PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

ESCUELA DE SISTEMAS

DISERTACIÓN DE GRADO PREVIA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS Y COMPUTACIÓN

TEMA:

DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL MÓDULO DE MANEJO DE TRANSPORTES Y RUTAS PARA EL SISTEMA DE GESTIÓN ACADÉMICA DE LA EMPRESA "RUTATEC", CON INTEGRACIÓN DE HERRAMIENTAS DE GEOLOCALIZACIÓN PARA LA LECTURA DE DATOS DEL GPS.

AUTOR:

ALEX PATRICIO CHÁVEZ CHÉRREZ

QUITO – 2013

Tabla de contenido

Índice de Ilustraciones 4

1. CAPÍTULO 1 5

1.1 QUÉ ES RUTATEC 5

1.1.1 PRODUCTOS Y SERVICIOS DE RUTATEC 6

1.2 QUÉ ES EL SISTEMA DE GESTIÓN ACADÉMICA (RUTADEMIC) 7

1.2.1 CARACTERÍSTICAS IMPORTANTES DE RUTADEMIC: 7

1.2.2 MÓDULOS DE RUTADEMIC 8

1.2.3 OPCIONES DE INSTALACIÓN DE RUTADEMIC 25

1.2.4 REQUERIMIENTOS DE INSTALACIÓN DE RUTADEMIC 25

1.3 JUSTIFICACIÓN 25

1.4 DEFINICIÓN DEL MÓDULO DE TRANSPORTES 26

1.4.1 QUÉ ES GEOLOCALIZACIÓN 26

1.5 HERRAMIENTA DE DESARROLLO 27

1.6 DEFINICIÓN DE METODOLOGÍA 28

1.6.1 QUÉ ES SCRUM 28

1.7 ARQUITECTURA DE LA SOLUCIÓN 31

2. CAPÍTULO 2 34

2.1 DOCUMENTO DE REQUERIMIENTOS 34

2.1.1 Introducción 34

2.1.2 Propósito 34

2.1.3 Alcance 34

2.1.4 Definiciones, acrónimos y abreviaturas 34

2.1.5 Referencias 34

2.1.6 Visión General 35

2.1.7 Descripción General 35

2.1.7.1 Perspectiva del Producto 35

Interfaces del sistema 35

Interfaces de usuario 35

2.1.8 Requerimientos específicos 35

2.1.8.1 Funcionales 35

2.1.8.2 Detalle de Requerimientos Funcionales 36

2.1.8.3 Especificación de casos de uso 39

2.1.8.4 Requerimientos no funcionales 39

Rendimiento 39

Seguridad 39
Fiabilidad 39
Disponibilidad 39
Mantenibilidad 39
Portabilidad 39
2.1.9 Otros requisitos 40
2.2 MODELADO DE CASOS DE USO 41
2.2.1 Caso de Uso General Simplificado 41
2.2.2 Descripción de Casos de Uso 42
2.2.2.1 Caso de Uso CUS-01 42
2.2.2.2 Caso de Uso CUS-02 42
2.2.2.3 Caso de Uso CUS-03 44
2.2.2.4 Caso de Uso CUS-04 45
2.2.2.5 Caso de Uso CUS-05 46
2.2.2.6 Caso de Uso CUS-06 47
2.2.2.7 Caso de Uso CUS-07 48
2.2.2.8 Caso de Uso CUS-08 49
2.2.2.9 Caso de Uso CUS-09 50
2.2.2.10 Caso de Uso CUS-10 50
2.2.2.11 Caso de Uso CUS-11 51
2.4 MODELADO DE BASE DE DATOS 55
3. CAPÍTULO 3 58
3.1 IMPLEMENTACIÓN CATÁLOGO DE CONDUCTORES 59
3.2 IMPLEMENTACIÓN DEL REPORTE DE RUTAS BASADO EN DATOS DEL GPS. 69
3.3 PRUEBAS UNITARIAS PANTALLA INGRESO NUEVO CONDUCTOR 88
3.4 PRUEBAS UNITARIAS REPORTE DE RUTAS BASADO EN DATOS DEL GPS 91
4. CAPÍTULO 4 95
4.1 CONCLUSIONES 95
4.2 RECOMENDACIONES 97
4.3 BIBLIOGRAFÍA 98

