

Universidad Central de Venezuela

Facultad de Ciencias  
Escuela de Computación



**Trabajo Especial de Grado:  
Desarrollo de Un Módulo Web  
Para la Gestión de Planes de Estudio en el  
Sistema de Gestión Académica CONEST**

José T. Hernández C.  
V-16.179.441

Tutora:  
Prof. Jossie Zambrano  
Octubre, 2012

## Resumen

La División de Control de Estudios es la dependencia administrativa de la Facultad de Ciencias que se encarga de coordinar y supervisar los procesos relacionados con la inscripción, control de notas y graduación de las licenciaturas de pregrado.

La División presta algunos de sus servicios a través del Sistema de Gestión Académica de Control de Estudios (CONEST) facilitando las operaciones de los procesos administrativos de la Facultad. Sin embargo, el sistema CONEST no cuenta con una aplicación que permita gestionar los planes de estudio de las distintas licenciaturas que ofrece la Facultad.

Este Trabajo Especial de Grado tiene como objetivo desarrollar un módulo web del sistema CONEST que permita gestionar los planes de estudio de las licenciaturas de la Facultad de Ciencias. Además de automatizar parte del trabajo de verificación de requisitos curriculares realizado por el Departamento de Grado de la División de Control de Estudios.

El método ágil de desarrollo de software implementado para estructurar, planificar y controlar el proceso de desarrollo del módulo web para la gestión de planes de estudio en el sistema CONEST es una adaptación de la Programación Extrema.

Palabras Clave: Control de Estudios, Gestión de Planes de Estudio, Verificación de Requisitos Curriculares, Aplicación Web, Ruby on Rails.

## Índice General

Resumen.....	2
Introducción .....	7
Capítulo 1. Planes de Estudio de la Facultad de Ciencias.....	9
1.1 Licenciatura en Biología.....	11
1.2 Licenciatura en Computación .....	15
1.3 Licenciatura en Física.....	17
1.4 Licenciatura en Geoquímica.....	21
1.5 Licenciatura en Matemática .....	22
1.6 Licenciatura en Química .....	26
Capítulo 2. Tecnologías Web .....	30
2.1 HTML5 .....	31
2.2 CSS3 .....	32
2.3 jQuery .....	33
2.4 Ruby on Rails.....	34
2.5 Haml .....	35
2.6 Sass.....	36
2.7 MySQL.....	36
Capítulo 3. Método Ágil de Desarrollo de Software .....	37
3.1 Programación Extrema (XP) .....	38
3.2 Adaptación del Método Ágil XP .....	40
3.2.1 Planificación .....	40
3.2.2 Diseño .....	41
3.2.3 Codificación.....	41
3.2.4 Pruebas .....	42
Capítulo 4. Marco Aplicativo .....	44
4.1 Iteración 0.....	44
4.1.1 Planificación .....	44
4.1.2 Diseño .....	50
4.1.3 Codificación.....	51
4.1.4 Pruebas .....	51

4.2 Iteración 1 .....	52
4.2.1 Planificación .....	52
4.2.2 Diseño .....	52
4.2.3 Codificación .....	61
4.2.4 Pruebas .....	72
4.3 Iteración 2 .....	76
4.3.1 Planificación .....	76
4.3.2 Diseño .....	77
4.3.3 Codificación .....	80
4.3.4 Pruebas .....	87
Conclusiones.....	90
Recomendaciones.....	92
Referencias Bibliográficas .....	93
Anexos.....	94

## Índice de Figuras

<b>Figura 1.</b> Plan de Estudio de Biología.....	13
<b>Figura 2.</b> Componente Docente de Biología. ....	15
<b>Figura 3.</b> Plan de Estudio de Computación. ....	17
<b>Figura 4.</b> Plan de Estudio de Física.....	19
<b>Figura 5.</b> Componente Docente de Física. ....	21
<b>Figura 6.</b> Plan de Estudio de Geoquímica.....	22
<b>Figura 7.</b> Plan de Estudio de Matemática. ....	24
<b>Figura 8.</b> Componente Docente de Matemática. ....	26
<b>Figura 9.</b> Plan de Estudio de Química. ....	28
<b>Figura 10.</b> Componente Docente de Química. ....	29
<b>Figura 11.</b> Arquitectura Cliente-Servidor. ....	31
<b>Figura 12.</b> Ciclo de Vida Para el Desarrollo Ágil de Software. ....	37
<b>Figura 13.</b> Formato de las Tarjetas de Historias. ....	40
<b>Figura 14.</b> Diseño General del Sistema CONEST.....	50
<b>Figura 15.</b> Listado de Planes de Estudio.....	53
<b>Figura 16.</b> Editar Planes de Estudio. ....	54
<b>Figura 17.</b> Tabla distribucion_requisito_plan.....	55
<b>Figura 18.</b> Listado de Requisitos Curriculares Distribuidos por Semestre. ....	55
<b>Figura 19.</b> Listado de Asignaturas.....	57
<b>Figura 20.</b> Editar Asignaturas. ....	58
<b>Figura 21.</b> Tabla materia_en_plan.....	59
<b>Figura 22.</b> Campos Obligatorios. ....	60
<b>Figura 23.</b> Diálogos Emergentes. ....	60
<b>Figura 24.</b> Mensajes de Confirmación. ....	61
<b>Figura 25.</b> Mensajes de Error. ....	61
<b>Figura 26.</b> Modelo plan.rb.....	61
<b>Figura 27.</b> Vista plan_estudio/index.html.haml. ....	62
<b>Figura 28.</b> Vista Parcial plan_estudio/_formulario.html.haml. ....	63
<b>Figura 29.</b> Método plan_estudio/guardar.....	64
<b>Figura 30.</b> Vista Parcial plan_estudio/_notificacion.html.haml. ....	64

<b>Figura 31.</b> Método plan_estudio/actualizar.....	64
<b>Figura 32.</b> Método plan_estudio/eliminar.....	65
<b>Figura 33.</b> Método plan_estudio/editar. ....	65
<b>Figura 34.</b> Vista plan_estudio/editar.html.haml. ....	66
<b>Figura 35.</b> Script plan_estudio.js.coffee.....	66
<b>Figura 36.</b> Vista Parcial plan_estudio/_distribucion_requisitos_curriculares.html.haml.....	67
<b>Figura 37.</b> Método plan_estudio/crear_requisito_curricular. ....	68
<b>Figura 38.</b> Modelo requisito_graduacion_plan.rb.....	68
<b>Figura 39.</b> Método asignatura/autocomplete.....	69
<b>Figura 40.</b> Script asignatura.js.coffee (Autocomplete). ....	70
<b>Figura 41.</b> Método asignatura/crear_prelacion. ....	71
<b>Figura 42.</b> Método asignatura/buscar_planes_estudio_asignatura.....	71
<b>Figura 43.</b> Script asignatura.js.coffee (Listas Desplegables).....	72
<b>Figura 44.</b> Verificación de Requisitos Curriculares. ....	77
<b>Figura 45.</b> Asignaturas Aprobadas/Asignaturas No Aprobadas. ....	78
<b>Figura 46.</b> Tabla historial_academico. ....	78
<b>Figura 47.</b> Resultado de la Verificación de Requisitos Curriculares.....	79
<b>Figura 48.</b> Tablas opcion, opcion_plan y requisito_graduacion_opcion_plan.....	79
<b>Figura 49.</b> Verificación de la Opción Profesional.....	80
<b>Figura 50.</b> Método plan_estudio/verificar_requisitos (Requisitos Curriculares). ....	81
<b>Figura 51.</b> Vista Parcial _verificar_requisitos_academicos.html.erb.....	84
<b>Figura 52.</b> Método plan_estudio/verificar_requisitos (Opción Profesional). ....	85
<b>Figura 53.</b> Vista Parcial _resumen_opcion_profesional.html.erb. ....	86

## Introducción

La Universidad Central de Venezuela (UCV) es la institución de educación superior más antigua de Venezuela, fundada en 1721. En la actualidad, mantiene una matrícula estudiantil que ronda los 54.222 estudiantes de pregrado y 8.317 estudiantes de postgrado, con 4.790 profesores y 4.514 empleados administrativos, distribuidos en 9 Facultades en Caracas, 2 Facultades en Maracay, 5 Núcleos de Estudios supervisados y 12 Estaciones Experimentales en diferentes regiones del país.

La UCV ofrece aproximadamente 351 carreras universitarias entre pregrado y postgrado. Las carreras que se imparten en esta casa de estudios están conformadas por un conjunto de asignaturas y requisitos curriculares que se deben satisfacer para obtener un título universitario, lo que se conoce como un plan de estudio.

La Facultad de Ciencias de la UCV está integrada por 5 Escuelas, donde se forman licenciados en las siguientes carreras: Biología, Computación, Física, Geoquímica, Matemática y Química.

La División de Control de Estudios (DCE) es la dependencia administrativa de la Facultad que se encarga de coordinar y supervisar los procesos relacionados con la inscripción, control de notas y graduación de las licenciaturas de pregrado.

El Departamento de Grado de la DCE utiliza los planes de estudio de las distintas licenciaturas cada semestre para verificar que los estudiantes que inscribieron Trabajo Especial de Grado cumplen con todos los requisitos curriculares y las normativas académicas que deben satisfacer para obtener un título universitario en la licenciatura en que se encuentran inscritos.

La Facultad de Ciencias ha desarrollado el Sistema de Gestión Académica de Control de Estudios (CONEST) que presta servicio a los estudiantes, profesores y empleados administrativos, facilitando las operaciones de los procesos administrativos de la Facultad.

Los trabajadores del Departamento de Grado actualmente realizan todas sus labores con un soporte mínimo del sistema CONEST, verificando de manera manual los requisitos curriculares de cada estudiante con lápices de colores y distintos documentos impresos. Esto representa un mayor esfuerzo humano y aumenta la probabilidad de cometer errores.

Es por ello que el objetivo de este Trabajo Especial de Grado es desarrollar un módulo web del sistema CONEST basado en tecnologías web, que automatice el trabajo de verificación de requisitos curriculares del proceso de graduación que se realiza en la DCE, así como la gestión de los planes de estudio de las distintas licenciaturas que ofrece la Facultad de Ciencias.

Para lograr dicho desarrollo, se utiliza el método ágil de desarrollo de software llamado Programación Extrema (XP), donde se realizan tres iteraciones implementando las fases propuestas en el método. En la primera iteración se analiza de manera general el módulo web, en

la segunda iteración se desarrolla la gestión de los planes de estudio de las licenciaturas de la Facultad de Ciencias, y en la tercera iteración de desarrolla la verificación de los requisitos curriculares de los estudiantes regulares de pregrado. Este trabajo de investigación está estructurado de la siguiente manera:

**Capítulo 1. Planes de Estudio de la Facultad de Ciencias:** En este capítulo se describen los planes de estudio y se presentan los resultados del análisis realizado a todos los planes de estudio de las licenciaturas que ofrece la Facultad de Ciencias.

**Capítulo 2. Tecnologías Web:** En este capítulo se describen las tecnologías web implementadas en el desarrollo del módulo web para la gestión de planes de estudio en el sistema CONEST.

**Capítulo 3. Método Ágil de Desarrollo de Software:** En este capítulo se describen las metodologías ágiles de desarrollo de software, la Programación Extrema (XP), y la adaptación del método ágil XP implementado en el proceso de desarrollo del módulo web para la gestión de planes de estudio en el sistema CONEST.

**Capítulo 4. Marco Aplicativo:** En este capítulo se presentan las iteraciones realizadas durante el desarrollo del módulo web para la gestión de planes de estudio en el sistema CONEST bajo el método ágil XP, especificando la planificación, el diseño, la codificación y las pruebas de las tarjetas de historias correspondientes.

Finalmente, se presentan las conclusiones, las recomendaciones y las referencias bibliográficas.

## **Capítulo 1**

### **Planes de Estudio de la Facultad de Ciencias**

Un plan de estudio (Teruel, 2000) es de manera general una descripción de los requisitos que se necesitan satisfacer para obtener un título universitario en un área del saber. Estos requisitos incluyen: aprobar el conjunto de asignaturas que la universidad considera obligatorias, el número de créditos que el estudiante debe aprobar en asignaturas electivas, y los requisitos o prelación que aplican entre las asignaturas.

En la Facultad de Ciencias de la UCV se imparten 6 licenciaturas: Biología, Computación, Física, Geoquímica, Matemática y Química. Los planes de estudio de las licenciaturas de la Facultad son gestionados por la DCE, la cual es una dependencia administrativa que se encarga de coordinar y supervisar los procesos relacionados con la inscripción, control de notas y graduación de los estudiantes regulares de pregrado.

La DCE requiere conocer la estructura y los reglamentos de cada uno de los planes de estudio para revisar, validar y procesar decisiones que en materia académica incidan en sus procesos administrativos para su debida implementación. Los planes de estudio son vitales en los procesos de inscripción y graduación, la DCE integra los recursos tanto humanos como tecnológicos para cerciorarse de que cada estudiante cumpla con todos los requisitos y las prelación para cursar las asignaturas, y que cada egresado haya cumplido y aprobado con todos los requisitos y normativas académicas para optar por un título universitario en las licenciaturas que ofrece la Facultad de Ciencias.

Los planes de estudio de las licenciaturas de la Facultad de Ciencias están conformados por un conjunto de asignaturas y reglas. Una asignatura es una unidad académica que forma parte de una carrera, por lo general dura un período académico de entre 15 y 16 semanas (excepto en el curso intensivo que tiene una duración de 7 u 8 semanas), está dirigida por uno o más docentes, cuenta con una lista fija de estudiantes, y este obtiene una nota y unidades de créditos por cada asignatura aprobada. Sus atributos comunes incluyen:

**Nombre:** Es el nombre completo de la asignatura por el cual es conocida. Ejemplos de nombres de asignaturas son: Sistemas Operativos, Comunicación de Datos, Lenguajes de Programación, etc.

**Abreviación:** Es el nombre corto de la asignatura, se utiliza para mantener la compatibilidad con otros sistemas que están limitados al uso de 15 caracteres. Ejemplos de abreviaciones de asignaturas son: Sist. Operativos, Comun. de Datos, etc.

**Código:** Es una representación numérica de dígitos que permite identificar la asignatura, los primeros 2 dígitos representan la Facultad, los siguientes 2 dígitos representan la Escuela, y los últimos 4 dígitos representan la asignatura. La Facultad de Ciencias está representada por el

número 03. Las Escuelas están representadas de la siguiente manera: 01 (Escuela de Biología), 02 (Escuela de Física), 03 (Escuela de Química), 04 (Escuela de Computación), 05 (Escuela de Matemática), y 06 (Escuela de Geoquímica). Ejemplos de códigos de asignaturas son: 03046004, 03046003, 03046204, etc.

**Unidades de Crédito (UC):** Las unidades de crédito se determinan en base al número de horas de clase semanales que un estudiante recibe por asignatura. Estas horas incluyen:

- **Horas Teóricas (HT):** 1 HT equivale a 1 UC.
- **Horas de Práctica (HP):** 2 HP equivalen a 1 UC.
- **Horas de Laboratorio (HL):** 3 HL equivalen a 1 UC.

**Calificación:** La nota final de las asignaturas se califica con un número comprendido entre 0 y 20. En la Facultad de Ciencias se utiliza otro método para calificar asignaturas y requisitos curriculares con notaciones alfabéticas: A (Aprobado), AP (Aplazado), EQ (Equivalencia), y RET (Retirado).

**Tipo:** Las asignaturas pueden ser cursadas según las siguientes modalidades: obligatorias, electivas, optativas.

Las asignaturas **obligatorias** son aquellas que todos los estudiantes están obligados a cursar por su importancia en la formación y el desarrollo educativo del estudiante. En general, están ubicadas en la primera mitad de la carrera.

Las asignaturas **electivas** son aquellas que el estudiante puede cursar para profundizar su formación en un determinado campo profesional, pueden ser cursadas como parte de una opción según lo establecido en el reglamento de la licenciatura. Se ubican en los últimos semestres de la carrera.

Las asignaturas **optativas** son aquellas que el estudiante puede cursar para ampliar su formación general, más allá de la pura formación profesional, y no son un requisito en todas las licenciaturas de la Facultad. En general, están ubicadas en los últimos semestres de la carrera. Las asignaturas del Programa de Cooperación Interfacultades (PCI) son otro tipo de asignaturas optativas que el estudiante puede cursar en otras Facultades de la Universidad.

Los planes de estudio de las licenciaturas de la Facultad de Ciencias agrupan las asignaturas por semestres, algunas de estas asignaturas guardan una relación de prelación, en función de conocimientos previos que el estudiante debe manejar para cursar las asignaturas. El objetivo del sistema de prelación es facilitar el avance del estudiante en la carrera ya que permite integrar y secuenciar conocimientos y habilidades entre asignaturas.

Los planes de estudio podrían cambiar con el transcurso del tiempo, a aquellos estudiantes que se encuentran en periodo de transición de un plan de estudio a otro es necesario aplicarles la tabla

de convalidaciones y equivalencias, siguiendo las normas acordadas por cada Escuela. Otro caso particular a tomar en cuenta es el de los estudiantes que se están reincorporando o cambiando de carrera, ya que se debe verificar las equivalencias de cada una de las asignaturas que apliquen según el caso.

La opción profesional es un conjunto de asignaturas que forman al estudiante en un determinado campo profesional. Las opciones buscan profundizar la formación del estudiante en el área de la carrera de su preferencia, estas opciones son definidas por las Escuelas, soportadas por los centros de investigación y administradas por el o los departamentos. Para optar por una opción el estudiante debe aprobar un número mínimo de asignaturas, el Seminario y el Trabajo Especial de Grado de la opción correspondiente. Además, el estudiante puede cursar asignaturas de cualquier opción bajo la modalidad de electiva, siempre y cuando cumpla con las prelación exigidas.

Es importante resaltar que cada licenciatura tiene su propio reglamento, por lo que cada plan de estudio exige diferentes requisitos para obtener el título de la carrera. Los estudios de licenciatura en la Facultad de Ciencias tienen una duración de 10 semestres, y 19 unidades de crédito por semestre (en promedio).

Además, el componente docente es un convenio que existe con la Facultad de Humanidades y Educación denominado Convenio Cooperativo de Formación Docente que les permite a los estudiantes de las licenciaturas de Biología, Física, Matemática y Química optar por un título universitario de Licenciatura en Educación con una mención científica. El estudiante debe aprobar las asignaturas correspondientes al área científica en la Facultad de Ciencias y las asignaturas correspondientes al área humanística en la Escuela de Educación.

A continuación se describen los planes de estudio de las licenciaturas que ofrece la Facultad de Ciencias.

### **1.1 Licenciatura en Biología**

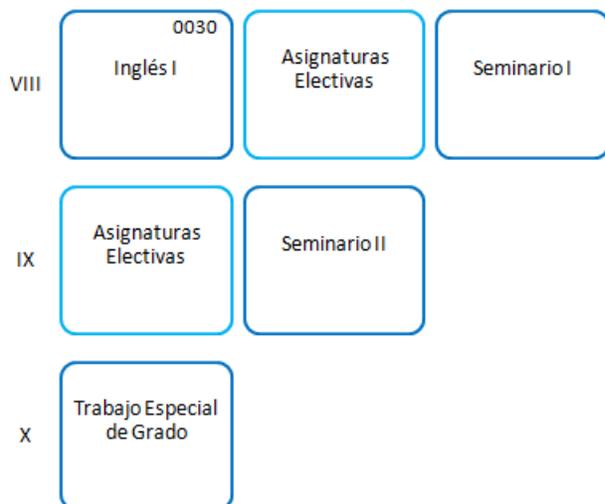
El plan de estudio de la Licenciatura en Biología se encuentra vigente desde el año 1968. El Licenciado en Biología se encarga de estudiar en el laboratorio y en condiciones naturales el origen, el desarrollo, la estructura y otros aspectos fundamentales de los organismos vivos.

El biólogo estudia e investiga diferentes aspectos de las células y los seres vivos. Diseña los experimentos, explora e investiga y aplica los resultados en la solución de problemas en las áreas de biología celular, botánica, ecología, tecnología de alimentos y zoología.

El plan de estudio de Biología está estructurado en dos partes. La primera parte corresponde al ciclo básico de la carrera, su objetivo es brindar una formación básica de los grandes principios de la biología. El ciclo básico tiene una duración de 7 semestres, está conformado por 34 asignaturas obligatorias para un total de 121 unidades de crédito.

La segunda parte corresponde al ciclo de opción de la carrera, su objetivo es profundizar la formación del estudiante en un determinado campo profesional dentro del campo de la biología. El ciclo de opción dispone de 3 semestres (VIII, IX, X), para la aprobación de las asignaturas correspondientes a las opciones, con 60 unidades de crédito distribuidas de la siguiente manera: 25 unidades de crédito en asignaturas obligatorias (Seminario I, Seminario II y Trabajo Especial de Grado), y 35 unidades de crédito en asignaturas electivas. Además, el estudiante deberá cursar Inglés I o presentar un examen de suficiencia con un valor de 4 unidades de crédito, adicionales a las 60 señaladas anteriormente. La Figura 1 muestra el plan de estudio de Biología y las asignaturas que lo integran.

I	8206 Matemáticas General I	2103 Física General I	3401 Principios de Química I			
II	8207 Matemáticas General II (8206)	2104 Física General II (2103, 8206)	3404 Lab. Principios de Química (3401, 3403)	3403 Principios de Química II (3401, 8206)	1902 Principios de Biología (3401)	
III	8208 Matemáticas General III (8207)	3425 Principios de Fisicoquímica (3403, 8207, 2104)	1115 Biología Vegetal (1902)	1116 Lab. Biología Vegetal (1902)		
IV	1926 Bioestadística (1902, 8207)	3450 Química Orgánica (3403, 3404)	3451 Lab. Química Orgánica (3403, 3404)	1818 Biología Animal (1902)	1819 Lab. Biología Animal (1902)	
V	1031 Bioquímica General (3450, 3451)	1032 Lab. Bioquímica General (3450, 3451)	2440 Física General III (2104)	1844 Genética General (1926, 1031, 1032)	1845 Lab. Genética General (1926, 1031, 1032)	
VI	1416 Fisiología I (1515, 1516, 1031, 1032)	1417 Lab. Fisiología I (1515, 1516, 1031, 1032)	1436 Biología Celular (1031, 1032, 3425)	1437 Lab. Biología Celular (1031, 1032, 3425)	1313 Ecología I (1115, 1116, 1818, 1819, 1926)	1314 Lab. Ecología I (1115, 1116, 1818, 1919, 1926)
VII	1418 Fisiología II (1436, 1437, 1818, 1819)	1419 Lab. Fisiología II (1436, 1437, 1818, 1819)	1850 Evolución (1844, 1845, 1926)	1315 Ecología II (1313, 1314)	1316 Lab. Ecología II (1313, 1314)	1927 Filosofía de la Ciencia



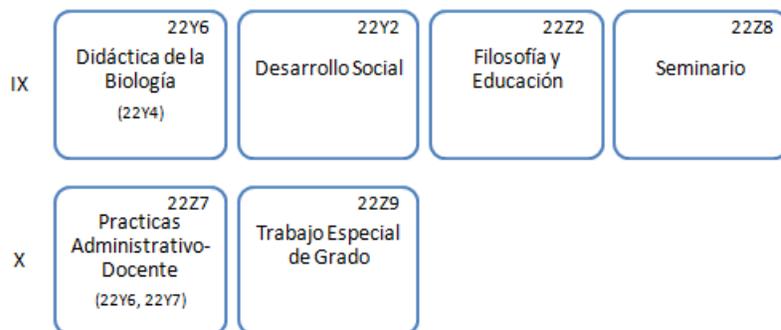
**Figura 1.** Plan de Estudio de Biología.

La Escuela de Biología ofrece las siguientes opciones profesionales: Botánica, Biología Celular, Tecnología de Alimentos, Ecología y Zoología. Para optar por una opción profesional el estudiante debe aprobar un mínimo de 18 unidades de crédito en asignaturas correspondientes a la opción de su preferencia, excepto en Tecnología de Alimentos donde es necesario aprobar todas las asignaturas de la opción.

Finalmente, para obtener el título de Licenciado en Biología el estudiante debe aprobar al menos 185 unidades de crédito.

Para los estudiantes que deseen ingresar al Programa Cooperativo de Formación Docente, deben tener aprobadas todas las asignaturas hasta el 5to semestre inclusive. El estudiante debe cursar 119 unidades de crédito en asignaturas dictadas por la Facultad de Ciencias y 52 unidades de crédito en asignaturas dictadas por la Escuela de Educación para optar por el componente docente. La Figura 2 muestra el componente docente de Biología.

I	8206 Matemática I	2103 Física General I	3401 Principios de Química I			
II	1902 Principios de Biología	8207 Matemática II (8206)	2104 Física General II (2103, 8206)	3403 Principios de Química II (8206, 3401)	3404 Lab. Principios de Química (3401, 3403)	
III	1115 Biología Vegetal (1902)	1116 Lab. Biología Vegetal (1902)	3425 Principios de Físicoquímica (8207, 3401)			
IV	3450 Química Orgánica (3403, 3404)	3451 Lab. Química Orgánica (3403, 3404)	1818 Biología Animal (1902)	1819 Lab. Biología Animal (1902)	1926 Bioestadística (8207)	
V	1031 Bioquímica General (3450, 3451)	1032 Lab. Bioquímica General (3450, 3451)	1844 Genética General (1926)	1845 Lab. Genética General (1926)	0030 Idiomas I	
VI	1416 Fisiología I (1115, 1116, 1031, 1032)	1417 Lab. Fisiología I (1115, 1116, 1031, 1032)	1436 Biología Celular (1031, 1032)	1437 Lab. Biología Celular (1031, 1032)	1313 Ecología I (1115, 1116, 1926, 1818, 1919)	1314 Lab. Ecología I (1115, 1116, 1926, 1818, 1919)
	22Y1 Introducción a la Pedagogía	0031 Idiomas II (0030)				
VII	22Y4 Didáctica General	1418 Fisiología II (1436, 1437)	1419 Lab. Fisiología II (1436, 1437)	1315 Ecología II (1313, 1314)	1316 Lab. Ecología II (1313, 1314)	1850 Evolución (1844, 1845)
VIII	22Y3 Psicopedagogía	22Y8 Evaluación	22Y7 Administración Escolar	1927 Filosofía de la Ciencia		



**Figura 2.** Componente Docente de Biología.

## 1.2 Licenciatura en Computación

El plan de estudio de la Licenciatura en Computación se encuentra vigente desde el año 2004. El Licenciado en Computación ejerce sus tareas profesionales dentro de tres amplias funciones, que se señalan a continuación:

- El desarrollo de sistemas y aplicaciones computacionales, donde debe estar preparado para resolver problemas que se presentan en las grandes, medianas, pequeñas organizaciones y a los individuos, proporcionándoles soluciones computacionales y en general sistemas que integren elementos computacionales.
- Funciones de gestión de plataforma computacionales y servicios informáticos, a través de esta función ejerce su rol de facilitador de la tecnología computacional. Este puesto de trabajo implica gestionar y administrar recursos informáticos para facilitar y soportar procesos y actividades en organizaciones, instituciones y directamente por usuarios finales.
- Investigación e innovación, el profesional cumple esta función a partir de la introducción de la tecnología computacional en las diversas actividades de la sociedad, esto es posible por la búsqueda innovadora y participación en la investigación pura y aplicada, la misma es profundizada, a partir de los estudios de cuarto nivel.

El plan de estudio de Computación está estructurado en dos partes. La primera parte corresponde al ciclo básico de la carrera, su objetivo es brindar una formación básica de los principios de la computación. El ciclo básico tiene una duración de 5 semestres, está conformado por 21 asignaturas obligatorias para un total de 107 unidades de crédito.

