



UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS
ESCUELA DE COMPUTACIÓN

Automatización de los procesos
asociados a la Planificación Docente
en CONEST

Trabajo Especial de Grado presentado ante la ilustre
Universidad Central de Venezuela
por la bachiller:
Josefina Johana Montilla Medina
C.I. 19018395

Tutores:
Prof. Jossie Zambrano
Prof. Sergio Rivas

Caracas, octubre 2014

Dedicatoria

Dedico este logro inicialmente a ti Papá Dios por permitirme vivir esta experiencia, por estar a mi lado cada día, dándome salud, fortaleza, sabiduría y por tu inmenso amor, sin ti, mis metas no tendrían inicio ni final.

A mis padres por su amor y apoyo incondicional, por su confianza, por todo el esfuerzo y sacrificio realizado, por sus consejos, por la paciencia que han demostrado, por transmitirme sus valores de responsabilidad y compromiso. Este triunfo es de Ustedes!

A mis hermanos en especial a ti Richard Jose M. Este logro es compartido, es un obsequio para ti. Gracias por los momentos especiales y hacerme sentir en tus últimos días que era parte importante de tu vida. Te Amo hermano espero que estés orgulloso de mi. Para ti también “Gordita, Crisbelina” Todo se puede alcanzar con esfuerzo y dedicación nunca te rindas, lucha por tus sueños.

Y finalmente a Ana Montilla por estar a mi lado y aunque no entendiera nada del tema disponía de su tiempo para leer mis redacciones, fuiste mi apoyo aun cuando no creía en mí.

Josefina Montilla

Agradecimientos

Agradezco a mi fuente de inspiración, a la mujer que día a día me llena de orgullo que sin darte cuenta me diste valor y fuerzas para continuar desde el inicio de este proceso, gracias por tu apoyo en todos los ámbitos. Te amo mamá

A ti Papá por apoyar siempre en mis estudios, por inculcarme su importancia y aun más por tus frases peculiares que me incentivaban. Gracias por todo Te quiero un montón.

A ti “Tía Vicki” por ser mi segunda madre, gracias por recibirme y adoptarme como una hija más en tu hogar, por tus cuidados y esfuerzos.

A ti Ana Montilla por ser mi luz en los días oscuros, porque indirectamente fuiste mi compañera de trabajo, por tu apoyo, por soportarme en los días de frustración y de mal humor. Gracias por estar allí cuando lo necesite “PIO PIO”!.

A ti “Tía Alexis” por darme alojamiento en tu hogar en mis primeros pasos en la universidad.

A mis hermanos.

A la Sra Julia y Anjuly Lugo por su ayuda, colaboración, por ser parte importante de mi formación y de mi vida. Gracias por todo!

A mis tutores Jossie Zambrano y Sergio Rivas por cada uno de los conocimientos transmitidos, por impulsarme en este proyecto, por orientarme, por la firmeza, dedicación y exigir más de mí. Gracias son excelentes personas, profesionales y amigos!

A la Casa que Vence las Sombras la UCV, a la Facultad de Ciencias, a la División de Control de Estudios especialmente a Milagros Graterol, Dinora Moreno por su orientación, ayuda y consejos.

A mis amigos del liceo especialmente al grupo de 9^o “C”, por formar parte de esta preparación.

Gracias a todas esas personas que formaron parte de mi vida durante mi carrera universitaria y a todos aquellos que alguna vez me apoyaron directa o indirectamente. Gracias!.

Josefina Montilla

Resumen

El inicio de un período académico en cualquier institución contempla un determinado número de procesos, los cuales se caracterizan por su complejidad y flujo de información, en este sentido la gestión académica es la más importante y para hacerla posible es necesario iniciar una Planificación Docente. De manera que la distribución y organización de los docentes, estudiantes, asignaturas y horarios recae sobre esta. Por ello es significativo disminuir el margen de error por la cantidad de personas involucradas, los tiempos de respuesta que implica hacerlo de forma manual, así como la cantidad de material y recursos físicos utilizados.

El objetivo de este Trabajo Especial de Grado consiste en automatizar los procesos asociados a la Planificación Docente, con la finalidad de generar un impacto positivo en los tiempos de respuesta y reducir la brecha de comunicación entre las dependencias de las escuelas y la División de Control de Estudios (DCE). La solución formará parte de la nueva versión de CONEST, sistema que es utilizado por la Facultad de Ciencias de la Universidad Central de Venezuela (UCV) para la gestión académica a nivel de Pregrado desde el año 2007.

El método de desarrollo de software utilizado para alcanzar el objetivo de esta investigación, es una adaptación de AgilUs. Y entre las tecnologías para la plataforma destacan: El lenguaje de programación Ruby 2.0, su framework Ruby on Rails 4.0 y el sistema manejador de bases de datos MySQL. Asimismo, esta investigación contempla el estudio de CONEST 1.0, tomando en cuenta la problemática que acarrea entorno al enfoque de este trabajo, permitiendo así encontrar una solución que ayude, facilite y fortalezca las actividades del personal involucrado.

Palabras clave: Aplicaciones Web, Gestión, Planificación, AgilUs, DCE, CONEST

Índice general

Introducción	10
1. Planificación Docente	13
1.1. Proceso de la Planificación Docente en Pregrado	13
1.1.1. Ofertada	16
1.1.2. Ejecutada	17
1.2. CONEST	17
1.2.1. Características de CONEST 3.0	18
1.2.2. Módulos de CONEST 3.0	18
2. Tecnologías y Herramientas	20
2.1. Tecnologías Web	20
2.1.1. HTML5	21
2.1.2. HAML (HTML Abstracción Markup Language)	23
2.1.3. JavaScript	26
2.1.4. JQuery	28
2.1.5. Ruby on Rails	30
2.1.6. MySQL	32
2.2. Herramientas	34
2.2.1. Asana	35
2.2.2. Zoho Projects	36

3. Método AgilUs y Desarrollo de Software	38
3.1. Método AgilUs	38
3.1.1. Diseño centrado en el usuario	39
3.1.2. Principios	42
3.1.3. Buenas prácticas	42
3.1.4. Ciclo de vida	43
3.2. Desarrollo del Módulo Planificación Docente	44
3.2.1. Iteración I Planificación Docente	45
3.2.2. Iteración II Gestión de la Planificación Docente	59
3.2.3. Iteración III Reportes y Horarios	75
3.2.4. Iteración IV Estado de la Calificación	86
3.3. Caso Estudio	97

Índice de figuras

1.1.1. Planificación tentativa de la escuela de Computación II-2010	14
1.1.2. Planificación docente de la Escuela de Biología I-2009	14
1.1.3. Flujo de Información de la Planificación Docente en CONEST 1.0.	16
2.1.1. Tecnologías del Sistema CONEST 3.0	21
2.1.2. Estructura semántica de html5 & HTML	22
2.1.3. Estructura básica en HTML5 y el equivalente en HAML	25
2.1.4. Ejemplo de código en ERB y su equivalente HAML	26
2.1.5. Código en JavaScript y su equivalente en CoffeeScript	27
2.1.6. Tema de JQuery color naranja.	28
2.1.7. Aplicación tradicional y Aplicación con Ajax	30
2.2.1. Interfaz de Asana	35
2.2.2. Vista general de Zoho Projects	37
3.1.1. Ciclo de vida del método AgilUs [1]	44
3.2.1. Vista de la Planificación Docente en CONEST 1.0	46
3.2.2. Vista de Planificación docente CONEST 1.0	47
3.2.3. Caso de Uso de la Planificación Docente, nivel 1	49
3.2.4. Modelo de Objetos del Dominio	50
3.2.5. Flujo de información de la Planificación Docente en CONEST 3.0.	51
3.2.6. Prototipo en Papel	52
3.2.7. Prototipo en Papel	52
3.2.8. Logotipo de CONEST 3.0	53
3.2.9. Colores Principales	53

3.2.10	Plantilla para la Planificación Docente	54
3.2.11	Prototipo ejecutable de los períodos académicos	55
3.2.12	Prototipo ejecutable del listado de las licenciaturas	56
3.2.13	Prototipo rápido del listado de las asignaturas	57
3.2.14	Interfaz de Usuario para visualizar una sección	58
3.2.15	Hoja de estilo	59
3.2.16	Notas de una tormenta de ideas Iteración II	60
3.2.17	Caso de Uso de la Planificación Docente, nivel 2	62
3.2.18	Prototipo rápido para agregar un periodo académico	63
3.2.19	Método para el número de secciones y capacidad	64
3.2.20	Prototipo rápido para agregar una asignatura	65
3.2.21	Prototipo para cerrar una asignatura	66
3.2.22	Prototipo rápido para el grupo docente	66
3.2.23	Prototipo rápido para cerrar asignatura	67
3.2.24	Ventana modal para agregar un período académico	70
3.2.25	Ventana modal para agregar una asignatura	71
3.2.26	Ventana modal para cambiar las licenciaturas	71
3.2.27	Ventana modal para modificar la capacidad	72
3.2.28	Ventana modal para el coordinador de la asignatura	72
3.2.29	Ventana modal para el cierre de asignatura con estudiantes	73
3.2.30	Ventana modal para modificar el nombre de la sección	73
3.2.31	Ventana modal para agregar grupo docente	74
3.2.32	Ventana modal para cerrar una sección	74
3.2.33	Tormenta de Ideas Iteración III	76
3.2.34	Prototipo en papel horarios en la vista de asignaturas	77
3.2.35	Resumen de la Planificación Docente en PDF	78
3.2.36	Plantilla para Documentos PDF	79
3.2.37	Resumen de la Planificación Docente en PDF	80
3.2.38	Ventana modal para el horario una asignatura	80
3.2.39	Prototipo rápido de listado de sección en PDF	81

3.2.4	Listado de sección en PDF	82
3.2.4	Listado de secciones en Excel	83
3.2.4	Resumen de Planificación Docente	84
3.2.4	Ubicación de la opción Resumen de Horarios	85
3.2.4	Resumen de Horarios interfaz	85
3.2.4	Tormenta de Ideas Iteración IV	87
3.2.4	Colores empleados para indicadores	87
3.2.4	Prototipo rápido del estatus de la calificación	88
3.2.4	Barra de progreso de calificación.	89
3.2.4	Prototipo rápido de configuración de asignatura	90
3.2.5	Interfaz para configurar una asignatura	91
3.2.5	Interfaz para visualizar las asignaturas en preinscripción	92
3.2.5	Listado de secciones sin calificar por licenciatura	93
3.2.5	Resultados de la prueba de aceptación beta	95
3.2.5	Interfaz para visualizar los periodos académicos	96
3.2.5	Interfaz para visualizar las licenciaturas	96
3.3.1.	Datos de la Planificación para el período 01-2014	98

Índice de cuadros

3.1. Análisis del sistema CONEST para su versión 1.0.	46
3.2. Evaluación heurística para la iteración II de Gestión	69
3.3. Encuesta aplicada en la prueba de aceptación	94

Introducción

Actualmente, la información se ha convertido en un recurso vital para las organizaciones y con el transcurrir del tiempo, son más grandes los volúmenes de datos e información disponibles, por ende, debe ser manipulada de forma correcta. En este sentido surge la necesidad de desarrollar sistemas automatizados que permitan disminuir esfuerzos, tiempo y costos; dando respuestas a necesidades particulares en el margen de eficiencia. Tales sistemas han demostrado su influencia y uso en todos los ámbitos: social, cultural, económico y educativo; particularmente este último ha aprovechado los beneficios de este recurso tanto a nivel laboral como académico.

Desde el año 2007, la Facultad de Ciencias de la Universidad Central de Venezuela se ha sumado a la implementación de aplicaciones Web para el cumplimiento de sus funciones internas. Estas aplicaciones se destacan de otros sistemas por su amplio alcance dentro del campo de la información y las comunicaciones, además no requieren de la instalación de un software, ni necesitan de un sistema operativo en específico y pueden ser accesibles desde cualquier dispositivo con acceso a Internet.

En la actualidad, la Facultad de Ciencias provee formación de pregrado a aproximadamente 3200 estudiantes, que pertenecen a las cinco escuelas, las cuales son: Biología, Computación, Física, Matemática y Química. Cada una de ellas constituye una licenciatura, a excepción de la escuela de Química que posee dos licenciaturas: Química y Geoquímica. Al igual que cualquier otra institución educativa, es vital realizar la gestión académica para facilitar la transformación de las condiciones institucionales, en solución a problemas o respuestas a interrogantes que surgen del proceso administrativo.

Para llevar a cabo la gestión académica la Facultad cuenta con la División de Control de Estudios (DCE), esta debe coordinar y organizar el plan de estudio de las distintas escuelas con los recursos disponibles, tales como: docentes, asignaturas, aulas, horarios y estudiantes. Por este motivo debe disponer de la Planificación Docente de cada escuela para dar inicio a un nuevo período académico y a los procesos que de él se derivan, como la preinscripción, inscripción y posteriormente la calificación.

Sin embargo, desarrollar una Planificación Docente es una tarea ardua que tiende a ser complicada y llena de dificultades. Una de las principales dificultades que puede conseguir

el jefe de departamento, quien es el responsable de este proceso, es coordinar a los docentes en las distintas asignaturas, acordar un horario para dictarlas, garantizar que las asignaturas del mismo semestre no solapen sus horarios, asimismo, debe determinar que asignaturas estarán en el proceso de preinscripción y establecer el número de secciones a ofertar. En ocasiones esta planificación puede tardar mucho tiempo en ser culminada por el número de personas involucradas, aumentando margen de error, los recursos físicos, materiales y tiempos de respuesta.

Después de finalizar con el proceso de desarrollo de una planificación, los jefes de departamentos deben consignar a la DCE el documento correspondiente, sin embargo, el personal de esta entidad recibe este documento en formatos diferentes en cuanto a organización y estructura de datos siendo poco legible, además, de contener datos que no son interés para el personal y en ocasiones con errores en los nombres y en los códigos de las asignaturas. Conjuntamente a este conflicto se suman los procedimientos de actualización de horarios o secciones de una asignatura por no ser eficientes, ya que, demoran mucho tiempo en ser informados a la DCE.

Otra dificultad que está ligada a la Planificación Docente, es la incertidumbre que posee el estudiante al momento de inscribir un nuevo período académico; debido a que no posee información de las asignaturas a ofertar, el docente, el horario y si requiere preinscripción o no, esta situación es una constante en este proceso. Y ciertamente conocer la planificación con anterioridad le permite al estudiante organizar las asignaturas acorde a sus necesidades, reduciendo el número de solicitudes estudiantiles que posteriormente se efectúan.

Dada su importancia para articular la gestión académica y a fin de solventar las dificultades expuestas, es indispensable automatizar los procesos asociados a la Planificación Docente, con la finalidad de generar un impacto positivo en los tiempos de respuesta y reducir la brecha de comunicación entre las dependencias de las escuelas y la DCE, siendo este el objetivo general de la investigación.

Para alcanzar este propósito, se presentan los siguientes objetivos específicos:

- Gestionar la Planificación Docente en términos de configuración, permitiendo la apertura de un nuevo período académico, ofertar y cerrar asignaturas, secciones, horarios, aulas y grupo docente, al igual que su modificación.
- Diseñar el flujo de información de la Planificación Docente entre las escuelas y la DCE, proponiendo un rol más participativo de los jefes de departamento en el sistema.
- Determinar un conjunto de asignaturas correspondientes a las licenciaturas, en la apertura de un nuevo período académico.
- Generar reportes y los documentos de interés para la Planificación Docente.

- Aplicar el método de desarrollo de software, AgilUS, dando lugar al levantamiento de información, a fin de identificar el alcance y los requerimientos de usuarios.

Por otra parte, tanto la configuración de las asignaturas como la administración y organización de los recursos que intervienen en la Planificación Docente son imprescindibles y de suma importancia para el logro de los objetivos.

Es significativo resaltar que la solución a desarrollar forma parte de la nueva versión de CONEST, sistema que es utilizado por la DCE y a través del cual interactúan docentes, estudiantes y personal administrativo, por esta razón es ideal, el método de desarrollo de software AgilUs, ya que, está orientado a integrar al usuario como parte del equipo de desarrollo desde las primeras etapas del ciclo de desarrollo permitiendo obtener un producto usable y adaptable a los requerimientos de usuarios. Adicionalmente, la plataforma del sistema debe estar a la vanguardia de las tecnologías, así como eliminar las fallas que posee la antigua versión.

Para cumplir con los objetivos anteriores se presenta el documento con la siguiente estructura:

Capítulo 1 Planificación Docente, muestra la descripción amplia del proceso que involucra a la Planificación Docente, así como las entidades participantes, a nivel de pregrado.

Capítulo 2 Tecnologías y Herramientas, presenta el conjunto de tecnologías asociadas al entorno de diseño y al desarrollo de la aplicación.

Capítulo 3 Método AgilUs y Desarrollo de Software, se divide en dos secciones, la primera contempla de manera general el método aplicado, AgilUS. Mientras que la segunda especifica el proceso de desarrollo de la solución.

Finalmente, se exponen los resultados obtenidos, las conclusiones, recomendaciones y las referencias consultadas durante este Trabajo Especial de Grado.

Capítulo 1

Planificación Docente

La Planificación Docente es un mecanismo que permite articular la gestión académica en un nuevo período lectivo, a través de la cual se consigue organizar un conjunto de recursos, estos dependen de cada institución y sus normativas, entre los cuales se destacan: Las asignaturas, secciones, aulas, horarios, docentes y estudiantes. Dicha planificación es presentada mediante un documento y puede estar sujeta a estudio, modificación y actualización.

Este capítulo, describe el proceso de Planificación Docente a nivel de Pregrado. Igualmente hace mención de las etapas, entidades involucradas y la importancia que implica realizar una buena planificación. Asimismo, se presenta el sistema CONEST.

1.1. Proceso de la Planificación Docente en Pregrado

Los responsables de desarrollar e integrar la Planificación Docente, son los jefes de departamentos; su objetivo es organizar a los factores que intervienen en un nuevo período lectivo, asimismo deben consignar a la DCE el documento asociado a la planificación, con el listado de las asignaturas tentativas, el número de secciones, la capacidad de estudiantes, el docente y el horario respectivo. Actualmente las escuelas emiten este documento en formatos distintos, a pesar que el producto final es el mismo. Véase ejemplo de ello, en las figuras 1.1.1 y 1.1.2.

PROGRAMACION DOCENTE TENTATIVA II-2010 (02/06/2010)																CNU+Conv+PI		Aplaz+Ret		Total	
Asignaturas Obligatorias																					
Sem.	Cód.	Asignatura	UC	Profesor(es)	Rol	Est.	Secciones			Prep.		Obs.									
SEMESTRE I																					
I	6201	Algoritmos y Programación	6	Yusneyi Carballo (DE)	T+C	266	T	P	L	I	II	44,3	210	102	312						
				Ivan Flores (DE)	T																
				Sup1 V. Leguizamo	T																
				Sup2 V. Leguizamo	T																
				Sup2 W. Hernández	T																
					T																
I	6301	Introducción a la Informática	4	Yosly Hernández	C	246	6	10		2	2	41,0	210	41	251						
				Mayerling Marquez (Cont. DE)	T																
				Eudes Carrero	T																
				Silvestre Jimenez (Inst. Conv. 6hrs.)	T																
				Rossana Díaz (Inst. Conv. 6hrs.)	T																
				Silvia Rivero	T																
				Concurso Instructor1 (DE) CISI	T																
I	6106	Matemáticas Discretas I	4	Luis Manuel Hernández (DE)	T+C	273	6	11		5	1	45,5	210	120	330						
				Carenne Ludeña (Agreg. 6 hrs.)	T																
				Ronald Pietri (Inst. Conv. 6hrs.)	T																
				Sergio Velasquez	T																
				Sup2 H. Navarro	T																
				Suplente2 Guy Vernahez	T																
3							12					6									
Asignaturas Obligatorias																					
Sem.	Cód.	Asignatura	UC	Profesor(es)	Rol	Est.	Secciones			Prep.		Obs.									
SEMESTRE II																					
II	6202	Algoritmos y Estructuras de Datos	5	Robinson Rivas (DE)	C	170	4	7		2	2	42,5	34	113	147						
				Karla Rivas (Cont. 6hrs.)	T																

Figura 1.1.1: Planificación tentativa de la escuela de Computación II-2010

Código		Asignatura	Num. Est.	Num. Secc.	Jurado	EMAIL
Cód. Tipo		Nombre	Est.	Secc.	Nombre	COORD. DE LA ASIGNATURA
1902	O	Principios de Biología (4 UC)	100	2	Pérez, María C	6555555
		Requisito:			García, Juan	3333333
		2401 Principios de Química I			Medina, Luis	8993444

Se debe indicar si la programación es:
A Ejecutar
En Ejecución

Programación Docente I-2009

Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Biología
DIVISIÓN DE CONTROL ESTUDIOS
Departamento:

Código de la asignatura
Tipo de Asignatura:
O: Obligatoria
E: Electiva

Se debe indicar el nombre de la asignatura, unidades de crédito y sus respectivos Requisitos

Número de estudiantes

Nombre y C.I. del Coord. y del jurado que firmará la(s) planilla(s) de Notas finales. Se debe señalar el Coordinador con (c)

Correo Electrónico del Coordinador de la asignatura.

Página 1

Control de estudios Prog. Asig Distribución Sabana I-2008 Hoja1

Figura 1.1.2: Planificación docente de la Escuela de Biología I-2009

El punto de partida para desarrollar e integrar una Planificación Docente, es coordinar a los docentes en las diferentes asignaturas de la escuela y acordar un horario para las mismas, se debe garantizar que los horarios de las asignaturas de un mismo semestre no se solapen, de igual forma el horario de las asignaturas obligatorias tienen prioridad en el orden de asignación sobre las electivas, además, es esencial identificar las asignaturas que estarán

disponibles en la preinscripción, ya que, la misma es utilizada con diferentes objetivos. Entre los cuales se puede mencionar:

- La elección de estudiantes en base a un conjunto determinado de criterios, debido a que algunas asignaturas electivas cuentan con cupo limitado.
- Medir la cantidad de estudiantes preinscritos en un asignatura, esto permite conocer si el número de estudiantes es suficiente para la apertura de una sección en el proceso de inscripción.
- Asignar horarios correspondientes a una asignatura, el estudiante se postula por uno o varios horarios en particular.

En segunda instancia, están las estimaciones tales como: el número de secciones y la cantidad de estudiantes en una sección. Este estimado debe ser ajustado a las necesidades del estudiantado, de manera que los cambios ocurridos después en el proceso de inscripción no impacten fuertemente en la gestión académica. Si la estimación de las secciones es muy grande y por razones ajenas no se dispone de un docente o no se inscriben estudiantes en la sección, se debe permitir el cierre de las secciones innecesarias, así como también debe ser posible reubicar a los estudiantes si es necesario.

Por otra parte, es posible que el comportamiento de la planificación sea semejante al anterior, por ello, los jefes de departamentos utilizan el concepto de semestres pares e impares, el semestre llamado “par” es el período académico donde la facultad tiene el mayor índice de estudiantes por nuevo ingreso. Por ende, realizan la comparación de la cantidad de inscritos, retirados, aplazados y de nuevo ingreso si aplica, con el semestre anterior, verificando que tenga la misma condición del actual (par o impar), para que la estimación sea la más ajustada.

Particularmente, se ejemplifica un estimado del número de secciones para la asignatura Elementos de la Matemática, generalmente puede existir una demanda que genere la apertura de dos (2) secciones. Para el resto de las asignaturas en la escuela de Matemática, la cantidad de estudiantes inscritos no amerita abrir más de una sección.

El plan de estudio de algunas escuelas se asemeja en los primeros semestres, es por este motivo que poseen asignaturas en común, sin embargo estas son coordinadas por una sola escuela por lo tanto debe contemplar la demanda del resto de las escuelas, es decir, cada escuela tendrá sus propias secciones, este tipo de asignaturas se conocen como de servicio. Un ejemplo, es el caso de “Física I” que debe tener la cantidad de secciones necesarias para las escuelas de Química, Biología y Matemática, aunque la escuela de Física es la responsable de asignar a los profesores y de establecer el horario para estas asignaturas. Otro caso de asignaturas de servicios son: Matemáticas I, II, III, IV y Métodos Matemáticos. Para “Matemática I” no se realizan divisiones de secciones por escuela.

En el proceso de desarrollo de una planificación están otros aspectos a considerar, entre ellos: Coordinar a los docentes en las asignaturas, identificar cuales requieren de preinscripción y la asignación de horarios, sin embargo, en esta última las asignaturas obligatorias tienen prioridad sobre cualquier otra y la responsabilidad de los horarios para las obligatorias recae sobre la DCE, la asignación se realiza evitando solapamientos entre asignaturas de un mismo semestre y las adyacentes a este, tratando de beneficiar al alumnado de la sección en general. Por otro lado la DCE espera realizar la inscripción, para iniciar la asignación de aulas.

Las consideraciones antes expuestas, permiten que los jefes de departamentos dispongan de los datos necesarios para crear el documento asociado a la planificación. No obstante, esta planificación puede dividirse en dos fases: ofertada y ejecutada, descritas a continuación.

1.1.1. Ofertada

Esta etapa se identifica, cuando los jefes de departamentos de cada escuela, realizan el documento correspondiente a la Planificación Docente para su posterior estudio y evaluación por el Consejo de Escuela, si la misma es aprobada se consigna a la DCE a través de una copia impresa. La DCE se encarga de acoplar el listado de las asignaturas al sistema CONEST. Véase en la siguiente figura 1.1.3.

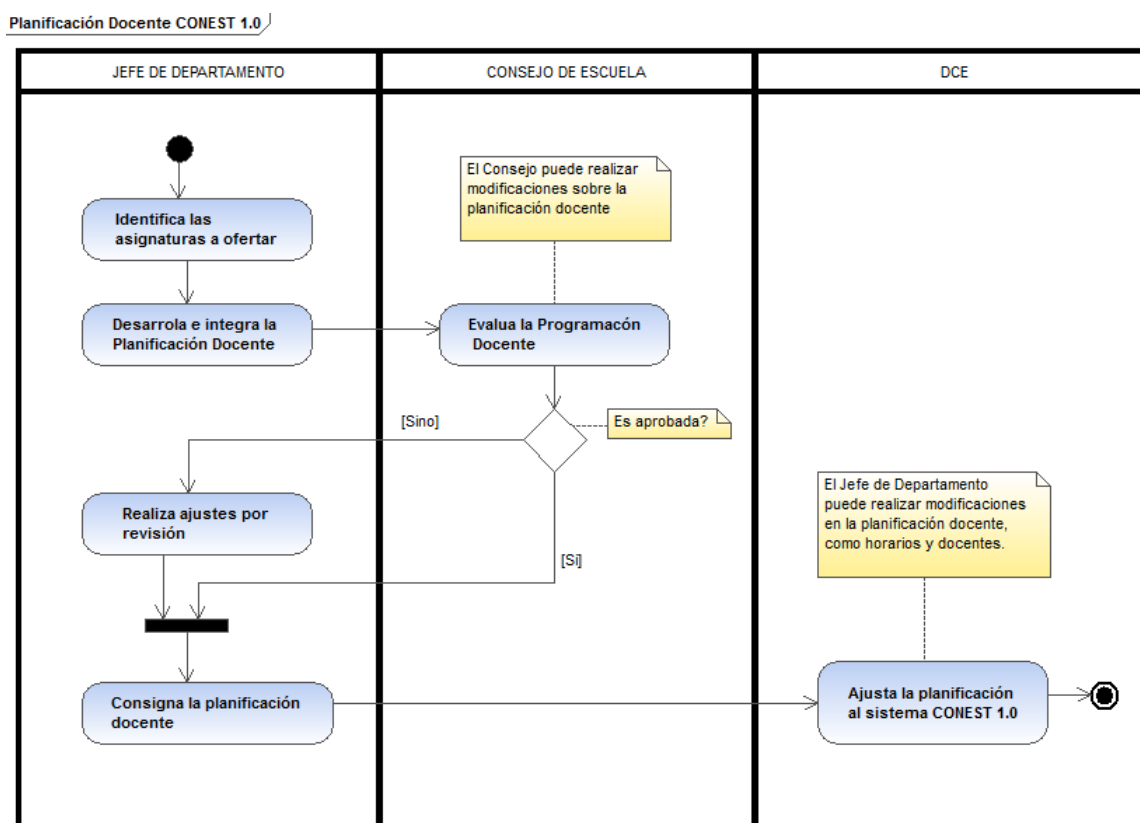


Figura 1.1.3: Flujo de Información de la Planificación Docente en CONEST 1.0.

Finalmente se realiza el proceso de preinscripción y después de obtener los resultados se verifica si existen ajustes que realizar, confirmando a la DCE para iniciar el proceso de inscripción del nuevo período académico. Es importante destacar que durante esta etapa la planificación puede estar sometida a actualizaciones.

1.1.2. Ejecutada

En esta etapa, la Planificación Docente debe contemplar a todos los docentes correspondientes a las secciones de cada asignatura, junto con los docentes que se desempeñan como jurados evaluadores, de manera que permita mantener el control y poder ejecutar el proceso de calificación, obteniendo sus respectivas actas.

Durante esta etapa el jefe de departamento puede realizar modificaciones solo en la asignación de docentes, con la única restricción que debe ser antes de la calificación.

