

**UNIVERSIDAD NACIONAL JOSÉ MARÍA
ARGUEDAS
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**



Presentado por

PERCY ALHUAY CARRASCO

**DESARROLLO DE UN SISTEMA DE GENERACIÓN
DE HORARIOS ACADÉMICOS CON ALGORITMOS
GENÉTICOS PARA LA ESCUELA PROFESIONAL
INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD
NACIONAL JOSÉ MARÍA ARGUEDAS EN EL AÑO
2015.**

Asesor

M. Sc. Ing. EDWIN ROQUE TITO

ANDAHUAYLAS – APURÍMAC - PERÚ

2016

DEDICATORIA

Dedico el presente proyecto de tesis principalmente a Dios que me dio la salud, la fuerza, la sabiduría, la alegría y me ha ayudado en todos los momentos difíciles de mi carrera universitaria.

A mis queridos padres Francisco Alhuay H. y Julia M. Carrasco C. quien me han dado la vida y han luchado incansablemente por darme la educación y velar por mi bienestar, ustedes han sido, son el ejemplo más grande que tengo y me siento muy orgulloso de ser su hijo.

A mis hermanas Ayude y Nayda, a mi hermano Michael que ustedes supieron apoyarme constantemente y por sus amores incondicionales.

EPÍGRAFE

La felicidad no consiste simplemente en estar bien, sino en estar haciendo algo que te llene la vida (Julián María).

No imites a nadie, sé siempre la mejor versión de ti mismo (David Fischman).

AGRADECIMIENTO

Agradezco a Dios por haberme protegido durante todo el camino y darme la fuerza para superar los obstáculos y dificultades de mi vida.

A mi madre por su demostración ejemplar a no desfallecer ni rendirnos ante nada y a perseverar mediante sus sabios consejos.

A los tres jurados de este proyecto y a mi asesor, mi enorme agradecimiento por su comprensión y experiencia compartida durante el desarrollo de mi proyecto.

A todos mis amigos/compañero que me apoyaron y con quienes he compartido momentos inolvidables.

INDICE

DEDICATORIA.....	iv
EPÍGRAFE	v
AGRADECIMIENTO	vi
INDICE	vii
RESUMEN	xiv
ABSTRACT	xv
CAPITULO 1 INTRODUCCIÓN	1
1.1. REALIDAD PROBLEMÁTICA	2
1.2. FORMULACIÓN DEL PROBLEMA.....	5
1.3. OBJETIVO	5
1.3.1. objetivo general	5
1.3.2. objetivos específicos	5
1.4. JUSTIFICACIÓN.....	6
1.5. VIABILIDAD DE LA INVESTIGACIÓN.....	7
1.5.1. Viabilidad técnica	7
1.5.1.1.Hardware.....	7
1.5.1.2.Software	7
1.5.2. Viabilidad temporal.....	8
1.5.3. Viabilidad económica.	8
1.5.3.1.Costos generales	8
1.5.3.2.Costos de hardware y software.....	8
1.5.3.3.Costo de personal	8
1.5.3.4.Costo inicial de desarrollo del sistema.....	9
1.5.3.5.Gastos operativos del sistema	9
1.5.3.6.Beneficios.....	9
1.5.3.7.Flujo de caja	11
1.5.4. Viabilidad operativa	12
1.5.4.1.Relación de costo y beneficio	12
1.6. LIMITACIÓN DEL ESTUDIO	12
CAPITULO 2 REVISIÓN DE LITERATURA	14
2.1. Antecedentes	14
2.1.1. Técnicas existentes para la resolución del problema	14
2.1.1.1.Técnicas secuenciales basadas en grafos	14
2.2. BASES TEÓRICAS.....	20
2.2.1. Algoritmo	20
2.2.2. Problema NP	20
2.2.2.1.La fase de suposición	21

2.2.2.2.La fase de verificación,	21
2.2.2.2.1.Problemas de optimización	21
2.2.2.2.2.Algoritmos Meta-heurísticos	22
2.2.2.2.3.Técnicas basadas en algoritmos de población	22
2.2.3. Conceptos de Timetabling	23
2.2.3.1.Problema Timetabling.	23
2.2.3.2.Solución al problema del Timetabling	23
2.2.3.2.1.Técnicas tradicionales.....	23
2.2.3.2.1.1.Programación entera.....	24
2.2.3.2.1.2.Programación lineal.	24
2.2.3.2.1.3.Bactracking	24
2.2.3.2.2.Técnicas no tradicionales.	26
2.2.3.2.2.1.Redes neuronales	26
2.2.3.2.2.2.Reconocido simulado.....	27
2.2.3.2.2.4.Algoritmo evolutivo.....	28
2.2.3.3.Algoritmo Genético.....	31
2.2.3.4.Elementos fundamentales del horario genético.	32
2.2.3.4.1.Estructura del cromosoma.	32
2.2.3.4.2.Estructura de la Población Inicial.	33
2.2.3.4.3.Operadores genéticos.....	34
2.2.3.5.Algoritmo Genético simple.	34
2.2.3.6.Descripción del funcionamiento del algoritmo genético.	35
2.2.4. Estándares para desarrollo de proyecto	36
2.2.4.1.ISO/IEC 9126	36
2.2.5. Metodología para gestión de proyecto.....	36
2.2.6. Metodología de desarrollo XP (Programig Express).....	39
2.2.7. Herramientas utilizadas en el desarrollo de software.	39
2.2.7.1.Java.	39
2.2.7.2.Hibernate.....	40
2.2.7.3.Netbeans	41
2.2.7.4.Librería JGAP	42
2.2.7.5.Base de datos	43
2.2.7.6.Mysql.....	44
2.3. Definición de términos básicos para creación de un sistema de horario.....	45
CAPITULO 3 MATERIALES Y METODOLOGÍA	47
3. Ingeniería del proyecto.....	47
3.1. GESTIÓN DE PROYECTO - GUÍA DE PMBOK.....	47
3.1.1. Gestión de Integración	47
3.1.2. Gestión del Alcance	50

3.1.3.	Gestión del Tiempo	52
3.1.4.	Gestión de costes.....	52
3.1.5.	Gestión de calidad.....	53
3.1.6.	Gestión de recursos humanos	55
3.1.7.	Gestión de comunicación	55
3.1.8.	Gestión de riesgos	56
3.1.9.	Gestión de compras	57
3.1.10.	Gestión de los interesados.....	57
3.2.	METODOLOGÍA DE DESARROLLO DE SOFTWARE XP (PROGRAMACIÓN EXTREM)	58
3.2.1.	Planificación	58
3.2.2.	Diseño	66
3.2.4.	Codificación.....	91
3.2.4.1.	Técnicas de AG aplicado en la programación.	96
3.2.5.	Pruebas	101
3.3.	REPRESENTACIÓN DEL PROBLEMA GENÉTICO – CON GRAFICO.....	101
3.3.4.	Operadores Genéticos	103
3.3.5.	Estándares de desarrollo de software.....	104
3.3.6.	Herramientas de desarrollo de software	105
CAPÍTULO 4 RESULTADOS Y DISCUSIONES		106
4.1.	VALORACIÓN DE LA SOLUCIÓN	106
4.1.1.	Primeros casos de análisis	106
4.1.2.	Segundo caso de análisis.....	108
4.2.	EVALUACIÓN Y COMPARACIÓN DE RESULTADOS.....	110
CONCLUSIÓN		111
RECOMENDACIÓN		113
REFERENCIAS BIBLIOGRÁFICAS		114
ANEXOS:		118
ANEXO2: Distribución de carga del DAITI de los contratados cambios 2016 - II.		121
ANEXO 3: Distribución de carga del DAITI de los contratados cambios 2015.		122
ANEXO 4: Distribución de carga del DAITI de los nombrados cambios 2015.		123
ANEXO 5: Distribución de carga electiva de docentes contratados académicos 2015 –II -		124
ANEXO 6: Distribución de carga electiva a docentes nombrados académicos 2015 –II -		125
ANEXO 7: Plan curricular de la Escuela Profesional de Ingeniería de Sistemas.....		126
ANEXO 8: Cuadro de equivalencias de las asignaturas del currículo 2007 y 2014-II....		129
ANEXO 9: Manual de uso del aplicativo de generación de horarios.....		131
ANEXO 10: Código del algoritmo genético en el proyecto		140
ANEXO 11: Resultados del horario por ciclo con algoritmo genético		151
ANEXO 12: Dato recogido del libro de acta de distribución de carga académica de la EPIS ...		156

ÍNDICE DE TABLA

Tabla 03. Diez principales países que emplean la tecnología de la información.	3
Tabla 04. Gastos Operativos	9
Tabla 05. Ahorro anual en Elaboración de horario	10
Tabla 06. Resumen de flujo de caja	11
Tabla 07: Área de gestión de alcance del proyecto.....	47
Tabla 08: Área de gestión de alcance del proyecto.....	50
Tabla 09: Gestión de tiempo del proyecto	52
Tabla 10: Área de gestión de costos	53
Tabla 11: Área de gestión de alcance del proyecto.....	53
Tabla 12: Área de gestión de recursos humanos	55
Tabla 13: Área de gestión de riesgo	56
Tabla 14: Área de gestión de compras	57
Tabla 15:Área de gestión de los recursos humanos	57
Tabla 16:Recolección de historias de registrar docente	58
Tabla 17:Recolección de historias de registrar salón	59
Tabla 18:Recolección de historias de registrar usuarios	59
Tabla 19:Recolección de historias de registrar curso	60
Tabla 20:Recolección de historias para modificar docente	60
Tabla 21:Recolección de historias para modificar salón	61
Tabla 22:Recolección de historias para modificar usuarios	61
Tabla 23:Recolección de historias para modificar curso	62
Tabla 24:Recolección de historias para eliminar docente	62
Tabla 25:Recolección de historias para eliminar salón	63
Tabla 26:Recolección de historias para eliminar usuarios	63
Tabla 27:Recolección de historias para eliminar curso	64
Tabla 28:Secuencia de actividades de Plan de entrega.....	64
Tabla 29:Modelo de Velocidad del proyecto.....	65
Tabla 30:Especificación de caso de uso registrar curso	70
Tabla 31:Especificación de caso de uso registrar curso	70
Tabla 32:Especificación de caso de uso registrar curso	71
Tabla 33: Especificación de caso de uso registrar curso	71
Tabla 34: Especificación de caso de uso modificar curso	72
Tabla 35:Especificación de caso de uso modificar docente	73
Tabla 36:Especificación de caso de uso modificar salón	73
Tabla 37:Especificación de caso de uso modificar curso	74
Tabla 38:Tarjeta CRC modulo usuario	87

Tabla 39:Tarjeta CRC modulo usuario	87
Tabla 40:Tarjeta CRC modulo docente	87
Tabla 41:Tarjeta CRC modulo horario	88
Tabla 42:Tarjeta CRC módulo salón.....	88
Tabla 43:Tarjeta CRC modulo usuario	88
Tabla 44:Tarjeta CRC modulo semestre	89
Tabla 45:Tarjeta CRC modulo usuario - mantenimiento	89
Tabla 46:Tarjeta CRC modulo curso – mantenimiento.....	89
Tabla 47:Tarjeta CRC modulo docente – mantenimiento.....	90
Tabla 48:Tarjeta CRC modulo horario – mantenimiento	90
Tabla 50:Tarjeta CRC modulo ciclo – Mantenimiento	91
Tabla 52:Representación del Gen en un cromosoma para el problema de horarios de EPIS.....	103
Tabla 53:Representación del gen para el problema de horarios de EPIS.....	103
Tabla 54:Representación del operador cruzamiento para el problema de horarios de EPIS.	104
Tabla 55:Primera prueba con el AG	106
Tabla 56:Segunda prueba con el AG.....	107
Tabla 57:Tercera prueba con el AG.....	107
Tabla 58:Cuarta prueba con el AG	107
Tabla 59:Cuadro de comparación de los Casos con AG.....	108
Tabla 60:Primera prueba con el AG con datos al 90%.....	108
Tabla 61:Segunda prueba con el AG con datos al 90%.....	109
Tabla 62:Tercera prueba con el AG con datos al 90%.....	109
Tabla 63: Cuarta prueba con el AG con datos al 90%	109

ÍNDICE DE FIGURAS

<i>Figura 01.</i> Diagrama de un algoritmo.	20
<i>Figura 02.</i> Programación lineal estructurado.	25
<i>Figura 03.</i> Programación Backtrackin -Lineal	26
<i>Figura 04.</i> El esquema general de un algoritmo evolutivo.....	29
<i>Figura 05.</i> Clasificación de técnicas de búsqueda	31
<i>Figura 06.</i> Definición de un algoritmo genético.....	32
<i>Figura: 07.</i> Representación de un cromosoma para el problema del horario	33
<i>Figura 08.</i> Ejemplo de población Inicial para horario.....	33
<i>Figura 09.</i> Seudocódigo del Algoritmo Genético	35
<i>Figura 10.</i> Esquema de un algoritmo genético simple.....	35
<i>Figura 12:</i> Arquitectura de Hibernate	41
<i>Figura 13:</i> Pantalla del generador de horarios.....	19
<i>Figura 14.</i> Arquitectura de sistema de una base de datos (2015).....	43
<i>Figura 14.</i> Diagrama de casos de uso registrar usuario.....	66
<i>Figura 15.</i> Diagrama de casos de uso registrar salón	67
<i>Figura 16.</i> Diagrama de casos de uso registrar curso	67
<i>Figura 17:</i> Diagrama de casos de uso registrar docente	68
<i>Figura 18:</i> Diagrama de casos de uso modificar curso	68
<i>Figura 19.</i> Diagrama de casos de uso modificar docente	69
<i>Figura 20.</i> Diagrama de casos de uso modificar salón.....	69
<i>Figura 21.</i> Diagrama de colaboración de registrar salón	75
<i>Figura 22.</i> Diagrama de colaboración de registrar curso.....	75
<i>Figura 23.</i> Diagrama de colaboración de registrar docente.....	76
<i>Figura 24.</i> Diagrama de colaboración de modificar docente.	76
<i>Figura 25.</i> Diagrama de colaboración de modificar curso	77
<i>Figura 26.</i> Diagrama de colaboración modificar salón	77
<i>Figura 27.</i> Diagrama de secuencia de registrar salón	78
<i>Figura 28.</i> Diagrama de secuencia de registrar Docente	79
<i>Figura 29.</i> Diagrama de secuencia de registrar curso	80
<i>Figura 30.</i> Diagrama de secuencia modificar salón.....	81
<i>Figura 31.</i> Diagrama de secuencia modificar docente	82
<i>Figura 33.</i> Diagrama de secuencia de registrar salón	83
<i>Figura 34.</i> Diagrama de componentes del sistema	84
<i>Figura 35.</i> Diagrama de despliegue	84
<i>Figura 36.</i> Diagrama de entidad relación.....	85
<i>Figura 37.</i> Diagrama de Clases	86
<i>Figura 38.</i> Presentación de Modelo Vista Controlador (M.V.L.)	92

<i>Figura 39.</i> Proceso de programación de modelo de negocio	92
<i>Figura 40.</i> Proceso de programación de modelo de conexión de datos	93
<i>Figura 41.</i> Proceso de programación de modelo de conexión de datos	94
<i>Figura 42:</i> Resultado del diseño de capa de presentación para crear nuevo docente	95
<i>Figura 43.</i> Interfaz para procedimiento de Mantenimiento.	96
<i>Figura 44.</i> Proceso de ejecución del algoritmo genético.	96
<i>Figura 45.</i> Programación de reproducir individuo.	97
<i>Figura 46.</i> Codificación del individuo mutar	97
<i>Figura 47.</i> Código de evaluar individuo	98
<i>Figura 48.</i> Código de imprimir población	98
<i>Figura 49.</i> Imprime el mejor individuo.	99
<i>Figura 50.</i> Ingresar datos del algoritmo genético	99
<i>Figura 51.</i> Resultado del horario del Ciclo I.....	100
<i>Figura 52.</i> Resultado del horario del Ciclo II.....	100
<i>Figura 53.</i> Resultado del horario del Ciclo III.....	100
<i>Figura 55.</i> Representación de la población inicial para el problema de horarios de UNAJMA. ..	102

RESUMEN

El presente proyecto informático tiene como objetivo buscar la solución al problema de Elaboración de horarios en la Escuela Profesional de Ingeniería de Sistemas de la Universidad Nacional José María Arguedas. La importancia de resolver el problema es optimizar el tiempo, dinero y la cantidad de personal que interviene en el proceso de Elaboración de horario académico, el software genético apoyará al personal responsable en la Elaboración de horario y dará un mejor resultado y/o satisfacción a los estudiantes y docentes de la Escuela Profesional Ingeniería de Sistema.

Durante la investigación del proyecto tecnológico, se analizó entre algunas técnicas basadas en algoritmos de población en búsqueda y optimización, tales como: la búsqueda tabú, los algoritmos miméticos, algoritmo colonia de hormigas, estrategias evolutivas, algoritmo evolutivo, reconocido simulado. Para identificar cuál de ellas posee las mejores características que permita resolver nuestra problemática se analizó distintas fuentes de información y se determinó utilizar Algoritmo Genético, lo cual pertenecen al grupo de técnicas de programación no tradicionales - metaheurísticas.

La solución al problema, se construyó con un algoritmo genético a partir de una población inicial generada por un algoritmo aleatorio. Se ha tomado como caso de estudio la Escuela Profesional Ingeniería de Sistemas de la Universidad nacional José María Arguedas. Para facilitar la búsqueda de esta solución se aplicaron los operadores de selección natural, casamiento, mutación, elitismo. Para su desarrollo se utilizó lenguaje de programación Java, con entorno de desarrollo **NetBeans**. Se obtuvieron los resultados de horario del primer ciclo a decimo ciclo, generados por el algoritmo genético.

En conclusión, que el algoritmo genético, de acuerdo a los resultados obtenidos de los dos casos de análisis del proyecto, no garantiza una buena solución al problema, si no se cumple con toda las necesidades o restricciones definidas, se recomienda que se debe cumplir toda las restricciones y necesidades de la institución para garantizar el mejor resultado del algoritmo genético.

Palabras claves: Timetabling, Algoritmo, Algoritmos Genéticos, Optimización, Ciclo Académica, Meta Heurísticas, técnicas no tradicionales, Algoritmo evolutivo.

ABSTRACT

The present computer project aims to find the solution to the problem of scheduling in the Professional School of Systems Engineering of the National University José María Arguedas. The importance of solving the problem is to optimize the time, money and the number of staff involved in the academic writing process, the genetic software will support the responsible staff in the preparation of the schedule and will give a better result and / or satisfaction to the students and teachers of the Professional School System Engineering.

During the investigation of the technological project, we analyzed some techniques based on algorithms of population in search and optimization, such as tabu search, mimetic algorithms, ant colony algorithm, evolutionary strategies, evolutionary algorithm, recognized simulated. In order to identify which one has the best characteristics to solve our problem, we analyzed different sources of information and determined to use Genetic Algorithm, which belong to the group of non-traditional programming techniques - metaheuristics.

The solution to the problem was constructed with a genetic algorithm from an initial population generated by a random algorithm. It has been taken as case study the Professional School Systems Engineering of the National University José María Arguedas. To facilitate the search for this solution, the operators of natural selection, marriage, mutation, and alcoholism were applied. For its development was used Java programming language, with NetBeans development environment. Results were obtained from the first cycle to the tenth cycle, generated by the genetic algorithm.

In conclusion, the genetic algorithm, according to the results obtained from the two cases of analysis of the project, does not guarantee a good solution to the problem, if it does not meet all the defined needs or constraints, it is recommended that all the Constraints and needs of the institution to ensure the best outcome of the genetic algorithm.

Keywords: Timetabling, Algorithm, Genetic Algorithms, Optimization, Academic Cycle, Meta Heuristics, nontraditional techniques, Evolutionary Algorithm.

CAPITULO 1 INTRODUCCIÓN

La creación de horarios académicos en las Universidades e instituciones de formación académica, siempre fue una preocupación al inicio de cada semestre académico, en optimizar la distribución del horario, que consiste en asignar para cada curso un docente, periodo, día y sala.

Al analizar la problemática a profundidad, se descubre que no es nada sencillo solucionar el problema, por lo tanto, la planificación y organización se requirió de un alto esfuerzo y bastante tiempo. En la planificación se descubrió que se debe aplicar las restricciones fuertes y suaves, por estas restricciones el problema se convirtió en muy complejo. Por lo tanto, el software de generación de horarios se llevará a cabo una mejora en el tiempo y un menor esfuerzo por parte de las personas que se encargan de la Elaboración.

La propuesta que se utilizó para este proyecto tecnológico es algoritmos genéticos para dar la resolución al problema de generación de horarios, tomando como dominio del problema los cursos que se dictan en la Facultad de la Escuela Profesional Ingeniería de Sistemas de la Universidad Nacional José María Arguedas.

1.1. Realidad problemática.

La tecnología desde sus inicios siempre dio un contexto diferente a las organizaciones en sus procesos de negocio, para el siglo XXI, la Tecnología de Información (TI), se ha convertido en uno de los factores más relevantes en cuanto al uso para las empresas y fue siempre un papel predominante para toda empresa (pública /privada) en el proceso de toma de decisiones, basado en una adecuada administración de la información, tiempo y recursos. En el siglo XXI, el elemento diferenciador entre empresas supervivientes y sobrevivientes, radica en el aprovechamiento y buen manejo de los recursos tecnológicos que les ofrece y la manera en que dichos recursos son explotados por cada una de las organizaciones, todos ellos relacionados con la manipulación de datos para proveer información clara, precisa y confiable que sea utilizada para la toma de decisiones oportuna y acertada. La toma de decisiones es una actividad crítica dentro de las organizaciones, por tal razón es imprescindible contar con herramienta automatizada de planificación que faciliten la consecución de resultados efectivos y en tiempos relativamente cortos, de tal manera que a través de su eficiente administración sea posible la toma de decisiones certera y oportuna, que la conduzcan a alcanzar las metas y objetivos planteados en una empresa.

Según Informe Mundial sobre Tecnología de la Información (IMTI, 2015) del Foro Económico Mundial (FEM), determina que la buena toma de decisiones certera y oportuna, conduce a alcanzar las metas y objetivos planteados dentro de una organización. Claros ejemplos de países que optaron por la buena toma de decisión, como se muestra en la **Tabla 03**, en el siglo XXI, se ha convertido en uno de los usuarios más eficaz en el uso de la tecnología digital/información del mundo, según el último FEM. Sin embargo, son las naciones europeas las que dominan la parte superior de la clasificación, al ocupar las diez primeras posiciones en el año 2015. Singapur es el único país asiático tras la débil actuación de Hong Kong o Corea del Sur.

Tabla 03:

Diez principales países que emplean la tecnología de la información.

Diez principales países que emplean tecnología de la información	
Índice de disposición a la conectividad 2015	Ranking mundial*
Singapur	1
Finlandia	2
Suecia	3
Países Bajos	4
Noruega	5
Suiza	6
Estados Unidos	7
Reino Unido	8
Luxemburgo	9
Japón	10

Fuente: Foro Económico Mundial, 2015
*Ranking 2015 de 143 economías.
**El Índice mide la forma en que las economías usan las oportunidades ofrecidas por las tecnologías de la información y la comunicación para un bienestar y una competitividad incrementadas.

Nota: Revista de foro economía mundial 2015.

<http://www.cdi.org.pe/InformeGlobaldeCompetitividad/index.html>

Por lo tanto, en las instituciones académicas como empresas industriales, entre otros, siempre han presentado dificultades para resolver la asignación de horarios. En el 2015 existen investigaciones y casos que tuvieron éxitos en Universidades/empresas, según el informe de la Universidad Politécnica de Valencia (España), menciona a continuación de sistemas implementados:

- ❖ Sistema de asignación Optimizada en Centros Distribución Comercial en Mercado S.A.(España).

Este proyecto, fue implementado en 2001, donde persigue la satisfacción total de las personas que componen, en conjunto de la plantilla, para lo que se han estandarizado diferentes tipos de horarios. Todos ellos contemplan una misma hora de entrada y salida, independientemente del día de la semana que se trabaje, y un número máximo de horas de trabajo diario y semanal, todo ello en horario de jornada continuada. Pero, además, los trabajadores conocen sus horarios con un mes de antelación, lo que sin duda les permite planificar y aprovechar al máximo su vida personal.

❖ Sistema Optihper- Turnos

El sistema OptiHper-Turnos, como adaptación del sistema informático Optihper, permite la asignación de turnos (M/T/N) en servicios 24/365 para (policía, bomberos, servicios hospitalarios, asistenciales, etc.). Puede asignar también los destinos del personal de acuerdo a sus cualificaciones.

Mientras tanto en Perú las instituciones tanto públicas y/o privadas de educación de nivel primario, secundaria, pre-universitarias y de estudios superiores, siempre presentan problemas en creación de horarios académicos al inicio del semestre académico, este problema de horarios dificulta el logro de las metas de cada institución de sector educación. La Universidad Nacional José María Arguedas, no es ajeno a la problemática específicamente la Facultada de Ingeniería de la Escuela Profesional Ingeniera de Sistemas, carece de un sistema de generación de horarios, software de este tipo sería un elemento básico para la adecuada administración de las clases como: cursos, días, horas y docentes, hasta el semestre académico 2015-II la EPIS, realiza de forma manual la generación de horarios. Una mala gestión al momento de elaborar la distribución de horarios puede significar una gran pérdida de tiempo por parte del personal involucrado en el proceso, lo cual se puede traducir en dinero perdido por la institución respectiva. Por otro lado, si no se escoge una distribución óptima, probablemente se estén desaprovechando varios recursos de la institución (aulas sin utilizar) y se puede estar sobrecargando a los docentes (pueden tener muchas horas seguidas de dictado). Los alumnos también se pueden ver afectados por esta distribución, ya que se puede programar muchas clases o evaluaciones consecutivas, lo cual afecta su rendimiento académico. En la EPIS, nunca antes se ha implementado una solución informática que brinde soporte eficiente para esta actividad, por supuesto hasta el semestre 2015-II, presenta insatisfacción, por los siguientes motivos mencionados:

- ❖ Tiempo prolongado entre una semana para la creación de los horarios.
- ❖ Los horarios generados de forma manual, ocasionan conflictos con horas.

- ❖ Al crear los horarios de forma manual se necesita transcribirlos en una hoja electrónica, para su futura publicación en los murales de institución.
- ❖ No existe alertas cuando hay conflictos, como cruce de horas para un mismo grupo o cuando a un profesor se le asigna más de una materia en la misma hora
- ❖ Uso excesivo de papel y energía.

En la escuela Profesional de ingeniería de sistemas de la UNAJMA, hay una necesidad para resolver un problema a través de un software que facilite y optimice el trabajo realizado.

1.2. Formulación del problema.

¿Se necesita desarrollar un sistema de generación de horarios académicos con algoritmos genéticos para la Escuela Profesional Ingeniería de Sistemas de la Universidad Nacional José María Arguedas en el año 2015?

1.3. Objetivo.

1.3.1. Objetivo General.

Desarrollar un sistema de generación de horarios académicos con algoritmos genéticos para la Escuela Profesional Ingeniería de Sistemas de la Universidad Nacional José María Arguedas en el año 2015.

1.3.2. Objetivos Específicos.

- a. Analizar el proceso actual de generación de horarios de la EPIS.
- b. Diseñar y construir un sistema de generación de horarios académicos con algoritmos genéticos para la Escuela Profesional Ingeniería de Sistemas de la Universidad Nacional José María Arguedas en el año 2015.
- c. Evaluar la eficiencia del algoritmo genético en los resultados de generación del horario en la Escuela Profesional de Ingeniería de Sistemas de la Universidad Nacional José María Arguedas en el año 2015.

1.4. Justificación.

En la Universidad Nacional José María Arguedas (UNAJMA), al inicio de cada semestre académico, genera los horarios de forma manual, este trabajo toma un determinado tiempo de una a dos semanas y los horarios realizados manualmente están sujetos a modificaciones debido a la inestabilidad en la disponibilidad de los profesores. Las modificaciones son constantes inclusive cuando las clases académicas ya iniciaron, situación que retrasa a los alumnos y profesores en culminar de manera óptima los sílabos de los cursos.

Debido a la necesidad, se plantea desarrollar un sistema de generación de horarios para la Escuela Profesional Ingeniería de Sistemas (EPIS) de la Facultad de Ingeniería, el sistema que se desarrollará tiene como finalidad brindarle un gran beneficio a la UNAJMA, debido a que se conseguirá disminuir el tiempo de ejecución del proceso, de esta manera se puede aprovechar al personal designado a este proceso para que cumplan otras funciones. También, permitirá una óptima distribución de horarios académicos, con lo cual se generará un uso adecuado de los recursos tanto profesores como aulas, con lo cual se disminuirá los tiempos muertos en los cuales los recursos pueden ser utilizados y se realizará una carga balanceada de las horas de dictado de los docentes. Por lo tanto, el propósito es optimizar los cruces de las asignaturas y horas, evitando los retrasos en cumplimiento de los sílabos de cada semestre académico.

El presente proyecto de tesis tecnológico tiene como objetivo generar horarios de clases con un mínimo de error y que se ajusta a la disponibilidad de los profesores de la EPIS. Para alcanzar el objetivo se propone utilizar un algoritmo genético basadas en método meta-heurístico, que permite generar horarios de clases en menor tiempo posible con una óptima distribución de horarios académicos de la EPIS, donde tendrán el beneficio de que los horarios de clases se encuentren distribuidos con el objetivo de que un alumno de un mismo ciclo académico pueda llevar todos los cursos especificados en la malla curricular, de esta manera se disminuyen los tiempos entre clase y clase. Además, se contará con una mejor distribución de exámenes por ciclo para que la carga académica disminuya.

1.5. Viabilidad de la investigación.

Se tiene la autorización del Presidente de la comisión de gobierno y del director de la Escuela Profesional ingeniería de sistemas, para el levantamiento de la información de la Universidad Nacional José María Arguedas. Con respecto a la recopilación de datos y la respectiva especificación de los requisitos se llevará a cabo mediante entrevistas al responsable de la EPIS, así como al departamento encargado de la distribución de horarios de la facultad. De esta manera se pretende poder obtener un conocimiento claro del proceso de Elaboración de horarios, para luego iniciar con el análisis de la solución. Es viable realizar el levantamiento de dicha información ya que se encuentra en la misma Universidad y no son datos confidenciales.

Teniendo la problemática se procederá a establecer las causas que amerita el desarrollo del sistema de generación de horarios, en esta sección se realiza un análisis de viabilidad según diversos criterios. Se evaluará la viabilidad desde un punto de vista técnico, operacional, económico, temporal y relación de costo y beneficio.

1.5.1. Viabilidad técnica.

Existen en el mercado las herramientas que se va utilizara, para el desarrollo del proyecto tecnológico, se evalúa en dos enfoques principales.

1.5.1.1. Hardware.

Para su desarrollo del proyecto se cuenta con un computador y para la implantación del proyecto tecnológico, los requisitos de funcionamiento son mínimos.

1.5.1.2. Software.

En cuanto al software, se cuenta con todas las aplicaciones necesarias para el desarrollo del proyecto y funcionamiento del mismo, los softwares que se va a utilizar no presentan una inversión alguna para su adquisición. El desarrollo de software se realizará en lenguaje Java, con NetBeans IDE 8.0.1, gestor de base de datos MYSQL- XAMP.

1.5.2. Viabilidad temporal.

Para este proyecto tecnológica, se cuenta con aproximadamente 12 meses para el desarrollo del producto; sin embargo, como se muestra en el plan de proyecto, se ha realizado un cronograma de avances que estima acabar el proyecto en 10 meses, considerando el tiempo de dedicación de 4 horas al día.

1.5.3. Viabilidad económica.

A continuación, se presenta la factibilidad económica para el desarrollo de software, donde se pone en manifiesto el equilibrio existente entre el costo del sistema y los beneficios que se derivan de este.

A continuación, se presenta un resumen de los costos que involucra el desarrollo del sistema e implantación del sistema propuesto y el proyecto será autofinanciado.

1.5.3.1. Costos generales.

Al lograr optimizar el proceso de generación de horarios, se minimizará el tiempo de trabajo manual, se reducirá los gastos en los materiales, un mínimo de error en asignar las horas y las asignaturas, también se disminuirá la cantidad de personal que cumple esta labor.

1.5.3.2. Costos de hardware y software.

Debido a que actualmente se cuenta con los equipos y recursos tecnológicos necesarios, para su desarrollo e implantación no será requerido ningún tipo de inversión en cuanto al software.

El software en donde se desarrollará es JAVA (IDE NetBeans IDE 8.0.1, MYSQL gestor de base de datos. Tiene la licencia gratuita.

1.5.3.3. Costo de personal.

El equipo de desarrollo de software conformado por el asesor con un costo de dos mil quinientos nuevos soles, que es un monto pagable, también un junior programador de dos mil nuevos soles y el investigador, no requiere de una inversión considerable.

1.5.3.4. Costo inicial de desarrollo del sistema.

Los costos de servicios, materiales, personal, costos de software y costos de hardware se detallaron cuadro de distribución de presupuestos del proyecto, corresponden a los 10 meses de Elaboración del proyecto sumando el total de costo del proyecto de tesis a **S/. 15,548.50** nuevo soles nuevos soles.

1.5.3.5. Gastos operativos del sistema.

Según la caja de flujo, los gastos operativos el primer año tiene cero gastos en mantenimiento y administración se muestra en la Tabla 04.

Tabla 04:

Gastos Operativos.

ITEM	DESCRIPCION	CANTIDAD	UNID. MEDIDA	PRECIO UNITARIO	PRECIO PARCIAL
I.01.00	GASTOS OPERATIVOS				S/. -
I.01.01	Administración y Mantenimiento	1	AÑO	-	-
COSTO TOTAL DE GASTOS OPERATIVOS					S/. -

Nota: Desarrollado por el tesista. Tesista

1.5.3.6. Beneficios.

Los beneficios serán estimados en base a algunos indicadores y otros aspectos que se lograrían al implantar el sistema, además se describirán los beneficios intangibles los cuales no serán calculados por ser beneficios sociales.

❖ **Beneficios tangibles.**

- Inversión en Elaboración de un horario académico de forma manual en los semestres académicos, registra s/. **S/. 7,088.80** nuevos soles en un año.
- Reducción del personal para el proceso de Elaboración de un horario académico. Para este beneficio se ha realizado el siguiente análisis que se demuestra en el cuadro.

Tabla 05:
Ahorro anual en Elaboración de horario.

SIN EL SOFTWARE			CON EL SOFTWARE		
ITEM	DESCRIPCION	Precio parcial	ITEM	DESCRIPCION	Precio parcial
1.00.00	BIENES	S/. 902.00	1.00.00	BIENES	S/. 50.00
1.01.00	Materiales de escritorio	S/. 542.00	1.01.00	Materiales de escritorio	S/. 25.00
1.02.00	Materiales Consumibles	S/. 360.00	1.02.00	Materiales Consumibles	S/. 25.00
2.00.00	SERVICIOS	S/. 1,120.00	2.00.00	SERVICIOS	S/. 50.00
2.01.00	Servicios diversos	S/. 1,120.00	2.01.00	Servicios diversos	S/. 50.00
4.00.00	RECURSOS HUMANOS	S/. 1,200.00	4.00.00	RECURSOS HUMANOS	S/. 100.00
4.01.00	Participacion por Area	S/. 1,000.00	4.01.00	Participacion por Area	S/. 50.00
5.00.00	GASTOS GENERALES	S/. 322.20	5.00.00	GASTOS GENERALES	S/. 20.00
	Imprevistos			Imprevistos	
	Imprevistos 10% del costo total	S/. 322.20		Imprevistos 10% del costo total	S/. 20.00
INVERSION TOTAL EN UN SEMESTRE		S/. 3,222.00	INVERSION TOTAL EN UN SEMESTRE		S/. 200.00
TOTAL EN DOS SEMESTRES		S/. 7,088.40	TOTAL EN DOS SEMESTRES		S/. 440.00
AHORRO ANUAL		S/. 6,648.40			

Nota: Desarrollado por el tesista.

Según el cálculo realizado se ha estimado un ahorro anual de **S/. 6,648.40 nuevos** soles en Elaboración del horario académico para la Escuela Profesional Ingeniera de sistemas de la UNAJMA ver la **Tabla 05**.

❖ **Beneficio al tener el sistema.**

- Reducción del tiempo de creación de Horario académico
- Reducción de error en conflicto de horas.
- Reducción de costo en materiales y recursos humanos.
- Mejor asignación de cursos y horas a los docentes.
- Mejora de la satisfacción de los alumnos.
- Incremento de la productividad del personal de la UNAJMA en otras actividades.

1.5.3.7. Flujo de caja.

En la **Tabla 06** se detalla de forma clara el flujo de caja del proyecto, demuestra que es factible implementar de acuerdo al resultado de VAN y TIR.

Tabla 06:

Resumen de flujo de caja.

	AÑO 0	AÑO 1	AÑO 2	AÑO 3	AÑO 4	AÑO 5	TOTAL
INVERSIÓN INICIAL	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)
BIENES	S/. 4,825.00	0.00	0.00	0.00	0.00	0.00	4,825.00
SERVICIOS	S/. 1,610.00	0.00	0.00	0.00	0.00	0.00	1,610.00
SOFTWARE	-	0.00	0.00	0.00	0.00	0.00	-
RECURSOS HUMANOS	S/. 7,700.00	0.00	0.00	0.00	0.00	0.00	7,700.00
GASTOS GENERALES	S/. 1,413.50	0.00	0.00	0.00	0.00	0.00	1,413.50
TOTAL INVERSIÓN INICIAL	15,548.50	0.00	0.00	0.00	0.00	0.00	15,548.50
GASTOS OPERATIVO	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)
ADMINISTRACIÓN Y GENERACIÓN DE HORARIO	0.00	0.00	0.00	0.00	0.00	0.00	-
TOTAL GASTOS OPERATIVOS	0.00	-	-	-	-	-	-
TOTAL DE GASTOS	15,548.50	-	-	-	-	-	15,548.50
TOTAL DE GASTOS ACUMULADOS	15,548.50	15,548.50	15,548.50	15,548.50	15,548.50	15,548.50	
BENEFICIOS	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)
INVERSIÓN EN ELABORACIÓN DE HORARIO ACADÉMICO	0.00	S/. 6,648.40	S/. 6,648.40	S/. 6,648.40	S/. 6,648.40	S/. 6,648.40	33,242.00
TOTAL BRUTO DE BENEFICIOS	0.00	6,648.40	6,648.40	6,648.40	6,648.40	6,648.40	33,242.00
TOTAL BRUTO DE BENEFICIOS ACUMULADOS	0.00	6,648.40	13,296.80	19,945.20	26,593.60	33,242.00	
TOTAL NETO DE BENEFICIOS	0.00	6,648.40	6,648.40	6,648.40	6,648.40	6,648.40	33,242.00
TOTAL NETO DE BENEFICIOS ACUMULADOS	0.00	6,648.40	13,296.80	19,945.20	26,593.60	33,242.00	
FLUJO DE CAJA NETO	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)	(S/.)
FLUJO DE CAJA NETO ANUAL	-15,548.50	6,648.40	6,648.40	6,648.40	6,648.40	6,648.40	17,693.50
FLUJO DE CAJA NETO ANUAL ACUMULADO	-15,548.50	-8,900.10	-2,251.70	4,396.70	11,045.10	17,693.50	
VAN	S/. 18,554.27						
TIR	32%						

Nota: Desarrollado por el tesista.

1.5.4. Viabilidad operativa.

Los beneficiarios tangibles del proyecto de investigación tecnológico serán los alumnos y la Universidad. La aplicación será bastante amigable con el usuario, en cuanto al entorno será capaz de realizar las funciones que solicite el usuario de manera sencilla

1.5.4.1. Relación de costo y beneficio.

El análisis de Costo – Beneficio presenta una ventaja para la UNAJMA, ya que se necesitará requisitos mínimos de software y hardware para su desarrollo e implantación del sistema de generación de horarios, donde no se necesitará gran cantidad de dinero para la tecnología que se requiere.

El sistema que se desarrollará traerá una mejora significativa en reducción del tiempo de trabajo, minimización de uso de recursos, efectivizar la asignación de los cursos y salones, con un mínimo de error y así lograr el objetivo de los sílabos de cada semestre académico.

1.6. Limitación del estudio.

La vital limitación para las investigaciones es la escasa información de trabajos previos, esto no significa la no existencia, sin embargo, ha sido difícil encontrar trabajos relacionados con la técnica no tradicional llamada algoritmos genéticos para problemas de horarios.

Otra limitación de esta investigación tecnológica es no poder realizar comparaciones por ser un proyecto tecnológico.

Se puede mencionar las limitaciones de proyecto de tesis tecnológico:

- El sistema es neto en idioma español
- El sistema que se implementará no se experimental, se limita a la relación de causa y efecto.
- No se puede instalar el sistema en versiones inferiores a Pentium IV.
- Se requiere como mínimo 1 GB de espacio en el disco duro para su instalación.
- Exclusivo para generación de horarios de clase para la EPIS.

- El sistema de horario no se responsabiliza en asignar los salones de acuerdo a la capacidad matriculado en los cursos.
- El software no divide en grupos de estudiantes
- El software asigna salones, no por las características del salón.
- No maneja los horarios individuales de los alumnos.

CAPITULO 2 REVISIÓN DE LITERATURA

2.1. Antecedentes.

2.1.1. Técnicas existentes para la resolución del problema.

La ASAP (Automated scheduling, optimization and planing group) es otro organismo cuyo fin principal es conducir investigación en modelos, heurísticas y algoritmos para la producción automática de soluciones de alta calidad a una gran variedad de problemas de planeación y optimización del mundo real, incluyendo Elaboración de horarios, logística, asignación de espacio y mantenimiento de inventarios.

2.1.1.1. Técnicas secuenciales basadas en grafos.

Los problemas de horarios pueden ser modelados como los problemas de coloración de grafos. Este problema consiste en representar a los eventos como los vértices y las restricciones fuertes como las aristas en grafo no dirigido. El objetivo del problema de los grafos es poder colorear todos los vértices, pero con la restricción que los vértices adyacentes no tengan el mismo color, esto se puede hacer de manera análoga en el problema del horario, con los vértices como eventos y los colores que se deben pintar en cada vértice reflejan un intervalo de tiempo particular, y las aristas entre los vértices reflejan el problema de la asignación de diferentes intervalos de tiempo que se le asigna a los eventos. Como al modelar el problema no se toman en cuenta las restricciones suaves es necesario que estas sean consideradas aparte, ya que esas restricciones determinan la calidad de la solución según Qu, Burke, Edmund y Barry (2006).

El método de coloreado de grafos aplicado en los problemas de horario es un método de construcción que ordena los eventos y se van asignando uno por uno, bajo el criterio de la dificultad que tengan al momento de realizar un horario. Debido a que la fase de ordenamiento es muy importante, a lo largo de la literatura existen diversos investigadores que han tratado de buscar mejores estrategias de ordenamiento. Burke, Edmund, Newall y Jim (2004) investigaron una nueva manera de realizar el proceso de ordenamiento dinámicamente

(por cada iteración al hallar la solución del problema el ordenamiento se iba adaptando). La actualización de cada iteración se lograba gracias a la experiencia obtenida en el proceso anterior dependiendo de la dificultad de asignación, de esta manera iba aprendiendo iterativamente. Además, cabe recalcar que de esta manera ya no es necesario hallar un ordenamiento inicial.

2.1.1.2. Técnicas basadas en restricciones.

Estas soluciones están basadas en las técnicas de programación de lógica de restricciones. Para el caso de los problemas de horarios se modela a un evento como una variable con dominios finitos, los valores de los intervalos de tiempo y salas se asignan secuencialmente para ir creando la solución. Además, se utiliza el backtraking, volver a través de los pasos, ya que cuando la asignación va avanzando y llega a un punto donde es imposible darle un valor a una variable se necesita volver al último estado donde se tenía otra posibilidad de elección, de esta manera va a ir formando otra solución posible. Este tipo de métodos generalmente tienen un costo computacionalmente elevado ya que el tiempo de ejecución aumenta exponencialmente con respecto al número de variables. Esta técnica por lo general no arroja buenas soluciones comparado con las otras soluciones existentes cuando el problema de optimización es complejo.

(Merlot, Boland y Stuckey, 2003) utilizaron la programación basadas en restricciones para producir soluciones iniciales. Luego, esta sirvió como entrada para mejorar las soluciones en otros métodos como los algoritmos de recocido simulado y escalando la montaña. Se aplicó para generar un horario de exámenes ordenándolos primero según el tamaño de intervalo de tiempo que tenía y luego se iba asignando uno por uno para formar una solución. Este híbrido fue probado en la Universidad de Melbourne.

Duong y Lam (2004) desarrollaron la programación basada en restricciones para crear una solución inicial y luego lo aplicaron en sus algoritmos de recocido simulado. Además, emplearon el backtraking para disminuir el esfuerzo de búsqueda. Este fue utilizado para solucionar el problema en la Universidad tecnológica HoChiMinh.

2.1.1.3. Técnicas basadas en búsquedas locales.

Angeles (2015) afirma que las búsquedas locales son aquellas que resuelven un problema realizando una búsqueda en su propio vecindario. El vecindario es creado a partir de la modificación mediante diversos operadores de una solución inicial, de esta manera se obtiene un espacio de búsqueda. Esta búsqueda posee una función objetivo para determinar qué solución es mejor para la generación de horarios. Dependiendo de los parámetros y las características del espacio de búsqueda se puede determinar el rendimiento y eficiencia del uso de este tipo de técnica. A continuación, se presentará dos técnicas muy nombradas en la literatura:

❖ **Búsqueda tabú.**

Este algoritmo consiste en la revisión de un espacio de búsqueda, pero tomando en cuenta de que no visita el mismo punto dos veces ya que este algoritmo posee una lista llamada *lista tabú* en la cual se conservan los últimos movimientos realizados. Ellos pueden determinar que se encontró una mejor solución en base a una estrategia de aspiración. Si es que no se encontró una solución revisando todo su óptimo local entonces puede saltar la búsqueda a otros vecindarios, si estos vecindarios no otorgan buenas soluciones este puede conservar la solución de su óptimo local.