Índice de Ilustraciones

Título: Ilustración 1 8
Título: Ilustración 2 9
Título: Ilustración 3 10
Título: Ilustración 4 11
Título: Ilustración 5 12
Título: Ilustración 6 13
Título: Ilustración 7 14
Título: Ilustración 8 15
Título: Ilustración 9 16
Título: Ilustración 10 17
Título: Ilustración 11 18
Título: Ilustración 12 19
Título: Ilustración 13 20
Título: Ilustración 14 21
Título: Ilustración 15 22
Título: Ilustración 16 23
Título: Ilustración 17 24
Título: Ilustración 18 29
Título: Ilustración 19 31
Título: Ilustración 20 33
Título: Ilustración 21 41
Título: Ilustración 22 42
Título: Ilustración 23 42
Título: Ilustración 24 44
Título: Ilustración 25 45
Título: Ilustración 26 46
Título: Ilustración 27 47
Título: Ilustración 28 48
Título: Ilustración 29 49
Título: Ilustración 30 50
Título: Ilustración 31 50
Título: Ilustración 32 51
Título: Ilustración 33 55
Título: Ilustración 34 56
Título: Ilustración 35 57
Título: Ilustración 36 53
Título: Ilustración 37 89
Título: Ilustración 38 90
Título: Ilustración 39 91
Título: Ilustración 40 92
Título: Ilustración 41 92
Título: Ilustración 42 93
Título: Ilustración 43 93
Título: Ilustración 44 94

1. CAPÍTULO 1

En este capítulo se explica qué es RUTATEC como empresa y hablaremos sobre su pilar fundamental que es el SISTEMA DE GESTIÓN ACADÉMICA. Además, se dará una explicación sobre el funcionamiento del módulo de TRANSPORTES el cuál será el desarrollo de la presente disertación. Hablaremos también sobre las herramientas que se van a utilizar en el desarrollo y de la metodología que se va a seguir.

1.1 QUÉ ES RUTATEC

"La idea de RUTATEC empezó siendo un proyecto de tesis de la Facultad de Sistemas de la Universidad Católica del Ecuador en el año 2006. El tema propuesto fue la creación de un Sistema Informático de Administración de Procesos basado en un requerimiento existente en un colegio de la ciudad de Quito. Una vez entregada la tesis y graduados los estudiantes, el colegio analizó el proyecto y se mostró muy satisfecho con los resultados, lo que abrió una gran oportunidad a los estudiantes.

En ese tiempo el colegio experimentaba problemas con el proveedor del servicio del sistema de Gestión Escolar, así que propuso a los ingenieros que desarrollen un sistema a la medida desde cero para la Gestión Escolar del colegio.

El Sistema de Gestión Escolar fue desarrollado originalmente con tecnología Microsoft: SQL Server[1], Visual Studio[2], la principal razón para utilizar esas herramientas fue que la Institución Educativa tenía firmado un contrato de "School Agreement"[3]. El desarrollo de la primera versión del Sistema de Gestión Escolar llevó aproximadamente 8 meses con 3 desarrolladores.

A partir de ese momento RUTATEC empezó a funcionar como una empresa de hecho que daba servicios especializado de desarrollo de software a medida e implantación de portales web colaborativos (tanto para colaboración interna como externa a la empresa) utilizando herramientas Microsoft como Visual Studio, SQL Server, SharePoint[4], etc., además del servicio entregado con el Sistema de Gestión Escolar.

En el año 2008 RUTATEC se unió a La Corporación MachángaraSoft que es un grupo empresarial conformado varias empresas especialistas y líderes en distintas líneas de negocio tecnológico. Especializándose y liderando la línea de tecnología aplicada a instituciones educativas.

Actualmente brinda productos y servicios tecnológicos, se caracteriza por ofrecer una mezcla única de innovación y personalización a sus clientes. Con seis años de experiencia en el

[1] **Microsoft SQL Server** es un sistema para la gestión de bases de datos producido por Microsoft basado en el modelo relacional.

[2] **Microsoft Visual Studio** es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, y Visual Basic .NET, al igual que entornos de desarrollo web.

[3] **School Agreement** es un programa integral de Licenciamiento por Volumen de Microsoft basado en suscripción que ofrece una forma flexible y redituable para que los clientes de educación puedan licenciar la última tecnología de Microsoft.