Es importante resaltar que la Planificación Docente es un elemento importante y necesario para articular la gestión en cualquier período académico. Por ello, puede estar sujeta a revisión, modificación, ampliación y actualización, no es estática; puede evolucionar en función de la información que se obtiene sobre los resultados parciales del desarrollo de un período académico.

1.2. CONEST

En esta sección se describe al Sistema de Control de Estudios (CONEST) de la Facultad de Ciencias, a nivel general, haciendo mención de su estructura, características más relevantes y descripción de sus módulos.

CONEST es el Sistema de Gestión Académica de la División de Control de Estudios de la Facultad de Ciencias, cuyo objetivo principal es automatizar los procesos administrativos de la gestión académica. La visión es apoyar el cumplimiento de las normas académicas a través de las tecnologías de información y comunicación (TIC). Este sistema se encuentra operativo desde el año 2007 con la participación de estudiantes, docentes y personal administrativo de esta comunidad. Es un desarrollo con tecnologías de software libre y como repositorio de datos MySQL, siendo modificable y adaptable a las necesidades del usuario.[5]

CONEST es la plataforma donde es posible administrar y gestionar toda la información involucrada en los procesos administrativos y académicos del personal docente y estudiantil, además, mantiene una relación con las dependencias internas, externas y escuelas de esta Facultad, para llevar a cabo sus tareas principales como lo son inscribir, reincorporar y calificar a los estudiantes, administrar la planta física donde se imparten clases, emitir documentos y reportes, entre otras.[3]

CONEST presta servicios las 24 horas del día, los 365 días del año, en los cuales se puede acceder a todas sus funcionalidades de manera remota sin la instalación de ningún tipo de software especial. En sus labores diarias; permite conocer el estado de los procesos de inscripción, grado y calificación, así como también el envío de información a los estudiantes y docentes a través de correos electrónicos.[3]

La complejidad en los procesos y la necesidad de nuevos requerimientos en el sistema ha generado un crecimiento poco planificado y desmesurado, como consecuencia, redundancia en las funcionalidades, fallas operativas, inconsistencias en la interfaz, desorganización del código, altos costos en tiempo y esfuerzo humano, mantenimiento correctivo y carencia de pruebas rigurosas. Por esta razón, surge la nueva versión de CONEST 3.0 cuyo objetivo es mejorar los procesos de la gestión académica, organizar la interfaz de usuario, actualizar las tecnologías y la plataforma, mejorando el rendimiento del sistema, así como la comunicación entre las dependencias de las escuelas y DCE.

1.2.1. Características de CONEST 3.0

- Implementado en Ruby on Rails bajo la arquitectura Modelo, Vista y Controlador (MVC). Herramientas de software libre.
- Manejado con la herramienta GIT que permite a los desarrolladores establecer sincronización de versiones del código y que trabajen en ambientes colaborativos donde pueda aprender del trabajo realizado por otros integrantes del grupo.[5]
- La interfaz de usuario de CONEST es común para todos los módulos en cuanto a la disposición de las funcionalidades.[3]
- Incentiva la participación de forma activa de Estudiantes, Docentes y Personal Administrativo que forman parte de la comunidad de la Facultad de Ciencias.[5]

1.2.2. Módulos de CONEST 3.0

CONEST 3.0 cuenta con un conjunto de módulos orientados a ofrecer servicios a los usuarios, considerando los diferentes roles que cumplen los usuarios en el sistema académico de la Facultad de Ciencias. A continuación se describen los módulos:

- Módulo Servicio al Estudiante: Este módulo ofrece varias funcionalidades tales como inscripción de asignaturas de un período académico, consulta del horario del semestre en curso, del expediente curricular, constancia de inscripción, chequeo de requisitos de pensum, retirar asignaturas y solicitud de constancia, entre otras.

- **Módulo Servicio al Docente:** Se efectúan acciones de calificación vía Web, lo que ayuda a recoger y centralizar la información de los docentes, este módulo además permite al docente descargar en formato Excel y PDF los listados de los estudiantes inscritos en las secciones de su correspondiente licenciatura, al igual que consultar el expediente curricular de tales estudiantes. Cabe destacar que si el docente cumple con el rol de jefe de departamento, adicionalmente, tiene las opciones de gestionar las asignaturas electivas, horarios y el grupo docente de la Planificación Docente, como descargar el resumen correspondiente en PDF.
- **Módulo de Ingreso:** Este gestiona el proceso de inscripción de los nuevos estudiantes en la Facultad de Ciencias. Ellos a través de un formulario introducen los datos personales y académicos necesarios para formalizar la inscripción, finalizado esto, el nuevo estudiante es inscrito en el primer semestre de su licenciatura [5].
- **Módulo de Administración:** Este módulo es utilizado por el personal administrativo de la DCE, consolidando todas las funciones de las instancias que han sido desarrolladas, es decir, se integra la administración de todos los módulos. Se encuentra en constante actualización y ofrece funcionalidades que dependiendo del rol que cumpla el personal, se despliegan algunas u otras. Estas se encuentran organizadas en tres grupos, los cuales contienen procesos relacionados a la asignación y optimización de las plantas físicas de la facultad para impartir clase de pregrado, los relacionados con los actos de grado y los relacionados con actividades administrativas en general. [5]

A continuación se describe el capítulo 2, destacando las tecnologías Web utilizadas para la plataforma y las herramientas que facilitaron el proceso de desarrollo.

Capítulo 2

Tecnologías y Herramientas

Este capítulo, presenta un conjunto de tecnologías y herramientas de software libre utilizadas en un entorno Web, siendo de interés y aplicables en la solución. Entre las tecnologías están: El lenguaje de marcado HTML5, HAML, JavaScript (CoffeeScript), JQuery (AJAX), Ruby on Rails y el sistema manejador de bases de Datos MySQL. Esta elección está basada en la plataforma sobre la cual se ha levantado el sistema CONEST 1.0, con la distinción de emplear versiones más recientes. Dentro de este marco también se describen algunas herramientas relacionadas, que no son propias de la aplicación pero facilitan el desarrollo y seguimiento del proyecto, entre ellas Asana y Zoho Projects.

2.1. Tecnologías Web

Las Tecnologías Web son un conjunto de técnicas que hacen posible desarrollar bienes y servicios, cubriendo las necesidades humanas y facilitando la adaptación al medio ambiente. Dando mejores resultados al momento de desarrollar aplicaciones Web.

El uso de la Tecnología Web ha creado un nuevo paradigma en el manejo de la información, permitiendo la distribución de información en hipertexto o hipermedios enlazados y accesibles a través de Internet, impulsado la creación de páginas, aplicaciones, sitios y portales. La siguiente figura 2.1.1 muestra las tecnologías aplicadas en el sistema CONEST.

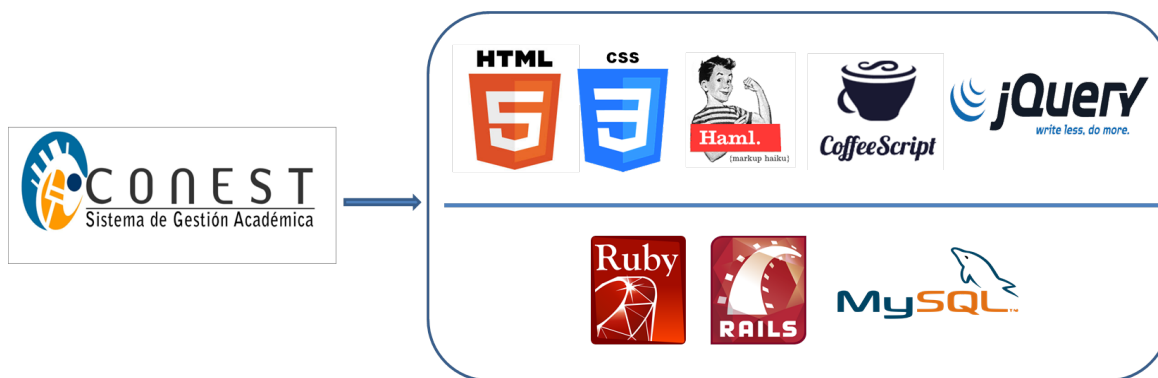


Figura 2.1.1: Tecnologías del Sistema CONEST 3.0

2.1.1. HTML5

HTML es el lenguaje con el que se escribe la estructura y la semántica del contenido de un documento Web.

HTML5 (*HyperText Markup Language*, versión 5) es la quinta revisión importante del lenguaje básico de la *World Wide Web*, HTML. HTML5 especifica dos variantes de sintaxis:

- Un «clásico» HTML (text/html), la variante conocida como HTML5.
- Una variante *XHTML* conocida como *XHTML5* que deberá ser XML (*XHTML*) (application/xhtml+xml).

Esta es la primera vez que HTML y *XHTML* se han desarrollado en paralelo. El desarrollo de este código es regulado por el Consorcio W3C desde el 2007.

2.1.1.1. Estructura Semántica

HTML5 proporciona varios elementos que perfeccionan la estructura en bloques, eliminando así DIV innecesarios. Este cambio en la semántica hace que la estructura de la web sea más coherente y fácil de entender por otras personas y los navegadores. En la Figura 2.1.2 se compran los nuevos elemento en HTML5.



Figura 2.1.2: Estructura semántica de html5 & HTML

2.1.1.2. Nuevos Elementos

- `<section></section>` Se utiliza para representar una sección “general” dentro de un documento o aplicación. Puede contener subsecciones.
- `<article></article>` Representa un contenido autónomo e independiente en un documento, página, aplicación, o sitio web con la intención de que pueda ser reutilizado y repetido.
- `<aside></aside>` Representa un contenido que está muy poco relacionado con el resto de la página, como una barra lateral.
- `<header></header>` Representa la cabecera de una sección.
- `<footer></footer>` Representa el pie de una sección, con información acerca de la página/sección que poco tiene que ver con el contenido de la página.
- `<nav></nav>` Representa una sección dedicada a la navegación.

El elemento `input`, adquiere relevancia, el cual ha sido ampliado, otorgando facilidades para validar el contenido sin JavaScript. Se permite todos estos tipos de dato:

- *Datetime*, `datetime-local`, `date`, `month`, `week`, `time`: Para que indicar una fecha/hora.

- *Number*: Para que el usuario indique un número.
- *Range*: Para indicar un rango entre dos números.
- *Email*: Para indicar un correo electrónico.
- *Url*: Para indicar una dirección web.
- *Search*: Para indicar una búsqueda.
- *Color*: Para indicar un color.

Otros elementos nuevos:

- *<audio>* y *<video>* Nuevos elementos que permitirán incrustar un contenido multimedia de sonido o de vídeo, respectivamente. Es una de las novedades más importantes e interesantes en este HTML5, ya que permite reproducir y controlar vídeos y audio sin necesidad de plugins como el de Flash.
- *<canvas>* Este es un elemento complejo que permite generar un lienzo de mapa de bits dependiente de la resolución, que permite dibujar gráficos de alta resolución, gráficos de juegos u otros efectos visuales. A través de un API se puede, mediante JavaScript, pintar todo tipo de formas gráficas para componer o mezclar imágenes. Se puede responder a eventos de usuario que por su habilidad para actualizar su contenido en tiempo real nos permite crear aplicaciones web enriquecidas en el lado del cliente similares a las que se pueden realizar con un plugin externo como Flash. [20]

2.1.1.3. Etiquetas eliminadas en HTML5

Las siguientes etiquetas están eliminadas en la nueva versión de HTML5 [20]:

- *<acronym>* *<applet>* *<basefont>* *<big>* *<xmp>* *<center>* *<dir>*
- ** *<frame>* *<u>* *<frameset>* *<noframes>* *<strike>* *<tt>*

2.1.2. HAML (HTML Abstracción Markup Language)

Haml es un lenguaje de marcado que se utiliza para describir de manera limpia y simple el código HTML de cualquier documento web, sin el uso de código embebido tradicional, eliminando la redundancia. Haml funciona como un reemplazo a los sistemas de plantillas de páginas embebidas, tales como PHP, ERB y ASP. Sin embargo, Haml evita la necesidad de codificar explícitamente HTML, porque en realidad es una descripción abstracta del HTML, con algo de código para generar contenido dinámico [9].

2.1.2.1. Principios de HAML

- El marcado debería ser elegante
 - El marcado no debería usarse solamente como una herramienta para lograr que los navegadores muestren una página como el autor lo desee. El marcado se debe entender, debería ser amigable y placentero.
- El marcado no debería repetirse
 - HTML implica grandes repeticiones. La mayoría de los elementos deben ser nombrados dos veces: Una vez antes de su contenido y otra después.
 - ERB agrega aun más repetición y caracteres innecesarios. Haml evita todo esto recurriendo al sangrado en vez del texto para determinar dónde empiezan y terminan los elementos y los bloques de código. Esto no solo da como resultado plantillas más compactas sino que además hace el código mucho más limpio a la vista.
- El marcado debería tener buen sangrado
 - Uno de los mayores problemas de los lenguajes de plantillas tradicionales es que no solo no fomentan el código bien sangrado sino que encima lo hacen difícil o incluso imposible de escribir. El resultado es un *XHTML* confuso e ilegible. Haml formatea las etiquetas de manera que todas estén bien sangradas y reflejen la estructura subyacente del documento.
- La estructura de HTML debería ser clara
 - XML y *XHTML* son formatos fundados sobre la idea de un documento estructurado. Debido a que la lógica de Haml se basa en el sangrado de elementos hijos, esta estructura se preserva naturalmente, lo que hace al documento mucho más fácil y lógico de leer.

En la siguiente figura 2.1.3 se observa la estructura básica de un documento en HTML5 y su equivalente en HAML.

escapeinside

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <meta charset="utf-8" />
5     <title>
6       Estructura basica de HTML5
7     </title>
8   </head>
9   <body>
10    <p>Hola mundo</p>
11    <footer>
12    </footer>
13  </body>
14 </html>
```

escapeinside

```
1 !!!
2 %html{:lang => "es"}
3   %head
4     %meta{:charset => "utf-8"}
5     %title Estructura basica de HAML
6   %body
7     %p Hola mundo
8     %footer
```

Figura 2.1.3: Estructura básica en HTML5 y el equivalente en HAML

La implementación original de HAML está diseñada para trabajar con código ruby embebido. Usando HAML se logran plantillas con ruby embebido y sobre todo legibles al tiempo que mantiene la flexibilidad de las plantillas ERB tradicionales.

ERB (Embedded Ruby) es el lenguaje que utiliza Ruby por defecto como plantillas para generar archivos HTML con código Ruby embebido. La filosofía que usa es escribir en HTML y para especificar código Ruby utiliza `<%%>` y `<%= %>`, mientras en HAML solo debe colocarse un `(-)` y para ingresar valores de variables Ruby solo debe colocarse `(=)`. A continuación, en la figura 2.1.4 contiene un ejemplo de código ERB y su equivalente en HAML.

	escapeinside
1	<div id="contenido">
2	<div class="izquierda columna">
3	Este es un ejemplo en HAML!
4	<p>
5	<%= Este es un parrafo %>
6	</p>
7	</div>
8	<div class="derecha columna">
9	<%= render :partial => "barra" %>
10	</div>
11	</div>

	escapeinside
1	#contenido
2	.izquierda.columna
3	%h2 Este es un ejemplo en HAML!
4	%p= Este es un parrafo
5	.derecha.columna
6	= render :partial => "barra"

Figura 2.1.4: Ejemplo de código en ERB y su equivalente HAML

2.1.3. JavaScript

JavaScript es un lenguaje orientado a objetos y su sintaxis básica es intencionalmente parecida a la de Java y C++ para facilitar el aprendizaje del lenguaje. Se pueden incluir los scripts en la cabecera del documento HTML, dentro de los elementos pertenecientes al “*body*” y definiéndolos en un archivo externo y realizando una referencia desde el documento HTML. Es necesario hacer referencia al hecho de que JavaScript es un lenguaje de corto alcance, es decir, sus acciones están limitadas al sitio web. Entre las funcionalidades similares con los lenguajes mencionados, posee los operadores ya conocidos y comúnmente utilizados (igualdad, relacionales, sumas, etc.) y su aplicación es de la misma forma a como se conoce.

Posee además las funcionalidades de desplegar cuadros de diálogos emergentes para informar o alertar de información relevante, que sea necesario la confirmación del usuario para continuar con su trabajo dentro del ambiente web. Un analizador de expresiones regulares complementa las funcionalidades del lenguaje y proporciona para los desarrolladores, una herramienta poderosa y de gran utilidad para la realización de validaciones. El acceso al DOM del HTML es un punto importante puesto que, a través de esto, es posible realizar modificaciones y aplicar dinamismo a la página. “*getElementById*” nos permitirá tener acceso a las etiquetas definidas en el documento por medio del “Id” establecido, cada uno de estas etiquetas poseen propiedades que pueden ser modificadas mediante las funciones proporcionadas por el lenguaje, por su parte “*getElementsByTagName*” ofrece las mismas

opciones que la función antes mencionada, pero a diferencia de esta, se hace a un tipo de etiquetas que se encuentre presente en el documento.

2.1.3.1. CoffeeScript

Es un lenguaje de programación que se compila a JavaScript. Su objetivo es reemplazar la gramática detallada de JavaScript con algo más conciso y elegante, sin añadir ningún tipo de gastos o le impide el uso de JQuery o cualquier otro marco de JavaScript. El lenguaje sintáctico es influenciado por Python, Ruby, Haskell, y YAML, CoffeeScript compila uno a uno en el equivalente JavaScript; acaba con puntos y comas innecesarios, aparatos ortopédicos, y declaraciones, y añade características de programación mas sofisticadas y funcionales como la comprensión de lista y la coincidencia de patrones [4].

El lenguaje tiene un seguimiento relativamente grande por parte de la comunidad Ruby. El soporte a CoffeeScript está incluido en Ruby on Rails desde su versión 3.1.

escapeinside

```
1 History.initCore = function() {
2   // Initialise
3   if ( typeof History.initCore.initialized !== 'undefined' ) {
4     // Already Loaded
5     var i=0; for (i=0;i<=5;i++) {
6       document.write("El número es " + i);
7       document.write("<br />");
8     }
9     return false;
10  } else {
11    History.initCore.initialized = true;
12  }
13 }
```

escapeinside

