



**UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO**  
**FACULTAD CIENCIAS DE LA INGENIERÍA**  
**CARRERA INGENIERÍA EN SISTEMAS**

Proyecto de investigación  
previo a la obtención del título  
de Ingeniero en Sistemas

**Título del proyecto de investigación:**  
**"OPTIMIZACIÓN BASADA EN COLONIA DE HORMIGAS PARA RESOLVER**  
**PROBLEMAS ASIGNACIÓN DE HORARIOS EN LA UNIVERSIDAD TÉCNICA**  
**ESTATAL DE QUEVEDO"**

**Autores:**

John Alexander Oñate Espinoza

Darwin Abel Tullmo Toapanta

**Director de Tesis:**

Lic. Amilkar Puris Cáceres PhD

**Quevedo – Ecuador.**

**2016**





**UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO**  
**FACULTAD CIENCIAS DE LA INGENIERÍA**  
**CARRERA INGENIERÍA EN SISTEMAS**

Presentado al Consejo Directivo como requisito previo a la obtención del título de  
**INGENIERO EN SISTEMAS**

Aprobado:

**ING. PAVEL NOVOA HERNÁNDEZ PhD**  
PRESIDENTE DEL TRIBUNAL DE TESIS

**ING. CARLOS MÁRQUEZ DE LA PLATA**  
MIEMBRO DEL TRIBUNAL DE TESIS

**ING. EDUARDO SAMANIEGO MENA**  
MIEMBRO DEL TRIBUNAL DE TESIS

CERTIFICACION DE REDACCION TECNICA DEL PROYECTO DE  
INVESTIGACION

La suscrita, Soc. Teddy Elizabeth De La Cruz Valdivieso M.Sc, Docentes de la Universidad Técnica Estatal de Quevedo, certifica que a los estudiantes **John Alexander Oñate Espinoza y Darwin Abel Tullmo Toapanta**, se les procedió a la respectiva revisión y a su vez las correcciones realizadas por los estudiantes de su proyecto titulado **“OPTIMIZACIÓN BASADA EN COLONIA DE HORMIGAS PARA RESOLVER PROBLEMAS ASIGNACIÓN DE HORARIOS EN LA UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO”** previo a la obtención del título de Ingeniero en Sistemas bajo mi revisión, habiendo cumplido con las disposiciones reglamentarias establecidas para el efecto

---

**Soc. Teddy Elizabeth De La Cruz Valdivieso M.Sc**  
REDACCIÓN TÉCNICA

## **DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS**

Nosotros, JOHN ALEXANDER OÑATE ESPINOZA Y DARWIN ABEL TULLMO TOAPANTA, declaramos que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Universidad Técnica Estatal de Quevedo, puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

---

John Alexander Oñate Espinoza  
C.I.: 0503600611

---

Darwin Abel Tullmo Toapanta  
C.I.: 0503097107

## **CERTIFICACIÓN DEL DIRECTOR DE TESIS**

El suscrito, PhD. Msc. Amilkar Puris Cáceres, Docente de la Universidad Técnica Estatal de Quevedo, certifica que los Egresados JOHN ALEXANDER OÑATE ESPINOZA Y TULLMO TOAPANTA DARWIN ABEL, realizaron la tesis de grado previo a la obtención del título de INGENIERO EN SISTEMAS titulada OPTIMIZACIÓN BASADA EN COLONIA DE HORMIGAS PARA RESOLVER PROBLEMAS ASIGNACIÓN DE HORARIOS EN LA UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO, bajo mi dirección, habiendo cumplido con las disposiciones reglamentarias establecidas para el efecto.

---

Lic. Amilkar Puris Cáceres PhD.

## **DEDICATORIA**

A mi familia por su apoyo siempre junto a mí durante el transcurso de esta meta por su compañerismo y buenos ratos vividos

A mis profesores por brindarme esos conocimientos adquiridos en el aula de clases

A mi tutor quien brindó su tiempo incondicional para que pueda culminar con éxito este proyecto de tesis

**John Alexander Oñate Espinoza**

A mis padres por estar conmigo en todo momento apoyándome con la realización de mi proyecto de investigación.

A mis abuelos que fueron fundamental en mi vida, que con sus sabios consejos hizo de mí un ser con bien.

A mis maestros que me impartieron conocimiento y saber, siendo ellos mi segundo hogar, que con sus sabias enseñanzas nos dejan huellas imborrables en nuestras mentes.

**Darwin Abel Tullmo Toapanta**

## **AGRADECIMIENTOS**

*A nuestros padres por toda su ayuda, por apoyarnos en todo momento, por la comprensión, por los valores que me han inculcado y sobre todo por ser un excelente ejemplo de vida.*

*A nuestras familias por el apoyo brindado siempre.*

*Al Dr. Amilkar Puris Cáceres y familia por permitirnos entrar en su hogar como un amigo más y sobre todo por compartir sus sabios conocimientos, su experiencia, su paciencia y su motivación para la realización de este trabajo investigativo.*

*A mis grades e incondicionales amigos y por ayudarnos a salir del estrés y por su gran apoyo durante los días de estudio.*

*Para ellos: Muchas gracias y que Dios los bendiga.*

***John Alexander Oñate Espinoza***

***Darwin Abel Tullmo Toapanta***



## RESUMEN EJECUTIVO

El constante avance de la tecnología y el incremento en demanda de automatización de la actualidad en los diferentes ámbitos, tanto en la educación, salud y economía, el software convencional no satisface los requerimientos de optimización y rapidez en cuanto a la planificación de tareas, en este caso la generación de horarios de clases, requiere una planificación con constantes cambios lo cual conlleva una inversión de tiempo y esfuerzo.

Con el presente proyecto de investigación se presenta la descripción y desarrollo de la aplicación web, que aplica la metaheurística de optimización colonia de hormiga, para la generación de horarios de la Universidad Técnica Estatal de Quevedo de la Facultad Ciencias de la Ingeniería, con la finalidad de disminuir la insatisfacción de los docentes y estudiantes durante el proceso de gestión de horarios, cabe recalcar como toda aplicación no entrega resultados 100% óptimos sino una aproximación a lo ideal, debido al modelo complejo por la cantidad de restricciones que se deben de cumplir y satisfacer.

Mediante el proceso de generación de horario se implementan cuatro módulos, el de Administración y el de Carga de Datos permiten gestión de la información y a su vez asignar a los docentes a un determinado paralelo, Generación de Horario en el módulo se puede generar los horarios y Reconstrucción de Horario se los utilizan para modificar el horario generado por paralelo.

Para el presente caso de estudios se utilizó la información de la Facultad Ciencias de la Ingeniería pertenecientes a la Universidad Técnica Estatal de Quevedo; en las pruebas realizadas se obtuvo resultados satisfactorios en las cuales intervinieron 25 paralelos que pertenecen al ciclo 1, 5 mallas de las siguientes carreras: la carrera de sistemas, eléctrica, mecánica, telemática y seguridad industrial, 86 docentes con sus respectivos cargos, esta información se obtuvo del distributivo proporcionado por el Ing. Efraín Díaz, uno de los encargados de realizar los distributivos en la Universidad Técnica Estatal de Quevedo.

**Palabras Claves:** Metaheurística, Optimización, Colonia de Hormiga, Probabilístico.

## ABSTRACT

The constant advance of the technology and the increase in the demand of automation of the present time in the different fields, in the education, health and economy, conventional software does not satisfy the requirements of optimization and quickness in the planning of tasks, in This case the generation of class schedules, requires planning with constant changes which entails an investment of time and effort.

This research project presents the description and development of the web application, which applies the ant colony optimization metaheuristics, for the generation of schedules of the Quevedo State Technical University of the Faculty of Engineering Sciences, with the purpose Of decreasing teacher and student dissatisfaction during the time management process, it should be emphasized that every application does not deliver 100% optimal results, but rather an approximation to the ideal, due to the complex model due to the number of restrictions that must be met and satisfy.

Through the process of generation of time four modules are implemented, the Administration and the Load of Data allow management of the information and in turn to assign to the teachers a certain parallel, Generation of Time in the module can generate the schedules and Schedule Reconstruction are used to modify the schedule generated by parallel.

For the present case study was used information from the Faculty of Engineering Sciences belonging to the State Technical University of Quevedo; In the tests carried out satisfactory results were obtained in which 25 parallels belonged to the cycle 1, 5 mesh of the following careers: systems, electrical, mechanical, telematics and industrial safety, 86 teachers with their respective positions, this information Was obtained from the distributive provided by Engineer Efrain Diaz, one of those in charge of performing distributions at Quevedo State Technical University.

**Keywords:** Metaheuristics, Optimization, Ant Colony, Probabilistic.

## ÍNDICE

<b>Introducción .....</b>	<b>1</b>
<b>CAPÍTULO I .....</b>	<b>2</b>
<b>CONTEXTUALIZACIÓN DE LA INVESTIGACIÓN .....</b>	<b>2</b>
<b>1.1. Problema de investigación.....</b>	<b>3</b>
<b>1.1.1. Planteamiento del problema. ....</b>	<b>3</b>
<b>1.1.2. Formulación del problema. ....</b>	<b>4</b>
<b>1.1.3. Sistematización del problema. ....</b>	<b>4</b>
<b>1.2. Objetivos. ....</b>	<b>4</b>
<b>1.2.1. General.....</b>	<b>4</b>
<b>1.2.2. Específicos.....</b>	<b>4</b>
<b>1.3. Justificación.....</b>	<b>4</b>
<b>1.3.1. Alcance.....</b>	<b>5</b>
<b>CAPÍTULO II.....</b>	<b>6</b>
<b>MARCO TEÓRICO.....</b>	<b>6</b>
<b>2.1. Marco Conceptual.....</b>	<b>7</b>
<b>2.1.1. Fundamentos De La Inteligencia Artificial. ....</b>	<b>7</b>
<b>2.1.2. Algoritmo. ....</b>	<b>8</b>
<b>2.1.3. Meta-heurística. ....</b>	<b>8</b>
<b>2.1.4. Los algoritmos evolutivos.....</b>	<b>8</b>
<b>2.1.5. Algoritmos Genéticos (AG). ....</b>	<b>9</b>
<b>2.1.6. Colonia de hormigas. ....</b>	<b>9</b>
<b>2.1.7. Asignación dinámica de recursos con técnicas bio inspirada para un sistema de control de temperatura MIMO. ....</b>	<b>11</b>
<b>2.1.8. Complejidad algorítmica.....</b>	<b>11</b>
<b>2.1.9. iText Software. ....</b>	<b>12</b>
<b>2.1.10. Algoritmos utilizados para la elaboración del software para generación de horarios.....</b>	<b>13</b>
<b>2.1.10.1. Búsqueda Primero el Mejor.....</b>	<b>13</b>
<b>2.1.10.2. Búsqueda local guiada. ....</b>	<b>13</b>

2.1.10.3. Ant Colony Optimization (ACO).....	14
2.1.10.4. Programación con restricciones.....	15
2.2. Marco referencial.....	15
2.2.1. Metaheurísticas aplicadas a un Problema de Asignación de Salones y Horarios a Asignaturas. ....	15
2.2.2. Asignación de horarios de clases universitarias mediante algoritmos evolutivos.....	15
2.2.3. Uso de algoritmos evolutivos para resolver el problema de asignación de horarios escolares en la facultad de Psicología de la universidad de veracruzana.....	16
2.2.4. Análisis, diseño e implementación de un sistema informático para el apoyo al proceso de asignación de la carga académica usando programación con restricciones. ....	16
2.2.5. Metaheurística de optimización mediante colonias de hormigas y aplicaciones. ....	17
2.2.6. Optimización inteligente: técnicas de inteligencia computacional para optimización. ....	17
2.2.7. Ant Colony Optimization. ....	17
2.2.8. Programación de Horarios de Clases y Asignación de Salas para la Facultad de Ingeniería de la Universidad Diego Portales Mediante un Enfoque de Programación Entera.....	18
CAPÍTULO III .....	19
METODOLOGÍA DE LA INVESTIGACIÓN.....	19
3.1. Localización.....	20
3.2. Tipo de investigación. ....	20
3.3. Métodos de investigación.....	20
3.4. Fuentes de recopilación de información. ....	21
3.5. Diseño de la investigación.....	21
3.6. Instrumentos de investigación. ....	21
3.7. Tratamiento de los datos. ....	21
3.8. Recursos humanos y materiales.....	22
3.8.1. Recursos Técnicos. ....	22
3.8.2. Talento Humano. ....	22

3.8.3. Recursos Técnicos. ....	22
3.8.3.1. Hardware. ....	22
3.8.3.2. Software. ....	23
3.8.4. Recursos Financiero. ....	23
CAPÍTULO IV.....	24
RESULTADOS Y DISCUSIÓN.....	24
4.1. Descripción del problema.....	25
4.1.1. Formulación matemática.....	26
4.2. Requerimientos para la implementación de un sistema de generación de horarios.....	27
4.3. Entorno computacional. ....	28
4.4. Diagrama de casos de uso.....	28
4.5. Modelado del sistema.....	38
4.5.1. Módulo de administración. ....	38
4.5.2. Módulo de carga de dato. ....	41
4.5.3. Módulo generación de horarios. ....	43
4.5.3.1. Grafo representativo del problema. ....	43
4.5.3.2. Función heurística.....	44
4.5.3.3. Rastros de feromona. ....	47
4.5.3.4. Valores de guía de la metaheurística.....	47
4.5.3.5. Codificación para resolver las restricciones. ....	55
4.5.3.6. Conjunto de datos. ....	56
4.5.3.7. Función objetivo.....	57
4.5.3.8. Selección de Horas. ....	57
4.5.3.9. Asignación de pesos a docentes. ....	58
4.5.3.10. Evaluación de la hormiga.....	58
4.5.3.11. Evaluación de horarios.....	58
4.1.1.1. Evaporación de la feromona y Actualización de Feromonas.....	59
4.1.1.2. Condiciones de parada. ....	59

4.1.1.3. Estudio experimental.....	60
4.1.2. Módulo de reconstrucción de horario.....	65
CAPÍTULO V .....	64
CONCLUSIONES Y RECOMENDACIONES .....	64
5.1. Conclusiones.....	65
5.2. Recomendaciones.....	67
CAPÍTULO VI.....	68
BIBLIOGRAFÍA .....	68
Bibliografía.....	69
CAPÍTULO VII.....	71
ANEXOS .....	71
Anexo 1. ....	72
Anexo 2. ....	73

## Índice de Tablas

TABLA: 1 Hardware utilizados en el desarrollo de la investigación.....	22
TABLA: 2 Software usado en el desarrollo del sistema de generación de horarios. ....	23
TABLA: 3 Costos de librería utilizados en la presente investigación.....	23
TABLA: 4 Restricciones a tener en cuenta para la evaluación del horario.....	26
TABLA: 5 Restricciones obligatorias para la construcción de horarios. ....	26
TABLA: 6 Requerimientos del sistema. ....	27
TABLA: 7 Caso de uso Inicio de sesión. ....	29
TABLA: 8 Caso de uso Administrar usuarios.....	30
TABLA: 9 Caso de uso Gestionar carrera.....	31
TABLA: 10 Caso de uso Gestionar facultad. ....	32
TABLA: 11 Caso de uso Gestionar docentes.....	33
TABLA: 12 Caso de uso gestionar materias. ....	34
TABLA: 13 Caso de uso manejo del distributivo .....	35
TABLA: 14 Caso de uso generación de horario.....	36
TABLA: 15 Caso de uso manejo de reportes.....	37
TABLA: 16 Estructura de datos de la tabla periodo. ....	38
TABLA: 17 Tipos de datos de la tabla facultad.....	39
TABLA: 18 Tipos de datos de la tabla carrera.....	39
TABLA: 19 Estructura de datos de la tabla paralelo.....	40
TABLA: 20 Formato de asignación a docentes con sus respectivos paralelos y materias. 40	
TABLA: 21 Tipo de docente según su carga y asignación del peso. ....	41
TABLA: 22 Tipo de datos para la tabla tipo materia. ....	42
TABLA: 23 Tipos de clasificaciones para docentes. ....	45
TABLA: 24 Asignación de hora a docente de manera Probabilístico.....	46
TABLA: 25 Asignación de hora a docente de manera Aleatoria. ....	46
TABLA: 26 Parámetros guías para la meta-heurística.....	47
TABLA: 27 Datos representativos para la distribución del horario.....	48
TABLA: 28 Matriz de feromona inicial en el proceso de distribución del horario.....	49
TABLA: 29 Cálculo del valor de materia para obtener el valor de los caminos del grafo. 49	
TABLA: 30 Representación en valores los posibles caminos del grafo. ....	50
TABLA: 31 Probabilidad de ir desde un docente x a un docente y. ....	51
TABLA: 32 Docentes asignados al horario propuesto en le grafo de la hormiga 2.....	53
TABLA: 33 Matiz de feromona llena con su respectiva evaporación y contribución. ....	54

TABLA: 34 Representación en valores los posibles caminos del grafo de la hormiga 2 después de la primera iteración. ....	55
TABLA: 35 Matriz de feromona. ....	59
TABLA: 36 Prueba 1 de 50 iteraciones para obtener un horario óptimo. ....	61
TABLA: 37 Prueba 2 de 100 iteraciones para obtener un horario óptimo. ....	62
TABLA: 38 Prueba 3 de 300 iteraciones para obtener un horario óptimo. ....	63
TABLA: 39 Prueba 4 de 500 iteraciones para obtener un horario óptimo. ....	64

### **Índice de figuras**

FIGURA: 1 Formato .xls para datos de docentes. ....	42
FIGURA: 2 Formato .xls para datos malla. ....	43
FIGURA: 3 Grafo de asignación por docente. ....	44
FIGURA: 4 Valores de probabilidad de selección de un docente. ....	51
FIGURA: 5 Primera ruta seleccionada para encontrar una solución. ....	52
FIGURA: 6 Ruta completa de la hormiga 1 en la selección de docentes. ....	52
FIGURA: 7 Docentes asignados al horario propuesto en le grafo de la hormiga 1. ....	53
FIGURA: 8 Ruta completa de la hormiga 2 en la selección de docentes. ....	53
FIGURA: 9 Promedio de 50 iteraciones. ....	61
FIGURA: 10 Promedio de 100 iteraciones. ....	62
FIGURA: 11 Promedio de 300 iteraciones. ....	63
FIGURA: 12 Promedio de 500 iteraciones. ....	64
FIGURA: 13 Esquema de la base de datos. ....	72
FIGURA: 14 Pantalla principal de la aplicación web. ....	73
FIGURA: 15 Administración gestión de datos para el horario. ....	73
FIGURA: 16 Asignación de docentes a paralelos. ....	74
FIGURA: 17 Carga de datos de docentes de un archivo en formato .xls ....	74
FIGURA: 18 Carga de datos de materias de un archivo en formato .xls. ....	75
FIGURA: 19 Asignación de horas a un docente y paralelo específico. ....	75
FIGURA: 20 Generación de horario. ....	76
FIGURA: 21 Reconstrucción de horario. ....	76
FIGURA: 22 Inicio de sesión para realizar gestiones en la aplicación web. ....	77



### Código Dublin

Título:	Optimización basada en Colonia de Hormigas para resolver problemas asignación de horarios en la Universidad Técnica Estatal de Quevedo.				
Autor:	Darwin Tullmo, John Oñate				
Palabras clave:	Metaheurística	Optimización	Probabilística	Colonia de Hormiga	
Fecha de publicación:	08-dic-16				
Editorial:	Quito: EPN, 2016.				
Resumen:	<p>Resumen .- En el presente proyecto de investigación se muestra la descripción y desarrollo de la aplicación web, que aplica la metaheurística de optimización Colonia de Hormiga, para la generación de horarios de la Universidad Técnica Estatal de Quevedo de la Facultad Ciencias de la Ingeniería, con la finalidad de disminuir la insatisfacción de los docentes y estudiantes durante el proceso de gestión de horarios, Mediante el proceso de generación de horario se implementan cuatro módulos, el de Administración y el de Carga de Datos permiten gestión de la información y a su vez asignar a los docentes a un determinado paralelo, Generación de Horario en el módulo se puede generar los horarios y Reconstrucción de Horario se los utilizan para modificar el horario generado por paralelo.</p> <p><b>Abstract .-</b> The present research project shows the description and development of the web application, which applies the optimization metaheuristics Colonia de Hormiga, for the generation of schedules of the Quevedo State Technical University of the Faculty of Engineering Sciences, in order to To reduce the dissatisfaction of teachers and students during the time management process. Through the process of generation of time, four modules are implemented, the Administration and the Data Loading program, which allow the management of information and, in turn, A certain parallel, Generation of Time in the module can be generated the schedules and Reconstruction of Time are used to modify the schedule generated by parallel.</p>				
Descripción:	97 hojas : dimensiones, 29 x 21 cm + CD-ROM 6162				
URI:					

## Introducción

Las instituciones educativas al iniciar con un nuevo ciclo académico establecen nuevos horarios para que los docentes impartan clases, las cuales presentan constantes cambios e implica un considerable uso del tiempo y está expuesta a equivocaciones cuando se realiza de forma manual.

La Universidad Técnica Estatal de Quevedo lleva un constante avance en la automatización de actividades por lo que se busca alcanzar los mejores resultados en el desarrollo de la ciencia y la tecnología, surge la necesidad de un nuevo sistema automático de generación de horarios que cumpla con un conjunto de requerimientos impuestas por la Universidad.

