. 2.** Graph of the academic curricula of the Engineering of Information Technologies

collaborative application, in order to get student critic path, determining its academic projection, and reducing the processing time.

Since each time a subject is visited, the graph must be tracking forward (to keep the subsequent subjects in the prediction list) and backward (to analyze if there is a prerequisite, and if it has been approved).

Both paths can be performed quickly in the weighted bidirectional graph, without having to go through the graph several times, as it happened when the acyclic directed graph has been applied.

Consequently, the weighted bidirectional graph has a better performance than the acyclic directed

graph, though the same algorithm be applied, in this case: the AOT.

### 3.4 Total Academic Projection

Considering the remaining graph, repeat steps 3.2 and 3.3, until there are no nodes to be processed and the academic projection is obtained, from the critical path is generated until the end of the race.4 Collaborative Application to Support the Tutorial Competence. The development of collaborative applications is a complex process, since these applications have to consider activities and protocols social of the organizations; shared workspace; interaction among users and between

**Table 1.** Division de labor for developing the collaborative application of tutorial competence

Role	Task	Activity
Tutor	Enter to collaborative application	Entering username
		Entering password
		Dar click on button
	Reviewing course	Dar click on course
		Selecting student o student page
		Consulting subjects grades
		Consulting curricula graphic map
	Consulting academic projection	
Student	Enter to collaborative application	Entering username
		Entering password
		Dar click on button
	Consulting subjects grades	Dar click on grades
		Consulting academic historical
		Selecting a period
		Reviewing grades of this period
	Consulting curricula graphic map	Reviewing graphic map
Consulting academic projection	Reviewing the remaining subjects by period	

them with shared resources; group awareness; group memory; shared and individual view; and adaptation. In according to Molina [2], four forms for developing this type of applications have been established:

- 1 Ad-hoc: The application is built in a completely adapted way to the specific problem to which it is intended to support.
- 2 Use of toolkits: These provide a higher level of programming abstraction by using functions and API (Application Programming Interfaces).
- 3 Use of components: They allow the construction of collaborative applications by using predefined building blocks.
- 4 Use of conceptual modeling: The process of collaborative environment development is based on conceptual modeling.

#### 4 Development of a Collaborative Application

The ontological model described in the section 2.4 is used in the development of a collaborative application for supporting the tutorial competence. This development is focused on three templates, which below are presented:

*Division of labor* (see Table 1): It establishes the organizational structure of the group; from it indicate the actions that the group members (users) should make in relation to the established roles for each of them.

This division must be governed by a police, which defines the roles that users can play. These roles establish the set of rights/obligations and status of the user; who can execute tasks, which are comprised of activities that use the prevailing shared resources. First, the roles are identified, then for role are defined the tasks that they can carry out, and by each task are identified the activities that constitutes them. *Access control*:

**Table 2.** Access control for developing the collaborative application of tutorial competence

Stage	Task	Role
User access	Enter to collaborative application	Tutor, Student
Grades	Consulting academic historical	Tutor, Student
	Selecting a period	Tutor, Student
	Reviewing grades of the period	Tutor, Student
Graphic map	Consulting curricula graphic map	Tutor, Student
Projection	Consulting academic projection	Tutor, Student

**Table 3.** Shared user interface for developing the collaborative application of tutorial competence

User Interface	Elements	Type	Widget
User access	Enter to collaborative application	Tutor, Student	None
	Consulting academic historical	Tutor, Student	Photographic of Tutor and Student
Grades	Selecting a period	Tutor, Student	Button of the period
	Reviewing grades of the period	Tutor, Student	Grades name
Graphic map	Consulting curricula graphic map	Tutor, Student	Academic map
Projection	Consulting academic projection	Tutor, Student	Tables of each projected period