La segunda parte corresponde al ciclo de opción de la carrera, su objetivo es profundizar la formación del estudiante en un determinado campo profesional dentro del campo de la computación. El ciclo de opción dispone de 5 semestres (VI, VII, VIII, IX, X), para la aprobación de las asignaturas correspondientes a las opciones, con 78 unidades de crédito distribuidas de la siguiente manera: 25 unidades de crédito en asignaturas obligatorias (Pasantía, Seminario y Trabajo Especial de Grado), 43 unidades de crédito en asignaturas electivas, y 10 unidades de

crédito en asignaturas obligatorias optativas. Además, el estudiante deberá cursar 8 unidades de crédito en asignaturas optativas, adicionales a las 78 señaladas anteriormente. La Figura 3 muestra el plan de estudio de Computación.

I	6201 Algoritmos y Programación	6301 Introducción a la Informática	8206 Matemáticas I	6106 Matemáticas Discretas I	
II	6202 Algoritmos y Estructuras de Datos (6201)	6001 Org. Y Estructura del Computador I (6106)	8207 Matemáticas III (8206)	6107 Matemáticas Discretas II (6106)	
III	6203 Ingeniería de Software (6202, 6301)	6002 Org. Y Estructura del Computador II (6001, 6201)	8208 Matemáticas III (8207)	0030 Inglés I	
IV	6004 Sistemas Operativos (6002, 6202)	6303 Bases de Datos (6203, 6107)	6104 Probabilidad y Estadística (8208, 6202, 6107)	6108 Matemáticas Discretas III (6202, 8207, 6107)	0031 Inglés II (0030)
V	6204 Lenguajes de Programación (6203, 6002, 6108)	6302 Sistemas de Información (6203, 6303)	6003 Comunicación de Datos (6002, 6202, 8208)	6105 Calculo Cientifico (8208, 6202)	
VI	Asignatura Obligatoria Optativa	Asignatura Obligatoria Optativa	Asignatura Optativa	Asignatura Electiva	
VII	Asignatura Electiva	Asignatura Electiva	Asignatura Electiva	Laboratorio	
VIII	Asignatura Electiva	Asignatura Electiva	6403 Pasantía	Asignatura Optativa	



**Figura 3.** Plan de Estudio de Computación.

La Escuela de Computación ofrece las siguientes opciones profesionales: Sistemas de Información, Bases de Datos, Modelos y Programación Matemática, Ingeniería de Software e Interacción Humano Computador, Sistemas Distribuidos y Paralelos, Inteligencia Artificial, Tecnologías Educativas, Cálculo Científico, Tecnologías en Comunicación y Redes de Computadoras, Computación Gráfica, y Aplicaciones en Internet. Para optar por una opción profesional el estudiante debe aprobar un mínimo de 15 unidades de crédito en asignaturas correspondientes a la opción de su preferencia.

Finalmente, para obtener el título de Licenciado en Computación el estudiante debe aprobar al menos 193 unidades de crédito.

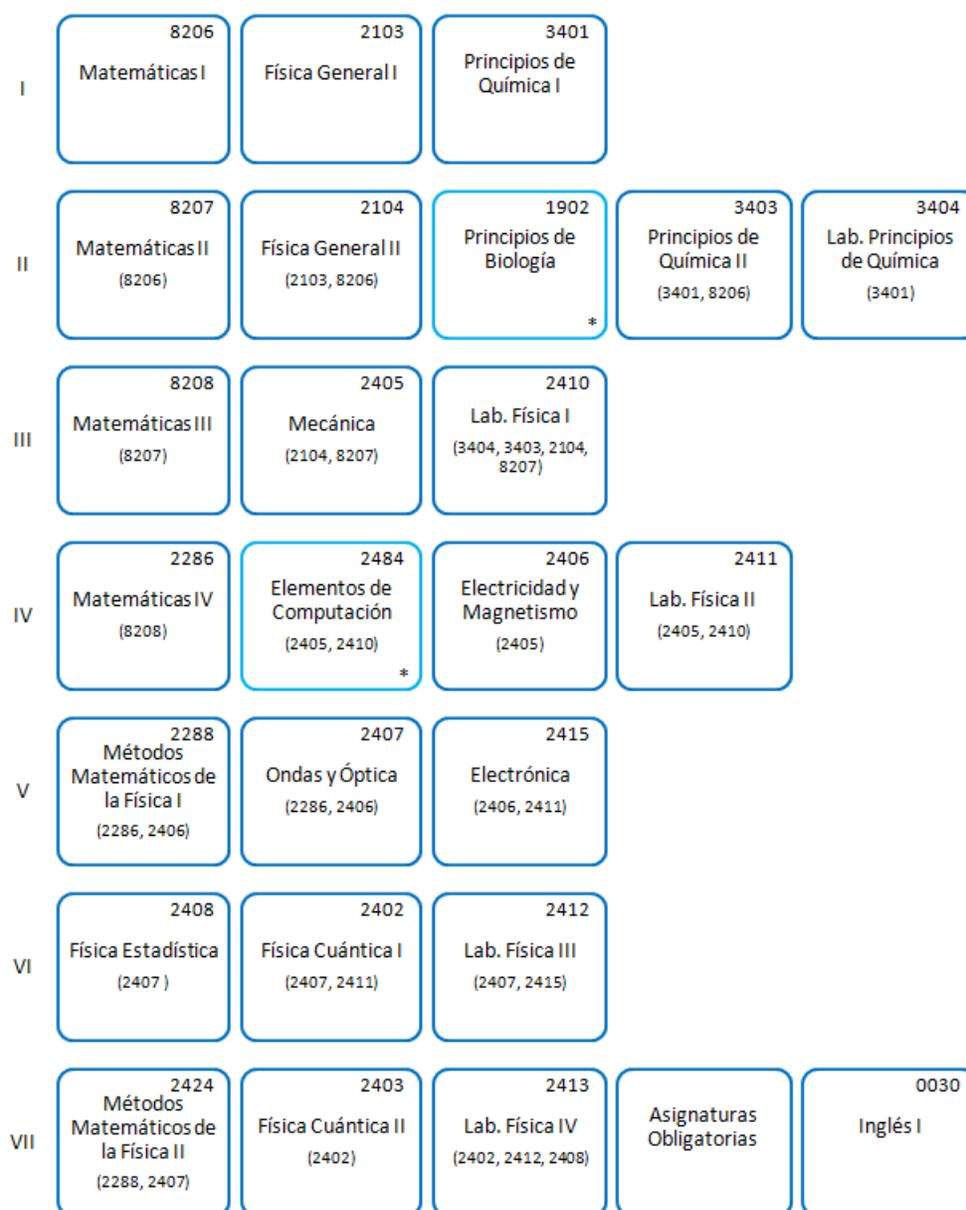
### 1.3 Licenciatura en Física

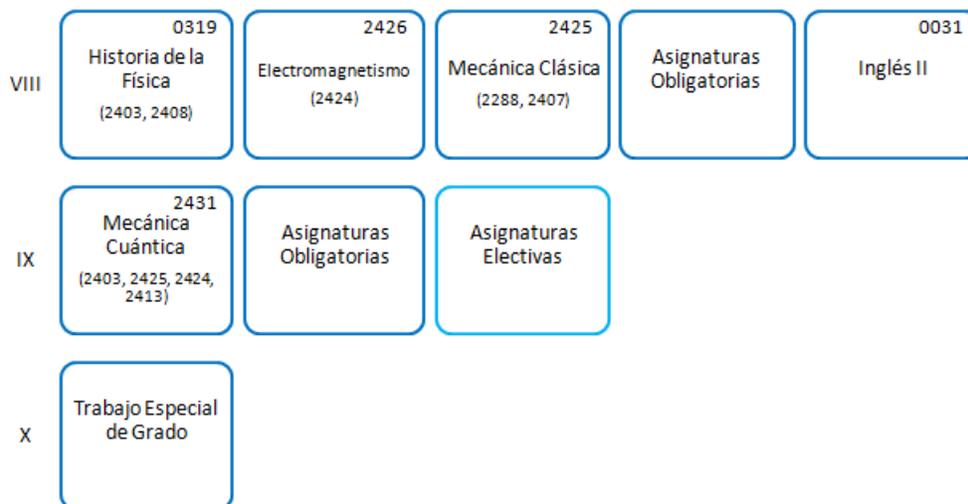
El plan de estudio de la Licenciatura en Física se encuentra vigente desde el año 1977. El Licenciado en Física en cualquier área de actuación, le corresponde ser un profesional que apoyado en conocimientos sólidos y actualizados de la física, debe ser capaz de tratar problemas nuevos y tradicionales, preocupándose por la búsqueda de nuevas formas del saber y del hacer científico y/o tecnológico.

El Físico estudia fenómenos de la naturaleza, como la energía en todas sus formas, la estructura de la materia, las leyes que rigen, los movimientos de los cuerpos y la interacción entre ellos. Trabaja en programas multidisciplinarios relacionados con la metalurgia, geofísica, astrofísica, oceanografía, química, electrónica y medicina. A través de la investigación pura y aplicada comprueba y propone nuevas leyes.

El plan de estudio de Física está estructurado en dos partes. La primera parte corresponde al ciclo básico de la carrera, su objetivo es brindar una formación básica de los grandes principios de la física. El ciclo básico tiene una duración de 9 semestres, está conformado por 26 asignaturas obligatorias y 1 asignatura electiva para un total de 142 unidades de crédito. El estudiante puede optar por al menos una de estas electivas: Principios de Biología y Elementos de Computación.

La segunda parte corresponde al ciclo de opción de la carrera, su objetivo es profundizar la formación del estudiante en un determinado campo profesional dentro del campo de la física. El ciclo de opción dispone de 4 semestres (VII, VIII, IX, X), para la aprobación de las asignaturas correspondientes a las opciones, con 35 unidades de crédito distribuidas de la siguiente manera: 15 unidades de crédito en asignaturas obligatorias (Trabajo Especial de Grado), y 20 unidades de crédito en asignaturas obligatorias y electivas (según las opciones). Además, el estudiante deberá cursar Inglés I e Inglés II o presentar un examen de suficiencia con un valor de 8 unidades de crédito, adicionales a las 35 señaladas anteriormente. La Figura 4 muestra el plan de estudio de Física.





**Figura 4.** Plan de Estudio de Física.

La Escuela de Física ofrece las siguientes opciones profesionales: Astrofísica, Física Computacional, Física Experimental, Geofísica, Instrumentación, Física Médica y Física Teórica. Para optar por una opción profesional el estudiante debe aprobar un mínimo de 20 unidades de crédito en asignaturas correspondientes a la opción de su preferencia.

Finalmente, para obtener el título de Licenciado en Física el estudiante debe aprobar al menos 185 unidades de crédito.

Para los estudiantes que deseen ingresar al Programa Cooperativo de Formación Docente, deben tener aprobadas todas las asignaturas hasta el 5to semestre inclusive. El estudiante debe cursar 111 unidades de crédito en asignaturas dictadas por la Facultad de Ciencias y 52 unidades de crédito en asignaturas dictadas por la Escuela de Educación para optar por el componente docente. La Figura 5 muestra el componente docente de Física.

I	8206 Matemática I	2103 Física General I	3401 Principios de Química I		
II	8207 Matemática II (8206)	2104 Física General II (2103, 8206)	3403 Principios de Química II (3401, 8206)	22W2 Introducción a la Computación (8206)	
III	8208 Matemática III (8207)	2405 Mecánica (8207, 2104)	2410 Lab. Física I (2104, 8207, 3409)		
IV	2286 Matemática IV (8208)	2406 Electricidad y Magnetismo (2405)	2411 Lab. Física II (2405, 2410)		
V	2415 Electrónica (2406, 2411)	2407 Ondas y Óptica (2282, 2406)	0030 Idiomas I		
VI	2402 Física Cuántica (2407, 2411)	2408 Física Estadística (2407)	22Y1 Introducción a la Pedagogía (2407, 2415)	22Y2 Desarrollo Social (2407)	
VII	2403 Física Cuántica II (2402)	22Y3 Psicopedagogía (2407, 2415)	22Y4 Didáctica General (2407, 2415)	22Y7 Administración Escolar (2407, 2415)	0031 Idiomas II (0030)
VIII	22Y6 Didáctica de la Física (22Y4)	22Y8 Evaluación (2407, 2415)	22Z2 Filosofía y Educación (2407, 2415)	22Y1 Historia de la Física (2403, 2408)	
IX	22Z8 Seminario (2407, 2415)	22Z7 Prácticas Administrativo-Docente (22Y6, 22Y7)			

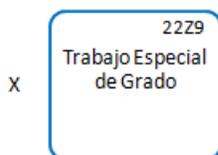
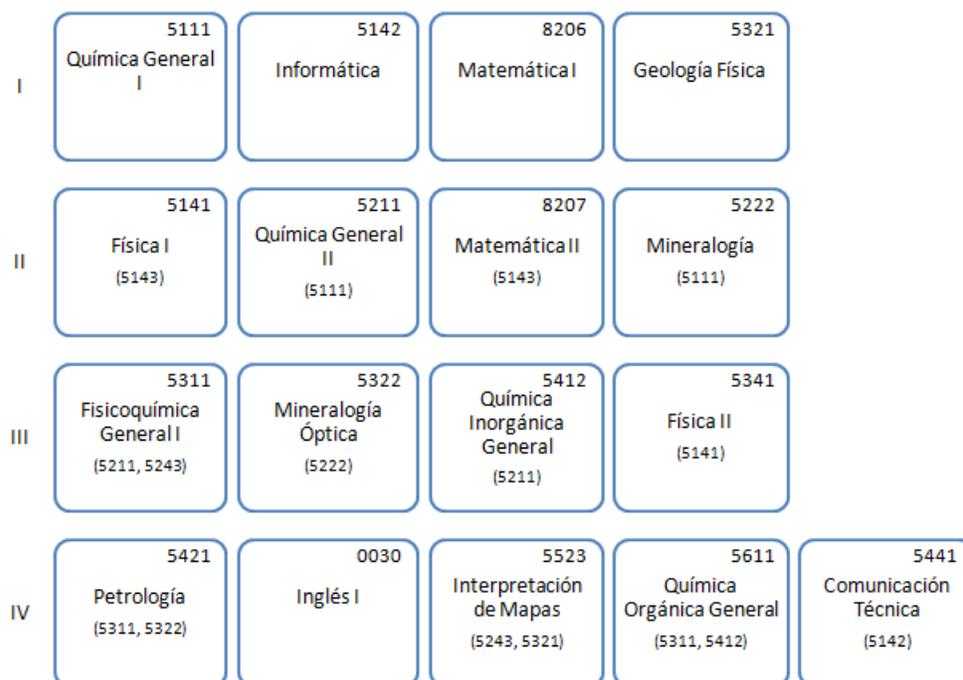


Figura 5. Componente Docente de Física.

#### 1.4 Licenciatura en Geoquímica

El plan de estudio de la Licenciatura en Geoquímica se encuentra vigente desde el año 2000. El Licenciado en Geoquímica estudia la composición de los diversos ambientes y materiales que constituyen el planeta, investiga los procesos y leyes que determinan dicha composición. Este profesional contribuye al entendimiento global del sistema tierra y a la búsqueda y uso racional de los materiales naturales requeridos por la sociedad moderna.

El plan de estudio de Geoquímica está conformado por 39 asignaturas obligatorias y 4 asignaturas electivas para un total de 191 unidades de crédito distribuidas de la siguiente manera: 171 unidades de crédito en asignaturas obligatorias y electivas, y 20 unidades de crédito en las asignaturas Proyecto Geoquímico I y Proyecto Geoquímico II. La Figura 6 muestra el plan de estudio de Geoquímica.



V	5521 Análisis Geoestructural (5421)	0031 Inglés II (0030)	5641 Economía General (5243, 5441)	5841 Ordenamiento Jurídico Venezolano (5441)	5531 Geoquímica (5142)
VI	5621 Métodos Geol. y Geoq. Campo (5523, 5521, 5531)	5522 Geomorfología (5523, 5521)	5622 Trabajo Geol. y Geoq. de Campo (5531, 5621)	5411 Química Analítica General (5211, 5341)	5642 Geoestadística (5142, 5243)
VII	5721 Suelos (5531)	5731 Geoquímica Orgánica (5531, 5611)	5732 Hidrogeoquímica (5531)	5822 Yacimientos Minerales (5531)	5511 Instrumental Analítico en Geoquímica (5411)
VIII	5821 Int. a la Geología de Venezuela (5531, 5521)	5741 Sociedad Venezolana (5841)	5831 Geoquímica de Campo (5622, 5642)	5832 Prospección Geoquímica (5322, 5642)	Electiva
IX	5941 Inglés III (0031)	5991 Proyecto Geoquímico I	Electiva	Electiva	
X	5999 Proyecto Geoquímico II (5991)	Electiva			

**Figura 6.** Plan de Estudio de Geoquímica.

Finalmente, para obtener el título de Licenciado en Geoquímica el estudiante debe aprobar al menos 191 unidades de crédito.

### 1.5 Licenciatura en Matemática

El plan de estudio de la Licenciatura en Matemática se encuentra vigente desde el año 2004. El Licenciado en Matemática está capacitado para comenzar a realizar labores de investigación y desarrollo de nuevas teorías; conceptos que posteriormente pueden tener aplicación en la ciencia, la industria, la computación, la estadística, etc. Resuelve problemas matemáticos complejos e incidentales en las investigaciones científicas y en las actividades económicas. Ejerce funciones de asesor sobre diversas aplicaciones del análisis matemático.

El plan de estudio de Matemática está conformado por 28 asignaturas obligatorias, 3 asignaturas electivas y 2 asignaturas optativas para un total de 194 unidades de crédito distribuidas de la

siguiente manera: 160 unidades de crédito en asignaturas obligatorias y electivas, 6 unidades de crédito en asignaturas optativas, y 28 unidades de crédito en las asignaturas Seminario I, Seminario II y Trabajo Especial de Grado. El estudiante puede optar por al menos una de estas asignaturas: Introducción a la Computación, Cálculo Numérico y Cálculo Científico. La Figura 7 muestra el plan de estudio de Matemática.

I	8206 Matemática I	2103 Física I	8101 Elementos de la Matemática			
II	8207 Matemática II (8206)	2104 Física II (2103)	8102 Álgebra I (8101)	Idioma		
III	8208 Matemática III (8207)	8005 Geometría Básica (8102)	8601 Introd. a la Computación (8206, 8102) *	Cálculo Numérico (8208, 8102) *	Cálculo Científico (8208, 8102) *	Idioma
IV	8401 Análisis I (8208)	8701 Probabilidades (8208)	8103 Álgebra II (8102)	Estudio Complementario		
V	8402 Análisis II (8401)	8801 Estadística (8701)	8503 Ec. Diferenciales Ordinarias (8208, 8103)	Estudio Complementario		
VI	8901 Topología (8402)	8403 Funciones Analíticas (8401)	8104 Álgebra III (8103)			
VII	8902 Teoría de la Medida (8402)	8003 Geometría Diferencia (8402)	8502 Ec. en Derivadas Parciales (8501)			
VIII	8903 Análisis Funcional (8901, 8902)	Asignatura Básica General	Electiva			

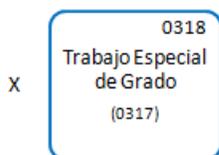


**Figura 7.** Plan de Estudio de Matemática.

Finalmente, para obtener el título de Licenciado en Matemática el estudiante debe aprobar al menos 194 unidades de crédito.

Para los estudiantes que deseen ingresar al Programa Cooperativo de Formación Docente, deben tener aprobadas todas las asignaturas hasta el 5to semestre inclusive. El estudiante debe cursar 101 unidades de crédito en asignaturas dictadas por la Facultad de Ciencias y 52 unidades de crédito en asignaturas dictadas por la Escuela de Educación para optar por el componente docente. En el III Semestre el estudiante puede optar por al menos una de estas asignaturas: Introducción a la Computación, Cálculo Numérico y Cálculo Científico. En el VI Semestre el estudiante puede optar por al menos una de estas asignaturas: Historia de la Matemática y Filosofía de la Ciencia. Además, la electiva Opción Docente es alguna de las siguientes asignaturas: Funciones Analíticas, Topología, Ecuaciones Diferenciales Ordinarias, Geometría Diferencial, Álgebra III o Teoría de la Medida. La Figura 8 muestra el componente docente de Matemática.

I	8206 Matemática I	2103 Física I	8101 Elementos de la Matemática			
II	8207 Matemática II (8206)	2104 Física II (2103)	8102 Álgebra I (8101)	Idioma		
III	8208 Matemática III (8207)	8005 Geometría Básica (8102)	8601 Introd. a la Computación (8206, 8102) *	Cálculo Numérico (8208, 8102) *	6105 Cálculo Científico (8208, 8102) *	Idioma
IV	8401 Análisis I (8208)	8701 Probabilidades (8208)	8103 Álgebra II (8102)			
V	8402 Análisis II (8401)	8801 Estadística (8701, 8601)	Electiva Opción Docente			
VI	8201 Historia de la Matemática (8208) *	1927 Filosofía de la Ciencia *	0301 Introducción a la Pedagogía	0302 Desarrollo Social		
VII	0303 Psicopedagogía	0304 Didáctica General	0307 Administración Escolar			
VIII	0308 Evaluación (0304)	0309 Didáctica de las Matemáticas (0304)	0311 Filosofía y Educación			
IX	0316 Prácticas Administrativo- Docentes (0307, 0309)	Seminario				



**Figura 8.** Componente Docente de Matemática.

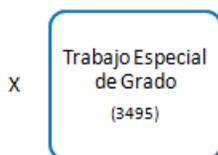
## 1.6 Licenciatura en Química

El plan de estudio de la Licenciatura en Química se encuentra vigente desde el año 1968. El Licenciado en Química estudia la materia, su composición, propiedades y transformaciones en la estructura. Asesora técnicamente o trabaja en el diseño y modificaciones de procesos químicos, soluciones de problemas industriales, innovación tecnológica, evaluación del impacto ambiental, mejoramiento y síntesis de productos, control de calidad, análisis de productos diversos.

El plan de estudio de Química está estructurado en dos partes. La primera parte corresponde al ciclo básico de la carrera, su objetivo es brindar una formación básica de los grandes principios de la química. El ciclo básico tiene una duración de 8 semestres, está conformado por 27 asignaturas obligatorias para un total de 114 unidades de crédito.

La segunda parte corresponde al ciclo de opción de la carrera, su objetivo es profundizar la formación del estudiante en un determinado campo profesional dentro del campo de la química. El ciclo de opción dispone de 5 semestres (VI, VII, VIII, IX, X), para la aprobación de las asignaturas correspondientes a las opciones, con 69 unidades de crédito distribuidas de la siguiente manera: 39 unidades de crédito en asignaturas obligatorias, y 30 unidades de crédito en las asignaturas Seminario de Investigación y Trabajo Especial de Grado. Además, el estudiante deberá cursar Inglés I e Inglés II o presentar un examen de suficiencia con un valor de 8 unidades de crédito, adicionales a las 69 señaladas anteriormente. La Figura 9 muestra el plan de estudio de Química.

I	2103 Física General I	8206 Matemáticas I	3401 Princ. Química		
II	2104 Física General II (2103, 8206)	8207 Matemáticas II (8206)	3403 Princ. Química II (3401, 8206)	3404 Lab. Princ. de Química (3401, 3403)	1903 Princ. de Biología
III	2440 Física General III (2104, 8207)	8208 Matemáticas III (8207)	3440 Q. Inorgánica I (3403)	2416 Lab. Física I (2104, 3404)	
IV	3416 Q. Orgánica I (3440)	2417 Lab. Física II (2440, 2416)	3442 Q. Inorgánica II (3440)	3441 Lab. I Q. Inorgánica (3404, 3440)	
V	3418 Q. Orgánica II (3416)	3443 Lab. II Q. Inorgánica (3442, 3441)	3460 Fisicoquímica I (8208, 2440)	3430 Q. Analítica I (3440)	
VI	3414 Lab. Q. Orgánica (F) I (3418, 3441)	3445 Lab. Q. Analítica (3430, 2417, 3441)	3462 Fisicoquímica II (3460)	Asignaturas Obligatorias	
VII	3432 Q. Analítica II (3430, 2440, 3416)	3423 Lab. Orgánica (F) II (3414, 3418)	Asignaturas Obligatorias	0030 Inglés I	
VIII	3424 Lab. Fisicoquímica Integrado (3445, 2416, 3462)	3426 Lab. Instrumental Analítico (3432, 3445)	Asignaturas Obligatorias	0031 Inglés II (0030)	
IX	3495 Seminario de Investigación	Asignaturas Obligatorias			

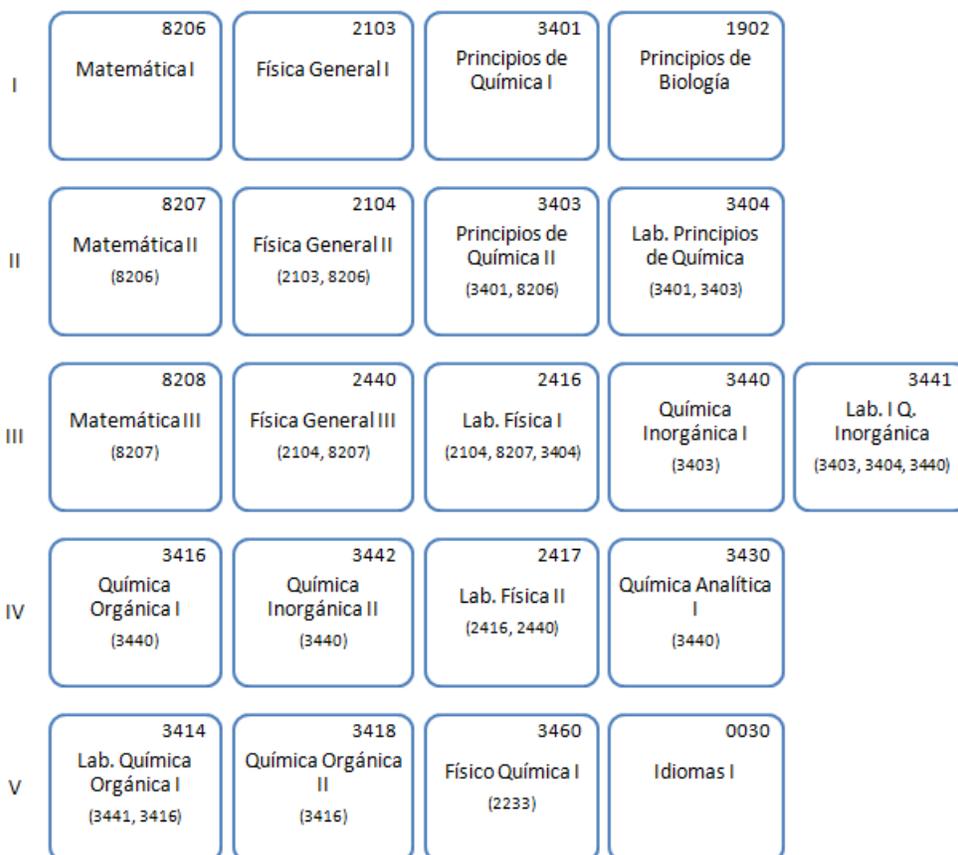


**Figura 9.** Plan de Estudio de Química.

La Escuela de Química ofrece las siguientes opciones profesionales: Básica, Geoquímica y Tecnología. Para optar por una opción profesional el estudiante debe aprobar un mínimo de 39 unidades de crédito en asignaturas, Seminario de Investigación y el Trabajo Especial de Grado correspondientes a la opción de su preferencia.

Finalmente, para obtener el título de Licenciado en Química el estudiante debe aprobar al menos 191 unidades de crédito.

Para los estudiantes que deseen ingresar al Programa Cooperativo de Formación Docente, deben tener aprobadas todas las asignaturas hasta el 5to semestre inclusive. El estudiante debe cursar 116 unidades de crédito en asignaturas dictadas por la Facultad de Ciencias y 52 unidades de crédito en asignaturas dictadas por la Escuela de Educación para optar por el componente docente. La Figura 10 muestra el componente docente de Química.