Los métodos exactos han demostrado poca ventaja ante problemas que tienen gran tamaño de información, es precisamente por este motivo el problema ha sido abordado por diferentes técnicas de optimización; la metaheurística es uno de los modelos que mejor solución dan, por lo cual nuestro problema se centrara en este tema.

La naturaleza encuentra las mejores soluciones a los problemas mediante la evolución continuada de las especies donde subyace el termino optimizar, la herramienta del presente trabajo se desarrolla en base a una técnica denominada optimización por colonia de hormiga (**Ant Colony Optimization - ACO**). De la cual se puede obtener la mayor ganancia o la menor pérdida de tiempo en la generación de horarios.

Se pone en práctica la metodología de Problemas de Satisfacción de Restricciones (**Constraint Satisfaction Problems – CSP**), en conjunto con ACO para el desarrollo de la aplicación web, ya que es efectiva de cierto tipo de problemas, típicamente combinatorios y de optimización.

Para el desarrollo de la aplicación web se hace uso de un software NetBeans de referencia internacional, desarrollado en la comunidad de JAVA mediante la plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos.

Java es la base para casi todo tipo de aplicaciones en red y es el estándar mundial para el desarrollo y distribución de software de la empresa, el contenido basado en la web, juegos

y aplicaciones móviles. Java cuenta con un ecosistema grande y madura con el apoyo de herramientas fuertes.<sup>1</sup>

---

<sup>1</sup> <http://www.oracle.com/>

**CAPÍTULO I**

**CONTEXTUALIZACIÓN DE LA INVESTIGACIÓN**

## **1.1. Problema de investigación.**

### **1.1.1. Planteamiento del problema.**

Hoy en día con el avance tecnológico en el área de las TIC's, es imprescindible contar con herramientas automatizadas de planificación que faciliten la obtención de resultados efectivos y en tiempos relativamente cortos. Las instituciones tanto públicas como privadas requieren diseñar y planificar adecuadamente sus actividades. En este contexto, la Universidad Técnica Estatal de Quevedo UTEQ, está realizando la asignación de actividades de forma manual.

Al realizar la planificación de las actividades de forma manual se requiere de más tiempo y esfuerzo, por lo que la universidad no cuenta con un software dedicado para la gestión de asignación tareas, debido a su magnitud ya que tiene 4 Facultades principales y 1 Facultad a distancia, y cada Facultad tiene de 3 a 12 carreras asignadas respectivamente, en el cual cada carrera cuenta de 3 a 7 paralelos habilitados para cada ciclo o semestre en cada periodo lectivo y 25 a 100 docentes asignados por cada Facultad.

El tiempo para diseñar y planificar la asignación de actividades requiere de dos a tres semanas, en vista de que se debe considerar los requerimientos como el tipo de docente con cargo de gestión u otras funciones que ejerzan, el tipo de materia y el paralelo habilitado, para evitar que más de una asignación tarea requiera de un mismo recurso o se asigne a un mismo periodo de tiempo. Entre otra de las consideraciones se toma más tiempo de lo esperado realizar una distribución equilibrada, de tal forma que los contratiempos generan malestares tanto para estudiantes como para profesores.

### **Diagnóstico.**

En la Universidad Técnica Estatal de Quevedo no existe una solución lo cual mitigue el problema de asignación de tareas dificultando tanto a docentes como a los estudiantes por los malestares generados.

### **Pronóstico.**

Demora en los procesos de asignación de horarios de clases, mal sentir de los docentes y estudiantes por tener constantes modificación en las primeras semanas de cada inicio de ciclo.

### **1.1.2. Formulación del problema.**

¿Cómo realizar el proceso de generación de horarios en la Universidad Técnica Estatal de Quevedo de forma tal que garantice el cumplimiento de los requerimientos necesarios?

### **1.1.3. Sistematización del problema.**

Se propusieron las siguientes sistematizaciones para el problema:

S1: ¿Cómo obtener la información relacionada como facultad, carrera, docentes, materias y paralelos para poder obtener un esquema general necesario para el problema de estudio?

S2: ¿De qué manera se puede optimizar la generación de horarios cumpliendo con los requerimientos?

S3: ¿Cómo incorporar cambios a un horario ya generado para realizar mejoras al proceso?

## **1.2. Objetivos.**

### **1.2.1. General.**

Desarrollar un software que permita encontrar una solución computacional, para generar horarios académicos para la Universidad Técnica Estatal de Quevedo utilizando Optimización basada en Colonia de Hormigas.

### **1.2.2. Específicos.**

- Implementar un módulo para la retroalimentación de toda la información que requiere la generación de horario.
- Obtener una solución computacional basada en la metaheurística de Optimización Colonia de Hormigas.
- Diseñar un módulo para realizar ajustes manuales al horario generado.

## **1.3. Justificación.**

En la Universidad Técnica Estatal de Quevedo especialmente la Facultad Ciencias de la Ingeniería se realizan los horarios a cada inicio de ciclo, tomando un tiempo aproximado de

dos a tres semanas. Adicionalmente, los horarios generados están sujetos a modificaciones debido insatisfacción tanto de los docentes como de los estudiantes. Los cambios en los horarios se producen inclusive cuando las clases ya han iniciado, situación que ha ocasionado retrasos e incomodidad.

Ya que los horarios se generan manualmente en ocasiones no se tienen en cuenta los requerimientos debido a la magnitud de la asignación de horario.

Con la implementación del sistema que permite minimizar el tiempo de generación de horario tomando en cuenta los requerimientos en su generación, aunque no ofrece un horario 100% optimo como todo software basado en probabilidad, pero si uno aceptable.

### **1.3.1. Alcance.**

Tendrá como alcance el análisis y mejora la generación de horarios. Este método heurístico basada en colonia de hormiga será aplicado bajo las condiciones y reglas proporcionadas por la Universidad Técnica Estatal de Quevedo.

Con el método heurístico se desea cubrir los siguientes puntos:

- Reducir el tiempo de generación de horarios para la facultad y eliminar los cruces que pueden darse entre los profesores, materias, paralelos y aulas.
- Permitir la asignación de los horarios cumpliendo con los requerimientos de la facultad como son: regirse a la disponibilidad de asignaturas de cada semestre, respetar las preferencias de la facultad, de las materias y de los profesores en cuanto a horarios de clase, cumplir con el número de sesiones semanales disponible de cada materia, controlar el número de sesiones de clases diarias para un paralelo y para un profesor.
- En caso del Docente contar con carga en otra Unidad, se debe considerar la no asignación en los días y horas indicadas. Esto también aplica a los profesores que dictan clases en otra facultad, si ya tiene asignado horario de trabajo debe de validar que no exista cruce.

## **CAPÍTULO II**

### **MARCO TEÓRICO**



## **2.1. Marco Conceptual.**

En el presente proyecto presenta en especial la metaheurística **ACO**, como los principales conceptos que abarcan la presente investigación y sus fundamentos, así como referencias a otras metaheurísticas.

### **2.1.1. Fundamentos De La Inteligencia Artificial.**

En una primera aproximación podemos definir la IA como una ciencia que tiene como objetivo el diseño y construcción de máquinas capaces de imitar el comportamiento inteligente de las personas. Una rama especializada de la Informática que investiga y produce razonamiento por medio de máquinas automáticas y que pretende fabricar artefactos dotados de la capacidad de pensar. Un saber que se encuadra dentro del proyecto de la modernidad que Kant condensó en la pregunta por el hombre, culmen, por otra parte, del racionalismo que se inicia con Descartes. Ahora bien, lo específico de esta nueva ciencia estriba en el uso de la analogía del ordenador no solamente como método de trabajo e investigación sino también de interpretación de la realidad. Esta orientación ha hecho posible el surgimiento de una nueva rama del saber denominado Ciencias cognitivas que tienen como objeto específico de estudio el análisis y la experimentación sobre el conocimiento visto como un proceso de adquisición, codificación, manipulación, producción y transferencia de nueva información [1].

Por extraño que pueda parecer, lo cierto es que no hay un consenso entre los científicos e ingenieros sobre lo que es la Inteligencia Artificial, y mucho menos se ha llegado a una definición exacta y concisa que nos permita dirimir qué programas son o no inteligentes. El problema es que ni siquiera tenemos la certeza de que seamos capaces de definir qué es la inteligencia (no artificial) [2].

Se dice que los humanos son una especie inteligente: saben hablar, resuelven problemas matemáticos, han llegado a la Luna... pero ¿acaso no son inteligentes el resto de animales que habitan el planeta? En efecto, un perro muestra comportamientos inteligentes e incluso es capaz de expresar sentimientos. ¿Dónde poner el límite de lo que es o no inteligente entonces? [2].

El primer intento de definir la Inteligencia Artificial lo hizo el matemático Alan Turing, que es considerado como el padre de la computación. Este científico inglés es más conocido por su máquina de Turing: una máquina conceptual que utilizó para formalizar los conceptos del modelo computacional que seguimos utilizando hoy día. En concreto demostró que con las operaciones básicas que podía desarrollar su máquina podía codificarse cualquier algoritmo, y que toda máquina capaz de computar tendría las mismas operaciones básicas que su máquina o un súper conjunto de estas [2].

### **2.1.2. Algoritmo.**

En matemáticas, lógica, ciencias de la computación y disciplinas relacionadas, un algoritmo (del griego y latín, dixit algorithmus) es un conjunto prescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien deba realizar dicha actividad [3].

### **2.1.3. Meta-heurística.**

Una Meta-heurística es un método heurístico para resolver un tipo de problema computacional general, usando los parámetros dados por el usuario sobre unos procedimientos genéricos y abstractos de una manera que se espera eficiente. Normalmente, estos procedimientos son heurísticos [3].

### **2.1.4. Los algoritmos evolutivos.**

Los algoritmos evolutivos son una familia de métodos computacionales inspirados por la selección natural que abordan dos temas principales: desarrollo de algoritmos para la optimización y la comprensión de la evolución biológica. El primero es el uso de los algoritmos evolutivos más comunes, siendo una de las principales herramientas para la optimización paralelo. De hecho, se han aplicado con éxito a diversos problemas de optimización, tales como el modelado de las propiedades eléctricas de las neuronas (Achard y De Schutter 2006 ; Druckmann et al. 2007 ; Keren et al. 2009 ; Smolinski y Prinz 2009 ), la escritura automática de los programas para la específica tareas, y el modelado de los mercados financieros [4].

### 2.1.5. Algoritmos Genéticos (AG).

Algoritmos genéticos (GA) como una herramienta para una búsqueda y una metodología de optimización ha llegado a una etapa madura. Ha encontrado muchas aplicaciones útiles en los campos científicos y de ingeniería. La principal razón de este éxito es sin duda debido a los avances que se han hecho en la fabricación de estado sólido de la microelectrónica que, a su vez, condujeron a la proliferación de ampliamente disponible, computadoras de bajo costo y rápidas [5].

La Asamblea General trabaja en el principio darwiniano de selección natural para que el conocido filósofo inglés, Herbert Spencer acuñó la frase "Supervivencia del más apto". Como un optimizador numérico, las soluciones obtenidas por la Asamblea General no son matemáticamente orientadas. En cambio, GA posee una flexibilidad intrínseca y la libertad de elegir deseable optima según las especificaciones de diseño. Los criterios de interés sean no lineales, con restricciones, discretas, multimodal o NP duro, el GA es enteramente igual al desafío. De hecho, debido a la singularidad del proceso evolutivo y la estructura genética de un cromosoma, el mecanismo de procesamiento de GA puede tomar la forma de multiobjetivo y paralelismo. Estos proporcionan una dimensión adicional para soluciones donde otras técnicas pueden han fracasado totalmente. Es, por lo tanto, el objetivo de este libro para reunir material relevante de GA que ya ha sido utilizado y demostrado en varias disciplinas de la ingeniería [5].

### 2.1.6. Colonia de hormigas.

Es común que para buscar solución a problemas complejos se usen métodos inspirados en la naturaleza. El principio del método de optimización de colonia de hormigas (**ACO por sus siglas en inglés, Ant Colony Optimization**), se basa en el estudio del comportamiento que utilizan las hormigas para marcar el camino más corto entre su colonia y el alimento. La mayoría de las especies de hormigas son prácticamente ciegas, por lo cual deben utilizar feromonas para marcar las rutas que van utilizando y luego otras hormigas pueden seguir ese rastro para el regreso a la colonia. Se puede decir entonces que las hormigas siguen el camino que tenga mayor probabilidad de uso o aquel cuya marca de feromona es más potente.

En la implementación de un algoritmo de colonia de hormigas, las posibles soluciones son denominadas hormigas artificiales, que puede ser una solución factible o no. Esta alternativa de solución estará construida en base a las reglas que emulan el comportamiento de las hormigas reales como son la explotación, exploración y evaporación de feromonas. Al comienzo existirá una colonia o un conjunto de soluciones posibles y en cada iteración las hormigas crearán alternativas en base a la información recopilada por las soluciones antecesoras. La cantidad de hormigas es un parámetro importante a definir en el diseño del algoritmo.

Se define además una estructura de memoria que almacena datos numéricos que emulan la feromona de las hormigas reales. Esta estructura, llamada rastro de feromona, guarda el grado de aceptación de hormigas o soluciones anteriores para una variable de estado que se encuentra en una alternativa de solución.

Este método se diferencia de otros métodos heurísticos en que la construcción de las soluciones las realiza en base al aprendizaje ganado anteriormente y que ha sido guardado en la matriz de feromonas, dejando en un segundo plano el impacto que pueda tener esta solución en la función objetivo. Al igual que otros métodos, el algoritmo de colonia de hormigas previene que se pueda converger hacia una solución local, para lo cual existe un proceso denominado evaporación de feromonas, el cual disminuye el rastro de feromona utilizando una función de probabilidad.

Algunas de las ventajas de este método en el problema de asignación de horarios, es que tienen alta probabilidad de encontrar algún óptimo global, pues los métodos convencionales regularmente convergen en óptimos locales. Además, la representación es fácil de analizar y comprender. Entre las desventajas del método de colonia de hormigas el alto grado de incertidumbre en cuanto al tiempo de convergencia del algoritmo y el alto costo de procesamiento que requiere. Al igual que otros métodos heurísticos no asegura el encontrar soluciones exactas, sino solo soluciones óptimas, que muchas veces son suficientes.

### **2.1.7. Asignación dinámica de recursos con técnicas bio inspirada para un sistema de control de temperatura MIMO.**

Como [6] explica que la primera hormiga que va en busca de comida, lo hace sin ningún tipo de información o señal que la guíe, pero al ir avanzando va dejando feromona por la ruta, ejerciendo una acción sobre la decisión de las hormigas precedentes las cuales escogen el camino que posee una mayor concentración de feromona, permitiendo que las hormigas encuentren la ubicación de las fuentes de comida al igual que su hormiguero. Se ha demostrado que los rastros de feromona permiten lentamente la optimización distribuida en la cual cada agente sencillo realiza una pequeña contribución en la búsqueda de la mejor solución.

### **2.1.8. Complejidad algorítmica.**

La Complejidad algorítmica consiste en encontrar la mejor solución en un conjunto discreto (finito o numerable), de aquí es de suma importancia conocer la factibilidad o no de un método (ya sea exacto o no) antes de invertir esfuerzo en una implementación.

El criterio para saber si un problema de Complejidad algorítmica es difícil o no, es la complejidad temporal del algoritmo, la que básicamente relaciona el tiempo de ejecución con el tamaño del procedimiento. Existen muchas opciones de solución para un problema, sin embargo, debemos escoger la mejor secuencia de pasos y con el menor tiempo de ejecución. Con este fin, para cada problema se determina una medida  $N$ , que representa el número de datos a procesar por el programa; entonces el tiempo de ejecución de un programa se mide en función de  $N$ , lo que se designa como  $T(N)$ .

Los VRP pertenecen en su mayoría a la clase NP-Hard (difícil), debido al gran consumo de recursos computacionales para encontrar una solución óptima que crece de forma exponencial en relación al tamaño del problema (cantidad de clientes) y de aquí el origen de la gran motivación académica e investigativa.

Los problemas de decisión se clasifican en clases de complejidad, la clase  $P$  contiene problemas de decisión que pueden ser resueltos en forma relativamente rápida mientras que la clase  $NP$  contiene problemas difíciles de resolver.

**Los problemas NP** – Completo se cree que son los más difíciles de resolver, y tienen la siguiente propiedad: si se encuentra un algoritmo eficaz polinomial para resolver uno de ellos, todos los demás se resolverán polinomialmente y por lo tanto será  $P = NP$ . Los problemas NP – completos pueden definirse como la intersección entre NP y NP -duro.

**La clase NP** – duro, contiene problemas de decisión tan complejos como los de la clase NP, pero no necesariamente forma parte de la clase de complejidad NP.

### 2.1.9. iText Software.

iText Software es un especialista líder en el mundo en las bibliotecas de software PDF programables. Sus soluciones se utilizan en ventas y marketing, jurídica y de gobierno, finanzas, TI, operaciones y recursos humanos por una diversa base de clientes que incluye a las instituciones médicas, bancos, gobiernos y empresas de tecnología. La solución iText primero fue diseñado por el visionario tecnología Bruno Lowagie en 2000 y ha conocido un crecimiento continuo en sus 15 años de existencia, junto con un sano deseo de mantenerse a la vanguardia de la innovación de productos. Se desarrolló a partir de la comunidad de código abierto, lo que es compatible con iText para el día de hoy a través de su modelo de negocio único de doble licencia.<sup>2</sup>

iText es un proyecto de software libre / abierto (F / OSS), que le permite leer a través de la API de código completo para ayudarle mejor en la creación de una solución. F / OSS no significa que esté libre de obligaciones, sólo significa que el código está disponible gratuitamente. Siguiendo las reglas de la Licencia Pública General de Affero (AGPL), es posible utilizar iText sin costo alguno. De lo contrario, es obligatorio comprar una licencia comercial para eliminar las restricciones copyleft.<sup>3</sup>

La licencia AGPL viene con las siguientes restricciones. Los puntos siguientes no son exhaustivas, sino simplemente las restricciones más importantes:

- El usuario no puede desplegarlo en una red sin revelar el código fuente completo de sus propias aplicaciones bajo la licencia AGPL.
- Debe distribuir todo el código fuente, incluyendo su propio producto y las aplicaciones basadas en la Web.

---

<sup>2</sup> <http://itextpdf.com/>

<sup>3</sup> <http://itextpdf.com/>

- Debe dar a conocer las modificaciones realizadas en iText.
- Debe mencionar prominente e incluir iText los derechos de autor y la licencia AGPL en los metadatos del archivo de salida.

## **2.1.10. Algoritmos utilizados para la elaboración del software para generación de horarios.**

### **2.1.10.1. Búsqueda Primero el Mejor.**

Es un caso en el cual se selecciona un nodo para la expansión basada en una función de evaluación  $f(n)$ .

Esta función evaluación devuelve un número que sirve para representar lo deseable o indeseable que sería la expansión de un nodo. Se expande primero aquel nodo que tiene mejor evaluación. Se escoge el que parece ser el mejor, pero puede no serlo.

Un componente clave de estos algoritmos es una función heurística, denotada  $h(n)$ : Se expande el nodo con  $f(n)$  menor

$h(n)$  = valor estimado del camino de coste mínimo desde el nodo  $n$  al nodo objetivo

Componentes:

- 1) Sea  $L$  una Lista de nodos iniciales.
- 2) Sea  $N$  el nodo más cercano a la meta (el mejor). Si  $L$  está vacía, falla.
- 3) Si  $N$  es la meta. Regrese la trayectoria desde el nodo inicial al nodo  $N$ .
- 4) Si  $N$  no es una meta. Buscar los hijos de  $N$ , colocarlos en  $L$ , etiquetándolos con la trayectoria desde el nodo inicial. Retorne al paso 2.

### **2.1.10.2. Búsqueda local guiada.**

Consiste básicamente en una la iteración de un procedimiento de búsqueda local y en la modificación de la función objetivo cada vez que termina la búsqueda local, mediante la inclusión de términos de penalización.

La metaheurística de GLS es una estrategia que guía la exploración del espacio de búsqueda con:

- Información de la estructura del problema
- Información del propio proceso de búsqueda.

Tanto el conjunto de soluciones como la estructura de la vecindad se mantienen inalterados; en cambio, la función objetivo modifica con la intención de ayudar al procedimiento de búsqueda local a escapar de los óptimos locales.

La metaheurísticas de Búsqueda Local Guiada (GLS, Guided Local Search) es una estrategia que guía la exploración del espacio de búsqueda con información de la estructura del problema y del propio proceso de búsqueda. Consiste básicamente en una la iteración de un procedimiento de búsqueda local y en la modificación de la función objetivo cada vez que termina la búsqueda local, mediante la inclusión de términos de penalización [6].

#### **2.1.10.3. Ant Colony Optimization (ACO).**

Optimización de colonia de hormigas (ACO) es una metaheurística basada en la población que puede ser utilizada para encontrar soluciones aproximadas a problemas de optimización difíciles.

Consiste en una técnica probabilística para solucionar problemas computacionales que pueden reducirse a buscar los mejores caminos o rutas en grafos.