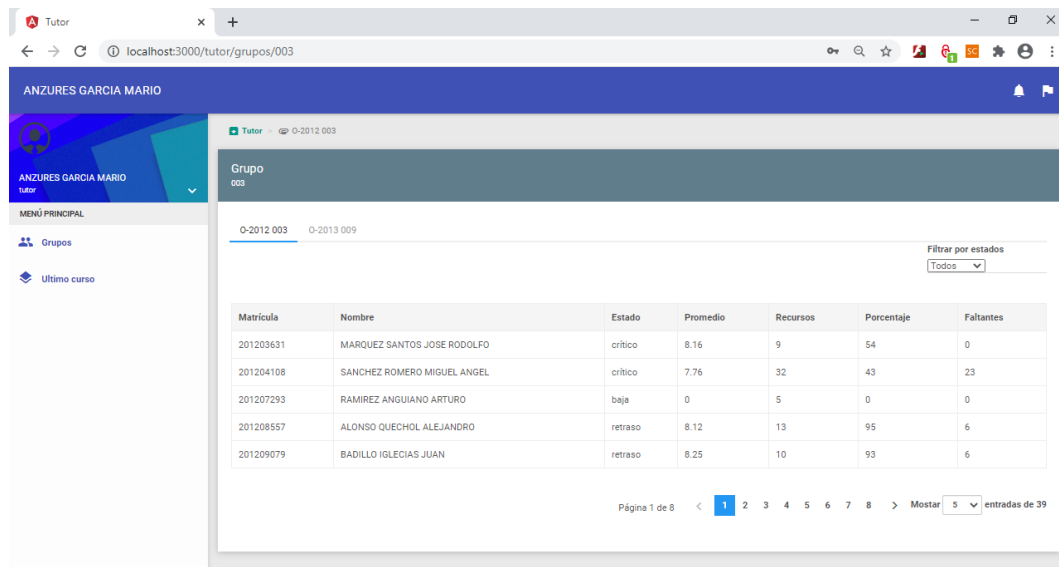
Key element in collaborative applications security (Table 2).

Special attention has been paid on mechanisms that allow identification and verification of the user identity, in an attempt to protect the application from unauthorized access; as well as; in those that enable the information to be protected according to security policies, by allowing access to shared resources only to authorized users.

In a collaborative application, the sharing information takes place at various stages, which in a coordination model is defined as each of the collaboration moments [6]. For example, a tutorial competence collaborative application has several stages: user access, support of the tutorial competition, which is divided into grades, graphic map, and projection. Each stage controls the roles that can participate in it, facilitating the authentication and interaction among users and the shared workspace:

- *Shared User interface:* It shows the resources and the interaction among users, and between them and the collaborative application (see Table 3). Since:
  - It displays all the tasks performed by one or several roles over resources, and where other users are involved.
  - It allows each user to be aware of what other's users are doing (group awareness), and therefore they know what happens in the shared workspace by using of widgets. Every collaborative activity triggers a notification, which sends messages to update the interface, concerning the modification of the resource, promoting greater cooperation and interaction among users in different collaborative tasks.
  - To do this, concurrency is used, for it helps to manage the permissions assigned to users, ensuring the mutually exclusive use.