(Gasparo y Schearf, 2001) aplicaron este algoritmo con la particularidad de que su vecindario de soluciones estaba conformado por soluciones que violaban algunas restricciones fuertes y débiles del problema, por lo tanto, se dedicaron a estudiar una estrategia de selección exhaustiva. Además, utilizaron una lista tabú dinámica y adaptaban la función de coste mientras que se realizaba la búsqueda. Este enfoque se aplicó en otros problemas y tuvo un resultado similar comparado con la utilización de otros algoritmos como miméticos.

❖ **Recocido simulado.**

Este método es basado en la analogía del simulado de recocido de los metales. El término de recocido se refiere a un proceso físico en el que un sólido es calentado mediante temperaturas altas para que este pase a una fase líquida y luego sea enfriado lentamente mediante la disminución de la temperatura. De esta manera se dice que las partículas enfriadas poseen menos energía.

En los problemas de optimización combinatoria esa fluctuación de energía aleatoria en el sistema se utiliza para escapar del mínimo local hacia el mínimo global. Si se hace analogía con el problema de timetabling, los estados del sistema vendrían a ser las soluciones factibles, la energía sería la función de coste, la temperatura vendría a ser el parámetro de control y el estado congelado la solución heurística del problema.

2.1.1.4. Técnicas basadas en algoritmos de población.

Son algoritmos metaheurísticos que se basan en el comportamiento natural. Son comúnmente usados para ayudar a resolver los problemas de optimización combinatorio como el problema de timetabling. A continuación, se presentará alguno de estos:

❖ **Algoritmo genético.**

Esta técnica consiste básicamente en tener una solución inicial representada en un cromosoma e ir evaluando este cromosoma mediante los operadores de mutación y cruce para que en la evolución se mejore la solución. Este método es comúnmente usado en la resolución de muchos problemas de timetabling según la afirmación de Qu, Burke, Edmund y Barry (2006). (Wong, Cote y Gely, 2002) utilizaron este método para resolver problema de generación de horarios de exámenes, este fue modelado para satisfacer las restricciones del problema. donde, se realizó un torneo de selección para escoger a los mejores padres para mejorar la producción de los candidatos.

❖ **Algoritmo memeticos.**

Este tipo de métodos combina la habilidad de la exploración de búsqueda de los algoritmos evolutivos con los métodos de búsquedas. Esta técnica presenta buenas soluciones, pero toma un alto tiempo computacional.

Los algoritmos Memeticos han sido aplicados a diferentes problemas de timetabling. En el 2008 se publicó “A Memetic Algorithm for the University Course Timetabling Problem” por Sadaf.N y Yang.S, en el cual se presenta una la solución al problema UCTP. Para ello se mezclaron dos técnicas de búsqueda local dentro de un algoritmo genético.

❖ **Algoritmo colonia de hormigas**

Esta técnica simula la búsqueda de los caminos más cortos que usan las hormigas cuando van a buscar su comida. Estas se guían dependiendo de la intensidad del olor de las feromonas que dejan en los caminos, en el más corto se intensifica el olor. Este tipo de algoritmos no son comúnmente usado en los problemas de timetabling pero (Naji, 2014) utilizó este algoritmo para resolver el problema de generación de horarios de exámenes.

2.1.1.5. Sistemas que existen en 2015.

2.1.1.5.1. Bunge.

Control de acceso a las distintas plantas del grupo (Puerto Gral. San Martín, Ramallo Tancacha, San, donde se controla el ingreso de empleados, empresas contratistas y visitas. Integrado con el sistema GestHor, se controla el presentimos de los empleados y las horas realizadas por cada una de las empresas contratistas, este sistema de gestión de tiempos y horarios del personal, implementado por Gesthor (2015) a las siguientes empresas.

- ❖ Grupo Techint (FINMA S.A.I.F.).
- ❖ Hospital Italiano (de Buenos Aires – Argentina).
- ❖ Terminales Portuarias Rio de la Plata.
- ❖ Autopista del Sol.
- ❖ Peñaflor.

2.1.1.5.2. UTTS (University timetabling scheduler)

Andrew (2002) desarrolló un sistema para la generación de horarios de clase y de exámenes para todas las facultades de la Universidad nacional de Singapur (UTTS). Este desarrollo básicamente se dividió en dos porciones uno que era la generación de clase y la otra de examen (UTTSExam).

El sistema se encuentra desarrollado en lenguaje java, usando Visual Age para java y Microsoft Access para la base de datos. El sistema está diseñado en tres niveles que es muy común cuando se desarrolla una aplicación cliente/ servidor. Estas tres capas son la vista donde se pueda apreciar la interfaz del software con todos sus componentes, la aplicación donde se encuentra la lógica y la persistencia.

Para la elaboración eficiente de horarios de sistemas se utiliza un algoritmo genético combinado con búsqueda tabú para mejorar la solución. En la **Figura 13** se puede observar una vista del software para la generación de horarios.

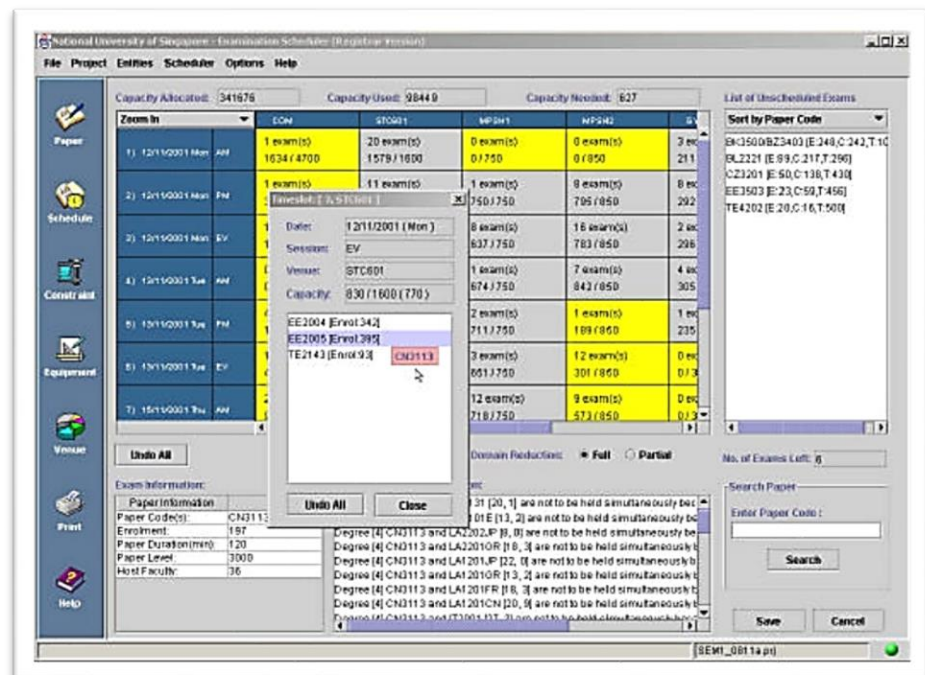


Figura 13. Pantalla del generador de horarios.

Fuente: de Imagen recuperada de American Association for Artificial Intelligence. Singapore. <https://www.aaai.org/Papers/AAAI/2002/AAAI02-125.pdf>

2.2. Bases teóricas

2.2.1. Algoritmo

Quispe (2007), cita en su artículo a distintos autores que definen el algoritmo.

- Secuencia finita de instrucciones, reglas o pasos que describen de forma precisa las operaciones de un ordenador que debe realizar para llevar a cabo una tarea en un tiempo más finito. (Donald E., 1968)
- Un algoritmo es un conjunto finito de pasos definidos, estructurados en el tiempo y formulados con base a un conjunto finito de reglas no ambiguas, que proveen un procedimiento para dar la solución o indicar la falta de esta a un problema en un tiempo determinado como se puede observar en la figura 01 (Quispe, 2007).

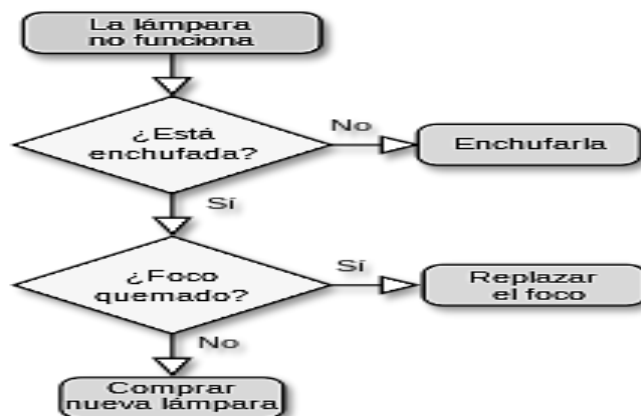


Figura 01. Diagrama de un algoritmo.

Fuente: Revista de algoritmos de Rodolfo Quispe-Otazu <http://www.rodolfoquispe.org/blog/que-es-un-algoritmo.php>

2.2.2. Problema NP

Según Ángeles (2015), publicó en su proyecto de tesis PUCP, que son aquellos problemas para los cuales existe un algoritmo determinístico los cuales pueden verificar si una solución es correcta en un tiempo polinomial. La resolución de un problema no determinístico consta de dos fases: Fase de suposición y fase de verificación (Alsuwaiyel, 1998).

2.2.2.1. La fase de suposición

Es aquella en la cual se genera una respuesta arbitraria encontrada mediante un algoritmo no determinístico en un número de pasos polinomiales. Esta respuesta puede ser o no una solución del problema.

❖ Algoritmo no determinístico

Un algoritmo no determinista es un algoritmo que con la misma entrada ofrece muchos posibles resultados. No se puede saber de antemano cuál será el resultado de la ejecución de un algoritmo no determinista.

2.2.2.2. La fase de verificación

Es en la cual se verifica mediante un algoritmo determinístico si la respuesta se encuentra en un formato apropiado y si la respuesta es una solución para la instancia de ese problema.

❖ Algoritmo determinístico.

Un algoritmo determinista es un algoritmo que, en términos informales, es completamente predictivo si se conocen sus entradas. Dicho de otra forma, si se conocen las entradas del algoritmo siempre producirá la misma salida, y la máquina interna pasará por la misma secuencia de estados (Redcientífica, 2015).

2.2.2.2.1. Problemas de optimización.

Un problema de optimización consiste en minimizar o maximizar el valor de una variable. En otras palabras, se trata de calcular o determinar el valor mínimo o el valor máximo de una función de una variable. Se debe tener presente que la variable que se desea minimizar o maximizar debe ser expresada como función de otra de las variables relacionadas en el problema. En ocasiones es preciso considerar las restricciones que se tengan en el problema, ya que éstas generan igualdades entre las variables que permiten la obtención de la función de una variable que se quiere minimizar o maximizar. Según el artículo Cálculo Diferencial e Integral publicado por (Canek, 2008).

2.2.2.2. Algoritmos Meta-heurísticos

Son métodos aproximados que se define formalmente como una estrategia de muy alto nivel para explorar espacios de búsqueda. Este proceso consiste en una generación iterativa, la cual tiene como objetivo encontrar de manera eficiente soluciones óptimas (Blum y Roli, 2003). Es importante mencionar que la exploración del espacio de búsqueda debe ser balanceada entre la diversificación y la intensificación.

Los algoritmos meta-heurísticos se pueden clasificar según las características que este presenta. A continuación, se mencionará los tipos de clasificación más importantes (Blum y Roli, 2003):

- ❖ Inspirados en la naturaleza vs lo no inspirados.
- ❖ Basado en poblaciones vs un solo punto de búsqueda.
- ❖ Función objetivo dinámica vs la estática.
- ❖ Una estructura de vecinos vs varias.
- ❖ Uso de mayor memoria vs a un uso menor.

2.2.2.2.3. Técnicas basadas en algoritmos de población

Son algoritmos meta-heurísticos que se basan en el comportamiento natural.

• Algoritmo genético

Esta técnica consiste básicamente en tener una solución inicial representada en un cromosoma e ir evaluando este cromosoma mediante los operadores de mutación y cruce para que en la evolución se mejore la solución. Este método es comúnmente usado en la resolución de muchos problemas de horarios. (Burke, 2006).

• Algoritmo meméticos

Este tipo de métodos combina la habilidad de la exploración de búsqueda de los algoritmos evolutivos con los métodos de búsquedas locales (EcuRed, 2015). Esta técnica presenta buenas soluciones, pero toma un alto tiempo computacional.

- **Algoritmo colonia de hormigas**

Esta técnica simula la búsqueda de los caminos más cortos que usan las hormigas cuando van a buscar su comida. Estas se guían dependiendo de la intensidad del olor de las feromonas que dejan en los caminos, en el más corto se intensifica el olor. De forma similar se realiza una analogía con los problemas de optimización combinatoria, donde cada hormiga se utiliza para construir una solución y las feromonas representan la información obtenida durante la búsqueda, la cual ayuda a generar soluciones en la siguiente etapa.

2.2.3. Conceptos de Timetabling

2.2.3.1. Problema Timetabling.

Timetabling en español es como un problema de programación horaria, consiste en asignar, de la mejor manera, una serie de eventos o recursos en periodos de tiempo denominados spots. Estos eventos se asignarán cumpliendo requisitos obligatorios o fuertes y se tratará de cumplir requisitos secundarios o débiles del problema, para agregar un alto grado de optimización y adaptabilidad a la solución. Sin embargo, es suficiente cumplir las restricciones más fuertes para aceptar una solución como válida (Aoron, 2012).

2.2.3.2. Solución al problema del Timetabling

2.2.3.2.1. Técnicas tradicionales.

Método que, debido a su forma de buscar, se les denomina métodos completos, ya que recorren todo el espacio de búsqueda, es decir, encuentra todas las soluciones posibles a un determinado problema, sin embargo, estos dependen del número de variables que influyen en el problema, es decir, el éxito de estas depende directamente del número de variables que intervienen en el problema.

2.2.3.2.1.1. Programación entera

Es un modelo que contiene restricciones y una función objetivo idénticas a las formuladas por planeación lineal. La única diferencia es que una o más de las variables de decisión tienen que tomar un valor entero en la solución final (Aoron, 2012).

2.2.3.2.1.2. Programación lineal.

La programación lineal es una técnica matemática relativamente reciente (siglo XX), que consiste en una serie de métodos y procedimientos que permiten resolver problemas de optimización en el ámbito, sobre todo, de las Ciencias Sociales (Aoron, 2012).

Nos centraremos en este tema en aquellos problemas simples de programación lineal, los que tienen solamente 2 variables, problemas bidimensionales.

Para sistemas de más variables, el procedimiento no es tan sencillo y se resuelven por el llamado método Simplex, dado por el matemático estadounidense (Danzig, 1951).

Los modelos matemáticos se dividen básicamente en Modelos Determinísticos (MD) o Modelos Estocásticos (ME) como se ve en la *Figura 02*. En el primer caso (MD) se considera que los parámetros asociados al modelo son conocidos con certeza absoluta, a diferencia de los Modelos Estocásticos, donde la totalidad o un subconjunto de los parámetros tienen una distribución de probabilidad asociada. Los cursos introductorios a la Investigación Operativa generalmente se enfocan sólo en Modelos Determinísticos.

2.2.3.2.1.3. Backtracking

Según el artículo (Programación, 2013) determina que el Backtracking o búsqueda atrás, es una técnica de programación para hacer búsqueda sistemática a través de todas las configuraciones posibles dentro de un espacio de búsqueda.



Figura 02. Programación lineal estructurado.

Fuente: Revista enfocado al área de la Investigación de Operaciones [http://www.Programación lineal.net/Programación_lineal.html](http://www.Programación%20lineal.net/Programación_lineal.html)

Se asemeja a un recorrido en profundidad dentro de un grafo dirigido. El grafo en cuestión suele ser un árbol como se muestra en la **Figura 03**, o por lo menos no contiene ciclos. Sea cual sea su estructura, existe sólo implícitamente. El objetivo del recorrido es encontrar soluciones para algún problema. Esto se consigue construyendo soluciones parciales a medida que progresa el recorrido; estas soluciones parciales limitan las regiones en las que puede encontrar una solución completa. El recorrido tiene éxito si, procediendo de esta forma, se puede definir por completo una solución. En este caso el algoritmo puede, o bien detenerse (si lo único que se necesita es una solución del problema) o bien seguir buscando soluciones alternativas (si deseamos examinarlas todas). Por otra parte, el recorrido no tiene éxito si en alguna etapa la solución parcial construida hasta el momento no se puede completar. En tal caso, el recorrido vuelve atrás exactamente igual que en un recorrido en profundidad, eliminando sobre la marcha los elementos que se

hubieran añadido en cada fase. Cuando vuelve a un nodo que tiene uno o más vecinos sin explorar, prosigue el recorrido de una solución.

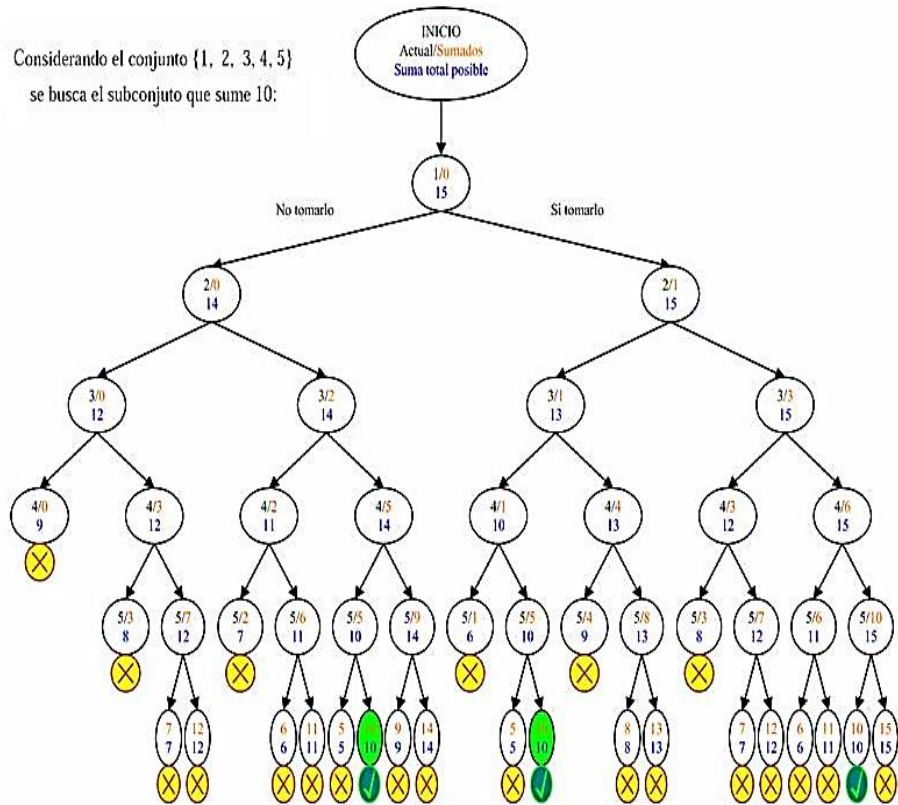


Figura 03. Programación Backtrackin -Lineal

Fuente: https://commons.wikimedia.org/wiki/File:Branch%26bound_low.jpg

2.2.3.2.2. Técnicas no tradicionales.

De lo contrario de las técnicas tradicionales, estas no encuentran todas las soluciones posibles en un problema, es decir acotan el espacio de búsqueda, por lo mismo caen en la categoría de método incompatibles se puede observar las técnicas de búsqueda la **Figura 05.**

2.2.3.2.2.1. Redes neuronales

Abramson (1991) publicó en un web científico de Inteligencia Artificial, sobre las Redes Neuronales como un campo muy importante dentro de la Inteligencia Artificial. Inspirándose en el

comportamiento conocido del cerebro humano (principalmente el referido a las neuronas y sus conexiones), trata de crear modelos artificiales que solucionen problemas difíciles de resolver mediante técnicas algorítmicas convencionales.

Las redes neuronales artificiales se basan en el circuito de procesamiento de entradas en el cual los pesos son sumados. Las funciones de peso serán llamadas desde ahora como atenuadores. En la implementación, las entradas a una neurona son pesadas multiplicando el valor de la entrada por un factor que es menor o igual a uno. El valor de los factores de peso es determinado por el algoritmo de aprendizaje.

2.2.3.2.2. Reconocido simulado

La simulación del proceso de recocido puede usarse para describir un proceso de generación de una sucesión de soluciones de un problema de optimización combinatoria, en donde se vayan obteniendo, conforme el proceso avanza, mejores soluciones al mismo. Para este propósito, se puede observar una analogía entre el sistema físico y un problema de optimización combinatoria en donde cada solución del problema puede verse como un estado del sólido y el valor de la función objetivo para el nivel de energía del sólido. Funciona así:

- El proceso asigna aulas procesando un grupo a la vez empezando con el primero.
- El proceso continúa hasta que todos los grupos son procesados.
- Cuando una solución es obtenida dividiendo el problema en conjuntos de horario, una mutación especial es generada con grupos traslapados. Esto podría generar una mala asignación de aula, para minimizar esto se utiliza el siguiente paso.
- Los grupos son procesados basados en el número de aulas, este número de aulas es optimizado y se termina con las horas de clase con el mínimo de paquetes a esa hora.

- La implementación del programa genera una base de datos con la nueva asignación de salones para el próximo semestre (Abramson, 1991).

2.2.3.2.2.3. Búsqueda Tabú

La búsqueda tabú es un método de optimización matemática, perteneciente a la clase de técnicas de búsqueda local. La búsqueda tabú aumenta el rendimiento del método de búsqueda local mediante el uso de estructuras de memoria: una vez que una potencial solución es determinada, se la marca como "tabú" de modo que el algoritmo no vuelva a visitar esa posible solución (Glover y Melian, 2003).

2.2.3.2.2.4. Algoritmo evolutivo

Los algoritmos evolutivos trabajan con una población de individuos, que representan soluciones candidatas a un problema. Esta población se somete a ciertas transformaciones y después a un proceso de selección, que favorece a los mejores. Cada ciclo de transformación y selección constituye una generación, de forma que después de cierto número de generaciones se espera que el mejor individuo de la población esté cerca de la solución buscada como se muestra en la **Figura 04**. Los algoritmos evolutivos combinan la búsqueda aleatoria, dada por las transformaciones de la población, con una búsqueda dirigida dada por la selección (Glover y Melian, 2003).

Principales Componentes:

- Población de individuos, que son una representación (no necesariamente directa) de posibles soluciones.
- Procedimiento de selección basado en la aptitud de los individuos para resolver el problema.

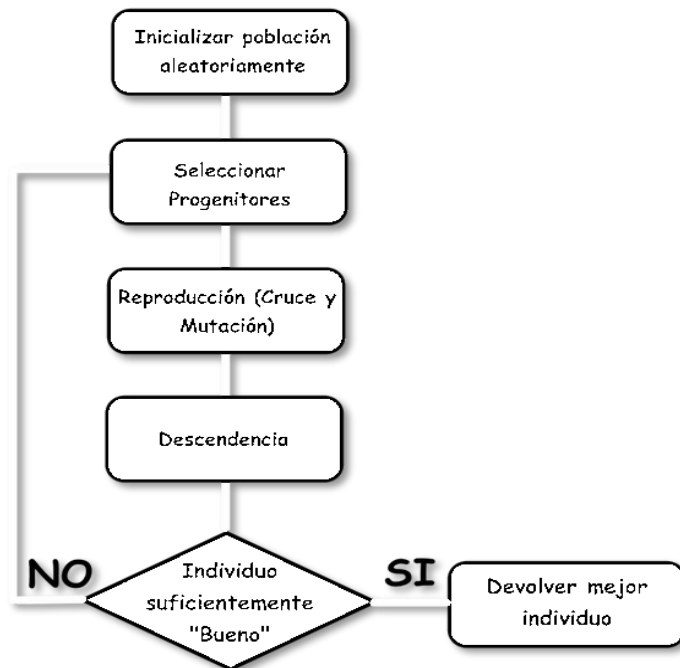


Figura 04. El esquema general de un algoritmo evolutivo

Fuente: Recuperado de: http://jarroba.com/wp-content/uploads/2013/05/Es-quema-general-de-un-AE-jarroba.com_.png

- **Estrategias evolutivas**

Las estrategias evolutivas son métodos computacionales que trabajan con una población de individuos que pertenecen al dominio.

Cada individuo de la población es un posible óptimo de la función objetivo; la representación de cada individuo de la población consta de 2 tipos de variables: las variables objeto y las variables estratégicas. Las variables objeto son los posibles valores que hacen que la función objetivo alcance el óptimo global y las variables estratégicas son los parámetros mediante los que se gobierna el proceso evolutivo o, en otras palabras, las variables estratégicas indican de qué manera las variables objeto son afectadas por la mutación.

- **Programación genética**

La programación evolutiva (PE) es una rama de la computación evolutiva. La programación evolutiva es prácticamente una variación de los algoritmos genéticos, donde lo que cambia es la representación de los individuos. En el caso de la PE los individuos son ternas (tripletas) cuyos valores representan estados de un autómata finito. Cada terna está formada por:

- ❖ El valor del estado actual;
- ❖ un símbolo del alfabeto utilizado;
- ❖ el valor del nuevo estado.

Estos valores se utilizan, como en un autómata finito, de la siguiente manera: Teniendo el valor del estado actual en el que nos encontramos, tomamos el valor del símbolo actual y si es el símbolo de nuestra terna, nos debemos mover al nuevo estado.

- **Algoritmo genético**

Los Algoritmos Genéticos (AGs) son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización. Están basados en el proceso genético de los organismos vivos. A lo largo de las generaciones, las poblaciones evolucionan en la naturaleza de acuerdo con los principios de la selección natural y la supervivencia de los más fuertes, postulados por Darwin (1859).

Por imitación de este proceso, los Algoritmos Genéticos son capaces de ir creando soluciones para problemas del mundo real. La evolución de dichas soluciones hacia valores óptimos del problema depende en buena medida de una adecuada codificación de las mismas.

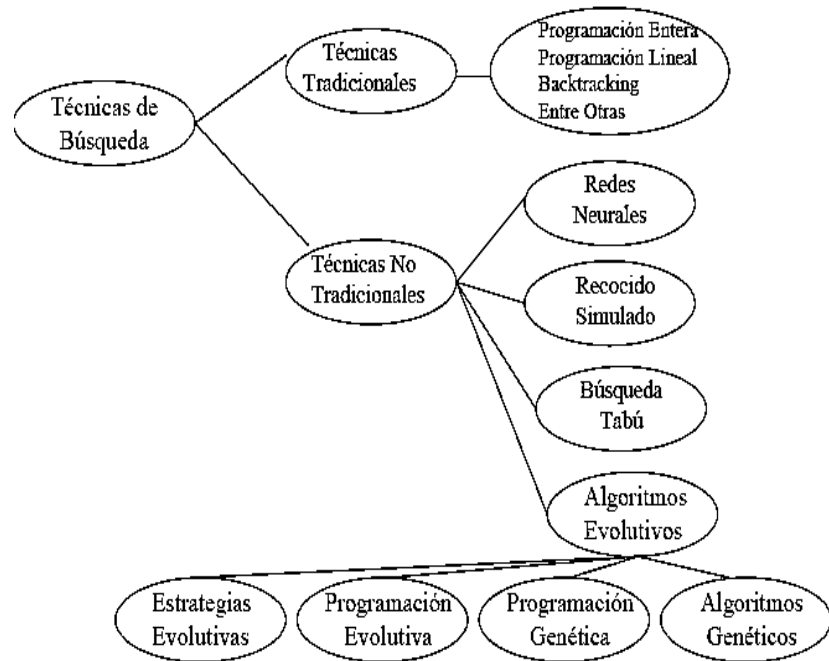


Figura 05. Clasificación de técnicas de búsqueda

Fuente: Tesis de “asignación de horarios con algoritmo evolutivo en la Universidad del Norte”

2.2.3.3. Algoritmo Genético

Pino, Martínez y Villanueva (2012) define que los Algoritmos Genéticos (AGs), son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización, es decir que son Algoritmos de búsqueda basados en los mecanismos de selección natural y genética natural. Combinan la supervivencia de los más compatibles entre las estructuras de cadenas, con una estructura de información ya aleatorizada, intercambiada para construir un algoritmo de búsqueda con algunas de las capacidades de innovación de la búsqueda humana se aclara la dedinición con la **Figura 06**.

El funcionamiento del algoritmo genético sigue el siguiente proceso:

1. El sistema genera varios centenares de individuos o soluciones posibles.
2. Los que mejor se ajustan a los requerimientos, obtienen más posibilidades de reproducirse.
3. Se procede al cruzamiento de individuos aleatoriamente

4. De acuerdo a una heurística se decide, si los individuos, además deben mutarse o no y así se obtiene una nueva generación de individuo o soluciones.
5. Nuevamente se evalúa a los que mejor se ajustan a los requerimientos y obtienen más posibilidades de reproducirse. Las peores son descartadas.
6. Continuación: los pasos 3 y 4 se reiteran docenas de miles de veces. Tras miles de generaciones, el mejor individuo-solución resuelve el problema de manera satisfactoria. No se puede aspirar al óptimo absoluto.

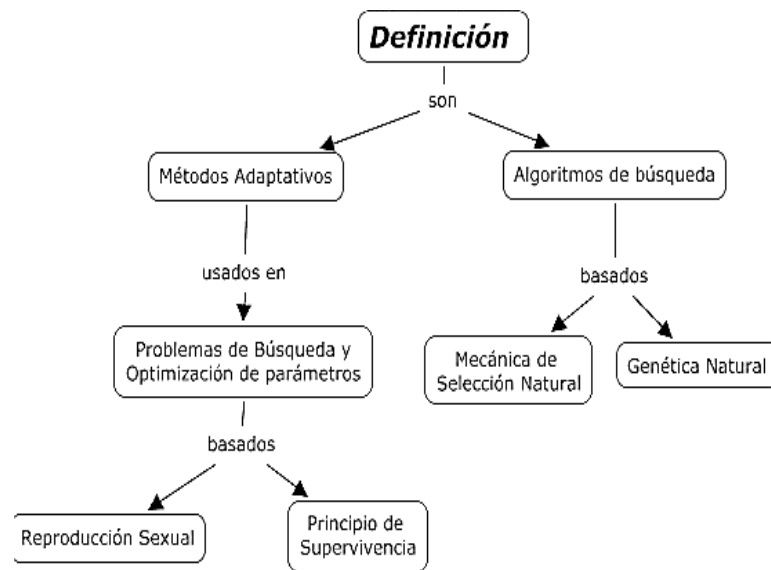


Figura 06. Definición de un algoritmo genético

Fuente: Recuperado <http://cmappublic2.ihmc.us/rid=1KNKCD8QT-1T75CLQ-1R4M/algoritmos%20geneticos.cmap>

2.2.3.4. Elementos fundamentales del horario genético.

2.2.3.4.1. Estructura del cromosoma.

Para el problema del horario, cada cromosoma representa un horario completo. Como ya se mencionó, el cromosoma está constituido por una cadena de genes y en este caso cada gen es una estructura compuesta al menos de cinco componentes se muestra en la **Figura 07**.

➤ Día

- Periodo
- Profesor
- Curso
- Aula

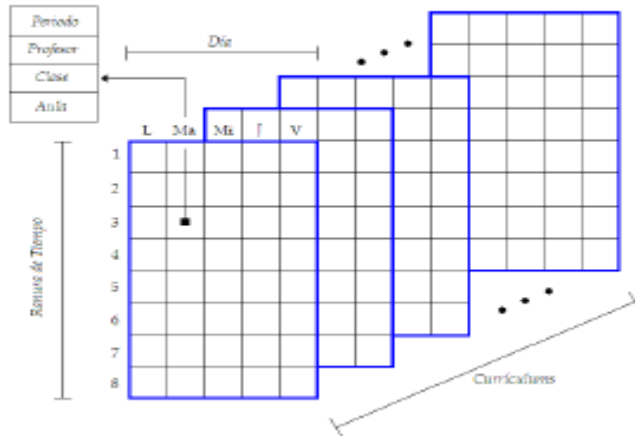


Figura: 07. Representación de un cromosoma para el problema del horario

Fuente: Desarrollado por el tesista

2.2.3.4.2. Estructura de la Población Inicial.

La población inicial está formada por un conjunto de horarios, generados de forma aleatoria, tratando de obtener diversidad y calidad en las posibles soluciones como se muestra en la **Figura 08**.

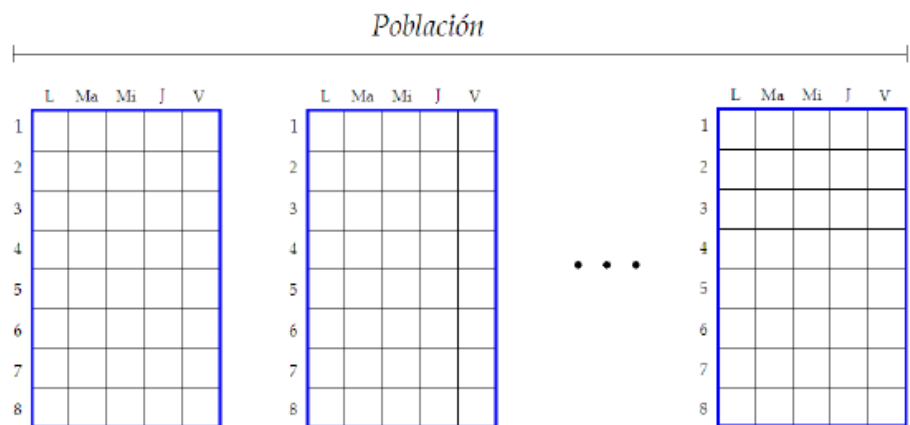


Figura 08. Ejemplo de población Inicial para horario

Fuente: Desarrollado por el tesista

2.2.3.4.3. Operadores genéticos

❖ Selección

Proceso que escoge los miembros de la población que serán utilizados en la reproducción (padres). Se eligen los más aptos. Existen varios métodos de selección la más conocida es la rueda de ruleta según Pino, Martínez y Villanueva (2012).

❖ Cruce

Indica la frecuencia con la que se producen cruces entre los cromosomas padre, es decir, que haya probabilidad de reproducción entre ellos. En caso de que no exista probabilidad de reproducción, los hijos serán copias exactas de los padres. En caso de haberla, los hijos tendrán partes de los cromosomas de los padres. Si la probabilidad de cruce es del 100% el hijo se crea totalmente por cruce, no por partes según Pino, Martínez y Villanueva (2012).

❖ Mutación

Nos indica la frecuencia con la que los genes de un cromosoma son mutados. Si no hay mutación, los descendientes son los mismos que había tras la reproducción. En caso de que haya mutaciones, parte del cromosoma descendiente es modificado y si la probabilidad de mutación es del 100%, la totalidad del cromosoma se cambia. En este caso, no se cambian simplemente unos bits del cromosoma, sino que se cambian todos, lo que significa que se produce una inversión en el cromosoma y no una mutación por lo que la población degenera muy rápidamente definido por Pino, Martínez y Villanueva (2012).

2.2.3.5. Algoritmo Genético simple.

Se muestra en la **Figura 09** la secuencia y estructura del algoritmo genético que se aplicara en el proyecto de generación de horario.

```

// Inicio Algoritmo Genetico
generarPoblacionInicial();
evaluarIndividuos();
for(int i = 0; i < ConfigGenetico.getLimiteGeneraciones(); i++){
    this.generacion++;
    reproducirIndividuos();//
    mutarIndividuos();
    evaluarIndividuos();
    imprimirMejorIndividuo();
}

```

Figura 09. Seudocódigo del Algoritmo Genético

Fuente: Tesis de generación de horarios con programación tradicional del instituto politécnico nacional (México, 2007).

2.2.3.6. Descripción del funcionamiento del algoritmo genético.

Es posible observar el esquema general del funcionamiento de un algoritmo genético a través de la ilustración en la **Figura 10**.

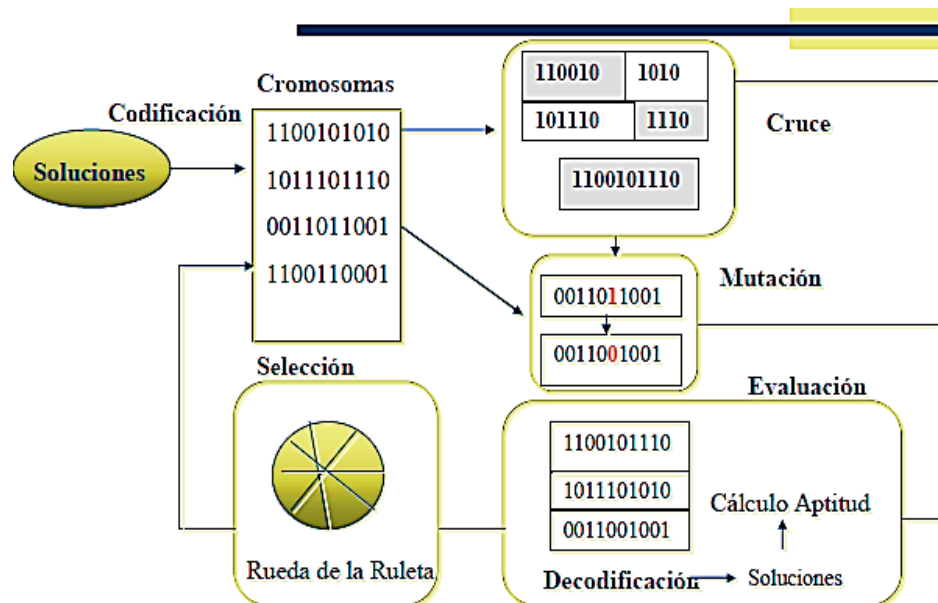


Figura 10. Esquema de un algoritmo genético simple.

Fuente: Artículo sobre evolución de algoritmo genético.

2.2.4. Estándares para desarrollo de proyecto

2.2.4.1. ISO/IEC 9126

La norma Internacional publicado por la IEEE (1992), es usada para la evaluación de la calidad de software, también conocido como ISO 9126 (o ISO/IEC 9126). Este estándar describe 6 características generales como se muestra en la **Figura 11**: Funcionalidad, Confiabilidad, Usabilidad, Eficiencia, Mantenibilidad, y Portabilidad.

La norma ISO/IEC 9126 permite especificar y evaluar la calidad del software desde diferentes criterios asociados con adquisición, requerimientos, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad y auditoria de software.



Figura 11. Composición de la calidad de medición del software.

Fuente: <http://actividadreconocimiento-301569-8.blogspot.pe/2013/03/norma-de-evaluacion-isoiec-9126.html>.

2.2.5. Metodología para gestión de proyecto.

Según el libro de Project Management Instituto (PMI, 201) de La guía del PMBOK, fundamenta como un estándar en la Administración de

proyectos, EL propósito de esta guía es la aplicación de conocimientos, procesos, habilidades, herramientas y técnicas. También proporciona y promueve un vocabulario común en el ámbito de la profesión de la dirección de proyectos y las buenas prácticas.

La Guía del PMBOK proporciona pautas para la dirección de proyectos tomados de forma individual. Define la dirección de proyectos y otros conceptos relacionados, y describe el ciclo de vida de la dirección de proyectos y los procesos conexos.

❖ **Agrupación de procesos.**

Los 5 grupos en los que la Guía PMBOK clasifica los procesos son:

1. **Inicialización:** Aquellos procesos aplicados para la definición de un proyecto nuevo, o una nueva fase de un proyecto existente, para la autorización de su inicio.
2. **Planeación:** Aquellos procesos requeridos para establecer el alcance del proyecto, definiendo objetivos y un curso de acción para alcanzar los objetivos del mismo.
3. **Ejecución:** Aquellos procesos aplicados para completar el trabajo definido, satisfaciendo las especificaciones del mismo.
4. **Monitoreo y control:** Aquellos procesos que siguen la trayectoria, revisan y regulan el progreso y el rendimiento del proyecto; identifican áreas de cambio requeridas en el plan, e inician dichos cambios.
5. **Cierre:** Aquellos procesos aplicados para finalizar todas las actividades a través de los grupos. Cierran formalmente el proyecto o fase.

❖ **Áreas de conocimiento.**

Cada una de las áreas de conocimiento comprende los procesos requeridos para lograr una efectiva gestión del proyecto. Las 10 áreas de conocimiento son las siguientes:

1. **Integración:** Incluye los procesos y actividades requeridos para identificar, definir, combinar, unificar y coordinar los mismo a realizar por los grupos de trabajo.
2. **Alcance:** Incluye los procesos requeridos para asegurar la realización de todo el trabajo a aplicar en el proyecto, y no solo realizar aquellos que completen el proyecto.
3. **Tiempo:** Incluye los procesos requeridos para la correcta administración de tiempo.
4. **Administración de costos:** Incluye los procesos involucrados en la planeación, estimación, presupuesto, financiamiento, costeo, administración y control de costos; con el objetivo de que el proyecto sea realizado con un presupuesto apropiado.
5. **Calidad:** Incluye los procesos y actividades involucrado en el rendimiento de organización, que define la política de calidad, objetivos y responsabilidades para que el proyecto satisfaga las necesidades por las que se hizo.
6. **Recursos humanos:** Incluye los procesos que organizan, administran y dirigen al equipo de trabajo.
7. **Comunicación:** Incluye los procesos requeridos para asegurar en tiempo y forma la planeación, recolección, creación, distribución, almacenaje, recuperación, administración, control, monitoreo y disposición de la información del proyecto.
8. **Riesgos:** Incluye los procesos que planean, identifican, analizan, y controlan los posibles o actuales riesgos del proyecto.
9. **Adquisición:** Incluye todos los procesos necesarios para la adquisición y compra de productos, bienes, servicios o resultados requeridos del exterior por el equipo de trabajo.
10. **Interesados:** Incluye todos los procesos requeridos para identificar los grupos u organización que impacta el proyecto; analizando sus expectativas y desarrollar las estrategias necesarias para impactar positivamente en la ejecución y decisiones de los interesados.

2.2.6. Metodología de desarrollo XP (Programig Express).

La programación Extrema es una metodología ligera de desarrollo de software que se basa en la simplicidad, la comunicación y la realimentación reutilización del código desarrollado y centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, su objetivo principal es aumentar la productividad al desarrollar el software.

2.2.7. Herramientas utilizadas en el desarrollo de software.

2.2.7.1. Java.

Según la página oficial de Java (2015), es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes.

Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma. Eso quiere decir que si hacemos un programa en Java podrá funcionar en cualquier ordenador del mercado. Es una ventaja significativa para los desarrolladores de software, pues antes tenían que hacer un programa para cada sistema operativo, por ejemplo, Windows, Linux, Apple, etc. Esto lo consigue porque se ha creado una Máquina de Java para cada sistema que hace de puente entre el sistema operativo y el programa de Java y posibilita que este último se entienda perfectamente.

La independencia de plataforma es una de las razones por las que Java es interesante para Internet, ya que muchas personas deben tener acceso con ordenadores distintos. Pero no se queda ahí, Java está desarrollándose incluso para distintos tipos de dispositivos

además del ordenador como móviles, agendas y en general para cualquier cosa que se le ocurra a la industria.

❖ **Pasado y presente**

Java fue pensado originalmente para utilizarse en cualquier tipo de electrodoméstico, pero la idea fracasó. Uno de los fundadores de Sun rescató la idea para utilizarla en el ámbito de Internet y convirtieron a Java en un lenguaje potente, seguro y universal gracias a que lo puede utilizar todo el mundo y es gratuito. Una de los primeros triunfos de Java fue que se integró en el navegador Netscape y permitía ejecutar programas dentro de una página web, hasta entonces impensable con el HTML.

Actualmente Java se utiliza en un amplio abanico de posibilidades y casi cualquier cosa que se puede hacer en cualquier lenguaje se puede hacer también en Java y muchas veces con grandes ventajas. Para lo que nos interesa a nosotros, con Java podemos programar páginas web dinámicas, con accesos a bases de datos, utilizando XML, con cualquier tipo de conexión de red entre cualquier sistema. En general, cualquier aplicación que deseemos hacer con acceso a través web se puede hacer utilizando Java.

2.2.7.2. Hibernate

King y Bauer (2010) afirma que Hibernate es una herramienta de Mapeo objeto-relacional (ORM) para la plataforma Java y disponible también para .Net con el nombre de NHibernat que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans de las entidades que permiten establecer estas relaciones. Hibernate es software libre, distribuido bajo los términos de la licencia GNU LGPL.

En otras palabras, Hibernate es un Framework que agiliza la relación entre la aplicación y la base de datos como se muestra en la **Figura 12**. Para poder aprender a utilizarlo es necesario contar con los conocimientos básicos de base de datos y SQL, así como manejar el lenguaje Java.

Hibernate busca solucionar el problema de la diferencia entre los dos modelos de datos coexistentes en una aplicación, el usado en la memoria de la computadora que es el orientado a objetos y el usado en las bases de datos (modelo relacional). Para lograr esto permite al desarrollador detallar cómo es su modelo de datos, qué relaciones existen y qué forma tienen. Con esta información.

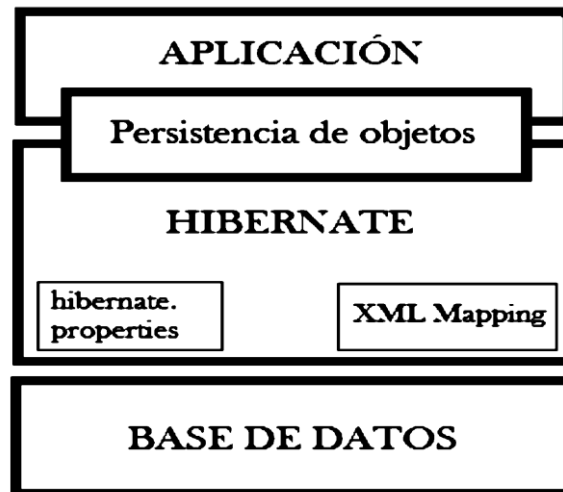


Figura 12: Arquitectura de Hibernate.