```
1 History.initCore = ->
2   # Initialise
3   if typeof History.initCore.initialized isnt "undefined"
4     # Already Loaded
5     i = 0
6     while i <= 5
7       document.write "El número es " + i
8       document.write "<br />"
9       i++
10    false
11  else
12    History.initCore.initialized = true
```

Figura 2.1.5: Código en JavaScript y su equivalente en CoffeeScript

2.1.4. JQuery

Es una biblioteca de JavaScript, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. JQuery es software libre y de código abierto, creada inicialmente por John Resig.

JQuery UI tienes una amplia gama de temas que puede ser descargados, aunque también se pueden crear nuevos temas personalizados. Un ejemplo de tema se puede visualizar en la figura 2.1.6.

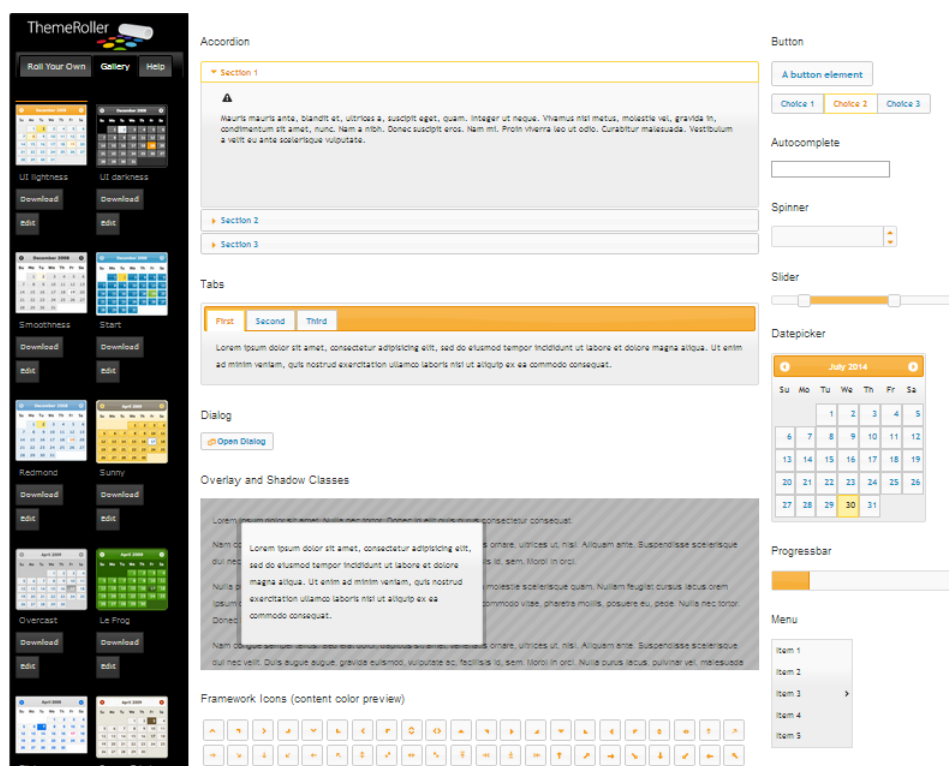


Figura 2.1.6: Tema de JQuery color naranja.

2.1.4.1. AJAX

Acrónimo de Asynchronous JavaScript And XML (JavaScript y XML asíncronos, por sus siglas en inglés) Es una técnica de desarrollo web para crear aplicaciones interactivas que mantienen una comunicación asíncrona con el servidor en segundo plano. AJAX involucra varias tecnologías, de naturaleza asíncronas, unidas con el fin de lograr cambios sobre una misma página sin necesidad de recargarla completamente, mejorando significativamente la interactividad, velocidad y usabilidad en las Aplicaciones Web.

- XHTML y CSS para el diseño que acompaña a la información.
- *Document Object Model* (DOM).

- XML y XSLT: Es el formato usado generalmente para la transferencia de datos solicitados al servidor, aunque cualquier formato puede funcionar.
- *XMLHttpRequest*: El acceso a los datos se realiza mediante *XMLHttpRequest*, objeto disponible en los navegadores actuales, que actúa como una interfaz para realizar peticiones HTTP y HTTPS a servidores web.
- JavaScript: Es el lenguaje interpretado en el que normalmente se efectúan las funciones de llamada de AJAX, cambiando dinámicamente el contenido de las paginas HTML a través de DOM.

En las Aplicaciones Web tradicionales, las acciones realizadas por el usuario en la página desencadena una serie de llamadas al servidor. Una vez procesada la petición, el servidor devuelve un HTML al navegador web y este se encarga de mostrar el contenido recargando la pagina. La figura 2.1.7 muestra a la izquierda la interacción tradicional mientras a la derecha muestra otra con AJAX.

2.1.4.2. Interacción de una aplicación Web AJAX

Una aplicación AJAX elimina el start-stop-start-stop de la interacción Web introduciendo un intermediario *AJAX Engine*.

El AJAX Engine (motor AJAX): Es el responsable de la comunicación con el servidor y de la actualización de la página. Esta interacción es asincrónica, así el usuario no tendrán una página en blanco.

Este tipo de interacción web, elimina la carga completa de la página y provee un mecanismo de cargas incrementales. Aprovechando la capacidad de procesamiento del cliente, lo encarga de más tareas en la ejecución de las aplicaciones. Dando mejoras en las capacidades gráficas para obtener una mayor interactividad.

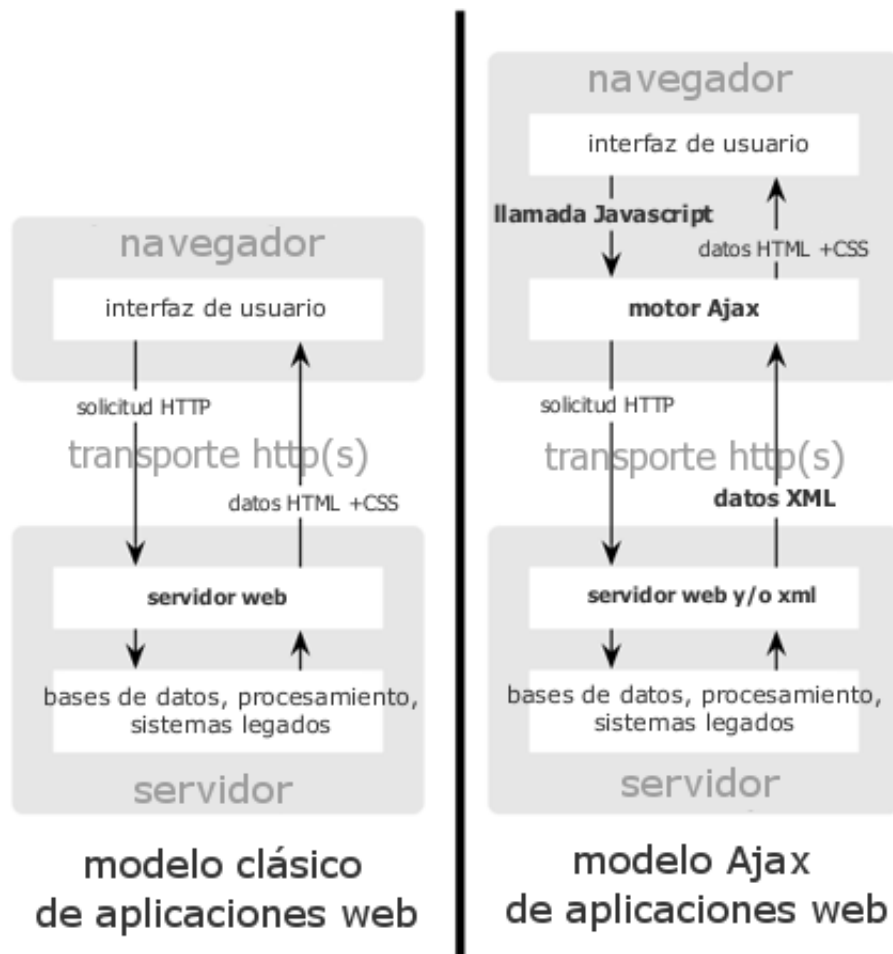


Figura 2.1.7: Aplicación tradicional y Aplicación con Ajax

2.1.5. Ruby on Rails

Rails es un framework de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma de la arquitectura Modelo, Vista y Controlador (MVC). El lenguaje de programación Ruby permite la metaprogramación, de la cual Rails hace uso, lo que resulta en una sintaxis más legible [15].

Entender el patrón MVC es clave para entender Rails. MVC divide la aplicación en tres capas, cada una con una función específica.

- **Vista:** Es la lógica de visualización, o cómo se muestran los datos de las clases del Controlador. Se compone de “templates” (plantillas) que son responsables de proporcionar representaciones apropiadas de los recursos de la aplicación. Las plantillas pueden venir en una variedad de formatos, pero la mayoría de plantillas de vista son HTML con código Ruby embebido (archivos. ERB).
- **Modelo:** Se encarga de definir los objetos que representan las entidades del problema,

las relaciones entre ellos y el procesamiento realizado. En Ruby on Rails, las clases del Modelo son gestionadas por ActiveRecord.

- **Controlador:** Es responsable de tramitar las solicitudes HTTP entrantes y ofrecer una respuesta adecuada. Por lo general, esto significa volver HTML, pero los controladores Rails también puede generar XML, JSON, PDFs, vistas específicas para móviles y mucho más.

En Rails, las capas *Controller* y *View* son manejados conjuntamente por el *Action Pack*. Estas dos capas están agrupados en un solo paquete debido a su interdependencia pesada. Esto es a diferencia de la relación entre *Active Record* y *Action Pack*, que son independientes. Cada uno de estos paquetes se puede utilizar de forma independiente fuera de *Rails*.

2.1.5.1. Rails 4.0

Ruby on Rails fue escrito por David Heinemeier Hansson a partir de su trabajo en Basecamp, una herramienta de gestión de proyectos, por 37signals. Fue liberado al público por primera vez en julio de 2004. Tras casi tres años desde el lanzamiento de la versión anterior (RoR 3.0) el equipo de trabajo de este entorno de desarrollo Web código abierto anuncia en junio de 2013, la versión 4.0. A continuación se hace mención de algunas de las características de la versión [15]:

- *StrongParameters*: Se reemplaza *attr_accessor*, *attr_protected*, y mueve el filtrado de parámetros hacia el controlador, en lugar del modelo.
- *Turbolinks*: Acelera el lado del cliente, que básicamente convierte la aplicación en una aplicación de una sola página javascript en términos de velocidad, pero sin ninguno de los inconvenientes del desarrollo (excepto, tal vez, los problemas de compatibilidad con algunos paquetes de *JavaScript* existentes).
- Mejoras en *Basecamp*, que permite un trabajo mucho más efectivo con JSON.
- *ActiveSupport:Queue*: muy comentado en la *RailsConf*, se añade el soporte para una cola de procesos a Rails.
- *Cache Digests*: Rails 4.0 introduce la generación de claves para la cache basada en un objeto y sus dependencias, gestión automática de las dependencias de las plantillas anidadas se renuevan adecuadamente cuando hay modificaciones.
- Mejoras en *ActiveRecord*
 - Los métodos *DROP_TABLE* y *remove_column* ahora son reversibles, siempre y cuando se da la información necesaria.

- Nuevo método *revert* volverá una migración total o el bloque dado.
 - Agrega *Relation#load*, para cargar explícitamente el registro y volver *self*.
 - *Model.all* ahora devuelve un *ActiveRecord :: Relation*, en lugar de un conjunto de registros.
 - Añade *PostgreSQL* Soporte de tipo array. Cualquier tipo de datos se puede utilizar para crear una columna de matriz, con plena migración y apoyo dumper support.
- *Railties*: Nuevas ubicaciones de prueba *test/models*, *test/helpers*, *test/controllers* y *test/mailers*. Multi-hilo por defecto.
 - Características obsoletas:
 - *Active Resource*
 - *Active Record Observers*
 - *Action Pack Page*
 - *Action Caching*.
 - *ActiveRecord::SessionStore*
 - *Page Caching*
 - *Sprockets*
 - *Performance tests*

2.1.6. MySQL

Es el sistema de gestión de bases de datos relacionales (*RDBMS* por sus siglas en inglés *Relational Database Management System*) desarrolla, distribuida y soporta la compañía comercial MySQL AB. Ha estado disponible desde 1996, pero su desarrollo data desde 1979 y ha ganado 3 años consecutivos el premio Linux Journal Reader's Choice Award [10].

MySQL es un servidor multi-usuarios muy rápido y robusto de ejecución de instrucciones en paralelo, es decir, que múltiples usuarios distribuidos a lo largo de una red local o Internet podran ejecutar distintas tareas sobre las bases de datos localizadas en un mismo servidor. Utiliza el lenguaje SQL (*Structured Query Language*), que fue desarrollado por IBM en 1981 y desde entonces es utilizado de forma generalizada en las bases de datos relacionales.

MySQL es un software de código abierto, significa que es posible para cualquiera usar y modificar el software, bajo la licencia de GNU GPL (*General Public License GNU*), aunque MySQL AB distribuye una versión comercial, en lo único que se diferencia de la versión libre, es el soporte técnico que ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de otra manera, se vulneraría la licencia GPL.

El software de base de datos MySQL es un sistema cliente/servidor que consiste en un servidor SQL multi-threaded que trabaja con diferentes bakends, programas y bibliotecas cliente, herramientas administrativas y un amplio abanico de interfaces de programación para aplicaciones (APIs).

2.1.6.1. Características de MySQL

Algunas de las características de MySQL se exponen a continuación:

- Interioridades y Portabilidad
 - Escrito en C y en C++.
 - Usa *GNU Automake*, *Autoconf*, y *Libtool* para portabilidad.
 - Probado con un amplio rango de compiladores diferentes
 - Usa tablas en disco *B-tree* (*MyISAM*) muy rápidas con compresión de índice.
 - Tablas hash en memoria, que son usadas como tablas temporales.
 - Proporciona sistemas de almacenamiento transaccionales y no transaccionales.
 - Uso completo de *multi-threaded* mediante *threads* del kernel. Pueden usarse fácilmente múltiple *CPUs* si están disponibles
 - Funciona en diferentes plataformas
- Tipos de Columnas
 - Diversos tipos de columnas: enteros con/sin signo de 1, 2, 3, 4, y 8 bytes de longitud, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET, ENUM, y tipos espaciales OpenGIS
 - Registros de longitud fija y longitud variable.
- Sentencias y Funciones
 - Soporte completo para operadores y funciones en las cláusulas de consultas *SELECT* y *WHERE*.
 - Soporte completo para las cláusulas *SQL GROUP BY* y *ORDER BY*. Soporte de funciones de agrupación.
 - Soporte para *LEFT OUTER JOIN* y *RIGHT OUTER JOIN* cumpliendo estándares de sintaxis *SQL* y *ODBC*.
 - *DELETE*, *INSERT*, *REPLACE*, y *UPDATE* devuelven el número de filas que han cambiado (han sido afectadas). Es posible devolver el número de filas que serían afectadas usando un flag al conectar con el servidor.

- Puede mezclar tablas de distintas bases de datos en la misma consulta
- Seguridad
 - Un sistema de privilegios y contraseñas que es muy flexible y seguro, que permite verificación basada en el host.
- Escalabilidad y Límites
 - Se permiten hasta 64 índices por tabla (32 antes de *MySQL* 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de *MySQL* 4.1.2).
 - Soporte a grandes bases de datos. Usamos *MySQL Server* con bases de datos que contienen 50 millones de registros.
- Conectividad
 - Los clientes pueden conectar con el servidor *MySQL* usando sockets TCP/IP en cualquier plataforma. En sistemas Windows de la familia NT (NT,2000,XP, o 2003), los clientes pueden usar named pipes para la conexión. En sistemas Unix, los clientes pueden conectar usando ficheros socket Unix
 - Los servidores Windows soportan conexiones con memoria compartida si se inicializan con la opción `–shared-memory`. Los clientes pueden conectar a través de memoria compartida usando la opción `–protocol=memory`.

Las tecnologías a las cuales se hicieron mención en este capítulo, forman parte del diseño y desarrollo del módulo de Planificación Docente. Aunque para lograr el éxito de dicha aplicación, se deben obtener los resultados deseados, normalmente estos son definidos en los requerimientos de la aplicación y son obtenidos a través de la aplicación de un método de desarrollo software.

2.2. Herramientas

Las herramientas son instrumentos u objetos utilizados con el fin de facilitar la realización de cualquier actividad. En este ámbito, se consideran a Zoho Projects y Asana como herramientas que aunque no son propias de la aplicación facilitan el desarrollo y seguimiento del módulo.

2.2.1. Asana

Es una aplicación Web y móvil diseñada para mejorar la comunicación y colaboración en equipo. Fue desarrollada por Dustin Moskovitz y Justin Rosenstein. Es lo suficientemente flexible para el trabajo y la comunicación en equipo. Asana es una aplicación colaborativa para la gestión de tareas. El producto tiene muchas funcionalidades, como espacios de trabajo, proyectos, proyectos personales, tareas, etiquetas, notas, comentarios y un buzón que organiza y actualiza la información en tiempo real. La característica clave de Asana es su simplicidad, proporciona una infraestructura simple que se puede utilizar de varias maneras [2].

2.2.1.1. Visión general de Asana

Al iniciar Asana, se presenta una vista única, simple y básica; se puede observar en la figura 2.2.1. La aplicación consta de tres secciones principales.

Una columna a la izquierda que dará una visión general del área de trabajo con sus proyectos. Al abrir la aplicación, si sólo tiene acceso a un espacio de trabajo, Asana la selecciona por usted. Este panel también es el lugar donde realiza actividades relacionadas con el área de trabajo, tales como: Agregar un integrante al equipo, crear un proyecto, crear un grupo y gestión de etiquetas.

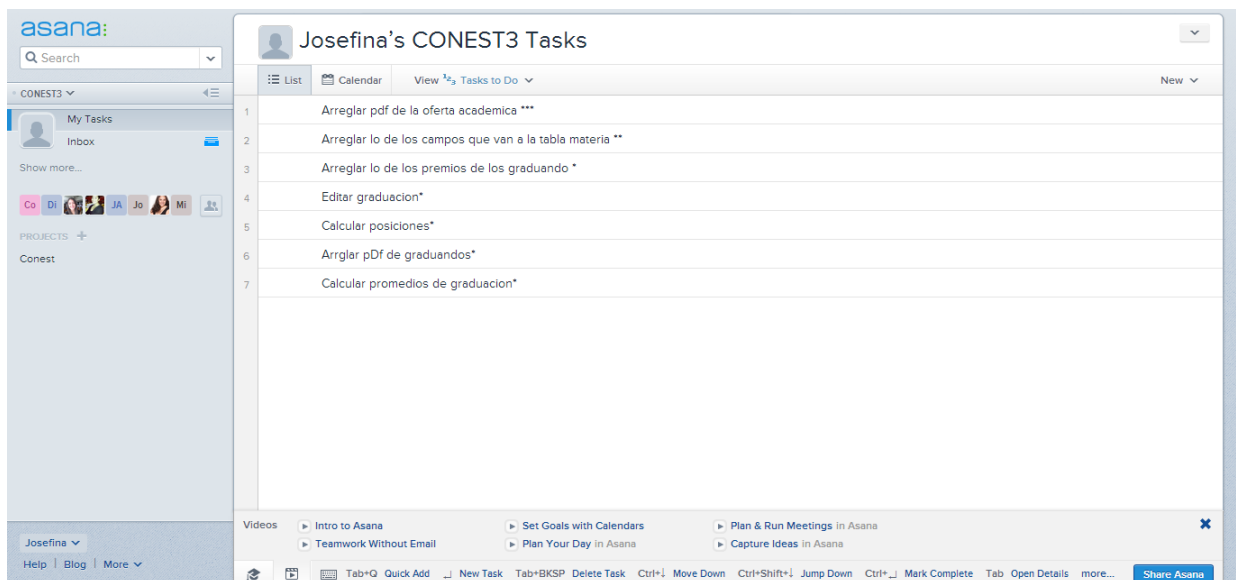


Figura 2.2.1: Interfaz de Asana

La sección central, es el más grande de Asana, es la vista de tareas para el proyecto seleccionado. Este punto de vista es, básicamente, una lista de tareas, donde pueden ser agrupados y etiquetados.

La tercera sección, en el lado derecho, no es tan grande como el del medio y se usa para darle detalles adicionales acerca del elemento seleccionado en el panel central. Por lo tanto, si selecciona una tarea, se mostrará quien se asignó la tarea, comentarios acerca de esta tarea, y así sucesivamente. Asana es muy interactiva: si un miembro del equipo añade un comentario, otros son notificados de forma instantánea.

El elemento más importante en Asana es la "tarea". Para cada tarea, puede configurar varias propiedades: Comentarios, Responsable, Fecha de vencimiento, Adjuntos, Etiquetas, Notificación Estado (hecho / no hecho) Para una tarea, también puede agregar subtareas: Si una tarea, por ejemplo, se compone de una serie de tareas, puede explicar exactamente lo que necesitan para ser completada en subtareas [2].

2.2.2. Zoho Projects

Zoho Projects es un software de gestión de proyectos en línea, colabora con los miembros de su equipo y revoluciona la forma de trabajar. También cuenta con módulos de control de errores [21]. Véase un ejemplo de la vista general de Zoho Projects en la figura 2.2.2.

2.2.2.1. Características

- Planificación de proyectos

Permite pasar menos tiempo en la planificación de un proyecto. Permite establecer tareas, tratando de disminuir las dependencias para completar el proyecto a tiempo. Asignar tarea a los miembros del equipo en base a las prioridades y manteniendo el control sobre el proyecto y su avance.

- Módulo de seguimiento de fallas o error

Informe y seguimiento de los problemas con el sistema de seguimiento de errores. Con esto, tratar de evitar estados críticos.

- Módulo de seguimiento de tiempo

Identificar cómo productivamente el tiempo se ha distribuido a través de una tarea. Realizar el seguimiento del tiempo dedicado al uso del módulo de seguimiento del tiempo y llevar un registro de las horas facturables.

- Informe del proyecto

Puede generar informes y analizar el rendimiento de un equipo con un conjunto de datos gráficos de fácil comprensión. Conseguir informes sobre el estado del proyecto y los recursos utilizados. También se puede determinar la tendencia en curso utilizando diagramas de Gantt.

■ Colaboración ONLINE

Para la toma de decisiones más rápida y mejor, se puede utilizar el incorporado de las salas de chat en línea y wikis para colaborar con colegas y clientes. Calendario de Reuniones en el calendario del proyecto y las notificaciones establecidas para recordar a los asistentes a la reunión.

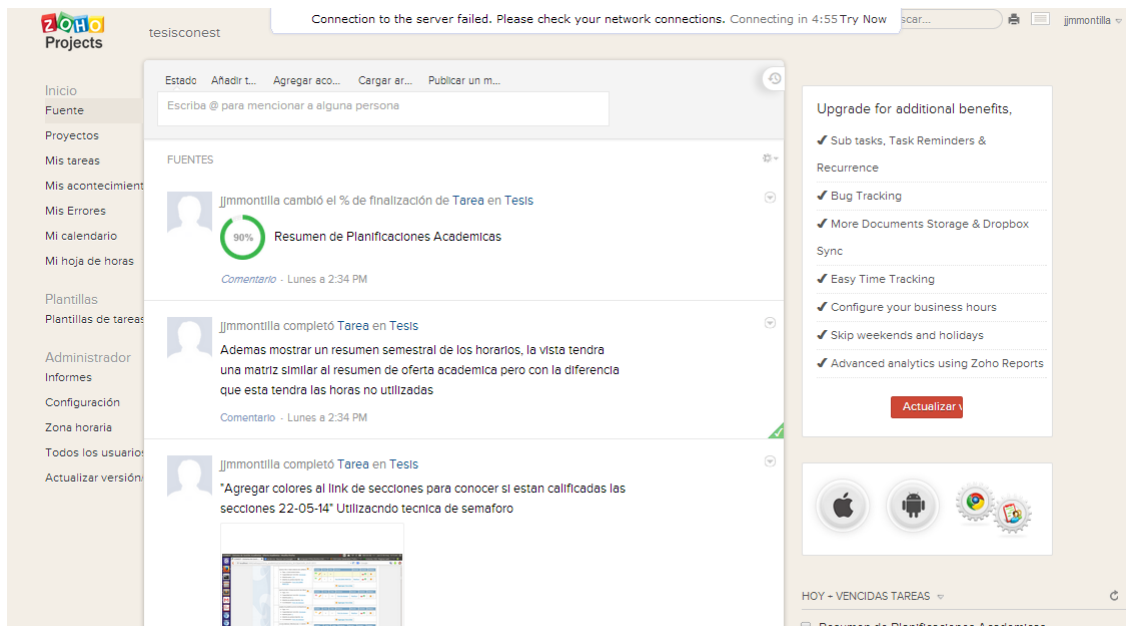


Figura 2.2.2: Vista general de Zoho Projects

Después de describir las tecnologías de mayor relevancia para la plataforma de CO-NEST 3.0. Así como también, se puntualizaron las herramientas de apoyo para el desarrollo del mismo. Se procede a detallar el método AgilUs y el desarrollo del software.

Capítulo 3

Método AgilUs y Desarrollo de Software

Los métodos ágiles de desarrollo de software han despertado gran interés en los últimos años debido a que proponen simplicidad y velocidad para crear sistemas.

Los procesos ágiles de desarrollo de software, también conocidos como metodologías livianas, surgen como alternativa para superar las debilidades notorias y reales en la ingeniería del software convencional, caracterizados por ser rígidos y dirigidos a la documentación exhaustiva que se generan en cada una de las actividades desarrolladas [13].

Los procesos ágiles están centrados en fortalecer las relaciones intrapersonales y los resultados finales, siendo esto su principal filosofía. Dando mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas. Este enfoque está mostrando su efectividad en proyectos con requisitos muy cambiantes y cuando se exige reducir drásticamente los tiempos de desarrollo pero manteniendo una alta calidad del producto.

A continuación, se describe el método de desarrollo de software AgilUs, dando énfasis en puntos tales como: Concepto, características, principios, buenas prácticas y ciclo de vida. Método en el cual se fundamenta el desarrollo de esta investigación. Particularmente, este método destaca entre otros por estar orientado a aplicaciones que involucran un alto grado de interacción con los usuarios y por la integración del concepto de usabilidad desde las primeras etapas del ciclo de vida del software. Bajo esta perspectiva es ideal ya que el módulo de Planificación Docente está dirigido a diferentes usuarios como Docentes, Estudiantes y Personal Administrativo.

3.1. Método AgilUs

Es un método de desarrollo ágil, resultado de una de las líneas de investigación desarrolladas en el Centro de Ingeniería de Software y Sistemas (ISYS) de la Escuela de Compu-

tación, Universidad Central de Venezuela. Incorpora el concepto de usabilidad, en la necesidad de desarrollar software usables. Se fundamenta en el análisis centrado en el usuario y en la participación de especialistas, con el objetivo de evolucionar el software, a fin de que éste alcance el mayor grado de usabilidad. AgilUs es un método de desarrollo iterativo e incremental, se centra en la construcción y desarrollo de las interfaces de usuario, estas deben guiar las decisiones en Ingeniería de Software.

El Método AgilUs busca proporcionar un conjunto de actividades organizadas para construir la usabilidad en el diseño de interfaces de usuario durante el desarrollo de un producto de software. El proceso de desarrollo de software engloba las actividades de requisitos, análisis, prototipaje y entrega; así como las evaluaciones de usabilidad correspondientes a cada etapa del proceso. Se realizan en ciclos iterativos hasta alcanzar el producto final. En cada etapa del proceso de desarrollo de software, se incluyen actividades propias para la construcción de la usabilidad.

AgilUs permite la construcción de la usabilidad desde las primeras etapas del ciclo de vida del software. Está orientado a desarrollos de aplicaciones que involucren un alto grado de interacción con los usuarios. El método propuesto hace énfasis en la obtención de ese feedback desde etapas tempranas del desarrollo, con la aplicación de tormentas de ideas, encuestas y prototipos en papel, entre otras, a lo largo de la evolución del software hasta obtener el producto final, incluido la entrega del producto. Así los resultados de estas evaluaciones permitirán asegurar la obtención de un producto usable.

Alcanzar la usabilidad en el desarrollo de software implica aplicar un Diseño Centrado en el Usuario, a continuación se explica en que consiste este enfoque y se mencionan algunas de las técnicas de interés para el desarrollo este trabajo [1].

3.1.1. Diseño centrado en el usuario

Es un conjunto de métodos y técnicas que tiene como paradigma central la inclusión del usuario como centro de desarrollo. A fin de realizar un diseño centrado en el usuario, se debe tomar en cuenta las características del usuario, las actividades que realiza y el escenario donde desempeña su actividad. Todos estos factores permitirán conocer cuáles son los requisitos y metas que se deben satisfacer en el sistema y cómo es la forma más apropiada de ofrecérselos a los usuarios para facilitar el uso del mismo.

3.1.1.1. Indagación

Técnicas de este tipo relevantes a este trabajo de investigación son: Tormentas de ideas, entrevistas, encuestas, cuestionarios y análisis de sistemas existentes.

- Tormentas de ideas: Esta técnica se debería aplicar en las etapas tempranas del proceso de desarrollo, cuando se conoce poco acerca del diseño real y hay necesidad de nuevas ideas, a fin de determinar los requerimientos de usabilidad del sistema a desarrollar.
- Encuestas, Cuestionarios: Esta técnica puede ser utilizada en cualquier etapa del proceso de desarrollo, dependiendo de las preguntas formuladas en la encuesta y de la información que se requiera por parte de los usuarios.
- Evaluación de sistemas existentes: Consiste en la revisión de versiones anteriores del mismo sistema, así como sistemas de la competencia o afines, con el objetivo de identificar problemas de usabilidad.

3.1.1.2. Inspección

Consiste de un tipo de evaluación que puede implicar observación, medición y comparación con respecto a ciertas características. Los métodos de inspección permiten determinar la madurez conceptual del producto que se está construyendo. Dentro de las técnicas de este tipo relevantes a este trabajo de investigación se encuentran: Evaluación heurística, listas de comprobación y guías de estilo. Dentro de las técnicas de este tipo relevantes a este trabajo de investigación se encuentran:

- Evaluación Heurística: es una técnica de inspección de usabilidad en la que los especialistas buscan problemas de usabilidad en la interfaz de usuario. Normalmente la realiza un grupo reducido de evaluadores que, con base en su propia experiencia y fundamentándose en principios de usabilidad (heurísticas), evalúan de forma independiente la interfaz de usuario, contrastando finalmente los resultados con el resto de evaluadores. Se recomienda que los evaluadores naveguen a través de toda la aplicación al menos dos veces para familiarizarse con su estructura y antes de comenzar con la evaluación propiamente dicha. La evaluación heurística puede ser utilizada en, prácticamente, cualquier momento del ciclo de desarrollo. Se puede proporcionar maquetas de papel o incluso especificaciones de diseño a los expertos y detectar una buena cantidad de problemas de usabilidad antes de que el trabajo real de producción de comienzo.
- Listas de comprobación: son recomendaciones que el equipo de desarrollo ha acordado considerar en el diseño de la interfaz de usuario. Hay que comenzar decidiendo qué tipo de listas se van a utilizar para juzgar los atributos y los métodos de interacción de la interfaz de usuario.
- Guías de estilo: Las guías de estilo para el diseño de interfaces de usuario resumen buenas prácticas y provee guías de bajo y alto nivel en el diseño de interfaces usables.

La adopción de guías de estilo específicas se puede establecer como parte de los requerimientos no funcionales o requerimientos de usabilidad. Como resultado de aplicar se obtiene una interfaz de usuario mejor diseñada.

3.1.1.3. Pruebas

Las pruebas de usabilidad son una forma de medir qué tan bien puede una persona usar un objeto hecho por el hombre, en particular un software. En general, una prueba consiste en hacer uso de un producto para ver cómo funciona o qué resultado tiene. Dentro de las técnicas de este tipo relevantes en esta investigación se pueden citar: Protocolo del Pensamiento Manifestado, Protocolo de preguntas y las pruebas de aceptación.

- Protocolo del Pensamiento Manifestado: El protocolo del pensamiento manifestado es una técnica de evaluación que consiste en observar a los usuarios mientras están realizando una tarea con el sistema, y a quien se le ha solicitado que exprese en voz alta sus pensamientos, sensaciones y opiniones mientras interactúa con el producto. Esta técnica se puede utilizar en cualquier fase del proyecto de desarrollo, no es costosa y es una buena fuente de información cualitativa.
- Pruebas de aceptación: Las pruebas de aceptación conducidas por el cliente verifican que el sistema satisface los requerimientos funcionales y no funcionales. Estas pruebas las realiza el cliente. Estas pruebas se realizan sobre el producto terminado e integrado o pudiera ser una versión del producto pactada previamente con el cliente.

Una prueba de aceptación puede ir desde un informal caso de prueba hasta la ejecución sistemática de una serie de pruebas bien planificadas. De hecho, las pruebas de aceptación pueden tener lugar a lo largo de semanas o meses, descubriendo así errores latentes o escondidos que pueden ir degradando el funcionamiento del sistema. Estas pruebas son muy importantes, ya que definen el paso nuevas fases del proyecto como el despliegue y mantenimiento.

La prueba alfa se lleva a cabo, por un cliente, en el lugar de desarrollo. Se usa el software de forma natural con el desarrollador como observador del usuario. Las pruebas alfa se llevan a cabo en un entorno controlado. Una vez logrado esto, se procede a realizar las pruebas y a documentar los resultados.

La prueba beta se lleva a cabo por los usuarios finales del software en los lugares de trabajo de los clientes. A diferencia de la prueba alfa, el desarrollador no está presente normalmente. Así, la prueba beta es una aplicación "en vivo" del software en un entorno que no puede ser controlado por el desarrollador. El cliente registra todos los problemas (reales o imaginarios) que encuentra durante la prueba beta e informa a intervalos regulares al desarrollador.

3.1.1.4. Prototipaje

La aplicación de la técnica de prototipaje consiste en la construcción de un modelo del sistema que se desarrolla. Esta técnica es fundamental en el desarrollo e implementación de los métodos descritos para la inspección y pruebas de un producto, debido a que, habitualmente, no se aplican al producto final, sino un prototipo del mismo con unas determinadas características.

- Prototipaje en papel: Esta técnica se caracteriza por el uso de materiales y equipo sencillos para crear una simulación basada en papel de la interfaz de un sistema con el objetivo de explorar los requerimientos de usuario.
- Prototipaje rápido: Esta técnica esta asociada a la idea de desarrollar diferentes conceptos propuestos mediante prototipos de software o hardware, para su posterior evolución [1].

3.1.2. Principios

El desarrollo de un producto de software en AgilUs esta centrado en los siguientes principios:

- Integra la Interacción Humano Computador (IHC) y la Ingeniería de Software (IS). IS e IHC son complementarias, no son disciplinas excluyentes. Un diseño centrado en el usuario impacta positivamente en la calidad del software (*ISO 9126-1*).
- La usabilidad debe considerarse desde el principio del desarrollo, es conveniente incluir la usabilidad desde el principio en el desarrollo como uno de los requisitos para impactar positivamente la calidad del producto final.
- La usabilidad determina la utilidad. Un software se considera útil en la medida que pueda ser usado a fin de producir resultados, en forma eficiente, intuitiva y satisfactoria para los usuarios.
