



**UNIVERSIDAD NACIONAL DE INGENIERÍA  
FACULTAD DE CIENCIAS Y SISTEMAS  
INGENIERÍA DE SISTEMAS**

**TRABAJO MONOGRÁFICO PARA OPTAR AL TÍTULO DE  
INGENIERO DE SISTEMAS**

**TÍTULO:**

**“SISTEMA DE INFORMACIÓN WEB PARA LA GESTIÓN DE CURSOS  
LIBRES EN LA FACULTAD DE CIENCIAS Y SISTEMAS”**

**AUTORES:**

<b>Br. Gabriel Ignacio Chávez Toruño</b>	<b>2010-32557</b>
<b>Br. Michael Bryang Chévez Angulo</b>	<b>2010-32661</b>

**TUTOR:**

**Msc. Ing. EVELYN ESPINOZA ARAGÓN**

**MANAGUA, DICIEMBRE DE 2017**



## **Dedicatoria**

Este Trabajo lo dedicamos a Dios porque de Él proviene todo y es el que nos dio fuerzas, sabiduría, inteligencia, paciencia y sobre todo vida para poder llegar a este día tan especial y además siempre nos ayudó a resolver cualquier inconveniente o problema que se nos presentara.

También lo dedicamos a nuestros padres y familiares pues ellos han recorrido todo este largo camino con nosotros y sin su ayuda incondicional este trabajo no habría sido posible.

A nuestros maestros porque sin el conocimiento que nos brindaron no lo podríamos haber logrado.

A nuestra tutora: sin su apoyo esto no habría sido posible

## Resumen

El presente trabajo monográfico tiene como tema principal “Sistema de Información web para la gestión de los cursos libres de la Facultad de Ciencias y Sistemas”. Para su realización, se ha dividido en dos partes: la gestión de requerimientos y el diseño del sistema.

En la primera parte del trabajo, gestión de requerimientos, se describe el entorno de la organización su misión, visión y cómo está formada su estructura organizacional; se detallan las necesidades y fuentes de información, los datos que deberán obtener de éstas y el tipo de instrumento a aplicar para obtener dichos datos, así como las fuentes secundarias que sirven de apoyo al desarrollo de la investigación. Luego, se interpretan los resultados obtenidos al procesar las fuentes.

Una vez identificado el entorno del negocio, así como las necesidades de información se pasa a describir cómo opera el negocio, quiénes son los involucrados y las actividades que llevan a cabo cada uno de ellos, lo que permite definir los procesos realizados a lo interno de la organización en relación a la gestión de los cursos, además se presentan los resultados de las encuestas y entrevistas realizadas a los alumnos y personal de los cursos libres.

En el **estudio de factibilidad técnica** se identificó los equipos y recursos con los que cuenta la facultad, y elaboró una propuesta sobre la infraestructura tecnológica necesaria para que el sistema pueda funcionar correctamente.

Con el propósito de determinar los costos del sistema se utilizó la metodología de COCOMO II, esta herramienta permite cuantificar el software, medir el conjunto de recursos involucrados en el ciclo de desarrollo y medir incluso el proceso de desarrollo. Esto incluye, por tanto, elementos que son directamente medibles (como las líneas de código), y otros que no lo son directamente, sino que son calculados a través de fórmulas o ecuaciones (como el esfuerzo de desarrollo o el coste del proyecto), además de cuantificar los costos de inversión en tecnología.

Para determinar la **factibilidad financiera** y conocer la rentabilidad del proyecto se realizó un flujo de efectivo que contiene los ingresos y gastos más la inversión que deberá realizar la facultad para implementar el sistema. **La factibilidad operativa** se determinó al poseer la estructura organizacional y los recursos humanos suficiente y en el aspecto legal se especificaron las restricciones de uso para sistemas como éste en nuestro país.

En el diseño del sistema se hace uso de las extensiones que provee UWE para desarrollo del modelo de casos de uso, conceptual, de presentación, de navegación, de contenido y el modelo relacional de la base de datos

Respecto a la implementación del sistema, la codificación del prototipo se realizó con Visual Studio Ultimate 2013 en el lenguaje de programación C#, HTML5, CSS3 y JavaScript mediante una solución con dos proyectos: el primero, que aplica el patrón arquitectural Modelo-Vista-Controlador con ASP.net MVC 4 bajo .Net Framework 4.5; y el segundo, que contiene las clases resultantes del mapeo que hace Entity Framework 5 y así como el DbContext para conectar con la base de datos de SQL Server 2014.

La seguridad del sistema se garantizó mediante el cifrado con un algoritmo SHA\_256 de las contraseñas en la base de datos, el inicio de sesión a través de hilos de procesos y el manejo de roles de usuario.

Por último, se realizaron pruebas de caja blanca y caja negra para verificar el funcionamiento óptimo del sistema.

## Índice

Introducción.....	1
Antecedentes.....	3
Justificación.....	5
Objetivos.....	7
Objetivo General: .....	7
Objetivos Específicos: .....	7
Marco Teórico.....	8
1. Definición de sistema de información .....	8
2. Tipos de sistemas de información .....	9
3. Beneficios de los sistemas de información web.....	10
3.1. Calidad de la información. ....	11
3.2. Mejora en los procesos de marketing. ....	11
4. Sistemas web vs sistemas de escritorio.....	13
5. Ingeniería Web. ....	15
5.1. Diferencia entre la ingeniería web y la ingeniería de software .....	15
5.2. Tecnologías de desarrollo web y sus alternativas .....	16
5.3. Patrones de diseño y arquitecturas de software .....	21
5.4. Metodologías para la gestión de un proyecto de SI Web .....	23
6. Sistemas de información web en los procesos de gestión educativos.....	29
Capítulo I: Entorno del negocio.....	30
1.1. Presentación de la organización.....	30
1.1.1. Misión .....	30
1.1.2. Visión.....	30
1.1.3. Estructura organizacional.....	31
1.2. Descripción del sistema del negocio.....	31
1.3. Recopilación de información .....	33
1.3.1. Fuentes primarias.....	34
1.3.2. Fuentes secundarias .....	35
Capítulo II: Gestión de requerimientos .....	36
1.1. Objetivos del sistema.....	39
1.2. Requerimientos funcionales .....	41
1.3. Requerimientos no funcionales.....	47
1.4. Limitaciones y restricciones del sistema .....	54
1.5. Matriz de rastreabilidad .....	55
Capítulo III: Análisis de la viabilidad.....	56
2.1. Estudio técnico. ....	56
2.1.1. Descripción de los requisitos mínimos.....	57
2.1.2. Requisitos para el alojamiento del sistema.....	58
2.2. Análisis económico .....	65
2.3. Análisis financiero .....	76
2.4. Estudio de factibilidad operacional .....	84
2.5. Consideraciones legales .....	86
Capítulo IV: Diseño .....	88

4.1.	Modelo de requerimientos .....	88
4.1.1.	Identificación de actores .....	88
4.1.2.	Modelo de caso de uso del negocio.....	89
4.1.3.	Diagramas de actividad del negocio .....	91
4.1.4.	Modelo del sistema .....	92
4.1.5.	Diagramas de Colaboración. ....	98
4.2.	Modelo de contenido: .....	99
4.3.	Modelo de navegación: .....	100
4.4.	Diagrama de Estados .....	101
4.5.	Modelo de procesos: .....	104
4.6.	Diagrama de Actividad .....	105
4.7.	Modelo de presentación:.....	106
Capítulo V:	Implementación .....	107
5.1.	Modelo físico .....	107
5.2.	Diagrama de componentes.....	107
5.3.	Diagrama de despliegue .....	108
5.4.	Medidas de Seguridad. ....	108
Capítulo VI:	PRUEBAS DEL SOFTWARE.....	109
6.1.	Pruebas de Caja Blanca. ....	109
6.1.1.	Prueba de condición en el sistema web.....	112
6.2.	PRUEBAS DE CAJA NEGRA. ....	113
6.2.1.	Análisis de las validaciones del sistema web .....	114
Conclusiones	.....	117
Recomendaciones	.....	118
Referencias bibliográficas.....		119

## **Anexos**

- 1. Entrevistas realizadas**
- 2. Encuesta realizada a los estudiantes**
- 3. Análisis de los datos recopilados**
- 4. Comunicaciones personales**
- 5. Cotizaciones e información de costos**
- 6. Financiamiento Bancario y calendario de pagos**
- 7. Aceptación del sistema**
- 8. Cálculos del desarrollo del modelo COCOMO II**
  - 8.1. Distribución de tiempo y esfuerzo por etapa**
  - 8.2. Costo de uso de medios técnicos (CUMT)**
  - 8.3. Costo de abastecimiento de materiales**
- 9. Modelo del negocio**
  - 9.1. Diagramas de actividades del negocio**
- 10. Modelo de requerimientos**
  - 10.1. Casos de uso**
  - 10.2. Plantillas de Coleman**
- 11. Modelado del sistema**
  - 11.1. Diagramas de secuencia**
  - 11.2. Diagramas de Colaboración**
  - 11.3. Diagramas navegacionales**
  - 11.4. Diagramas del modelo de presentación**
  - 11.5. Diagramas de estructura de procesos**
  - 11.6. Diagramas de Actividad.**
- 12. Diagrama de Componentes**
- 13. Modelo físico de la base de datos**
- 14. Complejidad ciclomática del sistema web**
- 15. Fuentes secundarias**
- 16. Imágenes del sistema web**
- 17. Código fuente del método putHorarios**

## Introducción

En Nicaragua y especialmente Managua existen varias instituciones tanto públicas (Instituto Nacional Tecnológico, Silvano matamoros, Manuel Antonio Olivares) como privadas (Instituto Latinoamericano de Computación, Universidad Centroamericana) que ofertan a la sociedad cursos técnicos de corta duración, para que los interesados desarrollen sus habilidades, actualicen sus conocimientos y lo apliquen en el mundo laboral.

En la Universidad Nacional de Ingeniería algunas facultades como la de “Ciencias y Sistemas” y “Electrónica y Computación” así como la Dirección de Posgrado ofrecen cursos formativos de corto plazo tanto para los estudiantes de una carrera profesional como para la sociedad en general. La Facultad de Ciencias y Sistemas brinda a la comunidad y público en general desde el año 2003. "cursos libres" con una orientación técnica a capacitar y mejorar conocimientos en las áreas de TI.

Los Cursos Libres de la Facultad de Ciencias y Sistemas enfrentan una serie de inconvenientes que hacen que los procesos tomen tiempo y frenan el crecimiento de este proyecto, por lo que su integración con la facultad no es total.

En este entorno se presenta un retraso para gestionar el volumen de datos generados en los procesos académicos y administrativos asociados a la limitación de personal y la desorganización de la información vinculada al registro manual de datos académicos, así como de costos, gastos y beneficios asociados a cada curso en particular, dificultando las labores que realiza el coordinador de los cursos, además de hacer que los diferentes trámites de notas y certificados no se realicen en un tiempo óptimo.

Debido a esta situación se diseña una investigación para desarrollar un “Sistema de Información web para la Gestión de los Cursos libres en la Facultad de Ciencias y Sistemas”, que fundamentado en la comprensión del entorno organizacional y del negocio permita que la creación de este sistema sea apto para su implementación, el cual asista en los procesos de gestión de notas, estudiantes,

matrículas, docentes, publicidad, agilice los procesos y mejore la atención de una manera más eficiente y eficaz beneficiando a la universidad, a la facultad y a los usuarios a los que se les ofrece los cursos.

## Antecedentes

Los cursos libres que brinda la Facultad de Ciencias y Sistemas empezaron a impartirse en el 2003, propuestos por el Ing. Luis Rocha Jambrina como una forma de extensión y vinculación de la Universidad con la comunidad y personas con deseos de capacitarse y actualizar sus conocimientos, bajo un perfil de enseñanza técnico y a corto plazo. El primer curso impartido fue operador de microcomputadoras, cuyo éxito impulsó la apertura de más cursos.

En la actualidad el coordinador de los cursos libres es el Ing. Danilo Humberto Noguera Rivera. Estos cursos son abiertos con una frecuencia trimestral y la mayoría de ellos son relativos al área de informática y al diseño gráfico. Durante cada temporada se ofertan de 7 a 8 cursos<sup>1</sup>; en la tercera temporada del año 2015 de los cursos libres ingresaron un total de 84 alumnos de los cuales el 67 por ciento eran del sexo masculino y un 33 por ciento del sexo femenino<sup>2</sup>.

Aunque es una iniciativa propia de la facultad, el personal es reducido y al estar este proyecto sustentado mediante ingresos propios, el personal no está dedicado a tiempo completo a la administración de esta organización por servir también a la carrera de Ingeniería de Sistemas, Técnico Superior en Ingeniería de Sistemas y la modalidad sabatina de la carrera.

A nivel universitario los procesos administrativos y académicos en las carreras que sirve esta alma máter, están regidos mediante el Sistema Informático de Registro Académico (**SIRA**) que se encarga de gestionar dichos procesos (horarios, docentes, carga académica, pre-matricula, matrícula, etc.). En contraste, en los cursos libres, el sistema mediante el cual se lleva un registro de información completo de estudiantes, docentes, diplomas expedidos, asignación de horarios, de espacios de laboratorio es de forma manual (no está automatizado) la consulta de los registros, los trámites de notas y certificados toman tiempo. Dichos registros están en Excel y en diversas ocasiones se ha

---

<sup>1</sup> Rocha Jambrina, L. (20 de agosto de 2015). Comunicación Personal

<sup>2</sup> Noguera Rivera, D.H. (11 de febrero de 2016). Comunicación Personal

manifestado la necesidad de gestionar la información de una manera más eficiente.

Para resolver estas situaciones, en noviembre de 2014 se presentó un proyecto web que gestionase las notas y contenidos de los cursos libres por parte de estudiantes del Técnico Superior de Informática y Sistemas; sin embargo el alcance del mismo resultó limitado en contraste con los problemas existentes y no se llegó a implementar.

## Justificación

En la actualidad la inclusión de la tecnología como un soporte transversal en los procesos de las organizaciones resulta una necesidad evidente, sobre todo en el ámbito educativo donde existen grandes volúmenes de información que deben de ser administrados para garantizar la eficiencia de las tareas.

La Facultad de Ciencias y Sistemas cuenta con procesos claramente definidos que se constituyen en los Cursos Libres que sirven a la comunidad. Sin embargo, la mayoría de actividades involucradas se realizan de forma manual por un personal reducido resultando errores en el proceso de la información, retrasos en la emisión de certificados y de informes de notas. Además no se cuenta con un mecanismo que permita determinar qué cursos son los más redituables.

Con este proyecto se pretende analizar los requerimientos en el estado actual del sistema de cursos libres para diseñar y desarrollar un Sistema de Información Web apto para su implementación, el cual asista en los procesos de gestión de notas, estudiantes, matrícula, docentes, además de retroalimentarse con historial para elegir publicidad y calcular beneficios obtenidos por cada curso asistiendo de esta forma a la toma de decisiones.

Este servirá para gestionar grandes cantidades de datos que el personal disponible no puede realizar de manera manual con lo que se reduce el tiempo que se emplea en procesar los datos, así como el número de actividades que deben de realizar para actualizarlos.

El sistema de información le será útil al personal administrativo y al coordinador de los cursos libres de la Facultad de Ciencias y Sistemas pues podrán agilizar sus procesos y tener un control más exacto sobre los ingresos y gastos por curso; los docentes podrán gestionar las notas desde cualquier lugar y dispositivo de su preferencia; y a los estudiantes e interesados en llevar cursos, se les atenderá con mayor rapidez, garantizando la integridad de la información, evitando el extravío de notas y facilitando su rectificación; reduciendo sus gastos de

transporte. Todos los usuarios del sistema tendrán acceso ubicuo a la información requerida.

Esto le será muy útil a la Facultad de Ciencias y Sistemas, pues podrá reducir cuellos de botella, llevando un ágil control de registro de los certificados expedidos y garantizar el acceso ubicuo a la información requerida por el personal.

## Objetivos

### **Objetivo General:**

- Implementar un sistema web que permita la gestión de los cursos libres de la Facultad de Ciencias y Sistemas.

### **Objetivos Específicos:**

- Definir los requisitos funcionales y no funcionales del sistema web para la gestión de los cursos libres de la Facultad de Ciencias y Sistemas sobre la base de la comprensión del entorno organizacional.
- Elaborar los análisis técnico, financiero, económico, operacional y legal para determinar la viabilidad del proyecto.
- Diseñar el sistema web aplicando la metodología de Ingeniería Web Basada en el Lenguaje de Modelado Unificado (UWE).
- Desarrollar un prototipo apto para su despliegue en el entorno operacional
- Aplicar las pruebas necesarias que garanticen el funcionamiento correcto del prototipo del sistema web y su aptitud para ser implantado.

## Marco Teórico

### 1. Definición de sistema de información

La información es un recurso intangible y esencial en los procesos que conforman una organización. Sus falencias pueden influir negativamente en la operatividad de la misma. Al respecto Vega (2005) expresa que la información es considerada uno de los recursos más valiosos de toda organización, no solo un subproducto de la conducción empresarial; pues al retroalimentar, su uso estratégico mediante un sistema que permita su gestión, define ventajas competitivas, su excelencia operacional y por ende su crecimiento. (p. 1)

Laudon & Laudon (2012) afirman que:

Un sistema de información recolecta, almacena y disemina la información proveniente del entorno de la empresa y sus operaciones internas, para apoyar las funciones organizacionales y la toma de decisiones, la comunicación, la coordinación, el control, el análisis y la visualización. Los sistemas de información transforman los datos en bruto y los convierten en información útil a través de tres actividades básicas: entrada, procesamiento y salida. (p. 33)

Debido a la realidad actual es muy difícil que un sistema de información carezca de elementos informáticos pues son un componente fundamental junto con las personas, los datos y la comunicación. Por ello O'Brien & Marakas (2006, p. 6) expresan el concepto como *“cualquier combinación organizada de personas, hardware, software, redes de comunicación y recursos de información que almacene, recupere, transforme y disemine información en una organización”*.

## 2. Tipos de sistemas de información

Cohen y Asín (2000, citado en Rodríguez, 2008, p. 531) plantean que los sistemas de información se pueden clasificar según los niveles de jerarquía organizacional en: sistemas transaccionales, de asistencia a los procesos operativos; sistemas de apoyo a la toma de decisiones que son utilizados por los mandos intermedios, y estratégicos, que permiten definir ventajas operativas para una relación eficiente con el medio externo a la organización. (Rodríguez, 2008, p. 531)

De manera similar, Laudon & Laudon (2012, p. 45) manifiestan que una *“empresa de negocios tiene sistemas para dar soporte a los distintos grupos de niveles de administración. Estos sistemas incluyen sistemas de procesamiento de transacciones (TPS), sistemas de información gerencial (MIS), sistemas de soporte de decisiones (DSS) y sistemas para inteligencia de negocios (BIS).”*

Para diferenciar el software de los SI O'Brien & Marakas (2006) expresan que:

El concepto de recursos de software comprende todos los grupos de instrucciones para el procesamiento de la información. Este concepto genérico de software incluye no sólo los conjuntos de instrucciones operativas llamados programas, los cuales dirigen y controlan el hardware informático, sino también los grupos de instrucciones para el procesamiento de información, llamados procedimientos, que las personas necesitan. (p. 28)

En la actualidad la web ha expandido el alcance de los SI y propiciado el surgimiento de nuevas herramientas para su desarrollo e implementación. Al respecto, Isakowitz & Vitali (1998) manifiestan que la plataforma de la web ha evolucionado constantemente desde sus inicios; en las últimas décadas ha pasado de servir para propósitos de marketing hasta convertirse en un soporte para las funciones organizacionales. Los sistemas de Información basados en web (WIS) son sistemas de información construidos con la tecnología de la red para aprovechar una mayor ubicuidad e interconexión, para servir al trabajo y los procesos internos de una organización (pp. 78-80).

Pressman y Lowe (2009) expresan este concepto de una manera más sencilla: *“Cuando una WebApp es combinada con hardware cliente y servidor, sistemas operativos, software de redes y navegadores, un sistema basado en web emerge.”* (p. 2)

Una WebApp es una categoría de software cuya orientación principal es hacia las redes. Pressman (2010), añade: *“desde que surgió Web 2.0, las webapps están evolucionando hacia ambientes de cómputo sofisticados que no sólo proveen características aisladas, funciones de cómputo y contenido para el usuario final, sino que también están integradas con bases de datos corporativas y aplicaciones de negocios.”* (p. 7)

La naturaleza de las WebApps y los SI basados en web, debido a su naturaleza, se han vuelto más comunes en las empresas. La razón de ello es que las *aplicaciones web constituyen un tipo de sistema de información de arquitectura cliente-servidor. Éstas aprovechan la ventaja de servir datos al usuario independientemente de su ubicación y dispositivo, que ofrece la red y la flexibilidad de sus diferentes tecnologías para que los mismos interactúen con datos que reciben a través de un cliente ligero que es el navegador web.* (Ferrer, 2012, pp. 17-27).

### **3. Beneficios de los sistemas de información web.**

Los sistemas de información tiene múltiples beneficios para una organización estos pueden ser tanto tangibles (aumento en la productividad, costos operacionales más bajos, reducción en costos de distribuidores externos, reducción en costos de oficina y profesionales, tasa reducida de crecimiento en los gastos, etc.) como intangibles (mejora el uso de los activos, el control de los recursos, la planificación organizacional, la información se hace oportuna, mejora la toma de decisiones, las operaciones, mayor satisfacción del cliente, etc.).

### 3.1. Calidad de la información.

Para que un sistema de información sea de verdadera utilidad para una organización debe presentar calidad en su información lo que significa que la misma debe poseer los siguientes atributos: tiempo, contenido y forma (O'Brien J.A. & Marakas G.M., 2006).

- En términos de **tiempo** debe ser oportuna, actualizada, frecuente y estar presente en cualquier instante del tiempo.
- El **contenido** de la información debe ser exacto, que se ajuste a las necesidades del usuario, íntegro, específico y debe ser capaz de medir el desempeño y el alcance de actividades de la organización.
- La **forma** en que se presenta la información debe ser clara, detallada y con cierto orden de forma que el usuario que la vea pueda entenderla de forma fácil y sencilla.

La calidad en la información es uno de los mayores beneficios de los sistemas de información ya que si las características antes mencionadas están presentes en los productos de información harán que la misma no sea obsoleta, inexacta o difícil de entender; a la vez ésta será muy significativa, útil y valiosa tanto para la organización como para las personas que laboran dentro de la misma, lo cual también agregará valor a la empresa.

### 3.2. Mejora en los procesos de marketing.

Otros de los beneficios relevantes de los sistemas de información web son las que mejoras que dan a los procesos de marketing. Los sistemas de información dedicados al funcionamiento de la mercadotecnia ayudan a que las personas encargadas de esa área puedan tener un contacto más ordenado y cómodo con los clientes, se puedan planear y fijar precios de los productos que van a salir al mercado así como en otras actividades que se requieren como publicidad, promoción de ventas, estrategias, investigaciones de mercado y pronósticos (O'Brien J.A. & Marakas G.M., 2006).

Los sitios Web y los servicios de internet/intranets hacen posible un **proceso de mercadotecnia interactiva**, en el que los clientes puedan llegar a ser socios en la creación, comercialización, adquisición y mejora de productos y servicios (O'Brien J.A. & Marakas G.M., 2006).

La mercadotecnia interactiva es un proceso en el cual una organización, empresa o negocio establece relaciones comerciales con sus clientes ya sean preferenciales o potenciales usando como medio de comunicación la internet, intranet y extranet con el objetivo de atraer y mantener clientes lo cuales permitirán a la empresa mejorar o crear los productos o servicios que ofrecen (O'Brien J.A. & Marakas G.M., 2006).

Una de las principales ventajas que implica hacer la publicidad web es que conlleva a un costo menor en comparación a un aviso televisivo y cuyo objetivo principal es facilitarle al usuario que visite el, sitio toda la información necesaria acerca de los productos o servicios que la empresa u organización ofrecen.

Otras ventajas de la publicidad web que se pueden mencionar además de su bajo costo, es el ahorro en el tiempo de producción; debido a que el tiempo ahorrado se puede utilizar para otras actividades productivas para la empresa, también se puede tener acceso a un mayor número de clientes y puesto que las redes sociales ofrecen gran cantidad de datos sobre los usuarios como lo son su edad, localización geográfica, sexo, aficiones, gustos, etc., lo que facilita encontrar las necesidades y preferencias de sus posibles clientes.

Son muchas las ventajas que presentan los sistemas de información web en la parte de marketing lo cuales son una gran herramienta de ayuda para los organizaciones hoy en día.

#### 4. Sistemas web vs sistemas de escritorio.

Existen muchas características que se pueden mencionar tanto de un sistema web como de un sistema de escritorio. En la siguiente tabla se mencionan las más relevantes de ambos:

Características.	Sistema Web	Sistema de escritorio
<b>Personalización, actualización y soporte</b>	Es suficiente con realizar los cambios en el servidor web	Hay que realizarlos en cada estación de trabajo (PC) donde se tenga la aplicación
<b>Accesibilidad y cobertura</b>	Cualquier lugar con acceso a Internet	Solo en el computador donde se haya instalado previamente el software
<b>Capacidad de usuarios concurrentes</b>	Alta debido a la arquitectura de clientes livianos que la pueden usar	Baja ya que la forma de diseño es centrada en un único usuario local
<b>Portabilidad</b>	El sistema puede ser usado con cualquier navegador de Internet	Solo funciona en el sistema operativo en el que está alojado o para el cual fue creado
<b>Infraestructura y movilidad</b>	Solo se tiene que conectar a la Internet	Está restringido a la ubicación del computador local.
<b>Seguridad eléctrica y lógica</b>	Es responsabilidad del proveedor de servicio	Es responsabilidad del administrador de la compañía y de cada usuario que usa el sistema localmente.

Tabla 1: Características de un sistema web y escritorio

Las ventajas de los sistemas web frente a los sistemas de escritorio radican en el amplio espectro de usuario que puede alcanzar independientemente de su ubicación, plataforma de SO que utilicen y sus necesidades particulares permitiendo reducir los costos de instalar y mantener equipos clientes con mayores recursos de hardware debido a que lo mínimo necesario es tener un equipo capaz de ejecutar un navegador web fluidamente. Ferrer (2012), desglosa estas ventajas de la siguiente forma:

- No requiere instalar software especial (en los clientes)
- Bajo coste en actualizar los equipos con una nueva versión
- Acceso a la última y mejor versión
- Información centralizada
- Seguridad y copias de seguridad
- Reducción de costes en los puestos cliente (mayor longevidad)

Los sistemas de escritorio, pueden ser monousuarios, aunque la mayoría utilizan intranets o internet para conectarse a otra localidad; su desarrollo, al ser pensado para cierto hardware más potente que para ejecutar un simple navegador y para una plataforma de SO, es más rápido; además, aprovechan los recursos del equipo para realizar operaciones complejas. Moreno (2011) destaca las siguientes:

- Permite un mejor aprovechamiento del hardware y software del equipo.
- Mejores tiempos de respuesta.
- No dependencia de internet, siempre online.
- Si se desea y el software lo permite, puede trabajar también por internet.
- Movilidad mediante VPNs o terminales remotas.

Las principales ventajas de las aplicaciones de escritorio son el aprovechamiento del hardware, menores costos y tiempos de desarrollo e independencia de internet; sin embargo, las aplicaciones web no requieren un uso intensivo de hardware por lo que la frecuencia de actualización de este es menor, y sus datos están centralizados. La implementación de un sistema web o de escritorio va a depender del tipo de proyecto que se desea implantar.

## 5. Ingeniería Web.

Murugesan, Deshpande, Hansen, & Ginige, A., definen la IWeb así: "La ingeniería web es el establecimiento y el uso de principios científicos rigurosos, de ingeniería y gestión, y enfoques disciplinados y sistemáticos para el desarrollo exitoso, implantación y mantenimiento de sistemas y aplicaciones basados en web que tengan alta calidad". (2001, p. 3)

### 5.1. Diferencia entre la ingeniería web y la ingeniería de software

Todo sistema debe ser sujeto de una serie de principios y métodos que le permitan resolver un problema de manera eficiente; esto es lo que define a una ingeniería.

La Ingeniería Web y la Ingeniería de Software se basan ambas en estratos que aseguran su correcta implementación y el desarrollo de un buen producto de software: calidad (firmeza, fiabilidad y placer al usuario), procesos (que permiten definir que hacer desde comprender el problema hasta su aceptación por el usuario final elaborando los productos de trabajo necesario), métodos (las formas técnicas para llevar a cabo las actividades de los procesos) y, herramientas y tecnologías para la web. (Pressmann & Lowe, 2009, pp.17-20)

A pesar de lo anterior, la diferencia entre la IWeb y la IS radica en el enfoque con el que se aplican sus principios:

Para Pressman y Lowe (2009, pp. 4-7) el desarrollo de Web Apps difiere de la IS contemporánea debido a la propia naturaleza de aquellas:

Los atributos de la mayoría de Web App que las hacen diferentes del software tradicional están determinados por el uso que hagan los clientes de los recursos para lo cual los datos transitan en la red donde se encuentra alojada la aplicación (intensidad de la red), la concurrencia de usuarios, respuesta a una carga de trabajo impredecible por cuanto se puede hacer distinto uso de ella en diferentes momentos, alto desempeño en la carga y procesamiento para mientras el usuario la utiliza, alta disponibilidad, orientación a datos en diferentes contenedores, necesidad de inmediatez de la Web App, estética y calidad en el diseño y contenido y énfasis en la seguridad debido a la gran cantidad de usuarios que pueden acceder.