VI	3432 Química Analítica II (3416, 2440, 3430)	3445 Lab. Química Analítica I (3430, 3416, 3441)	22Y1 Introducción a la Pedagogía	22Y3 Psicopedagogía	0031 Idiomas II (0030)
VII	22Y7 Administración Escolar	22Y4 Didáctica General	3462 Físico Química II (3460)	3461 Lab. Química Orgánica II (3416, 3460, 3442)	3433 Lab. Química Analítica II (3431)
VIII	22Y6 Didáctica de la Química (22Y4)	22Y8 Evaluación	22Y2 Desarrollo Social	3463 Lab. Físico Química I (3462, 3461)	3443 Lab. Química Inorgánica II (3440, 3441)
IX	3426 Instrumental Analítico (3432, 3433)	22Z2 Filosofía y Educación	22Z8 Seminario		
X	22Z7 Prácticas Administrativo-Docente (22Y6, 22Y7)	22Z9 Trabajo Especial de Grado			

**Figura 10.** Componente Docente de Química.

## Capítulo 2

### Tecnologías Web

Desde el 26 de Febrero de 2007 se encuentra operativo el sistema CONEST, con la finalidad de automatizar la gestión académica de la Facultad de Ciencias de la UCV.

El proyecto surge como una iniciativa de la Facultad de Ciencias para resolver una problemática administrativa, potenciar los servicios de gestión académica de la División de Control de Estudios (DCE) y modernizar la infraestructura informática, entre otras cosas, incentivando la participación de forma activa de estudiantes, docentes y personal administrativo que forman parte de esta comunidad.

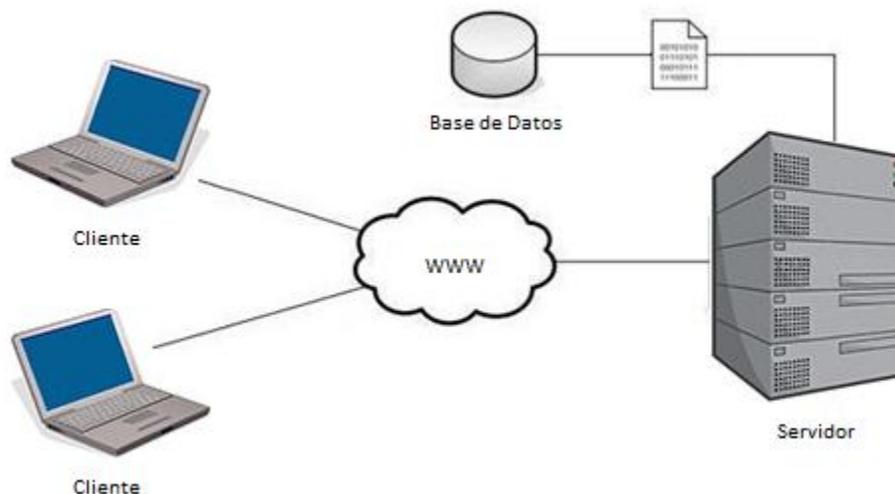
El sistema CONEST es una aplicación web desarrollada con software libre. Las aplicaciones web (Hassani, 2005) son aquellas que se acceden a través de una red tales como internet o una intranet. En otras palabras, es un software de aplicación codificado en un lenguaje soportado por un navegador web y que depende del mismo para ejecutar la aplicación.

Las aplicaciones web son populares debido a la accesibilidad de los navegadores web. La capacidad de actualizar y mantener aplicaciones web sin distribuir e instalar software en miles de clientes potenciales y la compatibilidad entre plataformas son otras razones de su popularidad.

Las aplicaciones web funcionan en una arquitectura cliente-servidor (Beck, 2005). En las arquitecturas cliente-servidor una aplicación se modela como un conjunto de servicios proporcionados por los servidores y un conjunto de clientes que usan esos servicios. Los clientes necesitan conocer que servidores están disponibles, pero normalmente no conocen la existencia de otros clientes.

Con frecuencia clientes y servidores se comunican a través de una red de comunicación en hardwares separados utilizando el Protocolo de Transferencia de Hipertexto (HTTP) (Patrick, 2004) el cual es protocolo de red para sistemas de colaboración de información hipermedia distribuidos. HTTP es la base de la comunicación de datos en la Worl Wide Web (WWW).

HTTP funciona como un protocolo petición-respuesta en la arquitectura cliente-servidor. En HTTP un navegador web, por ejemplo, actúa como un cliente, mientras que una aplicación que se está ejecutando en una computadora que hospeda un sitio web actúa como un servidor. El cliente envía un mensaje de solicitud HTTP al servidor. El servidor que almacena contenido, proporciona recursos tales como archivos HTML e imágenes o genera el contenido según sea necesario, o desempeña otras funciones por el cliente, devuelve una respuesta HTTP. Una respuesta contiene información sobre el estado de finalización de la solicitud y puede contener cualquier contenido solicitado por el cliente en el cuerpo del mensaje. La Figura 15 muestra un modelo lógico de una arquitectura cliente-servidor utilizando el protocolo HTTP.



**Figura 11.** Arquitectura Cliente-Servidor.

El término aplicación web generalmente está asociado con las tecnologías web que permiten crear herramientas o servicios que facilitan el intercambio interactivo de información, interoperabilidad, diseño centrado en el usuario y la colaboración en la WWW.

Las tecnologías web del lado del cliente permiten que se ejecuten elementos interactivos en el navegador del usuario. Esto hace posible que la navegación del sitio web sea a través de menús dinámicos y las validaciones del lado del cliente.

Ciertas operaciones se realizan del lado del cliente debido a que requieren acceso a información o funcionalidades que están disponibles en el cliente pero no en el servidor, o porque el usuario debe ingresar datos. A continuación se describe brevemente y se especifican las características más importantes de las tecnologías web del lado del cliente implementadas en el sistema CONEST:

## 2.1 HTML5

HTML5 (Hickson, 2010) es la quinta revisión del estándar HTML, actualmente en desarrollo. Como sus predecesores inmediatos (HTML 4.01 y XHTML 1.1), HTML5 es un estándar para estructurar y presentar contenido en la WWW.

El estándar incorpora nuevas características como los elementos <canvas>, <audio> y <video> así como la integración de contenido Gráfico Vectorial Escalable (SVG), que están diseñados para mejorar la inclusión y el manejo de contenido gráfico y multimedia en la web sin recurrir a extensiones y sus APIs. Otros elementos nuevos tales como <section>, <article>, <header> y <nav> están diseñados para mejorar la semántica de los documentos. Nuevos atributos también han sido introducidos, y otros removidos, para mejorar y simplificar la semántica de las expresiones.

Partes de HTML5 van a estar terminadas e implementadas en los navegadores progresivamente antes de que toda la especificación alcance el estado final de recomendación del World Wide Web Consortium (W3C). HTML5 incluye las siguientes características:

- El elemento canvas para el modo inmediato de dibujo 2D.
- Reproducción multimedia temporizada.
- Bases de datos para almacenamiento sin conexión de red.
- Edición de documentos.
- Drag-and-drop.
- Mensajería entre documentos.
- Gestión del historial del navegador.
- Microdata.

## 2.2 CSS3

CSS3 (Bos, 2005) es la tercera especificación del estándar CSS, actualmente en desarrollo. CSS es un lenguaje de hojas de estilo que se utiliza para describir la presentación (el aspecto y el formato) de un documento escrito en un lenguaje de marcado. Su aplicación más común es para estilizar páginas web escritas en HTML y XHTML.

CSS3 se basa en sus predecesores (CSS1 y CSS2), incluye una serie de selectores como [att^=""], [att\$=""] y [att\*=""] que permiten seleccionar elementos específicos de un documento HTML sin tener que depender de identificadores, clases y scripts innecesarios. Otros atributos nuevos tales como font-face, word-wrap, text-shadow, border-image y rgba mejoran en gran medida el trabajo de los desarrolladores, por lo que algunas de las tareas de CSS que consumen más tiempo son muy sencillas, permitiendo un mejor código, más limpio y ligero.

El desarrollo de CSS3 está dividido en módulos. La especificación anterior se ha separado en pedazos más pequeños y se han añadido nuevas funcionalidades. Varios de los módulos ya están terminados, incluyendo SVG, Media Queries y Namespaces, mientras que otros todavía están en desarrollo. Las nuevas funciones se incorporarán gradualmente en los navegadores web en los próximos años. CSS3 incluye las siguientes características:

- Atributos de textos como font-face para embeber fuentes de letras.
- Margen, borde, relleno y posicionamiento para la mayoría de los elementos.
- Color y opacidad de textos, fondos y otros elementos.
- Diseño de múltiples columnas sin utilizar elementos <div>.
- Transiciones y animaciones.
- Agregar esquinas redondeadas a elementos HTML sin imágenes de fondo.

## 2.3 jQuery

jQuery (The jQuery Project, 2010) es una librería de JavaScript que simplifica la manera de interactuar con los documentos HTML, seleccionar elementos DOM, desarrollar animaciones, manejar eventos y agregar interacción de manera asíncrona con el servidor a páginas web.

jQuery al igual que otras librerías, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requieren de mucho más código, es decir, con esta librería se logran resultados en menos tiempo. jQuery también permite crear extensiones sobre la librería de JavaScript, esto contribuye a la creación de páginas web con funciones nuevas y generalmente muy específicas. jQuery incluye las siguientes características:

- Selección de elementos DOM.
- Interacción y modificación del árbol DOM.
- Manejo de eventos.
- Manipulación de CSS.
- Efectos y animaciones.
- AJAX.
- Extensibilidad a través de plugins.
- Compatibilidad con todos los navegadores modernos.

Al igual que HTML5, jQuery establece una serie de elementos que reflejan el uso típico de los sitios web modernos. La ventaja de HTML5 con respecto a jQuery, es que está diseñado para integrar y presentar contenido multimedia en la web sin recurrir a extensiones de terceros y sus APIs. Sin embargo, la especificación de HTML5 es un trabajo en curso, y partes del estándar aún no están terminadas. jQuery es una solución más consolidada y soportada por todos los navegadores modernos.

Las tecnologías web del lado del servidor son comúnmente utilizadas en aplicaciones web interactivas que permiten a los usuarios acceder bases de datos y recuperar o modificar datos que se encuentran alojados en un servidor.

El servidor usa lenguajes como Ruby para procesar peticiones de los clientes, interpretar y generar respuestas utilizando información de archivos y bases de datos, y enviarlas de vuelta al cliente en un lenguaje soportado por el navegador. A continuación se describe brevemente y se especifican las características más importantes de las tecnologías web del lado del servidor implementadas en el sistema CONEST.

## 2.4 Ruby on Rails

Ruby on Rails (RoR) (Ruby on Rails, 2011) es un framework para desarrollar aplicaciones web escrito en el lenguaje de programación Ruby que está optimizado para mejorar la productividad y con un mínimo de configuración.

RoR incluye herramientas que facilitan las tareas más comunes de desarrollo, como generadores que pueden construir automáticamente algunas de las clases y páginas necesarias para un sitio web básico. También incluye WEBrick, un servidor web sencillo que se distribuye con Ruby, y Rake, un sistema de construcción, distribuido como una gema. Juntos estas herramientas proveen un ambiente de desarrollo.

RoR está dividido en varios paquetes, llamados ActiveRecord (un sistema de mapeo objeto-relacional para acceder a la base de datos), ActiveResource (provee servicios web), ActionPack, ActiveSupport y ActionMailer. Aparte de los paquetes estándar, los desarrolladores pueden hacer plugins para ampliar los paquetes existentes.

RoR supone que hay una “mejor” manera de hacer las cosas, y está diseñado para fomentar esa manera (y en algunos casos para rechazar las alternativas). La filosofía de RoR incluye los siguientes principios:

- **Don't Repeat Yourself (DRY):** DRY sugiere que escribir el mismo código una y otra vez es una mala práctica de programación.
- **Convention Over Configuration (CoC):** CoC significa que RoR hace suposiciones sobre lo que se quiere hacer y cómo se va a hacerlo, en lugar de exigir que se especifique cada detalle a través de archivos de configuración.

Como muchos frameworks web, RoR usa el patrón de diseño arquitectónico Modelo-Vista-Controlador (MVC) para organizar la programación de las aplicaciones web. MVC es una arquitectura de software, considerado actualmente como un patrón de arquitectura utilizado en la ingeniería de software. El patrón aísla la lógica de negocio de la aplicación de su interfaz de usuario, permitiendo el desarrollo, prueba y mantenimiento de cada uno de manera independiente. A continuación se describe brevemente los componentes del patrón de diseño arquitectónico MVC:

**Modelo:** Un modelo representa la información (datos) de la aplicación y las reglas para manipular esos datos. En el caso de RoR, los modelos se utilizan principalmente para gestionar la interacción con las tablas en la base de datos. En muchos casos, una tabla en la base de datos corresponderá a un modelo en la aplicación. La mayor parte de la lógica de negocio de la aplicación se concentra en los modelos.

**Vista:** Las vistas representan las interfaces de usuario de una aplicación, generalmente son documentos HTML con código Ruby embebido que sólo realiza tareas relacionadas con la presentación de los datos. Las vistas se encargan de proporcionar datos al navegador web u otras herramientas que se utilicen para realizar peticiones a la aplicación.

**Controlador:** Los controladores proveen un enlace entre los modelos y las vistas. En RoR, los controladores son responsables de procesar las peticiones entrantes desde el navegador web, interrogar a los modelos para obtener datos, y pasar dichos datos a la vista para que pueda presentarlos.

## 2.5 Haml

Haml (Haml, 2011) es un lenguaje de enmarcado liviano que se utiliza para describir el XHTML de cualquier documento web sin el uso de codificación en línea tradicional. Está diseñado para resolver muchas de las fallas en los motores de plantillas, a la vez que hace el enmarcado tan limpio como se pueda. Haml funciona en sustitución de sistemas de plantillas de páginas tales como RHTML. Sin embargo, Haml evita la necesidad de codificar XHTML explícitamente en la plantilla, porque el mismo es una descripción del XHTML, con código para generar contenido dinámicamente. La filosofía de Haml incluye los siguientes principios:

- **El enmarcado debe ser limpio:** El enmarcado no debe ser utilizado simplemente como una herramienta para hacer que los navegadores desplieguen una página de la manera en que su autor lo desea. Lo que se despliega no es lo único que la gente tiene que ver; también tienen que ver, modificar y entender el enmarcado. Por lo tanto, el enmarcado debe ser tan amigable y agradable como el resultado desplegado.
- **El enmarcado debe ser DRY:** XHTML implica muchas repeticiones. La mayoría de los elementos tienen que ser nombrados al menos dos veces; una vez antes de su contenido y una vez después. ERB agrega aún más repeticiones y caracteres innecesarios. Haml evita todo esto apoyándose en la indentación, no el texto, para determinar donde los elementos y bloques de código comienzan y terminan. Esto no solo resulta en plantillas más pequeñas, sino también hace que el código sea más limpio.
- **El enmarcado debe estar bien indentado:** Uno de los principales problemas de los lenguajes de plantillas tradicionales es que no solo no fomentan el código bien indentado, sino que activamente lo hacen difícil, o hasta imposible, de escribir. El resultado es confuso, XHTML ilegible. Haml formatea las etiquetas para que estén bien indentadas y reflejen la estructura subyacente del documento.
- **La estructura XHTML debe ser clara:** XML y XHTML son formatos construidos sobre el concepto de documentos estructurados. Esa estructura es reflejada en el enmarcado, y debería ser reflejada igualmente en el meta-enmarcado, como Haml. Debido a que la lógica de Haml está basada en la indentación de los elementos hijos, esta estructura se mantiene naturalmente, haciendo que el documento sea más fácil de leer y más entendible para las personas.

## 2.6 Sass

Sass (Sass, 2011) es un metalenguaje de hojas de estilo. Es un lenguaje de scripting que es interpretado a CSS. SassScript es el lenguaje de scripting propiamente. Sass consiste de dos sintaxis. La sintaxis original, llamada “la sintaxis indentada” usa una sintaxis similar a la de Haml. Utiliza la indentación para separar bloques de código y saltos de línea para separar reglas. La nueva sintaxis “SCSS” usa un formato de bloques similar al de CSS. Utiliza corchetes para denotar bloques de código y punto y coma para separar reglas dentro de un bloque.

Sass extiende CSS al proveer varios mecanismos disponibles en otros lenguajes de programación más tradicionales, particularmente lenguajes orientado a objetos, pero que no están disponibles en CSS3 como tal. Cuando SassScript es interpretado, crea un bloque de reglas de CSS para varios selectores definidos por el archivo Sass. El interpretador Sass traduce SassScript a CSS. Adicionalmente, Sass puede monitorear un archivo Sass (.sass o .scss) y traducirlo a un archivo .css. Sass incluye las siguientes características:

- Variables.
- Indentación.
- Mixins.
- Parametros.
- Herencia de selectores.

## 2.7 MySQL

MySQL (Obfusco, 2003) es un Sistema Manejador de Base de Datos Relacional (RDBMS) que se ejecuta como un servidor que facilita el acceso de múltiples usuarios a una serie de base de datos. MySQL está escrito en C y C++, funciona en muchas plataformas diferentes, incluyendo Linux, Solaris, Symbian, Mac OS X y Microsoft Windows, entre otros.

MySQL es principalmente un RDBMS y por lo tanto no posee una interfaz grafica de usuario para administrar base de datos o gestionar los datos que contiene adentro. Los usuarios pueden usar las herramientas de línea de comandos incluidas, o descargarse un software o aplicación web para administrar base de datos, construir la estructura de base de datos, y gestionar los registros de datos. MySQL incluye las siguientes características:

- Soporte de múltiples plataformas.
- Stored procedure.
- Triggers.
- Indexación de texto completo.
- Motores de almacenamiento independientes (MyISAM para las consultas, InnoDB para las transacciones, MySQL Archive para almacenar el historial de datos).

### Capítulo 3

## Método Ágil de Desarrollo de Software

El sistema CONEST se construyó utilizando un método ágil de desarrollo de software. El desarrollo ágil de software (Asierra, 2004) es un marco de trabajo conceptual usado en la ingeniería de software para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información. El software se desarrolla en unidades de tiempo llamadas iteraciones, la cual debe durar de una a cuatro semanas. Cada iteración del ciclo de vida incluye: planificación, análisis de requerimientos, diseño, codificación, revisión y documentación. La Figura 16 muestra el ciclo de vida general para el desarrollo ágil de software.



**Figura 12.** Ciclo de Vida Para el Desarrollo Ágil de Software.  
(Fuente: <http://sdc.net.au/services/application-development.aspx>).

Las metodologías ágiles de desarrollo de software están diseñadas para producir software útil de forma rápida. Generalmente, son procesos en los que se entrelazan la especificación, el desarrollo y las pruebas. El software no se desarrolla y utiliza en su totalidad, sino en una serie de incrementos, donde en cada incremento se incluye nuevas funcionalidades al sistema. Ejemplos de metodologías de desarrollo de software ágiles son: RAD, SCRUM y XP. Todos los métodos ágiles comparten un conjunto de principios:

- **Participación del cliente:** Los clientes deben estar fuertemente implicados en todo el proceso de desarrollo. Su papel es proporcionar y priorizar nuevos requerimientos del sistema y evaluar las iteraciones del sistema.
- **Entrega incremental:** El software se desarrolla en incrementos, donde el cliente especifica los requerimientos a incluir en cada incremento.

- **Personas ante procesos:** Se deben reconocer y explotar las habilidades del equipo de desarrollo. Se les debe dejar desarrollar sus propias formas de trabajar, sin procesos formales, a los miembros del equipo.
- **Aceptar el cambio:** Se debe contar con que los requerimientos del sistema cambian, por lo que el sistema se diseña para dar cabida a estos cambios.
- **Mantener la simplicidad:** Se debe centrar en la simplicidad tanto en el software a desarrollar como en el proceso de desarrollo. Donde sea posible, se trabaja activamente para eliminar la complejidad del sistema.

El método ágil de desarrollo de software utilizado para controlar el proceso de desarrollo del módulo web para la gestión de planes de estudio en el sistema CONEST es una adaptación de la Programación Extrema. A continuación se describe brevemente y se especifican las características más importantes de la Programación Extrema.

### 3.1 Programación Extrema (XP)

La Programación Extrema (XP) (Beck, 2005) es posiblemente el método ágil más conocido y ampliamente utilizado. En XP todos los requerimientos se expresan como escenarios (llamados historias de usuarios), los cuales se implementan directamente como una serie de tareas. Los programadores trabajan en parejas y desarrollan pruebas para cada tarea antes de escribir el código. Todas las pruebas se deben ejecutar satisfactoriamente cuando el código nuevo se integre al sistema. XP implica varias prácticas:

- **Planificación incremental:** Los requerimientos se registran en tarjetas de historias y las historias a incluir en una entrega se determinan según el tiempo disponible y su prioridad relativa. Los desarrolladores dividen estas historias en tareas de desarrollo.
- **Entregas pequeñas:** El mínimo conjunto útil de funcionalidad que proporcione valor de negocio se desarrolla primero. Las entregas del sistema son frecuentes e incrementalmente añaden funcionalidad a la primera entrega.
- **Diseño sencillo:** Solo se lleva a cabo el diseño necesario para cumplir los requerimientos actuales.
- **Desarrollo previamente probado:** Se utiliza un sistema de pruebas de unidad automatizado para escribir pruebas para nuevas funcionalidades antes de que estas se implementen.
- **Refactorización:** Se espera que todos los desarrolladores refactoricen el código continuamente tan pronto como encuentren posibles mejoras en el código. Esto conserva el código sencillo y mantenible.
- **Programación en parejas:** Los desarrolladores trabajan en parejas, verificando cada uno el trabajo del otro y proporcionando la ayuda necesaria para hacer siempre un buen trabajo.

- **Propiedad colectiva:** Las parejas de desarrolladores trabajan en todas las áreas del sistema, de modo que no desarrollen islas de conocimientos y todos los desarrolladores posean todo el código. Cualquiera puede cambiar cualquier cosa.
- **Integración continua:** En cuanto acaba el trabajo en un área, se integra en el sistema entero. Después de la integración, se deben pasar al sistema todas las pruebas de unidad.
- **Ritmo sostenible:** No se consideran aceptables grandes cantidades de horas extras, ya que a menudo el efecto que tienen es que se reduce la calidad del código y la productividad a medio plazo.
- **Cliente presente:** Debe estar disponible al equipo de XP un representante de los usuarios finales del sistema (el cliente) a tiempo completo. En un proceso de programación extrema, el cliente es miembro del equipo de desarrolladores y es responsable de formular al equipo los requerimientos del sistema para su implementación.

Una vez que se han desarrollado las tarjetas de historias, el equipo de desarrolladores las divide en tareas y estima el esfuerzo y recursos requeridos para su implementación. El cliente establece entonces la prioridad de las historias a implementar, eligiendo aquellas historias que pueden ser utilizadas inmediatamente para entregar un apoyo útil al negocio. Cuando los requerimientos cambian, las historias sin implementar también cambian o se pueden descartar. Si se requieren cambios en un sistema que ya se ha entregado, se desarrollan nuevas tarjetas de historias y, de nuevo, el cliente decide si estos cambios tienen prioridad sobre nuevas funcionalidades.

XP adopta un enfoque “extremo” para el desarrollo iterativo. Se pueden construir varias veces al día nuevas versiones del software y los incrementos se entregan al cliente cada dos meses aproximadamente. Cuando un programador construye el sistema para crear una versión nueva, debe ejecutar todas las pruebas automatizadas existentes además de las pruebas para las funcionalidades nuevas. El nuevo software generado solamente se acepta si se ejecutan satisfactoriamente todas las pruebas.

Para evitar algunos de los problemas de las pruebas y de las validaciones del sistema. XP pone más énfasis en el proceso de pruebas que otros métodos ágiles. Las pruebas del sistema son fundamentales en XP, en la que se ha desarrollado un enfoque que reduce la probabilidad de producir nuevos incrementos del sistema que introduzcan errores en el software existente. Las características clave de las pruebas en XP son:

- Desarrollo previamente probado.
- Desarrollo de pruebas incremental a partir de los escenarios.
- Participación del usuario en el desarrollo de las pruebas y en la validación.
- El uso de bancos de pruebas automatizadas.

A continuación se describe la adaptación del método ágil XP utilizado en el proceso de desarrollo del módulo web para la gestión de planes de estudio en el sistema CONEST.

## 3.2 Adaptación del Método Ágil XP

Las prácticas del método ágil XP se agrupan en cuatro fases; Planificación (incremental), Diseño (sencillo y refactorización), Codificación (entregas pequeñas, cliente presente, integración continua, ritmo sostenible, programación en parejas y propiedad colectiva) y Pruebas (desarrollo previamente probado).

A continuación se describe como se implementaron las prácticas del método ágil XP en cada fase durante el desarrollo del módulo web para la gestión de planes de estudio en el sistema CONEST.

### 3.2.1 Planificación

En la fase de planificación se desarrollan las tarjetas de historias, las cuales deben ser definidas con la participación del cliente para que proporcionen y prioricen los requerimientos.

Los requerimientos relacionados con la gestión de los planes de estudio de las licenciaturas de la Facultad de Ciencias y sus normativas, que inciden en el proceso de graduación de los estudiantes regulares de pregrado se registraron en tarjetas de historias.

Las tarjetas de historias están conformadas por un nombre, una descripción y el número de días estimado para desarrollar el requerimiento. La Figura 17 muestra el formato de las tarjetas de historias.

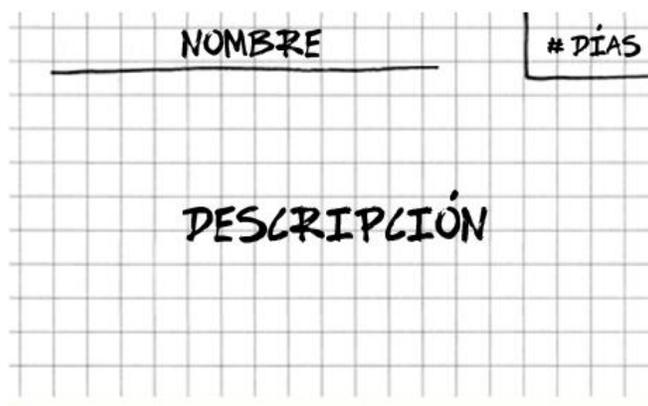


Figura 13. Formato de las Tarjetas de Historias.

Las tarjetas de historias no se dividieron en tareas de desarrollo por simplicidad y para reducir la documentación.

En esta fase también se planifican las entregas de los requerimientos del módulo web mediante iteraciones programadas, las cuales dividen la implementación de las tarjetas de historias y se entregan al cliente al final de cada iteración para hacer pruebas y correcciones.

Las iteraciones tienen una duración de tres meses, y las historias a incluir en una entrega se determinan según el número de días estimado para desarrollar los requerimientos y su prioridad relativa.

### **3.2.2 Diseño**

En la fase de diseño se esquematiza el módulo web, enfocándose en cubrir las necesidades inmediatas del cliente, ni más ni menos. En cada iteración se entregó el mayor valor de negocio produciendo el código más sencillo que cumpliera con los requerimientos definidos en las tarjetas de historias.

En esta fase también se revisó constantemente el código, refinando los diseños que se consideraron necesarios cuando se encontraban posibles mejoras en el código, con el fin de minimizar en lo posible las ineficiencias.

### **3.2.3 Codificación**

En la fase de codificación se desarrolla el módulo web en incrementos según la planificación de las entregas, creando un programa sencillo que aportara valor de negocio rápidamente, que se actualizaba de manera frecuente añadiendo funcionalidades a la primera entrega.

