UNIVERSIDAD DE ORIENTE

NÚCLEO DE MONAGAS

ESCUELA DE INGENIERÍA Y CIENCIAS APLICADAS

DEPARTAMENTO DE INGENIERÍA DE SISTEMAS

CIENCIAS DE LA COMPUTACIÓN

DESARROLLO DE UNA APLICACION WEB PARA LA GESTIÓN DINÁMICA DEL TIEMPO DE ESTUDIO EN LA CARRERA INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD DE ORIENTE – NÚCLEO MONAGAS

(Caso de estudio Taller de Programación Orientada a Objetos 0722111)

Monografía de Investigación, en Modalidad Cursos Especiales de Grado, presentado como requisito parcial para optar al título de Ingeniero de Sistemas

Autores:
Br. Mata Héctor
C.I: 25.273.405
Br. Rodríguez Santiago
C.I: 26.531.663

Asesor Académico:
Prof. (Ing.) Guevara Rommel

Maturín, Octubre del 2020

UNIVERSIDAD DE ORIENTE
NÚCLEO DE MONAGAS
ESCUELA DE INGENIERÍA Y CIENCIAS APLICADAS
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS
CIENCIAS DE LA COMPUTACIÓN

DESARROLLO DE UNA APLICACION WEB PARA LA GESTIÓN DINÁMICA DEL TIEMPO DE ESTUDIO EN LA CARRERA INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD DE ORIENTE – NÚCLEO MONAGAS

(Caso de estudio Taller de Programación Orientada a Objetos 0722111)

APROBADO POR:

Ing. Guevara Rommel
Tutor Académico

Ing. Reinoza Henry
Jurado Principal

Ing. Rodríguez Yeisland
Jurado Principal

Maturín, Octubre del 2020

UNIVERSIDAD DE ORIENTE
NÚCLEO DE MONAGAS

ESCUELA DE INGENIERÍA Y CIENCIAS APLICADAS
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS
SUB-COMISIÓN DE TRABAJO DE GRADO

ACTA DE EVALUACIÓN DEL TRABAJO DE GRADO

CTG-EICA-IS-2017

MODALIDAD: CURSOS ESPECIALES DE GRADO

ACTA N° 00000496-00002-03-2020

En Maturín, siendo las 8:00 am. del día 20 de octubre del 2020 reunidos en la Sala "A-5 Postgrado", Campus: Juanico del Núcleo de Monagas de la Universidad de Oriente, los miembros del jurado profesores: (Ing.) ROMMEL GUEVARA (Asesor Académico), (Ing.) YEISLAND RODRÍGUEZ (Jurado), (Ing.) HENRY REINOZA (Jurado). A fin de cumplir con el requisito parcial exigido por el Reglamento de Trabajo de Grado vigente para obtener el Título de **Ingeniero de Sistemas,** se procedió a la presentación del Trabajo de Grado, titulado: DESARROLLO DE UNA APLICACION WEB PARA LA GESTIÓN DINÁMICA DEL TIEMPO DE ESTUDIO EN LA CARRERA INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD DE ORIENTE – NÚCLEO MONAGAS (Caso de estudio Taller de Programación Orientada a Objetos 0722111). Por el Bachiller: HÉCTOR RAFAEL MATA BERMÚDEZ, C.I. 25.273.405. El jurado, luego de la discusión del mismo acuerdan calificarlo como: **APROBADO**

Br. HÉCTOR MATA
C.I.: 25.273.405

Prof. (Ing.) ROMMEL GUEVARA
C.I.: 10.300.053
Asesor Académico

Profa. (Ing.) YEISLAND RODRÍGUEZ
C.I.: 16.199.486
Jurado

Prof. (Ing.) HENRY REINOZA
C.I.: 8.030.340
Jurado

Profa. (Ing.) FRANCY TONONI
C.I.: 8.277.843
Sub-Comisión de Trabajo de Grado

Profa. (Lcda.) ALBA ORTIZ
C.I.: 14.009.373
Jefe de Departamento

Según establecido en resolución de Consejo Universitario N° 034/2009 de fecha 11/06/2009 y Artículo 13 Literal J del Reglamento de Trabajo de Grado de la Universidad de Oriente. NOTA: Para que esta acta tenga validez debe ser asentada en la hoja N°- 17 del 01° libro de Actas de Trabajos de Grado del Departamento de Ingeniería de Sistemas, EICA de la Universidad de Oriente y estar debidamente firmada por el (los) asesor (es) y miembros del jurado.

DEL PUEBLO VENIMOS / HACIA EL PUEBLO VAMOS

UNIVERSIDAD DE ORIENTE
NÚCLEO DE MONAGAS

ESCUELA DE INGENIERÍA Y CIENCIAS APLICADAS
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS
SUB-COMISIÓN DE TRABAJO DE GRADO

ACTA DE EVALUACIÓN DEL TRABAJO DE GRADO

CTG-EICA-IS-2016

MODALIDAD: CURSOS ESPECIALES DE GRADO

ACTA N° 00000495-00001-03-2020

En Maturín, siendo las 8:00 am. del día 20 de octubre del 2020 reunidos en la Sala "A-5 Postgrado", Campus: Juanico del Núcleo de Monagas de la Universidad de Oriente, los miembros del jurado profesores: (Ing.) ROMMEL GUEVARA (Asesor Académico), (Ing.) YEISLAND RODRÍGUEZ (Jurado), (Ing.) HENRY REINOZA (Jurado). A fin de cumplir con el requisito parcial exigido por el Reglamento de Trabajo de Grado vigente para obtener el Título de **Ingeniero de Sistemas,** se procedió a la presentación del Trabajo de Grado, titulado: DESARROLLO DE UNA APLICACION WEB PARA LA GESTIÓN DINÁMICA DEL TIEMPO DE ESTUDIO EN LA CARRERA INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD DE ORIENTE – NÚCLEO MONAGAS (Caso de estudio Taller de Programación Orientada a Objetos 0722111). Por el Bachiller: SANTIAGO RAFAEL RODRIGUEZ LISCANO, C.I. 26.531.663. El jurado, luego de la discusión del mismo acuerdan calificarlo como: **APROBADO**

Br. SANTIAGO RODRIGUEZ
C.I.:26.531.663

Prof. (Ing.) ROMMEL GUEVARA
C.I.: 16.386.953
Asesor Académico

Profa. (Ing.) YEISLAND RODRIGUEZ
C.I.: 16.199.486
Jurado

Prof. (Ing.) HENRY REINOZA
C.I.: 8.030.340
Jurado

Profa. (Ing.) FRANCY TONONI
C.I.: 8.277.84
Sub-Comisión de Trabajo de Grado

Profa. (Lcda.) ALBA ORTIZ
C.I.: 14.009.373
Jefe de Departamento

Según establecido en resolución de Consejo Universitario N° 034/2009 de fecha 11/06/2009 y Artículo 13 Literal J del Reglamento de Trabajo de Grado de la Universidad de Oriente. "NOTA: Para que esta acta tenga validez debe ser asentada en la hoja N°- 16 del 01° libro de Actas de Trabajos de Grado del Departamento de Ingeniería de Sistemas, EICA de la Universidad de Oriente y estar debidamente firmada por el (los) asesor (es) y miembros del jurado.

DEL PUEBLO VENIMOS / HACIA EL PUEBLO VAMOS

RESOLUCIÓN

De acuerdo al artículo 41 del Reglamento de Trabajo de Grado: "Los Trabajos de Grado son de la exclusiva propiedad de la Universidad de Oriente, y sólo podrán ser utilizados a otros fines con el consentimiento del Consejo de Núcleo respectivo, quién deberá participarlo previamente al Consejo Universitario, para su autorización".

DEDICATORIA

Este trabajo está dedicado a mis padres **Héctor Mata** y **Tamaiba Bermúdez** ya que son los pilares fundamentales de mi formación personal y profesional.

A mi hermana **Luemyadig Mata** quien ha sido un apoyo importante en mi vida.

A mis abuelos **Rafael Mata**, **Adolfina Brito**, **Agustina Velásquez**, aunque no estén físicamente sé que han guiado cada uno de mis pasos.

Y, por último, pero no menos importante a mi mascota **Holly** el miembro más importante de mi familia ❤

Héctor Mata

DEDICATORIA

Dedico el cumplimiento de este logro a **Dios**, por guiarme durante el camino y darme fuerzas para seguir adelante a pesar de las dificultades.

A mis **padres**, Carmen Alida Liscano y Rafael Abelardo Rodríguez por ser ejemplo de honestidad, esfuerzo y amor, por enseñarme todo lo que soy y siempre tomar en cuenta mis estudios y metas. Por apoyarme en todas mis ideas y ser unos padres incondicionales. Todo lo que he aprendido y seguiré aprendiendo es gracias a ustedes que me enseñaron lo que verdaderamente es importante en la vida y lo que vale la pena luchar. Los amo.

A mi **primo hermano** Rafael Velásquez, por estar ahí durante toda mi etapa de estudios superiores y por ayudarme en cuanto necesitara, por siempre ser un gran amigo y hermano mayor, por todas las vivencias que tuvimos durante tu residencia en la casa y por demostrar que siempre con el suficiente esfuerzo y dedicación se puede mejorar aún más.

A mi **familia**, por ser la mejor familia que me pudo tocar, quienes me han brindado ejemplos de superación, profesionalismo, amistad y cooperación. Por la unión que nos caracteriza, los bonitos recuerdos que tenemos juntos y por todas las demás experiencias que nos falta por vivir. Siempre estaré agradecido por lo que me han enseñado.

A todos aquellos **jóvenes venezolanos** que a pesar de las dificultades que nos ha tocado vivir seguimos buscando cumplir nuestras metas con esfuerzo, luchando contra las limitaciones que se nos imponen y realizando los sacrificios para superarlas. Para todos ellos mi admiración y mis mejores deseos.

Santiago Rodríguez

AGRADECIMIENTOS

A mis padres **Tamaiba Bermúdez** y **Héctor Mata**, quienes siempre han confiado en mí y día a día me han brindado su sabiduría y apoyo incondicional, con palabras, consejos, regaños y su propio ejemplo a seguir, pero sobretodo con muchísimo amor. Y a mi hermana **Luemyadig Mata**, quien me ha acompañado a lo largo de mis trepidantes aventuras y me ha ayudado mucho en este recorrido académico y profesional.

A mis tíos **Carmen Aurolina Bermúdez** y **Virgilio Brito** los cuales siempre han estado involucrados estrechamente en mi vida y sobretodo en mi travesía universitaria prestando apoyo en todos los aspectos.

A mis tíos **Juan Carlos González** y **Virginia Brito**, **Ludmilla Bermúdez** y **Jesús Abreu** por su apoyo y conocimiento que probaron ser invaluables en este trayecto que fue la realización de mi trabajo de grado. Y al resto de mi familia que de alguna u otra manera influenciaron mi desarrollo personal y académico.

A mi compañero **Santiago Rodríguez** por nunca desfallecer y mantenerse constante durante toda esta travesía.

A mi amigo **Omar Díaz** por estar pendiente incluso desde la Argentina, y compartir mi entusiasmo por Terraria.

A mi tutor académico **Rommel Guevara** por ayudarme enormemente en el desempeño de este proyecto y siempre creer en mi potencial. Y al profesor **Cesar Rivero** por ofrecerme su amistad desde el día 1 que ingrese a la universidad.

A todos esos compañeros de estudio que, a lo largo de la carrera, fueron de gran relevancia, siempre los llevare en mi corazón.

Y no puedo dejar a un lado a mi fiel y amada, **Holly María**, que, aunque es tan solo un perrito, siempre está a mi lado, llenado mi vida de alegrías y amor incondicional, ocupando uno de los lugares más importantes de mi corazón.

Hector Mata

AGRADECIMIENTOS

A mi madre **Carmen Alida Liscano** por siempre estar para mí y ayudarme en lo necesario, por escucharme cuando estaba agobiado y por orientarme constantemente durante mi vida. Te amo.

A mi padre **Rafael Abelardo Rodríguez** por siempre buscar lo mejor para nosotros y trabajar incansablemente en pro de que nunca nos faltase nada, por todo el apoyo que recibo de ti y por siempre ser un vivo ejemplo de superación. Te amo.

A la **Universidad de Oriente** "la casa más alta", por ser mi segunda casa durante todo este tiempo de estudios y permitirme formar parte, por enseñarme a analizar y a buscar constantemente información y por formar lo que soy ahora.

A mis compañeros de estudio y amigos, a **Félix Martínez** y **Dora Goncalves** por ser mis amigos durante todos los años de universidad, a **Brayan Vegas, Richard Solano, Carlos Oswaldo, Alberto Núñez** por todas las charlas y discusiones diarias fuera de aulas de clase. Agradecido por esos buenos momentos.

A mis amigos y hermanos **José Ruiz, Diego Olmedo y Manuel Haiek** que, a pesar de estar lejos actualmente, espero volver a encontrarnos y podamos compartir como cuando éramos más chamos.

A los **profesores** que a pesar de las circunstancias siguen dando lo mejor de sí y permiten que muchos jóvenes crezcan como ciudadanos y profesionales, mis más sinceros agradecimientos.

A los **profesores de áreas**, por formar parte junto a nosotros de esta etapa extra de formación y debate, agradecido por todas las experiencias y el conocimiento compartido. Particularmente con el profesor **Rommel Guevara**, por habernos ayudado tanto durante toda esta etapa y formar parte del objetivo alcanzado.

Y no menos importante a mi compañero de investigación **Héctor Mata** por haber compartido juntos todos los momentos de estrés, de alegría y por ser un gran amigo.

Santiago Rodríguez

INDICE GENERAL

RESOLUCIÓN ..... **v**
DEDICATORIA ..... **vi**
AGRADECIMIENTOS ..... **viii**
INDICE GENERAL ..... **x**
INDICE DE FIGURAS ..... **xiii**
INDICE DE GRAFICOS ..... **xiv**
INDICE DE TABLAS ..... **xv**
RESUMEN ..... **xvii**
INTRODUCCION **1**
CAPITULO I **3**
EL PROBLEMA Y SUS GENERALIDADES **3**
1.1 PLANTEAMIENTO DEL PROBLEMA 3
1.2 OBJETIVOS DE LA INVESTIGACIÓN 6
1.2.1 Objetivo general 6
1.2.2 Objetivos específicos 6
1.3 JUSTIFICACIÓN DE LA INVESTIGACIÓN 7
1.4 DELIMITACIÓN DE LA INVESTIGACIÓN 8
CAPITULO II **9**
MARCO TEORICO **9**
2.1 ANTECEDENTES DE LA INVESTIGACIÓN 9
2.2 BASES TEÓRICAS 11
2.2.1 Procrastinación 11
2.2.2 Sistema 12
2.2.3 Tecnologías de la Información y Comunicación (TIC) 12
2.2.4 Aplicación Web 12
2.2.5 Desarrollo web Full-Stack 13
2.2.6 Lenguajes de Marcado 13
2.2.7 HTML 14
2.2.8 Cascading Style Sheets CSS 15
2.2.9 Bootstrap 15
2.2.10 Lenguajes De Programación 16
2.2.11 JavaScript 16
2.2.12 jQuery 17
2.2.13 Node.js 17
2.2.14 Express 18
2.2.15 Base de Datos 19
2.2.16 NoSQL 19
2.2.17 MongoDB 20
2.2.18 Visual Studio Code 20

2.2.19 Git 21
2.2.20 GitHub 22
2.2.21 UML 22
2.2.22 Metodología 25
2.2.23 Metodología XP 25
2.2.23.1 Características de la metodología XP 25
2.2.23.2 Fases de la programación extrema 26
2.2.23.2.1 Planeación 26
2.2.23.2.2 Diseño: 27
2.2.23.2.3 Codificación 27
2.2.23.2.4 Pruebas 28
2.3 BASES LEGALES 28
2.3.1 Constitución de la República Bolivariana de Venezuela (1999) 29
2.3.2 Ley Orgánica de Ciencia, Tecnología e Innovación (2010) 30
2.4 DEFINICION DE TÉRMINOS BÁSICOS 30
CAPITULO III 33
DESARROLLO DE LOS OBJETIVOS 33
3.1 TIPO Y DISEÑO DE LA INVESTIGACION 33
3.2 NIVEL DE LA INVESTIGACION 34
3.3 DISEÑO METODOLOGICO CON TECNICAS E INSTRUMENTOS 34
3.3.1 Revisión documental 34
3.3.2 Encuestas estructuradas 35
3.3.3 Encuestas impersonales 35
3.4 POBLACIÓN Y MUESTRA 35
3.4.1 Población 35
3.4.2 Muestra 36
3.5 TECNICAS DE ANALISIS DE DATOS 37
3.6 DISEÑO OPERATIVO 38
3.6.1 Fase I: Planeación 38
3.6.2 Fase II: Diseño 39
3.6.3 Fase III: Desarrollo 39
3.7 PRIMERA FASE 43
3.7.1 Análisis de la situación actual 43
3.7.1.1 Descripción de la situación actual 43
3.7.1.2 Identificación de los agentes presentes en la situación actual 43
3.7.1.3 Diagrama de actividades del sistema actual 45
3.7.1.4 Análisis de los datos recolectados a través de encuestas 46
3.7.1.5 Identificación de focos problemáticos 51
3.7.2 Análisis de metas y submetas situación actual 54
3.7.2.1 Razonamiento Metas-Submetas Escenario Actual 54
3.7.2.2 Método de reducción de metas escenario actual 54
3.7.3 Análisis conjunto del escenario propuesto 55
3.7.3.1 Identificación de los agentes presentes en el escenario propuesto 55

3.7.3.2 Razonamiento metas-submetas escenario propuesto 57
3.7.3.3 Método de reducción de metas escenario propuesto 57
3.7.3.4 Método de razonamiento hacia atrás y hacia adelante del escenario propuesto 58
3.7.3.4.1 Razonamiento hacia atrás 58
3.7.3.4.2 Aplicando el razonamiento hacia atrás 59
3.7.3.4.3 Aplicando razonamiento hacia adelante 59
3.7.3.5 Identificación de reglas de condición – acción 60
3.7.3.5.1 Estudiantes 60
3.7.3.5.2 Base de datos 60
3.7.3.5.3 Aplicación WEB 61
3.7.4 Requerimientos de la aplicación 61
3.7.4.1 Historias de Usuario 61
3.7.4.2 Tareas de ingeniería 65
3.7.4.2.1 Tareas de ingeniería Primera Iteración 66
3.7.4.2.2 Tareas de ingeniería Segunda Iteración 70
3.7.4.2.3 Tareas de ingeniería Tercera Iteración 75
3.7.4.3 Plan de Entregas 79
3.7.4.4 Requerimientos funcionales 81
3.7.4.5 Requerimientos no funcionales 81
3.7.4.6 Requerimientos de software 82
3.7.4.7 Requerimientos de hardware 82
3.8 SEGUNDA FASE 83
3.8.1 Diseño base de la aplicación propuesta 83
3.8.1.1 Arquitectura del sistema 83
3.8.1.2 Interconexiones de los agentes presentes en el sistema 84
3.8.1.3 Explicación de los accesos de usuarios 84
3.8.2 Casos de uso del sistema 85
3.8.3 Diagramas de clases 87
3.8.4 Diagramas de secuencia 88
3.8.5 Diagramas de actividades 90
3.8.6 Determinación del modelo matemático de la aplicación propuesta 92
3.8.7 Definición autómata de la aplicación propuesta 98
3.9 TERCERA FASE 100
3.9.1 Codificación del sistema diseñado 100
3.9.2 Representación de la aplicación propuesta, diseño de interfaz 101
3.9.3 Casos de prueba 106
CAPITULO IV 110
CONCLUSIONES Y RECOMENDACIONES 110
4.1 CONCLUSIONES 110
4.2 RECOMENDACIONES 111
REFERENCIAS BIBLIOGRAFICAS 113
HOJAS METADATOS 117

INDICE DE FIGURAS

Figura 1. Diagrama modelo de un equipo de gestión de proyecto 23
Figura 2. Diagrama de actividades - sistema actual 45
Figura 3. Interconexión de los focos problemáticos 52
Figura 4. Diagrama de Causa y Efecto (Ishikawa) 53
Figura 5. Método de reducción de metas escenario actual 55
Figura 6. Método de reducción de metas escenario propuesto 58
Figura 7. Arquitectura del sistema 83
Figura 8. Interconexiones de agentes presentes en el sistema 84
Figura 9. Casos de uso general del sistema 85
Figura 10. Caso de uso de la generación del horario 86
Figura 11. Diagrama de caso de uso – visualizar dashboard 86
Figura 12. Diagrama de caso de uso dashboard – modificar, editar, reiniciar e imprimir 87
Figura 13. Diagrama general de clases 88
Figura 14. Diagrama de secuencia Inicio de Sesión 89
Figura 15. Diagrama de secuencia Registro de Usuario 89
Figura 16. Diagrama de secuencia Generador de Horario 90
Figura 17. Diagrama de actividades general del sistema 91
Figura 18. Diagrama de transición del autómata propuesto 98
Figura 19. Login del sistema 101
Figura 20. Registro del sistema 102
Figura 21. Selección de materias 103
Figura 22. Selección de horas disponibles 104
Figura 23. Horario Generado 104
Figura 24. Horario generado por el sistema – perfil de usuario 105

INDICE DE GRAFICOS

Grafico 1. Respuestas graficadas de la encuesta – Primera pregunta 47
Grafico 2. Respuestas graficadas de la encuesta – Segunda pregunta 48
Grafico 3. Respuestas graficadas de la encuesta – Tercera pregunta..... 49
Grafico 4. Respuestas graficadas de la encuesta – Cuarta pregunta 50

INDICE DE TABLAS

Tabla 1. Diseño cuadro Operativo 40
Tabla 2. Agentes del proceso actual 44
Tabla 3. Razonamiento Metas y Submetas situación actual 54
Tabla 4. Agentes del sistema 56
Tabla 5. Metas Submetas del escenario propuesto 57
Tabla 6. Historia de Usuario 1 62
Tabla 7. Historia de Usuario 2 62
Tabla 8. Historia de Usuario 3 63
Tabla 9. Historia de Usuario 4 63
Tabla 10. Historia de Usuario 5 64
Tabla 11. Historia de Usuario 6 64
Tabla 12. Historia de Usuario 7 65
Tabla 13. Tarea de ingeniería 1 66
Tabla 14. Tarea de ingeniería 2 66
Tabla 15. Tarea de ingeniería 3 67
Tabla 16. Tarea de ingeniería 4 67
Tabla 17. Tarea de ingeniería 5 67
Tabla 18. Tarea de ingeniería 6 68
Tabla 19. Tarea de ingeniería 7 68
Tabla 20. Tarea de ingeniería 8 68
Tabla 21. Conjunto de tareas de ingeniería – primera iteración 69
Tabla 22. Tarea de ingeniería 9 70
Tabla 23. Tarea de ingeniería 10 70
Tabla 24. Tarea de ingeniería 11 70
Tabla 25. Tarea de ingeniería 12 71
Tabla 26. Tarea de ingeniería 13 71
Tabla 27. Tarea de ingeniería 14 71
Tabla 28. Tarea de ingeniería 15 72
Tabla 29. Tarea de ingeniería 16 72
Tabla 30. Tarea de ingeniería 17 72
Tabla 31. Tarea de ingeniería 18 73
Tabla 32. Tarea de ingeniería 19 73
Tabla 33. Tarea de ingeniería 20 73
Tabla 34. Conjunto de tareas de ingeniería – segunda iteración 74
Tabla 35. Tarea de ingeniería 21 75
Tabla 36. Tarea de ingeniería 22 75
Tabla 37. Tarea de ingeniería 23 75
Tabla 38. Tarea de ingeniería 24 76
Tabla 39. Tarea de ingeniería 25 76

Tabla 40. Tarea de ingeniería 26 76
Tabla 41. Tarea de ingeniería 27 77
Tabla 42. Tarea de ingeniería 28 77
Tabla 43. Conjunto de tareas de ingeniería – tercera iteración 78
Tabla 44. Plan de entrega 1ra iteración. 80
Tabla 45. Plan de entrega 2da iteración. 80
Tabla 46. Plan de entrega 3ra iteración 80
Tabla 47. Pesos asignados variables a evaluar. 94
Tabla 48. Valores posibles - variables a evaluar. 94
Tabla 49. Ejemplo modelo matemático 1ra asignatura 95
Tabla 50. Ejemplo modelo matemático 2da asignatura 95
Tabla 51. Leyenda de estados y entradas del autómata del sistema propuesto 99
Tabla 52. Caso de prueba de aceptación 1 106
Tabla 53. Caso de prueba de aceptación 2 107
Tabla 54. Caso de prueba de aceptación 3 107
Tabla 55. Caso de prueba de aceptación 4 108
Tabla 56. Caso de prueba de aceptación 5 108
Tabla 57. Caso de prueba de aceptación 6 109
Tabla 58. Caso de prueba de aceptación 7 109

UNIVERSIDAD DE ORIENTE
NÚCLEO DE MONAGAS
ESCUELA DE INGENIERÍA Y CIENCIAS APLICADAS
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS
CIENCIAS DE LA COMPUTACIÓN

DESARROLLO DE UNA APLICACION WEB PARA LA GESTIÓN DINÁMICA DEL TIEMPO DE ESTUDIO EN LA CARRERA INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD DE ORIENTE – NÚCLEO MONAGAS

Autores:
Br. Mata Héctor
C.I: 25.273.405
Br. Rodríguez Santiago
C.I: 26.531.663

Asesor Académico:
Prof. (Ing.) Guevara Rommel
Fecha: Octubre del 2020

RESUMEN

Esta investigación tuvo como propósito optimizar el proceso de organización y distribución del tiempo de los estudiantes, el cual se realizaba de forma manual, presentando deficiencias al momento de su planificación y trayendo como consecuencia disminución en el rendimiento académico. El trabajo de investigación tecnológica, de nivel descriptivo, con apoyo de técnicas e instrumentos metodológicos, como revisión documental y encuestas estructuradas impersonales, empleando la metodología eXtreme Programming XP hace uso de técnicas y prácticas para el desarrollo de software. Utilizó un algoritmo basado en programación por metas que permitiera generar los horarios de estudio y crear una aplicación web que facilitara el uso del mismo, la implementación de ésta aumentaría el desempeño de los estudiantes.

Palabras claves: Aplicación web, procrastinación, estudiantes, horario, planificar.

INTRODUCCION

En los últimos años el uso de la tecnología en el campo académico se ha considerado una de las herramientas más importantes para el mejoramiento del rendimiento de los estudiantes. Permitiendo realizar tareas con gran velocidad y proporcionando acceso a múltiples referencias bibliográficas y diferentes fuentes de estudio. De la misma manera, el internet se ha vuelto una parte indispensable de las vidas de los ciudadanos.

Las aplicaciones y sistemas WEB permiten llevar estas herramientas a cualquier individuo que posea un dispositivo capaz de conectarse a la red, siendo el uso de dispositivos móviles o computadoras personales los más comunes. Actualmente, cada uno de éstos resulta un pilar importante para el desarrollo de las actividades diarias de la población y es evidenciable que el empleo de ellos incrementa la productividad en la realización de éstas tareas. Los estudiantes usan estas tecnologías en el ámbito académico para la realización de investigaciones y como medio para reforzar los conocimientos, sin embargo, a pesar de estos beneficios, la web dispone de muchas fuentes de distracción que pueden entorpecer la organización del tiempo libre, trayendo como consecuencias resultados negativos en el desempeño estudiantil.