Powell expresó que los sistemas Web "*envuelven una mezcla entre publicaciones impresas y desarrollo de software, entre marketing y computación, entre comunicaciones internas y relaciones externas, y entre arte y tecnología.*" (Murugesan, y otros, 2001, p. 3)

Debido a esto y los atributos de las Web Apps, antes mencionados, nace la IWeb que se guía siguiendo un enfoque de desarrollo ágil. La ingeniería web permite desarrollar aplicaciones web de calidad industrial a través de una serie de actividades y procesos definidos y estructurados coherentemente que respondan a los constantes cambios que demanda la estrategia del negocio, a nivel organizacional y en las necesidades de los diferentes stakeholders, administrando eficientemente el soporte del cambio en las diferentes etapas por las que pasa la Web App. (Pressman & Lowe, 2009, pp. 12-13).

## **5.2. Tecnologías de desarrollo web y sus alternativas**

Pressman y Lowe (2009, p. 20) expresan que durante las últimas décadas han surgido una serie de herramientas y de tecnologías que han hecho que las Web Apps sean más sofisticadas y penetrantes desde lenguajes de modelado y descripción pasando por los lenguajes de realidad virtual, desarrollo basado en

componentes, navegadores web que cumplen más estándares, así como de lenguajes y componentes de servidor que permiten un desarrollo flexible.

### 5.2.1. Tecnologías y lenguajes del lado del servidor y el cliente.

Los lenguajes del lado del cliente son aquellos que reconoce el navegador web del usuario. Los programas creados con estos, aunque residen en el servidor junto a la página web son transferidos al cliente para su representación (Cobo, 2005):

- HTML: Del inglés, HyperText Markup Language, se trata de un lenguaje de marcas que permite maquetar o estructurar el contenido del sitio web referenciando recursos para brindar una representación enriquecida, según Mateu (2004). La 5ta revisión más importante (HTML5) incorpora nuevas etiquetas para el desarrollo de una web semántica.
- CSS: Según Van Lancker (2013) *"las hojas de estilo son elementos agregados al lenguaje HTML que tomarán en cuenta la presentación del documento o de la aplicación Web"*
- Sobre JavaScript, en Mozilla Developer Network (2015) se expresa que *"es un lenguaje de scripting multiplataforma y orientado a objetos. Es un lenguaje pequeño y liviano. Dentro de un ambiente de host, JavaScript puede conectarse a los objetos de su ambiente y proporcionar control programático sobre ellos"*. Por esta razón es utilizado para enriquecer el contenido de los elementos de un sitio web y mejorar la interactividad con el usuario.
- Angular JS: Es un proyecto de código abierto creado y mantenido por Google, realizado en JavaScript que contiene un conjunto de librerías útiles para el desarrollo de aplicaciones web y propone una serie de patrones de diseño para llevarlas a cabo. En pocas palabras, es un *framework* para el desarrollo, sobre el lenguaje JavaScript con programación del lado del cliente. Una de las principales características de esta librería es que trabaja con el patrón de diseño Modelo-Vista-Controlador (MVC) además que

puede sincronizar los datos entre el modelo y la vista de una manera sencilla por medio de los *data-binding*.

Entre las principales ventajas de este *framework* se pueden mencionar las siguientes:

- Facilita el desarrollo y mantenimiento de aplicaciones de gestión o negocio.
- Potente, sencillo y extensible debido a que puede incorporar una serie de servicios web y librerías de terceros
- La mayoría de las funcionalidades de aplicación web se trasladan o radican en el lado cliente (*front-end*) lo que reduce la programación en el lado del servidor (*back-end*) lo que conlleva a aplicaciones web mucho más ligeras.
- No tiene dependencia de las librerías JQUERY
- Tiene su propia forma de enrutamiento la cual trabaja del lado del cliente por lo que no hay que hacer solicitudes de HTML al servidor

Los lenguajes del lado del servidor son reconocidos, compilados e interpretados por este, que debe gestionar elementos externos al equipo cliente y devolverlos en un formato comprensible para el navegador web. Son empleados cuando por razones de seguridad los cálculos no se pueden realizar en la computadora del usuario, o se debe acceder a información centralizada, situada en una base de datos en el servidor, (...) que maneja toda la información de las BD y cualquier otro recurso, como imágenes o servidores de correo y luego envía al cliente una página web con los resultados de todas las operaciones (Puente, n.d.)

- ASP.NET: Es un conjunto de tecnologías desarrolladas por Microsoft que permite desarrollar aplicaciones Web orientadas a la empresa reduciendo la creación de código. *"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."*(Microsoft Developer Network, 2007)

- PHP, según Cobo (2005) es un lenguaje open source interpretado del lado del servidor para ser robusto, versátil y modular cuyos programas ejecutados en el servidor son embebidos en el código HTML al enviar al cliente. Es multiplataforma y compatible con diferentes bases de datos.
- C# es un lenguaje de programación orientado a objetos que se ha diseñado para compilar diversas aplicaciones que se ejecutan en .NET Framework, permitiendo desarrollar aplicaciones webs del lado del servidor haciendo uso de las clases y Apis de aquel. (MSDN, 2015)

### **5.2.2. Motores de base de datos.**

Entre los motores de base de datos más utilizados se encuentran:

- SQL Server, desarrollado por Microsoft para el entorno empresarial para la gestión de bases de datos relacionales en una misma computadora o en red mediante el uso de Transact SQL y ANSI SQL (MSDN, 2015)
- MySQL es un SGBD de alto rendimiento, desarrollado en C/C++ por Oracle Corporation, multiplataforma y con manejo de usuarios y privilegios. Debido a esto es muy usado en internet. (Casillas, Ginesta y Mora, pp.:5-71 2007)

### 5.2.3. Entornos de desarrollo integrados (IDE).

Alonzo (2010) expresa que un IDE (acrónimo en inglés de integrated development environment), “es un programa informático compuesto por un conjunto de herramientas de programación. Este puede dedicarse en exclusiva a un sólo lenguaje de programación o bien, poder utilizarse para varios”. Sus componentes como el editor de código, intérprete, compilador, depurador y diseñador de GUI permiten desarrollar una gran variedad de aplicaciones.

A continuación, en la siguiente tabla<sup>3,4,5</sup> se muestran los IDE más comunes que se usan hoy en día:

IDE	Características	Lenguajes soportados	Sistema operativo	Proyectos que pueden desarrollar
<b>Visual Studio 2015</b>	Permite a desarrollar aplicaciones escritorio web y móvil en cualquier entorno que soporte la plataforma .NET.	C++, C#, Visual Basic .NET, F#, Java, Python, Ruby, PHP y ASP.NET	Windows.	Web, Escritorio, y móviles
<b>Eclipse</b>	Dispone de un Editor de texto con un analizador sintáctico, Tiene pruebas unitarias con JUnit, control de versiones con CVS, integración con Ant, asistentes ( <i>wizards</i> ) para creación de proyectos, clases, tests, etc., y refactorización. Asimismo. integración Hibernate. <sup>5</sup>	Java, ANSI C, C++, JSP, SH, PERL, PHP y SED.	Multiplataforma	Web y escritorio
<b>Netbeans</b>	Gestiona: la interfaz de usuario, configuración de usuario, almacenamiento, ventanas, y Marcos Asistente	JAVA, JSP, HASKELL, CPP, SH, YAC, LEX, PEARL, ANSI C y OBJC	Multiplataforma	Web y escritorio

Tabla 2: Características de algunos IDE

<sup>3</sup> Netbeans Community (2016). *Netbeans IDE-Overview*. Recuperado de <https://netbeans.org/features/index.html>

<sup>4</sup> Microsoft (2016). *IDE de Visual Studio*. Recuperado de <https://www.visualstudio.com/es/vs/>

<sup>5</sup> Fácil Cloud (2016). *Netbeans vs Eclipse ¿Cuál elegir?* Recuperado de <https://www.facilcloud.com/noticias/netbeans-o-eclipse-cual-elegir/>

## 5.3. Patrones de diseño y arquitecturas de software

### 5.3.1. Patrones de Diseño.

Describen una estructura de diseño que resuelve un problema particular dentro de un contexto específico y entre “fuerzas” que afectan la manera en la que se aplica y en la que se utiliza dicho patrón. (Pressman, 2010)

Un patrón de diseño se caracteriza como “una regla de tres partes que expresa una relación entre cierto contexto, un problema y una solución”

Los patrones de diseño se clasifican en:

- **Creacionales:** se centran en la “creación, composición y representación” de objetos.
- **Estructurales:** se centran en problemas y soluciones asociados con la manera en la que se organizan e integran las clases y objetos para construir una estructura más grande.
- **Conductuales:** se enfocan a problemas asociados con la asignación de responsabilidad entre los objetos y a la manera en la que se efectúa la comunicación entre ellos.

### 5.3.2. Arquitectura del software

La arquitectura del software de un programa o sistema de cómputo es la estructura o estructuras del sistema, lo que comprende a los componentes del software, sus propiedades externas visibles y las relaciones entre ellos. Esta permite a su vez:

- Analizar la efectividad del diseño para cumplir los requerimientos establecidos.
- Considerar alternativas arquitectónicas en una etapa en la que hacer cambios al diseño todavía es relativamente fácil.
- Reducir los riesgos asociados con la construcción del software.

### 5.3.2.1. Arquitectura Cliente/Servidor

Según Salazar (2010) la frase cliente / servidor define la forma en que se relacionan las estaciones de trabajo a través de las redes de comunicación. En ella, el cliente desea llevar a cabo una operación; en vez de realizarla por sí solo, le traslada esa operación al servidor, el cual recibe la petición a través de algún medio de comunicación y se encarga de realizarla y le devolverá un resultado.

Una de las arquitecturas más avanzadas de tipo cliente / servidor es la llamada arquitectura de tres capas. Los componentes de una arquitectura de tres capas son los siguientes (Salazar, 2010):

- **Presentación:** es la capa con la que interactúa el usuario. En este caso, está formada por los formularios con los que interactúa el usuario
- **Lógica de negocio:** esta capa está formada por las entidades empresariales, que representan objetos que van a ser manejados o consumidos por toda la aplicación. Es el puente de comunicación entre la capa de datos y la capa de presentación, aquí se verifica la información enviada por la capa anterior.
- **Datos:** aquí la información es grabada en una base de datos, por medio de clases que interactúan con estas y que realizan todas las operaciones con la base de datos de forma transparente para la capa de negocio.

## 5.4. Metodologías para la gestión de un proyecto de SI Web

### 5.4.1. ¿Qué metodología se puede usar para desarrollo de un proyecto web?

Según Loa Fragoso (2011) el proceso que se debe de elegir para el desarrollo de un sitio Web depende del tipo de proyecto a implementar o los acuerdos a los que se lleguen con el cliente. Se puede elegir un proceso ágil que produzca liberaciones de aplicaciones Web a un ritmo acelerado, o uno incremental para un proceso más lento y más elaborado, que necesite ser más detallado y analizado con detenimiento.

Las metodologías más recomendadas para el desarrollo de proyectos Web están constituidas por modelos de procesos ágiles, ya que las WebApps suelen tener actualizaciones constantemente, por lo que el proceso debe tener ciclos de desarrollo cortos, no obstante no se limita el uso de metodologías tradicionales como RUP las cuales llevan una documentación extensiva y mucha planificación. (Loa Fragoso, 2011). Para seleccionar una se deben considerar los siguientes puntos (Pressman, 2009).

#### **En los procesos ágiles:**

- **Las entregas se hacen de manera incremental.** las actividades del marco de trabajo ocurrirán de manera repetida conforme cada incremento se someta a ingeniería y se entregue.
- **Cambios frecuentes.** Los cambios pueden ocurrir como resultado de la evaluación de un incremento entregado o como consecuencia de cambiar las condiciones de los negocios.
- **Los plazos de entregas son muy cortos.** lo que disminuye la creación y revisión de gran cantidad de documentación, aunque no excluye la simple realidad de que el análisis crítico, el diseño y la prueba deban registrarse en alguna forma.

### En los procesos tradicionales:

- **Se busca un desarrollo de software predecible.** Lo que hace susceptible a los cambios ya que sus predicciones se centran en la documentación que se elabora excluyendo posibles cambios que puedan ocurrir en la organización o el negocio
- **Se hace énfasis en la documentación.** Con el fin de establecer una planificación y dar un seguimiento riguroso a muchas actividades que se darán durante el desarrollo del software

#### 5.4.1.1. Proceso Unificado de desarrollo (RUP)

El Proceso Unificado (también llamado Proceso Unificado Racional), es una metodología que reúne muchas de las mejores prácticas del desarrollo de software y que haciendo uso del UML y del diseño orientado a objetos brinda un enfoque iterativo e incremental para asegurar la calidad del software. (Rational Software, 2001).

Según Loa Fragoso (2011), un proyecto RUP define la fase Inicial, de comunicación con el cliente para identificar los requerimientos y planear los incrementos; la fase de elaboración para modelar los procesos y representar la arquitectura mediante UML; la fase de construcción donde se desarrollan e integran los componentes de software y manuales de usuario; y la fase de transición donde se despliega preliminarmente el sistema, definiendo los procesos de instalación y liberación final.

En el RUP también se describe actividades de trabajo como escribir casos de uso en disciplinas, donde una disciplina es un conjunto de actividades y artefactos relacionados en un área determinada (Fragoso, 2011):

- **Modelado empresarial:** Tiene como objetivo comprender la estructura y la dinámica de la organización, comprender problemas actuales e identificar posibles mejoras.

- **Requisitos:** Tiene como objetivo especificar o establecer lo que el sistema debe hacer, también define los límites del sistema, y una interfaz de usuario, realiza una estimación del costo y tiempo de desarrollo.
- **Análisis y Diseño:** Define la arquitectura del sistema y tiene como objetivo trasladar los requisitos en implementación.
- **Implementación:** Su objetivo es que los componentes se implementen en las clases de diseño, se asignen a los nodos, prueben individualmente e integrarlos en un sistema ejecutable (enfoque incremental).
- **Prueba:** Verifica la integración de los componentes, requisitos implementados y defectos resueltos antes de la distribución.
- **Despliegue:** Asegura que el producto está preparado para el cliente y procede a su entrega.
- **Gestión de Cambios y configuración:** Es esencial para controlar el número de artefactos producidos por la cantidad de personal que trabaja en un proyecto conjuntamente.
- **Gestión de proyectos:** Su propósito consiste en proveer pautas para: Administrar proyectos de software intensivos, planear, dirigir personal, ejecutar acciones, supervisar proyectos y administrar el riesgo
- **Entorno:** Su propósito es proveer a la organización que desarrollará el software, un ambiente en el cual basarse, el cual provee procesos y herramientas para poder desarrollar el software.

Fragoso (2011) expresa que estas disciplinas se distribuyen en cada una de las fases a lo largo del tiempo (Véase Ilustración 2):

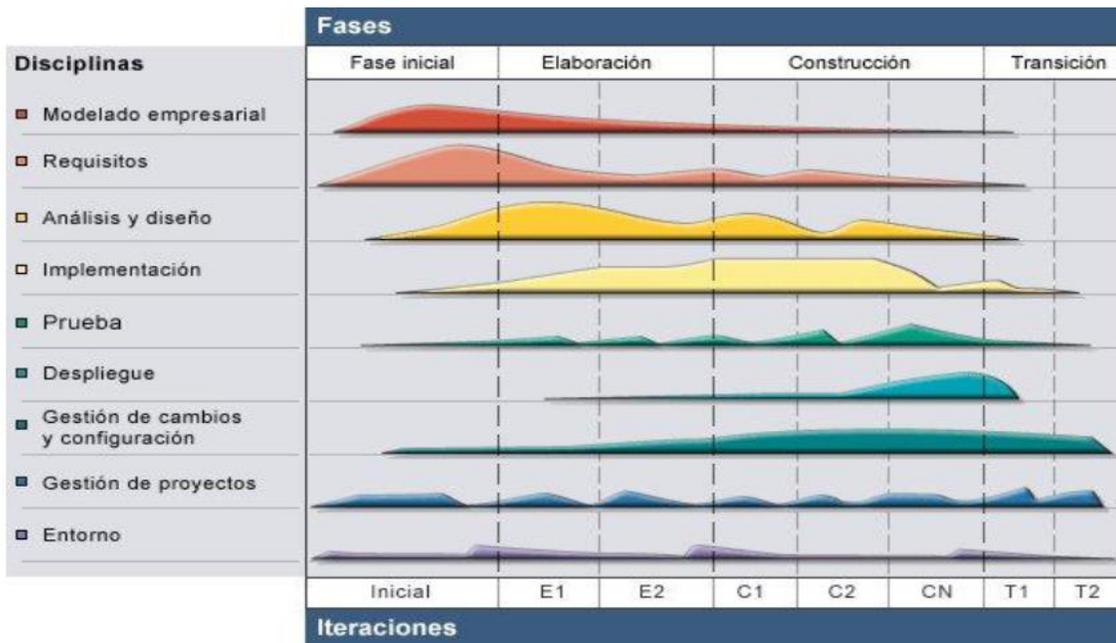


Figura 1: Ciclo de vida de RUP

Cuando se desarrollan estas disciplinas se genera algunos artefactos o productos de trabajos por cada fase del desarrollo del software (Fragoso, 2011):

Fase Inicial	Fase de elaboración	Fase de construcción	Fase de transición
<ul style="list-style-type: none"> <li>• Documento de la visión</li> <li>• Modelado inicial de casos de uso</li> <li>• Glosario inicial del proyecto</li> <li>• Caso inicial de negocio</li> <li>• Evaluación inicial del riesgo</li> <li>• Plan de proyecto, fases e iteraciones</li> <li>• Modelado del negocio</li> <li>• Uno o más prototipos</li> </ul>	<ul style="list-style-type: none"> <li>• Modelado de casos de uso</li> <li>• Requisitos suplementarios</li> <li>• Modelo de análisis</li> <li>• Descripción de la arquitectura de software</li> <li>• Prototipo arquitectónico</li> <li>• Modelo de diseño preliminar</li> <li>• Lista de riesgos</li> <li>• Plan de proyecto</li> </ul>	<ul style="list-style-type: none"> <li>• Modelado de diseño</li> <li>• Componentes del software</li> <li>• Incremento integrado del software</li> <li>• Plan y procedimientos de pruebas</li> <li>• Casos de pruebas</li> <li>• Documentación: del soporte, manuales de usuario, manuales de instalación, descripción del incremento actual.</li> </ul>	<ul style="list-style-type: none"> <li>• Incremento de software integrado</li> <li>• Reportes de las pruebas beta</li> <li>• Retroalimentación general del usuario.</li> </ul>

Figura 2: Artefactos originados por RUP

El subconjunto de artefactos se selecciona según las necesidades particulares del proyecto a desarrollar.

#### 5.4.1.2. Programación Extrema

Para Loa Fragoso (2005) XP (del inglés *extreme programming*), un modelo ágil, sigue la combinación de programar probando primero y desarrollo iterativo. Busca la satisfacción del cliente y la entrega temprana de software incremental con una

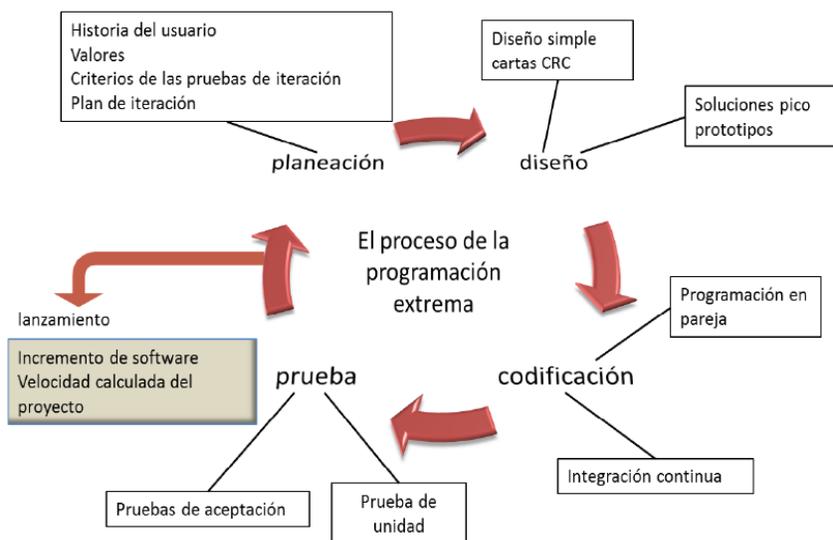


Figura 3: Ciclo de vida de XP

simplicidad general del desarrollo. Considera la codificación como la actividad principal. Su particularidad es tener como parte del equipo, al usuario final, para llegar al éxito del proyecto (Véase Ilustración 4).

Algunas de las características de XP son las siguientes:

- **Pruebas unitarias:** se basa en las pruebas realizadas a los principales procesos, de tal manera que se adelanta en algo hacia el futuro, para descubrir posibles errores.
- **Re-fabricación:** se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- **Programación en pares:** una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto, mientras uno conduce, el otro consulta el mapa.

#### 5.4.2. UWE-UML extendido para la web

UWE es un enfoque, una metodología basada en IS utilizada en la IWeb para el desarrollo de aplicaciones web utilizando UML y retomando RUP.

UWE detalla el proceso de autoría de aplicaciones con una definición exhaustiva del proceso de diseño que debe ser utilizado. Este proceso, iterativo e incremental, incluye flujos de trabajo y puntos de control, y sus fases coinciden con las propuestas en el Proceso Unificado de Modelado (Quiroga, 2015).

En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito.

Quiroga, G. y Coronel, F. (2013) expresan que algunas ventajas para considerar UWE son:

- Permite el desarrollo de aplicaciones y sistemas web que proveen páginas más apropiadas, que se adapten al usuario a partir de su dispositivo, preferencias o contexto.
- Esta metodología orientada a aspectos garantiza las demandas de cada usuario en particular separando sus requerimientos, enfoques, interfaces, adaptabilidad.
- Define un conjunto de procesos durante las etapas del desarrollo para mantener la integridad del diseño y la funcionalidad del sistema

El modelo que propone UWE está compuesto por seis etapas o sub-modelos<sup>6</sup>:

- **Modelo de casos de uso:** captura los requisitos del sistema.
- **Modelo de contenido:** conceptualiza del desarrollo del sistema
- **Modelo de usuario:** es modelo de navegación, en el cual se incluyen modelos estáticos y modelos dinámicos.
- **Modelo de estructura:** en el cual se encuentra la presentación del sistema y el modelo de flujo.
- **Modelo abstracto:** incluye el modelo de interfaz de usuario y el modelo de ciclo de vida del objeto.
- **Modelo de adaptación**

En cuanto a los requisitos, UWE los clasifica dependiendo del carácter de cada uno. Además distingue entre las fases de captura, definición y validación de requisitos e integra funcionalidades que abarcan áreas relacionadas con la web como la navegación, presentación, los procesos de negocio y los aspectos de adaptación.

---

<sup>6</sup> De la Rosa, M. J. (2010). *Estudio de UWE (UML-Based Web Engineering)* (p. 3). Madrid: Universidad Carlos III de Madrid

## 6. Sistemas de información web en los procesos de gestión educativos.

Es importante considerar las ventajas que permite un sistema de información para la gestión de procesos educativos, considerando los elementos que se encuentran involucrados, es decir Programas, Cursos, Docentes, Estudiantes, autoridades académicas, entre otros.

Las plataformas de gestión académicas para cursos, apoyan la gestión de los mismos, permitiendo administrarlos y mantenerlos tanto en datos como en registros para los cursos, modalidades, versiones, malla curricular, docentes, estudiantes, recursos y documentación sobre la gestión.

Los sistemas de información que se dedican a la gestión de sistemas escolares o cursos educativos cumplen con ciertas características: integran diferentes subsistemas (Módulos de finanzas, Modulo de matrícula, Módulos de gestión de notas, Docentes, etc.), lo que permite a la vez que cada persona de la institución pueda emplear y acceder para realizar diferentes actividades, a la vez se adapta a la ambiente de la organización y por ultimo da soporte a las actividades administrativas de la institución escolar o que gestionan de cursos académicos (Essink L., & Visscher A., 1987).

Algunos de los procesos más comunes que se automatizan en instituciones de académicas son los siguientes:

- **Matricula:** permite tener los datos sobre estudiantes y monitorear aulas, cursos y profesores o tutores asignados a los mismos.
- **Gestión de notas:** consultar notas por parte del estudiante así como ingresar y mantener un control de las mismas por parte del personal docente con la posibilidad de generar reportes de estas.
- **Gestión de cursos:** permite crear y asignar docentes por curso creando en ese mismo instante un acta de notas para dicho curso en particular.
- **Control de pagos:** se monitorea los pagos académicos u obligatorios que tiene que realizar el estudiante para obtener derecho al servicio con la facilidad de hacer estos pagos en cuotas con fechas establecidas.

## **Capítulo I: Entorno del negocio**

### **1.1. Presentación de la organización**

Los Cursos Libres que brinda la Facultad de Ciencias y Sistemas comenzaron a impartirse en 2003 como una iniciativa de extensión y vinculación de la universidad con la comunidad y personas con deseos de capacitarse y actualizar sus conocimientos, bajo un perfil de enseñanza técnico y a corto plazo. Para brindar sus servicios utiliza las mismas instalaciones de la Facultad.

#### **1.1.1. Misión**

Somos una unidad de la Facultad de Ciencias y Sistemas dedicada a capacitar y actualizar conocimientos en el área informática y tecnológica mediante la oferta de cursos de corta duración accesibles a los estudiantes y sociedad en general.

#### **1.1.2. Visión**

Ser líder y referente de la extensión universitaria en el país a través de la formación continua y técnica de excelencia, mediante la enseñanza de tecnologías informáticas modernas, para contribuir al desarrollo de la sociedad nicaragüense.

### 1.1.3. Estructura organizacional

Aunque no se ha formalizado (en papel), existe una estructura organizacional que se inserta dentro de la facultad-, y que se presenta como un organigrama por funciones en la Figura 1:

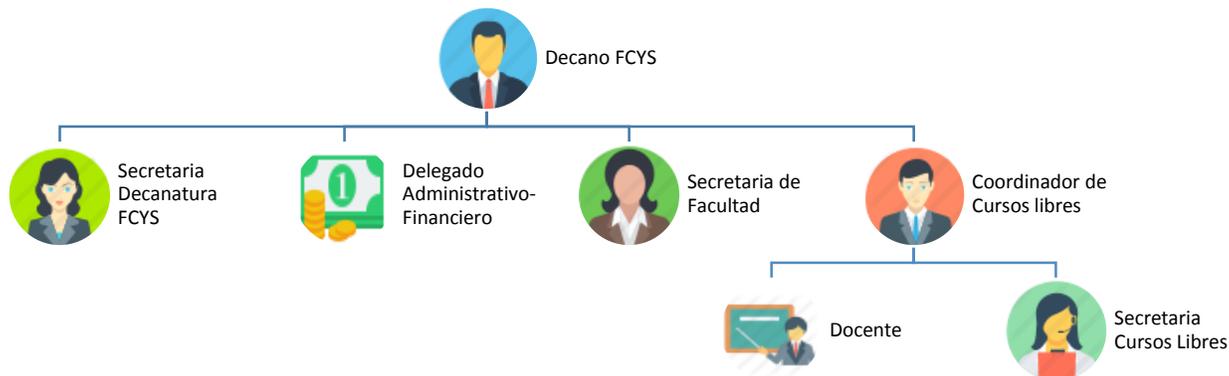


Figura 1: Organigrama de la Unidad de Cursos Libres de la Facultad de Ciencias y Sistemas (fuente: elaboración propia).

### 1.2. Descripción del sistema del negocio

Los cursos que se ofrecen son impartidos en una temporada que dura un trimestre. Al año se abren tres. De los 15 cursos que se proponen son aprobados 8 o 7, su aprobación depende de la cantidad de estudiantes que paguen por dicho curso. Para la aprobación de un curso se necesitan que el curso tenga un mínimo de 10 estudiantes inscritos pero el máximo de estudiantes que por curso son 25.

Previo a que se comience a impartir clases en una temporada de cursos libres (cada temporada tiene una duración de 3 meses, resultando 3 temporadas por año por los periodos de planificación) realiza el proceso de matrícula en el cual el estudiante debe presentar una serie de documentos que son requeridos para ser registrado, asignándolo al grupo del curso elegido.

El costo de cada curso para los estudiantes en general, es de C\$ 3,000 que puede ser pagado en 4 cuotas de C\$750 (quincenal o mensual). La primera cuota es en la práctica, la matrícula. El coordinador y su secretaria son los únicos autorizados

para matricular a un alumno. Los recibos y el efectivo recaudados en la gestión de pagos son entregados al delegado financiero del Decano.

Una vez matriculado el educando, puede comenzar a recibir sus clases del curso correspondiente (las cuales se guían por un plan temático), siendo registrada su asistencia por el docente. Esta asistencia es entregada cada día que se imparte el curso a la secretaria del coordinador.