Fuente: www.emmersonmiranda.blogspot.com.

2.2.7.3. Netbeans

La página oficial de (NetBeans, 2015) menciona que es un proyecto exitoso de código abierto con una gran base de usuarios, una comunidad en constante crecimiento. Sun Microsystems fundó el proyecto de código abierto NetBeans en el año 2000 y continúa siendo el patrocinador principal de los proyectos.

Existen dos productos el NetBeans IDE y NetBeans Platform.

❖ NetBeans IDE

Es un entorno de desarrollo - una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java - pero puede servir para

cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el NetBeans IDE. NetBeans IDE es un producto libre y gratuito sin restricciones de uso.

❖ **NetBeans Platform.**

Una base modular y extensible usada como estructura de integración para crear grandes aplicaciones de escritorio. Empresas independientes asociadas, especializadas en desarrollo de software, proporcionan extensiones adicionales que se integran fácilmente en la plataforma y que pueden también utilizarse para desarrollar sus propias herramientas y soluciones. Ambos productos son de código abierto y gratuito para uso tanto comercial como no comercial. El código fuente está disponible para su reutilización de acuerdo con la Common Development and Distribution License.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

2.2.7.4. Librería JGAP.

Según la página oficial (javaHispano, 2015) publica que JGAP son las siglas de java Genetic Algorithms Package (paquete de algoritmos genéticos para java). Es un componente de programación de algoritmos genéticos que se utiliza como un framework y es un software libre y se puede distribuir bajo la GNU Lesser Public License 2.1 o posterior.

- JGAP tiene clases e interfaces para representar.
- Genes (gene).

- Cromosomas (Chromosome).
- Individuos (Chromosome).
- La población (Genotype).
- La fusión de ajuste (FitnessFucion).
- Operadores genéticos.

2.2.7.5. Base de datos.

Pinto (2011) especifica a una base de datos como un conjunto o depósito de datos estructurados que se encuentran almacenados en memoria no volátil, organizados independientemente de su utilización y su implementación se encuentra en máquinas accesibles en tiempo real que pueden ser manipuladas por un conjunto de aplicaciones y usuarios se muestra en la **Figura 14**.

Ventajas de la base de datos:

- ❖ Evita la redundancia, la inconsistencia y los problemas de integridad de los datos.
- ❖ Mejora en la disponibilidad de los datos.
- ❖ Permite la administración de la seguridad de los datos, tanto en acceso como en operaciones.
- ❖ Independencia de los datos y el tratamiento.
- ❖ Menos costo de mantenimiento, puesto que el cambio de datos no implica el cambio en el programa o viceversa.

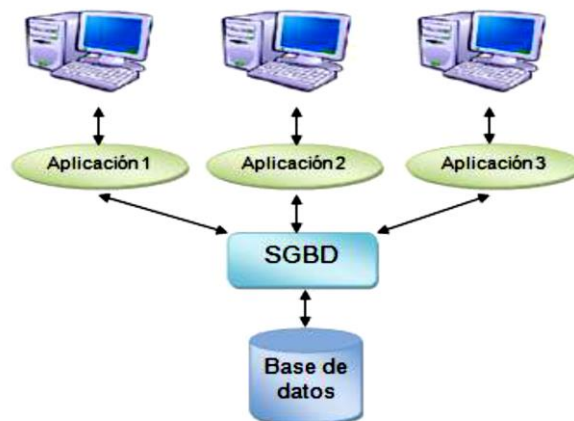


Figura 14. Arquitectura de sistema de una base de datos (2015).

Fuente: Recuperado de

<http://nethumans.com/solutions/development/Database.aspx>

A continuación, se listan los diferentes gestores de base de datos que existen en el mercado:

- ❖ Access.
- ❖ Oracle.
- ❖ MySQL.
- ❖ SQL Server.
- ❖ Mysql Workbech.

Dentro de las aplicaciones antes mencionadas se ha seleccionado MySQL como gestor de base de datos para almacenar todos los datos necesarios para la generación del sistema de horarios.

2.2.7.6.Mysql.

Según página oficial de (Oracle, 2015) determina que MySQL es el servidor de bases de datos relacionales más popular, desarrollado y proporcionado por MySQL AB. MySQL AB es una empresa cuyo negocio consiste en proporcionar servicios en torno al servidor de bases de datos MySQL.

Una base de datos es una colección estructurada de datos. La información que puede almacenar una base de datos puede ser tan simple como la de una agenda, un contador, o un libro de visitas, ó tan vasta como la de una tienda en línea, un sistema de noticias, un portal, o la información generada en una red corporativa. Para agregar, acceder, y procesar los datos almacenados en una base de datos, se necesita un sistema de administración de bases de datos, tal como MySQL.

Una base de datos relacional almacena los datos en tablas separadas en lugar de poner todos los datos en un solo lugar. Esto agrega velocidad y flexibilidad. Las tablas son enlazadas al definir relaciones que hacen posible combinar datos de varias tablas cuando se necesitan consultar datos. La parte SQL de "MySQL" significa "Lenguaje Estructurado de Consulta", y es el lenguaje más usado y estandarizado para acceso a bases de datos relacionales.

¿Por qué usar MySQL?

El servidor de bases de datos MySQL es muy rápido, seguro, y fácil de usar. Si eso es lo que se está buscando, se le debe dar una oportunidad a MySQL. Se pueden encontrar comparaciones de desempeño con algunos otros manejadores de bases de datos en la página de MySQL.

El servidor MySQL fue desarrollado originalmente para manejar grandes bases de datos mucho más rápido que las soluciones existentes y ha estado siendo usado exitosamente en ambientes de producción sumamente exigentes por varios años. Aunque se encuentra en desarrollo constante, el servidor MySQL ofrece hoy un conjunto rico y útil de funciones. Su conectividad, velocidad, y seguridad hacen de MySQL un servidor bastante apropiado para acceder a bases de datos en Internet.

2.3. Definición de términos básicos para creación de un sistema de horario.

❖ Optimización.

Un proceso que conduce a la solución óptima de un problema. Cuando se habla de un problema de optimización se tiene en cuenta tres elementos: a) conjunto de posibles soluciones o soluciones factibles, b) una estrategia de búsqueda y c) un criterio para discriminar y determinar cuál de todas las posibles soluciones es la mejor según la Enciclopedia de economía (2009).

❖ Aula.

El aula es el espacio donde se desarrolla el proceso de enseñanza aprendizaje formal, independientemente del nivel académico o de los conocimientos impartidos en cada uno de ellos según la Enciclopedia de economía (2009).

❖ Laboratorio.

Un laboratorio es un lugar físico que se encuentra especialmente equipado con diversos instrumentos y elementos de medida o equipo, en orden a satisfacer las demandas y necesidades de experimentos o investigaciones diversas según Enciclopedia de economía (2009).

❖ **Restricciones.**

Una limitación o una reducción ya sean naturales o impuestas, según corresponda según Enciclopedia de economía (2009).

❖ **Materia.**

Es el nombre que se le da a la asignatura impartida en un curso y que forma parte de un programa de estudios según Enciclopedia de economía (2009).

❖ **Horario de clases .**

Es un cuadro en donde se indica la distribución de las horas y días, materias, profesores, paralelos, aulas y laboratorios para el caso de la UNAJMA.

CAPITULO 3 MATERIALES Y METODOLOGÍA

3.1. Gestión de proyecto - Guía de PMBOK.

La tesis tecnológica se desarrolló bajo la guía del PMBOK, para lograr los objetivos planeados y la calidad del producto, la guía del PMBOK presenta las buenas prácticas en las 10 áreas de conocimiento de gestión de proyecto, estas áreas fueron aplicados en las 5 fases o etapas (Inicio, Planificación, ejecución, control y Cierre), de la guía, a continuación, se describirá cada área de conocimiento en gestión de proyectos.

3.1.1. Gestión de Integración.

En la **Tabla 07** se incluye todo los procesos y actividades requeridas para realizar un buen proyecto.

Tabla 07:

Área de gestión de alcance del proyecto.

PROYECTO	Desarrollo de un sistema de generación de horarios académicos con algoritmos genéticos para la Escuela Profesional Ingeniería de Sistemas de la Universidad Nacional José María Arguedas en el año 2015		
PATROCINADOR	Percy Alhuay Carrasco – Jefe del Proyecto		
PREPARADO Por	Percy Alhuay Carrasco – Jefe del Proyecto	FECHA	20/08/2015
REVISADO POR	Percy Alhuay Carrasco – Jefe del Proyecto	FECHA	22/08/2015
Aprobado por	Percy Alhuay Carrasco – Jefe del Proyecto	FECHA	28/08/2015
REVISIÓN	DESCRIPCIÓN		FECHA
	1	Preparación de acta de constitución (Percy Alhuay Carrasco)	28/08/2015
BREVE DESCRIPCIÓN DEL PRODUCTO O SERVICIO DEL PROYECTO			

<p>La aplicación es desarrollada con algoritmo genético, da la mejor solución en menor tiempo posible, se registra todos los datos posibles como de (Docentes, Salón, Curso, Usuarios, Carga Académica, entre otros), para generar para dar la mejor solución.</p>	
<p>ALINEAMIENTO DEL PROYECTO</p>	
<p>OBJETIVOS ESTRATÉGICOS DE LA ORGANIZACIÓN</p>	<p>PROPÓSITO DEL PROYECTO</p>
<p>Desarrollar talleres de capacitación en el uso y manejo del software en la Universidad en la facultad de Ingeniería de sistemas</p>	<p>Promover el funcionamiento del sistema con algoritmos genéticos en la Universidad</p>
<p>OBJETIVOS DEL PROYECTO</p>	
<p>a. Analizar el proceso actual de generación de horarios de la EPIS.</p>	
<p>b. Diseñar y construir un sistema de generación de horarios académicos con algoritmos genéticos para la Escuela Profesional Ingeniería de Sistemas de la Universidad Nacional José María Arguedas en el año 2015.</p>	
<p>c. Evaluar la eficiencia del algoritmo genético en los resultados de generación del horario en la Escuela Profesional de Ingeniería de Sistemas de la Universidad Nacional José María Arguedas en el año 2015.</p>	
<p>FACTORES CRÍTICOS DE ÉXITO DEL PROYECTO</p>	
<p><input type="checkbox"/> Proceso de adquisición de equipamiento y software dentro de los plazos establecidos y las características requeridas.</p> <p><input type="checkbox"/> Diseño de software acorde a los requerimientos de la parte interesada (stakeholder).</p>	
<p>REQUERIMIENTOS DE ALTO NIVEL</p>	
<p><input type="checkbox"/> La Aplicación debe ser construida exactamente como lo plantean los requisitos funcionales con algoritmos genéticos.</p> <p><input type="checkbox"/> La aplicación debe contar con el módulo de administración de usuarios.</p>	
<p>EXTENSIÓN Y ALCANCE DEL PROYECTO</p>	
<p>Fases del proyecto</p>	<p>Principales entregables</p>
<p>Fase I – Gestión del Proyecto - Planificación</p>	<p>Acta de constitución del proyecto. Identificación de los stakeholder.</p>
<p>Fase II – Análisis y Diseño</p>	<p>Definición de los casos de usos</p>
	<p>Definición de los diagramas en starUML</p>

	Definición de Entidad Relacional
	Definición de las interfaces.
	Metáfora del sistema
	Tarjetas CRM
Fase III – Requerimiento de Adquisiciones	Adquisición de laptop
	Adquisición de entorno de desarrollo , xampp
	MySQL-workbench-community
	Ganttproject
	NetBeans IDE
Fase IV – Desarrollo	Desarrollo de los módulos del sistema de generación de horarios
Fase V – Pruebas	Pruebas de la aplicación
Fase VI - Implementación	Aplicación totalmente construida.
INTERESADOS CLAVE	
· Docentes de la Facultad de ingeniería de sistemas.	
· Alumnos de la facultad de ingeniería de sistemas.	
· Analista del sistema	
· Programador	
RIESGOS	
<input type="checkbox"/>	Demora en la adquisición de las herramientas físicas y lógicas necesarias.
<input type="checkbox"/>	Poca información de datos en cuanto a algoritmos genéticos.
<input type="checkbox"/>	No contar con los recursos económicos suficientes.
<input type="checkbox"/>	Retraso en la programación
Hitos principales del proyecto	
<input type="checkbox"/>	Aprobación del acta constitución del proyecto.
<input type="checkbox"/>	Alcanzar todos los requerimientos funcionales.
<input type="checkbox"/>	Pruebas y correcciones de la aplicación.
PRESUPUESTO DEL PROYECTO	
El costo del proyecto que haciende a la suma de S/. 15548.5 será asumido en un 100 % por el jefe del proyecto.	

Nota: Desarrollado por el tesista.

3.1.2. Gestión del Alcance.

El objetivo de este proyecto es reducir el tiempo de generación de horarios académicos, reducir los costos en la utilización de materiales y reducción del personal en cuanto a esta actividad, se detalla en la siguiente **Tabla 08**, el alcance del proyecto de acuerdo al guía de PMBOK.

Tabla 08:

Área de gestión de alcance del proyecto.

DESCRIPCIÓN DEL ALCANCE DEL PRODUCTO	
<p>REQUISITOS: CONDICIONES O CAPACIDADES QUE DEBE POSEER O SATISFACER EL PRODUCTO PARA CUMPLIR CON CONTRATOS, NORMAS, ESPECIFICACIONES, U OTROS DOCUMENTOS FORMALMENTE IMPUESTOS.</p>	CARACTERÍSTICAS:
<p>1. El Sistema de generación de horarios para la EPIS, permitirá en este módulo que el personal encargado solo gestiones las actividades.</p> <p>Búsqueda del profesor</p> <ul style="list-style-type: none"> • Ingresar nombre • Ingresar los apellidos • Ingresar Email. • Seleccionar unidad académica • Seleccionar dedicación del docente • Seleccionar estado del docente <p>Nuevo profesor</p> <ul style="list-style-type: none"> • Datos personales • Datos laborales • Clases (horas y curso) <p>Detalle del profesor</p> <ul style="list-style-type: none"> • Información personal • Horas disponibles • Cursos disponibles 	<p>1. Módulo de Mantenimiento de Profesores</p>
<p>2. El sistema tendrá un módulo de salones, se menciona a continuación los alcances de software en este módulo. La vista principal será la pantalla de búsqueda.</p> <p>Filtro de búsqueda</p> <ul style="list-style-type: none"> • Aula • Tipo de salón • Estado • Capacidad • Pabellón 	<p>2. Módulo de Mantenimiento de Salones</p>

<p>Nueva aula</p> <ul style="list-style-type: none"> • Datos del salón • Información – Ayuda. <p>Detalle de aula</p> <ul style="list-style-type: none"> • Información del salón • Información del tipo de salón 	
<p>3. El sistema tendrá un módulo de Cursos, se menciona a continuación los alcances de software en este módulo. La vista principal será la pantalla de búsqueda del curso.</p> <p>Búsqueda de mantenimiento del curso</p> <ul style="list-style-type: none"> • Filtro de búsqueda <ul style="list-style-type: none"> ✓ Clave del curso ✓ Nombre del curso ✓ Unidad académica ✓ Especialidad ✓ Estado • Resultados de búsqueda <p>Nuevo curso</p> <ul style="list-style-type: none"> • Datos del curso <ul style="list-style-type: none"> ✓ Especialidad del curso ✓ Clave del curso ✓ Nombre del curso • Composición <ul style="list-style-type: none"> ✓ Clases ✓ Laboratorio ✓ Practica <p>Detalle del curso</p> <ul style="list-style-type: none"> • Información del curso • Composición del curso 	<p>3. Módulo de Mantenimiento de Cursos</p>
<p>4. El sistema tendrá un módulo de horarios, se menciona a continuación los alcances de software en este módulo. La vista principal será la pantalla de búsqueda del horario, resultado de los algoritmos.</p> <p>Mantenimiento de búsqueda de horario</p> <ul style="list-style-type: none"> • Grupo de horarios • Descripción • Fecha de creación • Asignación • Estado <p>Nuevo horario</p> <ul style="list-style-type: none"> • Datos del algoritmo genético <p>Visualización del horario</p> <ul style="list-style-type: none"> • Horarios de clases • Ciclo curricular 	<p>4. Módulo de Mantenimiento de Horarios</p>

Fuente: Desarrollado por el tesista.

3.1.3. Gestión del Tiempo.

Todas las actividades de gestión de tiempo se llevarán en MS PROJECT 2013, en el programa se detallan las actividades, secuencia profesional involucrado, duración, recursos a utilizar y desarrollo de cada actividad, de acuerdo a la metodología XP se muestra en la **Tabla 09** las actividades.

Tabla 09:

Gestión de tiempo del proyecto.

CRONOGRAMA DE HITOS DEL PROYECTO.	
HITO O EVENTO SIGNIFICATIVO	FECHA PROGRAMADA
Inicio del proyecto.	10 de agosto del 2015
2. Planificación.- Se realiza la captura de todos los requerimientos del sistema.	10 de agosto del 2015 a 21 de septiembre del 2015
3. Análisis. Esta fase incluye la definición de problemas y los objetivos.	22 de septiembre del 2015 a 08 de diciembre del 2015
4. Diseño Se realizará la parte de diseño de diagramas de acuerdo a metodología XP.	22 de septiembre del 2015 a 08 de diciembre del 2015
5. Construcción y prueba En esta fase encontraremos los posibles errores y aseguramiento de que la funcionalidad del sistema esté acorde a lo inicialmente planeado.	16 de septiembre del 2015 hasta 13 de noviembre del 2015
6. implantación del sistema. En esta fase se realiza la instalación y conexión con el servidor.	09 de diciembre del 2016 a 06 de mayo del 2016
7. Documentación.	06 de mayo del 2016 al 27 de junio del 2016
Fin del proyecto	27 de junio a 27 de junio del mismo año

Fuente: Desarrollado por el tesista

3.1.4. Gestión de costes.

El proyecto de desarrollo de software, tendrá cuatro fases (Planeación, Diseño, Codificación, Prueba), de acuerdo a la metodología XP, los costos se determinan en cada fase se puede observar en la **Tabla 10**.

Tabla 10:
Área de gestión de costos.

		GASTO POR FASES	
METODOLOGIA XP	1ª Fase: Planificación del proyecto.		Costo total
	o Historias de usuario.	La primera fase tendrá un costo total de todas las actividades en general	S/. 4,500.00
	o Release planning.		
	o Velocidad del proyecto.		
	o Programación en pareja.		
	o Reuniones diarias.		
	• 2ª Fase: Diseño.		Costo total
	o Diseños simples.	La segunda fase tendrá un costo total de todas las actividades en general	S/. 1,800.00
	o Metaforas de Sistema		
	o Funcionalidad extra.		
	o Tarjetas C.R.C.		
	• 3ª Fase: Codificación.		Costo total
	o Clientes siempre presentes	La tercera fase tendrá un costo total de todas las actividades en general	S/. 5,200.00
	o El código se escribe siguiendo estándares		
	• 4ª Fase: Pruebas.		Costo total
	o Prueba de unitario	La cuarta fase tendrá un costo total de todas las actividades en general	S/. 2,635.00
o Prueba de aceptacion			
o pueba de errores			
		GASTOS GENRALES	S/. 1,413.50
		TOTAL DE INVERSION	S/. 15,548.50

Nota: Desarrollado por el tesista.

3.1.5. Gestión de calidad.

Se basará el proyecto tecnológico en los lineamientos del estándar ISO/IEC 9126 de la IEEE, para medir la calidad de sistema de generación de horarios con algoritmos genético. Los lineamientos que se debe tomar en cuenta a detalle son: funcionabilidad, confiabilidad, usabilidad, eficiencia, Mantenibilidad y portabilidad y se muestra en la **Tabla 11** gestión de calidad del proyecto.

Tabla 11:
Área de gestión de alcance del proyecto.

NOMBRE DEL PROYECTO	Desarrollo de un sistema de generación de horarios académicos con algoritmos genéticos para la Escuela Profesional Ingeniería de Sistemas de la Universidad Nacional José María Arguedas en el año 2015
----------------------------	---

PREPARADO POR	Percy Alhuay Carrasco – Jefe del Proyecto
FECHA	15/10/2015
GESTIÓN DE LA CALIDAD DEL PROYECTO	
<p>En este proyecto se utilizará la ISO/IEC 9126 que es un estándar internacional para la evaluación de Software. El estándar está dividido en cuatro partes las cuales dirigen, respectivamente, lo siguiente: modelo de calidad, métricas externas, métricas internas y calidad en las métricas de uso.</p>	
<p>También se hará uso del estándar ISO/IEC 12207, que a través de la Norma técnica peruana proporciona un marco de referencia común para los procesos de ciclo de vida del software. Este estándar se concibió para aquellos interesados en adquisición de software, así como desarrolladores y proveedores. Este marco de referencia cubre el ciclo de vida del software desde la conceptualización de las ideas (recopilación de requisitos) hasta la culminación del software, lo cual permite una buena calidad del software. Además cubre el control y la mejora de estos procesos.</p>	
PLANIFICAR LA CALIDAD	
(Explicar y sustentar como se ha elaborado el presente plan de gestión de calidad)	
<p>El plan de gestión de calidad describe como el equipo de dirección del proyecto implementara la política de calidad de la organización ejecutante. Es un componente o un plan subsidiario del plan para la dirección del proyecto.</p>	
<p>El plan de gestión de la calidad proporciona entradas al plan general para la dirección del proyecto y aborda el control de calidad, el aseguramiento de la calidad y métodos de mejora continua de los procesos del proyecto. (PMBOK 2008).</p>	
REALIZAR ASEGURAMIENTO DE LA CALIDAD	
(Explicar cómo se va a realizar el proceso de aseguramiento de calidad)	
<p>Aseguramiento de calidad. Es responsable el jefe del proyecto ejecutar el aseguramiento de la calidad durante todo el proyecto, revisa el planeamiento de los procesos del proyecto contra lo ejecutado, planea acciones preventivas o correctivas según sean necesario. Se informa semanalmente al equipo de trabajo.</p>	

Nota: Desarrollado por el tesista.

3.1.6. Gestión de recursos humanos.

En esta área de conocimiento se visualizará la explicación en el siguiente **Tabla 12** sobre la administración del equipo de trabajo.

Tabla 12:
Área de gestión de recursos humanos.

GASTO POR FASES			
METODOLOGIA XP	1ª Fase: Planificación del proyecto.		Costo total
	o Asesor	S/. 700.00	S/. 1,800.00
	o Analista	S/. 1,100.00	
	• 2ª Fase: Diseño.		Costo total
	o Asesor	S/. 600.00	S/. 1,700.00
o Analista y Diseñador	S/. 1,100.00	
	• 3ª Fase: Codificación.		Costo total
	o Asesor	S/. 600.00	S/. 2,600.00
	o Programador	S/. 2,000.00	
	• 4ª Fase: Pruebas.		Costo total
	o Asesor	S/. 600.00	S/. 1,600.00
	o Tester	S/. 1,000.00	
	TOTAL DE INVERSION		S/. 7,700.00

Nota: Desarrollado por el tesista.

3.1.7. Gestión de comunicación.

El proyecto tiene una duración de un año días, de lunes a viernes, por tal razón es necesario que se lleve las sesiones y reuniones para culminar el proyecto tecnológico durante el periodo establecido.

- Reuniones con el asesor se llevan en el siguiente horario

Martes: 09:00 - 11:00 y de 11:00 - 13:00

Jueves: 15:00 - 17:00 y de 17:00 - 19:00

- Sesiones de conocimiento los días sábados de:

Sábados: 08:00 - 12:00

- Sesiones de desarrollo del producto por el equipo del tesista, se realiza:

Lunes a viernes de 1.5 a 2 horas diarias.

3.1.8. Gestión de riesgos.

Los problemas potenciales que pueden ocurrir en el proceso de desarrollo del proyecto tecnológico son en presupuesto, recursos humanos.

Se está trabajando con la metodología XP, en esta metodología por lo tanto presenta tres interacciones, esto hace que reduzca los riesgos en un menor porcentaje, de acuerdo a la guía de PMBOK se puede identificar los riesgos a través de (impacto e influencia) en el proyecto, esto se puede observar en el siguiente **Tabla 13**.

Tabla 13:
Área de gestión de riesgo.

		IMPACTO SOBRE EL PROYECTO	
		BAJO	ALTO
INFLUENCIA SOBRE EL PROYECTO	ALTA	Tesista <ul style="list-style-type: none"> Bach. Percy Alhuay Carrasco. Equipo del proyecto. <ul style="list-style-type: none"> Desarrollador Analista y Diseñador Asesor 	<ul style="list-style-type: none"> Presidente de la Comisión Organizadora Vicepresidente Académico Vicepresidente Administrativo Departamento académico.
	BAJA	Usuarios <ul style="list-style-type: none"> Alumnos Docentes Veedores: <ul style="list-style-type: none"> Centro Federado de Estudiantes de la EPIS. Círculos de Investigación de EPIS. 	<ul style="list-style-type: none"> Presidente de la Comisión Organizadora Vicepresidente Administrativo Director de Escuelas EPIS

Nota: Elaborado por Tesista.

3.1.9. Gestión de compras.

En esta área de conocimiento se detallará como se adquirió para cada actividad los materiales y recursos humanos en los 10 meses que tienen la duración del proyecto, se puede observar en el siguiente **Tabla 14**.

Tabla 14:
Área de gestión de compras.

ACTIVIDADES			GASTOS POR MESES											
ITEM	DESCRIPCION	Adquisición	AGOSTO	SEPTIEMBRE	OCTUBRE	NOVIEMBRE	DICIEMBRE	ENERO	FEBRERO	MARZO	ABRIL	MAYO	JUNIO	Precio parcial
1.00.00	BIENES	COMPRADO Y ALQUILADO	S/. 482.50	S/. 482.50	S/. 482.50	S/. 482.50	S/. 482.50	S/. 482.50	S/. 482.50	S/. 482.50	S/. 482.50	S/. 482.50	S/. 0.00	S/. 4,825.00
2.00.00	SERVICIOS	COMPRADO Y ALQUILADO	S/. 161.00	S/. 161.00	S/. 161.00	S/. 161.00	S/. 161.00	S/. 161.00	S/. 161.00	S/. 161.00	S/. 161.00	S/. 161.00	S/. 0.00	S/. 1,610.00
3.00.00	SOFTWARE	GRATUITO	S/. 0.00	S/. 0.00	S/. 0.00	S/. 0.00	S/. 0.00	S/. 0.00	S/. 0.00	S/. 0.00	S/. 0.00	S/. 0.00	S/. 0.00	S/. 0.00
4.00.00	RECURSOS HUMANOS	CONTRATADO	S/. 770.00	S/. 770.00	S/. 770.00	S/. 770.00	S/. 770.00	S/. 770.00	S/. 770.00	S/. 770.00	S/. 770.00	S/. 770.00	S/. 0.00	S/. 7,700.00
5.00.00	GASTOS GENERALES		S/. 141.35	S/. 141.35	S/. 141.35	S/. 141.35	S/. 141.35	S/. 141.35	S/. 141.35	S/. 141.35	S/. 141.35	S/. 141.35	S/. 0.00	S/. 1,413.50
		SUB TOTAL DE GASTOS	S/. 1,554.85	S/. 1,554.85	S/. 1,554.85	S/. 1,554.85	S/. 1,554.85	S/. 1,554.85	S/. 1,554.85	S/. 1,554.85	S/. 1,554.85	S/. 1,554.85	S/. 0.00	S/. 15,548.50

Nota: Elaborado por el tesista.

3.1.10. Gestión de los interesados.

Se puede identificar en el **Tabla 15** los **STAKEHOLDERS**, que pueden ser afectados con la implementación del proyecto.

Tabla 15:
Área de gestión de los recursos humanos.

ROL GENERAL	STAKEHOLDERS
ASESOR	Asesor del Proyectista
EQUIPO DE PROYECTO	Tesista <ul style="list-style-type: none"> Bach. Percy Alhuay Carrasco Equipo de gestión de proyecto <ul style="list-style-type: none"> Desarrollador Analista y Diseñador
COMINTE EJECUTIVO	<ul style="list-style-type: none"> Presidente de la comisión Organizadora. Vicepresidente Académico: Vice - Presidente Administrativo.
GERENTES DE OPERACIONES	Jefe de Registros Académicos. Directores de Escuelas EPIS.

GERENTES FUNCIONALES	Jefe de la Oficina de Sistemas de Información
USUARIOS / CLIENTES	Cliente: <ul style="list-style-type: none"> • Universidad Nacional José María Arguedas - EPIS. Usuario: <ul style="list-style-type: none"> • Docentes. • Alumnos.
OTROS STAKEHOLDERS	Veedores: <ul style="list-style-type: none"> • Centro Federado de Estudiantes de la EPIS. • Círculos de Investigación de LA EPIS.

Nota: Elaborado por el tesista.

3.2. Metodología de desarrollo de software XP (Programación Extrem).

La metodología de desarrollo de software empleada será XP, por ser una metodología ágil centrada en potenciar las relaciones interpersonales par el éxito del proyecto Las fases que lo componen se nuestra a continuación.

3.2.1. Planificación

- ❖ Historias de Usuario registrar
 - Se muestra en la **Tabla 16** registrar docente como administrador

Tabla 16:

Recolección de historias de registrar docente.

HISTORIA DE USUARIO	
Numero: 3	Usuario: Adminsitrador
Nombre historia:	Registro Docentes
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteracion asignada: 1
Programador responsable: Equipo xp	
Descripcion	
Se mostrara en pantalla una interfaz donde permitira al administrador ingresar los datos del Docentes	
Observacion	
Estos datos se almacenara en una base de datos	

Nota: Desarrollado por el tesista.

- Se muestra en la **Tabla 17** registrar salón como administrador

Tabla 17:

Recolección de historias de registrar salón.

HISTORIA DE USUARIO	
Numero: 1	Usuario: Adminsitrador
Nombre historia:	Registro Salon
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion Se mostrara pantalla una interfaz dodne permitira al administrador ingresar los datos del salon	
Observacion Estos datos se almacenara en una base de datos	

Nota: Desarrollado por el tesista.

- Se muestra en la **Tabla 18** registrar usuario como administrador.

Tabla 18:

Recolección de historias de registrar usuarios.

HISTORIA DE USUARIO	
Numero: 4	Usuario: Adminsitrador
Nombre historia:	Registro Uusuarios
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion Se mostrara en pantalla una interfaz donde permitira al administrador ingresar los datos del usuario	
Observacion Estos datos se almacenara en una base de datos	

Nota: Desarrollado por el tesista.

- Se muestra en la **Tabla 19** registrar curso como administrador.

Tabla 19:

Recolección de historias de registrar curso.

HISTORIA DE USUARIO	
Numero: 2	Usuario: Adminsitrador
Nombre historia:	Registro Curso
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion Se mostrara pantalla una interfaz dodne permitira al administrador ingresar los datos del Curso	
Observacion Cada curso tiene Id para ingresar Manualmente	

Nota: Desarrollado por el tesista.

❖ Historia de usuarios modificar

- Se muestra en la **Tabla 20** modificar docente como administrador.

Tabla 20:

Recolección de historias para modificar docente

HISTORIA DE USUARIO	
Numero: 3	Usuario: Adminsitrador
Nombre historia:	Modificar Docentes
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion Se mostrara en pantalla una interfaz donde permitira al administrador modificarlos datos del Docentes	
Observacion La modificacion se almacenara en una base de datos	

Nota: Desarrollado por el tesista

- Se muestra en la **Tabla 21** modificar salón como administrador.

Tabla 21:

Recolección de historias para modificar salón.

HISTORIA DE USUARIO	
Numero: 1	Usuario: Adminsitrador
Nombre historia:	Modificar Salon
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion Se mostrara pantalla una interfaz donde permitira al administrador modificar los datos ingresados del salon	
Observacion La modificacion se almacenara en una base de datos	

Nota: Desarrollado por el tesista.

- Se muestra en la **Tabla 22** modificar usuario como administrador.

Tabla 22:

Recolección de historias para modificar usuarios.

HISTORIA DE USUARIO	
Numero: 4	Usuario: Adminsitrador
Nombre historia:	Modificar Usuarios
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion Se mostrara en pantalla una interfaz donde permitira al administrador modificar los datos del usuario	
Observacion La modificacion se almacenara en una base de datos	

Nota: Desarrollado por el tesista.

- Se muestra en la **Tabla 23** modificar curso como administrador.

Tabla 23:

Recolección de historias para modificar curso.

HISTORIA DE USUARIO	
Numero: 2	Usuario: Adminsitrador
Nombre historia:	Modificar Curso
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion	
Se mostrara en pantalla una interfaz donde permitira al administrador modificar los datos del Curso	
Observacion	
La modificacion se almacenara en una base de datos	

Nota: Desarrollado por el tesista.

- ❖ Historia de usuarios eliminar.

- Se muestra en la **Tabla 24** eliminar docente como administrador.

Tabla 24:

Recolección de historias para eliminar docente.

HISTORIA DE USUARIO	
Numero: 3	Usuario: Adminsitrador
Nombre historia:	Eliminar Docentes
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion	
Se mostrara en pantalla una interfaz donde permitira al administrador eliminar los datos del docentes	
Observacion	
La eliminacion se almacenara en una base de datos	

Nota: Desarrollado por el tesista.

- Se muestra en la **Tabla 25** eliminar salón como administrador.

Tabla 25:

Recolección de historias para eliminar salón.

HISTORIA DE USUARIO	
Numero: 1	Usuario: Adminsitrador
Nombre historia:	Eliminar Salon
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion Se mostrara pantalla una interfaz donde permitira al administrador eliminar los datos ingresados del salon	
Observacion La eliminacion se almacenara en una base de datos	

Nota: Desarrollado por el tesista.

- Se muestra en la **Tabla 26** eliminar usuario como administrador.

Tabla 26:

Recolección de historias para eliminar usuarios.

HISTORIA DE USUARIO	
Numero: 4	Usuario: Adminsitrador
Nombre historia:	Eliminar Usuarios
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion Se mostrara en pantalla una interfaz donde permitira al administrador eliminar los datos del usuario	
Observacion La eliminacion se almacenara en una base de datos	

Nota: Desarrollado por el tesista.

- Se muestra en la **Tabla 27** eliminar curso como administrador.

Tabla 27:

Recolección de historias para eliminar curso.

HISTORIA DE USUARIO	
Numero: 2	Usuario: Adminsitrador
Nombre historia:	Eliminar Curso
Prioridad en negocio Alta	Riesgo en desarrollo Alta
Puntos estimados: 2	Iteraccion asignada: 1
Programador responsable: Equipo xp	
Descripcion Se mostrara en pantalla una interfaz donde permitira al administrador Eliminar los datos del Curso	
Observacion La eliminacion se almacenara en una base de datos	

Nota: Desarrollado por el tesista.

❖ Plan de entrega.

Es una planificación donde el desarrollador y cliente establecen los tiempos de implementación ideales de usuario se muestra en la **Tabla 28**.

Tabla 28:

Secuencia de actividades de Plan de entrega.

Plan Release					
# His Us	Tarea	Nombre Encargado	Itera	Fecha Inicio	Fecha Fin
1	Recolección de historias de usuario	Bach. Percy Alhuay	1a	10/08/15	21/09/2015
			2a	18/08/15	21/09/2015
2	Velocidad del proyecto	Bach. Percy Alhuay	1a	31/08/15	01/09/15
			2a	02/09/15	04/09/15
3	Recolección plan RELEASE	Bach. Percy Alhuay	1a	24/08/15	25/08/15
			2a	26/08/15	27/08/15
			3a	27/08/15	28/08/15

4	Recolección de diagrama de clases	Bach. Percy Alhuay	1a	22/09/15	01/12/15
			2a	01/12/15	08/12/15
5	Reuniones	Bach. Percy Alhuay	1a	15/09/15	18/09/15
			2a	19/09/15	19/09/15
			3a	21/09/15	21/09/15
7	Correcciones	Bach. Percy Alhuay	1a	20/08/15	21/09/15
			2a		
			3a		

Nota: Desarrollado por el tesista..

❖ Velocidad del Proyecto.

El número de historias de usuarios por iteración no fue una buena medida de la velocidad del proyecto debido a que no todos tenían el mismo nivel de dificultad y por tanto el mismo requerimiento de horas de desarrollo. Por eso se encontró que en la segunda iteración se trabajó menos horas semanales en comparación del primero como se muestra en la **Tabla 29**. El motivo de este resultado fue por el nivel de dificultad tanto, el número de horas requeridas para las historias de usuarios de la última fue el más bajo de todo el proyecto.

Tabla 29:

Modelo de Velocidad del proyecto.

	Iteración 1	Iteración 2	Iteración 3	Iteración 4
Horas	30	10	10	5
Semanas	3	2	2	1
Horas semanales	10	5	5	5
Historia de usuarios				
Velocidad del proyecto	12	8	8	8

Nota: Desarrollado por el tesista..

Si bien esta medida de velocidad del proyecto fue tomada en cuenta para el análisis de tiempo, y planificar las entregas de acuerdo a esta medida y precisa.

- ❖ Programación en Pareja.
 - Cada persona en el equipo debe conocer mucho del código de cada sección del proyecto.
 - El objetivo que ambos miembros del grupo conocieran cada aspecto de la implantación.
 - Rotar los programadores en partes del desarrollo.
- ❖ Reuniones Diarias.
 - Las historias son traducidas a tareas al comienzo de cada iteración.
 - Se estiman los tiempos para la realización de tareas en días ideales.
 - Las reuniones se realizarán en el sitio de trabajo del equipo
 - Evitar discusiones largas

3.2.2. Diseño.

❖ Diseños simples.

➤ Diagrama de casos de usos.

- En la **Figura 14** se muestra el caso de uso del usuario

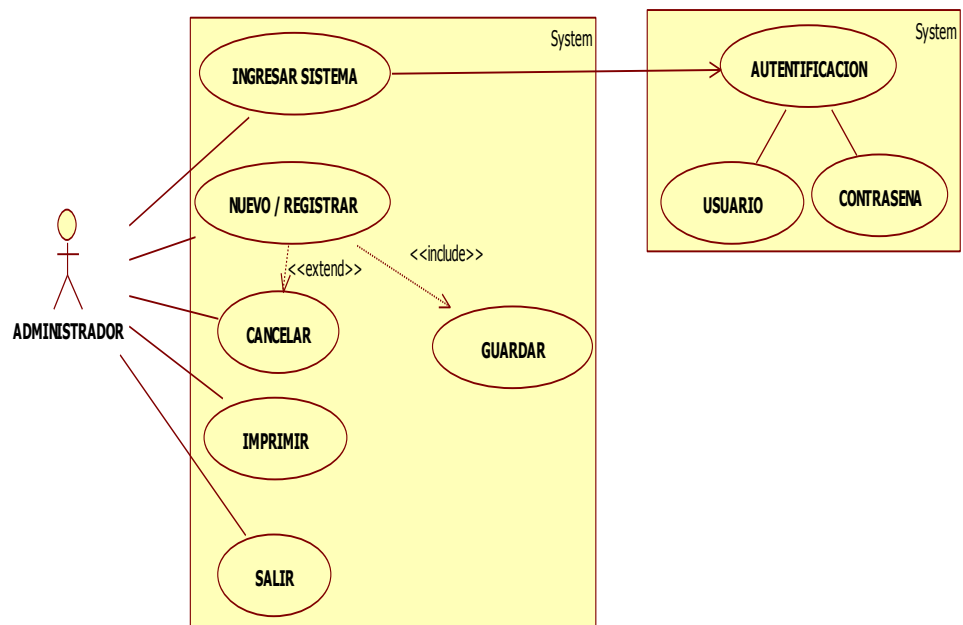


Figura 14. Diagrama de casos de uso registrar usuario

Fuente: Desarrollado por el tesista

o En la **Figura 15** se muestra el caso de uso del salón.

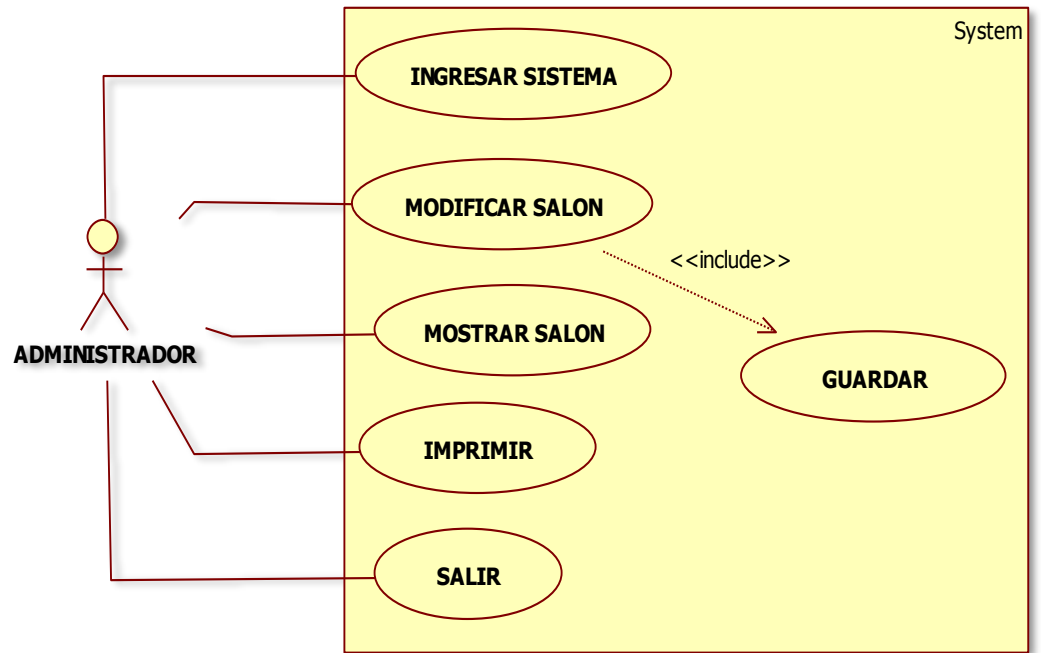


Figura 15. Diagrama de casos de uso registrar salón.

Fuente: Desarrollado por el tesista.

o En la **Figura 16** se muestra el caso de uso del curso.

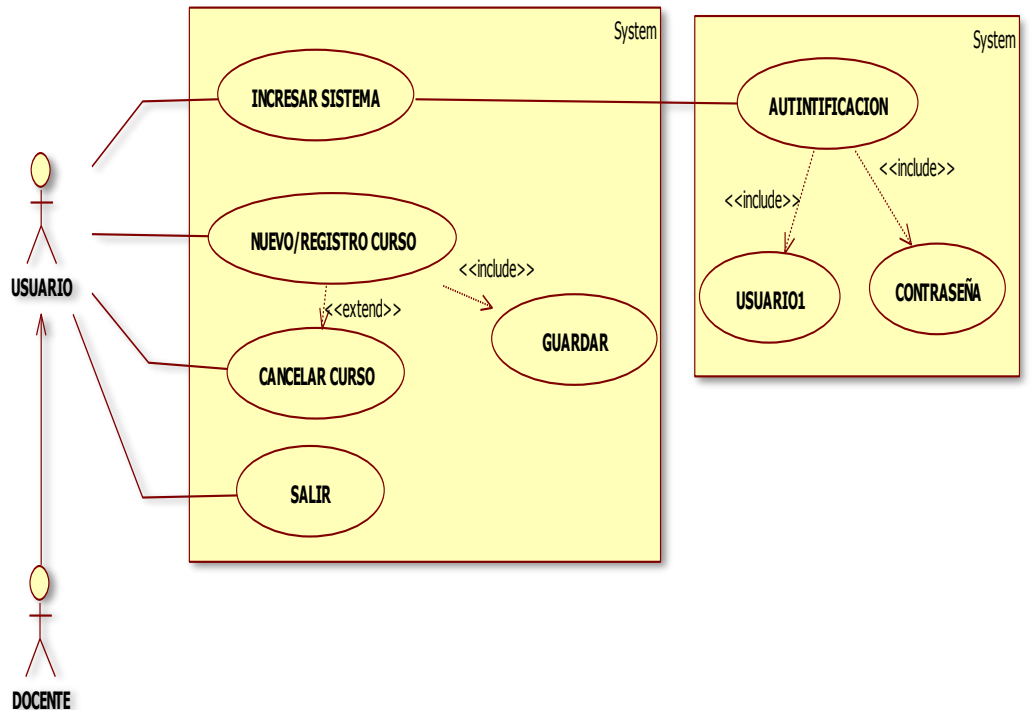


Figura 16. Diagrama de casos de uso registrar curso.

Fuente: Desarrollado por el tesista.

o En la **Figura 17** se muestra el caso de uso del docente.

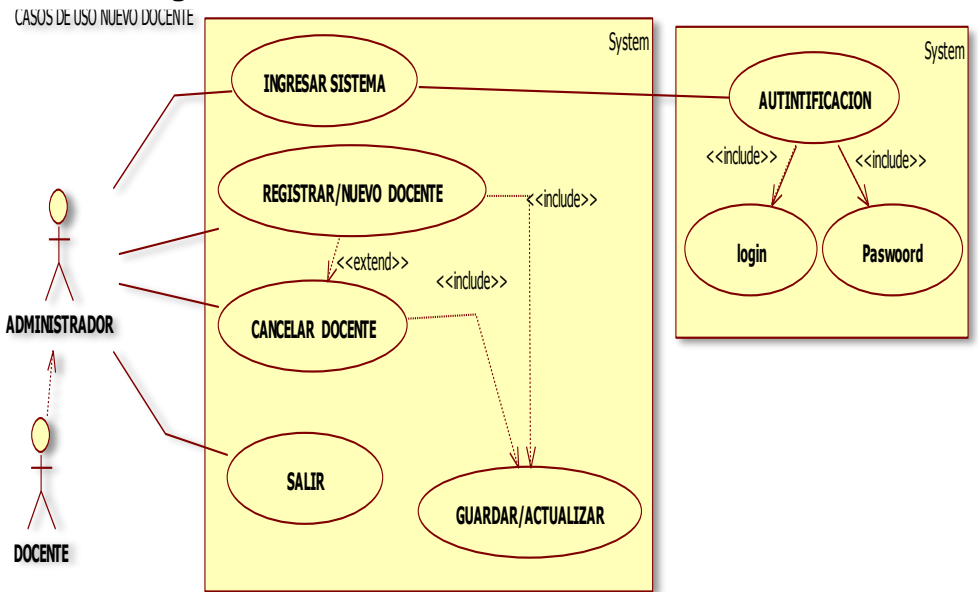


Figura 17. Diagrama de casos de uso registrar docente.