- Las hormigas tienden a elegir los caminos con mayor cantidad de feromona
- Tendencia a acumular más feromona en los caminos más cortos
- La volatilidad de la feromona ayuda a descartar los caminos menos transitados

Las hormigas artificiales (en adelante hormigas) incrementalmente construyen soluciones moviendo en el gráfico. El proceso de construcción solución es estocástico y está sesgada por un modelo de feromonas, es decir, un conjunto de parámetros asociados con los componentes del gráfico (ya sean nodos o bordes) cuyos valores se modifican en tiempo de ejecución por las hormigas.



#### **2.1.10.4. Programación con restricciones.**

La Programación por restricciones es un paradigma de la programación en informática, donde las relaciones entre las variables son expresadas en términos de restricciones (ecuaciones). Actualmente es usada como una tecnología de software para la descripción y resolución de problemas combinatorios particularmente difíciles, especialmente en las áreas de planificación y programación de tareas (calendarización).

## **2.2. Marco referencial.**

### **2.2.1. Metaheurísticas aplicadas a un Problema de Asignación de Salones y Horarios a Asignaturas.**

Según [7] se realiza una actualización del estado del arte de los problemas de programación de horarios y se construye un motor de metaheurísticas para la herramienta (software) de apoyo para la asignación de grupos y asignaturas a los salones, que semestralmente realiza la Bedelía de la Facultad de Ingeniería. El motor que construyeron incluye algunas de las metaheurísticas más importantes como Algoritmos Genéticos, Búsqueda Tabú y dentro de Optimización por Colonia de Hormigas los algoritmos Ant System, Ant Colony System y MAX – MIN Ant System.

Es un conjunto de procesos que se siguieron para unir múltiples herramientas para un fin en común utilizando como apoyo herramientas de software y las diferentes metaheurísticas para poder construir una aplicación que pueda asignar los diferentes horarios en una facultad como la que se realizó en Montevideo Uruguay en 2005.

### **2.2.2. Asignación de horarios de clases universitarias mediante algoritmos evolutivos.**

De acuerdo con [8] el problema de asignación de horarios en universidades radica en programar en un horizonte de planificación (generalmente una semana) las asignaturas que se dictan en un periodo, para las distintas asignaturas que la requieren, considerando los profesores necesarios en cada asignatura, utilizando algoritmo evolutivo se plantea una

solución mediante procedimientos de optimización metaheurística del problema timetabling propuesto de algoritmos genéticos.

La herramienta que utilizo [8] fue MATLAB, por la ventaja más importante al utilizar ese programa, por la eficiencia que posee al trabajar con grandes cálculos numéricos en vectores y matrices, las pruebas que se realizaron al programa con 10 generaciones, 50 generaciones, y 100 generaciones y con un operador de mutación de 80%, mientras que el operador de recombinaciones 20%, para las pruebas se encontraron soluciones optimas, sin aplicar los operadores genéticos para resolver el problema, debido a la gran cantidad de salones por lo que se restringió esa variable, para realizar un análisis y hallar la solución con un margen de error relativo en un tiempo computacional inferior a los 10 minutos.

### **2.2.3. Uso de algoritmos evolutivos para resolver el problema de asignación de horarios escolares en la facultad de Psicología de la universidad de veracruzana.**

De acuerdo a [10] probó con diversas combinaciones de sus parámetros (tamaño de población, número de generaciones, porcentaje de cruce y porcentaje de mutación). Los resultados obtenidos con probabilidad de mutación del 10%, cruce del 98%, número de individuos 100 y número de generaciones 1000, mismas que ofrecieron un mejor resultado con la función de calidad.

Es un conjunto de procesos que se siguieron para unir múltiples herramientas para un fin en común utilizando como apoyo herramientas de software y las diferentes metaheurísticas para poder construir una aplicación que pueda asignar los diferentes horarios en una facultad como la que se realizó en Venezuela 2007

### **2.2.4. Análisis, diseño e implementación de un sistema informático para el apoyo al proceso de asignación de la carga académica usando programación con restricciones.**

De acuerdo a [11] realizan la combinación de datos mediante técnicas de programación con restricciones para generar una o varias soluciones o alternativas que ayuden al directivo a escoger el horario que mejor se acomode a los requerimientos.

“Después de combinar de combinar datos y aplicando técnicas de programación con restricciones sobre ellos, se logró que la aplicación generara una o varias alternativas que el usuario elija la(s) que le permita(n) una mejor distribución de la carga académica” [11]

### **2.2.5. Metaheurística de optimización mediante colonias de hormigas y aplicaciones.**

De acuerdo con [9] dice que las feromonas forman un sistema indirecto de comunicación química entre animales de una misma especie, que transmiten información acerca del estado fisiológico, reproductivo y social, así como la edad, el sexo y el parentesco del animal emisor, las cuales son recibidas en el sistema olfativo del animal receptor, quien interpreta esas señales, jugando un papel importante en la organización y la supervivencia de muchas especies.

### **2.2.6. Optimización inteligente: técnicas de inteligencia computacional para optimización.**

Menciona [11] A menudo los problemas de optimización combinatoria compleja son muy difíciles de resolver en la práctica, ya que muchos, como el problema de optimización de la distribución, pertenecen a la clase de problemas NP-duros (es decir, no existe un algoritmo conocido que los resuelva en un tiempo polinomial), por lo que para solucionar dichos problemas es necesario entrar en el campo de la teoría de las Ciencias de la Computación. Si no se cuenta con una gran cantidad de casos resueltos con los que se pueda comparar y fundamentar la efectividad del algoritmo de optimización de colonia de hormigas, es necesario simular los procesos para poder hacer una comparación y determinar dicha eficiencia.

### **2.2.7. Ant Colony Optimization.**

El Ant System, desarrollado por [13] fue el primero de varios algoritmos basados en hormigas en ser desarrollado; además, es prototipo de varios algoritmos que conforman el paradigma de la optimización por colonia de hormigas (Ant Colony Optimization–ACO). El Ant System posee un grupo de agentes computacionales concurrentes y asíncronos que se mueven por los estados de un problema que corresponden a soluciones parciales. Los agentes se mueven aplicando una política local de decisión estocástica basada en dos

parámetros llamados el rastro de feromona ( $\tau$ ) y el atrayente ( $\eta$ ). Al moverse, cada hormiga construye una solución al problema de manera incremental. Cuando una hormiga completa una solución, o durante la etapa de construcción de la misma, la evalúa y modifica el valor del rastro de feromona en los componentes usados en ella.

#### **2.2.8. Programación de Horarios de Clases y Asignación de Salas para la Facultad de Ingeniería de la Universidad Diego Portales Mediante un Enfoque de Programación Entera.**

De acuerdo al modelo propuesto de [15] entrega de manera rápida y eficiente los horarios y asignaciones de sala de clase satisfaciendo todos los requerimientos obligatorios y condiciones deseables para la Facultad en un tiempo menor a los 5 minutos, son manejadas implícitamente mediante las variables de decisión, para su resolución utiliza la generación de columnas y la implementación de un algoritmo tipo branch-and-price, donde la función objetivo representa la minimización de clases auxiliares asignadas a bloques horarios que no pertenecen al día miércoles más la cantidad de clases asignadas al Auditorio de la Facultad (AU).

**CAPÍTULO III**

**METODOLOGÍA DE LA INVESTIGACIÓN**

En el presente proyecto se integra la localización y se utilizó las siguientes metodologías de investigación:

### **3.1. Localización.**

El proyecto se localiza en Quevedo, Provincia de Los Ríos, en la Universidad Técnica Estatal de Quevedo Av. Quito km. 1 1/2 vía a Santo Domingo de los Tsachilas.

### **3.2. Tipo de investigación.**

En el presente proyecto de investigación se lo clasifica como una investigación tipo descriptiva y experimental debido a su ámbito de estudio.

En la Descriptiva se describe los procesos que se llevan a la hora de detallar el esquema del proyecto y sus problemas a solucionar.

Experimental se plantean nuevas formas de mejorar los resultados en el proceso de manipulación de variables y realizando pruebas de su funcionalidad.

### **3.3. Métodos de investigación.**

Mediante el estudio inicial nos llevó a deducir la metodología de investigación y la técnica de investigación:

#### **Método de observación.**

Se utilizó principalmente lo que se podía visualizar muchos casos de docentes y estudiantes con problemas de choques de horarios y falta de asignación de cursos.

#### **Técnicas de la investigación.**

La técnica es indispensable en el proceso de la investigación científica, ya que integra la estructura por medio de la cual se organiza la investigación, La técnica pretende los siguientes objetivos en el cual utilizaremos la siguiente:

#### **La entrevista.**

La entrevista es una técnica de recopilación de información mediante una conversación profesional, el cual se realizó con los docentes encargados de la gestión de horarios para la Universidad Técnica Estatal de Quevedo.

### **3.4. Fuentes de recopilación de información.**

Como fuente de recopilación de la información tenemos a la Universidad Técnica Estatal de Quevedo, que tiene la información de los docentes, materias, cursos y estudiantes que la conforman.

### **3.5. Diseño de la investigación.**

Se diseñó la investigación con el director de proyecto, encargados de la gestión de horarios se seleccionaron los aspectos pertinentes a esta investigación, así como los aspectos subyacentes a la misma como, por ejemplo, la definición de los docentes con prioridad, el establecimiento de los grupos de experimentación etc.

### **3.6. Instrumentos de investigación.**

Deben ser instrumentos válidos y confiables, cuya función es recoger datos o información, que serán luego los resultados relevantes. Mide las variables de investigación, logra observaciones del objeto de estudio y valora los sucesos u objetos de interés para la investigación. Sirve para ordenar las observaciones y mediciones, para su posterior análisis. Los instrumentos más usados son: entrevistas, encuestas, observación directa, procedimientos experimentales, análisis de documentos, registros. Los instrumentos a aplicar dependen de las definiciones anteriores sobre el tipo, método y diseño de la investigación.

### **3.7. Tratamiento de los datos.**

No se utiliza un tratamiento de datos complejo ya que solo se solicita datos concretos en una hoja de Excel con extensión .xls, para a partir de ellas tomar como base y poder realizar las operaciones con la metaheurística colonia de hormigas que permite generar un horario que incumpla con menos restricciones.

### 3.8. Recursos humanos y materiales.

Se detalla el recurso técnico, humano y presupuesto que se utilizó en la presente investigación.

#### 3.8.1. Recursos Técnicos.

Es la suma de conocimientos obtenidos por educación, estudio y experiencia volcados a una disciplina específica de una ciencia (industrial, científica, arte, etc.) Estos recursos tienen un carácter más empírico que racional.

#### 3.8.2. Talento Humano.

- Tullmo Toapanta Darwin Abel
- Oñate Espinoza John Alexander
- Tutor de proyecto de investigación

#### 3.8.3. Recursos Técnicos.

##### 3.8.3.1. Hardware.

A continuación, se enlista los equipos utilizados en la realización del proyecto de investigación.

*TABLA: 1 Hardware utilizados en el desarrollo de la investigación.*

Cantidad	Equipo	Descripción
1	PORTÁTIL HP	Core I3 2.0 Ghz 6 GB RAM 512 GB DISCO DURO DVD REWRITER
1	PC ESCRITORIO	Core duo 2.0 Ghz y Superior 4 GB RAM O SUPERIOR 560 GB DISCO DURO CD REWRITER MONITOR, TECLADO, MOUSE

FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES



### 3.8.3.2. Software.

A continuación, se enlista los programas utilizados en la realización del proyecto de investigación.

*TABLA: 2 Software usado en el desarrollo del sistema de generación de horarios.*

<b>Tipo</b>	<b>Descripción</b>
<b>Sistema operativo</b>	<b>Windows 7, Windows 10</b>
<b>NetBeans 8.1</b>	<b>NetBeans IDE es un producto libre y gratuito sin restricciones de uso. NetBeans es un entorno de desarrollo de código libre.</b>
<b>MySQL</b>	<b>MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario de código abierto (RDBMS)</b>
<b>Tomcat 8</b>	<b>Tomcat como servidor web es de código abierto y está programado en Java.</b>
<b>Microsoft Office</b>	<b>Word, Excel, PowerPoint, Project</b>
<b>astah-professional-7_1</b>	<b>Este software permite la creación de diagramas de casos de uso.</b>
<b>FUENTE: CONOCIMIENTO PROPIO</b>	
<b>ELABORADO: LOS AUTORES</b>	

### 3.8.4. Recursos Financiero.

A continuación, se enlista los recursos financieros que se requirió.

*TABLA: 3 Costos de librería utilizados en la presente investigación.*

<b>CANT.</b>	<b>DESCRIPCIÓN</b>	<b>PRECIO UNITARIO</b>	<b>TOTAL</b>
<b>68</b>	<b>Impresiones Blanco y Negro</b>	<b>0,10</b>	<b>6,80</b>
<b>25</b>	<b>Impresiones Color</b>	<b>0,50</b>	<b>12,50</b>
<b>204</b>	<b>Copias</b>	<b>0,03</b>	<b>6,12</b>
<b>3</b>	<b>Anillados</b>	<b>1,00</b>	<b>3,00</b>
<b>3</b>	<b>CDs</b>	<b>1,5</b>	<b>4,50</b>
<b>TOTAL</b>			<b>\$32,92</b>
<b>FUENTE: CONOCIMIENTO PROPIO</b>			
<b>ELABORADO: LOS AUTORES</b>			

**CAPÍTULO IV**

**RESULTADOS Y DISCUSIÓN**

En este capítulo se presentan los resultados obtenidos en las pruebas y los procesos de generación de horario que aporta en la implementación del sistema. Se comienza indicando la descripción del problema, formulación matemática y casos de uso para medir el desempeño del algoritmo; posteriormente se indica el modelado del sistema el cual se divide en cuatro módulos donde se detallan las características de cada uno de ellos.

#### **4.1. Descripción del problema.**

El problema a mencionar se trata como un problema de planificación de horarios. Este problema representa un modelo complejo por la cantidad de restricciones que se deben de cumplir para generar un horario de clases para las facultades de la universidad siguiendo con un conjunto de requerimientos.

En nuestro caso lo definimos como un esquema de satisfacción de restricciones donde el objetivo es encontrar una distribución de horario que incumpla la menor cantidad de restricciones.

En la presente investigación se busca la forma de distribuir un horario utilizando la meta-heurística colonia de hormiga, para brindar una solución al problema de distribución de las horas de todos los docentes en los diferentes días de la semana.

- No debe existir cruce de horas en horario de docente
- No debe existir cruce de horas en materias de un mismo módulo
- Debe existir reservas de horas o de días para generación de horario para los docentes, Ejemplo un profesor a tiempo parcial puede solicitar que se asigne el dictado de clases en un determinado día a una determinada hora. (Mayor prioridad)
- Los Docentes con cargo de Gestión se debe priorizar el dictado de clases en las primeras horas
- Priorizar la no asignación de dos o más materias profesionales en un mismo día
- Priorizar el dictado de materias profesionales en las primeras horas. (Menor prioridad que 4)
- En caso del Docente contar con carga en otra Unidad, se debe considerar la no asignación en los días y horas indicadas. Esto también aplica a los profesores que

dictan clases en otra facultad, si ya tiene asignado horario de trabajo debe de validar que no exista cruce.

#### 4.1.1. Formulación matemática.

Se realizó la siguiente formulación matemática para la generación de horarios:

La siguiente función objetivo determina minimizar el incumplimiento de 4 restricciones de la Tabla # 4 que evalúan el horario sujeto a las restricciones obligatorias de la Tabla # 5.

$$Fo\ min = \sum_{i=1}^4 W_i R_i$$

Donde  $R_i$  son las restricciones para evaluar el horario y  $W_i$  determina la importancia de la restricción.

*TABLA: 4 Restricciones a tener en cuenta para la evaluación del horario.*

Nº	Descripción	Peso (W)
R1	Cantidad de horas por materias sin asignas en el horario.	<b>0,4</b>
R2	Cantidad de materia asignada al horario que exceden las 3 horas seguidas.	<b>0,3</b>
R3	Cantidad de docentes con diferentes materias que son asignados en el horario en un mismo paralelo y mismo día.	<b>0,2</b>
R4	Cantidad de docentes con cargo de decano (1) y coordinador (2) que están asignados fuera de las primeras horas.	<b>0,1</b>

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

También existen restricciones obligatorias que toda solución tiene que cumplir la cual se presenta a continuación.

*TABLA: 5 Restricciones obligatorias para la construcción de horarios.*

Código:	Detalle:
R1	No debe existir cruce de horas en horario de docente
R2	No debe existir cruce de horas en materias de un mismo módulo
R3	Debe existir reservas de horas o de días para generación de horario para los docentes, Ejemplo un profesor a tiempo parcial puede solicitar que se asigne el dictado de clases en un determinado día a una determinada hora. (Mayor prioridad)
R4	Los Docentes con cargo de Gestión se debe priorizar el dictado de clases en

	las primeras horas
R5	Priorizar la no asignación de dos o más materias profesionales en un mismo día
R6	Priorizar el dictado de materias profesionales en las primeras horas. (Menor prioridad que 4)
R7	En caso del Docente contar con carga en otra Unidad, se debe considerar la no asignación en los días y horas indicadas. Esto también aplica a los profesores que dictan clases en otra facultad, si ya tiene asignado horario de trabajo debe de validar que no exista cruce.

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

## **4.2. Requerimientos para la implementación de un sistema de generación de horarios.**

Se plantearon los requerimientos necesarios que se utilizan en ambientes web como:

*TABLA: 6 Requerimientos del sistema.*

Contenido	Descripción
Gestor de base de datos.	Contiene toda la información requerida para la generación de horarios
Servidor de Web (Apache Tomcat Vs 8)	Aloja las páginas web y gestiona los aplicativos y sus demandas.
Formulario de carga de datos de docentes	Importa los datos desde una hoja de cálculo .xls con formato preestablecido para docentes.
Formulario de carga de datos de las mallas curriculares (materias)	Importa los datos desde una hoja de cálculo .xls con formato preestablecido para las mallas curriculares (materias).
Formularios de Administración	Gestionan la información de periodo, facultad, carrera, docentes, mallas curriculares.
Formularios de Asignación clases	Se puede reservar una hora específica para un docente en el horario de clases.
Pantallas de visualización de horario	Se obtiene un horario completo de toda la

	facultad.
Formulario de reconstrucción de horario	Permite el cambio manual del horario, mover horas asignadas con sus respectivas validaciones.
Reportes	Visualización un horario específico, docentes por curso, otros.
<b>FUENTE: CONOCIMIENTO PROPIO</b>	
<b>ELABORADO: LOS AUTORES</b>	

### 4.3. Entorno computacional.

El sistema está programado en lenguaje Java (JSP) que es multiplataforma y como gestor de base de datos MySQL.

Las pruebas se desarrollaron en PCs con sistema operativo Windows 8, procesador Intel de 2.00 GHz y 4 GB de memoria RAM, en diferentes navegadores, obteniendo resultados satisfactorios en la usabilidad del sistema de generación de horarios.

### 4.4. Diagrama de casos de uso.

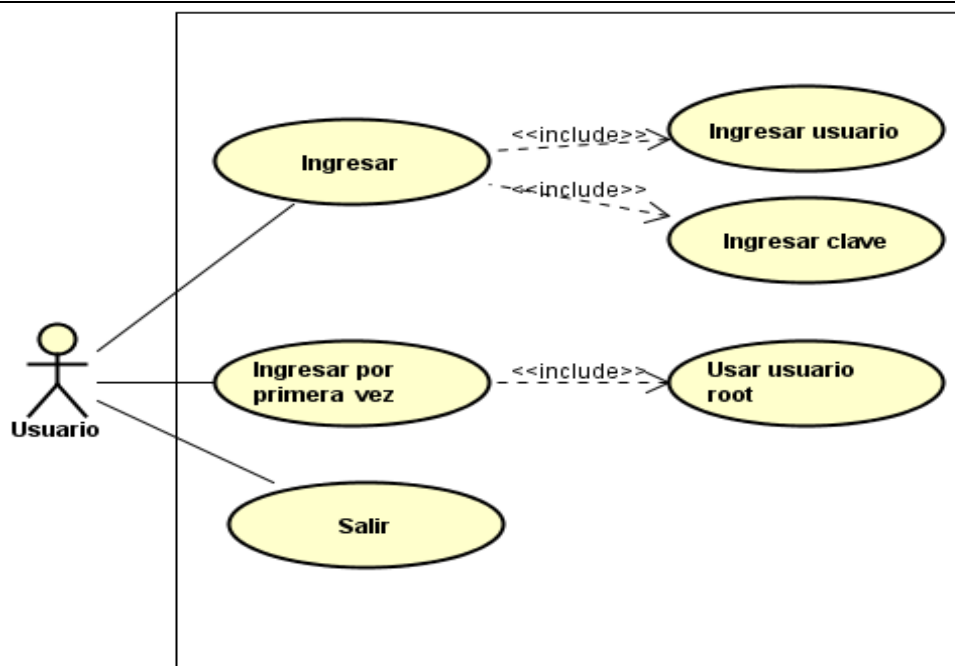
El diagrama de casos de uso representa una perspectiva del lado del Usuario final que manipula el sistema web desarrollado, además de las formas como interactúa el usuario con el sistema y sus elementos de las cuales se definen dos tipos de usuarios:

**Usuario administrador:** Una persona (docente) asignada para realizar gestiones de la información.

**Usuario:** Todos los que visitan el sistema web (docentes, estudiantes, y público en general)

Para modelar la funcionalidad del sistema se utiliza uno de los diversos elementos gráficos UML denominado diagrama de casos de uso conformado del sistema, casos de uso, actores y las relaciones.