Actualmente, cada docente determina la mejor forma de evaluar, según las características del curso, con las cual los alumnos deben demostrar sus capacidades y sus conocimientos adquiridos. Al final del curso, la secretaria de los cursos libres entrega un formato de acta de notas, que contiene la sumatoria de las evaluaciones; el acta es firmada por el coordinador y la secretaria académica de la Facultad de Ciencias y Sistemas. Llegado a este punto, si el estudiante ha alcanzado una calificación mayor o igual a 70 puntos y ha cancelado todas sus cuotas puede solicitar su certificado de aprobación pagando C\$100. Por último, el acta de notas es registrada en un libro, sellada y firmada por la secretaria académica, quedándose en posesión de él.

En cuanto al manejo de documentos, la principal persona encargada de mantener el registro es la secretaria de los cursos libres ella se encarga de llevar un control sobre los siguientes documentos: control de pagos, costo de publicidad, asistencia de los alumnos por curso, formato de acta de notas, listado de emisiones, impresión y entrega de diplomas, constancias de participación de los alumnos en los cursos.

La secretaria de los cursos libres, también, es la encargada de elaborar y enviar a cada docente que imparten los cursos por correo electrónico el acta de notas para que estos llenen este formato y luego se la envíe a ella para ella después pueda enviar dichas actas a decanatura para su respectiva firma.

Ocasionalmente los alumnos solicitan información acerca de sus notas, para obtener dicha información ellos consultan al docente que le imparte el curso. Y

también hay solicitudes de información de cada curso para esto se abocan con la secretaria de los cursos libres.

Antes de que finalice una temporada de cursos se definen los siguientes a impartir durante la próxima temporada; para ello cada docente puede proponer al coordinador algún curso, detallando el plan temático del mismo, y el horario en el que este puede ser impartido. Junto a esto el coordinador, en base a referencias históricas de mercado elabora una lista final que es enviada al decano para que autorice la erogación de los gastos publicitarios.

Se puede resumir lo anteriormente descrito en los siguientes puntos:

- Creación de grupos por curso
- Asignación de docentes
- Gestión de Matricula
- Gestión de pagos
- Asignación de grupo
- Asignación de horarios
- Creación de reportes
- Gestión de notas e historial académico

### **1.3. Recopilación de información**

La metodología de la Ingeniería Web basada en el Lenguaje de Modelado Unificado, UWE, tiene como primera fase el Análisis de requisitos, que consiste en la captura y definición de los requerimientos funcionales y no funcionales para reflejarlos en un modelo de casos de uso.

Para recopilar esta información se aplicaron entrevistas y encuestas, permitiendo comprender y documentar los requisitos de cada uno de los procesos a automatizar.

### 1.3.1. Fuentes primarias

Incluyen datos que se generan con el proyecto; hay que aplicar un instrumento para la obtención de información. La siguiente tabla detalla las fuentes primarias y las necesidades de información de la organización:

<b>Necesidades de Información</b>	<b>Datos</b>	<b>Fuente</b>	<b>Instrumento</b>
<b>Matrícula</b>			
Saber las cantidades de estudiantes que se matriculan.	<ol style="list-style-type: none"> <li>Número de estudiantes que se matriculan por curso.</li> <li>Cupos máximos disponibles por curso</li> </ol>	<ol style="list-style-type: none"> <li>Coordinador de los cursos libres</li> </ol>	<ol style="list-style-type: none"> <li>Entrevista</li> </ol>
Tiempo necesario para que un estudiante se matricule.	<ol style="list-style-type: none"> <li>Tiempo de espera para que el estudiante sea atendido</li> <li>Número de veces que llega el estudiante.</li> <li>Tiempo que toma la matrícula.</li> </ol>	<ol style="list-style-type: none"> <li>Estudiante.</li> <li>Secretaria de los cursos libres.</li> </ol>	<ol style="list-style-type: none"> <li>Encuesta.</li> <li>Entrevista</li> </ol>
La satisfacción del estudiante	<ol style="list-style-type: none"> <li>Grado de satisfacción en la atención durante la matrícula.</li> <li>Grado de satisfacción con los procesos</li> </ol>	<ol style="list-style-type: none"> <li>Estudiante</li> </ol>	<ol style="list-style-type: none"> <li>Encuesta.</li> </ol>
<b>Gestión de procesos administrativos</b>			
El flujo de los procesos	<ol style="list-style-type: none"> <li>Actividades de cada proceso</li> <li>Duración de cada actividad</li> <li>Personal involucrado</li> <li>Entradas y salidas de los procesos</li> </ol>	<ol style="list-style-type: none"> <li>Decano</li> <li>Coordinador de los cursos libres</li> <li>Estudiantes</li> </ol>	<ol style="list-style-type: none"> <li>Encuesta</li> </ol>
Actores y empleados que intervienen en los procesos administrativos y académicos	<ol style="list-style-type: none"> <li>Roles de los empleados</li> <li>Cantidad de empleados por funciones</li> </ol>	<ol style="list-style-type: none"> <li>Decano</li> <li>Coordinador de los cursos libres</li> </ol>	<ol style="list-style-type: none"> <li>Entrevista</li> </ol>
Gastos e ingresos	<ol style="list-style-type: none"> <li>Punto de Equilibrio de los cursos</li> <li>Cantidad de estudiantes matriculados por curso</li> </ol>	<ol style="list-style-type: none"> <li>Coordinador de cursos libres</li> <li>Asistente financiero</li> </ol>	<ol style="list-style-type: none"> <li>Entrevista</li> </ol>
<b>Tecnología</b>			
Tipo de equipo de comunicación. Conocer el presupuesto para equipos tecnológicos.	<ol style="list-style-type: none"> <li>Tipo de infraestructura.</li> <li>Compatibilidad en infraestructura física con la tecnología.</li> <li>Equipos de cómputo disponible.</li> <li>Ancho de banda disponible</li> </ol>	<ol style="list-style-type: none"> <li>Personal DTIC</li> </ol>	<ol style="list-style-type: none"> <li>Entrevista</li> </ol>

Tabla 3: Necesidades de información de en los Cursos Libres (fuente: elaboración propia)

### **1.3.2. Fuentes secundarias**

Incluyen datos que ya están disponibles como documentos, facturas, revistas etc. Las fuentes secundarias obtenidas fueron los formatos de las actas de notas y la hoja de asistencia inicial, proporcionadas por la secretaria de los cursos libres (véase anexo 15). Dichos documentos constituyen suministros y salidas de procesos (esto se detalla posteriormente en los casos de uso).

## Capítulo II: Gestión de requerimientos

La gestión de requerimientos asegura que un proyecto cumple con las expectativas de sus clientes e interesados, porque permite la recopilación de información útil sobre los procesos y las necesidades de cada uno de los agentes o usuarios que intervienen en estos permitiendo realizar un mejor diseño del sistema.

Un requerimiento o requisito es una característica que debe de poseer un sistema para satisfacer las necesidades del cliente. *“Los requerimientos para un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. Estos requerimientos reflejan las necesidades de los clientes de un sistema que ayude a resolver algún problema.”* (Sommerville, 2005).

En este sentido, la Ingeniería de Requerimientos es el proceso de descubrir, analizar, documentar y verificar estos servicios y restricciones.

Para definir los requerimientos del sistema se hará uso del software REM 1.2.2 para una fácil comprensión y mejor organización de los datos para establecer los requerimientos del software.

Antes de definir los requerimientos, se presentan los participantes (personas y organizaciones) directos o indirectos en el proceso de análisis de requerimientos del sistema. Se incluye a los desarrolladores, clientes y usuarios.

### A. Organizaciones

<b>Organización</b>	<b>Unidad de Cursos Libres de la FCYS</b>
<b>Dirección</b>	Semáforos Villa Progreso 300 mts E. Recinto Universitario Pedro Aráuz Palacios.
<b>Teléfono</b>	(505)22496429
<b>Fax</b>	<b>PD</b>
<b>Comentarios</b>	Ninguno

Tabla 4: Datos de la Unidad de Cursos Libres de la FCYS como organización (elaboración propia, fuente: Facultad de Ciencias y Sistemas).

<b>Organización</b>	<b>Grupo desarrollador</b>
<b>Dirección</b>	Managua, Nicaragua
<b>Teléfono</b>	<b>PD</b>
<b>Fax</b>	<b>PD</b>
<b>Comentarios</b>	Ninguno

Tabla 5: Organización grupo de desarrollo (fuente: elaboración propia).

#### A. Desarrolladores

<b>Participante</b>	<b>Gabriel Ignacio Chávez Toruño</b>
<b>Organización</b>	<a href="#">Grupo desarrollador</a>
<b>Rol</b>	Analista-programador
<b>Es desarrollador</b>	Sí
<b>Es cliente</b>	No
<b>Es usuario</b>	No
<b>Comentarios</b>	Analista programador del sistema de información

Tabla 6: Participante Gabriel Ignacio Chávez Toruño (fuente: elaboración propia).

<b>Participante</b>	<b>Michael Bryang Chávez Angulo</b>
<b>Organización</b>	<a href="#">Grupo desarrollador</a>
<b>Rol</b>	Analista-Programador
<b>Es desarrollador</b>	Sí
<b>Es cliente</b>	No
<b>Es usuario</b>	No
<b>Comentarios</b>	Analista programador del sistema de información

Tabla 7: Participante Michael Bryang Chávez Angulo (fuente: elaboración propia).

## B. Clientes y usuarios

<b>Participante</b>	<b>Carlos Sánchez</b>
<b>Organización</b>	<a href="#">Unidad de Cursos Libres de la FCYS</a>
<b>Rol</b>	Decano de la Facultad de Ciencias y Sistemas
<b>Es desarrollador</b>	No
<b>Es cliente</b>	Sí
<b>Es usuario</b>	Sí
<b>Comentarios</b>	Ninguno

Tabla 8: Participante Carlos Sánchez (fuente: elaboración propia)

<b>Participante</b>	<b>Danilo Humberto Noguera Rivera</b>
<b>Organización</b>	<a href="#">Unidad de Cursos Libres de la FCYS</a>
<b>Rol</b>	Coordinador de Cursos Libres FCYS
<b>Es desarrollador</b>	No
<b>Es cliente</b>	Sí
<b>Es usuario</b>	Sí
<b>Comentarios</b>	Ninguno

Tabla 9: Participante Danilo Humberto Noguera Rivera (fuente: elaboración propia).

<b>Participante</b>	<b>Alejandro Vallecillo</b>
<b>Organización</b>	<a href="#">Unidad de Cursos Libres de la FCYS</a>
<b>Rol</b>	Delegado Administrativo-financiero
<b>Es desarrollador</b>	No
<b>Es cliente</b>	Sí
<b>Es usuario</b>	Sí
<b>Comentarios</b>	Ninguno

Tabla 10: Participante Alejandro Vallecillo (fuente: elaboración propia).

<b>Participante</b>	<b>Rebeca Bonilla</b>
<b>Organización</b>	<a href="#">Unidad de Cursos Libres de la FCYS</a>
<b>Rol</b>	Asistente del Coordinador de Cursos Libres
<b>Es desarrollador</b>	No
<b>Es cliente</b>	Sí
<b>Es usuario</b>	Sí
<b>Comentarios</b>	Ninguno

Tabla 11: Participante Rebeca Bonilla (fuente: elaboración propia).

<b>Participante</b>	<b>Estudiante de los cursos libres</b>
<b>Organización</b>	<a href="#">Unidad de Cursos Libres de la FCYS</a>
<b>Rol</b>	Estudiante
<b>Es desarrollador</b>	No
<b>Es cliente</b>	No
<b>Es usuario</b>	Sí
<b>Comentarios</b>	Ninguno

Tabla 12: Participante estudiante (fuente: elaboración propia).

### 1.1. Objetivos del sistema

Los siguientes son los objetivos que se esperan alcanzar cuando el sistema web a desarrollar esté en producción:

- Gestionar los cursos libres impartidos
- Gestionar los usuarios (empleados, estudiantes y visitantes)
- Ayudar al coordinador determinar los cursos libres que se abrirán en cada temporada.
- Gestionar el proceso de matrícula e inscripción de curso
- Controlar los trámites para realizar el pago de aranceles y cuotas
- Gestionar el historial de notas de los estudiantes.
- Proporcionar datos e información fiables a los usuarios

Para identificar los requerimientos del sistema se deben identificar las necesidades de los usuarios, las cuales fueron extraídas de las entrevistas y encuestas:

<b>Usuarios</b>	<b>¿Qué necesita del sistema?</b>
<b>Coordinador de cursos libres</b>	<ul style="list-style-type: none"> <li>Registrar la información de los cursos libres ofertados</li> <li>Matricular estudiante</li> <li>Registrar los pagos de cuotas.</li> <li>Registrar pago de aranceles</li> <li>Registrar devoluciones de pago</li> <li>Autorizar devoluciones de pago (se realizan devoluciones si la solicita el estudiante y el grupo está cerrado)</li> <li>Registrar docente</li> <li>Asignar docente a un curso libre</li> <li>Realizar propuesta de cursos libres a impartir por temporada (a partir de datos históricos y preinscripciones) para hacer publicidad</li> <li>Aprobar propuesta de cursos</li> <li>Abrir un grupo de curso</li> <li>Cerrar un grupo de curso</li> <li>Elaborar informes de ingresos</li> <li>Aprobar acta de notas</li> </ul>
<b>Secretaria del coordinador de los cursos libres</b>	<ul style="list-style-type: none"> <li>Preinscribir estudiante en un curso</li> <li>Matricular estudiante en un curso.</li> <li>Registrar los pagos de cuotas</li> <li>Registrar pagos de aranceles</li> <li>Registrar devoluciones de pago</li> <li>Realizar devoluciones de pago</li> <li>Elaborar informes de ingresos</li> </ul>
<b>Decano</b>	Conocer los ingresos de la Unidad de Cursos Libres de forma detallada en cada temporada
<b>Delegado administrativo-financiero</b>	<ul style="list-style-type: none"> <li>Conocer los ingresos de la Unidad de Cursos Libres de forma detallada en cada temporada.</li> <li>Informe de pagos de los estudiantes</li> </ul>
<b>Docente</b>	<ul style="list-style-type: none"> <li>Registrar las notas de los estudiantes en el curso que ha impartido</li> <li>Proponer nuevos cursos</li> </ul>
<b>Estudiante</b>	<ul style="list-style-type: none"> <li>Conocer el catálogo de cursos que oferta la Unidad de Cursos Libres.</li> <li>Preinscribirse en un curso</li> <li>Conocer su historial académico</li> <li>Obtener constancias de notas</li> <li>Obtener información de cada uno de los cursos libres que son ofrecidos</li> </ul>
<b>Usuario no registrado</b>	<ul style="list-style-type: none"> <li>Conocer información de la empresa</li> <li>Conocer el catálogo de cursos</li> <li>Contactar a la organización</li> </ul>

Tabla 13: Necesidades de los usuarios que debe satisfacer el sistema a desarrollar

## 1.2. Requerimientos funcionales

Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. En algunos casos, los requerimientos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer.

<b>FRQ-0001</b>	<b>Registro de alumnos</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a>
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <a href="#">[FRQ-0003] Preinscripción al curso</a></li> <li>• <a href="#">[OBJ-0002] Gestionar perfil de los usuarios</a></li> </ul>
<b>Descripción</b>	El sistema deberá <i>guardar toda la información necesaria del estudiante para crear su perfil.</i>
<b>Importancia</b>	importante
<b>Urgencia</b>	inmediatamente
<b>Estado</b>	validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	Los datos del estudiante deberán de ser registrados para preinscribirse a un curso o durante la matrícula.

Tabla 14: Requerimiento funcional registro de alumnos (elaboración propia).

<b>FRQ-0002</b>	<b>Gestionar matrícula</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a> <a href="#">Rebeca Bonilla</a>
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <a href="#">[OBJ-0005] Gestionar el proceso de matrícula</a></li> </ul>
<b>Descripción</b>	El sistema deberá <i>proporcionar las interfaces y medios adecuados para realizar los subprocesos de registro del estudiante, de inscripción en el curso de su elección, de pago de cuotas y de salida de los reportes necesarios. Deberá llevar un control fiable histórico de los cursos en que se ha registrado un estudiante y permitir conocer la cantidad de estudiantes matriculados en cada curso y otra información similar.</i>
<b>Importancia</b>	vital
<b>Urgencia</b>	hay presión
<b>Estado</b>	validado
<b>Estabilidad</b>	media
<b>Comentarios</b>	El estudiante sólo podrá matricularse antes del período de inscripción

Tabla 15: Requerimiento funcional gestionar matrícula (elaboración propia).

<b>FRQ-0003</b>	<b>Preinscripción al curso</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Rebeca Bonilla</a>
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <a href="#">[OBJ-0001] Gestionar los cursos libres impartidos</a></li> <li>• <a href="#">[OBJ-0004] Ayudar al coordinador determinar los cursos libres que se abrirán en cada temporada.</a></li> </ul>
<b>Descripción</b>	El sistema deberá <i>permitir al estudiante reservar un cupo de los disponibles en el curso libre que seleccionó. Para ello deberá registrar sus datos personales previamente.</i>
<b>Importancia</b>	importante
<b>Urgencia</b>	puede esperar
<b>Estado</b>	validado
<b>Estabilidad</b>	media
<b>Comentarios</b>	Ninguno

Tabla 16: Requerimiento funcional preinscripción al curso (elaboración propia).

<b>FRQ-0004</b>	<b>Registro de docentes</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a>
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <a href="#">[OBJ-0003] Registrar empleados y personal</a></li> </ul>
<b>Descripción</b>	El sistema deberá <i>guardar toda la información necesaria del estudiante para crear su perfil.</i>
<b>Importancia</b>	importante
<b>Urgencia</b>	hay presión
<b>Estado</b>	validado
<b>Estabilidad</b>	media
<b>Comentarios</b>	Ninguno

Tabla 17: Requerimiento funcional registro de docentes (elaboración propia).

<b>FRQ-0005</b>	<b>Registro de notas</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a>
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <a href="#">[OBJ-0007] Gestionar el historial de notas</a></li> </ul>
<b>Descripción</b>	El sistema deberá <i>permitir que el docente registre en detalle las notas de sus estudiantes y que el estudiante vea el historial de las notas que ha obtenido en los cursos que ha llevado. El personal administrativo con los suficientes privilegios también podrá consultar este historial</i>
<b>Importancia</b>	vital
<b>Urgencia</b>	hay presión
<b>Estado</b>	validado
<b>Estabilidad</b>	media
<b>Comentarios</b>	Ninguno

Tabla 18: Requerimiento funcional registro de notas (elaboración propia).

<b>FRQ-0006</b>	<b>Registro de curso</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a> <a href="#">Ericka Espinoza</a>
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <a href="#">[OBJ-0001] Gestionar los cursos libres impartidos</a></li> <li>• <a href="#">[OBJ-0004] Ayudar al coordinador determinar los cursos libres que se abrirán en cada temporada.</a></li> </ul>
<b>Descripción</b>	El sistema deberá <i>almacenar la información de cada curso impartido y por impartir incluyendo el docente asignado.</i>
<b>Importancia</b>	vital
<b>Urgencia</b>	hay presión
<b>Estado</b>	validado
<b>Estabilidad</b>	media
<b>Comentarios</b>	Ninguno

Tabla 19: Requerimiento funcional registro de curso (elaboración propia).

<b>FRQ-0007</b>	<b>Administrar grupos de curso</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a>
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <a href="#">[OBJ-0001] Gestionar los cursos libres impartidos</a></li> <li>• <a href="#">[OBJ-0004] Ayudar al coordinador determinar los cursos libres que se abrirán en cada temporada.</a></li> </ul>
<b>Descripción</b>	El sistema deberá <i>controlar los grupos de curso, cantidad de alumnos inscritos, horarios, y los docentes asignados a cada grupo</i>
<b>Importancia</b>	vital
<b>Urgencia</b>	hay presión
<b>Estado</b>	validado
<b>Estabilidad</b>	media
<b>Comentarios</b>	Ninguno

Tabla 20: Requerimiento funcional administrar grupos de curso (elaboración propia).

<b>FRQ-0008</b>	<b>Gestión de pago</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Alejandro Vallecillo</a> <a href="#">Danilo Humberto Noguera Rivera</a> <a href="#">Rebeca Bonilla</a>
<b>Dependencias</b>	• <a href="#">[OBJ-0006] Controlar los trámites para realizar el pago de aranceles y cuotas</a>
<b>Descripción</b>	El sistema deberá <i>registrar los pagos hechos por el estudiante, tomando en cuenta los descuentos aplicables así como si aquel está becado. También deberá registrar la devolución de pagos.</i>
<b>Importancia</b>	vital
<b>Urgencia</b>	puede esperar
<b>Estado</b>	validado
<b>Estabilidad</b>	media
<b>Comentarios</b>	Ninguno

Tabla 21: Requerimiento funcional gestión de pago (elaboración propia).

<b>FRQ-0009</b>	<b>Gestionar personal administrativo</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a>
<b>Dependencias</b>	• <a href="#">[OBJ-0003] Registrar empleados y personal</a>
<b>Descripción</b>	El sistema deberá <i>guardar y modificar toda la información necesaria del personal administrativo para gestionar su perfil según sus privilegios.</i>
<b>Importancia</b>	importante
<b>Urgencia</b>	hay presión
<b>Estado</b>	validado
<b>Estabilidad</b>	media
<b>Comentarios</b>	Ninguno

Tabla 22: Requerimiento funcional gestionar personal administrativo (elaboración propia).

<b>FRQ-0010</b>	<b>Generar reportes</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a>
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>mostrar en pantalla así como imprimir informes de: matrícula, cursos, ingresos, pagos realizados y demás reportes que se utilizan actualmente</i>
<b>Importancia</b>	importante
<b>Urgencia</b>	puede esperar
<b>Estado</b>	validado
<b>Estabilidad</b>	media
<b>Comentarios</b>	Ninguno

Tabla 23: Requerimiento funcional generar reportes (elaboración propia).

<b>FRQ-0011</b>	<b>Gestión de usuario</b>
<b>Versión</b>	1.0 ( 19/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a>
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <a href="#">[FRQ-0003] Preinscripción al curso</a></li> <li>• <a href="#">[FRQ-0002] Gestionar matrícula</a></li> <li>• <a href="#">[FRQ-0004] Registro de docentes</a></li> <li>• <a href="#">[FRQ-0005] Registro de notas</a></li> <li>• <a href="#">[UC-0003] Registrar empleado</a></li> </ul>
<b>Descripción</b>	El sistema deberá <i>crear y modificar los usuarios en el sistema, controlando su acceso a las funcionalidades del mismo mediante los privilegios que les concede su rol en la organización.</i>
<b>Importancia</b>	vital
<b>Urgencia</b>	hay presión
<b>Estado</b>	validado
<b>Estabilidad</b>	alta
<b>Comentarios</b>	Ninguno

Tabla 24: Requerimiento funcional gestión de usuario (elaboración propia).

### 1.3. Requerimientos no funcionales

Los requerimientos no funcionales expresan, no se refiere a servicios específicos que debe de brindar el sistema; sino, a una serie de especificaciones de su forma de operar a fin de que sea útil a la organización. Sommerville (2005) expresa que:

Son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. Los requerimientos no funcionales a menudo se aplican al sistema en su totalidad. Normalmente apenas se aplican a características o servicios individuales del sistema.

A continuación se muestran los requerimientos obtenidos:

- A. Requerimientos del producto
  - a. Requerimiento de usabilidad

<b>NFR-0001</b>	<b>Requerimiento de usabilidad</b>
<b>Versión</b>	1.0 ( 27/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a> <a href="#">Michael Bryang Chávez Angulo</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>cumplir los requerimientos funcionales, trabajar correctamente con la Base de datos en SQL Server y ser compatible con las versiones actuales de los navegadores web</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	inmediatamente
<b>Estado</b>	validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 25: Requerimiento no funcional. Usabilidad (fuente; elaboración propia).

- b. Requerimientos de eficiencia:  
 i. Requerimiento de rendimiento

<b>NFR-0002</b>	<b>Requerimientos de Rendimiento</b>
<b>Versión</b>	1.0 ( 27/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>trabajar sin largos tiempos de espera para realizar las transacciones con la base de datos</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 26: Requerimiento no funcional. Rendimiento (fuente: elaboración propia).

- ii. Requerimiento de espacio

<b>NFR-0003</b>	<b>Requerimiento de Espacio</b>
<b>Versión</b>	1.0 ( 27/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>ocupar un espacio en disco duro no mayor de 2.5 GB incluida su BD</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 27: Requerimiento no funcional. Espacio (fuente: elaboración propia).

c. Requerimiento de fiabilidad

<b>NFR-0004</b>	<b>Requerimiento de Fiabilidad</b>
<b>Versión</b>	1.0 ( 27/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>llevar un control de los alumnos inscritos, de los grupos de cursos y del personal de la unidad de los cursos libres</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 28: Requerimiento no funcional. Fiabilidad (fuente: elaboración propia).

B. Requerimientos externos

a. Requerimientos legales

i. Requerimientos de privacidad

<b>NFR-0005</b>	<b>Requerimientos de Privacidad</b>
<b>Versión</b>	1.0 ( 27/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a> <a href="#">Michael Bryang Chávez Angulo</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a>
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>ser ejecutado únicamente por el personal de la Unidad de los Cursos Libres</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 29: Requerimiento no funcional. Privacidad (elaboración propia).

ii. Requerimientos de seguridad

<b>NFR-0006</b>	<b>Requerimientos de Confidencialidad</b>
<b>Versión</b>	1.0 ( 27/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a> <a href="#">Michael Bryang Chávez Angulo</a>
<b>Fuentes</b>	<a href="#">Danilo Humberto Noguera Rivera</a>
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>otorgar distintos niveles de acceso a los usuarios del sistema según su rol. No todos los usuarios tendrán permiso a todas las áreas del sistema</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 30: Requerimiento no funcional. Confidencialidad (fuente: elaboración propia).

<b>NFR-0013</b>	<b>Requerimiento de disponibilidad de la información</b>
<b>Versión</b>	1.0 ( 03/09/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>permitir que la información esté disponible para los usuarios autorizados en el momento que lo requieran</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 31: Requerimiento no funcional. Disponibilidad de la información (elaboración propia).

<b>NFR-0012</b>	<b>Requerimientos de integridad de la información</b>
<b>Versión</b>	1.0 ( 03/09/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>de mantener libre de modificaciones no autorizadas la información</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	pendiente de validación
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 32: Requerimiento no funcional. Integridad de la información (fuente: elaboración propia).

b. Requerimientos de software

<b>NFR-0007</b>	<b>Requerimientos de Software</b>
<b>Versión</b>	1.0 ( 27/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a> <a href="#">Michael Bryang Chávez Angulo</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>ejecutarse en un entorno windows server con asp.net 4.5+, SQL Server 2012+ y reporting services activado</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 33: Requerimiento no funcional. Software necesario (fuente: elaboración propia).

c. Requerimientos de HW virtual

<b>NFR-0008</b>	<b>Requerimientos de Hardware</b>
<b>Versión</b>	1.0 ( 27/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a> <a href="#">Michael Bryang Chávez Angulo</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>ejecutarse en un servidor o host web que posea al menos 3Gb de espacio en disco duro (con 2GB de BD) y 700 MB disponibles de RAM.</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 34: Requerimiento no funcional. Hardware (fuente: elaboración propia).

d. Requerimientos de red

<b>NFR-0009</b>	<b>Requerimiento de Arquitectura de Red</b>
<b>Versión</b>	1.0 ( 27/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a> <a href="#">Michael Bryang Chávez Angulo</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>ser de tipo cliente servidor</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	pendiente de validación
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 35: Requerimiento no funcional. Arquitectura de red (fuente: elaboración propia).

<b>NFR-0010</b>	<b>Requerimiento de transferencia de datos</b>
<b>Versión</b>	1.0 ( 03/09/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a> <a href="#">Michael Bryang Chávez Angulo</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>permitir una transferencia de datos mayor o igual a 4.54GB/mes.</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	hay presión
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 36: Requerimiento no funcional. Transferencia de datos (fuente: elaboración propia).

<b>NFR-0011</b>	<b>Requisito de conectividad</b>
<b>Versión</b>	1.0 ( 03/09/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>permitir acceso 24/7.</i>
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 37: Requerimiento no funcional. Conectividad (fuente: elaboración propia).

<b>NFR-0014</b>	<b>Requerimiento de concurrencia</b>
<b>Versión</b>	1.0 ( 03/09/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	El sistema deberá <i>permitir su utilización por al menos 30 usuarios a la vez.</i>
<b>Importancia</b>	Vital
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Alta
<b>Comentarios</b>	Ninguno

Tabla 38: Requerimiento no funcional. Concurrencia (fuente: elaboración propia).

#### 1.4. Limitaciones y restricciones del sistema

A continuación se describe funcionalidades que no realizará el sistema:

<b>CRQ-0001</b>	<b>No contabiliza las horas de clase impartidas</b>
<b>Versión</b>	1.0 ( 28/08/2016 )
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a>
<b>Fuentes</b>	?
<b>Dependencias</b>	Ninguno
<b>Descripción</b>	La información almacenada por el sistema deberá satisfacer la siguiente restricción: <i>El sistema no contabiliza la asistencia de los docentes ni las horas de clase que han impartido</i>
<b>Importancia</b>	vital
<b>Urgencia</b>	inmediatamente
<b>Estado</b>	validado
<b>Estabilidad</b>	alta
<b>Comentarios</b>	Ninguno

Tabla 39: Restricción del sistema: Horas de clase impartidas no son contabilizadas (fuente: elaboración propia).