El módulo web se desarrolló siguiendo los estándares de codificación del sistema CONEST, para asegurar la consistencia, facilitar la comprensión y la refactorización del código. A continuación se describe brevemente el estándar de codificación para el desarrollo del sistema CONEST.

Las reglas generales para escribir y documentar el código incluyen:

- En español.
- Sin eñes, acentos o caracteres especiales.
- Sustituir las eñes por enes.
- Omitir los acentos.
- Identar el código con 2 espacios en blanco.
- Usar comentarios para documentar cada método.
- Dejar un espacio en blanco antes y después de un signo igual, excepto en los parámetros por defecto de un método.
- No dejar espacios en blanco después de un método.

Las reglas generales para nombrar directorios, archivos, atributos, variables y constantes incluyen:

- Directorios, archivos, atributos y variables en minúscula.
- Constantes en mayúscula.
- Separar nombres compuestos con sub guiones.

Las reglas generales para nombrar clases incluyen:

- De manera capital (Primera letra en mayúscula y el resto en minúscula).
- No separar nombres compuestos.

Las reglas generales para nombrar métodos y parámetros incluyen:

- En minúscula.
- Separar nombres compuestos con sub guiones.
- Sí un método retorna un booleano debe terminar en ?.
- Sí un método modifica un objeto debe terminar en !.
- Sí un método arroja una excepción de error debe terminar en !.
- Sí un método no tiene parámetros debe terminar sin ().

Es importante resaltar que el cliente estuvo presente durante toda la fase de codificación del proyecto, trabajando activamente para revisar los requerimientos, las funcionalidades, señalar las prioridades y responder las preguntas del programador.

Se integró una versión completa del módulo web al sistema CONEST al final de la fase de codificación del proyecto.

Se trabajó 30 horas semanales, distribuidas entre los cinco días de la semana en jornadas diurnas, trabajando horas extras solo cuando se consideraba necesario.

El módulo web fue desarrollado por un solo programador, el cual trabajó todas las áreas del modulo web, impidiendo el desarrollo de islas de conocimientos.

### **3.2.4 Pruebas**

En la fase de pruebas se implementan pruebas funcionales y de aceptación al final de cada iteración, estas se generaron a partir de las tarjetas de historias que se desarrollaban durante una iteración.

Se definen los datos de entrada y los resultados esperados para cada tarjeta de historia, y el cliente verifica el correcto funcionamiento del requerimiento que se está probando. Cuando se da

por buena la prueba de aceptación, se considera que la historia correspondiente se ha completado.

El sistema de pruebas de unidad automatizado no se implementó para favorecer la productividad y las entregas rápidas y constantes durante el desarrollo del proyecto.

## **Capítulo 4**

### **Marco Aplicativo**

En este capítulo se describe el proceso de desarrollo del módulo web para la gestión de planes de estudio en el sistema CONEST.

El método ágil de desarrollo de software utilizado para estructurar, planificar y controlar el proceso de desarrollo de este Trabajo Especial de Grado es una adaptación de la Programación Extrema (XP). Es importante resaltar que algunas prácticas del método ágil XP no se aplicaron en este proyecto tales como el desarrollo previamente probado, la programación en parejas, la propiedad colectiva y la integración continua. Sin embargo, se aplicaron el resto de las prácticas tales como la planificación incremental, las entregas pequeñas, el diseño sencillo, la refactorización, el ritmo sostenible y la presencia del cliente.

Debido al método ágil de desarrollo de software utilizado en este Trabajo Especial de Grado, fue necesario definir y desarrollar un conjunto de iteraciones, en las cuales el software se desarrolla en incrementos. A continuación, se describen las iteraciones realizadas durante el desarrollo del módulo web para la gestión de planes de estudio en el sistema CONEST.

#### **4.1 Iteración 0**

En esta iteración se analiza de manera general el módulo web; se realiza un levantamiento de los requerimientos necesarios para la implementación, se desarrollan las tarjetas de historias y se esquematiza su funcionamiento.

##### **4.1.1 Planificación**

Los planes de estudio de las licenciaturas de la Facultad de Ciencias se estudiaron para determinar las características comunes y las diferencias de cada plan de estudio. Además, se analizó el proceso de graduación que lleva a cabo la DCE mediante entrevistas con los trabajadores del Departamento de Grado, para determinar la manera en que los planes de estudio influyen en dicho proceso.

El Departamento de Grado de la DCE utiliza los planes de estudio de las distintas licenciaturas cada semestre para verificar que los estudiantes que inscribieron Trabajo Especial de Grado cumplen con todos los requisitos curriculares y las normativas académicas que deben satisfacer para obtener un título universitario en la licenciatura en que se encuentran inscritos. En el proceso de graduación se verifican los siguientes requisitos curriculares:

- Aprobar el conjunto de asignaturas obligatorias.
- Aprobar el número de créditos en asignaturas electivas.

- Convalidaciones y equivalencias entre asignaturas, en caso de que aplique.
- Opciones profesionales.
- Componente docente.
- Servicio comunitario.

Es importante resaltar que los trabajadores del Departamento de Grado realizan todas sus labores con un soporte mínimo del sistema CONEST, verificando de manera manual los requisitos curriculares de cada estudiante con lápices de colores y distintos documentos impresos. Esto representa un mayor esfuerzo humano y aumenta la probabilidad de cometer errores.

Posteriormente, con el soporte y la supervisión de los tutores se estableció como el módulo web debe gestionar los planes de estudio para facilitar y simplificar las operaciones de los procesos de gestión académica de la DCE. Y por último, se revisó la base de datos actual del sistema CONEST para evaluar el soporte que provee y determinar las estructuras de datos necesarias para desarrollar las nuevas funcionalidades que incorporará el módulo web para la gestión de planes de estudio al sistema CONEST.

Tras estudiar los planes de estudio de las licenciaturas de la Facultad de Ciencias y el proceso de graduación de la DCE, se determinó que el módulo web para la gestión de planes de estudio en el sistema CONEST debe satisfacer los siguientes requerimientos:

- Gestionar los planes de estudio de las licenciaturas que ofrece la Facultad de Ciencias; requisitos curriculares, distribución de los requisitos curriculares por semestre, asignaturas, prelacones, convalidaciones y equivalencias, y opciones profesionales.
- Automatizar la verificación de requisitos curriculares del proceso de graduación del Departamento de Grado de la DCE.
- Aplicar un conjunto de pruebas para verificar el correcto funcionamiento del módulo web.
- Integrar el módulo web con el resto del sistema CONEST.

Las tarjetas de historias se desarrollan a partir de los requerimientos. En este Trabajo Especial de Grado se definieron las siguientes tarjetas de historias:

1. Diseñar una interfaz gráfica para verificar requisitos curriculares	1 Día
Definir la información necesaria y el formato adecuado para verificar requisitos curriculares en el módulo web	

2. Diseñar las estructuras de datos para verificar requisitos curriculares	1 Día
Diseñar las tablas, atributos, restricciones y asociaciones necesarias para soportar la verificación de requisitos curriculares en el módulo web	

3. Desarrollar la verificación de asignaturas obligatorias	2 Días
Verificar que un estudiante haya aprobado las asignaturas obligatorias en una carrera	
4. Desarrollar la verificación de asignaturas obligatorias optativas	1 Día
Verificar que un estudiante haya aprobado el número de créditos en asignaturas obligatorias optativas en una carrera	
5. Desarrollar la verificación de asignaturas electivas	1 Día
Verificar que un estudiante haya aprobado el número de créditos en asignaturas electivas en una carrera	
6. Desarrollar la verificación de asignaturas complementarias	1 Día
Verificar que un estudiante haya aprobado el número de créditos en asignaturas complementarias en una carrera	
7. Desarrollar la verificación de laboratorios	1 Día
Verificar que un estudiante haya aprobado el número de créditos en asignaturas de laboratorio en una carrera	
8. Desarrollar la verificación de pasantías	1 Día
Verificar que un estudiante haya aprobado el número de créditos en asignaturas de pasantía en una carrera	
9. Desarrollar la verificación de seminarios	1 Día
Verificar que un estudiante haya aprobado el número de créditos en asignaturas de seminario en una carrera	
10. Desarrollar la verificación de TEG	1 Día
Verificar que un estudiante haya aprobado el número de créditos en asignaturas de trabajo especial de grado en una carrera	
11. Desarrollar la verificación del servicio comunitario	1 Día
Verificar que un estudiante haya aprobado las asignaturas del servicio comunitario en una carrera	
12. Desarrollar la verificación de asignaturas excedentes	1 Día
Verificar si un estudiante ha aprobado más asignaturas electivas y/o asignaturas complementarias que las que se requieren en una carrera	
13. Desarrollar la verificación de convalidaciones	1 Día
Verificar si un estudiante ha aprobado una asignatura que haya sido convalidada por otra asignatura obligatoria en una carrera	
14. Diseñar una interfaz gráfica para la cabecera	1 Día
Diseñar un prototipo de interfaz para la cabecera de la vista para verificar requisitos curriculares	

15. Revisar los requisitos curriculares de la Licenciatura en Biología	1 Día
Definir los requisitos curriculares del plan de estudio de Biología que se verificaran en el módulo web	
16. Revisar los requisitos curriculares de la Licenciatura en Computación	1 Día
Definir los requisitos curriculares del plan de estudio de Computación que se verificaran en el módulo web	
17. Revisar los requisitos curriculares de la Licenciatura en Física	1 Día
Definir los requisitos curriculares del plan de estudio de Física que se verificaran en el módulo web	
18. Revisar los requisitos curriculares de la Licenciatura en Geoquímica	1 Día
Definir los requisitos curriculares del plan de estudio de Geoquímica que se verificaran en el módulo web	
19. Revisar los requisitos curriculares de la Licenciatura en Matemática	1 Día
Definir los requisitos curriculares del plan de estudio de Matemática que se verificaran en el módulo web	
20. Revisar los requisitos curriculares de la Licenciatura en Química	1 Día
Definir los requisitos curriculares del plan de estudio de Química que se verificaran en el módulo web	
21. Desarrollar un script de bases de datos	1 Día
Crear un script de bases de datos que contenga las asignaturas aprobadas por un estudiante de Computación	
22. Refactorizar el código para verificar requisitos curriculares	1 Día
Revisar y refinar el algoritmo para verificar requisitos curriculares	
23. Diseñar una interfaz gráfica para el resumen	1 Día
Diseñar un prototipo de interfaz para el resumen de la vista para verificar requisitos curriculares	
24. Desarrollar el resumen de los datos académicos	1 Día
Mostrar los datos académicos de un estudiante en una carrera	
25. Desarrollar el resumen de la verificación de los requisitos curriculares	1 Día
Mostrar el resultado de la verificación de los requisitos curriculares de un estudiante en una carrera	
26. Diseñar las estructuras de datos para verificar opciones profesionales	1 Día
Diseñar las tablas, atributos, restricciones y asociaciones necesarias para soportar la verificación de opciones profesionales en el módulo web	

27. Desarrollar el resumen de la verificación de la opción profesional	1 Día
Mostrar el resultado de la verificación de la opción profesional de un estudiante en una carrera	
28. Desarrollar el resumen general	1 Día
Mostrar el resultado general de la verificación de los requisitos curriculares y la verificación de la opción profesional de un estudiante en una carrera	
29. Refactorizar el código para mostrar los resúmenes	1 Día
Revisar y refinar el algoritmo para mostrar el resumen de la verificación de requisitos curriculares y el resumen de la verificación de opciones profesionales	
30. Desarrollar la selección de un estudiante y una carrera	1 Día
Crear un formulario para seleccionar un estudiante y una carrera que haya cursado en la Facultad de Ciencias	
31. Migrar el módulo web al sistema CONEST 3.1	2 Días
Migrar las funcionalidades del módulo web desarrolladas en el sistema CONEST 3.0 al sistema CONEST 3.1	
32. Diseñar las estructuras de datos para distribuir requisitos curriculares	1 Día
Diseñar las tablas, atributos, restricciones y asociaciones necesarias para soportar la distribución de requisitos curriculares en el módulo web	
33. Desarrollar el listado de planes de estudio	1 Día
Listar los planes de estudio	
34. Desarrollar la creación de planes de estudio	1 Día
Crear un formulario para crear un plan de estudio	
35. Desarrollar la edición de planes de estudio	1 Día
Crear un formulario para editar un plan de estudio	
36. Desarrollar la eliminación de planes de estudio	1 Día
Eliminar un plan de estudio	
37. Desarrollar el listado de requisitos curriculares	1 Día
Listar los requisitos curriculares de un plan de estudio	
38. Desarrollar la creación de requisitos curriculares	1 Día
Crear un formulario para crear un requisito curricular en un plan de estudio	
39. Desarrollar la edición de requisitos curriculares	1 Día
Crear un formulario para editar un requisito curricular de un plan de estudio	

40. Desarrollar la eliminación de requisitos curriculares	1 Día
Eliminar un requisito curricular de un plan de estudio	
41. Desarrollar el listado de requisitos curriculares distribuidos por semestre	1 Día
Listar los requisitos curriculares de un plan de estudio distribuidos por semestre	
42. Desarrollar la distribución de requisitos curriculares por semestre	1 Día
Distribuir los requisitos curriculares de un plan de estudio por semestre	
43. Desarrollar la redistribución de requisitos curriculares por semestre	1 Día
Redistribuir los requisitos curriculares de un plan de estudio en un semestre	
44. Desarrollar la eliminación de requisitos curriculares distribuidos por semestre	1 Día
Eliminar un requisito curricular de un plan de estudio en un semestre	
45. Desarrollar el listado de asignaturas	1 Día
Listar las asignaturas	
46. Desarrollar la creación de asignaturas	1 Día
Crear un formulario para crear una asignatura	
47. Desarrollar la edición de asignaturas	1 Día
Crear un formulario para editar una asignatura	
48. Desarrollar la eliminación de asignaturas	1 Día
Eliminar una asignatura	
49. Desarrollar el listado de asignaturas de un plan de estudio	1 Día
Listar las asignaturas de un plan de estudio	
50. Desarrollar la asignación de asignaturas de un plan de estudio	1 Día
Asignar una asignatura a un plan de estudio	
51. Desarrollar la edición de asignaturas de un plan de estudio	1 Día
Editar las asignaturas de un plan de estudio	
52. Desarrollar la eliminación de asignaturas de un plan de estudio	1 Día
Eliminar una asignatura de un plan de estudio	
53. Desarrollar un PDF para la vista para verificar requisitos curriculares	1 Día
Crear un documento PDF que contenga la verificación de los requisitos curriculares y el resumen de la verificación de los requisitos curriculares de un estudiante en una carrera	

54. Desarrollar las validaciones	2 Días
Validar el formato de todos los campos de los formularios del módulo web	
55. Desarrollar los mensajes de error	1 Día
Crear los mensajes de error de todos los formularios del módulo web	
56. Desarrollar los mensajes de confirmación	1 Día
Crear los mensajes de confirmación de todos los formularios del módulo web	
57. Desarrollar un script de bases de datos	1 Día
Crear un script de bases de datos que contenga las tablas, atributos, restricciones y asociaciones para soportar todas las funcionalidades del módulo web	

#### 4.1.2 Diseño

El sistema CONEST cuenta con una interfaz gráfica sencilla y organizada, que representa la información y las acciones disponibles a un usuario mediante iconos e indicadores visuales, y permite a los usuarios interactuar con el sistema a través de la manipulación directa de los elementos de la interfaz, para acceder las funcionalidades básicas del sistema y satisfacer los requerimientos respectivos.

El módulo web para la gestión de planes de estudio se desarrolló siguiendo la apariencia general y las técnicas de interacción del sistema CONEST para asegurar la consistencia, facilitar el aprendizaje y disminuir la carga cognitiva de los usuarios.

El módulo web resuelve la problemática administrativa de los requerimientos solicitados por los trabajadores de la DCE. La Figura 14 muestra el diseño general del sistema CONEST.



Figura 14. Diseño General del Sistema CONEST.

El módulo web para la gestión de planes de estudio en el sistema CONEST se desarrolló utilizando las herramientas tecnológicas que fueron descritas en el capítulo 2:

- Lenguaje de programación Ruby, versión 1.9.7.
- Framework para aplicaciones web Rails, versión 3.1.0.
- Gestor de paquetes RubyGems, versión 1.6.2.
- Lenguaje de enmarcado Haml, versión 3.1.2.
- Lenguaje de hojas de estilo Sass, versión 3.1.7.
- Lenguaje de programación CoffeScript, versión 3.1.0.
- Librería de JavaScript jQuery, versión 1.6.2.
- Manejador de bases de datos relacional MySQL, versión 5.1.

#### **4.1.3 Codificación**

La mayor parte de la gestión de los planes de estudio de las licenciaturas de la Facultad de Ciencias en el módulo web se realiza a través de listados y formularios. Los listados muestran los registros de la base de datos organizados en filas, y los usuarios pueden seleccionar un registro del listado para revisar sus datos detalladamente. Los formularios muestran los datos particulares de los registros, de los cuales se toman los datos ingresados por los usuarios, se realizan las validaciones correspondientes y se almacenan en la base de datos.

El módulo web se desarrolló siguiendo las guías de Ruby on Rails para evitar malas prácticas de programación y las vulnerabilidades de seguridad en el software.

#### **4.1.4 Pruebas**

Las pruebas funcionales se generaron a partir de las tarjetas de historias. Se definieron los datos de entrada y los resultados esperados para cada prueba, y los tutores, que son usuarios finales del sistema, verificaron el correcto funcionamiento de los requerimientos que se estaban probando al final de cada iteración.

Las pruebas de aceptación están dirigidas a los trabajadores de la DCE, los cuales realizaron una serie de tareas relacionadas con la gestión de los planes de estudio de las licenciaturas de la Facultad de Ciencias y la verificación de los requisitos curriculares de los estudiantes regulares de pregrado. Además, respondieron una encuesta para evaluar los requerimientos funcionales y los requerimientos no funcionales del módulo web.

## 4.2 Iteración 1

En esta iteración se describe el proceso de desarrollo de la gestión de los planes de estudio de las licenciaturas de la Facultad de Ciencias en el módulo web, siguiendo las prácticas del método ágil XP; especificando la planificación, el diseño, la codificación y las pruebas de las tarjetas de historias correspondientes.

### 4.2.1 Planificación

<b>Descripción</b>	Diseñar la interfaz gráfica, las estructuras de datos y desarrollar un algoritmo para gestionar los planes de estudio
<b>Tarjetas de Historias</b>	32. Diseñar las estructuras de datos para distribuir requisitos curriculares 33. Desarrollar el listado de planes de estudio 34. Desarrollar la creación de planes de estudio 35. Desarrollar la edición de planes de estudio 36. Desarrollar la eliminación de planes de estudio 37. Desarrollar el listado de requisitos curriculares 38. Desarrollar la creación de requisitos curriculares 39. Desarrollar la edición de requisitos curriculares 40. Desarrollar la eliminación de requisitos curriculares 41. Desarrollar el listado de requisitos curriculares distribuidos por semestre 42. Desarrollar la distribución de requisitos curriculares por semestre 43. Desarrollar la redistribución de requisitos curriculares por semestre 44. Desarrollar la eliminación de requisitos curriculares distribuidos por semestre 45. Desarrollar el listado de asignaturas 46. Desarrollar la creación de asignaturas 47. Desarrollar la edición de asignaturas 48. Desarrollar la eliminación de asignaturas 49. Desarrollar el listado de asignaturas de un plan de estudio 50. Desarrollar la asignación de asignaturas de un plan de estudio 51. Desarrollar la edición de asignaturas de un plan de estudio 52. Desarrollar la eliminación de asignaturas de un plan de estudio 54. Desarrollar las validaciones 55. Desarrollar los mensajes de error 56. Desarrollar los mensajes de confirmación 31. Migrar el módulo web al sistema CONEST 3.1
<b>Tiempo Estimado</b>	27 Días

### 4.2.2 Diseño

Como se mencionó anteriormente, la mayor parte de la gestión de los planes de estudio de las licenciaturas de la Facultad de Ciencias se realiza a través de listados y formularios.

El listado de planes de estudio muestra todos los planes de estudio en la base de datos organizados en filas. En la primera fila se muestra la cabecera, las columnas de la primera fila muestran la denominación de los atributos más relevantes de los planes de estudio. A partir de la segunda fila, se muestran los planes de estudio, las columnas de estas filas muestran los datos particulares de cada plan de estudio que correspondan con los atributos de la cabecera.

La última columna de las filas de los planes de estudio, muestran un icono para eliminar el plan de estudio correspondiente. Al colocar el cursor sobre el icono, se muestra una descripción emergente para informar al usuario sobre la funcionalidad del icono. Los usuarios deberán hacer clic en el icono, y confirmar la acción para eliminar un plan de estudio de la base de datos.

Los planes de estudio del listado se pueden ordenar alfabéticamente de manera descendiente y ascendiente por atributo, mediante unos iconos que se encuentran en cada columna de la cabecera. Luego del listado, se muestra un enlace para crear un plan de estudio nuevo. La Figura 15 muestra el listado de planes de estudio.

CONEST Sistema de Gestión Académica

Universidad Central de Venezuela | Facultad de Ciencias | Acerca de | Contáctenos | Ayuda

Inicio Preinscripción Inscripción Solicitudes Plan de Estudio Horarios

Planes de Estudio Lunes, 16 de Abril de 2011, 11:13 p.m.

**Listado de Planes de Estudio**

Nombre	Carrera	Régimen de Estudio	Fecha de Aplicación	
<a href="#">PLAN BIOLOGIA</a>	LICENCIATURA EN BIOLOGIA	SEMESTRAL	2009-03-27	A X
<a href="#">PLAN COMPUTACION</a>	LICENCIATURA EN COMPUTACION	SEMESTRAL	2009-03-27	A X
<a href="#">PLAN FISICA</a>	LICENCIATURA EN FISICA	SEMESTRAL	2009-03-27	A X
<a href="#">PLAN GEOQUIMICA</a>	LICENCIATURA EN GEOQUIMICA	SEMESTRAL	2009-03-27	A X
<a href="#">PLAN MATEMATICA</a>	LICENCIATURA EN MATEMATICA	SEMESTRAL	2009-03-27	A X
<a href="#">PLAN QUIMICA</a>	LICENCIATURA EN QUIMICA	SEMESTRAL	2009-03-27	A X

Para crear un plan de estudio nuevo haga click [aquí](#)

CONEST Versión 3.0.0 Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2011

**Figura 15.** Listado de Planes de Estudio.

Los usuarios pueden crear un plan de estudio nuevo por medio de un formulario. El formulario muestra todos los atributos de los planes de estudio mediante campos de entrada, los cuales deben ser llenados por los usuarios con la información solicitada, que se almacenará para su uso posterior o manipulación.

Los usuarios también pueden seleccionar un plan de estudio del listado para revisar sus datos particulares tales como el nombre, la carrera a la que pertenece, el régimen de estudio y la fecha de aplicación. Además, se muestra toda la información relacionada con los planes de estudio, como los requisitos curriculares, la distribución de los requisitos curriculares por semestre y las opciones profesionales, con la finalidad de acceder convenientemente y gestionar las distintas características de los planes de estudio en un solo lugar.

Se utilizó un menú de acordeones para organizar de manera estructurada las características de los planes de estudio. Los usuarios pueden minimizar y maximizar los acordeones para mostrar su contenido cuando sea necesario. Esta técnica permite mostrar una gran cantidad de información en poco espacio. La Figura 16 muestra la página web para editar planes de estudio.

Inicio Preinscripción Inscripción Solicitudes Plan de Estudio Horarios

Lunes, 16 de Abril de 2011, 11:13 p.m.

**Editar Plan de Estudio**

**Datos del Plan de Estudio**

Nombre  
PLAN COMPUTACION

Carrera  
LICENCIATURA EN COMPUTACION

Régimen de Estudio  
SEMESTRAL

Fecha de Aplicación  
2009-03-27

Editar

**Requisitos Curriculares**

**Distribución de los Requisitos Curriculares**

**Opciones Profesionales**

Módulo Administrativo  
Rivas Sergio  
V-13736933  
Cerrar sesión

CONEST Versión 3.0.0 Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2011

Figura 16. Editar Planes de Estudio.

Se pueden gestionar los datos particulares de los requisitos curriculares de los planes de estudio tales como el tipo de asignatura, el número mínimo de unidades de crédito y el número mínimo de asignaturas que se deben aprobar.

En la Facultad de Ciencias existen diferentes tipos de asignaturas tales como las obligatorias, obligatorias optativas, electivas, complementarias, laboratorios, pasantías, seminarios, TEG y servicio comunitario. Cada licenciatura exige aprobar un número de asignaturas de diferentes tipos para obtener el título de la carrera. Además, dichas asignaturas están distribuidas en 10

semestres. Se creó la tabla *distribucion\_requisito\_plan* en la base de datos para poder distribuir los requisitos curriculares de un plan de estudio por semestre. La Figura 17 muestra la tabla *distribucion\_requisito\_plan*.



Figura 17. Tabla *distribucion\_requisito\_plan*.

El listado de requisitos curriculares distribuidos por semestre muestra el número de asignaturas que se deben aprobar en cada semestre, de los requisitos curriculares de un plan de estudio organizados en filas. En la primera fila se muestra la cabecera, las columnas de la primera fila muestran los 10 semestres de la carrera. A partir de la segunda fila, se muestran los requisitos curriculares, las columnas de estas filas muestran el número de asignaturas requeridas por cada requisito curricular que correspondan con el semestre de la cabecera.

La última columna de las filas de los requisitos curriculares de un plan de estudio, muestran el número total de asignaturas distribuidas del número total de asignaturas que se deben aprobar del requisito curricular correspondiente. La Figura 18 muestra el listado de requisitos curriculares distribuidos por semestre.

Editar Plan de Estudio Lunes, 16 de Abril de 2011, 11:13 p.m.

[Datos del Plan de Estudio](#)

[Requisitos Curriculares](#)

[Distribución de los Requisitos Curriculares](#)

Tipo de Asignatura	I	II	III	IV	V	VI	VII	VIII	IX	X	
COMPLEMENTARIA	0	0	0	0	0	1	0	1	1	0	3/3
ELECTIVA	0	0	0	0	0	1	3	2	2	0	8/8
LABORATORIO	0	0	0	0	0	0	1	0	0	0	1/1
OBLIGATORIA	4	4	4	5	4	0	0	0	0	0	21/21
OBLIGATORIA OPTATIVA	0	0	0	0	0	2	0	0	0	0	2/2
PASANTIA	0	0	0	0	0	0	0	1	0	0	1/1
SEMINARIO	0	0	0	0	0	0	0	0	1	0	1/1
TRABAJO ESPECIAL DE GRADO	0	0	0	0	0	0	0	0	0	1	1/1

[Opciones Profesionales](#)

Figura 18. Listado de Requisitos Curriculares Distribuidos por Semestre.

Se pueden gestionar los datos particulares de los requisitos curriculares de un plan de estudio distribuidos por semestre tales como el tipo de asignatura, el semestre, y el número de asignaturas que se deben aprobar en dicho semestre.

También se pueden gestionar los datos particulares de los requisitos curriculares de las opciones profesionales registradas en un plan de estudio tales como el tipo de asignatura, el número mínimo de unidades de crédito y el número mínimo de asignaturas que se deben aprobar.

La gestión de los requisitos curriculares, la distribución de los requisitos curriculares por semestre y las opciones profesionales de los planes de estudio, se realiza de manera similar a la gestión de los planes de estudio.

El listado de asignaturas muestra todas las asignaturas en la base de datos organizadas en filas. En la primera fila se muestra la cabecera, las columnas de la primera fila muestran la denominación de los atributos más relevantes de las asignaturas. A partir de la segunda fila, se muestran las asignaturas, las columnas de estas filas muestran los datos particulares de cada asignatura que correspondan con los atributos de la cabecera.