Fuente: Desarrollado por el tesista.

o En la **Figura 18** se muestra el caso de uso curso para realizar Mantenimiento.

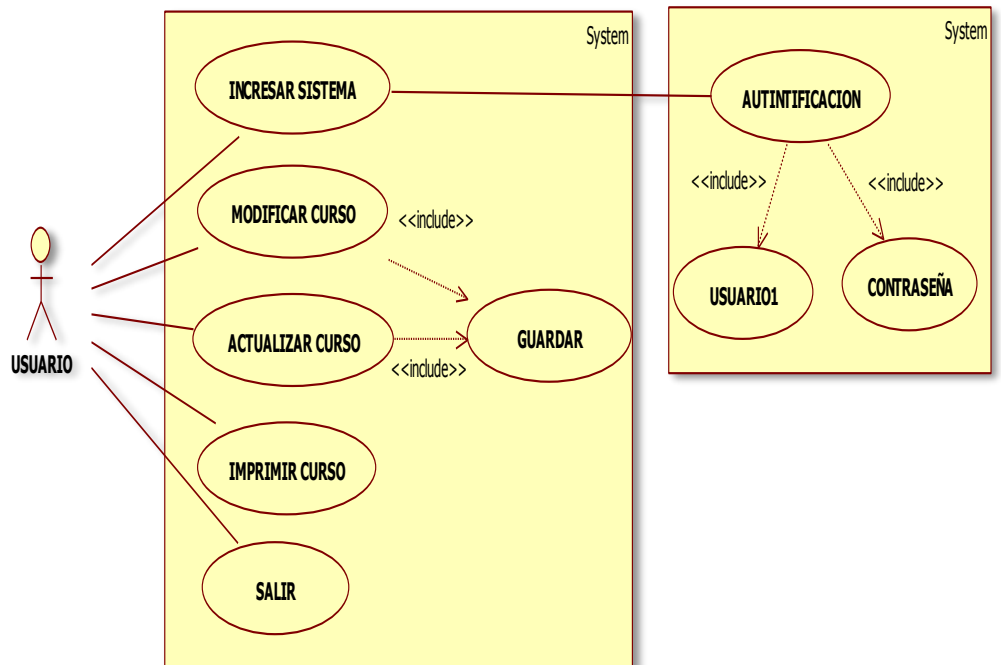


Figura 18. Diagrama de casos de uso modificar curso.

Fuente: Desarrollado por el tesista.

o En la **Figura 19** se muestra el caso de uso docente para modificar en el sistema.

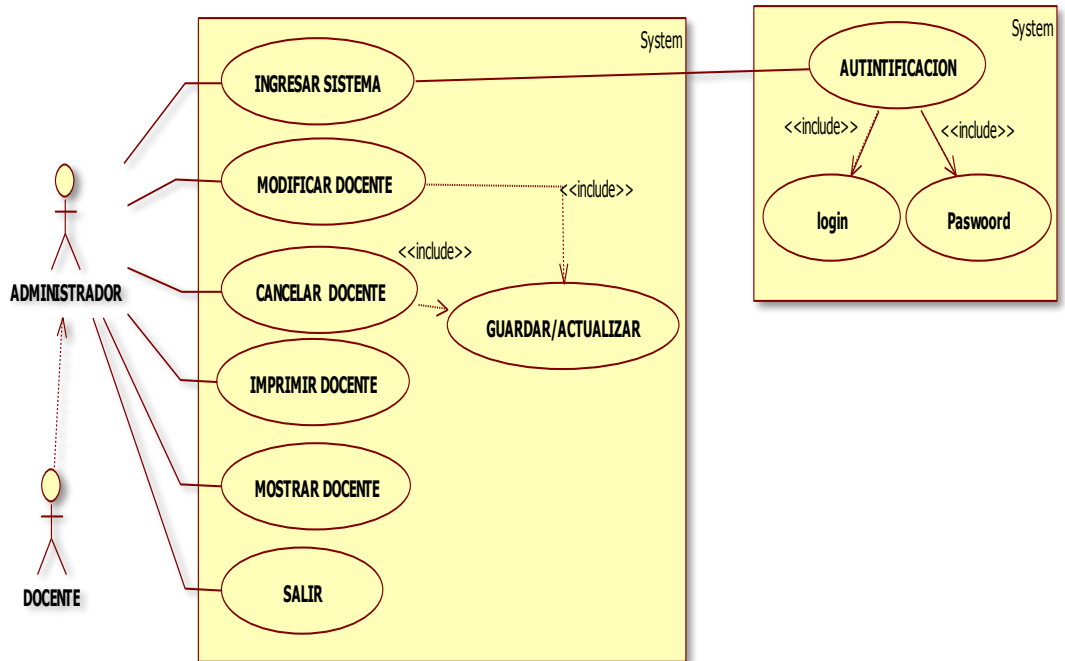


Figura 19. Diagrama de casos de uso modificar docente.

Fuente: Desarrollado por el tesista.

o En la **Figura 20** se muestra el caso de uso del salón para realizar mantenimiento.

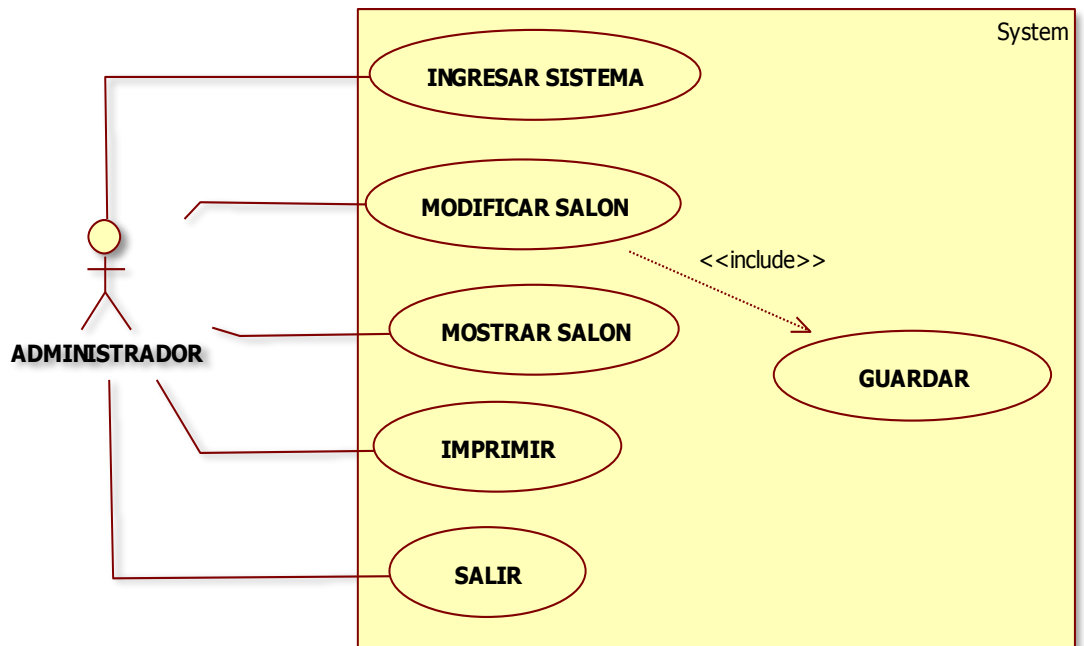


Figura 20. Diagrama de casos de uso modificar salón.

Fuente: Desarrollado por el tesista.

➤ Especificación de casos de uso.

- En la **Tabla 30** se detalla la especificación del curso para registrar

Tabla 30:

Especificación de caso de uso registrar curso.

Casos de Uso	Crear un nuevo curso
Tipo de C.U	Primario
Actores	Administrador, Secretario
Propósito	Insertar datos nuevos del curso en la base de datos.
Resumen	Administrador, secretario ingresa al sistema insertar datos nuevos para crear curso y luego presiona guardar.
Precondiciones	Tener información completa del curso.
Acción de los actores	Respuesta de sistema
<ul style="list-style-type: none"> • el usuario ingresa al sistema. • El usuario ingresa datos para crear nuevo curso. • El usuario crea y guarda los datos ingresados • El usuario elige a opción de salir. 	<ul style="list-style-type: none"> • El sistema pide datos para el acceso con login y el sistema muestra el interfaz principal. • El sistema recibe los datos del producto. • El sistema realiza la búsqueda en la base de datos y confirmar. • El sistema guardar los datos. • El sistema finaliza la aplicación.

Nota: Desarrollado por el tesista.

- En la **Tabla 31** se detalla la especificación de curso para registrar.

Tabla 31:

Especificación de caso de uso registrar curso.

Casos de Uso	Crear un nuevo docente
Tipo de C.U	Primario
Actores	Administrador, Secretario
Propósito	Insertar datos nuevos del docente en la base de datos.
Resumen	Administrador, secretario ingresa al sistema insertar datos nuevos para crear docente y luego presiona guardar.
Precondiciones	Tener información completa del D.
Acción de los actores	Respuesta de sistema
<ul style="list-style-type: none"> • el usuario ingresa al sistema. • El usuario ingresa datos para crear nuevo curso. 	<ul style="list-style-type: none"> • El sistema pide datos para el acceso con login y el sistema muestra el interfaz principal. • El sistema recibe los datos del docente

<ul style="list-style-type: none"> • El usuario crea y guarda los datos ingresados • El usuario elige a opción de salir. 	<ul style="list-style-type: none"> • El sistema realiza la búsqueda en la base de datos y confirmar. • El sistema guardar los datos. • El sistema finaliza la aplicación.
--	--

Nota: Desarrollado por el tesista.

- En la **Tabla 32** se detalla la especificación del salón para registrar.

Tabla 32:

Especificación de caso de uso registrar curso.

Casos de Uso	Crear un nuevo salón
Tipo de C.U	Primario
Actores	Administrador, Secretario
Propósito	Insertar datos nuevos del salón en la base de datos.
Resumen	Administrador, secretario ingresa al sistema insertar datos nuevos para crear salón y luego presiona guardar.
Precondiciones	Tener información completa del salón.
Acción de los actores	Respuesta de sistema
<ul style="list-style-type: none"> • el usuario ingresa al sistema. • El usuario ingresa datos para crear nuevo curso. • El usuario crea y guarda los datos ingresados • El usuario elige a opción de salir. 	<ul style="list-style-type: none"> • El sistema pide datos para el acceso con login y el sistema muestra el interfaz principal. • El sistema recibe los datos del salón. • El sistema realiza la búsqueda en la base de datos y confirmar. • El sistema guardar los datos. • El sistema finaliza la aplicación.

Nota: Desarrollado por el tesista.

- En la **Tabla 33** se detalla la especificación del curso para registrar.

Tabla 33:

Especificación de caso de uso registrar curso.

Casos de Uso	Crear un nuevo curso
Tipo de C.U	Primario
Actores	Administrador, Secretario
Propósito	Insertar datos nuevos del usuario en la base de datos.
Resumen	Administrador, secretario ingresa al sistema insertar datos nuevos para crear curso y luego presiona guardar.

Precondiciones	Tener información completa del curso.
Acción de los actores	Respuesta de sistema
<ul style="list-style-type: none"> • el usuario ingresa al sistema. • El usuario ingresa datos para crear nuevo curso. • El usuario crea y guarda los datos ingresados • El usuario elige a opción de salir. 	<ul style="list-style-type: none"> • El sistema pide datos para el acceso con login y el sistema muestra el interfaz principal. • El sistema recibe los datos del producto. • El sistema realiza la búsqueda en la base de datos y confirmar. • El sistema guardar los datos. • El sistema finaliza la aplicación.

Nota: Desarrollado por el tesista

○En la **Tabla 34** se detalla la especificación del curso para modificar en el sistema.

Tabla 34:

Especificación de caso de uso modificar curso.

Casos de Uso	Modificar curso.
Tipo de C.U	Primario
Actores	Administrador
Propósito	Modificar un dato que ya existe en la base de datos.
Resumen	El administrador ingresa al sistema y modifica datos existentes.
Precondiciones	Tener información completa y correcta para la buena solución.
Acción de los actores	Respuesta de sistema
<ul style="list-style-type: none"> • El administrador selecciona la pestaña del modificar. • El administrador selecciona el dato para modificar. • El administrador modifica y elige guardar los datos • El sistema recibe los datos del producto. • El administrador finaliza la aplicación. 	<ul style="list-style-type: none"> • El sistema carga la ventana de modificar. • El sistema muestra lo buscado para modificar. • Confirma lo modificado del dato • Se cierra todo el sistema

Nota: Desarrollado por el tesista.

○En la **Tabla 35** se detalla la especificación del docente para Modificar en el sistema.

Tabla 35:*Especificación de caso de uso modificar docente.*

Casos de Uso	Modificar docente
Tipo de C.U	Primario
Actores	Administrador
Propósito	Modificar un dato que ya existe en la base de datos.
Resumen	El administrador ingresa al sistema y modifica datos existentes.
Precondiciones	Tener información completa y correcta para la buena solución.
Acción de los actores	Respuesta de sistema
<ul style="list-style-type: none"> • El administrador selecciona la pestaña del modificar. • El administrador selecciona el dato para modificar. • El administrador modifica y elige guardar los datos • El sistema recibe los datos del producto. • El administrador finaliza la aplicación. 	<ul style="list-style-type: none"> • El sistema carga la ventana de modificar. • El sistema muestra lo buscado para modificar. • Confirma lo modificado del dato • Se cierra todo el sistema

Nota: Desarrollado por el tesista.

○En la **Tabla 36** se detalla la especificación del salón para modificar en el sistema.

Tabla 36:*Especificación de caso de uso modificar salón.*

Casos de Uso	Modificar salón.
Tipo de C.U	Primario
Actores	Administrador
Propósito	Modificar un dato que ya existe en la base de datos.
Resumen	El administrador ingresa al sistema y modifica datos existentes.
Precondiciones	Tener información completa y correcta para la buena solución.
Acción de los actores	Respuesta de sistema
<ul style="list-style-type: none"> • El administrador selecciona la pestaña del modificar. 	<ul style="list-style-type: none"> • El sistema carga la ventana de modificar.

<ul style="list-style-type: none"> • El administrador selecciona el dato para modificar. • El administrador modifica y elige guardar los datos • El sistema recibe los datos del producto. • El administrador finaliza la aplicación. 	<ul style="list-style-type: none"> • El sistema muestra lo buscado para modificar. • Confirma lo modificado del dato • Se cierra todo el sistema
---	---

Nota: Desarrollado por el tesista.

○ En la **Tabla 37** se detalla la especificación del usuario para modificar.

Tabla 37:

Especificación de caso de uso modificar curso

Casos de Uso	Modificar usuario
Tipo de C.U	Primario
Actores	Administrador
Propósito	Modificar un dato que ya existe en la base de datos.
Resumen	El administrador ingresa al sistema y modifica datos existentes.
Precondiciones	Tener información completa y correcta para la buena solución.
Acción de los actores	Respuesta de sistema
<ul style="list-style-type: none"> • El administrador selecciona la pestaña del modificar. • El administrador selecciona el dato para modificar. • El administrador modifica y elige guardar los datos • El sistema recibe los datos del producto. • El administrador finaliza la aplicación. 	<ul style="list-style-type: none"> • El sistema carga la ventana de modificar. • El sistema muestra lo buscado para modificar. • Confirma lo modificado del dato • Se cierra todo el sistema

Nota: Desarrollado por el tesista.

➤ Diagrama de colaboración

○ En la **Figura 21** se muestra toda la secuencia de Registrar salón.

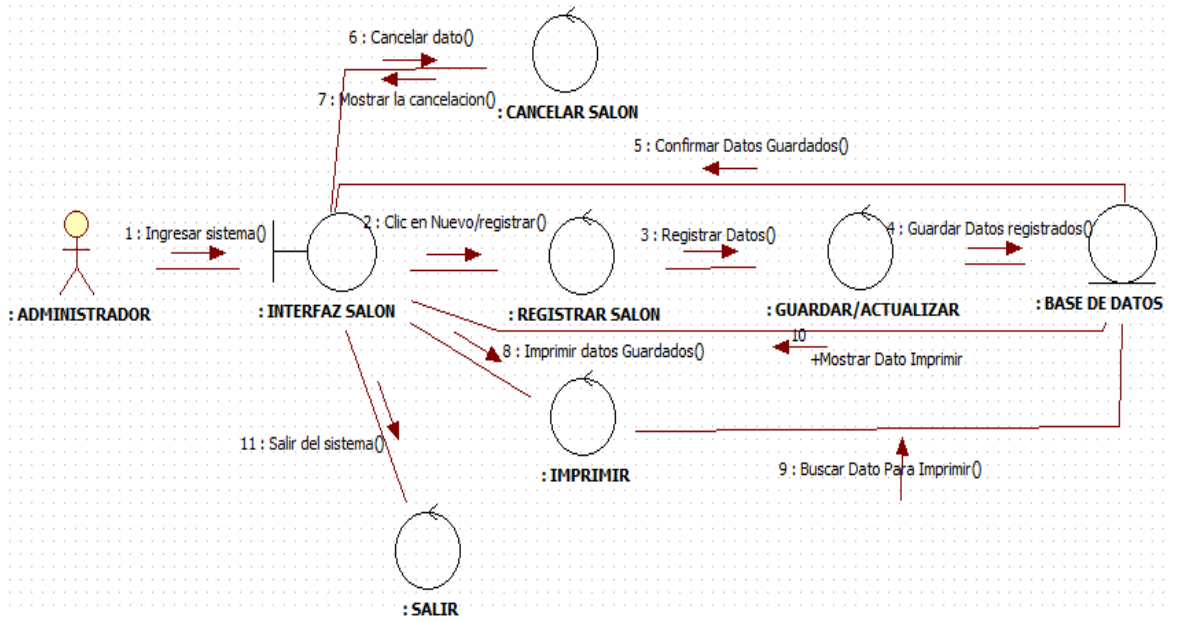


Figura 21. Diagrama de colaboración de registrar salón.
Fuente: Desarrollado por el tesista.

o En la **Figura 22** se muestra toda la secuencia de **Registrar curso**.

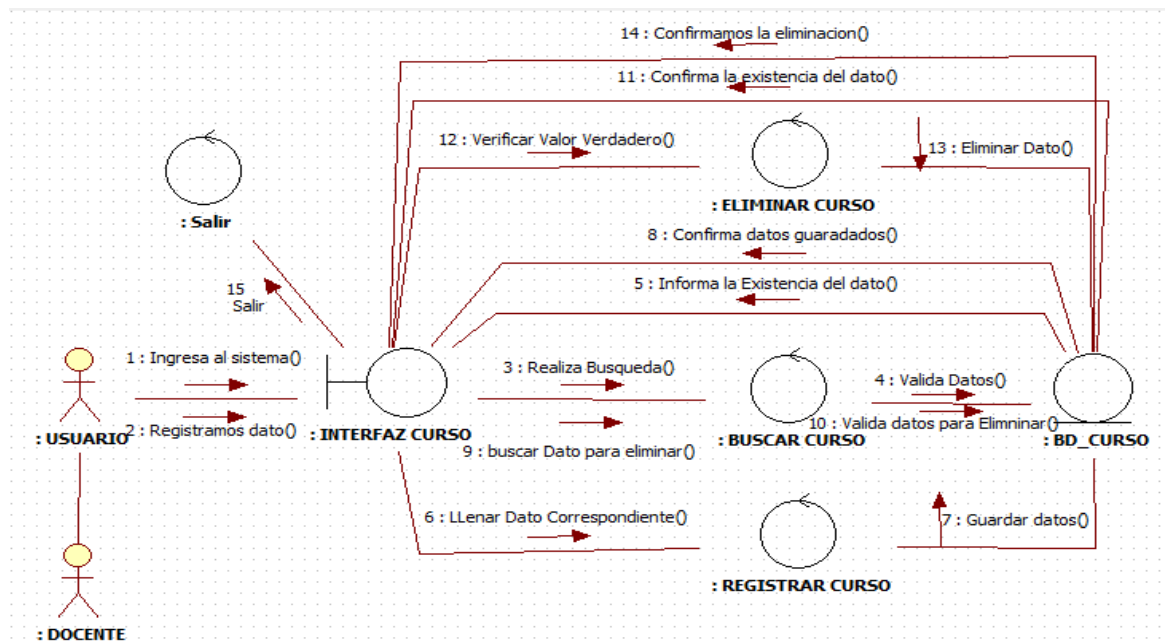


Figura 22. Diagrama de colaboración de registrar curso.
Fuente: Desarrollado por el tesista.

o En la **Figura 23** se muestra toda la secuencia de registrar docente.

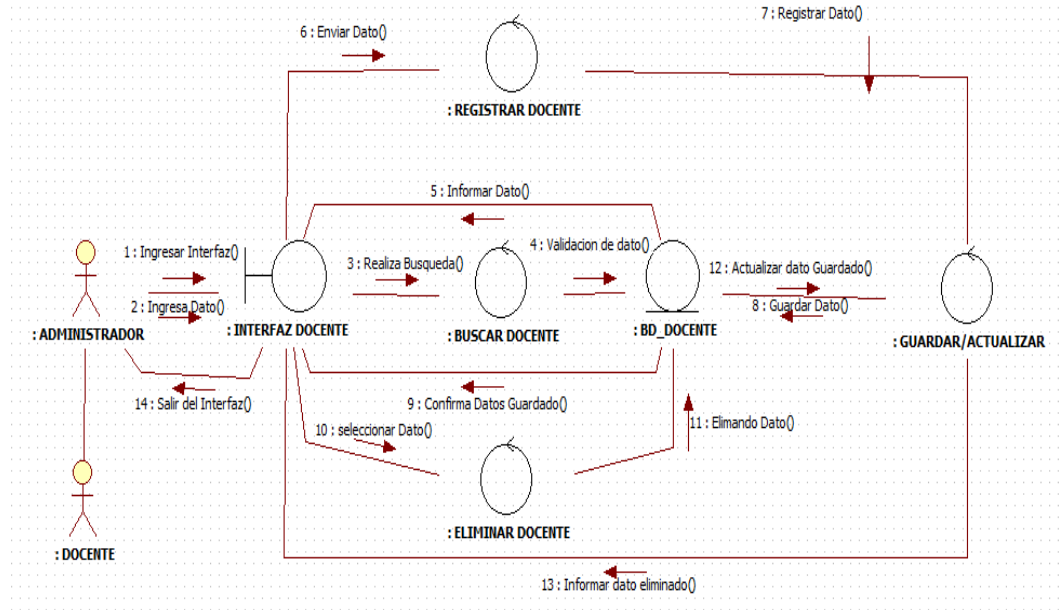


Figura 23. Diagrama de colaboración de registrar docente.

Fuente: Desarrollado por el tesista.

○ En la **Figura 24** se muestra toda la secuencia de **modificar docente**.

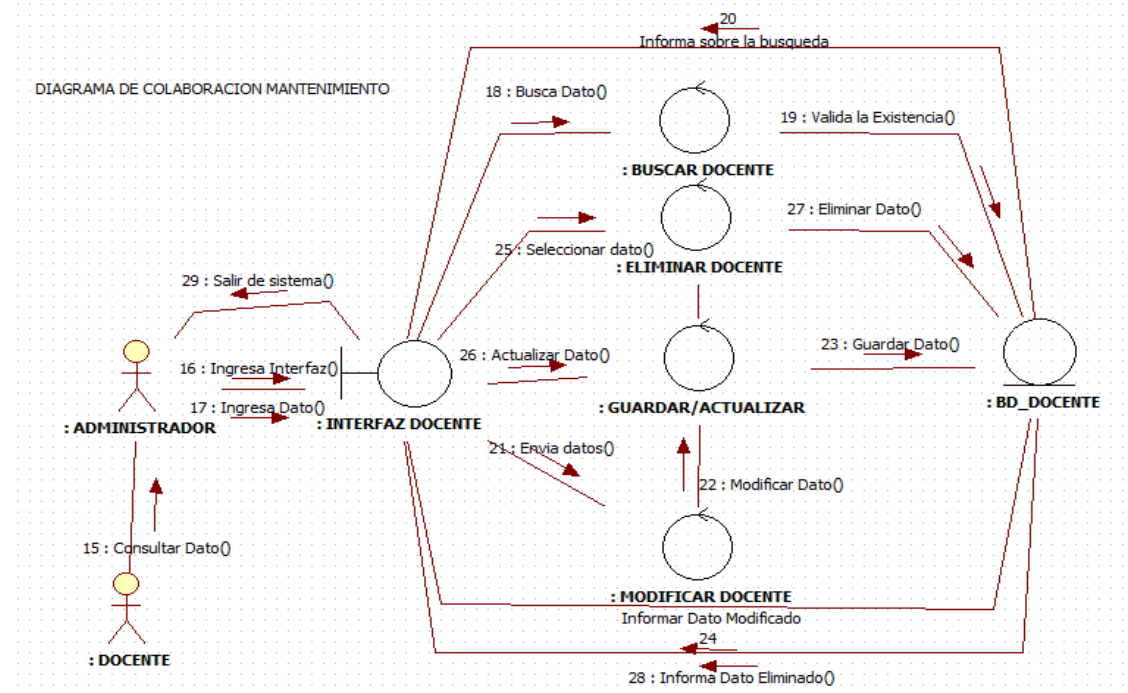


Figura 24. Diagrama de colaboración de modificar docente.

Fuente: Desarrollado por el tesista. del tesista.

○ En la **Figura 25** se muestra toda la secuencia de **modificar curso**.

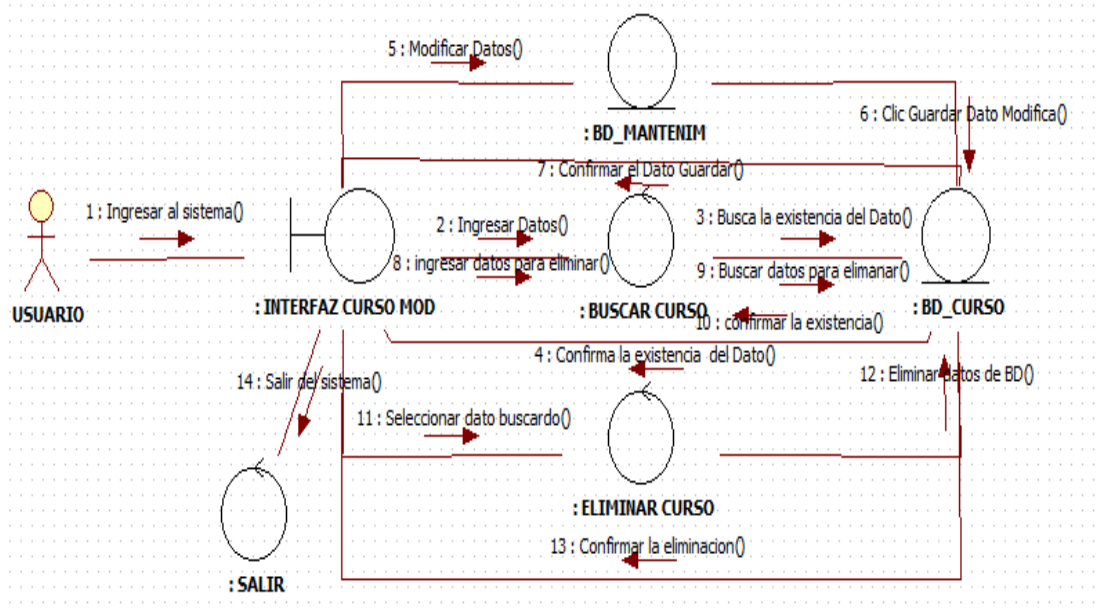


Figura 25. Diagrama de colaboración de modificar curso.

Fuente: Desarrollado por el tesista.

- En la **Figura 26** se muestra toda la secuencia de **modificar salón**.

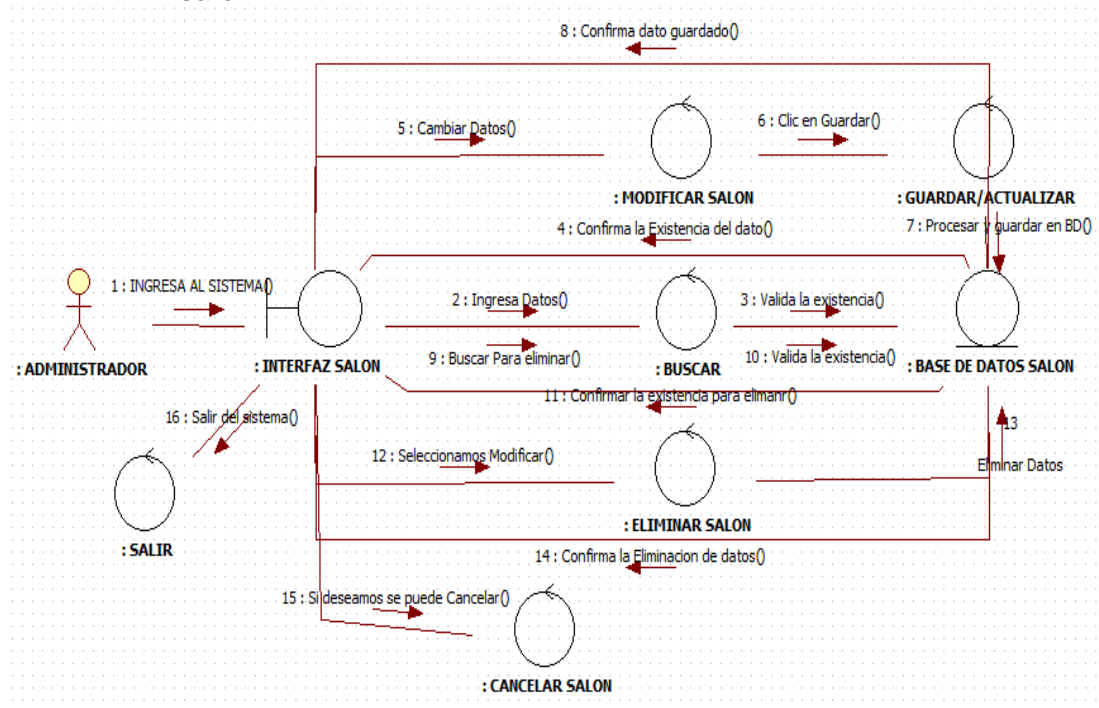


Figura 26. Diagrama de colaboración modificar salón.

Fuente: Desarrollado por el tesista.

➤ **Diagrama de secuencia.**

- En la **Figura 27** especifica la secuencia de actividades de salón para registrar.

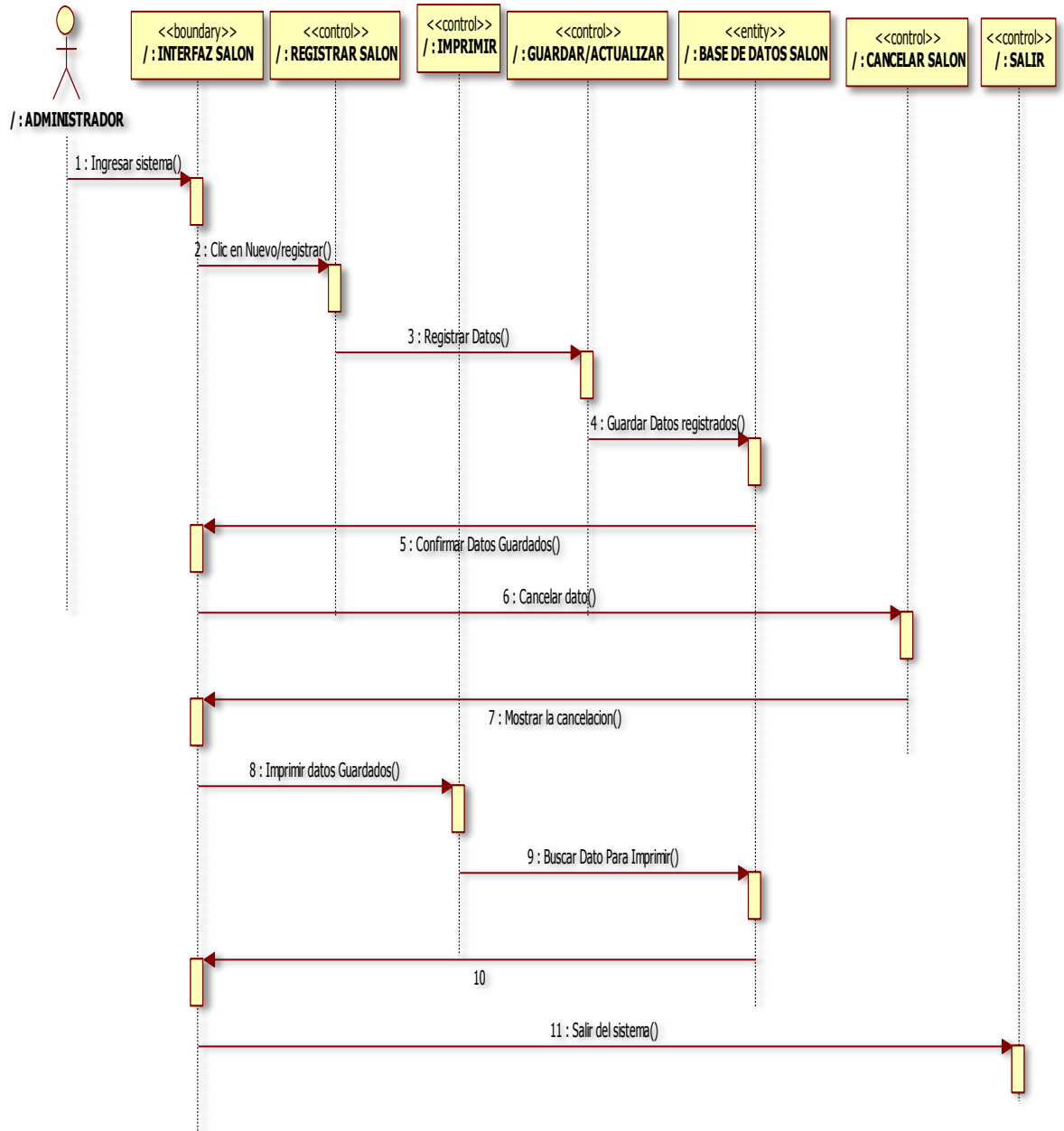


Figura 27. Diagrama de secuencia de registrar salón.

Fuente: Desarrollado por el tesista.

- o En la **Figura 28** especifica la secuencia de actividades de docente para registrar.

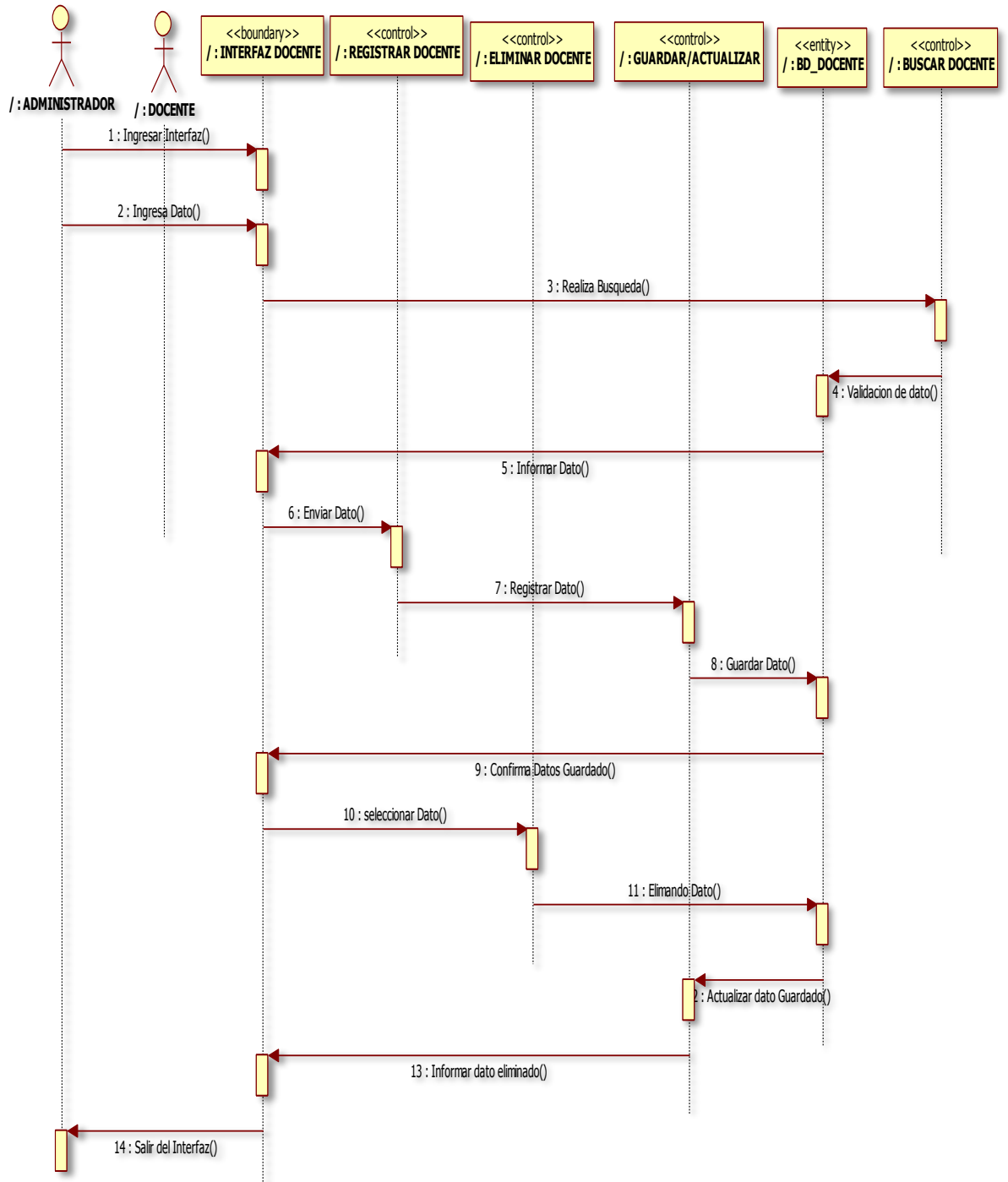


Figura 28. Diagrama de secuencia de registrar Docente.

Fuente: Desarrollado por el testista.

- o En la **Figura 29** especifica la secuencia de actividades de curso para registrar.

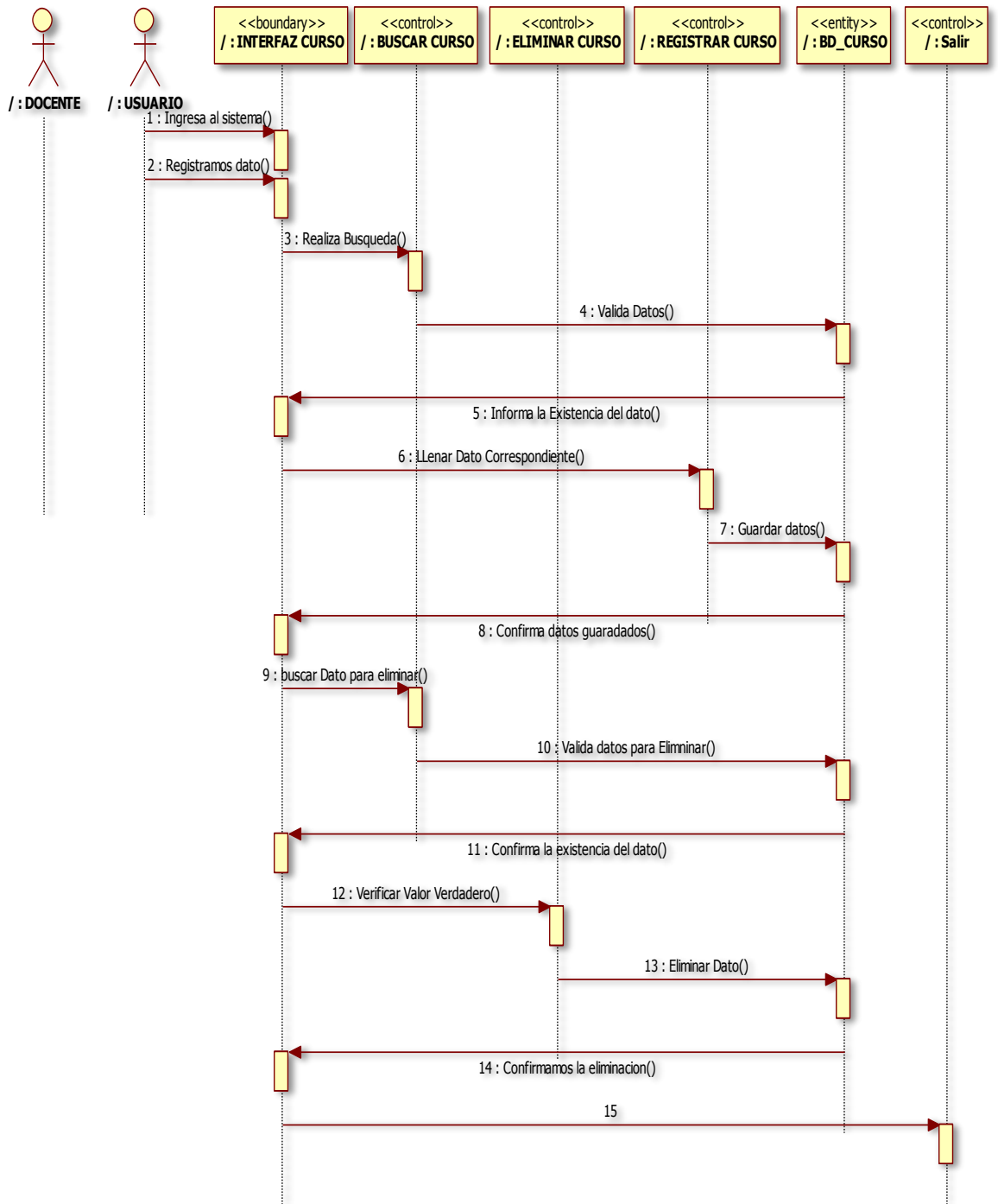


Figura 29. Diagrama de secuencia de registrar curso.

Fuente: Desarrollado por el tesista.

- o En la **Figura 30** especifica la secuencia de actividades de salón para modificar.

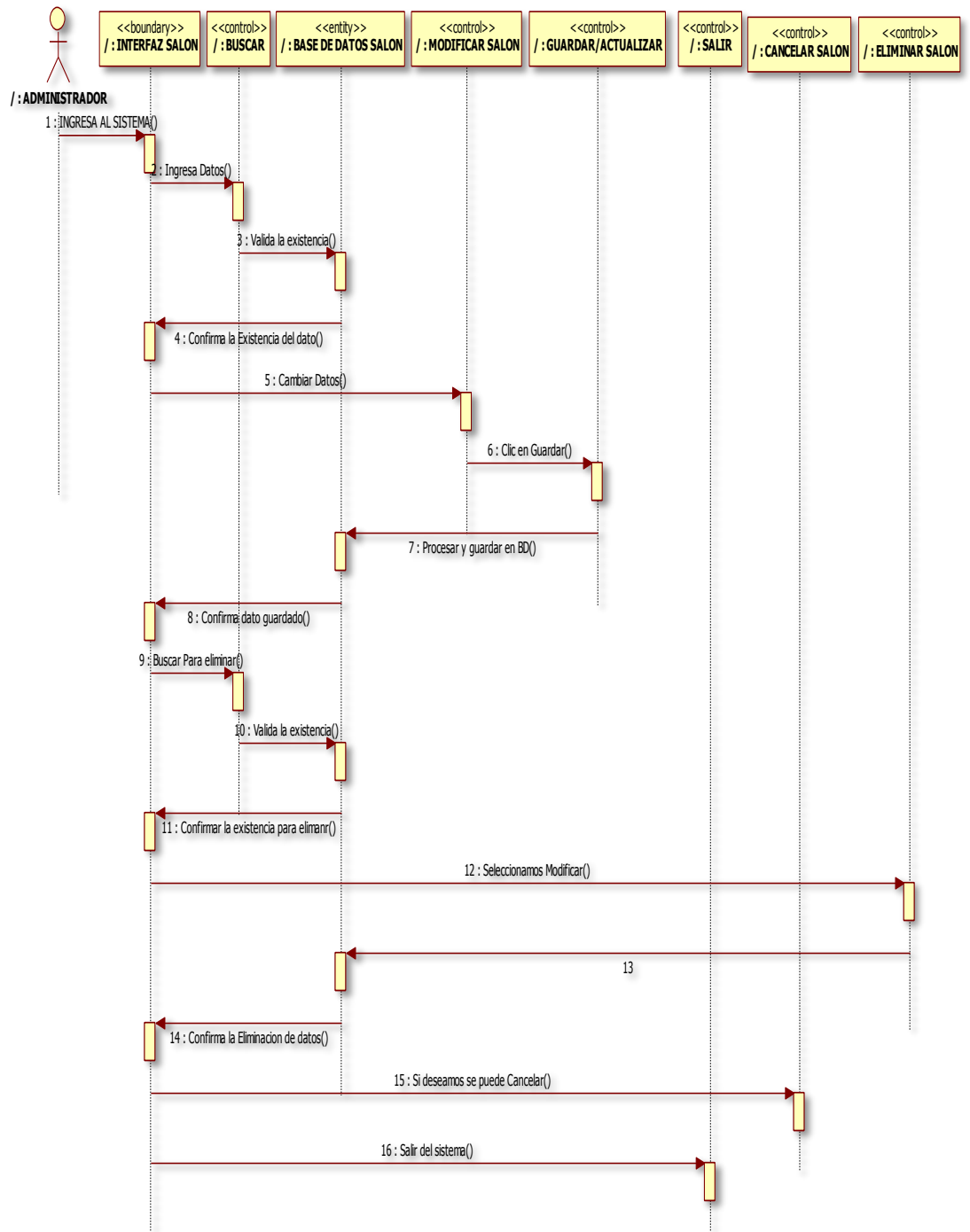


Figura 30. Diagrama de secuencia modificar salón

Fuente: Desarrollado por el tesista

- o En la **Figura 31** especifica la secuencia de actividades de docente para modificar.

