

Universidad Nacional Autónoma de Nicaragua, Managua
Facultad Regional Multidisciplinaria de Chontales
“Cornelio Silva Arguello”
UNAN - Managua / FAREM – Chontales



Seminario de Graduación para optar al Título de
Ingeniero en Sistemas de Información

Tema General

Desarrollo de sistemas web y/o aplicaciones móviles, utilizando arquitectura distribuida.

Subtema

Desarrollo de un sistema web y aplicaciones móviles, para la gestión de horarios de asignaturas, utilizando arquitectura distribuida, en la Universidad Nacional Autónoma de Nicaragua, FAREM – Chontales durante el segundo semestre del año 2016.

Presentado por:

- Br. Jorge Inocente Cruz Serrano.
- Br. Rosa Marina Lumbí Suárez.
- Br. Miguel Ángel Díaz Fernández.

Tutores:

- MSc. Miriam Patricia Téllez Marín.
- MSc. Saira María Urbina Cienfuegos.

Índice general

Carta de aprobación del tutor	1
Dedicatoria	2
Agradecimientos	3
Resumen.....	4
1. Introducción	6
2. Antecedentes	8
3. Definición del problema	9
3.1 Caracterización del problema objeto estudio.....	10
3.2 Delimitación del problema	11
4. Objetivos	12
4.1 General.....	12
4.2 Específicos	12
5. Marco teórico	13
5.1 Sistema distribuido	14
5.1.1 Definición	14
5.1.2 Características de los sistemas distribuidos	14
5.1.3 Arquitectura “Cliente - Servidor”	15
5.1.4 Clasificación de los sistemas “Cliente - Servidor”	16
5.2 Servicios web.....	18
5.2.1 Definición.....	18
5.2.2 Especificaciones que definen el servicio web	18
5.2.3 Componentes de los servicios web.....	19
5.2.4 Tecnologías y estándares de los servicios web.....	19
5.3 Protocolo	20
5.3.1 Definición.....	20
5.3.2 Protocolos utilizados en los sistemas distribuidos	21
5.4 Aplicaciones web	22
5.4.1 Definición.....	22
5.4.2 Características.....	22
5.4.3 Funcionamiento de las aplicaciones web.....	23
5.4.4 Tipo de aplicaciones web	24

5.5	Base de datos.....	26
5.5.1	Definición.....	26
5.5.2	Tipos de base de datos.....	26
5.5.3	Base de datos relacionales.....	27
5.5.3.1	Características de las base de datos relacionales.....	29
5.5.3.2	Ventajas de las bases de datos relacionales.....	29
5.5.3.3	Desventajas de las bases de datos relacionales.....	29
5.5.4	Sistemas gestores de base de datos.....	30
5.5.5	Sistemas de gestores de base de datos más utilizados.....	30
5.6	Aplicación móvil (App).....	32
5.6.1	Tipos de Apps.....	32
5.6.2	Entornos.....	33
5.6.2.1	Entornos de desarrollo android.....	33
5.6.2.2	Android.....	34
5.6.2.3	Entorno de desarrollo iOS.....	34
5.6.2.4	Entorno de desarrollo Windows phone.....	35
6.	Marco Metodológico.....	36
6.1	Sistema de variables.....	37
6.1.2	Hipótesis.....	37
6.1.3	Operacionalización de variables.....	38
6.2	Tipo de investigación.....	41
6.3	Métodos y técnicas de recolección.....	42
6.3.1	Observación.....	42
6.3.2	Encuesta.....	42
6.4	Técnicas de procesamiento y análisis de datos.....	43
7.	Marco administrativo.....	44
7.1	Cronograma de documento.....	45
7.2	Presupuesto de trabajo.....	46
8.	Marco de resultado.....	51
8.1	Cronograma del software.....	52
8.2	Descripción específica de los resultados obtenidos por cada objetivo.....	52
8.2.1	Introducción.....	52
8.2.2	Análisis.....	53

8.2.3	Diseño	53
8.2.4	Implementación.....	55
8.2.5	Evaluación	56
9.	Conclusiones	57
10.	Recomendaciones.....	58
11.	Glosario de términos.....	59
12.	Bibliografía.....	60
13.	Anexos	62
13.1	SRS.....	110
13.2	Manual técnico.....	129
13.3	Manual de usuario	139
13.4	Casos de prueba	156

Índice de tablas

Tabla 1 Primera Operacionalización _____	38
Tabla 2 Segunda Operacionalización _____	39
Tabla 3 Tercera Operacionalización _____	40
Tabla 4 Modelos de desarrollo de software _____	47
Tabla 5 Dimensiones del proyecto _____	48
Tabla 6 Costo del desarrollo _____	49
Tabla 7 Costo del documento _____	50
Tabla 8 Costo total del proyecto _____	50
Tabla 9 Herramientas para el diseño _____	54
Tabla 10 Herramientas para la implementación _____	56
Tabla 11 Palabras desconocidas _____	59

Índice de figuras

Figura 2 Anexo Cronograma de trabajo	1
Figura 3 Anexo Cronograma de software	2
Figura 4 Mapa de navegación web	37
Figura 5 Mapa de navegación móvil	38



Carta de aprobación del tutor

Universidad Nacional Autónoma De Nicaragua, Managua
Facultad Regional Multidisciplinaria de Chontales
“Cornelio Silva Arguello”

2017 “Año de la Universidad Emprendedora”

Departamento de Ciencia, Tecnología y Salud

Valoración Docente

En la medida que el Internet ha ganado ventaja como herramienta de comunicación versátil, la idea de los sistemas informáticos de escritorio evolucionó, dando paso a sitios web que se convierten en sistemas de información útiles, fiables y robustos, que pueden interactuar fácil con aplicaciones móviles a través de protocolos estandarizados para este fin.

Los sistemas web, para uso empresarial e institucional es la mejor opción a elegir al momento de actualización, innovación y crecimiento planificado para estas entidades. El tema “Desarrollo de un sistema web y aplicación móvil, para la **gestión de horarios de asignaturas**, utilizando arquitectura distribuida, en la Universidad Nacional Autónoma de Nicaragua, FAREM – Chontales durante el segundo semestre del año 2016”, se considera pertinente y de requerida aplicación en la Facultad para la optimización de tiempos y recursos destinados a la asignación de horarios, en donde se aprovecha la compatibilidad del sistema web con dispositivos móviles, acceso inmediato y seguridad en la información.

Se considera que los estudiantes **Jorge Inocente Cruz Serrano, Miguel Ángel Díaz Fernández, Rosa Marina Lumbí Suárez** reúnen los requisitos y méritos para la evaluación del jurado examinador en la defensa de trabajo de grado para optar al título de **Ingeniero en Sistemas de Información**.

M.Sc. Miriam Patricia Téllez Marín

M.Sc. Saira María Urbina Cienfuego



Dedicatoria

La presente tesis está dedicada primeramente a Dios, por permitirnos llegar a cumplir unas de nuestras metas. Por los triunfos y los momentos difíciles que nos han enseñado a valorarlo cada día más. A nuestros padres por sus consejos para hacer de nosotros mejores personas, por motivarnos, apoyarnos durante todo el trayecto de preparación y así culminar nuestra carrera. A nuestros profesores que compartieron sus conocimientos durante el desarrollo de nuestra formación profesional, por su tiempo y paciencia a nuestros tutores. A nuestros compañeros, amigos y todas aquellas personas que de una u otra manera han contribuido para el logro de nuestros objetivos.



Agradecimientos

Antes de todo primeramente a nuestro padre celestial por permitirnos hacer realidad uno de nuestros sueños, por darnos las fuerzas, salud, sabiduría por su amor que no tiene fin, nos permite sonreír ante todos nuestros logros que son resultado de su ayuda.

Gracias a nuestros padres que son una bendición y estar presente no solo en esta etapa tan importante de nuestra vida, sino en todo momento ofreciéndonos lo mejor de acuerdo a sus posibilidades.

Agradecemos a nuestros tutores por habernos brindado la oportunidad de recurrir a su capacidad y conocimiento científico, así como también haber tenido toda la paciencia del mundo por guiarnos durante todo el desarrollo de nuestra tesis.

Y para finalizar, también agradecemos a todos los que fueron nuestros compañeros de clase durante todos los niveles de Universidad, ya que gracias al compañerismo, amistad y apoyo moral han aportado en un alto porcentaje a nuestras ganas de seguir adelante en nuestra carrera profesional.



Resumen

El presente documento presenta las técnicas metodológicas, procesos de desarrollo y herramientas de implementación utilizadas para la automatización de los procesos de asignación de horarios de la UNAN FAREM Chontales e igualmente se presenta como una herramienta de comunicación educativa entre educador y educando.

- Parte de los problemas encontrados con el proceso son:
 1. Falta de comunicación entre encargados
 2. Poca o nula organización y comunicación
 3. Creación de horarios de forma manual

Como modelo para el desarrollo de un Software, nuestro proyecto está siguiendo las técnicas que encontramos en el modelo de desarrollo del sistema que sugiere Iam sommerville en el libro de ingeniería de software, encaminamos nuestro proyecto cumpliendo con los cuatro pasos que propone este paradigma: Análisis, diseño, desarrollo y validación.

Para conocer a fondo los factores que influyen el proceso de asignación de horarios se utilizaron técnicas de recolección de información tanto para conocer la naturaleza del problema, si en verdad existe una necesidad automatización de las tareas y las necesidades que debe cumplir el software.

La evaluación del costo de desarrollo del software final se detalla mediante técnicas de evaluación de costos para proyectos de desarrollo de programación, COCOMO es el estándar aplicado.

Para el análisis de los requerimientos funcionales y no funcionales, que son todas las necesidades, roles y característicos que requiere el cliente y los usuarios para el correcto funcionamiento del sistema, se utilizó un documento de especificación de requerimientos (SRS) el cual está basado en el Estándar IEEE 830 – 1998.



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



A través del estándar IEEE830, la recolección de todos los requerimientos funcionales y no funcionales que son las necesidades del cliente y lo que el cliente espera que el software realice.

Mediante las herramientas de trabajo como Visual Studio.Net, SQL Server, entre otras, se logró el desarrollo del software y sus funciones en base a los requisitos solicitados por el cliente.



1. Introducción

El presente documento tiene como objetivo describir las etapas para el desarrollo de un sistema web y aplicaciones móviles, para la gestión de horarios de asignaturas, utilizando arquitectura distribuida, en la Universidad Nacional Autónoma de Nicaragua, FAREM – Chontales cuyas herramientas están orientada a la automatización de asignación de horario, al apoyo y reforzamiento de la comunicación entre educando-educador, utilizando las herramientas y métodos descritos por Ian Sommerville, en su libro de “Ingeniería de Software” apoyado por el estándar IEEE830 para una adecuada especificación de requisitos del software, este documento se rige y estructura según normas de calidad internacionales para el desarrollo de productos de software.

Estas herramientas facilitará esta tarea a cada Docente, Coordinador de carrera, Director de departamento, además se verán beneficiados los estudiantes de la institución, donde podrán saber con anticipación su carga horaria cada semestre; permitiendo a los estudiantes que necesiten inscripción de asignaturas conocer los grupos al que pertenece cada clase y sabrán las posibilidades de cursarla según su carga horaria y los horarios de los grupos.

Como primera instancia, se logra definir y estructurar el problema existente con sus características, conociendo así la naturaleza del problema, se definen los objetivos que el proyecto deberá cumplir para efectuar su cometido de forma eficiente.

El marco teórico presenta los conceptos, definiciones, funciones y el uso de las diferentes tecnologías aplicadas al proyecto, estas seleccionadas convenientemente para lograr el mejor desempeño y fiabilidad de las herramientas desarrolladas.

Mediante el marco metodológico se presentan los datos obtenidos mediante la investigación, datos que apoyan el desarrollo, soportan la fiabilidad y eficiencia del software e igualmente detalla los métodos y herramientas con los cuales se obtuvo la información presentada.



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



El marco administrativo detalla el cronograma y presupuesto requeridos para finalizar el proyecto.

Finalmente, se presentan las conclusiones obtenidas y las recomendaciones que se cree que deben ser tomadas en cuenta para el correcto uso del software y sus componentes.



2. Antecedentes

En el mercado global existen múltiples herramientas tanto propietarias como libres para la gestión de horarios en ámbitos educativos, ejemplo de estas son:

- Agora
- Alexia
- ApliAula
- Aula1
- Clickedu
- DocCF
- Educare
- Esentia
- Gescola

En UNAN MANAGUA como parte de proyecto de maestría se desarrolló un prototipo de un sistema de horario, pero no se implementa debido a que cada región lo elabora de forma diferente y acorde a sus necesidades.

El presente proyecto tiene por antecedente una aplicación “Sistema de horario” diseñada a base de Microsoft Access 2007, inició a utilizarse UNAN FAREM Chontales en el año 2006 y se abandonó entre el año 2012 y 2013, esta estaba diseñada en Macros y su lenguaje de programación era Visual Basic 5.0.



3. Definición del problema

Este centro de estudios inicio sus actividades académicas en el año 1984, bajo la dirección del Instituto Nacional de Administración Pública (INAP); En 1988 el INAP paso a formar parte de las estructuras de la UNAN – Managua, atendido de manera específica por la facultad de Ciencias Económicas. En 1998 se amplió la oferta académica e incremento su matrícula. Se abrieron nuevas carreras profesionales. Por otro lado, se extendieron los servicios a todos los turnos.

En la actualidad la universidad sirve aproximadamente 21 carreras que se ofertan en los diferentes turnos, la matrícula existente es de alrededor de 2300 estudiantes y 120 de profesores.

Dada la gran población universitaria existente de alumnos y docentes, el proceso de asignación efectiva de los horarios para cada año de las diferentes carreras y la distribución de sus docentes en cada asignatura es un proceso que se vuelve más difícil de generar de forma manual y las herramientas genéricas que se obtienen en el mercado, no cumplen con las necesidades que este proceso representa en la universidad.

Por estas razones es que surge la necesidad de diseñar e implementar una aplicación web y móvil que permita la gestión de este proceso de asignación de horarios de la UNAN – FAREM Chontales durante el año 2016.



3.1 Caracterización del problema objeto estudio

Preguntas planteadas:

1. ¿Es necesaria la automatización del proceso?
2. ¿A quiénes beneficiará el desarrollo e implementación del proyecto?
3. ¿Qué características deberá tener un software para automatizar este proceso?
4. ¿Está dispuesto el personal que utilizará esta aplicación?
5. ¿Poseen las habilidades necesarias para utilizarlo?
6. ¿En qué medida se debe implementar la accesibilidad en el software?
7. ¿Qué tipo de plataforma utilizará, Por qué?
8. ¿En dónde se almacenarán los datos?
9. ¿De qué forma se distribuirá la aplicación?
10. ¿Quiénes tendrán acceso al software?



3.2 Delimitación del problema

Debido al incremento de la población universitaria que se presenta cada año, la generación y distribución de horario de asignatura se vuelve más exigente, además los estudiantes tienen que acudir hasta el recinto para darse cuenta de su horario de clases, lo cual es algo que podrían ver desde cualquier lugar con su dispositivo móvil en donde podrán tener acceso a los demás horarios para aquellos que necesiten inscripción de asignatura. Dado que no existe un software que dé respuesta a esta problemática es de mucho interés un sistema web para la parte del docente y la aplicación móvil para los estudiantes que permita la gestión de este proceso.



4. Objetivos

4.1 General

- Desarrollar sistema web y aplicaciones móviles, para la gestión de horarios de asignaturas, utilizando arquitectura distribuida, en la Universidad Nacional Autónoma de Nicaragua, FAREM – Chontales, durante el segundo semestre del año lectivo 2016.

4.2 Específicos

1. Identificar los diferentes requerimientos que se presentan en la generación de horarios, utilizando el estándar de especificación de requisitos de software IEEE 830.
2. Diseñar las interfaces y procesos en base a los requerimientos identificados, aplicando herramientas de diseño.
3. Implementar la codificación del diseño obtenido, a través de lenguajes de programación y tecnología web y móviles, orientada a una arquitectura distribuida.
4. Evaluar el desempeño del sistema web y las aplicaciones móviles, de acuerdo a los requerimientos establecidos.



5. Marco teórico



5.1 Sistema distribuido

Los sistemas distribuidos suponen un paso más en la evolución de los sistemas informáticos, entendidos desde el punto de vista de las necesidades que las aplicaciones plantean y las posibilidades que la tecnología ofrece. Siendo importante proporcionar una definición de sistema distribuido resultará interesante presentar, a través de la evolución histórica, los conceptos que han desembocado en los sistemas distribuidos actuales, caracterizados por la distribución física de los recursos en máquinas interconectadas.

5.1.1 Definición

Sistemas cuyos componentes hardware y software, que están en computadoras conectadas en red, se comunican y coordinan sus acciones mediante el paso de mensajes, para el logro de un objetivo. Se establece la comunicación mediante un protocolo preestablecido. [1]

El uso de este tipo en específico de sistemas permite compartir recursos disponibles en un servidor, a múltiples cantidades de cliente (Siempre que sea permitido), los cuales obtienen información del servidor y conocen la manera de cómo tratarla, dependiendo del tipo de protocolo de comunicación utilizado, se tratan los datos de distintas formas. [2]

5.1.2 Características de los sistemas distribuidos

- **Concurrencia:** Esta característica de los sistemas distribuidos permite que los recursos disponibles en la red puedan ser utilizados simultáneamente por los usuarios y/o agentes que interactúan en la red.
- **Carencia de reloj global:** Las coordinaciones para la transferencia de mensajes entre los diferentes componentes para la realización de una tarea, no tienen una temporización general, está más bien distribuida en los componentes.
- **Fallos independientes de los componentes:** Cada componente del sistema pudiera fallar de manera independientemente, y los demás continuar ejecutando sus acciones.



Esto permite el logro de las tareas con mayor efectividad, pues el sistema en su conjunto continúa trabajando. [1]

Para que un sistema pueda ser calificado como “Sistema distribuido” debe cumplir, sin excepciones, todas las características presentes.

Es fácil concluir las ventajas que presenta este tipo de sistemas, pero, los mismos representan también, una complicidad mayor de desarrollo e implementación, con respecto a los sistemas convencionales. [3]

5.1.3 Arquitectura “Cliente - Servidor”

Se considera un sistema en donde el cliente es una máquina que solicita un determinado servicio y se denomina servidor a la máquina que lo proporciona. Los servicios pueden ser:

- Ejecución de un determinado programa
- Acceso a un determinado banco de información
- Acceso a un dispositivo de hardware

La presencia de un medio físico de comunicación entre las máquinas, es un elemento primordial, y dependerá de la naturaleza de este medio la viabilidad del sistema.

Las arquitecturas Cliente – Servidor más comunes y más utilizadas son:

- Arquitectura Cliente-Servidor de dos capas: Consiste en una capa de presentación y lógica de la aplicación; y la otra de la base de datos. Normalmente esta arquitectura se utiliza en las siguientes situaciones:
 - Cuando se requiera poco procesamiento de datos en la organización
 - Cuando se tiene una base de datos centralizada en un solo servidor
 - Cuando la base de datos es relativamente estática
 - Cuando se requiere un mantenimiento mínimo [1]



- Arquitectura Cliente-Servidor de tres capas: Consiste en una capa de la Presentación, otra capa de la lógica de la aplicación y otra capa de la base de datos. Normalmente esta arquitectura se utiliza en las siguientes situaciones:
 - Cuando se requiera mucho procesamiento de datos en la aplicación.
 - En aplicaciones donde la funcionalidad este en constante cambio.
 - Cuando los procesos no están relativamente muy relacionados con los datos.
 - Cuando se requiera aislar la tecnología de la base de datos para que sea fácil de cambiar.
 - Cuando se requiera separar el código del cliente para que se facilite el mantenimiento.
 - Está muy adecuada para utilizarla con la tecnología orientada a objetos. [1]

5.1.4 Clasificación de los sistemas “Cliente - Servidor”

Según el nivel de abstracción del servicio que ofrecen, los sistemas cliente-servidor se clasifican en:

1. Representación distribuida: La interacción con el usuario se realiza en el servidor, el cliente hace de pasarela entre el usuario y el servidor.
2. Representación Remota: La lógica de la aplicación y la base de datos se encuentran en el servidor. El cliente recibe y formatea los datos para interactuar con el usuario.



3. Lógica Distribuida: El cliente se encarga de la interacción con el usuario y de algunas funciones triviales de la aplicación. Por ejemplo, controles de rango de campos, campos obligatorios, etc. mientras que el resto de la aplicación, junto con la base de datos, están en el servidor.
 4. Gestión Remota de Datos: El cliente realiza la interacción con el usuario y ejecuta la aplicación y el servidor es quien maneja los datos.
 5. Base de Datos Distribuidas: El cliente realiza la interacción con el usuario, ejecuta la aplicación, debe conocer la topología de la red, así como la disposición y ubicación de los datos. Se delega parte de la gestión de la base de datos al cliente.
 6. Cliente servidor a tres niveles: El cliente se encarga de la interacción con el usuario, el servidor de la lógica de aplicación y la base de datos puede estar en otro servidor.
 7. Niveles de una aplicación Web: El nivel de interfaz de usuario está compuesto por las páginas HTML que el usuario solicita a un servidor Web y que visualiza en un cliente Web (normalmente, un navegador). El nivel de lógica de negocio está compuesto por los módulos que implementan la lógica de la aplicación y que se ejecutan en un servidor de aplicaciones. El nivel de datos está compuesto por los datos, normalmente gestionados por un sistema de gestión de bases de datos (servidor de datos), que maneja la aplicación web.
- [4]



5.2 Servicios web

Existen múltiples definiciones acerca de servicios web es por eso que a continuación se muestra un concepto bastante claro. Con sus especificaciones, algunos de sus componentes, tecnologías y estándares de los servicios web.

5.2.1 Definición

Un servicio web expone un conjunto de servicios para ser consumidos a través de la red. En otras palabras, un servicio web especifica un conjunto de operación (funciones que retornan determinado valor, reciben un conjunto finito de parámetros, y retorna un resultado), a través de una URL, donde una aplicación Cliente remota los puede consumir (podría haber cuestiones de seguridad en el medio).

Cuando se expone un servicio web, se publica un archivo wsdl en el servidor web, donde se muestran esas operaciones, parámetros, tipos de retorno, dirección para invocar el servicio, etc. Existe otro enfoque para el diseño de web service, denominado Restful, donde, resumidamente, en vez de publicar operaciones, se publican identificadores de recursos, para poder accederlos de forma remota. [5]

5.2.2 Especificaciones que definen el servicio web

Un servicio web es una colección de protocolos abiertos y estándares usados para el intercambio de datos entre aplicaciones o sistemas.

Software ejecutándose en distintas plataformas, y escritos en distintos lenguajes de programación a través del uso de estos protocolos estándares se comunican entre sí.

Ejemplo:

- SOAP (Simple Access Object Protocol)
- WSDL (Web Service Definition Language)



- UDDI (Universal Description, Discovery & Integration)
- WS-Security (Web Service Security)
- WS-ReliableMessaging (Web Service ReliableMessagin)
- WS-Reliability (Web Service Reliability)
- WS-Addressing (Web Service Addressing)

5.2.3 Componentes de los servicios web

La plataforma básica de los servidores es XML (Extensible Markup Language) + HTTP (HyperText Transfer Protocol).

Todos los servicios web estándar utilizan los siguientes componentes:

- SOAP (Simple Object Access Protocol)
- UDDI (Universal Description, Discovery and Integration)
- WSDL (Web Services Description Language) [5]

5.2.4 Tecnologías y estándares de los servicios web

XML-RPC: Protocolo simple basado en XML para el intercambio de información entre sistemas. Los Requests son codificados en XML y enviados vía HTTP POST. Las respuestas son embebidas en el cuerpo de la respuesta HTTP. Es independiente de la plataforma.

SOAP: Protocolo de comunicación basado en XML para intercambio de mensajes entre sistemas. Especifica un formato para el intercambio de mensajes es independiente del lenguaje y de la plataforma. Es extensible, es desarrollado por la W3C.

WSDL: Es un formato estándar basado en XML para describir servicios web y mostrar cómo acceder a ellos.



UDDI: Es un lenguaje estándar basado en XML para describir, publicar y encontrar servicio web. Es independiente de plataforma y puede comunicarse mediante SOAP, CORBA y JAVA Rmi. [5]

5.3 Protocolo

En la actualidad contamos con muchos protocolos de comunicación comerciales, incluso algunas empresas de telecomunicaciones, han llegado a desarrollar sus propios protocolos, dependiendo de los servicios que ofrezcan a sus usuarios. Estos protocolos muchas veces aun sin darnos cuenta son usados por nosotros y nos ayudan a hacer tareas como los son el Internet, una transferencia por módem o una simple comunicación a un servicio en línea inteligente de algún banco. Para saber un poco más de ellos es importante conocer su definición y algunos protocolos utilizados en los sistemas distribuidos.

5.3.1 Definición

Es un conjunto bien conocido de reglas y formatos que se utilizan para la comunicación entre procesos que realizan una determinada tarea. Se requieren dos partes:

- Especificación de la secuencia de mensajes que se han de intercambiar.
- Especificación del formato de los datos en los mensajes.

Un protocolo permite que componentes heterogéneos de sistemas distribuidos puedan desarrollarse independientemente, y por medio de módulos de software que componen el protocolo, haya una comunicación transparente entre ambos componentes. Es conveniente mencionar que estos componentes del protocolo deben estar tanto en el receptor como en el emisor. [1]



5.3.2 Protocolos utilizados en los sistemas distribuidos

IP - Protocolo de Internet: Protocolo de la capa de Red, que permite definir la unidad básica de transferencia de datos y se encarga del direccionamiento de la información, para que llegue a su destino en la red.

TCP - Protocolo de Control de Transmisión: Protocolo de la capa de Transporte, que permite dividir y ordenar la información a transportar en paquetes de menor tamaño para su transporte y recepción.

HTTP - Protocolo de Transferencia de Hipertexto: Protocolo de la capa de aplicación, que permite el servicio de transferencia de páginas de hipertexto entre el cliente WEB y los servidores.

SMTP - Protocolo de Transferencia de Correo Simple: Protocolo de la capa de aplicación, que permite el envío de correo electrónico por la red.

POP3: Protocolo de Oficina de Correo: Protocolo de la capa de aplicación, que permite la gestión de correos en Internet, es decir, le permite a una estación de trabajo recuperar los correos que están almacenados en el servidor. [2]



5.4 Aplicaciones web

Cuando se habla de sistemas web también se refiere a aplicaciones web por eso posteriormente se conocerá su definición, sus principales características, funcionamiento, y los tipos de aplicaciones.

5.4.1 Definición

Los “sistemas web” o también conocido como “aplicaciones web” son aquellos que están creados e instalados no sobre una plataforma o sistemas operativos (Windows, Linux). Sino que se alojan en un servidor en Internet o sobre una intranet (red local). Su aspecto es muy similar a páginas web que vemos normalmente, pero en realidad los “Sistemas Web” tienen funcionalidades muy potentes que brindan respuestas a casos particulares. [6]

5.4.2 Características

Compatibilidad multiplataforma: Una misma versión de la aplicación puede correr sin problemas en múltiples plataformas como Windows, Linux, Mac, etc.

Actualización: Las aplicaciones web siempre se mantienen actualizadas y no requieren que el usuario deba descargar actualizaciones y realizar tareas de instalación.

Acceso inmediato y desde cualquier lugar: Las aplicaciones basadas en tecnologías web no necesitan ser descargadas, instaladas y configuradas. Además, pueden ser accedidas desde cualquier computadora conectada a la red desde donde se accede a la aplicación.

Menos requerimientos de hardware: Este tipo de aplicación consume muy poco espacio en disco y también es mínimo el consumo de memoria RAM en comparación con los programas instalados localmente. Tampoco es necesario disponer de computadoras con poderosos procesadores ya que la mayor parte del trabajo se realiza en el servidor en donde reside la aplicación.