En la Universidad de Oriente Núcleo de Monagas, los estudiantes de Ingeniería de Sistemas no cuentan con una herramienta para administrar un plan de estudio que les permita distribuir de forma óptima el tiempo que disponen, no obstante, la presente investigación busca desarrollar una aplicación web donde los alumnos puedan mejorar ésta administración a través de un algoritmo automatizado que distribuya horas de estudio en los tiempos libres disponibles a la semana permitiendo generar con el uso constante, hábitos de estudio que mejoren el rendimiento

académico. La investigación está estructurada en cuatro (4) capítulos conformados de la siguiente manera:

Capítulo I. El problema y sus generalidades: En este capítulo se describe la situación problema, evidenciando las carencias que se deben atender y las condiciones que se debe mejorar, se establece el objetivo general, los objetivos específicos, y se realiza la justificación de la investigación donde se explica por qué la institución se beneficiaría, así como también la delimitación del estudio.

Capítulo II. Marco Teórico: constituido por los antecedentes de la investigación los cuales construyen los cimientos de la investigación, las bases teóricas que representan el conjunto de conceptos relacionados con el enfoque de la investigación, las bases legales donde se exponen los lineamientos que la ley exige y la definición de términos básicos. En éste capítulo se presentan el conjunto de fundamentos teóricos que guiaron a la investigación.

Capítulo III. Desarrollo de los objetivos: se describe en éste capítulo el enfoque metodológico que guió el desarrollo del proyecto, además de establecer los requisitos del mismo, el análisis y presentación de los resultados de la recolección de datos. Se desarrollaron los procesos a partir de los cuales se definió la estructura de la aplicación y la codificación de la misma.

Capítulo IV. Conclusiones y recomendaciones: Comprende las conclusiones obtenidas a través de las diversas fases del desarrollo del trabajo de investigación y las recomendaciones finales que permitan servir de apoyo para investigaciones futuras en la misma área o como guía para la aplicabilidad, mantenimiento y mejora de la propuesta establecida por los investigadores.

CAPITULO I
EL PROBLEMA Y SUS GENERALIDADES

1.1 PLANTEAMIENTO DEL PROBLEMA

El acceso a la información académica nunca había estado tan disponible como en la actualidad, el internet se convirtió en una biblioteca digital donde se adquiere información relacionada a una gran diversidad de áreas del conocimiento, y la cantidad de estudios académicos junto con las innumerables referencias bibliográficas a la que un estudiante puede acceder para profundizar en su aprendizaje sigue aumentado cada día (Garcia & otros, 2019). Los avances tecnológicos en la sociedad introdujeron muchas facilidades con respecto a cómo se vivía en las décadas pasadas e igualmente crearon gran cantidad de distracciones que proporcionaron un ambiente idóneo para la procrastinación del individuo, sin diferenciar si este desempeña sus estudios en una academia o adquiere el conocimiento de forma autodidacta. No tener un buen hábito de estudio definido o una orientación de cómo abordar la adquisición de nuevo conocimiento es perjudicial para el rendimiento académico.

En la actualidad se estima que la procrastinación es uno de los problemas más comunes relacionados con el desempeño estudiantil, según un estudio realizado por el departamento de profesores de la Universidad Yonsei en Seúl, Corea, analizó 33 estudios en el área incorporando una muestra de 38.529 participantes, donde se evidenció la implicación de comportamientos relacionados con la procrastinación en la disminución del rendimiento académico (Kim & Seo, 2015). Los estudiantes con estos hábitos aplazan sus estudios con la mentalidad de realizar una actividad placentera y dejar para luego aquellas actividades que requieran de un mayor esfuerzo para su realización.

Comportamientos como el expuesto generan deficiencia en el aprendizaje y el desarrollo de competencias, el estudiante solo se enfoca en realizar y cumplir con la entrega de las asignaciones para la fecha límite establecida por su profesor, sin dedicarle el tiempo necesario a fin de asimilar el contenido. El problema es más evidente cuando ingresan a la universidad, no desarrollan un hábito de estudio durante el periodo de educación secundaria y mantienen esta forma de realizar las asignaciones que resultaba "exitosa" en la etapa anterior, pero que no les da los mismos resultados durante los estudios superiores, ocasionando como consecuencias una obtención de bajas calificaciones, reprobación de asignaturas, pérdida de tiempo, la deserción de estudio, cambios constantes de carrera, baja autoestima y desmotivación (Loyola Illescas, et al., 2018).

El internet facilitó la transferencia de información alrededor del mundo, pero a su vez, complico la capacidad del individuo de discernir entre el tiempo gastado diariamente consumiendo contenido de entretenimiento y cuánto le debe dedicar a la realización de las obligaciones personales, complicando así su organización diaria. La gestión del tiempo constituye un aspecto esencial en el trabajo académico de los estudiantes universitarios, resultando ser el factor más importante en la predicción del rendimiento, por delante de otros como la inteligencia o el total de horas dedicadas al estudio (Darren, Sinikka, Emory, Shannon Lund, & Tabitha, 2008).

En este sentido, mantener un mal hábito de estudio puede traer como consecuencia la manifestación de estrés y un bajo desempeño académico, a causa de que el estudiante no dispone de la organización necesaria para crear un plan de estudio idóneo que le permita asimilar el contenido programático de una asignatura, además, postergar este comportamiento fuera del ámbito académico traerá problemas a nivel laboral y personal debido a la ausencia de la responsabilidad.

La corrección de esta conducta durante las diversas etapas de formación académica y más aún a nivel universitario, garantizara el desarrollo de profesionales de calidad y ciudadanos responsables, capaces de generar ideas y tecnologías necesarias para el progreso del país. Para el año 2002, en Venezuela se ofrecía estudios profesionales a más de 800.000 personas en 145 instituciones de educación superior, las cuales tienen presencia física en 96 ciudades ofreciendo 370 carreras distintas a nivel de pregrado y alrededor de 200 especialidades de postgrado. (Víctor Morales, et al., 2003) Donde, según la clasificación actual realizada por el Laboratorio de Cibermetría del Consejo Superior de Investigaciones Científicas (CSIC) las 8 mejores universidades de Venezuela comprenderían a; la Universidad Central de Venezuela, la Universidad de los Andes, la Universidad Simón bolívar, la Universidad del Zulia y la Universidad de Oriente por citar algunas.

La Universidad de Oriente es un instituto de educación superior dedicado a la docencia y a la investigación en la región oriental de Venezuela, ofrece estudios en diversas especialidades en el área de ciencias y tecnología, con su población estudiantil repartida en cinco núcleos: Sucre, Anzoátegui, Bolívar, Nueva Esparta y Monagas, teniendo esta ultima un aproximado de 4.400 estudiantes inscritos para el periodo III-2019.

De acuerdo a datos aportados por el instructor de la asignatura Taller de Programación Orientada a Objetos de la carrera Ingeniería de Sistemas (UDO Núcleo de Monagas) donde se evaluó el desempeño de los estudiantes y el índice de aprobación con respecto a repitientes durante los periodos académicos comprendidos del I-2016 al III-2017, se evidenció, una tendencia donde más del 50% de los estudiantes inscritos reprueban, además, se observó que el índice de inscripción y aprobación de repitientes va disminuyendo con el paso de los periodos académicos. Hacer caso omiso a las deficiencias en los métodos de estudio de los estudiantes seguirá generando un porcentaje de resultados insatisfactorios, que traerán como

consecuencias; el retraso de la graduación de nuevos profesionales, el aumento del trabajo administrativo de los departamentos, la disminución de la calidad académica, dificultad de completar todas las unidades de los cronogramas programados en las asignaturas, y la deserción de estudiantes.

Por consiguiente, se planteó la necesidad de desarrollar una aplicación WEB para la administración dinámica del tiempo de los estudiantes en la Universidad de Oriente Núcleo de Monagas enfocados específicamente en el caso de los inscritos en la asignatura Taller de Programación Orientada a Objetos de la carrera Ingeniería de Sistemas. Con el objetivo de que los mismos puedan llevar un control de las horas dedicadas al estudio diario mediante el desarrollo de un horario dinámico por parte de la aplicación, tomando en cuenta las asignaturas inscritas en el periodo académico, las exigencias de cada una, el número de créditos, la importancia personal que le asigne el estudiante, si la asignatura es teórica o practica y el tiempo libre disponible.

1.2 OBJETIVOS DE LA INVESTIGACIÓN

1.2.1 Objetivo general

Desarrollar una aplicación WEB para la gestión dinámica del tiempo de estudio de los estudiantes de Ingeniería de Sistemas de la Universidad de Oriente, Núcleo de Monagas.

1.2.2 Objetivos específicos

1. Analizar la situación actual con respecto a la problemática presente, enfocado a la escuela de Ingeniería de Sistemas de la Universidad de Oriente, Núcleo de Monagas.

2. Establecer los requisitos funcionales y no funcionales de la aplicación para la etapa de diseño.
3. Diseñar una arquitectura tomando en cuenta las variables claves que permitan a la aplicación la óptima organización de horarios de estudios.
4. Desarrollar los módulos de la aplicación WEB que cumpla con los requisitos y la arquitectura establecida que permita organizar los horarios de estudios de los estudiantes.

1.3 JUSTIFICACIÓN DE LA INVESTIGACIÓN

El desarrollo de la disciplina a nivel académico es fundamental para garantizar profesionales de calidad, con el avance tecnológico actual es inevitable pensar en la no implementación de nuevas aplicaciones para solventar y ayudar con los problemas de los ciudadanos. En este sentido, la presente investigación busca alinearse con estos intereses y elaborar una herramienta para minimizar los inconvenientes ocasionados por la procrastinación y su relación con el bajo desempeño académico en los estudiantes de la carrera Ingeniería de Sistemas de la Universidad de Oriente Núcleo Monagas enfocándose en el caso de estudio de la asignatura Taller de Programación Orientada a Objetos (0722111).

Esta plataforma es beneficiosa para la Universidad de Oriente, Núcleo de Monagas pues permite elevar la calidad del estudiante universitario al ofrecerle un entorno amigable relacionando sus tiempos de ocio y trabajo con las exigencias requeridas por las asignaturas cursantes en su periodo de estudio, orientando la planificación del semestre y logrando como resultado un incremento del rendimiento académico.

La implementación de esta plataforma WEB permitirá reducir significativamente los problemas de los estudiantes de la carrera Ingeniería de Sistemas relacionados con la mala gestión del tiempo, mediante el análisis de las asignaturas inscritas en el periodo académico actual y la creación de un horario de estudio dinámico acorde con las exigencias, esto se lograra por medio de la incorporación de un algoritmo de comparación para los siguientes criterios; el número de créditos de cada asignatura, si el estudio es netamente teórico o también practico, la frecuencia de clases por semana y la dificultad de la asignatura, además, ofrecerá nombres de documentos académicos y bibliografías referentes a las asignaturas del estudiante, guiando y facilitando la búsqueda de información, de tal manera que se pueda enfocar rápidamente en los temas de importancia.

1.4 DELIMITACIÓN DE LA INVESTIGACIÓN

El sistema está orientado para el uso de los estudiantes de Ingeniería de Sistemas de la Universidad de Oriente Núcleo Monagas, recopila información pertinente al pensum académico de esta carrera en particular y fue diseñado con el objetivo de satisfacer las necesidades de esta comunidad. Una investigación posterior que estudie los diversos requerimientos del resto de las carreras universitarias permitirá la incorporación dentro del sistema y ampliar el alcance del mismo.

El sistema está hospedado en un entorno WEB para garantizar el fácil acceso y la compatibilidad entre diversos dispositivos y para que el sistema funcione correctamente se debe disponer como mínimo una conexión a internet de 1 Megabit de velocidad, una unidad de procesamiento Intel Pentium 4 compatible con el conjunto de instrucciones SSE2, 1024 megabytes (MB) de memoria RAM, 800 megabytes (MB) de espacio libre de almacenamiento y sistema operativo Windows 7, OS X Yosemite 10.10, Ubuntu 14.04/Debian 8 de 64bits o dispositivos Android/IOS con capacidad de navegar por internet.

CAPITULO II
MARCO TEORICO

En esta sección de la monografía de investigación se presentan los aportes obtenidos de diversos estudios que sirvieron como guía para la estructuración del proyecto, las bases conceptuales necesarias para entender las herramientas a utilizar en la elaboración de la aplicación WEB para la comunidad estudiantil de la Universidad de Oriente Núcleo de Monagas y el marco legal en el cual se encuentra inmerso la investigación.

2.1 ANTECEDENTES DE LA INVESTIGACIÓN

Según Fontaines (2012) los antecedentes de investigación se definen como una descripción de la evolución histórica de los estudios sobre un tema hasta exponer el estado actual del mismo, sirviendo estos de soporte para la investigación. Para este estudio, se utilizaron los siguientes antecedentes que sirvieron de base y contribuyeron en diferentes aspectos.

El trabajo de grado realizado por Fuentes Gimón & Lagrange Delfino (2010) titulado: *Desarrollo de una plataforma web que interconecte y organice grupos universitario pertenecientes al movimiento estudiantil.* Este trabajo se hizo con la finalidad de optar al título de Ingeniero de Sistemas en la Universidad Metropolitana de Caracas y que tenía como objetivo el desarrollar una red social que interconectara y proveyera herramientas para la comunicación a las distintas agrupaciones universitarias de Venezuela. Esta investigación fue tomada por emplear la metodología de desarrollo de la ingeniería de software, programación extrema también conocida como eXtreme Programming, debido que expone cada una de las

fases de esta metodología de manera clara, se tomó como una guía de apoyo para este trabajo de investigación.

Galletta & Rios (2017) realizaron la investigación titulada: *Plataforma web para publicación y búsqueda de viviendas apoyada en GIS y GP. Caso de desarrollo para Manasés Capriles*. Para optar al título de Ingeniero de Sistemas en la Universidad Metropolitana de Caracas, que tenía por objetivo el desarrollar una plataforma web para la consulta de inmuebles habitacionales con búsquedas basadas en múltiples criterios, optimizadas por Programación por Metas (GP) e integrada con Sistemas de Información Geográfica (GIS) para enriquecer los datos utilizados.

Los aportes que tuvo éste trabajo, fueron las tecnologías utilizadas en su desarrollo, para éste se utilizó un framework MEAN Full Stack para construir y diseñar la plataforma web, así mismo, en nuestro trabajo de investigación se empleará el uso de algunas de éstas tecnologías como el entorno de ejecución multiplataforma NodeJS, el módulo Express y la permanencia de datos en MongoDB.

La investigación realizada por Conde Jaules (2017) titulada: *Aplicación web para la gestión académica del Instituto de Educación Superior Tecnológico Público Churcampa, Región Huancavelica, 2016.* Para optar al título de Ingeniero de Sistemas en la Universidad Nacional de San Cristóbal de Huamanga, que tenía como objetivo el desarrollo de una aplicación web para la gestión académica del Instituto de Educación Superior Tecnológico público Churcampa, con el propósito de automatizar el proceso de gestión académica y optimizar los tiempos en actividades de gestión académica. Este trabajo se utilizó como base fundamental de cómo el uso de la tecnología puede contribuir positivamente en el desarrollo estudiantil, y los beneficios que estos entornos web brindan a los estudiantes, también en el diseño e interfaz del sistema.

2.2 BASES TEÓRICAS

Las bases teóricas representan la esencia conceptual del estudio, es el cuerpo de teorías y conceptos que sirven para interpretar el comportamiento del objeto, por lo tanto, permite poner en evidencia cuanto conoce el investigador sobre el tema al representar clara y estructuradamente lo investigado. (Gracia Cordoba, 2004 citado por Tomas Fontaines Ruiz, 2012) Las bases teóricas representan la explicación de los elementos esenciales que sustentan el proyecto de investigación y forman parte de las bases usadas para llegar a la solución del problema. Estas son las bases teóricas que sustentan la investigación:

2.2.1 Procrastinación

Proveniente del verbo latino procrastinare, la pocrastinación es el hábito de aplazar innecesariamente el inicio o la conclusión de deberes importantes hasta el momento de sentirse irritado o experimentar sensaciones de malestar e incomodidad (Solomon & Rothblum, 1984 citado por Rodríguez & Clariana, 2016). Según (Ferrari, Johnson y McCown, 1995 citado por Sánchez Hernández, 2010) en estudios clinicos realizados en la poblacion norteamericana mas del 40% de las personas han tenido algun tipo de problema con la procrastinacion, iniciandose en la etapa de la adolescencia y se ve acentuada en los estudiantes de educacion superior.

En latinoamerica se observa un 61% de las personas con problemas de algun tipo, con relacion a la procrastinación y el 20% presenta procrastinación cronica, mostrando patrones claros y continuos de comportamiento dilatorio en las tareas que son su responsabilidad (Ferrari, O'Callahan, y Newbegin (2004) citado por Quant & Sánchez, 2012).

2.2.2 Sistema

(Meadows, 2008) Define un sistema como un grupo de elementos, ya sean moléculas, células o personas que interconectadas y relacionadas de cierta manera producen un patrón único de comportamiento en el tiempo. Este sistema puede ser alterado por fuerzas ajenas del mundo exterior pero la respuesta a estas fuerzas es característico del sistema en sí.

En la década de los 60s nace la teoría general de sistemas por Ludwig von Bertalanffy donde se expone que en el mundo todo está conformado por sistemas y sub sistemas, los elementos que existen siempre están en constante relación con su entorno y las funciones de un sistema dependen de su estructura y la interacción de sus elementos.

2.2.3 Tecnologías de la Información y Comunicación (TIC)

Son aquellas tecnologías capaces de obtener por si mismas información de medios externos y utilizarlas en procesos para posteriormente tomar sus propias decisiones y transmitirlas de manera que podamos entenderla (Gil Díaz, 1985) citado por (Grande, Cañón, & Cantón, 2016). Por medio de internet las Tecnologías de la Información y Comunicación nos dan un vasto recurso de conocimiento y presentan ser una ayuda invaluable en el área de lo académico (Belloch, 2012).

2.2.4 Aplicación Web

Son herramientas a las que los usuarios pueden acceder de manera remota que están alojadas en un servidor web, a través de internet o incluso una intranet mediante un navegador, haciendo aún más sencillo su mantenimiento ya que no dependen de actualizar software a miles de usuario, además de ser multiplataforma debido a que

solo se necesita un dispositivo con un navegador web actualizado para acceder a estas. (Wikipedia, 2001).

2.2.5 Desarrollo web Full-Stack

Es el desarrollo de software de lado del cliente, así como también del servidor. En adición a HTML y CSS también debe dominarse un lenguaje de programación (JavaScript), manejar un servidor el cual se encargará de manejar la lógica en el sistema web (NodeJS) y una base de datos que permitirá la permanencia de los datos importantes para los usuarios y del mismo sistema (MongoDB) (Refsnes Data, 2000).

2.2.6 Lenguajes de Marcado

(Fernández-Valmayor, Navarro, Fernández-Manjón, & Sierra, 2006) "Los lenguajes de marcas surgieron, inicialmente, como lenguajes formados por el conjunto de códigos de formato que los procesadores de texto introducen en los documentos para dirigir el proceso de presentación (impresión) mediante una linotipia o impresora programable. Como en el caso de los lenguajes de programación, inicialmente estos códigos de formato estaban ligados a las características de una máquina, programa o procesador de textos concreto y, en ellos, inicialmente no había nada que permitiese al programador (formateador de documentos en este caso) abstraerse de las características del procesador de textos y expresar de forma independiente a éste la estructura y la lógica interna del documento."

Los lenguajes de marcado fueron creados con el propósito de organizar y estructurar la gran cantidad de documentos que presentaban las empresas, estos lenguajes permiten darle sentido al documento y que este no se pierda cuando es transmitido por internet o cuando es necesario su impresión. A diferencia de los

lenguajes de programación estos solo contienen información relevante a la estructura y relevancia del texto dentro de las etiquetas, no maneja aspectos lógicos y de operación.

2.2.7 HTML

(Pfaffenberger & Karow, 2000) En abreviación a Lenguajes de Marcas de Hipertexto, HTML (por sus siglas en inglés) es uno de los muchos lenguajes de marcado que aparecieron en los últimos años. En resumen, este leguaje proporciona las pautas para agregar el marcado y la estructura del texto usando símbolos especiales en los documentos en Internet. Estos símbolos describen las partes del documento, por ejemplo, se pueden utilizar las marcas para identificar cual es el resumen, los títulos entre otros.

HTML es un lenguaje de marcado con todas las ventajas que proporcionan estos lenguajes en lo que respecta a la maquetación del texto, pero este lenguaje va más allá, es un lenguaje que utiliza hipertextos, permitiendo la creación de texto interactivo llamado hiperlink que enlaza esa oración con una sección del mismo documento o cita externa, permitiendo el enlace entre diversos documentos y un acceso más inmediato a la información.

El World Wide Web Consortium (W3C) es el encargado de mantener y estandarizar las normas y estructuras del código HTML, es el encargado de evaluar y publicar las especificaciones de las diversas versiones de HTML, actualmente la última versión de HTML es la versión 5 que incorpora nuevos elementos, atributos y un conjunto de herramientas tecnológicas que permiten desarrollar sitios WEB más dinámicos.

Entre las ventajas de esta última versión se tiene:

- Semántica: Permite describir con mayor precisión cuál es el contenido de la página.
- Sin conexión y almacenamiento: Permite a las páginas web almacenar datos localmente en el lado del cliente y operar sin conexión de manera más eficiente.
- Gráficos y efectos 2D/3D: Proporciona nuevas características en lo que respecta a gráficos en la web como lo son canvas 2D, WebGL, SVG, etc.
- Multimedia: Otorga un excelente soporte para utilizar nativamente contenido multimedia como lo son audio y video.
- CSS3: Ofrece una gran variedad de opciones para hacer diseños más sofisticados. (Mozilla, 1998)

2.2.8 Cascading Style Sheets CSS

Describe la manera como los elementos HTML son mostrados en pantalla, papel u otro tipo de multimedia (Refsnes Data, 2000). Hoy en día prácticamente todos los navegadores tienen soporte de CSS al igual que muchas otras aplicaciones. Para escribir código de CSS no se necesita más que un editor de texto, sin embargo, existen muchas más herramientas disponibles que lo hacen mucho más sencillo. (World Wide Web Consortium, 1994)

2.2.9 Bootstrap

Bootstrap es un conjunto de herramientas de diseño de código abierto que permiten fácilmente modificar, animar, alinear, organizar y darle vida al desarrollo de HTML, CSS, JS y funcionalidades hechas en jQuery (Otto & Jacob Thornton, 2011).

Los desarrolladores pueden tomar ventaja del uso de las clases CSS ya definidas en Bootstrap para personalizar aún más la apariencia del contenido de sus aplicaciones.

2.2.10 Lenguajes De Programación

(Iverson, 1967) define el lenguaje de programación como un lenguaje que es adecuado para la descripción de algoritmos complejos, es un lenguaje conciso, corto en simbología, preciso y su uso es consistente en una amplia área de aplicación, además, debe exhibir claramente la secuencia en las que se realizan las operaciones. Los lenguajes de programación sirven para representar de una forma lógica un conjunto de instrucciones que permiten indicarle a un computador los pasos necesarios para realizar una acción en específico.

2.2.11 JavaScript

JavaScript es un lenguaje de programación interpretado que funciona directamente en el navegador, es orientado a objetos y su primer propósito fue enfocado al desarrollo WEB, JavaScript funciona mediante la creación de scripts que les agregan dinamismo y funcionalidad a los elementos HTML, utilizando el DOM (modelo de objeto de documento) proporcionado por los navegadores para manipular el lado visible por el cliente en una página.

Según (Haverbeke, 2018) JavaScript es un lenguaje de programación introducido al mercado en el año 1995 como una forma de agregar programas a las páginas web en el navegador Netscape, desde entonces fue adoptado por otros navegadores y es el lenguaje que permitió la creación de las modernas aplicaciones WEB que disponemos actualmente.

JavaScript no solo se quedó para el desarrollo de aplicaciones del lado del cliente, junto al entorno de ejecución multiplataforma Node.js permite también trabajar del lado del servidor, logrando que con un mismo lenguaje se pueda crear una aplicación WEB completa. El conjunto de librerías y frameworks de JavaScript lo han convertido en uno de los lenguajes más flexibles que existen actualmente, con React Native se pueden desarrollar aplicaciones móviles para Android, IOS o el framework Electron para el desarrollo de aplicaciones en Windows y Mac.

2.2.12 jQuery

Es una rápida, pequeña y enriquecida librería de JavaScript, hace cosas como el recorrido de documentos HTML y su manipulación, manejo de eventos, y animación sean mucho más sencillos, con una combinación de versatilidad y extensibilidad. jQuery también provee de un paradigma para manejar eventos que van más allá de la selección y manipulación básica del DOM. También apuesta por incorporar otras funcionalidades bastante usadas de JavaScript (The jQuery Foundation, 2005).

2.2.13 Node.js

(Nguyen, 2012) Node.js es una plataforma JavaScript del lado del servidor que consiste en una biblioteca central de funciones junto a un minimalista ecosistema de desarrollo. Se ejecuta sobre el motor V8 de JavaScript creado por los ingenieros de Google haciéndolo muy rápido para la ejecución de código. Node.js está construido en base al tiempo de ejecución de JavaScript de Google, fue creado para la rápida y escalable creación de aplicaciones en red. Node.js usa un modelo de I/O no bloqueante basado en eventos haciéndolo liviano y eficiente, ideal para la creación de aplicaciones multiplataforma de ejecución en tiempo real con un intensivo uso de datos.

Ventajas de Node.js

Desde el punto de vista de desarrollo WEB, Node.js otorga una gran cantidad de ventajas:

- Node.js fue diseñado para optimizar el rendimiento y la escalabilidad de las aplicaciones WEB.
- Su código está escrito en "JavaScript básico" lo que permite que los desarrolladores pierdan menos tiempo en aprender un nuevo contexto entre lenguajes cuando se está escribiendo código tanto para el Front-end como el Back-end.
- JavaScript es un lenguaje de programación relativamente nuevo y trae consigo los beneficios del avance de diseño de otros lenguajes cuando se compara con (Python, PHP, etc)
- Viene con un gestor de paquetes integrado NPM (Node Packet Manager) que proporciona acceso inmediato a una gran cantidad de paquetes reutilizables.
- Es portable, funciona en entornos como Microsoft Windows, OS X, Linux, Solaris, FreeBSD, entre otros. Además, tiene un amplio soporte en los diversos proveedores de hospedaje web, incorporando la documentación necesaria para el hospedaje de sitios que usen Node.js
- Tiene un ecosistema y una comunidad de desarrolladores bastante amplia y activa que facilita la solución de problemas y el soporte mutuo.

2.2.14 Express

Según su página WEB oficial. "Express es una infraestructura de aplicaciones web Node.js mínima y flexible que proporciona un conjunto sólido de características para las aplicaciones web y móviles." (StrongLoop, Inc., 2010) Express es un

Framework que aporta funcionalidades como el enrutamiento, opciones para la gestión de sesiones, manejos de cookies entre otros, además, aporta mecanismos para:

- Escritura de manejadores de peticiones con diferentes verbos HTTP en diferentes caminos URL (rutas).
- Integración con motores de renderización de "vistas" para generar respuestas mediante la introducción de datos en plantillas.
- Establecer ajustes de aplicaciones web como qué puerto usar para conectar, y la localización de las plantillas que se utilizan para renderizar la respuesta.
- Añadir procesamiento de peticiones "middleware" adicional en cualquier punto dentro de la tubería de manejo de la petición. (Mozilla, 1998)

2.2.15 Base de Datos

Formalmente una base de datos se refiere a un conjunto de datos relacionados y a la manera que están organizados. El acceso a éstos datos normalmente viene dado por el uso de un sistema para el manejo de bases de datos, y consiste en un grupo de software de computadora que le permite al usuario interactuar con una o más bases de datos. Dado la estrecha relación entre estos, el término “base de datos” es comúnmente utilizado para referirse a ambos términos. Podemos decir entonces que es una herramienta que nos permite recopilar y almacenar datos que puede ser utilizada posteriormente para en otras tareas, dando como consecuencia la permanencia de los mismo (Microsoft, 1991).