- El usuario determina la usabilidad. La usabilidad no es una propiedad abstracta. Un software sólo será considerado usable en un contexto específico y por un tipo de usuario específico. El objetivo es lograr que todos los usuarios del software encuentren usables las tareas que pueden realizar [1].

3.1.3. Buenas prácticas

En AgilUs se propone aplicar buenas prácticas de desarrollo de software que están enfocadas en satisfacer las demandas del usuario y el desarrollo iterativo e incremental, haciendo énfasis en lograr la usabilidad en cada etapa. Las prácticas son:

- Diseño centrado en el usuario
- Diseño basado en Prototipos
- Desarrollo ágil, iterativo e incremental
- Usabilidad como atributo de calidad
- Interacción continua con el usuario[1]

3.1.4. Ciclo de vida

En cada etapa del desarrollo se incluyen actividades para la construcción de la usabilidad. El ciclo de vida de AgilUS engloba la definición de requisitos, análisis, prototipaje y entrega. En la figura 3.1.1 se muestra un diagrama de la relación entre cada una de las etapas del ciclo de vida de AgilUs, con las actividades que se realizan y artefactos que se generan en cada etapa [1].

A continuación se describen las etapas de este método:

- Requisitos: Se realiza el análisis global del problema a solucionar, se estudian productos similares existentes, se genera un perfil de usuario, y se define la lista de requerimientos a desarrollar. Esta etapa es importante en el desarrollo del software, ya que un mal análisis de requisitos traería como consecuencia un software que no cumple con las necesidades del usuario.
- Análisis: Se lleva a cabo el análisis de la solución a desarrollar, se emplean diagramas de casos de uso y modelo de objetos del dominio, siguiendo la notación UML, para definir las funcionalidades que tendrá el producto a desarrollar.
- Prototipaje: Se implementa un prototipo rápido de la interfaz de usuario a partir de los patrones de interacción, el cual va evolucionando hasta convertirse en el producto final, se genera la guía de estilo, y se realizan evaluaciones de usabilidad apropiadas a esta etapa: las evaluaciones heurísticas y las listas de comprobación.
- Entrega: Se aplican las pruebas al sistema para certificar que la aplicación desarrollada sea un software usable y sin errores, finalmente se pone en producción la aplicación.

Se inscribe en la categoría de métodos ágiles debido a que reduce la cantidad de actividades y artefactos que se generan, motiva la participación del usuario y su carácter iterativo e incremental permite adaptar el desarrollo a los cambios. Incluye la usabilidad a fin de aplicar un enfoque de diseño centrado en el usuario y como un mecanismo para asegurar la calidad del software.

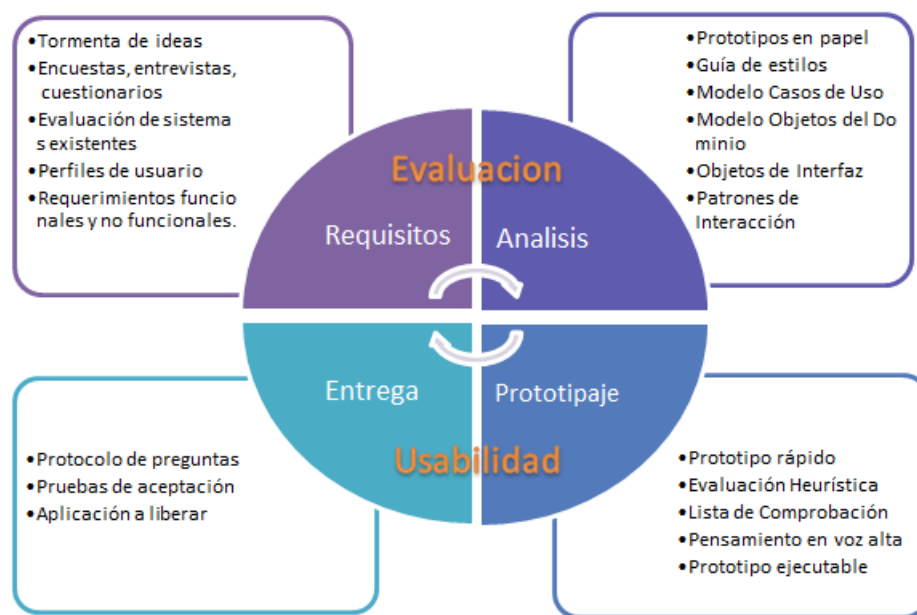


Figura 3.1.1: Ciclo de vida del método AgilUs [1]

Los procesos ágiles están centrados en fortalecer las relaciones intrapersonales y los resultados finales, siendo esto su principal filosofía. Dando mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas. Este enfoque está mostrando su efectividad en proyectos con requisitos muy cambiantes y cuando se exige reducir drásticamente los tiempos de desarrollo pero manteniendo una alta calidad del producto.

Finalmente después de estudiar el método de desarrollo de software AgilUs, es apropiado retomar el objetivo de este Trabajo, basado en desarrollar un módulo automatizado permite ayudar tanto al personal administrativo como docente de Pregrado a realizar las diferentes tareas involucradas en la Planificación Docente.

3.2. Desarrollo del Módulo Planificación Docente

En capítulos previos, se hace mención del método de desarrollo de software aplicado en esta investigación, el cual es AgilUs. En tal sentido, se puntualiza y describe como se llevó a cabo este proceso, resaltando las actividades y artefactos utilizados en las cuatro (4) etapas planteadas en [1]. Las cuales son: Requisitos, análisis, prototipaje y entrega. Adicionalmente, el método permite iterar tantas veces como sea necesario hasta conseguir el producto deseado, es por ello que el ciclo de desarrollo está dividido en cuatro (4) iteraciones: I Planificación Docente, II Gestión de la Planificación Docente, III Reportes y Horarios y IV Estado de la Calificación. Por otra parte, se estableció con el equipo de desarrollo efectuar reuniones

frecuentes de dos (2) a tres (3) semanas, con el fin de establecer los requisitos funcionales del sistema y permitiendo llevar un seguimiento del ritmo de trabajo.

3.2.1. Iteración I Planificación Docente

Al inicio de esta iteración se efectuó una reunión informal con el equipo de desarrollo para obtener la información sobre la Planificación Docente, al igual se discutió en términos generales el funcionamiento y la operatividad del sistema utilizado para ese momento, asimismo, se planteó disponer de cuatro (4) niveles de navegación después del inicio de sesión hasta visualizar o crear una planificación.

Esta iteración está centrada en diseñar la representación visual de la Planificación Docente, con la cual deberá interactuar el usuario después del inicio de sesión hasta la planificación desea, sin embargo, no se efectúa la implementación de las funcionalidades solo su identificación.

3.2.1.1. Requisitos

Se considera pertinente en esta etapa, utilizar técnicas de indagación que permitan conocer las funcionalidades y los requisitos que el sistema debe satisfacer y la manera más adecuada de facilitar a los usuarios el uso de los mismos.

Este ciclo inicia con el análisis general del sistema CONEST para su antigua versión 1.0, seguidamente se realiza el perfil de usuarios, los requerimientos funcionales y no funcionales. A continuación se describen cada una de estas actividades:

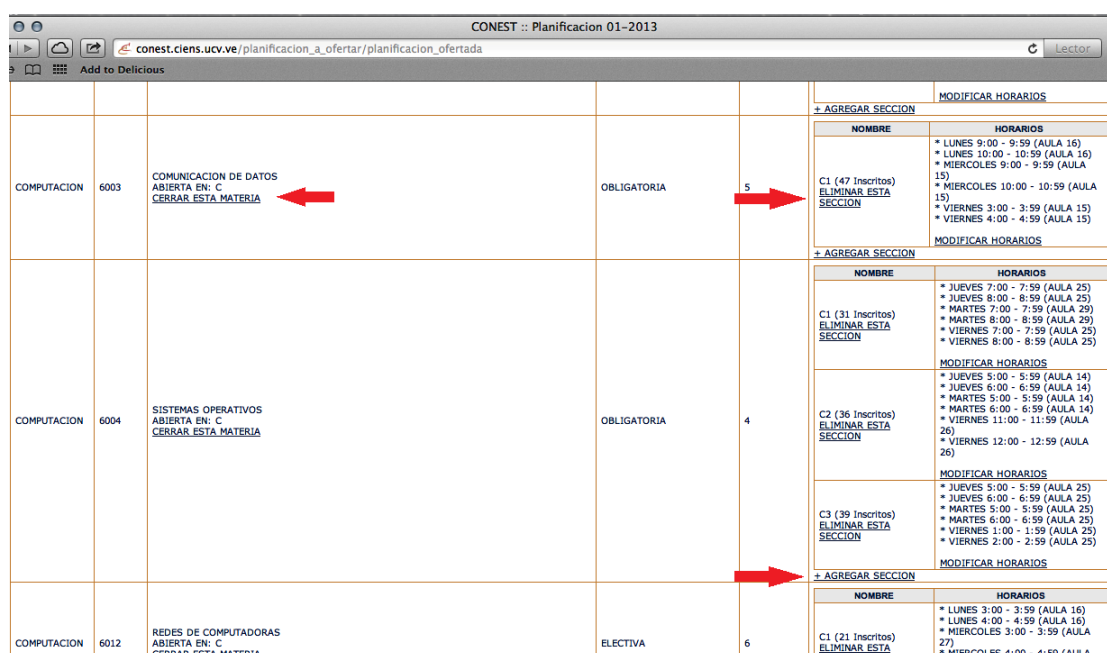
- **Análisis de Sistemas Existentes**

La finalidad de realizar un análisis a otros sistemas es puntualizar sus ventajas y aplicarlas en el desarrollo de la solución propuesta; así como minimizar las dificultades o errores que estos pudieran tener. La aplicación estudiada es el sistema CONEST para su antigua versión, en relación con la Planificación Docente. El siguiente cuadro 3.1 se representa la evaluación:

Cuadro 3.1: Análisis del sistema CONEST para su versión 1.0.

Nombre del sistema	CONEST
Tópicos a evaluar	Este sistema permite al usuario con rol administrativo elegir una planificación existente con las mismas características a la ofertada, con la posibilidad de configurarla y gestionarla, sin embargo, es posible arrastrar con asignaturas no deseadas, además, permite ver un resumen en HTML de la Planificación en la etapa ejecutada. Sin embargo, se observa la falta de organización, agrupación de datos asociados como ociosos, tornando la interfaz poco agradable y confusa. Además, el número de vistas o pasos utilizados por el usuario para realizar una acción tiende a ser excesiva. Observe un ejemplo en la figura 3.2.2
Funcionalidades	Las funciones están orientadas a la gestión de la planificación, tales como: agregar una materia, eliminar una materia, agregar una sección, eliminar una sección, modificar horarios y asignar jurado. En ejemplo de ello, está la figura 3.2.1
Personalización	No permite la personalización, ni filtros de información.
Comunicación	El sistema provee mensajes en la interfaz con información de los cambios realizados por el usuario en la Planificación.
Idiomas	Español
Opiniones de usuario	La interfaz de usuario en confusa, poco agradable y compleja.

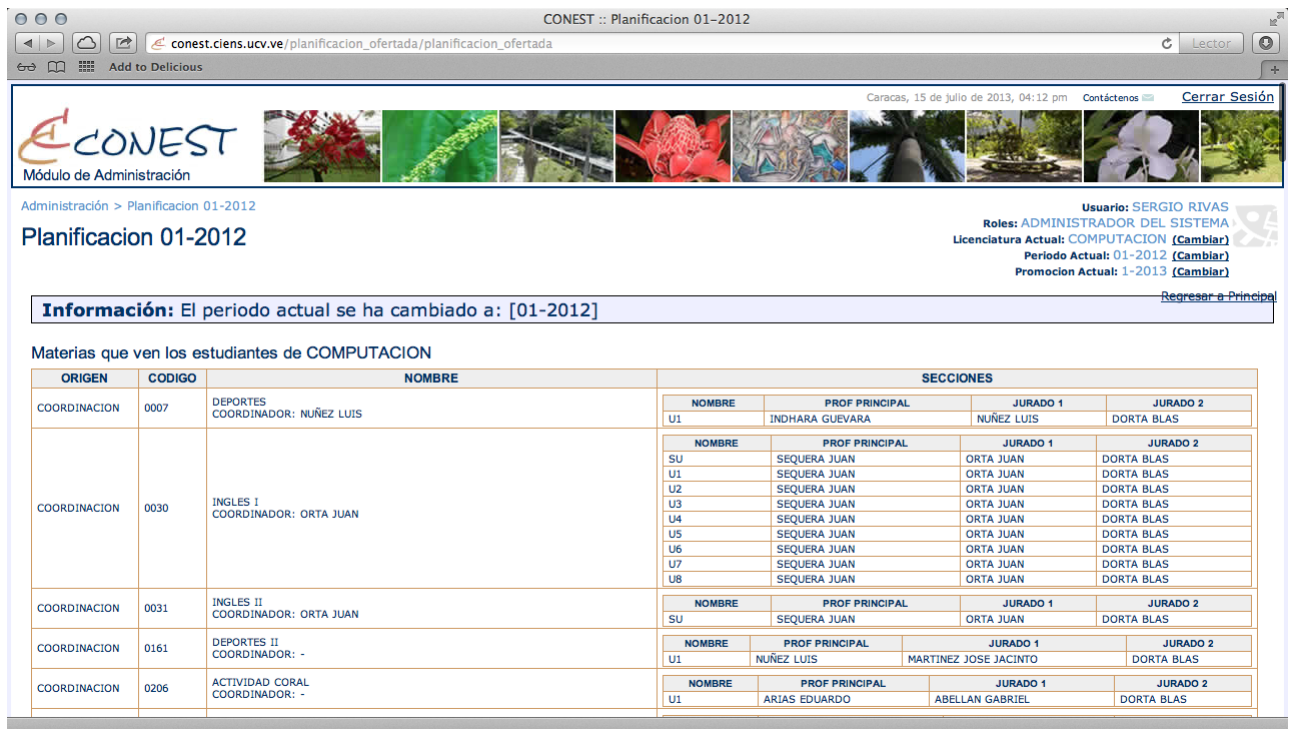
A continuación se muestra la interfaz de usuario de CONEST 1.0 para las opciones de gestión, en este caso de la planificación 01-2013 de etapa ofertada, las flechas rojas no forman parte de la interfaz, indican las acciones.



CONEST :: Planificación 01-2013				
conest.ciens.ucv.ve/planificacion_a_ofertar/planificacion_ofertada				
COMPUTACION	6003	COMUNICACION DE DATOS ABIERTA EN: C CERRAR ESTA MATERIA	OBLIGATORIA	5
COMPUTACION	6004	SISTEMAS OPERATIVOS ABIERTA EN: C CERRAR ESTA MATERIA	OBLIGATORIA	4
COMPUTACION	6012	REDES DE COMPUTADORAS ABIERTA EN: C CERRAR ESTA MATERIA	ELECTIVA	6

Figura 3.2.1: Vista de la Planificación Docente en CONEST 1.0

Otra vista que ejemplifica el análisis del sistema es la siguiente interfaz 3.2.2



CONEST
Módulo de Administración

Caracas, 15 de julio de 2013, 04:12 pm [Contáctenos](#) [Cerrar Sesión](#)

Administración > Planificación 01-2012

Planificación 01-2012

Usuario: SERGIO RIVAS
Roles: ADMINISTRADOR DEL SISTEMA
Licenciatura Actual: COMPUTACION [\(Cambiar\)](#)
Periodo Actual: 01-2012 [\(Cambiar\)](#)
Promoción Actual: 1-2013 [\(Cambiar\)](#)

Información: El periodo actual se ha cambiado a: [01-2012] [Regresar a Principal](#)

Materias que ven los estudiantes de COMPUTACION

ORIGEN	CODIGO	NOMBRE	SECCIONES			
COORDINACION	0007	DEPORTES COORDINADOR: NUÑEZ LUIS	NOMBRE	PROF PRINCIPAL	JURADO 1	JURADO 2
			U1	INDHARA GUEVARA	NUÑEZ LUIS	DORTA BLAS
COORDINACION	0030	INGLES I COORDINADOR: ORTA JUAN	NOMBRE	PROF PRINCIPAL	JURADO 1	JURADO 2
			SU	SEQUERA JUAN	ORTA JUAN	DORTA BLAS
			U1	SEQUERA JUAN	ORTA JUAN	DORTA BLAS
			U2	SEQUERA JUAN	ORTA JUAN	DORTA BLAS
			U3	SEQUERA JUAN	ORTA JUAN	DORTA BLAS
			U4	SEQUERA JUAN	ORTA JUAN	DORTA BLAS
			U5	SEQUERA JUAN	ORTA JUAN	DORTA BLAS
			U6	SEQUERA JUAN	ORTA JUAN	DORTA BLAS
			U7	SEQUERA JUAN	ORTA JUAN	DORTA BLAS
			U8	SEQUERA JUAN	ORTA JUAN	DORTA BLAS
COORDINACION	0031	INGLES II COORDINADOR: ORTA JUAN	NOMBRE	PROF PRINCIPAL	JURADO 1	JURADO 2
			SU	SEQUERA JUAN	ORTA JUAN	DORTA BLAS
COORDINACION	0161	DEPORTES II COORDINADOR: -	NOMBRE	PROF PRINCIPAL	JURADO 1	JURADO 2
			U1	NUÑEZ LUIS	MARTINEZ JOSE JACINTO	DORTA BLAS
COORDINACION	0206	ACTIVIDAD CORAL COORDINADOR: -	NOMBRE	PROF PRINCIPAL	JURADO 1	JURADO 2
			U1	ARIAS EDUARDO	ABELLAN GABRIEL	DORTA BLAS

Figura 3.2.2: Vista de Planificación docente CONEST 1.0

Este análisis facilitó el levantamiento de información y la identificación de los requerimientos de usuarios que son base fundamental en el desarrollo de este trabajo.

■ Perfil de Usuarios

La Planificación Docente no solo está dirigida al personal de la DCE, sino que además contempla a docentes y estudiantes que forman parte de la Facultad de Ciencias. Los últimos son en su mayoría jóvenes que desde temprana edad tienen experiencia con aplicaciones Web. No obstante los docentes son profesionales que se destacan en áreas como: biología, computación, física, matemática o química. Sus edades pueden variar entre 25 y 75 años, en algunos casos pueden no estar familiarizados con las tecnologías. El personal administrativo de la DCE tiene experiencia en la antigua versión del sistema CONEST, esto demuestra la alta pluralidad de usuarios que divisa el sistema.

En efecto de lo antes mencionado, es imprescindible lograr la interfaz sea simple e intuitiva, transmitiendo seguridad y comodidad en la navegación del sistema en general. Considerando que el usuario solo requiere de conocimientos generales para el uso del computador, navegación en Internet y conocer la dinámica de la Planificación Docente.

■ Requerimientos Funcionales y No Funcionales

Los resultados obtenidos en las técnicas de indagación son base indispensable sobre la cual se desprenden el conjunto inicial de requerimientos funcionales y no funcionales

para esta primera iteración.

Entre los requerimientos funcionales de esta iteración se asumen los siguientes:

- Consultar una Planificación Docente, cabe destacar que esta consulta involucra a otras subconsultas como son: Los períodos académicos y datos asociados, Las asignaturas y atributos de las mismas y secciones.
- Otro requerimiento es filtrar la Planificación Docente por escuela y semestre.

Requerimientos No Funcionales:

- Debe tener una interfaz sencilla pero al mismo tiempo intuitiva y con alto grado de usabilidad.

3.2.1.2. Análisis

Como seguimiento a la actividad de desarrollo para definir las soluciones a los requerimientos planteados, se dispone de los siguientes artefactos: modelo de caso de uso, modelo de objetos del dominio, prototipo en papel y guías de estilo. Las cuales se encuentran descritas a continuación:

■ Modelo de Casos de Uso

El Modelo de Casos de Uso describe las funcionalidades del sistema a partir de las interacciones del usuario. Los casos de uso constituye un flujo completo de eventos de interacción entre el actor y el sistema, definiendo una forma particular de usar el sistema. Las diferentes instancias de los casos de uso se denominan escenario. [7]

Bajo esta perspectiva, se diseña un diagrama de casos de uso que identifica las funcionalidades del módulo de Planificación Docente, dirigiendo el proceso de desarrollo del mismo. Se representa bajo la figura 3.2.3, donde se identifican cuatro (4) actores principales, los estudiantes, docentes, jefes de departamentos, que son docentes encargados de la Planificación Docente y al personal de la DCE, quienes acoplan y gestionan la planificación en el sistema. Dentro de las funcionalidades se encuentran:

1. Visualizar la Planificación: Esta función muestra a los usuarios principalmente cada una de las asignaturas, secciones, horarios y los docentes respectivos que pertenecen a una Planificación Docente. Todos los usuarios identificados pueden acceder a esta función, se aplican permisologías para la interfaz de usuario acorde al rol.
2. Gestionar Asignaturas: Hace referencia a la acción y a la consecuencia de administrar asignaturas por medio de opciones como: Agregar una asignatura, consultar el estado o atributos, modificar y cerrar una asignatura.

3. Gestionar Secciones: Permite al usuario manejar las secciones abarcando diferentes actividades, tales como: Abrir una nueva sección, consultar a los estudiantes de la sección, editar y cerrar una sección. Los usuarios que pueden realizar esta acción son el Jefe de Departamento y el personal de DCE.
4. Imprimir Listado de Sección: Tanto docentes como Jefes de departamentos y personal de DCE pueden disponer del listado de estudiantes de una sección en dos formatos PDF y Excel.
5. Administrar Grupo Docente: El grupo docente básicamente cuenta con tres participantes, el docente principal quien sería el responsable de la calificación y dos (2) jurados.
6. Gestionar Horario: Permite realizar diversas opciones entre las que destacan: Consultar horarios por asignaturas, por semestre, asignar el horario que corresponde con alguna asignatura, modificar y eliminar el horario.
7. Imprimir Resumen de Planificación: Esta función permite obtener el resumen de la Planificación Docente en formato PDF.

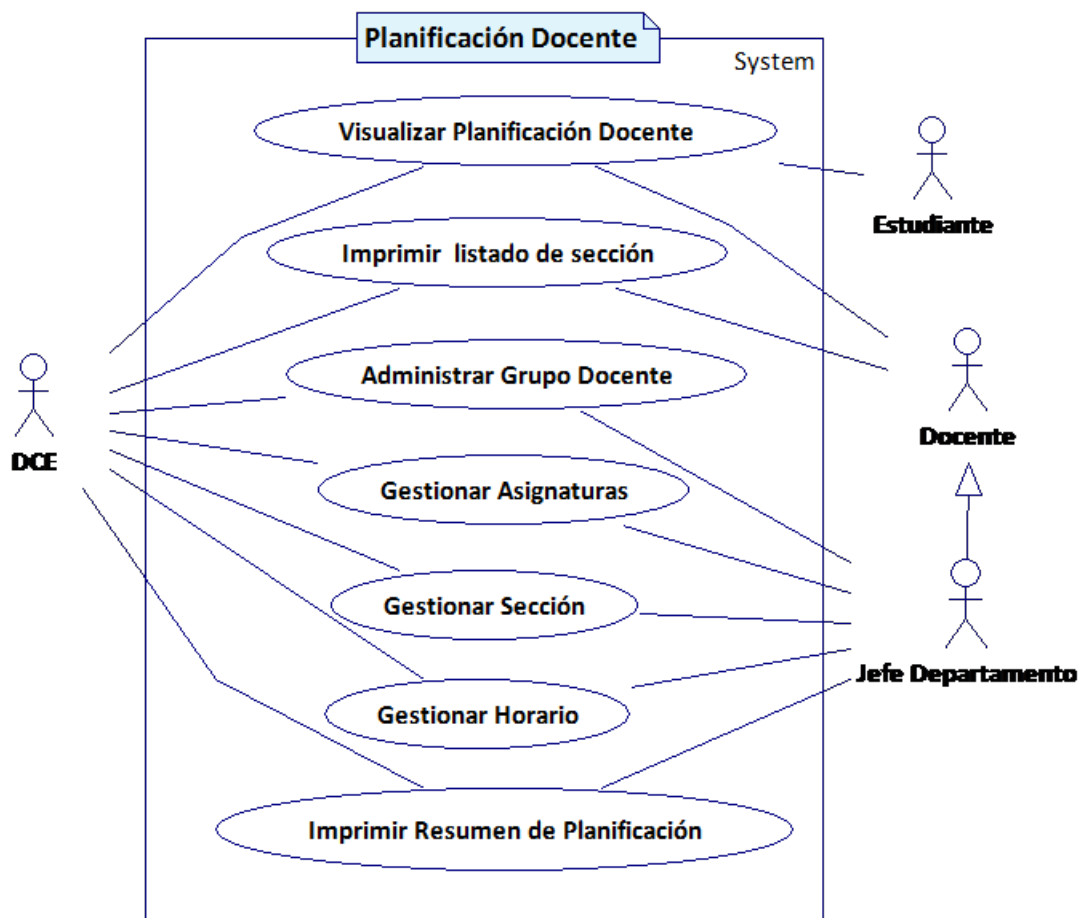


Figura 3.2.3: Caso de Uso de la Planificación Docente, nivel 1

- Modelo de Objetos del Dominio

Partiendo de la estructura del modelo de objetos del dominio a la que hace referencia la figura 3.2.4, se describe los datos más importantes del sistema y sus relaciones estáticas. Como resultado del análisis al modelo de objetos se agregaron nuevos atributos y registros a las tablas existentes en el modelo de datos. El modelo de datos es la estructura abstracta que documenta y organiza la información, se emplea para especificar los datos, las relaciones entre ellos, su semántica asociada y las restricciones de integridad [7].

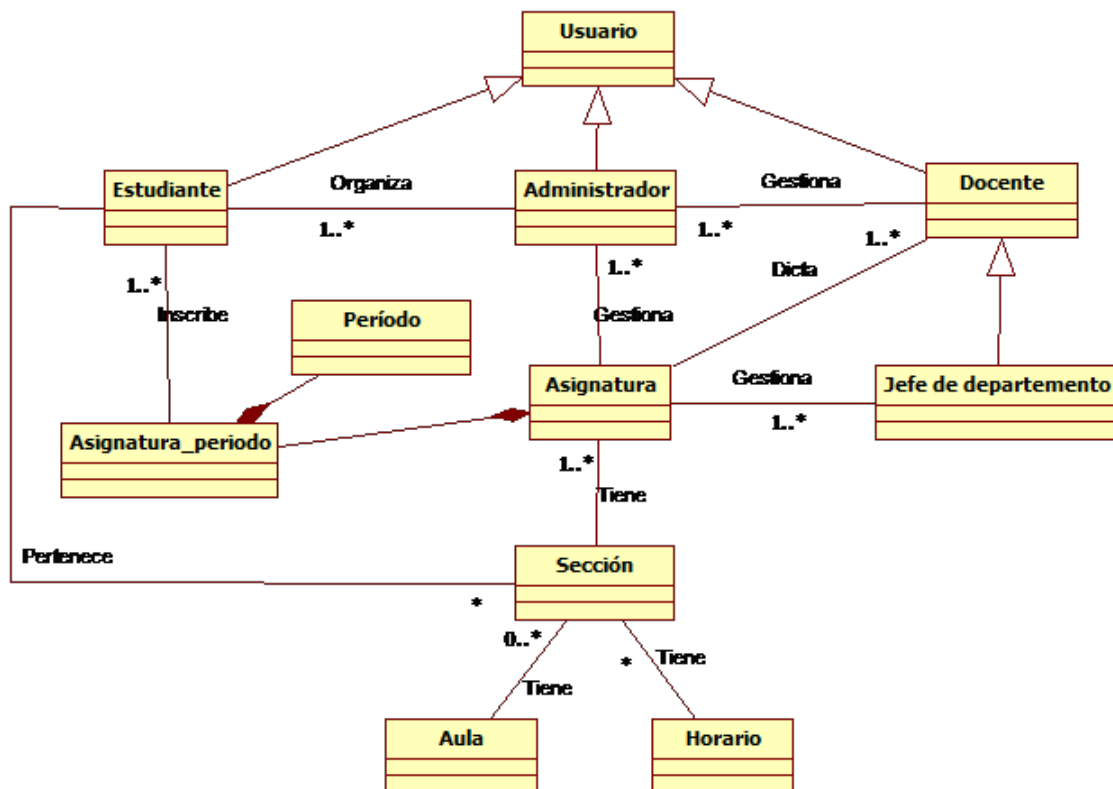


Figura 3.2.4: Modelo de Objetos del Dominio

- Flujo de Información

Se propone un flujo de información para la nueva versión de CONEST, con un rol más participativo por parte de los jefes de departamentos de las escuelas, estos tienen la oportunidad de ajustar las asignaturas electivas al sistema, el objetivo es disminuir el margen de error por el número de personas involucradas en el proceso, en este caso, la DCE solo verifica si la planificación concuerda con el documento asociado a la misma. Sin embargo, aún recae sobre la DCE la asignación de horarios y la manipulación de las asignaturas obligatorias en el sistema. El ejemplo del flujo de información en la figura 3.2.5.

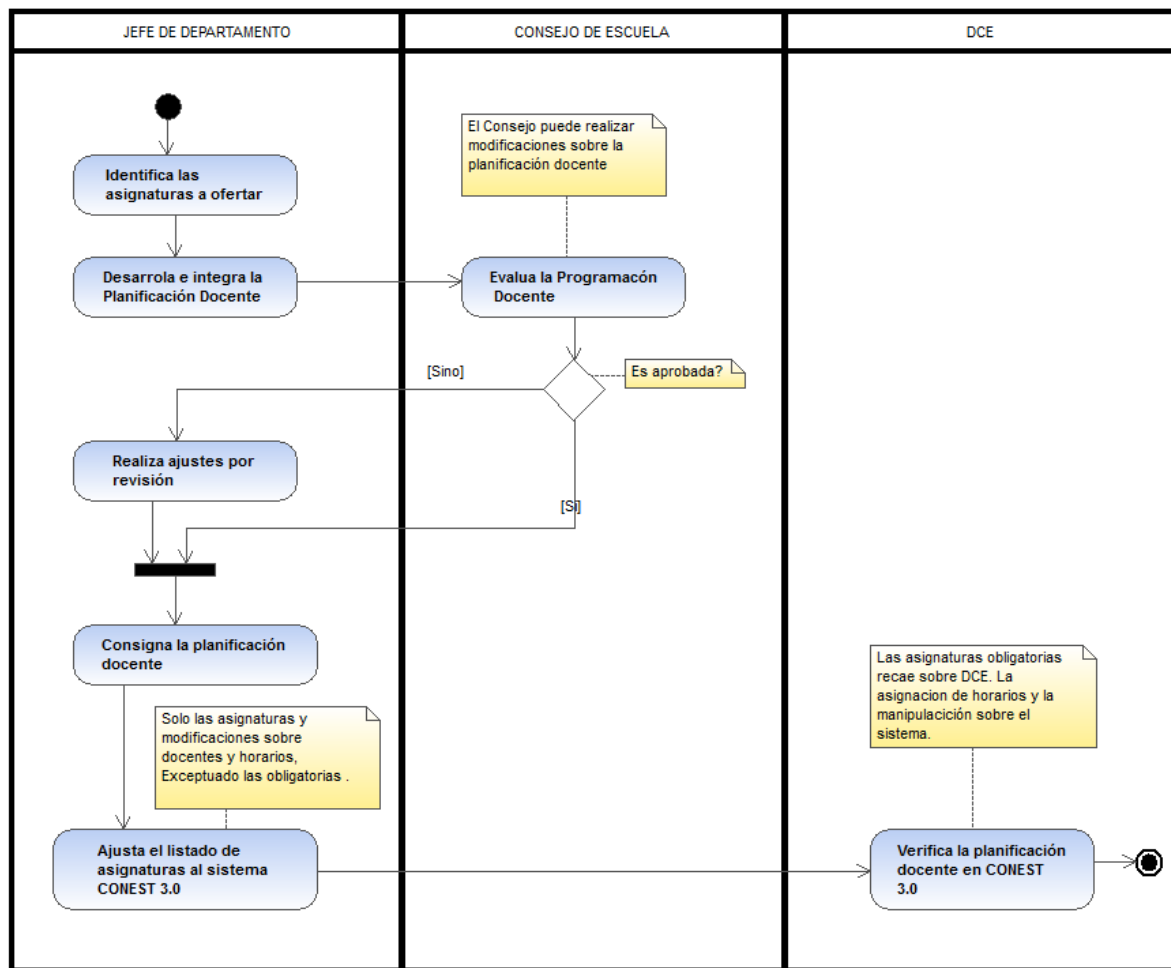


Figura 3.2.5: Flujo de información de la Planificación Docente en CONEST 3.0.

■ Prototipo en Papel

Las figuras 3.2.6 y 3.2.7 representan prototipos en papel, cuyo objeto es simular en lo posible la interfaz con la cual deberá interactuar el usuario, después del inicio de sesión hasta seleccionar la Planificación Docente deseada. En el primer prototipo se visualizan cuatro (4) vistas diferentes: La primera muestra un conjunto de períodos académicos, junto con una ventana modal que representa el agregar de un nuevo período, la segunda vista muestra las licenciaturas y otros datos, la tercera contiene el nombre de las asignaturas y la cuarta indica las secciones de una asignatura.

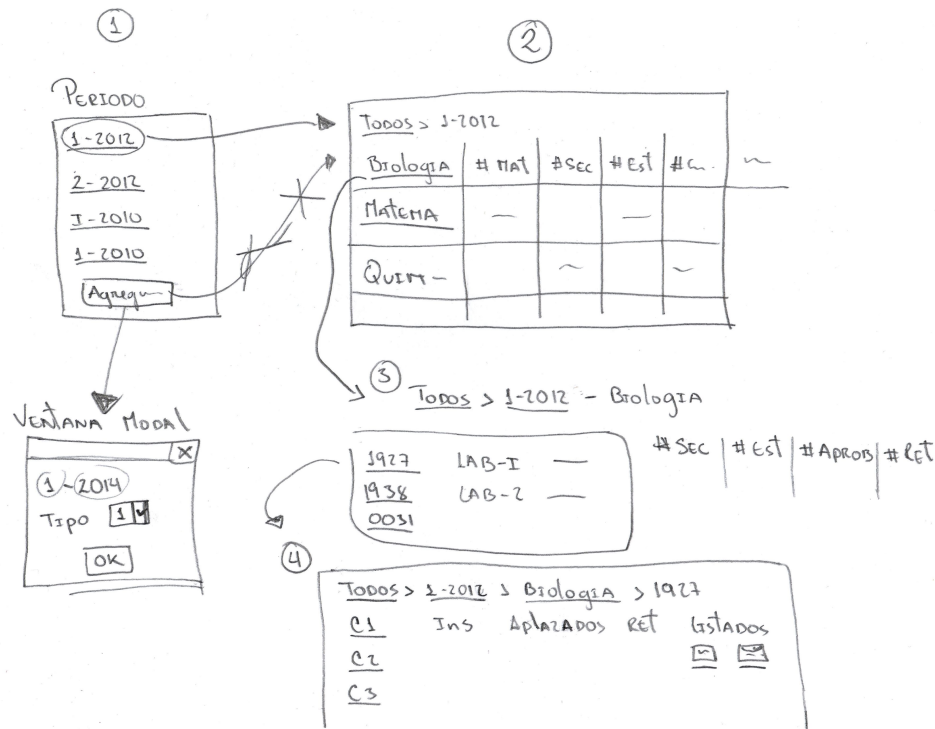


Figura 3.2.6: Prototipo en Papel

El siguiente prototipo, presentado en la figura 3.2.7 es una continuación del anterior y es la quinta vista, esta contiene a los estudiantes que pertenecen a una sección.

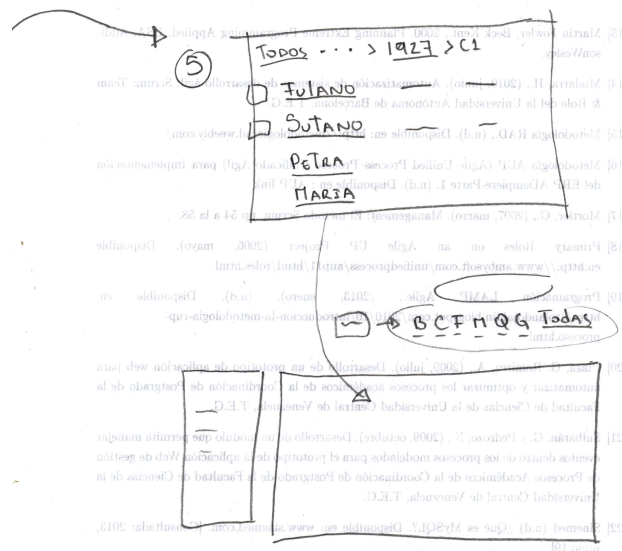


Figura 3.2.7: Prototipo en Papel

Los prototipos de papel permiten crear una imagen palpable de lo que será una futura aplicación. Su creación y manipulación es rápida y permite a los usuarios imaginarse lo que será la futura aplicación en funcionamiento sin interferencias. [14]

■ Guías de Estilo

La guía de estilo constituye un conjunto de buenas practicas y patrones básicos relacionados con la apariencia de una interfaz para su aplicación en el desarrollo de nuevas vistas dentro de un entorno concreto. Las guías de estilo se centran generalmente en el aspecto, resaltando puntos tales como el diseño y la maquetación. [8]

En efecto de lo antes mencionado, se realizó una guía de estilo siguiendo los lineamientos de CONEST 3.0, basada en los siguientes elementos: definición de colores y plantillas.

• Definición de Colores:

Los principales colores de la interfaz de usuarios están basados en el logotipo de la aplicación. La figura 3.2.8 muestra el logotipo de CONEST 3.0.



Figura 3.2.8: Logotipo de CONEST 3.0

La figura 3.2.9 simboliza los colores principales en toda la aplicación y la manera general de aplicarlos es la siguiente:

- El negro y blanco para la tipografía
- El azul para el encabezado de tablas, ventana modales, resaltar el fondo de un título y para representar algún enlace.
- El gris para el borde y pie de tablas.
- El amarillo y el naranja para resaltar mensajes o botones.

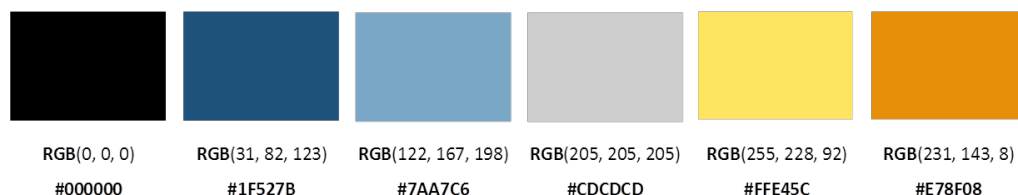


Figura 3.2.9: Colores Principales

• Plantillas:

Para efectos del módulo en desarrollo la plantilla utilizada se visualiza en la figura 3.2.10. El objetivo de este diseño es definir una estructura de página con cabecera y pie, un menú horizontal de navegación, un lateral izquierdo de información y una zona de contenidos.

Rails simplifica el ejercicio de las plantillas estas se realizan a través de los layouts, los cuales son segmentos de códigos que pueden repetirse en cada una de las vistas de la aplicación que lo requiera.

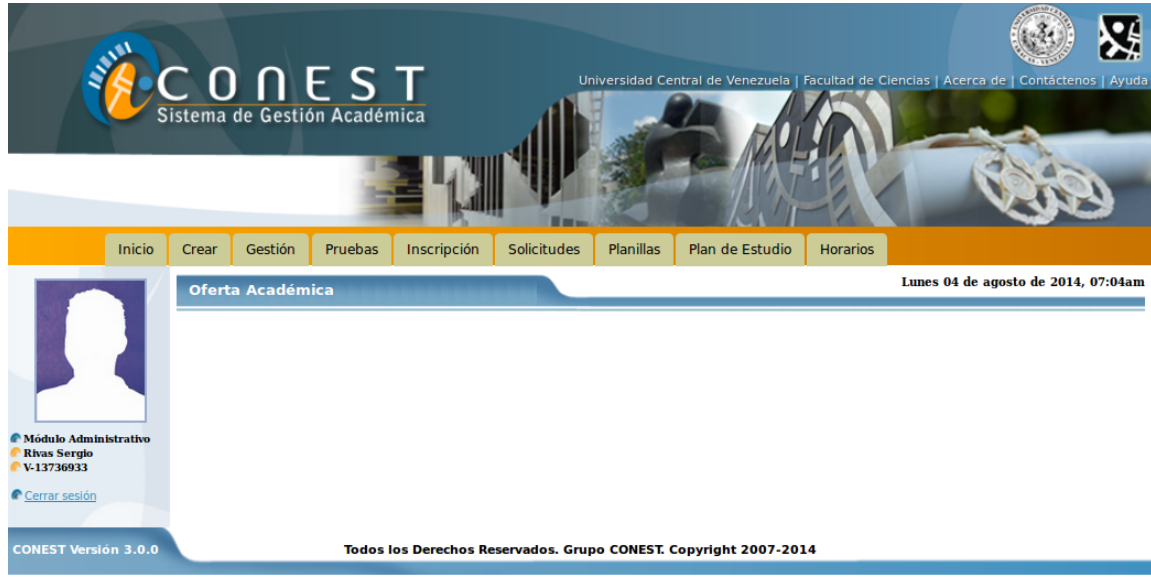


Figura 3.2.10: Plantilla para la Planificación Docente

Esta forma de diseño proporciona una separación entre la estructura y el contenido de aplicación.

3.2.1.3. Prototipaje

Esta etapa define e identifica en líneas generales la apariencia de la interfaz de usuario, la presentación y organización de los datos en los diferentes niveles de interacción con la Planificación Docente.

■ Prototipo Rápido

En avance al módulo de la Planificación Docente se realiza un prototipo rápido en base a la proyección de los datos existentes, sin realizar ninguna funcionalidad de gestión sobre la planificación solo de consulta, dando forma a la semántica, estructura y organización de los mismos en la interfaz de usuario, pero identificando los puntos de partida de cada una de las funcionalidades que debe permitir este módulo.

En este punto se debe disponer de un ambiente de desarrollo para el prototipo rápido bajo las siguientes características: Contar con el Manejador de Versiones de Ruby, la versión 2.0 de Ruby junto con Rails 4.0, conjuntamente las gemas necesarias para el funcionamiento de la aplicación y usar la herramienta de control de versiones Git para mantener la consistencia en el código y las versiones de la aplicación entre los distintos integrantes del grupo de desarrollo.

La primera interfaz, mostrada en la figura 3.2.11 muestra un subconjunto de los períodos académicos más recientes con datos que son significativos para el personal de DCE y representados a través de una tabla que consta de siete (7) columnas con los siguientes campos: Período académico, cantidad de estudiantes por nuevo ingreso, regulares y retirados, por otra parte el número de secciones ofertadas, calificadas y sin calificar.

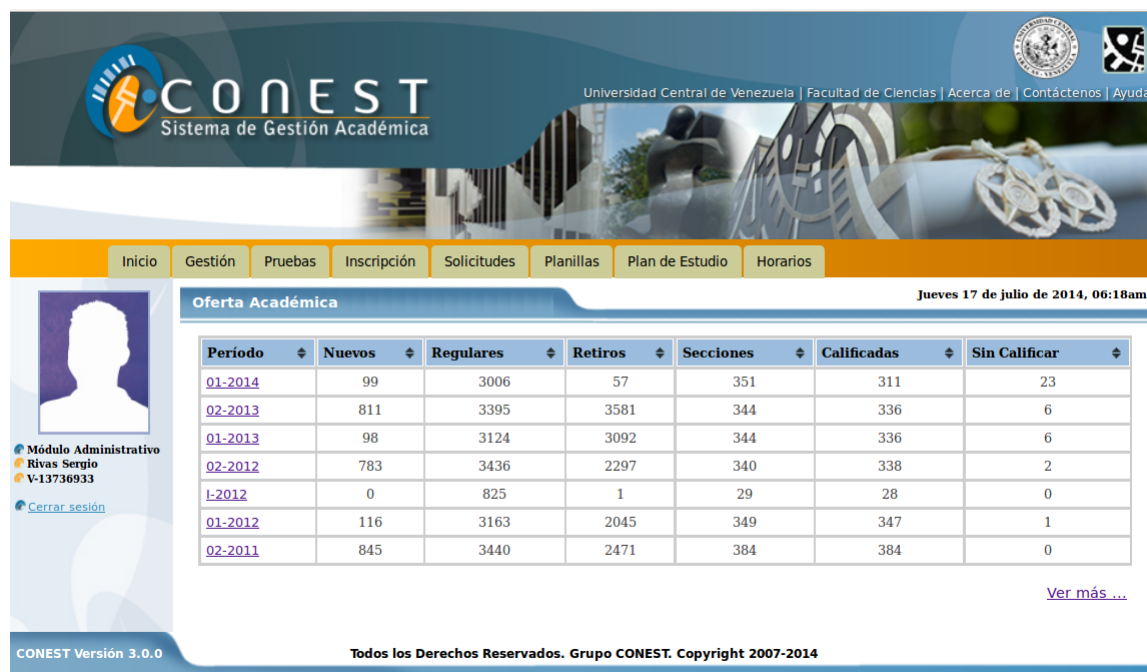


Figura 3.2.11: Prototipo ejecutable de los períodos académicos

Al momento de seleccionar un período académico el contenido de la interfaz cambia por las diferentes licenciaturas de facultad, nuevamente se presentan datos utilizados por el personal de DCE, al igual que en la vista anterior se utiliza una tabla, entre los datos están: Licenciatura, cantidad de estudiantes de nuevo ingreso, regulares, retirados, en artículo 3, en artículo 6, artículo 7 y numero de secciones ofertadas, calificadas y sin calificar sumando un total de diez (10) columnas. En la siguiente figura 3.2.12 se puede visualizar el contenido:

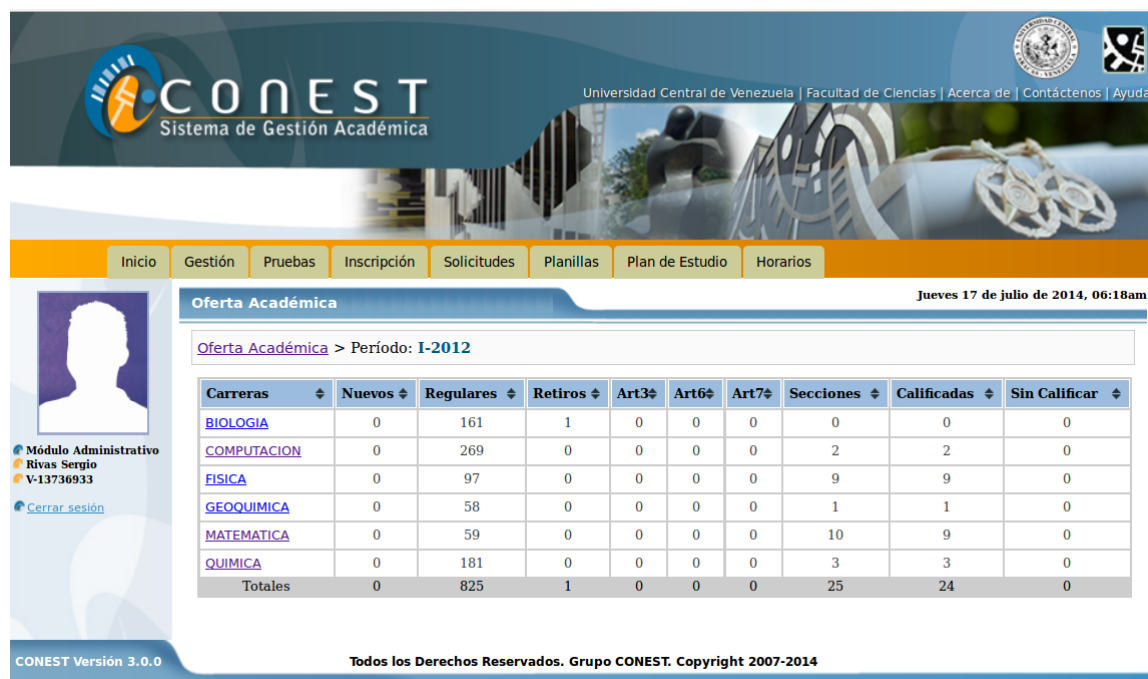


Figura 3.2.12: Prototipo ejecutable del listado de las licenciaturas

La vista anterior le proporciona al usuario enlaces que le permite conocer la Planificación Docente asociada a una licenciatura de acuerdo con su elección.

Se emplean tablas para exponer a la información de manera más organizada, el contenido de la figura 3.2.13 cuenta con una tabla de dos (2) columnas, la primera representa las asignaturas ofertadas con el código y nombre, asimismo, una pequeña lista de atributos como: capacidad de estudiantes por sección, iniciales de las licenciaturas para la cual está abierta, si está disponible en preinscripción y el coordinador de la asignatura. En la segunda columna se visualiza el conjunto de secciones, por medio de otra tabla con el nombre de la sección, el número de estudiantes, profesor, horario, listados y la opción de eliminar. También se simboliza la opción de agregar una sección por un botón.

En la parte superior de la tabla principal se cuenta con filtros de licenciaturas y semestres, en la parte inferior se dispone de un botón para agregar una nueva asignatura a la planificación, cabe destacar que ninguna de las acciones están realizadas para este punto simplemente es una representación.



CONEST

Sistema de Gestión Académica

Universidad Central de Venezuela

Facultad de Ciencias

Acerca de

Contactenos

Ayuda



Inicio

Gestión

Pruebas

Inscripción

Solicitudes

Planillas

Plan de Estudio

Horarios



Módulo Administrativo

Rivas Sergio

V-13736933

Camara sesión

Oferta Académica

Jueves 17 de julio de 2014, 06:20am

Oferta Académica > Período: I-2012 > Carrera: Q

Carrera: B | C | F | G | M | Q | Todas

Semestre: 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Todos

Asignatura	Sección																		
(2103) FÍSICA GENERAL I - Capacidad: xx - Abierta para: xx xx ... - Abierta en preinscripción: Si - Coordinador: Prof. xx	<table> <thead> <tr> <th>Nombre</th> <th># Est.</th> <th>Profesor</th> <th>Horario</th> <th>Listados</th> <th>Eliminar</th> </tr> </thead> <tbody> <tr> <td>U1</td> <td>18</td> <td>Prof. XX XX</td> <td>Modificar</td> <td>PDF - Excel</td> <td>- Eliminar</td> </tr> </tbody> </table> + Agregar Sección	Nombre	# Est.	Profesor	Horario	Listados	Eliminar	U1	18	Prof. XX XX	Modificar	PDF - Excel	- Eliminar						
Nombre	# Est.	Profesor	Horario	Listados	Eliminar														
U1	18	Prof. XX XX	Modificar	PDF - Excel	- Eliminar														
(3401) PRINCIPIOS QUÍMICA I - Capacidad: xx - Abierta para: xx xx ... - Abierta en preinscripción: Si - Coordinador: Prof. xx	<table> <thead> <tr> <th>Nombre</th> <th># Est.</th> <th>Profesor</th> <th>Horario</th> <th>Listados</th> <th>Eliminar</th> </tr> </thead> <tbody> <tr> <td>U1</td> <td>32</td> <td>Prof. XX XX</td> <td>Modificar</td> <td>PDF - Excel</td> <td>- Eliminar</td> </tr> </tbody> </table> + Agregar Sección	Nombre	# Est.	Profesor	Horario	Listados	Eliminar	U1	32	Prof. XX XX	Modificar	PDF - Excel	- Eliminar						
Nombre	# Est.	Profesor	Horario	Listados	Eliminar														
U1	32	Prof. XX XX	Modificar	PDF - Excel	- Eliminar														
(8206) MATEMÁTICA I - Capacidad: xx - Abierta para: xx xx ... - Abierta en preinscripción: Si - Coordinador: Prof. xx	<table> <thead> <tr> <th>Nombre</th> <th># Est.</th> <th>Profesor</th> <th>Horario</th> <th>Listados</th> <th>Eliminar</th> </tr> </thead> <tbody> <tr> <td>U1</td> <td>30</td> <td>Prof. XX XX</td> <td>Modificar</td> <td>PDF - Excel</td> <td>- Eliminar</td> </tr> <tr> <td>U2</td> <td>30</td> <td>Prof. XX XX</td> <td>Modificar</td> <td>PDF - Excel</td> <td>- Eliminar</td> </tr> </tbody> </table> + Agregar Sección	Nombre	# Est.	Profesor	Horario	Listados	Eliminar	U1	30	Prof. XX XX	Modificar	PDF - Excel	- Eliminar	U2	30	Prof. XX XX	Modificar	PDF - Excel	- Eliminar
Nombre	# Est.	Profesor	Horario	Listados	Eliminar														
U1	30	Prof. XX XX	Modificar	PDF - Excel	- Eliminar														
U2	30	Prof. XX XX	Modificar	PDF - Excel	- Eliminar														

+ Agregar Materia

CONEST Versión 3.0.0

Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2014

Figura 3.2.13: Prototipo rápido del listado de las asignaturas

Si el usuario requiere acceder a una sección para visualizar los estudiantes inscritos, lo puede efectuar a través del nombre de dicha sección, logrando realizar otras funciones que están fuera del alcance de este TEG pero son accesibles por medio de la Planificación Docente, tales como: Visualizar el historial académico y la calificación de la sección. En la siguiente se observa un ejemplo3.2.14.



CONEST

Sistema de Gestión Académica

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

Inicio

Gestión

Pruebas

Inscripción

Solicitudes

Planillas

Plan de Estudio

Horarios



Módulo Administrativo

Rivas Sergio

V-13736933

[Cerrar sesión](#)

Oferta Académica

Jueves 17 de julio de 2014, 06:20am

Oferta Académica > Período: I-2012 > Carrera: Q

2103 - FISICA GENERAL I - Sección: U1

Estudiantes

#	Carrera	Foto	Cédula	Nombre	Nota Final	Nota Rep	En Letra	Retirar
1	Q		21337350	ASCANIO MUÑOZ ARIANA STEPHANIA	10	-	-	Retirar
2	F		21337350	ASCANIO MUÑOZ ARIANA STEPHANIA	10		-	Retirar
3	F		22785112	BARRIOS ARVELAEZ CLAUDIA VEROWSKA	11	-	-	Retirar
4	M		21344404	BERMUDEZ COLMENARES THIRONE JOSE	13	-	-	Retirar
5	F		21345397	BLANCO APONTE MARIOLGA ESTEFANIA	13	-	-	Retirar
6	F		22522065	CENTEME HERNANDEZ DARLIN ERNESTINA	07		-	Retirar
7	M		83020425	ENCISO SIRIVICHE MARCO ANTONIO	03		-	Retirar

Figura 3.2.14: Interfaz de Usuario para visualizar una sección

En el contexto de implementación para el prototipo rápido se aplicó una hoja de estilo, la cual puede observarse en la figura 3.2.15. La hoja de estilo es una colección de reglas que define la apariencia de una aplicación Web. Estas reglas se refieren al modo en que aparecerá la interfaz de usuario, controlando por ejemplo el color, el fondo, tipografía, apariencia de los bordes, márgenes, alineación, espacio entre caracteres entre otros. Facilitando la consistencia y homogeneidad en el diseño y la imagen de la aplicación.