La última columna de las filas de las asignaturas, muestran un icono para eliminar la asignatura correspondiente. Al colocar el cursor sobre el icono, se muestra una descripción emergente para informar al usuario sobre la funcionalidad del icono. Los usuarios deberán hacer clic en el icono, y confirmar la acción para eliminar una asignatura de la base de datos.

Las asignaturas del listado se pueden ordenar alfabéticamente de manera descendiente y ascendiente por atributo, mediante unos iconos que se encuentran en cada columna de la cabecera. Luego del listado, se muestra un enlace para crear una asignatura nueva.

Es importante resaltar que existen más de 1900 registros de asignaturas en la base de datos, por este motivo, se utilizó un sistema de paginación para dividir los registros en conjuntos discretos. Esta técnica permite mostrar en el listado un número limitado de asignaturas en varias páginas. Los usuarios pueden navegar a través de las diferentes páginas para mostrar su contenido cuando sea necesario. Además, los usuarios pueden buscar asignaturas por nombre por medio de una barra de búsqueda que se encuentra antes del listado. La Figura 19 muestra el listado de asignaturas.

Inicio Preinscripción Inscripción Solicitudes Plan de Estudio Horarios

Lunes, 16 de Abril de 2011, 11:13 p.m.

**Asignaturas**

**Buscador de Asignaturas**

INGLES I

**Listado de Asignaturas**

Código	Nombre	Departamento	Observaciones	
0001	ELECTIVAS	GEOQUIMICA		X
0002	IDIOMAS	COORDINACION ACADEMICA		X
0003	HUMANIDADES	GEOQUIMICA		X
0004	IDIOMAS Y HUMANIDADES	GEOQUIMICA		X
0005	IDIOMA I	GEOQUIMICA		X
0006	IDIOMA II	COMPUTACION	CODIGOS ANTIGUOS	X
0007	DEPORTES	COORDINACION ACADEMICA	POR MIGRACION	X
0008	IDIOMA Y HUMANIDADES	MATEMATICA	CODIGOS ANTIGUOS	X
0009	ELECTIVAS	BIOLOGIA		X
0010	IDIOMAS Y HUMANIDADES	COMPUTACION		X
0013	PRINCIPIOS DE INGENIERIA QUIMICA	QUIMICA	CODIGOS ANTIGUOS	X
0014	GEOLOGIA GENERAL I	BIOLOGIA	CODIGOS ANTIGUOS	X
0015	GEOLOGIA GENERAL II	COMPUTACION	CODIGOS ANTIGUOS	X
0017	OCEANOGRAFIA	BIOLOGIA	CODIGOS ANTIGUOS	X
0022	PROCESOS UNITARIOS I	BIOLOGIA	CODIGOS ANTIGUOS	X
0023	MICROBIOLOGIA DE ALIMENTOS	BIOLOGIA	CODIGOS ANTIGUOS	X
0026	ALEMAN I	MATEMATICA	CODIGOS ANTIGUOS	X
0027	ALEMAN II	MATEMATICA	CODIGOS ANTIGUOS	X
0028	FRANCES I	COORDINACION ACADEMICA	CODIGOS ANTIGUOS	X
0029	FRANCES II	MATEMATICA	CODIGOS ANTIGUOS	X

1 2 3 4 5 ... [Siguiete >](#) [Última](#)

Para crear una asignatura nueva haga click [aquí](#)

CONEST Versión 3.0.0 Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2011

**Figura 19.** Listado de Asignaturas.

Los usuarios pueden crear una asignatura nueva por medio de un formulario. El formulario muestra todos los atributos de las asignaturas mediante campos de entrada, los cuales deben ser llenados por los usuarios con la información solicitada, que se almacenará para su uso posterior o manipulación.

Los usuarios también pueden seleccionar una asignatura del listado para revisar sus datos particulares tales como el código, el departamento al que pertenece, el nombre y el nombre corto. Además, se muestra toda la información relacionada con las asignaturas tales como los planes de estudio, las prelacones, las convalidaciones y las opciones profesionales, con la finalidad de acceder convenientemente y gestionar las distintas características de las asignaturas en un solo lugar.

Se utilizó un menú de acordeones para organizar de manera estructurada las características de las asignaturas. Los usuarios pueden minimizar y maximizar los acordeones para mostrar su contenido cuando sea necesario. La Figura 20 muestra la página web para editar asignaturas.

Sistema de Gestión Académica

Inicio Preinscripción Inscripción Solicitudes Plan de Estudio Horarios

Editar Asignatura Lunes, 16 de Abril de 2011, 11:13 p.m.

**Datos de la Asignatura**

Código: 6221

Departamento: DEPARTAMENTO DE COMPUTACION

Nombre: APLICACIONES CON LA TECNOLOGIA I Ejemplo: MATEMATICA I

Nombre Corto: APLIC.CON LA TEC.INTERNET Ejemplo: MATEMATICA I

Observaciones:   
 Ejemplo: POR MIGRACION

Editar

**Planes de Estudio**

**Prelaciones**

**Convalidaciones**

**Opciones Profesionales**

CONEST Versión 3.0.0 Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2011

Figura 20. Editar Asignaturas.

Existen asignaturas que se pueden cursar bajo diferentes modalidades en la misma carrera. De este modo, la asignatura Aplicaciones con la Tecnología Internet se puede cursar como obligatoria optativa o como electiva en la carrera de Computación, por ejemplo. Se crearon los campos *tipo\_materia\_id* y *es\_tipo\_principal* en la tabla *materia\_en\_plan* para poder registrar una asignatura varias veces en el mismo plan de estudio con tipos de asignaturas diferentes. Otro caso particular a tomar en cuenta es el de las asignaturas obligatorias y sus convalidaciones, el campo *es\_tipo\_principal* permite determinar cuáles son las asignaturas obligatorias de una carrera y cuáles son las asignaturas por las cuales son convalidadas. La Figura 21 muestra la tabla *materia\_en\_plan*.

```
mysql> desc materia_en_plan;
+-----+-----+-----+-----+-----+-----+
| Field                | Type                | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| carrera_id           | char(10)            | NO   | PRI | NULL    |       |
| plan_nombre          | char(20)            | NO   | PRI | NULL    |       |
| materia_codigo       | varchar(10)         | NO   | PRI | NULL    |       |
| tipo_materia_id      | char(10)            | NO   | PRI | NULL    |       |
| grupo_nota_id        | char(10)            | NO   | MUL | NULL    |       |
| nro_creditos         | int(10) unsigned   | NO   |     | 0       |       |
| nro_horas_teoría     | int(11)            | NO   |     | NULL    |       |
| nro_horas_practica   | int(11)            | NO   |     | NULL    |       |
| nro_horas_laboratorio | int(11)            | NO   |     | NULL    |       |
| ubicacion_en_carrera | int(10) unsigned   | NO   |     | NULL    |       |
| requiere_aula        | char(1)            | NO   |     | NULL    |       |
| requiere_horario     | char(1)            | NO   |     | NULL    |       |
| es_reparable         | char(1)            | NO   |     | NULL    |       |
| es_de_servicio       | char(1)            | NO   |     | 0       |       |
| se_programa_examen   | char(1)            | NO   |     | 0       |       |
| creditos_requisitos  | int(11)            | NO   |     | NULL    |       |
| puede_retirarse      | char(1)            | NO   |     | NULL    |       |
| es_tipo_principal    | char(1)            | NO   |     | 0       |       |
+-----+-----+-----+-----+-----+-----+
18 rows in set (0.00 sec)
```

Figura 21. Tabla materia\_en\_plan.

Se pueden gestionar los datos particulares de las asignaturas registradas en un plan de estudio tales como la carrera y el plan de estudio a los que pertenece, el tipo de asignatura, el método de calificación, el número de unidades de crédito que se obtienen por aprobarla, el número de horas teóricas, el número de horas de práctica y el número de horas de laboratorios que reciben los estudiantes en clases semanales, el semestre, si requiere un aula y un horario, si puede ser reparada, si es de servicio, si se programa un examen final, el número de unidades de créditos aprobados requeridos para cursarla y si puede ser retirada.

También se pueden gestionar los datos particulares de las relaciones de las asignaturas registradas en un plan de estudio tales como la carrera y el plan de estudio a los que pertenece, la asignatura que se debe aprobar con anterioridad para poder cursarla y si dicha asignatura se puede cursar en paralelo.

Además, se pueden gestionar los datos particulares de las convalidaciones de las asignaturas registradas en un plan de estudio tales como la carrera y el plan de estudio a los que pertenece, y la asignatura por la cual es convalidada.

También se pueden gestionar los datos particulares de las asignaturas registradas en las opciones profesionales de un plan de estudio tales como la carrera y el plan de estudio a los que pertenece, la opción profesional y el tipo de asignatura.

La gestión de los planes de estudio, las prelacones, las convalidaciones y las opciones profesionales de las asignaturas registradas en un plan de estudio se realiza de manera similar a la gestión de las asignaturas.

El módulo web mantiene a los usuarios informados sobre el estado de las operaciones que están realizando, a través de retroalimentación adecuada dentro de un plazo razonable de tiempo.

Los formularios están diseñados para comprobar que los campos obligatorios están llenos. Si los campos no lo están, se muestra una notificación emergente que indica la obligatoriedad, en la esquina superior derecha de cada uno de ellos. Además, no se puede enviar un formulario al servidor hasta que todos los campos obligatorios estén llenos. La Figura 22 muestra la notificación emergente para indicar los campos obligatorios de los formularios.



The screenshot shows a web interface for creating a study plan. The title is 'Crear Plan de Estudio' and the date/time is 'Lunes, 16 de Abril de 2011, 11:13 p.m.'. On the left, there is a user profile for 'Módulo Administrativo', 'Rivas Sergio', and 'V-13736933'. The main form is titled 'Datos del Plan de Estudio' and contains the following fields: 'Nombre' (empty), 'Carrera' (dropdown menu), and 'Régimen de Estudio' (set to 'SEMESTRAL'). Two red error messages are overlaid on the form, both stating '\* Este campo es requerido' (This field is required), one pointing to the 'Nombre' field and the other to the 'Carrera' dropdown.

Figura 22. Campos Obligatorios.

Los diálogos muestran mensajes complementarios sobre la información llenada por los usuarios en los campos de entrada de un formulario. Se utilizan en su mayoría con las listas desplegables, en las cuales las opciones disponibles dependen de los valores de otros campos de entrada, para informar a los usuarios que no hay ninguna opción disponible para los valores seleccionados. La Figura 23 muestra el dialogo emergente utilizado junto con las listas desplegables en cascada.



The screenshot shows a web interface for creating a study plan. The title is 'Plan de Estudio' and the date/time is 'Lunes, 16 de Abril de 2011, 11:13 p.m.'. On the left, there is a user profile for 'Módulo Administrativo', 'Rivas Sergio', and 'V-13736933'. The main form is titled 'Datos del Plan de Estudio' and contains the following fields: 'Plan de Estudio' (set to 'PLAN COMPUTACION'), 'Código' (set to '6221'), 'Opción Profesional' (dropdown menu set to 'APLICACIONES EN INTERNET'), and 'Tipo de Asignatura' (dropdown menu). An error dialog box is overlaid on the form, titled 'Tipo de Asignatura' and containing the message: 'No hay ningún tipo de asignatura que registrar en este plan de estudio.' (There is no type of assignment to register in this study plan.)

Figura 23. Diálogos Emergentes.

Los mensajes de confirmación y error están expresados en un lenguaje sencillo (sin código), con palabras, frases y expresiones que indiquen con precisión el resultado de la operación realizada,

así como también pueden sugerir una solución constructiva en caso de que sea necesario. Las Figuras 24 y 25 muestran los mensajes de confirmación y error respectivamente.

Planes de Estudio Lunes, 16 de Abril de 2011, 11:13 p.m.

El plan de estudio se activó satisfactoriamente.

Listado de Planes de Estudio

Nombre	Carrera	Régimen de Estudio	Fecha de Aplicación	
PLAN BIOLOGIA	LICENCIATURA EN BIOLOGIA	SEMESTRAL	2009-03-27	A X

Figura 24. Mensajes de Confirmación.

Planes de Estudio Lunes, 16 de Abril de 2011, 11:13 p.m.

Disculpe. No se puede eliminar el plan de estudio activo.

Listado de Planes de Estudio

Nombre	Carrera	Régimen de Estudio	Fecha de Aplicación	
PLAN BIOLOGIA	LICENCIATURA EN BIOLOGIA	SEMESTRAL	2009-03-27	A X

Figura 25. Mensajes de Error.

### 4.2.3 Codificación

Los planes de estudio se representan en el módulo web para la gestión de planes de estudio en el sistema CONEST por medio del modelo *plan.rb*, en el cual se definen las propiedades generales de los planes de estudio tales como la clave primaria, las asociaciones entre modelos, las validaciones, las retrollamadas y los métodos de la lógica de negocio. La Figura 26 muestra el modelo *plan.rb*.

```

1 class Plan < ActiveRecord::Base
2   set_primary_keys :nombre, :carrera_id
3
4   belongs_to :tipo_regimen_del_plan
5   belongs_to :carrera
6   belongs_to :plan_activo,
7     :class_name => 'PlanActivo',
8     :foreign_key => ['nombre', 'carrera_id']
9
10  validates :nombre, :carrera_id, :tipo_regimen_del_plan_id, :presence => true
11  validates :nombre, :length => {:maximum => 20}
12  validates :carrera_id, :tipo_regimen_del_plan_id, :length => {:maximum => 10}
13  validates :nombre, :format => {:with => /^[A-ZÁÉÍÓÚÑ ]+$/}
14
15  before_validation do |plan|
16    plan.nombre.upcase!
17  end
18
19  before_destroy do |plan|
20    plan_activo = PlanActivo.where("carrera_id = ?", plan.carrera_id).first
21
22    if plan_activo
23      if plan.nombre == plan_activo.plan_nombre
24        errors[:base] = "Disculpe. No se puede eliminar el plan de estudio activo."
25        return false
26      end
27    end
28  end
29 end

```

Figura 26. Modelo plan.rb.

Para mostrar el listado de planes de estudio, primero se buscan todos los registros en la base de datos, en el método *index* del controlador *plan\_estudio\_controller.rb*.

Luego, en la vista *index.html.haml*, se determina si la variable donde se guardó el resultado de la búsqueda de los planes de estudio contiene algún registro o si está vacía. Si la variable no está vacía, se itera sobre el conjunto de resultados, y se muestra en una tabla los datos particulares así como los enlaces para editar, activar y eliminar el plan de estudio correspondiente. Si la variable está vacía, se muestra un mensaje indicando que no se encontró ningún registro y un enlace para crear un plan de estudio nuevo.

El estilo de la tabla y la funcionalidad para ordenar alfabéticamente las columnas se obtienen a través de la clase *tablesorter*. Además, se enlaza la hoja de estilo *plan\_estudio.css.scss* para incorporar las reglas de presentación requeridas por la vista. La Figura 27 muestra la vista *index.html.haml* (Ver Figura 15).

```
1 = stylesheet_link_tag "plan_estudio"
2
3 - if notice || alert
4   = render :partial => "notificacion"
5
6 %fieldset
7   %legend Listado de Planes de Estudio
8   - if @planes.size > 0
9     %table.tablesorter
10      %thead
11        %tr
12          %th Nombre
13          %th Carrera
14          %th Régimen de Estudio
15          %th Fecha de Aplicación
16      %tbody
17        - @planes.each do |plan|
18          %tr
19            %td= link_to plan.nombre, :action => "editar", :nombre => plan.nombre, :carrera_id => plan.carrera_id
20            %td= plan.carrera.nombre
21            %td= plan.tipo_regimen_del_plan.descripcion
22            %td= plan.fecha_aplicacion
23            %td
24              - if plan.plan_activo
25                %b{:class => "activo"} A
26              - else
27                = link_to "A", {:action => "activar", :carrera_id => plan.carrera_id, :plan_nombre => plan.nombre},
28                  :title => "Activar", :class => "inactivo"
29            %td= link_to "X", {:action => "eliminar", :nombre => plan.nombre, :carrera_id => plan.carrera_id},
30                  :title => "Eliminar", :confirm => "¿Está seguro de eliminar el plan de estudio?", :class => "eliminar"
31      %p
32        Para crear un plan de estudio nuevo haga click
33        = link_to "aquí", :action => "crear"
34  - else
35    %p
36      No hay ningún plan de estudio activo. Para crear uno nuevo haga click
37      = link_to "aquí", :action => "crear"
```

Figura 27. Vista *plan\_estudio/index.html.haml*.

Para crear un plan de estudio, primero se crea una instancia nueva, en el método *crear* del controlador *plan\_estudio\_controller.rb*.

Luego, en la vista *crear.html.haml*, se muestra un formulario en el cual se especifica la variable donde se guardará la información de los campos de entrada, la acción que se ejecutará cuando se envíe el formulario al servidor, el identificador y la vista parcial de los campos de entrada.

Los atributos de los planes de estudio se muestran en la vista parcial `_formulario.html.haml` mediante campos de entrada. Los campos de entrada se llenan dependiendo de la acción que se esté realizando, de este modo, los usuarios llenan todos los campos de entrada del formulario cuando estén creando un plan de estudio nuevo, pero no podrán modificar la información de los campos de entrada correspondientes a la clave primaria cuando estén editando un plan de estudio existente.

El estilo de las notificaciones emergentes y la funcionalidad para indicar los campos obligatorios de los formularios se obtienen a través de la clase `validate[required]`. La Figura 28 muestra la vista parcial `_formulario.html.haml` (Ver Figura 16).

```

1 - if params[:action] == "crear" || params[:action] == "guardar"
2   %p
3     = f.label :nombre
4     = f.text_field :nombre, :class => "validate[required] text-input mayusculas"
5     Ejemplo: PLAN COMPUTACION
6   %p
7     = f.label :carrera_id, "Carrera"
8     = f.collection_select :carrera_id, Carrera.where("visible = ?", "1").all, :id, :nombre,
9     {:include_blank => true}, {:class => "validate[required]"}
10  %p
11  = f.label :tipo_regimen_del_plan_id, "Régimen de Estudio"
12  = f.text_field :tipo_regimen_del_plan_id, :value => "SEMESTRAL", :readonly => true
13  = f.hidden_field :tipo_regimen_del_plan_id, :value => "SEM"
14  %p
15  = f.label :fecha_aplicacion, "Fecha de Aplicación"
16  = f.text_field :fecha_aplicacion
17  %p
18  = f.submit "Crear"
19
20 - if params[:action] == "editar" || params[:action] == "actualizar"
21  %p
22  = f.label :nombre
23  = f.text_field :nombre, :readonly => true
24  %p
25  = f.label :carrera_id, "Carrera"
26  = f.text_field :carrera_id, :value => @plan.carrera.nombre, :readonly => true
27  = f.hidden_field :carrera_id, :value => @plan.carrera_id
28  %p
29  = f.label :tipo_regimen_del_plan_id, "Régimen de Estudio"
30  = f.text_field :tipo_regimen_del_plan_id, :value => "SEMESTRAL", :readonly => true
31  = f.hidden_field :tipo_regimen_del_plan_id, :value => "SEM"
32  %p
33  = f.label :fecha_aplicacion, "Fecha de Aplicación"
34  = f.text_field :fecha_aplicacion
35  %p
36  = f.submit "Editar"

```

**Figura 28.** Vista Parcial `plan_estudio/_formulario.html.haml`.

Para guardar un plan de estudio nuevo, primero se guarda la información de los campos de entrada del formulario, en el método `guardar` del controlador `plan_estudio_controller.rb`. Luego, se intenta almacenar el registro en la base de datos. Si ninguna de las validaciones falla y la operación de inserción en la base de datos se realiza exitosamente, se redirige a la vista `index.html.haml` y se muestra un mensaje de confirmación. En caso contrario, se muestra un mensaje de error. La Figura 29 muestra el método `guardar`.

```

169 def guardar
170   @plan = Plan.new(params[:plan])
171
172   if @plan.save
173     flash[:notice] = "El plan de estudio se creó satisfactoriamente."
174     redirect_to :action => "index"
175   else
176     @plan
177     render :crear
178   end
179 end

```

Figura 29. Método plan\_estudio/guardar.

Los mensajes de confirmación y error se muestran en la vista parcial `_notificacion.html.haml`. Los estilos de los mensajes se obtienen a través de las clases `ui-state-highlight.ui-corner-all` y `ui-state-error.ui-corner-all` respectivamente. La Figura 30 muestra la vista parcial `_notificacion.html.haml`.

```

1 - if notice
2   .ui-widget
3     .ui-state-highlight.ui-corner-all{:style => "margin: 8px; padding: 5px;}
4     %p
5       %span{:class => "ui-icon ui-icon-info", :style => "float: left; margin-right: .3em;}
6       = notice
7
8 - if alert
9   .ui-widget
10    .ui-state-error.ui-corner-all{:style => "margin: 8px; padding: 5px;}
11    %p
12      %span{:class => "ui-icon ui-icon-alert", :style => "float: left; margin-right: .3em;}
13      = alert

```

Figura 30. Vista Parcial plan\_estudio/\_notificacion.html.haml.

Para editar un plan de estudio, primero se busca el registro seleccionado del listado en la base de datos, en el método `editar` del controlador `plan_estudio_controller.rb`.

Luego, en la vista `editar.html.haml`, se muestra un formulario en el cual se especifica la variable donde se guardaron los datos particulares del plan de estudio que se mostrarán en los campos de entrada, la acción que se ejecutará cuando se envíe el formulario al servidor, el identificador y la vista parcial de los campos de entrada.

Para actualizar un plan de estudio existente, primero se busca el registro en la base de datos, en el método `actualizar` del controlador `plan_estudio_controller.rb`. Luego, se intenta almacenar los cambios en el registro en la base de datos. Si ninguna de las validaciones falla y la operación de actualización en la base de datos se realiza exitosamente, se redirige a la vista `index.html.haml` y se muestra un mensaje de confirmación. En caso contrario, se muestra un mensaje de error. La Figura 31 muestra un fragmento del método `actualizar`.

```

181 def actualizar
182   @plan = Plan.where("nombre = ? AND carrera_id = ?", params[:plan][:nombre], params[:plan][:carrera_id]).first
183
184   if @plan.update_attributes(params[:plan])
185     flash[:notice] = "El plan de estudio se editó satisfactoriamente."
186     redirect_to :action => "index"
187   else
188     @plan
189     render :editar
190   end
191 end

```

Figura 31. Método plan\_estudio/actualizar.

Para eliminar un plan de estudio existente, primero se busca el registro en la base de datos, en el método *eliminar* del controlador *plan\_estudio\_controller.rb*. Luego, se intenta borrar el registro de la base de datos. Si ninguna de las retrollamadas falla y la operación de eliminación en la base de datos se realiza exitosamente, se muestra un mensaje de confirmación. En caso contrario, se muestra un mensaje de error. La Figura 32 muestra el método *eliminar*.

```

343 def eliminar
344   plan = Plan.where("nombre = ? AND carrera_id = ?", params[:nombre], params[:carrera_id]).first
345
346   if plan.destroy
347     flash[:notice] = "El plan de estudio se eliminó satisfactoriamente."
348     redirect_to :action => "index"
349   else
350     flash[:alert] = "Disculpe. Se produjo un error al eliminar el plan de estudio."
351     flash[:alert] = plan.errors[:base][0] if plan.errors
352     redirect_to :action => "index"
353   end
354 end

```

Figura 32. Método *plan\_estudio/eliminar*.

En el método *editar* del controlador *plan\_estudio\_controller.rb*, también se buscan todos los requisitos curriculares, el número de asignaturas que los estudiantes deben aprobar en cada semestre de la carrera por requisito curricular y las opciones profesionales de un plan de estudio. La Figura 33 muestra un fragmento del método *editar*.

```

13 def editar
14   @requisitos_graduacion_plan = RequisitoGraduacionPlan.where("carrera_id = ? AND plan_nombre = ?",
15   @plan.carrera_id, @plan.nombre).all
16   @requisitos_graduacion_plan.each do |requisito_graduacion_plan|
17     case requisito_graduacion_plan.tipo_materia_id
18     when "0"
19       @obligatorias_1 = DistribucionRequisitoPlan.buscar_numero_asignaturas(@plan.carrera_id, @plan.nombre, "0", "1")
20       @obligatorias_2 = DistribucionRequisitoPlan.buscar_numero_asignaturas(@plan.carrera_id, @plan.nombre, "0", "2")
21       @obligatorias_3 = DistribucionRequisitoPlan.buscar_numero_asignaturas(@plan.carrera_id, @plan.nombre, "0", "3")
22       @obligatorias_4 = DistribucionRequisitoPlan.buscar_numero_asignaturas(@plan.carrera_id, @plan.nombre, "0", "4")
23       @obligatorias_5 = DistribucionRequisitoPlan.buscar_numero_asignaturas(@plan.carrera_id, @plan.nombre, "0", "5")
24       @obligatorias_6 = DistribucionRequisitoPlan.buscar_numero_asignaturas(@plan.carrera_id, @plan.nombre, "0", "6")
25       @obligatorias_7 = DistribucionRequisitoPlan.buscar_numero_asignaturas(@plan.carrera_id, @plan.nombre, "0", "7")
26       @obligatorias_8 = DistribucionRequisitoPlan.buscar_numero_asignaturas(@plan.carrera_id, @plan.nombre, "0", "8")
27       @obligatorias_9 = DistribucionRequisitoPlan.buscar_numero_asignaturas(@plan.carrera_id, @plan.nombre, "0", "9")
28       @obligatorias_10 = DistribucionRequisitoPlan.buscar_numero_asignaturas(@plan.carrera_id, @plan.nombre, "0", "10")
29
30       @MIN_MATERIAS_OBLIGATORIAS = requisito_graduacion_plan.min_materias
31       @obligatorias_distribuidas = 0
32
33       (1..10).each do |i|
34         @obligatorias_distribuidas += instance_variable_get("@obligatorias_#{@i}").
35         numero_materias if instance_variable_get("@obligatorias_#{@i}")
36       end
37     end
38   end
39 end
40
41 end
42
43 @opciones_profesionales = OpcionPlan.where("carrera_id = ? AND plan_nombre = ?", @plan.carrera_id, @plan.nombre).all
44 end

```

Figura 33. Método *plan\_estudio/editar*.

El algoritmo para buscar el número de asignaturas obligatorias optativas, electivas, complementarias, laboratorios, pasantías, seminarios y TEG que se deben aprobar en cada semestre, es similar al algoritmo para buscar el número de asignaturas obligatorias, que se muestra en la figura anterior.

En la vista *editar.html.haml*, se muestran todos los listados relacionados con los planes de estudio tales como el listado de requisitos curriculares, el listado de requisitos curriculares distribuidos por semestre y el listado de opciones profesionales, organizados en un menú de acordeones. La Figura 34 muestra un fragmento de la vista *editar.html.haml* (Ver Anexo I).

```

6 %fieldset
7   %legend{:class => "acordeon principal"} Datos del Plan de Estudio
8   .contenido-acordeon
9     = form_for @plan, :url => {:action => "actualizar"}, :html => {:id => "formulario"} do |f|
10     = render :partial => "formulario", :locals => {:f => f}
11
12 %fieldset
13   %legend{:class => "acordeon secundario"} Requisitos Curriculares
14   .contenido-acordeon
15   - if @requisitos_graduacion_plan.size > 0
16     %table.tablesorter
17       %thead
18         %tr
19           %th{:id => "tipo_asignatura"} Tipo de Asignatura
20           %th{:id => "min_creditos"} Min Créditos
21           %th{:id => "min_asignaturas"} Min Asignaturas
22       %tbody
23         - @requisitos_graduacion_plan.each do |requisito_graduacion_plan|
24           %tr
25             %td= link_to requisito_graduacion_plan.tipo_materia.descripcion,
26               :action => "editar_requisito_curricular", :carrera_id => @plan.carrera_id,
27               :plan_nombre => @plan.nombre, :tipo_materia_id => requisito_graduacion_plan.tipo_materia_id
28             %td= requisito_graduacion_plan.min_creditos
29             %td= requisito_graduacion_plan.min_materias
30             %td= link_to "X", {:action => "eliminar_requisito_curricular", :carrera_id => @plan.carrera_id,
31               :plan_nombre => @plan.nombre, :tipo_materia_id => requisito_graduacion_plan.tipo_materia_id},
32               :title => "Eliminar", :confirm => "¿Está seguro de eliminar el requisito curricular?", :class => "eliminar"
33           %p
34             Para crear un requisito curricular nuevo haga click
35             = link_to "aquí", :action => "crear_requisito_curricular", :carrera_id => @plan.carrera_id,
36               :plan_nombre => @plan.nombre
37         - else
38           %p
39             No hay ningún requisito curricular activo. Para crear uno nuevo haga click
40             = link_to "aquí", :action => "crear_requisito_curricular", :carrera_id => @plan.carrera_id,
41               :plan_nombre => @plan.nombre

```

Figura 34. Vista *plan\_estudio/editar.html.haml*.