TABLA: 7 Caso de uso Inicio de sesión.



**Caso de uso:** Inicio de sesión

**Identificador:** CU01

**Actores:** Usuario Administrador

**Propósito:** **Ingresar:** Permitir al usuario especificar su nombre y contraseña para iniciar sesión en el sistema.

**Ingresar por primera vez:** Usuario por defecto cuando el sistema recién está instalado

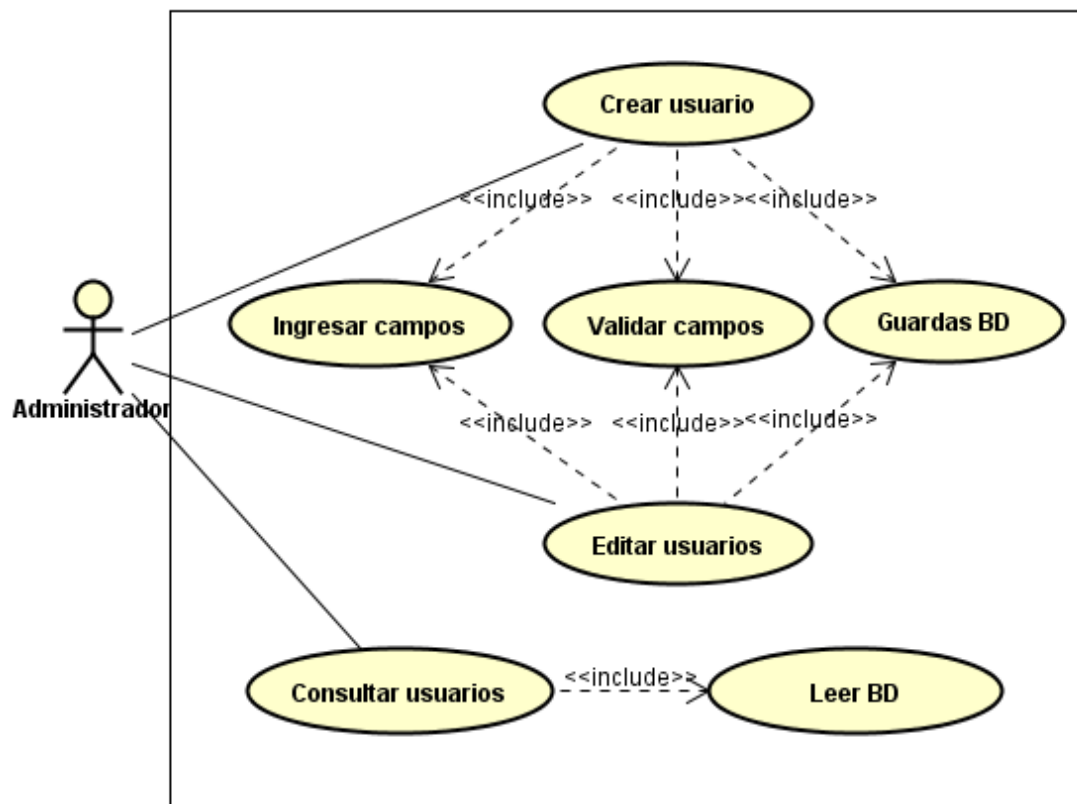
**Salir:** Cuando el usuario

**Resumen:** Este formulario permite que un usuario registrado ingrese sus datos como usuario y contraseña para tener acceso a las funciones del sistema.

**Pre-Requisitos:** El usuario debe estar registrado en el sistema.

**FUENTE:** CONOCIMIENTO PROPIO  
**ELABORADO:** LOS AUTORES

TABLA: 8 Caso de uso Administrar usuarios.



**Caso de uso:** Administrar usuarios

**Identificador:** CU02

**Actores:** Administrador

**Propósito:** **Crear usuario:** Permite agregar nuevo usuario en el sistema  
**Editar usuario:** Modificar los usuarios en el sistema  
**Consultar usuarios:** Permite obtener un listado de los usuarios en el sistema

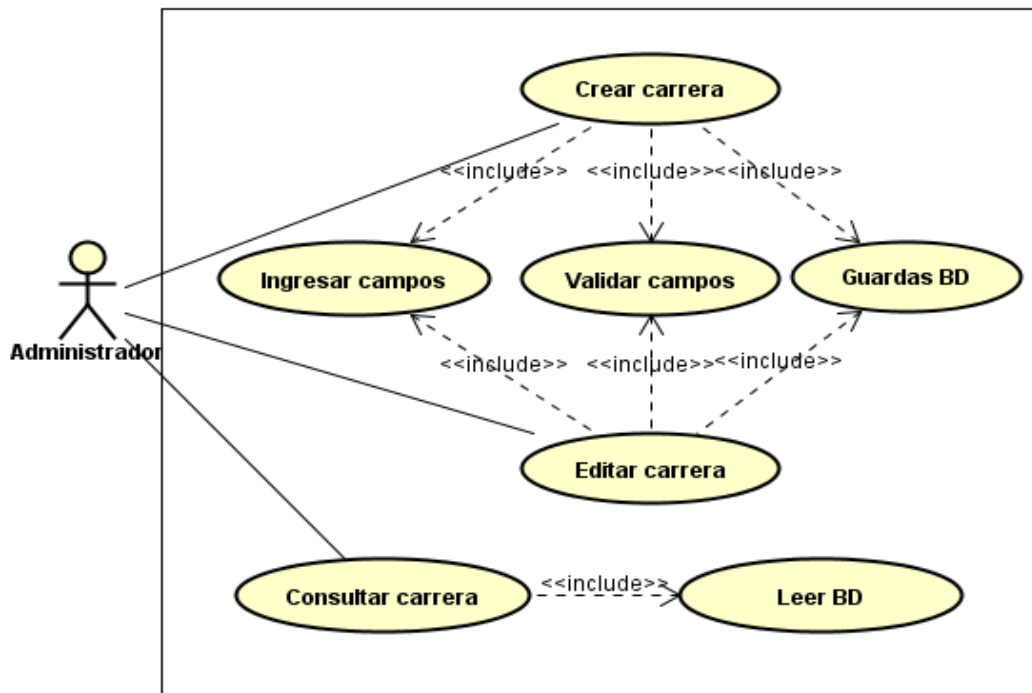
**Resumen:** Este formulario permite una gestión de usuarios en el sistema

**Pre-Requisitos:** El usuario administrador debe estar registrado en el sistema.

FUENTE: CONOCIMIENTO PROPIO  
 ELABORADO: LOS AUTORES



TABLA: 9 Caso de uso Gestionar carrera.



**Caso de uso:** Gestionar carrera

**Identificador:** CU03

**Actores:** Usuario Administrador

**Propósito:**

**Crear carrera:** Permite agregar nuevas carreras en el sistema

**Editar carrera :** Modificar las carreras en el sistema

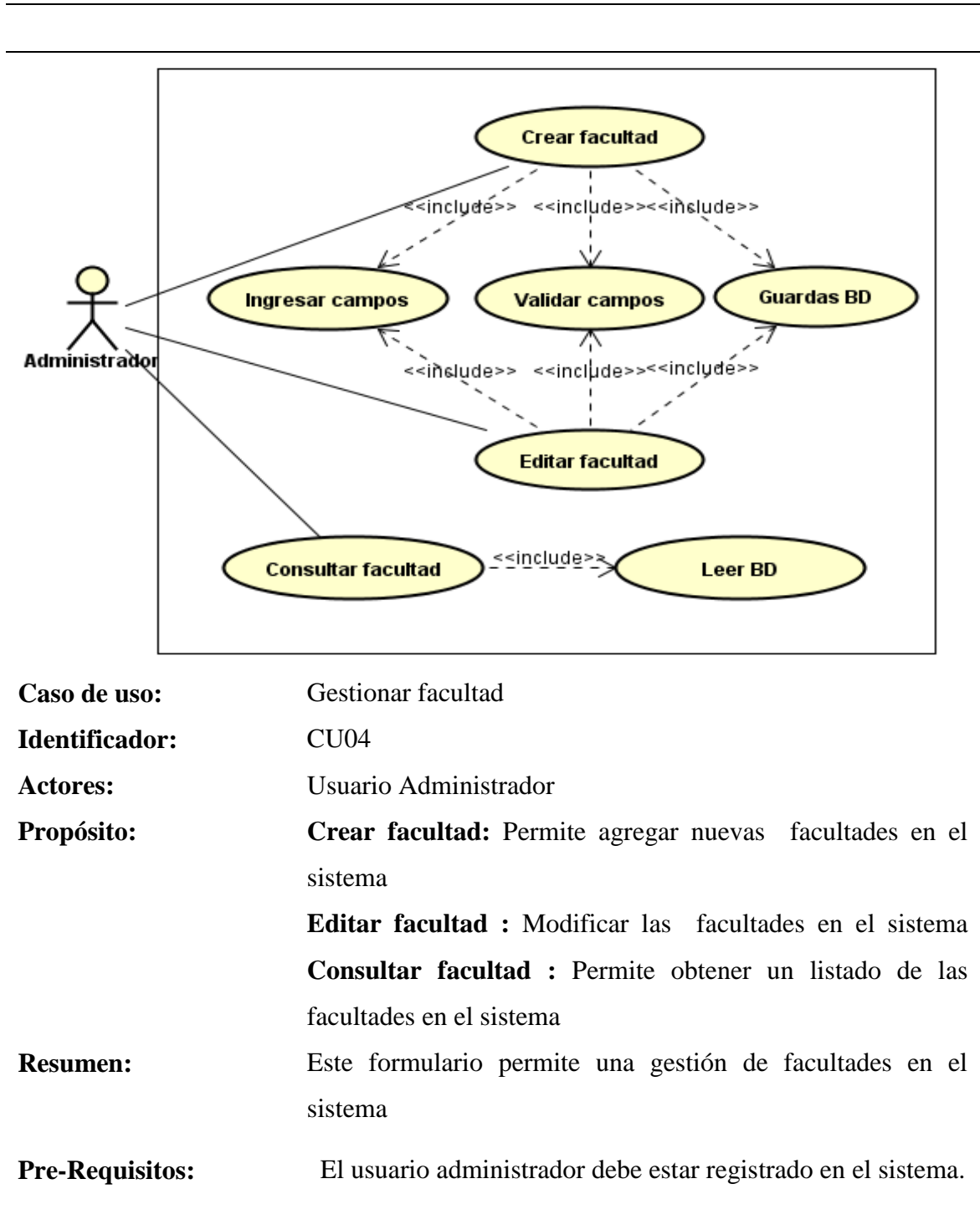
**Consultar carrera :** Permite obtener un listado de las carreras en el sistema

**Resumen:** Este formulario permite una gestión de carreras en el sistema

**Pre-Requisitos:** El usuario administrador debe estar registrado en el sistema.

**FUENTE:** CONOCIMIENTO PROPIO  
**ELABORADO:** LOS AUTORES

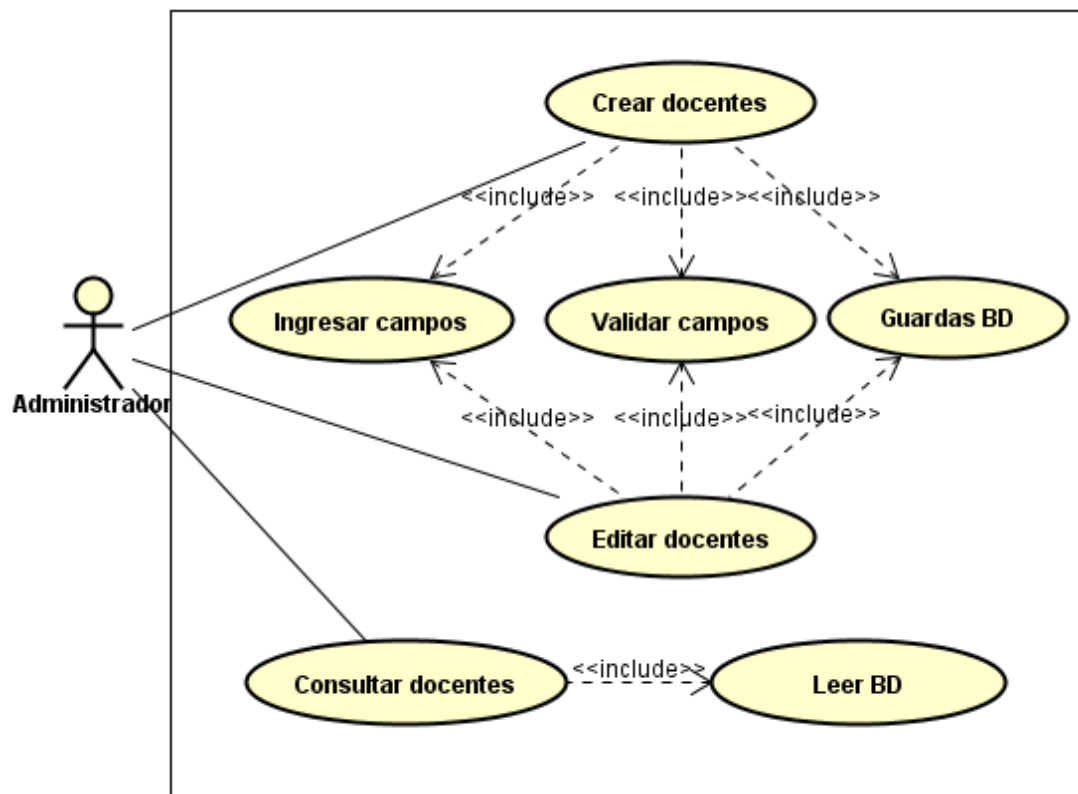
TABLA: 10 Caso de uso Gestionar facultad.



<b>Caso de uso:</b>	Gestionar facultad
<b>Identificador:</b>	CU04
<b>Actores:</b>	Usuario Administrador
<b>Propósito:</b>	<b>Crear facultad:</b> Permite agregar nuevas facultades en el sistema <b>Editar facultad :</b> Modificar las facultades en el sistema <b>Consultar facultad :</b> Permite obtener un listado de las facultades en el sistema
<b>Resumen:</b>	Este formulario permite una gestión de facultades en el sistema
<b>Pre-Requisitos:</b>	El usuario administrador debe estar registrado en el sistema.

FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES

TABLA: 11 Caso de uso Gestionar docentes.



**Caso de uso:** Gestionar docentes

**Identificador:** CU05

**Actores:** Usuario Administrador

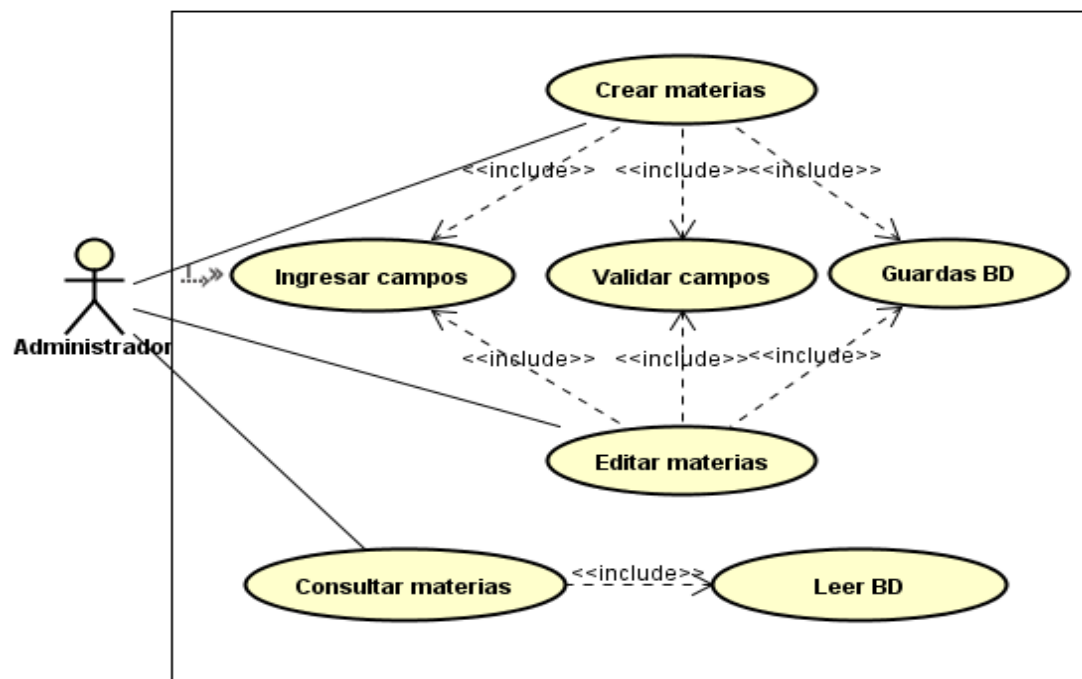
**Propósito:** **Crear docentes :** Permite agregar nuevos docentes en el sistema  
**Editar docentes:** Modificar los docentes en el sistema  
**Consultar docentes:** Permite obtener un listado de los docentes en el sistema

**Resumen:** Este formulario permite una gestión de los docentes en el sistema

**Pre-Requisitos:** El usuario administrador debe estar registrado en el sistema.

**FUENTE:** CONOCIMIENTO PROPIO  
**ELABORADO:** LOS AUTORES

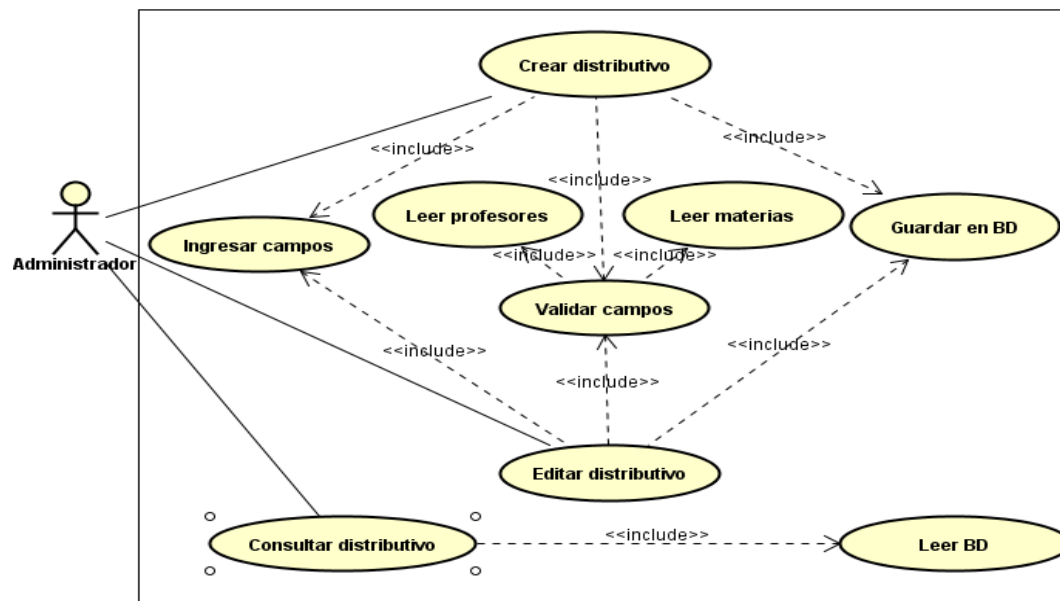
TABLA: 12 Caso de uso gestionar materias.



<b>Caso de uso:</b>	Gestionar materias
<b>Identificador:</b>	CU06
<b>Actores:</b>	Usuario Administrador
<b>Propósito:</b>	<p><b>Crear materias :</b> Permite agregar nuevas materias en el sistema</p> <p><b>Editar materias :</b> Modificar las materias en el sistema</p> <p><b>Consultar materias :</b> Permite obtener un listado de los docentes en el sistema</p>
<b>Resumen:</b>	Este formulario permite una gestión de las materias en el sistema
<b>Pre-Requisitos:</b>	El usuario administrador debe estar registrado en el sistema.

FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES

TABLA: 13 Caso de uso manejo del distributivo



**Caso de uso:** Manejo del distributivo

**Identificador:** CU07

**Actores:** Usuario Administrador

**Propósito:** **Crear distributivo:** Los distributivos son creados por periodo

**Editar distributivo:** Se podrá modificar el distributivo creado en el periodo seleccionado

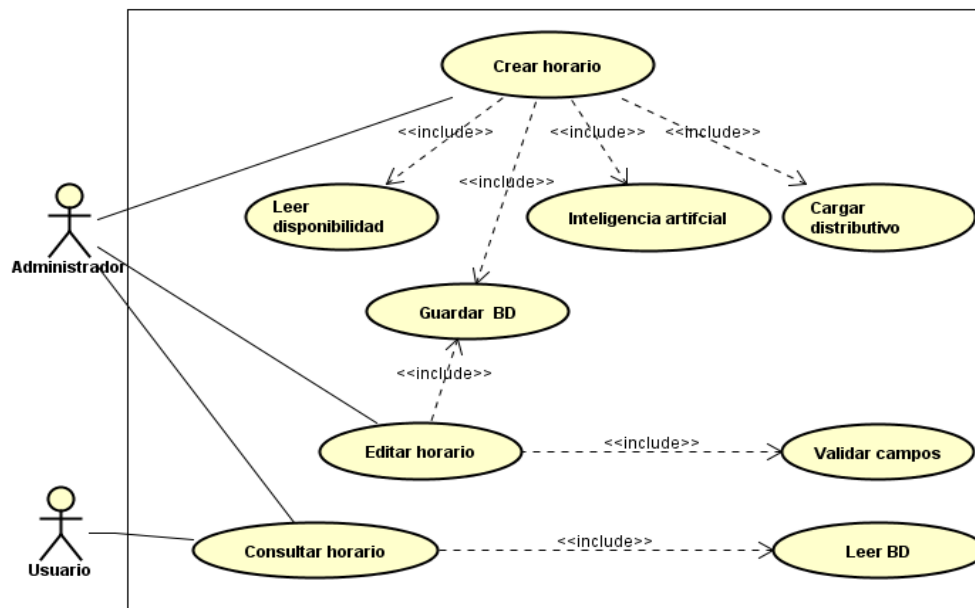
**Consultar distributivo:** Se podrá visualizar el distributivo del periodo seleccionado

**Resumen:** Este formulario permite construir el distributivo en el sistema con las clases definidas en docentes-materias

**Pre-Requisitos:** El usuario administrador debe estar registrado en el sistema.

**FUENTE:** CONOCIMIENTO PROPIO  
**ELABORADO:** LOS AUTORES

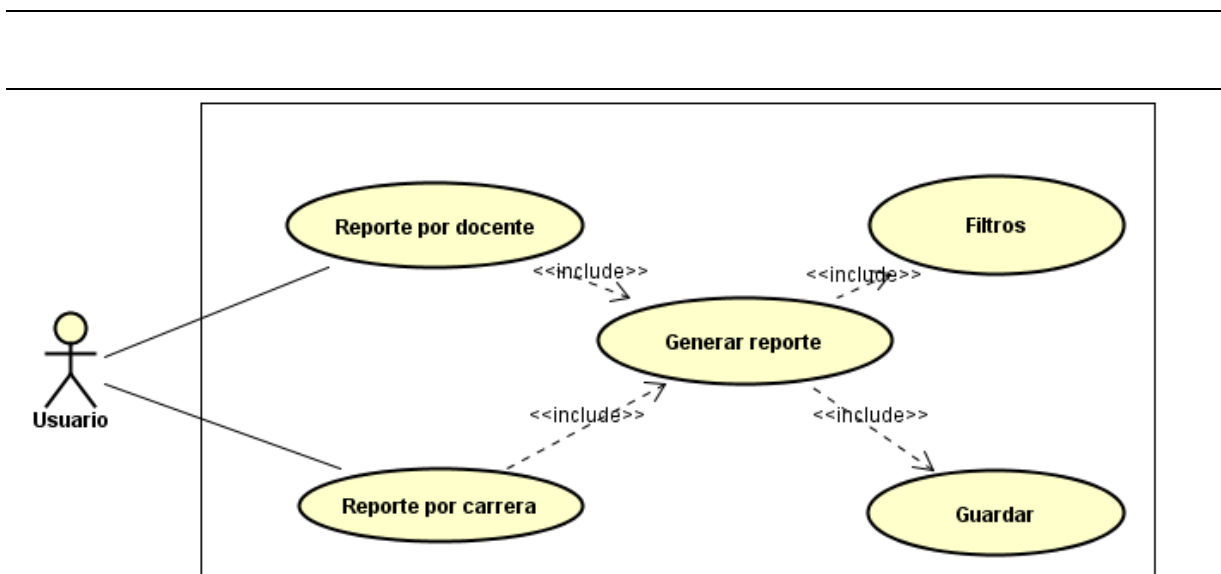
TABLA: 14 Caso de uso generación de horario.



<b>Caso de uso:</b>	Generación de horario
<b>Identificador:</b>	CU08
<b>Actores:</b>	Usuario Administrador Usuarios
<b>Propósito:</b>	<p><b>Crear horario:</b> El horario será creado por distributivo</p> <p><b>Edita horario:</b> El horario creado con la herramienta de inteligencia artificial podrá ser reconstruido de manera manual por el administrador</p> <p><b>Consultar horario:</b> Se podrá visualizar el horario por todos los usuarios</p> <p><b>Inteligencia artificial:</b> Se implementa el algoritmo de inteligencia artificial ACO</p> <p><b>Cargar distributivo:</b> Se debe seleccionar el distributivo previamente creado en el caso de uso CU07</p>
<b>Resumen:</b>	Este formulario permite la creación y reconstrucción del horario académico
<b>Pre-Requisitos:</b>	Los usuarios deben estar registrado en el sistema.

FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES

TABLA: 15 Caso de uso manejo de reportes.



<b>Caso de uso:</b>	Manejo de reportes
<b>Identificador:</b>	CU09
<b>Actores:</b>	<b>Usuario:</b> Usuario administrador y usuarios
<b>Propósito:</b>	<p><b>Reporte por docente:</b> El usuario podrá generar reportes del horario por docentes o grupo de docentes</p> <p><b>Reportes por carrera:</b> El usuario podrá generar reportes del horario desacuerdo con la carrera seleccionada</p> <p><b>Guardar:</b> Los reportes se guardan en el equipo en formato pdf</p>
<b>Resumen:</b>	Este formulario permite crear reportes
<b>Pre-Requisitos:</b>	Los usuarios deben estar registrado en el sistema.

FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES

## 4.5. Modelado del sistema.

De acuerdo al planteamiento del problema y a los requerimientos del sistema la aplicación se dividió en cuatro módulos los cuales representan la funcionalidad del sistema:

- Módulo de administración
- Módulo de carga de dato
- Módulo de generación de horarios
- Módulo de reconstrucción de horario

### 4.5.1. Módulo de administración.

En esta sección permite al usuario administrador gestionar el registro de la información referente al periodo, facultades, carreras, paralelos y la asignación de docentes con su respectiva materia que se tenga distribuida en la universidad.

A continuación, se muestran las tablas de manera jerárquica con sus respectivos detalles los cuales son ingresados de manera manual los cuales son:

- Descripción: Representa una descripción del tipo de información que incluye
- Dato: En esta se muestra un ejemplo de cómo se debe describir la información

Comenzando por la tabla periodo donde se describe el año, las fechas de inicio y fin del ciclo por el cual cursa la universidad, estado y un identificador automático numérico.

*TABLA: 16 Estructura de datos de la tabla periodo.*

Dato	Descripción
2016	Es el año que actual o periodo
02/05/2016	Se detalla la fecha de inicio del ciclo
30/03/2017	Se detalla la fecha de fin del ciclo
1	Se indica el ciclo que pertenece la fecha ejemplo: clico 1
A=Activo y P=Pasivo	Es la información que se usa y se los clasifica en A Y P, A=activo es válido para este periodo, P=pasivo indica que no se debe mostrar los datos o no se utilizan para realizar más procesos.

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**



La tabla facultad consiste en ingresar los siguientes datos: Nombre de la facultad, estado de la facultad y un identificador automático numérico.

*TABLA: 17 Tipos de datos de la tabla facultad.*

Dato	Detalla
FCI	Se ingresa el nombre de la facultad ejemplo: FCI, FCA, etc.
A=Activo y P=Pasivo	Es la información que se usa y se los clasifica en A Y P, A=activo es válido para este periodo, P=pasivo indica que no se debe mostrar los datos o no se utilizan para realizar más procesos.
<b>FUENTE: CONOCIMIENTO PROPIO</b>	
<b>ELABORADO: LOS AUTORES</b>	

La tabla carrera consiste en ingresar los siguientes datos: identificador de la carrera, Nombre de la carrera y estado de la carrera.

*TABLA: 18 Tipos de datos de la tabla carrera.*

Dato	Detalla
1,2,3,.....,n a,b,c,.....,zyz a1,b2,c3,..,xyz1	Es un identificador único que se requiere puede ser numérico, alfabético o alfanumérico.
Sistemas	Se ingresa el nombre de la carrera ejemplo: sistemas, eléctrica, etc.
A=Activo y P=Pasivo	Es la información que se usa y se los clasifica en A Y P, A=activo es válido para este periodo, P=pasivo indica que no se debe mostrar los datos o no se utilizan para realizar más procesos.
<b>FUENTE: CONOCIMIENTO PROPIO</b>	
<b>ELABORADO: LOS AUTORES</b>	

La tabla paralelo consiste en ingresar los siguientes datos: Estado del paralelo puede ser Activo o pasivo, nombre. Este último debe estar descrito con el semestre, nombre de la carrera y un identificador automático numérico.

*TABLA: 19 Estructura de datos de la tabla paralelo.*

Dato	Detalle
Primero A sistemas	Se ingresa el nombre del paralelo primero, segundo, tercero,
Primero B sistemas	etc, seguido de la carrera que pertenece. Ejemplo: primero
Segundo sistemas	sistemas, quinto electricidad, etc.
<b>FUENTE: CONOCIMIENTO PROPIO</b>	
<b>ELABORADO: LOS AUTORES</b>	

La tabla asignación docente clase forma con la siguiente información: el identificador del docente, el identificador de la materia, el identificador del paralelo, el identificador del periodo y un identificador automático numérico, este proceso se lo realiza mediante un formulario como se lo visualiza en el anexo de la **Figura: 15**.

*TABLA: 20 Formato de asignación a docentes con sus respectivos paralelos y materias.*

Dato	Detalle
1,2,...n	Se selecciona el periodo, el cual tiene asignado un identificador ejemplo: periodo 2016 ciclo 1 tiene el identificador 1, periodo 2016 ciclo 2 tiene el identificador 2.
1,2,3,...n	Se selecciona el paralelo, el cual tiene asignado un identificador: ejemplo: primero A sistemas tiene el identificador 1, segundo B sistemas tiene el identificador 2.
1,2,3,...n	Se selecciona la materia, el cual tiene asignado un identificador ejemplo: lógica matemática tiene el identificador 3, Ingles básico tiene el identificador 51.
1,2,3,...n	Se selecciona el docente, el cual tiene asignado un identificador ejemplo: Alarcón López Henry Felipe tiene el identificador 1, Jaramillo Chuqui Iván tiene el identificador 16.
A=Activo P=Pasivo	y Es la información que se usa y se los clasifica en A Y P, A=activo es válido para este periodo, P=pasivo indica que no se debe mostrar los datos o no se utilizan para realizar más procesos.
<b>FUENTE: CONOCIMIENTO PROPIO</b>	
<b>ELABORADO: LOS AUTORES</b>	

### 4.5.2. Módulo de carga de dato.

En esta sección permite al usuario administrador cargar la información referente a los docentes y Malla curricular (materias) que se tenga en la universidad. Dicha información se la puede cargar en un formato archivo formato .xls

Esta tabla está compuesta por 4 clasificaciones de docentes en las cuales se debe asignar el peso, dependiendo el cargo que ocupen los docentes, valorar con pesos entre 0 y 1, que sumen un total de 1, también tiene un identificador que va del 1 al 4 y su respectivo estado. Esta información debe ser ingresada antes de cargar los datos de los docentes.

*TABLA: 21 Tipo de docente según su carga y asignación del peso.*

Identificador	Tipo docente	Peso	Estado
1	Decano	0,4	A
2	Coordinador carrera	0,3	A
3	Gestión de área	0,2	A
4	Profesor	0,1	A

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

A continuación, se muestra un ejemplo de cómo cargar la información de los docentes en formato .xls. En la primera fila de la imagen se describe la información que se debe ingresar compuesta por (Cédula, Tipo docente, nombre, estado) el tipo docente se describe en la **Tabla 21**.

*FIGURA: 1 Formato .xls para datos de docentes.*

	A	B	C	D	E	F
1	cedula	tipo docente	nombre	estado		
2		4	ALARCON LOPEZ HENRY FELIPE	A		
3		4	ALCOCER QUINTEROS PATRICIO	A		
4		4	ALMEIDA ESPINEL PABLO	A		
5		4	BAJAÑA MENDIETA INES DE LA LUZ	A		
6		4	BAQUE MITE LEONARDO	A		
7		4	BARRENO OJEDA JUAN	A		
8		2	BARZOLA MIRANDA SONIA	A		
9		4	CABRERA VARGAS SEGUNDO	A		
10		4	CANCHIGNIA MARTINEZ HAYRON FABRICIO	A		
11		4	CEVALLOS MUÑOZ OMAR	A		
12		3	CHIRIBOGA CASANOVA WASHINGTON	A		
13		4	COELLO LEÓN CLEMENCIA	A		
14		3	DÍAZ MACÍAS EFRAIN	A		
15		4	FONFAY FLOR MARINA	A		
16		3	GUERRERO ULLOA GLEISTON	A		

**FUENTE: CONOCIMIENTO PROPIO**

**ELABORADO: LOS AUTORES**

Para la materia tiene que estar especificado con su respectivo tipo si es Básica, General, Investigación, Profesional, Proyecto integrador y Optativa con su color, código color, peso y estado. Esta información tiene que ser ingresada con anterioridad antes de cargar los datos de las materias.

*TABLA: 22 Tipo de datos para la tabla tipo materia.*

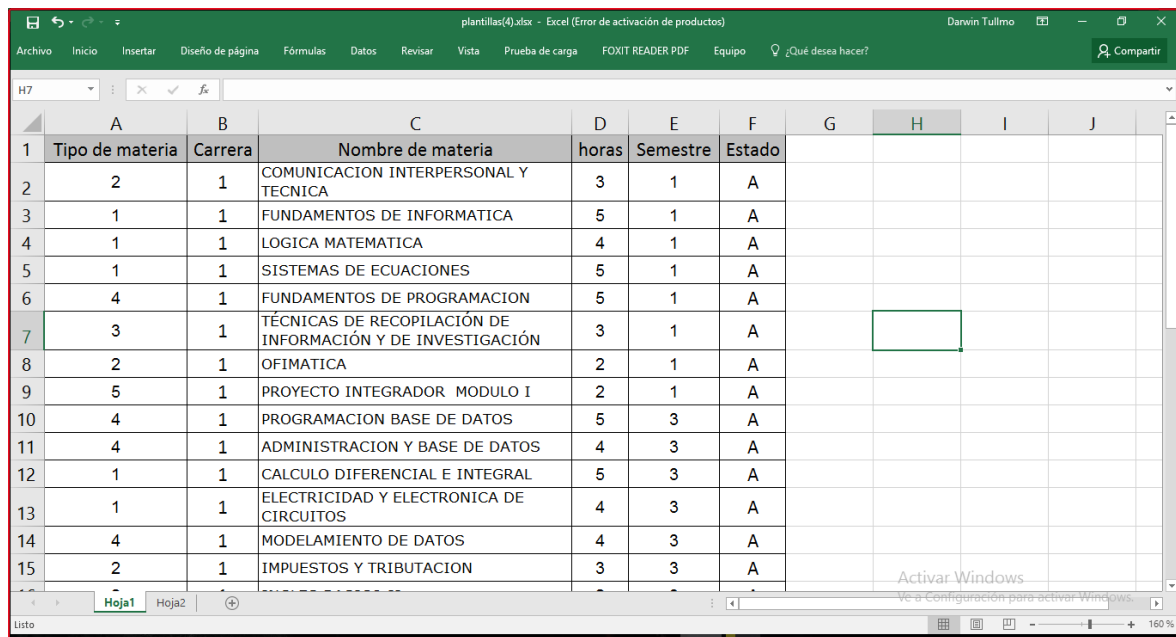
Número	Tipo materia	Color	Código color	Peso	Estado
1	Básica	greenyellow	#ADFF2F	0,12	A
2	General	yellow	#FFFF00	0,12	A
3	Investigación	orange	#FFA500	0,12	A
4	Profesional	lightblue	#ADD8E6	0,40	A
5	Proyecto integrador	lightgray	#D3D3D3	0,12	A
6	Optativa	chartreuse	#7FFF00	0,12	A

**FUENTE: CONOCIMIENTO PROPIO**

**ELABORADO: LOS AUTORES**

A continuación, se muestra un ejemplo de cómo cargar la información de las materias en formato .xls. En la primera fila de la imagen se describe la información que se debe ingresar compuesta por (Tipo materia, identificador de la carrera, nombre de la materia, horas, semestre, estado) el tipo de materia se describe en la **Tabla 22**.

*FIGURA: 2 Formato .xls para datos malla.*



	A	B	C	D	E	F	G	H	I	J
1	Tipo de materia	Carrera	Nombre de materia	horas	Semestre	Estado				
2	2	1	COMUNICACION INTERPERSONAL Y TECNICA	3	1	A				
3	1	1	FUNDAMENTOS DE INFORMATICA	5	1	A				
4	1	1	LOGICA MATEMATICA	4	1	A				
5	1	1	SISTEMAS DE ECUACIONES	5	1	A				
6	4	1	FUNDAMENTOS DE PROGRAMACION	5	1	A				
7	3	1	TÉCNICAS DE RECOPIACIÓN DE INFORMACIÓN Y DE INVESTIGACIÓN	3	1	A				
8	2	1	OFIMATICA	2	1	A				
9	5	1	PROYECTO INTEGRADOR MODULO I	2	1	A				
10	4	1	PROGRAMACION BASE DE DATOS	5	3	A				
11	4	1	ADMINISTRACION Y BASE DE DATOS	4	3	A				
12	1	1	CALCULO DIFERENCIAL E INTEGRAL	5	3	A				
13	1	1	ELECTRICIDAD Y ELECTRONICA DE CIRCUITOS	4	3	A				
14	4	1	MODELAMIENTO DE DATOS	4	3	A				
15	2	1	IMPUESTOS Y TRIBUTACION	3	3	A				

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

### 4.5.3. Módulo generación de horarios.

Este módulo es el encargado de aplicar un modelo metaheurística basado en colonia de hormigas para generar diferentes horarios. Valorando que el problema de generación de horarios este definido en la literatura como un posible cumplimiento de restricciones o limitaciones, este problema se define como un problema de satisfacción de restricciones.

Para la solución de este problema se aplicó un algoritmo de metaheurística ACO. En este escenario se definieron algunos elementos fundamentales de la metaheurística como son:

#### 4.5.3.1. Grafo representativo del problema.

Nosotros utilizamos una representación en forma de grafo donde cada nodo representa un profesor(docente) y los enlaces entre ellos son la continuidad de que a partir un profesor se puede asignar a otro profesor.

Con un ejemplo de 5 docente (profesores), D1 docente 1, D2 docente 2, D3 docente 3, D4 docente 4, D5 docente 5, se realiza la distribución de horas para cada docente de las diferentes materias que impantan.

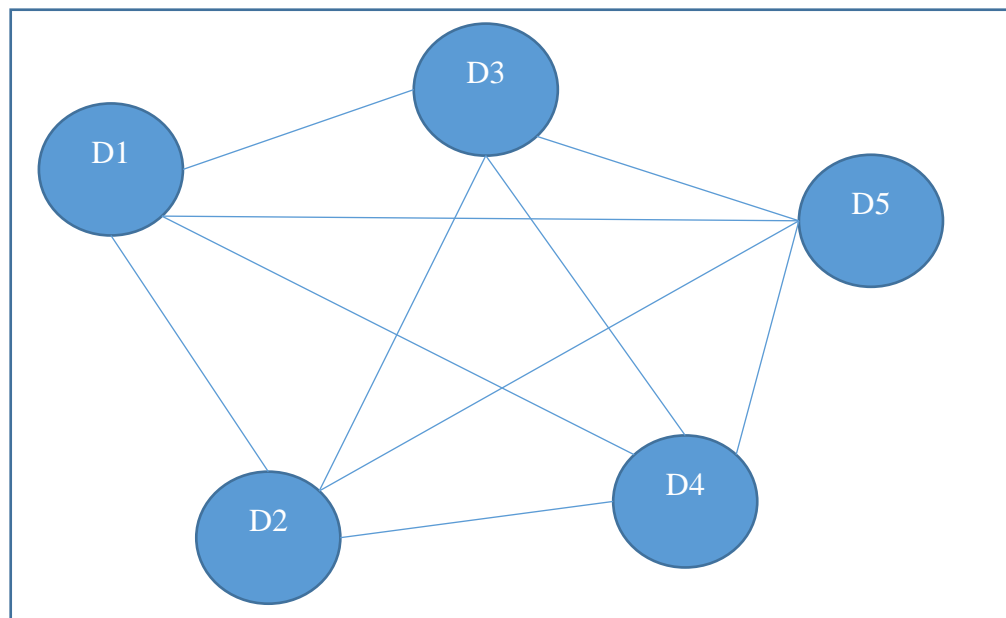
Sea  $G$  un grafo tal que  $G$  es igual al número de Vértices o nodos( $V$ ) que representan al docente asignado, en donde las aristas( $E$ ) representan los rastros de feromona entre sus nodos( $V$ ).

$V:=\{D1,D2,D3,D4,D5\}$

$E:=\{\{D1,D2\},\{D1,D3\},\{D1,D4\},\{D1,D5\}, \{D2,D3\},\{D2,D4\},\{D2,D5\},$   
 $\{D3,D4\},\{D3,D5\}, \{D4,D5\}\}$

El grafo  $G=(V, E)$  donde están los arcos y los nodos representan la forma para presentar el problema. Por ejemplo, presenta el problema con 5 docentes el grafo en la FIGURA: 3.

*FIGURA: 3 Grafo de asignación por docente.*



**FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES**

#### **4.5.3.2. Función heurística.**

Para este problema se definió dos funciones heurísticas las cuales determinan la probabilidad de elegir a un docente y posteriormente asignar en el horario cumpliendo con las restricciones establecidas, aplicando la función de probabilidad a la hora de distribuir el horario:

##### **1. Seleccionar el profesor dependiendo tipo de cargo.**

El **decano** tiene un mayor peso ya que requiere ocupar las primeras horas en el horario al igual que Coordinador carrera tiene un peso menor que el decano, pero también puede ocupar las primeras horas del horario, por lo que al aplicar las reglas de las restricciones

establecidas estos docentes con cargo de gestión son seleccionados, al contrario de los que realizan la gestión de área y solo profesor tienen menos probabilidad de ser seleccionados al principio.