```

4  #enlinea{
5      display:inline-block;
6      margin: 10px 6px 0px 6px;
7      //width: 120px;
8  }
9
10 #iconos{ display:inline-block; vertical-align: top; }
11
12 #algoritmo{ font-size: 12px; width: 352px; font-family: Verdana;}
13
14 #enlinea_radio{
15     display:inline-block;
16     margin: 4px 6px 0px 6px;
17     vertical-align: top;
18 }
19
20 a { color: #368EBD; }
21
22 #iconos a { font-weight: bold; }
23
24 #cont_oferta{ margin: auto; width: 80%; }
25
26 #cont_oferta select{ padding: 1px 4px 1px; }
27
28 #cont_oferta #select_inicio, #cont_oferta #select_fin{
29     display:inline-block;
30     width: 214px;
31     margin: 18px 0px 6px 0px;
32 }
33
34 #hr_elim_s{
35     margin:10px 0px;
36     border: 0;
37     height: 12px;
38     box-shadow: inset 0px 11px 10px -13px #9BB5DC;
39 }
40
41 #horario materia .celda-horario { font-size: 1em;}

```

Figura 3.2.15: Hoja de estilo

El prototipo rápido es el producto que se obtiene de todas las etapas y tareas anteriores no obstante, puede irse ajustando hasta llegar a un prototipo ejecutable listo para la fase de entrega y liberación.

3.2.1.4. Entrega

El producto desarrollado en la última etapa de esta iteración no se consideró listo para la liberación final, es solo una versión de prueba obtenida a partir del prototipo rápido y usado para la posterior evaluación con el equipo de desarrollo en la segunda reunión pautada. Durante la iteración se realizan pruebas unitarias a los métodos desarrollados y de integración con el sistema CONEST.

3.2.2. Iteración II Gestión de la Planificación Docente

Esta iteración se deriva de la anterior iteración por estar enfocada en generar el código fuente o la programación de las opciones de gestión sobre la Planificación Docente. Se pretende incorporar el concepto de usabilidad en cada una de las etapas del ciclo con ayuda del método AgilUS, permitiendo al usuario efectuar cada una de las funcionalidades de manera intuitiva, sencilla y satisfactoria.

3.2.2.1. Requisitos

Se considera la etapa con mayor valor en el ciclo de desarrollo, de manera que si se efectúa un análisis incorrecto sobre la lista de requerimientos se conseguiría un software que no cumple con las necesidades del usuario. Por esta razón se utilizaron técnicas como la tormenta de ideas y los requerimientos funcionales que permiten fortalecer la identificación de los requisitos.

■ Tormenta de Ideas

A manera ilustrativa, se presenta en la figura 3.2.16 notas que surgieron de la segunda reunión efectuada con el equipo de desarrollo. En la cual, se planteó como solución crear una Planificación Docente, sin partir de cero asignando automáticamente un conjunto de asignaturas acordes con el pensum de cada licenciatura, como las obligatorias, TEGs, Seminarios, Pasantías, Servicio Comunitario y Métodos. Asimismo, se planteó ofrecer al usuario un estimado del número de secciones y la capacidad de estudiantes por secciones al momento de agregar una asignatura a la planificación.

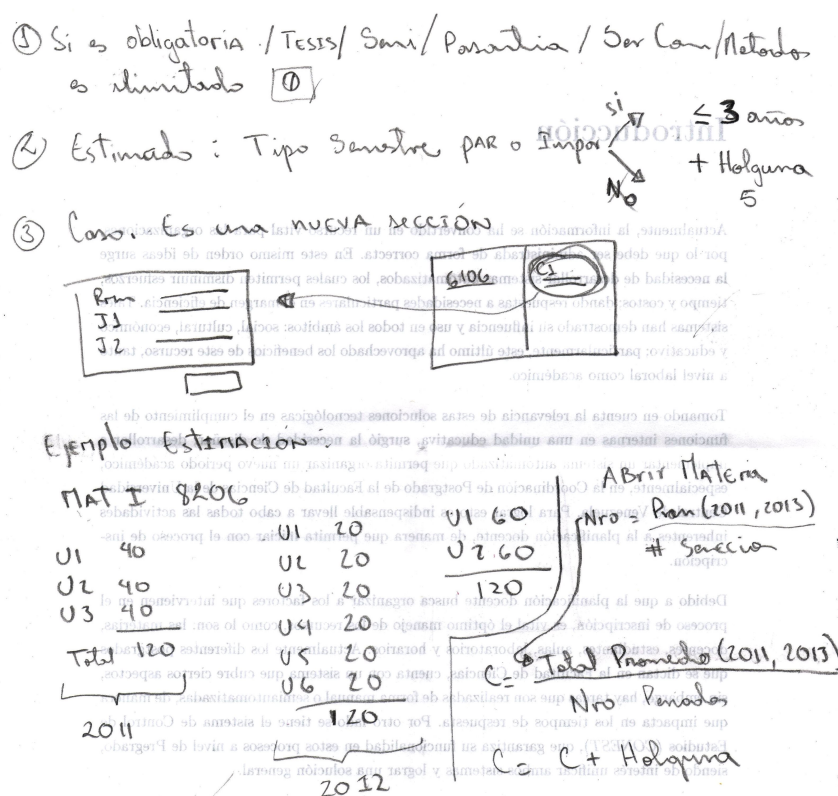


Figura 3.2.16: Notas de una tormenta de ideas Iteración II

■ Requerimientos Funcionales

Posteriormente de realizar el análisis de los requerimientos a través de la técnica anteriormente descrita, se logró obtener las funcionalidades pertinentes a esta segunda iteración destacando:

- La resolución de actividades que presenta el usuario al momento de diseñar la Planificación Docente que desea, ya sea añadiendo, modificando o cerrando algún elemento no deseado como una asignatura, una sección, un docente, horario o atributos de los anteriores recursos que son necesarios para la administración.

3.2.2.2. Análisis

Para establecer un plan de ejecución para la implementación de las funcionalidades, se elaboró un diagrama de casos de uso para modelar el funcionamiento del sistema y la interacción con el usuario. Es importante destacar que el diagrama es una extensión de otro modelo, mostrado en la figura 3.2.17. Posteriormente se describe de este artefacto:

■ Modelo de Casos de Uso

Particularmente la figura 3.2.17 es un diagrama de casos de uso que define las funcionalidades de gestión sobre la Planificación Docente. Entre los elementos más resaltantes están las asignaturas, secciones, grupo docente, horarios entre otros. Básicamente es un refinamiento de los casos de uso principales y sus relaciones en el diagrama visualizado en la figura 3.2.3, no constituyen un objetivo autónomo sólo sirven para complementar al nivel superior.

Siempre existe el caso de contemplar una modificación en la planificación, ya sea por un error en flujo de trabajo o simplemente para ampliarla. A continuación se listan los casos de usos:

1. Gestionar Asignaturas: Agregar, modificar y cerrar una asignatura.
2. Modificar atributos de asignatura: es la opción de configurar las característica de la asignatura, tales como: agregar coordinador, preinscripción y disponibilidad para otras licenciaturas.
3. Gestionar sección: Abrir, cerrar y cambiar el nombre de una sección.
4. Gestionar Grupo Docente: Incluye dos acciones la de asignar y modificar grupo docente.
5. Listado de sección: este caso permite al usuario obtener el listado en dos formatos PDF y Excel.
6. Gestionar Horario: Incluye casos de uso como: Asignar el horario, modificar y eliminar el horario a una sección
7. Visualizar horarios: Este caso de uso extiende a la visualización de la planificación, los horarios pueden ser observados por asignaturas o por semestre.
8. Visualizar asignaturas: Este caso de uso incluye a la visualización de la planificación y esta puede estar filtrada por tipo de asignatura, semestre o licenciatura.

9. Resumen de Planificación Docente: El usuario tiene la posibilidad de un respaldo de la planificación en un documento en formato PDF.

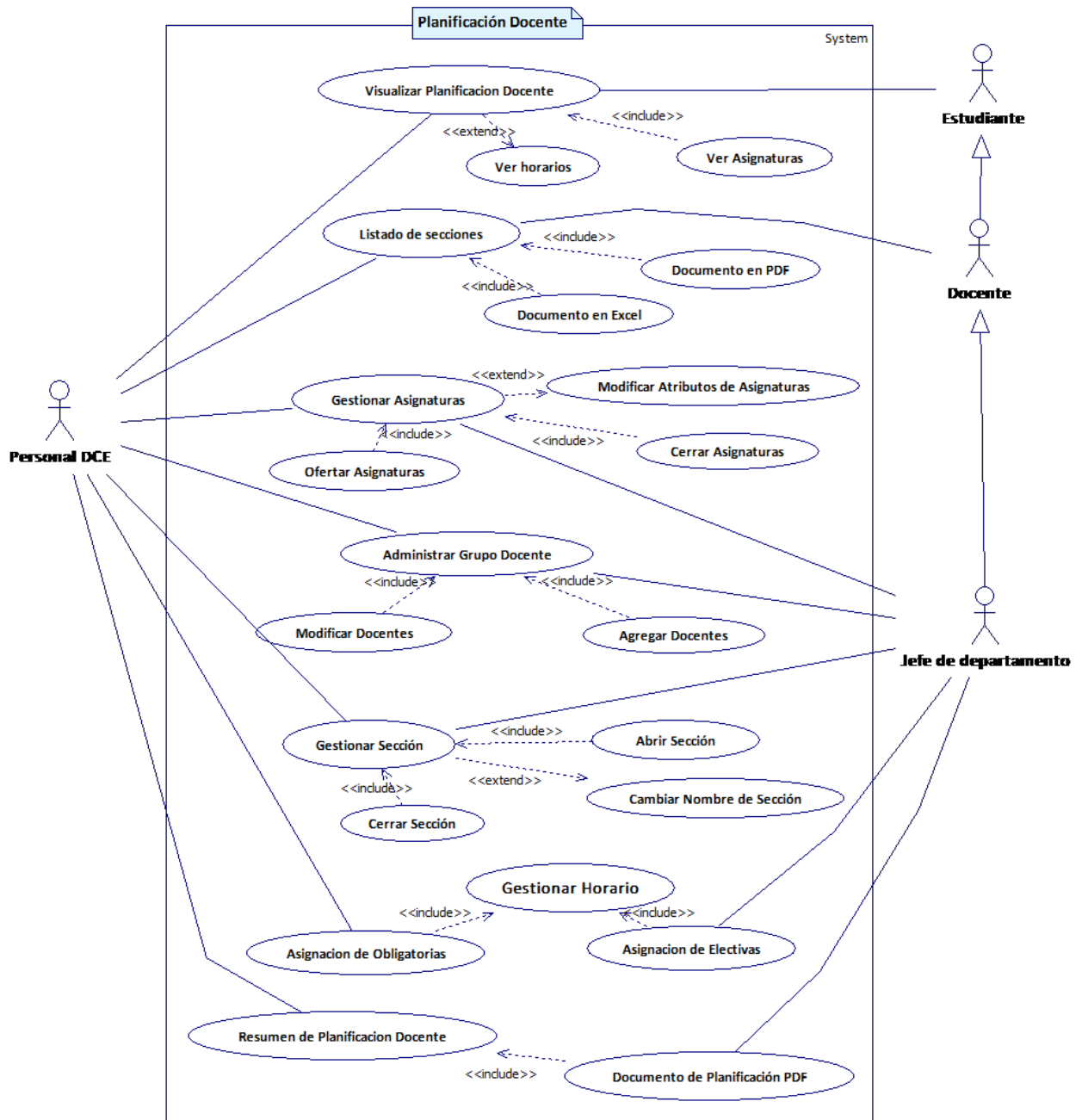


Figura 3.2.17: Caso de Uso de la Planificación Docente, nivel 2

3.2.2.3. Prototipaje

En esta etapa los requisitos propuestos en esta iteración se generan a través de los siguientes artefactos: prototipo rápido y prototipo ejecutable.

- Prototipo Rápido

La mayoría de las acciones realizadas en el módulo de Planificación Docente conlleva a la presencia de una ventana modal. A continuación, se describen algunas de estas acciones formulando un claro ejemplo de ello a través de la representación gráfica.

Antes de crear una nueva Planificación Docente es necesario abrir un nuevo período lectivo a través del sistema el cual estará asociado a la planificación. En la figura 3.2.18 se observa la vista que interactúa con el usuario cuando desea agregar un periodo académico, entre los datos solicitados por esta vista se encuentran: La fecha de inicio, fecha de culminación, tipo de período (Intensivo, periodo 01, período 02), si es obligatorio, si aplica reglamento y si está activo. Esta vista comprende la restricción de solo ser accesible por el usuario con el rol de administrador.

A screenshot of a web application window titled 'CONEST' with a close button in the top right corner. The main content area is titled 'Datos del Nuevo Periodo Académico:'. It contains several form fields: 'Fecha inicio' with a date picker showing '21', 'febrero', and '2014'; 'Fecha fin' with a date picker showing '21', 'febrero', and '2014'; 'Tipo' with a dropdown menu showing 'Intensivo'; 'Obligatorio' with an unchecked checkbox; 'Aplica reglamento' with an unchecked checkbox; and 'Activo' with an unchecked checkbox. An 'Aceptar' button is located at the bottom right of the form area.

Figura 3.2.18: Prototipo rápido para agregar un periodo académico

El prototipo que se presenta en la figura 3.2.20 consiste en la interfaz que permite agregar una nueva asignatura a la planificación, a través, de la cual se le solicita al usuario los siguientes campos:

- El código de la asignatura, es un campo obligatorio y de autocompletación a medida que el usuario introduce el código en el sistema, este busca el nombre respectivo al código.
- El coordinador de la asignatura, este campo no es necesario para agregar la asignatura, al igual que el anterior es de autocompletación por medio de la cédula del docente.
- El número de secciones, es un campo de tipo combobox que por defecto tiene una (1) sección. El sistema sugiere al usuario el número de secciones a ofertar acordes con la asignatura.

- Cupo por sección, es el último campo solicitado al usuario es de tipo combobox y por defecto tiene la capacidad en cero (0), el significado de este número consiste en que la asignatura no tiene límites de estudiantes, un ejemplo son las asignaturas obligatorias. Este campo también es sugerido por el sistema. La figura 3.2.19 muestra el método utilizado para calcular el número de secciones y el cupo por sección.