El estilo de los acordeones y la funcionalidad para minimizar y maximizar su contenido se obtienen a través de las clases *acordeon* y *contenido-acordeon*, las cuales se desarrollaron en el script *plan\_estudio.js.coffee*. La Figura 35 muestra un fragmento del script *plan\_estudio.js.coffee*.

```

22 $( ".principal" ).next().show()
23
24 $( ".secundario" ).addClass("subrayar")
25
26 $( ".secundario" ).next().hide()
27
28 $( ".principal" ).click ->
29   $( ".formErrorContent" ).hide()
30   $( ".formErrorArrow" ).hide()
31   if $( this ).next().is(":hidden") == true
32     $( ".formErrorContent" ).show()
33     $( ".formErrorArrow" ).show()
34
35 $( ".secundario" ).click ->
36   $( ".formErrorContent" ).hide()
37   $( ".formErrorArrow" ).hide()
38
39 $( ".acordeon" ).click ->
40   $( ".acordeon" ).addClass("subrayar")
41   $( ".contenido-acordeon" ).slideUp()
42   if $( this ).next().is(":hidden") == true
43     $( this ).removeClass("subrayar")
44     $( this ).next().slideDown()

```

Figura 35. Script *plan\_estudio.js.coffee*.

El listado de requisitos curriculares distribuidos por semestre, se muestra en la vista parcial `_distribucion_requisitos_curriculares.html.haml`. Se itera sobre el conjunto de requisitos curriculares del plan de estudio, y dentro de un ciclo, se determina la distribución del número de asignaturas del requisito curricular correspondiente en cada semestre de la carrera. Además, se muestra el número total de asignaturas distribuidas del número mínimo de asignaturas que se deben aprobar en dicho requisito curricular. La Figura 36 muestra un fragmento de la vista parcial `_distribucion_requisitos_curriculares.html.haml`.

```

1 - @requisitos_graduacion_plan.each do |requisito_graduacion_plan|
2   - if requisito_graduacion_plan.tipo_materia_id != "SC"
3     %tr
4       %td= requisito_graduacion_plan.tipo_materia.descripcion
5
6       - (1..10).each do |i|
7         - case requisito_graduacion_plan.tipo_materia_id
8           - when "O"
9             - obligatorias = instance_variable_get("@obligatorias_#{i}")
10            %td
11              - if obligatorias
12                = link_to obligatorias.numero_materias, (:action => "editar_distribucion", :carrera_id => @plan.carrera_id,
13                  :plan_nombre => @plan.nombre, :tipo_materia_id => "O", :ubicacion_en_carrera => i), :title => "Redistribuir"
14              - elsif @MIN_MATERIAS_OBLIGATORIAS > @obligatorias_distribuidas
15                = link_to "0", (:action => "crear_distribucion", :carrera_id => @plan.carrera_id,
16                  :plan_nombre => @plan.nombre, :tipo_materia_id => "O", :ubicacion_en_carrera => i), :title => "Distribuir"
17              - else
18                0
19            .
20            .
21          .
22        %td
23          - case requisito_graduacion_plan.tipo_materia_id
24            - when "O"
25              = @obligatorias_distribuidas.to_s + "/" + requisito_graduacion_plan.min_materias.to_s
26            - when "OO"
27              = @obligatorias_optativas_distribuidas.to_s + "/" + requisito_graduacion_plan.min_materias.to_s
28            - when "E"
29              = @selectivas_distribuidas.to_s + "/" + requisito_graduacion_plan.min_materias.to_s
30            - when "C"
31              = @complementarias_distribuidas.to_s + "/" + requisito_graduacion_plan.min_materias.to_s
32            - when "L"
33              = @laboratorios_distribuidos.to_s + "/" + requisito_graduacion_plan.min_materias.to_s
34            - when "P"
35              = @pasantias_distribuidas.to_s + "/" + requisito_graduacion_plan.min_materias.to_s
36            - when "S"
37              = @seminarios_distribuidos.to_s + "/" + requisito_graduacion_plan.min_materias.to_s
38            - when "T"
39              = @teg_distribuidos.to_s + "/" + requisito_graduacion_plan.min_materias.to_s

```

Figura 36. Vista Parcial `plan_estudio/_distribucion_requisitos_curriculares.html.haml`.

El algoritmo para determinar la distribución de las asignaturas obligatorias optativas, electivas, complementarias, laboratorios, pasantías, seminarios y TEG en cada semestre, es similar al algoritmo para determinar la distribución de las asignaturas obligatorias, que se muestra en la figura anterior.

Los tipos de asignaturas se restringen cuando se crea un requisito curricular nuevo en un plan de estudio, para prevenir que se registre un tipo de asignatura varias veces en el mismo plan de estudio. La Figura 37 muestra el método `crear_requisito_curricular`.

```

382 def crear_requisito_curricular
383   @titulo_pagina = "Crear Requisito Curricular"
384   @requisito_graduacion_plan = RequisitoGraduacionPlan.new
385
386   @carrera = Carrera.where("id = ?", params[:carrera_id]).first
387   @plan = Plan.where("nombre = ? AND carrera_id = ?", params[:plan_nombre], params[:carrera_id]).first
388   requisito_graduacion_plan = RequisitoGraduacionPlan.where("carrera_id = ? AND plan_nombre = ?",
389     @plan.carrera_id, @plan.nombre).all
390   tipo_materia_id = requisito_graduacion_plan.map(& :tipo_materia_id)
391   tipo_materia_id << "I" << "M" << "PCI"
392   @tipo_materia = TipoMateria.where("id NOT IN (?)", tipo_materia_id).all
393 end

```

Figura 37. Método plan\_estudio/crear\_requisito\_curricular.

Luego, en la vista *crear\_requisito\_curricular.html.haml*, se determina si la variable donde se guardó el resultado de la búsqueda de los tipos de asignaturas contiene algún elemento o si está vacía. Si no está vacía, se muestra un formulario, en el cual se especifica la variable donde se guardará la información de los campos de entrada, la acción que se ejecutará cuando se envíe el formulario al servidor, el identificador y la vista parcial de los campos de entrada. Si la variable está vacía, se muestra un mensaje indicando que no se encontró ningún elemento y un enlace para regresar a la vista *editar.html.haml* (Ver Anexo II).

Antes de guardar un requisito curricular en la base de datos, se verifica que el número mínimo de unidades de crédito y el número mínimo de asignaturas del requisito curricular no sean iguales a cero simultáneamente. Antes de actualizar un requisito curricular en la base de datos, también se verifica que el número mínimo de asignaturas que se deben aprobar no sea menor que el número total de asignaturas distribuidas en los 10 semestres de la carrera. La Figura 38 muestra un fragmento del modelo *requisito\_graduacion\_plan.rb*.

```

13 validate :minimo_asignaturas
14 validate :distribucion_requisito_curricular, :on => :update
15
16 def minimo_asignaturas
17   if min_creditos == 0 && min_materias == 0
18     errors.add(:min_creditos, "debe ser mayor que cero cuando el mínimo de materias sea igual a cero")
19   end
20 end
21
22 def distribucion_requisito_curricular
23   distribucion_requisito_plan = DistribucionRequisitoPlan.where("carrera_id = ? AND plan_nombre = ? AND tipo_materia_id = ?",
24     carrera_id, plan_nombre, tipo_materia_id).all
25   asignaturas_distribuidas = 0
26
27   if distribucion_requisito_plan.size > 0
28     distribucion_requisito_plan.each do |distribucion_requisito_plan|
29       asignaturas_distribuidas += distribucion_requisito_plan.numero_materias
30     end
31   end
32
33   if min_materias < asignaturas_distribuidas
34     errors.add(:min_materias, "no puede ser menor que el número de asignaturas distribuidas")
35   end
36 end

```

Figura 38. Modelo requisito\_graduacion\_plan.rb

Además, antes de guardar o actualizar la distribución de un requisito curricular en la base de datos, se verifica que el número total de asignaturas distribuidas en los 10 semestres de la carrera no sea mayor que el número mínimo de asignaturas que se deben aprobar en el requisito curricular correspondiente, en el modelo *distribucion\_requisito\_plan.rb*.

Los algoritmos para listar, crear, guardar, editar, actualizar y eliminar los requisitos curriculares, la distribución de los requisitos curriculares por semestre y las opciones profesionales de los planes de estudio, así como los requisitos curriculares de las opciones profesionales son similares a los algoritmos para listar, crear, guardar, editar, actualizar y eliminar los planes de estudio.

Las asignaturas se representan en el módulo web para la gestión de planes de estudio en el sistema CONEST por medio del modelo *materia.rb*, en el cual se definen las propiedades generales de las asignaturas tales como la clave primaria, las asociaciones entre modelos, las validaciones, las retrollamadas y los métodos de la lógica de negocio.

Para mostrar el listado de asignaturas, primero se buscan todos los registros en la base de datos en conjuntos (de 50 asignaturas cada uno), en el método *index* del controlador *asignatura\_controller.rb*.

Luego, en la vista *index.html.haml*, se determina si la variable donde se guardó el resultado de la búsqueda de las asignaturas contiene algún registro o si está vacía. Si la variable no está vacía, se itera sobre el primer conjunto de resultados, y se muestra en una tabla los datos particulares así como los enlaces para editar y eliminar la asignatura correspondiente. Luego de la tabla, se muestran los enlaces para navegar a través de los diferentes conjuntos de asignaturas. Si la variable está vacía, se muestra un mensaje indicando que no se encontró ningún registro y un enlace para crear una asignatura nueva.

El estilo de los enlaces y la funcionalidad para mostrar el contenido de los diferentes conjuntos de asignaturas en el listado se obtienen a través de la gema Kaminari.

Para buscar una asignatura por nombre, primero se buscan todos los registros cuyo nombre contenga la cadena de caracteres que el usuario ingresó en la barra de búsqueda, de manera asíncrona en la base de datos a medida que vaya tecleando, en el método *autocomplete* del controlador *asignatura\_controller.rb*. Luego, se muestran las asignaturas en un listado emergente debajo de la barra de búsqueda. La Figura 39 muestra el método *autocomplete*.

```
62 def autocomplete
63   materias = Materia.find(:all, :conditions => ["nombre LIKE ?", "%#{params[:term]}%"])
64   materias_json = []
65
66   materias.each do |materia|
67     materias_json << {:id => materia.codigo, :value => materia.codigo + " " + materia.nombre}
68   end
69
70   respond_to do |format|
71     format.json {render :json => materias_json.to_json}
72   end
73 end
```

Figura 39. Método asignatura/autocomplete.

El estilo del listado emergente y la funcionalidad para buscar asincrónicamente las asignaturas en la base de datos se obtienen a través de la librería jQuery, la cual se implementó en el script *asignatura.js.coffee*. La Figura 40 muestra un fragmento del script *asignatura.js.coffee*.

```
272   if conest.es_accion("index")
273     $("#materia_nombre").autocomplete
274       delay: 10,
275       minLength: 3,
276       source: "asignatura/autocomplete",
277       search: -> $("#materia_codigo").val(""),
278       select: (event, ui) ->
279         $("#materia_nombre").val(ui.item.value)
280         $("#materia_codigo").val(ui.item.id)
281
282     $("#materia_nombre").change ->
283       if $(this).val().length < 3
284         $("#materia_codigo").val("")
285
286     $("#formulario").submit ->
287       if !$("#materia_nombre").val() || !$("#materia_codigo").val()
288         $("#dialogo").dialog("option", "title", "Buscador de Asignaturas")
289         $("#dialogo").html('<p style="text-align:left;"><span class="ui-icon ui-icon-alert"
290           style="float:left;margin-right:3px" />Nombre inválido.</p>')
291         $("#dialogo").dialog("open")
292       return false
```

Figura 40. Script *asignatura.js.coffee* (Autocomplete).

Posteriormente, en el método *buscar\_asignatura* del controlador *asignatura\_controller.rb*, se determina si la asignatura que seleccionó el usuario del listado emergente es válida o si es inválida. Si la asignatura es válida, se redirige a la vista *editar.html.haml*. Si la asignatura no es válida, se muestra un mensaje de error.

En el método *editar* del controlador *asignatura\_controller.rb*, se busca la asignatura seleccionada del listado en la base de datos, así como los planes de estudio y las opciones profesionales en los que está registrada, las prelacones y las convalidaciones.

En la vista *editar.html.haml*, se muestra un formulario en el cual se especifica la variable donde se guardaron los datos particulares de la asignatura que se mostrarán en los campos de entrada, la acción que se ejecutará cuando se envíe el formulario al servidor, el identificador y la vista parcial de los campos de entrada.

Además, se muestran los listados relacionados con las asignaturas tales como el listado de planes de estudio, el listado de prelacones, el listado de convalidaciones y el listado de opciones profesionales, organizados en un menú de acordeones (Ver Anexo III).

Las carreras se restringen cuando se crea una prelación de una asignatura, para cerciorarse de que solo se puedan registrar prelacones en los planes de estudio en los cuales está registrada la asignatura. La Figura 41 muestra el método *crear\_prelacion*.

```

194 def crear_prelacion
195   @titulo_pagina = "Crear Prelación"
196   @prelacion = Prelacion.new
197
198   materia_en_plan = MateriaEnPlan.where("materia_codigo = ?", params[:materia_codigo]).all
199   materia_en_plan = materia_en_plan.map(& :carrera_id)
200   materia_en_plan << "!"@#"
201   @carrera = Carrera.where("id IN (?)", materia_en_plan).all
202   @materia = Materia.where("codigo = ?", params[:materia_codigo]).first
203 end

```

Figura 41. Método asignatura/crear\_prelacion.

Luego, en la vista *crear\_prelacion.html.haml*, se determina si la variable donde se guardó el resultado de la búsqueda de las carreras contiene algún elemento o si está vacía. Si no está vacía, se muestra un formulario, en el cual se especifica la variable donde se guardará la información de los campos de entrada, la acción que se ejecutará cuando se envíe el formulario al servidor, el identificador y la vista parcial de los campos de entrada. Si la variable está vacía, se muestra un mensaje indicando que no se encontró ningún elemento y un enlace para regresar a la vista *editar.html.haml*.

La lista desplegable de planes de estudio del formulario mencionado anteriormente se llena dinámicamente dependiendo del valor de la lista desplegable de carreras. Se buscan todos los planes de estudio en los que está registrada la asignatura que pertenezcan a la carrera que el usuario seleccione en la lista desplegable, de manera asíncrona en la base de datos cuando el usuario seleccione una opción, en el método *buscar\_planes\_estudio\_asignatura* del controlador *asignatura\_controller.rb*. La Figura 42 muestra el método *buscar\_planes\_estudio\_asignatura*.

```

256 def buscar_planes_estudio_asignatura
257   materia_en_plan = MateriaEnPlan.where("carrera_id = ? AND materia_codigo = ?",
258     params[:carrera_id], params[:materia_codigo]).all
259   materia_en_plan = materia_en_plan.map(& :plan_nombre)
260
261   planes = Plan.where("nombre IN (?)", materia_en_plan).all
262   planes_json = []
263
264   planes.each do |plan|
265     planes_json << {:nombre => plan.nombre}
266   end
267
268   respond_to do |format|
269     format.json {render :json => planes_json.to_json}
270   end
271 end

```

Figura 42. Método asignatura/buscar\_planes\_estudio\_asignatura.

La funcionalidad para buscar asíncronamente los planes de estudio en la base de datos se obtiene a través de la librería jQuery, la cual se implementó en el script *asignatura.js.coffee*. Además, se determina si la variable donde se guardó el resultado de la búsqueda de los planes de estudio contiene algún elemento o si está vacía. Si la variable no está vacía, se llenan las opciones de la lista desplegable con los planes de estudio. Si la variable está vacía, se muestra un mensaje indicando que no se encontró ningún plan de estudio para la carrera seleccionada en un dialogo emergente. La Figura 43 muestra otro fragmento del script *asignatura.js.coffee*.

```

98     $("#dialogo").dialog
99     autoOpen: false
100     hide: "highlight"
101     show: "highlight"
102
103     $("#prelacion_carrera_id").change ->
104     $("#prelacion_plan_nombre").empty()
105     $("#prelacion_materia_codigo_requisito").empty()
106     if $(this).val().length > 0
107     $.ajax
108     url: "buscar_planes_estudio_asignatura",
109     type: "GET",
110     data: {carrera_id: $("#prelacion_carrera_id").val(), materia_codigo: $("#prelacion_materia_codigo").val()},
111     dataType: "json",
112     success: (planes) ->
113     for i in [1..planes.length]
114     $("#prelacion_plan_nombre").append("<option>" + planes[i-1].nombre + "</option>")
115     $.ajax
116     url: "buscar_requisito_prelacion",
117     type: "GET",
118     data: {carrera_id: $("#prelacion_carrera_id").val(), plan_nombre: $("#prelacion_plan_nombre option:selected").
119     text(), materia_codigo: $("#prelacion_materia_codigo").val()},
120     dataType: "json",
121     success: (asignaturas) ->
122     if asignaturas.length > 0
123     for i in [1..asignaturas.length]
124     $("#prelacion_materia_codigo_requisito").
125     append("<option value=" + asignaturas[i-1].codigo + ">" + asignaturas[i-1].nombre + "</option>")
126     else
127     $("#dialogo").dialog("option", "title", "Requisito")
128     $("#dialogo").html('<p><span class="ui-icon ui-icon-alert" style="float:left;margin-right:3px" />
129     No hay ninguna asignatura disponible en este plan de estudio.</p>')
130     $("#dialogo").dialog("open")

```

**Figura 43.** Script asignatura.js.coffee (Listas Desplegables).

Por último, los algoritmos para listar, crear, guardar, editar, actualizar y eliminar los planes de estudio, las prelacones, las convalidaciones y las opciones profesionales de una asignatura es similar a los algoritmos para listar, crear, guardar, editar, actualizar y eliminar los planes de estudio.

#### 4.2.4 Pruebas

Las pruebas funcionales de esta iteración se basan en la creación, edición y eliminación de planes de estudio, asignaturas y opciones profesionales, así como la gestión de las características particulares de cada uno de ellos.

Caso de Prueba	Resultado Esperado	Resultado Obtenido
Crear un plan de estudio	Crear un plan de estudio nuevo	Se guardó la información básica del plan de estudio y se mostró un mensaje de confirmación
Editar un plan de estudio	Modificar un plan de estudio existente	Se actualizó la información básica del plan de estudio y se mostró un mensaje de confirmación
Eliminar un plan de estudio	Eliminar un plan de estudio existente	Se borró la información básica del plan de estudio y se mostró un mensaje de confirmación
Activar un plan de estudio	Registrar el plan de estudio activo de la carrera	Se registró el plan de estudio como el plan activo de la carrera y se mostró un mensaje de confirmación
Intentar eliminar un	Tratar de eliminar un plan	Se canceló la acción correctamente y se

plan de estudio activo	de estudio activo	mostró un mensaje de error
Crear un requisito curricular	Crear un requisito curricular nuevo en un plan de estudio	Se guardó el requisito curricular en el plan de estudio y se mostró un mensaje de confirmación
Editar un requisito curricular	Modificar un requisito curricular de un plan de estudio	Se actualizó el requisito curricular del plan de estudio y se mostró un mensaje de confirmación
Eliminar un requisito curricular	Eliminar un requisito curricular de un plan de estudio	Se borró el requisito curricular del plan de estudio y se mostró un mensaje de confirmación
Distribuir un requisito curricular	Distribuir un requisito curricular de un plan de estudio por semestres	Se guardó el número de asignaturas en cada semestre de un requisito curricular y se mostró un mensaje de confirmación
Redistribuir un requisito curricular	Redistribuir un requisito curricular de un plan de estudio por semestres	Se actualizó el número de asignaturas en cada semestre de un requisito curricular y se mostró un mensaje de confirmación
Intentar distribuir un número mayor de asignaturas	Tratar de distribuir un número de asignaturas mayor que el requerido por un requisito curricular	Se canceló la acción correctamente y se mostró un mensaje de error
Crear una opción profesional	Crear una opción profesional nueva	Se guardó la información básica de la opción profesional y se mostró un mensaje de confirmación
Editar una opción profesional	Modificar una opción profesional existente	Se actualizó la información básica de la opción profesional y se mostró un mensaje de confirmación
Eliminar una opción profesional	Eliminar una opción profesional existente	Se borró la información básica de la opción profesional y se mostró un mensaje de confirmación
Registrar una opción profesional	Registrar una opción profesional en un plan de estudio	Se registró la opción profesional en el plan de estudio y se mostró un mensaje de confirmación
Editar un registro de una opción profesional	Modificar una opción profesional registrada en un plan de estudio	Se actualizó la opción profesional registrada en el plan de estudio y se mostró un mensaje de confirmación
Eliminar un registro de una opción profesional	Eliminar una opción profesional registrada en un plan de estudio	Se borró la opción profesional registrada en el plan de estudio y se mostró un mensaje de confirmación
Crear un requisito curricular en una opción profesional	Crear un requisito curricular nuevo en una opción profesional	Se guardó el requisito curricular en la opción profesional y se mostró un mensaje de confirmación
Editar un requisito curricular de una opción profesional	Modificar un requisito curricular de una opción profesional	Se actualizó el requisito curricular de la opción profesional y se mostró un mensaje de confirmación
Eliminar un requisito curricular de una opción profesional	Eliminar un requisito curricular de una opción profesional	Se borró el requisito curricular de la opción profesional y se mostró un mensaje de confirmación

Crear una asignatura	Crear una asignatura nueva	Se guardó la información básica de la asignatura y se mostró un mensaje de confirmación
Editar una asignatura	Modificar una asignatura existente	Se actualizó la información básica de la asignatura y se mostró un mensaje de confirmación
Eliminar una asignatura	Eliminar una asignatura existente	Se borró la información básica de la asignatura y se mostró un mensaje de confirmación
Registrar una asignatura	Registrar una asignatura en un plan de estudio	Se registro la asignatura en el plan de estudio y se mostro un mensaje de confirmación
Editar un registro de una asignatura	Modificar una asignatura registrada en un plan de estudio	Se actualizó la asignatura registrada en el plan de estudio y se mostro un mensaje de confirmación
Eliminar un registro de una asignatura	Eliminar una asignatura registrada en un plan de estudio	Se borró la asignatura registrada en el plan de estudio y se mostro un mensaje de confirmación
Activar el tipo principal de una asignatura	Registrar el tipo principal de una asignatura en un plan de estudio	Se registró cual es el tipo principal de una asignatura registrada varias veces en el mismo plan de estudio y se mostró un mensaje de confirmación
Intentar eliminar el tipo principal de una asignatura	Tratar de eliminar el registro principal de una asignatura en un plan de estudio	Se canceló la acción correctamente y se mostro un mensaje de error
Crear una prelación	Crear una prelación nueva de una asignatura registrada en un plan de estudio	Se guardó la prelación de la asignatura registrada en el plan de estudio y se mostro un mensaje de confirmación
Editar una prelación	Modificar una prelación de una asignatura registrada en un plan de estudio	Se actualizó la prelación de la asignatura registrada en el plan de estudio y se mostro un mensaje de confirmación
Eliminar una prelación	Eliminar una prelación de una asignatura registrada en un plan de estudio	Se borró la prelación de la asignatura registrada en el plan de estudio y se mostro un mensaje de confirmación
Crear una convalidación	Crear una convalidación nueva de una asignatura registrada en un plan de estudio	Se guardó la convalidación de la asignatura registrada en el plan de estudio y se mostro un mensaje de confirmación
Editar una convalidación	Modificar una convalidación de una asignatura registrada en un plan de estudio	Se actualizó la convalidación de la asignatura registrada en el plan de estudio y se mostro un mensaje de confirmación
Eliminar una convalidación	Eliminar una convalidación de una asignatura registrada en un plan de	Se borró la convalidación de la asignatura registrada en el plan de estudio y se mostro un mensaje de

	estudio	confirmación
Registrar una asignatura en una opción profesional	Registrar una asignatura en una opción profesional de un plan de estudio	Se registró la asignatura en la opción profesional y se mostro un mensaje de confirmación
Editar un registro de una asignatura en una opción profesional	Modificar una asignatura registrada en una opción profesional de un plan de estudio	Se actualizó la asignatura registrada en la opción profesional y se mostro un mensaje de confirmación
Eliminar un registro de una asignatura en una opción profesional	Eliminar una asignatura registrada en una opción profesional de un plan de estudio	Se borró la asignatura registrada en la opción profesional y se mostro un mensaje de confirmación

### 4.3 Iteración 2

En esta iteración se describe el proceso de desarrollo de la verificación de los requisitos curriculares de los estudiantes regulares de pregrado en el módulo web, siguiendo las prácticas del método ágil XP; especificando la planificación, el diseño, la codificación y las pruebas de las tarjetas de historias correspondientes.

#### 4.3.1 Planificación

<b>Descripción</b>	Diseñar la interfaz gráfica, las estructuras de datos y desarrollar un algoritmo para verificar los requisitos curriculares de los estudiantes
<b>Tarjetas de Historias</b>	<ol style="list-style-type: none"> <li>1. Diseñar una interfaz gráfica para verificar requisitos curriculares</li> <li>2. Diseñar las estructuras de datos para verificar requisitos curriculares</li> <li>3. Desarrollar la verificación de asignaturas obligatorias</li> <li>4. Desarrollar la verificación de asignaturas obligatorias optativas</li> <li>5. Desarrollar la verificación de asignaturas electivas</li> <li>6. Desarrollar la verificación de asignaturas complementarias</li> <li>7. Desarrollar la verificación de laboratorios</li> <li>8. Desarrollar la verificación de pasantías</li> <li>9. Desarrollar la verificación de seminarios</li> <li>10. Desarrollar la verificación de TEG</li> <li>11. Desarrollar la verificación del servicio comunitario</li> <li>12. Desarrollar la verificación de asignaturas excedentes</li> <li>13. Desarrollar la verificación de convalidaciones</li> <li>14. Diseñar una interfaz gráfica para la cabecera</li> <li>15. Revisar los requisitos curriculares de la Licenciatura en Biología</li> <li>16. Revisar los requisitos curriculares de la Licenciatura en Computación</li> <li>17. Revisar los requisitos curriculares de la Licenciatura en Física</li> <li>18. Revisar los requisitos curriculares de la Licenciatura en Geoquímica</li> <li>19. Revisar los requisitos curriculares de la Licenciatura en Matemática</li> <li>20. Revisar los requisitos curriculares de la Licenciatura en Química</li> <li>21. Desarrollar un script de bases de datos</li> <li>22. Refactorizar el código para verificar requisitos curriculares</li> <li>23. Diseñar una interfaz gráfica para el resumen</li> <li>24. Desarrollar el resumen de los datos académicos</li> <li>25. Desarrollar el resumen de la verificación de los requisitos curriculares</li> <li>26. Diseñar las estructuras de datos para verificar opciones profesionales</li> <li>27. Desarrollar el resumen de la verificación de la opción profesional</li> <li>28. Desarrollar el resumen general</li> <li>29. Refactorizar el código para mostrar los resúmenes</li> <li>30. Desarrollar la selección de un estudiante y una carrera</li> <li>53. Desarrollar un PDF para la vista para verificar requisitos curriculares</li> <li>57. Desarrollar un script de bases de datos</li> </ol>
<b>Tiempo Estimado</b>	33 Días

### 4.3.2 Diseño

Los trabajadores del Departamento de Grado de la DCE verifican los requisitos curriculares de los estudiantes regulares de pregrado utilizando distintos documentos impresos, entre los cuales se destacan los planes de estudio de las licenciaturas de la Facultad de Ciencias y los kardex de los estudiantes.