Menos Bugs (errores): Son menos propensas a crear problemas técnicos debido a problemas de software y conflictos de hardware. Otra razón es que con aplicaciones basadas en web todos utilizan la misma versión, y los bugs (errores) pueden ser corregidos tan pronto como son descubiertos beneficiando inmediatamente a todos los usuarios de la aplicación.

Seguridad en los datos: Los datos se alojan en servidores con sistemas de almacenamiento altamente fiables y se ven libres de problemas que comúnmente sufren los ordenadores de usuarios comunes como virus y/o fallas de disco duro. [6]

5.4.3 Funcionamiento de las aplicaciones web

Una aplicación web es proporcionada por un servidor web y utilizada por usuarios que se conectan desde cualquier punto vía clientes web (browsers o navegadores). La arquitectura de un Sitio web tiene tres componentes principales: [7]

- Un servidor Web
- Una conexión de red
- Uno o más clientes

El servidor web distribuye páginas de información formateada a los clientes que las solicitan. Los requerimientos son hechos a través de una conexión de red, y para ello se usa el protocolo HTTP. Una vez que se solicita esta petición mediante el protocolo HTTP y la recibe el servidor web, éste localiza la página web en su sistema de archivos y la envía de vuelta al navegador que la solicitó. [8]



5.4.4 Tipo de aplicaciones web

Según su uso, las aplicaciones web se dividen en catorce tipos:

Comercio electrónico. Estas aplicaciones se utilizan para la venta de productos de consumo, debiendo ser capaces de gestionar los procesos de compra y pago. A modo de ejemplo, hay aplicaciones basadas en PrestaShop y tiendas en línea como Amazon.

Marketing y presentación de productos. En este tipo de aplicaciones Web, el objetivo principal no es la venta directa de productos al consumidor, sino publicitar de forma atractiva la imagen de una marca o línea de productos. Algunos ejemplos son sitios de marcas como bmw.de o apple.com.

Sitios de noticias y blogs. Estas aplicaciones están diseñadas para informar, y se nutren de entradas o noticias que se actualizan con regularidad. Ejemplos de esto son periódicos digitales como The Guardian, aplicaciones basadas en WordPress y blogs de activismo.

Correo electrónico, gestión de eventos y alojamiento de ficheros. Este tipo de aplicaciones se caracterizan por organizar el trabajo personal de los usuarios, como el correo, la agenda o los documentos. Algunos sitios a modo de ejemplo son Gmail, Sched o Dropbox.

Mapas, rutas y lugares. Las aplicaciones de este tipo tienen como objetivo ubicar en mapas lugares concretos o por categorías, permitiendo realizar búsquedas, generar rutas y con opciones de aumento. Algunos ejemplos son sitios como ViaMichelin o los de tipo Google Maps.

Foros de discusión. Estas aplicaciones web se emplean para dar soporte a discusiones u opiniones y están organizadas en categorías que contienen foros. Dentro de cada foro se encuentran los temas de discusión, donde los usuarios pueden abrir nuevos temas o contestar



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



a los ya existentes. Existen ejemplos de aplicaciones como phpBB o vBulletin y sitios como The Joomla! Forum.

Juego en línea. Estas son aplicaciones interactivas diseñadas para el entretenimiento. En muchos casos requieren de complementos instalados en el navegador como Java o Flash para funcionar. Ejemplos de este tipo de aplicaciones se pueden encontrar en sitios web como Bigpoint.com.

Redes sociales. Las aplicaciones web de este tipo tienen como objetivo interconectar personas entre sí, de modo que cada uno de los miembros va formando su propia red de contactos de forma independiente, con los que puede comunicarse y compartir noticias e intereses. Algunos ejemplos son los sitios web de Facebook y Twitter.

Vídeos, fotos, música y software. Este tipo de aplicaciones se caracterizan por sus posibilidades multimedia y de compartición en la red. En muchas ocasiones requieren de complementos como Flash para funcionar. Ejemplos de estas aplicaciones son sitios como YouTube, Instagram y Grooveshark.

Wikis, enciclopedias y diccionarios. Estas aplicaciones web tienen como objetivo permitir la consulta de definiciones, documentación o artículos enciclopédicos y la edición colectiva de entradas. Ejemplos de esto son aplicaciones basadas en MediaWiki, Wikipedia y Wiktionary.

Oferta / demanda. En este tipo de aplicaciones los usuarios han de ser capaces de publicar anuncios (ofreciendo o demandando un empleo, objetos de segunda mano u otros servicios), de modo que otros usuarios puedan consultar esta información y encontrar aquella que sea de su interés. Algunos ejemplos son portales como milanuncios.com e Infojobs.

Información y servicios de la comunidad. Este tipo de aplicaciones web son utilizados por la administración para poner a disposición de cualquier ciudadano servicios como los ofrecidos



por ayuntamientos, hacienda o seguridad social. A modo de ejemplo, se pueden citar agenciatributaria.es y Servef.

Banca electrónica. Este tipo de aplicación web está diseñado para poder realizar gestiones bancarias a distancia, como la consulta de datos económicos o transferencias. Se pueden citar ejemplos de banca en línea como Banca Internet Triodos o Banc Sabadell Online.

Mashups, buscadores y comparadores. Estas aplicaciones usan y combinan datos, presentaciones o funcionalidad procedentes de una o más fuentes para permitir la realización de búsquedas y comparaciones o crear nuevos servicios. Un ejemplo de esto es vuelosbaratos.es. [6]

5.5 Base de datos

Una base de datos (cuya abreviatura es BD) en el cual existen muchas definiciones complejas de diferentes escritores pero una de las más acertadas es la que se encuentra en el libro titulado introducción a las base de datos: el modelo relacional. En él se detallan los tipos de base de datos, características, ventajas, desventaja, y sistemas gestores de base de datos.

5.5.1 Definición

Es un fondo común de información almacenada en una computadora para que cualquier persona o programa computadora autorizada pueda acceder a ella, independientemente de su lugar de procedencia y del uso que haga de ella. Siendo un conjunto de datos comunes que se almacenan sin redundancia para ser útiles en diferentes aplicaciones. [3]

5.5.2 Tipos de base de datos

Las bases de datos jerárquicas: En una base de datos jerárquica se organizan los datos utilizando estructuras arborescentes (en árbol). Un ÁRBOL es una estructura jerárquica en



la que los elementos se suelen denominar NODOS y existen dependencias entre los nodos. La dependencia es de 1:M del tipo padre/hijo. Un hijo no puede tener más de un padre, pero un padre varios hijos.

Las bases de datos en red: Actualmente en desuso, en una base de datos en red se utiliza la estructura de grafo/red, como en el caso anterior los distintos objetos están relacionados entre sí mediante relaciones del tipo 1:M pero en este caso un objeto puede estar relacionado como hijo con varios elementos que serán sus padres. En este caso las relaciones que se crean se denominan SET y el equivalente al padre se denomina PROPIETARIO (OWNER) y el equivalente al hijo se denomina MIEMBRO (MEMBER).

Las bases de datos relacionales: Esta es la estructura que se ha impuesto para aplicaciones de gestión, consiste en organizar los datos en forma de tablas, las relaciones entre los objetos se consiguen incluyendo en la tabla del hijo, la clave del objeto padre. Como son las que utilizaremos durante todo el módulo hemos reservado un apartado especial para ellas.

Las bases de datos orientadas a objetos: Es un modelo más reciente, trata de almacenar en la base de datos los objetos completos (estado y comportamiento). La información que contienen se organiza en atributos y el comportamiento en operaciones.

Las bases de datos multidimensionales: En una base de datos multidimensional los datos se almacenan en tablas de múltiples dimensiones en vez de tablas bidimensionales como las del modelo relacional. Se utilizan para grandes volúmenes de información. [3]

5.5.3 Base de datos relacionales

Una base de datos relacional es una colección de elementos de datos organizados en un conjunto de tablas formalmente descritas desde la que se puede acceder a los datos o volver



a montarlos de muchas maneras diferentes sin tener que reorganizar las tablas de la base. La base de datos relacional fue inventada por E.F. Codd en IBM en 1970.

La interfaz estándar de programa de usuario y aplicación a una base de datos relacional es el lenguaje de consultas estructuradas (SQL). Los comandos de SQL se utilizan tanto para consultas interactivas para obtener información de una base de datos relacional y para la recopilación de datos para los informes.

Además de ser relativamente fáciles de crear y acceder, una base de datos relacional tiene la importante ventaja de ser fácil de extender. Después de la creación original de una base de datos, una nueva categoría de datos se puede añadir sin necesidad de que todas las aplicaciones existentes sean modificadas.

Una base de datos relacional es un conjunto de tablas que contienen datos provistos en categorías predefinidas. Cada tabla (que a veces se llaman 'relación') contiene una o más categorías de datos en columnas. Cada fila contiene una instancia única de datos para las categorías definidas por las columnas. Por ejemplo, una base de datos típica de ingreso de solicitudes de negocio incluiría una tabla que describiera a un cliente con columnas para el nombre, dirección, número de teléfono, y así sucesivamente. Otra tabla identificaría el pedido: producto, cliente, fecha, precio de venta, y así sucesivamente. Un usuario de la base de datos podría obtener una vista de la base de datos que se ajuste a sus necesidades. Por ejemplo, un gerente de sucursal podría preferir una vista o informe sobre todos los clientes que han comprado productos después de una fecha determinada. Un gerente de servicios financieros en la misma empresa podría, desde las mismas tablas, obtener un informe sobre las cuentas que deben ser pagadas. [4]



5.5.3.1 Características de las base de datos relacionales

- Una base de datos relacional se compone de varias tablas o relaciones.
- No pueden existir dos tablas con el mismo nombre ni registro.
- Cada tabla es a su vez un conjunto de registros (filas y columnas).
- La relación entre una tabla padre y un hijo se lleva a cabo por medio de las claves primarias y ajenas (o foráneas).
- Las claves primarias son la clave principal de un registro dentro de una tabla y éstas deben cumplir con la integridad de datos.
- Las claves ajenas se colocan en la tabla hija, contienen el mismo valor que la clave primaria del registro padre; por medio de éstas se hacen las relaciones.

5.5.3.2 Ventajas de las bases de datos relacionales

- Provee herramientas que garantizan evitar la duplicidad de registros.
- Garantiza la integridad referencial, así, al eliminar un registro elimina todos los registros relacionados dependientes.
- Favorece la normalización por ser más comprensible y aplicable.

5.5.3.3 Desventajas de las bases de datos relacionales

- Presentan deficiencias con datos gráficos, multimedia, CAD y sistemas de información geográfica.
- No se manipulan de forma manejable los bloques de texto como tipo de dato.
- Las Bases de Datos Orientadas a Objetos (BDOO) se propusieron con el objetivo de satisfacer las necesidades de las aplicaciones anteriores y así, complementar, pero no sustituir a las bases de datos relacionales.



5.5.4 Sistemas gestores de base de datos

Se trata de un conjunto de programas no visibles al usuario final que se encargan de la privacidad, la integridad, la seguridad de los datos y la interacción con el sistema operativo proporciona una interfaz entre los datos, los programas que los manejan y los usuarios finales.

Cualquier operación que el usuario realiza o solicita a la base de datos, está controlada por el gestor. [3]

5.5.5 Sistemas de gestores de base de datos más utilizados

MySQL: Es un sistema gestor de bases de datos relacionales rápido, sólido y flexible. Es idóneo para la creación de bases de datos con acceso desde páginas web dinámicas, así como para la creación de cualquier otra solución que implique el almacenamiento de datos, posibilitando realizar múltiples y rápidas consultas. Está desarrollado en C y C++, facilitando su integración en otras aplicaciones desarrolladas también en esos lenguajes.

Es un sistema cliente/servidor, por lo que permite trabajar como servidor multiusuario y de subprocesamiento múltiple, o sea, cada vez que se crea una conexión con el servidor, el programa servidor establece un proceso para manejar la solicitud del cliente, controlando así el acceso simultáneo de un gran número de usuarios a los datos y asegurando el acceso a usuarios autorizados solamente. Es uno de los sistemas gestores de bases de datos más utilizado en la actualidad, utilizado por grandes corporaciones como Yahoo! Finance, Google, Motorola, entre otras.

Microsoft SQL Server: Es un sistema gestor de base de datos relacionales producido por Microsoft. Es un sistema cliente/servidor que funciona como una extensión natural del sistema operativo Windows. Entre otras características proporciona integridad de datos, optimización de consultas, control de concurrencia y backup y recuperación.



Es relativamente fácil de administrar a través de la utilización de un entorno gráfico para casi todas las tareas de sistema y administración de bases de datos. Utiliza servicios del sistema operativo Windows para ofrecer nuevas capacidades o ampliar la base de datos, tales como enviar y recibir mensajes y gestionar la seguridad de la conexión. Es fácil de usar y proporciona funciones de almacenamiento de datos que sólo estaban disponibles en Oracle y otros sistemas gestores de bases de datos más caros.

PostgreSQL: Es un sistema gestor de bases de datos relacionales orientadas a objetos, derivado de Postgres, desarrollado en la universidad de California, en el departamento de ciencias de la computación de Berkeley. Es un gestor de bases de datos de código abierto, brinda un control de concurrencia multi-versión (MVCC por sus siglas en inglés) que permite trabajar con grandes volúmenes de datos; soporta gran parte de la sintaxis SQL y cuenta con un extenso grupo de enlaces con lenguajes de programación.

Posee características significativas del motor de datos, entre las que se pueden incluir las subconsultas, los valores por defecto, las restricciones a valores en los campos (constraints) y los disparadores (triggers). Ofrece funcionalidades en línea con el estándar SQL92, incluyendo claves primarias, identificadores entrecomillados, conversión de tipos y entrada de enteros binarios y hexadecimales.

El código fuente se encuentra disponible para todos sin costo alguno. Está disponible para treinta y cuatro plataformas con la última versión estable. Es totalmente compatible con ACID (acrónimo de Atomicity, Consistency, Isolation and Durability; en español: Atomicidad, Consistencia, Aislamiento y Durabilidad).

Posee una integridad referencial e interfaces nativas para lenguajes como ODBC, JDBC, C, C++, PHP, PERL, TCL, ECPG; PYTHON y RUBY. Funciona en todos los sistemas operativos Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), y Windows.



Debido a la liberación de la licencia, PostgreSQL se puede usar, modificar y distribuir de forma gratuita para cualquier fin, ya sea privado, comercial o académico. [9]

5.6 Aplicación móvil (App)

Las aplicaciones móviles son uno de los segmentos del marketing móvil que mayor crecimiento ha experimentado en los últimos años. Se pueden encontrar en la mayoría de los teléfonos, incluso en los modelos más básicos (donde proporcionan interfaces para el envío de mensajería o servicios de voz), aunque adquieren mayor relevancia en los nuevos teléfonos inteligentes. [10]

“Una App es toda aplicación la cual se ejecuta o funciona sobre determinada plataforma de software, estas y su funcionamiento están directamente ligados a la plataforma de Software y/o Hardware sobre las que se ejecutan”.

5.6.1 Tipos de Apps

Las aplicaciones pueden dividirse en diversas calificaciones, sin embargo, las más aceptadas en la comunidad desarrolladora son las siguientes:

- Según su entorno de funcionamiento.
- Según su funcionalidad.

Según su entorno de funcionamiento

- Funcionamiento de la App en sistema operativo: Según su entorno de ejecución, estas se ejecutan directamente desde el sistema operativo que las contiene (e.g. Android, iOS, BlackBerry OS, Windows Mobile).



- Funcionamiento de la App desde plataforma web: Dan lugar a las aplicaciones que ejecutan desde un entorno de navegador web.
- Funcionamiento sobre otra plataforma: Como Java/J2ME, BREW, Flash Lite o Silverlight (menos utilizadas en la actualidad).

Según su funcionalidad

- Comunicaciones
- Multimedia
- Juegos
- Productividad
- Viajes
- Compras
- Utilidades
- Entretenimiento
- Bienestar [10]

5.6.2 Entornos

En la actualidad existen múltiples entornos sobre los cuales es posible desarrollar aplicaciones móviles con el fin de alcanzar un público específico sobre las distintas plataformas móviles en el mercado, Las más utilizadas hasta la fecha son:

- Android
- Apple iOS
- Windows phone
- Firefox OS (En menor medida)

5.6.2.1 Entornos de desarrollo android



Google es el gestor principal de android, pero este sistema operativo surge como un producto de un consorcio llamado Handset Alliance. Android es un sistema operativo móvil basado en Linux y Java que ha sido liberado bajo la licencia Apache versión 2. El sistema busca, nuevamente, un modelo estandarizado de programación que simplifique las labores de creación de aplicaciones móviles y normalice las herramientas en el campo de la telefonía móvil. Al igual que ocurriera con Symbian, lo que se busca es que los programadores sólo tengan que desarrollar sus creaciones una única vez y así ésta sea compatible con diferentes terminales. Google promete una plataforma de desarrollo gratuita, flexible, económica en el desarrollo de aplicaciones y simple, diferenciada de los estándares que ofrecen Microsoft o Symbian. [11]

5.6.2.2 Android

Android: Es un sistema operativo desarrollado por la Open Handset Alliance un consorcio de empresas de hardware, software y telecomunicaciones. Basado en Linux, se diseñó especialmente para ser utilizado en dispositivos móviles.

Al estar desarrollado sobre Linux y licencias de código abierto, desde sus comienzos tuvo una excelente acogida por parte del mercado y por ello cuenta con una gran comunidad de desarrolladores de aplicaciones. [12]

5.6.2.3 Entorno de desarrollo iOS

iPhone OS es una versión reducida de Mac OS X optimizada para los procesadores ARM. Aunque oficialmente no se puede instalar ninguna aplicación que no esté firmada por Apple ya existen formas de hacerlo, la vía oficial forma parte del iPhone Developer Program (de pago) y hay que descargar el SKD que es gratuito. iPhone dispone de un interfaz de usuario realmente interesante, la única pega es la cantidad de restricciones que tiene, aunque quizás Apple se dé cuenta que para triunfar mucho más es mejor liberar y dar libertad a su sistema. [13]



5.6.2.4 Entorno de desarrollo Windows phone

Microsoft lanzó su propio Windows para móviles, antes conocido como Windows CE o Pocket PC, tiene una larga historia como segundón en el campo de los PDA u ordenadores de bolsillo, sin embargo, hace pocos meses superó por primera vez al hasta entonces líder, Palm OS. Windows Mobile es un sistema operativo escrito desde 0 y que hace uso de algunas convenciones de la interfaz de usuario del Windows de siempre. Una de las ventajas de Windows Mobile sobre sus competidores es que los programadores pueden desarrollar aplicaciones para móviles utilizando los mismos lenguajes y entornos que emplean con Windows para PC. En comparación, las aplicaciones para Symbian necesitan más esfuerzo de desarrollo, aunque también están optimizadas para cada modelo de teléfono [13]



6. Marco Metodológico



6.1 Sistema de variables

Es una propiedad que puede variar y cuya variación es susceptible de medirse

6.1.2 Hipótesis

El sistema contribuirá a evitar los “choques” que se dan entre las horas de los docentes, facilitará y acelerará el proceso de generación de los horarios de cada una de las carreras de la universidad, con el uso de este sistema el proceso de aprobación de los horarios será más rápido y eficiente, mediante la plataforma se mejorará la comunicación entre los docentes y los estudiantes y gracias a las tecnologías utilizadas, los procesos que se realizan en el sistema serán intuitivos o fáciles de aprender.



6.1.3 Operacionalización de variables

Objetivos	Variables	Descripción
<ul style="list-style-type: none">• Identificar los diferentes requerimientos que se presentan en la generación de horarios, utilizando el estándar de especificación de requisitos de software IEEE 830.	Identificación de requerimientos	Conocer el proceso requerido para la generación del horario de los diferentes departamentos en los que se agrupan las cada una de las carreras
<ul style="list-style-type: none">• Diseñar las interfaces y procesos en base a los requerimientos identificados, aplicando herramientas de diseño.	Diseñar aplicación	Luego de conocer el proceso requerido para generar el horario diseñar la herramienta en base a las necesidades encontradas
<ul style="list-style-type: none">• Implementar la codificación del diseño obtenido, a través de lenguajes de programación y tecnología web y móviles, orientada a una arquitectura distribuida.	Codificar y documentar	Codificar los procesos a realizar en base al diseño y documentar de forma correcta cada uno de los procesos que se toman en cuenta para este desarrollo.
<ul style="list-style-type: none">• Evaluar el desempeño del sistema web y las aplicaciones móviles, de acuerdo a los requerimientos establecidos.	Evaluar	Al finalizar con el desarrollo, se evalúa el nivel en que la herramienta cumple con los requerimientos y expectativas.

Tabla 1 Primera Operacionalización



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Variable	Dimensiones	Datos de entrada o indicadores
Identificación de requerimientos	Organizacional Procesos Tecnología	Roles Datos de entrada Datos de salida Asociaciones de proceso Procesos Restricciones
Diseñar la aplicación	Organizacional Tecnología Procesos	Diagramas UML Casos de Uso Flujograma Diagrama ER Bocetos
Codificar y documentar	Procesos Tecnología	Roles Datos de entrada Datos de salida Asociaciones de proceso Procesos Restricciones
Evaluar	Organizacional Procesos Tecnología	Roles Datos de entrada Datos de salida Asociaciones de proceso Procesos

Tabla 2 Segunda Operacionalización



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Variable	Datos de entrada o indicadores	Instrumento – herramienta
Identificación de requerimientos	Roles Datos de entrada Datos de salida Asociaciones de proceso Procesos Restricciones	Estándar IEEE 830 Observación
Diseño de la aplicación	Diagramas UML Casos de Uso Flujograma Diagrama E/R Bocetos	MockupBuilder Star UML Embarcadero Justinmind Prototyper
Codificar y documentar	Roles Datos de entrada Datos de salida Asociaciones de proceso Procesos Restricciones	Visual Studio Xamarin Studio Procesador de texto Sql Server
Evaluar	Roles Datos de entrada Datos de salida	Observación

Tabla 3 Tercera Operacionalización



6.2 Tipo de investigación

El presente documento es una investigación aplicada con un diseño transversal.

Una investigación aplicada es aquella que parte de una situación problemática que requiere ser intervenida y mejorada. Comienza con la descripción sistemática de la situación deficitaria, luego se enmarca en una teoría suficientemente aceptada de la cual se exponen los conceptos más importantes y pertinentes; posteriormente, la situación descrita se evalúa a la luz de esta teoría y se proponen secuencias de acción o un prototipo de solución. Supone el uso de los métodos de la investigación-acción-participación, es decir, relación directa con la comunidad afectada por la problemática. Las propuestas de solución deben integrar los conocimientos propios del Comunicador social-Periodista o del Comunicador Audiovisual-multimedia, según sea el caso. [14].

Se consideran transversales los estudios en los que se examinan la relación entre un problema y una serie de variables en una población determinada y en un momento del tiempo. Es decir, la presencia del problema y la exposición se observa simultáneamente, lo que dificulta la interpretación causa efecto. Por esta razón los estudios transversales son por definición descriptivos. [15]



6.3 Métodos y técnicas de recolección

Una investigación es científicamente válida al estar sustentada en información verificable, que responda lo que se pretende demostrar con la hipótesis formulada. Para ello, es imprescindible realizar un proceso de recolección de datos en forma planificada y teniendo claros objetivos sobre el nivel y profundidad de la información a recolectar. Se presenta en este artículo una serie de criterios a considerar para diseñar la herramienta de recolección de información, así como los métodos de recolección para lograr en una investigación resultados confiables. [16].

Para obtener la información necesaria para la presente investigación se utilizaron dos técnicas, muy conocidas y utilizadas ampliamente en la comunidad investigativa, estas son:

6.3.1 Observación

"El estudio del acto de enseñanza utiliza como principal instrumento de investigación, sistemas o planos de observación de acontecimientos o comportamientos". Como toda ciencia la enseñanza y el entrenamiento necesitan de grabaciones objetivas. La observación constituye un método de toma de datos destinados a representar lo más fielmente posible lo que ocurre, la realidad. [17].

6.3.2 Encuesta

Constituye el término medio entre la observación y la experimentación. En ella se pueden registrar situaciones que pueden ser observadas y en ausencia de poder recrear un experimento se cuestiona a la persona participante sobre ello. Por ello, se dice que la encuesta es un método descriptivo con el que se pueden detectar ideas, necesidades, preferencias, hábitos de uso [17]



6.4 Técnicas de procesamiento y análisis de datos

Para la recolección de las necesidades, requerimientos funcionales o no funcionales, roles, características que el sistema debe tener y actividades que debe cumplir se utilizó un documento de especificación de software (SRS) el cual está basado en el estándar IEEE 830 – 1998.

Las técnicas que se toman en cuenta para el desarrollo de esta aplicación web son las que encontramos en el paradigma de desarrollo según Ian Sommerville, en su libro de “Ingeniería de Software” última edición. [18]

Para la recolección de datos característicos del proceso de asignación de horarios se utilizó el método de encuestas.



7. Marco administrativo



7.1 Cronograma de documento

Es un apoyo importante, no solo para la organización, del tiempo, sino para el control de avance de la investigación y para facilitar en cualquier momento un reporte parcial de la misma; proporciona además una visión global del trabajo por realizar. [19]

Este cronograma se elaboró con el software Project Management que se encarga de simplificar la administración del proyecto, que brinda la posibilidad de organizar y controlar las tareas. [20]

A continuación muestra cada una de las fases del documento con su fecha de inicio y finalización, así mismo poder ver los avances de la documentación. *Ver anexo (Figura 1)*



7.2 Presupuesto de trabajo

Barry Boehm, en su libro clásico de economía de la ingeniería de Software, introduce una jerarquía de modelos de estimación de software con el nombre de COCOMO, por su nombre en inglés (Constructive, Cost, Model) Modelo constructivo de costos. La jerarquía de modelos de Boehm está constituida por los siguientes:

Modelo I: El modelo COCOMO básico calcula el esfuerzo y el costo del desarrollo de software en función del tamaño del programa, expresado en las líneas estimadas.

Modelo II: El Modelo COCOMO intermedio calcula el esfuerzo del desarrollo de software en función del tamaño del programa y de un conjunto de conductores de costos que incluye en la evaluación subjetiva del producto, del hardware, del personal y de los atributos del proyecto.

Modelo III: El Modelo COCOMO avanzado incorpora todas las características de la versión intermedia y lleva a cabo una evaluación del impacto de los conductores de costos en cada caso (análisis, diseño, etc.) del proceso de ingeniería de software. El Modelo COCOMO Básico Los Modelos COCOMO están definidos para tres tipos de proyectos de software. Utilizando la terminología de Boehm son:

1. Modelo Orgánico: Proyectos de software relativamente pequeños y sencillos en donde trabajan pequeños equipos, con buena experiencia en la aplicación, sobre un conjunto de requisitos poco rígidos (por ejemplo, un programa de análisis termal desarrollado para un grupo calorífico):
2. Modelo Semiacoplado: Proyectos de software intermedios en los que equipos, con variados niveles de experiencia, deben satisfacer requerimientos poco o medio rígidos.
3. Modo a la Medida: Proyectos de software que deben ser desarrollados en un conjunto de hardware, software y restricciones operativas y muy restringido. Técnicas de Estimación de Costos del Software.