[4] **Microsoft SharePoint Products and Technologies**, es una colección de productos y elementos de software que incluye, funciones de colaboración, basado en el Explorador web, módulos de administración de proceso, módulos de búsqueda y una plataforma de administración de documentos.

mercado Ecuatoriano, la empresa busca afianzar su crecimiento a través de la aplicación de estrategias de mercadeo efectivas.

Su producto estrella es el Sistema de Gestión Escolar para instituciones educativas llamado "RUTADEMIC", el cual es personalizado para cada cliente acorde a sus requerimientos." **Maldonado, Juan Pablo, (2013) Diseño y Desarrollo de un Plan de Marketing para el Sistema de Gestión Escolar "RUTATDEMIC" Desarrollado por RUTATEC, Tesis Maestría, Universidad Católica del Ecuador.**

1.1.1 PRODUCTOS Y SERVICIOS DE RUTATEC

I. Sistema de Gestión Escolar (RUTADEMIC)

Sistema que sirve para facilitar la administración y manejo de una institución educativa. Permite optimizar procesos permitiendo tener una institución más organizada, eficiente y que brinda un mejor servicio a todos los miembros de la comunidad educativa.

II. Educación en línea

La incorporación de nuevas tecnologías de la información aplicadas a la formación es un hecho que abre nuevas vías en el ámbito educativo. La educación virtual puede ser utilizada como complemento a la educación tradicional o como método de aprendizaje alternativo ya sea para capacitación de personal como estudiantes en general. En RUTATEC se implementan plataformas basadas en Moodle[5] ya que se cuenta con personal certificado en esta plataforma.

III. Desarrollo a medida

Las organizaciones tienen requerimientos muy específicos que no pueden ser encontrados en aplicaciones o sistemas existentes. RUTATEC ofrece el servicio de desarrollo de software a medida, garantizando un desarrollo de primer nivel que cumpla con las expectativas de nuestros clientes locales e internacionales

IV. Asesoría y consultoría:

a. Adquisición de hardware y software para instituciones educativas.
b. Optimización de procesos de gestión de instituciones educativas.
c. Optimización de recursos tecnológicos.
d. Utilización de tecnología en procesos de enseñanza.
e. Capacitación.

V. Gerencia de proyectos tecnológicos.

VI. Instalación y configuración de Infraestructura tecnológica.

[5] **Moodle** es una aplicación web de tipo Ambiente Educativo Virtual, un sistema de gestión de cursos, de distribución libre, que ayuda a los educadores a crear comunidades de aprendizaje en línea.

1.2 QUÉ ES EL SISTEMA DE GESTIÓN ACADÉMICA (RUTADEMIC)

El Sistema de Gestión Académica RUTADEMIC es una herramienta muy útil que facilita el manejo de centros educativos, mejorando la eficiencia de sus áreas de operación (Gestión Académica, Disciplina, Asistencia, Matriculación, Pensiones, Facturación, Manejo de Usuarios y Comunicación.

Permite optimizar y mejorar los procesos de gestión, ahorrando tiempo y recursos, brindando un excelente servicio tanto a profesores y personal administrativo como a padres de familia y estudiantes, es decir a toda la comunidad educativa de la institución.

1.2.1 CARACTERÍSTICAS IMPORTANTES DE RUTADEMIC:

1. **Acceso web:** El sistema puede instalarse en la institución educativa o en un servidor remoto en Internet. Todos los usuarios pueden acceder al sistema desde cualquier navegador de internet sin importar el dispositivo con el cual naveguen.
2. **Parametrizable:** Se puede adaptar a la estructura que maneja el establecimiento educativo, la institución no debe adaptarse al sistema, el sistema se configura según la estructura de la institución educativa.
3. **Fácil:** Muy fácil de aprender, su uso es muy intuitivo.
4. **Efectivo:** Facilita el trabajo y ahorra tiempo de docentes, personal administrativo, estudiantes y representantes.
5. **Seguro:** Administración de permisos a través de perfiles de usuarios, completamente compatible para trabajar con SSL[6] .
6. **Completo:** Los estudiantes y representantes pueden revisar notas, actividades pendientes, horarios, lista de compañeros, etc. desde el colegio y desde internet.
7. Reducción de tiempo de matriculación.
8. **Módulo de comunicación** muy completo que permite el intercambio de información constante entre todos los miembros de la comunidad educativa.
9. **Impresión de reportes** requeridos por el Ministerio de Educación (Régimen Escolar) y reportes internos para la institución educativa.
10. **Multiplataforma:** El sistema puede ser instalado sobre cualquier plataforma (Linux, Windows, Mac) lo único que se necesita es un servidor web como XAMPP[7].