### 1.5. Matriz de rastreabilidad

A continuación se muestra cómo los requerimientos planteados contribuyen a los objetivos definidos para el sistema:

TRM-0001	<a href="#">OBJ-0001</a>	<a href="#">OBJ-0002</a>	<a href="#">OBJ-0003</a>	<a href="#">OBJ-0004</a>	<a href="#">OBJ-0005</a>	<a href="#">OBJ-0006</a>	<a href="#">OBJ-0007</a>
<a href="#">FRQ-0001</a>	-	↗	-	-	-	-	-
<a href="#">FRQ-0002</a>	-	-	-	↗	-	-	-
<a href="#">FRQ-0003</a>	↗	-	↗	-	-	-	-
<a href="#">FRQ-0004</a>	-	↗	-	-	-	-	-
<a href="#">FRQ-0005</a>	-	-	-	-	-	↗	-
<a href="#">FRQ-0006</a>	↗	-	↗	-	-	-	-
<a href="#">FRQ-0007</a>	↗	-	↗	-	-	-	-
<a href="#">FRQ-0008</a>	-	-	-	-	↗	-	-
<a href="#">FRQ-0009</a>	-	↗	-	-	-	-	-
<a href="#">FRQ-0010</a>	-	-	↗	-	-	-	↗
<a href="#">FRQ-0011</a>	-	↗	-	-	-	-	-

Tabla 40: Matriz de rastreabilidad (fuente: elaboración propia).

## Capítulo III: Análisis de la viabilidad

Para todo sistema nuevo, el proceso de ingeniería de requerimientos, necesita que se elabore un estudio de viabilidad.

Llevar a cabo un estudio de la viabilidad comprende la evaluación y recopilación de la información, y la redacción de entregables. En el informe se pueden proponer cambios en el alcance, el presupuesto y la confección de agendas del sistema y sugerir requerimientos adicionales de alto nivel para este.

Se realizaron 5 estudios para este proyecto: la viabilidad técnica, que comprende todos los equipos de cómputo, ancho de banda de internet, proveedores de alojamiento etc.; la viabilidad económica, que comprende los costos de desarrollo del proyecto; la viabilidad financiera, que determina la rentabilidad; la viabilidad organizacional-operativa, que permitirá definir las condiciones mínimas para la implementación del software así como sus beneficios organizacionales; y por último, la viabilidad legal del proyecto para que este se desarrolle dentro del marco que la legislación nacional establece.

### 2.1. Estudio técnico.

El estudio técnico de un proyecto tiene el objetivo de determinar el equipo necesario para utilizar con eficiencia y eficacia los recursos que posee la entidad y de esta manera llegar a cumplir los objetivos que tiene la misma. Este estudio conlleva realizar una serie de análisis de los diversos factores que se ven involucrados para determinar costos, infraestructura, inversión en mano de obra, y otros aspectos que se verán a continuación.

Se pretende resolver todo lo que se relacione a la instalación y el funcionamiento del proyecto, en este caso, el del software en la Facultad de Ciencias y Sistemas, abarcando todo el proceso, que va desde la descripción del proceso, la obtención del equipo y todo lo involucrados en el proyecto.

### 2.1.1. Descripción de los requisitos mínimos

El sistema se desarrollará bajo la arquitectura cliente servidor en la plataforma .NET Framework bajo el patrón MVC 4; por tanto, existen una serie de factores a tomar en cuenta: las especificaciones del servidor, de los equipos clientes y el ancho de banda, así como los costos de hardware, instalación e infraestructura.

Requisitos mínimos para los equipos clientes	
Equipos de escritorio	Dispositivos móviles
Windows 7 SP1 o superior	500 Mb de memoria RAM
Memoria RAM de 2GB para 32 Bit y 4GB para 64 bits	Paquete de datos o conexión a internet
Mozilla Firefox 45.0.1+, Chrome 52+, Microsoft Edge	Navegador compatible con HTML5, CSS3 y ECMASCRIPT 5
Conexión a internet de 512Kb/s o superior.	

Tabla 41: Requisitos mínimos para que los dispositivos clientes puedan acceder al sistema

Requisitos mínimos para servidor (físico)
Windows Server 2012 R2 o superior
SQL Server 2012 Standard o superior
Procesador de 32 bits (compatible con Pentium III o superior) o 64 Bits (AMD Opteron, AMD Athlon 64, Intel Xeon compatible con Intel EM64T o Intel Pentium IV compatible con EM64T) de 2GHz o superior
Memoria RAM de 4GB
Espacio en disco Duro de 6GB
Microsoft .Net Framework 3.5, 4.0 y 4.5.

Tabla 42: Requisitos para el sistema web en el servidor

A continuación se comparan los recursos existentes contra los requerimientos anteriores:

Los equipos cliente a los que tiene acceso el personal administrativo presentan las siguientes características:

- Computadora del Coordinador de los cursos libres:
  - Procesador AMD A8 PRO-7600B R7, 10 Compute Core 4C(3M Caché 3.10 GHZ) 2 núcleos con 4 procesadores lógicos
  - Memoria RAM DDR3 de 8GB.
  - Sistema Operativo Windows 8.1 Pro de 64bits.
  - Disco Duro de 750GB de almacenamiento
  - Chrome 65

- Computadora del Decano, secretaría académica, asistente financiero, y secretaria de los cursos libres:
  - Procesador Intel Dual Core E7200 2.53 GHz (2m de cache) 1 núcleo y 2 procesadores lógicos.
  - Memoria RAM de 4gb.
  - Sistema Operativo Windows 7 de 64 bits.
  - Disco Duro de 750GB de almacenamiento.
  - Chrome 65

Se puede observar que las especificaciones de los equipos clientes existentes en la FCYS sobrepasan los requisitos mínimos de los mismos para la utilización del sistema.

En el aspecto del servidor, la Facultad no posee uno propio, y tampoco existen los recursos a nivel universitario (véase anexo 4) así que se presentan 2 alternativas: adquirir uno propio, invirtiendo en cableado e infraestructura complementaria, o contratar un servicio de alojamiento compatible.

### 2.1.2. Requisitos para el alojamiento del sistema

Considerando una cantidad máxima de 25 estudiantes por grupo de curso, 8 abiertos en promedio, por temporada, y un docente por cada curso, al Decano, al Asistente administrativo-financiero, la Secretaria de la FCYS, al Coordinador de los Cursos Libres y la secretaria del coordinador, la cantidad máxima de estaciones cliente que se conectarán al sistema será de 213 equipos.

La tabla 43 detalla los accesos al sistema para determinar la transferencia de datos mensual.

Descripción	Cantidad
Días	31
Entradas al sistema por días	20
Páginas accedidas en promedio por entrada	5
Peso promedio de las páginas accedidas	1536KB

Tabla 43: Descripción de accesos al sistema (fuente: elaboración propia).

Resultado:  $31 * 20 * 5 * 1536 = 4,761,600$  KB/mes  $\approx 4.54$  GB/mes

Esto indica que el alojamiento del sistema debe permitir una transferencia de datos mayor o igual a 4.54GB/mes.

Para que se ejecute apropiadamente el sistema informático, se requiere que el servidor tenga un espacio para aplicación de 1GB, SQL Server 2012+ con soporte para BD de un tamaño de al menos 2GB, Internet Information Services IIS 7+, Asp.net 4.5+, MVC4+ y Reporting Services.

En este sentido, se plantean 2 escenarios:

- **Escenario I: Equipos propios**

El otro escenario es que los cursos libres compren el equipo (Servidor) que alojaría el sitio web, así como el sistema gestor de base de datos que se requiere para la administración de registros. Deberá contratar al personal necesario para la instalación y administración de dicho equipo. El servidor que se ocupe del alojamiento debe poseer o superar las siguientes características de hardware:

- 4 gigabyte de RAM
- 120 Gigabyte de disco duro.
- Procesador de 64 bits a 2GHz.

A este equipo se le deberá de instalar Windows Server 2012 de 64 bits o superior, Internet Information Services 7 u 8, y SQL Server 2012 o superior para que el sistema funcione de manera correcta. Adicionalmente se deberá de invertir en una unidad UPS, y el cableado necesario para su interconexión (véase la tabla 44).

Equipo	Personal a ocupar	Costo por año (\$)	Total en 5 años(\$)
Contrato del dominio		50	250
Personal para administración de equipo	1	2744.41 <sup>7,8</sup>	13,722.09
Servidor Dell PowerEdge T30(Procesador Intel Xeon E3-1225 v5 3.3GHz Chipset Intel C236 Memoria RAM 8GB DDR4 (1x8G) 2133)			721.73
Monitor Dell E2016H			108
Windows Server 2016 Standard			1,456.27
Licencia SQL Server 2016 Standard			2,834.78
UPS FORZA FX-1500		189.75	569.25
Cableado categoría 5e			82.35
<b>Total Servidor Propio</b>			<b>\$ 19744.47</b>

Tabla 44: Costos al invertir en equipos y accesorios propios (fuente: elaboración propia a partir de cotización en Sevasa).

A continuación, se muestra el diagrama de red lógico (figura 2) y físico (figura 3) de esta alternativa:

<sup>7</sup> Baca Castellón, L. (17 de Agosto de 2017). Estos son los nuevos salarios mínimos que recibirán los trabajadores en Nicaragua. *La Prensa*. Recuperado de <http://www.laprensa.com.ni/2017/08/17/economia/2281209-a-las-puertas-segundo-ajuste-al-salario-minimo> el 18 de Octubre de 2017

<sup>8</sup> USD 1 equivale a C\$ 30.2662 al 20 de Noviembre de 2017 (fuente: BCN)

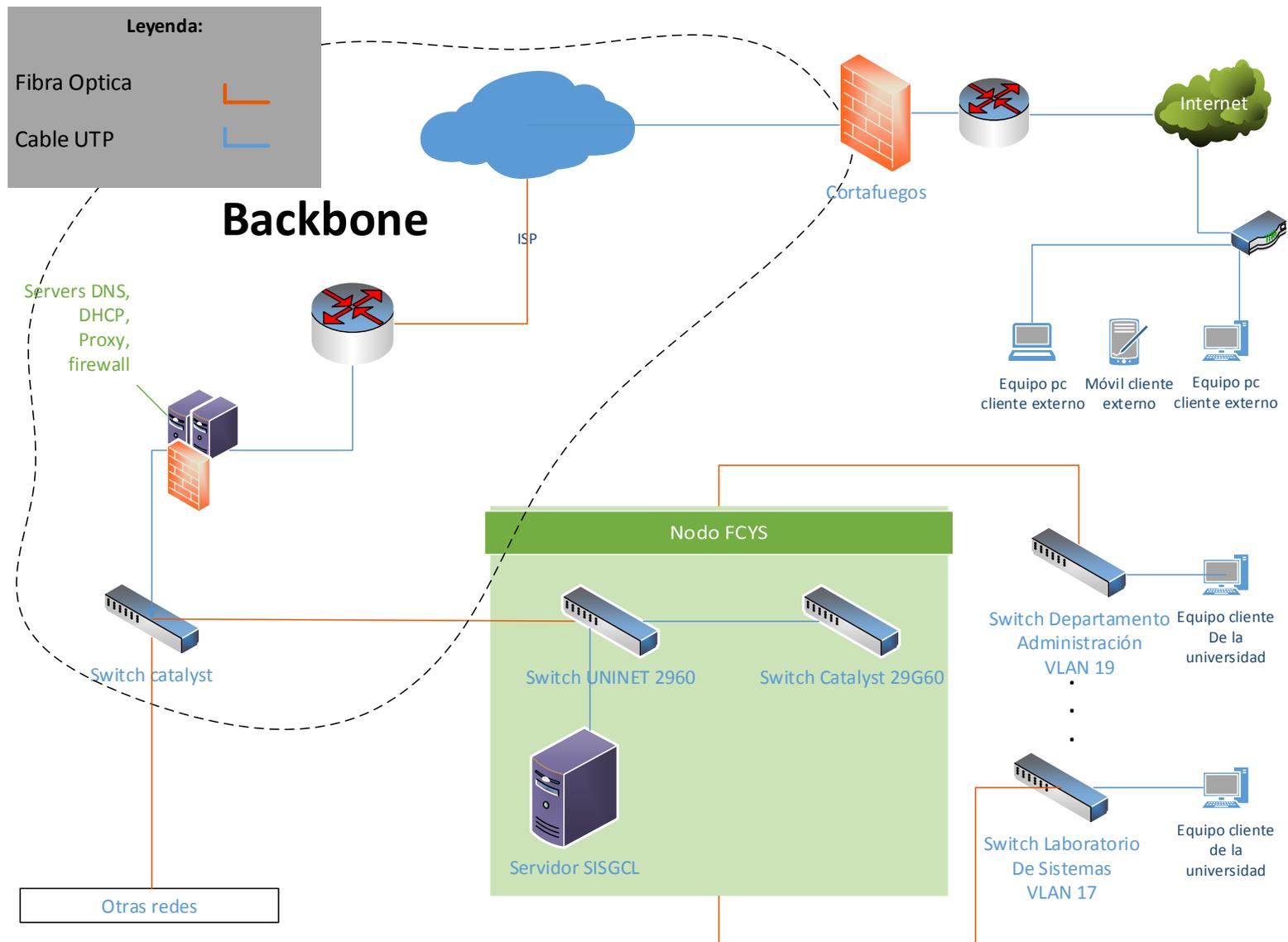


Figura 2: Diagrama de red lógico para el escenario de hosting propio (fuente: elaboración propia).

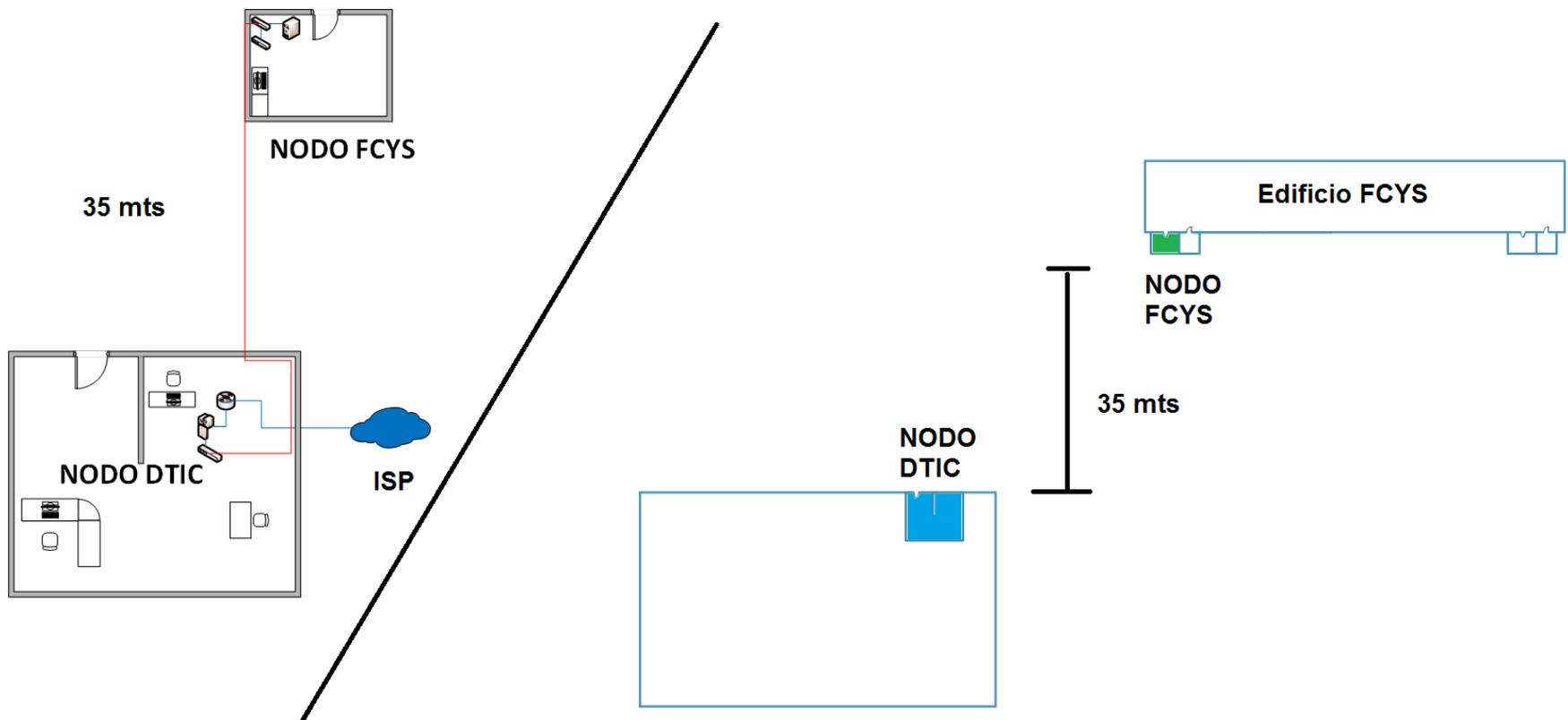


Figura 3: Diagrama de red físico para el escenario de hosting propio (fuente: elaboración propia).

- **Escenario II: Hosting externo.**

El sitio web se hospeda en un servidor externo haciéndolo accesible mediante internet, contratando un proveedor de servicios de alojamiento. A esto se le conoce como *alojamiento web*. Si se optase por este escenario no habría preocupación por el costo del servidor, costos de instalación, ni por la seguridad, ya que estas empresas cubren todos estos detalles; es importante realizar los pagos en tiempo y forma. Las alternativas de Web Host que cumplen con los requisitos necesarios para el sistema que se desarrolla son:

- MyAsp.net: Con sede en California, ofrece espacio de almacenamiento ilimitado y transferencia de datos ilimitada, FTP, SMTP, panel de control web, SQL Server 2014 con un límite de 10GB, soporte de ASP.net 3.5+, soporte de ASP.net MVC 4+. El costo por mes es de \$7.95, el pago por el registro del dominio es de \$10/año y el certificado SSL cuesta \$29/año.
- Windows Azure: ofrecido por Microsoft, es fácilmente escalable incurriendo en pequeñas cuotas extra. Soporta las últimas versiones de SQL Server, ASP.net 4+. La utilización de una BD es una cuota extra de \$4.98/mes permitiendo soportar SQL Server 2012+. Incluye SSL. Permite configurar la potencia de las máquinas virtuales a disposición del cliente. Su plan Básico ofrece una VM de 1.75 GB y 10 GB de espacio en disco. La cuota asciende a \$61/mes.

Considerando las prestaciones que ofrecen ambas alternativas, la mejor es MyAsp.net por ser la que representa una inversión menor. Los costos a 5 años se muestran en la tabla 45 y el diagrama de red en la figura 4.

Gastos operativos	Costo por año (\$)	Total a 5 años (\$)	Observaciones
Contrato del dominio	50	250	Se paga anualmente a Nic.Ni
MyAsp.net	95.4 (sin descuento)	414.99	
<b>Total Web Hosting</b>		<b>664.99</b>	

Tabla 45: Costo de hosting propio (fuente: elaboración propia).

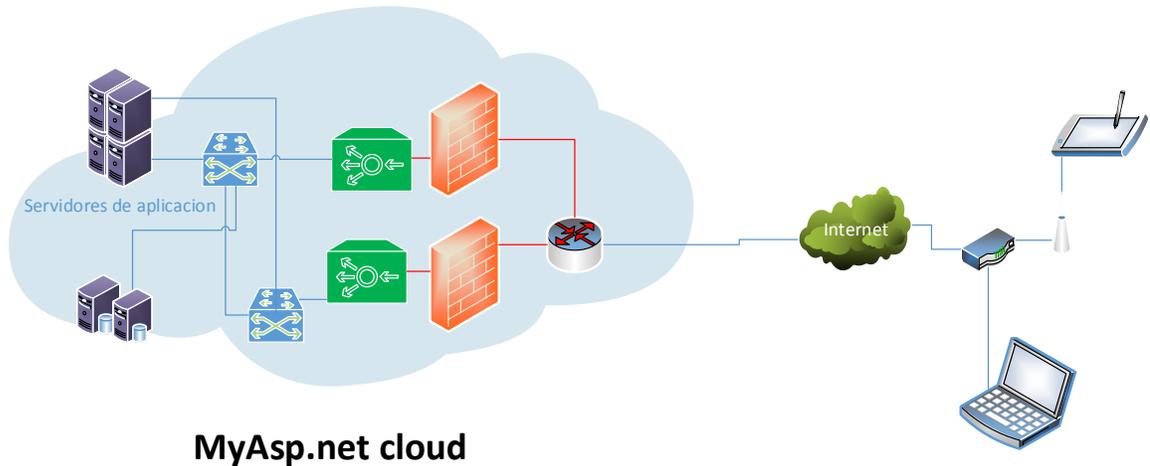


Figura 4: Diagrama de red para el escenario de alojamiento externo (fuente: elaboración propia)

Al comparar los dos escenarios se observa que la opción de web hosting que ofrece MyAsp.net representa una inversión menor en el horizonte del proyecto de 5 años; por tanto, es la opción a elegir para el desarrollo del proyecto y el funcionamiento del sistema web.

Del análisis técnico se puede concluir:

- La FCYS cumple los requisitos para el acceso y utilización del sistema web en sus equipos cliente.
- Al no tener un servidor propio se planteó contratar un servicio de alojamiento externo debido a que su costo a 5 años es mucho menor que si se adquiere un servidor físico.
- Se eligió el servicio de hosting de MyASP.net porque satisface los requisitos de banda ancha, capacidad de la BD y alojamiento del servidor, siendo compatible con ASP.net MVC 4.

## 2.2. Análisis económico

Para estimar el costo del sistema a desarrollar, el estudio de evaluación económica desarrolla COCOMO II (Modelo de Costos Constructivo), a través del cual se calculó el esfuerzo, el tiempo y los recursos necesarios para la ejecución del proyecto.

Este método utiliza como base el análisis de los puntos de función que proporciona una medida objetiva del grado de funcionalidad que el sistema tendrá, siendo útil en la evaluación, planificación, gestión y control de los procesos de desarrollo del software: permite cuantificar el tamaño de las aplicaciones a partir de los requisitos que especifica el usuario final; de los puntos de función derivan una serie de métricas para la gestión de la productividad, calidad y el coste en las diferentes etapas de su ciclo de vida a lo largo del tiempo.

Se debe de identificar las funcionalidades proveídas por los datos (Archivos de interfaces externas y archivos lógicos internos) y las transacciones (entradas, salidas y consultas):

- Archivos Lógicos internos<sup>9</sup>: Un archivo lógico interno es un grupo de datos lógicamente relacionados o informaciones de control, identificado por el usuario y mantenido dentro de las fronteras de la aplicación.
- Archivos de interfaces externas<sup>9</sup>: según la visión del usuario, son un grupo de datos lógicamente relacionados o informaciones de control utilizadas por la aplicación pero que es mantenida por otra (por eso es de interface externa).
- Entrada externa<sup>9,10</sup>: es un proceso elemental que representa las actividades de mantenimiento de los datos de los archivos lógicos referenciados (CRUD) y procesamiento los datos o información de control

---

<sup>9</sup> Giménez, J. (2002). *Funciones Tipo Dato*. Recuperado el 16 de Agosto de 2016 desde <http://www.oocities.org/gimenezpy/datos.htm>

<sup>10</sup> Giménez, J. (2002). *Funciones Tipo Trans*. Recuperado el 16 de Agosto de 2016 desde <http://www.oocities.org/gimenezpy/transaccion.htm>

que proceden del exterior de la aplicación. Dicho de otra forma, representan el flujo de información desde el exterior hacia el sistema.

- Salida externa<sup>11</sup>: es un proceso elemental que genera datos derivados o información desde el interior del sistema a través del procesamiento lógico de los datos recuperados desde archivos lógicos referenciados.
- Consulta Externa<sup>9</sup>: es un proceso elemental compuesto por una combinación de entrada y salida que resulta en una recuperación de información. La parte correspondiente a la salida en una consulta externa no posee datos derivados y ningún archivo lógico interno es mantenido durante el proceso.

Los puntos de función no ajustados, que indican la complejidad de la funcionalidad del sistema se indican en la tabla siguiente:

Tipo de función	Complejidad	Cantidad		Puntaje por complejidad	Total por tipo
Archivos Lógicos Internos	Simple	9	x	7	63
	Promedios	0	x	10	0
	Complejos	0	x	15	0
Archivos de Interfaz Externa	Simple	0	x	5	0
	Promedios	0	x	7	0
	Complejos	0	x	10	0
Entradas Externas	Simple	11	x	3	33
	Medias	7	x	4	28
	Complejas	3	x	6	18
Salidas Externas	Simple	1	x	4	4
	Medias	5	x	5	25
	Complejas	2	x	7	14
Consultas Externas	Simple	8	x	3	24
	Medias	4	x	4	16
	Complejas	0	x	6	0
<b>Total de Puntos de Función sin Ajustar</b>					<b>225</b>

Tabla 46: Puntos de función sin ajustar (fuente: elaboración propia).

<sup>11</sup> Kushal, S. (2007). *Calculating Function Points*. Recuperado el 16 de Agosto de 2016 desde <http://www.codeproject.com/Articles/18024/Calculating-Function-Points>

La sumatoria de los puntos de función bruta es de 225, la cual debe de ser ajustada calificando una serie de ítems para determinar un factor de ajuste (FA). Véase la siguiente tabla:

Características Generales	Grado de Influencia
1. Copias de seguridad y de recuperación fiables	4
2. Comunicación de datos	3
3. Funciones de procesamiento distribuido	3
4. Rendimiento crítico	1
5. Entorno operativo existente y fuertemente utilizado	1
6. Entrada de datos interactiva	4
7. Transacciones sobre múltiples pantallas	2
8. Actualización interactiva de archivos maestros	4
9. Entradas, salidas, archivos o peticiones complejas	3
10. Procesamiento interno complejo	1
11. Código reutilizable	3
12. Conversión e instalación	1
13. Múltiples instalaciones en diferentes organizaciones	0
14. Facilitar cambios y ser fácilmente reutilizadas	3
<b>Nivel de influencia</b>	<b>33</b>

Tabla 47: Cálculo del nivel de influencia de los PF (fuente: elaboración propia).

El factor de ajuste (FA) se determina empleando la siguiente ecuación:

$$FA = (0.65 + (0.01 * FC)) = (0.65 + (0.01 * 33)) = 0.98$$

Donde FC es el factor de complejidad o nivel de influencia

Con esto se obtiene un factor de ajuste de 0.98.

Luego los puntos de función ajustados (PFA) se determinan empleando la siguiente fórmula:

$$PFA = PFSA * FA = 225 * 0.98 = 220.5 \approx 221$$

Donde:

- PFSA es el total de los puntos de función sin ajustar

El lenguaje principal que los desarrolladores utilizarán en la aplicación es C#, el cual tiene un índice promedio de líneas de código por punto de función de 54 LOC/PF<sup>12</sup>. Este dato permite calcular el total de líneas de código:

$$LOC = PFA * LP$$

Donde:

- *LOC*: Total de líneas de código a emplear.
- *LP*: Líneas de código por punto de función

$$LOC = 221 * 54 = 11,934$$

A partir del cálculo anterior, se estima emplear 11,934 líneas de código para desarrollar la aplicación.

Para determinar la inversión es necesario utilizar técnicas para calcular el esfuerzo de las personas que planean, diseñan, desarrollan e implementan el sistema. Se empleará el modelo COCOMO II que utiliza como entrada la cantidad de líneas de código fuente de la aplicación (SLOC) que se calculó previamente.

COCOMO II tiene 3 niveles para diferente uso:

- Composición de aplicación
- Diseño temprano
- Nivel de post arquitectura

---

<sup>12</sup> Quantitative Software Management. (2016). *Function Point Languages Table*. Recuperado de <http://www.qsm.com/resources/function-point-languages-table>, el 16 de agosto de 2016.

Se utilizará el nivel de post arquitectura para calcular el esfuerzo:

$$E = A * Tama\tilde{n}o^B * \prod EM_i$$

Donde:

- E: Esfuerzo, estimado en personas-mes.
- A: coeficiente de calibración.
- EM<sub>i</sub>: Cada uno de los multiplicadores de esfuerzo.

Sommerville (2007) expresa que “*basándose en su gran cantidad de datos históricos, Boehm propone que el coeficiente A sea de 2,94. El tamaño del sistema es expresado en KSLOC*” (miles de líneas de código), el exponente B se calcula mediante el nivel de post arquitectura;  $\prod EM_i$ , la productoria de los multiplicadores de esfuerzo, se basa en 17 características del proyecto..

El exponente B, que representa el ahorro o gasto relativo de escala, es calculado a partir de los 5 factores de escala: Precedentes (PREC), flexibilidad del desarrollo (FLEX), resolución de arquitectura/riesgo (RESL), el grado de cohesión del equipo de trabajo (TEAM) y la madurez del proceso (PMAT).