La ecuación del COCOMO básico, aplicada al proyecto, tiene la siguiente forma:

$$E = aKLDC^b$$

$$D = cE^d$$

$$P = E/D$$

En donde E es el esfuerzo aplicado en personas-mes, D es el tiempo de desarrollo en meses cronológicos y KLDC es el número estimado en líneas de código distribuidas (expresada en miles) para el proyecto. Los coeficientes a y c y los exponentes b y d se muestran en la tabla siguiente

PROYECTO SOFTWARE	a	b	c	d
Orgánico	2,4	1,05	2,5	0,38
Semi-acoplado	3,0	1,12	2,5	0,35
Empotrado	2,8	1,20	2,5	0,32

Tabla 4 Modelos de desarrollo de software

Tanto a, b, c y d son constantes y presentan un valor diferente para cada modo de desarrollo, siendo nuevamente el orgánico en aplicado a continuación, conocemos que:

$$a = 2,4$$

$$b = 1,05$$

$$c = 2,5$$



$$d = 0,38$$

Lenguaje	Líneas de Código
C#	16758
HTML	7925
Sql	3895
JavaScript	2247
Total	30,825

Tabla 5 Dimensiones del proyecto

Siendo 22,225 LDC Obtenidas, de esta manera $KLDC=2.22$ y podemos obtener:

$$E = 2.4(3.08)^{1.05} = 3.25 \text{ Personas/Mes}$$

Se estima deben de existir 3.31 personas desarrollado para el software cada mes.

$$D = 2.5(3.25)^{0.38} = 3.9 \text{ Meses de desarrollo}$$

Se obtiene que es necesario trabajar 3.4 meses con 3.31 personas

Productividad:

PR(Productividad) según las cifras obtenidas seria:

$$PR = LDC/E = (30,825/3.25) = 9,484 \text{ LDC/} \mathbf{Por \text{ persona}}$$

La productividad indica que, deberán ser escritas 9,484 Líneas de Código por cada persona parte del equipo de desarrollo.

Según las cifras obtenidas será necesario un equipo de 3.25 personas trabajando alrededor de 3.25 meses y se deberán escribir al menos 9,484 líneas de código cada mes, Dado que las



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



personas no se pueden dividir de esa manera y el tiempo límite es de 3 meses, se hace un cálculo de costo en base a esto, sabiendo que según el COCOMO el tamaño del equipo desarrollador y la cantidad de tiempo son coherentes con lo obtenido.

Costo:

Cada persona trabaja en promedio 3 horas al día de lunes a viernes, teniendo en el mes un total de 60 horas al mes por persona.

Total Horas Trabajadas * 3 Personas: $(60H * 3Personas) * 3Meses = 540 H.$

Horas Trabajadas Por persona = $60H * 3Meses = 180 Horas.$

LDC totales por persona = $9484 / 3 = 3161.33 LDC$ al Mes.

Costo de desarrollo

Gasto	Precio Unitario	Precio Total
Pago a desarrolladores	\$15 * Hora	\$ 8100
Viáticos	\$50 * Persona	\$ 150
Electricidad		\$100
Precio Total Del Desarrollo		\$ 8350

Tabla 6 Costo del desarrollo



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Costo del documento

Gasto	Precio Unitario	Precio Total
Horas	C\$50 * 40 horas	C\$ 2,000
Electricidad	C\$5 * 40 horas	C\$ 200
Papelera	-	C\$100
Impresión	-	C\$50
Fotocopias	C\$25 * 2copias	C\$50
Varios (Transporte, comunicación internet)	-	C\$500
Precio Total De Documentación		C\$ 2900
Cambio Dólar (C\$29.35)		\$99

Tabla 7 Costo del documento

Costo total del Proyecto

Concepto	Costo
Costo de desarrollo	8350
Costo de documentación	\$99
Total	\$8449

Tabla 8 Costo total del proyecto



8. Marco de resultado



8.1 Cronograma del software

Es una técnica de medida de desempeño de un proyecto que integra variables fundamentales trabajo y tiempo (Comienzo y fin) en lo cual se puede determinar que tan bien el proyecto está cumpliendo los objetivos a una fecha específica de control o evaluación. [20]

En el cronograma de software se muestran cada una de las tareas que se llevaron a cabo, la duración especificando la fecha en que se inició y se culminó. *Ver anexo (Figura 2)*

8.2 Descripción específica de los resultados obtenidos por cada objetivo

8.2.1 Introducción

Estas etapas son un reflejo del proceso que sigue a la hora de resolver cualquier tipo de problema. Ya en 1945, mucho antes de que existiese la ingeniería de software, el matemático George Polya describió este proceso en su libro *How to solve it* (el primero que describe la utilización de técnicas heurísticas en la resolución de problemas). [21]

Básicamente resolver un problema requiere:

- Comprender el problema (análisis)
- Plantear una posible solución, considerando soluciones alternativas (diseño)
- Llevar a cabo la solución planteada (implementación)
- Comprobar que el resultado obtenido es correcto (pruebas)



8.2.2 Análisis

Es el proceso mediante el cual se intenta descubrir que es lo que realmente se necesita y se llega a una comprensión adecuada de los requerimientos del sistema, las características que el sistema debe de poseer. [22]

Mediante el proceso de identificación de los requerimientos se utilizó el estándar de especificación de requisitos del software IEEE 830 y técnicas para recopilación de información fue la observación y la encuesta que se procesó con la herramienta SPSS, obteniendo así los resultados. Este es un software de análisis estadístico que presenta las funciones principales necesarias para realizar el proceso analítico de principio a fin.

8.2.3 Diseño

Mientras que los modelos utilizados en la etapa representan los requisitos del usuario desde distintos puntos de vista (el qué), los modelos que se utilizan en la fase de diseño representan las características del sistema que nos permitirán implementarlo de forma efectiva (el cómo). [23]

Para el diseño del sistema web y aplicaciones móviles se utilizaron las siguientes herramientas:

Herramientas	Función
Balsamiq Mockups 3	Es una de las herramientas más útil para el prototipado, el cual permite modelar los requerimientos del usuario de tal modo que se acercan mucho a la realidad, mostrando de manera muy clara, la representación esquemática de la página web.
Justinmind prototyper	Consiste en crear bocetos interactivos sin necesidad de generar código de manera



	sencilla. En esta ocasión se utilizó para modelar las funciones de la aplicación móvil.
starUML	Es una herramienta para el modelamiento de software basado en los estándares UML (Unified Modeling Language). Esta hizo posible los casos de usos del sistema en donde muestra la interacción que tendrán los usuarios según su rol.
ER/Studio	Embarcadero es un software de modelado de datos fácil de usar y multinivel, para el diseño y construcción de bases de datos a nivel físico y lógico. Esta herramienta permitió crear el diagrama de normalización, presentando cualquier abstracción, percepción y conocimiento del sistema de información, formado por un conjunto de objetos denominados entidades y relaciones, incorporando una representación visual

Tabla 9 Herramientas para el diseño



8.2.4 Implementación

Una vez que sabemos qué funciones debe desempeñar nuestro sistema de información (análisis) y hemos decidido cómo vamos a organizar sus distintos componentes (diseño), es el momento de pasar a la etapa de implementación, pero nunca antes. Antes de escribir una sola línea de código (o de crear una tabla en nuestra base de datos) es fundamental haber comprendido bien el problema que se pretende resolver y haber aplicado principios básicos de diseño que nos permitan construir un sistema de información de calidad.

Para la fase de implementación hemos de seleccionar las herramientas adecuadas, un entorno de desarrollo que facilite nuestro trabajo y un lenguaje de programación apropiado para el tipo de sistema que vayamos a construir. La elección de estas herramientas dependerá en gran parte de las decisiones de diseño que hayamos tomado hasta el momento y del entorno en el que nuestro sistema deberá funcionar. [24]

Herramienta	Función
SQL Server 2012	Es un sistema de gestión de base de datos del modelo relacional, cuya principal función es la de almacenar y consultar datos solicitados por otras aplicaciones, sin importar si están en la misma computadora, si están conectadas a una red local o si están conectadas a través de internet. Con esta herramienta se crearon las tablas con sus procedimientos almacenados,
Xamarin Studio 2013	Es un IDE moderno y sofisticado utilizado para crear aplicaciones iOS, Mac y Android, presentando la funcionalidad que lo convierte en una excelente herramienta para crear aplicaciones móviles y de escritorio.



	Esta herramienta se utilizó para las aplicaciones móviles ya que trabaja con los archivos .axml Android para construir visualmente interfaces de usuario.
Visual Studio 2013	Es el IDE (Entorno de desarrollo integrado) de programación por excelencia de la plataforma .NET; por medio de este software se desarrolló el sistema, ya que brinda un conjunto completo de herramientas de desarrollo.

Tabla 10 Herramientas para la implementación

8.2.5 Evaluación

Una vez generado el código, comienzan la evaluación del software a través de las pruebas del software. De acuerdo a Pressman, el proceso de pruebas se centra en los procesos lógicos internos del sistema, asegurando que todas las sentencias se han comprobado y en los procesos externos funcionales, es decir, la realización de las pruebas para la detección de errores. [25]

Para las pruebas se utilizó las técnicas de cajas negras que se aplican sobre el sistema empleando un determinado conjunto de datos de entrada y observando las salidas que se producen para determinar si la función se está desempeñando correctamente.

Las técnicas de caja negra o funcional, obtienen casos a partir de los requisitos funcionales del programa a probar, por lo que no se tiene en cuenta la forma en que se codifica esa funcionalidad, sino que se consideran únicamente las entradas y salidas. [26]



9. Conclusiones

1. A lo largo de la presente investigación se dio a conocer la importancia de desarrollar un sistema web y aplicaciones móviles para la gestión de horario de asignaturas, con el estándar de especificación de requisitos del software IEEE 830, Se lograron identificar todos los requisitos que son considerados de mayor importancia en el desarrollo del proyecto.
2. Utilizando las herramientas de diseño adecuado y planeación necesarias, se definió el diseño requerido para los distintos módulos de la web y aplicaciones móviles, siendo posible obtener que sean rápidas, intuitivas y accesibles para el usuario final.
3. Empleando conocimientos, técnicas de desarrollo y su respectiva codificación; los procesos, funcionamientos del sistema web y aplicaciones móviles han sido implementada de manera exitosa.
4. A través de la evaluación de los distintos requerimientos tales como: requisitos comunes de las interfaces, requisitos funcionales y no funcionales entre otros, se conoció que mediante el desempeño del sistema web y aplicaciones móviles, no está a la perfección, pero en su mayoría cumple con las necesidades preestablecidas las cuales dan pauta principal para el desarrollo de software.



10. Recomendaciones

1. Para que en un futuro no se lleguen a presentar situaciones indeseadas, es necesaria la capacitación del personal que utilizará el sistema y aplicaciones móviles.
2. Revisar el manual técnico y de usuario en caso de una duda o se le presente un problema.
3. Establecer una conexión del sistema de asignación de horario directamente con el sistema central de la UNAN Managua para maximizar su utilidad.
4. La aplicación está diseñada para adaptarse de la forma necesaria para los distintos tamaños de pantallas, es recomendado utilizar Mozilla Firefox o Google Chrome para obtener mejores resultados.



11. Glosario de términos

Palabra	Significado
CAD	En inglés computer-aided design (diseño asistido por ordenador)
Heurísticas	Un arte, técnica o procedimiento práctico o informal, para resolver problemas.
HTML	Por su sigla en inglés de HyperText Markup Language (lenguaje de marcas de hipertexto).
HTTP	HyperText Transfer Protocol en español Protocolo de Transferencia de Hipertexto
INAP	Instituto Nacional de Administración Pública
PDA	PDA, del inglés personal digital assistant, asistente digital personal, computadora de bolsillo, organizador personal o agenda electrónica
Restful	(Representational state transfer) en español Transferencia de estado representacional.
SPSS	(Statistical Product and Service Solutions) Soluciones estadísticas de productos y servicios
Symbian	Fue un sistema operativo propiedad de Nokia, y que en el pasado fue producto de la alianza de varias empresas de telefonía móvi
WSDL,	Las siglas de Web Services Description Language, es un formato del Extensible Markup Language (XML) que se utiliza para describir servicios web

Tabla 11 Palabras desconocidas



12. Bibliografía

- [1] J. D. T. K. G. Coulouris, *Distributed Systems: Concepts And Design*, 2005.
- [2] L. Fabian, «Protocolos de sistemas distribuidos,» 2010. [En línea]. Available: <http://protocolosdesistemasdistribuidos.blogspot.com/>.
- [3] Departamento de ciencias de las computacion, «Fundamentos de diseño de las bases de datos,» de *Introducción a las bases de datos*, 2010.
- [4] A. Tanenbaun, *Distributed Systems: Principlres and Paradigms*, 2006.
- [5] A. Pastorini, «Servicios Web».
- [6] J. P. Maestras, «Instroducción a los Sistemas Web,» de *Tecnologías Web*, 2014.
- [7] W. S. Lescano, «Aprende a Programar,» 2009. [En línea]. Available: http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=102:ique-es-y-para-que-sirve-el-lenguaje-de-etiquetas-xml-extensible-markup-language&catid=46:lenguajes-y-entornos&Itemid=163.
- [8] U. d. Granada, «Universidad de Granada,» 2015. [En línea]. Available: http://www.ugr.es/~rescate/practicum/el_m_todo_de_observaci_n.htm.
- [9] J. Sierra, «Explorable.com,» 2009. [En línea]. Available: <https://explorable.com/es/muestreo-por-conveniencia>.
- [10] M. M. Asosiation, *Libro Blanco De Apps*, 2011.
- [11] G. p. l. c. d. a. móviles, *Andrez Rodriguez Escudero, Pereira*, 2013.
- [12] J. 2. D. 2. Escribano Arrechea, «Internet móvil para emprendedores,» de *Internet móvil para emprendedores.*, Madrid, ESPAÑA: Larousse, Ediciones Pirámide, 2013., 28 December 2016., p. 268.
- [13] F. Castellanos, *Desarrollo para aplicaciones moviles*, Lérganes, 2013.
- [14] U. d. L. Sabana, «Universidad de La Sabana,» 2011. [En línea]. Available: <http://www.unisabana.edu.co/carreras/comunicacion-social-y-periodismo/trabajo-de-grado/opciones-de-trabajo-de-grado/investigacion-aplicada/>.



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



- [15] U. d. Valencia, «www.uv.es,» [En línea]. Available: <http://www.uv.es/invsalud/invsalud/disenyo-tipo-estudio.htm>.
- [16] M. Torres, «<http://www.tec.url.edu.gt/>,» 2013. [En línea]. Available: http://www.tec.url.edu.gt/boletin/URL_03_BAS01.pdf.
- [17] M. Pieron, *Técnicas de investigación*, 1986.
- [18] I. Sommerville, «Ingeniería del software,» de *Estándar*, Pearson Educación , 2005, p. 687.
- [19] M. G. Moreno, «Introducción a la metodología de investigación educativa,» de *informe de investigación*, México, Editorial progreso, 2007, p. 272.
- [20] F. J. T. López, «Administración de proyectos informáticos,» de *Project Management Institute*, ECOE , 2013.
- [21] R. L. Glass, «Facts and fallacies of software engineering,» de *With reference of the mathematician George Polya*, Wesley, Addison, 2003.
- [22] F. Berzal, «El ciclo de vida de un sistema de información,» 10 Marzo 2011. [En línea]. Available: <http://flanagan.ugr.es/docencia/2005-2006/2/apuntes/ciclovida.pdf>. [Último acceso: 25 Enero 2017].
- [23] M. Fowler, *Patterns of Enterprise Application Architecture*, Wesley: Addison, 2003.
- [24] S. McConnell, *Code Complete: A practical Handbook of software construction*, Microsoft Press, 2004.
- [25] R. S. Pressman, *Ingeniería del software: un enfoque práctico*, McGraw-Hill, 2006.
- [26] I. R. R. J. J. D. C. Javier Tuya, *Técnicas cuantitativas para la gestión en la ingeniería del software*, Netbiblo, 2007.
- [27] I. d. I. E. Y. Electronicos, *Norma IEEE para software de usuario*, New York: Consejo de normas, 2001.
- [28] I. Sommerville, *Ingeniería de Software*, Madrid: Pearson Educación S.A., 2005.



13. Anexos

Tabla de contenido

1.	Cronogramas	1
2.	Instrumento de recolección de información	3
2.1	Resultados de las encuestas.....	6
3.	Casos de usos.....	17
3.1	Caso de uso usuario docente.....	17
3.2	Caso de uso usuario director de departamento	18
3.3	Caso de uso usuario coordinador	19
3.4	Caso de uso usuario administrador	20
3.5	Caso de uso usuario estudiante	21
4.	Base de datos.....	22
4.1	Diagrama de entidad/relación	22
4.2	Diccionario de datos	23
5.	Mapa de navegación.....	37
5.1	Web.....	37
5.2	Móvil	38
6	Bocetos	39
6.1	Web.....	39
6.1.1	Pantalla de Inicio de sesión	39
6.1.2	Pantalla de solicitud de usuario.....	39
6.1.3	Página principal	40
6.1.4	Pantalla de asignación de horario.....	40
6.1.5	Listado de grupos	41
6.1.6	Listado de carreras	41
6.1.7	Listado de docente.....	42
6.2	Bocetos móviles	43
6.2.1	Pantalla principal.....	43
6.2.2	Horario	43
6.2.3	Asignaturas.....	44
6.2.4	Asignaturas (mensaje de aviso)	44
6.2.5	Grupos.....	45

6.2.6	Turnos de carrera.....	45
6.2.7	Carreras.....	46
6.2.8	Actividades.....	46



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



1. Cronogramas

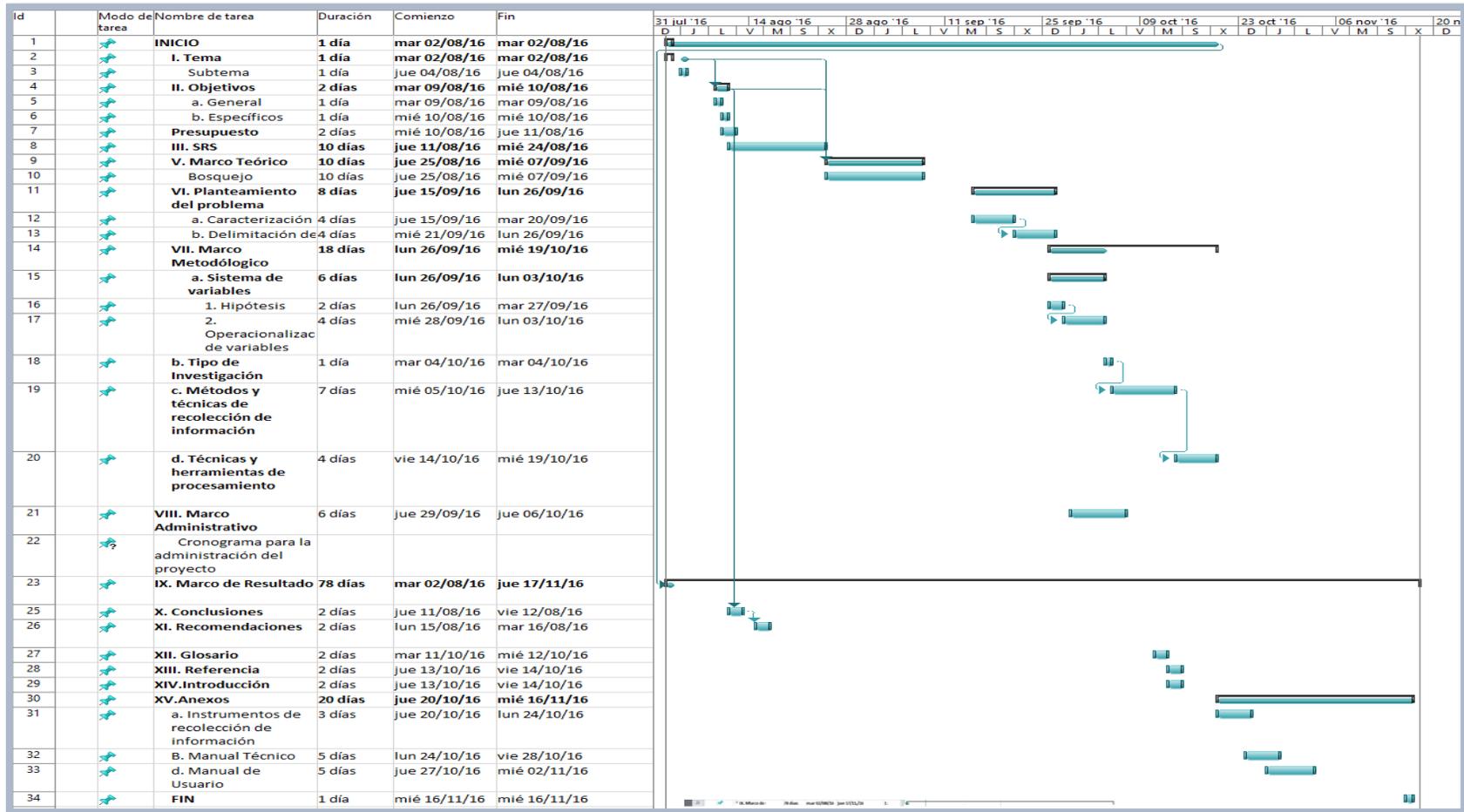


Figura 1 Anexo Cronograma de trabajo



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información

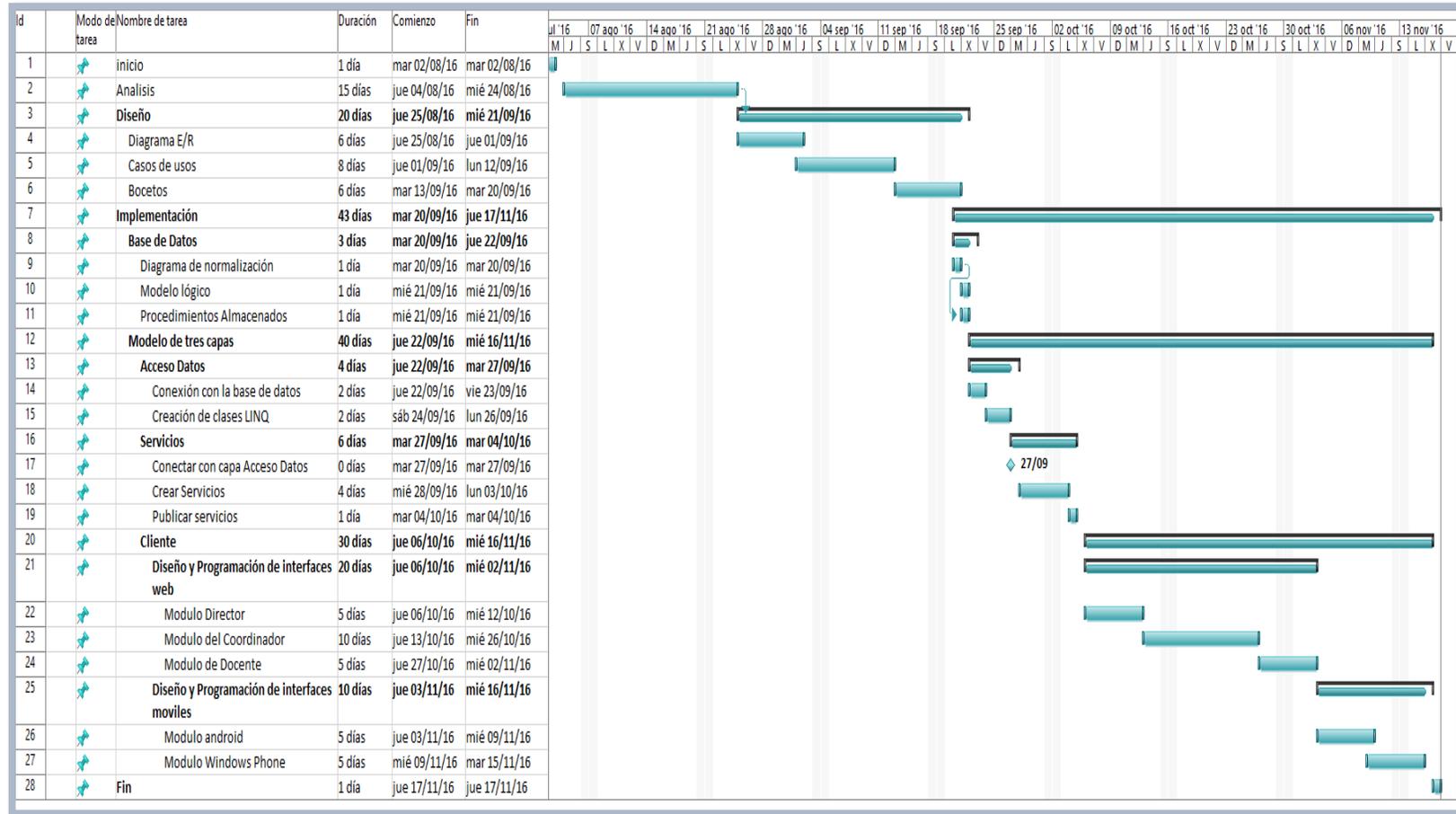


Figura 2 Anexo Cronograma de software



2. Instrumento de recolección de información

Universidad Nacional Autónoma de Nicaragua

Estimado Coordinador esperamos su apoyo para la presente encuesta que tiene como objetivo Evaluar el grado de interés/aceptación, de igual forma conocer la importancia de la Aplicación SAHO (Sistema de Asignación de Horario) según las necesidades que se presentan para usted en la asignación de horario de la carrera que coordina.

1. ¿De qué departamento es coordinador?

<input type="checkbox"/>	Ciencias, Tecnología y Salud
<input type="checkbox"/>	Ciencias Económicas y Administrativas
<input type="checkbox"/>	Educación y Humanidades

2. ¿Cuánto tiempo ha estado en su cargo?

3. De sus múltiples actividades ¿cuáles son las más complicadas?

4. ¿Cómo evalúa el nivel de complicación de la asignación de horario?

<input type="checkbox"/>	Bajo
<input type="checkbox"/>	Medio
<input type="checkbox"/>	Alto



5. ¿Considera usted que es necesario la automatización de este proceso?

<input type="checkbox"/>	Si
<input type="checkbox"/>	No

6. ¿Cuál considera usted que es su nivel de manejo de las TIC?

<input type="checkbox"/>	1
<input type="checkbox"/>	2
<input type="checkbox"/>	3
<input type="checkbox"/>	4
<input type="checkbox"/>	5

7. ¿Estaría usted dispuesto a mejorar su nivel de manejo de TIC?

<input type="checkbox"/>	Si
<input type="checkbox"/>	No

8. De las siguientes características ¿Cuáles conoce y considera necesarias en una aplicación?

<input type="checkbox"/>	Rapidez
<input type="checkbox"/>	Fácil de Usar
<input type="checkbox"/>	Accesible a la información



<input type="checkbox"/>	Interfaces “Cómodas”
<input type="checkbox"/>	Disponibilidad para múltiples dispositivos
<input type="checkbox"/>	Letras, botones grandes y colores

9. ¿Está familiarizado con los conceptos de “Arrastrar” y “Soltar” en las TI?

<input type="checkbox"/>	Si
<input type="checkbox"/>	No
<input type="checkbox"/>	Un poco

10. Describan sus propias palabras las funciones que considera que debería de poseer una aplicación para la asignación de horario.