[6] **Secure Sockets Layer** (**SSL)** En español «capa de conexión segura», SSL proporciona autenticación y privacidad de la información entre extremos sobre Internet mediante el uso de criptografía. Habitualmente, sólo el servidor es autenticado (es decir, se garantiza su identidad) mientras que el cliente se mantiene sin autenticar.

[7] **XAMPP** es un servidor independiente de plataforma, software libre, que consiste principalmente en la base de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script: PHP y Perl. El nombre proviene del acrónimo de **X** (para cualquiera de los diferentes sistemas operativos), **A**pache, **M**ySQL, **P**HP, **P**erl.

1.2.2 MÓDULOS DE RUTADEMIC

El sistema de gestión académica RUTADEMIC está compuesto por los siguientes módulos:

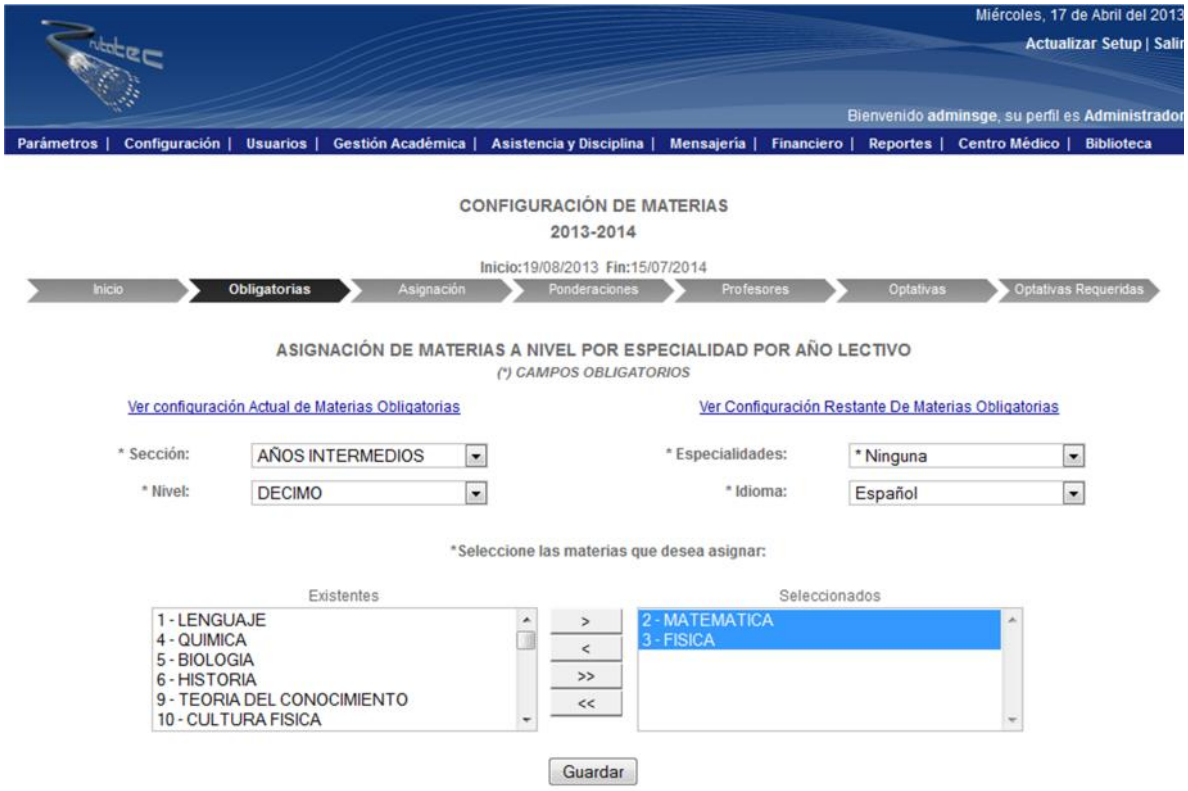
1. **Módulo de configuración.**

   En este módulo se definen y registran las reglas y políticas de funcionamiento de la institución educativa en el sistema para cada año lectivo: secciones, cursos, paralelos, rangos de supletorios, países, provincias, colores, logo de la institución, etc.