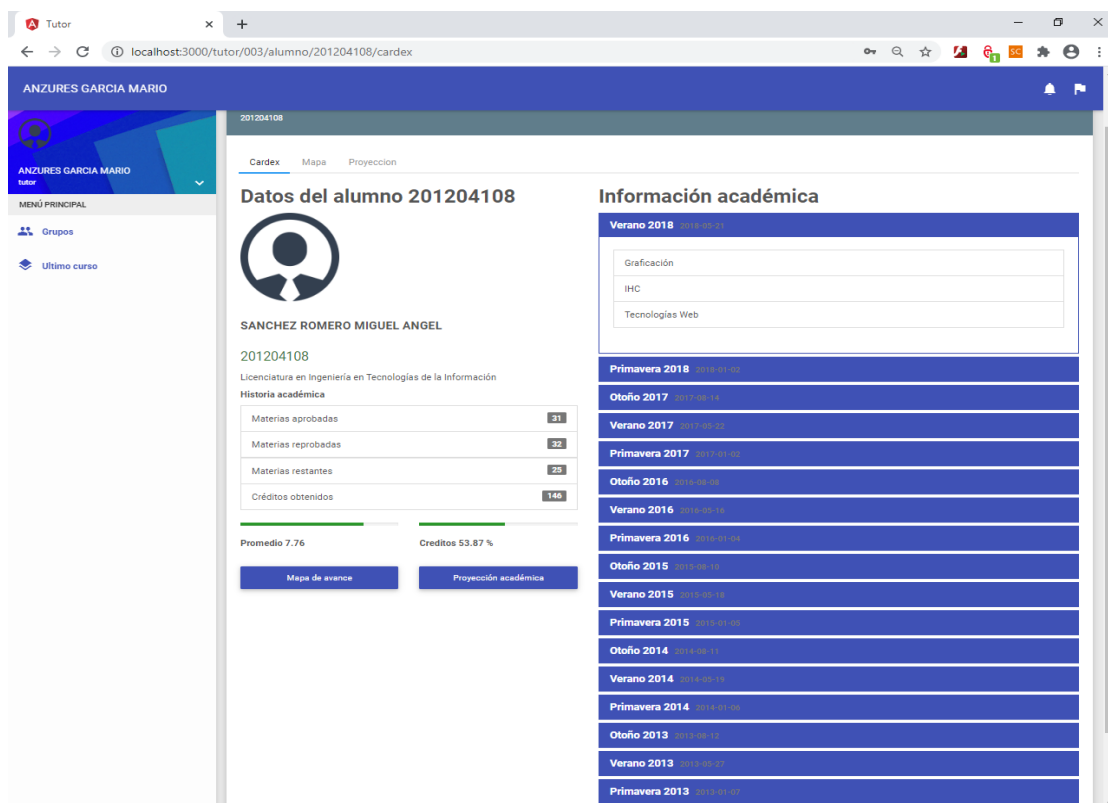




The screenshot shows a web browser window with the URL `localhost:3000/tutor/grupos/003`. The interface is for a user named ANZURES GARCIA MARIO. The main content area displays a table of students in 'Grupo 003' for the semester 'O-2012 003'. The table has columns for Matricula, Nombre, Estado, Promedio, Recursos, Porcentaje, and Faltantes. A filter dropdown is set to 'Todos'. The page shows 'Página 1 de 8' and 'Mostrar 5 entradas de 39'.

Matricula	Nombre	Estado	Promedio	Recursos	Porcentaje	Faltantes
201203631	MARQUEZ SANTOS JOSE RODOLFO	critico	8.16	9	54	0
201204108	SANCHEZ ROMERO MIGUEL ANGEL	critico	7.76	32	43	23
201207293	RAMIREZ ANGUIANO ARTURO	baja	0	5	0	0
201208557	ALONSO QUECHOL ALEJANDRO	retraso	8.12	13	95	6
201209079	BADILLO IGLECIAS JUAN	retraso	8.25	10	93	6

Fig. 3. Tutor user interface



The screenshot shows a web browser window with the URL `localhost:3000/tutor/003/alumno/201204108/cardex`. The interface is for a student named SANCHEZ ROMERO MIGUEL ANGEL (ID: 201204108). The page displays 'Datos del alumno' and 'Información académica'. The 'Historia académica' section shows a bar chart with the following data:

Categoría	Cantidad
Materias aprobadas	31
Materias reprobadas	32
Materias restantes	20
Créditos obtenidos	148

Summary statistics: Promedio 7.76, Créditos 53.87%. The 'Información académica' section shows a list of semesters from Primavera 2013 to Verano 2018, each with a corresponding 'Graficación' (grading) area.