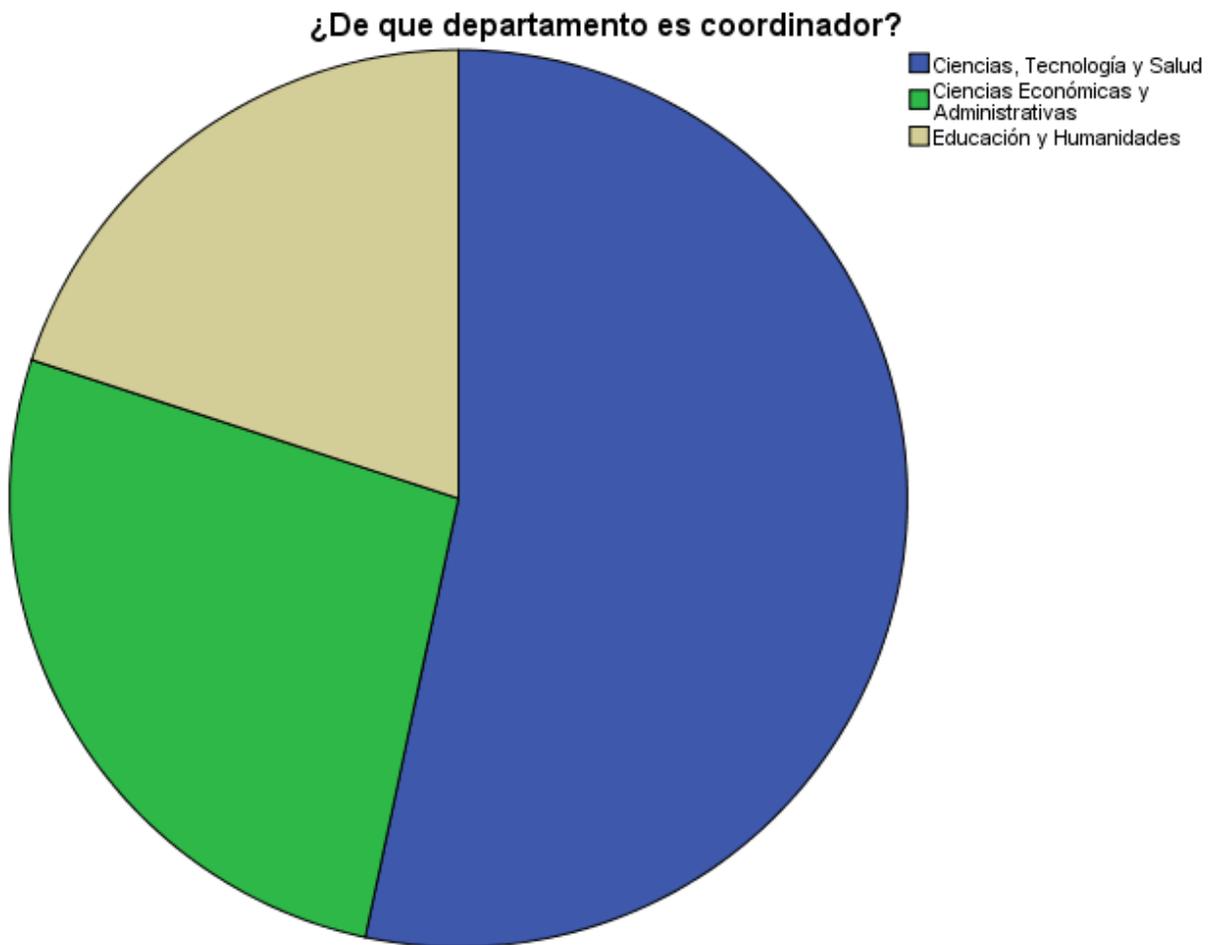
11. Al existir la aplicación para generar horario, ¿Estaría dispuesto a usarla?

<input type="checkbox"/>	Si
<input type="checkbox"/>	No

12. Escriba algún punto que usted considera importante y que no se presentó en las preguntas anteriores.



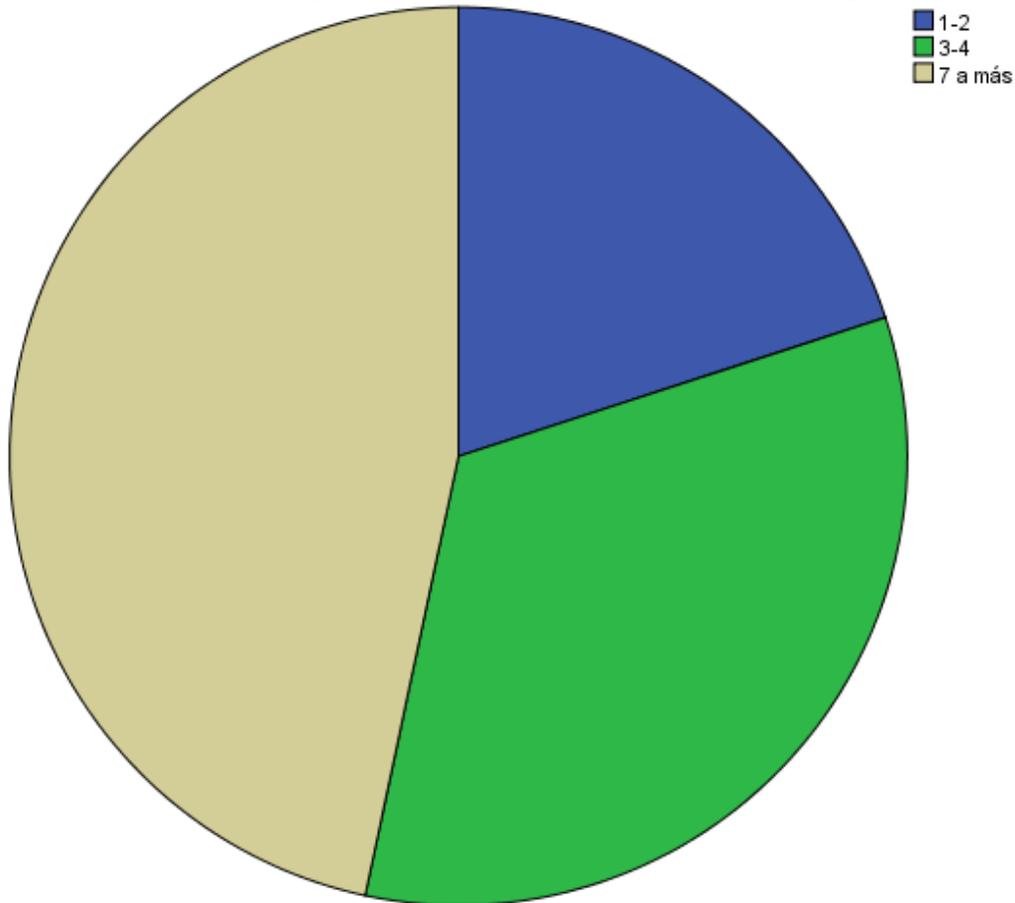
2.1 Resultados de las encuestas



De estos resultados se observa que la mayoría de los encuestados son coordinadores de una carrera de Ciencias Tecnología y salud, mientras que Ciencias económicas y administrativas en conjunto con Educación y humanidades comparten una cantidad similar de encuestados.



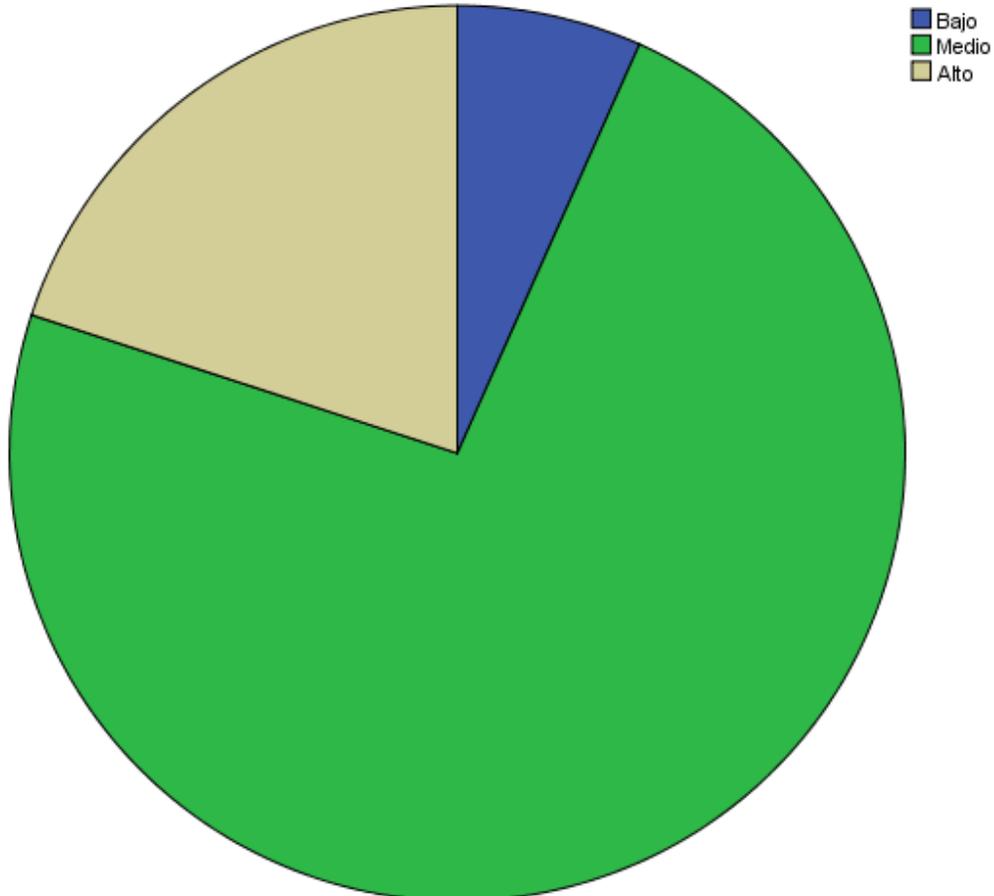
¿Cuanto tiempo ha estado en su cargo?



Más del 46% de los encuestados han permanecido en su cargo por al menos 7 años lo que representa una ventaja en los resultados de la encuesta por lo que sabemos que los datos serán obtenidos mayormente representativos dado que han sido obtenidos con información brindada por personas con experiencia.



¿Cómo evalúa el nivel de complicación de la asignación de Horarios?

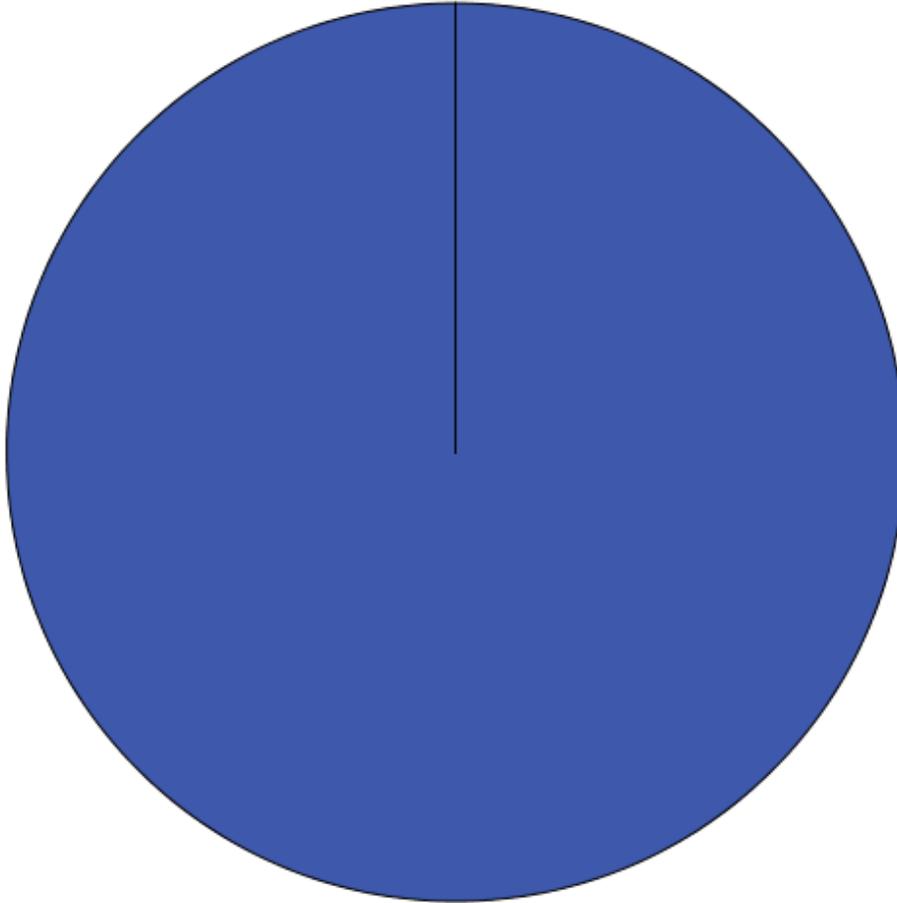


Según el 73% de los encuestados La asignación de horarios de forma manual representa una tarea de nivel de dificultad media, el 20 % afirma ser una tarea de gran dificultad y la mínima representada por el 7% concluyen es que es una tarea fácil de realizar.



¿Considera usted que es necesario la automatización de este proceso?

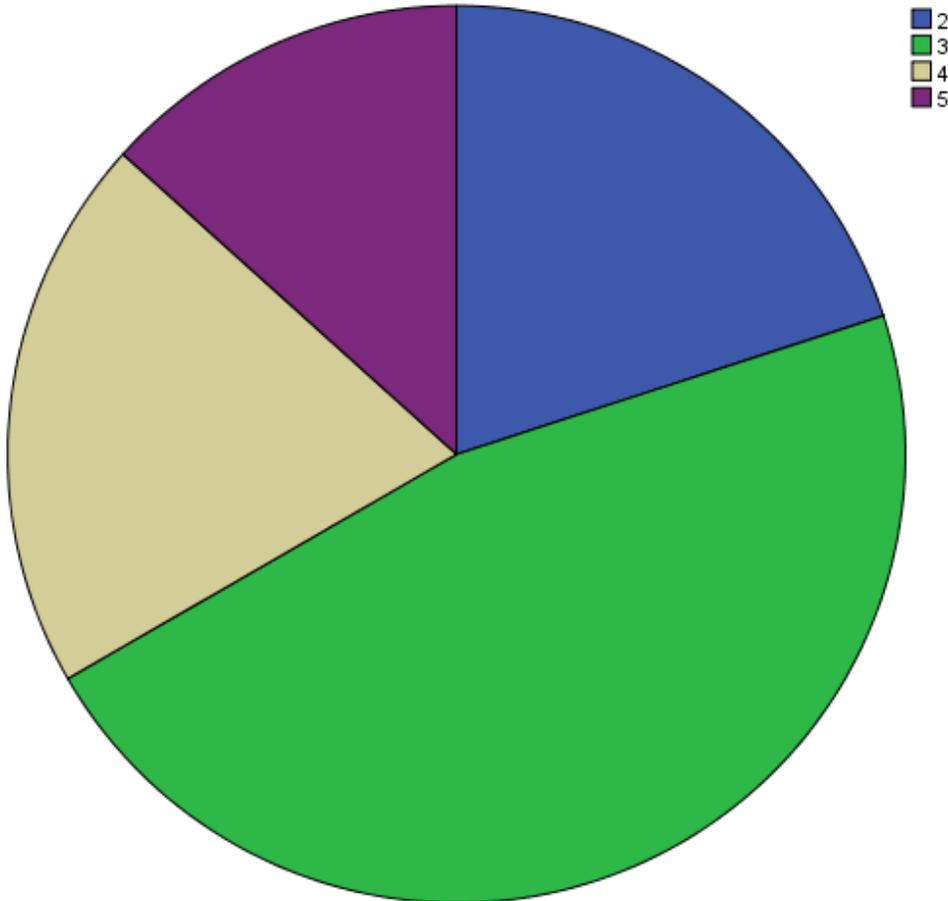
■ Si



Por opinión unánime se ha concluido que es necesario la automatización del proceso de asignación de horarios.



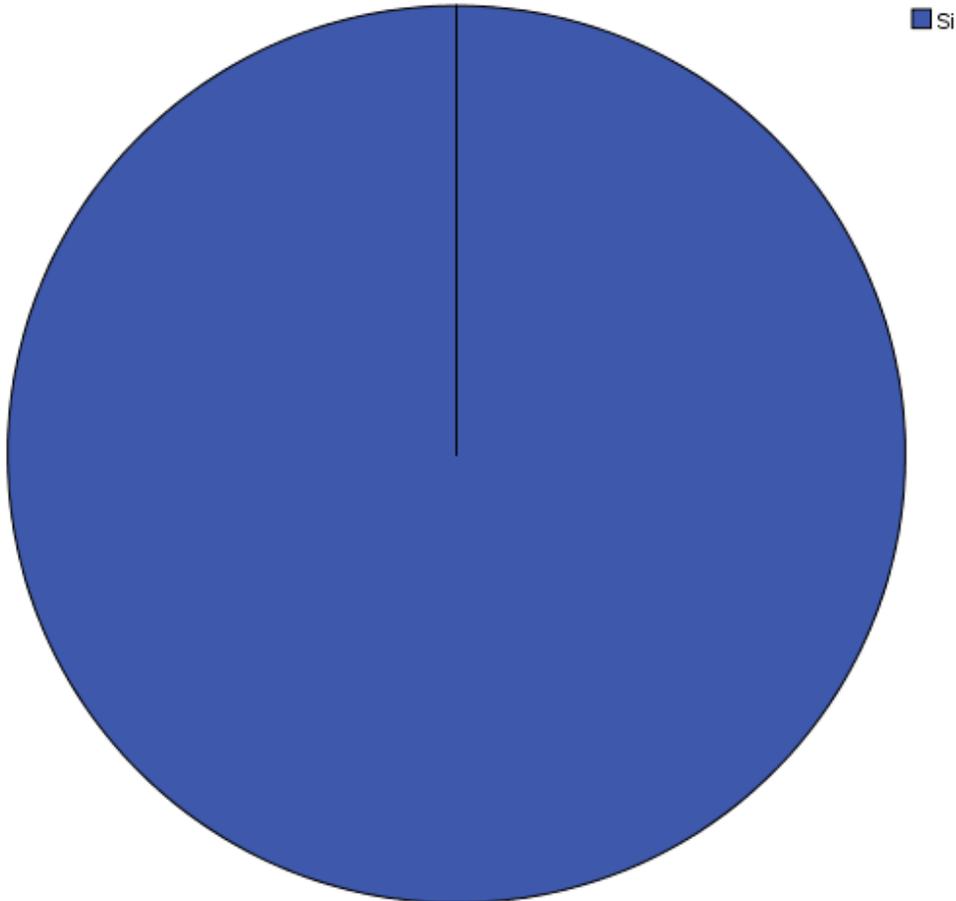
¿Cuál considera usted que es su nivel de control sobre las TIC?



Siendo 1 el nivel menos hábil y 5 el más hábil, se concluye que el 80% de los docentes coordinadores encargados de asignar horarios a quienes se considera futuros usuarios del sistema, poseen las habilidades necesarias para realizar sus tareas haciendo uso de esta herramienta informática; El otro 20% se entiende que necesitan de capacitación para el uso de la aplicación.



¿Estaría usted dispuesto a mejorar su nivel de manejo de TIC?



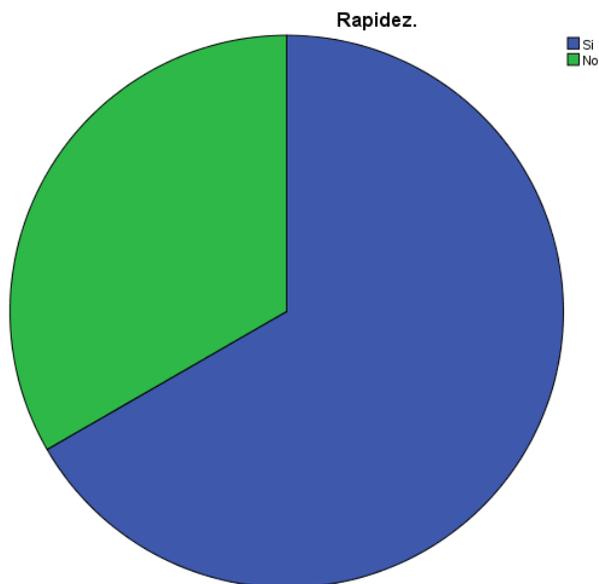
El 100% de los usuarios a los que se orienta el sistema están dispuestos a mejorar sus habilidades de uso de las tecnologías de información y comunicaciones como manera de apoyar la implementación del sistema.



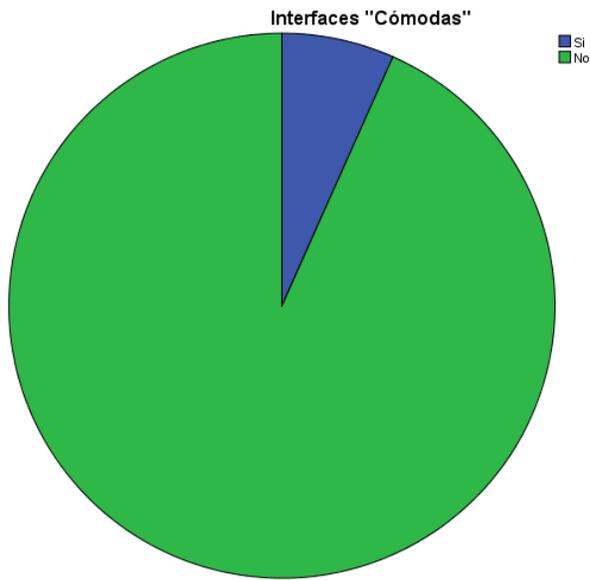
Las siguientes características representan conocimiento de los usuarios y la necesidad de la implementación de estas en el sistema.



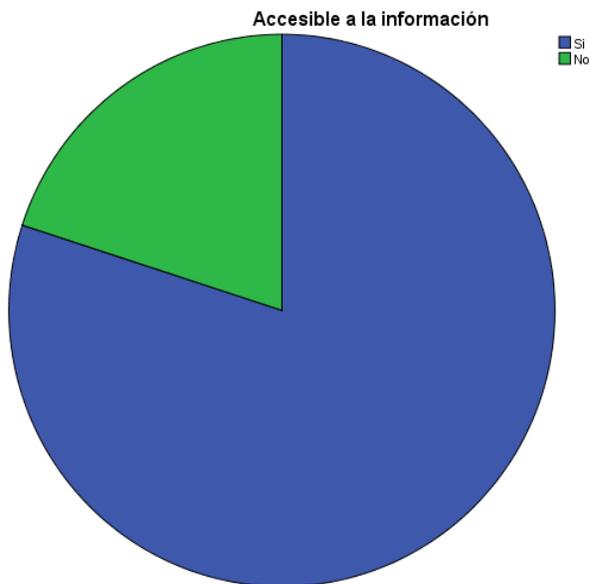
73% de los coordinadores conocen esta característica y la consideran necesaria al momento de implementar el sistema.



67% de los coordinadores conocen esta característica y la consideran necesaria al momento de implementar el sistema.



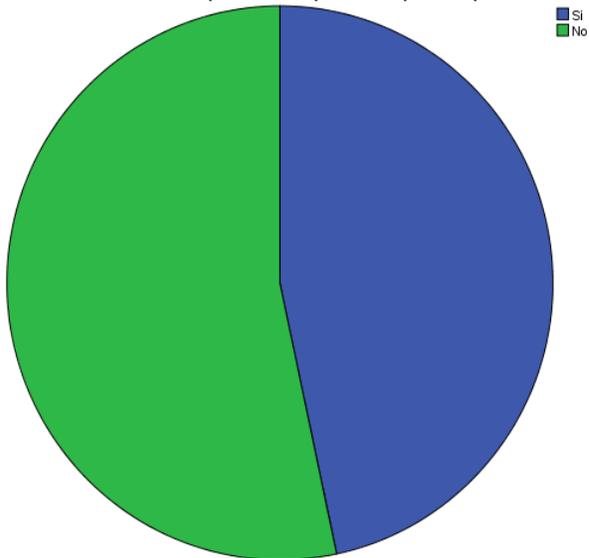
93% de los coordinadores desconocen esta característica o no la consideran necesaria al momento de implementar el sistema.



80% de los coordinadores conocen esta característica o la consideran necesaria al momento de implementar el sistema.

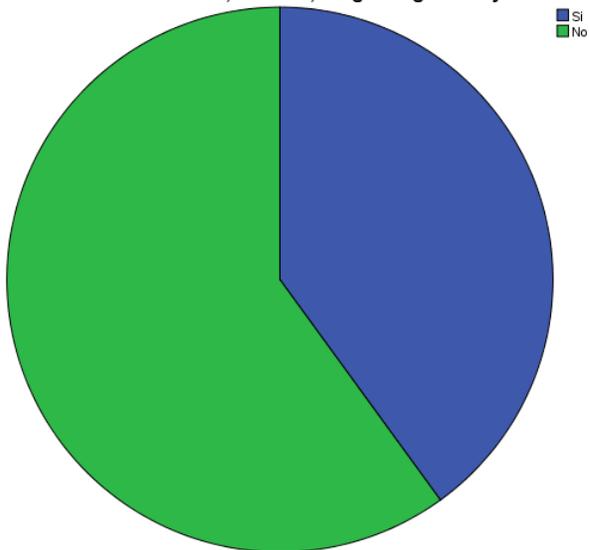


Disponibilidad para múltiples dispositivos.



53% de los coordinadores desconocen esta característica o no la consideran necesaria al momento de implementar el sistema.

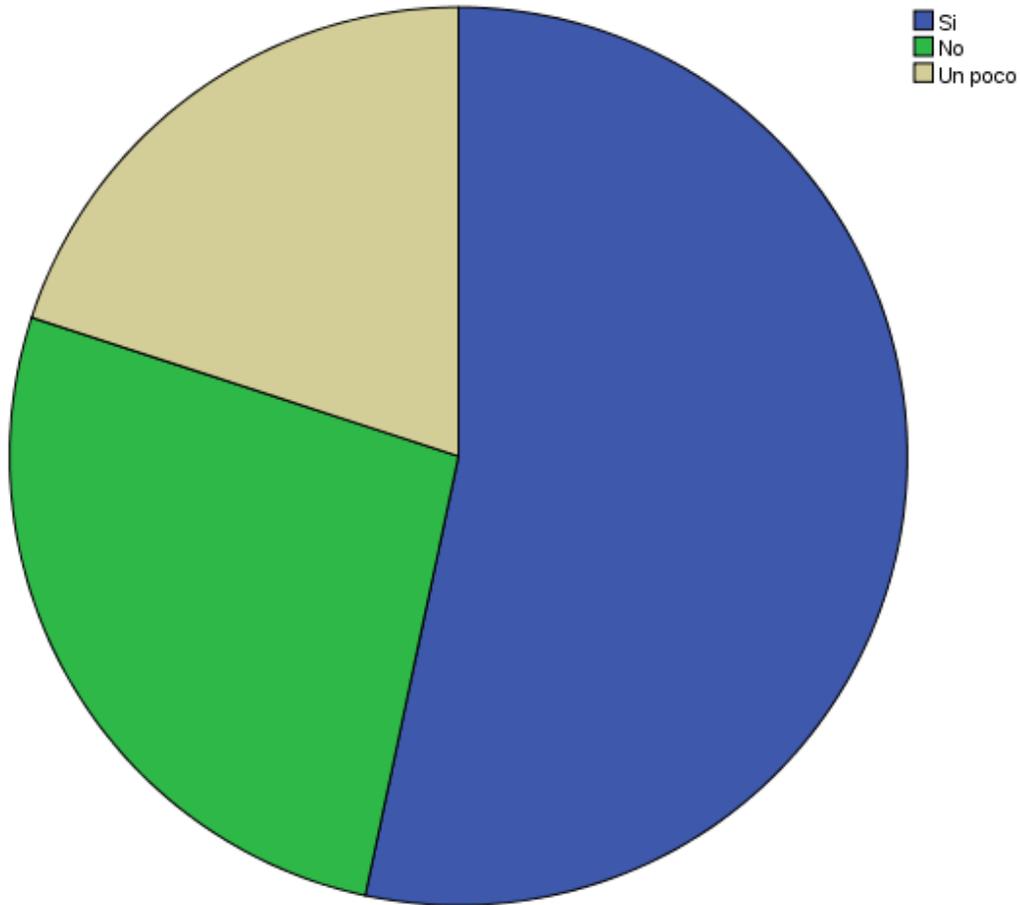
Lestras, botones, imágenes grandes y colores.