   Nos permite definir las materias para cada nivel, definir la disponibilidad de cada profesor, asignar el profesor que dictará cada materia en cada curso, definir criterios de evaluación, dirigentes de curso, aulas, materias optativas, equivalencia de calificaciones cualitativas, equivalencia entre asignaturas del colegio y del Ministerio, la forma en que se calculará el promedio académico y de disciplina, etc.



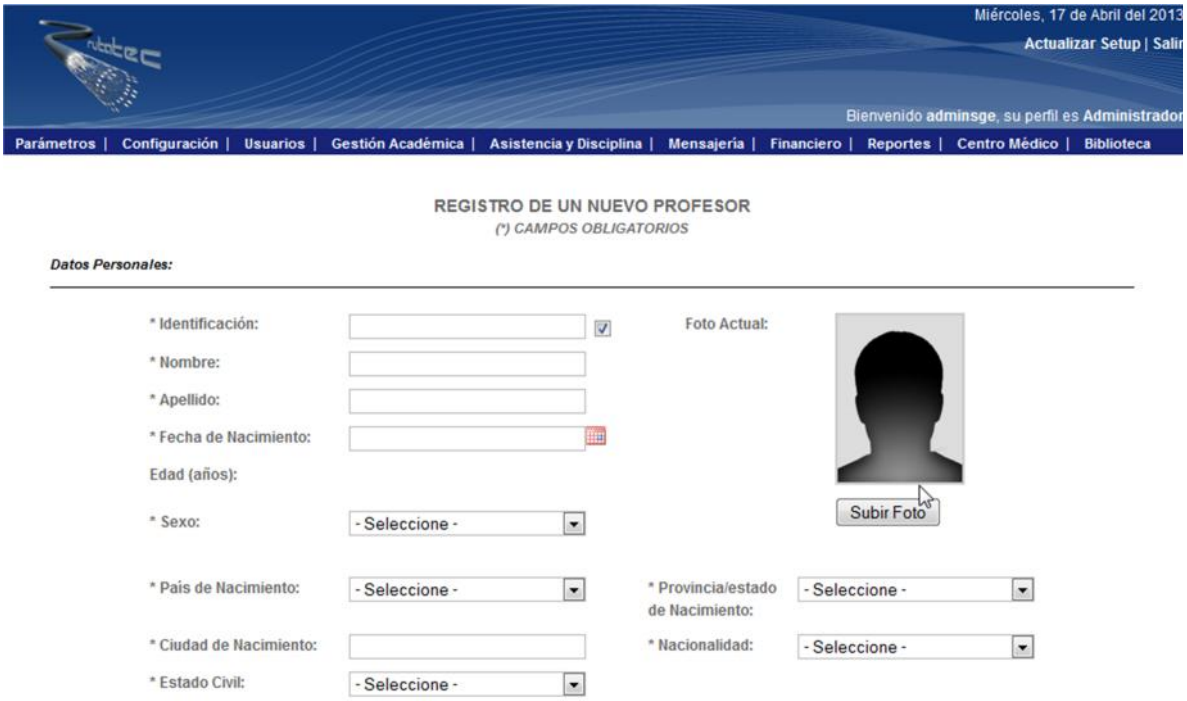
Título: Ilustración 1
Descripción: Captura de Pantalla de la interfaz de Configuración de Año Lectivo RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez



Título: Ilustración 2
Descripción: Captura de Pantalla de la interfaz de Configuración de Materias RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

2. Módulo de administración de usuarios

Este módulo permite crear y editar usuarios del sistema, asignar el perfil correspondiente, los permisos que tendrá cada usuario dentro del sistema. Para el caso de profesores, los datos solicitados son todos los que pide el Ministerio de Educación más otros adicionales, tales como las cargas familiares que tiene, la fecha de ingreso al magisterio, etc.