Para el proyecto actual la cantidad de KSLOC es de 11.934 MF.

La ecuación que permite determinar a B es:

$$B = 0.91 + 0.01 * \sum_{i=1}^5 SF_i$$

Donde:

$SF_i$  es cada uno de los factores de escala mencionados anteriormente.

Los valores para el presente proyecto están descritos en la tabla siguiente:

PREC	FLEX	RESL	TEAM	PMAT
Nominal	Nominal	Bajo	Nominal	Bajo
3.72	3.04	5.65	2.19	6.24

Tabla 48: Factores de escala (fuente: elaboración propia).

A partir de la tabla anterior se tiene:

$$B = 0.91 + 0.01 * 20.84$$

$$B = 0.91 + 0.2084$$

$$B = 1.1184$$

Se tiene un valor de gasto de escala de 1.1184

### **Estimación del esfuerzo:**

Para calcular el factor de ajuste de esfuerzo compuesto, se emplean 17 drivers o indicadores que se multiplican entre sí: confiabilidad requerida del software (RELY), tamaño de la base de datos (DATA), complejidad del producto (CPLX), desarrollo del software para reusabilidad (RUSE), documentación ajustada al ciclo de vida del software (DOCU), restricciones del tiempo de ejecución (TIME), restricciones de almacenamiento principal (STOR), volatilidad de la plataforma (PVOL), capacidad de los analistas (ACAP), capacidad de los programadores (PCAP), continuidad del personal (PCON), experiencia en las aplicaciones (APEX), experiencia en la plataforma (PLEX), experiencia en el lenguaje y herramientas (LTEX), uso de herramientas de software (TOOL), desarrollo multisitio (SITE), y calendario de desarrollo requerido (SCHED). Véase la tabla 50:

La productoria de estos multiplicadores es:

$$\prod EM_i = 0.37476649$$

Ahora se puede estimar el esfuerzo:

$$E = A * Tamaño^B * \prod EM_i$$

$$E = 2.94 * 11.934^{1.1184} * 0.37476649$$

$$E = 17.6354109$$

El esfuerzo es de 17.635 personas-meses.

INDICADOR	NIVEL	VALOR
<b>INDICADORES DEL PRODUCTO</b>		
RELY	Bajo	0.92
DATA	Alto	1.14
CPLX	Nominal	1
RUSE	Bajo	0.91
DOCU	Nominal	1
<b>INDICADORES DE LA PLATAFORMA</b>		
TIME	Nominal	1
STOR	Nominal	1
PVOL	Bajo	0.87
<b>INDICADORES DEL PERSONAL</b>		
ACAP	Nominal	1
PCAP	Alto	0.87
PCON	Alto	0.92
AEXP	Alto	0.89
PEXP	Alto	0.88
LTEX	Alto	0.91
<b>INDICADORES DEL PROYECTO</b>		
TOOL	Alto	0.86
SITE	Alto	0.92
SCED	Nominal	1

Tabla 49: Valores de los drivers de costo en el modelo de post arquitectura (fuente: elaboración propia).

### Estimación del tiempo de desarrollo (TDES):

$$TDES = 3.67 * E^{0.28+(0.002*\sum SF_i)}$$

$$TDES = 3.67 * 17.6354109^{0.28+(0.002*20.84)}$$

$$TDES = 3.67 * 17.6354109^{0.32168}$$

$$TDES = 9.23854075 \text{ meses}$$

El tiempo estimado para el desarrollo del sistema es aproximadamente 9 meses y 7 días.

El número de personal requerido es

$$CH = E/TDES$$

Donde:

- CH es la cantidad de hombres (personal requerido).
- E: Esfuerzo en personas-meses.
- TDES: Tiempo de desarrollo.

Sustituyendo los valores en la ecuación anterior, se tiene:

$$CH = \frac{17.6354109}{9.23854075} = 1.90889572 \text{ personas}$$

A partir del dato anterior, se necesitan 2 personas para el proyecto.

### Estimación de la productividad:

La productividad de estos analistas-desarrolladores se estima a como sigue:

$$P = \frac{KLOC * 1000}{TDES}$$

$$P = \frac{11.934 * 1000}{9.23854075} = 1291.76245$$

Se estima que estos desarrolladores tendrán una productividad de 1291.76 líneas de código/hombre-máquina.

### Distribución de tiempo y esfuerzo por etapa:

A partir de la información anterior y mediante interpolación de la tabla 1 en el Anexo 8.1 se obtuvo los siguientes valores:

Etapa	%Esfuerzo	Esfuerzo	%TDes	TDes	CH(E/TDes)
Estudio Preliminar	7	1.26	18.327833	1.6932244	1
Análisis	17	3.06	25.163917	2.3247787	1
Diseño y desarrollo	60.50825	10.891485	51.344333	4.7434672	2
Pruebas e implementación	22.49175	4.048515	23.49175	2.1702949	2

Tabla 50: Distribución de esfuerzo y tiempo por etapa (fuente: elaboración propia).

Estos valores permiten asignar a cada etapa del desarrollo del proyecto los recursos necesarios (esfuerzo, tiempo de desarrollo y cantidad de hombres).

### **Cálculo de los costos del proyecto:**

Para obtener el costo total en el cual incurrirá la Unidad de Cursos Libres, se debe realizar una serie de cálculos relacionados a los costos directos e indirectos del desarrollo del sistema web.

El valor del proyecto se calcula mediante la siguiente ecuación:

$$CTP = CD + CI$$

Donde:

- CTP: Costo total
- CD: Costos directos
- CI: Costos indirectos

A su vez, los costos directos están constituidos por: la fuerza de trabajo (CFT), la utilización de medios técnicos (CUMT), y el abastecimiento técnico de materiales (ATM):

$$CD = CFT + \sum CUMT + ATM$$

Los costos indirectos representan el 15% de los costos directos.

Para estimar el costo de la fuerza de trabajo, se considera un sueldo de C\$6,921.93<sup>13</sup> para cada uno de los desarrolladores, siendo calculada por etapa a partir de los valores del tiempo de desarrollo (TDes, véase la tabla 50) y la siguiente ecuación:

$$CFT = \text{Salario} * \text{Cantidad de Analistasprogramadores} * TDes$$

---

<sup>13</sup> Baca Castellón, L. (17 de Agosto de 2017). Estos son los nuevos salarios mínimos que recibirán los trabajadores en Nicaragua. *La Prensa*. Recuperado de <http://www.laprensa.com.ni/2017/08/17/economia/2281209-a-las-puertas-segundo-ajuste-al-salario-minimo-el-18-de-October-de-2017>

Etapa	CFT
<b>Estudio Preliminar</b>	C\$ 11,720.33
<b>Análisis</b>	C\$ 16,091.89
<b>Diseño y desarrollo</b>	C\$ 65,667.61
<b>Pruebas e implementación</b>	C\$ 30,045.13
<b>ΣCFT</b>	C\$ 123,524.95

Tabla 51: Costo de la fuerza de trabajo (fuente: elaboración propia).

El costo de uso de medios técnicos hace referencia al consumo energético de los equipos de cómputo empleados para el desarrollo del sistema, el cual se calculó en 0.0965 KW por hora (véase anexo 8.2, tabla 2) y considerando un precio promedio de C\$ 5.7575 KW/h<sup>14</sup>.

Distribuyéndolo por etapa, el CUMT sería:

Etapa	CUMT
<b>Estudio Preliminar</b>	C\$ 112.01
<b>Análisis</b>	C\$ 272.02
<b>Diseño y desarrollo</b>	C\$ 1,936.41
<b>Pruebas e implementación</b>	C\$ 719.79
<b>ΣCUMT</b>	C\$ 3,040.24

Tabla 52: Costo de uso de medios técnicos.

Añadiendo un costo de abastecimiento de materiales (ATM) de C\$ 772.83 (Véase anexo 8.3, tabla 3), el costo indirecto equivale a C\$19,100.703.

El costo total del proyecto es:

$$CTP = CD + CI$$

$$CTP = C\$127,338.02 + C\$19,100.73$$

$$CTP = C\$146,438.72$$

<sup>14</sup> Banco Central de Nicaragua (2017). *Precios de la Energía*. Recuperado de <http://www.bcn.gob.ni/estadisticas/precios/Energia/2-15.htm> el 11 de noviembre de 2016

### **Relación costo-beneficio:**

A partir de los resultados obtenidos de los cálculos estimados a través de la metodología de COCOMO II, el costo de inversión del proyecto de C\$146,438.72 representa una cantidad que no se recuperara en un periodo de tiempo; sin embargo, la implementación de la aplicación web generará beneficios intangibles para los usuarios, los cuales son de gran relevancia para la comunidad universitaria por lo que es conveniente continuar con el desarrollo del proyecto.

La implementación del sistema genera beneficios intangibles para que la organización vuelva más eficientes sus procesos internos, como el registro de estudiantes, registro de notas académicas y acercarse con el público meta, agregando el hecho de su modernización y avance en pro de su visión.

### **Conclusiones del análisis económico**

A lo largo de este estudio se encontró que:

- El software desarrollado utilizará un estimado de 11,934 líneas de código.
- Debido a que el exponente B del esfuerzo compuesto es de 1.1184, mayor a 1, el proyecto exhibe una *deseconomía* de escala: La productividad del proceso de desarrollo de sistema disminuirá si aumenta el tamaño del mismo.
- El esfuerzo estimado fue de 17.635 personas-meses.
- El tiempo estimado para el desarrollo del sistema es de 9.23854 meses (9 meses y 7 días).
- Se necesitan 2 personas para el proyecto.
- Estos desarrolladores tendrán una productividad de 1291.76 líneas de código/hombre-máquina.
- El costo total del proyecto es de C\$146,438.72. Aunque no se recupera en un periodo tiene muchos beneficios intangibles de gran relevancia por lo que es correcto analizarlo financieramente a mediano plazo.

### 2.3. Análisis financiero

Se realiza un estudio financiero para determinar si el proyecto será rentable a través del tiempo de evaluación, que en este caso, se realizó a 5 años. Para realizar este estudio se hicieron estimaciones de ingresos y gastos, con una tasa de crecimiento dada por la inflación en el país. A continuación se presentan los siguientes detalles necesarios para realizar el estudio:

- **Pronóstico de matrículas (temporadas altas y bajas).**

Años	temporada 1	temporada 2	temporada 3	total de estudiantes por año
2010	43	17	0	<b>60</b>
2011	40	21	0	<b>61</b>
2012	31	43	14	<b>88</b>
2013	42	29	47	<b>118</b>
2014	21	34	23	<b>78</b>
2015	0	34	84	<b>118</b>
2016	49	52	84	<b>185</b>
2017	81	53	0	<b>134</b>

**Tabla 53: Datos históricos de la cantidad de matrículas, periodo 2010-2017.**  
Elaboración propia (fuente: unidad de cursos libres).

Con la información de la **tabla 53** se pronosticará los ingresos futuros de la Unidad de Cursos Libres durante la duración del proyecto (5 años). Para ello se usará una de las funciones de Excel, llamada pronóstico que permitirá conocer un estimado de los estudiantes que tomarán un curso. Sólo se debe multiplicar el precio de cada curso (C\$ 3000) para conocer los ingresos de los siguientes años. La tabla 54 detalla muestra los resultados:

Años	Pronóstico de estudiantes por temporada
2018	169
2019	183
2020	197
2021	211
2022	225

Tabla 54: Pronóstico de la cantidad de estudiantes que ingresan (fuente: elaboración propia).

- **Presupuesto de Pago Salarial anual.** (pagos a los docentes, coordinador y secretaria del coordinador).

A continuación se detalla los pagos de salario del personal de la Unidad de Cursos Libres datos y que formarán parte de los **gastos del proyecto**:

Cargo	personal	salario mensual (C\$)	Salario por temporada(C\$)	Total por año (C\$)	Observaciones
<b>Docente</b>	8		8700	208800	<ul style="list-style-type: none"> <li>• *Los salarios de los docentes son C\$ 8400 (no titulados) y C\$ 9000 a los titulados</li> <li>• Se abren 8 cursos en promedio por temporada (3 temporadas por año) y es un docente por curso</li> </ul>
<b>Secretaria del coordinador</b>	1	5080.9	15242.7	60970.8	Pago mensual
<b>Coordinador</b>	1	3666.66	10999.98	32999.94	pago mensual durante 9 meses

Tabla 55: Salarios del personal de los cursos libres Elaboración propia (fuente: Unidad de cursos libres VER ANEXO 4)

- **Gastos.**

A continuación se presenta un detalle de todos los gastos que tendrá que incurrir la Unidad de Cursos Libre durante el desarrollo del proyecto

**Gastos Operativos.**

Concepto	Total por año	Observaciones
<b>Pago de salario de los docentes</b>	208800	<b>Tabla 55</b> salario de los docentes
<b>Material Didáctico</b>	2160	<b>Ver Anexo 5</b>

Tabla 56 Gastos Operativo elaboración propia (fuente cursos libres)

En los gastos operativos se añade la emisión de diplomas el cual tiene un costo de **C\$ 30 córdobas** por diploma

**Gastos Administrativos.**

Concepto	Total por temporada(C\$)	Total por año (C\$)	observaciones
<b>Otros Gastos administrativos</b>	240	720	<b>Ver Anexo 5</b>
<b>Pago del dominio</b>		1531.13	Fuente <b>tabla 47</b> costo del hosting; precio del dólar al 20 de noviembre C\$ 30.6226
<b>Pago de salarios (personal administrativo)</b>	34942.68	302770.74	<b>Tabla 55</b> suma del salario del coordinador y su secretaría
<b>Total de Gastos Administrativos</b>		<b>C\$ 93,970.74</b>	

Tabla 57: Gastos administrativos elaboración propia, fuente unidad de cursos libres

### Gastos de venta.

Concepto	Total por temporada(C\$)	Total por año (C\$)	observaciones
<b>Publicidad</b>	36,232.53	108697.59	<b>Ver Anexo 5</b>

Tabla 58 Gastos de venta (fuente cursos libres)

- **Inversión del proyecto.**

A continuación se detalla las inversiones que totales para el proyecto.

Concepto	Costo total (\$)	Observaciones
<b>Costo total de software</b>	146438.72	
<b>Alojamiento del sitio web en MyAsp.net</b>	12708.07	Fuente <b>tabla 47</b> costo del hosting; precio del dólar al 20 de noviembre C\$ 30.6226
<b>Total de inversión del proyecto:</b>	<b>C\$ 159146.79</b>	

Tabla 59 Detalle de inversión de proyecto elaboración propia

Antes de calcular el Flujo neto de efectivo se debe de calcular la Tasa mínima atractiva de retorno (TMAR) que es la rentabilidad mínima deseada.

Para ello, se tiene los siguientes datos:

- Inversión inicial C\$ 159146.79 (Tabla 59)
- Inflación nacional acumulada: 4% promedio
- TMAR: 4%( Inflación nacional acumulada) + 10%(premio al riesgo) = 14%

Estos datos permitirán calcular el Flujo neto de efectivo; luego se expresará el análisis de la viabilidad del proyecto:

- **Flujo Neto de efectivo (Análisis del flujo)**

Con los datos anteriores se puede elaborar el siguiente **Flujo de efectivo** donde los ingresos son calculados multiplicando el pronóstico de estudiantes a matricularse (**Tabla 54**) por el precio de los cursos (\$3000) más la ganancia por la emisión de diplomas multiplicado por el pronóstico de alumnos. Los gastos crecen con la inflación del país

<b>FLUJO DE EFECTIVO SIN FINANCIAMIENTO (C\$)</b>						
<b>Descripción</b>	<b>AÑOS</b>					
	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>Total de ingresos</b>		518501.07	562432.14	606464.50	650603.84	694856.15
<b>Gastos Operativos</b>						
Gasto total de emisión de diplomas		5066.79	5710.71	6398.05	7131.24	7912.84
Salario de Docentes		208800.00	221328.00	234607.68	248684.14	263605.19
Material didáctico		2160.00	2246.40	2336.26	2429.71	2526.89
<b>Total Gastos operativos</b>		216026.79	229285.11	243341.99	258245.08	274044.92
<b>Utilidad operativa</b>		<b>302474.29</b>	<b>333147.03</b>	<b>363122.51</b>	<b>392358.75</b>	<b>420811.23</b>
<b>Gastos administrativos</b>		96221.87	101888.91	107895.41	114261.69	121009.32
<b>Gastos de ventas (Publicidad)</b>		108697.59	113045.49	117567.31	122270.01	127160.81
<b>Total de gastos</b>		<b>204919.46</b>	<b>214934.41</b>	<b>225462.72</b>	<b>236531.69</b>	<b>248170.13</b>
<b>Costo de Inversión</b>	-159146.79					
<b>FNE</b>	<b>-159146.79</b>	<b>97554.83</b>	<b>118212.62</b>	<b>137659.79</b>	<b>155827.06</b>	<b>172641.10</b>
<b>FNE (ACUMULADO)</b>		<b>-61591.97</b>	<b>56620.65</b>	<b>194280.45</b>	<b>350107.51</b>	<b>522748.61</b>

Tabla 60: Flujo neto de efectivo sin financiamiento elaboración propia.

Realizando los cálculos de los indicadores de rendimiento del proyecto sin financiamiento, se tiene que:

- El **valor presente neto (VPN)** del proyecto, con una TMAR del 14% es de C\$ 292231.33, siendo éste mayor que 1, por tanto el proyecto es aceptable.
- El **período de recuperación** (pr) es de **dos años**; es decir, que la Unidad de Cursos Libres recuperará su inversión a partir del segundo año de operación con el sistema.
- La **relación costo-beneficio** (RCB) es de 1.34, mayor que 1, quiere decir que los beneficios superan a los costos y además por cada córdobas invertido se esperara obtener 1.34 córdobas.

#### **Flujo neto con financiamiento.**

Para comprobar si los beneficios del proyecto serán mayores haciendo un préstamo bancario, se elabora un flujo con financiamiento partiendo de los mismos datos. El préstamo bancario se hará con una entidad del país a una tasa del 18% anual (para ver detalle del calendario de pagos revisar **Anexo 6**). El detalle del se muestra a continuación:

<b>Detalle del Financiamiento.</b>	
<b>PLAZO MESES</b>	60
<b>FINANCIAMIENTO:</b>	159,146.79
<b>TASA INICIAL FIJA</b>	18.00000%
<b>PAGOS MENSUALES:</b>	4,062.58
<b>FECHA DEL DESEMBOLSO.</b>	27-nov-17
<b>FECHA DE INICIO DEL PAGO.</b>	27-dic-17
<b>PAGO MENSUAL CON SEGUROS</b>	<b>4,148.64</b>
<b>TOTAL INTERESES PAGADOS AL VENCIMIENTO</b>	<b>27,292.98</b>

Tabla 61 Detalle de financiamiento fuente: banco LAFISE

FLUJO DE EFECTIVO CON FINANCIAMIENTO (C\$)						
Descripción	AÑOS					
	0	1	2	3	4	5
<b>Total de ingresos</b>		518501.07	562432.14	606464.50	650603.84	694856.15
<b>Gastos operativos</b>		216026.79	229285.11	243341.99	258245.08	274044.92
<b>Utilidad Operativa</b>		<b>302474.29</b>	<b>333147.03</b>	<b>363122.51</b>	<b>392358.75</b>	<b>420811.23</b>
<b>Gastos administrativos</b>		96221.87	101888.91	107895.41	114261.69	121009.32
<b>Gastos Financieros</b>						
pago de seguro del préstamo		86.06	86.06	86.06	86.06	86.06
Intereses del préstamo		27292.98	23032.17	17984.50	11816.17	4482.21
<b>Total Gastos financieros</b>		27379.04	23118.23	18070.56	11902.23	4568.27
<b>Gastos de ventas (Publicidad)</b>		108697.59	113045.49	117567.31	122270.01	127160.81
<b>Total de Gastos</b>		<b>232298.50</b>	<b>238052.64</b>	<b>243533.28</b>	<b>248433.92</b>	<b>252738.40</b>
<b>Utilidad Neta antes de la amortización</b>		70175.79	95094.39	119589.23	143924.83	168072.83
<b>Amortización al préstamo</b>		48750.96	48750.96	48750.96	48750.96	48750.96
<b>Costo de Inversión</b>	-159146.79					
<b>Financiamiento</b>	159146.79					
<b>FNE</b>	<b>0</b>	<b>21424.83</b>	<b>46343.43</b>	<b>70838.27</b>	<b>95173.87</b>	<b>119321.87</b>
<b>FNE (ACUMULADO)</b>		<b>21424.83</b>	<b>67768.26</b>	<b>138606.53</b>	<b>233780.39</b>	<b>353102.26</b>

Tabla 62: Flujo neto de efectivo con financiamiento elaboración propia.

Realizando los cálculos de los indicadores de rendimiento del proyecto con financiamiento, se tiene que:

- El **valor presente neto (VPN)** del proyecto con una TMAR del 18% es de C\$ 195800.43, siendo este mayor que 1, por tanto el proyecto es aceptable.
- El **período de recuperación** (pr) es de **un** año; es decir, que la Unidad de Cursos Libres recuperará su inversión a partir del primer año de operación con el sistema.
- La **relación costo-beneficio** (RCB) es de 2.04, mayor que 1: los beneficios superan a los costos y además, por cada córdoba invertido se esperará obtener C\$ 2.04.

### **Conclusiones del análisis Financiero**

Después de realizar este estudio se puede expresar:

- El proyecto es aceptable a través de cualquier indicador de rendimiento no importando si se realiza con o sin financiamiento.
- El periodo de recuperación sin financiamiento es de **1 año** y con financiamiento es de **2 años**
- El indicador de rendimiento del **VPN** muestra que el proyecto desarrollado sin financiamiento representará mayores ganancias (**C\$ 96430.90 córdobas más**) para la **Unidad de Cursos Libres** que en vez realizar un préstamo bancario.
- El indicador de rendimiento **RBC** muestra que realizando un préstamo bancario se obtendrá mayores beneficios (**2.04 córdobas más por cada córdoba invertido**) contra la ejecución del proyecto con capital propio.
- Queda a criterio de la Unidad de los Cursos Libres la opción a tomar para desarrollar el proyecto.

## 2.4. Estudio de factibilidad operacional

El objetivo de este estudio es, principalmente, definir si existen las condiciones mínimas necesarias para garantizar la factibilidad de la implementación, tanto en lo funcional, como en la disponibilidad de recursos humanos para conocer si el sistema será adecuado dentro de la organización y podrá ser utilizado por el personal.

Se puede decir a nivel operacional, que el personal ya tiene definidos los roles y tareas, los cuales están escritos en un manual de funciones. Esto permite un mayor control en la gestión y administración de los cursos. Además, dicho personal tiene conocimiento acerca del uso de computadoras por lo que la implementación del sistema a nivel operacional es factible.

De acuerdo al análisis de la aceptación, los actores de la organización manifestaron esperar reducir el gasto de papelería, que los tramites sean más rápidos y tener un registro histórico de datos que pueda ser consultado fácilmente y permita al Coordinador de cursos tomar decisiones. Así mismo, el Decano y el Coordinador manifestaron su interés y entusiasmo con el desarrollo del mismo.

Además, una encuesta realizada a los estudiantes (Ver Anexos 2 y 3) mostró que están de acuerdo con la implementación de un sistema que permita gestionar su información y los cursos que están llevando. Algunas de las funcionalidades que ellos consideran más relevantes para el sistema son las siguientes:

- Consulta de información de cada curso.
- Gestión de notas.
- Gestión de historial académico.
- Gestión de la información básica de los usuarios.
- Matrícula en línea

A nivel operativo, podemos afirmar que es factible desarrollar el proyecto porque:

- Las actividades, tareas y funciones que realiza cada actor de la organización están bien definidas.
- Los usuarios conocen la forma de utilizar equipos de computadoras
- Los estudiantes, docentes y personal administrativo manifiestan estar de acuerdo con el desarrollo de un sistema informático para la gestión en los Cursos Libres.

## 2.5. Consideraciones legales

Los proyectos informáticos deben de cumplir con el marco legal aplicable nacional y/o internacional de forma tal que se cumpla con las restricciones pertinentes en cuanto a uso de licencias del software adquirido, protección de la información de los usuarios y cláusulas contractuales con el cliente.

Aunque en Nicaragua no existe una ley que norme específicamente el desarrollo de software existen un conjunto de leyes que actúan como un marco generalizado dentro del cual es recomendable desarrollar cualquier proyecto como la Ley de Derecho de Autor y Derechos Conexos (Ley No 312) y su reglamento; la Ley de Protección de Datos Personales (Ley No. 787). En el caso particular del presente proyecto se debe considerar la Normativa de Culminación de Estudios de la Universidad Nacional de Ingeniería que establece propiedades de derechos de autoría y su explotación.

El uso de programas de cómputo como las herramientas de desarrollo debe de ser conforme la Ley de derechos de Autor, asegurando la posesión de la debida licencia que otorgue para su correspondiente utilización. En el caso de Visual Studio existe una versión *community* y en SQL Server una versión *developer* completamente funcionales y que permiten codificar el sistema sin incurrir en costos de licencias adicionales

Según el Artículo 4 de la Ley 312, el derecho de autor corresponde al mismo por el sólo hecho de la creación de la obra, comprendiendo facultades morales y patrimoniales que confieren al autor la plena disposición y el derecho exclusivo de explotación de la obra, sin más limitaciones que las establecidas en la presente Ley.

Esta ley define un programa de cómputo como: un conjunto de instrucciones expresadas mediante palabras, códigos, gráficos, diseños o cualquier otra forma que, al ser incorporadas en un dispositivo de lectura autorizada, es capaz de hacer que ordenador, un aparato electrónico o similar capaz de elaborar informaciones,

ejercite determinada tarea u obtenga determinado resultado. También forma parte del programa su documentación y sus manuales de uso.

En consistencia con la Ley de Protección de Datos, si el sistema a desarrollar obtiene, utiliza o trata datos de personas, éstos deben ser meritorios y proporcionales según el fin para el que se colectan; así mismo deben de ser recopilados por un medio idóneo que permita otorgar su consentimiento (físico o electrónico). El motivo y la forma de uso de estos datos o información deben ser conocidos por los usuarios que accedan al sistema.

No se les podrá solicitar más datos de los necesarios a los usuarios para el uso del sistema al que ellos le darán ya que el Arto. 9 de la Ley de Protección de Datos establece que *“ninguna persona que solicite la prestación o adquisición de bienes o servicios está obligada a brindar a las instituciones públicas y privadas mayor información o datos personales que aquellos que sean adecuados, proporcionales y necesarios para la prestación de los mismos.”* (Ley 787, 2012)

De igual manera en las etapas del ciclo de vida del sistema se deberá considerar la obligatoriedad legal de implementar las medidas técnicas necesarias para *“garantizar la integridad, confidencialidad y seguridad de los datos personales”*. (Artos. 9 y 10, Ley 787, 2012).

### **Conclusiones:**

Desde el punto de vista legal, es factible desarrollar el sistema si se toma en consideración el marco regulatorio nacional y las licencias de software.

## Capítulo IV: Diseño

La especificación del metamodelo de UWE indica que los modelos de diseño de una aplicación web deben de expresar cuatro áreas fundamentales: los aspectos de la información de la aplicación; la estructura de hipertexto y la funcionalidad de la navegación; el esquema de la presentación; y la funcionalidad de la aplicación.

### 4.1. Modelo de requerimientos

#### 4.1.1. Identificación de actores

Los participantes ya sea miembros de la organización o agentes externos, se describen a continuación como actores del negocio.

- **Coordinador de los cursos libres:** se encarga de conjugar los procesos internos que se realizan: gestión de grupos, planificación de horarios y también la gestión de pagos e informes.
- **Secretaria del coordinador:** apoya en la gestión de costos, de pagos e informes al coordinador de los cursos libres.
- **Docente:** imparte cursos a los estudiantes y entrega informes de las evaluaciones realizadas
- **Decano:** es el principal responsable, autoriza los gastos y solicita informes financieros
- **Asistente administrativo de la FCYS:** solicita informes financieros.
- **Secretaria Académica de la FCYS:** Firma las actas de notas y las registra en el libro correspondiente.
- **Estudiante:** es la persona a la que se le brinda el servicio de enseñanza, requiriendo conocer su historial de notas, estado de pagos e informes académicos.

#### **4.1.2. Modelo de caso de uso del negocio**

Este modelo describe los escenarios básicos que conforman el giro del negocio de la organización, comprendiendo los procesos de negocio y detallando los actores y casos de uso, desde una visión externa al modelo de negocio. La figura 4 refleja el modelo del negocio para los cursos libres de la FCYS.