```
def calcular_capacidad_secciones(materia, carrera, periodo) #----retorna [0] es la capacidad x secciones [1] nro de se
i = 1
k = 0
resultado = []
nro_secciones_total = 0
aux_capacidad = 0
historico = 2
holgura = 5
periodo_id = periodo.split('/')
while k < historico do
  aux_periodo = "#{periodo_id[0]}-#{(periodo_id[1].to_i-i).to_s}"
  secciones = OfertaEnPeriodo.where(periodo_academico_id: aux_periodo, materia_codigo: materia, carrera_id: carrera)
  unless secciones.blank?
    secciones = secciones[0].secciones
    secciones.each do |c|
      if (c.nombre_seccion != 'EQ') && (c.nombre_seccion != 'SU')
        aux_capacidad = c.capacidad+aux_capacidad
        nro_secciones_total = nro_secciones_total + 1
      end
    end
  end
  k += 1
  i += 1
end

if nro_secciones_total == 0
  aux_nro_secciones = 2
else
  aux_nro_secciones = nro_secciones_total
end

nro_secciones_total = 1 if nro_secciones_total == 0 # Revisar esto o
capacidad = aux_capacidad/nro_secciones_total
capacidad = capacidad + holgura if capacidad != 0
resultado[0] = capacidad
if nro_secciones_total == 1 && aux_nro_secciones == 2
  resultado[1] = 0
else
  resultado[1] = aux_nro_secciones/historico
  if (aux_nro_secciones.to_f/historico)-(aux_nro_secciones/historico) >= 0.1
    resultado[1] = (aux_nro_secciones/historico)+1
  end
end
return resultado
end # ---- Cierre de calcular capacidad y secciones ----#
```

Figura 3.2.19: Método para el número de secciones y capacidad

Con ayuda de la técnica de desarrollo para aplicaciones interactivas AJAX, se logró realizar las dos sugerencias del número de secciones y cupo por sección, la cual se obtiene después de introducir el código de la asignatura. A continuación se muestra el conjunto de líneas de código empleadas:

```
escapeinside
1 window.agregar_materia_coffee =>
2   $('#autocomplete_materia__codigo').blur =>
3     if ($('#materia_codigo').val() != ""
4       materia_codigo = ($('#materia_codigo').val()
5       periodo_id = ($('#periodo_id').val()
6       carrera_id = ($('#carrera_id').val()
```