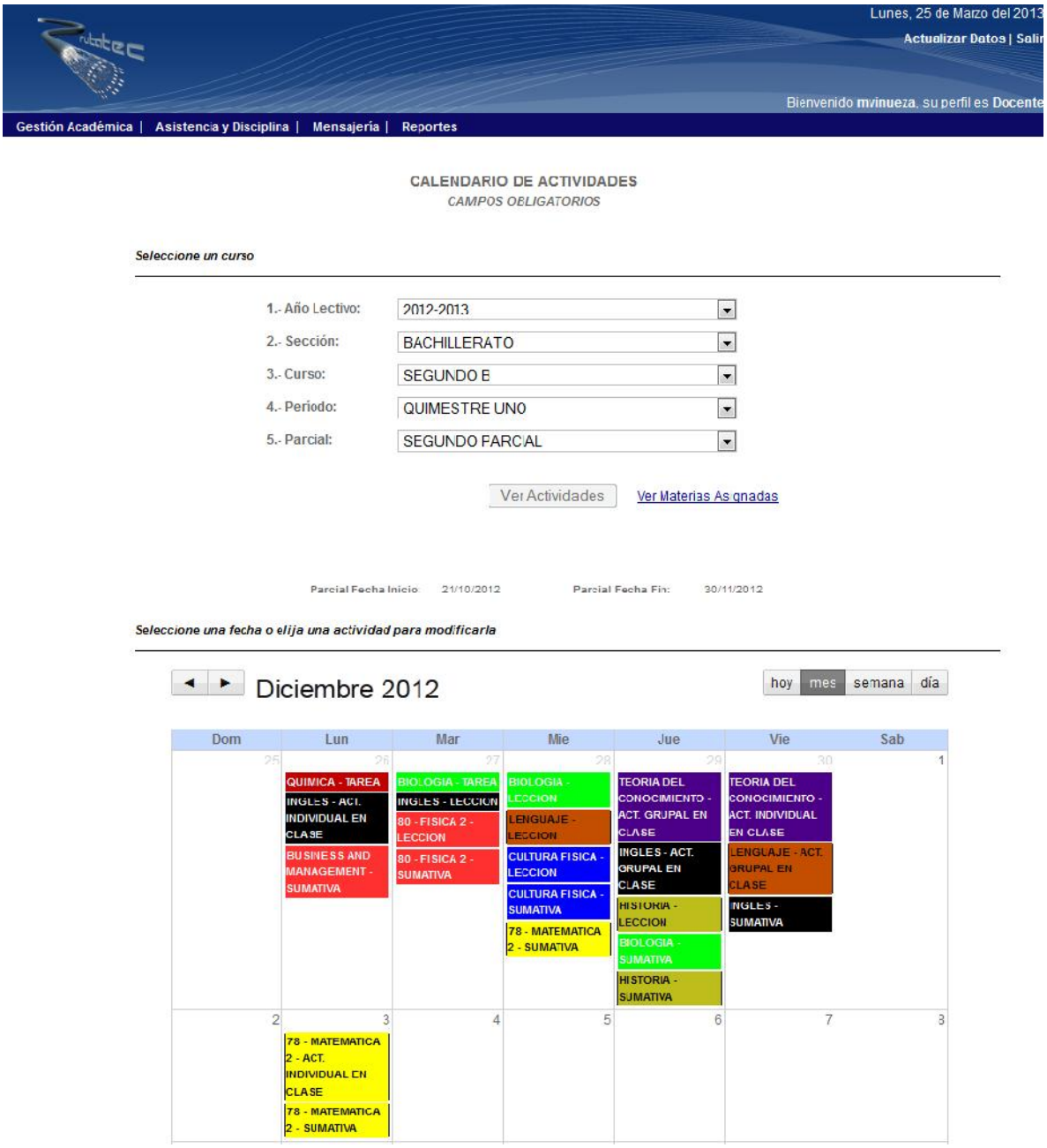
Título: Ilustración 3
Descripción: Captura de Pantalla de la interfaz de Ingreso de Usuarios RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