60% de los coordinadores conocen esta característica o la consideran necesaria al momento de implementar el sistema.



¿Está familiarizado con los conceptos de "Arrastrar" y "Soltar" en las TI?

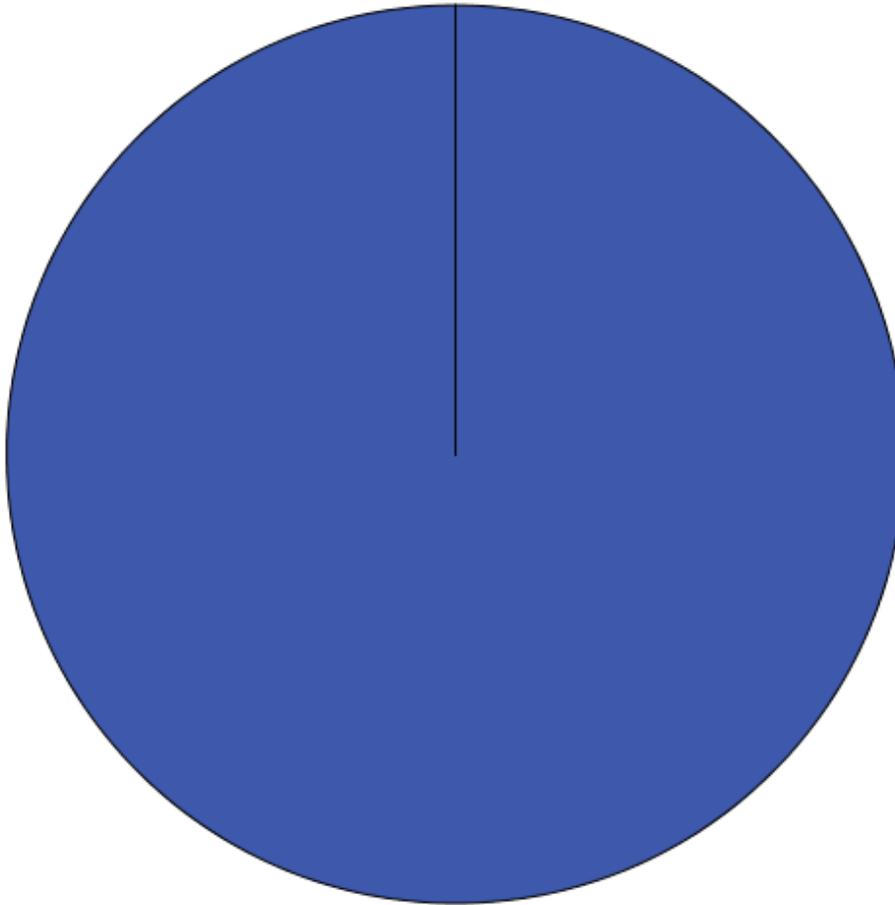


El 73% de los coordinadores tienen una noción del concepto "Arrastrar y Soltar", el cual es una de las características más emblemáticas del sistema.



Al existir la aplicación para generar horario, ¿Estaría dispuesto a usarla?

■ Si



El 100% de los profesores encuestados están dispuestos a hacer uso de una aplicación que automatice la asignación de horarios en la universidad.

De los resultados obtenidos se puede concluir que el personal encargado de generar el horario de la UNAN Managua FAREM chontales presenta las mismas dificultades y no se encuentran renuentes a una aplicación la cual este orientada a la automatización de dicho proceso, por lo cual además de ser viable, el sistema propuesto es necesario y el personal muestra actitudes positivas con respecto a la implementación y uso del mismo.

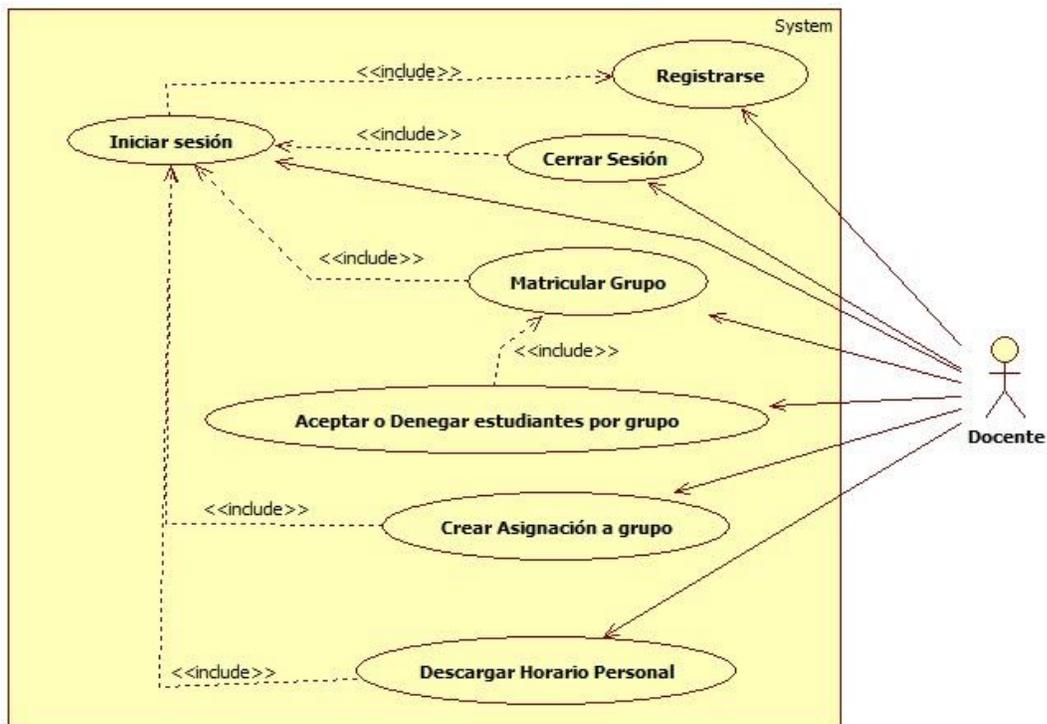


3. Casos de usos

Un diagrama de casos de uso es una forma de diagrama de comportamiento UML mejorado

3.1 Caso de uso usuario docente

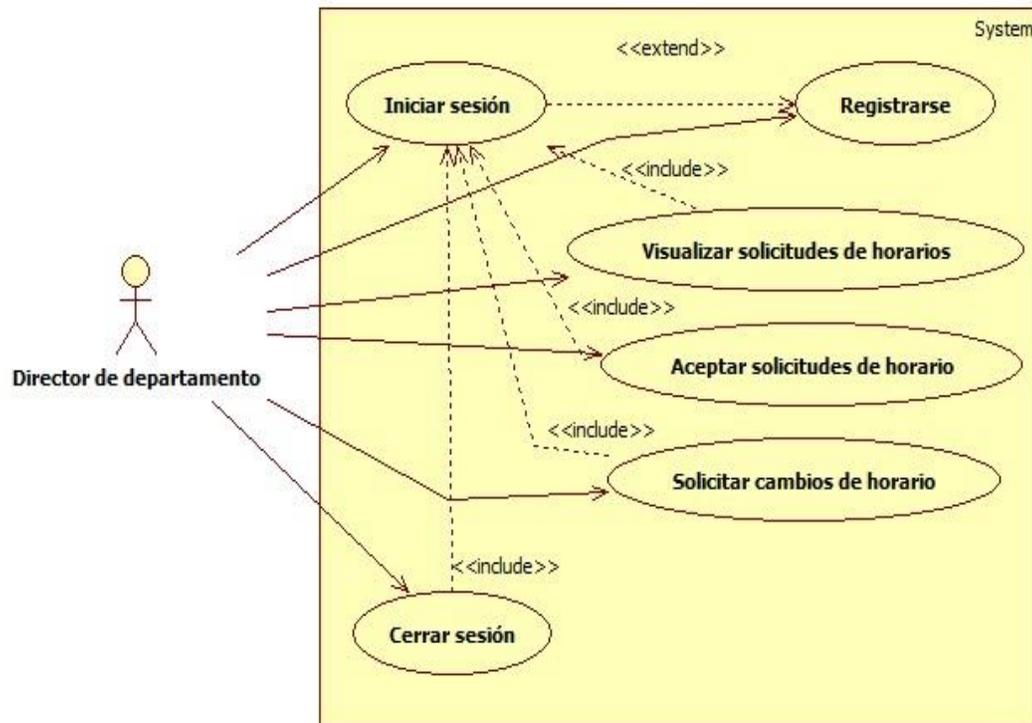
El docente podrá iniciar sesión o registrarse, dentro del sistema realizará las siguientes funciones: matricular grupo, aceptar o denegar estudiante por grupo, crear asignación a grupo y descargar el horario personal.





3.2 Caso de uso usuario director de departamento

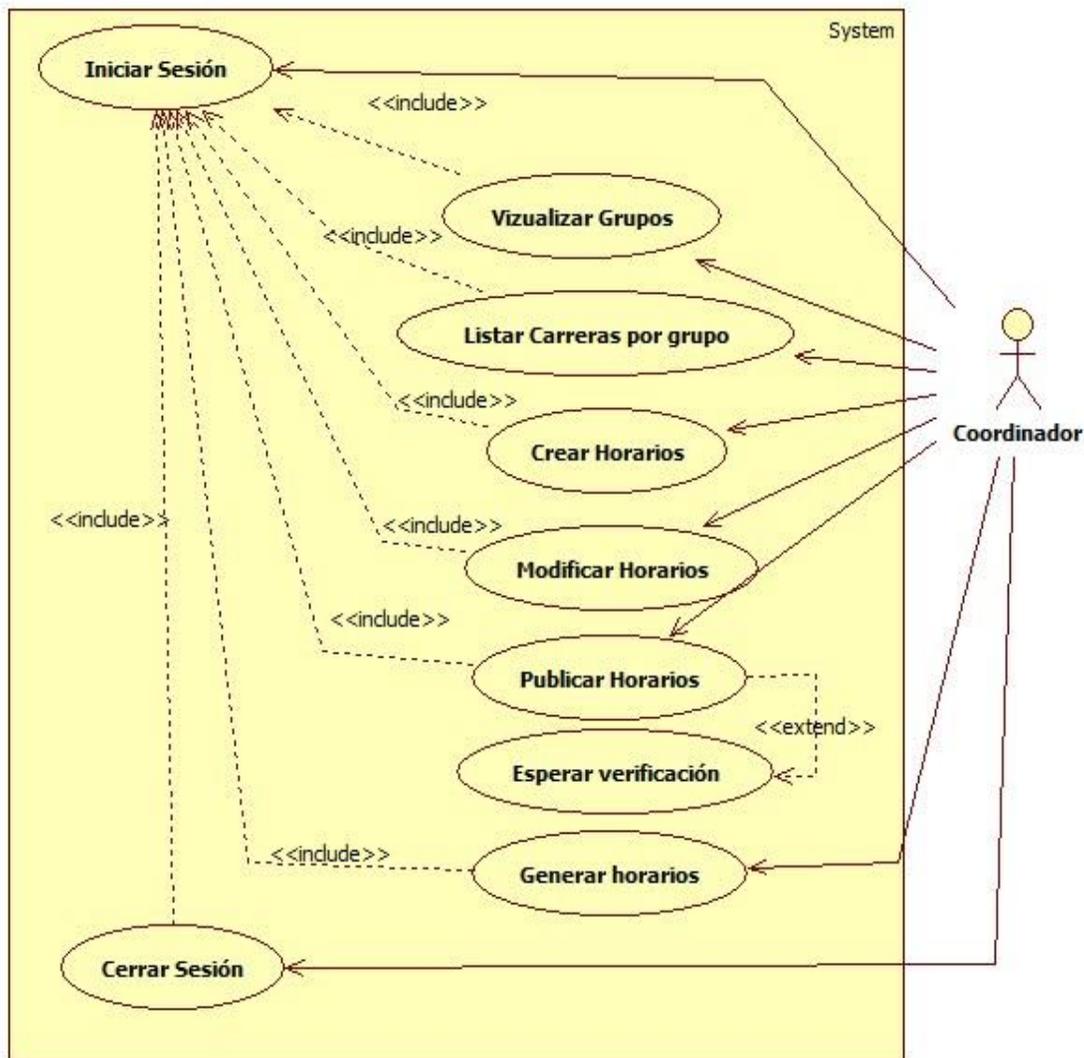
Cuando el director de departamento este registrado puede acceder, si no se tiene que registrar cuando entre en el sistema podrá visualizar las solicitudes de horario, aceptar solicitudes de horario, solicitar cambios de horario y cerrar sesión cuando lo amerite.





3.3 Caso de uso usuario coordinador

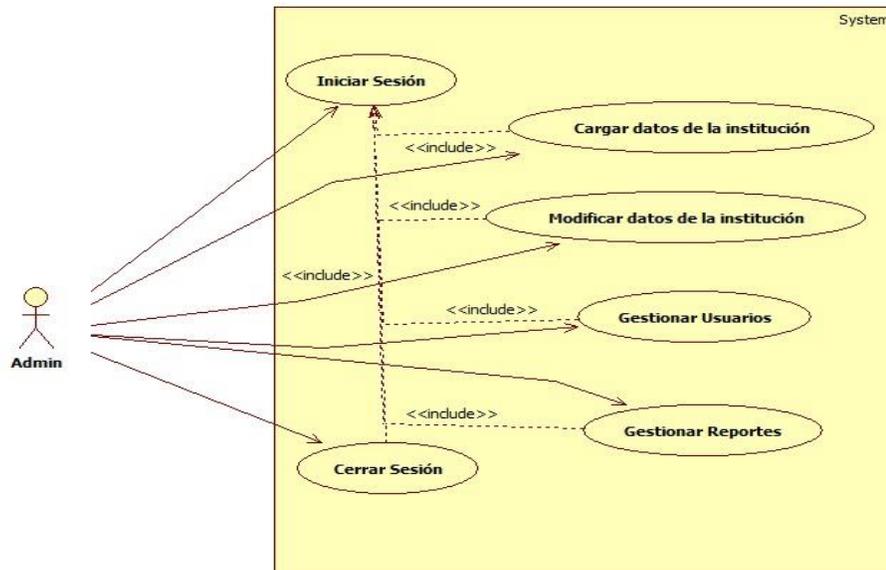
Usuario coordinador una vez registrado, puede hacer los siguientes funciones: visualizar los grupos, listar carreras por grupo, crear horario, modificar horario, publicar horario, esperar horario, esperar verificación, generar sesión o cerrar sesión.





3.4 Caso de uso usuario administrador

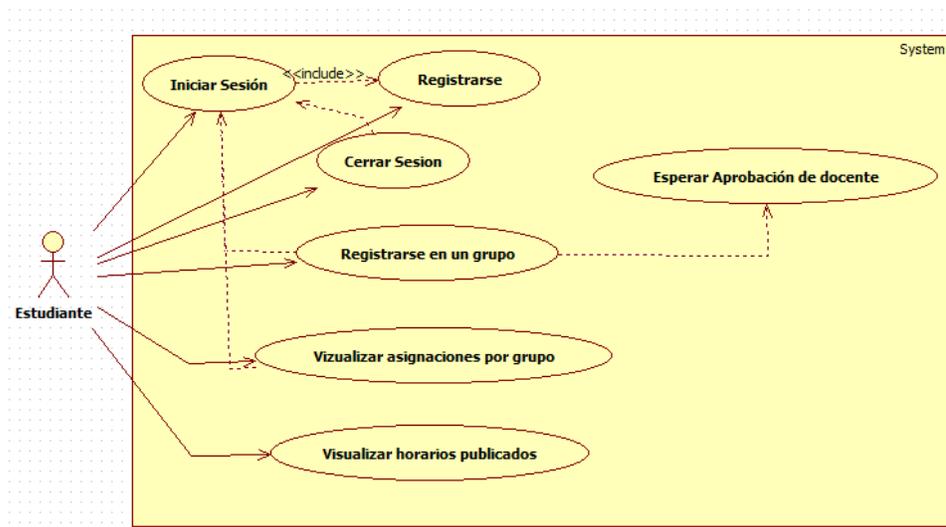
El usuario administrador se encargará de cargar datos de la institución, modificar datos de la institución, gestionar usuarios, gestionar reportes y cerrar sesión.





3.5 Caso de uso usuario estudiante

El usuario estudiante podrá iniciar después de haberse registrado y posteriormente registrarse en un grupo, esperar aprobación de docente, visualizar asignaciones por grupo, visualizar horarios publicados.

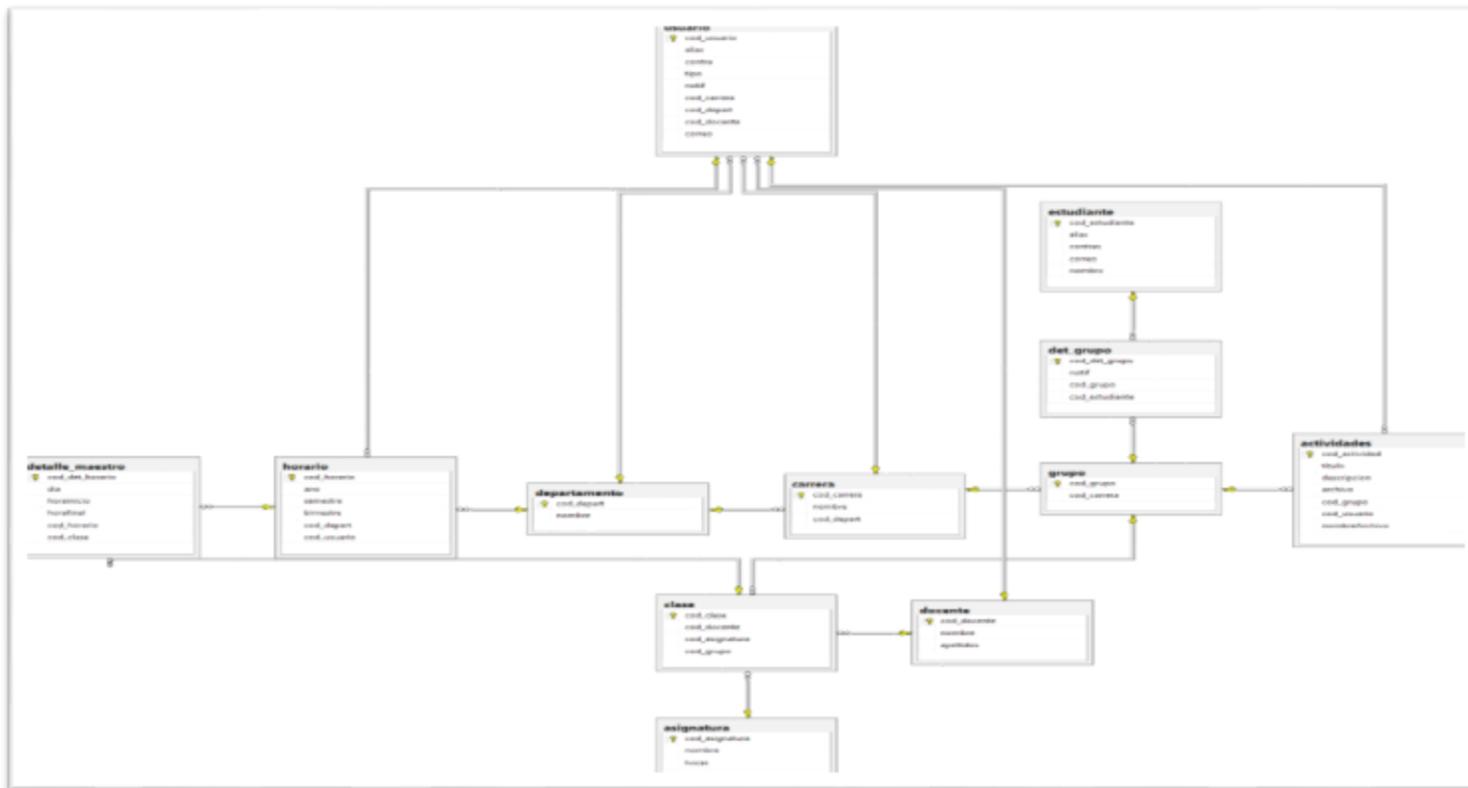




4. Base de datos

Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso

4.1 Diagrama de entidad/relación





4.2 Diccionario de datos

Tablas:

Tabla dbo.actividades (6 columna)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_actividad	int	X		
	Titulo	varchar(30)			
	Descripción	text			
	Archivo	binary(50)			
FK	cod_grupo	int			
FK	cod_usuario	int			
	nombreArchivo	text			

Indexes:

PK_actividades (Primary Key) (Clustered)

cod_actividad

Relaciones:

dbo.grupo (cod_grupo)

dbo.usuario (cod_usuario)

Tablas dbo.asignatura (4 columna)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_asignatura	varchar(6)			
	Nombre	text			
	Horas	int			

Indexes:

PK_asignatura (Primary Key) (Clustered)

5. cod_asignatura

Relaciones:

dbo.clase (cod_asignatura)

Tabla dbo.carrera (2 columna)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_carrera	varchar(10)			
	Nombre	text			



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



FK	cod_depart	int			
----	------------	-----	--	--	--

Indexes:

PK_carrera (Primary Key) (Clustered)

cod_carrera **Relaciones:**

dbo.departamento (cod_depart)

Tabla **dbo.clase** (4 columna)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_clase	int	X		
FK	cod_docente	int			
FK	cod_asignatura	varchar(6)			
FK	cod_grupo	int			

Indexes:

PK_clase (Primary Key) (Clustered)

cod_clase **Relaciones:**

dbo.asignatura (cod_asignatura) **dbo.docente** (cod_docente)

dbo.grupo (cod_grupo)

Tabla **dbo.departamento** (4 columna)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_depart	int	X		
	nombre	text			

Indexes:

PK_departamento (Primary Key) (Clustered)

cod_depart

Tabla **dbo.det_grupo** (4 columnas)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_det_grupo	int			
	Notif	bit			



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



FK	cod_grupo	int			
FK	cod_estudiante	int			

Indexes:

PK_det_grupo (Primary Key) (Clustered) cod_det_grupo

Relaciones:

dbo.estudiante (cod_estudiante)

dbo.grupo (cod_grupo)

Tabla dbo.detalle_maestro (6 columnas)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_det_horario	int	X		
	Dia	smallint			
	Horainicio	time			
	Horafinal	time			
FK	cod_horario	int			
FK	cod_clase	int			

Indexes:

PK_detalle_maestro (Primary Key) (Clustered)

cod_det_horario **Relaciones:**

dbo.clase (cod_clase)

dbo.horario (cod_horario)

Tabla dbo.docente (9 columnas)

Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto	Columna
PK	cod_docente	int	X		
	Nombre	text			
	Apellidos	text			

Indexes:

PK_docente (Primary Key) (Clustered)

cod_docente



Tabla dbo.estudiante (5 columnas)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_estudiante	int	X		
	alias	varchar(15)			
	contras	varchar(15)			
	correo	text			
	nombre	text			

Indexes:

PK_estudiante (Primary Key) (Clustered)

cod_estudiante

Tabla dbo.grupo (2 columnas)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_grupo	int	X		
FK	cod_carrera	varchar(10)			

Indexes:

PK_grupo (Primary Key) (Clustered)

cod_grupo **References:**

dbo.carrera (cod_carrera)

Table dbo.horario (6 columnas)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_horario	int	X		
	ano	int			
	semestre	bit			
	bimestre	bit			
FK	cod_depart	int			
FK	cod_usuario	int			

Indexes:



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



PK_horario (Primary Key) (Clustered)

cod_horario

Relaciones:

dbo.departamento (cod_depart)

dbo.usuario (cod_usuario)

Tabla dbo.usuario (9 columnas)

	Columna	Tipo de Datos	Identificatoria	Permitir valor nulo	Valor por defecto
PK	cod_usuario	int	X		
	Alias	varchar(15)			
	Contra	varchar(15)			
	Tipo	int			
	Notif	int			
FK	cod_carrera	varchar(10)			
FK	cod_depart	int			
FK	cod_docente	int			
	Correo	text		X	

Indexes:

PK_usuario (Primary Key) (Clustered)

cod_usuario **Relaciones:**

dbo.carrera (cod_carrera) **dbo.departamento** (cod_depart)

dbo.docente (cod_docente)



Procedimientos almacenados:

Procedimiento dbo.aceptarGrupo

Parámetros	Tipo de datos	Defecto	Salida
@notif	bit		
@cod_grupo	int		
@cod_estudiante	int		

Procedimiento dbo.aceptarUsuario

Parámetros	Tipo de datos	Defecto	Salida
@id	int		
@notif	int		

Procedimiento dbo.asignarHorario

Parámetros	Tipo de datos	Defecto	Salida
@dia	bit		
@horainicio	time		
@horafinal	time		
@cod_horario	int		
@cod_clase	int		

Procedimiento dbo.borarDetalleHorario

Parámetros	Tipo de datos	Defecto	Salida
@cod_det_horario	int		

Procedimiento dbo.borrarActividad

Parámetros	Tipo de datos	Defecto	Salida
@cod	int		

Procedimiento dbo.cambiarDatosUsuario

Parámetros	Tipo de datos	Defecto	Salida
@id	int		
@alias	varchar(15)		
@passwr	varchar(15)		
@corr	text		

Procedimiento dbo.cambiarUsuario

Parámetros	Tipo de datos	Defecto	Salida
@id	int		



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



@alias	varchar(15)		
@contra	varchar(15)		

Procedimiento dbo.cargarActividadesGrupo

Parámetros	Tipo de datos	Defecto	Salida
@codgrupo	int		

Resultado

Columna	Tipo de dato	Null
cod_actividad	int	
Titulo	varchar(30)	
Descripción	text	
Archivo	binary(50)	
cod_grupo	int	
cod_usuario	int	
nombreArchivo	text	

Procedimiento dbo.codDocente

Parámetros	Tipo de datos	Defecto	Salida
@codUsuario	int		

Resultado:

Column	Data Type	Nullable
cod_docente	int	

Procedimiento dbo.comprobar_asignacion

Parámetros	Tipo de datos	Defecto	Salida
@cod_docente	int		
@cod_asignatura	varchar(6)		
@horainicio	time		
@horafinal	time		
@dia	smallint		
@cod_grupo	int		

Procedimiento dbo.crearActividad

Parámetros	Tipo de datos	Defecto	Salida
------------	---------------	---------	--------



@titulo	varchar(30)		
@descripcion	text		
@archivo	binary(1)		
@codgrupo	int		
@codusuario	int		
@nombrear	text		

Procedimiento dbo.crearAsignaura

Parámetros	Tipo de datos	Defecto	Salida
@cod	varchar(6)		
@nombre	text		
@horas	int		

Procedimiento dbo.crearCarrera

Parámetros	Tipo de datos	Defecto	Salida
@cod_carrera	varchar(10)		
@nombre	text		
@cod_departamento	int		

Procedimiento dbo.crearClase

Parámetros	Tipo de datos	Defecto	Salida
@codDocente	int		
@codAsignatura	varchar(6)		

Procedimiento dbo.crearDepartamento

Parámetros	Tipo de datos	Defecto	Salida
@nombre	text		

Procedimiento dbo.crearDetalleMaestro

Parámetros	Tipo de datos	Defecto	Salida
@dia	smallint		
@horainicio	time		
@horafinal	time		
@cod_horario	int		
@cod_clase	int		

Procedimiento dbo.crearDocente



Parámetros	Tipo de datos	Defecto	Salida
@nombre	text		
@apellido	text		

Procedimiento dbo.crearEstudiante

Parámetros	Tipo de datos	Defecto	Salida
@alias	varchar(15)		
@contras	varchar(15)		
@nombre	text		
@correo	text		

Procedimiento dbo.crearGrupo

Parámetros	Tipo de datos	Defecto	Salida
@cod_carrea	varchar(10)		

Procedimiento dbo.crearHorario

Parámetros	Tipo de datos	Defecto	Salida
@ano	int		
@semestre	bit		
@bimestre	bit		
@coddepartamento	int		
@codusuario	int		

Procedimiento dbo.crearUsuario

Parámetros	Tipo de datos	Defecto	Salida
@usr	varchar(15)		
@pass	varchar(15)		
@tipo	int		
@codcar	varchar(10)		
@coddept	int		
@coddoc	int		
@email	text		

Procedimiento dbo.datosUsuarioDocente



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Parámetros	Tipo de datos	Defecto	Salida
@user	int		

Resultado:

Columna	Tipo de dato	Null
cod_usuario	int	
Alias	varchar(15)	
Contra	varchar(15)	
Tipo	int	
Notif	int	
cod_carrera	varchar(10)	
cod_depart	int	
cod_docente	int	
Correo	text	X
cod_docente	int	
Nombre	text	
Apellidos	text	

Procedimiento dbo.horarioDocente

Parámetros	Tipo de datos	Defecto	Salida
@codDocente	int		

Resultado:

Columna	Tipo de dato	Null
Maestro	varchar(max)	
Asignatura	text	
Grupo	int	
Dia	smallint	
Inicio	time	
Final	time	

Procedimiento dbo.iniciar_sesion

Parámetros	Tipo de datos	Defecto	Salida
@alias	varchar(15)		



@contra	varchar(15)		
---------	-------------	--	--

Resultado:

Columna	Tipo de dato	Null
cod_usuario	int	
Nombre	text	
Apellidos	text	
Alias	varchar(15)	
Contra	varchar(15)	
Tipo	int	

Procedimiento dbo.listarAsignaturasDocentes

Parámetros	Tipo de datos	Defecto	Salida
@cod_grupo	int		

Resultado:

Columna	Tipo de dato	Null
cod_clase	int	
cod_grupo	int	
cod_docente	int	
cod_asignatura	varchar(6)	
Asignatura	text	
Horas	int	
Docente	varchar(max)	

Procedimiento dbo.listarCarreras

Sin parametros.