```

7      parametros_adicionales = '&carrera_id='+carrera_id+'&periodo_id='+
      periodo_id+'&materia_codigo='+materia_codigo
8
9      $.ajax
10     url: 'resul_capacidad_secciones?'+parametros_adicionales,
11     success: (datos) ->
12         resultado = JSON.parse(datos)
13         unless resultado.existe == null
14             $('#mensaje_alerta p #texto').text resultado.existe;
15             $('#mensaje_alerta').show();
16         else
17             $('#mensaje_alerta p #texto').text '';
18             $('#mensaje_alerta').hide();
19             nro_secciones = parseInt(resultado.nro_secciones)
20             capacidad = parseInt(resultado.capacidad)
21             $('#nro_secciones option').eq(nro_secciones-1).prop('selected',true)
22             ;
23             $('#capacidad option').eq(
24                 capacidad).prop('selected',true);

```

Figura 3.2.20: Prototipo rápido para agregar una asignatura

Otra actividad de gestión sobre la Planificación Docente, reside en cerrar una sección no deseada. La figura 3.2.21 muestra la ventana de confirmación con información que corresponde a la sección como los estudiantes inscritos y el coordinador, adicionalmente el usuario debe indicar si desea notificar por medio de correo electrónico el cierre de la sección y/o distribuir a los estudiantes inscritos.

CONEST

Eliminar La Seccion EQ

Estudiantes inscritos 3 Coordinador MARIA RAICES

Notificar por Correo Electronico

Coordinador ☒ Estudiantes ☐

Distribuir estudiantes ☐

Aceptar

Figura 3.2.21: Prototipo para cerrar una asignatura

A cada sección ofertada le concierne un grupo docente que está compuesto por tres (3) integrantes, el docente principal quien sería el responsable de la calificación y dos (2) jurados. La siguiente interfaz 3.2.22 es utilizada para las acciones de añadir y modificar al grupo docente correspondiente. Esta vista tiene como restricción que los integrantes del grupo sean distintos entre sí.

CONEST

Jurado de la sección U1

Jurado principal
16497888 - Medina Zaida

Jurado
11917857 - QuiÑones Jackeline

Jurado
3184527 - Dunia a Emery r

Guardar

Figura 3.2.22: Prototipo rápido para el grupo docente

Por otra parte si se requiere cerrar una asignatura que no corresponde con la Planificación Docente, surgen dos estados de la asignatura a considerar antes de cerrarla, el primero y el ideal es que no contenga estudiantes y el segundo estado nace por motivos erróneos de ofertar la asignatura en inscripción conllevando a tener estudiantes inscritos motivo por el cual se muestra un mensaje indicando que primero debe eliminar las secciones con estudiantes o debe distribuir al alumnado en otras asignaturas. Para ejemplo esta la figura 3.2.23:

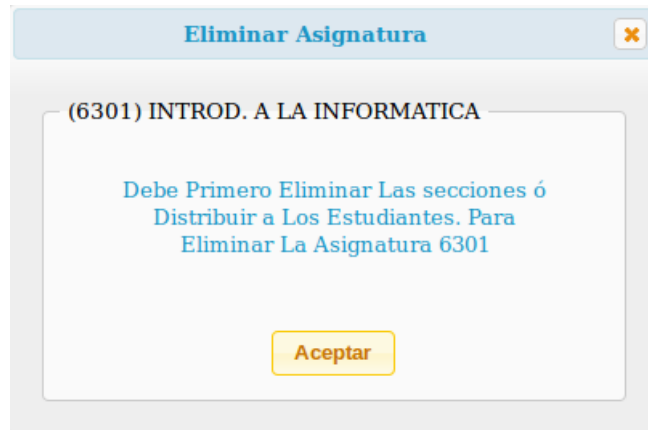


Figura 3.2.23: Prototipo rápido para cerrar asignatura

El prototipo rápido realizado fue base fundamental en la resolución del problema, a partir de este se agregaron y modificaron elementos sin perder la esencia del prototipo, un notable ejemplo de esto es el color de la ventana modal de la última vista que se presenta, esta cambia de color naranja a azul permaneciendo así hasta la entrega final.

■ Evaluación Heurística

La técnica de inspección utilizada es la de evaluación heurística, se empleó con el fin de comprobar aspectos asociados al diseño, navegación, operación y utilidad del sistema. Se usaron las diez (10) heurísticas planteadas en [11] y siguiendo la escala de evaluación que va de uno (1) a cinco (5), siendo uno de menor importancia y cinco un estado fatal.

1. Visibilidad del estado del sistema: El sistema siempre debe mantener a los usuarios informados sobre el estado del sistema, a través de una adecuada retroalimentación y en un tiempo razonable.
2. Similitud entre el sistema y el mundo real: El sistema debe hablar el idioma de los usuarios, con palabras, frases y conceptos familiares para el usuario, en lugar de términos orientados al sistema, haciendo que la información aparezca en un orden natural y lógico.
3. Control y libertad del usuario: Los usuarios a menudo eligen funciones del sistema por error y necesitarán una salida emergente para cambiar el estado no deseado sin tener que pasar por un diálogo extendido. Soporte de deshacer y rehacer.
4. Consistencia y cumplimiento de estándares: Los usuarios no deberían tener que preguntarse si diferentes palabras, situaciones o acciones significan lo mismo. Debe seguir las normas y convenciones de la plataforma.
5. Prevención de errores: Mide la preparación del sistema para prevenir y gestionar errores.

6. Reconocimiento en lugar de recordar: Minimizar la carga de la memoria del usuario. El usuario no debería tener que recordar la información de una parte a la otra.
7. Flexibilidad y eficiencia de uso: Las instrucciones para el uso del sistema deben ser visibles o fácilmente accesibles siempre que se necesiten. Los aceleradores no vistos por el usuario principiante, mejoran la interacción para el usuario experto de tal manera que el sistema puede servir para usuarios inexpertos y con experiencia.
8. Los diálogos estéticos y diseño minimalista: Los diálogos no deben contener información que es irrelevante o raramente necesaria. Cada unidad extra de información en un diálogo compite con las unidades relevantes de información y disminuye su visibilidad relativa.
9. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores: Los mensajes de error se deben expresar en lenguaje sencillo (sin códigos), indica con precisión el problema y sugerir una solución constructiva.
10. Ayuda y documentación: Aunque es mejor si el sistema puede ser utilizado sin la documentación, puede ser necesario proporcionar ayuda y documentación. Ésta tiene que ser fácil de buscar, centrada en las tareas del usuario, tener información de las etapas a realizar y que no sea muy extensa.

En seguimiento a esta actividad se exhibe el cuadro 3.2, que contiene algunas de fallas e inconvenientes encontrados en el prototipo después de aplicar la evaluación a ciertos usuarios del sistema.

Cuadro 3.2: Evaluación heurística para la iteración II de Gestión

Nro	Problema	Heurística	Valoración
1	Los campos de las fechas deben ser representados a través de calendarios en vez de combo box	H6	2
2	No se cuenta con un indicador de los campos obligatorios.	H1, H5, H6	3
3	Los mensajes son muy largos y en ocasiones están mal redactados, falta de aceptos	H9, H4	2
4	No se permite cambiar o modificar las licenciaturas para las cuales se oferto una asignatura	H3	5
5	No se puede cambiar el nombre de una sección	H3	3
6	El color de la letras del mensaje para cerrar una asignatura no corresponde con el resto	H4	1
7	No se cuenta con subtítulos en la acción que indique en qué punto de la selección de datos se encuentra	H10	2

■ Prototipo Ejecutable

Después de obtener los resultados de la evaluación heurística se procedió a realizar cambios en el prototipo antes descrito, mejorando la apariencia, organización y agregando nuevas funcionalidades como validaciones.

Es importante destacar, el uso de JavaScript, de su Framework JQuery y su librería JQueryUI los cuales fueron empleados para: ordenar (TableSorter), mostrar el calendario y seleccionar una fecha (Datepicker) y mostrar las ventanas modales.

Efectuando un contraste entre las figuras 3.2.18 y 3.2.24 correspondientes a la interfaz usada por el usuario para agregar un nuevo período académico. Se observa que en la última figura mejora la organización y añade indicativos en los campos que son obligatorios por medio de un asterisco. Asimismo, se modificaron los campos de fecha de inicio y fin, en ellos se utilizó un calendario para agregar la fecha este forma parte del plugin de JQuery y es conocido como Datepicker.

Agregar Período Académico

Nuevo Periodo Académico:

Fecha Inicio *

Fecha Fin *

Tipo *

Obligatorio ☐

Aplica Reglamento ☐

Activo ☒

Aceptar

Figura 3.2.24: Ventana modal para agregar un período académico

Datepicker

September 2014

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

(a) Datepicker

Una vez que el usuario acepta agregar el período, el sistema compara el tipo de semestre en caso de ser par (período 01) o impar (período 02) se cargan un conjunto de asignaturas partiendo de una planificación anterior con la misma condición. Entre el conjunto de asignaturas se encuentran las obligatorias y en casos especiales los TEGs, Seminarios, Pasantías, Servicio Comunitario y Métodos. Este proceso ocurre para cada una de las licenciaturas y tomando en cuenta el pensum respectivo. En el caso del semestre de tipo intensivo solo se crea el período académico y la planificación vacía sin asignaturas.

La vista para agregar una asignatura no sufrió grandes cambios como se muestra en la figura 3.2.25

Agregar Asignatura

Datos de la Asignatura

Asignatura *
0031

Coordinador Docente
[Empty text box]

Nro de Secciones **Cap. por Sección**

1 0

Guardar

Figura 3.2.25: Ventana modal para agregar una asignatura

Siguiendo esta línea es necesario considerar que las equivocaciones o errores son inherentes en los procesos sobre todo en aquellos con mayor flujo de información, motivo por el cual surge la siguiente opción en la figura 3.2.26 esta permite cambiar y modificar las licenciaturas para las cuales se ofertó la asignatura pero siempre dentro del marco de su pensum.

CONEST

Asignatura Abierta En:

Biología ☐	Computación ☐
Física ☐	Geoquímica ☒
Matemática ☐	Química ☐

Guardar

Figura 3.2.26: Ventana modal para cambiar las licenciaturas

En el mismo orden de ideas encontramos la opción de cambiar la capacidad por sección que posee una asignatura, como ejemplo la siguiente figura 3.2.27:

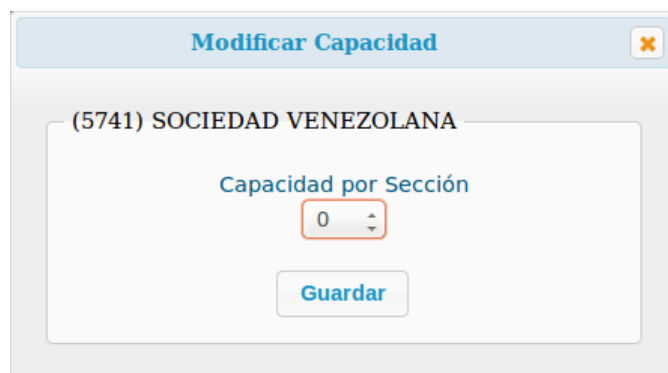


Figura 3.2.27: Ventana modal para modificar la capacidad

De igual forma el sistema ofrece la opción de agregar y modificar al Coordinador de la asignatura. Usando la misma interfaz para ambas opciones a través de una ventana modal como se visualiza en la figura 3.2.28, permitiendo al usuario mantener la noción de la tarea que está realizando y sobre cual elemento la está efectuando. Los campos para agregar o modificar al grupo docente son de autocompletación, colocando la cédula el sistema busca el nombre asociado.

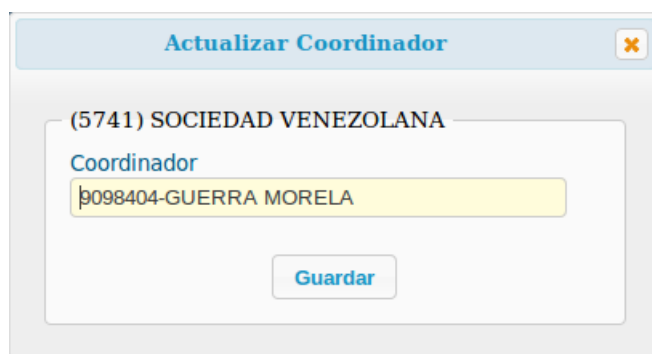


Figura 3.2.28: Ventana modal para el coordinador de la asignatura

En esta fase de entrega la presentación del color de la tipografía del mensaje que corresponde al cierre de la asignatura se cambió a negro y se agregó un área de texto para conocer los motivos de la eliminación de la asignatura, esta vista aparece en el caso ideal cuando la asignatura no tiene estudiantes inscritos. A continuación en la figura 3.2.29 muestra un ejemplo:

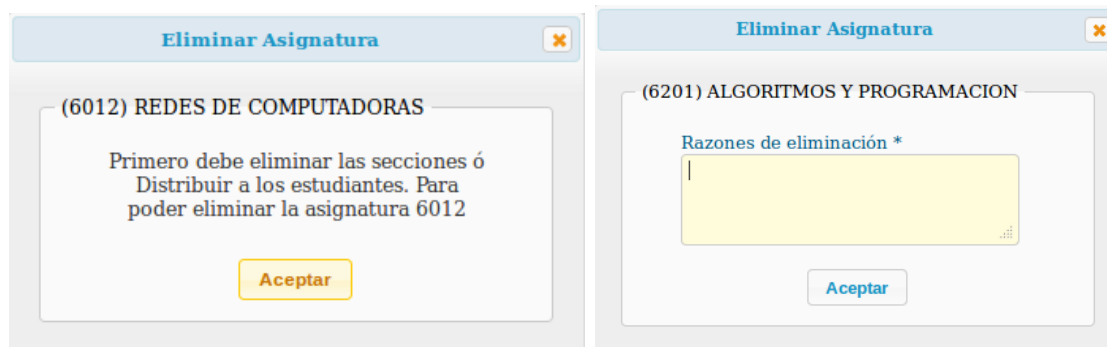


Figura 3.2.29: Ventana modal para el cierre de asignatura con estudiantes

Otro hecho a considerar es querer agregar una nueva sección a la asignatura, el sistema presenta al usuario una ventana de confirmación, si la acción es afirmativa se usan los atributos de las otras secciones para crear la nueva.

Si bien es cierto que por algún motivo el usuario desea modificar el nombre de la sección, también se puede realizar, ya que, está disponible en el módulo a través de la siguiente interfaz 3.2.30.

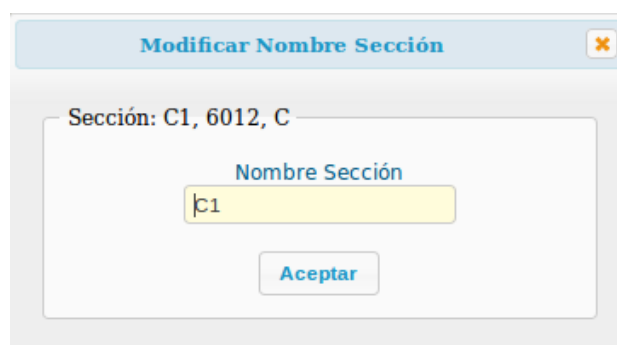


Figura 3.2.30: Ventana modal para modificar el nombre de la sección

Por otra parte la vista mostrada en la figura 3.2.31 solo estuvo sometida a un cambio, asociado al subtítulo que indica en que punto de la selección de datos se encuentra como la sección, al código de la asignatura y la inicial de la licenciatura a la cual se está modificando.

Figura 3.2.31: Ventana modal para agregar grupo docente

Al igual que el cierre de las asignaturas sucede con las secciones hay que tener en cuenta dos estados antes de cerrar dicha sección, el primero y el más ideal es que no contenga estudiantes y el segundo estado es el caso contrario. Si no se tienen estudiantes, el usuario debe introducir en la segunda modal las razones del cierre que está en la figura 3.2.32. En caso contrario el usuario tiene dos (2) opciones la primera eliminar la sección sin interesar a los estudiantes inscritos, la segunda distribuir en otras secciones restantes, el ejemplo de esta ventana modal es la que se visualiza a la izquierda en la figura 3.2.32. Cabe resaltar que en esta fase se consideró innecesaria la notificación tanto para el docente como para el alumnado, motivo por el cual se eliminó.

Figura 3.2.32: Ventana modal para cerrar una sección

3.2.2.4. Entrega

Para esta etapa se consideró necesario realizar pruebas unitarias, de integración y de aceptación, con el fin de verificar el cumplimiento de las funcionalidades en respuesta a las necesidades de los usuarios. En este sentido, el módulo debe generar los resultados esperados y sin errores en base a las entradas del usuario.

Las pruebas realizadas permitieron corregir, cambiar y recodificar parte del código empleado. En ejemplo de ello, están las pruebas realizadas a los formularios de agregar un período académico, una asignatura y el grupo docente, se comprobó que al introducir datos erróneos en los campos se muestra el mensaje correspondiente y, en el caso contrario que desplegara tanto los datos esperados como las acciones pertinentes. Las pruebas de integración estuvieron asociadas a la unificación de las funcionalidades desarrolladas en esta iteración con el resto del sistema y finalmente las pruebas de aceptación se efectuaron con ayuda del personal de la DCE, éstas permitieron mejorar significativamente el módulo e identificar nuevas funcionalidades al mismo.

3.2.3. Iteración III Reportes y Horarios

La iteración está centrada en desarrollar los reportes del listado de las secciones, el resumen por licenciatura de la Planificación Docente y la presentación los horarios tanto en los soportes como en resumen por semestres y en la visualización general de la planificación.

3.2.3.1. Requisitos

Como seguimiento de esta actividad y siendo primordial en la obtención de información se planteó desarrollar la siguiente actividad:

- **Tormenta de Ideas**

Las ideas expuestas en la figura 3.2.33 forman parte del estudio efectuado por el equipo de desarrollo y la apreciación de algunos usuarios de la DCE sobre el producto obtenido hasta ese momento. Esta evaluación se llevó a cabo mediante una reunión, en la cual, se discutieron aspectos de estructura y organización para el listado de los estudiantes que pertenecen a una sección utilizando los distintos formatos PDF y Excel, así como la presencia de los horarios en el resumen por licenciatura de la Planificación Docente en PDF y HTML. Otros puntos tomados en cuenta son la revisión y modificación de los mensajes expuestos a los usuarios y los usados en el registro de bitácora.

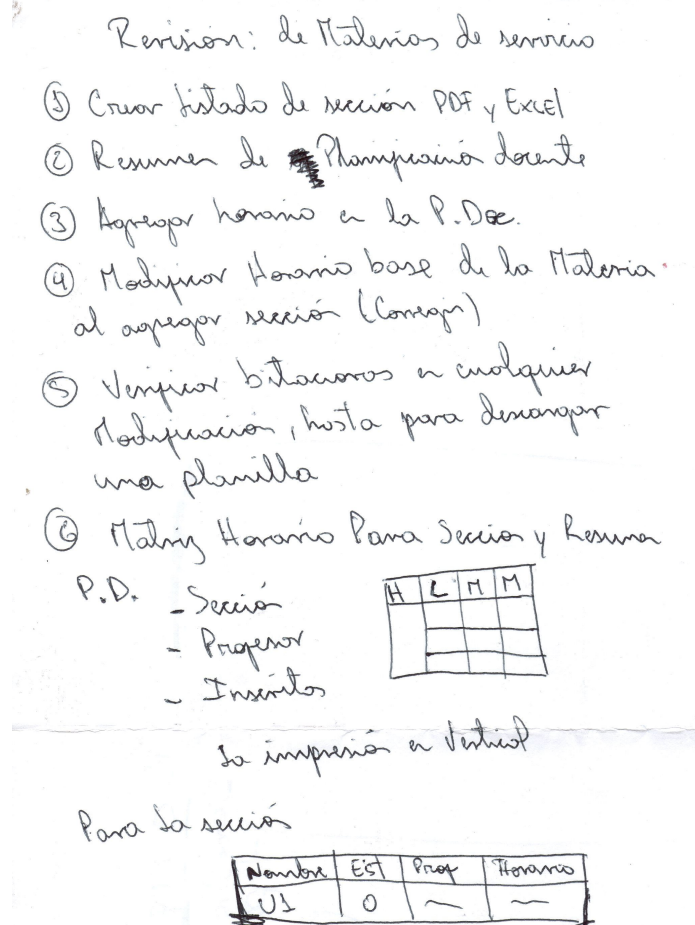


Figura 3.2.33: Tormenta de Ideas Iteración III

3.2.3.2. Análisis

Tras la solución de los requerimientos se engloba un plan de diseño y ejecución, de manera que para hacerlo posible se utilizan los siguientes artefactos: El prototipo en papel y la guía de estilo. A continuación se desglosan:

■ Prototipo en Papel

No cabe duda que la realización de los prototipos en papel tiende a ser rápida, además, son independientes de la tecnología y de herramientas específicas, económicos en recursos y permiten la detección temprana de errores de concepto y problemas de funcionalidades.

El siguiente prototipo 3.2.34 describe la ubicación del horario según las asignaturas presentes en la Planificación Docente de acuerdo a los filtros seleccionados, adicionalmente se suma un nuevo filtro relacionado con el tipo de asignatura de ejemplo están: Obligatorias, electivas, complementarias, servicio comunitario, pasantías entre otras.

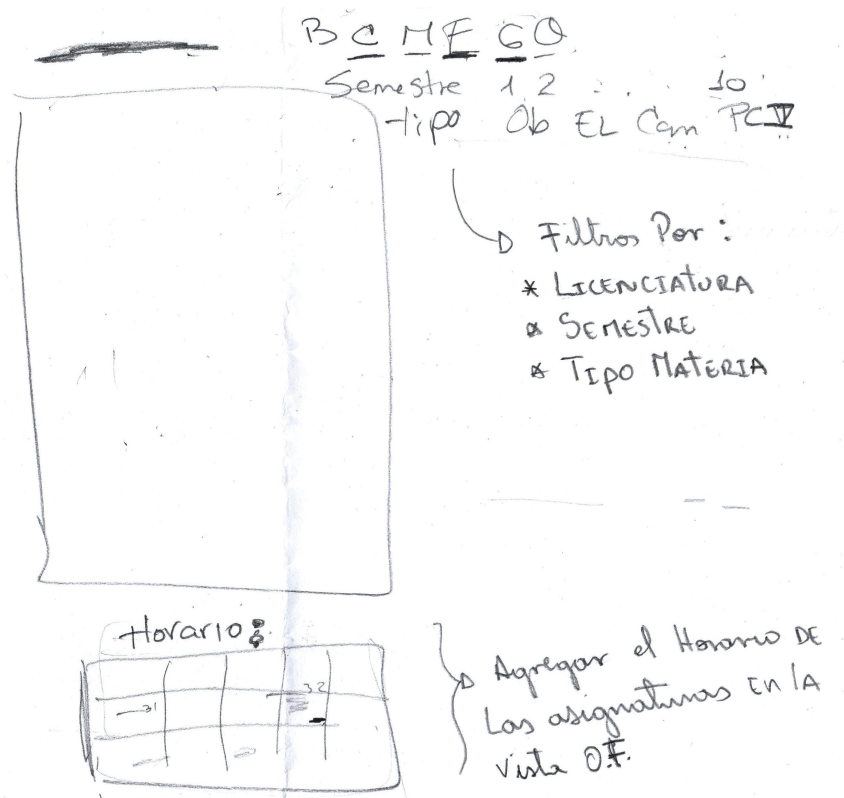


Figura 3.2.34: Prototipo en papel horarios en la vista de asignaturas

Con el fin de esclarecer algunas interrogantes pautadas por el autor se consideró ideal realizar un encuentro informal con el equipo de desarrollo. Las inquietudes residían en la organización y los datos que estarían presente en el resumen de la Planificación Docente por licenciatura en formato PDF, a partir de allí, se elaboró un prototipo en papel que se muestra a continuación en la figura 3.2.35

ENCABEZADO

Resumen Oferta
ACADEMICA

3.2. Método Ágil

SEMESTRE 1

3. Datos	
- Se Preinscribe	
- Docente principal	
- Es de Servicio	
- Se necesita Aula	
- ASIGNATURA	
- (COD. NOMBRE)	
- Tipo	
- Nro. SECCIONES	

SEMESTRE 2

HORARIO	
Hora	
1 ~	
2 ~	
3 ~	

- hora
- Docente
- Sección
- Aula

Figura 3.2.35: Resumen de la Planificación Docente en PDF

El resumen inicia con el membrete de la Facultad, el título, el periodo académico, la licenciatura y continúa con una tabla que especifica las asignaturas y sus datos, tales como: Coordinador, se preinscribe, es de servicio, requiere aula, tipo de asignatura y número de secciones. Las tablas son agrupadas por semestre y al final de cada tabla aparece el horario asociado si aplica.

■ Guía de Estilo

Dentro de este marco la guía de estilo relata la plantilla utilizada en los documentos con formatos PDFs.

● Plantilla

Los documentos emitidos por el sistema deben poseer un encabezado y un pie de página. El encabezado contiene el logotipo de la Universidad y el de la Facultad de Ciencias, así como el membrete de la Facultad mencionada. Por otra parte el

pie de página tiene el teléfono, fax, correo y dirección electrónica del sistema. En la siguiente figura 3.2.36 se muestra la plantilla usada para los documentos PDFs.

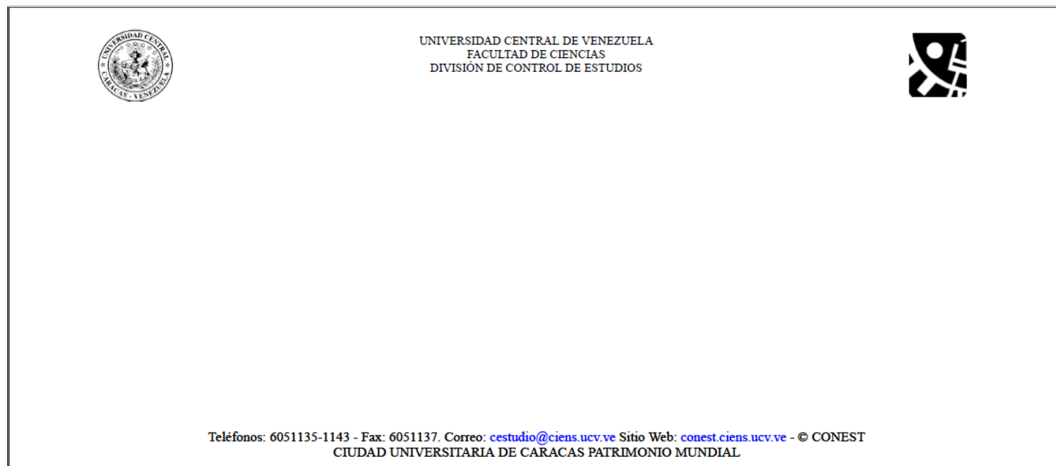


Figura 3.2.36: Plantilla para Documentos PDF

3.2.3.3. Prototipaje

Los prototipos rápidos jugaron un papel esencial en el desarrollo de los requerimientos planteados para esta iteración, ya que, sobre los mismos se efectuaron modificaciones y cambios necesarios para obtener un producto que se ajusta a los objetivos de esta investigación.

■ Prototipo Rápido

Se elaboró un prototipo rápido para el listado de la sección en PDF, el mismo permitió palpar el documento efectuando modificaciones sobre los campos no deseados y agregar datos al documento que son útiles para el personal administrativo y docente. Cabe destacar que los datos del listado en el formato Excel son similares al PDF.

De igual manera se elaboró un prototipo rápido para el resumen de la Planificación Docente por licenciatura, se puede observar una representación en la figura 3.2.37. En secciones previas se realizó la descripción general del documento sin embargo, el esquema para horario de las asignaturas se llevó a cabo mediante una tabla que consta de seis (6) columnas, la primera columna indica la hora a dictar la asignatura y las restantes contemplan los días de la semana (Lunes, martes, miércoles, jueves y viernes) junto con el código y sección. Al igual que el listado de sección este prototipo facilitó la evaluación y modificación del documento.

 UNIVERSIDAD CENTRAL DE VENEZUELA FACULTAD DE CIENCIAS COORDINACIÓN ACADÉMICA DIVISION DE CONTROL DE ESTUDIOS OFERTA ACADÉMICA 							
PERIODO ACADÉMICO: 02-2013 LICENCIATURA: QUÍMICA							
SEMESTRE: 0							
ASIGNATURA	TIPO	COORDINADOR	PREINSCRIPCIÓN	AULA	DE SERVICIO	SECCIONES	
(0007) DEPORTES	C					1	
(0031) INGLÉS II	O			SI	SI	9	
(0206) ACTIVIDAD CORAL	C					1	
(0250) BIOTECNIA	C					1	
(0387) RIES. COTID. ECOSUM.	PCI					1	
(0391) ETICA SABERES Y SOCIEDAD	C					1	
(0468) MANEJO INTEGRAL DE RESIDU	C			SI		1	
(0900) SEMINARIO SERVICIO COMUN	SC					1	
(0901) SERVICIO COMUNITARIO	SC					1	
(1927) FILOSOFIA DE LA CIENCIA	C			SI		2	
(3495) SEM. DE INVESTIGACION	S					3	
(3496) T.E.G. (OPC. BASICA)	T					3	
(3497) T.E.G. (OPC. GEOQUIMICA)	T					1	
(3498) T.E.G. (OPC. TECNOLOGIA)	T					5	
(3538) PASANTIAS INDUSTRIALES	P	LOPEZ CARMEN				1	
(3929) MANEJO DE DATOS QUIMICOS	E	CASTILLO JIMMY				1	

HORA	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES
1:00 - 1:59			0007 - Sec. U1		
2:00 - 2:59					

HORA	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES

Figura 3.2.37: Resumen de la Planificación Docente en PDF

En ocasiones es necesario que el usuario visualice el horario de una asignatura en específico, por esta razón, se elaboró un prototipo con esta opción mediante una ventana modal. Esta opción se ubica en la Planificación Docente en la columna de las asignaturas y en la celda junto con las características de la misma. La figura 3.2.38 es un ejemplo del horario de una asignatura.

Horario de la Asignatura					
Hora	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES
3:00 - 5:00		(6106)- MATEMÁTICAS DISCRETAS I * C1 - ROJAS OTILIO (AULA 19) * C2 - FELIPE VÍCTOR (AULA 17) * C3 - MANNARINO ILIANA (AULA 24) * C4 - DOMÍNGUEZ OLGA (AULA 20)		(6106)- MATEMÁTICAS DISCRETAS I * C1 - ROJAS OTILIO (AULA 19) * C2 - FELIPE VÍCTOR (AULA 12) * C3 - MANNARINO ILIANA (AULA 10) * C4 - DOMÍNGUEZ OLGA (AULA 20)	(6106)- MATEMÁTICAS DISCRETAS I * C1 - ROJAS OTILIO (AULA 19) * C2 - FELIPE VÍCTOR (AULA 27) * C3 - MANNARINO ILIANA (AULA 21) * C4 - DOMÍNGUEZ OLGA (AULA 20)

Figura 3.2.38: Ventana modal para el horario una asignatura

Después de realizar el prototipo rápido se efectuó una reunión con el equipo de trabajo para evaluar el mismo y se identificaron algunas fallas en relación a la estructura y a la ausencia de información relevante para el usuario. Los participantes de este proceso sugirieron agregar algunos aspectos al listado de la sección en PDF destacando en ellos el horario de la sección, la nota y el resumen si la sección está calificada. Por otra parte, se consideró necesario cambiar los datos expuestos en el horario de los documentos como los visualizados en la interfaz. Algunos de los cambios se pueden observar en la figura 3.2.39.



LISTADO DE ESTUDIANTES

PERIODO ACADÉMICO: 01-2013
ASIGNATURA: (5142) INFORMÁTICA
DOCENTE COORDINADOR: Espinoza Cesar
DOCENTE PRINCIPAL: Espinoza Cesar
SECCIÓN: G1

NRO	LICENCIATURA	CÉDULA	NOMBRES Y APELLIDOS	CORREO
1	GEOQUIMICA	18088940	HERNANDEZ HERNANDEZ ISABEL CRISTINA	uncorreoservidor.com
2	GEOQUIMICA	18749481	DE ABREU PEÑALOZA DANIEL ALEJANDRO	uncorreoservidor.com
3	GEOQUIMICA	19737805	LOPEZ GUARAMATO JOHN YAVETH	uncorreoservidor.com
4	GEOQUIMICA	20975396	SOLORZANO GUGLIETTA JORGE LUIS	uncorreoservidor.com
5	GEOQUIMICA	21346547	GONZALEZ PEÑA ANA GABRIELA	uncorreoservidor.com
6	GEOQUIMICA	22035728	DE QUEZADA NUÑEZ JOHAN MARIO	uncorreoservidor.com
7	GEOQUIMICA	22905077	PAREJO APONTE DANIELA	uncorreoservidor.com
8	GEOQUIMICA	23608871	MORALES CORRO YUSBELY MARIANA	uncorreoservidor.com
9	GEOQUIMICA	24073346	FAVARO SANOJA ALICIA	uncorreoservidor.com
10	GEOQUIMICA	24461422	BERMÚDEZ CARTAYA YAIMAR ALEJANDRA	uncorreoservidor.com
11	GEOQUIMICA	24843430	JIMENEZ URBINA ALFREDO ALEXANDER	uncorreoservidor.com
12	GEOQUIMICA	24898163	MERKL SIERRA DANIEL ERNESTO	uncorreoservidor.com
13	GEOQUIMICA	25038155	RIVERA TOVAR JHON JAIRO	uncorreoservidor.com
14	GEOQUIMICA	25070188	GERIG REYES JOANY ISABEL	uncorreoservidor.com
15	GEOQUIMICA	25232653	GUILLEN RIVAS YANNILET DEL VALLE	uncorreoservidor.com
16	GEOQUIMICA	25234218	RANGEL OROPEZA HERYK JOSE	uncorreoservidor.com

↓ Foto
↑ Version Inmate
(con iniciales de los segundos)
Agregar Horario y Aulas

FECHA

Página 1 de 1

FALTA (PIE DE PAGINA EN LA ULTIMA)

Figura 3.2.39: Prototipo rápido de listado de sección en PDF

■ Prototipo Ejecutable

La figura 3.2.40 muestra al listado de la sección en PDF, en una versión lista para la liberación. De manera que la misma se logró aplicando cada una de las modificaciones planteadas en la lista de comprobación, entre ellas: El horario de la sección, la fecha de emisión por el sistema, la foto del estudiante, la nota y el resumen. Las dos (2) últimas forman parte de una condición aplicada al documento, de tener estudiantes calificados se cambia la columna estado por la de nota y se agrega el resumen con los estudiantes aprobados, aplazados, retirados, número de estudiantes por calificar, nota

mínima, promedio y nota máxima.

EMITIDO: 18/07/2014 06:32AM



UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS
DIVISIÓN DE CONTROL DE ESTUDIOS

G1 - (5741) SOCIEDAD VENEZOLANA



PERÍODO ACADÉMICO: 02-2013
COORDINADOR: GUERRA MORELA
DOCENTE: GUERRA MORELA

HORA	LUNES	MARTES	MÉRCOLES	JUEVES	VIERNES
8:00 - 10:00		(5741) SOCIEDAD VENEZOLANA *GU-GUERRA MORELA			
10:00 - 11:00		(5741) SOCIEDAD VENEZOLANA *GU-GUERRA MORELA			

NRO	FOTO	LIC	CÉDULA	NOMBRE Y APELLIDO	CORREO	NOTA
1		G	21103691	CHISPO G. JETSY L.	uncorroo@servidor.com	RET
2		G	21582372	DÍAZ V. LUIS A.	uncorroo@servidor.com	17
3		G	23960835	QUEVEDO G. ROSA A.	uncorroo@servidor.com	16
4		G	20129921	SANCHEZ S. REINALDO D.	uncorroo@servidor.com	12

RESUMEN

Indicador	Valor
Nro. de estudiantes aprobados	3
Nro. de estudiantes aplazados	0
Nro. de estudiantes retirados	1
Nro. de estudiantes sin calificar	0
Nota mínima	12
Promedio	15,00
Nota máxima	17

Teléfonos: 6051135-1143 - Fax: 6051137. Correo: control@ciencia.ucv.ve Sitio Web: control.ciencia.ucv.ve - © CONEST
CIUDAD UNIVERSITARIA DE CARACAS PATRIMONIO MUNDIAL

Figura 3.2.40: Listado de sección en PDF

El listado de la sección en Excel, mostrado en la figura 3.2.41 es otro aspecto importante en la lista de requerimientos, éste se desarrolló con la estructura de datos usando en el formato PDF. El Excel consta de siete (7) columnas: El número del estudiante en la lista, la licenciatura, la cédula, los nombres, los apellidos, el correo electrónico, el estado si se retiró de la asignatura (RET). Adicionalmente, se le aplicó al documento una condición, de estar calificado al menos un estudiante se modifica la columna de estado por una de notas.

EMITIDO: 18/07/2014 06:04AM



UNIVERSIDAD CENTRAL DE VENEZUELA
 FACULTAD DE CIENCIAS
 DIVISION DE CONTROL DE ESTUDIOS
 OFERTA ACADÉMICA



PERÍODO ACADÉMICO: 02-2013
LICENCIATURA: GEOQUÍMICA

SEMESTRE: 1

ASIGNATURA	TIPO	COORDINADOR	PREINSCRIPCIÓN	AULA	DE SERVICIO	SECCIONES
(5111) QUÍMICA GENERAL I	O	BARROS CARLOS		SI		1
(5142) INFORMÁTICA	O	ESPINOZA CESAR		SI		1
(5221) GEOLOGÍA FÍSICA	O	OUTIERREZ JOSÉ		SI		1

HORA	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES
7:00 - 9:00		(5221) GEOLOGÍA FÍSICA *01-OUTIERREZ JOSÉ (03)		(5221) GEOLOGÍA FÍSICA *01-OUTIERREZ JOSÉ (03)	(5111) QUÍMICA GENERAL I *01-BARROS CARLOS (22)
9:00 - 11:00		(5111) QUÍMICA GENERAL I *01-BARROS CARLOS (22)		(5111) QUÍMICA GENERAL I *01-BARROS CARLOS (22)	(5142) INFORMÁTICA *01-ESPINOZA CESAR (22)

PÁG. 1 de 9

Figura 3.2.42: Resumen de Planificación Docente

El resumen de horarios en la figura 3.2.44, es una opción accesible desde la barra de menú y está ubicada en el submenú de la pestaña de procesos, opción disponible para usuario con rol de administrador, ver la figura 3.2.43. Dicho resumen dispone de dos filtros; el primero por licenciatura y el segundo por semestre. El horario se presenta mediante una tabla con la hora, los días de la semana y la asignatura. En secciones anteriores se hizo referencia a un arreglo de colores para identificar las asignaturas en la tabla de horarios, el campo de la asignatura contiene: El código, nombre, secciones junto con los nombres de los docentes y aula si aplica.



Figura 3.2.43: Ubicación de la opción Resumen de Horarios

Horario	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES
7:00 - 9:00	(6002) - ORG. Y ESTRUCT. DEL COMP. II * C1 - PARADA JAIME (AULA 21) * C2 - LARES JESUS (AULA 15)	(6203) - INGENIERIA DE SOFTWARE * C2 - RIVAS KEYLA (AULA 33)	(6002) - ORG. Y ESTRUCT. DEL COMP. II * C1 - PARADA JAIME (AULA 10) * C2 - LARES JESUS (AULA 17)	(0030) - INGLES I * U7 (AULA 32) (6203) - INGENIERIA DE SOFTWARE * C2 - RIVAS KEYLA (AULA 33)	(6203) - INGENIERIA DE SOFTWARE * C2 - RIVAS KEYLA (AULA 26)
9:00 - 11:00	(8208) - MATEMATICA III * U1 - KEY MARIA (AULA 25) * U2 - VILLEGAS BERTHA (AULA 26)	(8208) - MATEMATICA III * U1 - KEY MARIA (AULA 25) * U2 - VILLEGAS BERTHA (AULA 27)	(0030) - INGLES I * U3 (AULA 27) * U4 (8208) - MATEMATICA III * U1 - KEY MARIA (AULA 15) * U2 - VILLEGAS BERTHA (AULA 14)	(8208) - MATEMATICA III * U1 - KEY MARIA (AULA 26) * U2 - VILLEGAS BERTHA (AULA 28)	(6002) - ORG. Y ESTRUCT. DEL COMP. II * C1 - PARADA JAIME (AULA 19) * C2 - LARES JESUS (AULA 33)
11:00 - 1:00	(8208) - MATEMATICA III * U3 - VALERA MAIRA (AULA 06)	(0030) - INGLES I * U1 (AULA 25) (6203) - INGENIERIA DE SOFTWARE * C1 - NIÑO NORELVA (AULA 21) (8208) - MATEMATICA III * U3 - VALERA MAIRA (AULA 30)	(8208) - MATEMATICA III * U3 - VALERA MAIRA (AULA 30)	(6203) - INGENIERIA DE SOFTWARE * C1 - NIÑO NORELVA (AULA 21) (8208) - MATEMATICA III * U3 - VALERA MAIRA (AULA 30)	(6203) - INGENIERIA DE SOFTWARE * C1 - NIÑO NORELVA (AULA 30)

Figura 3.2.44: Resumen de Horarios interfaz

3.2.3.4. Entrega

En cualquier codificación es necesario garantizar el correcto funcionamiento del sistema, por ello es indispensable realizar pruebas unitarias y de integración durante el ciclo de desarrollo del software para obtener un producto usable que efectue cada una de las actividades pautadas en la Planificación Docente.

De acuerdo a los objetivos planteados, se lograron alcanzar algunos con ayuda de los prototipos rápidos mientras que otros se obtuvieron durante el prototipo ejecutable, el cual es utilizado en la etapa de entrega. Aun en este ciclo el prototipo desarrollado no se considera listo para su puesta en producción.

3.2.4. Iteración IV Estado de la Calificación

Para los jefes de departamento y el personal administrativo es importante auditar y conocer el estado de la calificación, es por esta razón, que los requerimientos funcionales de la iteración tienden a ser informativos y complementarios para el módulo. Adicionalmente, se realizan correcciones, revisiones y pruebas para verificar si el objetivo de la investigación se está cumpliendo a cabalidad.

3.2.4.1. Requisitos

La mayoría de los requisitos han sido obtenidos y puntualizados con ayuda de las técnicas de indagación, aunque entre ellas, la más resaltante es la tormenta de ideas por su uso frecuente en cada una de las iteraciones.

- **Tormenta de Ideas**

El inicio de cada iteración se caracteriza por las reuniones que estuvieron pautadas con el equipo de desarrollo, ya que, las mismas facilitaron la evaluación constante del sistema y la identificación de nuevos requisitos. De un proceso similar se obtuvieron los nuevos requerimientos para el módulo en desarrollo los cuales estuvieron plasmados mediante una tormenta de ideas, un ejemplo de ello es la figura 3.2.45. Entre los requisitos encontramos: La visualización de una barra de progreso para el proceso de calificación desde la interfaz de la Planificación Docente, una opción que permita recordar a los docentes que deben realizar la calificación de la sección a través de un correo electrónico, indicadores de calificación utilizando la metáfora del semáforo y finalmente se planteó agregar el horario del estudiante en la vista inicial del mismo.

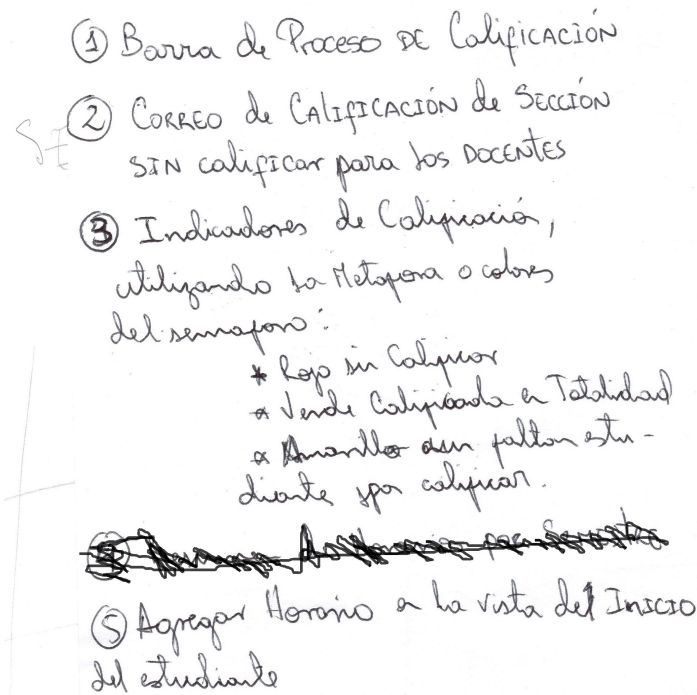


Figura 3.2.45: Tormenta de Ideas Iteración IV

3.2.4.2. Análisis

En torno a la información recopilada en la primera etapa de esta iteración, se consideró pertinente llevar a cabo el análisis de la solución de los requerimientos a desarrollar. A continuación se describen algunas de las actividades empleadas en este proceso:

■ Guía de Estilo:

La siguiente guía de estilo se empleó básicamente en los indicadores y en la barra de progreso de calificación, en esta guía se definen los colores y parte del estilo de la tipografía. Entre los colores empleados están: El rojo, naranja y verde. Representación gráfica de los colores en la figura 3.2.46.



Figura 3.2.46: Colores empleados para indicadores

3.2.4.3. Prototipaje

En ocasiones es ideal crear un esquema de los aspectos más complejos, de esta manera se diseñaron prototipos rápidos para estructurar la información en las vistas y para luego refinar los métodos de las funcionalidades en los prototipos ejecutables.

■ Prototipo Rápido

El prototipo rápido visualizado la figura 3.2.47 inició con la implementación de los indicadores de calificación, los cuales permiten al usuario identificar si una sección está calificada en su totalidad, parcialmente o continúa sin ser calificada. Para lograr captar esta idea se utilizó como metáfora los colores del semáforo de la siguiente manera:

- El color rojo indica que la sección está sin calificar.
- El color naranja indica que la sección está parcialmente calificada.
- El color verde indica que la sección está calificada en su totalidad.

(8203) TEC.Y RECURSOS DE APREN

Figura 3.2.47: Prototipo rápido del estatus de la calificación

Cuando el usuario no reconoce el significado de los colores, debe ubicarse con el puntero encima del nombre de la sección y aparecerá un título con el significado correspondiente. Cabe destacar, que los indicadores solo se muestran al estar activos ciertos parámetros del sistema asociados al proceso de calificación. En el siguiente código de programa escrito en CoffeeScript se muestra parte de la implementación de estos títulos.

escapeinside

```

1 $ ->
2   $(".rojo").attr "title", "Sección sin calificar"
3   $(".verde").attr "title", "Sección calificada"
4   $(''.naranja').attr "title", 'Sección sin terminar de calificar '
```

Después de realizar los indicadores se continuó por implementar la barra de progreso de calificación por asignatura. A continuación se muestra parte del código en CoffeeScript y la vista preliminar de la misma en 3.2.48

escapeinside