3. Módulo de profesores.

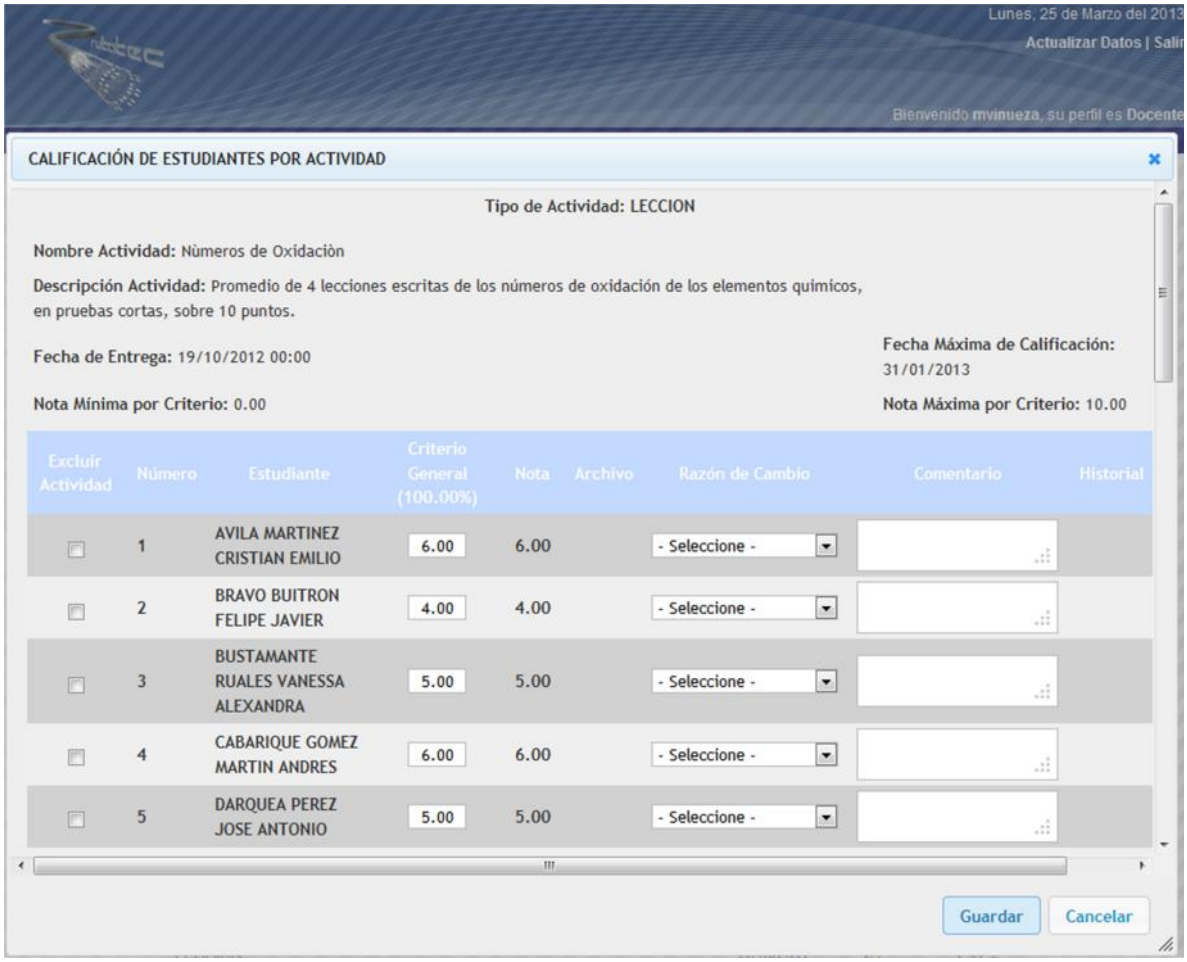
Con este módulo los profesores de la institución podrán crear actividades para sus estudiantes, definir el tipo de actividad y el puntaje que tendrá sobre el promedio de la asignatura, cargar documentos relacionados a cada actividad, definir la fecha de entrega y luego ingresar la calificación correspondiente.

El docente también tiene la opción de cargar el plan de estudios de cada asignatura y acceder al centro de mensajes.

Los dirigentes de curso o directores departamentales tendrán acceso especial para poder analizar las actividades, calificaciones e información adicional de todo lo que les competa.