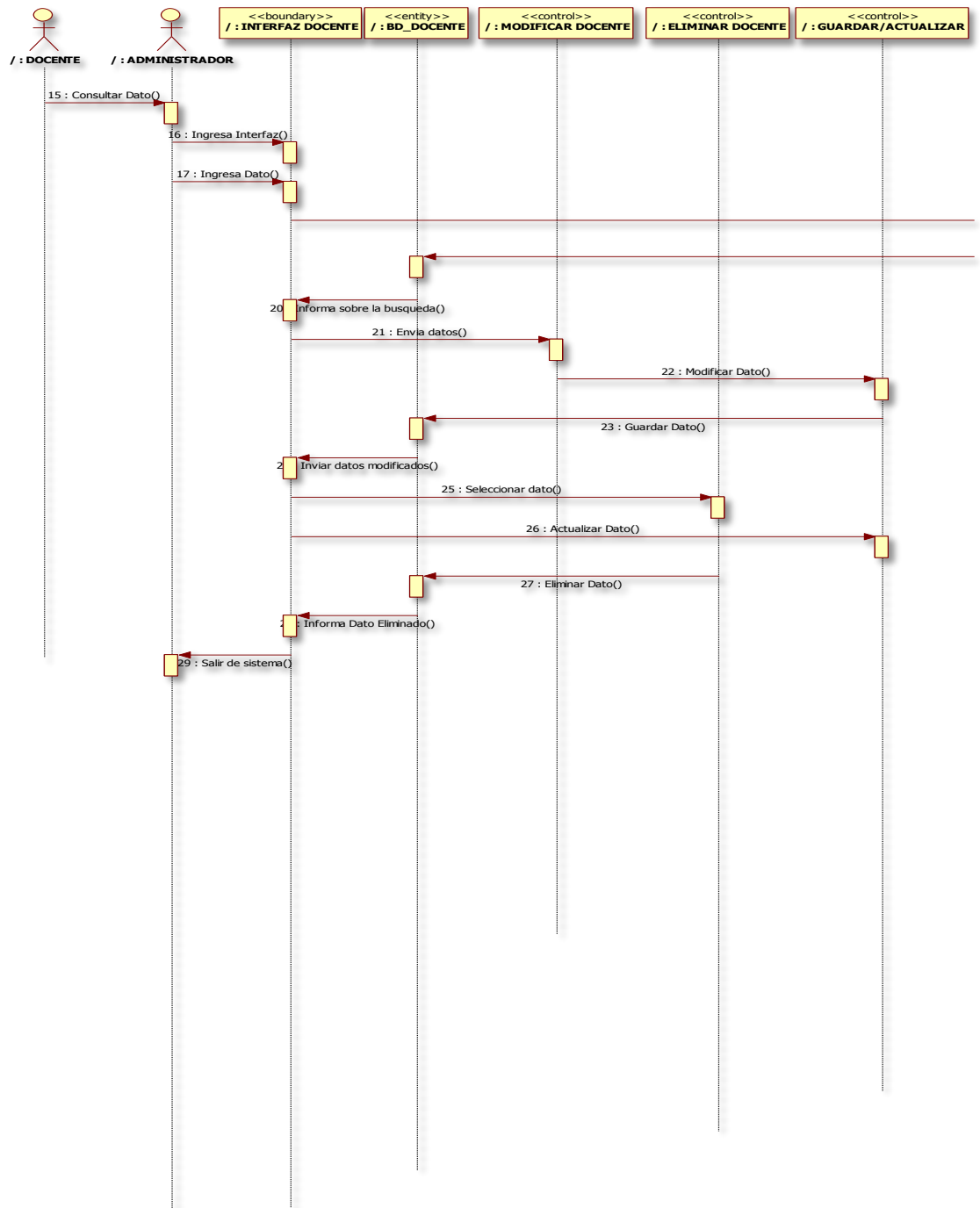


Figura 31. Diagrama de secuencia modificar docente.

Fuente: Desarrollado por el tesista.

- o En la **Figura 33** especifica la secuencia de actividades de curso para modificar.

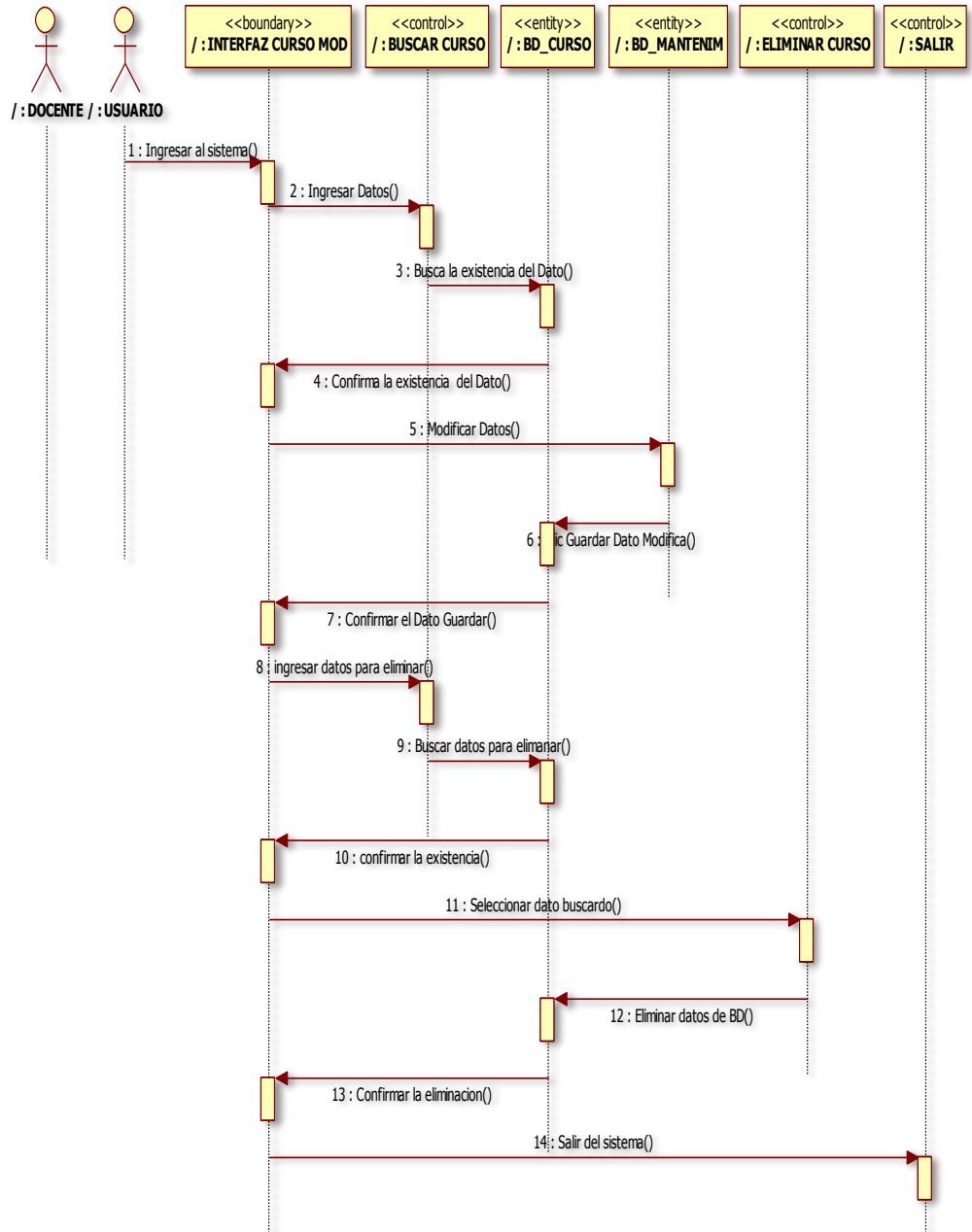


Figura 33. Diagrama de secuencia de registrar salón.

Fuente: Desarrollado por el tesista.

- o En la **Figura 34** especifica el despliegue paso desde un .EXE Diagrama de componentes

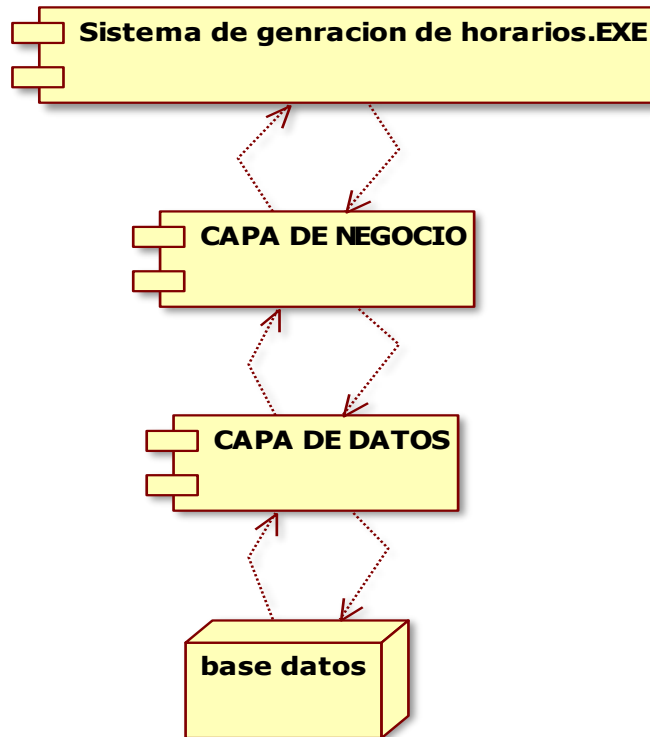


Figura 34. Diagrama de componentes del sistema.

Fuente: Desarrollado por el tesista.

- o En la **Figura 35** se muestra el diagrama de despliegue.

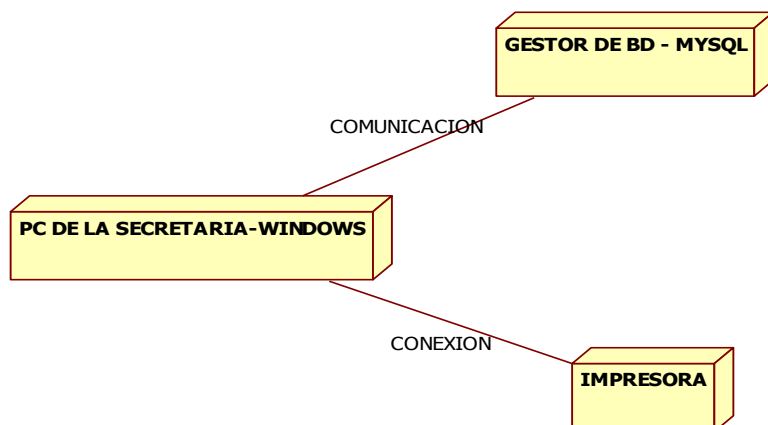


Figura 35. Diagrama de despliegue.

Fuente: Desarrollado por el tesista.

- o En la **Figura 36** se muestra diagrama de entidad relación del proyecto.

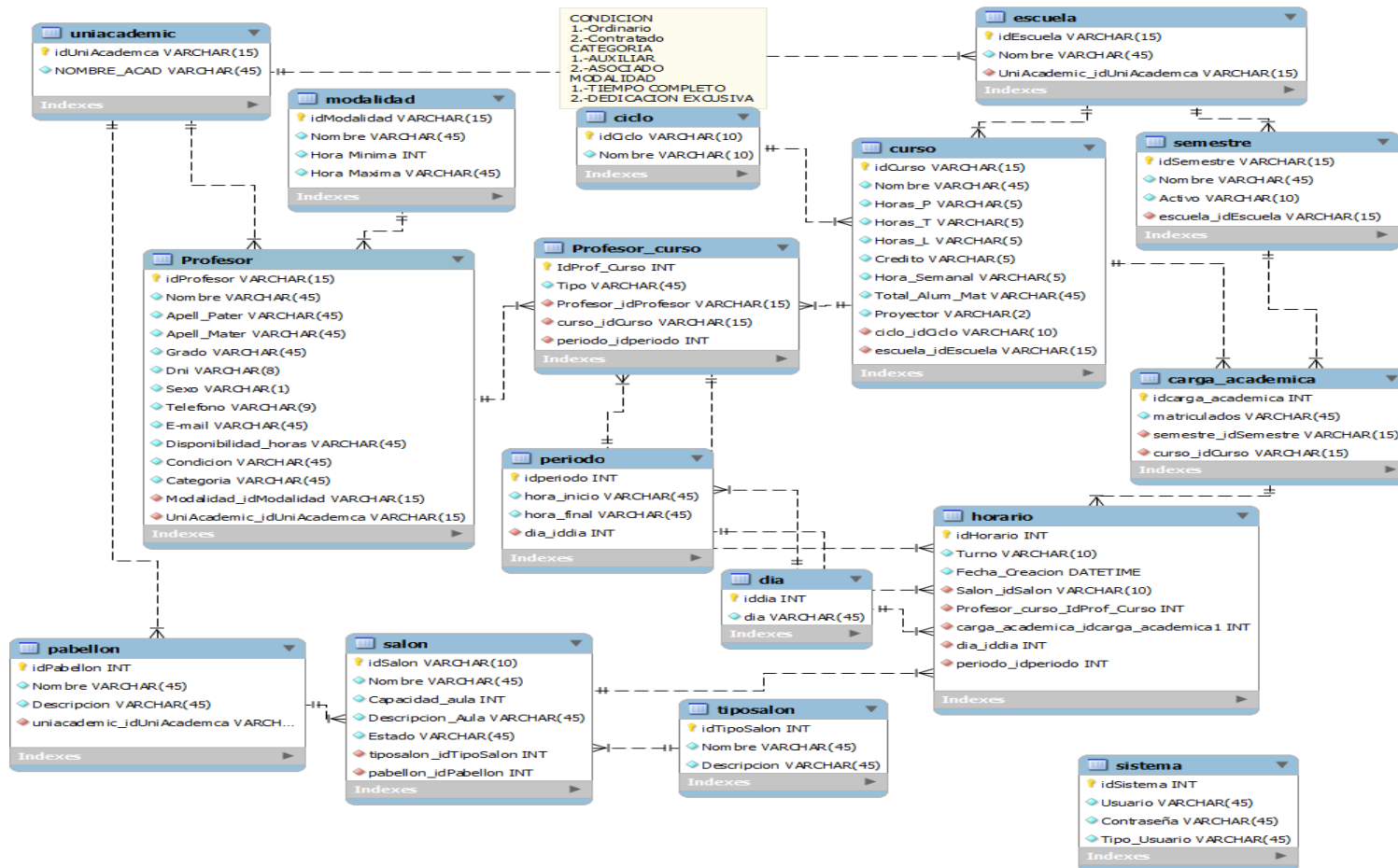


Figura 36. Diagrama de entidad relación.

Fuente: Desarrollado por el tesista.

❖ **Metáfora del sistema.**

- ✓ Se plasmará la arquitectura del sistema en una “historia”.
- ✓ Se detallará la convención de nombres de forma clara para los objetos del sistema.

❖ **Tarjetas C.R.C.**

- La **Tabla 38** muestra la tarjeta CRC modulo usuario.

Tabla 38:

Tarjeta CRC modulo usuario.

MODULO USUARIO REGISTRAR	
REPONSABILIDADES	COLABORACIONES
Insertar eliminar Mostrar Buscar Nuevo Cancelar	

Fuente: Desarrollado por el tesista.

- La **Tabla 38** muestra la tarjeta CRC modulo curso.

Tabla 39:

Tarjeta CRC modulo usuario.

MODULO CURSO REGISTRAR	
REPONSABILIDADES	COLABORACIONES
Insertar, eliminar Nuevo, Cancelar Mostrar Buscar	Ciclo Escuela

Fuente: Desarrollado por el tesista.

- La **Tabla 40** muestra la tarjeta CRC modulo docente.

Tabla 40:

Tarjeta CRC modulo docente.

MODULO DOCENTE REGISTRAR	
REPONSABILIDADES	COLABORACIONES
Insertar eliminar Buscar	Unidad Académica Modalidad

Nuevo	
Cancelar	

Fuente: Desarrollado por el tesista

- o La **Tabla 41** muestra la tarjeta CRC módulo Horario.

Tabla 41:

Tarjeta CRC modulo horario.

MODULO HORARIO REGISTRAR	
REPONSABILIDADES	COLABORACIONES
Exportar a Excel	
Mostrar	
Buscar	
Imprimir	

Fuente: Desarrollado por el tesista.

- o La **Tabla 42** muestra la tarjeta CRC módulo salón.

Tabla 42:

Tarjeta CRC módulo salón.

MODULO SALON REGISTRAR	
REPONSABILIDADES	COLABORACIONES
Insertar	
eliminar	
Mostrar	Pabellón
Buscar	Tipo Salón
Nuevo	
Cancelar	

Fuente: Desarrollado por el tesista.

- o La **Tabla 44** muestra la tarjeta CRC módulo ciclo

Tabla 43:

Tarjeta CRC modulo usuario.

MODULO CICLO REGISTRAR	
REPONSABILIDADES	COLABORACIONES
Insertar	
eliminar	
Mostrar	
Buscar	
Nuevo	
Cancelar	

Fuente: Desarrollado por el tesista.

- o La **Tabla 44** muestra la tarjeta CRC modulo semestre

Tabla 44:

Tarjeta CRC modulo semestre.

MODULO SEMESTRE REGISTRAR	
REPONSABILIDADES	COLABORACIONES
Insertar eliminar Mostrar Buscar Nuevo Cancelar	Unidad académica

Fuente: Desarrollado por el tesista.

- o La **Tabla 45** muestra la tarjeta CRC modulo usuario para mantenimiento.

Tabla 45:

Tarjeta CRC modulo usuario – mantenimiento.

MODULO USUARIO MANTENIMIENTO	
REPONSABILIDADES	COLABORACIONES
Modificar Eliminar Buscar Mostrar Nuevo Cancelar	

Fuente: Desarrollado por el tesista.

- o La **Tabla 46** muestra la tarjeta CRC modulo curso para mantenimiento.

Tabla 46:

Tarjeta CRC modulo curso – mantenimiento.

MODULO CURSO MANTENIMIENTO	
REPONSABILIDADES	COLABORACIONES
Modificar Eliminar Buscar Mostrar Nuevo Cancelar	Ciclo Escuela

Fuente: Desarrollado por el tesista.

- o La **Tabla 47** muestra la tarjeta CRC modulo docente para mantenimiento

Tabla 47:

Tarjeta CRC modulo docente – mantenimiento.

MODULO DOCENTE MANTENIMIENTO	
REPOSABILIDADES	COLABORACIONES
Modificar Eliminar Buscar Mostrar Nuevo Cancelar	Unidad Académica Modalidad

Fuente: Desarrollado por el tesista.

- o La **Tabla 48** muestra la tarjeta CRC módulo horario – mantenimiento

Tabla 48:

Tarjeta CRC modulo horario – mantenimiento.

MODULO SALON MANENIMIENTO	
REPOSABILIDADES	COLABORACIONES
Exportar a Excel Buscar Mostrar imprimir salir	

Fuente: Desarrollado por el tesista.

- o La **Tabla 49** muestra la tarjeta CRC módulo salón para mantenimiento.

Tabla 49:

Tarjeta CRC módulo salón – mantenimiento.

MODULO SALON MANTENIMIENTO	
REPOSABILIDADES	COLABORACIONES
Modificar Eliminar Buscar Mostrar Nuevo Cancelar	Pabellón Tipo Salón

Fuente: Desarrollado por el tesista

- o La **Tabla 50** muestra la tarjeta CRC Módulo ciclo para Mantenimiento.

Tabla 50:
Tarjeta CRC modulo ciclo – Mantenimiento.

MODULO CICLO MANTENIMEINTO	
REPONSABILIDADES	COLABORACIONES
Modificar Eliminar Buscar Mostrar Nuevo Cancelar	

Fuente: Desarrollado por el tesista.

- o La **Tabla 51** muestra la Tarjeta CRC modulo semestre para mantenimiento.

Tabla 51:
Tarjeta CRC modulo semestre – Mantenimiento.

MODULO SEMESTRE MANTENIMIENTO	
REPONSABILIDADES	COLABORACIONES
Modificar Eliminar Buscar Mostrar Nuevo Cancelar	Unidad académica

Fuente: Desarrollado por el tesista.

3.2.4. Codificación.

En esta etapa se desarrollará el sistema de generación de horarios con la técnica de algoritmos genéticos, en lenguaje de programación Java IDE Neatbeanst de algoritmo genético y se desarrollar en capas Modelo, Lógico y Vista (MLV) la estimación del tiempo es mayor en esta etapa.

El proyecto está programado de acuerdo a los estándares y filosofía de JAVA, donde podemos visualizar a detalla en los siguiente **Figura 38**.

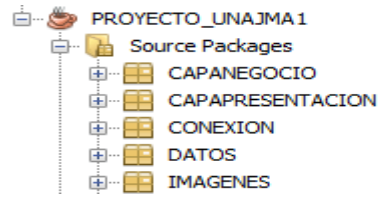


Figura 38. Presentación de Modelo Vista Controlador (M.V.L.).
Fuente: Elaboración de tesista.

- **Capa de negocio.**

En esta capa se crea las clases para: Curso, Profesor, Salón, Ciclo, Usuario, Entre otros. Donde se puede mencionar que se crea y/o inserta getter y setter (get y set), mostramos en la siguiente **Figura 29**.

```

public class SALON {
    private String Idsalon="", nombre, Descripcion_Aula,Disponibilidad, Estado;
    private int TipoSalon_idTipoSalon,Pabellon_idPabellon, Capacidad_aula;
public SALON(){
}
private OPERADORSALON opsalon= new OPERADORSALON();
public String getIdsalon() {
    return Idsalon;
}
public void setIdsalon(String Idsalon) {
    this.Idsalon = Idsalon;
}
public String getNombre() {
    return nombre;
}
public void setNombre(String nombre) {
    this.nombre = nombre;
}
public String getDescripcion_Aula() {
    return Descripcion_Aula;
}
public void setDescripcion_Aula(String Descripcion_Aula) {
    this.Descripcion_Aula = Descripcion_Aula;
}
public String getDisponibilidad() {
    return Disponibilidad;
}
public void setDisponibilidad(String Disponibilidad) {
    this.Disponibilidad = Disponibilidad;
}
}

```

Figura 39. Proceso de programación de modelo de negocio
Fuente: Desarrollado por el tesista.

- **Conexión.**

Esta capa hace que se conecten con el gestor de base de datos ver la **Figura 40**.

```
public OPERADOR() throws Exception{
    Class.forName("org.gjt.mm.mysql.Driver");
    conectar= DriverManager.getConnection("jdbc:mysql://localhost/proyecto","root","");
}

public static OPERADOR getInstance()throws Exception{
    if(instase == null){
        instase = new OPERADOR();
    }
    return instase;
}

public void finaliza()throws Exception{
    if(conectar.isClosed()){
        conectar=null;
        conectar.isClosed();
    }
}

public Connection getConnection(){
    return conectar;
}
```

Figura 40. Proceso de programación de modelo de conexión de datos.

Fuente: Desarrollado por el tesista.

- **Datos.**

En esta capa se crea el operador netamente para las consultas con la base de datos como: Operador Curso, Operador Profesor, Operador Salón, Operador Ciclo, Operador Usuario, Entre otros como se puede ver la **Figura 41**.

```

public void InsertarCurso(CURSO Ccurso) throws Exception{
    conexion=OPERADOR.getInstance().getConnection();
    PreparedStatement insertar=conexion.prepareStatement("INSERT INTO curso VALUES('"+Ccurso.getIdnombre()+"',
    insertar.executeUpdate();
}

public ResultSet cargarComboCurso(String dato) throws Exception{
    conexion=OPERADOR.getInstance().getConnection();
    String sql= "SELECT * FROM curso where Ciclo_idCiclo='"+dato+"'";
    PreparedStatement rsl=conexion.prepareStatement(sql);
    ResultSet rsls=rsl.executeQuery();
    return rsls;
}

public ResultSet cargarDatosCurso(String dato) throws Exception{
    conexion=OPERADOR.getInstance().getConnection();
    String sql= "SELECT * FROM curso where idCurso='"+dato+"'";
    PreparedStatement rsl=conexion.prepareStatement(sql);
    ResultSet rsls=rsl.executeQuery();
    return rsls;
}

public void borraCurso(CURSO salon) throws SQLException, Exception {
    conexion=OPERADOR.getInstance().getConnection();
    PreparedStatement consultaBorrar = conexion.prepareStatement("DELETE FROM curso WHERE NOMBRE=?");
    consultaBorrar.setString(1, salon.getNombre());
    consultaBorrar.executeUpdate();
    //conexion.close();
}

public ArrayList<CURSO> listarcursos(String valor) throws SQLException, Exception{
    //CONECTOS CON BASE DE DATOS
    conexion=OPERADOR.getInstance().getConnection();

    PreparedStatement consulta = conexion.prepareStatement("SELECT * FROM curso WHERE ciclo idCiclo LIKE '%"+v

```

Figura 41. Proceso de programación de modelo de conexión de datos.

Fuente: Desarrollado por el tesista.

- **Capa de presentación**

En esta capa el cliente interactúa con el software, se crea aquí las Interfaz de: crear nuevo docente, salón, curso, usuario, también Mantenimiento y para asignar datos para el horario ver la **Figura 42** y **Figura 43**.

➤ **Nuevo docente.**

The image shows a software window titled "NUEVO DOCENTE" with a standard Windows-style title bar. The window is divided into three main sections: "DATOS PERSONALES", "DATOS LABORALES", and "CLASES".

- DATOS PERSONALES:** This section contains several input fields and dropdown menus. It includes fields for "CODIGO", "DNI", "GRADO" (with a dropdown menu showing "Ing."), "TELEFONO", "NOMBRE", "EMAIL", "AP. PATERNO", "SEXO" (with a dropdown menu showing "F"), and "AP. MATERNO".
- DATOS LABORALES:** This section contains fields for "UNIDAD ACADEMICA" (dropdown), "DISP. DE HORAS", "MODALIDAD" (dropdown), "CATEGORIA" (dropdown showing "PRINCIPAL"), and "CONDICION" (dropdown showing "ORDINARIO").
- CLASES:** This section contains two fields: "PREFERENCIA DE HORAS" and "PREFERENCIA DE CURSO".

At the bottom of the window, there is a horizontal bar with four large, colorful icons: a red circle with a white diagonal line, a green plus sign, a blue floppy disk with a green arrow pointing down, and a blue circle with a white X. Each icon is followed by an ellipsis (...).

Figura 42: Resultado del diseño de capa de presentación para crear nuevo docente.

Fuente: Elaborado por el tesista.

➤ Interfaz de Mantenimiento

The screenshot shows a web interface for maintenance. At the top, there is a search filter section with the text 'Filtro de búsqueda' and 'Buscar por nombre' followed by a text input field. Below this is a grid of form fields for filtering: 'GRADO' (dropdown menu with 'Ing.' selected), 'NOMBRE:' (text input), 'TELEFONO' (text input), 'MODALIDAD' (dropdown menu), 'AP. PATERNO:' (text input), 'UNIDAD ACADEMICA:' (dropdown menu), 'SEXO:' (dropdown menu with 'F' selected), 'AP. MATERNO:' (text input), 'EMAIL:' (text input), 'CONDICION:' (dropdown menu with 'ORDINARIO' selected), 'DNI:' (text input), 'DISPON. HORAS:' (text input), and 'CATEGORIA:' (dropdown menu with 'PRINCIPAL' selected). Below the filters is a section labeled 'Resultado de búsqueda' which is currently empty. At the bottom of the interface is a horizontal toolbar with a left arrow, a trash can icon, a floppy disk icon with a pencil, a circular refresh icon, and a blue circle with a white 'X' icon, followed by an ellipsis.

Figura 43. Interfaz para procedimiento de Mantenimiento.

Fuente: Desarrollado por el tesista.

3.2.4.1. Técnicas de AG aplicado en la programación.

Para su desarrollo se preparó la técnica de algoritmo genético, como se muestra en la siguiente **Figura 44**.

➤ Proceso de algoritmo genético.

```
// Inicio Algoritmo Genetico
generarPoblacionInicial();
evaluarIndividuos();
for(int i = 0; i < ConfigGenetico.getLimiteGeneraciones(); i++){
    this.generacion++;
    reproducirIndividuos(); //
    mutarIndividuos();
    evaluarIndividuos();
    imprimirMejorIndividuo();
}
```

Figura 44. Proceso de ejecución del algoritmo genético.

Fuente: Desarrollado por el tesista. del tesista.

➤ **Reproducir Individuo.**

Como se definió en el capítulo 2 sobre individuos del algoritmo genético, se desarrolló como se muestra en la **Figura 45**.

```
private void reproducirIndividuos(){
    // Variables que usaremos
    Individuo individuo1, individuo2;
    Individuo hijo1, hijo2;
    // Reproduce la población (Con la cantidad de individuos configur
    for(int i = 0; i < poblacion.size()/2; i++){
        //Seleccionamos los padres
        individuo1 = poblacion.get(i*2);// el elemento par
        individuo2 = poblacion.get(i+1);// el siguiente elemento
        // Obtenemos los hijos
        hijo1 = individuo1.geHijo(individuo2);
        hijo2 = individuo2.geHijo(individuo1);
        // Los agregamos a la poblacion (SIGUIENTE GENERACION)
        poblacion.set(i*2, hijo1);
        poblacion.set(i+1, hijo2);
    }
    // si la poblacion tiene tamaño impar agregamos el ultimo elemen
    // a la nueva generación
}
```

Figura 45. Programación de reproducir individuo.

Fuente: Desarrollado por el tesista.

➤ **Mutar Individuo.**

En esta **Figura 46** se observa el objeto mutar individuo, el individuo mutara de acuerdo a sus características.

```
private void mutarIndividuos(){
    for(int i = 0; i < poblacion.size(); i++){
        poblacion.get(i).mutar();
    }
}
```

Figura 46. Codificación del individuo mutar.

Fuente: Desarrollado por el tesista.

➤ **Evaluar Individuo.**

En esta **Figura 47** se muestra como se evalúa a cada individuo.

```
private void evaluarIndividuos(){
    int i = 0;
    Individuo otroIndividuo;
    if(mejorIndividuo == null){
        mejorIndividuo = poblacion.get(0);
        mejorIndividuo.evaluate();
        i = 1;
    }
    for(; i < poblacion.size(); i++){// desde el segundo elemento
        otroIndividuo = poblacion.get(i);
        otroIndividuo.evaluate();
        if(mejorIndividuo.getPonderado() > otroIndividuo.getPonderado()){
            mejorIndividuo = otroIndividuo;
        }
    }
}
```

Figura 47. Código de evaluar individuo.

Fuente: Desarrollado por el tesista.

➤ **Imprime Población.**

En la **Figura 48** se muestra para imprimir en consola la población

```
public void imprimirPoblacion(){
    // IMPRIMIMOS LA POBLACIÓN
    System.out.println("IMPRIMIENDO POBLACIÓN GENERACION: "+this.generacion+"\n");
    for(int i = 0; i < poblacion.size(); i++){
        System.out.println("*****INDIVIDUO "+i+"*****");
        System.out.println("* PUNDTUACION: "+poblacion.get(i).getPonderado());
        poblacion.get(i).imprimir();
    }
    System.out.println("FIN IMPRESIÓN POBLACIÓN\n");
}
```

Figura 48. Código de imprimir población.

Fuente: Desarrollado por el tesista.

➤ **Imprime mejor individuo.**

La **Figura 49** muestra al mejor individuo de acuerdo a su ponderación.

```

public void imprimirMejorIndividuo(){
    // IMPRIMIMOS EL MEJOR INDIVIDUO
    System.out.println("#####[ EL MEJOR INDIVIDUO]#####");
    System.out.println("#[ PUNTUACION]+"mejorIndividuo.getPonderado());
    //mejorIndividuo.imprimir();
    System.out.println("#####[ FIN MEJOR INDIVIDUO]#####");
}

```

Figura 49. Imprime el mejor individuo.

Fuente: Desarrollado por el tesista.

- Esta **Figura 50** es el resultado final, donde generamos las codificaciones para la población de individuos.

Figura 50. Ingresar datos del algoritmo genético.

Fuente: Desarrollado por el tesista.

- En la **Figura 51** se observa el resultado final de la lógica del algoritmo genético por ciclos el horario.

Mostrando Horarios

Ciclo 1

Horas	lunes	martes	miercoles	jueves	viernes	sab...	do...
7:00am	IAB13 PR0004 SA0004	SA0004	IAB11 SA0004	IAB11 SA0004	SA0004		
8:00am	IAB14 SA0004	IAB12 SA0004	IAB11 SA0004	IAB12 SA0004	IAB11 SA0004		
9:00am	IAB12 SA0004	IAB12 SA0004	IAB16 SA0004	IAB13 PR0004 SA0004	IAB14 SA0004		
10:00am	IAB12 SA0004	IAB13 PR0004 SA0004	IAB12 SA0004	SA0004	IAB14 SA0004		
11:00am	IAB16 SA0004	IAB13 PR0004 SA0004	IAB13 PR0004 SA0004	SA0004	IAB16 SA0004		
12:00am	SA0004	IAB13 PR0004 SA0004	SA0004	SA0004	IAB16 SA0004		

Figura 51. Resultado del horario del Ciclo I.

Fuente: Desarrollado por el tesista.

- En la **Figura 52** se observa el resultado final de la lógica del algoritmo del ciclo II.

Mostrando Horarios

Ciclo 2

Horas	lunes	martes	miercoles	jueves	viernes	sab...	do...
7:00am	IAB26 SA0002	IAB23 PR0004 SA0002	IAB26 SA0002	IAB21 SA0002	IAB21 SA0002		
8:00am	IAB21 SA0002	IAB23 PR0004 SA0002	IAB27 SA0002	IAB22 SA0002	IAB22 SA0002		
9:00am	IAB27 SA0002	IAB22 SA0002	IAB22 SA0002	IAB21 SA0002	IAB27 SA0002		
10:00am	IAB27 SA0002	IAB22 SA0002	IAB27 SA0002	IAB23 PR0004 SA0002	IAB23 PR0004 ...		
11:00am	IAB23 PR0004 SA0002	IAB23 PR0004 SA0002	SA0002	SA0002	SA0002		
12:00am	SA0002	SA0002	SA0002	SA0002	SA0002		

Figura 52. Resultado del horario del Ciclo II.

Fuente: Desarrollado por el tesista.

- En la **Figura 53** se observa el resultado final de la lógica del algoritmo por ciclo III.

Mostrando Horarios

Ciclo 3

Horas	lunes	martes	miercoles	jueves	viernes	sab...	do...
7:00am	IAB31 PR0001 SA0001	IAB37 SA0001	IAB33 PR0001 SA0001	IAB37 SA0001	IAB36 SA0001		
8:00am	IAB31 PR0001 SA0001	IAB37 SA0001	IAB31 PR0001 SA0001	IAB37 SA0001	IAB36 SA0001		
9:00am	IAB31 PR0001 SA0001	IAB37 SA0001	IAB35 SA0001	IAB33 PR0001 SA0001	IAB36 SA0001		
10:00am	IAB34 SA0001	IAB36 SA0001	IAB34 SA0001	IAB33 PR0001 SA0001	IAB35 SA0001		
11:00am	IAB34 SA0001	IAB35 SA0001	IAB34 SA0001	IAB35 SA0001	IAB37 SA0001		
12:00am	IAB33 PR0001 SA0001	SA0001	SA0001	SA0001	SA0001		

Figura 53. Resultado del horario del Ciclo III.

Fuente: Desarrollado por el tesista.

3.2.5. Pruebas

Se realizará prueba a lo largo del desarrollo del proyecto con framework JUnit 3.8.1, con el fin de asegurar en todo momento la realización de lo planteado en el diseño en este proceso no solo participará el equipo de desarrollo, también participará la Escuela Profesional ingeniería de sistemas, sobre todo en las pruebas de aceptación.

Cabe señalar que el diseño de pruebas se realizar para todas las partes del sistema como una práctica para garantizar el buen funcionamiento.

Prueba Unitaria

- Las pruebas serán escritas ante que los métodos de desarrollo
- Se creará las unidades de prueba con JUnit, una herramienta de Java especializada para la creación de unidades de prueba.

Prueba de aceptación

- Se deberá diseñar las pruebas de aceptación con base en las historias de usuarios.
- Cuando se encuentra un error
- Al encontrar un error debe escribirse primero la prueba antes que corregirlo.

3.3. Representación del problema genético – con Gráfico

Para utilizar algoritmo genético, es necesario definir una población inicial y ésta será un conjunto de cromosomas. Será representado como una matriz tridimensional, cuyos ejes coordinados representan, los 5 días de las semanas (eje x), los siete (07) periodos de clases (eje y) y los diez (10) semestres de la Escuela Profesional Ingeniería de Sistemas (eje z) como se muestra en la **Figura 54**.

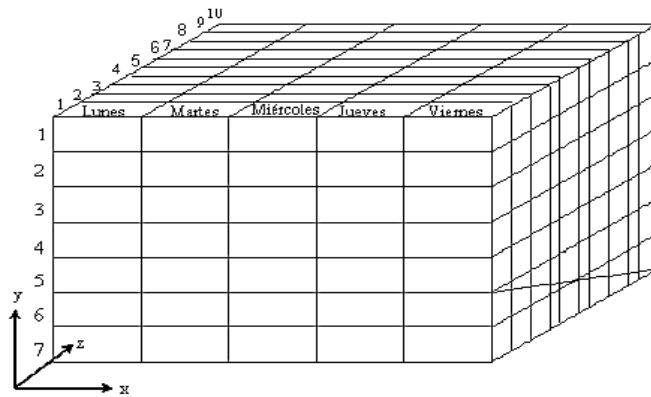


Figura 54. Representación cromosomas para el problema de horarios de UNAJMA.

Fuente: Desarrollado por el Tesista.

- En la **Figura 55** se muestra la representación de la población Inicial por individuos para caso del problema.

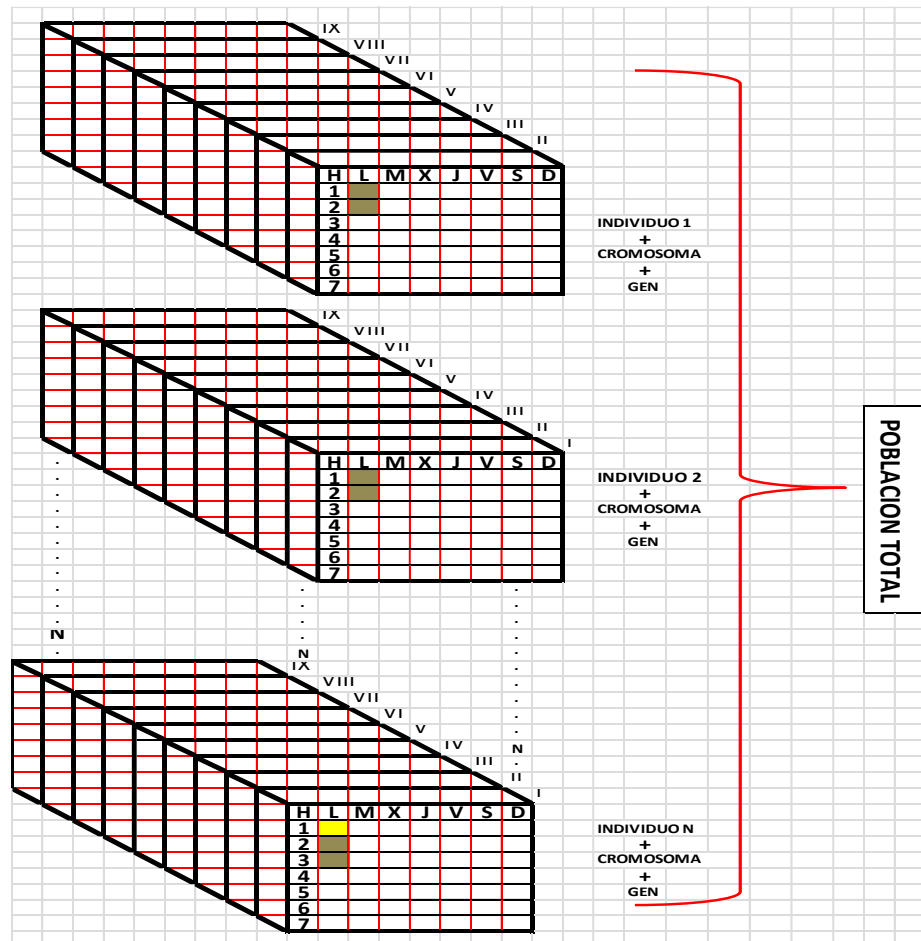


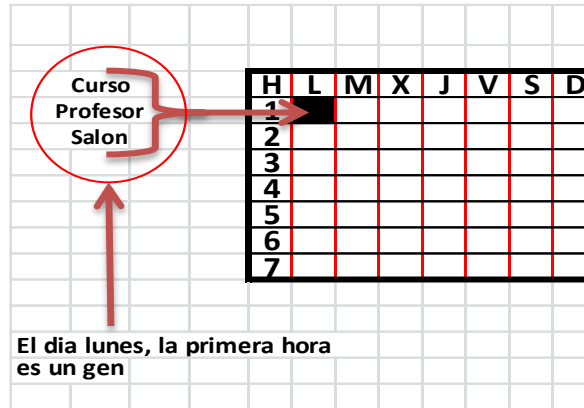
Figura 55. Representación de la población inicial para el problema de horarios de UNAJMA.

Fuente: Desarrollado por el tesista.

- En la Tabla 52 se muestra de cómo trabaja un cromosoma en el proyecto.

Tabla 52:

Representación del Gen en un cromosoma para el problema de horarios de EPIS.



Nota: Desarrollado por el tesista.

En el caso del gen, este será definido como el valor que contiene cada elemento de la matriz se puede ver en **Tabla 53** la representación de un gen.

Tabla 53:

Representación del gen para el problema de horarios de EPIS.

Profesor	Salón	Asignatura
----------	-------	------------

Nota: Desarrollado por el tesista.

Las variables profesor, salón y asignatura sus valores correspondientes, es decir, la variable profesor tendrá el nombre del profesor que imparte la asignatura, la variable salón el número de salón en la que se imparte la asignatura.

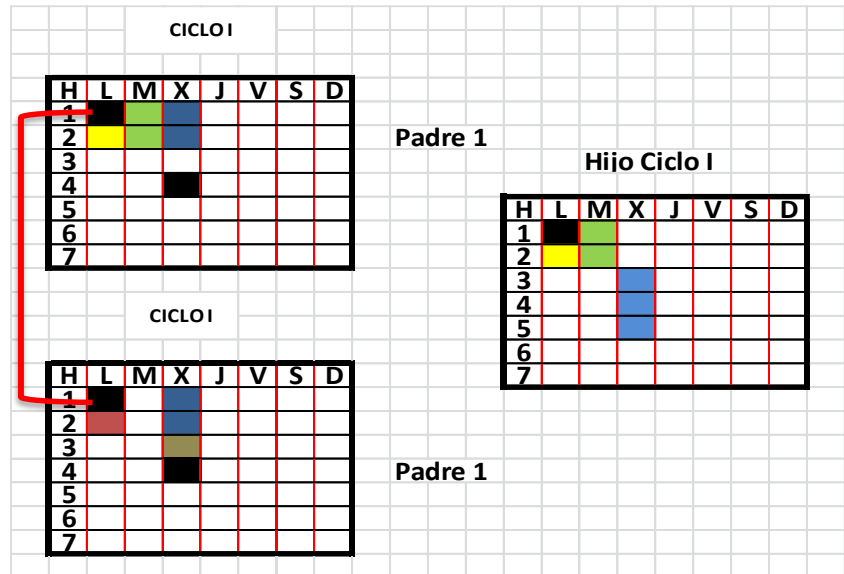
3.3.4. Operadores Genéticos.

- **Operador de cruzamiento.**

El cruce entre padres debe tratar de que sus descendientes posean la información genética más fuerte. Para esto, debe evitar las colisiones ver en la **Tabla 54**.

Tabla 54:

Representación del operador cruzamiento para el problema de horarios de EPIS.



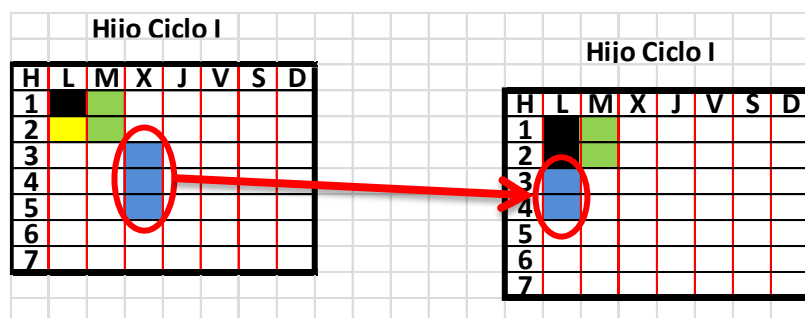
Nota: Desarrollado por el tesista.

➤ **Operador de mutación.**

El operador de mutación simplemente sigue la idea de cambiar un gen por otro mejor. El gen que mutara es seleccionado de formar aleatoria cuando hay una colisión esto ocurre, caso contrario si es el mejor no muta ver en la **Tabla 55**.

Tabla 55:

Representación del operador de mutación para el problema de horarios de EPIS.



Nota: Desarrollado por el tesista.

3.3.5. Estándares de desarrollo de software.

Se desarrolló con la ISO/IEC 9126.

3.3.6. Herramientas de desarrollo de software.

Las herramientas tecnológicas que se va utilizar para su desarrollo del proyecto son las siguientes:

Herramientas de desarrollo para diagramas.