*TABLA: 23 Tipos de clasificaciones para docentes.*

<b>Tipo de cargo</b>	<b>Peso</b>
Decano	0,4
Coordinador carrera	0,3
Gestión de área	0,2
Profesor	0,1

**FUENTE: CONOCIMIENTO PROPIO**

**ELABORADO: LOS AUTORES**

Luego de seleccionar el docente tomando en cuenta las restricciones se procede al siguiente paso que se detalla a continuación.

## **2. Asignación de horas a horario.**

Para la asignación de horas al horario se toma en cuenta las restricciones del tipo de materia ya que se les da una prioridad a las materias profesionales, también depende del docente que se seleccionó si tiene o no cargo de gestión, para tener una posibilidad más alta de quedarse con las primeras horas del horario siempre y cuando cumpla con las restricciones.

Los tipos de materias y sus respectivos pesos se presentan en la **Tabla 23**.

Se plantearon dos funciones para la selección de horas, la función heurística y la función probabilística.

### **Heurística.**

La heurística es aplicable a cualquier ciencia e incluye la elaboración de medios auxiliares, principios, reglas, estrategias y programas que faciliten la búsqueda de vías de solución a problemas; o sea, para resolver tareas de cualquier tipo para las que no se cuenta con un procedimiento algorítmico de solución.<sup>4</sup>

Se asigna las horas de forma probabilística, teniendo en cuenta que tendrán mayor probabilidad las primeras horas los docentes con cargo de gestión, a continuación, se muestra un ejemplo con 5 docentes, los cuales fueron seleccionados en el siguiente orden { PROF\_1, PROF\_2, PROF\_3, PROF\_4, PROF\_5}.

<sup>4</sup> <http://www.scepuras.org/>

*TABLA: 24 Asignación de hora a docente de manera Probabilístico.*

HORA	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES
1	PROF_1	PROF_2	PROF_3	PROF_1	PROF_4
2	PROF_1	PROF_2	PROF_3	PROF_1	PROF_4
3	PROF_4	PROF_5	PROF_4	PROF_5	
4	PROF_4	PROF_5	PROF_4	PROF_5	
5				PROF_5	
6					

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

Como se puede observar en la tabla los docentes con cargo de gestión son asignados las primeras horas de preferencia.

### **Aleatoria.**

En definitiva, permite ofrecer una descripción de la probabilidad de que se adoptan ciertos valores. No se sabe de manera precisa qué valor adoptará la variable cuando sea determinada o medida, pero sí se puede conocer cómo se distribuyen las probabilidades vinculadas a los valores posibles.

Se asigna de forma aleatoria llenando las primeras horas en el horario, teniendo en cuenta que tendrán las mismas posibilidades todos los docentes a obtener las primeras horas.

*TABLA: 25 Asignación de hora a docente de manera Aleatoria.*

HORA	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES
1	PROF_4	PROF_2	PROF_1	PROF_4	PROF_4
2	PROF_4	PROF_2	PROF_1	PROF_4	PROF_4
3	PROF_1	PROF_4	PROF_2	PROF_3	
4	PROF_1	PROF_4	PROF_2	PROF_3	
5		PROF_3			
6		PROF_3			

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

Se realizaron pruebas de con la función aleatoria, los resultados fueron los deseados ya que distribuía las horas más equitativamente al asignar a los docentes.

Se usa las dos formas para la asignación de las horas en el horario tanto la asignación heurística para que los docentes con cargo de gestión estén en las primeras horas y la aleatoria para que todos los docentes puedan también ser asignados a las primeras horas.

En la cual el que más probabilidad de quedar en las primeras horas será la materia con mayor peso, y las otras materias pueden ocupar cualquier posición en el horario dependiendo a qué tipo de docente se le haya asignado la materia, ya que eso influenciara para la toma de decisión en el algoritmo.



#### 4.5.3.3. Rastros de feromona.

Para representar este elemento se utilizó la matriz cuadrada de  $M \times M$  donde  $M$  es la cantidad de docentes presentes en el distributivo

#### 4.5.3.4. Valores de guía de la metaheurística.

Los siguientes valores son los parámetros que guían a los algoritmos en el proceso de hallar posibles soluciones a la hora de generar horarios, entregando posibles horarios a utilizar.

*TABLA: 26 Parámetros guías para la meta-heurística.*

Parámetro	Símbolo	Valor	Descripción
Número interacciones		500	Es el número de veces que repite el proceso de generación de horario.
Número hormigas		30	Representa cada horario por hormiga.
Índice de evaporación	$\tau$	0,1	Por cada iteración la matriz feromona se actualiza
Alfa	$\alpha$	1	Nivel de importancia de la información heurística
Beta	$\beta$	1	Nivel de importancia de la información memorística
Índice mínimo feromona		0,8	Valor por defecto para el parámetro de feromona

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

A continuación, un ejemplo para resolver un problema de asignación de horarios utilizando la metaheurística colonia de hormigas para lo cual se describen dos ecuaciones importantes las cuales son:

- 1) Ecuación para la probabilidad de seleccionar un camino.

$$p_{xy}^k = \frac{(\eta_{xy})^\alpha (\tau_{xy})^\beta}{\sum (\eta_{xy})^\alpha (\tau_{xy})^\beta}$$

Donde la  $k$  hormiga se mueve de un estado  $x$  a un estado  $y$  con probabilidad. Cada arco del grafo contiene 2 tipos de importación.

- $\tau_{xy}$  representa la cantidad de feromona depositada entre  $x$  y  $y$  es decir, contiene un valor de las rutas de un docente  $x$  a un docente  $y$  la cual aumenta o disminuye dependiendo de la calidad de la solución la cual es evaluada por la ecuación.
- $\eta_{xy}$  Esta información representa materia + tipo docente la cual no es modificada en la ejecución del algoritmo

Los valores de  $\alpha$  y  $\beta$  representan la deseabilidad de la información la cual toma valor de 1 para este ejemplo.

## 2) Ecuación para la actualización de feromona

$$\tau_{xy} = (1 - \rho)\tau_{xy} + \sum_k \Delta \tau_{xy}^k$$

$\tau_{xy}$  Es cantidad de feromona de las rutas de un docente  $x$  a un docente  $y$ . Este proceso permite actualizar la feromona y aportar la mejor solución donde  $\Delta \tau_{xy}^k$  es la cantidad de feromonas depositadas por la  $K$  hormiga

A continuación, en la Tabla # 27 se presentan los datos con las que se trabaja en este ejemplo los cuales constan de 2 docentes con cargo de gestión y 3 docentes a tiempo completo, 24 materias con sus respectivos tipos de materia.

*TABLA: 27 Datos representativos para la distribución del horario.*

Docente	Cantidad de materia	Tipo de materia	Tipo docente
D1	1	Básica	Decano
	2	Profesional	
D2	2	Investigación	Docente
	1	General	
	2	Básica	
D3	3	General	Coordinador
	2	Profesional	
	2	Básica	
D4	2	General	Docente
	1	Profesional	
	3	Básica	

D5	2	Investigación	Docente
	1	General	

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

Ahora para poder emplear el algoritmo de colonia de hormigas es necesario representar el problema como un grafo lo cual está representado en la Figura # 3.

Cada nodo (docente) contiene la información para  $\eta_{xy}$  que está unido por enlaces llamados arcos los cuales contiene la información de feromona para  $\tau_{xy}$ . A continuación, se muestra una Tabla # 28 una matriz cuadrada MxM la cual contiene la información de feromona con un valor inicial por defecto de 0,8.

*TABLA: 28 Matriz de feromona inicial en el proceso de distribución del horario.*

Docentes	D1	D2	D3	D4	D5
D1	0,8	0,8	0,8	0,8	0,8
D2	0,8	0,8	0,8	0,8	0,8
D3	0,8	0,8	0,8	0,8	0,8
D4	0,8	0,8	0,8	0,8	0,8
D5	0,8	0,8	0,8	0,8	0,8

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

Ahora que se conoce la información de cada docente (nodo) se procede a calcular el valor para  $\eta_{xy}$ . Se comienza calculando el valor de materia, como se muestra en la Tabla # 29,  $D1 = 1 \cdot 0,1 + 2 \cdot 0,5$  donde el valor de 1 y 2 representa la cantidad de materias multiplicando por 0,1 y 0,5 que representa el tipo de materia. Siguiendo con el mismo ejemplo de D1 ahora que tenemos el cálculo de materia que como resultado dio 1,1 se puede obtener el valor final para  $\eta$  que es sumándole el tipo de docente.

*TABLA: 29 Cálculo del valor de materia para obtener el valor de los caminos del grafo.*

	Calculo de materias	Materia + tipo docente (longitud)
D1	$1 \cdot 0,1 + 2 \cdot 0,5 = 1,1$	$1,1 + 0,4 = 1,5$
D2	$2 \cdot 0,1 + 1 \cdot 0,1 + 2 \cdot 0,1 = 0,5$	$0,5 + 0,1 = 0,6$
D3	$3 \cdot 0,1 + 2 \cdot 0,5 = 1,3$	$1,3 + 0,3 = 1,6$
D4	$2 \cdot 0,1 + 2 \cdot 0,1 + 1 \cdot 0,5 = 0,9$	$0,9 + 0,1 = 1$

D5	$3*0,1 + 2*0,1 + 1*0,1 = 0,6$	$0,6 + 0,1 = 0,7$
----	-------------------------------	-------------------

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

A continuación, en la Tabla # 30 donde se representa los posibles caminos del grafo con su respectiva longitud y cantidad de feromona.

*TABLA: 30 Representación en valores los posibles caminos del grafo.*

<b>Caminos</b>	<b>Longitud(<math>\eta</math>)</b>	<b>T</b>
D1-D2	0,6	0,8
D1-D3	1,3	0,8
D1-D4	1	0,8
D1-D5	0,7	0,8
D2-D1	1,5	0,8
D2-D3	1,3	0,8
D2-D4	1	0,8
D2-D5	0,7	0,8
D3-D1	1,5	0,8
D3-D2	0,6	0,8
D3-D4	1	0,8
D3-D5	0,7	0,8
D4-D1	1,5	0,8
D4-D2	0,6	0,8
D4-D3	1,3	0,8
D4-D5	0,7	0,8
D5-D1	1,5	0,8
D5-D2	0,6	0,8
D5-D3	1,3	0,8
D5-D4	1	0,8

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

Tomando como nodo inicial al D1 que representa al docente 1 tenemos 4 posibles caminos los cuales son hacia los docentes D2, D3, D4, D5. Para seleccionar al siguiente nodo se realizan las operaciones respectivas. Para ello en la siguiente tabla se muestra los cálculos para hallar los valores de la ecuación.

Una vez encontrados los valores de  $\eta$  y T que consiste en sumar los valores antes mencionados. Se debe realizar la sumatoria de todos los posibles caminos  $\sum (\eta_{xy})^\alpha (\tau_{xy})^\beta$  y poder encontrar la probabilidad de ir desde un docente x a un docente y, tal como se presenta en la Tabla # 31, en la ruta D1-D2 su  $P_{xy}$  es igual a  $0,06/0,36 = 0,16666667$ .

*TABLA: 31 Probabilidad de ir desde un docente x a un docente y.*

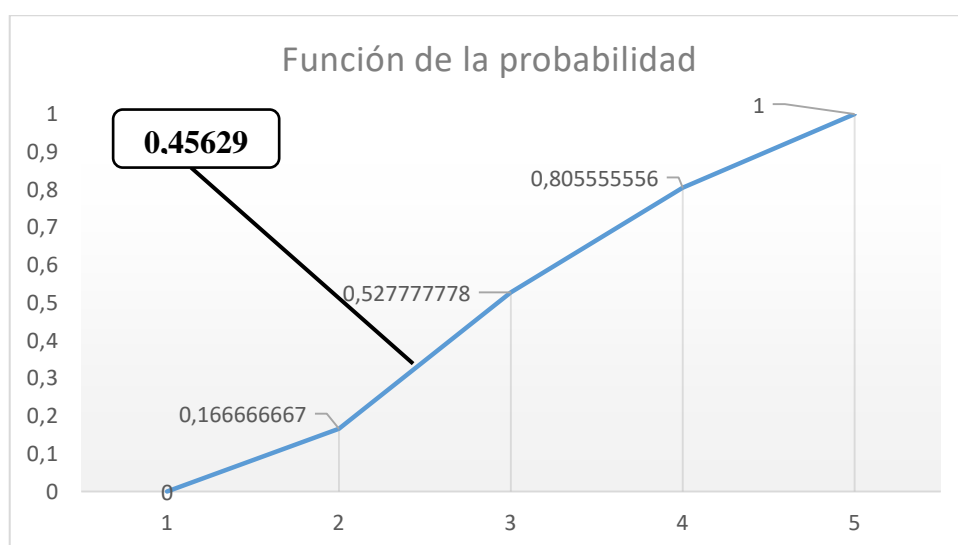
	$\eta(e) \cdot T(e)$	$P_{xy}$
D1-D2	0,06	0,16666667
D1-D3	0,13	0,36111111
D1-D4	0,1	0,27777778
D1-D5	0,07	0,19444444
$\Sigma$	0,36	1

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

Ahora le toca seleccionar a la hormiga cual camino debe tomar, para ello se procede a genera un valor aleatorio entre 0 y 1 utilizando la función `randon()` el cual es 0,45629.

Los valores en  $P_{xy}$  están ponderados de tal manera que la sumatoria es igual a 1. En la Figura: 4 se muestra de manera gráfica de estos valores. Tomando como ejemplo un caso cotidiano, con 4 arcos de baloncesto de diferentes tamaños, hay mayor probabilidad de encestar en el aro más grande. De la misma manera el docente que tenga mayor peso tendrá más posibilidad de ser seleccionado ya que tiene un conjunto mayor de números en el cual puede caer el número aleatorio.

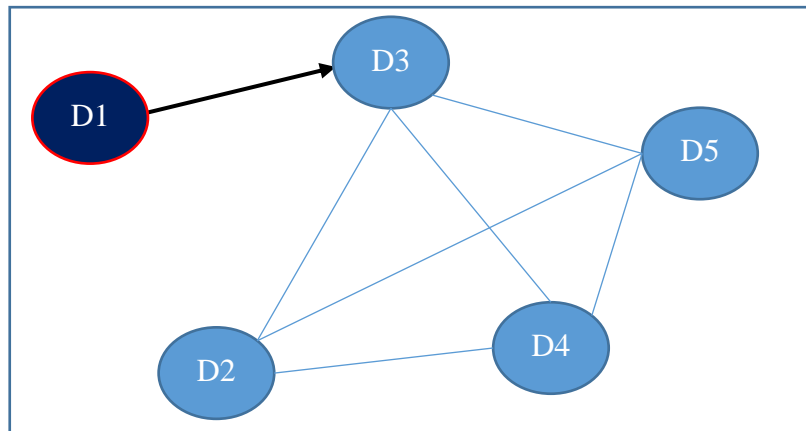
*FIGURA: 4 Valores de probabilidad de selección de un docente.*



**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

En la Figura: 5 del grafo para esta primera ruta se representa de la siguiente manera.

*FIGURA: 5 Primera ruta seleccionada para encontrar una solución.*



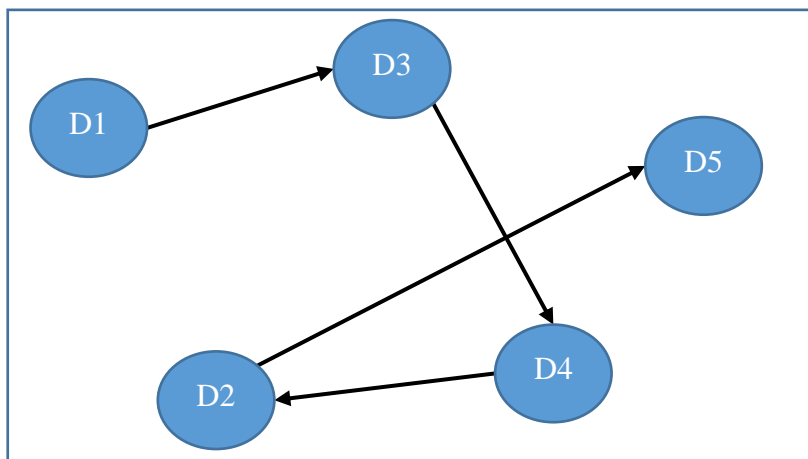
**FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES**

De esta manera algoritmo crear una ruta de D1 -> D3. Este mismo proceso se repite para los siguientes docentes hasta completar la ruta que es igual a encontrar una solución.

Después de haber realizado el proceso que lleva aplicar la ecuación de probabilidad ahora toca evaluar esa ruta o solución, para ello se tomara la muestra de 2 hormigas con su respectiva representación gráfica y la asignación para 1 paralelo la cual se describe a continuación.

Proponiendo que la hormiga 1 como se la representa en la Figura: 6 tiene la siguiente solución gráfica y de asignación.

*FIGURA: 6 Ruta completa de la hormiga 1 en la selección de docentes.*



**FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES**

D1 -> D3 -> D4 -> D2 -> D5

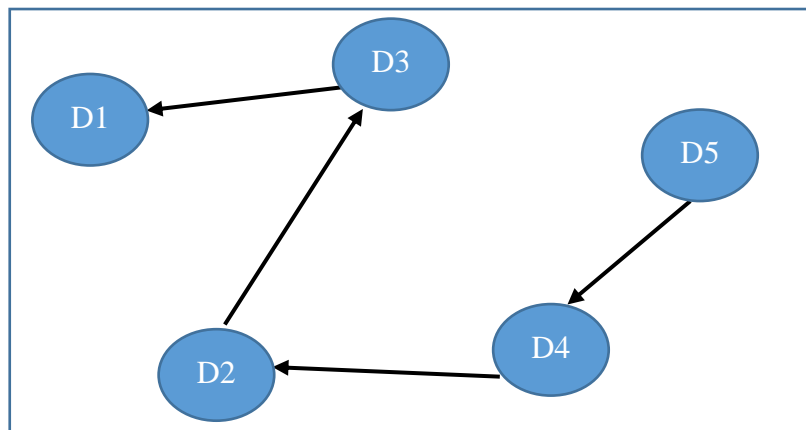
*FIGURA: 7 Docentes asignados al horario propuesto en le grafo de la hormiga 1.*

	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
7:30 - 8:30	D3	D1	D4	D1	D3
8:30 - 9:30	D3	D1	D4	D1	D3
9:30 - 10:30	D3	D4	D5	D1	D3
10:30 - 11:30	D5	D4	D5	D2	D2
11:30 - 12:30	D5			D2	D2
12:30 - 13:30				D2	

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

Proponiendo que la hormiga 2 como se la representa en la Figura: 8 tiene la siguiente solución gráfica y de asignación.

*FIGURA: 8 Ruta completa de la hormiga 2 en la selección de docentes.*



**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

D5 -> D4 -> D2 -> D3 -> D1

*TABLA: 32 Docentes asignados al horario propuesto en le grafo de la hormiga 2.*

	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
7:30 - 8:30	D5	D4	D5	D4	D2
8:30 - 9:30	D5	D4	D5	D4	D2
9:30 - 10:30	D1	D3	D2	D1	D2
10:30 - 11:30	D1	D3	D2	D1	D3
11:30 - 12:30		D3		D1	D3
12:30 - 13:30					D3

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

Para poder evaluar la solución de las 2 hormigas es necesario asignar las clases en su respectivo día y hora, este proceso de asignación consiste en ir ubicando a los docentes en las primeras horas con sus respectivas materias según como lo define el grafo y cumpliendo con las restricciones obligatorias vistas en la Tabla # 32. En este ejemplo se han asignado a todos los docentes en 1 paralelo.

La siguiente función objetivo anteriormente vista va a permitir identificar la cantidad de restricciones incumplidas  $For\ min = \sum_{i=1}^4 W_i R_i$  cuyas restricciones son las vistas en la Tabla # 4.

$$\text{Hormiga 1} = 0*0,4 + 0*0,3 + 0*0,2 + 0*0,1 = 0$$

$$\text{Hormiga 2} = 0*0,4 + 0*0,3 + 0*0,2 + 4*0,1 = 0,4$$

En la hormiga 1 no se incumplieron restricciones mientras que en la hormiga 2 se incumplió con 1 restricciones:

- Cantidad de docentes con cargo de decano (1) y coordinador (2) que están asignados fuera de las primeras horas.