Fig. 4. Student user interface: Academic historical

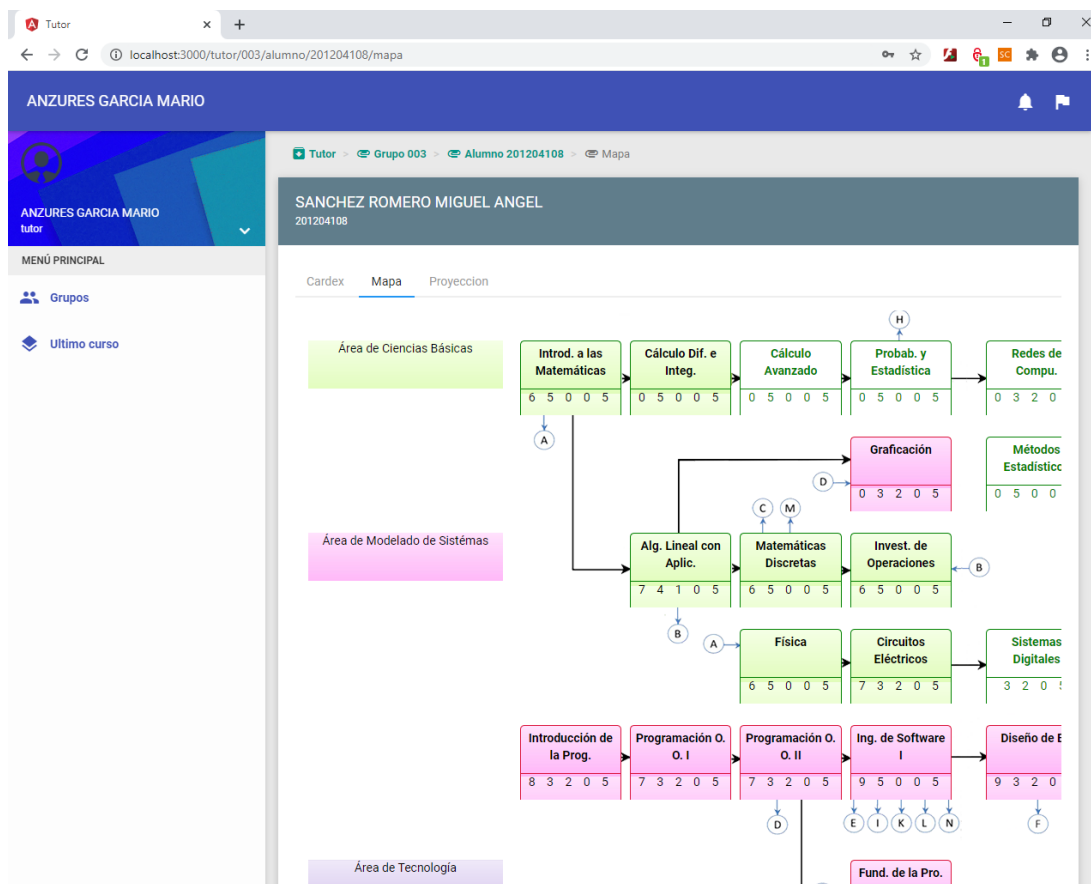


Fig. 5. Student user interface: Graphic map

#### 4.1 Collaborative Application to Competence Tutorial

The case of study consists to develop the tutorial competence collaborative application of the career of the EIT of the DCS. This career has a population of 498 students, whom academic curricula should be processed; implicating what a large amount of information can be analyzed to obtain the academic projection of that number of students. The collaborative application for supporting of competence tutorial was developed from tables defined in the section 4.1, in order to allow the communication, collaboration, and coordination between the tutor and student. Consequently, two user interfaces, principally, has been designed, one for the tutor role (see Figure 3) and another for the student role (see Figure 4).

The tutor user interface (see Figure 3) shows its name, two options: groups, and lasts courses, by default the information of its group is presented. For each group a student's list collected by pages is displayed. In order to the tutor guides and orients to students in a better manner, the first can consult the list of approved subjects in the last period by each student.

When a student of this list is chosen, its information is exhibited in the Figure 4, which is the user interface of the student role. This interface contains the academic historical, graphic map, and projection.

- The academic historical presents a summary of the academic information, with respect to the approved, unapproved subjects, obtained