2.2.16 NoSQL

NoSQL abarca una gran variedad de tecnologías de bases de datos no relacionales, y provee un mecanismo de almacenamiento y recuperación de datos

diferente a las relaciones tabulares de las bases de datos relacionales, todo esto en respuesta a las demandas presentes de crear aplicaciones modernas, la motivación para este enfoque es apostar por un diseño más simple (Merriman, Horowitz, & Ryan, 2008).

2.2.17 MongoDB

(Merriman, Horowitz, & Ryan, 2008) MongoDB es un poderoso, flexible y escalable almacén de datos orientado a documentos, combina la rápida escalabilidad junto a las características más útiles de las bases de datos relacionales. Según la definición encontrada en la página oficial de MongoDB "MongoDB es una base de datos distribuida, basada en documentos y de uso general que ha sido diseñada para desarrolladores de aplicaciones modernas y para la era de la nube. Es una base de datos documental, lo que significa que almacena datos en forma de documentos tipo JSON." (Merriman, Horowitz, & Ryan, 2008).

MongoDB es una base de datos NoSQL orientada a documentos, esta base de datos maneja los datos mediante un formato propietario de intercambio de datos llamado BSON basado en el término JSON (JavaScript Object Notation), este formato consiste en una lista ordenada de objetos cada uno con sus campos, entre los elementos soportados están; la cadena de caracteres (String), enteros de 32 y 64bits, arreglos, variables booleanas, variables nulas, expresiones regulares, código JavaScript entre otros.

2.2.18 Visual Studio Code

Según (Sole, 2019) Visual Studio Code es un editor de código fuente basado en Electron, un Framework para crear aplicaciones multiplataforma con tecnologías nativas, combina la simplicidad de un poderoso editor de código con las herramientas

que un desarrollador necesita para apoyar el desarrollo durante el ciclo de vida de la aplicación.

La definición extraída de la página web oficial de Visual Studio Code nos dice que: Visual Studio Code es un ligero editor de código fuente, pero potente en relación a su aplicabilidad, con licencia gratis para uso privado o comercial que se ejecuta en el escritorio y está disponible para Windows, MacOs, y Linux. Viene con soporte para lenguajes como JavaScript, TypeScript y Node.js además tiene un mecanismo para agregar extensiones a otros lenguajes (como C++, #, Java, Python, PHP, Go) y tiempos de ejecución (como .NET y Unity).

Este editor de código esta optimizado con soporte para operaciones de desarrollo como depuración, ejecución de tareas y control de versiones con integración en Git y GitHub. Su objetivo es proporcionar las herramientas que un desarrollador necesita para mantener una creación ágil de código y depuración, dejando los trabajos más complejos a la utilización de IDE más complicados de manejar.

2.2.19 Git

Git es un sistema de control de versión libre y de código abierto diseñado para manejar proyectos pequeños como también proyectos grandes con velocidad y eficiencia. Está diseñado para coordinar el trabajo entre programadores, pero también puede ser utilizado para tener control de los cambios en cualquier tipo de archivo. Cada directorio Git en cada computadora es un repositorio completo independientemente del acceso que se tenga a la red o al servidor central (Chacon & Long, 2008).

2.2.20 GitHub

GitHub es una plataforma de desarrollo, para manejar proyectos, desarrollo de software, y permite alojar y revisar código, provee a los equipos un lugar donde compartir el progreso de sus trabajos, aportando un abanico de herramientas y seguridad para garantizar un desempeño de alta calidad. Provee control de acceso, rastreo de bugs, petición de funciones, manejo de tareas y páginas wiki para cada proyecto (Friedman, y otros, 2007).

2.2.21 UML

(Si Alhir, 2003) UML es un lenguaje visual para modelar y comunicar el comportamiento de los sistemas mediante el uso de diagramas y texto de soporte. En la figura 1 se puede visualizar las siguientes relaciones de ejemplo:

- Un gerente dirige a un equipo para la ejecución de un proyecto.
- Cada gerente tiene un nombre y un número de teléfono, y puede iniciar un proyecto y terminar un proyecto.
- Cada proyecto tiene un nombre, una fecha de inicio y una fecha de culminación.
- Cada equipo tiene una descripción.

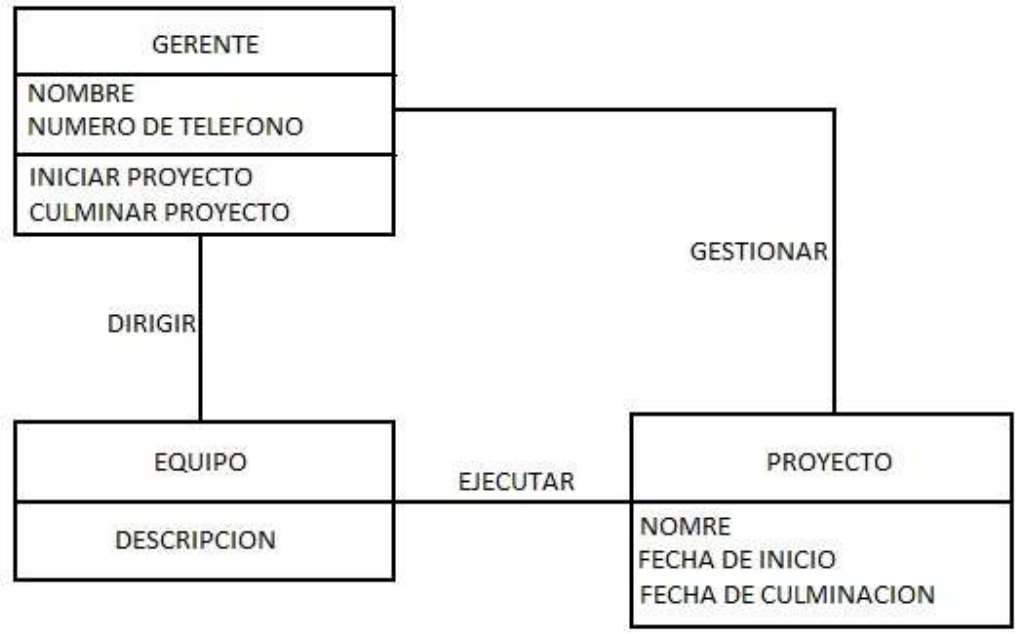


Figura 1. Diagrama modelo de un equipo de gestión de proyecto
Fuente: Autores (2020)

El lenguaje unificado de diagrama o notación (UML) es un lenguaje grafico utilizado para visualizar, construir y documentar un sistema en específico. Existen dos tipos de diagramas, los estructurales y los de comportamiento.

Diagramas estructurales, estos diagramas muestran la estructura estática de los objetos de un sistema y se dividen en:

- **Diagrama de clases:** Muestran las clases de un sistema, atributos y operaciones de cada clase en relación con las otras clases.
- **Diagrama de componentes:** Ejemplifica la relación estructural entre los componentes de un sistema, comúnmente usado en sistemas que incorporen un gran número de componentes que se comunican entre sí.

- **Diagrama de despliegue:** Muestra en hardware y software utilizado por el sistema. Estos diagramas son útiles cuando la solución se despliega a varios equipos con configuraciones diferentes.
- **Diagrama de objetos:** Estos diagramas se utilizan para mostrar cómo se representarán los objetos y sus relaciones en el sistema en un momento dado con respecto a un estado de ejecución.
- **Diagrama de paquetes:** Representa las dependencias entre los paquetes utilizados por el sistema, muestra como el sistema está dividido en grupos lógicos.
- **Diagrama de estructura compuesta:** Se utilizan para mostrar las estructuras internas de una clase y cómo interactúan estas con los demás elementos del sistema para realizar una acción en concreto.

Diagramas de Comportamiento, estos diagramas representan los aspectos dinámicos del sistema.

- **Diagrama de actividades:** Representan los flujos de trabajo de forma visual, mediante la implementación de gráficos que muestren el proceso de trabajo operativo en cualquier componente de un sistema.
- **Diagrama de casos de uso:** Estos diagramas representan las interacciones que puede tener el sistema con respecto a su entorno y van orientadas desde el punto de vista del usuario final.
- **Diagrama de máquina de estados:** Su aplicación va orientada a la representación del comportamiento de un objeto y documentar como responde a determinados eventos, tomando en cuenta las relaciones internas y externas que tenga.

2.2.22 Metodología

Para (Viñán Villagrán, Navarrete Chávez, Puente Riofrio, Pino Burgos, & Caicedo Benavides, 2018) “La Metodología de la Investigación Científica es una de las etapas específicas de un trabajo o proyecto que parte de una posición teórica y conduce una selección de técnicas concretas (o métodos) acerca del procedimiento destinado a la realización de tareas vinculadas a la investigación, el trabajo o el proyecto”. Es el conjunto de pasos necesarios para la organizada elaboración de una investigación. En el caso de metodologías para el desarrollo de software estas consisten en un marco de trabajo usado para estructurar, planear y controlar todo el proceso referente al ciclo de vida del sistema de información.

2.2.23 Metodología XP

Según (Bautista Q., 2010) La programación extrema o eXtreme Programming (XP) es un enfoque de la ingeniería de software formulado por Kent Beck, fundamentado en la realimentación continua entre el cliente y el equipo de desarrollo, una comunicación fluida entre los participantes, la simplicidad en las soluciones implementadas y el coraje para enfrentar los cambios. XP es el proceso ágil más destacado dentro del desarrollo de software y se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad.

2.2.23.1 Características de la metodología XP

- Metodología basada en prueba y error para obtener un software funcional.
- Fundamentada en principios.

- Está orientada hacia quien produce y usa software (el cliente tiene una participación activa).
- Los requisitos pueden cambiar.
- Grupo pequeño y muy integrado (2-12 personas).
- Combina las que han demostrado ser las mejores prácticas para desarrollar software, y las lleva al extremo.

2.2.23.2 Fases de la programación extrema

La Programación Extrema consta de 4 fases:

2.2.23.2.1 Planeación

En ésta primera fase se plantea un diálogo continuo con el cliente para abordar los requisitos y los casos de usos necesarios para desarrollar.

- **Historias de los usuarios**, son pequeñas oraciones orientadas a guiar a los desarrolladores con respecto a lo que el sistema debe realizar, éstas constan de 3 a 4 líneas escritas por el cliente en un lenguaje no técnico.
- **Plan de Entrega**, con las historias ya definidas se establece un plan de acción para cada versión del programa, tomando en cuenta que hojas de usuario se irán desarrollando y las fechas en las que se publicarán las versiones.
- **Plan de Iteración**, las iteraciones serán divididas aproximadamente cada 3 semanas y al inicio de cada iteración se deben tomar en cuenta las historias de los usuarios a desarrollar en esa iteración y de ser necesario las historias que no cumplieron las expectativas de la iteración pasada. Estas historias son divididas en tareas para los programadores.

- **Programación en pareja:** la metodología XP recomienda la programación en parejas pues mejora la productividad y localidad del software desarrollado, consiste en disponer de dos programadores en un mismo equipo donde uno codifica y el otro analiza si el código está bien implementado o si se puede mejorar
- **Reuniones Diarias,** son reuniones con el objetivo de exponer los problemas presentados, se busquen soluciones y se formulen ideas, todos los miembros deben estar presente y el peso del voto debe ser equitativo para el grupo.

2.2.23.2.2 Diseño:

La segunda fase hace énfasis en la elaboración de diseños simples y claros.

- **Simplicidad,** un diseño simple es desarrollado mucho más rápido que uno complejo, por eso XP propone elegir el diseño funcional más simple.
- **Soluciones 'Spike',** Cuando se presentan situaciones donde se es difícil hallar una solución idónea y no se pueda estimar el tiempo necesario para implementar una historia de usuario, se pueden desarrollar pequeñas soluciones de prueba llamadas 'Spike' para explorar diversas soluciones.
- **Refactoring,** XP promueve el uso de la recodificación, consiste en escribir nuevamente parte de un módulo sin cambiar su funcionalidad con el objetivo que mejorar el código, hacerlo más simple, conciso y entendible.

2.2.23.2.3 Codificación

- **Uso de Estándares,** XP promueve la programación basada en estándares, con el objetivo que sea fácilmente entendible por todo el equipo y facilite la codificación.

- **Integraciones permanentes,** todos los desarrolladores deben trabajar en la última versión del sistema, tratar de mejorar versiones antiguas puede acarrear un lastre para el equipo y es mejor seguir creando versiones diarias que mejoren al actual.
- **Propiedad colectiva,** Todos los miembros de un equipo pueden aportar ideas en cualquier parte del código.
- **Ritmo Sostenido,** La metodología indica que debe llevarse un ritmo sostenido de desarrollo, la meta de esta filosofía en planificar con antelación el desarrollo de la aplicación y evitar sobrecargar el equipo.

2.2.23.2.4 Pruebas

- **Pruebas unitarias,** Todos los módulos del programa deben pasar las pruebas establecidas antes de ser publicados.
- **Detección y corrección de errores,** Cuando se encuentren nuevos errores en el programa estos deben ser corregidos inmediatamente, y se deben de tomar las precauciones para que no vuelvan a ocurrir junto con el desarrollo de las pruebas necesarias para verificar que el error fue solucionado.
- **Pruebas de aceptación,** Las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada (Joskowicz, 2008).

2.3 BASES LEGALES

Según (Pérez, 2002) define las bases legales como “el conjunto de leyes, reglamentos, decretos, etc…, que establecen el basamento jurídico sobre el cual se sustenta la investigación.” En otras palabras, las bases legales representan el conjunto

legal que fundamenta la investigación y que guarda relación con el tema desarrollado, analizado e interpretado por los autores de la investigación.

2.3.1 Constitución de la República Bolivariana de Venezuela (1999)

La constitución representa los principios legales que rigen a la sociedad venezolana, constando de 350 artículos relacionados a diversas áreas legales que ejercen dictámenes referentes al poder público, los deberes y derechos humanos, el espacio geográfico y la división política, la seguridad de la nación, la protección legal, las reformas constitucionales, el sistema socio-económico y los principios fundamentales del país.

> Artículo 108:
>
> Los medios de comunicación social, públicos y privados, deben contribuir a la formación ciudadana. El estado garantizará servicios públicos, de radio, televisión, redes de bibliotecas y de informática, con el fin de permitir el acceso universal a la información, los centros educativos deben incorporar el conocimiento y aplicación de las nuevas tecnologías a su innovación, según los requisitos que establezca la ley (p.96).

> Artículo 110:
>
> El estado reconocerá el interés público de la ciencia, la tecnología, el conocimiento, la innovación y sus aplicaciones, los servicios de información necesaria por ser instrumentos fundamentales para el desarrollo económico, social y político del país, así como para la seguridad y soberanía nacional, para el fomento y desarrollo de esas actividades, el estado destinará recursos suficientes y creará el sistema nacional de los principios éticos y legales que deben mejorar las actividades de investigación científica, humanística y tecnológica (p.98).

Los artículos anteriores hacen referencia a que el estado garantizara el apoyo a las ideas de innovación referentes a herramientas tecnológicas que permitan elevar la calidad de vida de los ciudadanos y aporten al crecimiento del país, resaltando el uso

de las nuevas tecnologías para la distribución de la información y la creación de ciudadanos capaces de fomentar conocimiento orientándose así a los principios enmarcados en la ley. Estos artículos sustentan el trabajo en cuanto a la aprobación del estado para la creación de una aplicación WEB que permita aumentar la calidad académica de profesionales en el país.

2.3.2 Ley Orgánica de Ciencia, Tecnología e Innovación (2010)

Artículo 3:

> Forma parte del sistema nacional de ciencias y tecnología e innovación, las instituciones públicas o privadas que generen o desarrollen conocimientos científicos y tecnológicos, como procesos de innovación y las personas que se dediquen a la planificación, administración, ejecución y aplicación de actividades que posibiliten la vinculación efectiva entre la ciencia y tecnología y la sociedad (p.9).

Según este artículo las instituciones de educación del país sean públicas o privadas, deben ser parte de la generación de conocimiento tecnológico y de investigación con el pro de aumentar la relación que existe entre la sociedad venezolana y los avances tecnológicos en el mundo y en el país. Este articulo sustenta la investigación por la orientación de la misma de crear una aplicación que utilizando las tecnologías actuales permita mitigar el impacto de la procrastinación en la comunidad estudiantil de la escuela de Ingeniería de Sistemas de la Universidad de Oriente enlazando la tecnología y la vida cotidiana del ciudadano.

2.4 DEFINICION DE TÉRMINOS BÁSICOS

Lenguaje de Programación Interpretado: Su código no requiere ser compilado, mayormente consisten de funciones que son interpretadas en tiempo real por un intérprete, esto permite maximizar la eficiencia de los programas. Estos lenguajes

usualmente se ven en la programación web y la creación de programas cliente/servidor.

Modelo de Objeto de Documento (DOM): Es una interfaz de programación de aplicaciones para documentos HTML y XML. Es la base de todo programa hecho en estos formatos y define la estructura lógica de los documentos y el modo en que se accede y manipula un documento.

Framework: Es un esquema o patrón para el desarrollo e implementación de una aplicación. Es un conjunto de conceptos, prácticas y criterios que ayudan a enfocar un tipo de problemática en particular, nos sirve como referencia para enfrentar y resolver problemas similares.

Librería: Es un grupo de implementaciones, codificadas en un lenguaje de programación, que ofrece una interfaz para la funcionalidad que se invoca. Estas le dan una mayor utilidad a los programas que se están desarrollando añadiéndole nuevas funciones e incluyendo algoritmos ya preestablecidos.

Front-End: Es el desarrollo frontal de una aplicación, esto quiere decir todo lo que interactúa con el usuario final, de lado del cliente. La interfaz, el diseño, las transiciones y animaciones son manejados por el desarrollo de front-end.

Back-End: Es el desarrollo del motor de la aplicación, es el encargado de manejar la lógica del programa y los datos que utiliza para sus operaciones, es el diseño de lado servidor al que el usuario final no tiene acceso.

Enrutamiento: Es la función de conseguir un camino entre todos los posible en una red de paquetes cuyas topologías poseed una gran conectividad, buscando la mejor ruta. Para encontrar esta ruta optima se tener en cuenta la tabla de enrutamiento y

algunos otros parámetros como la métrica, la distancia administrativa y el ancho de banda.

Cookies: Son pequeños archivos que se alojan en la computadora de un usuario, están diseñados para contener una modesta cantidad de data específica a un cliente en particular y un sitio web, y puede ser accesada por el servidor web o la computadora cliente. Esto permite al servidor enviar una página diseñada particularmente para ese usuario.

Middleware: Es un intercomunicador de varias aplicaciones de software y ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre una plataforma. Funciona como una capa de abstracción de software.

CAPITULO III
DESARROLLO DE LOS OBJETIVOS

3.1 TIPO Y DISEÑO DE LA INVESTIGACION

De acuerdo con el tema de estudio, esta investigación está enmarcada en un proyecto factible, constó de fases orientadas a la recolección de información, diseño y desarrollo de una aplicación WEB como método viable para solucionar problemas relacionados al desempeño académico de los estudiantes de ingeniería de Sistemas en la Universidad de Oriente - Núcleo Monagas. (Arias, 2006) Define un proyecto factible como "Una propuesta de acción para resolver un problema practico o satisfacer una necesidad. Es indispensable que dicha propuesta se acompañe de una investigación, y demuestre su factibilidad o posibilidad de realización".

La propuesta del proyecto busca aportar una herramienta tecnológica para dar solución a problemas de rendimiento académico referentes a una mala gestión del tiempo libre y oriente a los estudiantes para el desarrollo de buenos hábitos de estudios. Según (Arias, 2014, pág. 6) la investigación tecnológica "Es la búsqueda y obtención de nuevos conocimientos prácticos y aplicados a corto plazo en la creación, producción o desarrollo de bienes y servicios innovadores, artefactos, materiales, prototipos o maquinarias que contribuyan a resolver problemas, satisfacer necesidades y mejorar la calidad de vida de la sociedad". Por ende, la investigación actual está enmarcada en esta definición porque tiene como objetivo, el desarrollo de un prototipo tecnológico que no está implementado actualmente, capaz de traer soluciones a problemas presentes en la Universidad de Oriente – Núcleo de Monagas.

3.2 NIVEL DE LA INVESTIGACION

Esta investigación está enmarcada en un nivel descriptivo de conocimiento, debido a que se fundamenta principalmente en hechos y procesos que no han sido desarrollados previamente en el área donde se plantea el proyecto. El producto de esta monografía de investigación es una descripción detallada del problema y una solución fundamentada por las evidencias encontradas. La información necesaria para el desarrollo del proyecto fue recolectada a través del análisis de referencias bibliográficas y la realización de encuestas a los estudiantes de Ingeniería de Sistemas de la Universidad de Oriente – Núcleo de Monagas.

3.3 DISEÑO METODOLOGICO CON TECNICAS E INSTRUMENTOS

Para la adquisición de información relevante al problema es necesario realizar técnicas que permitan la recolección y análisis de los datos obtenidos, en este proyecto se hizo énfasis en el uso de encuestas estructuradas de ámbito impersonal con el objetivo de delimitar los aspectos importantes a tomar en cuenta y disminuir el tiempo requerido para la toma de estos datos. A continuación, se describen con mayor profundidad las técnicas utilizadas en la investigación:

3.3.1 Revisión documental

Este proceso se empleó para obtener información relacionada con la problemática presente y con las herramientas necesarias para la realización de la aplicación, ayudó a obtener una idea más precisa en la obtención de la solución del problema y sustentar el proyecto con investigaciones referentes al tema, los elementos a indagar fueron, trabajos de grado relacionados, páginas WEB, libros, revistas científicas, entre otros.

3.3.2 Encuestas estructuradas

Para (Muñoz Martínez, 2020) la encuesta estructurada es la herramienta principal de algunos tipos de entrevista por la cual se recolecta información. Es el tipo de encuesta que esta previamente elaborada y tiene unos parámetros fijos y que no cambian con el tiempo, así mismo las preguntas mantienen un orden especifico, además facilitar su administración y evaluación.

3.3.3 Encuestas impersonales

Según (Fontaines, 2012) las encuestas son una fusión entre la entrevista y la observación estructurada, se caracterizan por la obtención de información específica delimitada en un protocolo o instrumento de investigación, que pueden ser analizada estadísticamente y los resultados extrapolados a la población en la que fueron extraídos. Para este proyecto se realizaron encuestas impersonales, maquetadas en una aplicación WEB y enviadas a grupos relacionados con la carrera de ingeniería de Sistemas de la Universidad de Oriente con el fin de obtener datos provenientes de los estudiantes con respecto a la planificación de sus estudios.

3.4 POBLACIÓN Y MUESTRA

3.4.1 Población

Fontaines (2012) define a la población como la totalidad de unidades seleccionadas mediante criterios establecidos por el investigador, que pueden ser observadas durante la investigación y que se encuentran relacionadas con el universo hipotético de estudio. Los elementos que componen la población deben presentar características comunes que los delimiten de otros individuos.

En la Universidad de Oriente - Núcleo de Monagas se imparten clases para las carreras de Ingeniería de Petróleo, disponiendo de un total de 926 estudiantes inscritos, Licenciatura en Contaduría Pública (896), Ingeniería de Sistemas (807), Licenciatura en Gerencia de Recursos Humanos (531), Licenciatura en Administración (383), Ingeniería Agronómica (381), Licenciatura en Tecnología de los Alimentos (293) e Ingeniería en Producción Animal (224), representando así el total de 4441 inscritos para el periodo III-2019 (Datos obtenidos de las autoridades del núcleo). Para objeto de esta monografía se tomó como población a la escuela de Ingeniería de Sistemas dada la naturaleza de la investigación y por ser ésta la escuela que incorpora la asignatura Taller de Programación Orientada a Objetos siendo parte de nuestro caso de estudio.

3.4.2 Muestra

Sampieri, Collado, & Lucio (2006) definen a la muestra cómo; un subgrupo representativo de la población de interés, los cuales serán la base de la recolección de datos y que dada su importancia deben delimitarse usando herramientas estadísticas que garanticen su veracidad, permitiendo que los resultados encontrados en la muestra sirvan para la generalización o extrapolación a la población.

Con una población de 807 estudiantes inscritos en la escuela de Ingeniería de Sistema de la Universidad de Oriente Núcleo de Monagas para el periodo académico III-2019, se calculó el tamaño de la muestra representativa mediante la siguiente ecuación, tomando en cuenta un nivel de confianza del 80% y un error de estimación del 6%.

Ec. 1 $$n = \frac{Z^2 \times p \times q \times N}{e^2 \times (N-1) + (Z^2 \times p \times q)}$$

Donde:

n - Es el tamaño de la muestra.

Z – Nivel de confianza.

e – Error de estimacion o Muestreo.

p, q – Porcentaje entre alternativas ante respuestas dicotomicas.

N – Tamaño de la poblacion

Sustituyendo en la ecuacion se puede observar que el resultado obtenido es de aproximandamente 100 estudiantes a encuestar:

$$\text{Ec. 2 } n = \frac{(1.28)^2 \times 0.5 \times 0.5 \times 807}{(0.06)^2 \times (807-1) + ((1.28)^2 \times 0.5 \times 0.5)}$$

$$\text{Ec. 3 } n = \frac{330.55}{2.90 + 0.41}$$

$$\text{Ec. 4 } n = 99.863 \approx 100$$

3.5 TECNICAS DE ANALISIS DE DATOS

Para (Hurtado, 2000) el análisis de los datos puede ser utilizado en investigaciones descriptivas cuando se quiere hacer un diagnóstico y evidenciar contenidos significativos entre diversas técnicas de recolección de datos como lo son las entrevistas, conversaciones u observaciones. Para el análisis de los datos arrojados por las encuestas se optó por realizar las siguientes técnicas de análisis de datos para describir los comportamientos de los estudiantes para con sus estudios:

1. Índices de resultados
2. Porcentajes

3.6 DISEÑO OPERATIVO

Para el desarrollo de esta investigación se optó por usar una metodología ágil como los es la Programación Extrema o XP por sus siglas en ingles. Se escogió esta metodología porque su versatilidad permite mantener un orden entre los procesos necesarios para llevar a cabo el proyecto además de ir secuencialmente mejorando la versión desarrollada hasta la fecha límite de entrega establecida, el uso de historias de usuario permite obtener información directa de los estudiantes y generar un plan de acción para incorporar funcionalidades al proyecto y la programación en pares genera un ambiente de productividad que ayuda a evitar errores de código y aumenta la comunicación. En la elaboración de la aplicación se tomaron en cuenta las tres primeras fases de la metodología, estando estas relacionadas con los objetivos específicos planteados, las fases son:

3.6.1 Fase I: Planeación

En esta primera fase de la metodología XP se busca evaluar y describir la situación actual y la problemática presente, representando lo diferentes puntos que se deben tomar en cuenta para poder pasar a la parte del diseño de la solución. En esta fase se va a recoger y analizar información referente a la descripción de la situación actual, análisis de encuestas realizadas a los estudiantes de la carrera de Ingeniería de Sistemas de la Universidad de Oriente Núcleo Mongas, la identificación de los agentes y los problemas presentes, la estructuración de metas y la representación gráfica del proceso actual de planificación de horarios de estudio de los estudiantes. La metodología programación extrema permite organizar parte de los requerimientos solicitados por los clientes, en historias de usuarios que representan fichas resumidas de las especificaciones que desea el cliente para posteriormente generar tareas específicas para cada historia y establecer el proceso de codificación de la aplicación

3.6.2 Fase II: Diseño

En esta segunda fase se busca organizar la información obtenida de la etapa anterior y presentar una estructura que permita generar la solución que será codificada en la fase de desarrollo. Las actividades de esta sección permitirán identificar la arquitectura del sistema propuesto, la representación gráfica de los casos de usos y las diversas funcionalidades del mismo, la interconexión de los agentes que representan el sistema, el acceso del usuario final a las diversas pantallas, la definición del modelo matemático que permitirá la funcionalidad principal de la aplicación y la definición del autómata de proceso central.