La página web para verificar requisitos curriculares muestra todas las asignaturas (tanto aprobadas como no aprobadas por un estudiante) de una carrera organizadas en filas. En la primera fila se muestra la cabecera, las columnas de la primera fila muestran la denominación de los atributos más relevantes del kardex. A partir de la segunda fila, se muestran las asignaturas, las columnas de estas filas muestran los datos particulares de cada asignatura que correspondan con los atributos de la cabecera.

Las asignaturas están agrupadas por semestre, de manera similar al plan de estudio de la licenciatura correspondiente. La Figura 44 muestra un fragmento de la página web para verificar requisitos curriculares.

Inicio Preinscripción Inscripción Solicitudes Plan de Estudio Horarios

Lunes, 16 de Abril de 2011, 11:13 p.m.

**Verificar Requisitos**

UNIVERSIDAD CENTRAL DE VENEZUELA  
FACULTAD DE CIENCIAS  
COORDINACION ACADEMICA  
DIVISION DE CONTROL DE ESTUDIOS  
LICENCIATURA EN COMPUTACION

VERIFICACION DE REQUISITOS ACADEMICOS FECHA: DD-MM-AAAA

ESTUDIANTE: **HERNANDEZ CORREA JOSE TADEO**  
CEDULA: **16179441**

SEM	PERIODO	CODIGO	NOMBRE ASIGNATURA	TIPO	UC	NOTA	EXAMEN	SECCION
I	02-2004	6106	MATEMATICAS DISCRETAS I	OBLIGATORIA	4	14	FINAL	B3
	02-2004	6201	ALGORITMOS Y PROGRAMACION	OBLIGATORIA	6	12	FINAL	B5
	02-2004	6301	INTRODUCCION A LA INFORMATICA	OBLIGATORIA	4	12	FINAL	B4
	01-2004	8206	MATEMATICA I	OBLIGATORIA	6	13	FINAL	C2
II	01-2005	6001	ORGANIZACION Y ESTRUCTURA DEL COMPUTADOR I	OBLIGATORIA	6	11	FINAL	D1
	02-2005	6107	MATEMATICAS DISCRETAS II	OBLIGATORIA	4	10	FINAL	A2
	01-2005	6202	ALGORITMOS Y ESTRUCTURAS DE DATOS	OBLIGATORIA	5	10	FINAL	C1
	I-2005	8207	MATEMATICA II	OBLIGATORIA	6	11	FINAL	A3
III	01-2006	0030	INGLES I	OBLIGATORIA	4	19	FINAL	A2
	01-2006	6002	ORGANIZACION Y ESTRUCTURA DEL COMPUTADOR II	OBLIGATORIA	5	12	FINAL	D1
	02-2005	6203	INGENIERIA DE SOFTWARE	OBLIGATORIA	5	12	FINAL	A3
	02-2005	8208	MATEMATICA III	OBLIGATORIA	6	11	FINAL	C2

Figura 44. Verificación de Requisitos Curriculares.

Se utilizaron diferentes colores y grados de opacidad para diferenciar las asignaturas aprobadas y las asignaturas que aún no han sido aprobadas por un estudiante en una carrera. La Figura 45 muestra el efecto utilizado para diferenciar las asignaturas aprobadas y las no aprobadas.



IX	01-2010	6246	APLICACIONES CON TECNOLOGIA INTERNET II	ELECTIVA	5	10	FINAL	C2
	02-2009	6346	SISTEMAS DE BASES DE DATOS DISTRIBUIDAS	ELECTIVA	5	11	FINAL	C1
	02-2006	8203	TECNICA Y RECURSOS DE APRENDIZAJE	COMPLEMENTARIA	3	15	FINAL	A1
	01-2011	6419	SEMINARIO (APLICACIONES INTERNET)	SEMINARIO	5	19	FINAL	C1
X	-	-	TRABAJO ESPECIAL DE GRADO	-	-	-	-	-

**Figura 45.** Asignaturas Aprobadas/Asignaturas No Aprobadas.

Una asignatura se puede registrar varias veces en el mismo plan de estudio con tipos de asignaturas diferentes. Se creó el campo *tipo\_materia\_id* en la tabla *historial\_academico* para poder determinar los tipos de las asignaturas aprobadas por un estudiante en una carrera. La Figura 46 muestra la tabla *historial\_academico*.

```
mysql> desc historial_academico;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| carrera_id    | char(10)      | NO   | PRI | NULL    |       |
| plan_nombre   | char(20)      | NO   | PRI | NULL    |       |
| materia_codigo | varchar(10)   | NO   | PRI | NULL    |       |
| tipo_materia_id | char(10)     | NO   |     | NULL    |       |
| nombre_seccion | char(10)      | NO   | PRI | NULL    |       |
| periodo_academico_id | varchar(10) | NO   | PRI | NULL    |       |
| estudiante_cedula | varchar(20)  | NO   | PRI | NULL    |       |
| tipo_status_materia_id | char(10)   | NO   | MUL | NULL    |       |
| tipo_examen_id | char(10)      | NO   | MUL | NULL    |       |
| tipo_nota_id_reparacion | char(10)   | YES  | MUL | NULL    |       |
| tipo_nota_id_definitiva | char(10)   | YES  | MUL | NULL    |       |
| tipo_nota_id_final | char(10)     | YES  | MUL | NULL    |       |
| fecha_hora_calificacion | datetime    | YES  |     | NULL    |       |
| usuario_que_califico | varchar(255) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
14 rows in set (0.00 sec)
```

**Figura 46.** Tabla *historial\_academico*.

Las asignaturas del servicio comunitario se muestran luego de las asignaturas de un plan de estudio, así como el excedente de asignaturas electivas y/o complementarias, en caso de que un estudiante haya aprobado más asignaturas que las requeridas en una carrera.

El resumen de la verificación de los requisitos curriculares muestra el número de asignaturas que se deben aprobar (por cada requisito curricular) en una carrera organizadas en filas. En la primera fila se muestra la cabecera, las columnas de la primera fila muestran la denominación de los atributos que se verificaron de los requisitos curriculares. A partir de la segunda fila, se muestran los tipos de asignaturas de los requisitos curriculares, las columnas de estas filas muestran el resultado de la verificación de cada requisito curricular que corresponda con los atributos de la

cabecera. Se utilizaron iconos visuales para señalar los requisitos curriculares que un estudiante no satisface. La Figura 47 muestra el resumen de la verificación de los requisitos curriculares.



TIPO	TOTAL ASIGNATURAS	OBSERVACIONES
OBLIGATORIAS	21	El Estudiante cumple con todas las Asignaturas Obligatorias
OBLIGATORIAS OPTATIVAS	2	El Estudiante cumple con todas las Asignaturas Obligatorias Optativas
ELECTIVAS	8	El Estudiante cumple con todas las Asignaturas Electivas
COMPLEMENTARIAS	3	El Estudiante cumple con todas las Asignaturas Complementarias
LABORATORIOS	1	El Estudiante cumple con todos las Asignaturas de Laboratorio
PASANTIAS	1	El Estudiante cumple con todas las Asignaturas de Pasantias
SEMINARIOS	1	El Estudiante cumple con todas las Asignaturas de Seminario
TRABAJO ESPECIAL DE GRADO	1	Falta 1 Asignatura(s) de Trabajo Especial de Grado ❌
SERVICIO COMUNITARIO	2	El Estudiante cumple con todas las Asignaturas de Servicio Comunitario

Figura 47. Resultado de la Verificación de Requisitos Curriculares.

Existen opciones profesionales que se pueden obtener en diferentes planes de estudio. De este modo, la opción profesional Aplicaciones con la Tecnología Internet se puede obtener en los planes de estudio de Computación de los años 2000 y 2004, por ejemplo. Se crearon las tablas *opcion*, *opcion\_plan* y *requisito\_graduacion\_opcion\_plan* en la base de datos para poder registrar una opción profesional en varios planes de estudio, así como sus requisitos curriculares. La Figura 48 muestra las tablas *opcion*, *opcion\_plan* y *requisito\_graduacion\_opcion\_plan*.

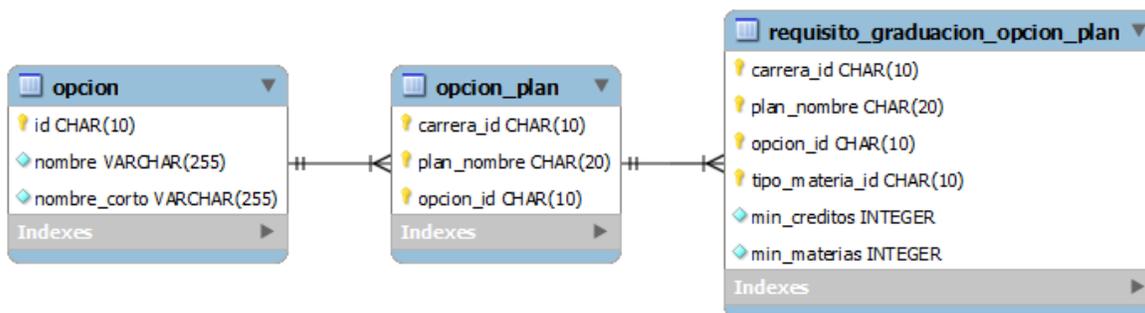


Figura 48. Tablas *opcion*, *opcion\_plan* y *requisito\_graduacion\_opcion\_plan*.

El resumen de la verificación de la opción profesional muestra todas las asignaturas (tanto aprobadas como no aprobadas por un estudiante) de una opción profesional organizadas en filas. En la primera fila se muestra la cabecera, las columnas de la primera fila muestran la denominación de los atributos de las asignaturas. A partir de la segunda fila, se muestran las asignaturas, las columnas de estas filas muestran los datos particulares de cada asignatura que correspondan con los atributos de la cabecera. La Figura 49 muestra el resumen de la verificación de la opción profesional.



VERIFICACION DE LA OPCION PROFESIONAL

CODIGO	NOMBRE ASIGNATURA	CREDITOS
6023	COMERCIO ELECTRONICO	5
6211	INTERACCION HUMANO-COMPUTADOR	5
6221	APLICACIONES CON LA TECNOLOGIA INTERNET	5
6246	APLICACIONES CON TECNOLOGIA INTERNET II	5
6419	SEMINARIO (APLICACIONES INTERNET)	5
6420	TRABAJO ESPECIAL DE GRADO (APLICACIONES INTERNET)	15

**Figura 49.** Verificación de la Opción Profesional.

Por último, se muestra el resultado general de la verificación de los requisitos curriculares de un estudiante en una carrera, en el cual se muestra un mensaje indicando si satisface o no satisface todos los requisitos curriculares necesarios para obtener un título universitario.

### 4.3.3 Codificación

Para verificar los requisitos curriculares de un estudiante en una carrera, primero se busca el plan de estudio activo de la carrera, las asignaturas obligatorias, el número mínimo de unidades de crédito y el número mínimo de asignaturas obligatorias optativas, electivas, complementarias, laboratorios, pasantías, seminarios, TEG y servicio comunitario que se deben aprobar, las convalidaciones y las asignaturas que ha aprobado en la base de datos, en el método *verificar\_requisitos* del controlador *plan\_estudio\_controller.rb*. La Figura 50 muestra un fragmento del método *verificar\_requisitos*.

```

701 def verificar_requisitos
702   @titulo_pagina = "Verificar Requisitos"
703
704   @estudiante = Estudiante.find(params[:estudiante_cedula])
705   @estudiante_en_carrera = EstudianteEnCarrera.where("carrera_id = ? AND estudiante_cedula = ?",
706     params[:carrera_id], params[:estudiante_cedula]).first
707   @plan_activo = PlanActivo.buscar_plan_activo(params[:carrera_id])
708
709   @requisito_graduacion_plan = RequisitoGraduacionPlan.buscar_requisitos_graduacion(@plan_activo.carrera_id, @plan_activo.plan_nombre)
710   @requisito_graduacion_plan.each do |requisito_graduacion_plan|
711     case requisito_graduacion_plan.tipo_materia_id
712     when "0"
713       @obligatorias_1 = MateriaEnPlan.buscar_obligatorias_semestre(@plan_activo.carrera_id, @plan_activo.plan_nombre, "1")
714       @obligatorias_2 = MateriaEnPlan.buscar_obligatorias_semestre(@plan_activo.carrera_id, @plan_activo.plan_nombre, "2")
715       @obligatorias_3 = MateriaEnPlan.buscar_obligatorias_semestre(@plan_activo.carrera_id, @plan_activo.plan_nombre, "3")
716       @obligatorias_4 = MateriaEnPlan.buscar_obligatorias_semestre(@plan_activo.carrera_id, @plan_activo.plan_nombre, "4")
717       @obligatorias_5 = MateriaEnPlan.buscar_obligatorias_semestre(@plan_activo.carrera_id, @plan_activo.plan_nombre, "5")
718       @obligatorias_6 = MateriaEnPlan.buscar_obligatorias_semestre(@plan_activo.carrera_id, @plan_activo.plan_nombre, "6")
719       @obligatorias_7 = MateriaEnPlan.buscar_obligatorias_semestre(@plan_activo.carrera_id, @plan_activo.plan_nombre, "7")
720       @obligatorias_8 = MateriaEnPlan.buscar_obligatorias_semestre(@plan_activo.carrera_id, @plan_activo.plan_nombre, "8")
721       @obligatorias_9 = MateriaEnPlan.buscar_obligatorias_semestre(@plan_activo.carrera_id, @plan_activo.plan_nombre, "9")
722       @obligatorias_10 = MateriaEnPlan.buscar_obligatorias_semestre(@plan_activo.carrera_id, @plan_activo.plan_nombre, "10")
723
724       @OBLIGATORIAS_ES_REQUISITO = true
725       @MIN_CREDITOS_OBLIGATORIAS = requisito_graduacion_plan.min_creditos
726       @MIN_MATERIAS_OBLIGATORIAS = requisito_graduacion_plan.min_materias
727
728       @obligatorias_aprobadas = []
729       @creditos_obligatorias = 0
730       @materias_obligatorias = 0
731     when "00"
732       @obligatorias_optativas_1 = DistribucionRequisitoPlan.buscar_numero_asignaturas
733         (@plan_activo.carrera_id, @plan_activo.plan_nombre, "00", "1")
734       @obligatorias_optativas_2 = DistribucionRequisitoPlan.buscar_numero_asignaturas
735         (@plan_activo.carrera_id, @plan_activo.plan_nombre, "00", "2")
736       @obligatorias_optativas_3 = DistribucionRequisitoPlan.buscar_numero_asignaturas
737         (@plan_activo.carrera_id, @plan_activo.plan_nombre, "00", "3")
738       @obligatorias_optativas_4 = DistribucionRequisitoPlan.buscar_numero_asignaturas
739         (@plan_activo.carrera_id, @plan_activo.plan_nombre, "00", "4")
740       @obligatorias_optativas_5 = DistribucionRequisitoPlan.buscar_numero_asignaturas
741         (@plan_activo.carrera_id, @plan_activo.plan_nombre, "00", "5")
742       @obligatorias_optativas_6 = DistribucionRequisitoPlan.buscar_numero_asignaturas
743         (@plan_activo.carrera_id, @plan_activo.plan_nombre, "00", "6")
744       @obligatorias_optativas_7 = DistribucionRequisitoPlan.buscar_numero_asignaturas
745         (@plan_activo.carrera_id, @plan_activo.plan_nombre, "00", "7")
746       @obligatorias_optativas_8 = DistribucionRequisitoPlan.buscar_numero_asignaturas
747         (@plan_activo.carrera_id, @plan_activo.plan_nombre, "00", "8")
748       @obligatorias_optativas_9 = DistribucionRequisitoPlan.buscar_numero_asignaturas
749         (@plan_activo.carrera_id, @plan_activo.plan_nombre, "00", "9")
750       @obligatorias_optativas_10 = DistribucionRequisitoPlan.buscar_numero_asignaturas
751         (@plan_activo.carrera_id, @plan_activo.plan_nombre, "00", "10")
752
753       @OBLIGATORIAS_OPTATIVAS_ES_REQUISITO = true
754       @MIN_CREDITOS_OBLIGATORIAS_OPTATIVAS = requisito_graduacion_plan.min_creditos
755       @MIN_MATERIAS_OBLIGATORIAS_OPTATIVAS = requisito_graduacion_plan.min_materias
756
757       @obligatorias_optativas_aprobadas = []
758       @obligatorias_optativas_asignadas = 0
759       @creditos_obligatorias_optativas = 0
760       @materias_obligatorias_optativas = 0
761     .
762     .
763   end
764
765   @convalidacion = Convalidacion.buscar_convalidaciones(@plan_activo.carrera_id, @plan_activo.plan_nombre)
766
767   @asignaturas_aprobadas = HistorialAcademico.buscar_asignaturas_aprobadas
768     (@plan_activo.carrera_id, @plan_activo.plan_nombre, @estudiante.cedula)
769   @asignaturas_aprobadas.each do |asignatura_aprobada|
770     case asignatura_aprobada.tipo_materia_id
771     when "0"
772       @obligatorias_aprobadas << {:sem => asignatura_aprobada.materia_en_plan.ubicacion_en_carrera,
773         :periodo => asignatura_aprobada.periodo_academico_id,
774         :codigo => asignatura_aprobada.materia_codigo,
775         :seccion => asignatura_aprobada.nombre_seccion,
776         :nombre => asignatura_aprobada.materia_nombre,
777         :tipo => asignatura_aprobada.tipo_materia.descripcion,
778         :uc => asignatura_aprobada.materia_en_plan.nro_creditos,
779         :nota => asignatura_aprobada.tipo_nota_id_definitiva,
780         :examen => asignatura_aprobada.tipo_examen.descripcion}
781
782       @creditos_obligatorias += asignatura_aprobada.materia_en_plan.nro_creditos
783       @materias_obligatorias += 1
784     .
785     .
786   end
787
788 end
789

```

Figura 50. Método plan\_estudio/verificar\_requisitos (Requisitos Curriculares).

El algoritmo para buscar el número de asignaturas electivas, complementarias, laboratorios, pasantías, seminarios, TEG y servicio comunitario que se deben aprobar en una carrera, es similar al algoritmo para buscar el número de asignaturas obligatorias optativas que se deben aprobar, que se muestra en la figura anterior.

Además, el algoritmo para buscar las asignaturas obligatorias optativas, electivas, complementarias, laboratorios, pasantías, seminarios, TEG y servicio comunitario aprobadas por un estudiante en una carrera, es similar al algoritmo para buscar las asignaturas obligatorias aprobadas por un estudiante, que se muestra en la figura anterior.

Es importante resaltar que las asignaturas de tipo método y PCI son consideradas como asignaturas electivas y complementarias respectivamente, en caso de que un estudiante haya aprobado alguna asignatura de estos tipos en una carrera.

Luego, en la vista parcial *\_verificar\_requisitos\_academicos.html.erb*, se verifica dentro de un ciclo si un estudiante ha aprobado todas las asignaturas obligatorias (o las asignaturas por las cuales han sido convalidadas) y el número de asignaturas obligatorias optativas, electivas, complementarias, laboratorios, pasantías, seminarios y TEG requeridas en cada semestre de una carrera. La Figura 51 muestra un fragmento de la vista parcial *\_verificar\_requisitos\_academicos.html.erb* (Ver Figura 44).

```

41 <% (1..10).each do |i| %>
42 <% obligatorias = instance_variable_get("@obligatorias_#{i}") %>
43 <% obligatorias_optativas = instance_variable_get("@obligatorias_optativas_#{i}") %>
44
45 <% sem = 0 %>
46
47 <% unless obligatorias.blank? %>
48 <% obligatorias.each do |obligatoria| %>
49 <% status = 0 %>
50 <% @obligatorias_aprobadas.each do |aprobada| %>
51 <% if(obligatoria.materia_codigo == aprobada[:codigo]) %>
52 <% status = 1 %>
53 <tr>
54 <td><p><%= aprobada[:periodo] %></p></td>
55 <td><p><%= aprobada[:codigo] %></p></td>
56 <td><p><%= aprobada[:nombre] %></p></td>
57 <td><p><%= aprobada[:tipo] %></p></td>
58 <td><p><%= aprobada[:uc] %></p></td>
59 <td><p><%= aprobada[:nota][0,2] %></p></td>
60 <td><p><%= aprobada[:examen] %></p></td>
61 <td><p><%= aprobada[:seccion] %></p></td>
62 </tr>
63 <% end %>
64 <% end %>
65
66 <% if(status == 0 && !@convalidaciones.blank?) %>
67 <% @convalidaciones.each do |convalidacion| %>
68 <% if(obligatoria.materia_codigo == convalidacion.es_convalidada_por_materia_codigo) %>
69 <% @obligatorias_aprobadas.each do |aprobada| %>
70 <% if(convalidacion.materia_codigo == aprobada[:codigo]) %>
71 <% status = 1 %>
72 <tr>
73 <td><p><%= aprobada[:periodo] %></p></td>
74 <td><p><%= aprobada[:codigo] %></p></td>
75 <td><p><%= aprobada[:nombre] %></p></td>
76 <td><p><%= aprobada[:tipo] %></p></td>
77 <td><p><%= aprobada[:uc] %></p></td>
78 <td><p><%= aprobada[:nota][0,2] %></p></td>
79 <td><p><%= aprobada[:examen] %></p></td>
80 <td><p><%= aprobada[:seccion] %></p></td>
81 </tr>
82
83 <tr>
84 <td colspan="9">
85 <div class="notificacion">
86 <table>
87 <tr>
88 <td><b>;</b></td>
89 <td><p>LA ASIGNATURA <%= convalidacion.materia_codigo %> HA SIDO CONVALIDADA POR LA ASIGNATURA
90 <%= obligatoria.materia_codigo %> <%= obligatoria.materia.nombre %> DE
91 <%= obligatoria.nro_creditos %> UC</p></td>
92 </tr>
93 </table>
94 </div>
95 </td>
96 </tr>
97 <% end %>
98 <% end %>
99 <% end %>
100 <% end %>
101 <% end %>
102
103 <% if(status == 0) %>
104 <tr class="sin-calificar">
105 <td><p></p></td>
106 <td><p><%= obligatoria.materia_codigo %></p></td>
107 <td><p><%= obligatoria.materia.nombre %></p></td>
108 <td><p><%= obligatoria.tipo_materia.descripcion %></p></td>
109 <td><p></p></td>
110 <td><p></p></td>
111 <td><p></p></td>
112 <td><p></p></td>
113 </tr>
114 <% end %>
115 <% end %>
116 <% end %>

```

```

118 <% unless obligatorias_optativas.blank? %>
119 <% (1..obligatorias_optativas.numero_materias).each do |j| %>
120 <% if (@materias_obligatorias_optativas - @obligatorias_optativas_asignadas > 0) %>
121 <tr>
122 <td><p><%= @obligatorias_optativas_aprobadas[@obligatorias_optativas_asignadas][:periodo] %></p></td>
123 <td><p><%= @obligatorias_optativas_aprobadas[@obligatorias_optativas_asignadas][:codigo] %></p></td>
124 <td><p><%= @obligatorias_optativas_aprobadas[@obligatorias_optativas_asignadas][:nombre] %></p></td>
125 <td><p><%= @obligatorias_optativas_aprobadas[@obligatorias_optativas_asignadas][:tipo] %></p></td>
126 <td><p><%= @obligatorias_optativas_aprobadas[@obligatorias_optativas_asignadas][:uc] %></p></td>
127 <td><p><%= @obligatorias_optativas_aprobadas[@obligatorias_optativas_asignadas][:nota][0,2] %></p></td>
128 <td><p><%= @obligatorias_optativas_aprobadas[@obligatorias_optativas_asignadas][:examen] %></p></td>
129 <td><p><%= @obligatorias_optativas_aprobadas[@obligatorias_optativas_asignadas][:seccion] %></p></td>
130 </tr>
131
132 <% @obligatorias_optativas_asignadas += 1 %>
133 <% else %>
134 <tr class="sin-calificar">
135 <td><p></p></td>
136 <td><p></p></td>
137 <td><p>OBLIGATORIA OPTATIVA</p></td>
138 <td><p></p></td>
139 <td><p></p></td>
140 <td><p></p></td>
141 <td><p></p></td>
142 <td><p></p></td>
143 </tr>
144 <% end %>
145 <% end %>
146 <% end %>

```

Figura 51. Vista Parcial \_verificar\_requisitos\_academicos.html.erb.

El algoritmo para verificar si un estudiante ha aprobado el número de asignaturas electivas, complementarias, laboratorios, pasantías, seminarios y TEG en cada semestre de una carrera, es similar al algoritmo para verificar si ha aprobado el número de asignaturas obligatorias optativas, que se muestra en la figura anterior.

En la vista parcial *\_verificar\_servicio\_comunitario.html.erb*, se verifica si un estudiante ha aprobado mas asignaturas electivas y/o complementarias que las requeridas en una carrera, así como se verifica si ha aprobado todas las asignaturas del servicio comunitario.