Resultado:

Columna	Tipo de dato	Null
cod_carrera	varchar(10)	
Nombre	text	
cod_depart	int	



Procedimiento dbo.listarDepartamentos

Sin parametro.

Resultado:

Columna	Tipo de dato	Null
cod_depart	int	
Nombre	text	

Procedimiento dbo.listarDocentes

Sin parametros.

Resultado:

Columna	Tipo de dato	Null
cod_docente	int	
Nombre	text	
Apellidos	text	

Procedimiento dbo.listarGruposPorCarrera

Parámetros	Tipo de datos	Defecto	Salida
@codCarrera	varchar(15)		

Resultado:

Columna	Tipo de dato	Null
cod_grupo	int	
cod_carrera	varchar(10)	

Procedimiento dbo.listarSolicitudUsuario

Sin parámetros.

Resultado:

Columna	Tipo de dato	Null
cod_usuario	int	
Nombre	text	



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Apellidos	text	
Departamento	text	
Carrera	text	
tipoUsuario	varchar(11)	X

Procedimiento dbo.modificarActividad

Parámetros	Tipo de datos	Defecto	Salida
@cod	int		
@titulo	varchar(30)		
@descripcion	text		
@archivo	binary(1)		
@codgrupo	int		
@codusuario	int		

Procedimiento dbo.modificarEstudiante

Parámetros	Tipo de datos	Defecto	Salida
@cod_estudiante	int		
@alias	varchar(15)		
@contras	varchar(15)		
@nombre	text		
@correo	text		

Procedimiento dbo.modificarUsuario

Parámetros	Tipo de datos	Defecto	Salida
@id	int		
@alias	varchar(15)		
@contra	varchar(15)		
@tipo	int		
@notif	bit		
@correo	text		

Procedimiento dbo.obtenerCodDocente



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Parámetros	Tipo de datos	Defecto	Salida
@codUsuario	int		

Resultado:

Columna	Tipo de dato	Null
cod_docente	int	

Procedimiento dbo.TotalSolicitudUsuario

Sin parametros.

Resultado:

Columna	Tipo de dato	Null
(Sin columna)	int	X



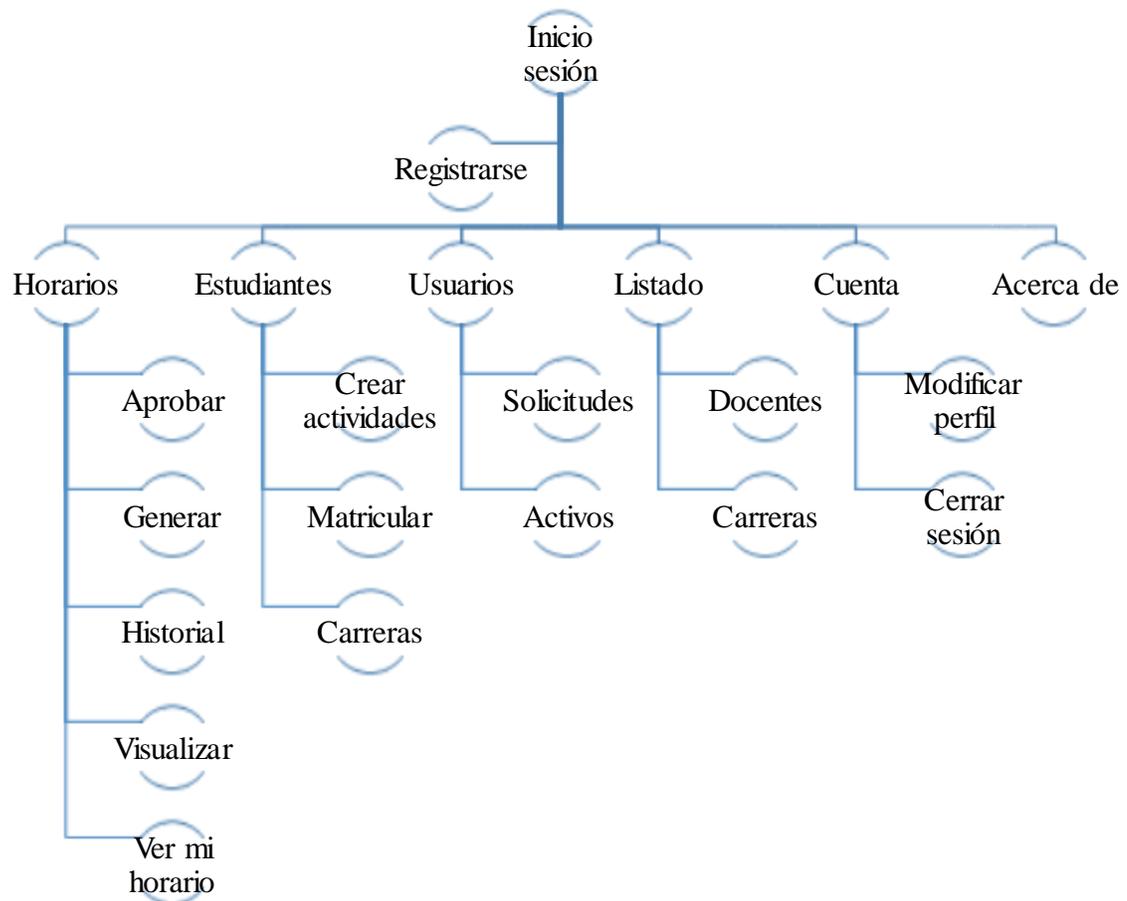
5. Mapa de navegación

Los mapas de navegación proporcionan una representación esquemática de la estructura del hipertexto, indicando los principales conceptos incluidos en el espacio de la información y las interrelaciones que existen entre ellos

5.1 Web

A continuación, se muestra de forma esquemática como está estructurado el sistema web, con el objetivo de orientar al usuario que acceda al software.

Figura 3 Mapa de navegación web

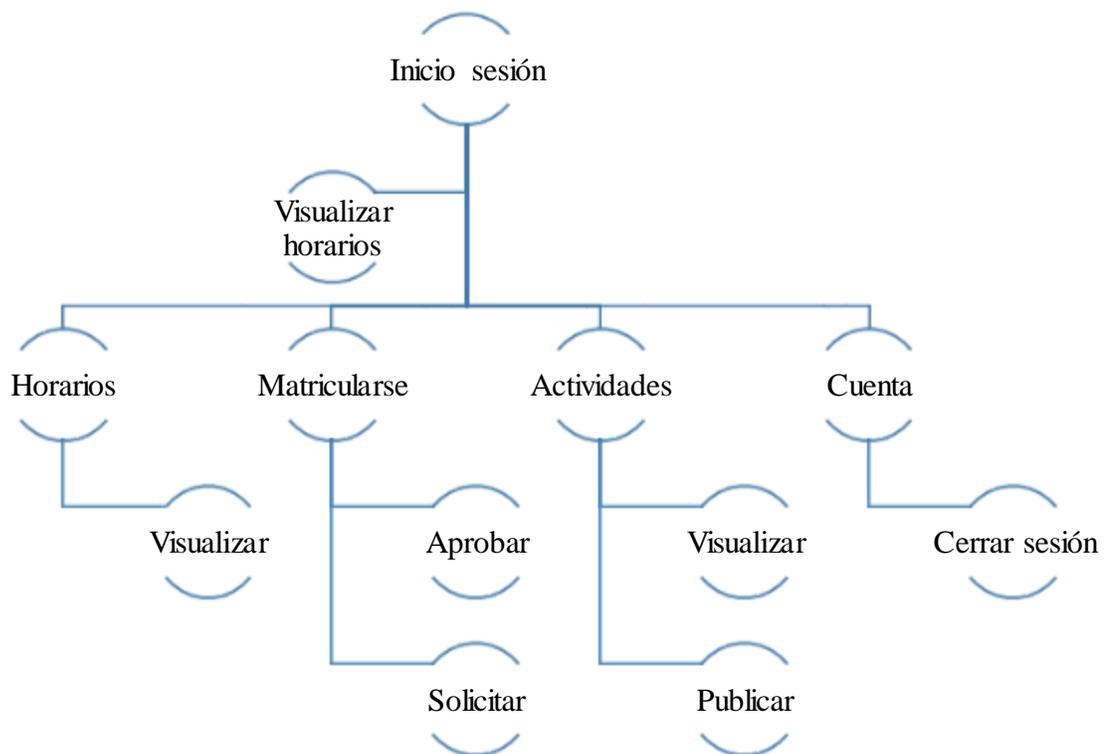




5.2 Móvil

El mapa de navegación expresa las vías de acceso de la interfaz del usuario principal a través de la aplicación.

Figura 4 Mapa de navegación móvil





6 Bocetos

Se refiere al esquema o al proyecto que sirve de bosquejo para cualquier obra.

6.1 Web

6.1.1 Pantalla de Inicio de sesión

El usuario tiene que ingresar sus credenciales (usuario y contraseña) para iniciar sesión o solicitar usuario si es necesario.

El boceto muestra una ventana de navegador con el título "Login" y la URL "http://www.unan.edu.ni/saho/login". En la esquina superior derecha hay un botón "Solicitar Usuario". En el centro de la página hay un icono cuadrado con puntos, seguido de dos campos de entrada etiquetados "Usuario" y "Contraseña", y un botón "Iniciar" debajo de ellos.

6.1.2 Pantalla de solicitud de usuario

Al dar click en solicitar usuario se le presentaran los siguientes campos que deben ser rellenados con sus respectivos datos



Solicitud de Usuario

Nombre

Apellido

Correo electrónico

Cargo

Departamento

Carrera

Iniciar

6.1.3 Página principal

En esta página principal se muestra el menú de navegación en el sistema el cual se puede tener acceso dependiendo su rol de usuario.

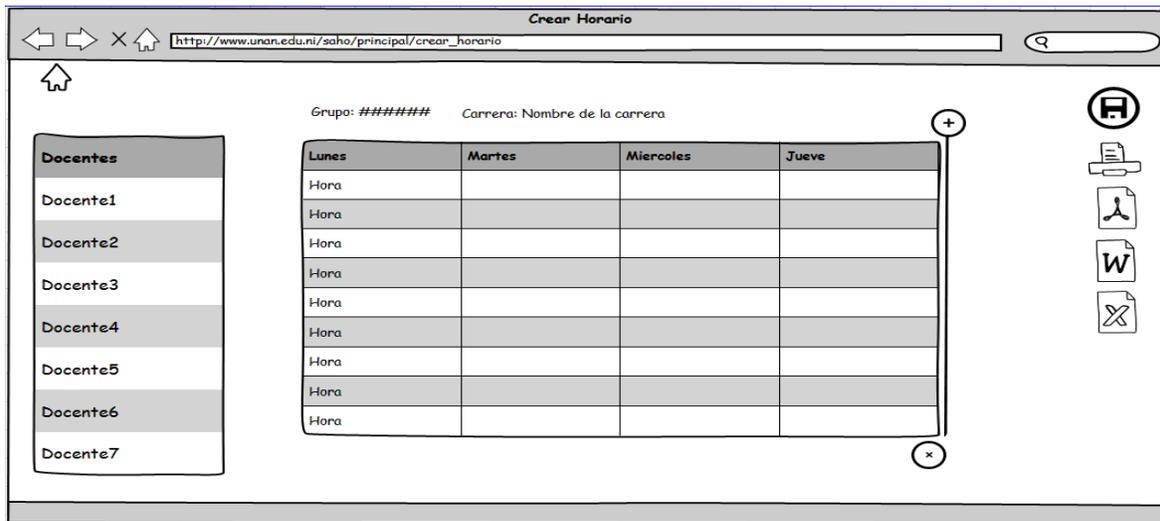
Principal

Grupos Asignaturas Docentes Horarios guardados

Bienvenido

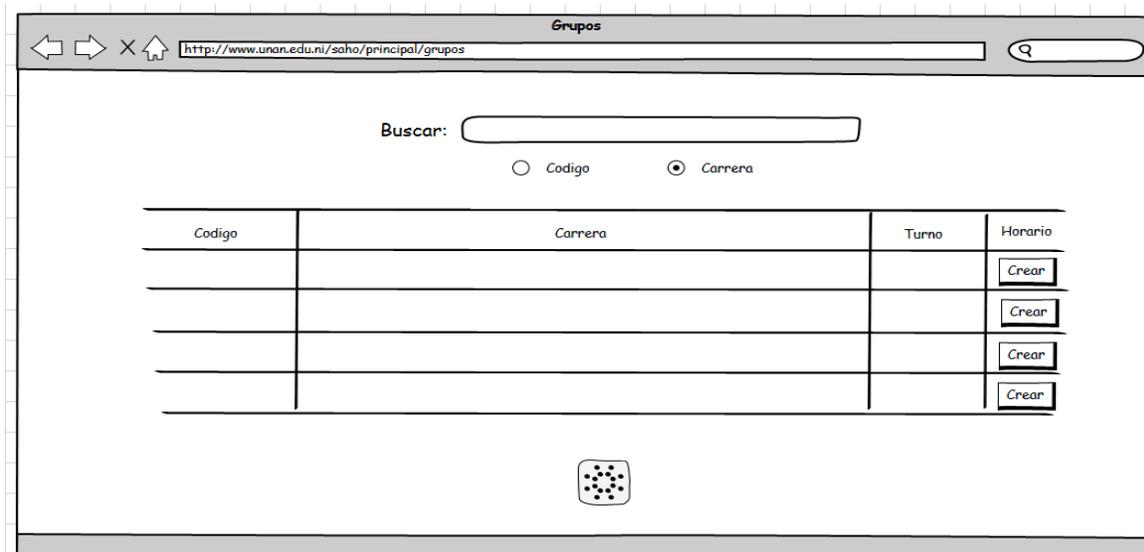
6.1.4 Pantalla de asignación de horario

Una vez creado el horario con la función de “arrastrar y soltar” se podrá guardar en distintos tipos de formatos.



6.1.5 Listado de grupos

En el listado de grupos, el usuario podrá realizar una búsqueda, ya sea por código o carrera.



6.1.6 Listado de carreras

En el listado de carreras el usuario podrá realizar una búsqueda por medio del código, nombre, o turno; en la liste se muestran datos de ejemplos.



Codigo	Nombre	Turno	Años existentes
#####	Blah Blah	Blah	#
#####	Blah Blah	Blah	#
#####	Blah Blah	Blah	#
#####	Blah Blah	Blah	#
#####	Blah Blah	Blah	#
#####	Blah Blah	Blah	#
#####	Blah Blah	Blah	#
#####	Blah Blah	Blah	#

6.1.7 Listado de docente

En el listado de docente se podrá realizar una búsqueda según el código o nombre, esto para hacerlo de una forma más rápida y eficiente

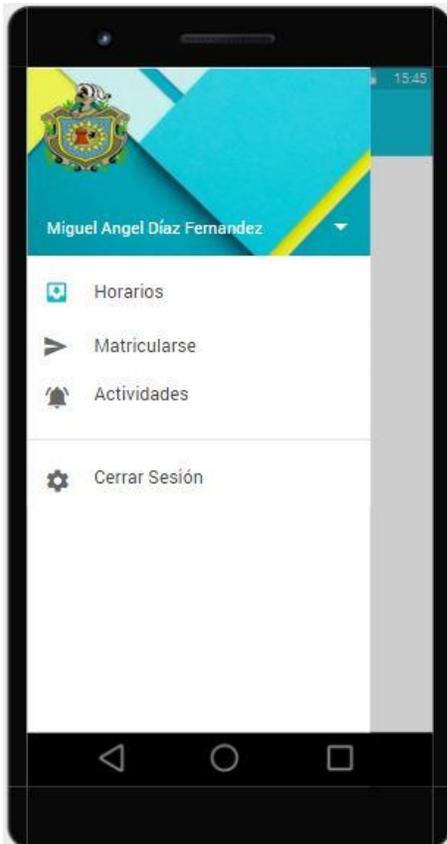
Codigo	Nombre	Apellidos	Departamento
#####	Blah Blah	Blah Blah	Blah Blah
#####	Blah Blah	Blah Blah	Blah Blah
#####	Blah Blah	Blah Blah	Blah Blah
#####	Blah Blah	Blah Blah	Blah Blah
#####	Blah Blah	Blah Blah	Blah Blah
#####	Blah Blah	Blah Blah	Blah Blah
#####	Blah Blah	Blah Blah	Blah Blah
#####	Blah Blah	Blah Blah	Blah Blah
#####	Blah Blah	Blah Blah	Blah Blah
#####	Blah Blah	Blah Blah	Blah Blah



6.2 Bocetos móviles

6.2.1 Pantalla principal

Una vez que el usuario haiga iniciado sesión se le mostrara un menú desplegable con el menú de navegación en donde podrá acceder a las diferentes funciones; o cerrar sesión se así lo prefiere.



6.2.2 Horario

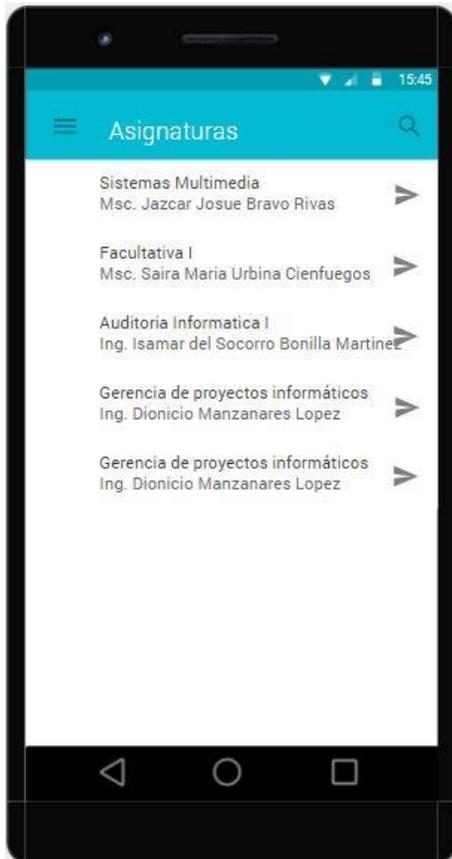
Los estudiantes podrán visualizar todos los horarios de las diferentes carreras.

	Lunes	Martes	Miercoles	Jueves
1:00-1:45	Sistemas Multimedia	Investigación Aplicada	Sistemas Multimedia	Investigación Aplicada
1:50-2:35				
2:40-3:25		Auditoria Informatica I		Auditoria Informatica I



6.2.3 Asignaturas

Las asignaturas se mostrarán según el grupo que haiga seleccionado el estudiante.



6.2.4 Asignaturas (mensaje de aviso)

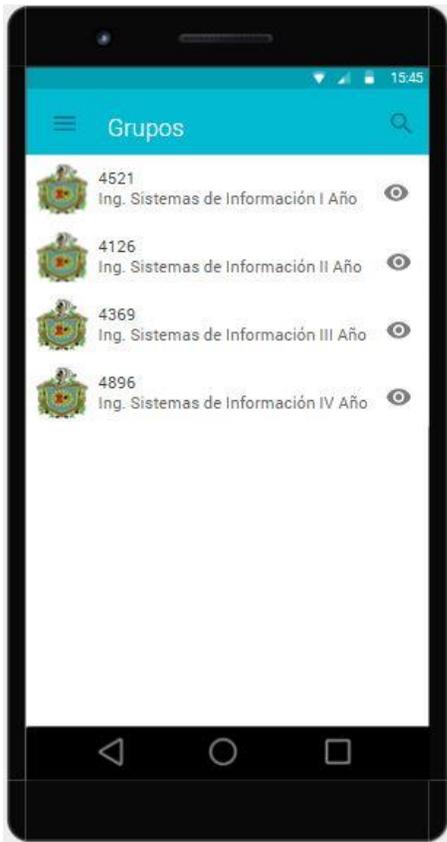
Cuando el alumno de click en matricular asignatura se le presentará un mensaje de aviso en donde podrá continuar la solicitud o cancelar.





6.2.5 Grupos

Se muestra los diferentes grupos en donde el estudiante podrá seleccionar al que pertenece.



6.2.6 Turnos de carrera

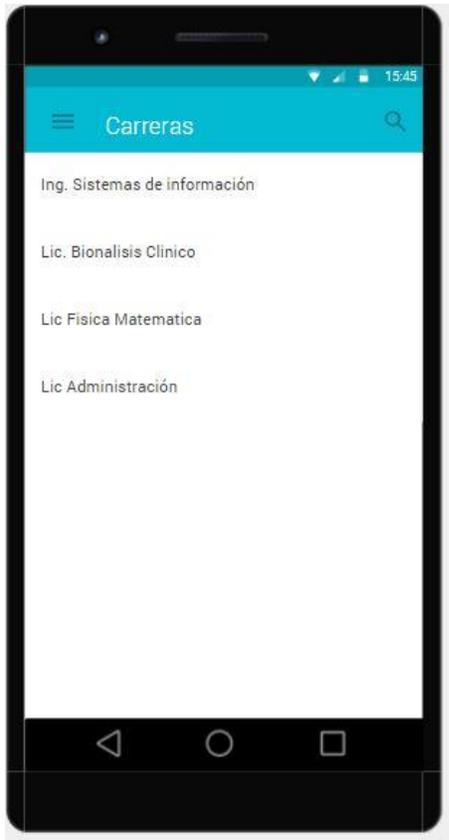
El estudiantes visualizará los diferentes turnos de las carreras.





6.2.7 Carreras

Se le mostrará un listado de todas las carreras.



6.2.8 Actividades

En las actividades se mostrarán aquellos puntos importantes según la asignatura en donde el alumno visualizará cada una de ellas.





13.1 SRS

(Especificación de requisitos del software)



Tabla de contenido

1	Introducción	1
1	Propósito	1
1.2	Alcance	1
1.3	Personal involucrado	1
1.4	Definiciones, acrónimos y abreviaturas	3
1.5	Referencias	3
1.6	Resumen	3
2	Descripción general	4
2.1	Perspectiva del producto	4
2.2	Funcionalidad del producto	4
2.3	Características de los usuarios	5
2.4	Restricciones	6
2.5	Suposiciones y dependencias	6
2.6	Evolución previsible del sistema	6
3	Requisitos específicos	7
3.1	Requerimientos Funcionales	7
3.2	Requerimientos no Funcionales	9
3.1	Requisitos comunes de los interfaces	11
3.1.1	Interfaces de usuario	11
3.1.2	Interfaces de hardware	11
3.1.3	Interfaces de software	12
3.1.4	Interfaces de comunicación	12
3.2	Requisitos funcionales	12
3.2.1	Requisito funcional 1	12
3.2.2	Requisito funcional 2	12
3.2.3	Requisito funcional 3	13
3.2.4	Requisito funcional 4	13
3.2.5	Requisito funcional 5	13
3.2.6	Requisito funcional 6	14



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



3.2.7 Requisito funcional 7	14
3.2.8 Requisito funcional 8	14
3.2.9 Requisito funcional 9	14
3.2.10 Requisito funcional 10	14
3.2.11 Requisito funcional 11	15
3.3 Requisitos no funcionales	15
3.3.1 Requisitos de rendimiento	15
3.3.2 Seguridad	15
3.3.3 Fiabilidad	16
3.3.4 Disponibilidad	16
3.3.5 Mantenibilidad	16
3.3.6 Portabilidad	16



1 Introducción

Mediante este documento se pretende dar a conocer las especificaciones de requisitos de software aplicadas al Sistema de asignación de horarios, el cual está orientado a la universidad UNAN FAREM-Chontales, el presente cumple con las normas establecidas por el estándar IEEE 830.

1.1 Propósito

Por medio de este documento se pretende definir de forma clara, concisa y practica los requerimientos funcionales y no funcionales, necesarios para desarrollo del sistema de asignación de horarios orientado a el personal docente, específicamente quienes son coordinadores de carrera y directores de departamento en la UNAN FAREM – Ch de la ciudad de Juigalpa durante el segundo semestre del año lectivo 2015.

1.2 Alcance

SAHO (Sistema de asignación de horarios) será un sistema web, el cual tendrá soporte para los navegadores de más alta demanda en el mercado y específicamente los utilizados en el área de asignación de horarios por los usuarios finales, SAHO será desarrollado con herramientas que faciliten y sobre todo agilicen el proceso de asignación de horarios.

1.3 Personal involucrado

Nombre	Jorge Inocente Cruz Serrano
Rol	Programador, Administrador, Diseñador
Categoría profesional	Bachiller
Responsabilidades	Diseñar, programar y administrar SAHO
Información de contacto	jorcruz2@gmail.com



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Aprobación	
Nombre	Miguel Díaz Fernández
Rol	Programador, Administrador, Diseñador
Categoría profesional	Bachiller
Responsabilidades	Diseñar, programar y administrar SAHO
Información de contacto	ruymi007@gmail.com
Aprobación	

Nombre	Rosa Marina Lumbí Suarez
Rol	Diseñar, programar y administrar SAHO
Categoría profesional	Bachiller
Responsabilidades	Diseñar, programar y administrar SAHO
Información de contacto	marinaluml1@hotmail.com
Aprobación	

Nombre	Miriam Patricia Téllez Marín
Rol	Docente Tutor
Categoría profesional	Master
Responsabilidades	Revisar los procesos de diseño y desarrollo de SAHO
Información de contacto	mirpatema@gmail.com
Aprobación	

Nombre	Saira Urbina Cienfuegos
Rol	Docente Tutor



Categoría profesional	Master
Responsabilidades	Revisar los procesos de diseño y desarrollo de SAHO
Información de contacto	sairacien5@hotmail.com
Aprobación	

1.4 Definiciones, acrónimos y abreviaturas

Concepto	Definición
UNAN	Universidad Nacional Autónoma de Nicaragua
FAREM	Facultad Regional Multidisciplinaria
CH/Ch	Chontales
Sistema	Herramienta de software
Software	Parte lógica de la computadora que permite la interacción de sus partes con el operador o usuario.
BD	Base de datos.
SAHO	Sistema de asignación de horarios

1.5 Referencias

Referencia	Título	Ruta	Fecha	Autor
1	IEEE830		1998	IEEE

1.6 Resumen

El presente documento describe y explica los requerimientos necesarios que deben ser cumplidos para el correcto desarrollo del software y su satisfactorio manejo por parte del usuario final, se da a conocer los detalles, así como su finalidad, sus restricciones y lo que el sistema en concreto hará y no hará, y sin exageradas tecnicidades se pretende entregar de modo satisfactorio un documento y un sistema que cumpla con



las normas que se plantean en el estándar IEEE 830 – 1998 y con las expectativas de usuarios directos e indirectos y de igual forma satisfaga las exigencias de evaluadores.