1. FIS	25. Intelligencia de Negocios	49. Agentes
2. DHPC	26. Programación de Dispositivos Móviles	50. Desarrollo Basado en Modelos
3. DHTIC	27. Trabajo Colaborativo	51. Cómputo Orientado a Servicios
4. Innovación y Talento Emprendedor	28. Administración de Proyectos	52. Programación Declarativa
5. Introducción a las Matemáticas	29. Proyectos I+D I	53. Web Semántica
6. Álgebra Lineal con Aplicaciones	30. Proyectos I+D II	54. Tratamiento de Información
7. Cálculo Diferencial e Integral	31. Servicio Social	55. Sistema de Información Geográfica
8. Cálculo Avanzado	32. Práctica Profesional	56. Ingeniería de Conocimiento
9. Matemáticas Discretas	33. Métodos Estadísticos	57. Intelligencia Antifalsal
10. Física	34. Redes de Computadoras	58. Procesamiento Digital de Imágenes
11. Probabilidad y Estadística	35. Redes y Servicios	59. Cómputo Ubicuo
12. Investigación de Operaciones	36. Diseño Lógico y Digital	60. Técnicas de Razonamiento Inteligible
13. Circuitos Eléctrico	37. Diseño de Bases de Datos	61. Aprendizaje Artificial
14. Introducción a la Programación	38. Administración de Bases de Datos	62. Gestión Electrónica
15. Programación Orientada a Objetos I	39. Teoría Gen. de Sist. y Sist. de Informa.	63. Logística
16. Programación Orientada a Objetos II	40. Ingeniería Web	64. Teoría de Decisiones
17. Ingeniería de Software I	41. Modelado de Procesos de Negocio	65. Lenguajes y Traductores
18. Fundamentos de la Programación Lógica	42. Tecnologías Web	66. Organización de Computadoras
19. Graduación	43. Administración de Sistemas Operativos	67. Desarrollo de Sitios Web
20. Sistemas Operativos	44. Cómputo Distribuido	68. Tecnologías Emergentes para los TICs
21. HAA	45. Métodos Formales	69. Mantenimiento de Equipo y Redes de Compu.
22. Reducción	46. Minería de Datos	70. Arte y Representación I
23. Ingeniería de Software II	47. Interacción Humano-Computadora	71. Animación per Computadora
24. Administración de Redes	48. Servicio Web	72. Mercadotecnia

Fig. 6. List of subjects of the EIT academic program

Lista de asignaturas por cursar:  
[24, 31, 45, 30, 25, 27, 32, 26, 49, 50, 51, 52, 53, 54, 55, 56, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72]

Lista de asignaturas que se pueden cursar:  
[68, 58, 65, 24, 26, 25, 61, 45, 27, 30, 52, 63, 69, 64, 54, 49, 50, 51, 31, 32]

Lista de asignaturas para el curso Verano 2017:  
[68, 58, 65]

Lista de proyecciones:  
[  
    [46, 41, 44, 40, 20, 28],  
    [35, 48, 29, 19, 43, 57],  
    [68, 58, 65],  
]

Créditos restantes: 60

Fig. 7. Subjects' projections list

Otoño 2018	Carga actual	Primavera 2019	Verano 2019
Admón. de Proyectos		Cálculo Dif. e Integ.	Cálculo Avanzado
Minería de Datos		TG de Sist. y Sist. de I.	Ingeniería Web
IHC		Tecnologías Web	Mod. de P. de Negocios
		Computo Distribuido	
		Graduación	
		Fund. de la Pro. Lógica	
Otoño 2019	Carga actual	Primavera 2020	
Probab. y Estadística		Redes de Compu.	
Métodos Formales		Servicio Social	
Proyectos I+D I		Tratamiento de Info.	
Intelig. de Negocios		Agentes	
Trabajo Colaborativo		Lenguajes y Traduct.	
Servicios Web			

Fig. 8. Academic projections

credit and percentage, so far. As well as, a list of periods already completed, where the student can see the obtained grades by subject in each one these.

- The map shows the student academic curricula in a graphic way (see figure 5), in which the subjects colored in white, are the ones that should be given to the student.
- The academic projection performs and indicates the subject that the student must take each period until to complete it academic program, both the student and the tutor can access to this interface. In this projection, a weighted bidirectional graph represents the EIT academic program and, an AOT generates the critical path of the student. Consequently, for a new student the list of the all subjects of the EIT academic program, is created, which includes optional subjects, both disciplinary and DESIT (Figure 5).

Then, weight of each node are determined, and the zero grade nodes are identified -in bold- (**24, 31, 45, 30, 25, 27, 32, 26, 49, 50, 51, 52, 53, 54, 55, 56, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72**); subsequently they are added to the list.