3.6.3 Fase III: Desarrollo

En esta tercera y última fase se codificará la solución propuesta establecida y diseñada en la etapa anterior, se explicará brevemente el proceso de desarrollo de la aplicación junto con las tecnologías utilizadas junto a su aplicación en el código y se representará gráficamente las diferentes pantallas que el usuario final podrá visualizar en su computador, explicando la funcionalidad de cada una de ellas.

Las fases fueron sustentadas por un conjunto de actividades que generaran el valor agregado a cada etapa del desarrollo de la investigación, siguiendo un proceso coherente y organizando las metas que se deben ir cumpliendo para lograr la solución a la problemática presente, el siguiente cuadro ejemplifica gráficamente la relación entre los objetivos, la metodología a usar, las fases y las actividades para cada etapa.

Tabla 1. Diseño cuadro Operativo

OBJETIVOS ESPECÍFICOS	METODOLOGÍA	FASES	ACTIVIDADES
1. Analizar la situación actual con respecto a la problemática presente, enfocado a la escuela de Ingeniería de Sistemas de la Universidad de Oriente, Núcleo de Monagas.	Extreme Programing / UML	Fase I: Planeación	- Análisis de la situación actual - Análisis metas y submetas de la situación actual
2. Establecer los requisitos funcionales y no funcionales de la aplicación para la etapa de diseño.	Extreme Programing / UML	Fase I: Planeación	- Análisis conjunto de la aplicación propuesta - Establecimiento de los requerimientos de la aplicación

OBJETIVOS ESPECÍFICOS	METODOLOGÍA	FASES	ACTIVIDADES
3. Diseñar una arquitectura tomando en cuenta las variables claves que permitan a la aplicación la óptima organización de horarios de estudios.	Extreme Programing / UML	Fase II: Diseño	-Diseño base de la aplicación propuesta - Representación mediante diagramas de caso de uso, clases, secuencia, y actividades las funciones que tendrá la aplicación. - Determinación del modelo matemático para el diseño de la aplicación propuesta. - Definición autómata de la aplicación propuesta.

OBJETIVOS ESPECÍFICOS	METODOLOGÍA	FASES	ACTIVIDADES
4. Desarrollar los módulos de la aplicación WEB que cumpla con los requisitos y la arquitectura establecida que permita organizar los horarios de estudios de los estudiantes.	Extreme Programing / UML	Fase III: Desarrollo	- Codificación de la arquitectura diseñada. - Representación de la aplicación propuesta, diseño de interfaz. - Casos de prueba

Fuente: Autores (2020)

3.7 PRIMERA FASE

3.7.1 Análisis de la situación actual

3.7.1.1 Descripción de la situación actual

Actualmente, la tecnología ha permitido el acceso a herramientas de asistencia para las tareas del individuo, ofreciéndole soluciones alternativas a problemas cotidianos, por ejemplo, el alcance a información bibliográfica no disponible en físico en una biblioteca, como también cursos de estudio en línea. Sin embargo, no solo ha posibilitado la obtención de información sino también múltiples maneras de ocio.

Debido a esto, muchos jóvenes emplean su tiempo en actividades de holganza teniendo por consecuencia altos niveles de procrastinación, además, carecen de herramientas efectivas para generar un plan de estudio y organizar su tiempo óptimamente. Es por ello que surgió la motivación a crear una aplicación web que proporcione al estudiante con instrumento para crear un programa que canalice de manera eficaz sus horas libres en la cultivación de su formación académica.

3.7.1.2 Identificación de los agentes presentes en la situación actual

La organización de los horarios para el estudio queda totalmente a responsabilidad de los estudiantes, la organización de sus horas libres, el establecimiento de prioridades y el cumplimiento de lo estipulado se realiza la mayoría de las veces sin una estructura definida. A continuación, se muestran los agentes presentes en el proceso actual:

Tabla 2. Agentes del proceso actual

Profesores	Estudiantes	Departamentos

Fuente: Autores (2020)

Profesores: Son los agentes encargados de organizar el plan de evaluación de cada asignatura y de impartir los conocimientos necesarios para el cumplimiento de los requisitos de la normativa académica conforme a cada especialidad. También sirven de guía para los estudiantes, al momento de referenciar contenido que les pueda asistir en la materia.

Estudiantes: Son los actores que buscan cumplir con las exigencias establecidas y cumplir con el pensum académico de la especialidad para obtener el título profesional, sin embargo, presentan carencias para organizar su tiempo de estudio y características de la procrastinación que dificultan el enfocarse en las tareas de estudio.

Departamentos: Son los responsables de la planificación académica de las distintas materias que se cursaran a lo largo de las carreras, además del planteamiento del contenido programático. Cada una de estas dispone de un departamento especializado definido por la escuela a la que pertenecen, y con un departamento de cursos básicos.

3.7.1.3 Diagrama de actividades del sistema actual

Figura 2. Diagrama de actividades - sistema actual
Fuente: Autores (2020)

En el sistema actual se puede observar como el estudiante inscribe las materias que va a cursar, luego comienza a asistir a clases hasta que recibe la fecha de presentación de las evaluaciones. No es sino hasta este momento cuando decide

estudiar el contenido de las asignaturas para estas pruebas, finalmente presenta los exámenes en los días previstos.

3.7.1.4 Análisis de los datos recolectados a través de encuestas

La primera fase del proyecto consiste en el análisis y la exploración de la situación actual, conocer los aspectos importantes a tomar en cuenta, como está estructurado el proceso y que variables son relevantes a la investigación. Para obtener datos referentes a los comportamientos de los estudiantes de Ingeniería de Sistemas de la Universidad de Oriente Núcleo de Monagas se optó por realizar una encuesta vía WEB con las siguientes preguntas:

1- ¿Usas libros o recursos online para profundizar lo impartido por el profesor?
2- ¿Cuánto tiempo le dedicas a tus estudios semanalmente?
3- ¿Estudias el contenido impartido de forma constante o solo cuando anunciaron fecha de examen?
4- ¿Con cuánto tiempo de anticipación al examen estudias intensivamente?

Las encuestas impersonales fueron realizadas mediante la página WEB SurveyMonkey, siendo ésta una empresa de Estados Unidos especializada en aportar soluciones de encuestas vía internet. Esta página permite que los usuarios registrados pueden generar encuestas y enviar estas mediante un enlace web a través de correo electrónico o aplicaciones de mensajerías de texto a los diversos encuestados. El proceso de recolección de datos se realizó durante el lapso comprendido entre los días 25/01/2020 al 07/02/2020. Los resultados obtenidos se pueden observar en las siguientes graficas:

Grafico 1. Respuestas graficadas de la encuesta – Primera pregunta
Fuente: Autores (2020)

Esta pregunta se realizó con el objetivo de entender cuál es la importancia de las bibliografías en la vida académica de un estudiante y como estas forman parte esencial para de desarrollo de los mismos. Con las respuestas obtenidas se evidencia como la gran mayoría de los estudiantes encuestados, específicamente el 97% de estos, no se limita solamente a lo que imparte el profesor de la asignatura y busca entender mejor los puntos explicados a través de recursos académicos fuera de las aulas de clases.

Grafico 2. Respuestas graficadas de la encuesta – Segunda pregunta
Fuente: Autores (2020)

Esta pregunta se realizó con el objetivo de obtener una perspectiva general de la cantidad de horas que los estudiantes de Ingeniería de Sistemas aplican a sus estudios semanalmente, de las respuestas obtenidas en esta pregunta se puede destacar que el 54% de los encuestados tienden a estudiar menos de 1 hora diaria/semanal y solo el 26% de los estudiantes que respondieron la encuesta estudian más de 10 horas semanales.

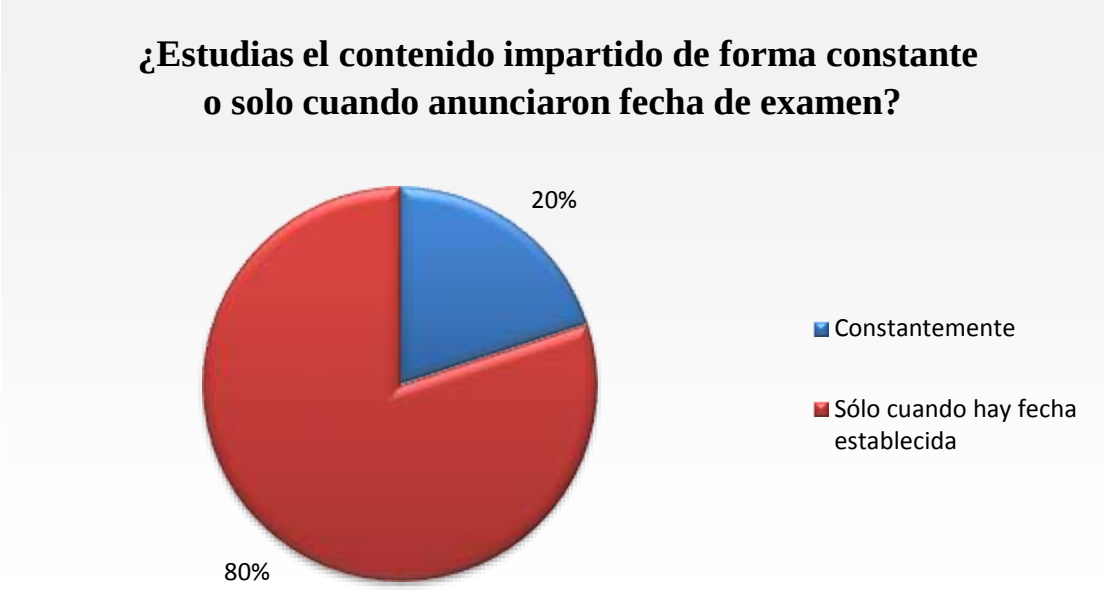


Grafico 3. Respuestas graficadas de la encuesta – Tercera pregunta
Fuentes: Autores (2020)

La realización de esta pregunta se hizo con el objetivo de mostrar las costumbres de estudio de los estudiantes y como estos aplazan estas actividades. De las respuestas obtenidas en esta pregunta se puede destacar que solo el 20% de los estudiantes estudia constantemente durante el transcurso de su periodo académico, siendo este porcentaje muy cercano a los datos obtenidos en la pregunta anterior donde solo el 26% de estudiantes estudian más de 10 horas semanales. Por otra parte, la mayoría de los estudiantes encuestados tienden a comenzar a estudiar solo cuando hay fecha definitiva para presentar un examen.

Grafico 4. Respuestas graficadas de la encuesta – Cuarta pregunta
Fuente: Autores (2020)

Con esta pregunta se logró identificar que el 45% de los estudiantes encuestados comienzan a estudiar de forma rigurosa y constantemente solo a 3 días de presentar el examen, siendo éste un porcentaje bastante importante de las respuestas obtenidas, evidenciando la carencia existente en hábitos de estudios que permitan garantizar el tiempo necesario para asimilar el contenido programático de la asignatura. Por otro lado, el 55% de los estudiantes encuestados estudian con un margen de tiempo mucho mayor.

Los resultados de las encuestas realizadas permitieron obtener información referente a los hábitos de estudio de un grupo de estudiantes de Ingeniería de Sistemas de la Universidad de Oriente – Núcleo de Monagas, esta información sirvió para reflejar un porcentaje importante de estudiantes con carencias de buenos hábitos de estudio, sirviendo este aspecto como un fundamento clave para la elaboración del proyecto. Estos datos obtenidos permitirán optimizar el algoritmo de distribución de

horas de estudios desarrollado para la aplicación, creando un plan de estudio capaz de guiar a los estudiantes con la mejora de sus hábitos académicos, aumentando el número de horas de estudio semanales.

3.7.1.5 Identificación de focos problemáticos

Se pudo identificar los focos problemáticos utilizando las técnicas e instrumentos de recolección de datos, haciendo uso de las encuestas realizadas a los estudiantes de la carrera de ingeniería de sistemas. Con análisis de los resultados obtenidos y la constante comunicación con los estudiantes se observó que dentro de la problemática planteada se evidencia lo siguiente:

1. **Procrastinación:** Se pudo evidenciar en las encuestas como los estudiantes presentan comportamientos relacionados con la procrastinación, debido a que el 80% de los estudiantes encuestados solo decide estudiar una vez conocen la fecha de presentación de las evaluaciones, además el 54% de estos alumnos suelen emplear menos de 6 horas a la semana para estudiar.
2. **Deficiencia para organizar el tiempo:** a pesar de haber avances en las aplicaciones que hoy en día son empleadas, la manera en la que muchos todavía organizan un plan de tareas es algo anticuada, empleando el uso de Microsoft Excel o incluso un calendario de papel. Lo que obstaculiza la creación de una distribución optima del tiempo para dedicar a estudiar el contenido que se plantea en las materias que cursan.
3. **Carencia de motivación en los estudiantes:** debido a factores externos los estudiantes presentan desmotivación, estos pueden ser, problemas personales, baja autoestima, falta de interés y sentido, e incumplimiento de metas. Por consecuencia esto genera una deficiencia en el rendimiento académico que a su vez genera aún más desmotivación en el estudiante.

4. **Poco uso de las nuevas tecnologías para el área académica:** los estudiantes prefieren utilizar los recursos tecnológicos a su disposición para otros propósitos que no involucran el estudio, invirtiendo la mayor parte de su tiempo en actividades de ocio, desaprovechando así beneficios que estas puedan aportar.
5. **Ausencia de un buen hábito de estudio:** La mayoría de los estudiantes no disponen de un buen hábito de estudio desde el inicio de la etapa de educación superior, dificultando su avance en las diversas asignaturas del pensum académico propiciando la manifestación de estrés y un ambiente sub óptimo de estudio.

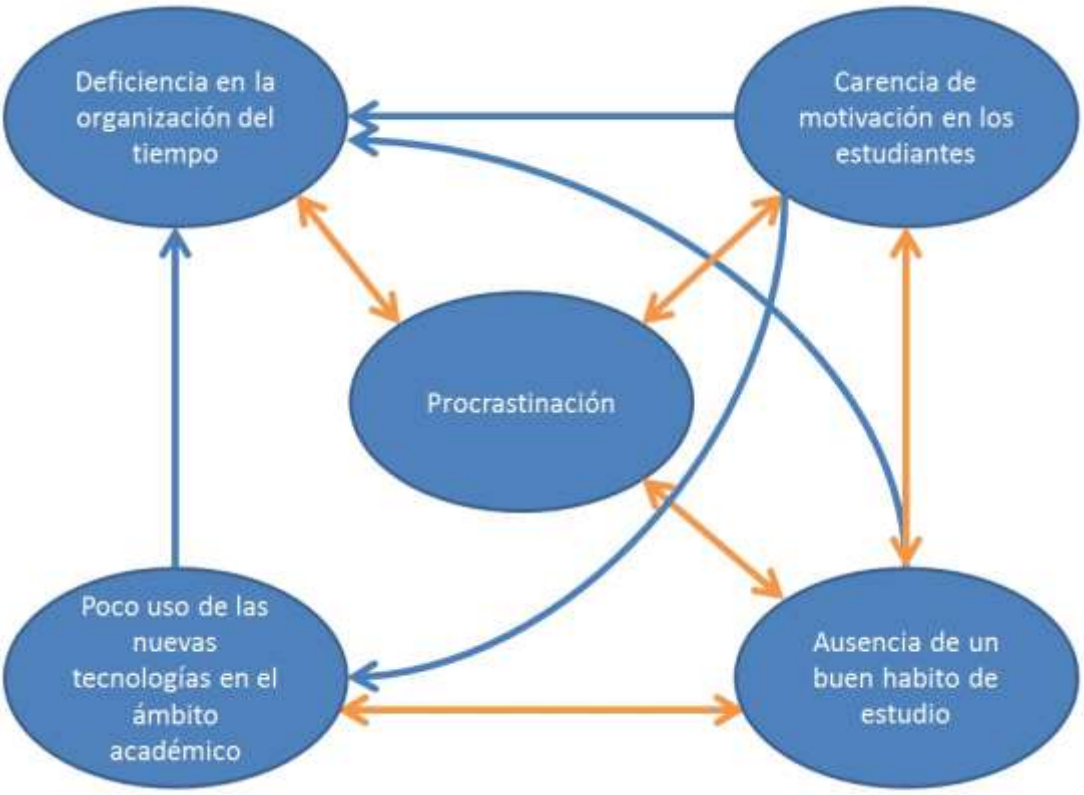


Figura 3. Interconexión de los focos problemáticos
Fuente: Autores (2020)

Posterior la obtención de los focos problemáticos mediante la recolección de datos, se procedió a realizar el diagrama de interconexión entre los diferentes focos representado en la figura 3, permitiendo observar las relaciones que presentan estos y como su existencia conjunta representa la problemática principal a abordar por los investigadores, el análisis de los focos problemáticos permite obtener una visión clara de los aspectos que inciden en la situación actual.

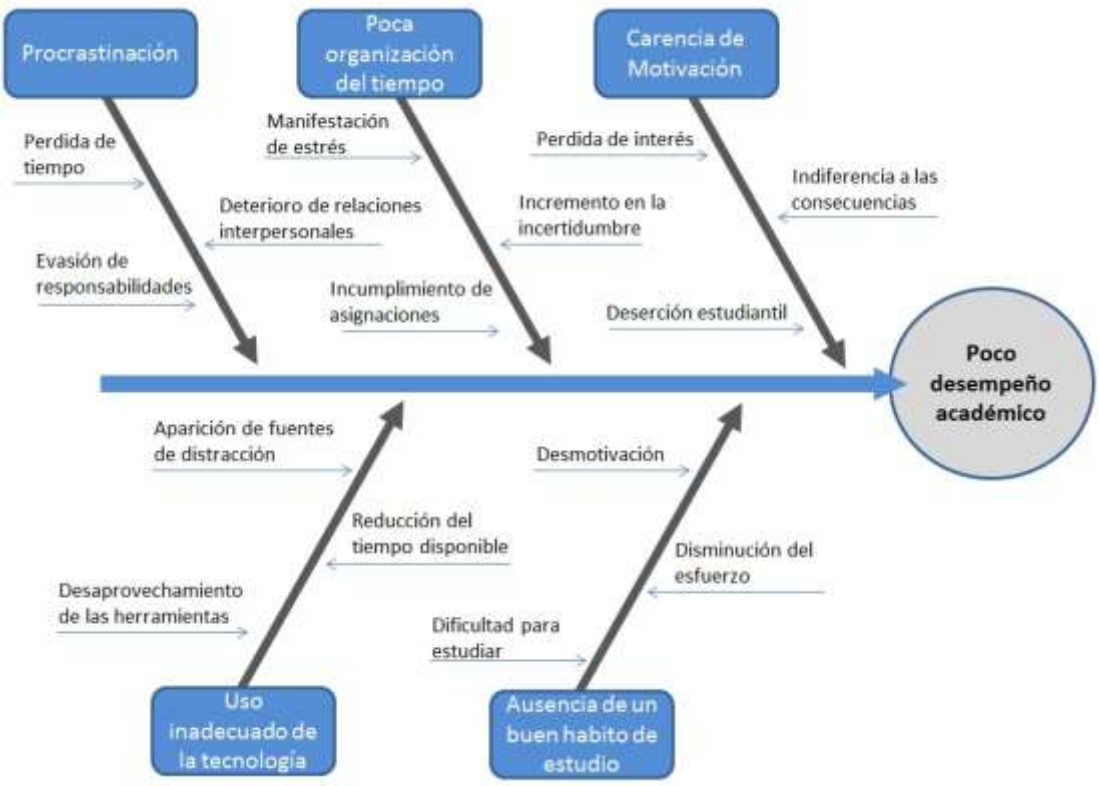


Figura 4. Diagrama de Causa y Efecto (Ishikawa)
Fuente: Autores (2020)

El diagrama de causas y efecto o diagrama de Ishikawa representado en la figura 4, desglosa los diferentes focos problema obtenidos en los análisis previos y ejemplifica las diferentes causas que genera cada foco, donde la unión de estos diversos factores estructura la problemática general presente en la situación actual. El

diagrama de Ishikawa permite desarrollar un análisis con mayor precisión, facilitando la organización de las ideas y la obtención de soluciones.

3.7.2 Análisis de metas y submetas situación actual

3.7.2.1 Razonamiento Metas-Submetas Escenario Actual

El razonamiento de Metas y Submetas permite al individuo organizar cuál será su objetivo principal de largo plazo y los objetivos secundarios que permitirán alcanzar la meta definitiva, tener claro este proceso permite desglosar los aspectos importantes a considerar durante todo el proyecto, en el escenario actual los estudiantes debido a sus experiencias previas analizan las asignaturas inscritas en el periodo académico y eligen cuales son prioritarias sobre otras, otorgándole una mayor importancia y gestionando su tiempo libre para estudiar.

Tabla 3. Razonamiento Metas y Submetas situación actual

Meta	Sub-metas
Aprobar las asignaturas	Inscribir un horario adecuado
	Realizar un plan de estudio acorde
	Utilizar recursos académicos para estudiar
	Aprobar las asignaciones planteadas

Fuente: Autores (2020)

3.7.2.2 Método de reducción de metas escenario actual

El proceso de reducción de metas consiste en ir desglosando las metas superiores en metas inferiores y estas a acciones mecánicas o básicas que permitan cumplir con cada una de las metas establecidas, el proceso de reducción de metas

permite especificar de una mejor manera las acciones necesarias que debe realizar un individuo para llevar a cabo las actividades o acciones que le permitirán ir cumpliendo con las sub-metas y posteriormente su meta actual.

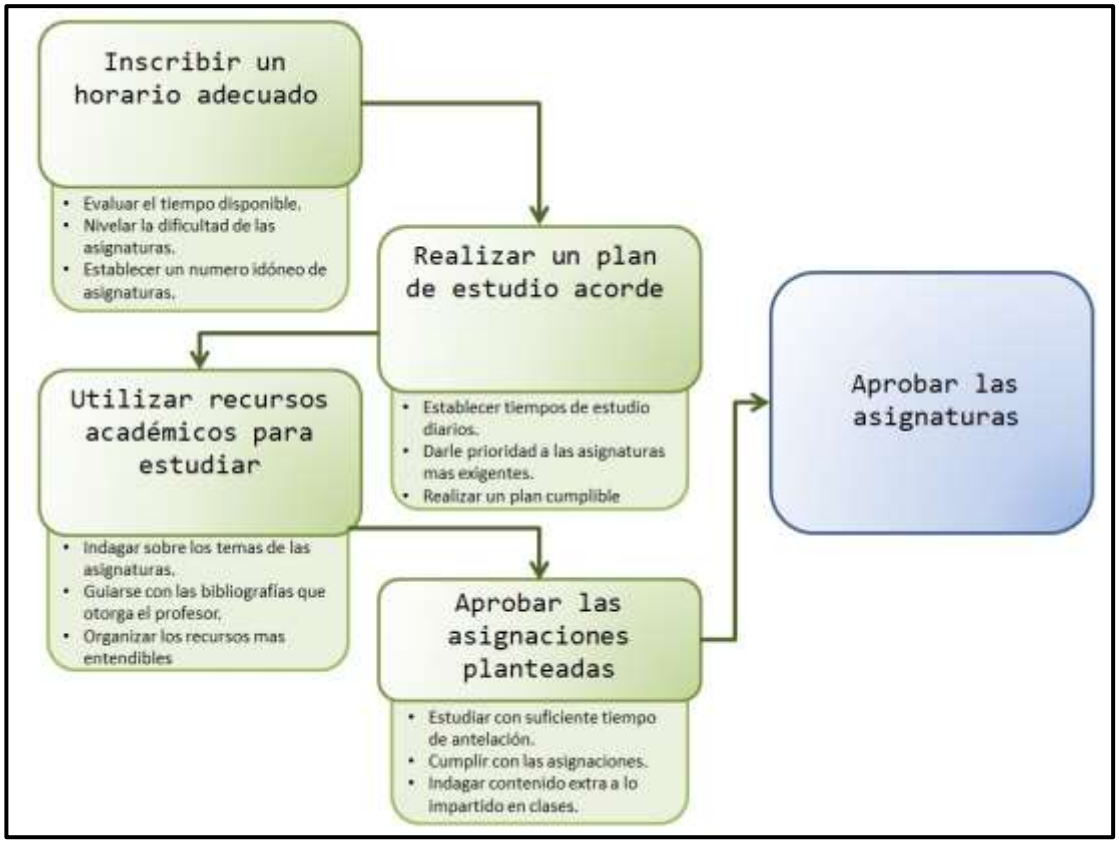


Figura 5. Método de reducción de metas escenario actual
Fuente: Autores (2020)

3.7.3 Análisis conjunto del escenario propuesto

3.7.3.1 Identificación de los agentes presentes en el escenario propuesto

Dentro del proceso propuesto existen diversos agentes que interactúan entre sí con el fin de llevar a cabo la funcionalidad establecida, este segmento de la monografía busca explicar brevemente el rol de cada elemento en la aplicación y por

qué están incorporados en el proceso, a continuación, se muestran los agentes presentes en el sistema:

Tabla 4. Agentes del sistema

Estudiantes	Base de Datos	Aplicación WEB

Fuente: Autores (2020)

Estudiantes: Son aquellos actores finales a los cuales está orientada la funcionalidad de la aplicación, los estudiantes tienen complicaciones a la hora de establecer sus horas de estudios semanales por lo tanto la solución de la aplicación orientada al desarrollo de un plan de estudio acorde a las asignaturas que inscribieron en el periodo académico permitirá solventar este inconveniente y ayudará a mejorar su desempeño académico.

Base de Datos: Es el elemento que permite la permanencia de datos en la aplicación web, se encarga de almacenar la información referente a las asignaturas por cursar junto a sus referencias bibliográficas y los datos de registro de los estudiantes; las credenciales de inicio de sesión (usuario y contraseña), los datos generales del estudiante (nombre y apellido, carrera afín), el almacenamiento de las materias seleccionadas y los horarios generados.

Aplicación Web: Es el segmento central de solución propuesta, consiste en un entorno web que permite generar a los estudiantes un horario de estudio mediante la aplicación de un algoritmo lógico que infiere entre el conjunto de asignaturas inscritas por el estudiante, los tiempos libres disponibles a la semana y las importancias personales establecidas para la dedicación de tiempo de estudio.

3.7.3.2 Razonamiento metas-submetas escenario propuesto

En el sistema propuesto se hizo énfasis en generar una aplicación WEB que permita facilitarle el trabajo al estudiante con el proceso de generación de un horario de estudio y su correlación con los tiempos libres disponibles a la semana, el sistema tomara constara de 3 fases y sub metas establecidas, donde el cumplimiento de la misma genere una aplicación que ayude a facilitar el enfoque del estudiante a lo más importante para con sus estudios y mejore así su rendimiento académico.

Tabla 5. Metas Submetas del escenario propuesto

Meta	Sub-metas
Aumentar el índice aprobatorio mejorando los hábitos de estudio	Analizar las asignaturas inscritas
	Relacionar el conjunto de asignaturas seleccionadas
	Generar un plan de estudio personalizado

Fuente: Autores (2020)

3.7.3.3 Método de reducción de metas escenario propuesto

Analizando la meta general establecida para la aplicación y el conjunto de submetas que permitan llevar a cabo el cumplimiento de esta, se desarrolla el proceso de reducción de metas con el fin de obtener actividades o acciones base que permitan expandir y explicar de una mejor manera el conjunto de elementos que se deben llevar a cabo para poder cumplir con las submetas y posteriormente llegar a la meta final.