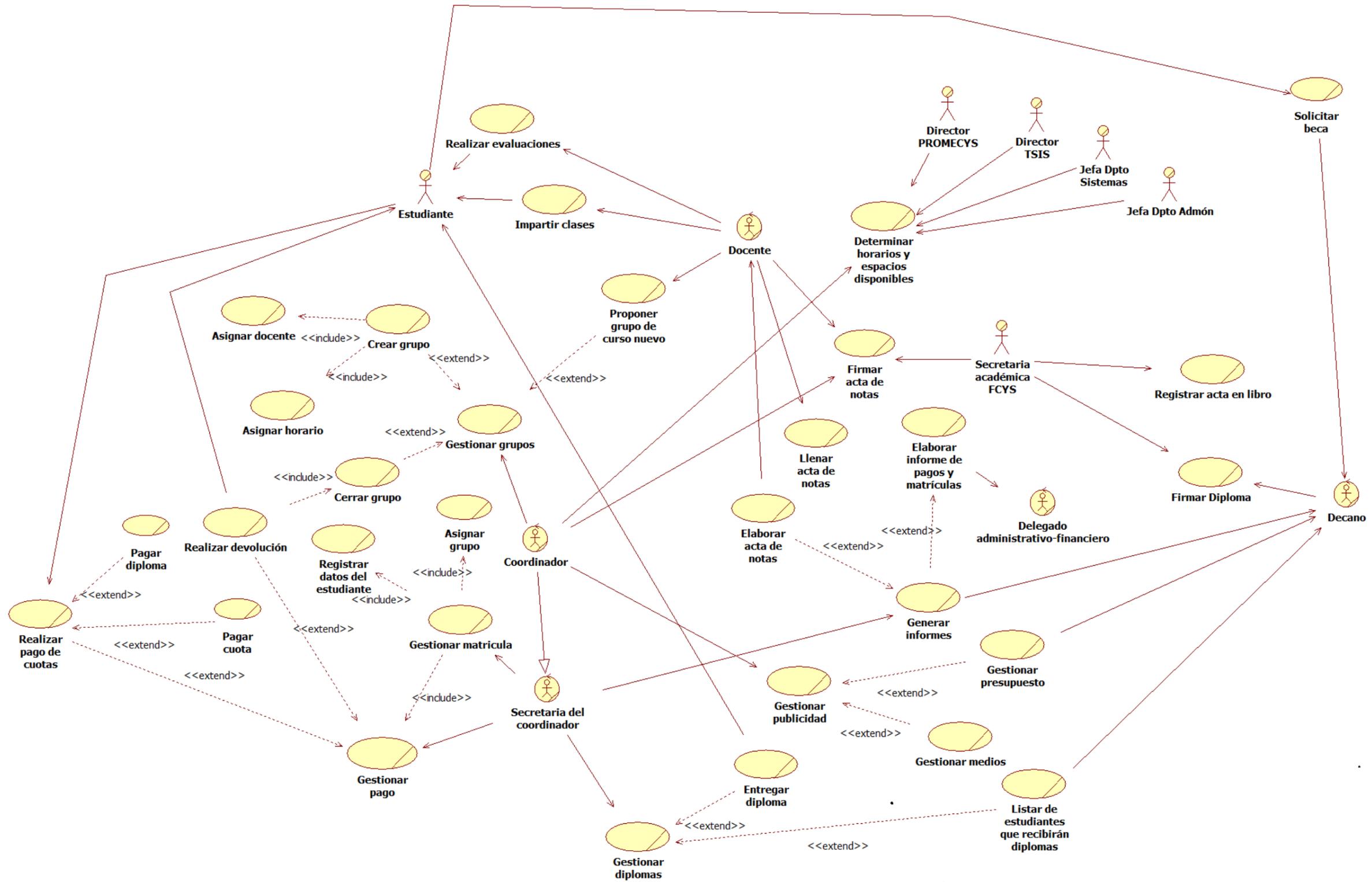


Figura 5: Diagrama de caso de uso del negocio (fuente: elaboración propia).

### 4.1.3. Diagramas de actividad del negocio

Representa los flujos de control de las operaciones en la organización; éste es el siguiente paso del modelo unificado RUP en el que se basa UW (véase anexo 9.1).

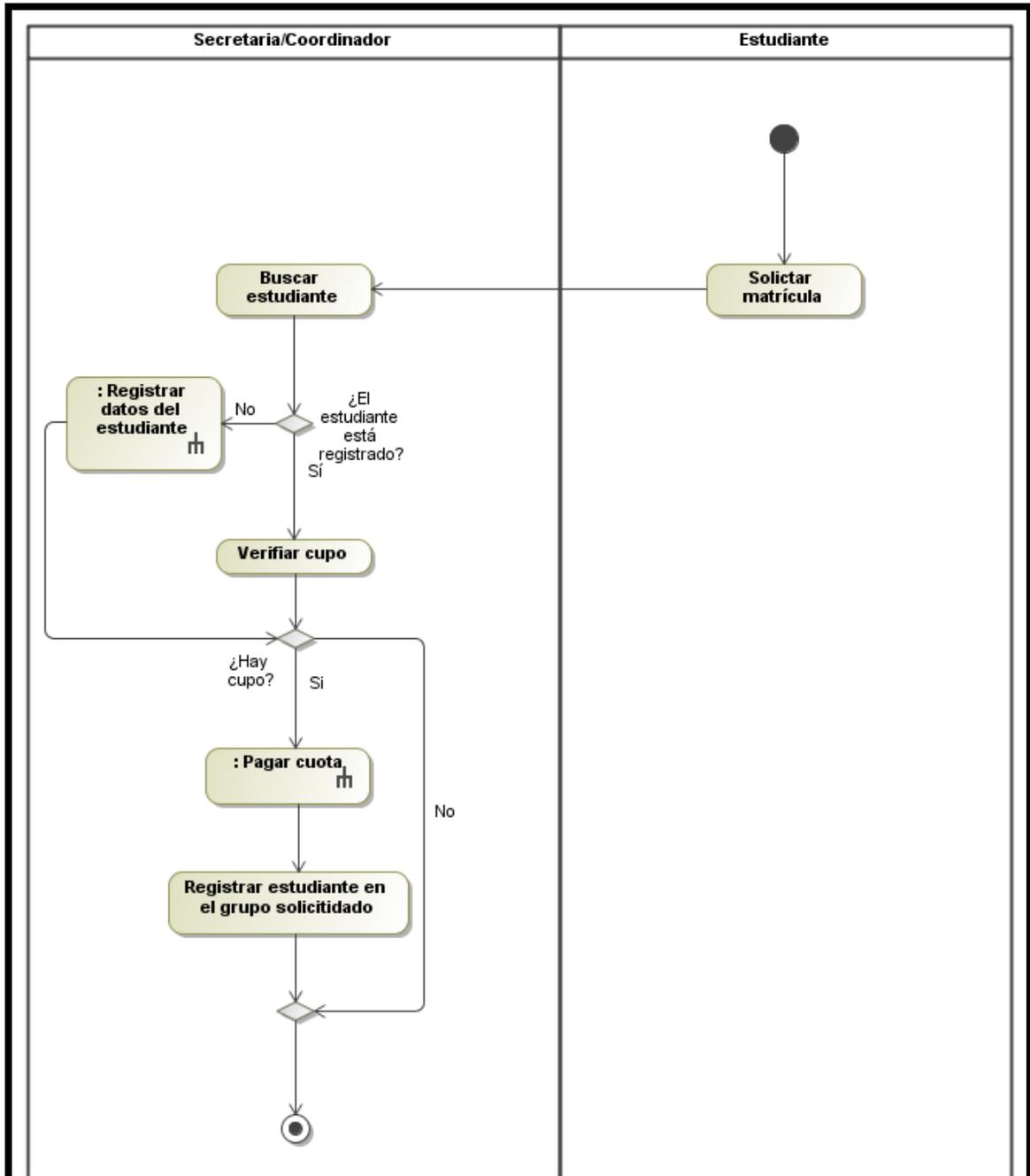


Figura 6: Diagrama de actividad del proceso de gestión de matrícula (fuente: elaboración propia)

#### 4.1.4. Modelo del sistema

##### 4.1.4.1. Actores del sistema

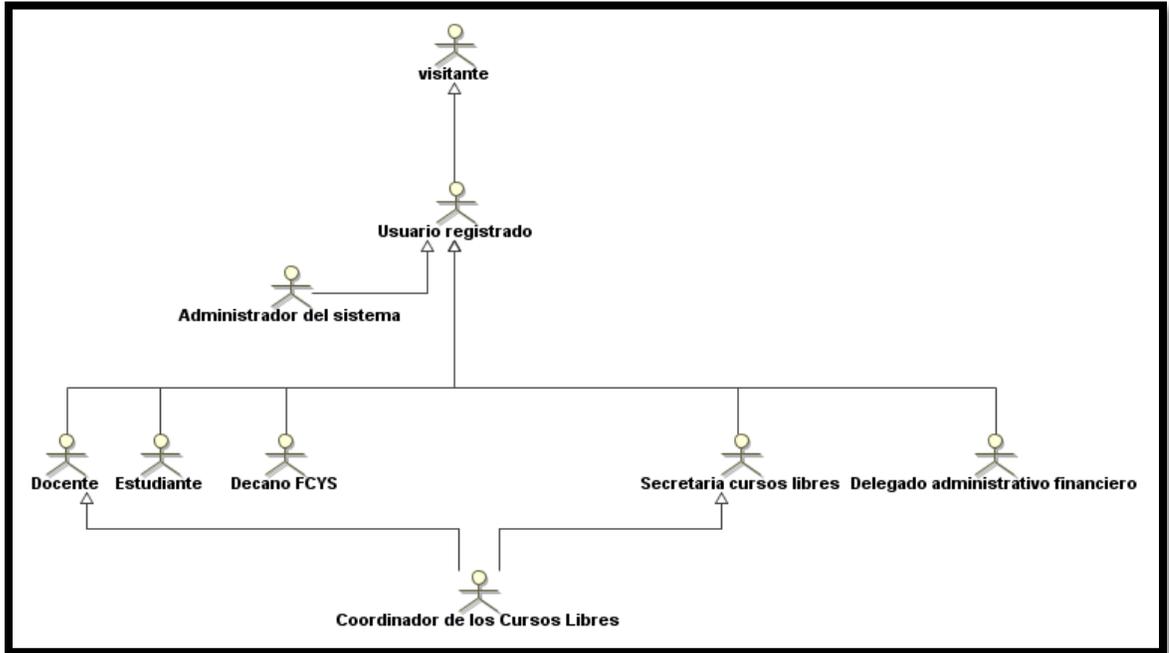


Figura 7: Jerarquía de actores del sistema

#### 4.1.4.2. Diagrama de paquetes funcionalidad del sistema

Un diagrama de paquetes muestra cómo un sistema está dividido en agrupaciones lógicas mostrando las dependencias entre ellas.

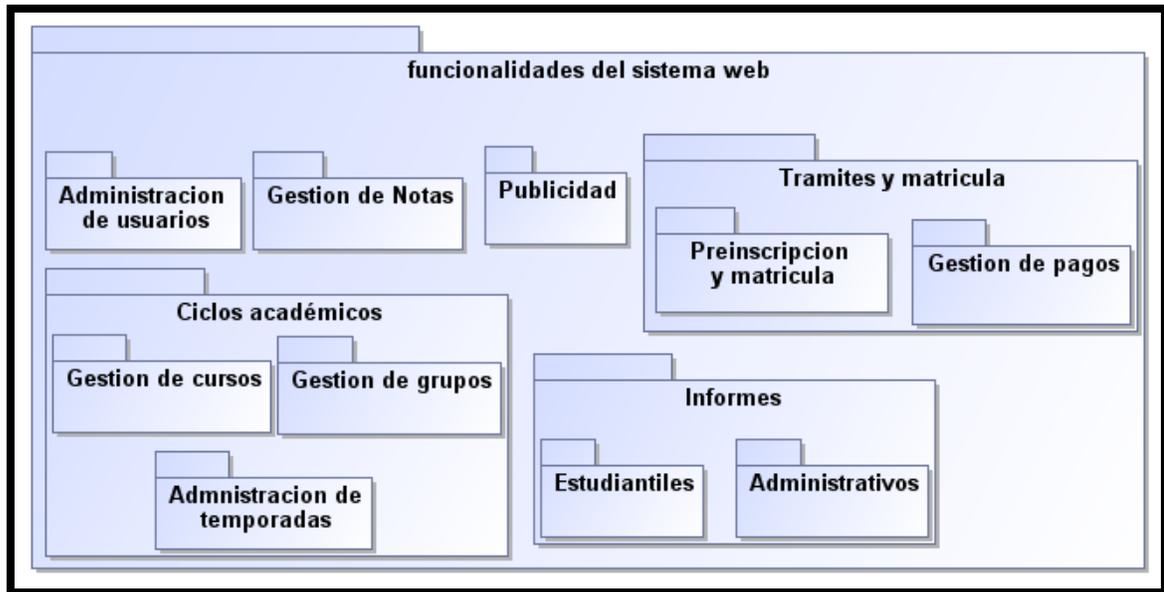


Figura 8: Diagrama de paquetes (fuente: elaboración propia).

Suministran una descomposición de la jerarquía lógica de un sistema. Están normalmente organizados para maximizar la coherencia interna dentro de cada paquete y minimizar el acoplamiento externo entre cada uno de ellos. Con estas líneas maestras sobre la mesa, los paquetes son elementos de gestión. Cada paquete puede asignarse a un individuo o a un equipo, y las dependencias entre ellos pueden indicar el orden de desarrollo requerido.

#### 4.1.4.3. Diagrama de casos de uso del sistema

Describe los procesos de un negocio, vinculados al campo de acción, y cómo se benefician e interactúan los socios y clientes en estos procesos (véase figura 9).

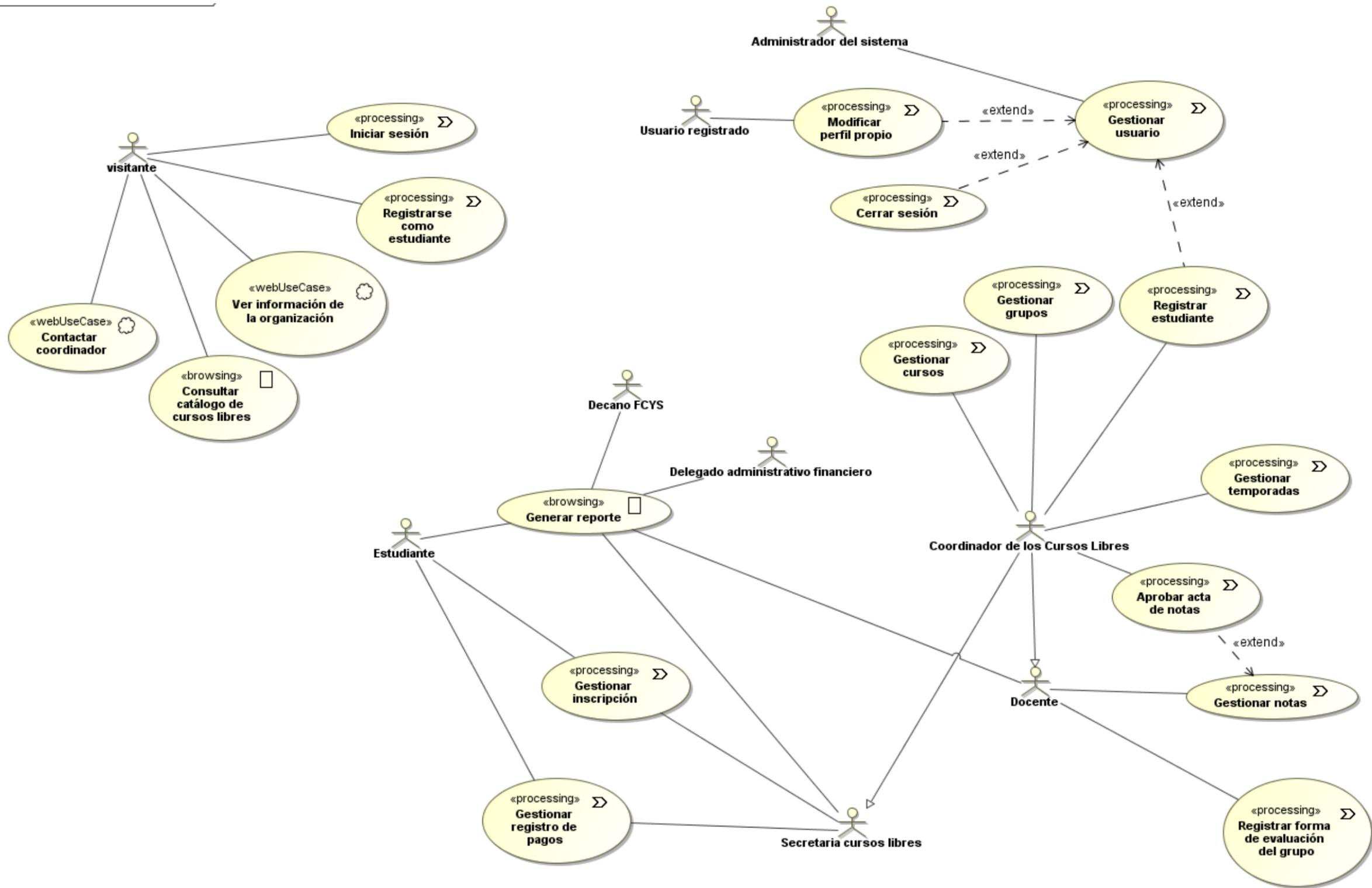


Figura 9: Modelo del negocio. Diagrama de caso de uso del negocio (fuente: elaboración propia).

## Descripción de los casos de uso:

Un diagrama de caso de uso es la representación gráfica de una serie de tareas que deberán realizarse para completar un proceso dentro del sistema. En la **tabla 63** se describe el caso de uso de la **figura 10**; es el caso de uso del sistema gestionar inscripción.

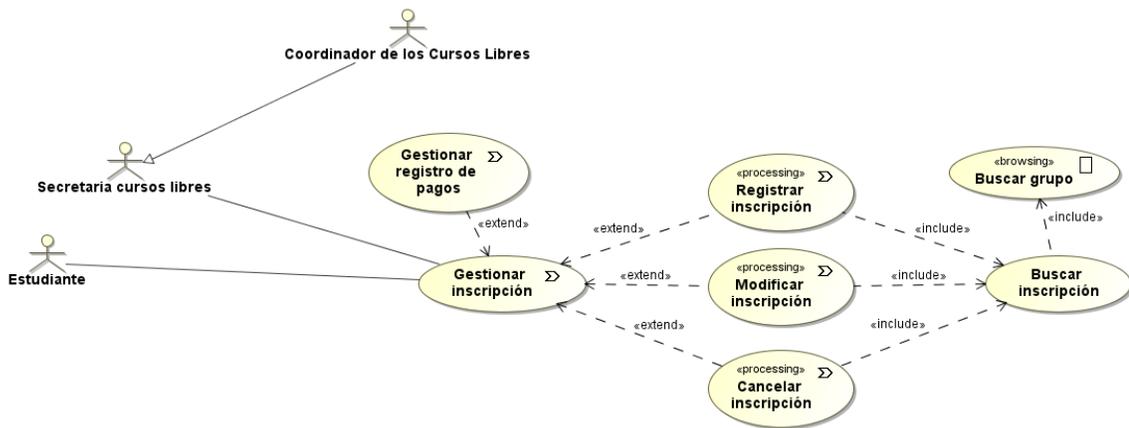


Figura 10: Diagrama de caso de uso del sistema. Escenario: Gestionar inscripción presencial (fuente: elaboración propia).

<b>UC-0026</b>	<b>Inscribir estudiante en grupo</b>	
<b>Versión</b>	1.0 ( 08/12/2017 )	
<b>Autores</b>	<a href="#">Gabriel Ignacio Chávez Toruño</a> <a href="#">Michael Bryang Chávez Angulo</a>	
<b>Fuentes</b>	?	
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <a href="#">[OBJ-0002] Gestionar los usuarios: empleados, estudiantes y visitantes externos</a></li> <li>• <a href="#">[OBJ-0004] Gestionar el proceso de matrícula</a></li> </ul>	
<b>Descripción</b>	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <i>al dar clic en el botón matricular el estudiante quedará registrado en el curso que eligió</i>	
<b>Precondición</b>	Haber iniciado sesión un usuario con permisos suficientes	
<b>Secuencia normal</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor <a href="#">_Secretaria del coordinador de los cursos libres (ACT-0007)</a> busca al estudiante
	2	El actor <a href="#">_Secretaria del coordinador de los cursos libres (ACT-0007)</a> selecciona el curso y el grupo donde inscribir al estudiante verificando si tiene descuento o beca
	3	El actor <a href="#">_Secretaria del coordinador de los cursos libres (ACT-0007)</a> indica el tipo de inscripción ( preinscripción o matrícula)
	4	El sistema verifica que existan cupos disponibles
	5	El sistema <i>inscribe al estudiante grupo elegido y notifica al usuario. El estado de su inscripción indica que espera el pago de matrícula</i>
	6	Se realiza el caso de uso <a href="#">Registrar abono al pago (UC-0022)</a> ( se debe de pagar el concepto de matrícula)
<b>Postcondición</b>	el estado de la inscripción debe ser matriculado	
<b>Excepciones</b>	<b>Paso</b>	<b>Acción</b>
	4	Si <i>no hay cupos</i> , el sistema <i>alerta al usuario</i> , a continuación este caso de uso <i>queda sin efecto</i>
<b>Rendimiento</b>	<b>Paso</b>	<b>Tiempo máximo</b>
	-	-
<b>Frecuencia esperada</b>	<b>PD</b>	
<b>Importancia</b>	vital	
<b>Urgencia</b>	hay presión	
<b>Estado</b>	pendiente de verificación	
<b>Estabilidad</b>	media	
<b>Comentarios</b>	Ninguno	

Tabla 63: Diagrama de caso de uso del negocio: Inscribir presencialmente estudiante en grupo (fuente: elaboración propia).

#### 4.1.4.4. Diagramas de secuencia

Muestra la forma en que los objetos se comunican entre sí al transcurrir el tiempo. Formalmente se utilizan para validar los casos de uso. Todos los diagramas de secuencia representan el flujo que se sigue para cada escenario en el proceso del sistema (**Ver anexo 11.1 Diagramas de secuencia**). A continuación (figura 11) se muestra uno de los diagramas de secuencia más relevantes como es la inscripción de estudiantes:

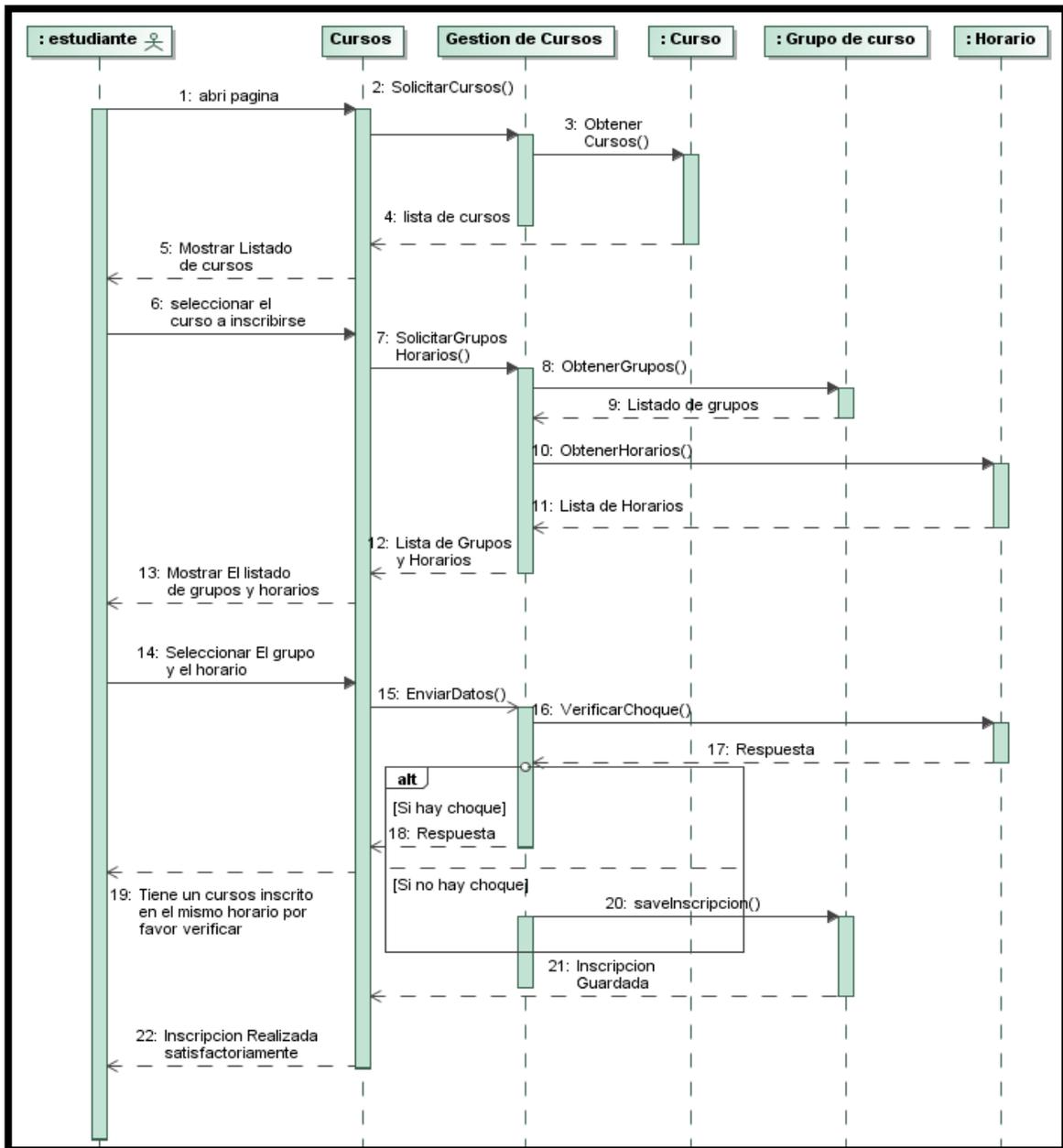


Figura 11: Diagrama de Secuencia Inscripción de estudiantes (fuente: elaboración propia).

#### 4.1.5. Diagramas de Colaboración.

Explican gráficamente cómo los objetos interactúan a través de mensajes para realizar las tareas. Muestra cómo las instancias específicas de las clases trabajan juntas para conseguir un objetivo común e implementa las asociaciones del diagrama de clases mediante el paso de mensajes de un objeto a otro. Dicha implementación es llamada "enlace". **(Ver anexo 11.2 Diagramas de colaboración)**. A continuación se muestra en la figura 12 el siguiente diagrama de colaboración.

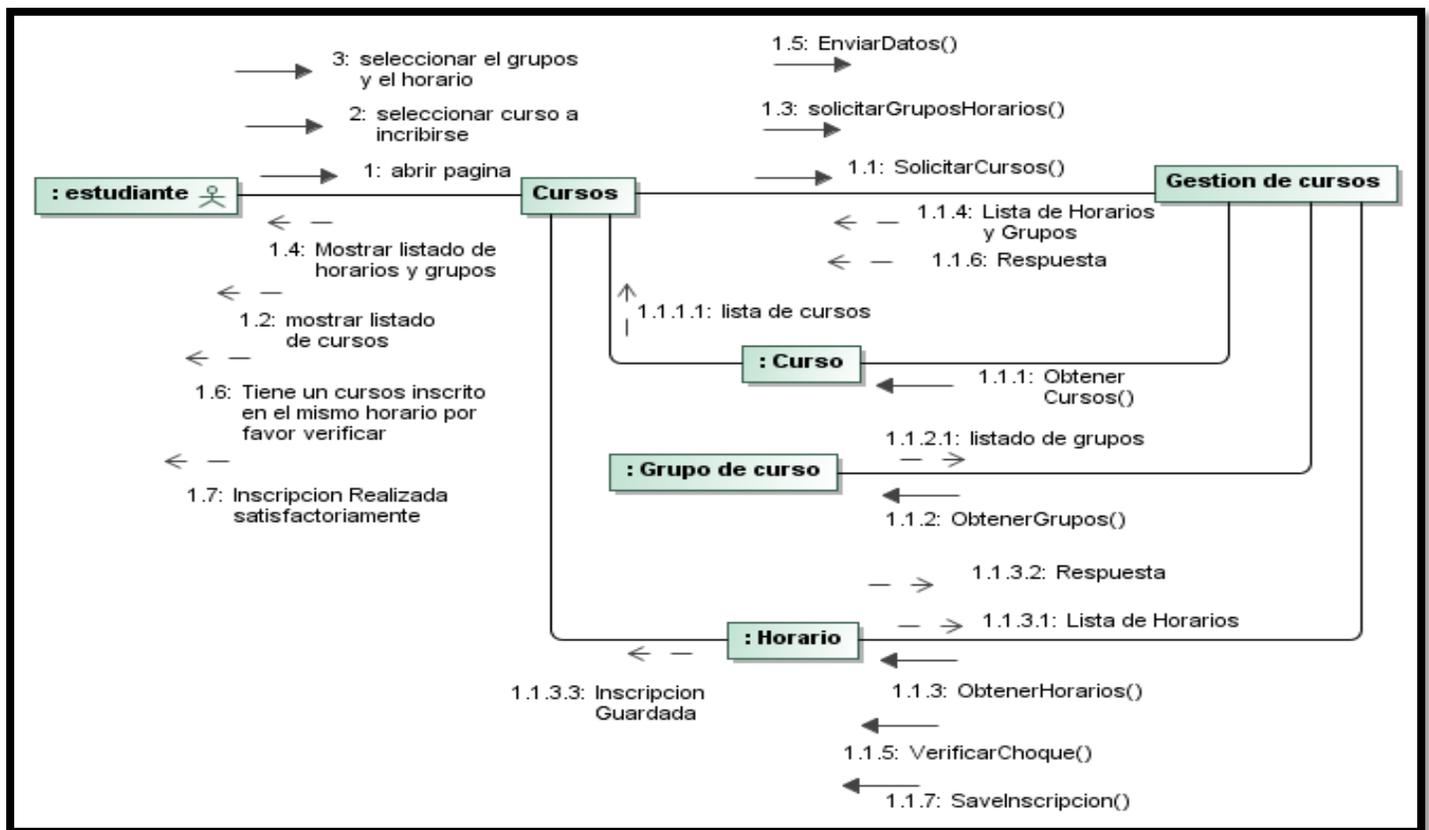


Figura 12: Diagrama de Colaboración Inscripción de estudiantes (fuente: elaboración propia).