2 Descripción general

2.1 Perspectiva del producto

La aplicación, que no será una extensión del sistema central de la UNAN MANAGUA ni parte de el en ningún aspecto, obtendrá sin embargo información generada por el mismo, por lo que los datos encontrados serán en concreto los que existen en el sistema central y oficial, por lo cual el cometido de la aplicación se logrará sin mayor esfuerzo.

2.2 Funcionalidad del producto

La aplicación poseerá una funcionalidad sencilla e intuitiva, con métodos de arrastrar y soltar(Drag&Drop) para la asignación de docente y/o asignaturas en un calendario con formato de días sencillo y de tamaño favorable para ser administrado de forma intuitiva, de manera cómoda se podrá colocar maestros en sus clases y clases en el horario, dichas acciones siendo realizadas cómodamente mediante un navegador web, será posible obtener una lista de las asignaturas disponibles las cuales tienen los docentes que las imparten, se contara con niveles de usuario, notificaciones y revisiones de horarios por parte del director de departamento.



2.3 Características de los usuarios

Tipo de usuario	Director de departamento
Formación	Profesional
Habilidades	Técnicas básicas de uso de computadoras e internet
Actividades	Verificar horarios, aprobar o descartar horarios

Tipo de usuario	Coordinador de carrera
Formación	Profesional
Habilidades	Técnicas básicas de uso de computadoras e internet
Actividades	Generar horarios, visualizar horarios creados, visualizar los docentes y asignaturas disponibles por grupo y carrera.

Tipo de usuario	Docente
Formación	Profesional
Habilidades	Técnicas básicas de uso de computadoras e internet
Actividades	Visualizar horarios creados, Colgar asignaciones por grupos.

Tipo de usuario	Estudiante
Formación	Bachiller
Habilidades	Técnicas básicas de uso de computadoras e internet
Actividades	Visualizar horarios creados, visualizar las asignaciones por grupo.

Tipo de usuario	Administrador
-----------------	---------------



Formación	Profesional
Habilidades	Técnicas básicas de uso de computadoras e internet
Actividades	Asignar usuarios, administrar información, edición de datos

2.4 Restricciones

Desarrollo y Diseño

Plataforma web:

- Diseño con HTML5 y CSS3, Utilizando framework mdbootstrap.
- Programación con lenguaje C# y javascript.

Plataforma Android:

- Diseño con XML.
- Programación con lenguaje C# utilizando componente XAMARIN.
- Manejo de datos locales con sqlite.

Plataforma WindowsPhone:

- Diseño con XML.
- Programación con lenguaje C# utilizando componente XAMARIN.
- Manejo de datos locales con SQL server móvil(compact edition)

2.5 Suposiciones y dependencias

Dependencias para la plataforma web en base a los navegadores, navegadores no soportado o desactualizados podrían no permitir el desempeño correcto de la aplicación.

Dependencia para las plataformas móviles, solamente poseer un dispositivo compatible, estas plataformas no disponen del módulo de asignación de horarios.

2.6 Evolución previsible del sistema

Unión con el sistema central de la universidad UNAN FAREM Chontales



3 Requisitos específicos

3.1 Requerimientos Funcionales

Número de requisito	ReqF-01
Nombre de requisito	Presentar automáticamente las asignaturas y sus respectivos docente
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input checked="" type="checkbox"/> Alta /Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqF-02
Nombre de requisito	Después de generar horarios solicitar revisión y permitir guardar e imprimir
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqF-03
Nombre de requisito	Login de usuarios
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	-
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqF-04
Nombre de requisito	Obtener datos de formato generado por el sistema central de la UNAN MANAGUA
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Número de requisito	ReqF-05
Nombre de requisito	Solo puede ser usado por los coordinadores de carrera y directores de departamento
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqF-06
Nombre de requisito	No permitir asignar a una clase más horas de las definidas
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqF-07
Nombre de requisito	No permitir asignar al docente en horas que tiene clases
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqF-08
Nombre de requisito	Permitir solo bloques de 2 o 3 horas
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input type="checkbox"/> Alta/Eencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqF-09
Nombre de requisito	No permitir la misma asignatura más de una vez en un día



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input type="checkbox"/> Alta/Eencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqF-10
Nombre de requisito	Los coordinares de carrera solo pueden ver los horarios de su carrera
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input type="checkbox"/> Alta/Eencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqF-11
Nombre de requisito	Los directores de departamento deberán poder visualizar todos los horarios de las carreras que pertenecen a su departamento
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

3.2 Requerimientos no Funcionales

Número de requisito	ReqNF-01
Nombre de requisito	Disponible mediante Internet
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	-
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqNF-02
Nombre de requisito	Soporte para Mozilla y Chrome



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	-
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqNF-03
Nombre de requisito	Interfaces atractivas e intuitivas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	-
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqNF-04
Nombre de requisito	Elementos y letras de tamaño moderado para accesibilidad
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	-
Prioridad del requisito	<input type="checkbox"/> Alta/Eencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqNF-05
Nombre de requisito	Responsivo
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	-
Prioridad del requisito	<input type="checkbox"/> Alta/Eencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	ReqNF-06
Nombre de requisito	Asignación mediante arrastrar y soltar
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional



Número de requisito	ReqNF-07
Nombre de requisito	Evaluar datos antes de guardar
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Docente Evaluador
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

3.1 Requisitos comunes de los interfaces

El sistema web de asignación de horarios de docentes para la UNAN FAREM Chontales constará con las características que nos ofrecen herramientas de diseño web en este caso, Bootstrap, una interfaz amigable, sencilla, con colores que faciliten la visualización de cada uno de sus elementos, Las aplicaciones móviles contarán con un diseño similar sin dejar de lado las diferencias de semántica entre plataformas.

3.1.1 Interfaces de usuario

La interfaz con la que constará esta aplicación web será minimalista pero elegante, para el ingreso se cuenta con un Login de usuario, en caso de no poseer uno se cuenta con solicitud de credenciales para obtener uno, además, colores claros, elementos relativamente grandes, aunque las propiedades de tamaño finalmente dependen finalmente de la configuración del navegador.

Las aplicaciones móviles contarán con un diseño minimalista, agradable a la vista y los elementos poseerán un tamaño básico, siguiendo la semántica de diseño de las aplicaciones móviles según su plataforma, pero ambos diseños compartirán los estilos.

3.1.2 Interfaces de hardware

Plataforma web: No aplica.

Plataforma Android: Cualquier celular con Android 4.0 o superior.

Plataforma WindowsPhone: Cualquier celular con Windows 8 o superior.



3.1.3 Interfaces de software

Dado que la aplicación utiliza los navegadores de internet y no utiliza componentes externos como flash, Silverlight y sus semejantes, los requisitos físicos son los requeridos por los navegadores a utilizar, siendo a los que está dirigida la aplicación Mozilla Firefox y Google Chrome en sus versiones más actualizadas.

Las plataformas móviles requerirán Android 4.0 o superior para la plataforma Android y Windows 8 o superior para la Plataforma WindowsPhone.

No existe integración de la aplicación con otras herramientas de software, es completamente independiente.

3.1.4 Interfaces de comunicación

Se utiliza el protocolo HTTP para el ingreso a la página. Los servicios utilizan el protocolo SOAP (Simple Object Access Protocole) para la conexión entre la base de datos y las aplicaciones.

3.2 Requisitos funcionales

3.2.1 Requisito funcional 1

Nombre: Presentar automáticamente las asignaturas y sus respectivos docentes.

Descripción: Dado que la información con la cual el sistema obtendrá los datos de todas las asignaturas y sus respectivos docentes, al momento de crear un horario de un determinado grupo en una carrera se deberá mostrar automáticamente las asignaturas de dicho grupo y su(s) docente es cuestión.

3.2.2 Requisito funcional 2

Nombre: Después de generar horarios solicitar revisión y permitir guardar e imprimir.



Descripción: Después de finalizar un horario, el usuario coordinador lo podrá guardar (Aunque guardar no necesariamente implica que el horario esté finalizado) para posteriormente modificarlos o en caso de que no se necesiten más modificaciones, solicitar una revisión por parte del usuario Director de departamento, si la aprobación es positiva, el horario podrá ser impreso o guardado en los diferentes formatos disponibles.

3.2.3 Requisito funcional 3

Nombre: Login de usuarios.

Descripción: Para utilizar esta herramienta de software, se deberá primeramente iniciar sesión.

3.2.4 Requisito funcional 4

Nombre: Obtener datos de formato generado por el sistema central de la UNAN MANAGUA.

Descripción: Puesto que los datos necesarios para generar horarios se encuentran en la base central de UNAN MANAGUA y su sistema permite generar un reporte digital, este último será manipulado para llenar la base de datos propia del SAHO.

3.2.5 Requisito funcional 5

Nombre: Solo puede ser usado por los coordinadores de carrera y directores de departamento.

Descripción: Además del usuario administrador que se encargará de la gestión de los datos, el único personal de la UNAN MANAGUA - FAREM Ch a quien le será permitido utilizar la aplicación es a los coordinadores de carrera y directores de departamento, los cuales tendrán que hacer una solicitud de credenciales si no las poseen.



3.2.6 Requisito funcional 6

Nombre: No permitir asignar a una clase más horas de las definidas.

Descripción: En concreto cada una de las asignaturas tiene horas totales definidas, por lo cual no se debe permitir ubicar en el horario exceso de horas en cada asignatura.

3.2.7 Requisito funcional 7

Nombre: No permitir asignar al docente en horas que tiene clases.

Descripción: Al momento de asignar clases, se debe verificar que el docente esté disponible en ese rango de horas.

3.2.8 Requisito funcional 8

Nombre: Permitir solo bloques de 2 o 3 horas.

Descripción: Las clases siempre poseen 5 o 6 horas por semana, dividida en 2 encuentros, por lo cual, por propósitos de ellos, las clases deben ser asignadas en bloques de 2 o 3 horas no más, y no menos.

3.2.9 Requisito funcional 9

Nombre: No permitir la misma asignatura en más de una vez en un día.

Descripción: Las asignaturas solo pueden ser en bloques de 2 o 3 horas una sola vez al día.

3.2.10 Requisito funcional 10

Nombre: Los coordinadores de carrera solo pueden ver los horarios de su carrera.



Descripción: Los usuarios coordinadores podrán generar horarios, visualizar y modificar solamente para su carrera.

3.2.11 Requisito funcional 11

Nombre: Los directores de departamento deberán poder visualizar todos los horarios de las carreras que pertenecen a su departamento.

Descripción: Luego de que los horarios son generados por los coordinadores de carrera y estos solicitan una revisión, los directores de departamento podrán ver y aprobar o no los horarios, solamente de las carreras que pertenecen a su departamento.

3.3 Requisitos no funcionales

3.3.1 Requisitos de rendimiento

El sistema deberá ser capaz de administrar de forma simultánea que cada uno de los coordinadores de carrera trabajen en conjunto solicitando sus servicios y ejecutando procesos, los procesos deberán ser procesados de forma eficiente, en donde se pretende que el 70 % de estos sean realizados en 3 segundos o menos, pero teniendo en cuenta el factor de la velocidad de la red y de los terminales en que se esté ejecutando.

3.3.2 Seguridad

Puesto que se utilizarán servicios, las páginas de UI trabajará de forma remota del servidor de los servicios, solamente realizando solicitudes, esto permite integridad de la BD y en conjunto con la pantalla de Login permite una gestión sencilla de la seguridad.



3.3.3 Fiabilidad

Se pretende que los procesos se realicen de forma totalmente fiable, procurando que los únicos errores en el sistema sean los que no se pueden controlar, siendo esto los errores humanos por el mal uso de las herramientas.

3.3.4 Disponibilidad

El sistema pretende una disponibilidad de 100% siempre y cuando se cuente con una conexión a internet.

3.3.5 Mantenibilidad

Es necesario que en cada uno de los semestres se vuelva a llenar la base de datos con los cambios de docentes y asignaturas.

3.3.6 Portabilidad

Disponible en los diferentes sistemas operativos de escritorio más usados en el mercado, su dependencia es meramente con el navegador web.



13.2 Manual técnico



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Tabla de contenido

1. Introducción	1
2. Objetivo:	1
3. Requerimientos técnicos	1
3.1 Requerimientos mínimos de hardware.....	1
3.2 Requerimientos mínimos de software.....	1
4. Herramientas utilizadas para el desarrollo	2
4.1 ASP.....	2
4.2 SQL Server.....	2



1. Introducción

La finalidad de todo manual técnico es la de proporcionar al lector la lógica con la **que** se ha desarrollado una aplicación, la cual se sabe que es propia de cada programador; por lo que se considera necesario ser documentada.

Aclarando que este manual no pretende ser un curso de aprendizaje de cada una de las herramientas empleadas para el desarrollo del sitio, sino documentar su aplicación en el desarrollo del sitio. Para un mayor detalle acerca de cada una de las herramientas utilizadas, y su forma de operación y aplicación, se recomienda consultar los manuales respectivos de cada una de ellos.

2. Objetivo:

Proporcionar una guía para el lector, del desarrollo de la interfaz y programación del Sistema de Asignación de Horario (SAHO) en ciudad de Juigalpa Chontales.

3. Requerimientos técnicos

3.1 Requerimientos mínimos de hardware

- Procesador : 2.3 GHz (Dos Núcleos)
- Memoria RAM : 4 Gigabytes (GB)
- Disco Duro : 500 GB

3.2 Requerimientos mínimos de software

- Conexión a internet desde cualquier navegador
- Cualquier sistema operativo Windows desde XP



4. Herramientas utilizadas para el desarrollo

4.1 ASP

ASP.NET es un modelo de desarrollo Web unificado que incluye los servicios necesarios para crear aplicaciones Web empresariales con el código mínimo. ASP.NET forma parte de .NET Framework y al codificar las aplicaciones ASP.NET tiene acceso a las clases en .NET Framework. El código de las aplicaciones puede escribirse en cualquier lenguaje compatible con el Common Language Runtime (CLR), entre ellos Microsoft Visual Basic, C#, JScript .NET y J#. Estos lenguajes permiten desarrollar aplicaciones ASP.NET que se benefician del Common Language Runtime, seguridad de tipos, herencia, etc.

4.2 SQL Server

SQL Server es un sistema de gestión de bases de datos relacionales (RDBMS) de Microsoft que está diseñado para el entorno empresarial. SQL Server se ejecuta en T-SQL (Transact - SQL), un conjunto de extensiones de programación de Sybase y Microsoft que añaden varias características a SQL estándar, incluyendo control de transacciones, excepción y manejo de errores, procesamiento fila, así como variables declaradas.

- Soporte de transacciones.
- Soporta procedimientos almacenados.
- Incluye también un entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y los terminales o clientes de la red sólo acceden a la información.
- Además, permite administrar información de otros servidores de datos.



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



5. Estructura de los proyectos

EL servidor de la aplicación contiene dos proyectos, que representan las capas de los servicios y la capa de datos o conexión a la información.

La capa de datos: se encarga de obtener una conexión directa a nuestra(s) bases de datos, en el presente proyecto, esta capa la conforma dos tipos de objetos, clases “.cs” de lenguaje de programación C# que representan propiedades de objetos presentes en la base de datos, siendo estos, específicamente tablas de la base de datos o un conjunto de tablas de la base de datos con información vincula, estas clases permiten crear objetos con este formato para presentar la información.

Ejemplo de una propiedad en el proyecto:

```
Explorador de control de código fuente
capaDatos
  - capaDatos.Clases.objetoHorarioDocente
    namespace capaDatos.Clases
    {
        5referencias
        public class objetoHorarioDocente
        {
            string nombre, asignatura;
            int grupo, dia;
            string horainicio, horafinal;

            0referencias
            public string Nombre
            {
                get { return nombre; }
                set { nombre = value; }
            }

            1referencia
            public string Asignatura
            {
                get { return asignatura; }
                set { asignatura = value; }
            }

            1referencia
            public int Grupo
            {
                get { return grupo; }
                set { grupo = value; }
            }
        }
    }
```

Las propiedades presentan variables privadas que son controladas mediante las propiedades GET y SET de C#, estas en conjunto son las que permiten dar un formato a los tipos de datos obtenidos desde la base de datos del software.

Por otra parte, la conexión a la base de datos es realizada mediante la tecnología LINQ que permite además de una conexión con la base de datos, permite conocer los objetos presentes en ella e instanciarlos para su posterior uso, estas son clases “.dbml”.

LINQ SQL conexión, tablas y procedimientos del proyecto:



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



La imagen muestra una interfaz de desarrollo de software (IDE) con dos paneles principales. El panel izquierdo contiene un diagrama de base de datos que muestra varias tablas (como 'tbl_usuario', 'tbl_grupo', 'tbl_docente') y sus relaciones mediante líneas con flechas. El panel derecho, titulado 'Explorador de control de código fuente', muestra una lista de métodos LINQ con sus respectivas firmas de parámetros, como:

- acceptarGrupo (System.Boolean notif, System.Int32 cod_grupo, System.Int32 cod_estudiante)
- acceptarSolicitudMatricula (System.Int32 codSolicitud)
- acceptarUsuario (System.Int32 id, System.Int32 notif)
- asignarHorario (System.Boolean dia, System.TimeSpan horainicio, System.TimeSpan horafinal, System.Int32 cod_h)
- borarDetalleHorario (System.Int32 cod_det_horario)
- borrarActividad (System.Int32 cod)
- cambiarDatosUsuario (System.Int32 id, System.String alias, System.String passwrd, System.String corr)
- cambiarUsuario (System.Int32 id, System.String alias, System.String contra)
- cargarActividadesGrupo (System.Int32 codgrupo)
- cargarSolicitudesEstudiante (System.Int32 codDocente)
- codDocente (System.Int32 codUsuario)
- comprobar_asignacion (System.Int32 cod_docente, System.String cod_asignatura, System.TimeSpan horainicio, Sy
- contarNotificacionesSolicitudMatricula (System.Int32 cod_docente)
- crearActividad (System.String titulo, System.String descripcion, System.Data.Linq.Binary archivo, System.Int32 cod)
- crearAsignatura (System.String cod, System.String nombre, System.Int32 horas)
- crearCarrera (System.String cod_carrera, System.String nombre, System.Int32 cod_departamento)
- crearClase (System.Int32 codDocente, System.String codAsignatura)
- crearDepartamento (System.String nombre)
- crearDetalleMaestro (System.Int16 dia, System.TimeSpan horainicio, System.TimeSpan horafinal, System.Int32 cod)
- crearDocente (System.String nombre, System.String apellido)
- crearEstudiante (System.String alias, System.String contras, System.String nombre, System.String correo)
- crearGrupo (System.String cod_carrea)
- crearHorario (System.Int32 ano, System.Boolean semestre, System.Boolean bimestre, System.Int32 coddepartamen
- crearUsuario (System.String usr, System.String pass, System.Int32 tipo, System.String codcar, System.Int32 coddept
- datosEnvioCorreo (System.String correo)
- datosUsuario (System.Int32 cod)
- datosUsuarioDocente (System.Int32 user)
- editarDocente (System.Int32 id, System.String name, System.String lastname)
- horarioDocente (System.Int32 codDocente)

Se observa que la conexión nos permite una vista detallada de las tablas y sus conexiones, y también, un listado de los procedimientos almacenados disponibles a ejecutar con pocas líneas de código, gracias a la misma tecnología LINQ.

La capa de servicios o simplemente los servicios se encargan de enviar información específica a un cliente que la solicita, esto mediante paquetes SOAP a los cuales se les da forma gracias al uso de las propiedades en la capa de datos, los servicios contienen “WebMethods” los cuales son métodos disponibles mediante la web y que devuelven información de una solicitud específica con un formato predefinido.

Ejemplo de uso de Servicios y WebMethods:



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



```
[WebMethod]
0 Referencias
public List<listar_Carreras> listarCarreras()
{
    try
    {
        bd = new dataConexionDataContext();

        List<listar_Carreras> lista = new List<listar_Carreras>();

        var resultado = bd.listarCarreras();

        foreach (var row in resultado)
        {
            listar_Carreras i = new listar_Carreras()
            {
                cod = row.cod_carrera,
                nombre = row.nombre,
                departamento = row.cod_depart
            };

            lista.Add(i);
        }

        return lista;
    }
    catch
    {
        return null;
    }
}
```

Los [WebMethods] deben cumplir un par de criterios para ser accesibles y visibles:

1. Antes de iniciar el método en cuestión en el servicio se antepone [WebMethod]
2. Todo método web debe ser público “public” para poder recibir peticiones, de lo contrario solo estará disponible para otros métodos que compartan la misma clase.

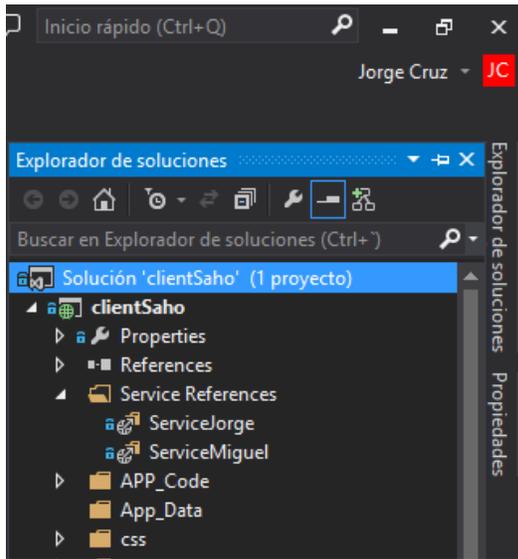
El proceso de desarrollo del proyecto servidor se da justo después de finalizar la creación de la base de datos y todos sus objetos, seguido de esto se crea una conexión a esta bd mediante la tecnología LINQ “.dbml”, se obtienen los objetos para instanciarlos y después se procede a crear las clases “.cs” de propiedades para dar formato a la información, finalizada esto se procede a conectar los servicios con los objetos de la base de datos y las clases, se diseñan los métodos web para su posterior uso con el cliente.

Los procesos de programación del lado del cliente se encargan de capturar la información o respuesta dada determinada solicitud al servidor, seguidamente estos están disponibles para presentarlos o procesarlos de la forma que sea necesaria.

Primeramente, para poder utilizar recursos del servicio web se debe de crear una referencia al mismo, al crear la referencia podemos instanciar y utilizar el servicio y sus métodos, la a él o los servicios se presenta dentro del apartado “Service References” de la aplicación.



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Una vez contamos con una conexión, solo resta instanciarlos para utilizarlos

```
ServiceMiguel.serviceMiguelSoapClient serv= new  
ServiceMiguel.serviceMiguelSoapClient();
```

Finalmente, solo resta capturar información según convenga, ejemplo:

```
login.aspx.cs login.aspx  
clientSaho clientSaho.login  
}  
Referencias  
protected void btentrar_Click(object sender, EventArgs e)  
{  
    try  
    {  
        string usuario = Request.Form["usuario"];  
        string pass = Request.Form["contra"];  
  
        var resultado = serv.iniciarSesion(usuario, pass);  
  
        foreach (var row in resultado)  
        {  
            Session["cod_usuario"] = row.cod_usuario;  
            Session["nombres"] = row.nombre;  
            Session["apellidos"] = row.apellidos;  
            Session["alias"] = row.alias;  
            Session["contra"] = row.contra;  
            Session["tipo"] = row.tipo;  
            Session["cod_carrera"] = row.cod_carrera;  
            Session["cod_docente"] = row.cod_docente;  
            Response.Redirect("/index");  
        }  
    }  
    catch  
    {  
        return;  
    }  
}
```

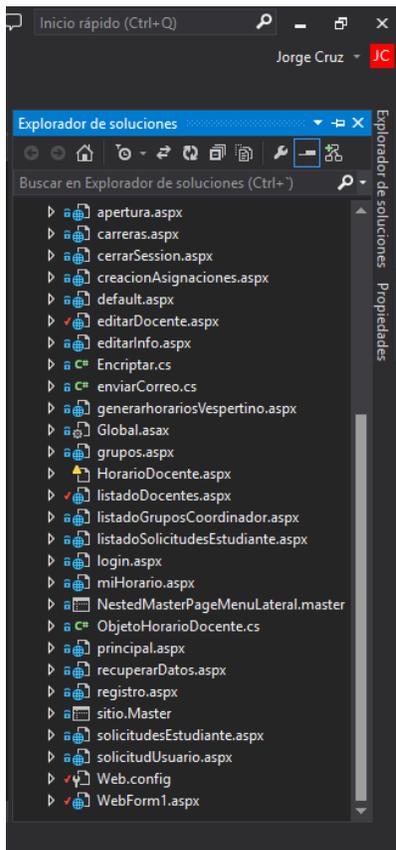
Nota: Las variables de sesión o cookies permiten almacenar información que esté disponible en el navegador para facilitar su uso en distintas partes de la aplicación.



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



La estructura del proyecto cliente se divide en dos elementos fundamentales, clases “.cs” permiten controlar mediante programación los datos y elementos de la página web, y páginas web ASP .NET “.aspx” las cuales permiten el uso de etiquetas HTML e incluyen etiquetas o controles de ASP .NET que poseen mayor flexibilidad de programación y las mismas ventajas de los controles puros de HTML.



6. Diseño de interfaces

El Diseño de páginas e interfaces de usuario, se ha hecho utilizando las siguientes herramientas

- Software Utilizado: VisualStudio.Net 2013 y SQL Server 2012
- Lenguajes de diseño: HTML5, CSS3, AXML



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



- Lenguaje de programación: C# ASP.NET, MONO C# for Android

El cliente o capa presentación está construido mediante clases de programación de C# y páginas web de ASP .NET en las cuales se detalla la estructura física de la página.

El proyecto combina el uso de etiquetas sencillas de HTML y controles más personalizables como los son los de ASP .NET.

La diferencia aunque obvia de los dos tipos de controles es que los de HTML puro se declaran con etiquetas sencillas `<control></control>` mientras que los de ASP .NET se definen `<asp:control></asp:control>`

Ejemplo de controles de ASP .NET

```
<div class="col-md-3">
  <asp:Button ID="btnBuscar" runat="server" OnClick="btnBuscar_Click" Text="Buscar" CssClass="btn btn-default" />
</div>
```

Todos los controles de ASP .NET deben de contener las propiedades `ID=<ID unico del elemento>` y `runat=<nombre del servidor>`.

Ejemplo de controles de HTML

```
<div class="md-form">
  <i class="fa fa-envelope prefix"></i>
  <input type="text" name="usuario" id="usuario" class="form-control" runat="server" autofocus required>
  <label for="usuario">Nombre de usuario</label>
</div>
```

Todos los controles de HTML deben de contener las propiedades `id=<ID unico del elemento>` y `runat=<nombre del servidor>` para poder ser controladas mediante programación.