- ✓ StarUML
- ✓ Rational Rose Enterprise.
- ✓ Erwin data Modeler 7.3 (Modelo Entidad Relacional).

Herramientas para desarrollar la programación.

- ✓ Java.
- ✓ Base de dato MySQL.

Recursos de software empleados.

- ✓ Ms Project 2013.
- ✓ Microsoft Excel 2013.
- ✓ Microsoft Word 2013.

CAPÍTULO 4 RESULTADOS Y DISCUSIONES

4.1. Valoración de la solución.

Para comparar el desempeño del algoritmo respecto a otro software de generación de horarios, primero se aplicará los tiempos de ejecución.

4.1.1. Primeros casos de análisis.

Contaremos con dos casos de análisis diferentes para esta sección, ambos se basarán en la información proporcionada que corresponde al ciclo 2015-II de la Escuela Profesional de Ingeniería de Sistemas, la cual fue elaborada manualmente por el personal responsable.

Los casos de análisis son:

- ❖ El primer caso de análisis, tomara el 40% de información necesaria para la generación del horario académico de 1ro ciclo a 5to ciclo académico.

A continuación, detallamos las informaciones tomadas para ver la solución:

- Se cuenta con un total de 20 cursos hasta 4to ciclo, se incorporó sus respectivos datos a la base de datos de software.
- Se cuenta con 10 profesores con sus respectivos datos en la base de datos del software.
- Se cuenta con 10 salones.

Con estos datos básicos y necesarios se comprobó en cuatro tiempos los porcentajes de solución.

- En la **Tabla 55** se puede ver la primera prueba del caso.

Tabla 55:

Primera prueba con el AG.

Nombre	N°	M. Error	Tiempo
Tamaño de población	10	70%	5 s
Límite de generación	10		5 s
Mejor individuo	515		

Nota: Desarrollado por el tesista.

- En la **Tabla 56** se puede ver la segunda prueba con los datos reales.

Tabla 56:

Segunda prueba con el AG.

Nombre	N°	M. Error	Tiempo
Tamaño de población	100	60%	12 s
Límite de generación	100		12 s
Mejor individuo	430		

Nota: Desarrollado por el tesista.

- En la **Tabla 57** se puede ver la tercera prueba de caso con los datos reales.

Tabla 57:

Tercera prueba con el AG.

Nombre	N°	M. Error	Tiempo
Tamaño de población	500	30%	30 s
Límite de generación	500		30 s
Mejor individuo	400		

Nota: Desarrollado por el tesista.

- En la **Tabla 58** se puede ver la cuarta prueba con los datos reales.

Tabla 58:

Cuarta prueba con el AG.

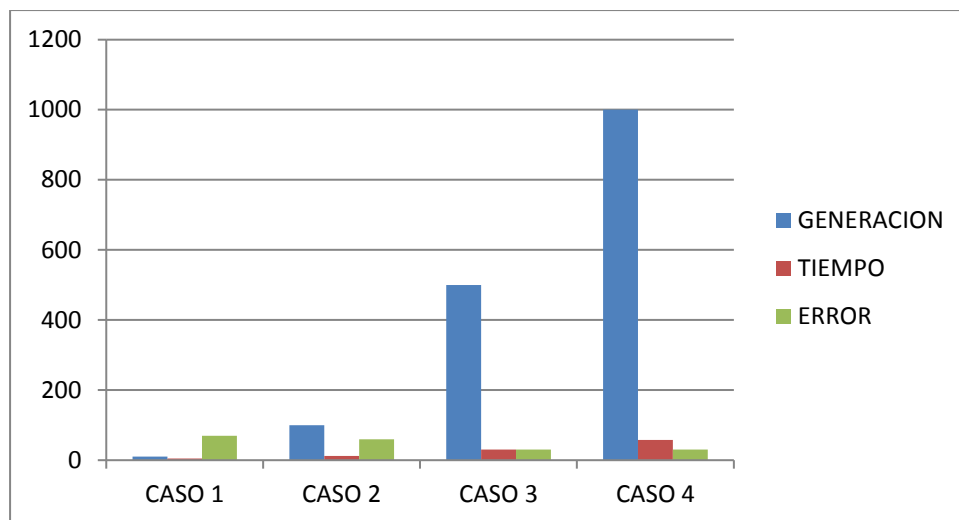
Nombre	N°	M. Error	Tiempo
Tamaño de población	1000	30%	58 s
Límite de generación	1000		58s
Mejor individuo	400		

Nota: Desarrollado por el tesista.

De acuerdo al primer análisis determinado con 40% de información, con distintos tamaños de generación se probó y se precisa que la mejor prueba fue con los datos de la tercera prueba, el mejor individuo con su operación de aptitud mínima de 400 que es el mejor individuo de todos para dar una buena solución al problema, en la siguiente **Tabla 59** se muestra los 4 casos.

Tabla 59:

Cuadro de comparación de los casos con AG.



Nota: Desarrollado por el tesista.

4.1.2. Segundo caso de análisis.

- ❖ El segundo caso de análisis, tomara el 90 % de información necesaria para la generación del horario académico de 1ro ciclo a 10mo ciclo académico.

A continuación, detallamos las informaciones tomadas para ver la solución:

- Se cuenta con un total de 62 cursos hasta 10mo ciclo, se incorporó sus respectivos datos a la base de datos de software.
- Se cuenta con 19 profesores con sus respectivos datos en la base de datos del software.
- Se cuenta con 10 salones

Con estos datos básicos y necesarios se comprobó en cuatro tiempos los porcentajes de solución.

- En la **Tabla 60** se puede ver la primera prueba con los datos reales.

Tabla 60:

Primera prueba con el AG con datos al 90%.

Nombre	N°	M. Error	Tiempo
Tamaño de población	10	70%	7 s
Límite de generación	10		7 s
Mejor individuo	485		

Nota: Desarrollado por el tesista.

- En la **Tabla 61** se puede ver la segunda prueba con los datos reales.

Tabla 61:

Segunda prueba con el AG con datos al 90%.

Nombre	N°	M. Error	Tiempo
Tamaño de población	100	60%	12 s
Límite de generación	100		12 s
Mejor individuo	485		

Nota: Desarrollado por el tesista.

- En la **Tabla 62** se puede ver la tercera prueba con datos reales

Tabla 62:

Tercera prueba con el AG con datos al 90%.

Nombre	N°	M. Error	Tiempo
Tamaño de población	500	30%	40 s
Límite de generación	500		40 s
Mejor individuo	490		

Fuente: Desarrollado por el tesista.

- En la **Tabla 56** se puede ver la cuarta prueba con los datos reales.

Tabla 63:

Cuarta prueba con el AG con datos al 90%.

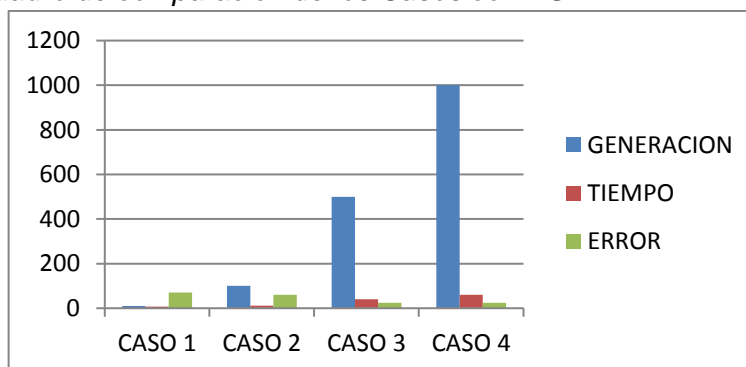
Nombre	N°	M. Error	Tiempo
Tamaño de población	1000	30%	58 s
Límite de generación	500		58s
Mejor individuo	460		

Nota: Desarrollado por el tesista.

El segundo caso de análisis se probó con un total de datos reales recuperados en etapa de análisis y se puede ver en resumen los resultados en la **Tabla 64**.

Tabla 64:

Cuadro de comparación de los Casos con AG.



Nota: Desarrollado por el tesista.

4.2. Evaluación y comparación de resultados.

Al inicio del proyecto de tesis se propuso una solución a través de un algoritmo genético, al problema de generación de horarios para la Facultad de Ingeniería de Sistemas de la Universidad Nacional José María Arguedas. La evaluación que se realizó al algoritmo genético fue en dos momentos donde se demuestra que el tiempo de ejecución no varía por la cantidad de datos, la variación es mínima, quiere decir que el algoritmo genético puede trabajar en 99% normal con más datos para su generación.

De acuerdo a las tablas de comparación para el mejor individuo se debe trabajar con tamaño de población máximo 1000 y tamaño de generación máximo 500 por que nos da al mejor individuo/ponderación esto hace que tenga menos cruces de horarios y sea el más factible.

La primera prueba se realizó con los datos del semestre 2015-II, el segundo con los datos del semestre 2016-I y el ultimo se realizó con los datos del semestre 2016-II, donde en el semestre 2016-II se trabajó con 62 cursos, 18 docentes de los cuales 12 nombrados y 7 contratados las asignaciones se dieron a 95% a los docentes el resultado se puede ver en el anexo1.

En el mercado existe software genérico (empaquetados) de horario uno de ellos es Asc Horarios que es bastante aceptado por los clientes, de acuerdo al uso que se dio en las pruebas puedo indicar que, es un software privado, no exactamente satisface las necesidades para una universidad, más la satisfacción seria para una institución educativa primaria y secundaria.

El sistema de generación de horario con algoritmo genético es muy distinto en su procesamiento de datos, a otros horarios que existen en el mercado donde los otros están desarrollados con programación lineal no heurística, mientras que el software genético tiene programación heurística, su proceso de generación es igual a los seres vivos.

CONCLUSIÓN.

- A partir de la investigación tecnológica se concluye que el desarrollo del sistema de generación de horarios con algoritmos genéticos, fue un trabajo complejo debido a la gran cantidad de parámetros, se trató de cumplir con la mayoría de las restricciones planteadas de acuerdo a la información requerida de la escuela profesional ingeniería de sistemas, donde se probó con datos en 40% y 90% de información, el resultado esperado fue en un tiempo menos esperado así logrando los objetivos de optimizar el tiempo y los cruces de cursos. Dos parámetros importantes al momento de asignar de acuerdo a los datos recopilados se debe respetar.
 - Los docentes Nombrados pueden elegir el día y la hora de acuerdo al orden de la categoría que cada uno tiene.
 - Los docentes contratados no pueden escoger los días ni las horas, se les asignara automáticamente con el sistema.

- Para diseñar y construir se tomó las restricciones planteadas y los diagramas correspondientes de la Escuela Profesional Ingeniería de Sistemas. Para este proyecto se ha tratado de abarcar la mayoría de condiciones que existen actualmente en la Escuela Profesional de Ingeniería de Sistemas, La construcción del cromosoma para este problema no ha sido una tarea fácil ya que existen diversas relaciones que se tienen que tomar en cuenta para armar complemente un horario y todas estas deben estar expresadas en el cromosoma. Inicialmente se había pensado en implementar un cromosoma binario, pero debido a la gran cantidad de relaciones la manipulación de este hubiera sido compleja y se implementó con cromosoma no binaria.

- Se pudo observar con debilidad el trabajo del método de algoritmo genético, de acuerdo a los resultados obtenidos de los dos casos de análisis del proyecto, no garantizó una buena solución al problema, porque existe deficiencia al generar debido a que no se cumplió con toda las necesidades o restricciones definidas al problema, se recomienda que se debe cumplir toda las restricciones y necesidades de la institución para garantizar el mejor resultado del algoritmo genético y llegar a una buena conclusión sobre el proceso del algoritmo genético. La deficiencia del horario es: No puede

dividir en dos grupos por la cantidad de alumnos, No puede asignar los salones y laboratorios de acuerdo a las características que presenta, pero el horario genera sin estas dos restricciones y es favorable los resultados.

RECOMENDACIÓN.

- Para garantizar la seguridad de la información, se recomienda la manipulación del personal autorizado.
- Se deberá realizar capacitación en el manejo a la persona responsable, para el manejo del software.
- Se recomienda, en el futuro aplicar y mejorar este proyecto, para un mejor resultado.
- Se recomienda una investigación más exhausta para llegar al 100% de resultado y sacar mejor el provecho al algoritmo genético.

REFERENCIAS BIBLIOGRÁFICAS

- Abramson. (1991). "**La construcción de los horarios escolares Uso de recocido simulado: Secuencial y algoritmos paralelas**", Management Science, PWS Publishing, Boston, 1998.
- Alsuwaiyel, M. (1998). "**Algorithms: Design Techniques and Analysis**" - fuente recuperado: <https://www.abebooks.co.uk/servlet/BookDetailsPL?bi=10820649130&searchurl=isbn%3D9810237405%26sortby%3D17>
- Andrew, L. (2002). "**UTTSExam: A Campus-Wide University Exam-Timetabling System**". American Association for Artificial Intelligence. Singapore. <https://www.aaai.org/Papers/AAAI/2002/AAAI02-125.pdf>
- Ángeles, A.N. (2015). "**Elaboración de una solución metaheurística usando un algoritmo genético que permita elaborar la distribución de los horarios académicos-Problemas NP**" publicado en su proyecto de tesis PUCP. Fuente recuperado: <http://tesis.pucp.edu.pe/repositorio/handle/123456789/6057>
- Aoron, R. (2012). "**investigación operativa**", fuente recuperada: http://invope2arl.blogspot.com/2012/04/Programación-entera_09.html
- Blum, C., y Roli, A. (2003). "**Meta-heurísticas de optimización combinatoria: Descripción y comparación conceptual**". ACM Computing Surveys. New York. Vol 35 Issue 3.
- Burke, (2006). "**basado en casos selección heurística para los problemas de horarios**". Diario de Programación. REINO UNIDO. Volumen 9, Número 2. fuente recuperado: <http://link.springer.com/journal/10951/9/2/page/1#page-1>
- Burke, Edmund, Newall y Jim (2004). "**Solución de problemas de horarios de examen a través de la adaptación de los ordenamientos heurísticos**". Annals of Operational Research. Netherlands, vol 129.

- Canek, A. (2008). Publica en un artículo sobre “**Calculo diferencial e integral**”
“fuente recuperada: <http://es.slideshare.net/pascualonofre/calculo-diferencial-42683193>
- Danzig, G. (1951). matemático estadounidense donde resuelve “**Método Simplex**”
–fuente recuperada: <https://sites.google.com/site/agricolausil0/home/4---contenido/4-1---geometria-analitica/4-1-3---introduccion-a-la-Programación-lineal>
- Duong, T., y Lam, K. (2004). “**Combinación de programación de restricciones y recocido simulado en el horario de exámenes universitario** “*La II Conferencia Internacional de Ciencias de la Computación.*” Fuente recuperada: <https://www.coursehero.com/file/p70ou8f/Duong-and-Lam-2004-also-employed-constraint-programming-to-generate-initial/>
- EcuRed. (2015). “**Algoritmo mimético** “. Página oficial de Ecured. Fuente recuperada. http://www.ecured.cu/index.php/Algoritmo_mem%C3%A9tico
- Enciclopedia de Economía. (2009). cita en el artículo “**la gran enciclopedia económica sobre la Optimización**”, fuente: <http://www.economia48.com/spa/d/optimizacion/optimizacion.htm>
- Foro Económico Mundial. (2015). Según su portal principal define “**Informe Mundial sobre tecnología de la Información**” fuente recuperada: <https://www.weforum.org/es/agenda/2015/04/que-paises-son-lideres-en-tecnologia-digital/>
- Gasparo, L., y Schaerf, A. (2001). “**Técnicas de búsqueda tabú para los horarios de examen**”. La práctica y la teoría del Automatizado de Horarios. la tercera Conferencia Internacional. Springer Lecture Notes in Computer Science, vol. 2079.
- Gesthor B. (2015). “**Gestión de tiempos y horarios del personal**”: fuente: <http://www.hdasoft.com/gesthor.html>

Glover, F., y Melian, B. (200). "**Inteligencia artificial, revista iberoamericana de inteligencia artificial**". (2003) pp. 29-48 issn: 1137-3601. (www.aepia.org/revista). <http://www.redalyc.org/articulo.oa?id=92571902>

Institute of Electrical and Electronics Engineers (1992). "**ISO 9126**" recuperado de: https://es.wikipedia.org/wiki/Institute_of_Electrical_and_Electronics_Engineers

King, G., Bauer, C. (2010). **Hibernate**. Recuperado el 2015 de Agosto de 2015, de http://docs.jboss.org/hibernate/core/3.5/reference/es-ES/html_single/

La ASAP (**Automated scheduling, optimization and planing group**): Programación automatizada, la optimización y el grupo de cepillado

López. (2015). **Una solución basada en agentes al problema de generación de horarios** (Vol 2. Pag. 73-85)

Merlot, Boland y Stuckey (2003). "**Un algoritmo híbrido para el problema de horarios examen**". La práctica y la teoría del Automatizado de Horarios: de la cuarta Conferencia Internacional. Springer Lecture Notes in Computer Science, vol. 2740

Naji, A. (2004). "**Comparación de los algoritmos meta-heurísticas para el problema de horarios de examen**". Matemáticas Aplicadas y Computación, Vol.16.

NetBeans. (2015). "**Definición de NetBeans**" Fuente recuperado: <https://netbeans.org/features/java/>

Oracle. (2015). "**Definición de Mysql**" recuperado de: <https://www.mysql.com/> y <https://www.oracle.com/mysql/index.html>

- Pino, R., Martínez, C. y Villanueva, F. (2012). ***Application of GRASP methodology to Vehicle Routing Problem (VRP)***. Disponible en: <http://elrond.informatik.tu-freiberg.de/papers/WorldComp2012/ICA6018.pdf>
- Pinto, M. (2011). "***Base de datos***". Recuperado el 20 de Agosto de 2015, fuente: http://www.mariapinto.es/e-coms/bases_datos.htm
- ProgramaciónNet. (2013). Publicado sobre "***Backtracking***" fuente recuperada: <https://www.cis.upenn.edu/~matuszek/cit594-2012/Pages/backtracking.html>
- Project Management Institute. (2008). ***Project Management Body of Knowledge (PMBOK Guide. Cuarta Edición)*** fuente recuperada: www.pmi.org
- Qu, Burke, Edmund y Barry. (2006). "***Una encuesta de las metodologías de búsqueda y Enfoques automatizados para los horarios de examen***". Informe Técnico Ciencias de la Computación NOTTCS-TR-2006-4.UK.
- Quispe, R. (2007). cita en su artículo "***Algoritmo y Programación***" fuente: <http://www.rodolfoquispe.org/blog/que-es-un-algoritmo.php>
- RedCientífica. (2015). Ciencia Tecnología y Pensamiento "***Algoritmo***" fuente recuperada: http://www.redcientifica.com/gaia/prog/pseu_c.htm
- Wong, Cote y Gely. (2002). "***Final de los horarios de examen: un enfoque práctico***". IEEE Conferencia Canadiense de Ingeniería Eléctrica y Computación.

ANEXOS:

ANEXO 1: Distribución de carga del DAITI de los nombrados 2016 - II.



N°	CÓDIGO DE ASIGNATURA	CARGA LECTIVA	CONDICIÓN	CATEGORIA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO DIRECTIVO
						HT	HP	TH	
1	TRABAJO LECTIVO		ORDINARIO	PRINCIPAL	DE	HT	HP	TH	MG. CECILIA EDITH GARCIA RIVAS PLATA DIRECTORA DE LA EPIS
	IAB56	INVESTIGACION TECNOLOGICA E INOVACION				2	2	4	
	IAA84	FORMULACION Y EVALUACION DE PROYECTOS INFORMATICOS				3	2	5	
	IAA95	TESIS I				3	0	3	
TOTAL HORAS LECTIVAS								12	
N°	CÓDIGO DE ASIGNATURA	CARGA LECTIVA	CONDICIÓN	CATEGORIA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO DIRECTIVO
						HT	HP	TH	
2	TRABAJO LECTIVO		ORDINARIO	ASOCIADO	TC	HT	HP	TH	DR. ANGEL FERNANDO NAVARRO RAYMUNDO DIRECTOR DAITI
	IAA71	INGENIERIA DE SOFTWARE I				2	4	6	
	IAA91	INGENIERIA DE SOFTWARE III				2	4	6	
TOTAL HORAS LECTIVAS								12	
N°	CÓDIGO DE ASIGNATURA	CARGA LECTIVA	CONDICIÓN	CATEGORIA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO DIRECTIVO
						HT	HP	TH	
3	TRABAJO LECTIVO		ORDINARIO	ASOCIADO	DE	HT	HP	TH	M.SC. JULIO CESAR HUANCA MARIN
	IAB13	INTRODUCCION A LA INFORMATICA GRUPO B				2	4	6	
	IAB33	DISEÑO WEB				2	2	4	
	IAB34	ALGORITMO Y ESTRUCTURA DE DATOS				2	4	6	
TOTAL HORAS LECTIVAS								16	
N°	CÓDIGO DE ASIGNATURA	CARGA LECTIVA	CONDICIÓN	CATEGORIA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO DIRECTIVO
						HT	HP	TH	
4	TRABAJO LECTIVO		ORDINARIO	ASOCIADO	DE	HT	HP	TH	ING. JUAN JOSÉ ORE CERRÓN
	IAA98	TOPICOS ESPECIALES				2	2	4	
	IAA68	MULTIMEDIA				0	2	2	
	IAAA5	TESIS II				2	2	4	
	IAB43	PROGRAMACION				2	4	6	
TOTAL HORAS LECTIVAS								16	
N°	CÓDIGO DE ASIGNATURA	CARGA LECTIVA	CONDICIÓN	CATEGORIA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO DIRECTIVO
						HT	HP	TH	
5	TRABAJO LECTIVO		ORDINARIO	ASOCIADO	TC	HT	HP	TH	ING. RICHARD ARTEMIO FLORES CONDORI
	IAB42	TEORIA GENERAL DE SISTEMAS				3	0	3	
	IAA74	SIMULACION DE SISTEMAS				2	2	4	
	IAB31	TEORIA Y DISEÑO DE BASE DE DATOS				2	2	4	
	IAA82	INGENIERIA DE LA INFORMACION GRUPO A				3	2	5	
TOTAL HORAS LECTIVAS								16	
N°	CÓDIGO DE ASIGNATURA	CARGA LECTIVA	CONDICIÓN	CATEGORIA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO DIRECTIVO
						HT	HP	TH	
6	TRABAJO LECTIVO		ORDINARIO	ASOCIADO	TC	HT	HP	TH	ING. HERWIN ALYN HUILLCEN BACA
	IAB52	REDES I				3	2	5	
	IAB35	COMPUTADORES Y SISTEMAS OPERATIVOS I				2	2	4	
	IAAA6	PROGRAMACION AVANZADA GRUPO A				2	2	4	
	IAA93	SISTEMA DE COMUNICACIONES I				3	2	5	
TOTAL HORAS LECTIVAS								18	
N°	CÓDIGO DE ASIGNATURA	CARGA LECTIVA	CONDICIÓN	CATEGORIA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO DIRECTIVO
						HT	HP	TH	
7	TRABAJO LECTIVO		ORDINARIO	ASOCIADO	TC	HT	HP	TH	ING. ENRIQUE EDGARDO CONDOR TINOCO SECRETARIO GENERAL
	IAA62	ARQUITECTURA DE PROCESADORES II				3	2	5	
	IAA65	INVESTIGACION DE OPERACIONES I				2	2	4	
TOTAL HORAS LECTIVAS								9	

N°	CÓDIGO DE ASIGNATURA	CARGA LECTIVA	CONDICIÓN	CATEGORIA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO DIRECTIVO	
						HT	HP	TH		
8	TRABAJO LECTIVO						HT	HP	TH	ING. FLOR DE LUZ PALOMINO VALDIVIA
	IIAB23	METODOLOGÍA DE LA PROGRAMACIÓN	ORDINARIO	AUXILIAR	DE	2	4	6		
	IIAB13	INTRODUCCION A LA INFORMATICA Y COMPUTACIÓN				2	4	6		
	IIAB41	BASE DE DATOS I GRUPO A				2	4	6		
TOTAL HORAS LECTIVAS								18		
9	TRABAJO LECTIVO						HT	HP	TH	M.SC. EDWIN ROQUE TITO DIRECTOR DE REGISTROS ACADÉMICOS
	IIAB51	BASE DE DATOS II	ORDINARIO	AUXILIAR	DE	2	4	6		
IIAA72	PLANEAMIENTO INFORMATICO	3				2	5			
TOTAL HORAS LECTIVAS								11		
10	TRABAJO LECTIVO						HT	HP	TH	ING. EDWING ALCIDES MAQUERA FLORES
	IIAA83	SISTEMAS DE INFORMACION GERENCIAL	ORDINARIO	AUXILIAR	TC	3	2	5		
IIAA75	INVESTIGACION DE OPERACIONES II	2				2	4			
IIAB32	MATEMATICA DISCRETAS	2				2	4			
IIAA76	COMPUTACION GRAFICA	2				2	4			
TOTAL HORAS LECTIVAS								17		
11	TRABAJO LECTIVO						HT	HP	TH	ING. ROBERTO QUISPE QUISPE
	IIAB44	ELECTRONICA BASICA	ORDINARIO	AUXILIAR	TC	2	4	6		
IIAB54	SISTEMAS DIGITALES	2				2	4			
IIAA94	AUTOMATIZACION	3				2	5			
IIAB21	DIBUJO PARA INGENIERIA	2				2	4			
TOTAL HORAS								19		



ANEXO2: Distribución de carga del DAITI de los contratados cambios 2016 - II.

PLAZA 01							DOCENTE
CÓDIGO	CARGA LECTIVA	CONDICIÓN	MODALIDAD	HORAS LECTIVAS			Ing. Ruben Apaza Apaza
TRABAJO LECTIVO				HT	HP	TH	
IIAA81	INGENIERIA DE SOFYWARE II	CONTRATADO	TIEMPO COMPLETO	2	4	6	
IIAB53	PROGRAMACION WEB			2	4	6	
IIAA66	PROGRAMACIÓN III			3	4	7	
				TOTAL HORAS	19		
PLAZA 02							DOCENTE
CÓDIGO	CARGA LECTIVA	CONDICIÓN	MODALIDAD	HORAS LECTIVAS			Ing. Humberto Silveira Reynaga
TRABAJO LECTIVO				HT	HP	TH	
IIAA78	INGENIERIA EMPRESARIAL	CONTRATADO	TIEMPO COMPLETO	2	2	4	
IIAA66	PROGRAMACION AVANZADA GRUPO B			2	2	4	
IIAAA3	SISTEMAS DE COMUNICACIONES II			3	2	5	
IIAAA1	INTELIGENCIA ARTIFICIAL			3	2	5	
				TOTAL HORAS	18		
PLAZA 03							DOCENTE
CÓDIGO	CARGA LECTIVA	CONDICIÓN	MODALIDAD	HORAS LECTIVAS			Ing. Leonidas Asto Huaman
TRABAJO LECTIVO				HT	HP	TH	
IIAA87	APRENDIZAJE ELECTRONICO	CONTRATADO	TIEMPO COMPLETO	2	2	4	
IIAA63	REDES II			2	2	4	
IIAA93	ADMINISTRACION DE RECURSOS INFORMATICO			3	2	5	
IIAB32	MATEMATICA DISCRETA GRUPO B			2	2	4	
				TOTAL HORAS	17		
PLAZA 04							DOCENTE
CÓDIGO	CARGA LECTIVA	CONDICIÓN	MODALIDAD	HORAS LECTIVAS			M.Sc. Ivan Soria Solis
TRABAJO LECTIVO				HT	HP	TH	
IIAA61	TEORÍA Y DISEÑO DE BASE DE DATOS II	CONTRATADO	TIEMPO COMPLETO	3	2	5	
IIAAA4	CONTROL DE CALIDAD DE SOFTWARE			3	2	5	
IIAA82	INGENIERIA DE LA INFORMACION GRUPO B			3	2	5	
IIAAA2	SEGURIDAD Y AUDITORIA DE SISTEMAS			3	2	5	
				TOTAL HORAS	20		
PLAZA 05							DOCENTE
CÓDIGO	CARGA LECTIVA	CONDICIÓN	MODALIDAD	HORAS LECTIVAS			M.Sc. Carlos Yinmel Castrol Buleje
TRABAJO LECTIVO				HT	HP	TH	
IIAA93	SISTEMAS DE COMUNCACIONES I GRUPO B	CONTRATADO	TIEMPO COMPLETO	3	2	5	
IIAB41	BASE DE DATOS I GRUPO B			2	4	6	
IIAB45	COMPUTADORES Y SISTEMAS OPERATIVOS II			2	4	6	
				TOTAL HORAS	17		
PLAZA 06							DOCENTE
CÓDIGO	CARGA LECTIVA	CONDICIÓN	MODALIDAD	HORAS LECTIVAS			Ing. Richard Carrion Abollaneda
TRABAJO LECTIVO				HT	HP	TH	
IIAA83	SISTEMAS DE INFORMACION GERENCIAL GRUPO B	CONTRATADO	TIEMPO PARCIAL	3	2	5	
				TOTAL HORAS	5		
PLAZA 07							DOCENTE
CÓDIGO	CARGA LECTIVA	CONDICIÓN	MODALIDAD	HORAS LECTIVAS			Ing. Wilfredo Soto Palomino
TRABAJO LECTIVO				HT	HP	TH	
IIAB13	INTRODUCCION A LA INFORMATICA GRUPO B	CONTRATADO	TIEMPO COMPLETO	2	4	6	
IIAB33	DISEÑO WEB			2	2	4	
IIAB34	ALGORITMO Y ESTRUCTURA DE DATOS			2	4	6	
				TOTAL HORAS	16		
PLAZA 08							DOCENTE
CÓDIGO	CARGA LECTIVA	CONDICIÓN	MODALIDAD	HORAS LECTIVAS			Ing. Max Fredi Quispe Aguilar
TRABAJO LECTIVO				HT	HP	TH	
IIAB44	ELECTRONICA BASICA GRUPO B	CONTRATADO	TIEMPO COMPLETO	2	4	6	
IIAB54	SISTEMAS DIGITALES GRUPO B			2	2	4	
IIAA94	AUTOMATIZACION GRUPO B			3	2	5	
IIAB21	DIBUJO PARA INGENIERIA GRUPO B			2	2	4	
				TOTAL HORAS	19		

ANEXO 3: Distribución de carga del DAITI de los contratados cambios 2015.

 									
Universidad Nacional José María Arguedas <i>Identidad y Excelencia para el Trabajo Productivo y el Desarrollo</i>									
DEPARTAMENTO ACADÉMICO DE INGENIERÍA Y TECNOLOGÍA INFORMÁTICA DISTRIBUCION CARGA LECTIVA DOCENTES CONTRATADOS SEMESTRE ACADÉMICO 2015 - II									
N°	CÓDIGO ASIGN	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO
1	IIAB31	TEORÍA Y DISEÑO DE BASE DE DATOS (Grupo A)	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	2	2	4	Ing. Humberto Silvera Reynaga.
	IIAB33	DISEÑO WEB				2	2	4	
	IIAA42	ALGORITMOS Y ESTRUCTURA DE DATOS II				2	4	6	
	IIAA65	INVESTIGACION DE OPERACIONES I				2	2	4	
						TOTAL HORAS	18		
2	IIAA76	COMPUTACIÓN GRÁFICA	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	0	4	4	Ing. Ivan Soria Solis.
	IIAA91	INGENIERÍA DE SOFTWARE III				2	4	6	
	IIAA98	TÓPICOS ESPECIALES				2	2	4	
	IIAA72	PLANEAMIENTO INFORMÁTICO				3	2	5	
						TOTAL HORAS	19		
3	IIAB34	ALGORITMO Y ESTRUCTURA DE DATOS	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	2	4	6	Ing. Leonidas Asto Huaman.
	IIAA66	PROGRAMACIÓN III				3	4	7	
	IIAA2	SEGURIDAD Y AUDITORÍA DE SISTEMAS				3	2	5	
						TOTAL HORAS	18		
4	IIAA53	ARQUITECTURA DE PROCESADORES I	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	3	2	5	Ing. Carlos Yinmel Castro Buleje
	IIAA62	ARQUITECTURA DE PROCESADORES II				3	2	5	
	IIAA78	INGENIERIA EMPRESARIAL				2	2	4	
	IIAA94	AUTOMATIZACIÓN				3	2	5	
						TOTAL HORAS	19		
5	IIAA81	INGENIERÍA DE SOFTWARE II	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	2	4	6	Ing. Wilson Candia Quispe
	IIAA83	SISTEMAS DE INFORMACION GERENCIAL				3	2	5	
	IIAA1	INTELIGENCIA ARTIFICIAL				3	2	5	
	IIAA6	PROGRAMACIÓN AVANZADA				2	2	4	
						TOTAL HORAS	20		
6	IIAB23	METODOLOGIA DE LA PROGRAMACION (Grupo B)	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	2	4	6	Ing. Richard Carrion Abollaneda
	IIAB31	TEORÍA Y DISEÑO DE BASE DE DATOS (Grupo B)				2	2	4	
	IIAB32	MATEMATICA DISCRETA				2	2	4	
	IIAAA4	CONTROL DE CALIDAD DE SOFTWARE				3	2	5	
						TOTAL HORAS	19		
7	IIAB13	INTRODUCCIÓN A LA INFORMÁTICA Y COMPUTACIÓN (Grupo B)	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	2	4	6	Ing. Ruben Apaza Apaza
	IIAA42	ALGORITMOS Y ESTRUCTURA DE DATOS II (Grupo B)				2	4	6	
	IIAA66	PROGRAMACIÓN III (Grupo B)				3	4	7	
						TOTAL HORAS	19		



ANEXO 4: Distribución de carga del DAITI de los nombrados cambios 2015.

				Universidad Nacional José María Arguedas						
		Identidad y Excelencia para el Trabajo Productivo y el Desarrollo								
DEPARTAMENTO ACADÉMICO DE INGENIERÍA Y TECNOLOGÍA INFORMÁTICA										
DISTRIBUCION CARGA LECTIVA DOCENTES NOMBRADOS SEMESTRE ACADÉMICO 2015 - II										
N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
			ORDINARIA	PRINCIPAL	TIEMPO COMPLETO				Cecilia edith Garcia Rivas Plata	
N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
		TRABAJO LECTIVO				HT	HP	TH		
1	I1AA71	INGENIERÍA DE SOFTWARE I	ORDINARIO	ASOCIADO	TIEMPO COMPLETO	2	4	6	Dr. ING. ANGEL FERNANDO NAVARRO RAYMUNDO Director de Escuela	
	I1AA5	TESIS II				2	2	4		
						TOTAL HORAS	10			
N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
		TRABAJO LECTIVO				HT	HP	TH		
2	I1AA45	PROGRAMACIÓN I	ORDINARIO	ASOCIADO	DEDICACIÓN EXCLUSIVA	2	4	6	ING. JUAN JOSE ORE CERRON	
	I1AA55	INGENIERÍA ADMINISTRATIVA				2	2	4		
	I1AA95	TESIS I				3	0	3		
			TOTAL HORAS	13						
N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
		TRABAJO LECTIVO				HT	HP	TH		
3	I1AA51	TEORÍA Y DISEÑO DE BASE DE DATOS I	ORDINARIO	ASOCIADO	TIEMPO COMPLETO	3	4	7	ING. RICHARD ARTEMIO FLORES CONDORI	
	I1AA74	SIMULACIÓN DE SISTEMAS				2	2	4		
	I1AA82	INGENIERÍA DE INFORMACIÓN				3	2	5		
			TOTAL HORAS	16						
N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
		TRABAJO LECTIVO				HT	HP	TH		
4	I1AA43	SISTEMAS OPERATIVOS II	ORDINARIO	ASOCIADO	TIEMPO COMPLETO	3	2	5	ING. HERWIN ALAYN HUILLCEN BACA	
	I1AA93	SISTEMAS DE COMUNICACIONES I				3	2	5		
	I1AA43	SISTEMAS DE COMUNICACIONES II				3	2	5		
			TOTAL HORAS	15						
N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
		TRABAJO LECTIVO				HT	HP	TH		
5	I1AA52	PROGRAMACIÓN II	ORDINARIO	ASOCIADO	TIEMPO COMPLETO	2	4	6	ING. ENRIQUE EDGARDO CONDOR TINOCO Secretario General	
	I1AA92	ADMINISTRACIÓN DE RECURSOS INFORMÁTICOS				3	2	5		
			TOTAL HORAS	11						
N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
		TRABAJO LECTIVO				HT	HP	TH		
6	I1AB21	DIBUJO PARA INGENIERÍA (Nuevo Plan Curricular)	ORDINARIO	AUXILIAR	DEDICACIÓN EXCLUSIVA	2	2	4	ING. FLOR DE LUZ PALOMINO VALDIVIA	
	I1AB35	COMPUTADORES Y SISTEMAS OPERATIVOS I (Nuevo Plan Curricular)				2	2	4		
	I1AA56	REDES I				2	2	4		
	I1AA63	REDES II				2	2	4		
			TOTAL HORAS	16						
N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
		TRABAJO LECTIVO				HT	HP	TH		
7	I1AA44	SISTEMAS DIGITALES	ORDINARIO	AUXILIAR	DEDICACIÓN EXCLUSIVA	2	4	6	M. Sc. ING. EDWIN ROQUE TITO Director de Departamento Académico de Ingeniería y Tecnología Informática	
	I1AA61	TEORÍA Y DISEÑO DE BASE DE DATOS II				3	2	5		
			TOTAL HORAS	11						
N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
		TRABAJO LECTIVO				HT	HP	TH		
8	I1AA54	TEORÍA GENERAL DE SISTEMAS	ORDINARIO	AUXILIAR	TIEMPO COMPLETO	3	0	3	ING. EDWING ALCIDES MAQUERA FLORES Director de la Oficina de Admisión Permanente	
	I1AA68	MULTIMEDIA				0	2	2		
	I1AA75	INVESTIGACIÓN DE OPERACIONES II				2	2	4		
	I1AA84	FORMULACIÓN Y EVALUACIÓN DE PROYECTOS INFORMÁTICOS				3	2	5		
			TOTAL HORAS	14						
N°	CÓDIGO ASIGN	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
		TRABAJO LECTIVO				HT	HP	TH		
9	I1AB13	INTRODUCCIÓN A LA INFORMÁTICA Y COMPUTACIÓN (Grupo A)	ORDINARIO	AUXILIAR	DEDICACIÓN EXCLUSIVA	2	4	6	ING. JULIO CESAR HUANCA MARIN	
	I1AB23	METODOLOGÍA DE LA PROGRAMACION (Grupo A)				2	4	6		
	I1AA88	COMERCIO ELECTRÓNICO				2	2	4		
			TOTAL HORAS	16						
			ordinario	auxiliar	tc				roberto quis quispe	

**ANEXO 5: Distribución de carga electiva de docentes contratados académicos
2015 –II -Contratados 2015 – II**

 Universidad Nacional José María Arguedas <i>Identidad y Excelencia para el Trabajo Productivo y el Desarrollo</i> 									
DEPARTAMENTO ACADÉMICO DE INGENIERÍA Y TECNOLOGÍA INFORMÁTICA									
DISTRIBUCION CARGA LECTIVA DOCENTES CONTRATADOS SEMESTRE ACADÉMICO 2015 - II									
N°	CÓDIGO ASIGN	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO
1	IIAB31	TEORÍA Y DISEÑO DE BASE DE DATOS (Grupo A)	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	2	2	4	Ing. Humberto Silvera Reynaga.
	IIAB33	DISEÑO WEB				2	2	4	
	IIAA42	ALGORITMOS Y ESTRUCTURA DE DATOS II				2	4	6	
	IIAA65	INVESTIGACION DE OPERACIONES I				2	2	4	
						TOTAL HORAS	18		
2	IIAA76	COMPUTACIÓN GRÁFICA	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	0	4	4	Ing. Ivan Soria Solis.
	IIAA91	INGENIERÍA DE SOFTWARE III				2	4	6	
	IIAA98	TÓPICOS ESPECIALES				2	2	4	
	IIAA72	PLANEAMIENTO INFORMÁTICO				3	2	5	
						TOTAL HORAS	19		
3	IIAB34	ALGORITMO Y ESTRUCTURA DE DATOS	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	2	4	6	Ing. Leonidas Asto Huaman.
	IIAA66	PROGRAMACIÓN III				3	4	7	
	IIAA42	SEGURIDAD Y AUDITORÍA DE SISTEMAS				3	2	5	
						TOTAL HORAS	18		
4	IIAA53	ARQUITECTURA DE PROCESADORES I	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	3	2	5	Ing. Carlos Yinmel Castro Buleje
	IIAA62	ARQUITECTURA DE PROCESADORES II				3	2	5	
	IIAA78	INGENIERIA EMPRESARIAL				2	2	4	
	IIAA94	AUTOMATIZACIÓN				3	2	5	
						TOTAL HORAS	19		
5	IIAA55	INGENIERÍA ADMINISTRATIVA	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	2	2	4	Ing. Wilson Candia Quispe
	IIAA81	INGENIERÍA DE SOFTWARE II				2	4	6	
	IIAAA1	INTELIGENCIA ARTIFICIAL				3	2	5	
	IIAA66	PROGRAMACIÓN AVANZADA				2	2	4	
						TOTAL HORAS	19		
6	IIAB23	METODOLOGIA DE LA PROGRAMACION (Grupo B)	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	2	4	6	Ing. Richard Carrion Abollaneda
	IIAB31	TEORÍA Y DISEÑO DE BASE DE DATOS (Grupo B)				2	2	4	
	IIAB32	MATEMÁTICA DISCRETA				2	2	4	
	IIAAA4	CONTROL DE CALIDAD DE SOFTWARE				3	2	5	
						TOTAL HORAS	19		
7	IIAB13	INTRODUCCIÓN A LA INFORMÁTICA Y COMPUTACIÓN (Grupo B)	CONTRATADO	Equivalente a Auxiliar	TIEMPO COMPLETO	2	4	6	Ing. Ruben Apaza Apaza
	IIAA42	ALGORITMOS Y ESTRUCTURA DE DATOS II (Grupo B)				2	4	6	
	IIAA66	PROGRAMACIÓN III (Grupo B)				3	4	7	
						TOTAL HORAS	19		

**ANEXO 6: Distribución de carga electiva a docentes nombrados académicos
2015 –II -nombrados 2015 – II**

 Universidad Nacional José María Arguedas Identidad y Excelencia para el Trabajo Productivo y el Desarrollo 									
DEPARTAMENTO ACADÉMICO DE INGENIERÍA Y TECNOLOGÍA INFORMÁTICA									
DISTRIBUCION CARGA LECTIVA DOCENTES NOMBRADOS SEMESTRE ACADÉMICO 2015 - II									
N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO
1		TRABAJO LECTIVO	ORDINARIO	ASOCIADO	TIEMPO COMPLETO	HT	HP	TH	Dr. ING. ANGEL FERNANDO NAVARRO RAYMUNDO Director de Escuela
	I1AA71	INGENIERÍA DE SOFTWARE I				2	4	6	
	I1AA45	TESIS II				2	2	4	
	TOTAL HORAS					10			
2		TRABAJO LECTIVO	ORDINARIO	ASOCIADO	DEDICACIÓN EXCLUSIVA	HT	HP	TH	ING. JUAN JOSE ORE CERRON Secretario General
	I1AA45	PROGRAMACIÓN I				2	4	6	
	I1AA95	TESIS I				3	0	3	
	TOTAL HORAS					9			
3		TRABAJO LECTIVO	ORDINARIO	ASOCIADO	TIEMPO COMPLETO	HT	HP	TH	ING. RICHARD ARTEMIO FLORES CONDORI
	I1AA51	TEORÍA Y DISEÑO DE BASE DE DATOS I				3	4	7	
	I1AA74	SIMULACIÓN DE SISTEMAS				2	2	4	
	I1AA82	INGENIERÍA DE INFORMACIÓN				3	2	5	
TOTAL HORAS			16						
4		TRABAJO LECTIVO	ORDINARIO	ASOCIADO	TIEMPO COMPLETO	HT	HP	TH	ING. HERWIN ALAYN HUILLCEN BACA
	I1AA43	SISTEMAS OPERATIVOS II				3	2	5	
	I1AA93	SISTEMAS DE COMUNICACIONES I				3	2	5	
	I1AA43	SISTEMAS DE COMUNICACIONES II				3	2	5	
TOTAL HORAS			15						
5		TRABAJO LECTIVO	ORDINARIO	ASOCIADO	TIEMPO COMPLETO	HT	HP	TH	ING. ENRIQUE EDGARDO CONDOR TINOCO
	I1AA52	PROGRAMACIÓN II				2	4	6	
	I1AA83	SISTEMAS DE INFORMACION GERENCIAL				3	2	5	
	I1AA92	ADMINISTRACIÓN DE RECURSOS INFORMÁTICOS				3	2	5	
TOTAL HORAS			16						
6		TRABAJO LECTIVO	ORDINARIO	AUXILIAR	DEDICACIÓN EXCLUSIVA	HT	HP	TH	ING. FLOR DE LUZ PALOMINO VALDIVIA
	I1AB21	DIBUJO PARA INGENIERIA (Nuevo Plan Curricular)				2	2	4	
	I1AB35	COMPUTADORES Y SISTEMAS OPERATIVOS I (Nuevo Plan Curricular)				2	2	4	
	I1AA56	REDES I				2	2	4	
I1AA63	REDES II	2	2	4					
TOTAL HORAS			16						
7		TRABAJO LECTIVO	ORDINARIO	AUXILIAR	DEDICACIÓN EXCLUSIVA	HT	HP	TH	M. Sc. ING. EDWIN ROQUE TITO Director de Departamento Academico de Ingenieria y Tecnologia Informatica
	I1AA44	SISTEMAS DIGITALES				2	4	6	
	I1AA61	TEORÍA Y DISEÑO DE BASE DE DATOS II				3	2	5	
TOTAL HORAS			11						
8		TRABAJO LECTIVO	ORDINARIO	AUXILIAR	TIEMPO COMPLETO	HT	HP	TH	ING. EDWIN ALCIDES MAQUERA FLORES Director de la Oficina de Admision Permanente
	I1AA54	TEORÍA GENERAL DE SISTEMAS				3	0	3	
	I1AA68	MULTIMEDIA				0	2	2	
	I1AA75	INVESTIGACIÓN DE OPERACIONES II				2	2	4	
I1AA84	FORMULACIÓN Y EVALUACIÓN DE PROYECTOS INFORMÁTICOS	3	2	5					
TOTAL HORAS			14						
9		TRABAJO LECTIVO	ORDINARIO	AUXILIAR	DEDICACIÓN EXCLUSIVA	HT	HP	TH	ING. JULIO CESAR HUANCA MARIN
	I1AB13	INTRODUCCIÓN A LA INFORMÁTICA Y COMPUTACIÓN (Grupo A)				2	4	6	
	I1AB23	METODOLOGÍA DE LA PROGRAMACIÓN (Grupo A)				2	4	6	
	I1AA88	COMERCIO ELECTRÓNICO				2	2	4	
TOTAL HORAS			16						

ANEXO 7: Plan curricular de la Escuela Profesional de Ingeniería de Sistemas.