En el método *verificar\_requisitos* del controlador *plan\_estudio\_controller.rb*, también se determina la opción profesional de un estudiante en una carrera, en función de los seminarios y TEG que haya aprobado o inscrito en el semestre en curso. Además, se buscan todos los requisitos curriculares de la opción profesional en la base de datos. La Figura 52 muestra otro fragmento del método *verificar\_requisitos*.

```

1034 if(!@TEG_aprobados.blank?)
1035   @materia_en_opcion_plan = MateriaEnOpcionPlan.buscar_opcion_profesional
1036   (@plan_activo.carrera_id, @plan_activo.plan_nombre, @TEG_aprobados[0][:codigo], "T")
1037 else
1038   @TEG_inscrito = HistorialAcademico.buscar_asignatura_sin_calificar
1039   (@plan_activo.carrera_id, @plan_activo.plan_nombre, "T", @estudiante.cedula)
1040
1041   if(@TEG_inscrito)
1042     @materia_en_opcion_plan = MateriaEnOpcionPlan.buscar_opcion_profesional
1043     (@plan_activo.carrera_id, @plan_activo.plan_nombre, @TEG_inscrito.materia_codigo, "T")
1044   elsif(!@seminarios_aprobados.blank?)
1045     @materia_en_opcion_plan = MateriaEnOpcionPlan.buscar_opcion_profesional
1046     (@plan_activo.carrera_id, @plan_activo.plan_nombre, @seminarios_aprobados[0][:codigo], "S")
1047   end
1048 end
1049
1050 if(!@TEG_inscrito && @seminarios_aprobados.blank?)
1051   @seminario_inscrito = HistorialAcademico.buscar_asignatura_sin_calificar
1052   (@plan_activo.carrera_id, @plan_activo.plan_nombre, "S", @estudiante.cedula)
1053
1054   if(@seminario_inscrito)
1055     @materia_en_opcion_plan = MateriaEnOpcionPlan.buscar_opcion_profesional
1056     (@plan_activo.carrera_id, @plan_activo.plan_nombre, @seminario_inscrito.materia_codigo, "S")
1057   else
1058     @materia_en_opcion_plan = MateriaEnOpcionPlan.new(:opcion_id => "SINMEN")
1059   end
1060 end
1061
1062 @requisito_graduacion_opcion_plan = RequisitoGraduacionOpcionPlan.buscar_requisitos_opcion
1063 (@plan_activo.carrera_id, @plan_activo.plan_nombre, @materia_en_opcion_plan.opcion_id)
1064 @requisito_graduacion_opcion_plan.each do |requisito_graduacion_opcion_plan|
1065   case requisito_graduacion_opcion_plan.tipo_materia_id
1066   when "E"
1067     @electivas_obligatorias_opcion = MateriaEnOpcionPlan.buscar_asignaturas_opcion
1068     (@plan_activo.carrera_id, @plan_activo.plan_nombre, "E", @materia_en_opcion_plan.opcion_id, "1")
1069     @electivas_obligatorias_opcion += MateriaEnOpcionPlan.buscar_asignaturas_opcion
1070     (@plan_activo.carrera_id, @plan_activo.plan_nombre, "M", @materia_en_opcion_plan.opcion_id, "1")
1071     @electivas_opcion = MateriaEnOpcionPlan.buscar_asignaturas_opcion
1072     (@plan_activo.carrera_id, @plan_activo.plan_nombre, "E", @materia_en_opcion_plan.opcion_id, "0")
1073     @electivas_opcion += MateriaEnOpcionPlan.buscar_asignaturas_opcion
1074     (@plan_activo.carrera_id, @plan_activo.plan_nombre, "M", @materia_en_opcion_plan.opcion_id, "0")
1075
1076     @ELECTIVAS_ES_REQUISITO OPCION = true
1077     @MIN_CREDITOS ELECTIVAS OPCION = requisito_graduacion_opcion_plan.min_creditos
1078     @MIN_MATERIAS ELECTIVAS OPCION = requisito_graduacion_opcion_plan.min_materias
1079
1080     @creditos_electivas_opcion = 0
1081     @materias_electivas_opcion = 0
1082   when "S"
1083     @seminarios_obligatorios_opcion = MateriaEnOpcionPlan.buscar_asignaturas_opcion
1084     (@plan_activo.carrera_id, @plan_activo.plan_nombre, "S", @materia_en_opcion_plan.opcion_id, "1")
1085
1086     @SEMINARIOS_ES_REQUISITO OPCION = true
1087     @MIN_CREDITOS SEMINARIOS OPCION = requisito_graduacion_opcion_plan.min_creditos
1088     @MIN_MATERIAS SEMINARIOS OPCION = requisito_graduacion_opcion_plan.min_materias
1089
1090     @creditos_seminarios_opcion = 0
1091     @materias_seminarios_opcion = 0
1092   end
1093 end

```

Figura 52. Método plan\_estudio/verificar\_requisitos (Opción Profesional).

El algoritmo para buscar los TEG que se deben aprobar en una opción profesional, es similar al algoritmo para buscar los seminarios que se deben aprobar, que se muestra en la figura anterior.

En la vista parcial *\_resumen\_opcion\_profesional.html.erb*, se verifica dentro de un ciclo si un estudiante ha aprobado todas las asignaturas electivas, seminarios y TEG que se deben aprobar en una opción profesional. La Figura 53 muestra un fragmento de la vista parcial *\_resumen\_opcion\_profesional.html.erb* (Ver Figura 49).

```

20 <% if(@ELECTIVAS_ES_REQUISITO OPCION) %>
21 <% unless @electivas_obligatorias_opcion.blank? %>
22 <% @electivas_obligatorias_opcion.each do |obligatoria| %>
23 <% status = 0 %>
24 <% @electivas_aprobadas.each do |aprobada| %>
25 <% if(obligatoria.materia_codigo == aprobada[:codigo]) %>
26 <% status = 1 %>
27 <tr>
28 <td><p><%= aprobada[:codigo] %></p></td>
29 <td><p><%= aprobada[:nombre] %></p></td>
30 <td class="ultima-columna"><p><%= aprobada[:uc] %></p></td>
31 </tr>
32 <% @creditos_electivas_opcion += aprobada[:uc] %>
33 <% @materias_electivas_opcion += 1 %>
34 <% end %>
35 <% end %>
36
37 <% if(status == 0) %>
38 <tr class="sin-calificar">
39 <td><p><%= obligatoria.materia_codigo %></p></td>
40 <td><p><%= obligatoria.materia.nombre %></p></td>
41 <td class="ultima-columna"><p><%= obligatoria.materia_en_plan.nro_creditos %></p></td>
42 </tr>
43 <% end %>
44 <% end %>
45 <% end %>
46
47 <% aux = 0 %>
48 <% unless @electivas_opcion.blank? %>
49 <% @electivas_opcion.each do |electiva| %>
50 <% @electivas_aprobadas.each do |aprobada| %>
51 <% if(electiva.materia_codigo == aprobada[:codigo]) %>
52 <tr>
53 <td><p><%= aprobada[:codigo] %></p></td>
54 <td><p><%= aprobada[:nombre] %></p></td>
55 <td class="ultima-columna"><p><%= aprobada[:uc] %></p></td>
56 </tr>
57 <% @creditos_electivas_opcion += aprobada[:uc] %>
58 <% @materias_electivas_opcion += 1 %>
59 <% aux += 1 %>
60 <% end %>
61 <% end %>
62 <% end %>
63 <% end %>
64
65 <% if(@MIN_MATERIAS_ELECTIVAS OPCION > @materias_electivas_opcion) %>
66 <% if(!@electivas_obligatorias_opcion.blank? && aux >= 0) %>
67 <% (@electivas_obligatorias_opcion.size + aux..@MIN_MATERIAS_ELECTIVAS OPCION - 1).each do |i| %>
68 <tr class="sin-calificar">
69 <td><p></p></td>
70 <td><p>ELECTIVA</p></td>
71 <td class="ultima-columna"><p></p></td>
72 </tr>
73 <% end %>
74 <% elsif(@electivas_obligatorias_opcion.blank?) %>
75 <% (@materias_electivas_opcion..@MIN_MATERIAS_ELECTIVAS OPCION - 1).each do |i| %>
76 <tr class="sin-calificar">
77 <td><p></p></td>
78 <td><p>ELECTIVA</p></td>
79 <td class="ultima-columna"><p></p></td>
80 </tr>
81 <% end %>
82 <% end %>
83 <% end %>
84 <% end %>

```

Figura 53. Vista Parcial \_resumen\_opcion\_profesional.html.erb.

Por último, en la vista *\_revision\_requisitos\_notas.erb*, se verifica si un estudiante satisface todos los requisitos curriculares para obtener el título de una carrera, y se muestran cada uno de los requisitos curriculares que no satisface en caso contrario.

#### 4.3.4 Pruebas

Las pruebas funcionales de esta iteración se basan en la verificación de los requisitos curriculares de los estudiantes regulares de pregrado de la Facultad de Ciencias.

<b>Caso de Prueba</b>	<b>Resultado Esperado</b>	<b>Resultado Obtenido</b>
Buscar un estudiante con una carrera	Buscar un estudiante inscrito en una sola carrera	Se buscó la carrera en la cual está inscrito el estudiante para verificar sus requisitos curriculares
Buscar un estudiante con más de una carrera	Buscar un estudiante inscrito en más de una carrera	Se buscaron las carreras en las cuales está inscrito el estudiante y se mostraron en un dialogo emergente
Buscar un estudiante sin carrera	Buscar un estudiante que no esté inscrito en ninguna carrera	Se mostro un mensaje de error en un dialogo emergente
Verificar los requisitos de un estudiante de primer semestre	Verificar los requisitos curriculares de un estudiante del primer semestre de una carrera	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular
Verificar los requisitos de un estudiante de Biología	Verificar los requisitos curriculares de un estudiante que este terminando la carrera de Biología	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular
Verificar los requisitos de un estudiante de Computación	Verificar los requisitos curriculares de un estudiante que este terminando la carrera de Computación	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular
Verificar los requisitos de un estudiante de Física	Verificar los requisitos curriculares de un estudiante que este terminando la carrera de Física	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular
Verificar los requisitos de un estudiante de Geoquímica	Verificar los requisitos curriculares de un estudiante que este terminando la carrera de Geoquímica	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular
Verificar los requisitos de un estudiante de Matemática	Verificar los requisitos curriculares de un estudiante que este terminando la carrera de Matemática	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular
Verificar los requisitos de un estudiante de Química	Verificar los requisitos curriculares de un estudiante que este terminando la carrera de Química	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular
Verificar los requisitos de un estudiante con convalidaciones	Verificar los requisitos curriculares de un estudiante con asignaturas convalidadas en una carrera	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular

Verificar los requisitos de un estudiante con excedente de asignaturas	Verificar los requisitos curriculares de un estudiante que haya aprobado más asignaturas que las requeridas en una carrera	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular
Verificar los requisitos de un estudiante sin opción	Verificar los requisitos curriculares de un estudiante sin opción profesional en una carrera	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular
Verificar los requisitos de un estudiante con opción	Verificar los requisitos curriculares de un estudiante con opción profesional en una carrera	Se verificaron los requisitos curriculares del estudiante y se mostró el resumen curricular

En esta iteración también se realizaron las pruebas de aceptación (con cuatro trabajadores de la DCE), las cuales se basan en la gestión de los planes de estudio de las licenciaturas de la Facultad de Ciencias y la verificación de los requisitos curriculares de los estudiantes regulares de pregrado.

Las principales funcionalidades del módulo web se dividieron en tres grupos diferentes, con el fin de permitir que los trabajadores que participaran en las pruebas, probaran las funcionalidades específicas del módulo web que correspondieran con las labores de su trabajo. En las pruebas de aceptación los trabajadores de la DCE realizaron las siguientes tareas:

#### Planes de Estudio

1. Crear plan de estudio con datos introducidos por el usuario
2. Activar plan de estudio
3. Intentar eliminar plan de estudio activo
4. Crear requisito curricular con datos introducidos por el usuario
5. Distribuir requisito curricular con datos introducidos por el usuario
6. Editar distribución requisito curricular con datos introducidos por el usuario
7. Intentar editar distribución requisito curricular con total asignaturas distribuidas > mínimo asignaturas
8. Intentar editar requisito curricular con mínimo asignaturas < total asignaturas distribuidas

#### Asignaturas

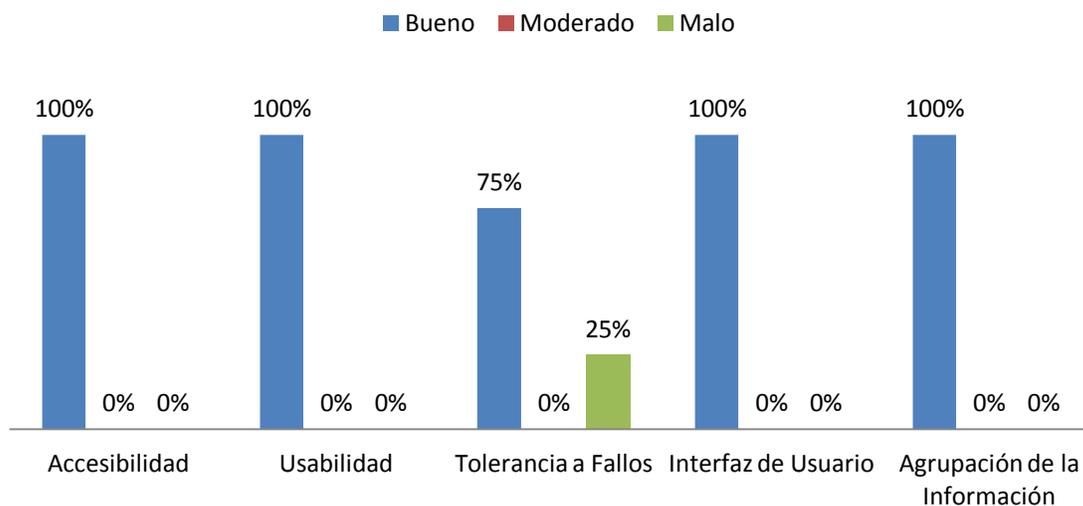
1. Crear asignatura con datos introducidos por el usuario
2. Registrar asignatura en plan de estudio con datos introducidos por el usuario
3. Activar asignatura principal
4. Intentar eliminar asignatura principal
5. Crear convalidación con datos introducidos por el usuario
6. Buscar asignatura

## 7. Registrar asignatura en opción profesional con datos introducidos por el usuario

### Opciones Profesionales/Verificación Requisitos Curriculares

1. Crear opción profesional con datos introducidos por el usuario
2. Registrar opción profesional en plan de estudio con datos introducidos por el usuario
3. Crear requisito curricular opción profesional con datos introducidos por el usuario
4. Intentar eliminar profesional con registros historial académico
5. Buscar estudiante predefinido
6. Validar requisitos curriculares de estudiante predefinido

Luego de realizar las pruebas de aceptación, los trabajadores de la DCE respondieron una encuesta para evaluar los requerimientos funcionales y los requerimientos no funcionales del módulo web. A continuación se presenta el resultado de la encuesta realizada por los trabajadores de la DCE:



## Conclusiones

El objetivo general de este Trabajo Especial de Grado es desarrollar un módulo web basado en tecnologías web, que permita gestionar los planes de estudio de las licenciaturas de la Facultad de Ciencias, así como automatizar el trabajo de verificación de requisitos curriculares del proceso de graduación realizado por la División de Control de Estudios (DCE).

Se facilitaron las labores de los trabajadores del Departamento de Grado de la DCE mediante la automatización de la verificación de los requisitos curriculares de los estudiantes regulares de pregrado de la Facultad de Ciencias, delegando las labores monótonas al módulo web, permitiendo que los trabajadores se dediquen a labores más importantes. Además, los estudiantes de la Facultad de Ciencias pueden verificar los requisitos curriculares de su carrera y generar los documentos PDF relacionados, desde cualquier computadora que disponga de una conexión a internet.

El sistema CONEST no contaba con una aplicación que permitiera gestionar los planes de estudio de las distintas licenciaturas que ofrece la Facultad de Ciencias, este trabajo se realizaba manipulando directamente la base de datos del sistema CONEST. Esto representa un mayor esfuerzo humano, y requiere de una persona con conocimientos tecnológicos para realizar el trabajo sin comprometer la integridad de la base de datos.

En este sentido, se simplificó y se redujo la carga de trabajo de los administradores de la DCE mediante la automatización de la gestión de los planes de estudio de las licenciaturas que ofrece la Facultad de Ciencias, estos se estudiaron para determinar las características comunes y las diferencias de cada plan de estudio, así como las validaciones y las relaciones entre las asignaturas y las opciones profesionales que los conforman. Al contar con las funcionalidades que ofrece el módulo web, los administradores de la DCE deben cumplir un perfil más general para gestionar los planes de estudio de las licenciaturas de la Facultad de Ciencias en el sistema CONEST.

Se realizaron un conjunto de pruebas funcionales para verificar el correcto funcionamiento del módulo web con la participación de los tutores (que son usuarios finales del sistema). Además, se realizaron las pruebas de aceptación con los trabajadores de la DCE. Los resultados de estas pruebas fueron positivos, y los usuarios que participaron en las mismas mostraron interés en utilizar las funcionalidades del módulo web para realizar sus labores.

El módulo web no se puso en producción con el resto del sistema CONEST, debido a los cambios realizados en las estructuras de datos durante el desarrollo de este proyecto. Estos cambios afectarían el funcionamiento del resto de los módulos del sistema CONEST sin una migración previa de la base de datos y la refactorización de la codificación.

El módulo web para la gestión de planes de estudio en el sistema CONEST se desarrolló utilizando tecnologías web actuales, entre las cuales se destacan el lenguaje de programación Ruby y el

framework de desarrollo de aplicaciones web Rails. Esta experiencia fue muy enriquecedora ya que pude profundizar mis conocimientos previos del framework y mejorar mis habilidades para desarrollar aplicaciones web siguiendo las prácticas de programación recomendadas en las páginas de Ruby on Rails Guides y Rails on Rails Screencasts.

Es importante resaltar que durante el proceso de desarrollo de este Trabajo Especial de Grado, se publicó una actualización importante del framework Rails (versión 3.1). Por este motivo, se actualizó el framework y se refactorizó el código del módulo web para cumplir con los estándares de programación de la nueva versión de Rails.

Otro aspecto importante a tomar en cuenta, es la adaptación del método ágil XP implementado para estructurar, planificar y controlar el proceso de desarrollo del módulo web. Algunas prácticas del método ágil XP no se aplicaron en este proyecto tales como el desarrollo previamente probado, la programación en parejas, la propiedad colectiva y la integración continua. Sin embargo, se aplicaron el resto de las prácticas del método, lo cual facilitó la planificación y las entregas del proyecto, dividiendo los requerimientos en iteraciones, que se fueron desarrollando progresivamente según su prioridad.

Finalmente, gracias al desarrollo de este Trabajo Especial de Grado se cumplieron satisfactoriamente los principales objetivos de este proyecto, logrando un aporte significativo al sistema CONEST, ya que hasta el momento no contaba con una aplicación que permitiera gestionar los planes de estudio de las distintas licenciaturas que ofrece la Facultad de Ciencias, así como verificar los requisitos curriculares de los estudiantes regulares de pregrado.

## Recomendaciones

A pesar de los beneficios que ahora pueden disfrutar los trabajadores de la DCE y los estudiantes de pregrado de la Facultad de Ciencias a través del módulo web para la gestión de planes de estudio en el sistema CONEST, se pueden considerar algunas mejoras y nuevas funcionalidades que permitan el mantenimiento preventivo y correctivo del sistema CONEST, como las que se mencionan a continuación:

- Los planes de estudio están intrínsecamente relacionados con los procesos administrativos y las normativas académicas de las licenciaturas de la Facultad de Ciencias, por este motivo, se recomienda refactorizar el sistema CONEST para que soporte los cambios realizados en las estructuras de datos durante el desarrollo de este Trabajo Especial de Grado. Siendo el módulo de inscripción, uno de los más afectados ya que este debe determinar cuáles son las asignaturas que puede cursar un estudiante basándose en su avance en la carrera, considerando las prelación entre las asignaturas y el tipo principal de las asignaturas registradas múltiples veces en un mismo plan de estudio, ambas, características fundamentales desarrolladas en el módulo web para la gestión de planes de estudio en el sistema CONEST.
- En el caso de las convalidaciones, los Consejos de Escuela de la Facultad de Ciencias aprueban las convalidaciones generales de las asignaturas que correspondan con su carrera, los trabajadores de la DCE no cuentan con un mecanismo para conocer las convalidaciones aprobadas por los distintos Consejos de Escuela, por este motivo, se recomienda desarrollar una aplicación que le permita a los trabajadores de la DCE conocer los detalles de las nuevas convalidaciones para registrarlas en el módulo web.
- Con respecto a la verificación de requisitos curriculares, cada semestre se verifican los requisitos curriculares de los estudiantes que inscribieron TEG. Los trabajadores del Departamento de Grado de la DCE utilizan un documento impreso, para conocer cuáles son dichos estudiantes. Se recomienda desarrollar una aplicación que le permita a los trabajadores del Departamento de Grado consultar cuáles son los estudiantes que inscribieron TEG en el semestre actual, así como imprimir el historial académico y la verificación de requisitos curriculares correspondientes, en caso de ser necesario.
- El componente docente es un convenio que existe con la Facultad de Humanidades y Educación que les permite a los estudiantes de las licenciaturas de Biología, Física, Matemática y Química optar por un título universitario de Licenciatura en Educación con una mención científica. Se recomienda crear los planes de estudio de los componentes docentes que ofrece la Facultad de Ciencias utilizando el módulo web para poder verificar los requisitos curriculares de los estudiantes inscritos en dichas carreras.

## Referencias Bibliográficas

- Asierra. (14 de Junio de 2004). *Desarrollo ágil de software - Wikipedia, la enciclopedia libre*. Recuperado el 15 de Enero de 2011, de Wikipedia, la enciclopedia libre: [http://es.wikipedia.org/wiki/Desarrollo\\_%C3%A1gil\\_de\\_software](http://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software)
- Beck, K. (2005). *Extreme Programming Explained: Embrace Change*. Boston: Pearson Education, Inc.
- Bos, B. (15 de Diciembre de 2005). *CSS current work & how to participate*. Recuperado el 20 de Diciembre de 2010, de World Wide Web Consortium (W3C): <http://www.w3.org/Style/CSS/current-work.en.html>
- Haml. (2011). *#haml.about*. Recuperado el 23 de Mayo de 2011, de #haml: <http://haml-lang.com/>
- Hassani, K. (3 de Abril de 2005). *Web application - Wikipedia, the free encyclopedia*. Recuperado el 18 de Enero de 2011, de [http://en.wikipedia.org/wiki/Web\\_application](http://en.wikipedia.org/wiki/Web_application)
- Hickson, I. (19 de Octubre de 2010). *HTML5*. Recuperado el 20 de Diciembre de 2010, de World Wide Web Consortium (W3C): <http://dev.w3.org/html5/spec/Overview.html>
- Obfusco, F. (24 de Noviembre de 2003). *MySQL - Wikipedia, the free encyclopedia*. Recuperado el 20 de Diciembre de 2010, de Wikipedia, the free encyclopedia: <http://en.wikipedia.org/wiki/MySQL>
- Patrick. (27 de Julio de 2004). *Hypertext Transfer Protocol - Wikipedia, the free encyclopedia*. Recuperado el 20 de Diciembre de 2010, de Wikipedia, the free encyclopedia: [http://en.wikipedia.org/wiki/Hypertext\\_Transfer\\_Protocol](http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol)
- Ruby on Rails. (2011). <http://rubyonrails.org/documentation>. Recuperado el 20 de Diciembre de 2010, de <http://rubyonrails.org>: <http://rubyonrails.org/documentation>
- Sass. (2011). *Sass - Syntactically Awesome Stylesheets*. Recuperado el 23 de Mayo de 2011, de Sass - Syntactically Awesome Stylesheets: <http://sass-lang.com/about.html>
- Teruel, A. (20 de Septiembre de 2000). *Justificaciones del diseño de Delta Pensum 1.x: Las clases Pensum, Calendario Curricular*. Recuperado el 22 de Febrero de 2010, de <http://www ldc.usb.ve/~teruel/ci4712/dp/dp1x/whyPensum.html>
- The jQuery Project. (2010). *Main Page - jQuery JavaScript Library*. Recuperado el 20 de Diciembre de 2010, de jQuery: The Write Less, Do More, JavaScript Library: [http://docs.jquery.com/Main\\_Page](http://docs.jquery.com/Main_Page)

## Anexo I: Editar Plan de Estudio

Inicio Preinscripción Inscripción Solicitudes Plan de Estudio Horarios



**Módulo Administrativo**  
 Rivas Sergio  
 V-13736933  
[Cerrar sesión](#)

**Editar Plan de Estudio** Lunes, 16 de Abril de 2011, 11:13 p.m.

[Datos del Plan de Estudio](#)

---

[Requisitos Curriculares](#)

Tipo de Asignatura	Mín Créditos	Mín Asignaturas	
COMPLEMENTARIA	8	3	X
ELECTIVA	40	8	X
LABORATORIO	3	1	X
OBLIGATORIA	107	21	X
OBLIGATORIA OPTATIVA	10	2	X
PASANTIA	5	1	X
SEMINARIO	5	1	X
SERVICIO COMUNITARIO	0	2	X
TRABAJO ESPECIAL DE GRADO	15	1	X

Para crear un requisito curricular nuevo haga click [aquí](#)

[Distribución de los Requisitos Curriculares](#)

---

[Opciones Profesionales](#)

---

CONEST Versión 3.0.0
Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2011

Inicio Preinscripción Inscripción Solicitudes Plan de Estudio Horarios



**Módulo Administrativo**  
 Rivas Sergio  
 V-13736933  
[Cerrar sesión](#)

**Editar Plan de Estudio** Lunes, 16 de Abril de 2011, 11:13 p.m.

[Datos del Plan de Estudio](#)

---

[Requisitos Curriculares](#)

---

[Distribución de los Requisitos Curriculares](#)

---

[Opciones Profesionales](#)

Nombre	
APLICACIONES EN INTERNET	X
BASES DE DATOS	X
COMPUTACION GRAFICA	X
INGENIERIA DE SOFTWARE	X
INTELIGENCIA ARTIFICIAL	X
TECNOLOGIAS EN COMUNICACION Y REDES DE COMPUTADORAS	X
SIN OPCION	X
SISTEMAS DE INFORMACION	X
TECNOLOGIAS EDUCATIVAS	X

Para registrar una opción profesional nueva haga click [aquí](#)

CONEST Versión 3.0.0
Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2011

## Anexo II: Crear Requisito Curricular

The screenshot shows the CONEST (Sistema de Gestión Académica) web application. The header includes the CONEST logo and navigation links for Universidad Central de Venezuela, Facultad de Ciencias, Acerca de, Contáctenos, and Ayuda. A secondary navigation bar contains menu items: Inicio, Preinscripción, Inscripción, Solicitudes, Plan de Estudio, and Horarios. The main content area is titled "Crear Requisito Curricular" and shows the current date and time: "Lunes, 16 de Abril de 2011, 11:13 p.m.". Below this, there is a section for "Datos del Requisito Curricular" with a message: "No hay ningún requisito curricular que crear. Para regresar haga click [aquí](#)". On the left side, there is a user profile section with a silhouette icon, the text "Módulo Administrativo", "Rivas Sergio", "V-13736933", and a "Cerrar sesión" link. The footer contains the version "CONEST Versión 3.0.0" and the copyright notice "Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2011".

## Anexo III: Editar Asignatura

Lunes, 16 de Abril de 2011, 11:13 p.m.

Inicio   Preinscripción   Inscripción   Solicitudes   Plan de Estudio   Horarios

---



Módulo Administrativo  
Rivas Sergio  
V-13736933

[Cerrar sesión](#)

### Editar Asignatura

[Datos de la Asignatura](#)

---

[Planes de Estudio](#)

Carrera	Plan de Estudio	Tipo de Asignatura	P	X
LICENCIATURA EN BIOLOGIA	PLAN BIOLOGIA	OBLIGATORIA	P	X
LICENCIATURA EN COMPUTACION	PLAN COMPUTACION	OBLIGATORIA	P	X
LICENCIATURA EN FISICA	PLAN FISICA	OBLIGATORIA	P	X
LICENCIATURA EN GEOQUIMICA	PLAN GEOQUIMICA	OBLIGATORIA	P	X
LICENCIATURA EN MATEMATICA	PLAN MATEMATICA	OBLIGATORIA	P	X
LICENCIATURA EN QUIMICA	PLAN QUIMICA	OBLIGATORIA	P	X

Para registrar un plan de estudio nuevo haga click [aquí](#)

[Prelaciones](#)

---

[Convalidaciones](#)

---

[Opciones Profesionales](#)

---

CONEST Versión 3.0.0      Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2011

Lunes, 16 de Abril de 2011, 11:13 p.m.

Inicio   Preinscripción   Inscripción   Solicitudes   Plan de Estudio   Horarios

---



Módulo Administrativo  
Rivas Sergio  
V-13736933

[Cerrar sesión](#)

### Editar Asignatura

[Datos de la Asignatura](#)

---

[Planes de Estudio](#)

---

[Prelaciones](#)

Carrera	Plan de Estudio	Requisito	X
LICENCIATURA EN BIOLOGIA	PLAN BIOLOGIA	8206 MATEMATICA I	X
LICENCIATURA EN COMPUTACION	PLAN COMPUTACION	8206 MATEMATICA I	X
LICENCIATURA EN FISICA	PLAN FISICA	8206 MATEMATICA I	X
LICENCIATURA EN GEOQUIMICA	PLAN GEOQUIMICA	8206 MATEMATICA I	X
LICENCIATURA EN MATEMATICA	PLAN MATEMATICA	8206 MATEMATICA I	X
LICENCIATURA EN QUIMICA	PLAN QUIMICA	8206 MATEMATICA I	X

Para crear una prelación nueva haga click [aquí](#)

[Convalidaciones](#)

---

[Opciones Profesionales](#)

---

CONEST Versión 3.0.0      Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2011