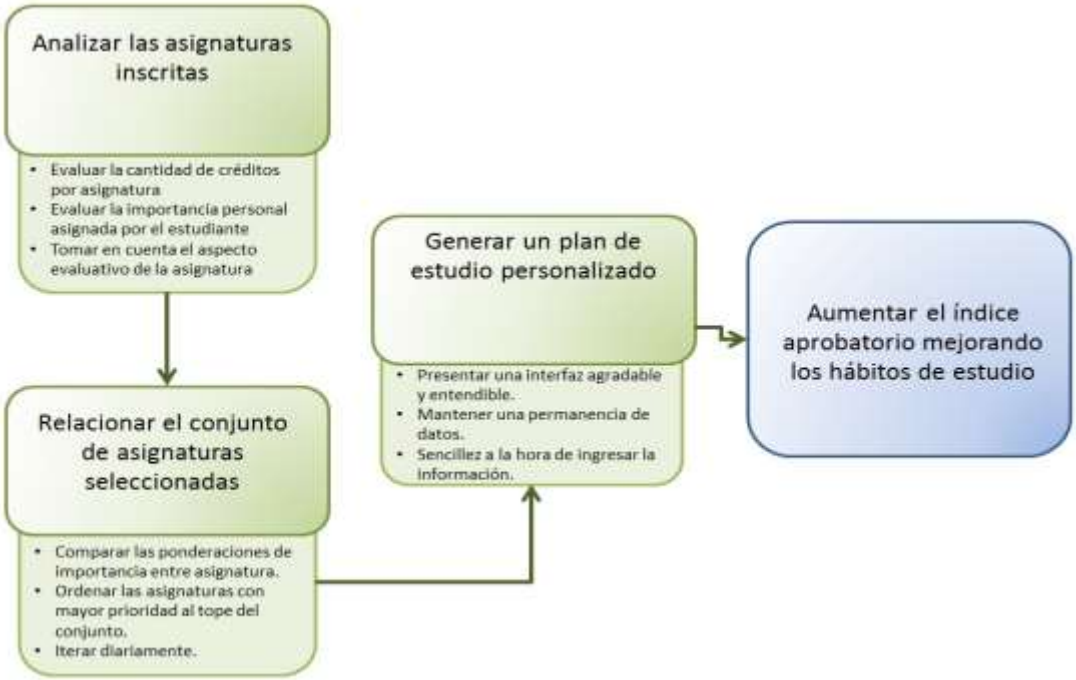


Figura 6. Método de reducción de metas escenario propuesto
Fuente: (Autores 2020)

3.7.3.4 Método de razonamiento hacia atrás y hacia adelante del escenario propuesto

3.7.3.4.1 Razonamiento hacia atrás

Meta: Aumentar el índice aprobatorio mejorando los hábitos de estudio.

Creencia: Si se establece un plan de estudio utilizando una aplicación WEB para su desarrollo, tomando en cuenta las diversas exigencias de cada asignatura y el tiempo disponible del estudiante, se facilitará la gestión del usuario incentivando a la creación de un buen hábito de estudio y posteriormente un aumento en el rendimiento académico.

Conclusión general: Procesando la información suministrada por el estudiante y realizando operaciones lógicas para generar el horario semanal de forma correcta se permitirá crear una aplicación que sirva como herramienta para la gestión del tiempo de los estudiantes.

3.7.3.4.2 Aplicando el razonamiento hacia atrás

Si se cumple con las asignaciones planteadas por el profesor

Entonces se aprobarán las asignaturas inscritas para el periodo académico actual.

Si se aumentan las horas estudiadas por semana

Entonces habrá más probabilidades de aprobar y cumplir con las asignaciones

Si se aplica un horario de estudio establecido de forma semanal

Entonces se aumentará el número de horas estudiadas por semana y se mejorará el hábito de estudio

Si se utiliza la aplicación WEB de forma correcta y se suministran los datos solicitados.

Entonces se generará un horario de estudio semanal personalizado, tomando en cuenta los diversos criterios de evaluación de importancia de cada asignatura.

3.7.3.4.3 Aplicando razonamiento hacia adelante

Reglas:

R1= Ingresar los datos solicitados por la aplicación

R2= Cumplir con el horario generado por la aplicación

R3= Actualizar la información suministrada al sistema durante con el periodo académico.

Hechos: el sistema se comportará como un asistente de estudiantil que facilitará los procesos de análisis y elaboración de un plan de estudio, generando un horario semanal acorde con los datos suministrados y los requerimientos establecidos por el usuario.

Procedimiento de inferencia lógica: Al aplicar las reglas R1, R2 y R3 el sistema dispondrá de la información necesaria para poder evaluar a que asignatura se le debe dar prioridad con respecto a otra y las horas disponibles para la estructuración de un horario de estudio semanal personalizado, la actualización de esta información permitirá que sea dinámico a los eventos que ocurran en el periodo académico y el cumplimiento del mismo garantizara un mejor desempeño del estudiante.

3.7.3.5 Identificación de reglas de condición – acción

3.7.3.5.1 Estudiantes

Si Utilizo la aplicación WEB de forma correcta se me generara un horario de estudio personalizado.

Si no tendré que evaluar directamente según mis creencias cuales son las asignaturas prioritarias y establecer tiempos de estudio semanales

3.7.3.5.2 Base de datos

Si guardo los datos necesarios para el proceso de inferencia de forma correcta se generará un plan de estudio personalizado para cada estudiante y tendrá permanencia en el servidor.

Si no se guardan los datos el proceso de inferencia no será capaz de establecer las variables necesarias para generar la funcionalidad principal.

3.7.3.5.3 Aplicación WEB

Si hago un proceso de inferencia correcto entre las asignaturas inscritas en el periodo académico actual del estudiante y sus horas disponibles a la semana poder desarrollar y mostrar un horario que sirva de herramienta para el usuario final.

Si no se verá afectado el rendimiento académico del estudiante

3.7.4 Requerimientos de la aplicación

3.7.4.1 Historias de Usuario

Junto a las encuestas impersonales realizadas en la primera fase del proyecto, también se utilizó un proceso de la Programación Extrema que es el del desarrollo de historias de usuarios, estas historias representan las ideas del cliente escritas de forma simple y concisa que permitan al desarrollador obtener ideas y los requisitos necesarios que debe incorporar el sistema, para el desarrollo de las historias de usuarios se pidió información constante a tres compañeros de estudio y un profesor que aportaron opiniones sobre las funcionalidades principales y secundarias que debe incorporar el sistema.

Tabla 6. Historia de Usuario 1

Historia de usuario	
Numero: 1	Usuario: Estudiante
Nombre: Trato personalizado para cada estudiante.	Puntos estimados: 3
Prioridad en negocio: Alta	Iteración: 1
Riesgo en desarrollo: Baja	
Descripción: Que pueda tener mi cuenta y se guarde mi información para cuando vuelva a utilizar la aplicación.	

Fuente: Autores (2020)

Tabla 7. Historia de Usuario 2

Historia de usuario	
Numero: 2	Usuario: Estudiante
Nombre: Referencias bibliográficas en las asignaturas	Puntos estimados: 2
Prioridad en negocio: Media	Iteración: 1
Riesgo en desarrollo: Baja	
Descripción: Que las asignaturas puedan tener información bibliográfica acerca de las asignaturas inscritas para el periodo actual.	

Fuente: Autores (2020)

Tabla 8. Historia de Usuario 3

Historia de usuario	
Numero: 3	Usuario: Estudiante
Nombre: Gestión de cuenta	Puntos estimados: 2
Prioridad en negocio: Alta	Iteración: 2
Riesgo en desarrollo: Baja	
Descripción: Poder eliminar mi cuenta o editar mi información	

Fuente: Autores (2020)

Tabla 9. Historia de Usuario 4

Historia de usuario	
Numero: 4	Usuario: Estudiante
Nombre: Selección de asignaturas	Puntos estimados: 3
Prioridad en negocio: Alta	Iteración: 2
Riesgo en desarrollo: Baja	
Descripción: Poder seleccionar las asignaturas y mis datos de tiempos disponibles a la semana de forma fácil.	

Fuente: Autores (2020)

Tabla 10. Historia de Usuario 5

Historia de usuario	
Numero: 5	Usuario: Estudiante
Nombre: Generación de Horario	Puntos estimados: 6
Prioridad en negocio: Alta	Iteración: 2
Riesgo en desarrollo: Medio	
Descripción: Que la información resultante sea de fácil entendimiento y sirva para organizar mis tiempos de estudio.	

Fuente: Autores (2020)

Tabla 11. Historia de Usuario 6

Historia de usuario	
Numero: 6	Usuario: Profesor
Nombre: Personalización posterior en los datos suministrados.	Puntos estimados: 6
Prioridad en negocio: Alta	Iteración: 3
Riesgo en desarrollo: Media	
Descripción: Que los datos suministrados puedan ser cambiados por el usuario durante el transcurso del periodo académico.	

Fuente: Autores (2020)

Tabla 12. Historia de Usuario 7

Historia de usuario	
Numero: 7	Usuario: Estudiante
Nombre: Impresión de la pantalla de inicio junto al resultado del proceso	Puntos estimados: 1
Prioridad en negocio: Media	Iteración: 3
Riesgo en desarrollo: Baja	
Descripción: Que pueda imprimir la información referente al horario generado y las asignaturas que está cursando.	

Fuente: Autores (2020)

Estas historias de usuario sirvieron para organizar las funcionalidades que se incorporaran en el desarrollo de la aplicación y que representan los elementos en concreto que desean de los clientes, los "puntos estimados" expuestos en las tablas están relacionados a un valor arbitrario de complejidad establecido por los desarrolladores, obtenido de la comparación de la historia pivote del proyecto (Historia de usuario 7) siendo esta la historia más sencilla a desarrollar y que sirve como base de medición para evaluar la complejidad de las demás historias.

3.7.4.2 Tareas de ingeniería

Las tareas de ingeniería sirvieron para organizar el conjunto de actividades que se requerían hacer para ir cumpliendo con las historias de usuario establecidas y darle funcionalidad a la aplicación, este conjunto de tareas permite especificar más técnicamente lo descrito de forma general en las historias de usuarios y organizan

mejor la estructura del conjunto de módulos requeridos en el proceso de desarrollo del proyecto.

3.7.4.2.1 Tareas de ingeniería Primera Iteración

Tabla 13. Tarea de ingeniería 1

Tarea de ingeniería	
Número: 1	Número de historia: 1
Nombre de la tarea: **Crear módulo de registro**	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Héctor Mata	
Descripción: Se debe crear un módulo donde se puedan registrar nuevos usuarios.	

Fuente: Autores (2020)

Tabla 14. Tarea de ingeniería 2

Tarea de ingeniería	
Número: 2	Número de historia: 1
Nombre de la tarea: **Creación de la base de datos en MongoDB**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Héctor Mata	
Descripción: Se debe crear las schemas de la base de datos para el registro de usuarios y el enlace en la aplicación.	

Fuente: Autores (2020)

Tabla 15. Tarea de ingeniería 3

Tarea de ingeniería	
Número: 3	Número de historia: 1
Nombre de la tarea: **Diseño de la interfaz de usuario para el registro.**	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Héctor Mata	
Descripción: Se debe crear una interfaz acorde al registro de usuario que sea de fácil entendimiento.	

Fuente: Autores (2020)

Tabla 16. Tarea de ingeniería 4

Tarea de ingeniería	
Número: 4	Número de historia: 1
Nombre de la tarea: **Enlace de la interfaz de usuario con la BD**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Héctor Mata	
Descripción: Se debe enlazar los datos suministrados por el usuario con la base de datos para suministrar la información.	

Fuente: Autores (2020)

Tabla 17. Tarea de ingeniería 5

Tarea de ingeniería	
Número: 5	Número de historia: 2
Nombre de la tarea: **Crear módulo de referencias bibliográficas**	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Santiago Rodríguez	
Descripción: Se debe crear una sección donde se puedan observar la información referente a las asignaturas seleccionadas.	

Fuente: Autores (2020)

Tabla 18. Tarea de ingeniería 6

Tarea de ingeniería	
Número: 6	Número de historia: 2
Nombre de la tarea: **Guardar información bibliográfica en la BD**	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Santiago Rodríguez	
Descripción: Se debe almacenar información bibliográfica de las asignaturas de la carrera en la base de datos.	

Fuente: Autores (2020)

Tabla 19. Tarea de ingeniería 7

Tarea de ingeniería	
Número: 7	Número de historia: 2
Nombre de la tarea: **Diseño de la interfaz de usuario para visualizar las bibliografías.**	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Santiago Rodríguez	
Descripción: Se debe crear una interfaz general para la visualización de las bibliografías pertinentes.	

Fuente: Autores (2020)

Tabla 20. Tarea de ingeniería 8

Tarea de ingeniería	
Número: 8	Número de historia: 2
Nombre de la tarea: **Enlazar interfaz con la base de datos**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Santiago Rodríguez	
Descripción: Se debe enlazar la solicitud de bibliografías referentes a las asignaturas elegidas por el usuario final con la BD.	

Fuente: Autores (2020)

Tabla 21. Conjunto de tareas de ingeniería – primera iteración

Numero de tareas	Numero de Historia	Nombre de la tarea
1	1	Crear módulo de registro
2	1	Creación de la base de datos en MongoDB
3	1	Diseño de la interfaz de usuario para el registro.
4	1	Enlace de la interfaz de usuario con la BD
1	2	Crear módulo de referencias bibliográficas
2	2	Guardar información bibliográfica en la BD
3	2	Diseño de la interfaz de usuario para visualizar las bibliografías.
4	2	Enlazar interfaz con la base de datos

Fuente: Autores (2020)

3.7.4.2.2 Tareas de ingeniería Segunda Iteración

Tabla 22. Tarea de ingeniería 9

Tarea de ingeniería	
Número: 9	Número de historia: 3
Nombre de la tarea: **Crear módulo de configuración de cuenta**	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Santiago Rodríguez	
Descripción: Se debe crear un módulo donde se puedan editar información básica de los usuarios registrados.	

Fuente: Autores (2020)

Tabla 23. Tarea de ingeniería 10

Tarea de ingeniería	
Número: 10	Número de historia: 3
Nombre de la tarea: **Consultar información referente al usuario en la BD**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Santiago Rodríguez	
Descripción: Se debe crear una consulta de datos para obtener la información pertinente del usuario registrado.	

Fuente: Autores (2020)

Tabla 24. Tarea de ingeniería 11

Tarea de ingeniería	
Número: 11	Número de historia: 3
Nombre de la tarea: **Diseño y creación de la interfaz para editar información básica**	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Héctor Mata	
Descripción: Se debe crear una interfaz sencilla donde se pueda visualizar y editar la información básica del usuario registrado.	

Fuente: Autores (2020)

Tabla 25. Tarea de ingeniería 12

Tarea de ingeniería	
Número: 12	Número de historia: 3
Nombre de la tarea: **Validación de los datos suministrados en la BD**	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Héctor Mata	
Descripción: Se debe crear un proceso de validación para saber que los datos suministrados fueron coherentes con los tipos de datos establecidos en los Schemas.	

Fuente: Autores (2020)

Tabla 26. Tarea de ingeniería 13

Tarea de ingeniería	
Número: 13	Número de historia: 4
Nombre de la tarea: **Crear módulo de selección de asignaturas y tiempos libres.**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Santiago Rodríguez	
Descripción: Se debe crear un módulo donde se puedan seleccionar las asignaturas a cursar en el periodo académico actual y las horas disponibles a la semana.	

Fuente: Autores (2020)

Tabla 27. Tarea de ingeniería 14

Tarea de ingeniería	
Número: 14	Número de historia: 4
Nombre de la tarea: **Crear registro de asignaturas en la BD**	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Héctor Mata	
Descripción: Se debe crear una propiedad que permita guardar las asignaturas seleccionadas para cada usuario en la base de datos.	

Fuente: Autores (2020)

Tabla 28. Tarea de ingeniería 15

Tarea de ingeniería	
Número: 15	Número de historia: 4
Nombre de la tarea: **Crear interfaz de selección de asignatura y tiempos libres.**	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Santiago Rodríguez	
Descripción: Se debe crear una interfaz fácil de navegar donde se muestren las asignaturas del pensum académico ordenadas por semestre a cursar y una tabla semanal donde se seleccionen las horas de trabajo, de estudio y las horas disponibles para cada usuario registrado.	

Fuente: Autores (2020)

Tabla 29. Tarea de ingeniería 16

Tarea de ingeniería	
Número: 16	Número de historia: 4
Nombre de la tarea: **Registro y envío de materias seleccionadas a la BD**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Héctor Mata	
Descripción: Se deben registrar las materias seleccionadas en un arreglo y enviarlos a la base de datos en formato JSON para su almacenamiento.	

Fuente: Autores (2020)

Tabla 30. Tarea de ingeniería 17

Tarea de ingeniería	
Número: 17	Número de historia: 5
Nombre de la tarea: **Crear módulo de generación de horario**	
Tipo de tarea: Desarrollo	Puntos estimados: 4
Programador responsable: Santiago Rodríguez	
Descripción: Se debe crear un módulo donde se utilice la información suministrada previamente (Materias seleccionadas, Horas libres) para generar un horario de estudio personalizado.	

Fuente: Autores (2020)

Tabla 31. Tarea de ingeniería 18

Tarea de ingeniería	
Número: 18	Número de historia: 5
Nombre de la tarea: **Crear registro del horario definitivo en la BD**	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Santiago Rodríguez	
Descripción: Se debe crear una propiedad para cada usuario en la base de datos que almacene un arreglo bidimensional con la información definitiva generada por el algoritmo de generación de horario.	

Fuente: Autores (2020)

Tabla 32. Tarea de ingeniería 19

Tarea de ingeniería	
Número: 19	Número de historia: 5
Nombre de la tarea: **Crear interfaz de generación de horario**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Héctor Mata	
Descripción: Se debe crear una interfaz donde se le presente al usuario el horario generado por la aplicación y la información definitiva de cada usuario. (El Dashboard de la aplicación)	

Fuente: Autores (2020)

Tabla 33. Tarea de ingeniería 20

Tarea de ingeniería	
Número: 20	Número de historia: 5
Nombre de la tarea: **Registro y envío del horario generado a la BD**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Santiago Rodríguez	
Descripción: Se debe enviar el arreglo generado a la base de datos para su almacenamiento en formato JSON.	

Fuente: Autores (2020)

Tabla 34. Conjunto de tareas de ingeniería – segunda iteración

Numero de tareas	Numero de Historia	Nombre de la tarea
1	3	Crear módulo de configuración de cuenta
2	3	Consultar información referente al usuario en la BD
3	3	Diseño y creación de la interfaz para editar información básica
4	3	Validación de los datos suministrados en la BD
1	4	Crear módulo de selección de asignaturas y tiempos libres.
2	4	Crear registro de asignaturas en la BD
3	4	Crear interfaz de selección de asignatura y tiempos libres.
4	4	Registro y envío de materias seleccionadas a la BD
1	5	Crear módulo de generación de horario
2	5	Crear registro del horario definitivo en la BD
3	5	Crear interfaz de generación de horario
4	5	Registro y envío del horario generado a la BD

Fuente: Autores (2020)

3.7.4.2.3 Tareas de ingeniería Tercera Iteración

Tabla 35. Tarea de ingeniería 21

Tarea de ingeniería	
Número: 21	Número de historia: 6
Nombre de la tarea: **Crear módulos de edición de importancia y horas libres**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Santiago Rodríguez	
Descripción: Se deben crear módulos para editar la importancia o las horas libres suministradas por el usuario, con el objetivo que estos puedan editar la información durante el transcurso del semestre y recibir nuevos horarios de estudio.	

Fuente: Autores (2020)

Tabla 36. Tarea de ingeniería 22

Tarea de ingeniería	
Número: 22	Número de historia: 6
Nombre de la tarea: **Consulta de asignaturas inscritas y horario definitivo a la base de datos**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Santiago Rodríguez	
Descripción: Se debe hacer una consulta de los elementos suministrados en la base de datos para poder mostrar la información pertinente al usuario.	

Fuente: Autores (2020)

Tabla 37. Tarea de ingeniería 23

Tarea de ingeniería	
Número: 23	Número de historia: 6
Nombre de la tarea: **Diseño de interfaz de edición de importancias y horas libres**	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Héctor Mata	
Descripción: Se deben crear secciones donde se puedan editar las importancias de las asignaturas seleccionadas por el usuario y las horas libres establecidas.	

Fuente: Autores (2020)

Tabla 38. Tarea de ingeniería 24

Tarea de ingeniería	
Número: 24	Número de historia: 6
Nombre de la tarea: **Generar horario nuevo y almacenar en la BD**	
Tipo de tarea: Desarrollo	Puntos estimados: 3
Programador responsable: Santiago Rodríguez	
Descripción: Con la información actualizada se debe generar un nuevo horario y este ser actualizado en la base de datos para su permanencia.	

Fuente: Autores (2020)

Tabla 39. Tarea de ingeniería 25

Tarea de ingeniería	
Número: 25	Número de historia: 7
Nombre de la tarea: **Crear módulo de impresión**	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Héctor Mata	
Descripción: Se debe crear un módulo donde se pueda imprimir los datos representados en el Dashboard para que el usuario final pueda tener una copia física de lo suministrado en la aplicación.	

Fuente: Autores (2020)

Tabla 40. Tarea de ingeniería 26

Tarea de ingeniería	
Número: 26	Número de historia: 7
Nombre de la tarea: **Gestionar los elementos a imprimir**	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Héctor Mata	
Descripción: Se deben establecer propiedades para delimitar que elementos y como se debe imprimir el reporte.	

Fuente: Autores (2020)

Tabla 41. Tarea de ingeniería 27

Tarea de ingeniería	
Número: 27	Número de historia: 7
Nombre de la tarea: **Incorporación en la interfaz del dashboard**	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Héctor Mata	
Descripción: Se debe incorporar una sección que permita al usuario seleccionar la opción de impresión en el dashboard diseñado.	

Fuente: Autores (2020)

Tabla 42. Tarea de ingeniería 28

Tarea de ingeniería	
Número: 28	Número de historia: 7
Nombre de la tarea: **Validación de la impresión**	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Santiago Rodríguez	
Descripción: Se debe validar que la calidad y la posición delos elementos en la impresión permiten que el usuario final pueda observar los diferentes aspectos sin dificultad.	

Fuente: Autores (2020)

Tabla 43. Conjunto de tareas de ingeniería – tercera iteración

Numero de tareas	Numero de Historia	Nombre de la tarea
1	6	Crear módulos de edición de importancia y horas libres
2	6	Consulta de asignaturas inscritas y horario definitivo a la base de datos
3	6	Diseño de interfaz de edición de importancias y horas libres
4	6	Generar horario nuevo y almacenar en la BD
1	7	Crear módulo de impresión
2	7	Gestionar los elementos a imprimir
3	7	Incorporación en la interfaz del dashboard
4	7	Validación de la impresión

Fuente: Autores (2020)

Luego de analizar y establecer el conjunto de actividades necesarias para cumplir con las historias de usuario, elegir la complejidad de cada una de ellas y distribuir el trabajo entre los desarrolladores, se establecieron un conjunto de módulos que representan segmentos de funcionalidad específicos para cada sección de la aplicación, los cuales están descritos a continuación:

a. **Módulo de registro**: En este módulo los estudiantes pueden crearse una cuenta nueva en la aplicación y especificar sus datos personales, nombre de usuario, nombre y apellido, contraseña de la cuenta, y carrera afín.

b. **Módulo de referencias bibliográficas**: Este módulo está incorporado en el Dashboard, permite al estudiante observar bibliografías referentes a las asignaturas del curso, en la etapa final de la aplicación muestra las bibliografías de las asignaturas seleccionadas.
c. **Módulo de Configuración**: Este módulo permite configurar y editar los datos suministrados en el proceso de creación de la cuenta.
d. **Módulo de Selección de asignaturas y tiempos libres**: Este módulo permite la selección de las asignaturas y los tiempos libres de forma inicial. Posterior a la creación de la cuenta el modulo permite establecer los datos necesarios para generar el horario inicial del usuario.
e. **Módulo de Generación de horarios**: Este es el modulo encargado de la funcionalidad principal del sistema, permite evaluar los criterios de las asignaturas seleccionadas por el usuario, las importancias que este le suministro a cada asignatura, el tiempo libre disponible a la semana y generar un horario personalizado.
f. **Modulo para la edición de importancia y horas**: Con este módulo los estudiantes tienen la posibilidad de editar durante el transcurso del periodo académico las horas libres a la semana que establecieron en un principio cuando se crearon la cuenta o la importancia seleccionada, con el objetivo de generar nuevos horarios acordes a las circunstancias.
g. **Módulo de impresión de reporte**: El módulo de impresión está incorporado en el Dashboard y permite generar un registro físico de las diferentes informaciones que se representan en la pantalla de inicio de cada usuario, posterior a la generación del horario de estudio.

3.7.4.3 Plan de Entregas

De acuerdo a las historias de usuarios y a las tareas de ingeniería establecidas se estableció un plan de acción y de entrega acorde a las exigencias de cada historia de

usuario , en las siguientes tablas se pueden observar el conjunto de historias de usuarios que fueron seleccionadas para cada iteración y el tiempo que se estima tome el desarrollo de las mismas, representado por los puntos de la última columna, donde cada 2 puntos representan una semana de desarrollo.

Tabla 44. Plan de entrega 1ra iteración.

1era Iteración. Primera versión del Sistema				
Historias Incluidas	**Nombre**	**Prioridad**	**Riesgos**	**Puntos**
1	Trato personalizado para cada estudiante.	Alta	Baja	3
2	Referencias bibliográficas en las asignaturas	Medio	Bajo	2
Aproximadamente dos semanas y media de desarrollo.				Total: 5

Fuente: Autores (2020)

Tabla 45. Plan de entrega 2da iteración.

2era Iteración. Segunda versión del Sistema				
Historias Incluidas	**Nombre**	**Prioridad**	**Riesgos**	**Puntos**
3	Gestión de cuenta	Alta	Bajo	2
4	Selección de asignaturas	Alta	Bajo	3
5	Generación de Horario	Alta	Medio	6
Aproximadamente cinco semanas y media de desarrollo.				Total: 11

Fuente: Autores (2020)

Tabla 46. Plan de entrega 3ra iteración

3era Iteración. Tercera versión del Sistema				
Historias Incluidas	**Nombre**	**Prioridad**	**Riesgos**	**Puntos**
6	Personalización posterior en los datos suministrados.	Alta	Medio	6
7	Impresión de la pantalla de inicio junto al resultado del proceso	Medio	Bajo	1
Aproximadamente tres semanas y media de desarrollo.				Total: 7

Fuente: Autores (2020)

3.7.4.4 Requerimientos funcionales

La aplicación propuesta debe cumplir con ciertos criterios específicos que permitan desarrollar la función principal establecida, los cuales son los siguientes:

1. El sistema debe garantizar la permanencia de los datos mediante la implementación de una base de datos.
2. Permite a los usuarios la creación, modificación de sus cuentas y la recuperación de contraseñas.
3. Debe desarrollar horarios con información referente a la Universidad de Oriente Núcleo de Monagas
4. Incorporar claves de acceso personalizadas para cada usuario y el resguardo de las mismas.
5. Debe permitir la manipulación de los datos iniciales (importancia personal en las asignaturas, horas libres suministradas) y la actualización de las mismas durante el transcurso del periodo académico, con el fin de permitir generar nuevos horarios acordes a la evolución del semestre.
6. Validar inicio de sesión para usuarios previamente registrados.

3.7.4.5 Requerimientos no funcionales

La aplicación propuesta debe incorporar ciertos requerimientos orientados al usuario final y su utilización que complementan la función principal de la aplicación:

1. La base de datos será creada en MongoDB.
2. La interfaz gráfica debe ser agradable y de fácil entendimiento para el usuario final
3. El proceso de la aplicación debe desarrollarse de forma rápida y sin errores.
4. La paleta de colores y el conjunto de imágenes a usar debe ser referente al tema y al entorno.

3.7.4.6 Requerimientos de software

Los requerimientos de software son el conjunto de herramientas utilizadas al momento de realizar la aplicación, estos programas y tecnologías servirán como base para el desarrollo y la ejecución de la aplicación en el lado del servidor:

1. Sistema operativo: Windows 7 o superior, OS X Yosemite 10.10, Ubuntu 14.04/Debian 8 de 64bits.
2. Base de datos: MongoDB – Mongoose
3. Navegadores web.
4. Entorno de programación: Visual Studio Code, Express, Passport.js, HTML, CSS, JavaScript, Node.js, jQuery, Bootstrap.