#### 4.2. Modelo de contenido:

Comprende clases necesarias para conceptualizar el dominio de información de la aplicación. La siguiente imagen detalla el modelo de contenido para el sistema de información web de los cursos libres.

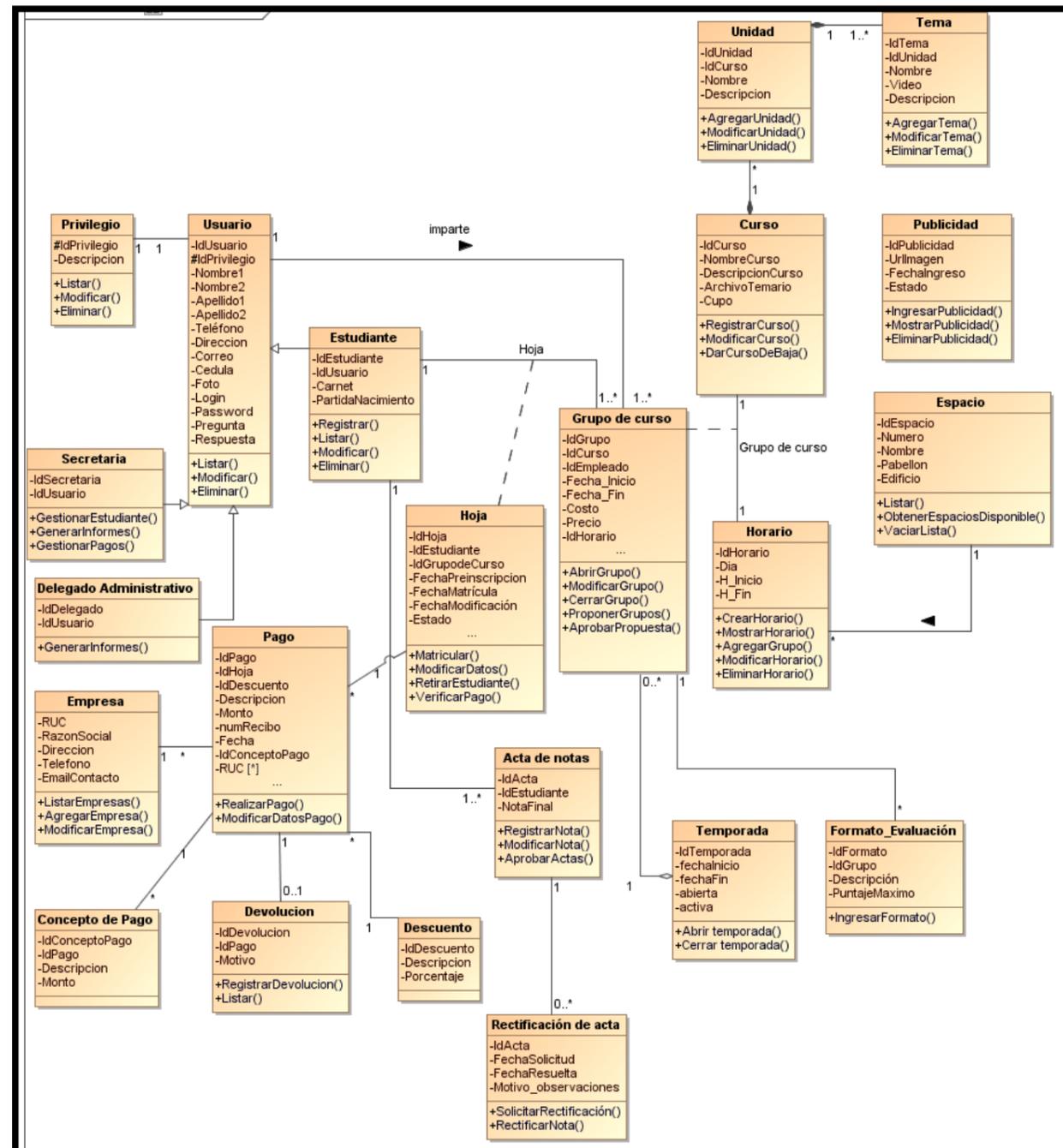


Figura 13: Modelo de contenido (fuente: elaboración propia).

### 4.3. Modelo de navegación:

El modelo de navegación pretende representar las rutas de navegación que los posibles usuarios tomarán durante su experiencia con el sistema, los nodos y enlaces de la estructura de hipertexto en la figura 14 se presenta el siguiente diagrama de navegación. (Véanse los diagramas de navegación en la sección 11.3 de los anexos).

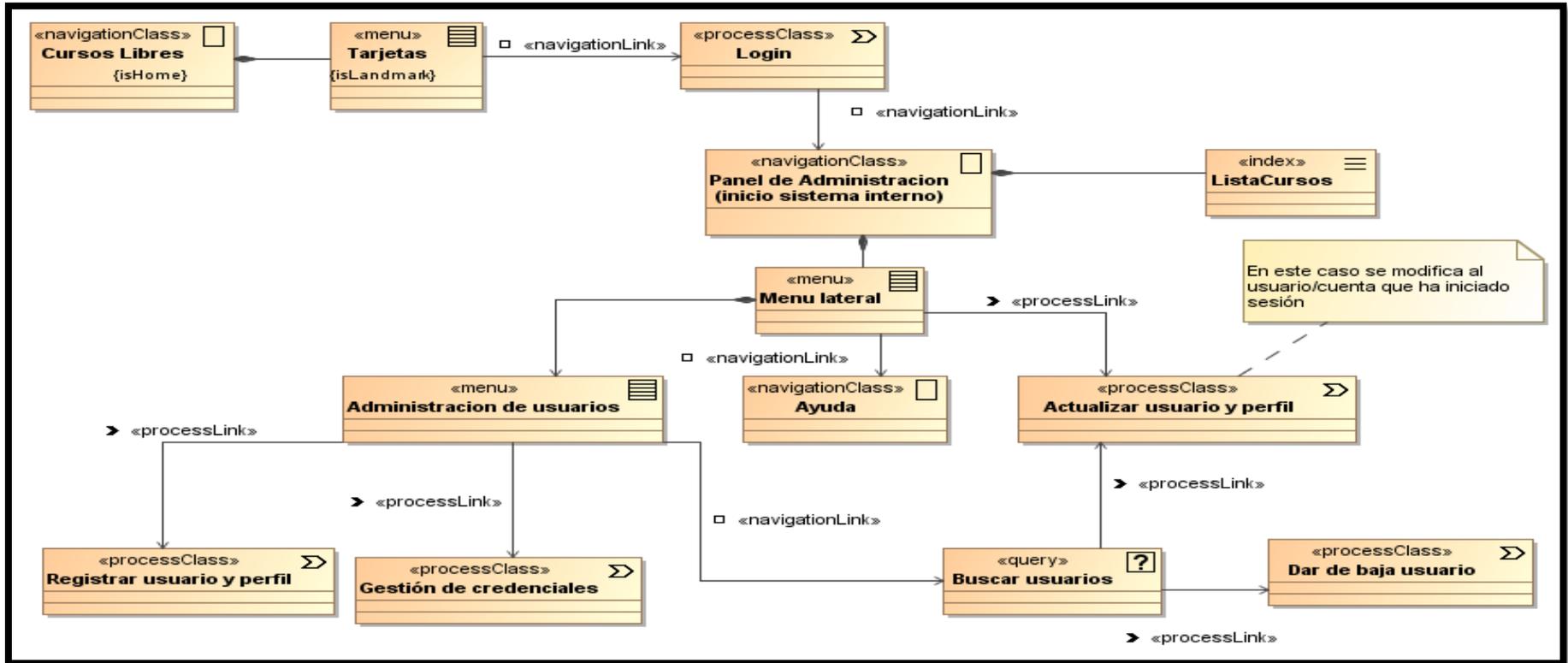


Figura 14: Modelo de navegación (fuente: elaboración propia).

#### 4.4. Diagrama de Estados

Muestra las diferentes condiciones que pueden atravesar un objeto a lo largo de su vida y los estímulos que llevan a aquellas. Las figuras 15, 16 y 17 muestran los diagramas de estado de tres clases dentro del sistema que cambian su estado eventualmente según las acciones de los usuarios: **Hoja, Grupos de curso y Actas de notas** respectivamente.

Para preinscribirse, el estudiante, una vez autenticado, ingresa en la página de cursos y selecciona el de su preferencia; el sistema verifica la disponibilidad de grupos, horarios y cupos del curso, luego el usuario selecciona el grupo donde desea inscribirse y el sistema registrara la preinscripción. Otra alternativa, es la matrícula directa sin necesidad de una preinscripción previa para ello el estudiante debe dirigirse donde el coordinador o la secretaria del mismo, los cuales harán el registro de la matrícula en el sistema (el sistema también verificara la disponibilidad grupos, horarios y cupos). En caso de no haber disponibilidad en el grupo, sólo se notifica que no hay disponibilidad y se termina el proceso.

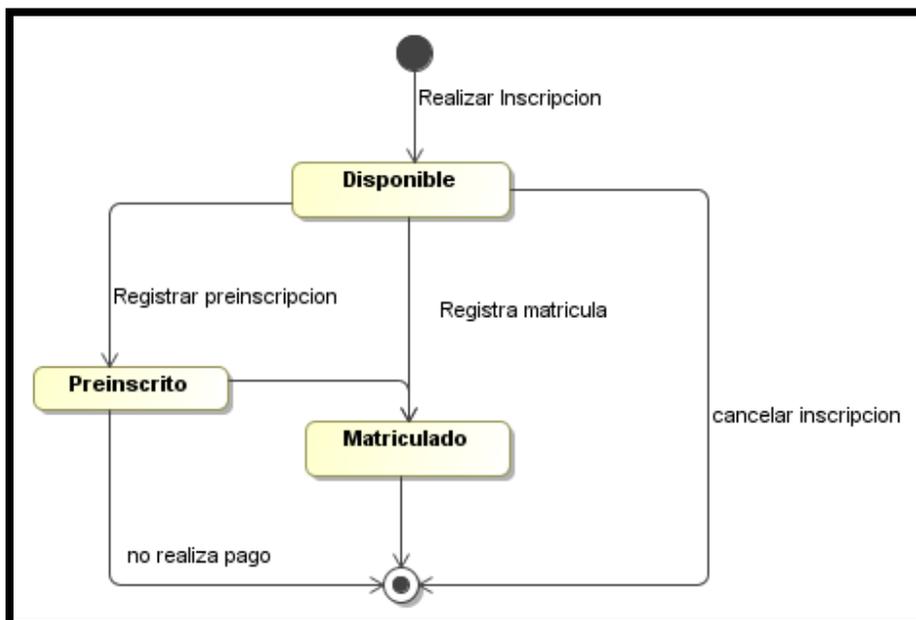


Figura 15: Diagrama de estado de Inscripción del estudiante (fuente: elaboración propia)

Al momento de crear un grupo pueden ocurrir 2 escenarios: que sean creados por el coordinador o propuestos por el docente. El coordinador de los cursos libres es el encargado de abrir, cerrar y descartar cualquier grupo de curso por los motivos que el considere.

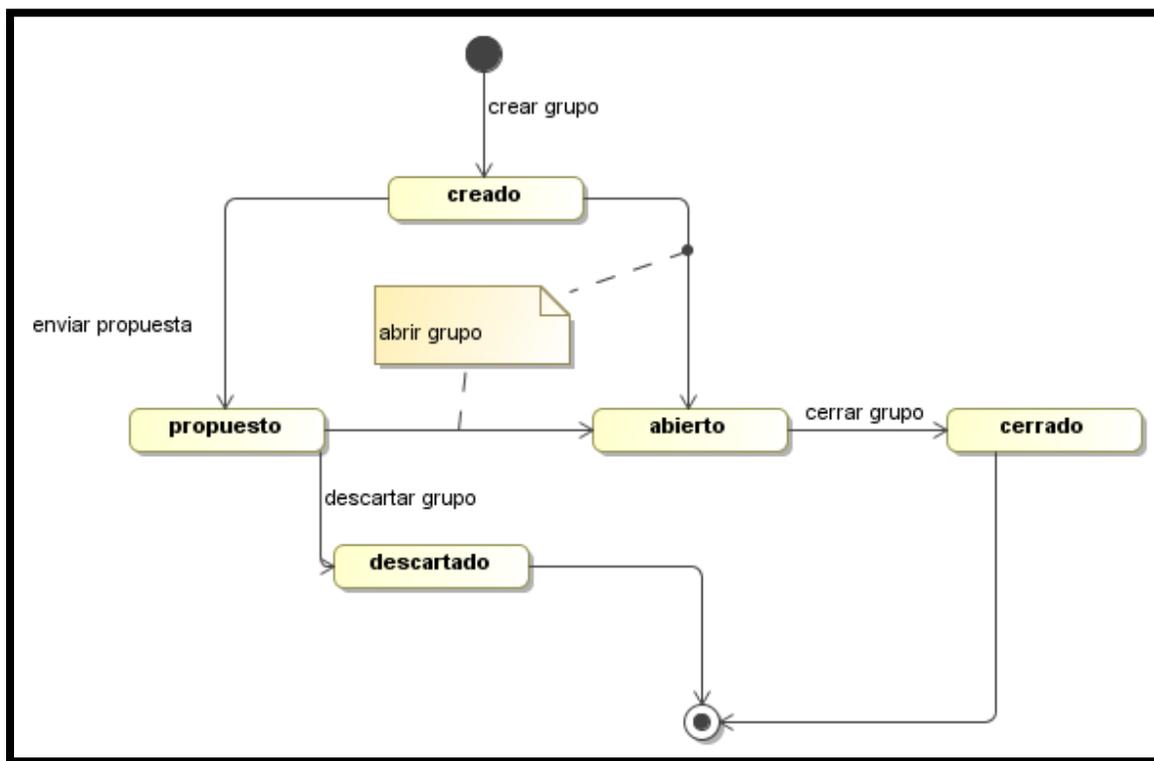


Figura 16: Diagrama de estado Grupos de Curso (fuente: elaboración propia)

Las notas de los alumnos son registradas únicamente por docente, según las evaluaciones que el mismo haya establecido para el curso. Las rectificaciones se llevan a cabo cuando el estudiante comunica al coordinador de los Cursos Libres que hay un error en su nota; éste mediante el sistema ingresa una solicitud de rectificación, la cual llega al docente y luego el sistema le habilita la edición de notas.

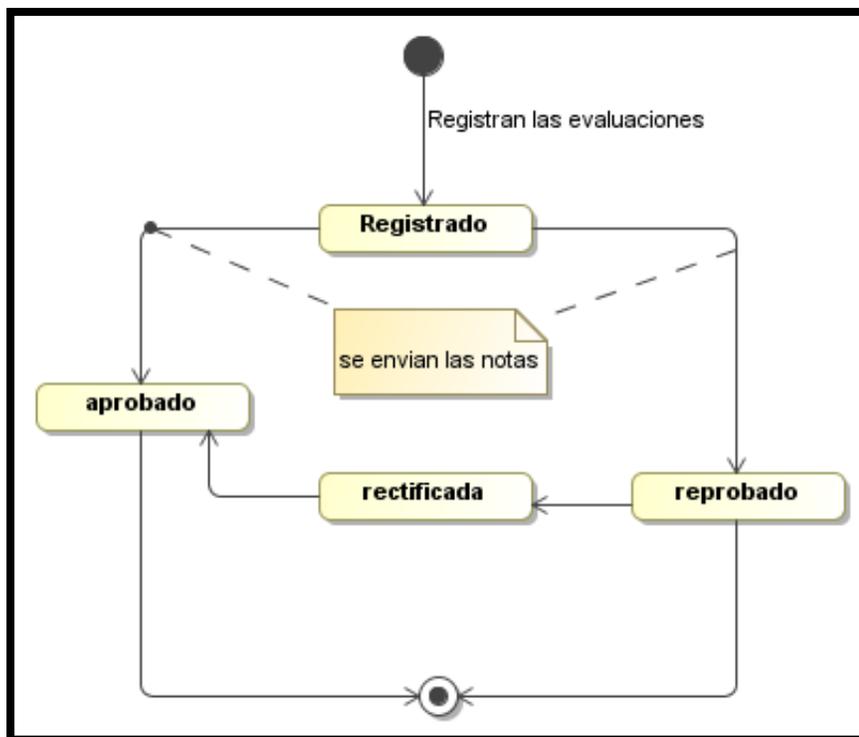


Figura 17: Diagrama de estado Gestión de Notas (fuente: elaboración propia)

#### 4.5. Modelo de procesos:

La siguiente figura ilustra la funcionalidad y transacciones que otorgan dinamismo al sistema (Véanse los diagramas en la sección 11.5 de los anexos).

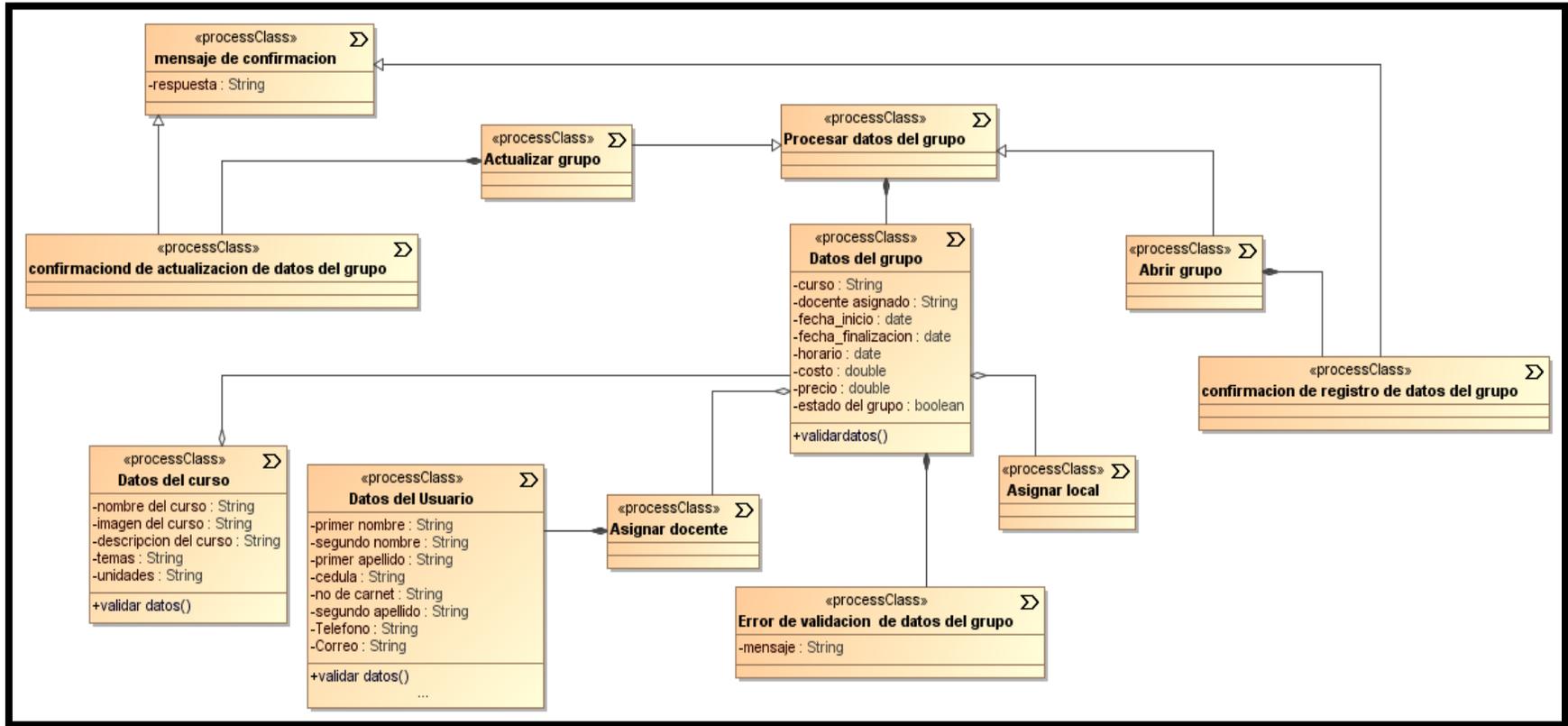


Figura 18: Diagrama de estructura de proceso. Escenario Gestión de Grupos (fuente: elaboración propia).

## 4.6. Diagrama de Actividad

Este tipo de diagramas detallan mediante flujos de trabajo, el comportamiento del sistema para realizar los procesos. Estas acciones las pueden llevar a cabo personas, componentes de software o equipos. (Ver los diagramas de actividad en la sección 11.6 de los anexos).

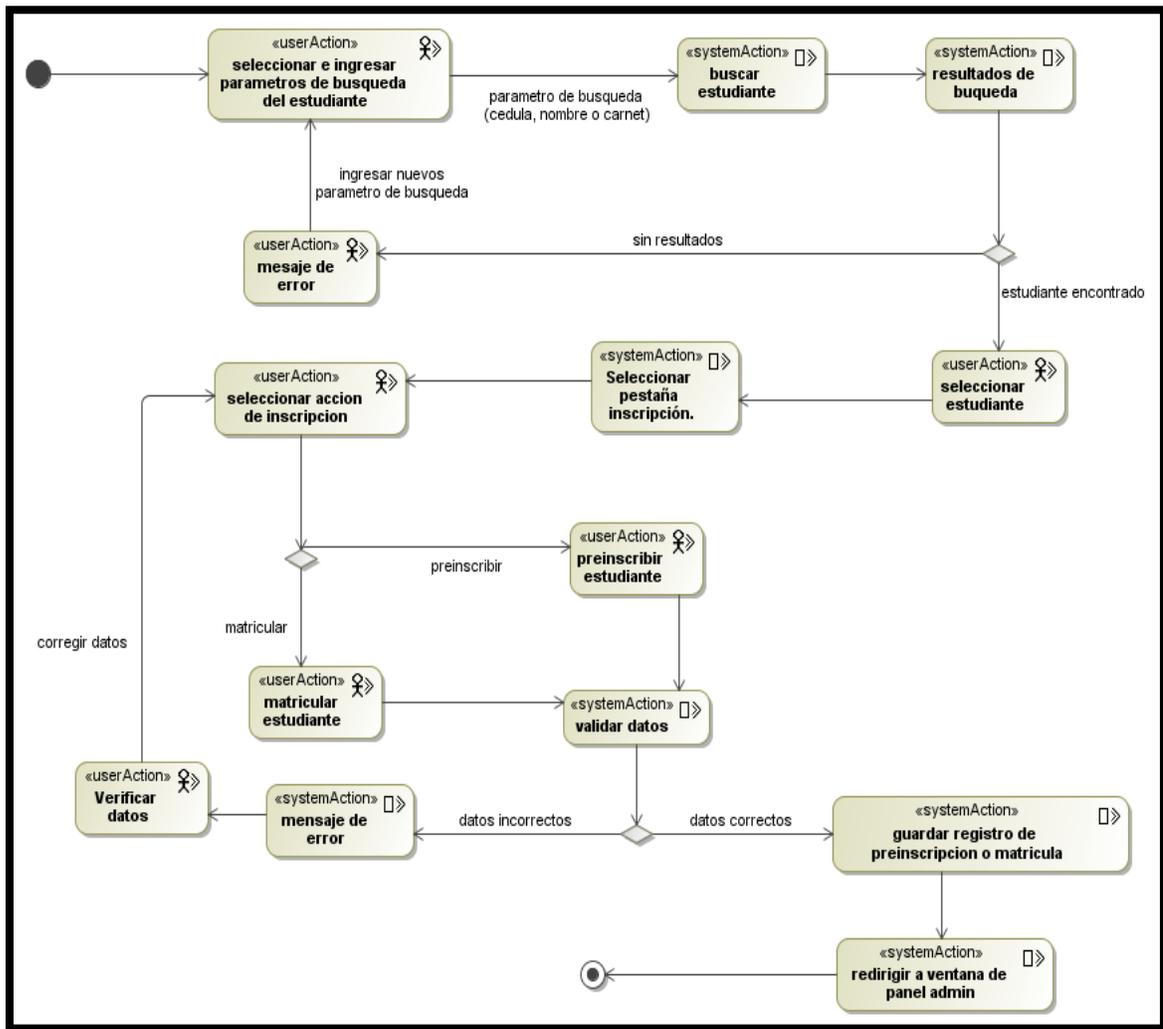


Figura 19: Diagrama de actividad. Escenario Gestión de matrícula (fuente: elaboración propia).

#### 4.7. Modelo de presentación:

Provee una visión abstracta del diseño de la interfaz de usuario de una aplicación web. Se basa en el modelo de navegación, del cual representa la estructura básica (sin tomar en cuenta colores, fuentes o tecnología específica) de sus nodos. Cada uno de ellos está representado mediante *clases*, *páginas* y *grupos de presentación*.

La figura 20 muestra la vista de inicio de sesión. El resto de diagramas puede consultarse en la sección 11.4 de los anexos.

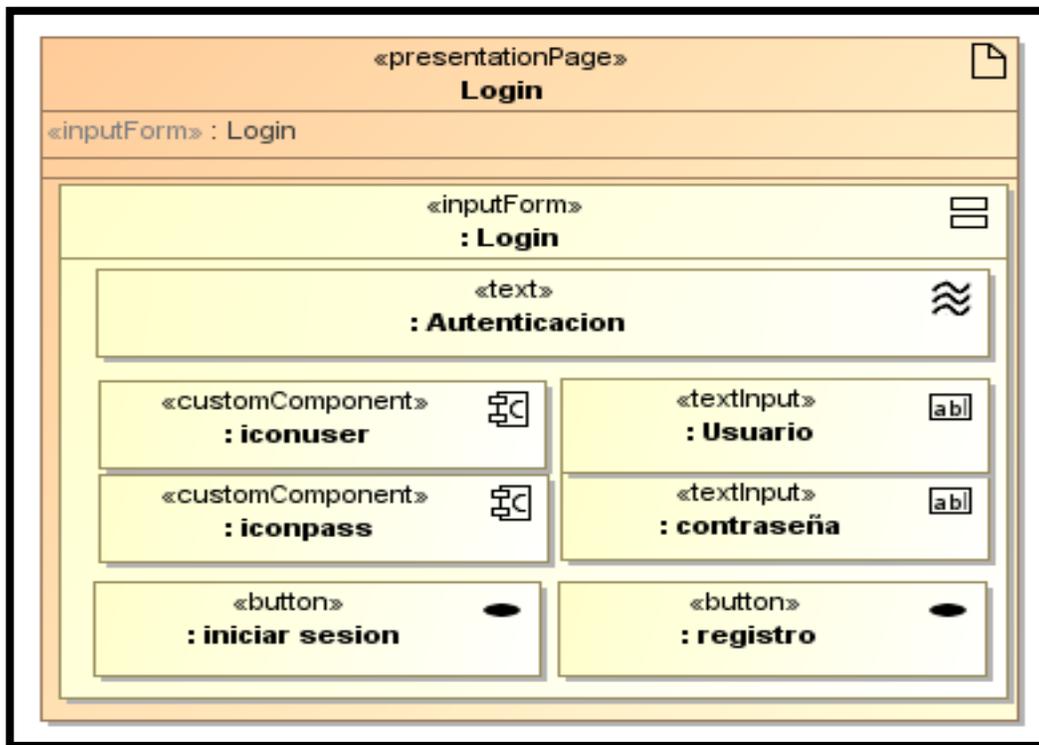


Figura 20: Modelo de presentación. Página de inicio de sesión (fuente: elaboración propia)

## Capítulo V: Implementación

Para la codificación del proyecto se utilizó una arquitectura cliente-servidor bajo el entorno de desarrollo .NET, en el lenguaje de programación C# y bajo el patrón MVC4. La capa de datos fue abstraída mediante Entity Framework 5, permitiendo la conexión con la base de datos de SQL Server.