Three nodes are selected from the previous list; since the period to be studied is summer, 2019 (see the third list in Figure 6). This list of three nodes is added to the student's projection (see the fourth list in Figure 7, which displays the projections for fall 2018, spring 2019, and 2019 summer). The algorithm continues running until the full projection is shown (see Figure 8).

## 5 Conclusions and Future Work

The main contribution of this work is a weighted bidirectional graph, which allows modeling the ITI academic curriculum, in such a way that it determines the students' critical path, by providing adequate academic projections, so that they and their tutor can make convenient decisions according to their necessities.

In addition, the processing time of the 498 students' curricula, which make up the ITI enrollment is considerably reduced, and the academic projection is calculated when the student

and/or tutor require them. From the experience of the development of the collaborative application, the following is extracted: the main benefits derived from this experience are the combination of graphs with artificial intelligence, and the promotion of the communication, coordination, and collaboration between tutor and student.

The tutor has access to the information of all the students (of one or several groups) in a single shared space, which allows them to carry out an academic follow-up through a constant communication and feedback; as well as the automatic update of the students' progress. This way, the professors can successfully complete their tutorial competence. The student receives a direct guide from the tutor, and has a projection according to their profile and academic progress, at any given time.

The future work will be focused on applying the weighted bidirectional graph to the other careers of the DCS, as well as adding chat, and videoconference within the collaborative application, to provide a more direct and simple communication.

## References

1. **Ruíz-Iglesias, M. (2008).** *Marco conceptual de la formación basada en competencias en ciencias pedagógicas, material de la maestría internacional en competencias profesionales.* UANL/UCLM.
2. **Rial-Sánchez, A. (2008).** Diseñar por competencias, un reto para los docentes universitarios en el espacio europeo de la educación superior. *Innovación Educativa*, No. 18, pp. 169–187.
3. **Ellis, C.A., Gibs, S.J., & Rein, G.L. (1991).** Groupware: Some issues and experiences. *Communication of the ACM*, Vol. 34, No.1, pp. 39– 8.
4. **Dharwadker, A. & Pirezada, S. (2011).** *Graph theory.* Amazon.
5. **Edmonds, J. & Johnson, E.L. (2003).** Matching: a well-solved class of linear programs, Combinatorial Structures and their Applications. *Proceedings of the Calgary Symposium*, pp. 27– 30.
6. **Shirinivas, S.G., Vetrivel, S., & Elango, N.M. (2010).** Application of graph theory in computer science an overview. *International Journal of Engineering Science and Technology*, Vol. 2, No. 9, pp. 4610–4621.
7. **Pirezada, S. & Dharwadker, A. (2007).** Applications of graph theory. *Journal of The Korean Society for Industrial and Applied Mathematics*, Vol. 11, No. 4, pp. 19–38.
8. **Foltête, J.C., Girardet, X., & Clauzel, C. (2014).** A methodological framework for the use of landscape graphs in land-use planning. *Landscape and Urban Planning*, Vol. 124, pp. 140–150.
9. **Jacobs, H.H. (1997).** *Mapping the big picture: integrating curriculum and assessment K-12.* Association for Supervision and Curriculum Development.
10. **Jacobs, H.H. (2005).** *Getting results with curriculum mapping.* Association for Supervision and Curriculum Development.
11. **Jacobs, H.H. (2009).** *The curriculum mapping planner: Templates, tools, and resources for effective professional development.* Association for Supervision and Curriculum Development.
12. **Udelhofen, S. (2005).** Keys to curriculum mapping: strategies and tools to make it work. *Thousand Oaks, CA: Sage Publications, Inc.*
13. **White, H.D. & Calhoun, K. (1984).** Mapping a curriculum by computer. *Journal of the American Society for Information Science*, Vol. 35, No. 2, pp. 82–89. DOI: 10.1002/asi.4630350204.
14. **Starr, M.L. & Krajcik, J.S. (1990).** Concept maps as a heuristic for science curriculum development: toward improvement in process and product. *Journal of Research in Science Teaching*, Vol. 27, No. 10, pp. 987–1000. DOI:10.1002/tea.3660271007.
15. **Gestwicki, P. (2008).** Work in progress - Curriculum visualization. *Proceedings of the 38th Annual Frontiers in Education Conference*, pp. 13– 14. DOI: 10.1109/FIE.2008.4720392.
16. **Anzures-García, M., Sánchez-Gálvez, L.A., Hornos, M.J., & Paderewski-Rodríguez, P. (2018).** Tutorial function groupware based on a workflow ontology and a directed acyclic graph. *Journal IEEE Latin American Transactions*, Vol. 16, No. 1, pp. 294–300. DOI: 10.1109/TLA.2018.8291487.
17. **Erümit, A., Nabyev, V., Kösa, T., Kokoç, M., & Aksoy, A. (2019).** A general analytical model for problem solving teaching: BoS. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, Vol. 10, No. 2, pp. 476–500. DOI: 10.16949/turkbilmat.486084.
18. **Bo-Yang, B., Liu, Z., & Meloche, J.A. (2010).** Visualization of the Chinese academic web based