Una vez identifica la hormiga que haya incumplido la menor cantidad de restricciones se aplica la ecuación 2 para esa ruta. De tal forma que la matriz se actualiza en  $0,1 * \text{restricciones cumplidas}$  y la evaporación es de 0,02 para las rutas no seleccionadas. La matriz de feromona actualizada seria la siguiente:

*TABLA: 33 Matiz de feromona llena con su respectiva evaporación y contribución.*

Docentes	D1	D2	D3	D4	D5
D1	0,78	0,78	1,2	0,78	0,78
D2	0,78	0,78	0,78	0,78	1,2
D3	0,78	0,78	0,78	1,2	0,8
D4	0,78	1,2	0,78	0,78	0,78
D5	0,78	0,78	0,78	0,78	0,78

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

Este ejemplo se desarrolló con 2 hormigas y 1 iteración, para obtener una mejor solución en el caso de que aún no se cumplan con todas las restricciones o la mayor cantidad de restricciones se debe realizar nuevas iteraciones con nuevas hormigas en base a esta información actualizada:



*TABLA: 34 Representación en valores los posibles caminos del grafo de la hormiga 2 después de la primera iteración.*

<b>Caminos</b>	<b>Longitud(<math>\eta</math>)</b>	<b>T</b>
D1-D2	0,6	0,78
D1-D3	1,3	1,2
D1-D4	1	0,78
D1-D5	0,7	0,78
D2-D1	1,5	0,78
D2-D3	1,3	0,78
D2-D4	1	0,78
D2-D5	0,7	1,2
D3-D1	1,5	0,78
D3-D2	0,6	0,78
D3-D4	1	1,2
D3-D5	0,7	0,78
D4-D1	1,5	0,78
D4-D2	0,6	1,2
D4-D3	1,3	0,78
D4-D5	0,7	0,78
D5-D1	1,5	0,78
D5-D2	0,6	0,78
D5-D3	1,3	0,78
D5-D4	1	0,78

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

#### **4.5.3.5. Codificación para resolver las restricciones.**

Tomando como referencia un enfoque basado en programación lógica, podemos emplear la siguiente codificación para resolver las restricciones duras y optimizar las restricciones blandas:

d = Días ( d1, d2, d3, d4, d5)

h = Horas (h1, h2, h3,...hNmax)

m = Materias(m1,m2,m3,...mN)

p = Profesores (p1, p2, p3,...pN)

pr = paralelos (pr1, pr2, pr3,...prN)

ar\_hp = Horario de profesores (Arreglo Multidimensional)

ar\_hcr = Horario de paralelos (Arreglo Multidimensional)

#### 4.5.3.6. Conjunto de datos.

A continuación, vamos a definir el conjunto de datos que serán representados en forma de vector para la solución del problema.

##### **Profesor.**

Quien imparte conocimientos a los alumnos sobre un determinado número de materias. Donde cada profesor  $pr_i \in \{pr_1, pr_2, pr_2, \dots \dots pr_n\}$  está asociado a las materias con un nivel de preferencia y ciertas restricciones. Donde  $n$  representa al número total de profesores.

##### **Materia.**

Representa a cada una de las materias vistas de acuerdo a la malla de cada carrera. Está dividida por semestres según el periodo que el estudiante está cursando.

Donde cada materia  $m_i \in \{m_1, m_2, m_2, \dots \dots m_n\}$  esta asignada a ciertos paralelos las cuales tienes sus respectivas horas, es decir, haciendo referencia a sus **horas/clases**.

Podemos determinas 2 tipos de materias: Optativas y obligatorias. Donde la materia obligatoria es la que debe ser cumplida para finalizar la carrera mientras que la materia opcional no, pero es necesario completar con 3 de ellas. Entonces de esta manera definimos que  $n$  es el número total de materias.

##### **Paralelo.**

Representa a un determinado número paralelos que se presentan en el semestre  $p_j \in \{p_1, p_2, p_2, \dots \dots p_n\}$  así mismo se conoce que está asociado a un determinado número de materias y dirigido por profesores ( $p$ ). Donde se debe establecer un número mínimo y máximo de alumnos para que sea funcional. Entonces de esta manera definimos que  $n$  es el número total de paralelos.

##### **Rango de horario.**

Un dato importante en la elaboración de horarios ya que representara el intervalo entre el valor máximo y el valor mínimo; por ello, se representa por una hora cada rango.

### **Características de la solución.**

Para emplear los algoritmos hay que tener en cuenta que una función  $I_i(s)$  indica si la característica  $i$  está presente en la solución  $s$ :

$$I_i(s) \begin{cases} 1: i \text{ presente en la solución } s \\ 0: i \text{ no está presente en la solución } s \end{cases}$$

### **Paralelo-Materia.**

$$pm_{ij} \begin{cases} 1: \text{si el paralelo } i \text{ está en una materia } j \\ 0: \text{caso contrario} \end{cases}$$

### **Profesor-Materia.**

Esta solución presenta más de 2 características ya que existe un nivel de preferencia en las materias a las cuales se puede dictar donde 1 representa un valor alto, 2 representa el segundo valor más alto, etc.

$$prm_{ij} \begin{cases} 1: \text{si el profesor } i \text{ está en una materia } j \\ 0: \text{caso contrario} \end{cases}$$

### **Periodo-profesor**

$$pepr_{ij} \begin{cases} 1: \text{si el profesor } i \text{ está en una materia } j \\ 0: \text{caso contrario} \end{cases}$$

Se debe tomar en consideración los docentes que estarán disponibles.

#### **4.5.3.7. Función objetivo.**

Para hallar la función objetivo se minimiza el número de restricciones incumplidas.

$$F = 1 - (1 / ((\text{restricciones } 1 * \text{peso}) + (\text{restricciones } 2 * \text{peso}) + (\text{restricciones } 3 * \text{peso}) + (\text{restricciones } 4 * \text{peso})))$$

#### **4.5.3.8. Selección de Horas.**

Se plantearon dos funciones para la selección de horas, la función heurística y la función probabilística.

#### **4.5.3.9. Asignación de pesos a docentes.**

Se le da un valor a cada cargo que posee un docente después de identificar los cargos de mayor relevancia.

Función a considerar en relación peso docente.

$$F(H_1) = (0,4 * \text{primero}) + (0,3 * \text{segundo}) + (0,2 * \text{tercero}) + (0,1 * \text{cuarto})$$

#### **4.5.3.10. Evaluación de la hormiga.**

Se evalúa cada horario independiente mente ya que se genera de una forma diferente e igual dependiendo del proceso que realice la función de asignación de hora a cada docente.

Se muestra el horario de dos hormigas para luego ser evaluadas con las respectivas restricciones generales, no tiene que incumplir o mínimo tener un menor número de incumplimientos.

**Función de la evaluación de la hormiga.**

$$FeH = 1 - ( 1 / ( \text{restricciones 1} * \text{peso} ) + ( \text{restricciones 2} * \text{peso} ) + ( \text{restricciones 3} * \text{peso} ) + ( \text{restricciones 4} * \text{peso} ) )$$

#### **4.5.3.11. Evaluación de horarios.**

Se considera las siguientes restricciones para la evaluación de horario.

1. Cantidad de horas por materias sin asignas en el horario y su peso = 0,4.
2. Cantidad de materia asignada al horario que exceden las 3 horas seguidas y su peso = 0,3.
3. Cantidad de docentes con diferentes materias que son asignados en el horario en un mismo paralelo y mismo día y su peso = 0,2.
4. Cantidad de docentes con cargo de decano (1) y coordinador(2) que están asignados fuera de las primeras horas y su peso = 0,1.

$$Fo \min = \sum_{i=1}^4 W_i R_i$$

##### **1.- Cantidad de horas por materias sin asignas en el horario.**

Esto conlleva a que hay docentes sin ubicar en horas de clases y el horario puede no ser el deseado.

##### **2.- Cantidad de materia asignada al horario que exceden las 3 horas seguidas.**

Un docente no debe dictar más de dos o tres horas seguidas en un mismo paralelo.

##### **3.- Cantidad de docentes con diferentes materias que son asignados en el horario en un mismo paralelo y mismo día.**

El docente no debe ser asignado más de una vez en el mismo paralelo el mismo día con el mismo curso.

#### **4.- 4. Cantidad de docentes con cargo de decano (1) y coordinador(2) que están asignados fuera de las primeras horas.**

Como anteriormente se asignó a los docentes 1 y 2, deben de permanecer lo más posible en las primeras horas caso contrario recibirá una mala puntuación por parte de la **función de evaluación**.

##### **4.1.1.1. Evaporación de la feromona y Actualización de Feromonas.**

Cuando todas las hormigas han completado una solución, los rastros son actualizados por y evaporados.

Para el proceso de evaporación, se procede a restar un 0.1 a toda la matriz de feromona y luego la aportación de  $\tau_{ij}$  donde representa el recorrido de la mejor solución.

*TABLA: 35 Matriz de feromona.*

	Docente1	Docente2	Docente3	Docente4	Docente5	DocenteN...
Docente1						
Docente2						
Docente3						
Docente4						
Docente5						
Docente6						
DocenteN...						

**FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES**

#### **Función de Evaporación y Actualización.**

$$M_F[i,j] = (1-P) * M_F[i,j]$$

##### **4.1.1.2. Condiciones de parada.**

Se utiliza dos condiciones de paradas para el algoritmo colonia de hormiga en la generación de horarios.

- El cumplimiento del número de iteraciones máximas ingresadas al algoritmo.
- Si el número de restricciones incumplidas es igual a 0.

#### 4.1.1.3. Estudio experimental.

En esta sección se presentan los resultados obtenidos por el algoritmo colonia de hormiga en las pruebas que se ejecutan al horario para conocer los parámetros óptimos para la generación de horarios.

Se realizó 4 pruebas con los datos obtenidos en el distributivo académico de la FCI para el ajuste de los parámetros del algoritmo de forma que genere una solución óptima minimizando los recursos.

Los resultados evaluados de la solución del grupo de hormigas al generar un horario. Los parámetros utilizados se describen a continuación:

- **Hormigas:** Cada hormiga representa una solución.
- **Iteraciones:** Es el número de veces que se repite el proceso.
- **Restricciones incumplidas:** Número de veces que incumple una regla.
- **Tiempo:** Periodo determinado durante el que se realiza el proceso en segundos.

Los datos que se utilizan corresponden al distributivo 2016 de la FCI que cuenta con:

- 25 paralelos que pertenecen al ciclo 1.
- 5 mallas de la carrera de sistemas, eléctrica, mecánica, telemática y seguridad industrial.
- 86 docentes con sus respectivos cargos.
- 166 clases, compuestas por docentes con sus respectivas materias

### Prueba 1.

En siguiente tabla se presenta los resultados de las pruebas realizadas con 50 iteraciones.

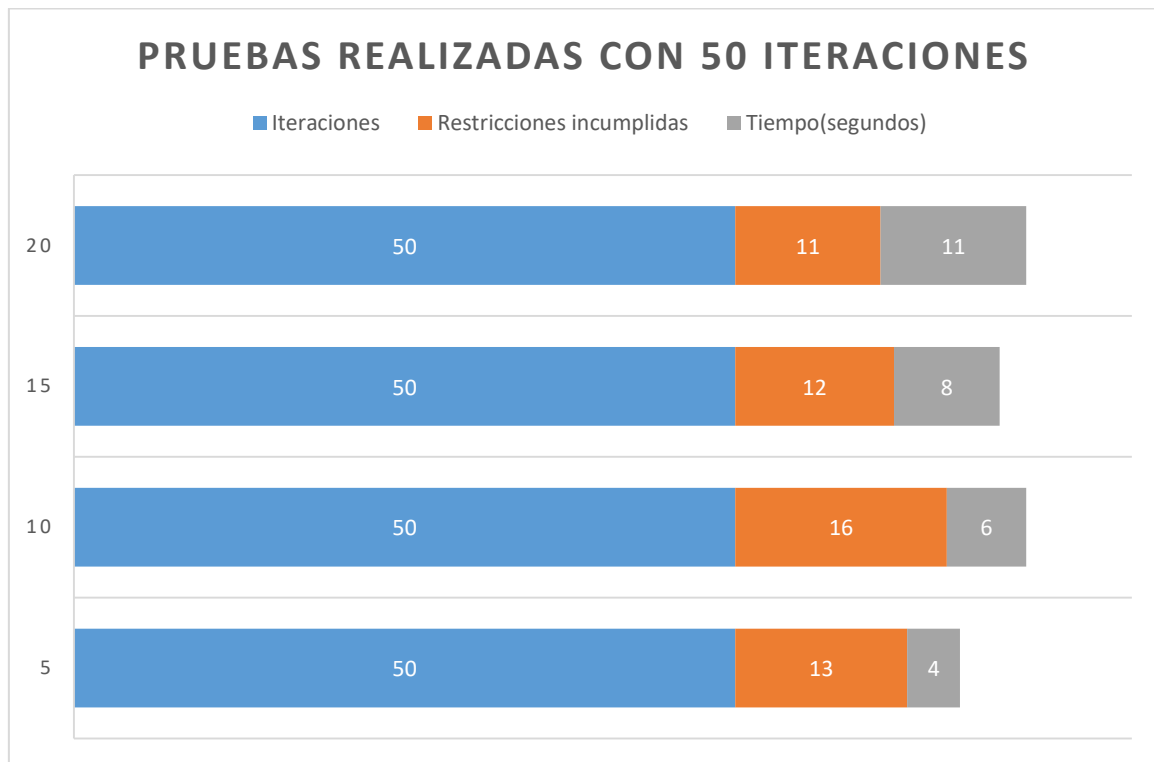
*TABLA: 36 Prueba 1 de 50 iteraciones para obtener un horario óptimo.*

Nº hormiga	Iteraciones	Restricciones incumplidas	Tiempo(segundos)
5	50	13	4
10	50	16	6
15	50	12	8
20	50	11	11

**FUENTE: CONOCIMIENTO PROPIO**

**ELABORADO: LOS AUTORES**

*FIGURA: 9 Promedio de 50 iteraciones.*



**FUENTE: CONOCIMIENTO PROPIO**

**ELABORADO: LOS AUTORES**

De los datos de la Tabla 37 y la Figura: 9 se analiza que tienen un alto índice de incumplimientos de restricciones por lo que no se debe tomar un número de iteraciones bajas para la generación de horarios.

## Prueba 2.

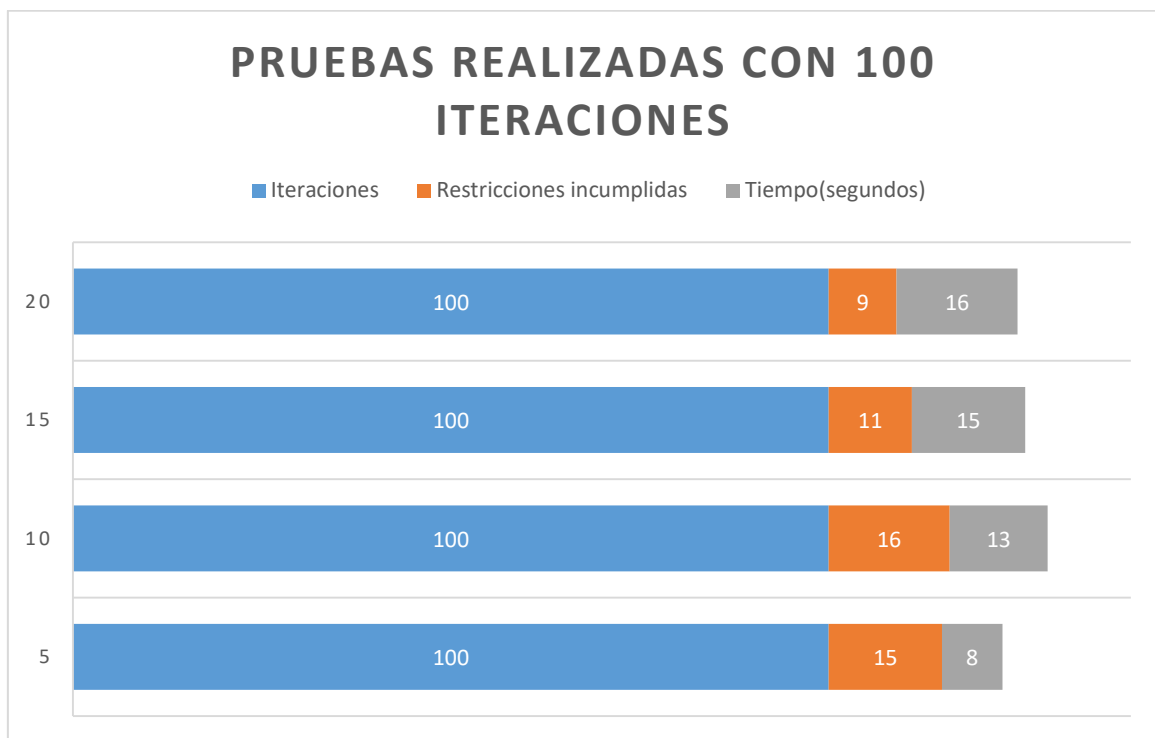
En siguiente tabla se presenta los resultados de las pruebas realizadas con 100 iteraciones.

*TABLA: 37 Prueba 2 de 100 iteraciones para obtener un horario óptimo.*

Nº hormiga	Iteraciones	Restricciones incumplidas	Tiempo(segundos)
5	100	15	8
10	100	16	13
15	100	11	15
20	100	9	16

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

*FIGURA: 10 Promedio de 100 iteraciones.*



**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

De los datos de la Tabla # 37 y la Figura: 10 se analiza que tienen un alto índice de incumplimientos de restricciones por lo que no se debe tomar un número de iteraciones bajas para la generación de horarios.



### Prueba 3.

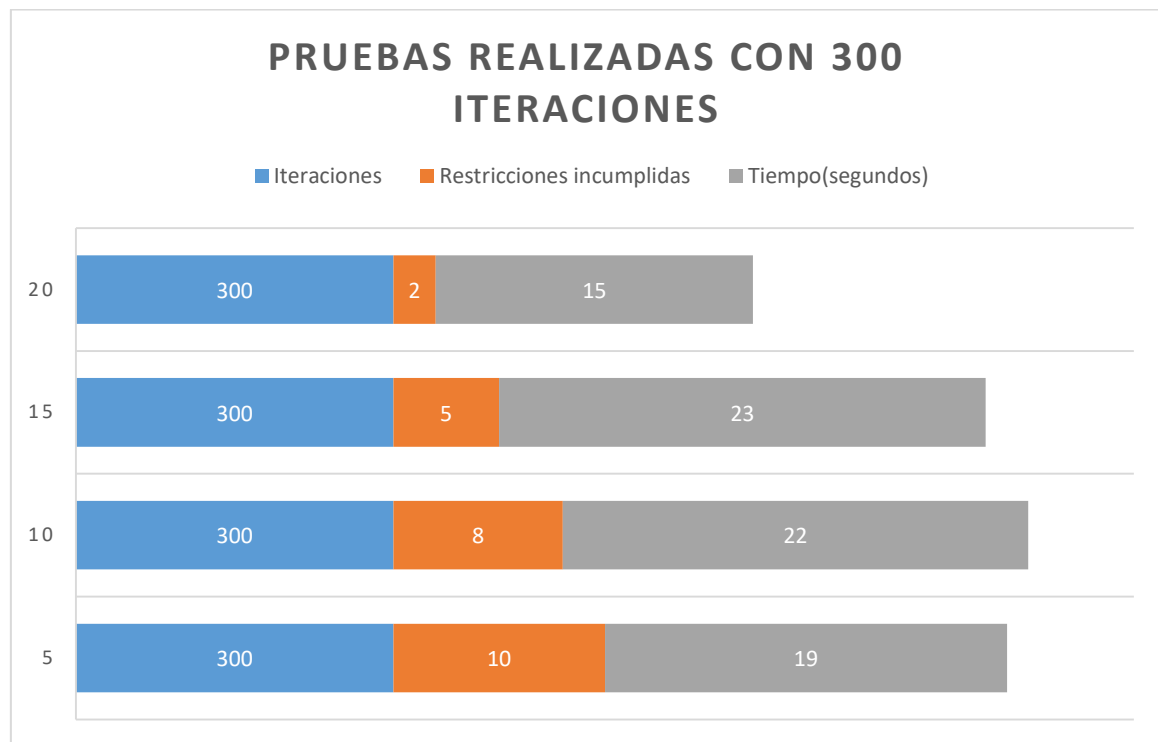
En siguiente tabla se presenta los resultados de las pruebas realizadas con 300 iteraciones.

*TABLA: 38 Prueba 3 de 300 iteraciones para obtener un horario óptimo.*

N° hormiga	Iteraciones	Restricciones incumplidas	Tiempo(segundos)
5	300	10	19
10	300	8	22
15	300	5	23
20	300	0	12

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

*FIGURA: 11 Promedio de 300 iteraciones.*



**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

De los datos de la Tabla # 38 y la Figura 11 se analiza que tienen un bajo índice de incumplimientos de restricciones con un número de iteraciones altas en la generación de horarios.