		Universidad Nacional José María Arguedas						
<i>Identidad y Excelencia para el Trabajo Productivo y el Desarrollo</i>								
PLAN CURRICULAR DE LA ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS								
PRIMER CICLO								
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	
1	IIAB13	EEE	Introducción a la Informática y Computación	2	4	6	4	
TOTAL HORAS				2	4	6	4	
SEGUNDO CICLO								
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	
2	IIAB21	EEE	Dibujo para Ingeniería	2	2	4	3	
3	IIAB23	EEE	Metodología de la Programación	2	4	6	4	
TOTAL HORAS				4	6	10	7	
TERCER CICLO								
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	
4	IIAB31	EEE	Teoría y Diseño de Base de Datos	2	2	4	3	
5	IIAB32	EEE	Matemática Discreta	2	2	4	3	
6	IIAB33	EEE	Diseño Web	2	2	4	3	
7	IIAB34	EEE	Algoritmo y Estructura de Datos	2	4	6	4	
8	IIAB35	EEE	Computadores y Sistemas Operativos I	2	2	4	3	
TOTAL HORAS				10	12	22	16	
CUARTO CICLO								
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	
9	IIAB41	EEE	Base de Datos I	2	4	6	4	
10	IIAB42	EEE	Teoría General de Sistemas	3	0	3	3	
11	IIAB43	EEE	Programación	2	4	6	4	
12	IIAB44	EEE	Electrónica Básica	2	4	6	4	
13	IIAB45	EEE	Computadores y Sistemas Operativos II	3	2	5	4	
TOTAL HORAS				12	14	26	19	
QUINTO CICLO								
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	
14	IIAB51	EEE	Base de Datos II	2	4	6	4	
15	IIAB52	EEE	Redes I	3	2	5	4	
16	IIAB53	EEE	Programación Web	2	4	6	4	
17	IIAB54	EEE	Sistemas Digitales	2	2	4	3	
18	IIAB56	EEE	Investigación Tecnológica e Innovación	2	2	4	3	
TOTAL HORAS				11	14	25	18	

SEXTO CICLO

N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS
19	IIAB61	EEE	Análisis y Diseño de Sistemas	3	2	5	4
20	IIAB62	EEE	Redes II	2	4	6	4
21	IIAB64	EEE	Gestión de Empresas Tecnológicas	2	2	4	3
22	IIAB65	EEE	Inteligencia de Negocios	2	4	6	4
23	IIAB66	EEE	Investigación de Operaciones	3	2	5	4
			TOTAL HORAS	12	14	26	19

SÉPTIMO CICLO

N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS
24	IIAB71	EEE	Ingeniería de Software I	3	2	5	4
25	IIAB72	EEE	Soluciones Móviles y Cloud Computing	2	4	6	4
26	IIAB73	EEE	Sistemas Integrados de Gestión ERP	2	4	6	4
27	IIAB74	EEE	Formulación y Evaluación de Proyectos Informáticos	3	2	5	4
28	IIAB75	EEE	Simulación de Sistemas	2	2	4	3
			TOTAL HORAS	12	14	26	19

OCTAVO CICLO

N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS
29	IIAB81	EEE	Ingeniería de Software II	3	2	5	4
30	IIAB82	EG	Tecnologías Ecoeficientes	2	2	4	3
31	IIAB83	EEE	Seguridad Informática	2	4	6	4
32	IIAB84	EEE	Comunicaciones I	2	4	6	4
33	IIAB85	EEE	Gerencia de TI	3	2	5	4
			TOTAL HORAS	12	14	26	19

NOVENO CICLO

N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS
34	IIAB91	EEE	Taller de Investigación Aplicada I	4	0	4	4
35	IIAB92	EEE	Gestión de Procesos de Negocio	2	2	4	3
36	IIAB93	EEE	Comunicaciones II	2	4	6	4
37	IIAB94	EEE	Sistemas Distribuidos	2	4	6	4
38	IIAB95	EEE	Inteligencia Artificial	2	4	6	4
			TOTAL HORAS	12	14	26	19

DÉCIMO CICLO

N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS
30	IIABA1	EEE	Taller de Investigación Aplicada II	7	0	7	7
40	IIABA2	EEE	Auditoría de TI	2	2	4	3
41	IIABA3	EEE	Legislación y Delitos Informáticos	3	0	3	3
42	IIABA4	EG	Deontología Profesional	2	0	2	2
43	IIABA5	EEE	Planeamiento Estratégico de Tecnologías de Información	3	2	5	4
			TOTAL HORAS	17	4	21	19

CURSOS ELECTIVOS							
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS
1	IIAB76	EEE	Computación Gráfica	2	2	4	3
2	IIAB77	EEE	Base de Datos Orientado a Objetos	2	2	4	3
3	IIAB86	EEE	Programación de Videojuegos	2	2	4	3
4	IIAB87	EEE	Interfaces Humano Computador	2	2	4	3
5	IIAB88	EEE	Arquitectura de Computadores	2	2	4	3
6	IIAB96	EEE	Reconocimiento de Patrones	2	2	4	3
7	IIAB97	EEE	Arquitectura de Data Centers	2	2	4	3
8	IIAB98	EEE	Autómatas Programables	2	2	4	3
9	IIABA6	EEE	Robótica	2	2	4	3
10	IIABA7	EEE	Tópicos Especiales	2	2	4	3
11	IIABA8	EEE	Ingeniería de Cableado Estructurado	2	2	4	3
12	IIABA9	EEE	Programación avanzada	2	2	4	3
EST. = ESTUDIOS							
EG = ESTUDIOS GENERALES							
EEE = ESTUDIOS ESPECÍFICOS Y DE ESPECIALIDAD							
RESUMEN DE CRÉDITOS Y COMPONENTES							
N°	CONDICIÓN	N° ASIGNATURAS	HT	HP	TH	CRÉDITOS	
1	OBLIGATORIOS	43	104	110	214	159	
2	ELECTIVOS	12	24	24	48	36	
	TOTAL	55	128	134	262	195	

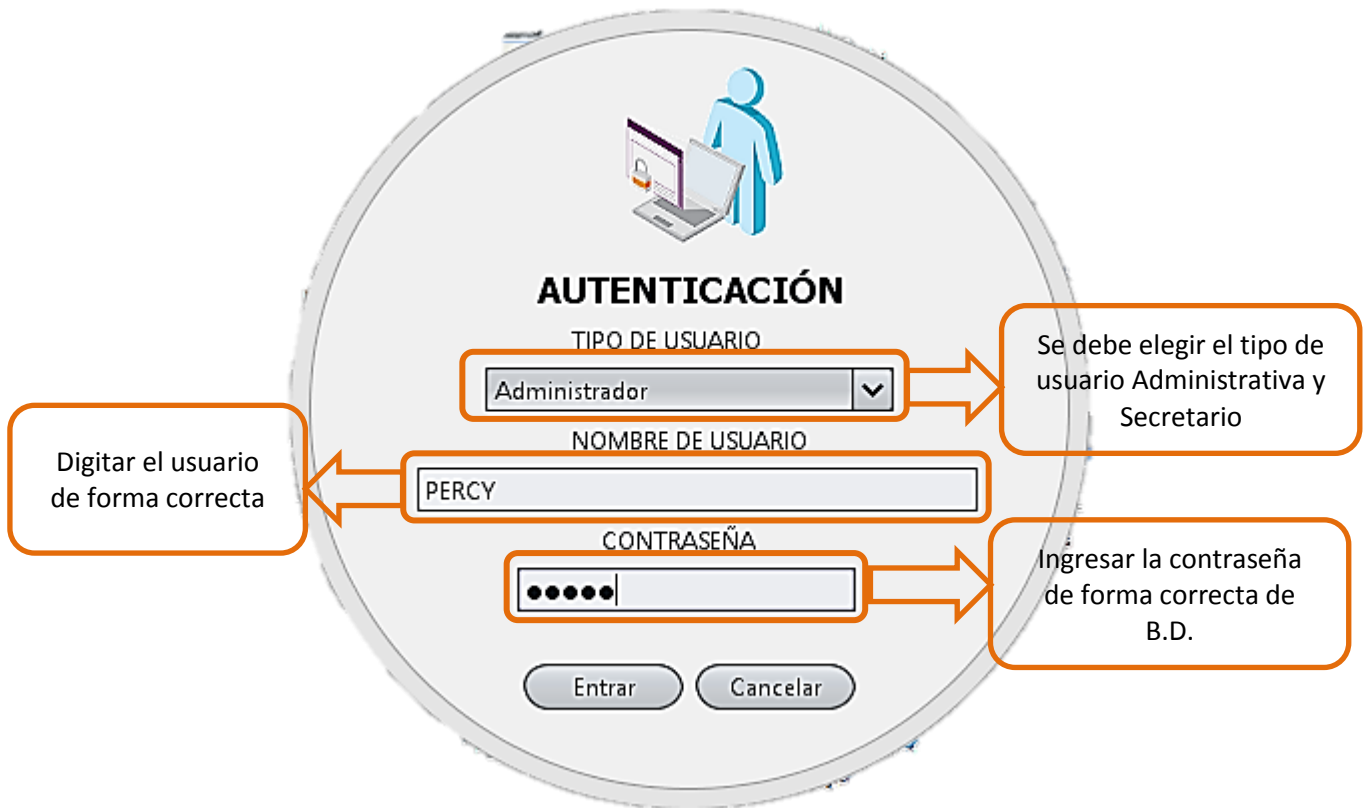
ANEXO 8: Cuadro de equivalencias de las asignaturas del currículo 2007 y 2014-II

 Universidad Nacional José María Arguedas <i>Identidad y Excelencia para el Trabajo Productivo y el Desarrollo</i> 														
CUADRO DE EQUIVALENCIAS DE LAS ASIGNATURAS DEL CURRÍCULO 2007 Y 2014-II														
CURRÍCULO 2014 - II								CURRÍCULO 2007						
PRIMER CICLO														
Nº	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H. T.	H. P.	T. H.	CREDITOS
1	IAB11	EG	Comunicación Oral y Escrita	2	2	4	3	IAA15	1	LENGUAJE	3	0	3	3
2	IAB12	EG	Matemática I	2	4	6	4	IAA11	1	MATEMÁTICA BÁSICA I	3	2	5	4
3	IAB13	EEE	Introducción a la Informática y Computación	2	4	6	4	IAA16	1	INFORMÁTICA I	2	4	6	4
4	IAB14	EG	Técnicas de Aprendizaje Universitario	1	2	3	2	IAA14	1	UNIVERSITARIO	2	0	2	2
6	IAB16	EG	Filosofía y Ética	3	0	3	3	IAA24	2	FILOSOFÍA	2	0	2	2
SEGUNDO CICLO														
Nº	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H. T.	H. P.	T. H.	CREDITOS
1	IAB21	EEE	Dibujo para Ingeniería	2	2	4	3	IAA26	2	DISEÑO ASISTIDO POR COMPUTADORA	2	2	4	3
2	IAB22	EG	Matemática II	1	4	5	3	IAA12	1	MATEMÁTICA BÁSICA II	3	2	5	4
3	IAB23	EEE	Metodología de la Programación	2	4	6	4	IAA25	2	INFORMÁTICA II	2	4	6	4
5	IAB26	EG	Económica	2	0	2	2	IAA37	3	ECONOMÍA	2	0	2	2
7	IAB27	EG	Cálculo I	2	4	6	4	IAA21	2	CÁLCULO I	2	4	6	4
TERCER CICLO														
Nº	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H. T.	H. P.	T. H.	CREDITOS
1	IAB31	EEE	Teoría y Diseño de Base de Datos	2	2	4	3	IAA51	5	TEORÍA Y DISEÑO DE BASE DE DATOS I	3	4	7	5
4	IAB34	EEE	Algoritmo y Estructura de Datos	2	4	6	4	IAA34	3	ALGORITMO Y ESTRUCTURA DE DATOS I	2	4	6	4
5	IAB35	EEE	Computadores y Sistemas Operativos I	2	2	4	3	IAA35	3	SISTEMAS OPERATIVOS I	2	2	4	3
6	IAB36	EG	Cálculo II	2	2	4	3	IAA31	3	CÁLCULO II	2	4	6	4
7	IAB37	EG	Física	2	4	6	4	IAA22	2	FÍSICA I	2	4	6	4
CUARTO CICLO														
Nº	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H. T.	H. P.	T. H.	CREDITOS
1	IAB41	EEE	Base de Datos I	2	4	6	4	IAA61	6	TEORÍA Y DISEÑO DE BASE DE DATOS II	3	2	5	4
2	IAB42	EEE	Teoría General de Sistemas	3	0	3	3	IAA54	5	TEORÍA GENERAL DE SISTEMAS	3	0		3
3	IAB43	EEE	Programación	2	4	6	4	IAA45	4	PROGRAMACIÓN I	2	4	6	4
5	IAB45	EEE	Computadores y Sistemas Operativos II	3	2	5	4	IAA43	4	SISTEMAS OPERATIVOS II	3	2	5	4
6	IAB46	EG	Estadística Descriptiva	2	2	4	3	IAA33	3	ESTADÍSTICA I	2	2	4	3

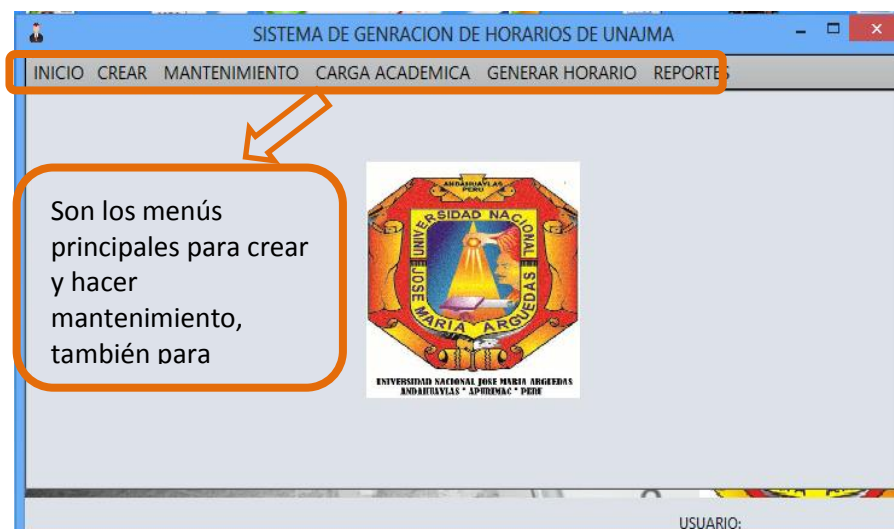
QUINTO CICLO																	
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H.	T.	H.	P.	T.	H.	CREDITOS
2	IIAB52	EEE	Redes I	3	2	5	4	IIAA56	5	REDES I	2	2	4	4	3		
3	IIAB53	EEE	Programación Web	2	4	6	4	IIAA52	5	PROGRAMACIÓN II	2	4	6	4	4		
4	IIAB54	EEE	Sistemas Digitales	2	2	4	3	IIAA44	4	SISTEMAS DIGITALES	2	4	6	4	4		
5	IIAB55	EG	Estadística Inferencial	2	2	4	3	IIAA46	4	ESTADÍSTICA II	2	2	4	4	3		
SEXTO CICLO																	
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H.	T.	H.	P.	T.	H.	CREDITOS
1	IIAB61	EEE	Análisis y Diseño de Sistemas	3	2	5	4	IIAA71	7	INGENIERÍA DE SOFTWARE I	2	4	6	4	4		
2	IIAB62	EEE	Redes II	2	4	6	4	IIAA63	6	REDES II	2	2	4	4	3		
3	IIAB63	EG	Costos y Presupuestos	3	0	3	3	IIAA73	7	INGENIERÍA DE COSTOS	2	2	4	4	3		
5	IIAB65	EEE	Inteligencia de Negocios	2	4	6	4	IIAA82	8	INGENIERÍA DE INFORMACIÓN	3	2	5	4	4		
6	IIAB66	EEE	Investigación de Operaciones	3	2	5	4	IIAA65	6	INVESTIGACIÓN DE OPERACIONES I	2	2	4	4	3		
SÉPTIMO CICLO																	
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H.	T.	H.	P.	T.	H.	CREDITOS
1	IIAB71	EEE	Ingeniería de Software I	3	2	5	4	IIAA81	8	INGENIERÍA DE SOFTWARE II	2	4	6	4	4		
4	IIAB74	EEE	Formulación y Evaluación de Proyectos Informáticos	3	2	5	4	IIAA84	8	FORMULACIÓN Y EVALUACIÓN DE PROYECTOS INFORMÁTICOS	3	2	5	4	4		
5	IIAB75	EEE	Simulación de Sistemas	2	2	4	3	IIAA74	7	SIMULACIÓN DE SISTEMAS	2	2	4	4	3		
OCTAVO CICLO																	
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H.	T.	H.	P.	T.	H.	CREDITOS
1	IIAB81	EEE	Ingeniería de Software II	3	2	5	4	IIAA91	9	INGENIERÍA DE SOFTWARE III	2	4	6	4	4		
3	IIAB84	EEE	Comunicaciones I	2	4	6	4	IIAA93	9	SISTEMAS DE COMUNICACIONES I	3	2	5	4	4		
5	IIAB85	EEE	Gerencia de TI	3	2	5	4	IIAA92	9	ADMINISTRACIÓN DE RECURSOS INFORMÁTICOS	3	2	5	4	4		
NOVENO CICLO																	
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H.	T.	H.	P.	T.	H.	CREDITOS
1	IIAB91	EEE	Taller de Investigación Aplicada I	4	0	4	4	IIAA95	9	TESIS I	3	0	3	3	3		
2	IIAB93	EEE	Comunicaciones II	2	4	6	4	IIAA3	10	SISTEMAS DE COMUNICACIONES II	3	2	5	4	4		
5	IIAB95	EEE	Inteligencia Artificial	2	4	6	4	IIAA1	10	INTELIGENCIA ARTIFICIAL	3	2	5	4	4		
DÉCIMO CICLO																	
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H.	T.	H.	P.	T.	H.	CREDITOS
1	IIABA1	EEE	Taller de Investigación Aplicada II	7	0	7	7	IIAA5	10	TESIS II	2	2	4	3	3		
4	IIABA5	EEE	Planeamiento Estratégico de Tecnologías de Información	3	2	5	4	IIAA72	7	PLANEAMIENTO INFORMÁTICO	3	2	5	4	4		
CURSOS ELECTIVOS																	
N°	CÓDIGO	EST.	ASIGNATURA	HT	HP	TH	CRÉDITOS	CODIGO	CICLO	ASIGNATURAS	H.	T.	H.	P.	T.	H.	CREDITOS
1	IIAB76	EEE	Computación Gráfica	2	2	4	3	IIAA76	7	COMPUTACIÓN GRÁFICA	0	4	4	4	2		
4	IIAB79	EG	Realidad Peruana	2	2	4	3	IIAA23	2	HISTORIA ECONÓMICA Y SOCIAL DEL PERÚ	3	0	3	3	3		
7	IIAB88	EEE	Arquitectura de Computadores	2	2	4	3	IIAA53	5	ARQUITECTURA DE PROCESADORES I	3	2	5	4	4		
8	IIAB89	EG	Contabilidad General	2	2	4	3	IIAA64	6	CONTABILIDAD GENERAL	3	0	3	3	3		
14	IIAB7	EEE	Tópicos Especiales	2	2	4	3	IIAA98	9	TÓPICOS ESPECIALES	2	2	4	4	3		
16	IIABA9	EEE	Programación avanzada	2	2	4	3	IIAA6	10	PROGRAMACIÓN AVANZADA	2	2	4	4	3		
EST. = ESTUDIOS																	
EG = ESTUDIOS GENERALES																	
EEE = ESTUDIOS ESPECÍFICOS Y DE ESPECIALIDAD																	

ANEXO 9: Manual de uso del aplicativo de generación de horarios

- Para hacer uso del software ejecutar el aplicativo (.exe) y luego muestra acceso de autenticación ingresar los datos correctos que está en la base de datos



- Después de ingresar los datos de manera correcta al acceso de autenticación, muestra esta ventana principal del sistema de generación de horario con algoritmo genético.



- Luego en la ventana principal podemos iniciar a crear nuevos cursos, profesores, salones, usuarios, ciclos, departamento.



- Luego de hacer clic en crear y seleccionar nuevo docente, nos muestra la ventana para crear un nuevo docente y podemos ingresar los datos correspondientes.

NUEVO DOCENTE

DATOS PERSONALES

CODIGO: PR0010 DNI:

GRADO: Ing. TELEFONO:

NOMBRE: EMAIL:

AP. PATERNO: SEXO: F

AP. MATERNO:

DATOS LABORALES

UNIDAD ACADEMICA: SELECCIONE UNO DISP. DE HORAS: CONDICION

MODALIDAD: SELECCIONE UNO CATEGORIA: PRINCIPAL ORDINARIO

CLASES

PREFERENCIA DE HORAS:

PREFERENCIA DE CURSO:

Buttons:

- Luego de hacer clic en crear y seleccionar nuevo curso, nos muestra la ventana para crear un nuevo curso y podemos ingresar los datos correspondientes.

NUEVO CURSO

Datos del Curso

COD CURSO: SEMESTRE: SELECCIONES UNO ▼

NOMBRE: CICLO: SELECCIONES UNO ▼

CREDITO: 2 ▼ ESCUELA: SELECCIONES UNO ▼

PROYECTO: Si ▼

COMPOSICION:

	N° de horas semanal	N° de horas max	N° horario
<input type="checkbox"/> Horas Teoricas	<input type="text"/>	<input type="text"/>	Total de horas <input type="text"/>
<input type="checkbox"/> Horas de Laboratorio	<input type="text"/>	<input type="text"/>	Cantidad de Matriculados: <input type="text"/>
<input type="checkbox"/> Horas Practica	<input type="text"/>	<input type="text"/>	

+

⊘

↓

×

- Luego de hacer clic en crear y seleccionar nuevo salón, nos muestra la ventana para crear un nuevo salón y podemos ingresar los datos correspondientes.

NUEVO SALON

DATOS DEL SALON BUSCAR POR NUMERO:

NUMERO	<input type="text"/>	TIPO DE AULA:	SELECCIONES UNO: <input type="button" value="v"/>
DESCRIPCION:	<input type="text"/>	PABELLON:	SELECCIONES UNO: <input type="button" value="v"/>
CAPACIDAD DE CLASE:	<input type="text"/>	CODIGO	SA0005
DISPONIBILIDAD:	<input type="text"/>	ESTADO	<input type="text"/>

INFORMACION

CODIGO	NUMERO	CAP_AULA	DESCRIPCI...	DISPONIBIL...	ESTADO	PABELLON	TIPO SALON
SA0001	S001	30	BUNEO		BUENO	1	2001
SA0002	S002	30	BUENO		BUENO	1	2001
SA0003	S003	25	BUNEO		BUENO	1	2001
SA0004	S004	30	BUENO		BUENO	1	2001

- Luego de hacer clic en crear y seleccionar nuevos usuarios, nos muestra la ventana para crear un nuevo usuario y podemos ingresar los datos correspondientes.

SISTEMA DE CREACION DE USUARIOS



CODIGO

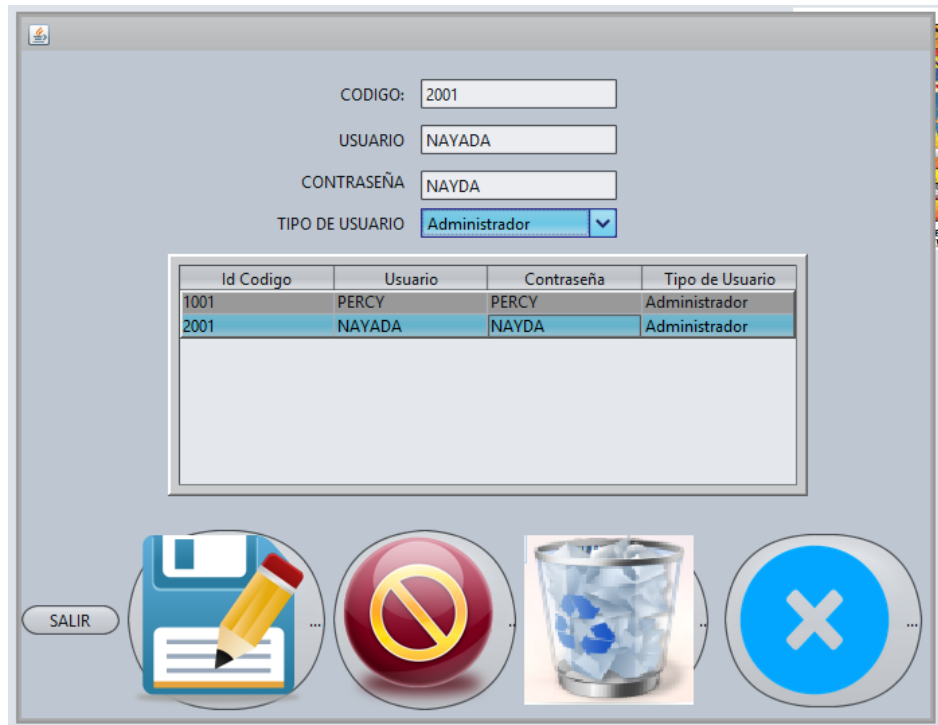
USUARIO

CONTRASEÑA

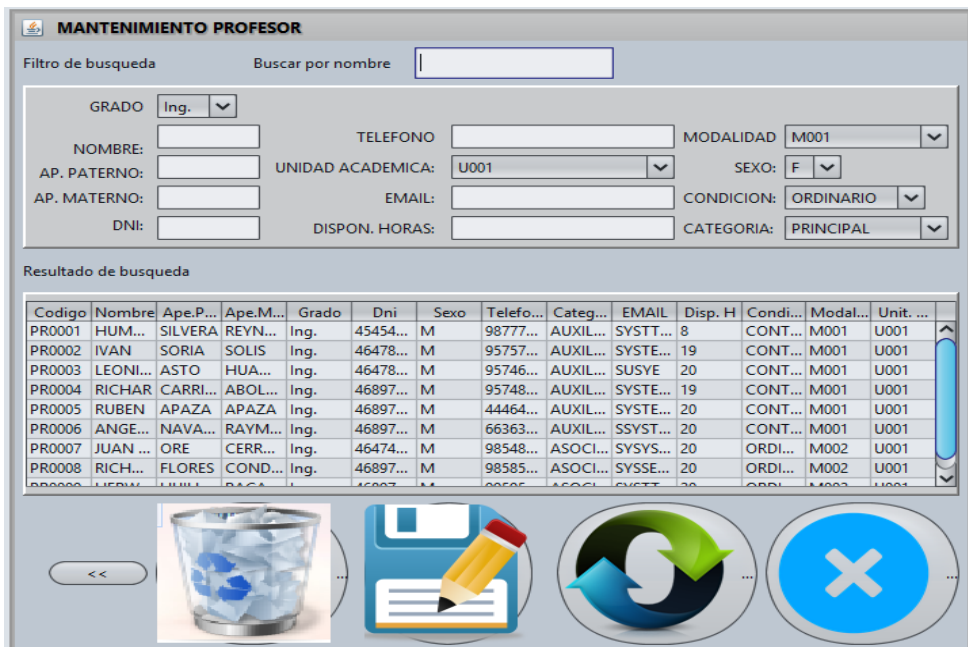
TIPO DE USUARIO Administrador Secretaria



- Para realizar Mantenimiento hacer clic en “Mantenimiento” y luego seleccionar Mantenimiento del usuario, como datos funcionales tenemos Actualizar, Modificar, Cancelar, Salir.



- Para realizar Mantenimiento hacer clic en “Mantenimiento” y luego seleccionar Mantenimiento del profesor, como datos funcionales tenemos Actualizar, Modificar, Cancelar, Salir y una tabla Jtable1 para ver los datos de mysql.



- Para realizar Mantenimiento hacer clic en “Mantenimiento” y luego seleccionar Mantenimiento del profesor, como datos funcionales tenemos Actualizar, Modificar, Cancelar, Salir y una tabla Jtable1 para ver los datos de mysql.

Mantenimiento Curo

FILTRO DE BUSQUEDA

ID CURSO: ESCUELA:

NOMBRE DEL CURSO: SEMESTRE:

HORA TEORICA: CICLO:

HORA PRACTICA: CREDITOS:

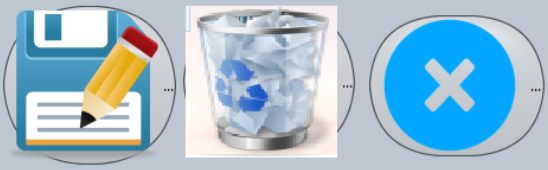
HORA DE LABORATORIO: PROYECTOR:

TOTAL MATRICULADO: HORAS SEMALES:

RESULTADO DE LA BUSQUEDA

Codigo	Nombre	H.P.	H.T.	H.L.	Credito	Hora Se...	Proyector	Total Mat	Ciclo	Escuela
IIAA42	ALGORIT...	0	0	0	4	6	Si	30	004	E001
IIAB11	COMUNI...	0	0	0	3	4	Si	30	001	E001
IIAB12	MATEM...	4	0	0	4	6	No	30	001	E001
IIAB13	INTROD...	4	0	0	4	6	Si	30	001	E001
IIAB14	TECNICA...	2	1	0	2	3	No	30	001	E001
IIAB16	FILOSOFI...	2	0	2	3	4	Si	30	001	E001
IIAB21	DIBUJO P...	0	0	0	3	4	Si	30	002	E001

<< Exportar IMPRIMIR



- Para realizar Mantenimiento hacer clic en “Mantenimiento” y luego seleccionar Mantenimiento del profesor, como datos funcionales tenemos Actualizar, Modificar, Cancelar, Salir y una tabla Jtable1 para ver los datos de mysql.

MANTENIMIENTO SALON

FILTRO DE BUSQUEDA

PABELLON: TIPO SALON:

AULA: ESTADO:


CAPACIDAD AULA DESCRIPCION:

CAP. EVALUACION DISPONIBILIDAD

RESULTADO DE LA BUSQUEDA INGRESE POR CODIGO MOSTRAR TODO

Codigo	Numero	Cap Aula	Descripcion	Estado	Tipo de Salon	Pabellon
SA0001	S001	30	BUNEO	BUENO	1	2001
SA0002	S002	30	BUENO	BUENO	1	2001
SA0003	S003	25	BUNEO	BUENO	1	2001
SA0004	S004	30	BUENO	BUENO	1	2001

<< EXportar a Excel IMPRIMIR



- Después de ingresar los datos de manera correcta, asignar sus cursos a cada uno de los docentes.

A screenshot of a software window with a light blue background. At the top left is a small icon. The window contains four input fields: 'CODIGO' (empty text box), 'TIPO' (empty text box), 'PROFESOR' (dropdown menu with 'PR0001' selected), and 'CURSO' (dropdown menu with 'IIAA42' selected). Below these fields is a large empty rectangular area. At the bottom right, there are two buttons: 'GUARDAR' and 'SALIR'.

- Después de asignarle sus cursos a cada uno de los docentes ir a menú "GENERAR HORARIO" y asignarle los datos correspondientes clic en generar.

A screenshot of a software window titled 'Datos Generales' and 'Datos del Algoritmo Genetico'. The 'Datos Generales' section has two dropdown menus: 'Facultad:' with 'Ingenieria' selected and 'Escuela:' with 'Ingenieria de Sistemas' selected. The 'Datos del Algoritmo Genetico' section has four input fields: 'Minimo Tamaño Poblacion', 'Maximo Tamaño Poblacion', and 'Tasa de Mutacion'. A blue callout box with the text 'CLIC AQUÍ PARA GENERAR HORARIO' has two red arrows pointing to the 'GENERAR' button at the bottom right. The 'SALIR' button is also visible next to it.

- Después de asignarle sus cursos a cada uno de los docentes ir a menú “GENERAR HORARIO” y “CLIC EN REPORTE DE HORARIO”, veremos los horarios asignados por ciclos.

Mostrando Horarios

Ciclo 3

Horas	lunes	martes	miércoles	jueves	viernes	sab...	do...
7:00am	IAB35 SA0001	IAB35 SA0001	IAB35 SA0001	IAB34 SA0001	IAB31 PR0004 SA0001		
8:00am	IAB35 SA0001	IAB34 SA0001	IAB33 PR0001 SA0001	IAB31 PR0004 SA0001	IAB36 SA0001		
9:00am	IAB37 SA0001	IAB33 PR0001 SA0001	IAB33 PR0001 SA0001	IAB31 PR0004 SA0001	IAB34 SA0001		
10:00am	IAB36 SA0001	IAB34 SA0001	IAB36 SA0001	IAB37 SA0001	IAB36 SA0001		
11:00am	IAB31 PR0004 SA0001	IAB37 SA0001	IAB37 SA0001	IAB33 PR0001 SA0001	IAB37 SA0001		
12:00am	IAB37 SA0001	SA0001	SA0001	SA0001	SA0001		

- Después de asignarle sus cursos a cada uno de los docentes ir a menú “GENERAR HORARIO” y “CLIC EN REPORTE DE HORARIO”, veremos los horarios asignados por ciclos.

Mostrando Horarios

Ciclo 3

Ciclo 1

Ciclo 2

Ciclo 3

Ciclo 4

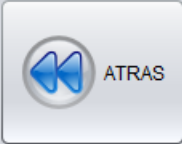
	lunes	martes	miércoles	jueves	viernes	sab...	do...
Ciclo 2	IAB35 SA0001	IAB35 SA0001	IAB35 SA0001	IAB34 SA0001	IAB31 PR0004 SA0001		
Ciclo 3	IAB35 SA0001	IAB34 SA0001	IAB33 PR0001 SA0001	IAB31 PR0004 SA0001	IAB36 SA0001		
Ciclo 4	IAB37 SA0001	IAB33 PR0001 SA0001	IAB33 PR0001 SA0001	IAB31 PR0004 SA0001	IAB34 SA0001		
10:00am	IAB36 SA0001	IAB34 SA0001	IAB36 SA0001	IAB37 SA0001	IAB36 SA0001		
11:00am	IAB31 PR0004 SA0001	IAB37 SA0001	IAB37 SA0001	IAB33 PR0001 SA0001	IAB37 SA0001		
12:00am	IAB37 SA0001	SA0001	SA0001	SA0001	SA0001		


- Después ir a menú Carga Académica y hacer clic en Horario por docente, para ver sus horas asignados a cada uno de los docentes.


DOCENTE: HUMBERTO SILVERA

HORARIO DEL DOCENTE:

Horas	lunes	martes	miercoles	jueves	viernes
7:00am	-- SA006	-- SA006	ALGORITMOS Y...	-- SA006	-- SA006
8:00am	-- SA006	TEORIA Y DISE...	-- SA006	-- SA006	ALGORITMOS Y...
9:00am	ALGORITMOS Y...	-- SA006	-- SA006	TEORIA Y DISE...	TEORIA Y DISE...
10:00am	TEORIA Y DISE...	ALGORITMOS Y...	-- SA006	ALGORITMOS Y...	DISEÑO WEB ...
11:00am	ALGORITMOS Y...	-- SA006	DISEÑO WEB ...	DISEÑO WEB ...	-- SA006
12:00am	-- SA006	-- SA006	-- SA006	-- SA006	-- SA006

 ATRAS

 IMPRIMIR



ANEXO 10: Código del algoritmo genético en el proyecto

➤ código de la clase administrador genético

ADMINISTRADOR GENETICO

```
public AdministradorGenetico(Dataset datos){
    this.datos = datos;
}

public void ejecutarAlgoritmoGenetico(){
    if(this.datos == null) return;// Si no hay datos
    ConfigGenetico.cerrar();// Evito que se realicen cambios a las
    configuraciones.
    // Inicio Algoritmo Genetico
    generarPoblacionInicial();
    evaluarIndividuos();
    imprimirPoblacion();
    for(int i = 0; i < ConfigGenetico.getLimiteGeneraciones(); i++){
        this.generacion++;
        reproducirIndividuos();
        mutarIndividuos();
        evaluarIndividuos();
    }
    // Fin Algoritmo Genetico
    ConfigGenetico.abrir();//
}
/**
 * Genera la población inicial.
 */
private void generarPoblacionInicial(){
    if(poblacion != null) return;// Ya no se genera un población inicial
    poblacion = new ArrayList();
    // Crear la población (Con la cantidad de individuos configurada -
    Dimensión población)
    for(int i = 0; i < ConfigGenetico.getDimensionPoblacion(); i++){
        // Crear un individuo
        poblacion.add(new Individuo(datos));
    }
}

private void reproducirIndividuos(){
    // Variables que usaremos
    Individuo individuo1, individuo2;
    Individuo hijo1, hijo2;
    for(int i = 0; i < poblacion.size()/2; i++){
        //Seleccionamos los padres
        individuo1 = poblacion.get(i*2);// el elemento par
        individuo2 = poblacion.get((i*2)+1);// el siguiente elemento
        // Obtenemos los hijo
```

```

        hijo1 = individuo1.getHijo(individuo2);
        hijo2 = individuo2.getHijo(individuo1);
        // Los agregamos a la poblacion (SIGUIENTE GENERACION)
        poblacion.set(i*2, hijo1);
        poblacion.set(i*2+1, hijo2);

    }
    // si la poblacion tiene tamaño impar agregamos el ultimo elemento
    // a la nueva generación
}

/**
 * Altera los individuos para acercarse más a la solución óptima.
 */
private void mutarIndividuos(){
    for(int i = 0; i < poblacion.size(); i++){// Recorro todos los elementos
        poblacion.get(i).mutar();
    }
}

private void evaluarIndividuos(){
    /* Calculamos las ponderaciones para cada individuo de lapoblacion*/
    // variables que usaremos
    int i = 0;
    Individuo otroIndividuo;
    if(mejorIndividuo == null){
        mejorIndividuo = poblacion.get(0);
        mejorIndividuo.evaluate();
        i = 1;
    }
    // Hallamos los ponderados de todos los individuos
    for(; i < poblacion.size(); i++){// desde el segundo elemento
        otroIndividuo = poblacion.get(i);
        otroIndividuo.evaluate();
        // Si el ponderado de un individuo es cero significa que es perfecto
        // Debería detenerse el algoritmo para ahorrar ciclos.
        // Se asume que para el siguiente if los ponderados no son negativos
        if(mejorIndividuo.getPonderado() > otroIndividuo.getPonderado()){
            // entonces el otro individuo es mejor
            // El ponderado menor corresponde al mejor individuo
            mejorIndividuo = otroIndividuo;// (ojo) si se desea almacenar los
mejores Individuos
            // de todas las generaciones, mejor seria clonarlos.
        }
    }
}
}

```

➤ CODIGO DEL INDIVIDUO

```
public Individuo(ArrayList<Cromosoma> cromosomas, Dataset datos){
    this.cromosomas = cromosomas;
    this.dataset = datos;
}
private void nacer(Dataset datos){
    ArrayList<Gen> fila;//Para los periodos (filas) del cromosoma
    ArrayList<ArrayList<Gen>> tablaGen;// Almacenara las filas
    String uncurso, unsalon, unprofesor;// Para el gen
    Gen unGen;
    datos.prepararParaIndividuo();//
    for(int ci = 0; ci < datos.getTotalCiclos(); ci++){// Los semestres (ciclos)
        tablaGen = new ArrayList<>();
        for(int i = 0; i < ConfigGenetico.getNumeroPeriodos(); i++){
            fila = new ArrayList();
            for(int j = 0; j < ConfigGenetico.getDias(); j++){
                unsalon = datos.getSalonAleatorio(ci);//
                if(datos.estaAsignado(ci, (byte)i, (byte)j)){// Está asignado
                    uncurso = datos.getCurso(ci, (byte)i, (byte)j);//
                    unprofesor = datos.getProfesorAsignado(uncurso);
                    unGen = new Gen(uncurso, unsalon, unprofesor);
                    unGen.setInamovible(true);// Iformamos que el Gen será
inamovible
                }else{// Aleatoriamente
                // Elegir normalmente
                uncurso = datos.getCursoAleatorio(ci);
                unprofesor = datos.getProfesorAleatorio(uncurso);//
                unGen = new Gen(uncurso, unsalon, unprofesor);
                }
                fila.add(unGen);
            }
            tablaGen.add(fila);
        }
        cromosomas.add(new Cromosoma(tablaGen));
    }
}
public Individuo getHijo(Individuo pareja){
    ArrayList<Gen> fila;//Para los periodos (filas) del cromosoma
    ArrayList<ArrayList<Gen>> tablaGen;// Almacenara las filas
    ArrayList<Cromosoma> cromosomasHijo = new ArrayList<>();
    Individuo elHijo;
    Cromosoma cromosomaDonante;
    int cicloElegido = (int)(Math.random()+cromosomas.size());
    for(int ci = 0; ci < cromosomas.size(); ci++){// Ciclos
        tablaGen = new ArrayList<>();
```

```

        if(cicloElegido == ci)cromosomaDonante = pareja.getCromosoma(ci);
        else cromosomaDonante = cromosomas.get(ci);
        for(int i = 0; i < cromosomaDonante.sizePeriodos(); i++){// Periodos
(Filas)
            fila = new ArrayList();
            for(int j = 0; j < cromosomaDonante.sizeDias(); j++){// Dias
(Columnas)
                //Creamos y agregamos el gen a una fila del cromosoma
                fila.add(cromosomas.get(ci).getGen(i, j).getClon());
            }
            // Agregamos la fila (periodo) a la tablaGen
            tablaGen.add(fila);
        }
        cromosomasHijo.add(new Cromosoma(tablaGen));
    }
    elHijo = new Individuo(cromosomasHijo, dataset);
    return elHijo;
}
public void mutar(){
    mutarCruceIntercambio();
    mutarUnirCursos();
    mutarCursoBloque();
}
private void mutarCruceIntercambio(){
    Gen gnComparar, gnSiguiete, gnTemp1, gnTemp2;
    int periodo, dia;
    cruce:for(int ci = 0; ci < cromosomas.size()-1; ci++){//Para seleccionar
un ciclo
        for(int i = 0; i < cromosomas.get(0).sizePeriodos(); i++){// periodos
(Filas)
            for(int j = 0; j < cromosomas.get(0).sizeDias(); j++){// dias
(Columnas)
                gnComparar = cromosomas.get(ci).getGen(i, j);
                if(gnComparar.getCurso().equals(""))continue;// si es nulo ya no
buscamos
                if(gnComparar.isInamovible()) continue;// o podemos romper el
ciclo
                for(int nextci = ci+1; nextci <cromosomas.size(); nextci++){// Los
otros ciclos
                    gnSiguiete = cromosomas.get(nextci).getGen(i, j);
                    if(gnComparar.esProfesorIguale(gnSiguiete)){// Existe cruce
                        if(j < cromosomas.get(0).sizeDias()-1)// el penultimo dia
(columna)
                            dia =
j+(int)(Math.random()*(cromosomas.get(0).sizeDias() - j));
                            //}else if(j>0){// Por si es que la cantidad de dias es 2

```



```

        gnTemp2 = unCromosoma.getGen(periodo, dia);//
        unCromosoma.setGen(gnTemp2, k, j);
        unCromosoma.setGen(gnTemp1, periodo, dia);
        //}
        break unir;
    }
    posant = (byte)k;
}
}
}
}
}
}
}
private void mutarCursoBloque(){
    // MOVEMOS LA PRIMERA CLASE ENCONTRADA QUE SUPERA
MÁS DE TRES HORAS
    //unCromosoma = cromosomas.get(ciAleatorio);
    // Variables que usaremos
    Gen gen, nextgen, gnTemp1, gnTemp2;
    int periodo, dia;
    Cromosoma unCromosoma;
    byte posant;// Valor anterior del gen igual
    int ciAleatorio = (int)(Math.random()*cromosomas.size());
    unCromosoma = cromosomas .get(ciAleatorio);
    int canConsecutivo;
    separar:for(int j = 0; j < unCromosoma.sizeDias(); j++){// columnas
(dias)
        for(int i = 0; i < unCromosoma.sizePeriodos(); i++){// filas (periodos)
            gen = unCromosoma.getGen(i, j);
            posant = (byte)i;
            if(gen.getCurso().equals(""))continue;// si es nulo ya no buscamos
            if(gen.isInamovible())continue;// si es inamovible ya no buscamos
            canConsecutivo = 0;
            for(int k = i + 1; k < unCromosoma.sizePeriodos(); k++){
                nextgen = unCromosoma.getGen(k, j);
                if(nextgen.isInamovible())continue;// (ojo)
                if(gen.esCursoIguar(nextgen)){// Existe repeticion en un dias
                    if((posant + 1) != k){// No son consecutivos
                        break;//Pasamos a otro
                    }else{
                        canConsecutivo++;
                    }
                }
            }
        }
    }
    if(canConsecutivo>ConfigGenetico.PER_CONSECUTIVOS){
        if(j < unCromosoma.sizeDias()-1)// el penultimo dia
(columna)
            dia = j+(int)(Math.random()*(unCromosoma.sizeDias()
- j));
}
}