### 5.1. Modelo físico

El modelo físico describe de la implementación de la base de datos, las estructuras de almacenamiento, la arquitectura de datos y las relaciones entre ellos, facilitando la comprensión de la información de una manera gráfica. **(Véase anexo 13)**

### 5.2. Diagrama de componentes

Este diagrama describe los elementos físicos y sus realizaciones en el entorno. Representa cómo un sistema de software es dividido en componentes y muestra las dependencias entre ellos. **(Ver Anexo 12).**

### 5.3. Diagrama de despliegue

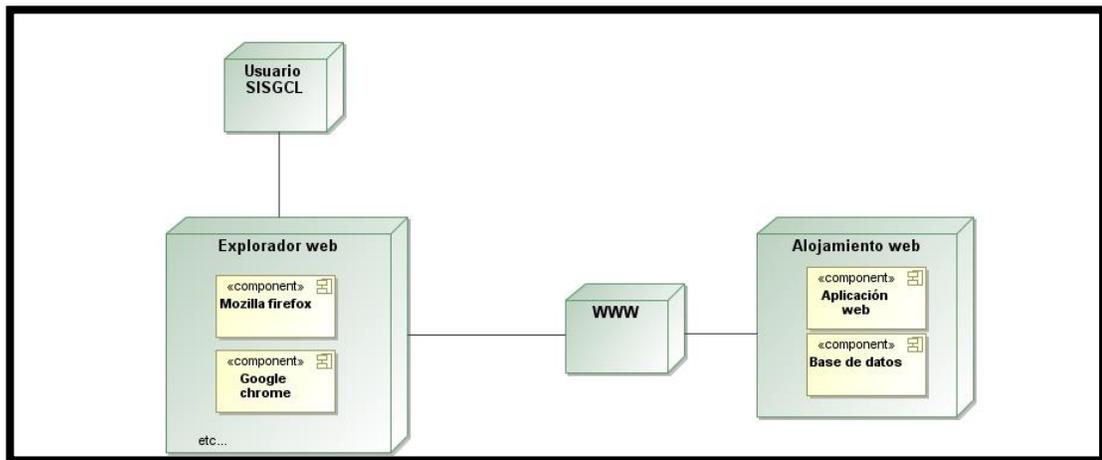


Figura 21: Diagrama de despliegue (fuente: elaboración propia).

### 5.4. Medidas de Seguridad.

Las medidas de seguridad empleadas en el sistema fueron:

- Agrupación de usuarios del sistema por roles y permisos.
- Cifrado de la información al momento de iniciar sesión
- Bloqueo de cuenta de usuario al realizar tres Intentos fallidos de inicio de sesión
- Cifrado de las contraseñas a nivel de base de datos mediante el algoritmo SHA\_256.
- Sesión del usuario en el sistema web expira en un día
- Cifrado de parámetros en las URL del sistema web.
- Implementación del Recaptcha de Google para validar el registro de estudiantes
- Verificación de autenticación del usuario en cada petición a una URL del sistema vía POST, GET, DELETE, PUT y mediante Ajax.

## Capítulo VI: PRUEBAS DEL SOFTWARE

### 6.1. Pruebas de Caja Blanca.

La prueba de caja blanca es un método de diseño de casos de pruebas que se basa en un estudio minucioso de la lógica y estructura del software. Se realizarán las pruebas de: ruta básica y camino condicional.

#### Prueba de ruta básica:

Permite medir la complejidad lógica de una unidad funcional de código. *Ésta prueba se aplicó al método Postpreinscribir del controlador GestiónDeCursosApiController.*

En la figura 22 se puede observar que:

- El número de nodos (N) es 8
- El número de regiones (R) es 2
- Existen 8 aristas (E)
- Existe 1 nodo predicado (NP)

La complejidad ciclomática, es decir  $V(G)$  se puede calcular de tres formas

$$V(G) = R = 2$$

$$V(G) = E - N + 2 = 8 - 8 + 2 = 2$$

$$V(G) = NP + 1 = 2$$

Las 2 posibles rutas para el método son:

Ruta 1:1-2-3-4

Ruta 2:1-2-4-5-6-7-8

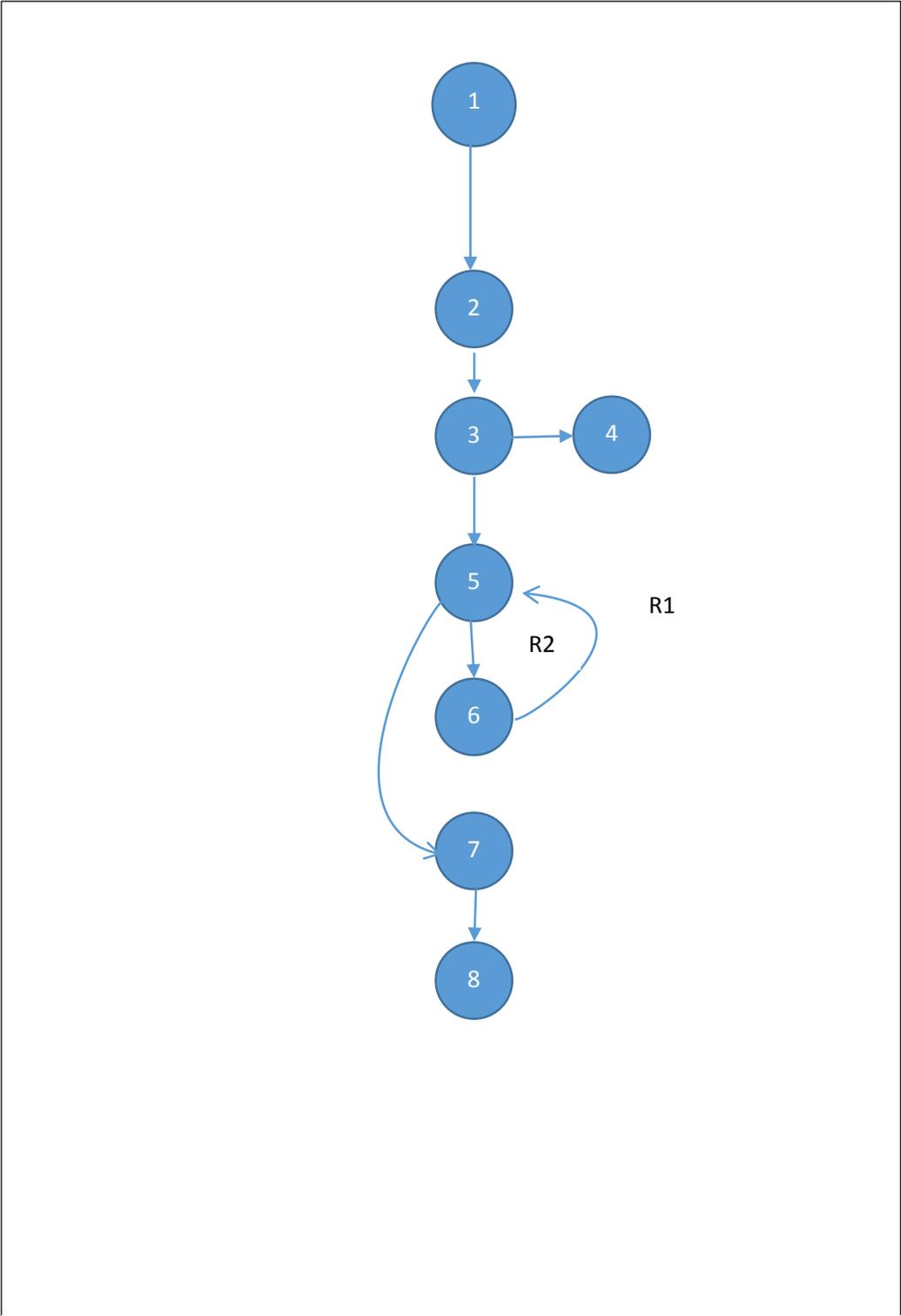


Figura 22: Caminos o rutas de la complejidad ciclomática (fuente: elaboración propia).

Como se puede observar en la siguiente tabla, la complejidad ciclomática de los métodos de este controlador es baja por lo que el esfuerzo para mantener el software es menor.

Miembro	Complejidad ciclomática
datouser	1
GestionDeGruposApiController	1
Getaprobargrupos	2
getcalendario	3
GetCursos	2
GetEspacios	4
GetHorarios	2
GetProfesores	1
obtenerAllhorarios	1

Tabla 64: Complejidad ciclomática de los métodos de GestionDeGruposApiController

La prueba de caja blanca también se aplicó a los controladores para: gestionar alumnos inscritos, gestionar cursos, notas y pagos (véase anexo 14).

### 6.1.1. Prueba de condición en el sistema web.

Esta técnica de caja blanca pertenece a las pruebas de Estructura de Control, y consiste en probar las condiciones lógicas del sistema, en una sección de código. Para aplicar la prueba se toma como base, el método Login del controlador LoginController, y se obtiene el siguiente grafo de flujo:

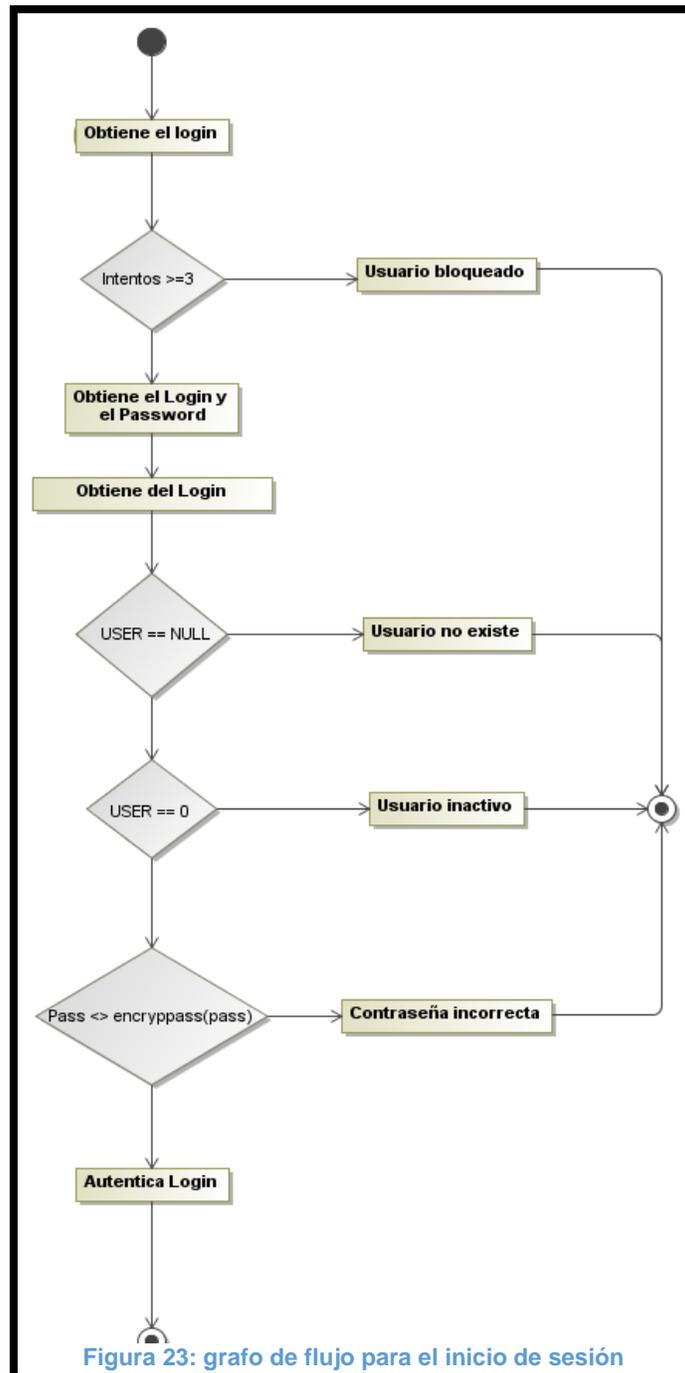


Figura 23: grafo de flujo para el inicio de sesión

Se probaron los errores en:

- El operador relacional, presente en el primer if, donde se prueba si el usuario ha realizado 3 intentos de inicio de sesión o más; bloquea la cuenta del usuario, y manda un mensaje indicando que ha superado el máximo de intentos de autenticación.
- El operador relacional, presente en el último condicional, donde se prueba cuándo la variable *pass* es diferente a la contraseña cifrada en la base de datos y cuándo son iguales.
- La variable lógica de todas las condiciones cambiando los valores de las variables en juego al azar.

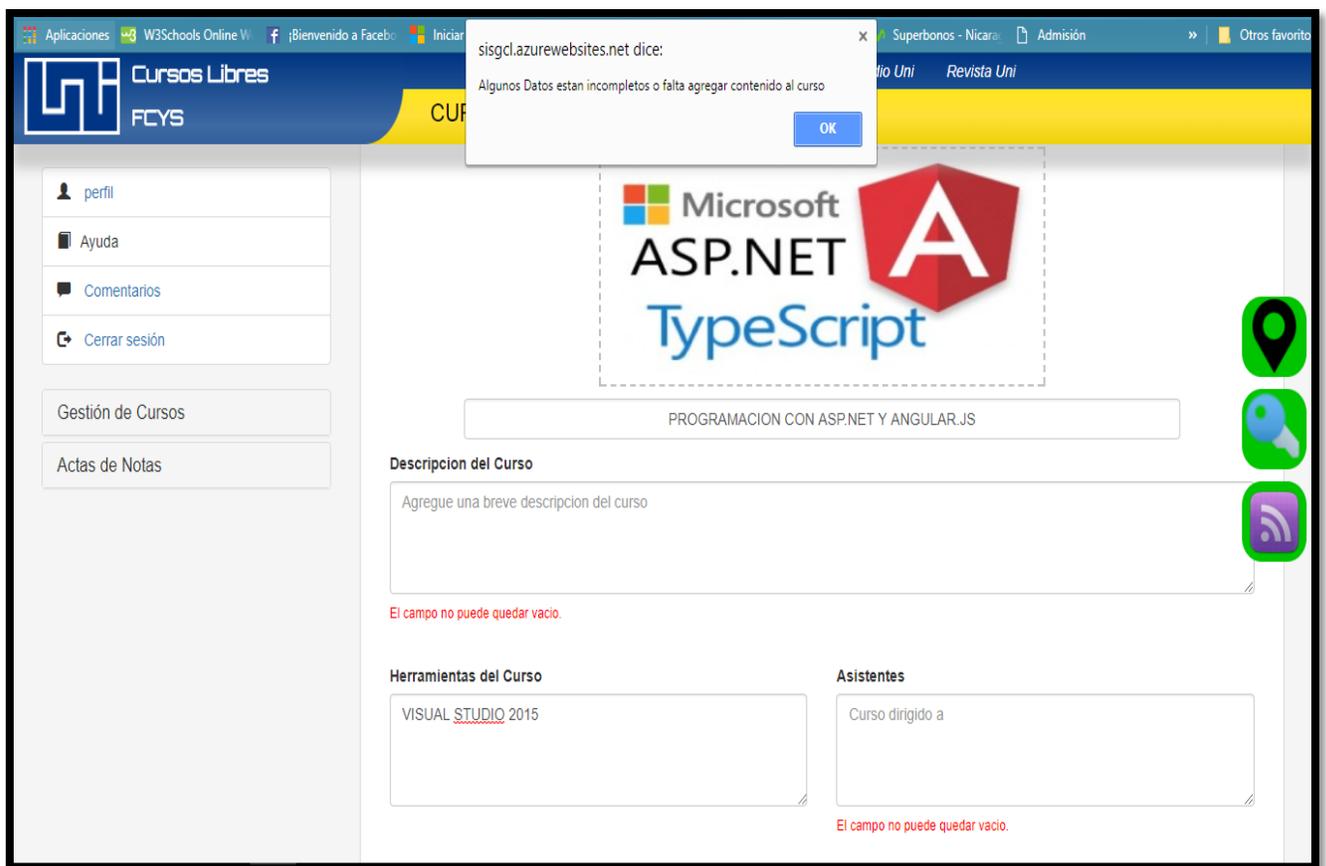
Se obtuvieron resultados satisfactorios, respecto a la prueba de condición, ya que los datos que arrojaba el sistema eran correctos.

## **6.2. PRUEBAS DE CAJA NEGRA.**

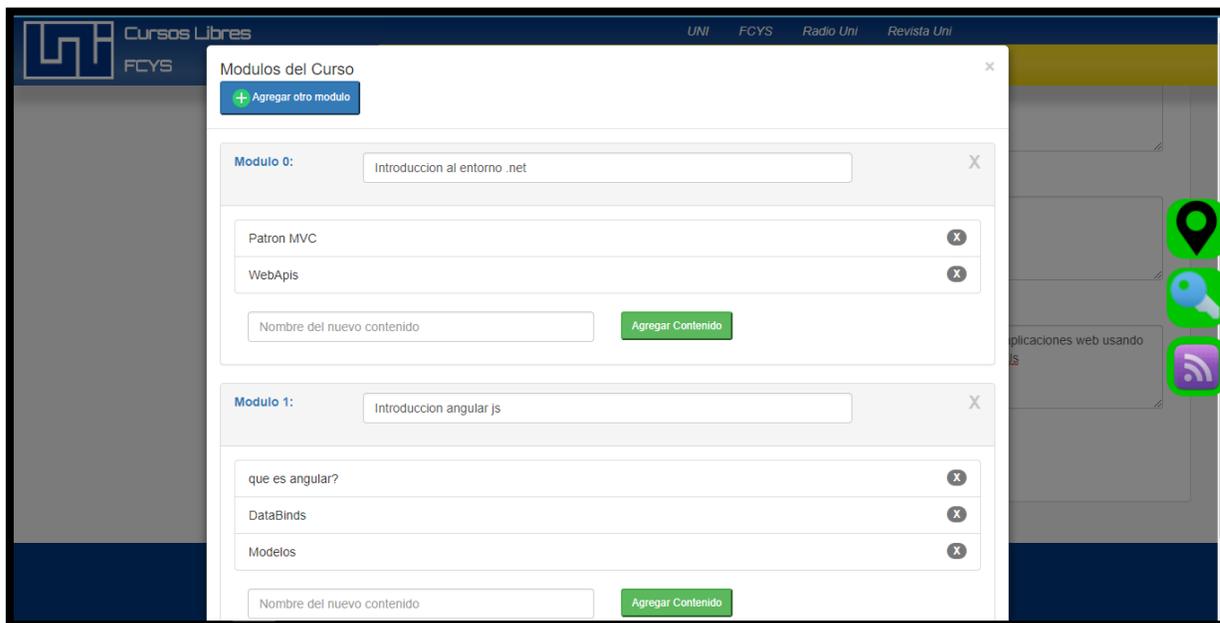
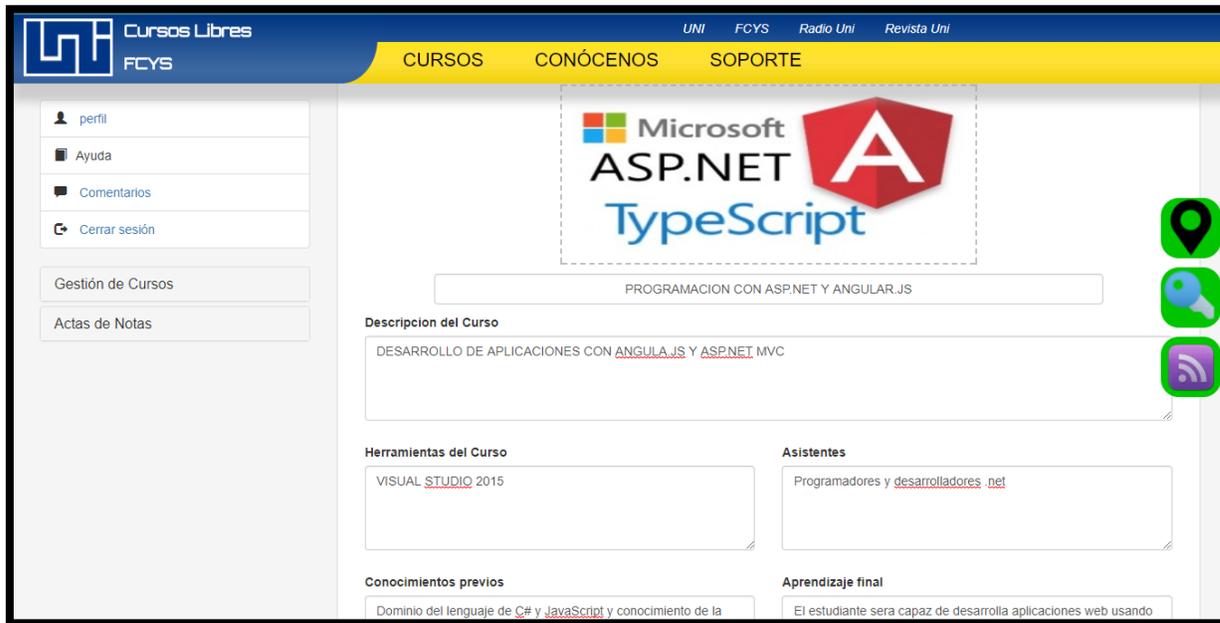
Las pruebas de caja negra es un método de diseño de casos de prueba que se refieren a las pruebas que se realizan a la interfaz del software y su integridad. Las pruebas que se aplicaran al sistema serán las siguientes:

### 6.2.1. Análisis de las validaciones del sistema web

Debido a que los errores se dan generalmente al agregar el contenido en la interfaz de entrada, se realizan este tipo de pruebas, que consiste en seleccionar casos y probar los extremos, para ver cómo se comporta el sistema. Se toma como base, la interfaz de Crear un nuevo curso, a continuación, se evalúa cada campo, estableciendo como requisito que cada campo es requerido y no puede quedar vacío y en el caso el contenido debe de haber al menos 1 tema y un subtema, en este caso, se probará con 2 campos vacíos y despliega correctamente el mensaje de error.



La siguiente prueba corresponde a la creación del curso con su contenido correspondiente para garantizar que este guarde correctamente



Total de cursos disponibles: 9

escriba el nombre del curso a buscar



ver todos los horarios

PROGRAMACION CON ASP.NET Y  
ANGULAR.JS



Ver Detalle del curso

Ver Horarios del curso

Previous 1 2 3 Next



## Conclusiones

- Se recopilaron los requerimientos solicitados por el usuario donde se logró identificar el interés por la seguridad de la información, la posibilidad de preinscripción en línea, registro de pagos y conocer información de los cursos online.
- Se analizó la viabilidad del sistema mediante 5 estudios: en cuanto al aspecto técnico, se eligió contratar el servicio de alojamiento externo de MyAsp.net; desde el punto de vista financiero, el proyecto es rentable; a nivel de la organización el permitirá manejar eficientemente la información; a nivel legal el sistema respetará el marco legal aplicable y de derechos de autor asegurando el acceso y la protección de datos
- Durante el análisis y diseño se elaboraron los diagramas de casos de uso, colaboración, secuencia, actividades, de contenido, estados, presentación y procesos
- El sistema se desarrolló utilizando ASP.NET MVC 4, EntityFramework 5 Angular JS y SQL Server, comprobándose su integridad, su simplicidad lógica y su funcionamiento mediante pruebas de caja blanca y caja negra.

## Recomendaciones

- El personal y docentes de la Unidad de Cursos Libres deberá de ser concientizado de los cambios en las operaciones que producirá el sistema de información.
- Capacitar y entrenar al personal y docentes de los Cursos Libres en el uso del sistema.
- Crear normas y procedimientos para el manejo del sistema y establecer planes de contingencia para salvaguardar la información.

## Referencias bibliográficas

1. Alonzo Velázquez, J. (2015). Lenguaje de Programación: Introducción a C/C++ (IDE). Guanajuato: Universidad de Guanajuato
2. Arellano Rodríguez, M. (2008). Sistemas de información: ¿adecuación a los cambios tecnológicos o herramienta de gestión? *Revista de Ciencias Sociales*, 14(3), 528-545
3. Cáceres, P., & Marcos, E. (2002). Procesos Ágiles para el Desarrollo de Aplicaciones Web. Madrid, España: Universidad Rey Juan Carlos recuperado de <http://www.dlsi.ua.es/~jaime/webe/articulos/s112.pdf>, el 10 de octubre de 2015
4. Casillas Santillán, L. A., Costal Costa, D., Gibert Ginestà, M., Camps Paré, R., Pérez Mora, O., & Martín Escofet, C. (2007). *Bases de datos: Software libre*. Barcelona, España: Universidad Abierta de Cataluña
5. Quirola, G. & Coronel, F. (2013). ANÁLISIS, DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA WEB PARA EL CONTROL DE UN TALLER TÉCNICO AUTOMOTRIZ EN PLATAFORMA PHP - MYSQL UTILIZANDO METODOLOGÍA WEB UWE PARA LA EMPRESA METROAUTOCERFRANCIA LTDA. (Tesis de Ingeniería). Universidad de las Fuerzas Armadas - ESPE. Sangolquí, Ecuador
6. Cobo, A. (2005). *PHP y MySQL: Tecnología para el desarrollo de aplicaciones web*. España: Ediciones Díaz de Santos
7. Essink, L., & Visscher, A. (1987). The Design and Impact of Management information system In Educational Organization en *Higher Education and New Technologies*. (pp. 367-387). United Kingdom: PERGAMON PRESS
8. Ferrer, Juan. (2012). Conceptos generales de la arquitectura de aplicaciones web en *Implantación de aplicaciones web* (pp. 17-27). Madrid, España: Ra-Ma Editorial, S.A
9. Isakowitz, T., Bieber, M. & Vitali, F. (1998). Web information systems. *Communications of the ACM*, 41(7), 78-80
10. Laudon, K. C. & Laudon, J. P. (2012). *Sistemas de Información Gerencial*. Naucalpan de Juárez, Estado de México: Pearson Educación de México
11. Loa Fragoso, C. Y., (2011). *Sistema informático para la automatización de los servicios diversos del sistema de gestión de la calidad en el PUEG de la UNAM*. (Tesis de Ingeniería). Universidad Nacional Autónoma de México. México, D.F.
12. Mateu, C. (2004). *Desarrollo de Aplicaciones Web*. Barcelona: Universidad Abierta de Cataluña
13. Microsoft Developer Network. (2007). *Información General sobre ASP.NET*. Recuperado de <https://msdn.microsoft.com/es-es/library/4w3ex9c2%28v=vs.100%29.aspx>
14. Microsoft Developer Network. (2015). *Microsoft SQL Server*. Recuperado de <https://msdn.microsoft.com/es-es/library/bb545450.aspx>
15. Microsoft Developer Network. (2015). *Visual C#*. Recuperado de

- <https://msdn.microsoft.com/es-es/library/kx37x362.aspx>
16. Moreno, J. (2011). El Blog del Software de Gestión en la PYME [Mensaje de Blog]. Recuperado a partir de <http://www.prismasoftwaregestion.com/blog/software-web-vs-software-no-web-o-de-escritorio-22/>
  17. Mozilla Developer Network. (2015). *Introducción a JavaScript*. Recuperado de <https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Introducci%C3%B3n>
  18. Murugesan, S., Deshpande, Y., Hansen, S., & Ginige, A. (2001). Web engineering: A new discipline for development of web-based systems. En *Web Engineering* (pp. 3-13). Springer Berlin Heidelberg.
  19. O'Brien, J. A. & Marakas, M. G. (2006). *Sistemas de Información Gerencial*. (7a ed.). D.F, México: McGRAW-HILL / INTERAMERICANA
  20. Orjuela Duarte, A., & Rojas C., M. (2008). Las metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo en *Avances en Sistemas e Informática*, 5(2), 159-171.
  21. Pressman, R.S. & Lowe, D. (2009). *Web Engineering: A practitioner approach*. 1221 Avenue of the Americas, New York, NY 10020, United States: McGRAW-HILL
  22. Pressman, R.S. (2010). *Ingeniería del software: Un enfoque práctico*. (7a ed.). D.F., México: McGRAW-HILL/INTERAMERICANA EDITORES, S.A. DE C.V.
  23. Puente, O.M. (n.d.). Programación Web Instituto Tecnológico de Matehuala [Mensaje de Blog]. Recuperado de <https://programacionwebisc.wordpress.com/2-3-lenguajes-de-programacion-del-lado-del-servidor/>
  24. Quiroga, A. (23 de Marzo de 2015). Proyecto de Grado Ingeniería de Sistemas [Mensaje de Blog]. Recuperado a partir de <http://proyectogradoingenieriasistemas.blogspot.com/2015/03/metodologia-uwe-uml-uml-based-web.html>
  25. Rational Software. (2001). Rational Unified Process Best Practices for Software Development Teams. Recuperado de [https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251\\_bestpractices\\_TP026B.pdf](https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf)
  26. Revista TIC NEWS Julio (2014). *Mejoras de la Publicidad web con I.A.* Managua, Dpto. de Nicaragua: TIC News
  27. Romero Galindo, R. (2012). *Análisis, Diseño e implementación de un sistema de información aplicado a la gestión Educativa en centros de educación especial*. (Tesis de Ingeniería). Pontificia Universidad Católica de Perú. Lima, Perú.
  28. Van Lancker, L. (2013). *HTML5 y CSS3: Domine los estándares de las aplicaciones Web*. (2ª ed.). España: Ediciones ENI.
  29. Vega, E. B. (12 de Julio de 2015). *Los sistemas de información y su importancia para las organizaciones y empresas*. Recuperado de <http://www.monografias.com/trabajos24/tics-emprendas/tics-emprendas.shtml>
  30. Wikiversidad. (2015). *Metodologías pesadas de desarrollo software*. Recuperado de [https://es.wikiversity.org/wiki/Metodolog%C3%ADas\\_pesadas\\_de\\_desarrollo\\_software](https://es.wikiversity.org/wiki/Metodolog%C3%ADas_pesadas_de_desarrollo_software), el 20 de septiembre de 2015