```
1 $ ->
2 $( 'td #progressbar ' ).each ->
3   valor = $(this).attr('value')
4   $(this).progressbar value: parseInt(valor)
```

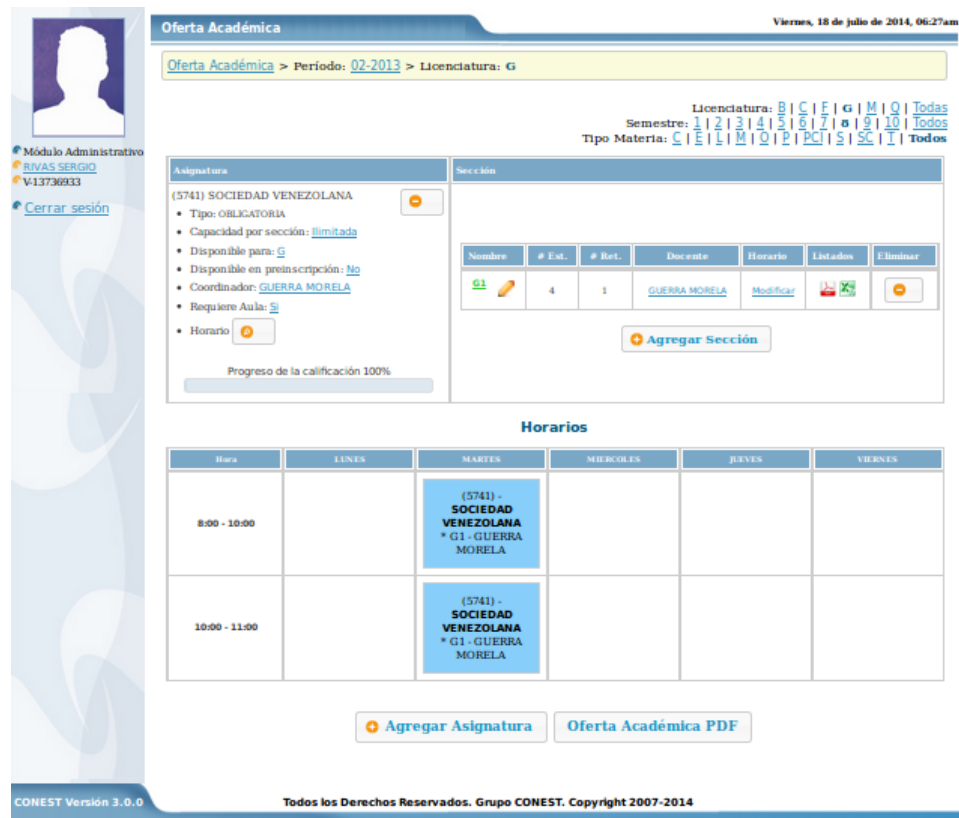


Figura 3.2.48: Barra de progreso de calificación.

Otro prototipo rápido elaborado es aquel asociado la activación de la asignatura para el proceso de preinscripción, la organización de la interfaz permite visualizar al conjunto de estudiantes agrupados y listados por aceptados y rechazados en el proceso si aplica. Como ejemplo grafico esta la siguiente figura 3.2.49.

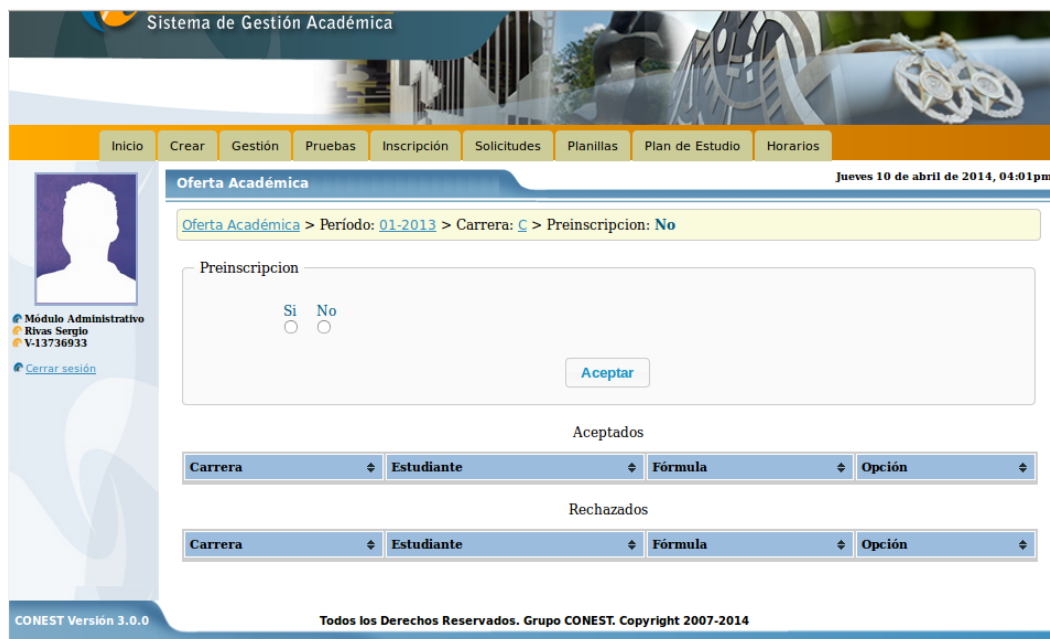


Figura 3.2.49: Prototipo rápido de configuración de asignatura

■ Prototipo Ejecutable

Después de culminar los prototipos rápidos estuvieron expuestos a evaluación para asegurar la obtención de un prototipo ejecutable usable con el menor número de modificaciones posteriores.

La figura 3.2.49 representa el prototipo ejecutable para la configuración de la asignatura para el proceso de preinscripción esta opción es accesible desde la interfaz de la Planificación Docente. Para la elaboración se usó la estructura mostrada en la figura 3.2.49 continuando con la implementación las funciones que permiten cambiar el estatus de preinscripción de la asignatura, el algoritmo de selección, así como modificar la opción que solo permitir a los estudiantes preinscritos.

En el listado de los estudiantes aceptados y rechazados se modificaron los datos visibles para el usuario, tales como: La posición en el listado, la foto del estudiante, inicial de la escuela, nombre completo, fórmula, eficiencia, créditos aprobados, promedio de notas, prioridad de la preinscripción por parte del estudiante y finalmente la acción. La última permite al usuario cambiar el estado del estudiante de aceptado ha rechazado y viceversa.


CONEST
 Sistema de Gestión Académica


 Universidad Central de Venezuela | Facultad de Ciencias

Inicio Social Procesos Usuarios Solicitudes Planillas Aulas Planes de Estudio Sistema


 Módulo Administrativo
 RIVAS SERGIO
 V-13736933
 Cerrar sesión

Oferta Académica
Viernes, 18 de julio de 2014, 06:17am

[Oferta Académica](#) > Período: [01-2014](#) > Licenciatura: [B](#)

Preinscripción: (2145) FOTOGRAFIA I (01-2014)

Configuración

Si ☒ No ☐ Algoritmo de Selección
 50 EFICIENCIA, 50 CREDITOS APROBADOS

Permitir solo a los estudiantes preinscritos
 Si ☐ No ☒

Guardar

Aceptados

#	Foto	Lic	Estudiante	Fórmula	Eficiencia	Cred. Apro.	Promedio	Prioridad	Acción
1		F	RANITZ MORALES RALPH PEDRO 82238694	88.760	0.870	181	11.726	1	
2		B	TORREALBA ORDAZ CARLOS LUIS 20913933	87.500	1.000	150	15.360	1	

Figura 3.2.50: Interfaz para configurar una asignatura

Otra manera de acceder a la interfaz de usuario para configurar las asignaturas de preinscripción es por medio del listado de licenciaturas en la columna que indica el número de preinscritos, este es un enlace que despliega las asignaturas ofertadas para la preinscripción, ver la figura 3.2.51, el nombre de la asignatura es el enlace final para la opción de configuración de preinscripción.

The screenshot displays the CONEST Sistema de Gestión Académica interface. At the top, the logo and name 'CONEST Sistema de Gestión Académica' are visible, along with the university's name 'Universidad Central de Venezuela | Facultad de Ciencias'. A navigation bar includes links like 'Inicio', 'Social', 'Procesos', 'Usuarios', 'Solicitudes', 'Planillas', 'Aulas', 'Planes de Estudio', and 'Sistema'. The main content area is titled 'Preinscritos por asignatura' and shows data for the period '01-2014' and the subject 'BIOLOGIA'. A table lists pre-enrollment statistics for three subjects: FOTOGRAFIA I, INGLES I, and LABORATORIO DE PRINCIPIOS DE QUIMICA. A sidebar on the left shows a user profile for 'RIVAS SERGIO' and a 'Cerrar sesión' link. The footer indicates 'CONEST Versión 3.0.0' and 'Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2014'.

Código	Asignatura	Preinscritos	Cupos	Aceptados
2145	FOTOGRAFIA I	23	38	1
0030	INGLES I	27	1440	27
3404	LABORATORIO DE PRINCIPIOS DE QUIMICA	55	90	29
TOTAL		105	1568	57

Figura 3.2.51: Interfaz para visualizar las asignaturas en preinscripción

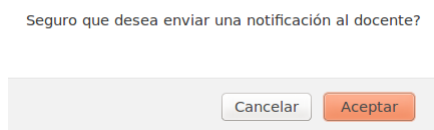
Entre los requisitos pautados para la iteración se encuentra la notificación a los docentes principales que aún no han calificado las secciones antes de cerrar el proceso de calificación mediante el uso de correos electrónicos, esta opción puede realizarse de dos formas la primera es global, en otras palabras enviar un correo a todos los docentes que no han calificado o realizarlo por sección uno a uno. Como representación gráfica está la figura 3.2.52, esta indica el semestre, el nombre de la asignatura, la sección, el coordinador de la asignatura, docente principal, los estudiantes inscritos y la opción de enviar el correo representado por un icono.

The screenshot shows the CONEST Sistema de Gestión Académica interface. The header includes the CONEST logo and the text 'Universidad Central de Venezuela | Facultad de Ciencias'. A navigation bar contains links: Inicio, Social, Procesos, Usuarios, Solicitudes, Planillas, Aulas, Planes de Estudio, and Sistema. The main content area is titled 'Secciones sin calificar' and shows the date 'Jueves 18 de septiembre de 2014, 06:28am'. Below this, there is a breadcrumb trail: 'Oferta Académica > Período: 02-2013 > Licenciatura: F'. A table lists the sections without grades:

Semestre	Asignatura	Sección	Coordinador	Docente	#Est	Opciones
7	(2471) INT. TEORIA CLA. CAMPOS	F1	ARIAS PIO	ARIAS PIO	5	
7	(2472) INTROD A LA RELATIVIDAD	F1	ARIAS PIO	ARIAS PIO	1	

On the left side, there is a user profile section with a silhouette icon, the text 'Módulo Administrativo', the name 'RIVAS SERGIO', the ID 'V-13736933', and a 'Cerrar sesión' link. The footer shows 'CONEST Versión 3.0.0' and 'Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2014'.

Figura 3.2.52: Listado de secciones sin calificar por licenciatura



(a) Ventana Emergente de Confirmación de Correo Electrónico

3.2.4.4. Entrega

Como seguimiento a esta etapa y última del ciclo de vida del desarrollo del software, es pertinente evaluar al sistema con el fin de comprobar si cumple a cabalidad con los objetivos planteados. En general, una prueba consiste en hacer uso de un producto para ver cómo funciona y qué resultado tiene [1]. Esta etapa se estructura en dos puntos la evaluación de usabilidad y aplicación liberada, a continuación la descripción:

1. Evaluación de Usabilidad:

Se realizó una prueba de aceptación llevada a cabo por usuarios que hacen uso frecuente de la Planificación Docente, dando a conocer su opinión general en base a los escenarios de uso del módulo de Planificación Docente, bajo la realización de cuatros (4) tareas claves: Agregar un período académico, agregar una asignatura a la planificación, agregar una sección y cerrar una sección. Esta prueba permitió descubrir errores menores escondidos a la vista del desarrollador.

En un escenario similar a la anterior prueba, se efectuó otra con la diferencia de que esta se realizó en un ambiente real y de producción. El objetivo de estas pruebas además

de certificar la funcionalidad del sistema es que permita lograr un producto usable. La prueba de aceptación se aplicó una encuesta que consta de cinco (5) preguntas directas y dos (2) preguntas de análisis a dos personas de la DCE con rol de administrador, Dinora Moreno y Milagros Graterol. En el siguiente cuadro 3.3 puede visualizar la estructura de la encuesta aplicada:

Cuadro 3.3: Encuesta aplicada en la prueba de aceptación

Nro	Pregunta	Si	No
1)	¿Considera usted que este módulo apoya y facilita la realización de las tareas en el proceso de Planificación Docente?		
2)	¿Opina usted que el módulo cumple con sus expectativas?		
3)	¿Cree usted que los errores humanos pueden verse reducidos al usar este sistema?		
4)	¿Considera usted que la interfaz del módulo es sencilla e intuitiva?		
5)	¿Los documentos generados por el módulo son satisfactorios y útiles?		
6)	¿Posee el Jefe de Departamento mayor participación en el sistema para distribuir la carga de la planificación?		
7)	¿Tiene usted alguna observación o recomendación que realizar al módulo de Planificación Docente?		

La figura 3.2.53 muestra la gráfica del resultado de la encuesta realizada por las cinco preguntas directas.

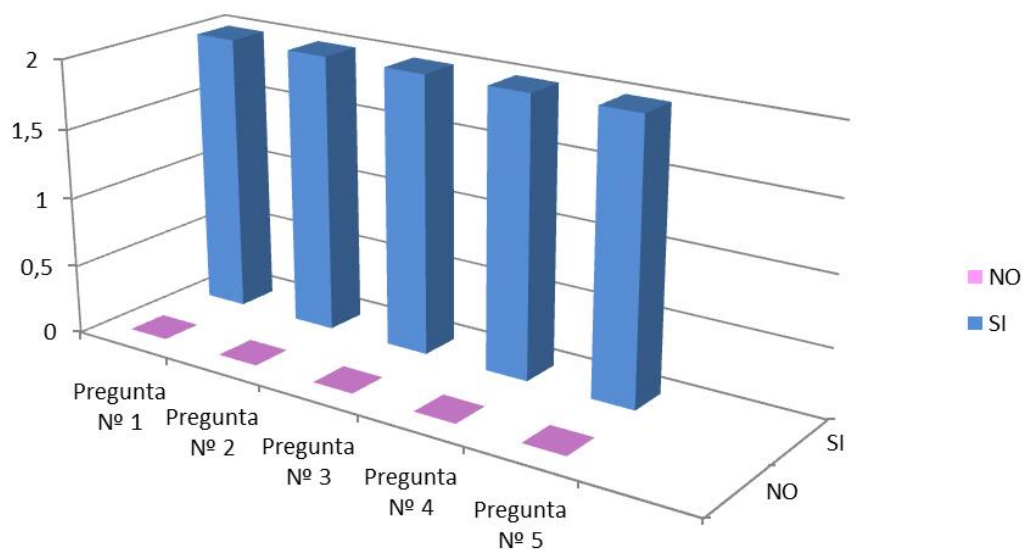


Figura 3.2.53: Resultados de la prueba de aceptación beta

Para concretar los resultados de la encuesta aplicada, los participantes dieron en su mayoría respuestas afirmativas (Si) a las preguntas directas.

Por otra parte para la pregunta seis (6) las respuestas textuales son las siguientes:

- Dinora Moreno: “A nivel del sistema sí, porque el carga directamente la planificación en el sistema y verifica que este correcta, o hace alguna modificación si lo amerita”
- Milagros Graterol: “Si, porque si no oferta las materias con los horarios, la responsabilidad es del docente (Jefe de Departamento)”

Ambas respuestas afirman que sí, el Jefe de Departamento tiene mayor participación en el sistema para distribuir la carga de la Planificación Docente. Finalmente para la última pregunta, no se obtuvieron observaciones ni recomendaciones. En resumen se puede alegar en base a los resultados obtenidos que el módulo de Planificación Docente fue aceptado y cumple con los requisitos del usuario.

2. Aplicación Liberada: Como resultado de aplicar las técnicas y artefactos del método AgilUs para cada una de sus etapas, se obtuvo un producto usable que previene posibles errores humanos, de manera que cumple con los requerimientos del usuario y los objetivos de este TEG. Es un software listo y puesto en producción para el período académico 02-2013 y 01-2014.

A continuación se presentan las vistas más significativas del módulo desarrollado:

La figura 3.2.54, es la interfaz que visualiza el usuario con rol de administrador después de iniciar sesión, en ella se listan los últimos cuatro (4) períodos académicos agregados

y cada período es un enlace que posteriormente lleva a la Planificación Docente. Desde esta vista es posible abrir un nuevo período académico, asociado a la planificación y junto a él se agregan automáticamente un conjunto de asignaturas (obligatorias, seminarios, pasantías, tesis y métodos) para cada licenciatura.



CONEST
Sistema de Gestión Académica

Universidad Central de Venezuela | Facultad de Ciencias

Inicio Social Procesos Usuarios Solicitudes Planillas Aulas Planes de Estudio Sistema

Oferta Académica Viernes, 18 de julio de 2014, 06:14am

Período	Nuevos	Regulares	Preinscritos	Retiros	Secciones	Nómina
01-2014	102	3038	824	183	355	
02-2013	811	3395	723	3571	345	
01-2013	98	3124	956	3092	344	
02-2012	783	3436	761	2297	340	

[Ver más ...](#)

[Agregar Período](#)

CONEST Versión 3.0.0 Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2014

Figura 3.2.54: Interfaz para visualizar los periodos académicos

Después de seleccionar un periodo académico, se despliega la vista mostrada en la figura 3.2.55 con las licenciaturas de la Facultad de Ciencias, los nombre de las licenciaturas son enlaces que corresponde con su Planificación Docente. Cada una de las vistas de la planificación tiene migas de pan para indicarle al usuario en que interfaz se encuentra.



CONEST
Sistema de Gestión Académica

Universidad Central de Venezuela | Facultad de Ciencias

Inicio Social Procesos Usuarios Solicitudes Planillas Aulas Planes de Estudio Sistema

Oferta Académica Viernes, 18 de julio de 2014, 06:25am

[Oferta Académica](#) > Período: **02-2013**

Licenciatura	Nuevos	Regulares	Preinscritos	Retiros	Secciones	Calificadas	Sin Calificar	Nómina
BIOLOGÍA	148	598	48	587	139	96	0	NM
COMPUTACION	195	1155	259	1168	182	137	0	NM
FISICA	152	411	98	458	100	86	2	NM
GEOQUIMICA	81	255	19	306	32	30	1	NM
MATEMATICA	101	293	13	379	84	76	0	NM
QUIMICA	134	683	286	673	81	66	0	NM
Totales	811	3395	723	3571	618	491	3	-

CONEST Versión 3.0.0 Todos los Derechos Reservados. Grupo CONEST. Copyright 2007-2014

Figura 3.2.55: Interfaz para visualizar las licenciaturas

La interfaz de la Planificación Docente posee variantes en los datos y las acciones disponibles de acuerdo al rol de cada usuario:

- Personal Administrativo de DCE

La vista del personal administrativo dispone de tres (3) filtros, las opciones de gestión y visualización sobre la planificación sin restricción por el tipo de asignatura, además, incluye las opciones de los Docentes, al igual que el horario de la planificación.

Un aspecto significativo de este TEG es brindar mayor protagonismo al Jefe de Departamento en el sistema, ya que, en el recae el mayor flujo de información y toma de decisiones sobre la planificación. El Jefe de Departamento puede cargar un conjunto de asignaturas en el sistema a excepción de las asignaturas obligatorias, pasantías, seminarios, tesis y métodos.

- Docentes

Las opciones para el usuario con rol de administrador están limitadas a la asignación del grupo docente, la visualizar y calificar una sección, el listado de las secciones (PDF y Excel), modificar al coordinador, la visualización del horario en la planificación y de una asignatura en particular. Además dispone de los tres (3) filtros.

- Estudiantes

Para los usuarios con rol de estudiante solo pueden visualizar la Planificación Docente, es decir, ver las asignaturas y algunos datos de la misma, de igual manera pueden visualizar secciones y docentes principales con los respectivos. El estudiante dispone de dos (2) filtros para la planificación y también puede visualizar el horario de la misma. Es importante destacar que el estudiante puede acceder a la Planificación Docente antes y durante el proceso de inscripción, de manera que permite reducir el número de solicitudes estudiantiles.

3.3. Caso Estudio

El 28 de Abril de 2014, se lanzó la nueva versión CONEST 3.0 para el período académico 02-2013, estando disponible para la comunidad universitaria de la Facultad de Ciencias, siendo posible visualizar el resultado de este TEG. El módulo de Planificación Docente fue puesto en producción sin tener culminadas algunas funcionalidades, de manera que durante un lapso de dos (2) a tres (3) semanas el sistema estuvo sujeto a prueba, modificación y optimización de los procesos.

Antes de la apertura del período académico 01-2014, se planteó al Jefe de Departamento de la Escuela de Computación que integrara las asignaturas de tipo electiva al sistema,

utilizando las funcionalidades de gestión de la Planificación Docente. Este proceso se llevó a cabo de manera satisfactoria, permitiendo ser un ejemplo para el resto de las escuelas y atenuando la participación de las mismas en períodos futuros. Esta participación, es muestra clara de que se puede lograr el flujo de información propuesto en el diagrama 3.2.5 y evitar las dependencias que tiene la DCE con las escuelas. En la figura 3.3.1 se puede observar algunos datos asociados a la Planificación Docente.

Oferta Académica							
Periodo	Nuevos	Regulares	Preinscritos	Retiros	Secciones	Nómina	
02-2014	0	0	0	0	255		
01-2014	103	3060	824	928	374		
02-2013	811	3396	723	3573	345		
01-2013	98	3127	956	3092	345		

Oferta Académica							
Oferta Académica > Período: 01-2014							
Licenciatura	Nuevos	Regulares	Preinscritos	Retiros	Secciones	Calificadas	Nómina
BIOLOGIA	24	542	90	171	91	83	NM
COMPUTACION	24	1045	256	270	152	78	NM
FISICA	14	350	118	114	70	51	NM
GEOQUIMICA	8	235	18	84	27	26	NM
MATEMATICA	13	244	16	84	58	49	NM
QUIMICA	20	644	326	205	75	59	NM
Totales	103	3060	824	928	473	346	-

Figura 3.3.1: Datos de la Planificación para el período 01-2014

Actualmente, se está utilizando la solución propuesta para la integración de las asignaturas electivas en el sistema para el período académico 02-2014. Con la diferencia de la participación de todas las escuelas en el proceso, además, se efectuaron cambios significativos en el sistema para la apertura de un nuevo período académico, ya que, la solución de calcular el número de secciones y la capacidad de la asignatura en base al promedio de dos semestres anterior con la misma característica de semestre “par o impar”, es más sencilla, esto se debe a que el comportamiento de la planificación es poco variable, además no se cargaban los horarios correspondientes. Por este motivo se crearon plantillas de los horarios y las asignaturas.

Para finalizar se presentan las conclusiones, recomendaciones, referencias bibliográficas consultadas en esta investigación.

Conclusiones y recomendaciones

Conclusión

El objetivo de este Trabajo Especial de Grado, consistió en automatizar los procesos asociados a la Planificación Docente, con la finalidad de generar un impacto positivo en los tiempos de respuesta y reducir la brecha de comunicación entre las distintas escuelas y la DCE. Este objetivo se logró con éxito, debido a que se consiguió desarrollar las funcionalidades propuestas para la planificación, las cuales están relacionadas con: la apertura de un período académico, la administración de los recursos, tales como asignaturas, secciones, grupo docente, horarios y aulas, asimismo, la configuración de las asignaturas, la generación de reportes y documentos de interés, el resumen de horarios por semestre y el estado de la calificación por medio de indicadores. Todas estas funcionalidades forman parte de la nueva versión del sistema CONEST y son evidenciadas mediante la ejecución del sistema, accediendo a través de las opciones correspondientes.

La Planificación Docente es un proceso complejo, razón por la cual involucró realizar un análisis exhaustivo de la situación, que permitiera comprender al detalle la problemática y las consecuencias derivadas. Las actividades realizadas para el conocimiento y resolución del problema estuvieron asociadas con las discusiones, en las cuales el equipo de desarrollo se involucró permitiendo la comprensión amplia desde el punto de vista del usuario. También, fue importante el análisis de la antigua versión del sistema para identificar las fallas del mismo, en relación al proceso de planificación siendo fundamental en fase de desarrollo.

El alcance de los objetivos propuestos se debe a las bases del marco conceptual y a la aplicación del método de desarrollo AgilUs. Este último demostró ser efectivo y de gran ayuda en el desarrollo de las funcionalidades y en el diseño de interfaces de usuario. Esta afirmación está basada en los resultados obtenidos en las pruebas de aceptación. Este método proporcionó un conjunto de técnicas y artefactos que permitieron la retroalimentación y evaluación del sistema, facilitando la corrección rápida y efectiva. Adicionalmente, integró al usuario final como parte del equipo de desarrollo ofreciendo un producto usable que responde a sus necesidades. Sin embargo, se considera que el método ofrece múltiples artefactos para el levantamiento de información y la evaluación del sistema, pero no para la lógica de implementación de los requerimientos.

En el inicio del proceso de desarrollo se consideró como una dificultad realizar las actividades de implementación, al mismo tiempo que era necesario el aprendizaje del lenguaje Ruby y la lógica del framework Ruby on Rails, debido a que solo se contemplaban conocimientos básicos del mismo. Sin embargo, el proceso de adaptación fue rápido y no causó impedimentos en el logro de los objetivos. En líneas generales el framework fue de gran ayuda en la organización del código de la aplicación y en el desarrollo, por estar orientado a la arquitectura Modelo, Vista y Controlador.

Sobre las tecnologías del lado del cliente no hubo inconvenientes en cuanto al uso, fueron apropiadas y facilitaron el desarrollo de las funcionalidades para el logro de los objetivos. En este sentido, la librería de jQuery y jQuery UI ayudaron significativamente en el desarrollo de los métodos de las ventanas modales, la autocomplementación de los campos y los calendarios para los formularios. La librería Tablesorter facilitó el ordenamiento de tablas dinámicamente. Además, la técnica AJAX se utilizó de manera significativa para lograr cambios sobre la interfaz sin necesidad de recargarla. Del mismo modo las herramientas Asana y Zoho Projects facilitaron el seguimiento del sistema, así como, la comunicación con el equipo de desarrollo.

Después de un largo proceso de desarrollo, el producto fue puesto en producción y a prueba con la Escuela de Computación para el semestre 01-2014, actualmente está siendo utilizada para la planificación del próximo semestre, con la participación del resto de las escuelas. Esto permitió demostrar que se puede cumplir el flujo de trabajo con un rol más protagónico por parte del jefe de departamento en el sistema, reduciendo la brecha de comunicación entre las dependencias de las escuelas y la DCE.

Por otra parte, este aporte ofrece al estudiantado y a la DCE, reducir el número de solicitudes estudiantiles asociados con el cambio de horario, ya que, los alumnos pueden disponer de la Planificación Docente con anterioridad al proceso de inscripción. Otro hecho que se suma es visualizar la planificación con tiempo para la toma de decisiones y la auditoría de datos.

Finalmente, la autora destaca su participación con el equipo de trabajo de CONEST como exitosa, gratificante, de enseñanza profesional y personal, así como la satisfacción de dejar un proyecto en producción y a la disposición de la comunidad universitaria adquiriendo la posibilidad de seguir creciendo.

Recomendaciones

Entre las características de cualquier aplicación se encuentra que debe ser escalable y sostenible en el tiempo, en este sentido, si surge la necesidad de agregar un nuevo requerimiento de usuario a la solución desarrollada, debe permitir incluir la nueva funcionalidad.

Bajo esta perspectiva se recomienda para trabajos futuros, estandarizar el documento de la Planificación Docente proveniente de las escuelas, de manera que el usuario pueda cargar al sistema el archivo en formato Excel y crear automáticamente la planificación con las asignaturas correspondientes. Esto es posible, ya que, el lenguaje Ruby dispone de gemas que permiten leer un documento en Excel.

Asimismo, se recomienda automatizar las actividades de coordinación entre los docentes y jefes de departamento de cada escuela, a fin de reducir los esfuerzos por localizar a un docente. Al igual que debe permitir conocer las propuestas de los docentes para dictar una asignatura, reduciendo los tiempos de respuesta.

Por otra parte se deben efectuar actividades de mantenimientos correctivos, preventivos y adaptativos a la aplicación.

Este trabajo, puede ser una guía o modelo para desarrollar un sistema que automatice el proceso de Planificación Docente a nivel de Postgrado.

Referencias

- [1] Acosta, A., (2011, Septiembre). AgilUs: Construcción ágil de la Usabilidad. pp 4-8.
- [2] Asana (n.d). Disponible en <https://asana.com/>. [Consultada: 2013, Octubre 28]
- [3] Boyer, Y. Mendez, N., (2008, Mayo). Automatización de procesos relacionados con las solicitudes estudiantiles y actividades administrativas y de docencia de la Facultad de Ciencias, T.E.G.
- [4] CoffeScript. Disponible en: <http://coffeescript.org/#overview>. [Consultada: 2013, Octubre 23]
- [5] Escalante, S. Ovalles, G., (2010, Octubre). Desarrollo de los Módulos de Constancias y de Comprobantes para el Sistema CONEST Postgrado, T.E.G.
- [6] ESTRUCTURA ACADÉMICA (n.d). Disponible en: www.ciens.ucv.ve
- [7] González, Z. Modelo de Requisitos Centro ISYS Escuela de Computación Facultad de Ciencias UCV, s.f. Disponible en: Modelo requisitos. [Consultada: 2014, Julio 28]
- [8] Guías de Estilo. Villa, L. (2004, Mayo). Desarrollo Web. Disponible en: [guia de estilo](#) . [Consultada: 2014, Agosto 14]
- [9] Haml (HTML Abstracción Markup Language). Disponible en: <http://haml.info/>. [Consultada: 2013, Octubre 23]
- [10] Manual de referencia de MySQL 5.0, s.f. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/features.html>. [Consultada: 2013, Junio 19]
- [11] Nielsen, J. (1995, January). 10 Heurísticas para el diseño de interfaz de usuarios. Disponible en: <http://www.nngroup.com/articles/ten-usability-heuristics/> [Consultada: 2014, Septiembre 22]
- [12] Primary Roles on an Agile UP Project (2006, Mayo). Disponible en: <http://www.ambysoft.com/unifiedprocess/aup11/html/roles.html>

- [13] Programación LAMP Agile., (2013, Enero). (n.d). Disponible en: <http://rolandojaldin.blogspot.com/2010/10/introduccion-la-metodologia-rup-proceso.html>
- [14] Prototipo en Papel. Villa, L. (s.f). Web Taller. Disponible en: prototipo papel. [Consultada: 2014, Agosto 14]
- [15] RailsGuides. Ruby on Rails Guides (v4.0.0). Disponible en: <http://guides.rubyonrails.org/>. [Consultada: 2013, Junio 19].
- [16] Rivero, K. y De Castro, G., (2007) Desarrollo de una Aplicación Web para la División de Control de Estudios que automatice la Asignación de Horarios y Aulas en el proceso de inscripción de los alumnos de Pregrado de la Facultad de Ciencias. T.E.G.
- [17] Sinemed (n.d) ¿Qué es MySQL?. Disponible en: www.sinemed.com. [Consultada: 2013, Junio 19]
- [18] Soporte. (2013, 9 de Julio). Wikipedia, La enciclopedia libre. Disponible en Wiki soporte. [Consultada: 2014, Agosto 28]
- [19] Soporte. Desde Definicion ABC. Disponible en: [soporte abc](http://soporte.abc.com). [Consultada: 2014, Agosto 28]
- [20] w3schools. HTML5 Introduction. Disponible en: [/www.w3schools.com](http://www.w3schools.com). [Consultada: 2013, Junio 19]
- [21] Zoho Projects. (n.d). Disponible en: <https://www.zoho.com/projects/> [Consultada: 2013, Octubre 28]