- on social network analysis. *Journal of Information Science*, Vol. 36, No. 2, pp. 131–143. DOI: 10.1177/0165551509353373.
19. **Sommaruga, L. & Catenazzi, N. (2007).** Curriculum visualization in 3D. *Proceedings of the Twelfth international conference on 3D web technology*, pp. 177–180. DOI: 10.1145/1229390.1229423.
  20. **Kabicher, S. & Motschnig-Pitrik, R. (2009).** Coordinating curriculum implementation using Wiki-supported graph visualization. *Proceedings Ninth IEEE International Conference on Advanced Learning Technologies*, pp. 742–743, DOI: 10.1109/ICALT.2009.54.
  21. **Mitchell, G.G. & Howlin, C.P. (2009).** Intelligent learning systems where are they now? *China-Ireland International Conference on Information and Communications Technologies (CIICT)*, pp. 256–259.
  22. **Zucker, R. (2009).** ViCurriAS: a curriculum visualization tool for faculty, advisors, and students. *Journal of Computing Sciences in Colleges*, Vol. 25, No. 2, pp. 138–145.
  23. **Siirtola, H., Raiha, K.J., & Surakka, V. (2013).** Interactive curriculum visualization. *Proceedings of the 17th International Conference on Information Visualisation*, pp. 108–117. DOI: 10.1109/IV.2013.13.
  24. **Anzures-García, M., Sánchez-Gálvez, L.A., Hornos, M.J., & Paderewski-Rodríguez, P. (2018).** Workflow ontology to support knowledge management in a group's organizational structure. *Computación y Sistemas*, Vol. 22, No. 1, pp. 163–178. DOI: 10.13053/cys-22-1-2781.
  25. **Anzures-García, M., Sánchez-Gálvez, L.A., Hornos, M.J., & Paderewski-Rodríguez, P. (2015).** Knowledge base for the development of collaborative. *Applications Engineering Letters*, Vol. 23, No. 2, pp. 65–71.
  26. **Anzures-García, M., Sánchez-Gálvez, L.A., Hornos, M.J., & Paderewski-Rodríguez, P. (2016).** Knowledge-based workflow ontology for group organizational structure. *Research in Computing Science: Advances in Computer Science and Engineering*, Vol. 123, pp. 79–90.
  27. **Anzures-García, M., Sánchez-Gálvez, L.A., Hornos, M.J., & Paderewski-Rodríguez, P. (2017).** Software engineering: Methods, modeling, and teaching. *Proceeding Facilitating the development of collaborative applications with the MVC architectural pattern*, Vol. 15, pp. 268–290.
  28. **Anzures-García, M., Sánchez-Gálvez, L.A., Hornos, M.J., & Paderewski-Rodríguez, P. (2016).** A semantic formalism to model the group interaction. *Research in Computing Science: Advances in Computer Science and Engineering*, Vol. 118, pp. 137–147.

Article received on 29/10/2019; accepted on 05/03/2020.  
Corresponding author is Luz A. Sánchez-Gálvez.