3.7.4.7 Requerimientos de hardware

Los requerimientos de hardware son las especificaciones físicas a las cuales deben disponer los dispositivos finales para la ejecución efectiva de la aplicación, como la ésta se ejecuta en un entorno web los requerimientos del sistema no son elevados, cualquier dispositivo con acceso a internet y un navegador web puede acceder a la aplicación, como requerimientos base se tienen:

1. Intel Pentium 4
2. 1Gb de memoria RAM
3. 800Mb de almacenamiento en el disco duro para la instalación de Google Chrome
4. Al menos 1Mbit de velocidad de navegación para el rendimiento fluido.

3.8 SEGUNDA FASE

3.8.1 Diseño base de la aplicación propuesta

3.8.1.1 Arquitectura del sistema

La arquitectura del sistema representa las relaciones que existen entre los diversos elementos del sistema, y como estaría representado el flujo de información entre estos componentes, el siguiente modelo se diseñó con el objetivo de ejemplificar los diversos elementos del sistema y su comunicación:

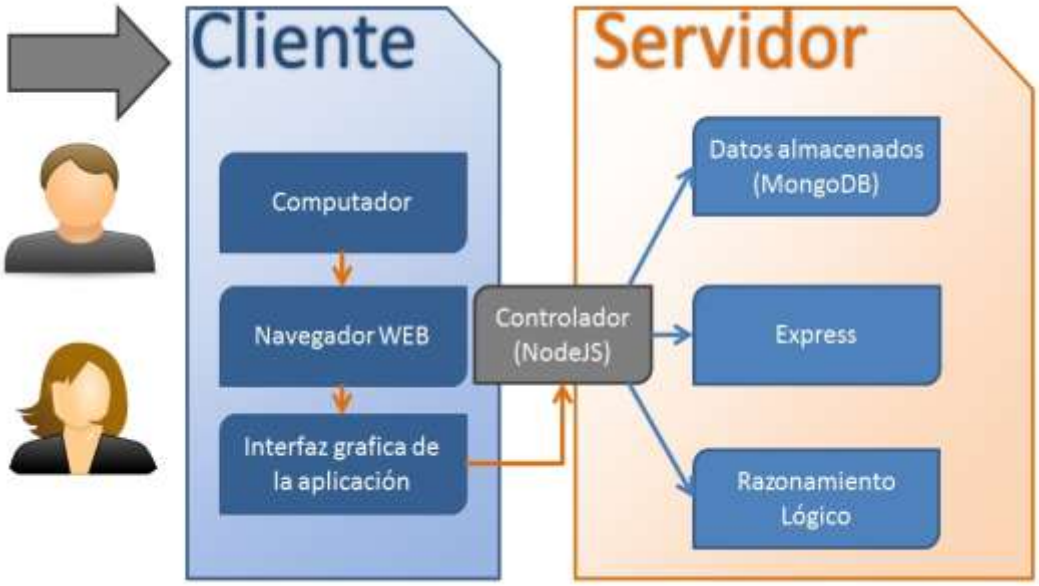


Figura 7. Arquitectura del sistema
Fuente: (Autores 2020)

3.8.1.2 Interconexiones de los agentes presentes en el sistema

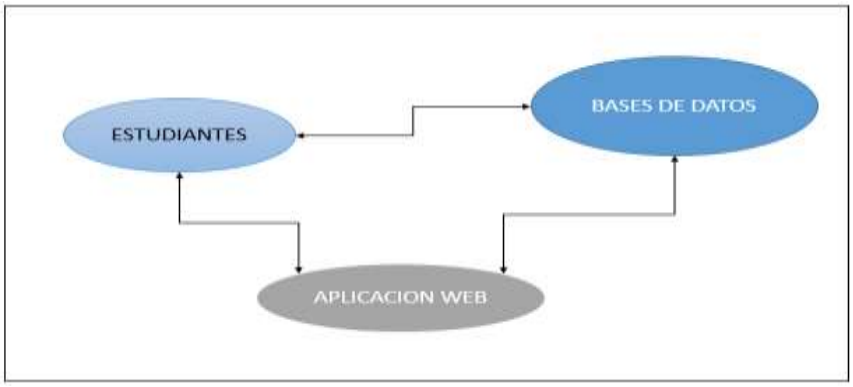


Figura 8. Interconexiones de agentes presentes en el sistema
Fuente: Autores (2020)

En la figura 5 se puede ver claramente la relación entre los agentes, en este caso, los estudiantes, con el resto del sistema de la aplicación web, donde podrán crear sus cuentas usuario y generar sus horarios, y a su vez ambos están relacionados con la base de datos para permitir la permanencia de los datos y almacenar la información de importancia.

3.8.1.3 Explicación de los accesos de usuarios

El inicio del sistema se da en el login, lugar donde el usuario procede a ingresar sus credenciales previamente registradas, ésta información es enviada a la base de datos donde se comparará la información almacenada con la suministrada por el usuario y de ser correcta puede autenticar la sesión, de lo contrario el usuario puede registrarse en el sistema llenando el formulario de registro que le permitirá igualmente acceder al sistema una vez completado el proceso, a continuación, se muestra el índex de la aplicación, si es la primera vez que se entra a la aplicación, comenzara el proceso de generación del horario, aquí debe seleccionar las materias que está cursando, el nivel de importancia que estas tienen para el usuario, las horas

que tendrá disponibles para dedicar a estudiar y posteriormente podrá visualizar su horario, una vez hecho esto puede ingresar al sistema en cualquier momento y revisar la información competente cada vez que quiera.

3.8.2 Casos de uso del sistema

Los casos de uso del sistema son un medio de representar gráficamente el comportamiento del sistema y los agentes que actúan en él, la relación que guardan entre si y cómo interactúan el uno con el otro, por medio de una serie de componentes y figuras establecidos por los estándares de UML, se puede evidenciar el procedimiento. A continuación, se muestran los casos de uso del sistema propuesto:

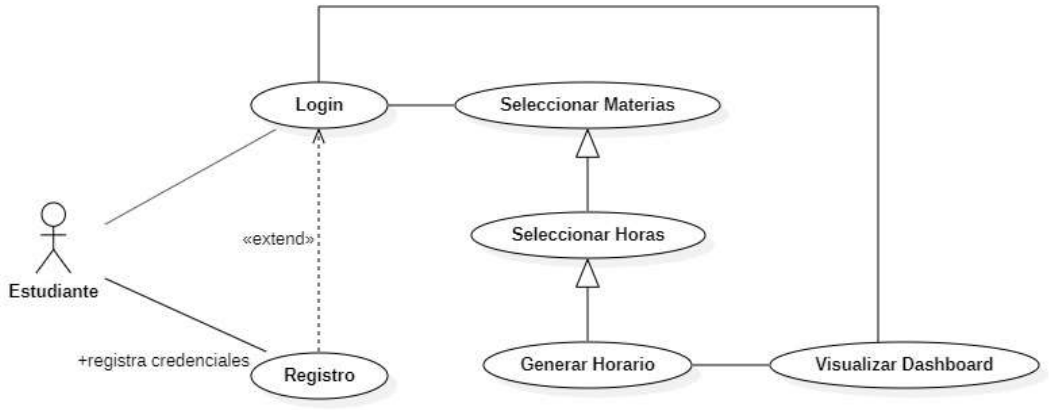


Figura 9. Casos de uso general del sistema
Fuente: Autores (2020)

En el caso de uso presentado se puede apreciar como el usuario interactúa con el sistema, primero registrándose en él por medio de la interfaz de registro, para luego a través de la pantalla de login autenticar sus credenciales y acceder al sistema que generara el horario de estudio a partir de las materias que seleccione el usuario y las horas disponibles que elija en su primer acceso a la aplicación, posteriormente puede visualizar en cualquier momento en su dashboard el plan de estudio creado.

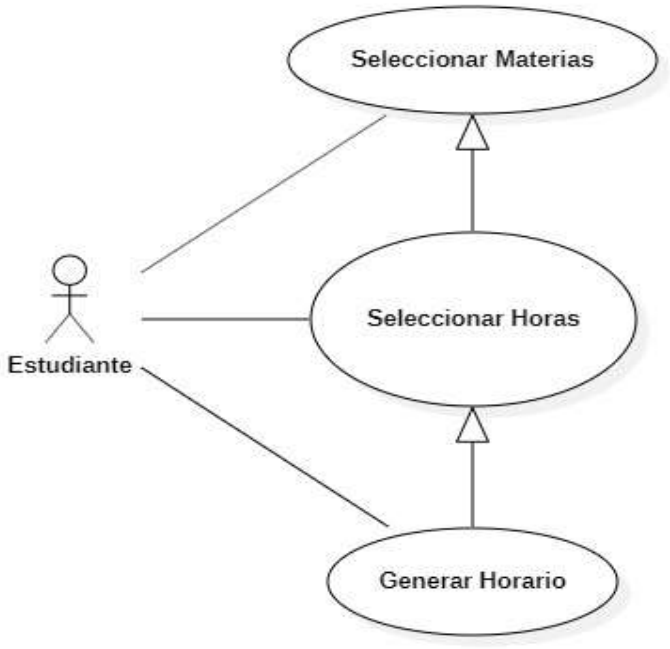


Figura 10. Caso de uso de la generación del horario
Fuente: Autores (2020)

El usuario luego de autenticar sus credenciales, puede generar su horario en el sistema, seleccionando las materias que está cursando, asignarle niveles de importancia, luego seleccionar que horas estarán están disponibles para estudiar y cuales están ocupadas para otras actividades, a continuación, la aplicación se encargara de distribuir de manera eficaz el tiempo adecuado para cada una de las asignaturas seleccionadas.

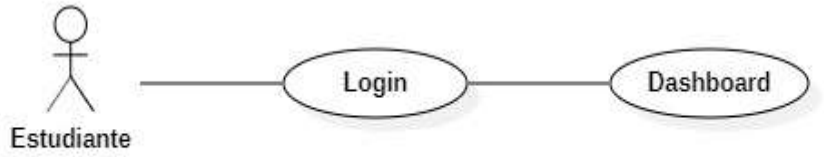


Figura 11. Diagrama de caso de uso – visualizar dashboard
Fuente: Autores (2020)

El usuario una vez haya creado su horario de estudio mediante el proceso de generación de la aplicación, puede ingresar en cualquier momento al sistema nuevamente y será redireccionado automáticamente a su perfil en el dashboard, donde tiene acceso a revisar la distribución de tiempo de las materias que incluyo para el plan de estudio.

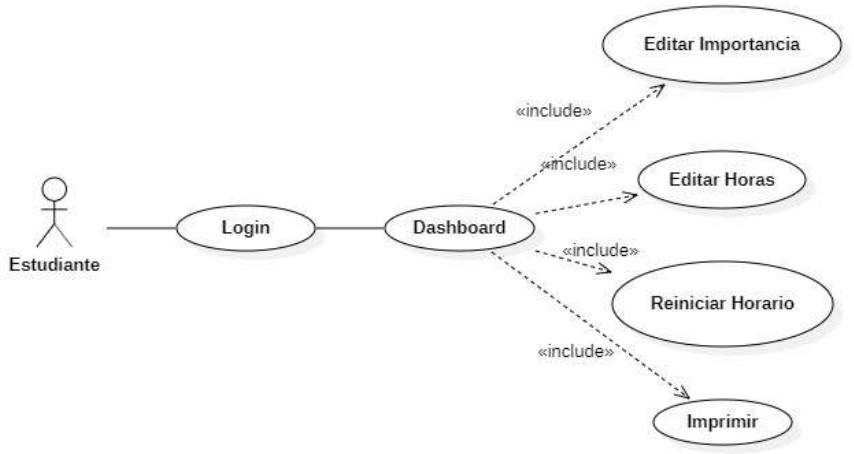


Figura 12. Diagrama de caso de uso dashboard – modificar, editar, reiniciar e imprimir
Fuente: Autores (2020)

El usuario una vez haya creado su horario con la aplicación web puede visualizarlo en cualquier momento accediendo a su cuenta personal del sistema, donde será llevado al dashboard en el cual podrá hacer una serie de cambios si así lo desea, entre ellos puede modificar la importancia de las materias que selecciono previamente, modificar las horas disponibles, imprimirlo si quiere tener una copia física del horario, o reiniciar el proceso y volver a generar un nuevo plan de estudio.

3.8.3 Diagramas de clases

El diagrama de clases representa las funciones y a los objetos que pertenecen al sistema, mostrando que atributos los componen y la relación que guardan estos. Entre

estos se puede apreciar los usuarios del sistema, el horario generado por cada uno y las materias seleccionadas que conforman las carreras y el plan de estudio deseado. A continuación, se muestra un diagrama general de clases:

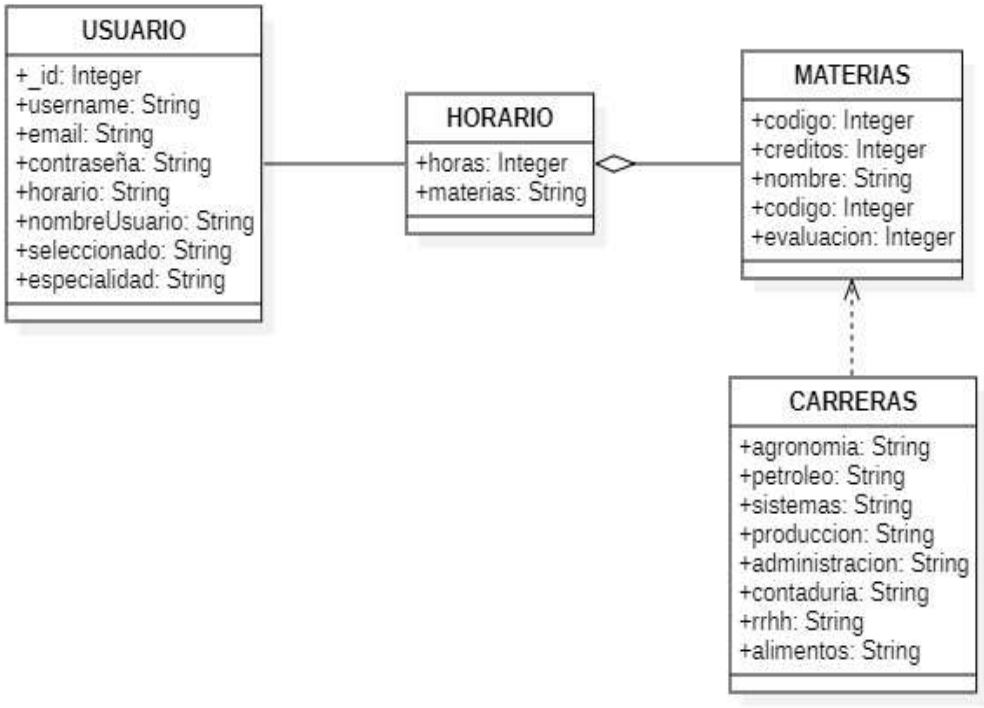


Figura 13. Diagrama general de clases
Fuente: Autores (2020)

3.8.4 Diagramas de secuencia

Estos diagramas son una representación graficas de las acciones que pueden tomar los usuarios en el sistema y permite comprender los pasos a seguir para un funcionamiento adecuado del mismo.

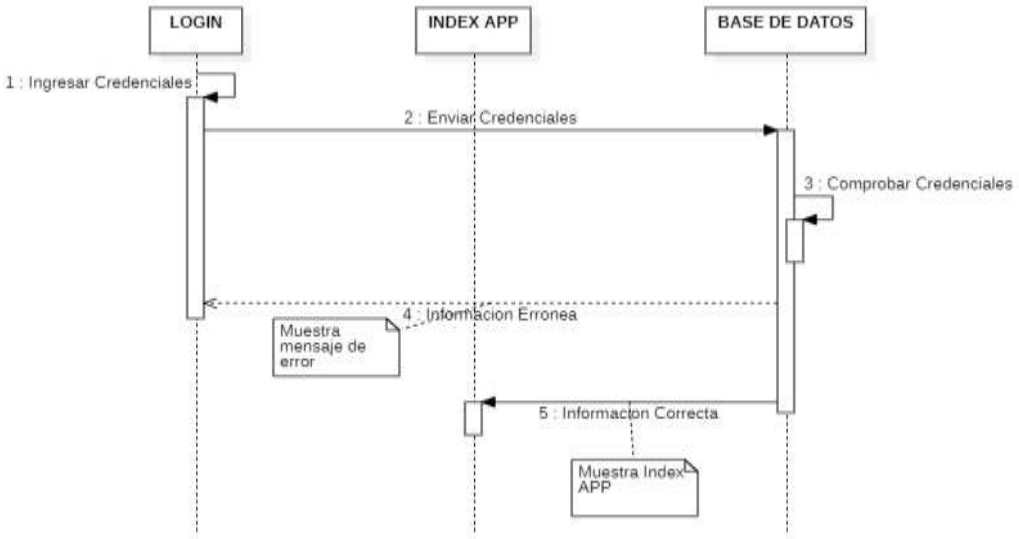


Figura 14. Diagrama de secuencia Inicio de Sesión
Fuente: Autores (2020)

Este diagrama muestra el inicio de sesión del sistema, como el usuario ingresa sus credenciales en el login, el cual se comunica con la base de datos para verificar la información, de ser esta correcta ingresara al índice de la aplicación web, de lo contrario se muestra un mensaje de error y se procede a rectificar la información suministrada.

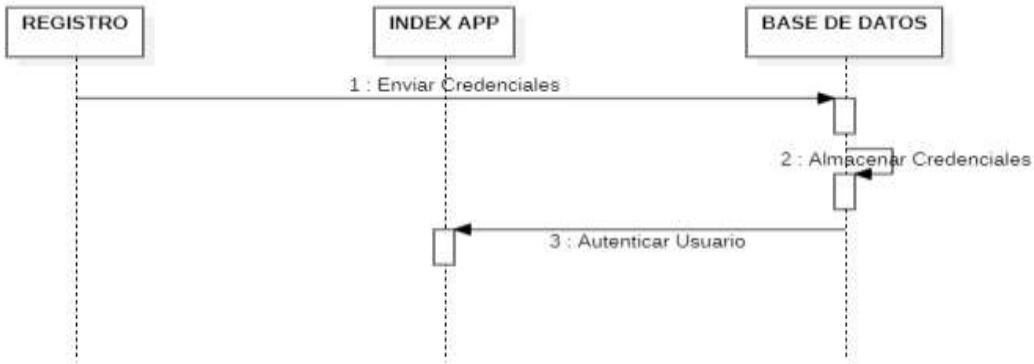


Figura 15. Diagrama de secuencia Registro de Usuario
Fuente: Autores (2020)

Se puede observar en el diagrama como el usuario a través de la pantalla de registro, llena el formulario con sus datos, ingresando sus credenciales en la interfaz de registro, a continuación, son enviados al back-end hasta la base de datos para luego ser almacenados en esta, y posteriormente ser autenticado por el sistema para luego mostrar el índex de la aplicación.

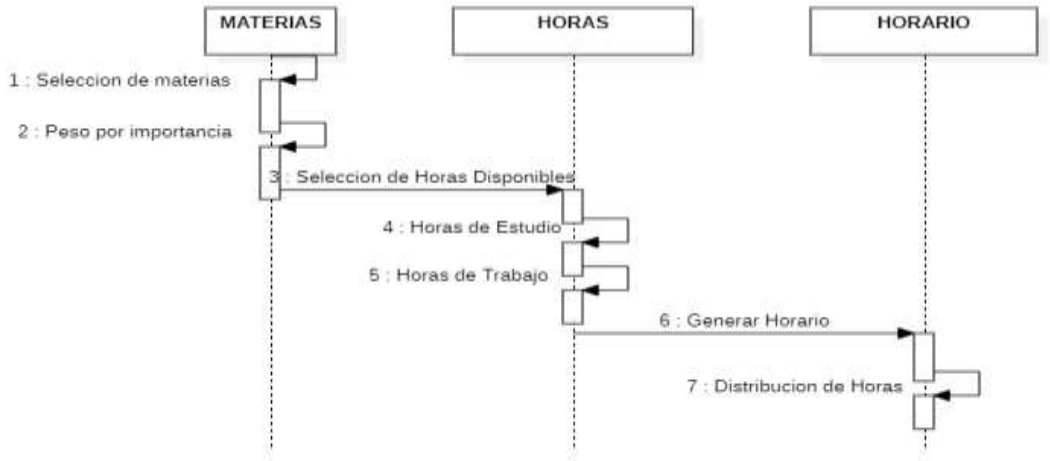


Figura 16. Diagrama de secuencia Generador de Horario
Fuente: Autores (2020)

Este diagrama muestra el proceso para generar el horario de estudio, el usuario luego de ingresar al sistema procede a seleccionar las materias que está cursando actualmente, asignándole un peso de importancia que será tomado en cuenta al momento de distribuir las horas de estudio, luego seleccionara que horas tiene disponibles y cuales no están disponibles, finalmente el sistema se encargar de distribuir de manera efectiva el tiempo libre.

3.8.5 Diagramas de actividades

El diagrama que se presenta a continuación representa las actividades que se pueden realizar en el sistema propuesto, expone los procesos de la aplicación desde el

ingreso a la aplicación si ya se posee una cuenta de usuario o registrar una de no tenerla, empezar la generación del horario seleccionando las materias que se están cursando y las horas que se tienen disponibles, finalmente el usuario puede visualizar su horario una vez creado en cualquier momento desde su dashboard y modificarlo posteriormente si así lo desea.

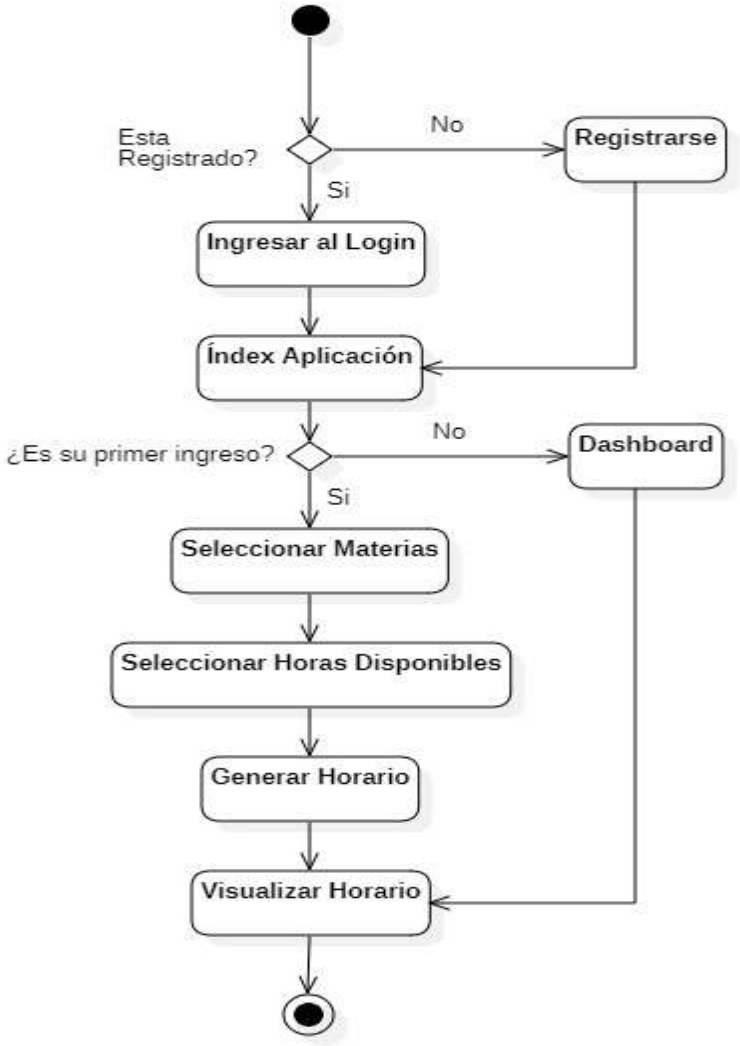


Figura 17. Diagrama de actividades general del sistema
Fuente: Autores (2020)

3.8.6 Determinación del modelo matemático de la aplicación propuesta

Para el desarrollo de la aplicación se optó por usar como guía un modelo de programación por metas, este tipo de modelo matemático de optimización es la técnica apropiada para los análisis de decisión necesarios a la hora de evaluar problemas donde existen múltiples objetivos es conflicto, la programación por metas es una técnica que permite manejar problemas de decisión donde está presente una meta simple con múltiples sub metas o también metas múltiples con varios subtemas. Además, no es necesario que el conjunto de datos a evaluar sea homogéneo entre sí, estos pueden tener diferentes características y no solo se limita a un solo tipo de valor, como ejemplo, en la solución de un problema económico esta variedad estaría representada en el uso de varias monedas; Pesos, Libras, Dólares.

La propuesta del modelo a usar por la aplicación WEB para seleccionar entre las múltiples asignaturas y tomar una decisión con respecto al nivel de relevancia entre las mismas tomo inspiración con el modelo "Weighted Goal Programing" (WGP) que permite otorgarle valores numéricos de importancia personalizados o "pesos" a las diversas variables a considerar con el objetivo de tener una ponderación que le permita al método la efectividad necesaria para discernir entre la importancia de cada una de las decisiones posibles y otorgar una solución idónea al usuario final.

$$z = \sum_{i=1}^{k} (pi \times dpi + qi \times dqi)$$

Para construir un modelo basado en WGP se debe usar una función a minimizar *z*, la cual será la sumatoria de todas las desviaciones en la evaluación, *pi* y *qi* (positivas y negativas) multiplicadas por los respectivos pesos asignados, *dpi* y d*qi*, que para el caso del proyecto sería para cada una de las asignaturas seleccionadas por

el estudiante. En el contexto de la aplicación WEB los conjuntos de materias seleccionadas por el usuario serán almacenadas mediante un arreglo de objetos con propiedades representativas de cada elemento.

Entre las variables a evaluar se tomaron 4 variables para la realización del algoritmo: una variable elegida por el usuario que representa la importancia personal que le asignó a determinada asignatura, el tipo de evaluación de la asignatura si requiere estudio teórico o teórico-práctico, el número de créditos establecidos para esa asignatura por el pensum curricular de la carrera, y una variable de minimización relacionada directamente a las repeticiones de las asignaturas asignadas a cada segmento de hora en la semana. Dada la naturaleza del algoritmo, este se realizará de forma iterativa comparando los valores obtenidos y estableciendo las prioridades entre las asignaturas constantemente hasta finalizar la planificación semanal. El modelo se representa de la siguiente manera:

$$z = 40 + \sum_{i=1}^{k} (p_i \times dp_i) - 10 \times \sum_{j=0} j$$

Donde los valores de k estarán comprendidos entre: $1 \leq k \leq 3$.

Y los pesos asignados a cada variable p*i* serian*;*

Tabla 47. Pesos asignados variables a evaluar.

Variable	Peso
dp_1	15
dp_2	5
dp_3	5

Fuente: Autores (2020)

Estos pesos fueron establecidos por los desarrolladores de la aplicación mediante pruebas del algoritmo y los resultados obtenidos, la ponderación tiene un valor máximo de 100 y un valor mínimo de 0, la multiplicación de estos pesos junto a los valores posibles de las variables a evaluar son los que le dan al algoritmo la capacidad de discernir entre las diversas asignaturas inscritas.

Tabla 48. Valores posibles - variables a evaluar.