```



```

        for(int i = 0; i < unCromosoma.sizePeriodos(); i++){// Filas
(Periodos)
            gen = unCromosoma.getGen(i, j);
            posant = (byte)i;
            canConsecutivo = 0;
            if(gen.getCurso().equals(""))continue;// Nos saltamos los cursos
vacios
            if(gen.isInamovible())continue;// Nos saltamos los inamovibles
            for(int k = i + 1; k < unCromosoma.sizePeriodos(); k++){
                nextgen = unCromosoma.getGen(k, j);
                if(gen.esCursoIguales(nextgen)){// El curso se repite el mismo
día
                    if((posant + 1) != k){// no son consecutivos
                        puntuacion += ConfigGenetico.RF;// Aumentamos +1
                        canConsecutivo = 0;
                    }else{// Son consecutivos
                        canConsecutivo++;
                        if(canConsecutivo >
ConfigGenetico.PER_CONSECUTIVOS){
                            // castigar con Restriccion Fuerte(+5)
                            puntuacion += ConfigGenetico.RF;
                        }
                    }
                    posant = (byte)k;
                }
            }
        }
    }
    return puntuacion;
}
private int getEvaluarCruce(){
    Cromosoma unCromosoma;
    short puntuacion = 0;
    Gen gnComparar, gnSiguiente;
    for(int ci = 0; ci < cromosomas.size()-1; ci++){//Para seleccionar un ciclo
        unCromosoma = cromosomas.get(ci);
        for(int i = 0; i < unCromosoma.sizePeriodos(); i++){// periodos (Filas)
            for(int j = 0; j < unCromosoma.sizeDias(); j++){// dias (Columnas)
                gnComparar = unCromosoma.getGen(i, j);
                if(gnComparar.getProfesor().equals(""))continue;
                if(gnComparar.isInamovible())continue;// No comparamos l{os
inamovibles
                for(int nextci = ci+1; nextci < cromosomas.size(); nextci++){// Los
otros ciclos
                    gnSiguiente = cromosomas.get(nextci).getGen(i, j);

```

```

        // Comparamos
        if(gnComparar.esProfesorIguales(gnSiguiente)){// Existe cruce
            // CASTIGAMOS
            puntuacion += ConfigGenetico.RF;// Aumentamos +5
            break;
        }
    }
}
return puntuacion;
}
private int getEvaluarCursosYPeriodos(){
    Cromosoma unCromosoma;
    short puntuacion = 0;
    ArrayList<String> cacheCursos;// aqui pondremos los cursos
    String curso, crsmCurso;
    int perOblCurso, perCrsmCurso = 0, perDiferencia;
    cacheCursos = new ArrayList<>();
    for(int ci = 0; ci < cromosomas.size(); ci++){
        unCromosoma = cromosomas.get(ci);
        curso = unCromosoma.getGen(0, 0).getCurso();//Obtenemos el Curso
del primer Gen
        //Comparamos y contamos cada una de las ocurrencias
        for(int i = 0; i < unCromosoma.sizePeriodos(); i++){// Filas (peridoos)
            for(int j = 0; j < unCromosoma.sizeDias(); j++){// Columnas (dias)
                if(unCromosoma.getGen(i, j).isInamovible())continue;
                crsmCurso = unCromosoma.getGen(i, j).getCurso();
                if(crsmCurso.equals(""))continue;// Nos saltamos los vacios
                if(curso.equals(crsmCurso)){
                    perCrsmCurso++;//Aumentamos una hora
                }else{// Son distintos
                    //Agregamos el curso del cromosoma distinto a la cache
                    cacheCursos.add(crsmCurso);
                }
            }
        }
        // Compruebo cuantas veces me pase de la raya jeje
        if(!curso.equals("")){
            perOblCurso = dataset.getTotalPeriodosCurso(curso);
            perDiferencia = perOblCurso - perCrsmCurso;
            puntuacion += Math.abs(perDiferencia)*ConfigGenetico.RF;
        }
        // Iniciamos todo de nuevo pero más eficiente
        while(!cacheCursos.isEmpty()){// mientras no esté vacío
            perCrsmCurso = 0;

```

```

        curso = cacheCursos.get(0);
        for(int i = 0 ; i < cacheCursos.size(); i++){
            crsmCurso = cacheCursos.get(i);
            if(curso.equals(crsmCurso)){// contamos
                perCrsmCurso++;
                cacheCursos.remove(i);//
                i-=1;// restamos uno para mantener la logica
            }
        }
        // Castigamos
        if(curso.equals("")){
            perOblCurso = dataset.getTotalPeriodosCurso(curso);
            perDiferencia = perOblCurso - perCrsmCurso;
            puntuacion += Math.abs(perDiferencia)*ConfigGenetico.RF;
        }
    }
}
return puntuacion;
}
private int getEvaluarProfesoresYPeriodos(){
    Cromosoma unCromosoma;
    short puntuacion = 0;
    String profesor, crsmProfesor;
    int perOblProfesor, perCrsmProfesor = 0, perDiferencia;
    ArrayList<String> cacheProfesor = new ArrayList<>();
    profesor = cromosomas.get(0).getGen(0, 0).getProfesor();
    for(int ci = 0; ci < cromosomas.size(); ci++){
        unCromosoma = cromosomas.get(ci);
        for(int i = 0; i < unCromosoma.sizePeriodos(); i++){// periodos (filas)
            for(int j = 0; j < unCromosoma.sizeDias(); j++){// dias (columnas)
                if(unCromosoma.getGen(i, j).isInamovible())continue;
                crsmProfesor = unCromosoma.getGen(i, j).getProfesor();
                if(crsmProfesor.equals("")) continue;// si es nulo no lo tomamos
en cuenta
                if(profesor.equals(crsmProfesor)){// Profesor cumple un periodo
                    perCrsmProfesor++;
                }else{//Guardamos el profesor en la cache.
                    cacheProfesor.add(crsmProfesor);
                }
            }
        }
    }
}
if(!profesor.equals("")){
    // Castigamos
    perOblProfesor = this.dataset.getTotalPeriodosProfesor(profesor);
    perDiferencia = perOblProfesor - perCrsmProfesor;
}

```

```

puntuacion += Math.abs(perDiferencia)*ConfigGenetico.RF;
}
while(!cacheProfesor.isEmpty()){
    // Inicializamos
    profesor = cacheProfesor.get(0);
    perCrsmProfesor = 0;
    for(int i = 0; i < cacheProfesor.size(); i++){
        crsmProfesor = cacheProfesor.get(i);
        if(profesor.equals(crsmProfesor)){
            perCrsmProfesor++;
            cacheProfesor.remove(i);
            i -= 1;
        }
    }
    //Castigamos
    perOblProfesor = this.dataset.getTotalPeriodosProfesor(profesor);
    perDiferencia = perOblProfesor - perCrsmProfesor;
    puntuacion += Math.abs(perDiferencia)*ConfigGenetico.RF;
return puntuacion;
}

```

ANEXO 11: Resultados del horario por ciclo con algoritmo genético CICLO I

HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
7:00am	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206
8:00am	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206
9:00am	TECNICAS DE APRENDIZAJE UNIVERSITARIO -- S-206	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206	FILOSOFIA Y ETICA -- S-206	INTRODUCCION A LA INFORMATICA Y COMPUTACION Flor de Luz S-206	TECNICAS DE APRENDIZAJE UNIVERSITARIO -- S-206
10:00am	FILOSOFIA Y ETICA -- S-206	COMUNICACION ORAL Y ESCRITA -- S-206	FILOSOFIA Y ETICA -- S-206	TECNICAS DE APRENDIZAJE UNIVERSITARIO -- S-206	MATEMATICA I - S-206
11:00am	COMUNICACION ORAL Y ESCRITA -- S-206	COMUNICACION ORAL Y ESCRITA -- S-206	COMUNICACION ORAL Y ESCRITA -- S-206	MATEMATICA I -- S-206	MATEMATICA I - S-206
12:00am	MATEMATICA I -- S-206	MATEMATICA I -- S-206	MATEMATICA I -- S-206	-- -- S-206	-- -- S-206

CICLO II

HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
7:00am	METODOLOIA DE LA PROGRAMACIÓN Flor de Luz S-202	MATEMATICA II -- S-202	METODOLOIA DE LA PROGRAMACIÓN Flor de Luz S-202	CALCULO I -- S-202	METODOLOIA DE LA PROGRAMACIÓN Flor de Luz S-202
8:00am	METODOLOIA DE LA PROGRAMACIÓN Flor de Luz S-202	MATEMATICA II -- S-202	METODOLOIA DE LA PROGRAMACIÓN Flor de Luz S-202	MATEMATICA II -- S-202	METODOLOIA DE LA PROGRAMACIÓN Flor de Luz S-202
9:00am	ECONOMIA -- S-202	ECONOMIA -- S-202	CALCULO I -- S-202	MATEMATICA II -- S-202	MATEMATICA II -- S-202
10:00am	CALCULO I -- S-202	CALCULO I -- S-202	CALCULO I -- S-202	-- -- S-202	-- -- S-202
11:00am	DIBUJO PARA INGENIERIA ROBERTO S-202	-- -- S-202	-- -- S-202	-- -- S-202	DIBUJO PARA INGENIERIA ROBERTO S-202
12:00am	DIBUJO PARA INGENIERIA ROBERTO S-202	-- -- S-202	-- -- S-202	-- -- S-202	DIBUJO PARA INGENIERIA ROBERTO S-202

CICLO III

HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
7:00am	FISICA -- S-208	DISEÑO WEB JULIO CESAR S-208	CALCULO II -- S-208	DISEÑO WEB JULIO CESAR S-208	FISICA -- S-208
8:00am	FISICA -- S-208	DISEÑO WEB JULIO CESAR S-208	FISICA -- S-208	DISEÑO WEB JULIO CESAR S-208	FISICA -- S-208
9:00am	CALCULO II -- S-208	COMPUTACION Y SISTEMAS OPERATIVOS I HERWIN ALAY S-208	CALCULO II -- S-208	TEORIA Y DISEÑO DE BASE DE DATOS RICAR ARTEMIO S-208	Matematica Discreta Edwin Alcides S-208
10:00am	ALGORITMOS Y ESTRUCTURA DE DATOS JULIO CESAR S-208	COMPUTACION Y SISTEMAS OPERATIVOS I HERWIN ALAY S-208	TEORIA Y DISEÑO DE BASE DE DATOS RICAR ARTEMIO S-208	TEORIA Y DISEÑO DE BASE DE DATOS RICAR ARTEMIO S-208	Matematica Discreta Edwin Alcides S-208
11:00am	ALGORITMOS Y ESTRUCTURA DE DATOS JULIO CESAR S-208	Matematica Discreta Edwin Alcides S-208	TEORIA Y DISEÑO DE BASE DE DATOS RICAR ARTEMIO S-208	FISICA -- S-208	CALCULO II -- S-208
12:00am	-- -- S-208	Matematica Discreta Edwin Alcides S-208	ALGORITMOS Y ESTRUCTURA DE DATOS JULIO CESAR S-208	COMPUTACION Y SISTEMAS OPERATIVOS I HERWIN ALAY S-208	ALGORITMOS Y ESTRUCTURA DE DATOS JULIO CESAR S-208

CICLO IV

HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
7:00am	Teoría General de sistemas RICAR ARTEMIO S-201	Estadística Discriptiva -- S-201	Estadística Discriptiva -- S-201	Computadores y Sistemas Operativos II Carlos Yinmel S-201	Estadística Discriptiva -- S-201
8:00am	Teoría General de sistemas RICAR ARTEMIO S-201	Computadores y Sistemas Operativos II Carlos Yinmel S-201	Computadores y Sistemas Operativos II Carlos Yinmel S-201	Estadística Discriptiva -- S-201	Computadores y Sistemas Operativos II Carlos Yinmel S-201
9:00am	Programación JUAN JOSE S-201	Computadores y Sistemas Operativos II Carlos Yinmel S-201	Programación JUAN JOSE S-201	-- -- S-201	Programación JUAN JOSE S-201
10:00am	Programación JUAN JOSE S-201	-- -- S-201	Programación JUAN JOSE S-201	-- -- S-201	Programación JUAN JOSE S-201
11:00am	BASE DE DATOS I Flor de Luz S-201	-- -- S-201	-- -- S-201	-- -- S-201	-- -- S-201
12:00am	BASE DE DATOS I Flor de Luz S-201	-- -- S-201	-- -- S-201	-- -- S-201	-- -- S-201

CICLO V

HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
7:00am	Investigación Tecnológica e Innovación Cecilia Edith S-205	Investigación Tecnológica e Innovación Cecilia Edith S-205	Estadística Inferencial -- S-205	Estadística Inferencial -- S-205	Programación Web RUBEN S-205
8:00am	Investigación Tecnológica e Innovación Cecilia Edith S-205	Investigación Tecnológica e Innovación Cecilia Edith S-205	Estadística Inferencial -- S-205	Programación Web RUBEN S-205	Programación Web RUBEN S-205
9:00am	Redes I HERWIN ALAY S-205	Sistemas Digitales ROBERTO S-205	Redes I HERWIN ALAY S-205	Sistemas Digitales ROBERTO S-205	Programación Web RUBEN S-205
10:00am	Redes I HERWIN ALAY S-205	Sistemas Digitales ROBERTO S-205	Redes I HERWIN ALAY S-205	Sistemas Digitales ROBERTO S-205	Estadística Inferencial -- S-205
11:00am	Base de datos II Edwin S-205	Programación Web RUBEN S-205	Redes I HERWIN ALAY S-205	Programación Web RUBEN S-205	-- -- S-205
12:00am	Base de datos II Edwin S-205	-- -- S-205	Base de datos II Edwin S-205	-- -- S-205	-- -- S-205

CICLO VI

HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
7:00am	ARQUITECTURA DE PROCESADORES II Enrique Edgardo S-209	INVESTIGACIÓN DE OPERACIONES I Enrique Edgardo S-209	ARQUITECTURA DE PROCESADORES II Enrique Edgardo S-209	INVESTIGACIÓN DE OPERACIONES I Enrique Edgardo S-209	ADMINISTRACIÓN DE PERSONAL -- S-209
8:00am	ARQUITECTURA DE PROCESADORES II Enrique Edgardo S-209	INVESTIGACIÓN DE OPERACIONES I Enrique Edgardo S-209	ARQUITECTURA DE PROCESADORES II Enrique Edgardo S-209	INVESTIGACIÓN DE OPERACIONES I Enrique Edgardo S-209	CONTABILIDAD GENERAL -- S-209
9:00am	ARQUITECTURA DE PROCESADORES II Enrique Edgardo S-209	TEORÍA Y DISEÑO DE BASE DE DATOS II IVAN S-209	TEORÍA Y DISEÑO DE BASE DE DATOS II IVAN S-209	REDES II LEONIDAS S-209	CONTABILIDAD GENERAL -- S-209
10:00am	PROGRAMACIÓN III RUBEN S-209	REDES II LEONIDAS S-209	PROGRAMACIÓN III RUBEN S-209	PROGRAMACIÓN III RUBEN S-209	PROGRAMACIÓN III RUBEN S-209
11:00am	PROGRAMACIÓN III RUBEN S-209	TEORÍA Y DISEÑO DE BASE DE DATOS II IVAN S-209	PROGRAMACIÓN III RUBEN S-209	TEORÍA Y DISEÑO DE BASE DE DATOS II IVAN S-209	REDES II LEONIDAS S-209
12:00am	ADMINISTRACIÓN DE PERSONAL -- S-209	CONTABILIDAD GENERAL -- S-209	REDES II LEONIDAS S-209	PROGRAMACIÓN III RUBEN S-209	MULTIMEDIA JUAN JOSE S-209

CICLO VII

HORA	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES
7:00am	INGENIERÍA DE SOFTWARE I ANGEL FERNANDO S-204	INVESTIGACIÓN DE OPERACIONES II Edwin Alcides S-204	INGENIERÍA DE SOFTWARE I ANGEL FERNANDO S-204	COMPUTACIÓN GRÁFICA Edwin Alcides S-204	INGENIERÍA DE SOFTWARE I ANGEL FERNANDO S-204
8:00am	INGENIERÍA DE SOFTWARE I ANGEL FERNANDO S-204	INVESTIGACIÓN DE OPERACIONES II Edwin Alcides S-204	INGENIERÍA DE SOFTWARE I ANGEL FERNANDO S-204	COMPUTACIÓN GRÁFICA Edwin Alcides S-204	INGENIERÍA DE SOFTWARE I ANGEL FERNANDO S-204
9:00am	SIMULACIÓN DE SISTEMAS RICAR ARTEMIO S-204	INVESTIGACIÓN DE OPERACIONES II Edwin Alcides S-204	SIMULACIÓN DE SISTEMAS RICAR ARTEMIO S-204	INGENIERÍA EMPRESARIAL HUMBERTO S-204	INGENIERÍA EMPRESARIAL HUMBERTO S-204
10:00am	SIMULACIÓN DE SISTEMAS RICAR ARTEMIO S-204	INVESTIGACIÓN DE OPERACIONES II Edwin Alcides S-204	SIMULACIÓN DE SISTEMAS RICAR ARTEMIO S-204	INGENIERÍA DE COSTOS - S-204	INGENIERÍA EMPRESARIAL HUMBERTO S-204
11:00am	INGENIERÍA EMPRESARIAL HUMBERTO S-204	PLANEAMIENTO INFORMÁTICO Edwin S-204	INGENIERÍA DE COSTOS -- S-204	INGENIERÍA DE COSTOS - S-204	INGENIERÍA DE COSTOS -- S-204
12:00am	COMPUTACIÓN GRÁFICA Edwin Alcides S-204	PLANEAMIENTO INFORMÁTICO Edwin S-204	-- -- S-204	PLANEAMIENTO INFORMÁTICO Edwin S-204	-- -- S-204

CICLO VIII

HORA	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES
7:00am	INGENIERÍA DE INFORMACIÓN RICAR ARTEMIO S-203	FORMULACIÓN Y EVALUACIÓN DE PROYECTOS Cecilia Edith S-203	INGENIERÍA DE INFORMACIÓN RICAR ARTEMIO S-203	FORMULACIÓN Y EVALUACIÓN DE PROYECTOS Cecilia Edith S-203	SISTEMA DE INFORMACIÓN GERENCIAL RICAR S-203
8:00am	INGENIERÍA DE INFORMACIÓN RICAR ARTEMIO S-203	FORMULACIÓN Y EVALUACIÓN DE PROYECTOS Cecilia Edith S-203	INGENIERÍA DE INFORMACIÓN RICAR ARTEMIO S-203	FORMULACIÓN Y EVALUACIÓN DE PROYECTOS Cecilia Edith S-203	GESTIÓN FINANCIERA -- S-203
9:00am	INGENIERÍA DE INFORMACIÓN RICAR ARTEMIO S-203	FORMULACIÓN Y EVALUACIÓN DE PROYECTOS Cecilia Edith S-203	APRENDIZAJE ELECTRÓNICO LEONIDAS S-203	SISTEMA DE INFORMACIÓN GERENCIAL RICAR S-203	SISTEMA DE INFORMACIÓN GERENCIAL RICAR S-203
10:00am	APRENDIZAJE ELECTRÓNICO LEONIDAS S-203	GESTIÓN FINANCIERA -- S-203	GESTIÓN FINANCIERA -- S-203	GESTIÓN FINANCIERA -- S-203	SISTEMA DE INFORMACIÓN GERENCIAL RICAR S-203
11:00am	INGENIERÍA DE SOFTWARE II ANGEL FERNANDO S-203	APRENDIZAJE ELECTRÓNICO LEONIDAS S-203	SISTEMA DE INFORMACIÓN GERENCIAL RICAR S-203	APRENDIZAJE ELECTRÓNICO LEONIDAS S-203	-- -- S-203
12:00am	INGENIERÍA DE SOFTWARE II ANGEL FERNANDO S-203	-- -- S-203	-- -- S-203	-- -- S-203	-- -- S-203

CICLO IX

HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
7:00am	TÓPICOS ESPECIALES JUAN JOSE S-207	INGENIERÍA DE SOFTWARE III ANGEL FERNANDO S-207	TÓPICOS ESPECIALES JUAN JOSE S-207	INGENIERÍA DE SOFTWARE III ANGEL FERNANDO S-207	TESIS I Cecilia Edith S-207
8:00am	TÓPICOS ESPECIALES JUAN JOSE S-207	INGENIERÍA DE SOFTWARE III ANGEL FERNANDO S-207	TÓPICOS ESPECIALES JUAN JOSE S-207	INGENIERÍA DE SOFTWARE III ANGEL FERNANDO S-207	TESIS I Cecilia Edith S-207
9:00am	AUTOMATIZACIÓN ROBERTO S-207	INGENIERÍA DE SOFTWARE III ANGEL FERNANDO S-207	AUTOMATIZACIÓN ROBERTO S-207	INGENIERÍA DE SOFTWARE III ANGEL FERNANDO S-207	TESIS I Cecilia Edith S-207
10:00am	AUTOMATIZACIÓN ROBERTO S-207	ADMINISTRACIÓN DE RECURSOS INFORMÁTICOS LEONIDAS S-207	AUTOMATIZACIÓN ROBERTO S-207	ADMINISTRACIÓN DE RECURSOS INFORMÁTICOS LEONIDAS S-207	ADMINISTRACIÓN DE RECURSOS INFORMÁTICOS LEONIDAS S-207
11:00am	AUTOMATIZACIÓN ROBERTO S-207	SISTEMAS DE COMUNICACIONES I HERWIN ALAY S-207	ADMINISTRACIÓN DE RECURSOS INFORMÁTICOS LEONIDAS S-207	-- -- S-207	-- -- S-207
12:00am	-- -- S-207	SISTEMAS DE COMUNICACIONES I HERWIN ALAY S-207	-- -- S-207	SISTEMAS DE COMUNICACIONES I HERWIN ALAY S-207	-- -- S-207

CICLO X

HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
7:00am	SISTEMAS DE COMUNICACIONES II HUMBERTO S-210	SEGURIDAD Y AUDITORÍA DE SISTEMAS IVAN S-210	SEGURIDAD Y AUDITORÍA DE SISTEMAS IVAN S-210	SISTEMAS DE COMUNICACIONES II HUMBERTO S-210	INTELIGENCIA ARTIFICIAL HUMBERTO S-210
8:00am	SISTEMAS DE COMUNICACIONES II HUMBERTO S-210	SISTEMAS DE COMUNICACIONES II HUMBERTO S-210	SISTEMAS DE COMUNICACIONES II HUMBERTO S-210	INTELIGENCIA ARTIFICIAL HUMBERTO S-210	SEGURIDAD Y AUDITORÍA DE SISTEMAS IVAN S-210
9:00am	SEGURIDAD Y AUDITORÍA DE SISTEMAS IVAN S-210	TESIS II JUAN JOSE S-210	CONTROL DE CALIDAD DE SOFTWARE IVAN S-210	TESIS II JUAN JOSE S-210	CONTROL DE CALIDAD DE SOFTWARE IVAN S-210
10:00am	INTELIGENCIA ARTIFICIAL HUMBERTO S-210	TESIS II JUAN JOSE S-210	SEGURIDAD Y AUDITORÍA DE SISTEMAS IVAN S-210	TESIS II JUAN JOSE S-210	INTELIGENCIA ARTIFICIAL HUMBERTO S-210
11:00am	CONTROL DE CALIDAD DE SOFTWARE IVAN S-210	CONTROL DE CALIDAD DE SOFTWARE IVAN S-210	PROGRAMACIÓN AVANZADA HERWIN ALAY S-210	CONTROL DE CALIDAD DE SOFTWARE IVAN S-210	PROGRAMACIÓN AVANZADA HERWIN ALAY S-210

ANEXO 12: Dato recogido del libro de acta de distribución de carga académica de la EPIS

14



ACTA DE REUNION - DAITI - REAFIRMACION CARGA ACADEMICA
Y HORARIOS - 2014-

Siendo las 10:00 del día 10 de Marzo del 2014, se inicia la reunión. No habiendo quorum, solo asistieron

- Ing Herwin Alayn Huillcan Bacs
- Ing Flor de Luz Palomino Valderrama
- Ing Richard Flores Condori
- Ing Edwin Roque Tito

Luego se procedió a hacer entrega de dispositivos (Disco Duro Externo) capacidad 1 TB, convocando por segunda vez para el día Martes 11 de Marzo del presente de carácter obligatorio.

No habiendo más que hablar, se levanta la sesión y firman los presentes, siendo las 10:30.

Ing. Richard Flores C.

Ing. Flor de Luz Palomino V.

Herwin Huillcan Bacs.

UNIVERSIDAD NACIONAL
JOSÉ MARÍA ARGUEDAS
Msc. Ing. Edwin Roque Tito
JEFE DEL DEPARTAMENTO ACADÉMICO
DE INGENIERÍA Y TECNOLOGÍA INFORMÁTICA



Acta de Reunión DAITI - Ratificación de Carga Horaria para labores Académicas

Siendo las 11:20 am. del día 11 de Marzo del 2014, se da inicio a la reunión de docentes del departamento Académico de Ingeniería y Tecnología Informática con la asistencia del

Jug. Juan José Oro Cerrón

Jug. Herwin Huilken Boca

Jug. Enrique Condor Tinaco

Jug. Edwin Ramos Velasquez

Jug. Roberto Quispe Quispe

Jug. Flor de Ley palmiro Valdivia

Jug. Edwin Rogue Tito

El jefe de DAITI Ing. Rogue da a conocer que las horas serán repartidas en el turno de la mañana y solicita opiniones y sugerencias a los miembros del departamento, el Ing. Enrique Condor argumenta que puede existir 2 grupos en todo caso se podría generar un grupo en la tarde. Se da a conocer el orden de elección de las asignaturas, respetando el reglamento correspondiente y queda establecido de la siguiente manera:

- x ————— x ————— x ————— x ————— x
- ▷ Ing. Cecilia Garcia Rivas plata : - teoría General de sistemas (V ciclo) 03 HT
- Planeamiento Informático (VII ciclo) 05 HT
- Formulación y evaluación de Proj. Informáticas (VIII 05 HT)
- ▷ Ing. Angel NAVARRO Raymundo: - Ingeniería Empresarial (04 H.) VII
▷ Sistemas de Información general (05 VIII)
▷ Diseño Asistido por PC (04 II)
▷ Seguridad y Auditorio Inf. (05 X)
- ▷ Ing. Edwin Ramm Velasquez → Algoritmos y estruct. de datos I (06 III)



Algoritmos y estructura de datos II (06 IV)

Ing. Juan José Ore Corroñ: ▶ Ing. Administrativa (04 horas I)

▶ TESIS II (04 horas II)

Ing. Hermin Huillcas Baca: ▶ Redes I (04 horas III)

▶ Redes II (04 horas IV)

▶ Sistemas de Comunicación I (05 horas V)

Ing. Richard Flores Condori: - Teoría y diseño de BD I (07 horas VI)

- Simulación de Sistemas (04 horas VII)

Ing. Enrique Condor Tinoco: ▶ Programación II (06 horas VIII)

▶ Administración de R. Inform. (05 horas IX)

Ing. Flor de Luz Palomino: ▶ Informática I (06 horas X)

▶ Informática II (06 horas XI)

Ing. Edwin Requena Tito: - Ing. de Software I (06 horas XII)

- Control de Calidad de SW (05 horas XIII)

Ing. Edwin Maguero Flores: ▶ Sistemas Operativos II (05 horas XIV)

▶ Sistemas de Comunicación II (05 horas XV)

▶ Inteligencia Artificial (05 horas XVI)


Ing. Roberto Quispe Quispe: ▶ Sistemas Digitales I (06 horas XVII)

▶ Arg. de Procesadores I (05 horas XVIII)

▶ Automatización (05 horas XIX)

El director saliente Ing. Edwin Román V. comunica al recibido en calidad de donación una impresora multifuncional de Marca Kyocera, Modelo ECOSYS FS-1025 MFP para uso exclusivo de los docentes del departamento Academia de Ingeniería y Tecnología Informática, siendo la Jpm. del día 14 de Marzo se cierra el acta y se firman al pie en señal de conformidad.


Ing. Richard Flores


Ing. Enrique Condor Tinoco



ACTO DE REUNION PARA DISTRIBUCION DE CARGA LECTIVA N.º 2014-II

Siendo las 4: p.m del 23 de julio del 2014 en la ciudad de TOTA San Jeronimo. Se reunieron los docentes ordinarios del DAITI con la finalidad de acordar los siguientes temas.

- Distribucion de carga horaria lectiva 2014-II de los docentes ordinarios del DAITI.

El jefe de departamento Roque Tito a explicado la forma que se eligiran las asignaturas y sera de acuerdo a los siguientes:

- a) Ing. Cecilia Garcia Rivas plato
 - Planamiento informático (7) 5h
 - Tesis I (9) 3h.
 - Ing. administrativa (5) 4h
- b) Ing. Angel Navarro
 - Ing. Software I (7) 6h
 - Formulacion de proyectos inf. (8) 5h
 - seguridad y auditoria de sistemas (10) 5h.
- c) Ing. Edwin Ramos.
 - algoritmo estructura y datos I (3) 6h.
 - algoritmo estructura y datos II (4) 6h.
- d) Ing. Juan Jose Ore Carran
 - sistemas de Inf. Gerencial 2 (8) 5h
 - Tesis II (10) 4h
- e) Ing. Richard Flores C.
 - Teoria y Diseño de Base de datos I (5) 7h.
 - Ing. de la informacion (8) 5h.



f) Ing. Herwin Huilca Bacu

Sistemas Operativos I (3) 4h.

Sistema Comunicaciones I (9) 5h.

Redes I (5) 4h.

g) Ing. Enrique Cordero Trino

Programación II (5) 6h.

Admin. Recursos Informáticos (9) 5h.

h) Ing. Flor de los pelomino

Informática I (1) 5h.

Informática II (2) 6h.

i) Ing. Edwin Roque Tito

Teoría y Diseño B. D. II (6) 5h.

Teoría General de Sistemas (5) 3h.

Ing. Empresarial (2) 4h.

j) Ing. Edwin Maguila

Sistemas Operativos II (4) 5h.

Investigación de Operaciones I (6) 4h.

Temas Especialidades (9) 4h.

k) Ing. Roberto Quispe

Sistemas Digitales (4) 6h.

Arquitectura Procesador I (3) 5h.

Arquitectura Procesador II (6) 5h.

En cada caso los docentes también seleccionaron el horario respectivo de sus asignaturas seleccionadas.



Acta de reunión de Docentes
del Departamento Académico de Ingeniería y Tecnología.
Siendo 11:05 a.m del día 10 de diciembre del 2014,
en las instalaciones de la sede académica el Doctoral-
UNAJTMS en la sala de docentes se llega a los siguientes
acuerdos:

El jefe de departamento Msc. Ing. Edwin Roque Tito
explica sobre los horas y la asignación de cursos en
el departamento académico, luego se procede con la
designación de cursos de la siguiente forma:

1º) Ing. Cecilia Rojas Plata:

- Ingeniería Administrativa	(II)	(4 hrs)
- Formulación de Proyectos	(VIII)	(5)
- Tesis I	(IX)	(3)
- Tesis II	(X)	4
Total		<u>16 hrs</u>

2º) Ing. Angel Navarro Raymundo

- Ingeniería de Software I	(VII)	(6)
- " " " II	(VIII)	(6)
- Tópicos Especiales	(IX)	(4)
Total		<u>16 hrs</u>

3º) Ing. Edwin Ramos Velásquez

- Teoría y diseño de Base de Datos II	(II)	(5)
- Sistemas Digitales	(IV)	(6)
- Algoritmos y Estructura de Datos I	(III)	(6)
Total		<u>17</u>



4^{to}) Ing. Juan Jose Ori Cerroñ
- Programación I (IV) (6)
- Gestión Financiera (VIII) (4)
- Inteligencia Artificial (IX) (5)
Total 15

5^{to}) Ing. Richard Flores Condori
- Teoría y Base de Datos I (I) (7)
- Simulación de Sistemas (VII) (4)
- Ingeniería de Información (VIII) (5)
Total 16

6^{to}) Ing. Herwin Huilcen Baca
- Sistemas Operativos II (IV) (5)
- Redes II (VI) (4)
- Comunicaciones I (IX) (5)
- Redes I (V) (4)
Total 18

7^{mo}) Ing. Enrique Condori Tiucro
- Programación II (V) (6)
- Computación Gráfica (VII) (4)
- Administración de Recursos Informáticos (IX) (5)
Total 15 hrs

8^{vo}) Ing. Flor Palomino Valderrama
- Introducción a la informática (I) (6)
- Metodología de la programación (II) (6)
- Dibujo para ingeniería (III) (4)

9^{no}) Ing. Edwin Roque Tito
- Teoría General de Sistemas (V) (3)
- Sist. Operativos (III) (4)
- Comercio electrónico (VIII) (4)
- Sist. de Información General (VII) (5)
Total 16



10^{mo}) Ing. Roberto Quirope Quirope.

- Arquitectura de Procesadores I	(I)	(5)
- " " " " II	(II)	(5)
- Automatización	(IX)	(5)
Total.		15

El ingeniero Edwin Roque Tito informa sobre el traspaso de Departamento Académico el cual se acuerda observarlo para su anulación por lo siguiente: "Caso del Ing. Julio C. Huancamorán"

- Los cursos de la resolución ^{444-2012-CO-UNDA} 024, no pertenecen al departamento, en decir es una plaza de ingeniero de sistemas
- Que se debe llevar el proceso por el conducto regular (consulta al Departamento Académico)
- La especialidad con la que ingresa a la docencia universitaria, es de Lic. en Estadística, como se evidencia en la resolución 258-2012-LONDFU.

Richard Flores

A.N.P.

Paula Zamora R.

Flor Palomino B.

Edwin Ramos V.

E. Roque T.

Enrique Condori T.

U. Quirope B.

Roberto Quirope





Acta de Reunión Docentes Ordinarios 12/agosto/ 2015

En la sala de docentes de la EPIS sede Zotalal a horas 10:20 am del día Miércoles 12 de agosto de 2015, se da inicio la Distribución de carga académica 2015-II para docente nombrado:

Se da la carga académica de acuerdo a la categoría y antigüedad; de la siguiente manera:

- Dr. Angel F. Navarro Raymundo elige los cursos siguientes; con un total de 10 horas.
 - Ingeniería de Software I; 6 horas; 7^{mo} semestre
 - Tesis II; 4 horas; 10^{mo} semestre
- Ing. Juan José Ore Cerrón con los siguientes cursos; con un total de 9 horas.
 - Tesis I; 3 horas; 9^{no} semestre
 - Programación I; 6 horas;
- Ing. Richard A. Flores Condeni con los siguientes cursos; con un total de 16 horas.
 - Ingeniería de la Información; 5 horas; 8^{vo} semestre
 - Simulación de Sistemas; 4 horas; 7^{mo} semestre
 - Teoría y Diseño de Base Datos I; 7 horas; 5^{to} semestre
- Ing. Herwin Alayn Huillisen Baca con los siguientes cursos; con un total 15 horas.
 - Sistema Operativos II; 5 horas; 4^{to} semestre
 - Sistemas de Comunicaciones I; 5 horas; 9^{no} semestre
 - Sistemas de Comunicaciones II; 5 horas; 10^{mo} semestre



- Ing. Enrique Edgardo Cándor Tinoco con los siguientes cursos; con un total de 15 horas
 - programación II; 6 horas; 5^{to} semestre
 - Administración de Recursos Informáticos; 5 horas; 9^{no} semestre
 - Computación Gráfica; 4 horas; 7^{mo} semestre

- Ing. Flor de Luz Palomino Valdivia con los siguientes cursos; con un total de 16 horas
 - Dibujo para Ingeniería; 4 horas; 2^{do} semestre de del Nuevo Plan Curricular.
 - Computadores y Sistemas Operativos I; 4 horas; 3^{er} semestre del Nuevo Plan Curricular.
 - Redes I; 4 horas; 5^{to} semestre
 - Redes II; 4 horas; 6^{to} semestre

- Ing. Edwin Roque Tizo con los siguientes cursos; con un total de 11 horas
 - Teoría y Diseño de Base de Datos II; 5 horas; 6^{to} semestre
 - Sistemas Digitales; 6 horas; 4^{to} semestre

- Ing. Edwin Alcides Maquera Flores con los siguientes cursos; con un total de 11 horas
 - Investigación de Operaciones II; 4 horas; 7^{mo} semestre
 - Formulación y Evaluación de Proyectos; 5h; 8^{vo} semestre
 - Multimedia; 2 horas; 6^{to} semestre

El Ing. Enrique Edgardo Cándor Tinoco hace un cambio de curso: el curso de Computación Gráfica de 4 horas del 7^{mo} semestre por Sistema de Información Gerencial de 5 horas



del 8^{vo} semestre, sumando un total de 16 horas.

- Ing. Julio César Huanca Marín con los siguientes cursos; con un total de 16 horas.

- Introducción a la Informática y Computación; 6 horas, 1^{er} semestre del Nuevo Plan
- Metodología de la programación; 6 horas; 2^{do} semestre del Nuevo Plan
- Comercio Electrónico; 4 horas; 8^{vo} semestre

Se concluye la reunión a 11:23 am satisfactoriamente.

Ing. J. A. C.

Enrique C.T.

Ing. Richard Flores

Edwin Roque T.

ENRIQUE MASOERA

DEPARTAMENTO ACADÉMICO DE INGENIERÍA Y TECNOLOGÍA INFORMÁTICA
DISTRIBUCION CARGA LECTIVA DOCENTES NOMBRADOS SEMESTRE ACADÉMICO 2015 - I

N°	CÓDIGO ASIG	CARGA LECTIVA	CONDICIÓN	CATEGORÍA	MODALIDAD	HORAS LECTIVAS			PROFESOR/CARGO	
						HT	HP	TH		
1	TRABAJO LECTIVO			ORDINARIO	PRINCIPAL	DEDICACIÓN EXCLUSIVA	HT	HP	TH	Mg. ING. CECILIA EDITH GARCIA RIVAS PLATA
	IIAA72	PLANEAMIENTO INFORMÁTICO					3	2	5	
	IIAA84	FORMULACIÓN Y EVALUACIÓN DE PROYECTOS INFORMÁTICOS					3	2	5	
	IIAA95	TESIS I					3	0	3	
	IIAA5	TESIS II					2	2	4	
						TOTAL HORAS			17	
2	TRABAJO LECTIVO			ORDINARIO	ASOCIADO	TIEMPO COMPLETO	HT	HP	TH	Mg. ING. ANGEL FERNANDO NAVARRO RAYMUNDO
	IIAA71	INGENIERÍA DE SOFTWARE I					2	4	6	
	IIAA55	INGENIERÍA ADMINISTRATIVA					2	2	4	
						TOTAL HORAS			10	
3	TRABAJO LECTIVO			ORDINARIO	ASOCIADO	DEDICACIÓN EXCLUSIVA	HT	HP	TH	ING. EDWIN OCTAVIO RAMOS VELASQUEZ
	IIAA34	ALGORITMOS Y ESTRUCTURA DE DATOS I					2	4	6	
	IIAA61	TEORÍA Y DISEÑO DE BASE DE DATOS II					3	2	5	
						TOTAL HORAS			11	
4	TRABAJO LECTIVO			ORDINARIO	ASOCIADO	DEDICACIÓN EXCLUSIVA	HT	HP	TH	ING. JUAN JOSE ORE CERRON
	IIAA45	PROGRAMACIÓN I					2	4	6	
	IIAA78	INGENIERIA EMPRESARIAL					2	2	4	
	IIAAA1	INTELIGENCIA ARTIFICIAL					3	2	5	
5	TRABAJO LECTIVO			ORDINARIO	ASOCIADO	TIEMPO COMPLETO	HT	HP	TH	ING. RICHARD ARTEMIO FLORES CONDORI
	IIAA51	TEORÍA Y DISEÑO DE BASE DE DATOS I					3	4	7	
	IIAA74	SIMULACIÓN DE SISTEMAS					2	2	4	
	IIAA82	INGENIERÍA DE LA INFORMACIÓN					3	2	5	
						TOTAL HORAS			16	
6	TRABAJO LECTIVO			ORDINARIO	ASOCIADO	TIEMPO COMPLETO	HT	HP	TH	ING. HERMAN ALAYN HUILLEEN BACA
	IIAA56	REDES I					2	2	4	
	IAI18	COMPUTACIÓN II (EPIA)					2	4	6	
						TOTAL HORAS			15	
7	TRABAJO LECTIVO			ORDINARIO	ASOCIADO	TIEMPO COMPLETO	HT	HP	TH	ING. ENRIQUE EDUARDO CONDOR TINOCO
	IIAA52	PROGRAMACIÓN II					2	4	6	
	IIAA76	COMPUTACIÓN GRÁFICA					0	4	4	
	IIAA92	ADMINISTRACIÓN DE RECURSOS INFORMÁTICOS					3	2	5	
						TOTAL HORAS			15	
8	TRABAJO LECTIVO			ORDINARIO	AUXILIAR	DEDICACIÓN EXCLUSIVA	HT	HP	TH	ING. FLOR DE LUZ PALOMINO VALDIVIA
	IIB13	INTRODUCCIÓN A LA INFORMÁTICA Y COMPUTACIÓN (Nuevo Plan Curricular)					2	4	6	
	IIB21	DIBUJO PARA INGENIERIA (Nuevo Plan Curricular)					2	2	4	
	IIB23	METODOLOGIA DE LA PROGRAMACION (Nuevo Plan Curricular)					2	4	6	
						TOTAL HORAS			16	
9	TRABAJO LECTIVO			ORDINARIO	AUXILIAR	DEDICACIÓN EXCLUSIVA	HT	HP	TH	M. Sc. ING. EDWIN ROQUE TITO
	IIAA35	SISTEMAS OPERATIVOS I					2	2	4	
	IIAA54	TEORIA GENERAL DE SISTEMAS					3	0	3	
	IIAA65	INVESTIGACION DE OPERACIONES I					2	2	4	
	IIAA88	COMERCIO ELECTRÓNICO					2	2	4	
						TOTAL HORAS			15	
10	TRABAJO LECTIVO			ORDINARIO	AUXILIAR	TIEMPO COMPLETO	HT	HP	TH	ING. EDWING ALCIDES MAQUERA FLORES
	IIAA68	MULTIMEDIA					0	2	2	
	IIAA75	INVESTIGACIÓN DE OPERACIONES II					2	2	4	
	IIAA83	SISTEMAS DE INFORMACION GERENCIAL					3	2	5	
						TOTAL HORAS			11	
11	TRABAJO LECTIVO			ORDINARIO	AUXILIAR	TIEMPO COMPLETO	HT	HP	TH	ING. ROBERTO QUISPE QUISPE
	IIAA53	ARQUITECTURA DE PROCESADORES I					3	2	5	
	IIAA62	ARQUITECTURA DE PROCESADORES II					3	2	5	
	IIAA44	SISTEMAS DIGITALES					2	4	6	
						TOTAL HORAS			16	
12	TRABAJO LECTIVO			ORDINARIO	AUXILIAR	TIEMPO COMPLETO	HT	HP	TH	ING. JULIO CESAR HUANCA MARÍN
	IIAA42	ALGORITMOS Y ESTRUCTURA DE DATOS II					2	4	6	
	IIAA66	PROGRAMACIÓN III					3	4	7	
	IIAA6	PROGRAMACIÓN AVANZADA					2	2	4	
						TOTAL HORAS			17	





Universidad Nacional José María Arguedas
Identidad y Excelencia para el Trabajo Productivo y el
Desarrollo



UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS
SECRETARÍA GENERAL
TRÁMITE ADMINISTRATIVO
RECIBIDO
N° REG: 3589 FEBRU 21 12006
HORA: 2:50
FOLIO: 04 FIRMA: [Firma]

INFORME N°001-2016-P.A.C/AND

AL : Mg. Cecilia Edith Garcia Rivas Plata
Directora de la EPIS

DEL : Bach. Percy Alhuay Carrasco
Tesista del proyecto de generación de horario para EPIS.

ASUNTO : Hacerle conocer los horarios generados del I a X ciclo de
la EPIS como prueba.

FECHA : San Jerónimo, 12 de diciembre del 2016

Tengo el agrado de dirigirme a su digna persona para poder saludarle muy cordialmente. Mediante el presente documento hacerle conocer a su despacho de la EPIS, como tesista se planteó **desarrollar sistema de generación de horarios con algoritmo genético para la EPIS**, para obtener mi título, lo cual fue aprobado bajo resolución, donde actualmente me genera resultados adjunto documento.

- Horario generado de I a X ciclo.

Es cuanto informo a usted, para su conocimiento y fines.

Sin otro en particular, aprovecho la ocasión para referirle mi especial consideración y entendimiento.

Atentamente,


Bach. Percy Alhuay Carrasco

DNI: 46897622



Universidad Nacional José María Arguedas
 Identidad y Excelencia para el Trabajo Productivo y el
 Desarrollo



CARGO

INFORME N°001-2016-P.A.C/AND

AL : M.sc. Edwin Roque Tito Director de Registros Académicos
 DIRECTOR DE REGISTROS ACADÉMICOS

DEL : Bach. Percy Alhuay Carrasco
 Tesista del proyecto de generación de horario para EPIS.

ASUNTO : Hacerle conocer los horarios generados del I a X ciclo de
 la EPIS como prueba.

FECHA : San Jerónimo, 12 de diciembre del 2016

Tengo el agrado de dirigirme a su digna persona para poder saludarle muy cordialmente. Mediante el presente documento hacerle conocer a su despacho de la EPIS, como tesista se planteó **desarrollar sistema de generación de horarios con algoritmo genético para la EPIS**, para obtener mi título, lo cual fue aprobado bajo resolución, donde actualmente me genera resultados adjunto documento.

- Horario generado de I a X ciclo.

Es cuanto informo a usted, para su conocimiento y fines.

Sin otro en particular, aprovecho la ocasión para referirle mi especial consideración y entendimiento.

Atentamente,

Bach. Percy Alhuay Carrasco

DNI: 46897622