Finalmente los aspectos graficos de las interfaces se detalla con el uso del framawork MDBootstrap en conjunto con sus elementos Javascript y deben ser siempre incluidos dentro de los recursos del proyecto.



13.3 Manual de usuario



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Índice

1. Introducción	1
2. Objetivos	1
3. Dirigido.....	1
4. Lo que debe conocer	1
5. Hardware.....	1
6. Software.....	2
7. Glosario	2
8. Pantalla principal (Inicio de sesión).....	3
9. Regístrate.....	4
10. Rol como docente	5
11. Rol como de coordinador de carrera	6
12. Rol como Director de departamento	7
13. Usuario (Coordinador) Pantalla principal.....	7
14. Usuario (Coordinador) Horario	7
14.1 Modificar datos de usuario.....	9
14.1.1 Modificar datos de usuario “continuación”	10
15. Usuario (Coordinador) Crear	11
15.1 Usuario (Coordinador) Crear “continuación”	12
15.2. Usuario (Coordinador) Estudiantes	13
16. Usuario (Director) Pantalla principal.....	14
17. Usuario (Director) Modificar datos de usuarios	15



Manual de usuario

1. Introducción

El presente manual de usuario, describirá los objetivos e información clara y concisa de cómo utilizar el Sistema de Asignación de Horario (SAHO), siendo de mucha importancia este manual ante y/o durante la visualización de las paginas ya que lo que guiara paso a paso en el manejo de cada una de sus funciones para hacer posible este manual se utilizó el estándar IEEE 1063 [27]

2. Objetivos

- ❖ Guiar paso a paso al usuario a manejar el Sistema de Asignación de Horario (SAHO).
- ❖ Brindar una descripción detallada e ilustrada a las distintas opciones de la aplicación.

3. Dirigido

Este manual está orientado a los usuarios finales los cuales son: Docente, Director y Coordinador cada uno con su respectivas funciones.

4. Lo que debe conocer

Conocimientos básicos en:

- ❖ Internet
- ❖ Navegación Web
- ❖ Familiarizado con el proceso de crear horario.
- ❖ Especificaciones técnicas

5. Hardware

- ❖ Procesador : 2.3 GHz (Dos Núcleos)
- ❖ Memoria RAM : 4 Gigabytes (GB)
- ❖ Disco Duro : 500 GB



6. Software

Cualquier sistema operativo

Navegadores (Google chrome, Mozilla Firefox u otros)

7. Glosario

Palabra	Significado
Credenciales	Una credencial es una orden o un documento que atestigua o autoriza la cualificación, competencia o autoridad otorgada a un individuo por un tercero con autoridad.
Footer	En español pie de página, se pueden mostrar los derechos de autor, información propia de la página.
Hardware	Se refiere a todas las partes físicas de un sistema informático.
SAHO	Sistema de Asignación de Horario
Software	Es el equipo lógico o soporte lógico de un sistema informático



8. Pantalla principal (Inicio de sesión)

A continuación se muestra la pantalla principal donde el usuario podrá iniciar sesión, una vez que ingrese sus credenciales correctamente y haiga presionado el botón Entrar.

Si a un caso no se ha registrado, abajo del botón Entrar de click en Regístrate y posteriormente se le presentarán una serie de pasos. O si olvido su contraseña la podrá recuperar con un mensaje que le llegará al correo con el cual se registró.

En la parte del footer (Pie de página) se muestran los derechos de autor con los nombres de los creadores de la página web.

Universidad Nacional Autónoma de Nicaragua, Managua
FAREM-Chontales

Sistema de asignación de horarios
SAHO 1.0

Inicio de Sesión:

Nombre de usuario

Contraseña

Entrar

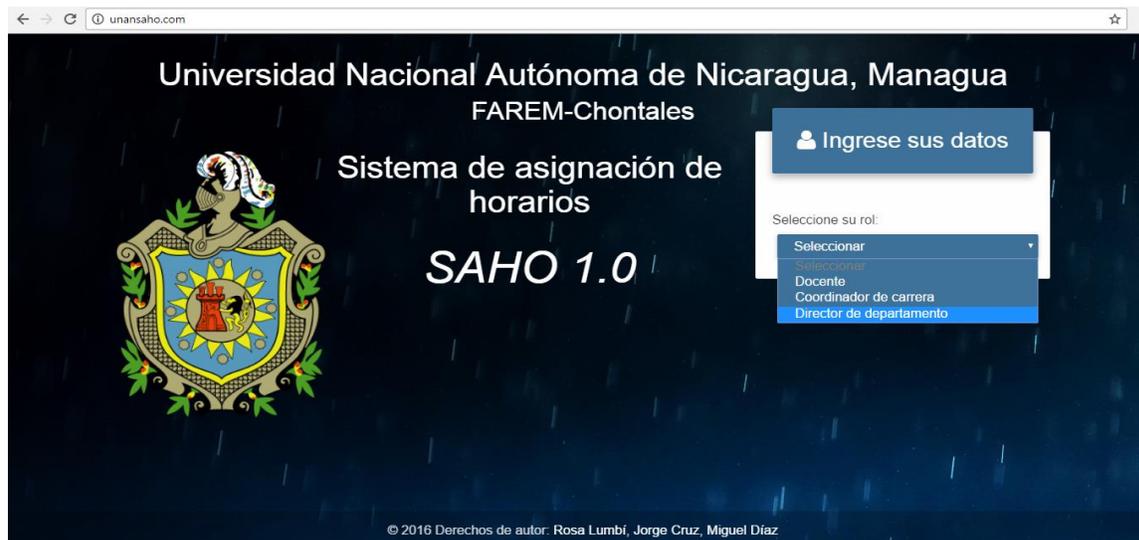
¿Eres coordinador o docente? [Regístrate](#)

¿Se te olvidó tu [Contraseña?](#)



9. Regístrate

En el cuadro donde dice ingrese sus datos podrá dar click en seleccionar y se desplegará una lista de los diferentes usuarios tales como: Docente, Coordinador de carrera, Director de departamento.





10. Rol como docente

Una vez seleccionado el rol como Docente, de click en seleccionar y automáticamente se desplegará una lista de los docentes.

Universidad Nacional Autónoma de Nicaragua, Managua
FAREM-Chontales

Sistema de asignación de horarios
SAHO 1.0

Ingrese sus datos

Seleccione su rol:
Docente

Seleccionar el docente
Seleccionar el docente
Miguel Angel Diaz Fernandez
Rosa Marina Lumbi Suárez
Jorge Inocente Cruz Serrano
Robison Aquiles Obando Aragon
Saira Maria Urbina Cienfuegos
Jonathan Moreno
Miriam Patricia Tellez
Alfredo Antonio Rodriguez Tellez

© 2016 Derechos de autor: Rosa Lumbí, Jorge Cruz, Miguel Díaz



11. Rol como de coordinador de carrera

Cuando se selecciona el rol como coordinador de carrera, de click en seleccionar carrera en donde se desplegará una lista de las diferentes carreras que ya están registradas y posteriormente de click para seleccionar el docente al que pertenece.

Universidad Nacional Autónoma de Nicaragua, Managua
FAREM-Chontales

Sistema de asignación de horarios
SAHO 1.0

Ingrese sus datos

Selección su rol:

- Coordinador de carrera
- Selección la carrera
- Selección el docente

Selección el docente:

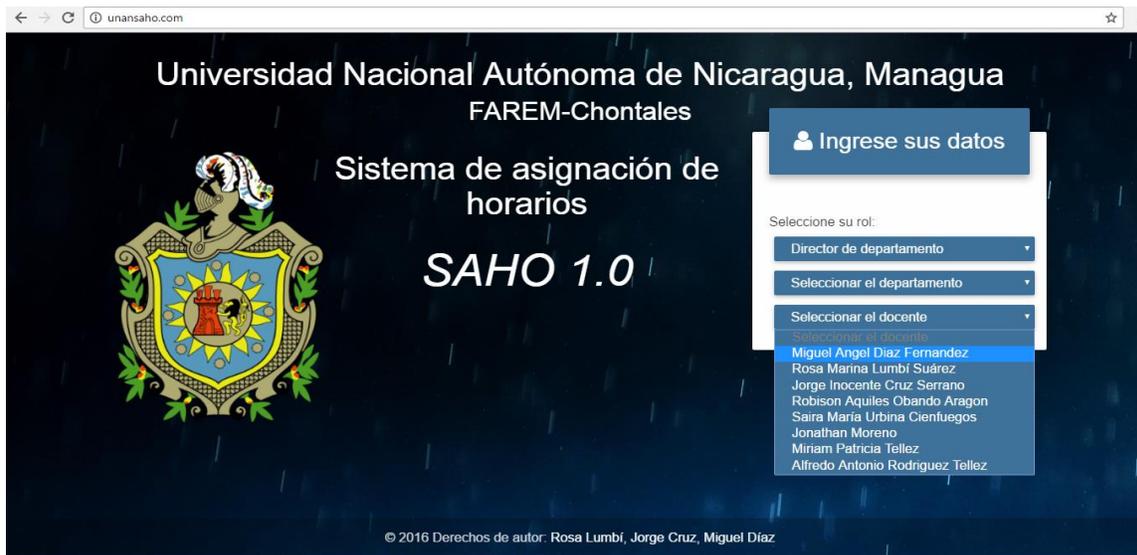
- Miguel Angel Diaz Fernandez
- Rosa Marina Lumbí Suárez
- Jorge Inocente Cruz Serrano
- Robison Aquiles Obando Aragon
- Saira María Urbina Cienfuegos**
- Jonathan Moreno
- Miriam Patricia Tellez
- Alfredo Antonio Rodriguez Tellez

© 2016 Derechos de autor: Rosa Lumbí, Jorge Cruz, Miguel Díaz



12. Rol como Director de departamento

Habiendo seleccionado el rol como director de departamento, de click en seleccionar el departamento donde le aparecerá una lista de los diferentes departamentos donde una vez haiga seleccionado, seleccionar el docente al que pertenece.



13. Usuario (Coordinador) Pantalla principal

Presione click en el nombre del usuario para que se le presente las opciones de modificar perfil y cerrar sesión.

También se presenta un icono donde están las solicitudes por parte de los usuarios (alumnos).

A lado izquierdo se encuentra el menú, de click en uno de ellos y se le desplegará cada una de sus opciones.

14. Usuario (Coordinador) Horario

De click en horarios y podrá ir a las opciones de crear, historial, visualizar, ver mi horario.



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



The screenshot shows a web browser window with the URL `unansaho.com`. The page title is "Sistema de asignación de horarios". The interface includes a dark blue sidebar on the left with the following menu items: "Horarios" (with a sub-menu arrow), "Crear", "Historial", "Visualizar", "Ver Mi Horario", "Estudiantes" (with a sub-menu arrow), "Listados" (with a sub-menu arrow), and "Acerca de..." (with an information icon). The main content area is currently empty. In the top right corner, there is a user profile icon with a notification badge showing "10" and the text "Hola andiferm". A dropdown menu is open, showing "Modificar Perfil" and "Cerrar Sesión". The footer contains the text "© 2016 Derechos de autor: Rosa Lumbí, Jorge Cruz, Miguel Díaz" next to a small logo.



14.1 Modificar datos de usuario

Habiendo dado click en modificar perfil se le presentarán los datos, en algunos casos no se podrán modificar ya que estará deshabilitada esa opción por ejemplo “Su rol de usuario”.

The screenshot shows a web browser window with the URL 'unansaho.com'. The page title is 'Sistema de asignación de horarios'. The user is logged in as 'Hola! andiferm'. The main content area is titled 'Modificar datos de usuario' and contains the following fields:

- Su rol de usuario: (disabled)
- Nombre de usuario:
- Contraseña:
- Correo:
- Carrera: (disabled)



14.1.1 Modificar datos de usuario “continuación”

Una vez que haiga modificado los datos presione el botón Guardar.

The screenshot shows a web browser window with the URL 'unansaho.com'. The page title is 'Sistema de asignación de horarios'. The user is logged in as 'andifermd@gmail.com'. The page has a dark blue sidebar with navigation options: 'Horarios', 'Estudiantes', 'Listados', and 'Acerca de...'. The main content area contains a form for user data. The form fields are: 'Email' (andifermd@gmail.com), 'Carrera' (Ingeniería en Sistemas de Información), 'Datos de docente' (Nombre: Miguel Angel, Apellidos: Diaz Fernandez), and a 'Guardar' button. The footer of the page contains the copyright notice: '© 2016 Derechos de autor: Rosa Lumbí, Jorge Cruz, Miguel Díaz'.



15. Usuario (Coordinador) Crear

De click en crear para visualizar y se presentaran la lista de grupos dependiendo la carrera en el cual es coordinador.

The screenshot shows a web browser window with the URL 'unansaho.com'. The page title is 'Sistema de asignación de horarios'. A navigation menu on the left includes 'Horarios', 'Crear', 'Historial', 'Visualizar', 'Ver Mi Horario', 'Estudiantes', 'Listados', and 'Acerca de...'. The main content area displays a table titled 'Lista de grupos de su carrera'.

Carrera	Grupo	Crear
Ing. En sistemas de la información	1	

© 2016 Derechos de autor: Rosa Lumbí, Jorge Cruz, Miguel Díaz



15.1 Usuario (Coordinador) Crear “continuación”

De click en el icono de crear y se le mostrará un formato que consiste en arrastrar y soltar las asignaturas con sus debidas horas, si desea puede descartar los cambios o de click en el icono rojo con una rayo si desea imprimir o enviar el horario.

Asignaturas	Horas		Lunes	Martes	Miércoles	Jueves
Facultativa II	4	1:00-1:45				
		1:50-2:35				
Auditoria Informatica II	5	2:40-3:25				
		3:35-4:15				
Practicas de Profesionalización	6	4:20-5:00				
		5:15-6:00				
Seminario de Graduación	6					

DESCARTAR



15.2. Usuario (Coordinador) Estudiantes

De click en Estudiantes y se desplegará un submenú en donde podrá crear actividad, le dará un nombre, agregará detalles, también puede subir un archivo una vez seleccionado. Y automáticamente lo creará presionando el botón crear.

La imagen muestra una interfaz web en un navegador con la URL `localhost:26345/crearactividad`. El título de la página es "Sistema de asignación de horarios". En la parte superior derecha hay un ícono de notificación con el número "10" y el nombre de usuario "Hola andiferm".

El menú lateral izquierdo contiene los siguientes ítems: "Horarios", "Estudiantes" (seleccionado), "Crear Actividades", "Matricular", "Carreras", "Listados" y "Acerca de...".

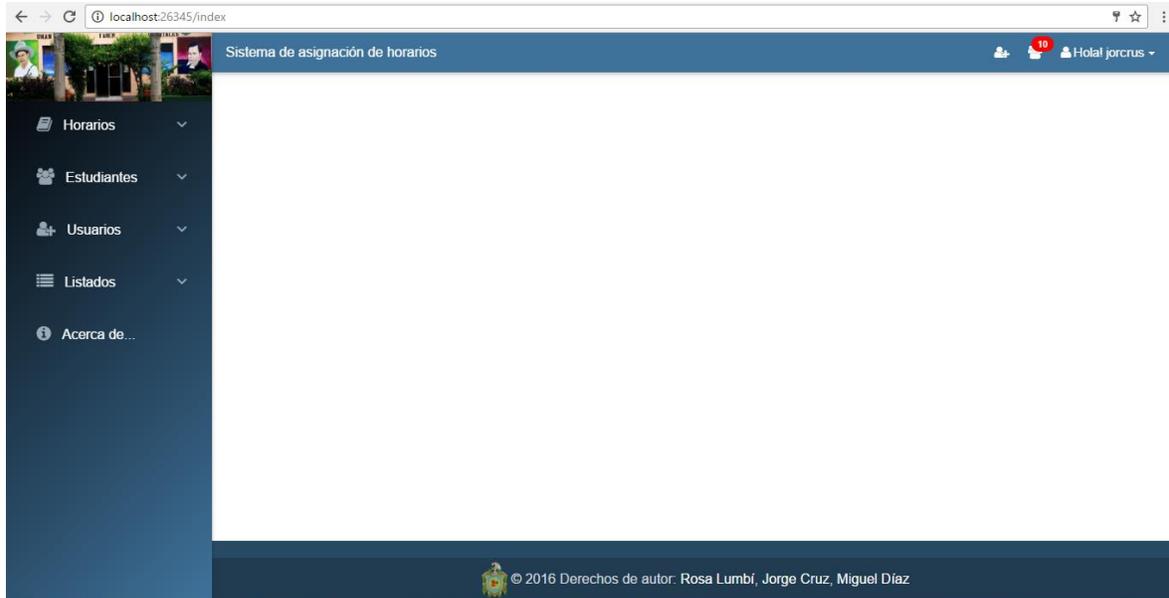
El contenido principal de la página está dentro de un recuadro con el título "Crear Actividad". Incluye los siguientes campos:

- Un botón azul "Crear Actividad" al principio.
- Una etiqueta "Grupo:".
- Un campo de texto "Nombre de la actividad".
- Una etiqueta "Detalles:".
- Un campo de texto "Detalles:".
- Un botón "Seleccionar archivo" y el texto "Ningún archivo seleccionado".
- Un botón verde "Crear" al final.



16. Usuario (Director) Pantalla principal

A continuación se presenta un menú además un icono de aprobación de solicitud de los docentes, coordinador y alumnos.





17. Usuario (Director) Modificar datos de usuarios

De click en el nombre del usuario y tendrá acceso a modificar perfil o cerrar sesión, si así lo desea.

Habiendo dado click en modificar perfil se le presentarán los datos, en algunos casos no se podrán modificar ya que estará deshabilitada esa opción por ejemplo “Su rol de usuario”.

Sistema de asignación de horarios

Modificar datos de usuario

Su rol de usuario:
[Desactivado]

Nombre de usuario:
jorcus

Contraseña:
livelong

Correo:
jorcus2@gmail.com

Datos de docente

Nombre:
Jorge Inocente

Apellidos:
Cruz Serrano

Guardar

© 2016 Derechos de autor: Rosa Lumbí, Jorge Cruz, Miguel Díaz



13.4 Casos de prueba (Caja negra)

Tabla de contenido

Registro de cambios	1
Control de distribución	1
1. Introducción	2
2. Objeto	2
3. Alcance	2
4. Trazabilidad de casos de pruebas – requisitos	3
5. Definición de los casos de pruebas	4



Hoja de control

Organismo	Equipo SAHO		
Proyecto	Sistema De Asignación de Horarios SAHO		
Entregable	Planes de Pruebas Funcionales		
Autores	Miguel Díaz, Jorge Cruz, Rosa Lumbí		
Versión/Edición	01	Fecha Versión	Enero, 2017
Aprobado por		Fecha Aprobación	

Registro de cambios

Versión doc	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
01	Versión inicial	Miguel Díaz Fernández	Enero, 2017
01	Versión inicial	Jorge Cruz Serrano	Enero, 2017
01	Versión inicial	Rosa Marina Lumbí	Enero, 2017

Control de distribución

Nombre y Apellidos
Miguel Ángel Díaz
Jorge Cruz Serrano
Rosa Marina Lumbí



1. Introducción

Aclarando que este manual no pretende ser un curso de aprendizaje de cada una de las herramientas empleadas para el desarrollo del sitio, sino documentar su aplicación en el desarrollo del sitio. Para un mayor detalle acerca de cada una de las herramientas utilizadas, y su forma de operación y aplicación, se recomienda consultar los manuales respectivos de cada una de ellos.

2. Objetivo

El objetivo de este documento es recoger los casos de pruebas que verifican que el sistema satisface los requisitos especificados. Deberá contener la definición de los casos de prueba, la matriz de trazabilidad entre casos de pruebas y requisitos, y la estrategia a seguir en la ejecución de las pruebas.

3. Alcance

El alcance del presente documento está dado por los requisitos del proyecto de software y se limita por las características que deben de cumplir según dichos requisitos, siendo estos los que brindan la pauta principal para las necesidades y funcionalidades que se deben cumplir con el desarrollo del software, las validaciones de entradas, salidas, procesos y rendimiento son los factores principales a ser evaluados.



4. Trazabilidad de casos de pruebas – requisitos

La siguiente tabla presenta las evaluaciones que realiza cada caso de prueba en correspondencia de cada uno de los requisitos presentados para el proyecto de software.

	RF-01	RF-02	RF-03	RF-04	RF-05	RF-06	RF-07	RF-08	RF-09	RF-10	RF-11
<CP-01>			X								
<CP-02>			X								
<CP-03>			X		X						
<CP-04>	X	X				X	X				
<CP-05>				X							
<CP-06>						X		X	X		
<CP-07>		X			X					X	X



5. Definición de los casos de pruebas

Las tablas a continuación describen cada uno de los casos de prueba requeridos para la evaluación de los requisitos previamente definidos para el proyecto, se describe su función, prerequisites necesarios para realizar las pruebas, los pasos seguidos, el resultado que se desea obtener y el resultado real obtenido para cada uno de ellos. Un caso de prueba puede evaluar uno o más requisitos funcionales.

Prueba de validación de inicio con datos no existentes	CP-01	
	¿Prueba de despliegue?	No
Descripción: Pruebas del comportamiento de la aplicación en caso de ingreso de datos que no existen en los registros de la base de datos del software.		
Prerrequisitos		
Pasos: <ol style="list-style-type: none">1. Ingreso del usuario: “andiferm”2. Ingreso de la contraseña: “password”3. Ejecutar inicio de sesión		
Resultado esperado: Mensaje de notificación de credenciales erróneas		
Resultado obtenido: Mensaje de notificación de credenciales erróneas		



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Prueba de validación de inicio con datos existentes	CP-02	
	¿Prueba de despliegue?	No
Descripción: Pruebas del comportamiento de la aplicación en caso de ingreso de datos que existen en los registros de la base de datos del software.		
Prerrequisitos <ul style="list-style-type: none">Existencia de las credenciales en la Base Datos.		
Pasos: <ol style="list-style-type: none">Ingreso del usuario: “andiferm”Ingreso de la contraseña: “andiferm”Ejecutar inicio de sesión		
Resultado esperado: Inicio de sesión exitoso		
Resultado obtenido: Inicio de sesión exitoso		



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Prueba de validación de inicio para diferentes tipos de usuario	CP-03	
	¿Prueba de despliegue?	No
Descripción: Pruebas de condicionamiento de la interfaz y su comportamiento para diferentes tipos usuarios		
Prerrequisitos <ul style="list-style-type: none">• Existencia de las credenciales en la Base Datos.		
Pasos: <ol style="list-style-type: none">1. Ingreso como docente con las credenciales “sairacien” “sairacien”2. Ingreso como coordinador de carrera con las credenciales “jorcrus” “jorcrus”3. Ingreso como director de departamento con las credenciales “mirpatema” “mirpatema”4. Ingreso como administrador con las credenciales “andiferm” “andiferm”		
Resultado esperado: Inicio de sesión exitoso Pantallas y menús diferentes para cada tipo de usuario		
Resultado obtenido: Inicio de sesión exitoso Pantallas y menús diferentes para cada tipo de usuario		



Prueba de validación de datos de horarios disponibles y manejo de datos en el horario	CP-04	
	¿Prueba de despliegue?	No
Descripción: Pruebas de manejo de la información y presentación al usuario		
Prerrequisitos <ul style="list-style-type: none">• Existencia de las credenciales de inicio en la base de datos• Existencia de datos para horario (Docente, clase, carga horaria) en la base de datos		
Pasos: <ol style="list-style-type: none">1. Ingreso como coordinador de carrera con las credenciales “jorcrus” “jorcrus”2. Comprobar los grupos asignados en el apartado de “Crear Horario”3. Seleccionar un grupo4. Verificar las clases, docentes y horas5. Asignar un docente en una hora6. En otra sesión, asignar al mismo docente en la misma hora en otro grupo7. Finalizar uno de los horarios8. Guardar9. Exportar y/o imprimirlo		
Resultado esperado: Inicio de sesión exitoso Visualización de grupos asignados Visualización de información de clases, docentes y horas Asignar exitosamente al docente en el primer grupo Fallo de asignación del docente en la misma hora, en otro grupo Guardado exitoso Exportación o Impresión exitosa		
Resultado obtenido: Inicio de sesión exitoso Visualización de grupos asignados Visualización de información de clases, docentes y horas Asignar exitosamente al docente en el primer grupo Fallo de asignación del docente en la misma hora, en otro grupo Guardado exitoso Pantallas y menús diferentes para cada tipo de usuario		



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Prueba de obtención de datos del Sistema Central de la UNAN	CP-05	
	¿Prueba de despliegue?	No
Descripción: Pruebas de manejo de la información y presentación al usuario		
Prerrequisitos <ul style="list-style-type: none">• Existencia de las credenciales de inicio en la base datos• Existencia de formato obtenidos del sistema de la universidad		
Pasos: <ol style="list-style-type: none">1. Iniciar como administrador con las credenciales “andiferm” “andiferm”2. Ingresar al módulo de obtención de datos3. Seleccionar el formato4. Guardar		
Resultado esperado: Inicio de sesión exitoso Visualización módulo de carga de formato Selección de formato Guardado exitoso		
Resultado obtenido: Inicio de sesión exitoso Inexistencia del módulo de carga del formato Guardado no posible		



Prueba de manejo de datos de horario	CP-06	
	¿Prueba de despliegue?	No
Descripción: Pruebas de manejo de la información y presentación al usuario		
Prerrequisitos <ul style="list-style-type: none">• Existencia de las credenciales de inicio en la base datos• Existencia de datos para horario (Docente, clase, carga horaria) en la base de datos		
Pasos: <ol style="list-style-type: none">1. Iniciar como coordinador de carrera con las credenciales “jorcus” “jorcus”2. Visualizar grupos asignados3. Seleccionar un grupo4. Asignar una clase de 2 horas5. Asignar una clase de 3 horas6. Asignar una clase de 4 horas7. Asignar una clase de 1 hora8. Asignar una clase dos veces en el mismo día9. Borrar asignaciones realizadas		
Resultado esperado: Inicio de sesión exitoso Visualizar grupos asignados Asignado de clase de 2 horas exitoso Asignado de clase de 3 horas exitoso Asignado de clase de 4 horas erróneo Asignado de clase de 1 hora erróneo Asignado de clase en el mismo día erróneo Borrado de asignaciones exitoso		
Resultado obtenido: Inicio de sesión exitoso Visualizar grupos asignados Asignado de clase de 2 horas exitoso Asignado de clase de 3 horas exitoso Asignado de clase de 4 horas exitoso Asignado de clase de 1 hora exitoso Asignado de clase en el mismo día exitoso Borrado de asignaciones exitoso		



Seminario de Graduación para optar al Título de Ingeniero en Sistemas de Información



Prueba de visualización de datos por usuario	CP-07	
	¿Prueba de despliegue?	No
Descripción: Pruebas de manejo de la información y presentación al usuario		
Prerrequisitos <ul style="list-style-type: none">• Existencia de las credenciales de inicio en la base de datos• Existencia de horarios creados en la base de datos.		
Pasos: <ol style="list-style-type: none">1. Iniciar como coordinador de carrera con las credenciales “jorcrus” “jorcrus”2. Ir al módulo de visualización de horarios3. Cerrar sesión4. Iniciar como director de departamento con las credenciales “mirpatema” “mirpatema”5. Ir al módulo de visualización de horarios		
Resultado esperado: Inicio de sesión exitoso como coordinador de carrera Visualización horarios de la carrera asignada al usuario Exportar o Imprimir Inicio de sesión exitoso como director de departamento Visualización de los horarios de todas las carreras que pertenecen al departamento asignado al usuario Exportar o Imprimir		
Resultado obtenido: Inicio de sesión exitoso como coordinador de carrera Visualización horarios de la carrera asignada al usuario Inicio de sesión exitoso como director de departamento Visualización de los horarios de todas las carreras que pertenecen al departamento asignado al usuario		