Variable	Valores posibles
p_1	-2,-1,0,1,2
p_2	1,2
p_3	1,2,3,4

Fuente: Autores (2020)

Los valores posibles serán registrados durante el proceso de selección de asignaturas, cada asignatura tiene sus propiedades específicas que la diferencian una de otra, relacionándose junto a la variable de selección personal que se le otorga al usuario final para que este pueda personalizar en cierto margen el resultado a obtener en el horario generado, para un caso hipotético donde todas las variables tengan su máximo valor el resultado de la ponderación en la primera iteración sin tomar en

cuenta la variable de repetición sería: 100 {40(peso base) + 2x15 + 2x5 + 4x5} o en el caso que se tengan los valores mínimos, el resultado sería 20 {40(peso base) – 2x15 + 1x5 + 1x5 }

Por ejemplo; para desarrollar un plan de horario de estudio entre dos asignaturas de la carrera de Ingeniería de Sistemas utilizando el algoritmo previamente descrito, donde cuyos criterios a tomar en cuenta están representados por los siguientes valores y el máximo de horas académicas de estudio son de 5 por día (200 minutos), se tiene:

Primera Asignatura:

Tabla 49. Ejemplo modelo matemático 1ra asignatura

Criterio a evaluar	Valor
Importancia personal	0
Tipo de evaluación	1 (solo evaluación teórica)
Número de créditos	3
Variable de repetición	0

Fuente: Autores (2020)

Segunda Asignatura:

Tabla 50. Ejemplo modelo matemático 2da asignatura

Criterio a evaluar	Valor
Importancia personal	2
Tipo de evaluación	2 (teórica-practica)
Número de créditos	4
Variable de repetición	0

Fuente: Autores (2020)

Sustituyendo en la ecuación para la iteración inicial se obtienen estos resultados, que le dan prioridad a la segunda asignatura:

$$z_1 = 40 + \ (0 \times 15) + (1 \times 5) + (3 \times 5) - (10 \times 0) = 60$$

$$z_2 = 40 + \ (2 \times 15) + (2 \times 5) + (4 \times 5) - (10 \times 0) = 100$$

Iteración 2, como ya se incluyó una hora para la segunda asignatura en el cuadro programático semanal, la variable de repetición aumenta, quedando:

$$z_1 = 40 + \ (0 \times 15) + (1 \times 5) + (3 \times 5) - (10 \times 0) = 60$$

$$z_2 = 40 + \ (2 \times 15) + (2 \times 5) + (4 \times 5) - (10 \times 1) = 90$$

Iteración 3:

$$z_1 = 40 + \ (0 \times 15) + (1 \times 5) + (3 \times 5) - (10 \times 0) = 60$$

$$z_2 = 40 + \ (2 \times 15) + (2 \times 5) + (4 \times 5) - (10 \times 2) = 80$$

Iteración 4:

$$z_1 = 40 + \ (0 \times 15) + (1 \times 5) + (3 \times 5) - (10 \times 0) = 60$$

$$z_2 = 40 + \ (2 \times 15) + (2 \times 5) + (4 \times 5) - (10 \times 3) = 70$$

Iteración 5, en este caso se sigue asignando la segunda asignatura debido a que en la comparación uno debe ser menor que otro, al no cumplirse se selecciona la misma asignatura de la iteración anterior:

$$z_1 = 40 + \ (0 \times 15) + (1 \times 5) + (3 \times 5) - (10 \times 0) = 60$$

$$z_2 = 40 + \ (2 \times 15) + (2 \times 5) + (4 \times 5) - (10 \times 4) = 60$$

Segundo día.

Iteración 6:

$$z_1 = 40 + \ (0 \times 15) + (1 \times 5) + (3 \times 5) - (10 \times 0) = 60$$

$$z_2 = 40 + \ (2 \times 15) + (2 \times 5) + (4 \times 5) - (10 \times 5) = 50$$

Iteración 7, se vuelve a seleccionar la primera asignatura:

$$z_1 = 40 + \ (0 \times 15) + (1 \times 5) + (3 \times 5) - (10 \times 1) = 50$$

$$z_2 = 40 + \ (2 \times 15) + (2 \times 5) + (4 \times 5) - (10 \times 5) = 50$$

Iteración 8:

$$z_1 = 40 + \ (0 \times 15) + (1 \times 5) + (3 \times 5) - (10 \times 2) = 40$$

$$z_2 = 40 + \ (2 \times 15) + (2 \times 5) + (4 \times 5) - (10 \times 5) = 50$$

Iteración 9:

$$z_1 = 40 + \ (0 \times 15) + (1 \times 5) + (3 \times 5) - (10 \times 2) = 40$$

$$z_2 = 40 + \ (2 \times 15) + (2 \times 5) + (4 \times 5) - (10 \times 6) = 40$$

Iteración 10:

$$z_1 = 40 + \ (0 \times 15) + (1 \times 5) + (3 \times 5) - (10 \times 2) = 40$$

$$z_2 = 40 + \ (2 \times 15) + (2 \times 5) + (4 \times 5) - (10 \times 7) = 30$$

El algoritmo realizara iteración tras iteración para obtener las asignaturas que incorporara para cada día de la semana, hasta que el valor de z llegue hasta cero, si eso ocurre y aún quedan horas disponibles a la semana, el valor inicial de z es

reiniciado al valor que se calculó en la iteración inicial y continúa el proceso, al finalizar la selección de las asignaturas de la semana, agrupa diariamente por bloques continuos de asignatura y selecciona un espacio de tiempo en el día que le permita al estudiante cumplir con las 5 horas académicas (200 minutos) programadas de una forma continua. El proceso iterativo de la aplicación WEB permitirá desarrollar un plan de estudio acorde a las exigencias de las asignaturas y la importancia personal que el estudiante le asigne a cada una.

3.8.7 Definición autómata de la aplicación propuesta

El autómata de la aplicación es la representación gráfica de los diversos estados que ejecuta la aplicación para llevar a cabo el proceso funcional principal, cada fase estará compuesta por un estado actual y una instrucción que permitirá avanzar por cada estado presente hasta llegar a un nodo terminal donde finaliza el proceso principal de la aplicación:

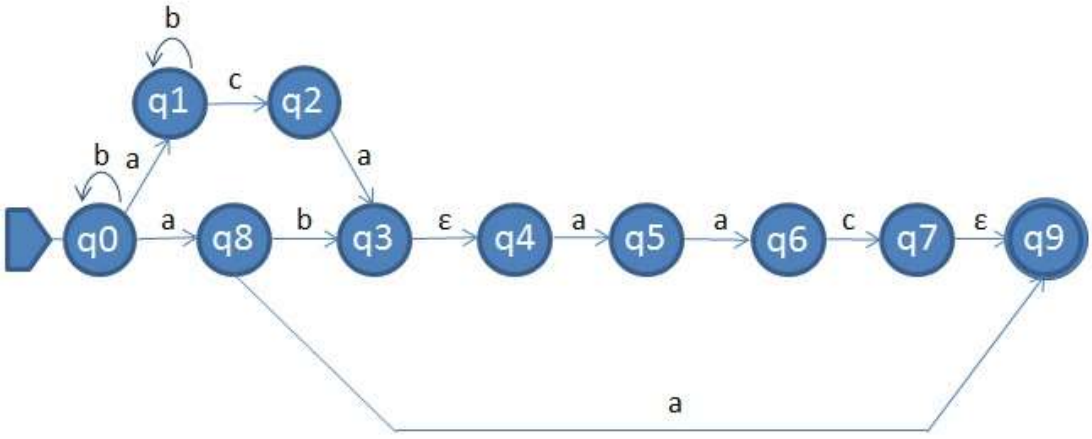


Figura 18. Diagrama de transición del autómata propuesto
Fuente: (Autores 2020)

Tabla 51. Leyenda de estados y entradas del autómata del sistema propuesto

Leyenda de Estados	
Q0	Inicio de la aplicación
Q1	Registro de usuario
Q2	Guardado de credenciales en la base de datos
Q3	Solicitud de asignaturas a la base de datos para la carrera pertinente
Q4	Registrar asignaturas inscritas en el periodo académico
Q5	Registrar tiempos ocupados/libres semanales
Q6	Comparación de criterios y generación del horario de estudio solicitado
Q7	Almacenar horario generado y materias seleccionadas en la base de datos
Q8	Evaluar que se haya establecido un horario en la base de datos
Q9	Mostrar pantalla de inicio con la información pertinente

Leyenda de Entradas	
a	Evaluación satisfactoria
b	Evaluación errónea
c	Registro en la base de datos
ε	Transición vacía

Fuente: Autores (2020)

El estado inicial (Q0) representa el inicio de sesión del sistema, en este estado el estudiante puede dirigirse a la creación de una cuenta (Q1) o ingresar sus credenciales y pasar al estado (Q8) donde se evalúa si ya completó el proceso de generación del horario. De ser satisfactorio este proceso se dirige al estado (Q9) donde se muestra toda la información pertinente, de no ser satisfactorio se continua desde el estado (Q3) que representa el inicio del proceso a aplicar cuando se crea una nueva cuenta en el sistema, la aplicación solicita las asignaturas pertinentes de la carrera del estudiante a la base de datos e informa que se debe seleccionar las asignaturas cursantes de este periodo (Q4), los tiempos libres y las horas ocupadas en

la semana (Q5) para así generar el horario de estudio (Q6), posteriormente el sistema guarda la información en la base de datos (Q7) y dirige al usuario al estado final (Q9).

3.9 TERCERA FASE

3.9.1 Codificación del sistema diseñado

El sistema diseñado fue codificado según los lineamientos expuestos en la fase anterior, creando cada módulo que se requerían para el cumplimiento de los objetivos. Además, se empleó JavaScript como lenguaje de programación usando el entorno de ejecución multiplataforma NodeJS y MongoDB como base de datos, empleando las herramientas que brindan para agilizar la construcción de la aplicación web, así como también, se aplicó el uso del editor de texto Visual Studio Code y el servicio GitHub para llevar un control de versiones del proceso de desarrollo.

En primer lugar, se creó el sistema de autenticación y registro de usuarios, se implementó un protocolo Passport para el login que permitiera a los usuarios tener una cuenta personal en la aplicación, además se incluyó un formulario de registro conectado a una base de datos NoSQL de tecnología Mongo. La autenticación del login y el registro de usuarios tienen comunicación con MongoDB.

Luego se hizo un diseño preliminar de interfaces para la pantalla de login, registro y perfil, así como también para el generador de horarios. Posteriormente se realizó el algoritmo de la aplicación que permita la distribución de las horas de estudio con las materias que curse el alumno, para crear el plan de estudio.

Finalmente se agregaron funciones adiciones al sistema como lo son la habilidad de modificar la importancia de las materias seleccionadas, cambiar las horas disponibles en el horario, imprimirlo en caso de estar satisfecho con el

resultado o reiniciarlo totalmente y hacer uno nuevo desde cero. También se mejoró el diseño de la interfaz para acomodar las nuevas funcionalidades que posee el sistema para mostrar adecuadamente información pertinente a los usuarios y una navegación más intuitiva por el sitio web.

3.9.2 Representación de la aplicación propuesta, diseño de interfaz

A continuación, se presentarán el conjunto de interfaces desarrolladas en base al diseño planteado, estas fueron creadas con el objetivo de establecer un entorno agradable y de fácil manejo mediante la iteración de ideas entre los desarrolladores y estudiantes que sirvieron como fuente de opiniones y críticas constructivas al diseño final de la aplicación, esta sección está dividida entre pantallas junto a una explicación breve de la funcionalidad de cada una.

Figura 19. Login del sistema
Fuente: Autores (2020)

En la figura 19 se muestra la interfaz de inicio de sesión, aquí el usuario puede ingresar sus datos como username y contraseña para autenticarse en el sistema si esta previamente registrado, posteriormente procede al índice donde puede formar su horario o visualizar uno que se haya creado con anterioridad, de lo contrario puede registrarse, presionando el botón de registrarse el cual lo llevara al formulario de registro.

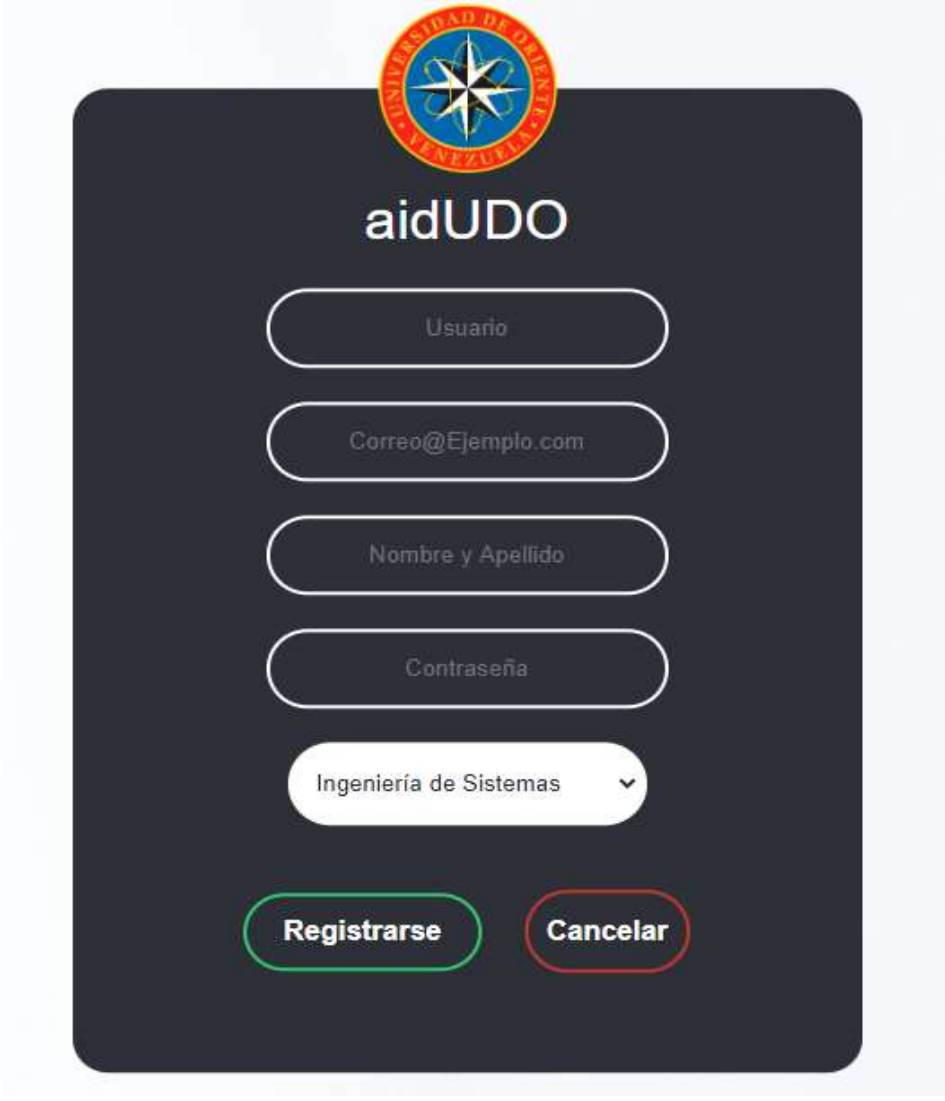


Figura 20. Registro del sistema
Fuente: Autores (2020)

En la figura 20 se muestra el formulario de registro de usuario en el sistema, aquí se debe llenar los campos de username, correo electrónico, nombre y apellido, contraseña y seleccionar que carrera cursa, luego será autenticado por la base de datos y se le mostrará el índice de la aplicación donde podrá generar un horario de estudio por primera vez.

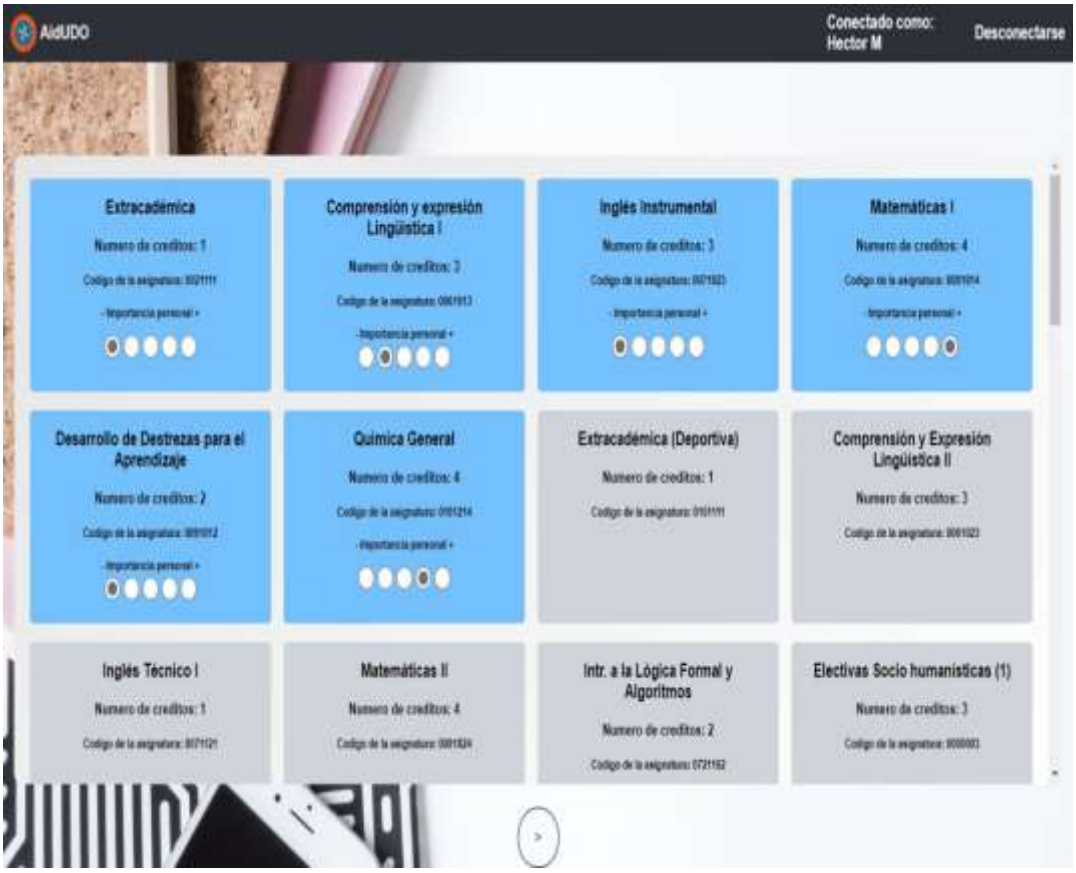


Figura 21. Selección de materias
Fuente: Autores (2020)

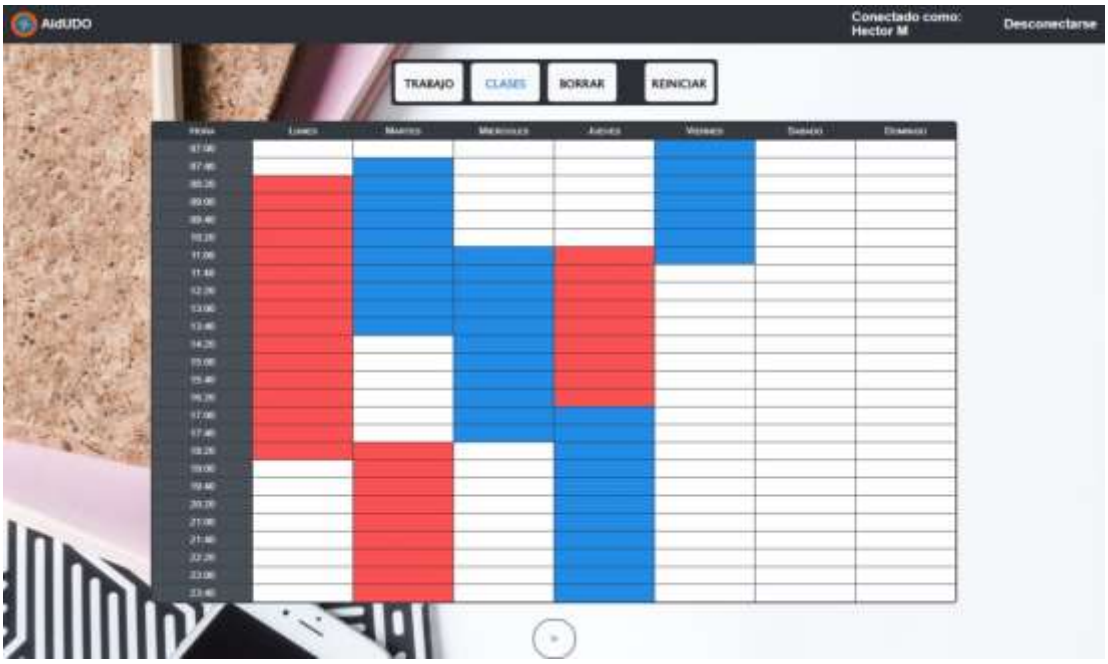


Figura 22. Selección de horas disponibles
Fuente: Autores (2020)

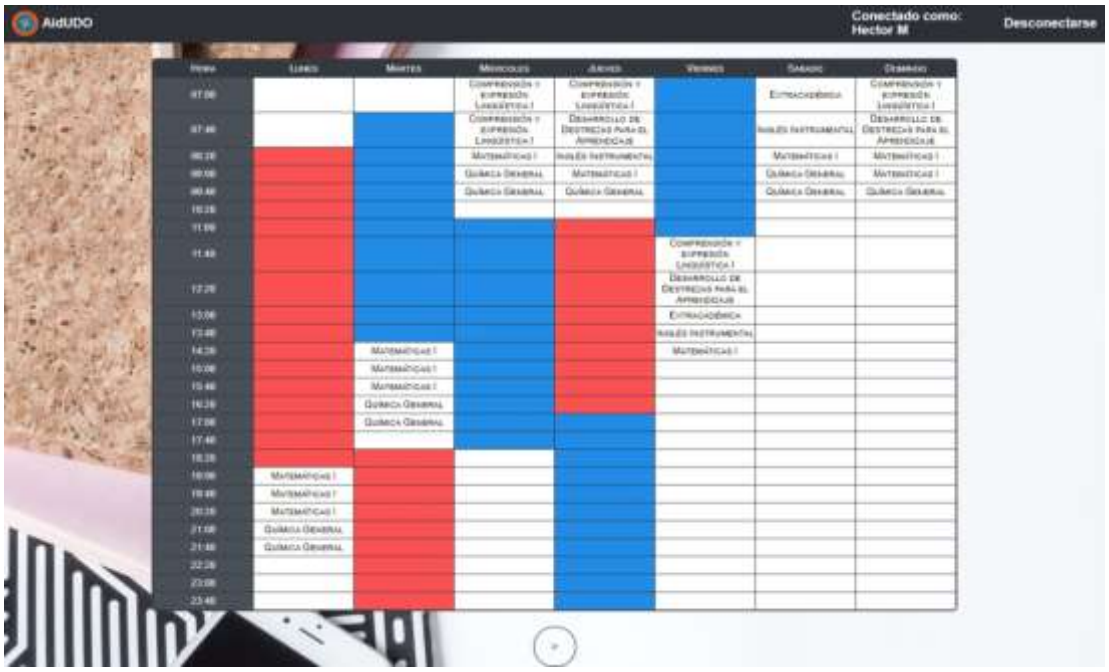


Figura 23. Horario Generado
Fuente: Autores (2020)

En la figura 21 se puede apreciar como el usuario puede seleccionar las materias que está cursando, además de asignarle un nivel de importancia que el sistema tomara en cuenta a la hora de generar el horario, en la figura 22 se seleccionan las horas ocupadas por trabajo (representadas con el color rojo) y las horas a las que se asiste a clases (representadas por el color azul) dejando las horas disponibles para estudio en blanco, por último en la figura 23 se muestra una previsualizacion del horario generado a partir de los parámetros previamente asignados.

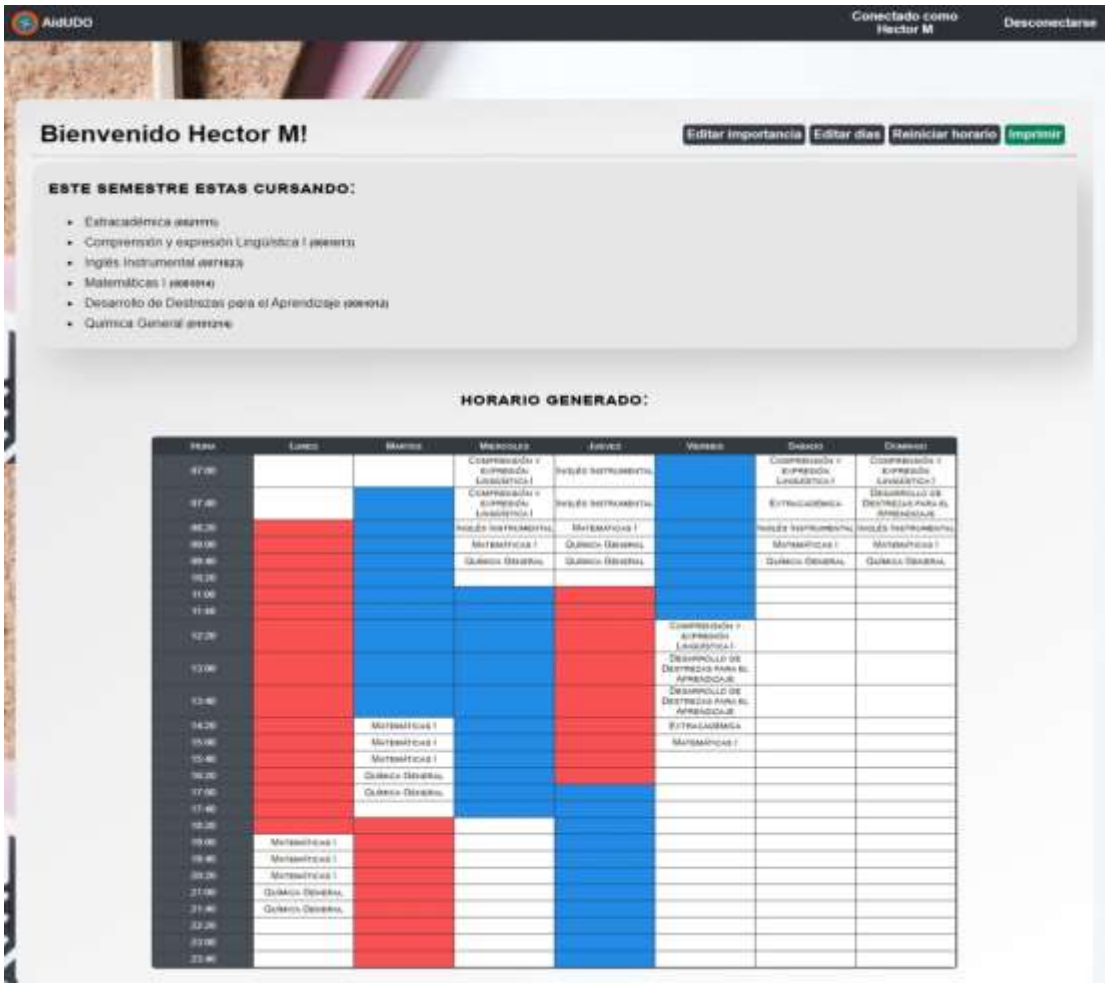


Figura 24. Horario generado por el sistema – perfil de usuario
Fuente: Autores (2020)

La figura 24 muestra el horario generado junto con un grupo de información de importancia para el usuario, aquí se puede visualizar que materias se está cursando actualmente, su código de asignatura, así como también el propio horario con la distribución de tiempo que se seleccionó en los pasos anteriores, además, puede modificar la importancia, cambiar los días y su disposición, o reiniciar totalmente el horario, también imprimirlo una vez satisfecho con el resultado.

3.9.3 Casos de prueba

Las pruebas de aceptación, son pruebas de caja negra que permiten conocer cuando funciona el sistema y tienen como objetivo asegurar que las funcionalidades del mismo cumplen con lo que se espera de ellas. La metodología XP propone el uso de tarjetas de pruebas de aceptación para certificar que la funcionalidad y los requerimientos son satisfactorios.