#### Prueba 4.

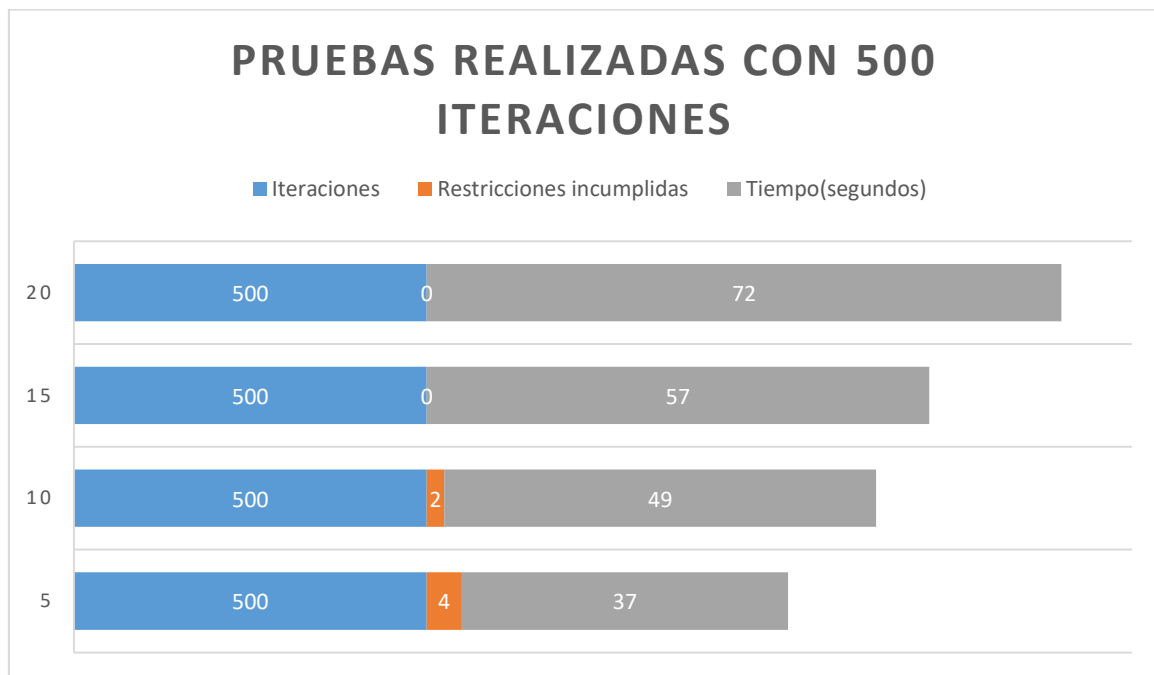
En siguiente tabla se presenta los resultados de las pruebas realizadas con 500 iteraciones.

*TABLA: 39 Prueba 4 de 500 iteraciones para obtener un horario óptimo.*

N° hormiga	Iteraciones	Restricciones incumplidas	Tiempo(segundos)
5	500	4	37
10	500	2	49
15	500	0	57
20	500	0	72

**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

*FIGURA: 12 Promedio de 500 iteraciones.*



**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**

De los datos de la Tabla # 39 y la Figura 12 se analiza que tienen un bajo índice de incumplimientos de restricciones con un número de iteraciones altas en la generación de horarios. tomando en cuenta los resultados de las pruebas se define que con 500 iteraciones y 20 hormigas se puede obtener un horario casi libre de restricciones incumplidas y con un tiempo variable.

#### **4.1.2. Módulo de reconstrucción de horario.**

El modulo reconstrucción de horario permite al usuario administrador realizar cambios tangibles al horario, permitiendo cambiar al docente a otra hora o día que el docente solicite.

Tal como se representa en el anexo de la **Figura 21**: con sus respectivas restricciones obligatorias para evitar un posible choque de horas, verificando este proceso antes de guardar y antes de seleccionar el horario.

## **CAPÍTULO V**

### **CONCLUSIONES Y RECOMENDACIONES**

## 5.1. Conclusiones.

En el presente trabajo de tesis se desarrolló la implementación de una aplicación web que permita encontrar una solución computacional al problema de generación de horario para la Universidad Técnica Estatal de Quevedo, del cual se puede concluir lo siguiente:

- Se realizó un estudio de la información necesaria para crear horarios que se maneja en la Universidad Técnica Estatal de Quevedo como facultad, carrera, horarios de clase en las tres secciones, además se obtuvo el distributivo académico 2015-2016 en el área de coordinación de la FCI donde pudimos identificar la estructura de clases que maneja datos tales como Docentes con sus respectivo cargo de gestión, Materias definidas por facultad y carrera. Como resultado se obtuvo el diseño de la base de dato y del módulo de retroalimentación para que la aplicación permita cargar la información necesaria para la gestión de horarios.
- El problema se modela como un problema de satisfacción de restricciones donde la función objetivo está representada por:
  - Cantidad de horas por materias sin asignas en el horario
  - Cantidad de materia asignada al horario que exceden las 3 horas seguidas.
  - Cantidad de docentes con diferentes materias que son asignados en el horario en un mismo paralelo y mismo día.
  - Cantidad de docentes con cargo de decano (1) y coordinador (2) que están asignados fuera de las primeras horas.

Además se realizó una ponderación de las cuatro restricciones para evaluar la calidad del horario. En el proceso de búsqueda se aplicó el algoritmo Optimización Colonia de Hormigas con una función heurística que determina cual profesor es más conveniente planificar su horario en función del nivel de prioridad que tenga. Los resultados computados en las pruebas realizadas en base a los siguientes factores: Numero de cursos y numero de clases por periodo., con esta información se puede encontrar una solución a partir de 500 iteraciones y 20 hormigas entre 57 y 72 segundos. Cumpliendo con todas las restricciones obligatorias e incumpliendo la menor cantidad de restricciones.

- Un modelo de reconstrucción es un proceso que definitivamente mejora la calidad de un resultado; Al aplicar dicho modelo en un horario elaborado por el módulo de generación de horario que utiliza la optimización de colonia de hormigas donde cabe destacar que los resultados no son 100% precisos sino más bien soluciones lo más aproximadas a la optimización del problema, por lo tanto la implementación del modelo de reconstrucciones permite mejorar la calidad del resultado por medio de la intervención manual del administrador a cargo del sistema.

Terminamos concluyendo que esta aplicación resulta ser una buena opción para mejorar la gestión y elaboración de horarios que actualmente se maneja en la Universidad Técnica Estatal de Quevedo, cuando el costo computacional es la prioridad.

## **5.2. Recomendaciones.**

Se plantearon las siguientes recomendaciones de acuerdo a los resultados obtenidos en el proceso de creación de horarios.

A la hora de cargar los datos se recomienda utilizar solo un archivo Excel, pero antes ingresar información requerida por la aplicación, el usuario administrador debe tener el distributivo clasificado y ordenado de acuerdo a los nuevos requerimientos que se puedan ser establecidos.

Se recomienda probar con otras metaheurísticas como: Algoritmos genéticos, Inteligencia de enjambre, Búsqueda dispersa (Scatter Search), Redes Bayesianas entre otras más para mejorar el algoritmo y presentar más opciones a seleccionar que permitan encontrar las posibles soluciones.

En el proceso de reconstrucción de horarios se podría mejorar al utilizar otros frameworks web, para tener una experiencia mucho menos rustico de los procesos, con la finalidad de mejorar la interface donde interactúa el usuario administrador.

Para que la aplicación pueda intercambiar información con otro software de manera segura, es bastante recomendable establecer una validación de los datos mediante esquemas o documentos de definición de datos, de modo que se muestren de manera explícita todos los datos, sus tipos, y las restricciones y validaciones que se efectúan sobre estos.

**CAPÍTULO VI**  
**BIBLIOGRAFÍA**



## Bibliografía

- [1] L. Á. MUNÁRRIZ, FUNDAMENTOS DE INTELIGENCIA ARTIFICIAL, II ed., Murcia: Selegráfica S.L., 1994.
- [2] A. G. Serrano, Inteligencia artificial: fundamentos, práctica y aplicaciones, España: RC Libros, 2012.
- [3] G. a. P. Bratle, Fundamentos de Algoritmos, Madrid: Prentice Hall, 1997.
- [4] S. Druckmann, Evolutionary Algorithms, New York: Springer New York, 2015.
- [5] K. T. a. S. K. K.F. Man, Genetic Aigorithms, Berlin: Springer, 2001.
- [6] M. Muñoz, «Asignación dinámica de recursos con técnicas bio inspirada para un sistema de control de temperatura MIMO,» Universidad del Valle, Facultad de Ingeniería. , Santiago de Cali, 2005.
- [7] Julio Brito Santana, Clara Campos Rodríguez, Félix C. García López, Miguel García Torres,Belén Melián Batista, José A. Moreno Pérez, J. Marcos Moreno Vega, «Metaheurísticas,» Universidad de La Laguna, Canarias, España, 2004.
- [8] S. Costabel, Metaheurísticas aplicadas a un Problema de Asignación de Salones y Horarios a Asignaturas, Montevideo: Montevideo, 2006.
- [9] I. J. M. M. Caballero, *Asignación de horarios de clases universitarias mediante algoritmos evolutivos.*, Barranquilla: [www.acofi.edu.co](http://www.acofi.edu.co), 2010.
- [10] L. F. J. Pitol, «Uso de algoritmos evolutivos para resolver el problema de asignación de horarios escolares en la facultad de Psicología de la universidad de veracruzana,» Universidad de veracruzana, Veracruz, 2008.
- [11] B. M. G. ALVAREZ y J. A. Q. CÁRDENAS, «2.2.4. Análisis, diseño e implementación de un sistema informático para el apoyo al proceso de asignación de la carga académica usando programación con restricciones,» UNIVERSIDAD TECNOLÓGICA DE PEREIRA, PEREIRA, 2010.

- [12] E. Menéndez, «Metaheurística de optimización mediante colonias de hormigas y aplicaciones.,» Universidad Central de las Villas, España, 2010.
- [13] S. C. O. F. d. V. I. y. H. F. Alonso, Optimización inteligente: técnicas de inteligencia computacional para optimización., Málaga: Universidad de Málaga. Servicio de Publicaciones e Intercambio Científico, 2004.
- [14] L. M. G. a. F. d. L. V. Maniezzo, Ant Colony Optimization, Berlin: Springer-Verlag Berlin Heidelberg, 2004.
- [15] R. H. J. M. P. y P. A. R. , «Programación de Horarios de Clases y Asignación de Salas para la Facultad de Ingeniería de la Universidad Diego Portales Mediante un Enfoque de Programación Entera.,» *Revista Ingeniería de Sistemas*, vol. XXII, nº 22, pp. 121-141, 2013.

## **CAPÍTULO VII**

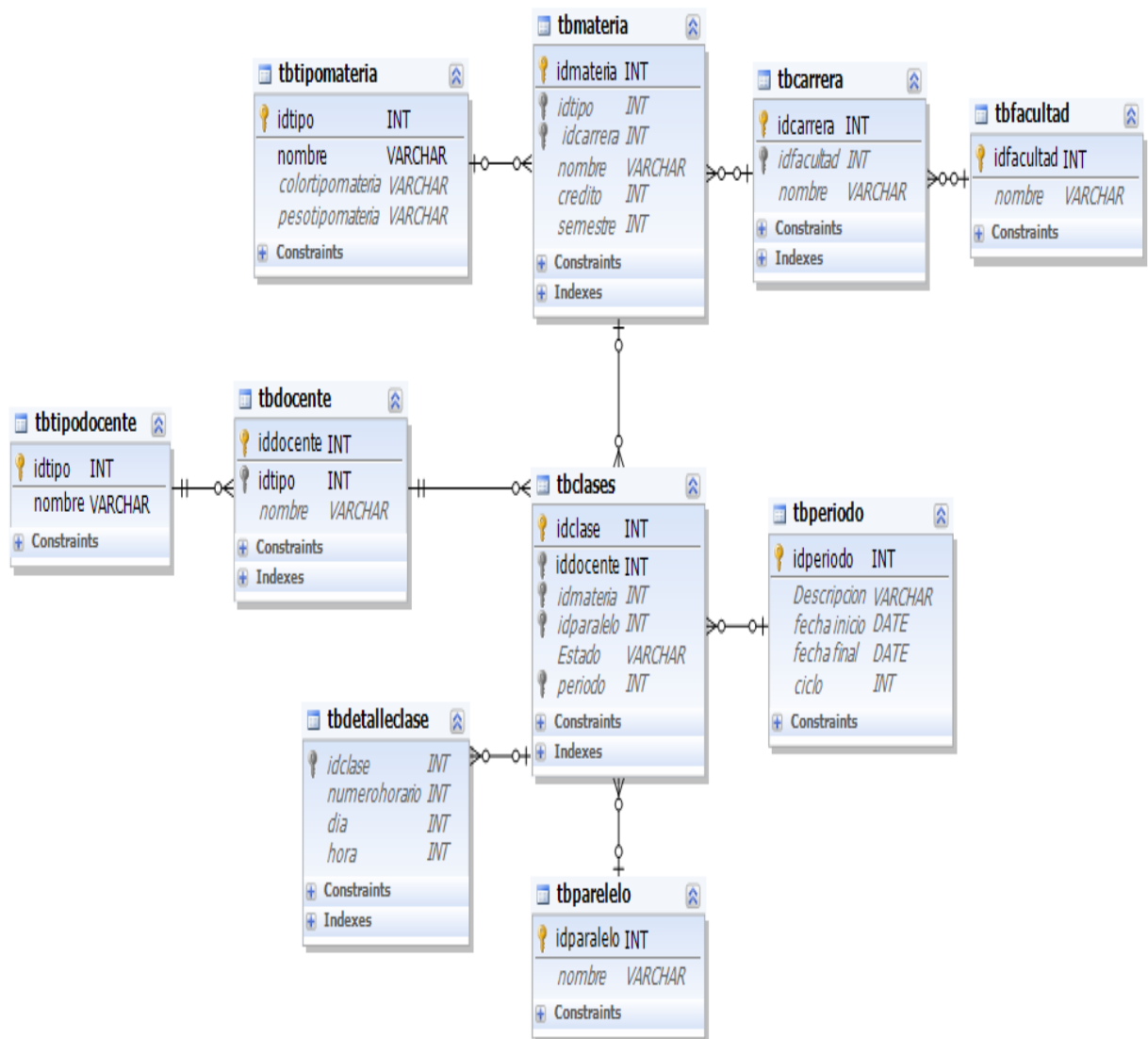
### **ANEXOS**

## Anexo 1.

### MANUAL DEL PROGRAMADOR.

#### Esquema de la base de datos.

FIGURA: 13 Esquema de la base de datos.



FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES

## Anexo 2.

### MANUAL DEL USUARIO.

Captura de pantallas de la aplicación y descripción de sus funcionalidades

**Pantalla principal:** contiene el menú y sus respectivos submenús con las diferentes funcionalidades que la caracterizan

*FIGURA: 14 Pantalla principal de la aplicación web.*



FUENTE: CONOCIMIENTO PROPIO

ELABORADO: LOS AUTORES

En la opción administración se compone de dos módulos el Módulo de administración y el módulo de carga de datos que se presentan en las capturas de pantallas.

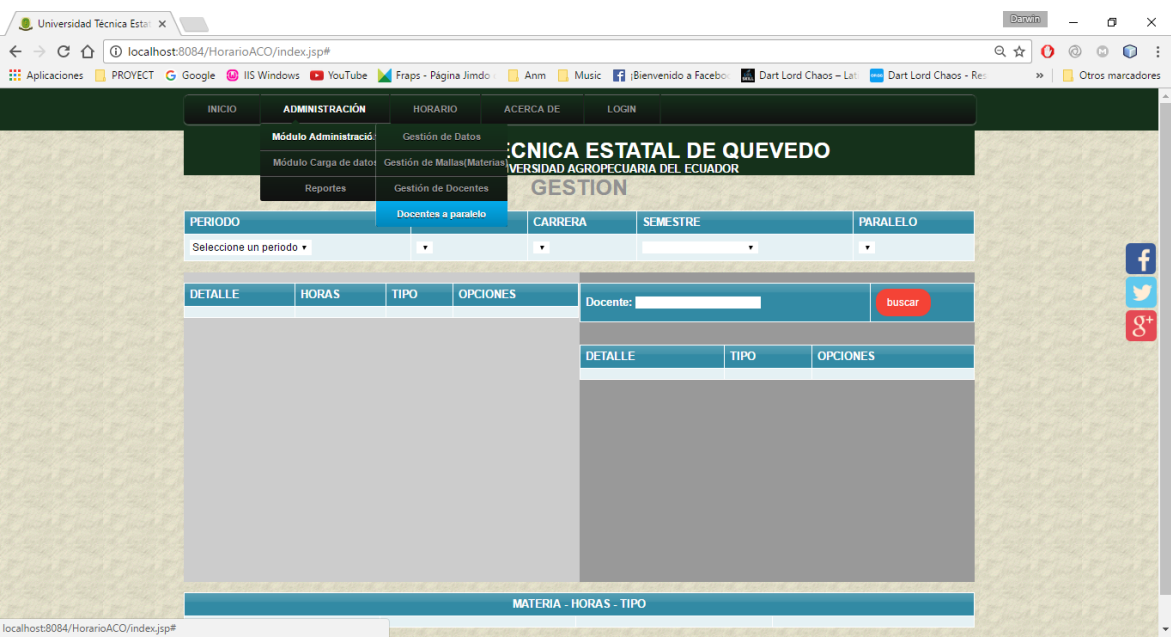
*FIGURA: 15 Administración gestión de datos para el horario.*



FUENTE: CONOCIMIENTO PROPIO

ELABORADO: LOS AUTORES

FIGURA: 16 Asignación de docentes a paralelos.



FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES

FIGURA: 17 Carga de datos de docentes de un archivo en formato .xls



FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES

FIGURA: 18 Carga de datos de materias de un archivo en formato .xls.

op 1	OP 2	OPCIONES	OPCIONES	OPCIONES	OPCIONES	OPCIONES
Tipo de materia	Carrera	Nombre de materia	horas	Semestre	Estado	editar
2	1	COMUNICACION INTERPERSONAL Y TECNICA	3	1	A	editar
1	1	FUNDAMENTOS DE INFORMATICA	5	1	A	editar
1	1	LOGICA MATEMATICA	4	1	A	editar
1	1	SISTEMAS DE ECUACIONES	5	1	A	editar
		SISTEMAS DE PROGRAMACION	5	1	A	editar

FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES

En la opción Horario se encuentra los siguientes módulos el módulo de generación de horario y el módulo de reconstrucción de horarios que se presentan en las capturas de pantallas.

FIGURA: 19 Asignación de horas a un docente y paralelo específico.

PERIODO	CARRERA	PARALELO
Ciclo 1 de 2016	Sistemas	Primero A sistemas

DOCENTE	MATERIA	HORAS	Acción
JARAMILLO CHUQUI IVAN	FUNDAMENTOS DE INFORMATICA	5	Seleccionar
CEDEÑO BARZOLA JIMY ALDRIN	LOGICA MATEMATICA	4	Seleccionar
CEDEÑO BARZOLA JIMY ALDRIN	SISTEMAS DE ECUACIONES	5	Seleccionar
CHIRIBOGA CASANOVA WASHINGTON	FUNDAMENTOS DE PROGRAMACION	5	Seleccionar

CARRERA:  PARALELO:

CARRERA: SISTEMAS DE ECUACIONES PARALELO: Primero A sistemas

Revisar Guardar hora

HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES



FIGURA: 20 Generación de horario.

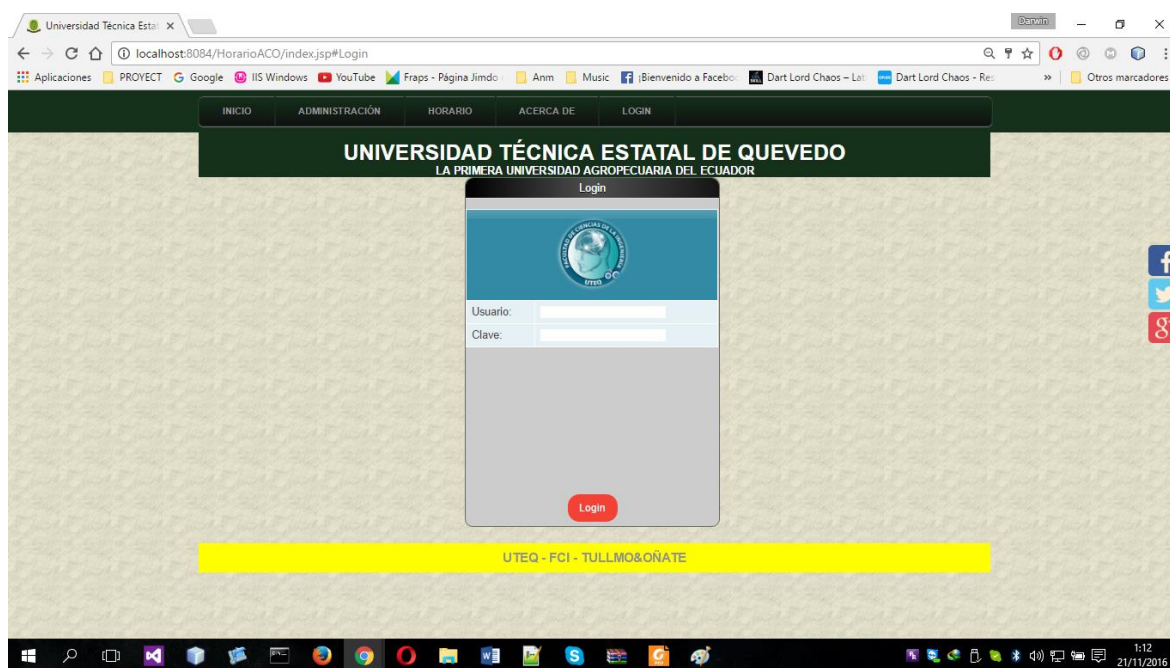
FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES

FIGURA: 21 Reconstrucción de horario.

FUENTE: CONOCIMIENTO PROPIO  
ELABORADO: LOS AUTORES



*FIGURA: 22 Inicio de sesión para realizar gestiones en la aplicación web.*



**FUENTE: CONOCIMIENTO PROPIO**  
**ELABORADO: LOS AUTORES**