Tabla 52. Caso de prueba de aceptación 1

<table>
<tr><td colspan="2">Caso de prueba de aceptación</td></tr>
<tr><td>Nombre: Registro de usuarios</td><td>Código: 001</td></tr>
<tr><td colspan="2">Descripción: Verificar el comportamiento de la base de datos al registrar un usuario nuevo.</td></tr>
<tr><td colspan="2">Pre-condiciones:
a) No debe haberse registrado previamente en el sistema.</td></tr>
<tr><td colspan="2">Ejecución de prueba:
1. Ingresar al sistema
2. Presionar “registrarse”
3. Ingresar los datos del formulario de registro
4. Presionar “registrarse”</td></tr>
<tr><td>Resultado esperado:
1. Registro de usuario exitoso
2. Almacenamiento de los datos en la base de datos
3. Redirección al índex de la aplicación</td><td>Evaluación de la prueba: Exitosa</td></tr>
</table>

Fuente: Autores (2020)

Tabla 53. Caso de prueba de aceptación 2

<table>
<tr><td colspan="2">Caso de prueba de aceptación</td></tr>
<tr><td>Nombre: Validación de usuario</td><td>Código: 002</td></tr>
<tr><td colspan="2">Descripción: Verificar el comportamiento del sistema al ingresar un usuario y contraseña válidos.</td></tr>
<tr><td colspan="2">Pre-condiciones:
a) El nombre de usuario y la contraseña deben estar previamente registrados en el sistema.</td></tr>
<tr><td colspan="2">Ejecución de prueba:
1. Ingresar al sistema
2. Ingresar los datos de login, usuario y contraseña
3. Presionar “ingresar”</td></tr>
<tr><td>Resultado esperado:
1. Inicio de sesión exitoso
2. Redirección al índex de la aplicación</td><td>Evaluación de la prueba:
Exitosa</td></tr>
</table>

Fuente: Autores (2020)

Tabla 54. Caso de prueba de aceptación 3

<table>
<tr><td colspan="2">Caso de prueba de aceptación</td></tr>
<tr><td>Nombre: Validación de usuarios</td><td>Código: 003</td></tr>
<tr><td colspan="2">Descripción: Verificar el comportamiento del sistema al ingresar un usuario y contraseña inválidos.</td></tr>
<tr><td colspan="2">Pre-condiciones:
a) El nombre de usuario y la contraseña no deben estar previamente registrados en el sistema.</td></tr>
<tr><td colspan="2">Ejecución de prueba:
1. Ingresar al sistema
2. Ingresar los datos de login, usuario y contraseña
3. Presionar “ingresar”</td></tr>
<tr><td>Resultado esperado:
1. Inicio de sesión fallido
2. Mostrar mensaje de error: “usuario o contraseña invalido”</td><td>Evaluación de la prueba:
Exitosa</td></tr>
</table>

Fuente: Autores (2020)

Tabla 55. Caso de prueba de aceptación 4

Caso de prueba de aceptación	
Nombre: Generar horario	**Código:** 004
Descripción: Verificar el comportamiento del sistema al generar un horario de estudio personalizado.	
Pre-condiciones: a) Tener acceso al Sistema b) Haber validado usuario con anterioridad. c) No debe haberse creado un horario de estudio.	
Ejecución de prueba: 1. Ingresar al sistema 2. Seleccionar las materias cursantes y el nivel de importancia, presionar la flecha que indica siguiente al final de la página. 3. Marcar las horas de trabajo, estudio y disponibles, presionar la flecha que indica siguiente al final de la página. 4. Previsualizar el horario generado, presionar la flecha que indica siguiente al final de la página.	
Resultado esperado: 1. Generación de horario exitosa 2. Redirección a la pantalla dashboard del sistema	**Evaluación de la prueba: Exitosa**

Fuente: Autores (2020)

Tabla 56. Caso de prueba de aceptación 5

Caso de prueba de aceptación	
Nombre: Visualizar horario	**Código:** 005
Descripción: Verificar el comportamiento del sistema para visualizar un horario creado en el perfil de usuario	
Pre-condiciones: a) Tener acceso al sistema. b) Validación exitosa de usuario. c) Haber generado un horario previamente.	
Ejecución de prueba: 1. Ingresar al sistema 2. Ingresar al dashboard del usuario	
Resultado esperado: 1. Se muestra el horario exitosamente	**Evaluación de la prueba: Exitosa**

Fuente: Autores (2020)

Tabla 57. Caso de prueba de aceptación 6

Caso de prueba de aceptación	
Nombre: Reiniciar horario	**Código:** 006
Descripción: Verificar el comportamiento del sistema al reiniciar el horario que ya se había generado.	
Pre-condiciones: a) Tener acceso al sistema. b) Validación exitosa de usuario c) Haber creado un horario previamente	
Ejecución de prueba: 1. Ingresar al sistema 2. Ingresar al dashboard de usuario 3. Pulsar el botón "Reiniciar horario" 4. Repetir el proceso de creación de horario 5. Visualizar el nuevo horario en el dashboard	
Resultado esperado: 1. Reinicio del proceso de generación de horario correcto 2. Visualización exitosa del nuevo horario en el dashboard de usuario	**Evaluación de la prueba: Exitosa**

Fuente: Autores (2020)

Tabla 58. Caso de prueba de aceptación 7

Caso de prueba de aceptación	
Nombre: Imprimir horario	**Código:** 007
Descripción: Verificar el comportamiento del sistema al intentar imprimir un horario	
Pre-condiciones: a) Tener acceso al sistema b) Validación exitosa de usuario c) Haber generado un horario previamente	
Ejecución de prueba: 1. Ingresar al sistema 2. Ir al dashboard de usuario 3. Presionar el botón de imprimir 4. Imprimir el horario con una impresora	
Resultado esperado: 1. Comunicación exitosa con la impresora 2. Impresión del horario	**Evaluación de la prueba: Exitosa**

Fuente: Autores (2020)

CAPITULO IV
CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

Esta monografía de investigación fue realizada como base para la elaboración de una aplicación WEB en la Universidad de Oriente Núcleo de Monagas que sirva como apoyo para la orientación de los estudiantes de Ingeniería de Sistemas con la planificación de sus horarios de estudio y el acceso a referencias bibliográficas de las asignaturas inscritas en el periodo. Posterior a las fases de análisis de la situación, el diseño y el desarrollo de la aplicación se obtuvieron las siguientes conclusiones.

1. El análisis de la situación actual relacionada con el proceso de elaboración de plan de estudio de los estudiantes de Ingeniería de Sistemas de la Universidad de Oriente - Núcleo de Monagas, permitió evidenciar las deficiencias presentes en los estudiantes a la hora de gestionar el tiempo y sus comportamientos relacionados con la procrastinación, sirviendo de razón para el desarrollo de una aplicación WEB que permita generar horarios de estudio personalizados para los estudiantes ayudando a mejorar sus hábitos académicos.
2. La determinación de los requerimientos para la solución propuesta mediante la recolección de historias de usuario y el análisis de los agentes presentes en la aplicación, permitió estructurar los aspectos importantes, las metas a cumplir y las funcionalidades que se van a tomar en cuenta para la realización del proyecto, estableciendo una estimación del tiempo de culminación junto al plan de acción para los desarrolladores.
3. El diseño de la aplicación generó la estructura de los diferentes procesos establecidos en la solución propuesta y como estarán organizados en el producto final, mediante el uso del lenguaje unificado de modelado UML,

herramienta que sirvió para estructurar de forma detallada y explicativa las funcionalidades previamente establecidas y como estarán relacionadas entre sí, para obtener una base general de lo que representara la aplicación, haciendo uso de diagramas de casos de usos, diagrama de clases, de secuencias, y actividades.

4. La aplicación WEB diseñada y desarrollada incorpora los requerimientos propuestos, cumpliendo con la funcionalidad establecida al inicio de la investigación y se establece como herramienta para el apoyo de la comunidad estudiantil, aportando una mayor unión entre el ámbito académico y las tendencias tecnológicas en la actualidad.

4.2 RECOMENDACIONES

Las recomendaciones establecidas en esta monografía de investigación están orientadas a la aplicabilidad del proyecto y el crecimiento del mismo posterior a su implementación. Estas sugerencias servirán para establecer la aplicación WEB de forma idónea y lograr que se cumplan los objetivos planteados mediante una mayor integración con el entorno universitario.

1- Implementar el diseño establecido de la aplicación WEB en la Universidad de Oriente Núcleo de Monagas con el objetivo de otorgar una herramienta idónea a una problemática que no está siendo abordada actualmente en la institución y así disminuir la situación presente con la gestión del tiempo de los estudiantes y su desempeño académico.

2- Desarrollar un estudio más amplio tomando en cuenta las diversas carreras que se dictan en la Universidad de Oriente Núcleo de Monagas con el fin de abarcar la mayor cantidad de estudiantes y que la herramienta esté disponible para toda la comunidad estudiantil, junto a la actualización de las asignaturas de cada carrera y las referencias bibliográficas acordes a los pensum de estudio afines.

3- Desarrollar la migración e integración de la base de datos para que sirva en conjunto con el departamento de control de estudios en la delimitación del acceso de los usuarios que pueden utilizar la aplicación, restringiendo el uso a los estudiantes que estén activos en el periodo académico actual y garantizando una mejor seguridad de los datos almacenados, aprovechando la infraestructura de la universidad.
4- Expandir la aplicación del proyecto a otros núcleos tomando en cuenta sus pensum de estudios de las carreras impartidas e incorporar funcionalidades extras al sistema, con el objetivo de que la aplicación sea utilizada como un asistente estudiantil general, especifico de la Universidad de Oriente a nivel nacional.
5- Implementar campañas de información en la universidad referente a las tecnologías de la actualidad con el fin de apoyar el desarrollo de más soluciones en el entorno y aumentar el alcance de aplicabilidad de los proyectos desarrollados nacionalmente.
6- Para el mantenimiento de la aplicación es necesario disponer de desarrolladores con conocimientos en base de datos, programación orientada a objetos y conocimientos del entorno de desarrollo de la aplicación.
7- El avance tecnológico de la actualidad hace necesario que las aplicaciones y soluciones sean reestructuradas y actualizadas a las últimas tendencias, con el objetivo de que se mantenga vigente aprovechando los beneficios de las futuras tecnologías relacionadas con seguridad, rendimiento y diseño de interfaces.

REFERENCIAS BIBLIOGRAFICAS

Arias, F. (2006). El proyecto de investigación: Introducción a la metodología científica. (5ta ed.). Caracas, Venezuela.

Arias, F. (2014). Apuntes sobre investigación tecnológica. Caracas.

Bautista Q., J. M. (2010). Programación Extrema (XP).

Belloch, C. (2012). Las Tecnologías de la Información y Comunicación en el aprendizaje. Universidad de Valencia, Departamento de Métodos de Investigación y Diagnóstico en Educación. Recuperado el 22 de Enero de 2020, de http://www.uv.es/bellochc/pedagogia/EVA1.pdf

Cegarra Sánchez, J. (2004). Metodología de la investigacion cientifica y tecnologica. Madrid, España.

Chacon, S., & Long, J. (23 de Julio de 2008). git --fast-version-control. Recuperado el 19 de Enero de 2020, de git Web site: https://git-scm.com/

CONDE JAULES, N. (2017). Aplicación web para la gestión académica del Instituto de Educación Superior Tecnológico Público Churcampa, Región Huancavelica, 2016. Ayacucho, Peru.

Darren, G., Sinikka, D., Emory, S., Shannon Lund, G., & Tabitha, P. B. (2008). Time Diary and Questionnaire Assessment of Factors Associated With Academic and Personal Success Among University Undergraduates. Journal of American College Health, 706-715.

Fernández-Valmayor, A., Navarro, A., Fernández-Manjón, B., & Sierra, J. L. (2006). Lenguajes de programación, lenguajes de marcado y modelos hipermedia: una visión interesada de la evolución de los lenguajes informáticos. Estudios de Lingüística del Español (ELiEs), 24.

Fontaines, T. (2012). Metodologia de la investigacion. Caracas, Venezuela.

Friedman, N., Anderson, E., Ballinger, K., Brescia, E., Fuller, T., Heisman, L., . . . Warner, J. (9 de Octubre de 2007). GitHub. Recuperado el 19 de Enero de 2020, de GitHub: https://github.com/

Fuentes Gimón, I. A., & Lagrange Delfino, J. A. (2010). DESARROLLO DE UNA PLATAFORMA WEB QUE INTERCONECTE Y ORGANICE GRUPOS UNIVERSITARIOS PERTENECIENTES AL MOVIMIENTO ESTUDIANTIL. Caracas, Distrito Capital, Venezuela.

Galletta, F. P., & Rios, A. (2017). Plataforma web para publicación y búsqueda de viviendas apoyada en GIS y GP. Caso de desarrollo para Manasés Capriles. Caracas, Distrito Capital, Venezuela.

Garcia, & otros. (2019). Uso del internet como base para el aprendizaje. Revista Atlante: Cuadernos de Educación y Desarrollo.

Grande, M., Cañón, R., & Cantón, I. (2016). TECNOLOGIAS DE LA INFORMACIÓN Y LA COMUNICACIÓN: EVOLUCIÓN DEL CONCEPTO Y CARACTERÍSTICAS. INTERNATIONAL JOURNAL OF EDUCATIONAL RESEACH AND INNOVATION, 218-230.

Haverbeke, M. (2018). Eloquent JavaScript.

Hurtado, J. (2000). Metodologia de la investigacion holística (3ra ed.). Caracas, Venezuela.

Iverson, K. (1967). A Programming Language.

Joskowicz, J. (2008). Reglas y Prácticas en eXtreme Programming.

Kim, K. R., & Seo, E. H. (2015). Kim, K. R., & Seo, E. H. (2015). The relationship between procrastination and academic performance: A meta-analysis. Seoul, Corea.

Loyola Illescas, E., Jara Cobos, R. V., Tobar Pesántez, L., Solano Gallegos, S., Pérez Mejía, A. V., Medina Aguirre, S., . . . Álvarez Rodas, L. (2018). Repitencia estudiantil en la Universidad Politécnica Salesiana. Ecuador.

Mata Bermudez, L. J. (2016). PLAN DE MANEJO ECOTURISTICO PARA LA RUTA CARIPE – CERRO NEGRO, PERTENECIENTE AL MUNICIPIO CARIPE DEL ESTADO MONAGAS, VENEZUELA. Maturín, Monagas, Venezuela.

Meadows, D. H. (2008). Thinkinh in Systems - A Primer.

Merriman, D., Horowitz, E., & Ryan, K. (8 de Julio de 2008). The database for modern applications: MongoDB. Recuperado el 19 de Enero de 2020, de MongoDB Web site: https://www.mongodb.com/

Microsoft. (02 de Mayo de 1991). Microsoft. Recuperado el 19 de Enero de 2020, de Microsoft: https://support.office.com/es-es/article/Conceptos-basicos-sobre-bases-de-datos-a849ac16-07c7-4a31-9948-3c8c94a7c204

Morales, V., Medina Rubio, E., & Alvarez Bedoya, N. (2003). La Educacion Superior en Venezuela. Caracas, Distrito Capital, Venezuela.

Mozilla. (24 de Enero de 1998). MDN web docs. Recuperado el 22 de Enero de 2020, de MDN HTML5: https://developer.mozilla.org/es/docs/HTML/HTML5

Mozilla. (24 de Eneri de 1998). MDN web docs moz://a. Recuperado el 20 de Enero de 2020, de MDN web docs moz://a: https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs/Introduction

Muñoz Martínez, J. D. (14 de Marzo de 2020). Blog | David Muñoz. Obtenido de Blog | David Muñoz: https://scarlex01.wixsite.com/pr-blog-es

Nguyen, D. (2012). Jump Start Node.Js.

Otto, M., & Jacob Thornton. (19 de Agosto de 2011). Bootstrap. Recuperado el 19 de Enero de 2020, de Bootstrap: https://getbootstrap.com/

Pfaffenberger, B., & Karow, B. (2000). HTML 4 Bible, 2nd Edtition.

Quant, D. M., & Sánchez, A. (2012). PROCRASTINACIÓN, PROCRASTINACIÓN ACADÉMICA: CONCEPTO E IMPLICACIONES. Revista Vanguardia Psicológica, 45-59.

Refsnes Data. (21 de Marzo de 2000). W3Schools Online Web Tutorials. Recuperado el 19 de Enero de 2020, de W3Schools: https://www.w3schools.com/

Rodríguez, A., & Clariana, M. (2016). Procrastinación en Estudiantes Universitarios: su Relación con la Edad y el Curso Académico. Revista Colombiana de Psicologia, 45-60.

Sampieri, R., Collado, C., & Lucio, P. (2006). Metodologia de la investigacion (4ta ed.). Mexico DF, Mexico.

Sánchez Hernández, A. M. (2010). Procrastinación académica: un problema en la vida universitaria. Huixquilucan, Mexico.

Si Alhir, S. (2003). Learning UML. Estados Unidos: O'Reilly.

Sole, A. D. (2019). Visual Studio Code Distilled.

StrongLoop, Inc. (15 de Marzo de 2010). Express. Recuperado el 20 de Enero de 2020, de Express - Infraestructura de aplicaciones web Node.js: https://expressjs.com/es/

The jQuery Foundation. (10 de Diciembre de 2005). jQuery. Recuperado el 19 de Enero de 2020, de jQuery: https://jquery.com/

Villota García, S. C., Zamora López, G. G., & Llanga Vargas, E. F. (2019). Uso del internet como base para el aprendizaje. Revista Atlante: Cuadernos de Educación y Desarrollo.

Viñán Villagrán, J. A., Navarrete Chávez, F. F., Puente Riofrio, M. I., Pino Burgos, S. d., & Caicedo Benavides, F. U. (Mayo de 2018). Metodología de la investigación científica como instrumento en la producción y realización de una investigación. Revista Atlante: Cuadernos de Educación y Desarrollo.

Wikipedia. (13 de Enero de 2001). Wikipedia. Recuperado el 19 de Enero de 2020, de La enciclopedia libre: https://es.wikipedia.org/wiki/Aplicacion_web

World Wide Web Consortium. (06 de Julio de 1994). W3C. Recuperado el 22 de Enero de 2020, de W3C CSS: https://www.w3.org/Style/CSS/Overview.en.html

HOJAS METADATOS

Hoja de Metadatos para Tesis y Trabajos de Ascenso - 1/6

Título	**Desarrollo de una aplicación web para la gestión dinámica del tiempo de estudio en la carrera ingeniería de sistemas de la universidad de oriente – núcleo Monagas**

El Título es requerido. El subtítulo o título alternativo es opcional.

Autor(es)

Apellidos y Nombres	**Código CVLAC / e-mail**	
Mata Bermúdez, Héctor Rafael	**CVLAC**	**C.I: 25.273.405**
	e-mail	**hrmb26@gmail.com**
Rodríguez Liscanio, Santiago Rafael	**CVLAC**	**C.I: 26.531.663**
	e-mail	**santiagoliscano@gmail.com**

Se requiere por lo menos los apellidos y nombres de un autor. El formato para escribir los apellidos y nombres es: "Apellido1 InicialApellido2., Nombre1 InicialNombre2". Si el autor está registrado en el sistema CVLAC, se anota el código respectivo (para ciudadanos venezolanos dicho código coincide con el número de la Cedula de Identidad). El campo e-mail es completamente opcional y depende de la voluntad de los autores.

Palabras o frases claves:

aplicación web
procrastinación
horario
planificación
estudiantes
curso especial de grado

El representante de la subcomisión de tesis solicitará a los miembros del jurado la lista de las palabras claves. Deben indicarse por lo menos cuatro (4) palabras clave.

Hoja de Metadatos para Tesis y Trabajos de Ascenso - 2/6

Líneas y sublíneas de investigación:

Área	**Sub-área**
Tecnología y Ciencias Aplicadas	Ingeniería de Sistemas

Debe indicarse por lo menos una línea o área de investigación y por cada área por lo menos un subárea. El representante de la subcomisión solicitará esta información a los miembros del jurado.

Resumen (Abstract):

La presente investigación tuvo como objeto principal el Desarrollo de un plan estratégico para la empresa Grin Center, C.A. Ubicada en la Ciudad de Maturín estado Monagas que optimice los procesos gerenciales. Además de ayudar a proveer y anticiparse a situaciones que puedan afectar la empresa, ya sea de forma positiva o negativa, fomentar la cultura organizacional, mejorar sus habilidades y destrezas en el manejo de planes de negocios de tal forma que se pueden coordinar las rutinas presentes en las distintas áreas de Grin Center, C.A. para visualizar el éxito y los riesgos, y de ayuda para dirigir los procesos administrativos tanto en este negocio como cualquier sucursal que pueda tener en el futuro. Para tal objetivo se realizó un estudio de la situación actual de la organización utilizando la técnica Diagrama Causa-Efecto seguidamente se aplicaron distintas metodologías introducidas en el Análisis Estratégico, basadas en los lineamientos estratégicos de Fred David, Peter Checkland, Humberto Serna Gomez, Thomson y Strickland, Finalmente para la elaboración de los planes de acción a seguir se utilizaron las teorías de George Morrisey con la finalidad de establecer un formato que permita considerar las actividades, responsables, duración, recursos y retroalimentación de cada plan. La investigación se encuentra constituida por cinco capítulos, ubicando a esta en una investigación de tipo proyectiva, basada en un diseño de fuente mixto y de nivel compresivo. Las técnicas e instrumentos de recolección de datos usados fueron la observación directa, entrevistas no estructuradas y la revisión documental. Los resultados obtenidos son los distintos planes de acción a seguir que se generaron para optimizar los procesos gerenciales en Grin Center C.A.

Hoja de Metadatos para Tesis y Trabajos de Ascenso - 3/6

Contribuidores:

Apellidos y Nombres	Código CVLAC / e-mail	
Guevara Cabrera Rommel José	**ROL**	CA ☐ AS ■ TU ☐ JU ☐
	CVLAC	**C.I. 10.306.053**
	e-mail	**rguevara@udo.edu.ve**
Reinoza Henry José	**ROL**	**CA** ☐ **AS** ☐ **TU** ☐ **JU** ■
	CVLAC	**C.I. 8.030.340**
	e-mail	**hjreinoza@gmail.com**
Rodríguez Villasmil Yeisland Carolina	**ROL**	**CA** ☐ **AS** ☐ **TU** ☐ **JU** ■
	CVLAC	**C.I 16.199.486**
	e-mail	**yeisland@udo.edu.ve**

Se requiere por lo menos los apellidos y nombres del tutor y los otros dos (2) jurados. El formato para escribir los apellidos y nombres es: "Apellido1 InicialApellido2., Nombre1 InicialNombre2". Si el autor esta registrado en el sistema CVLAC, se anota el código respectivo (para ciudadanos venezolanos dicho código coincide con el numero de la Cedula de Identidad).. La codificación del Rol es: CA = Coautor, AS = Asesor, TU = Tutor, JU = Jurado.

Fecha de discusión y aprobación:

Año	Mes	Día
2020	10	20

Fecha en formato ISO (AAAA-MM-DD). Ej: 2005-03-18. El dato fecha es requerido.

Lenguaje: spa Requerido. Lenguaje del texto discutido y aprobado, codificado usuando ISO 639-2. El código para español o castellano es spa. El código para ingles en. Si el lenguaje se especifica, se asume que es el inglés (en).

Hoja de Metadatos para Tesis y Trabajos de Ascenso - 4/6

Archivo(s):

Nombre de archivo
NMOCTG_MBHR2020

Caracteres permitidos en los nombres de los archivos: **A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z 0 1 2 3 4 5 6 7 8 9 _ - .**

Alcance:

Espacial: _____ (opcional)
Temporal: _____ (opcional)

Título o Grado asociado con el trabajo:

Ingeniero de Sistemas

Dato requerido. Ejemplo: Licenciado en Matemáticas, Magister Scientiarium en Biología Pesquera, Profesor Asociado, Administrativo III, etc

Nivel Asociado con el trabajo: Ingeniería

Dato requerido. Ejs: Licenciatura, Magister, Doctorado, Post-doctorado, etc.

Área de Estudio:

Tecnología y Ciencias Aplicadas

Usualmente es el nombre del programa o departamento.

Institución(es) que garantiza(n) el Título o grado:

Universidad de Oriente Núcleo Monagas

Si como producto de convenciones, otras instituciones además de la Universidad de Oriente, avalan el título o grado obtenido, el nombre de estas instituciones debe incluirse aquí.

Hoja de metadatos para tesis y trabajos de Ascenso- 5/6

UNIVERSIDAD DE ORIENTE
CONSEJO UNIVERSITARIO
RECTORADO

CU Nº 0975

Cumaná, 04 AGO 2009

Ciudadano
Prof. JESÚS MARTÍNEZ YÉPEZ
Vicerrector Académico
Universidad de Oriente
Su Despacho

Estimado Profesor Martínez:

Cumplo en notificarle que el Consejo Universitario, en Reunión Ordinaria celebrada en Centro de Convenciones de Cantaura, los días 28 y 29 de julio de 2009, conoció el punto de agenda **"SOLICITUD DE AUTORIZACIÓN PARA PUBLICAR TODA LA PRODUCCIÓN INTELECTUAL DE LA UNIVERSIDAD DE ORIENTE EN EL REPOSITORIO INSTITUCIONAL DE LA UDO, SEGÚN VRAC Nº 696/2009"**.

Leído el oficio SIBI – 139/2009 de fecha 09-07-2009, suscrita por el Dr. Abul K. Bashirullah, Director de Bibliotecas, este Cuerpo Colegiado decidió, por unanimidad, autorizar la publicación de toda la producción intelectual de la Universidad de Oriente en el Repositorio en cuestión.

Comunicación que hago a usted a los fines consiguientes.

UNIVERSIDAD DE ORIENTE
SISTEMA DE BIBLIOTECA
RECIBIDO POR
FECHA 5/8/09 HORA 5:20 PM

Cordialmente,

JUAN A. BOLAÑOS CURIEL
Secretario

REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD DE ORIENTE
SECRETARÍA
CONSEJO UNIVERSITARIO

C.C. Rectora, Vicerrectora Administrativa, Decanos de los Núcleos, Coordinador General de Administración, Director de Personal, Dirección de Finanzas, Dirección de Presupuesto, Contraloría Interna, Consultoría Jurídica, Director de Bibliotecas, Dirección de Publicaciones, Dirección de Computación, Coordinación de Teleinformática, Coordinación General de Postgrado.

JABC/YOC/maruja

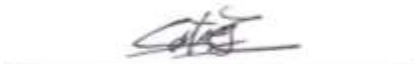
Hoja de metadatos para tesis y trabajos de Ascenso- 6/6

De acuerdo al Artículo 41 del reglamento de Trabajos de Grado:

Los Trabajos de Grado son de la exclusiva propiedad de la Universidad de Oriente, y sólo podrán ser utilizados a otros fines con el consentimiento del Consejo de Núcleo respectivo, quién deberá participarlo previamente al Consejo Universitario, para su autorización.

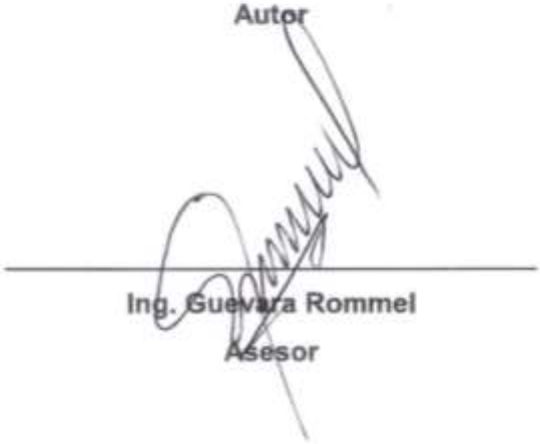
Mata Héctor

Autor



Rodriguez Santiago

Autor



Ing. Guevara Rommel

Asesor