Título: Ilustración 4
Descripción: Captura de Pantalla de la interfaz de Ingreso de Actividades RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

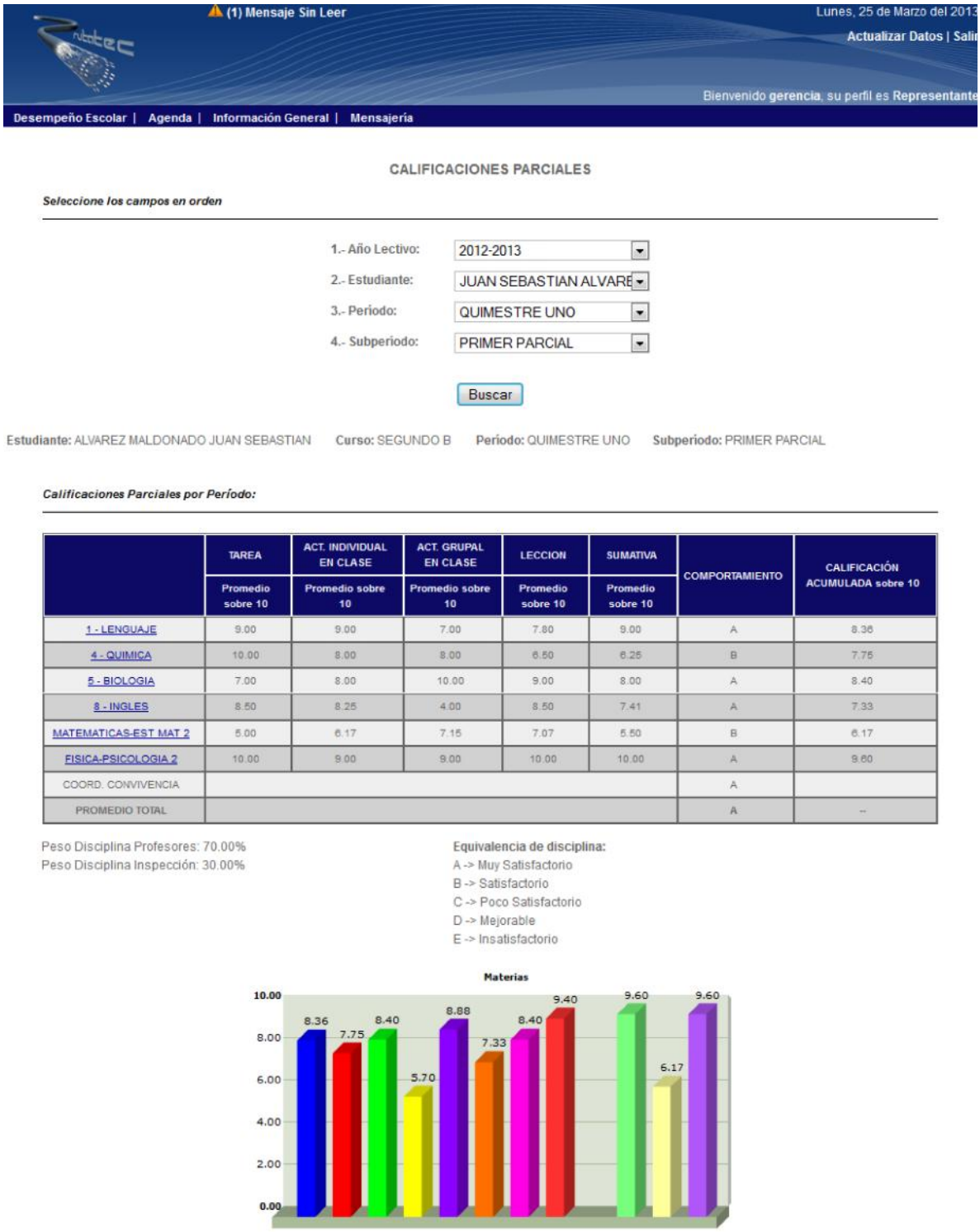


Título: Ilustración 5
Descripción: Captura de Pantalla de la interfaz de Ingreso de Calificaciones RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

4. Módulo de estudiantes.

Los estudiantes pueden acceder al sistema para consultar las actividades pendientes, su historial de calificaciones, lista de compañeros, revisar comunicados de la institución, enviar y recibir mensajes de otros usuarios (dependiendo de las políticas de cada institución), horario de clases, asistencia, disciplina, transporte, etc.

Adicionalmente los estudiantes pueden registrar eventos o actividades propias que serán desplegados en el calendario junto con las actividades enviadas por los profesores.



Título: Ilustración 6
Descripción: Captura de Pantalla de la interfaz de Detalle de Calificaciones RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

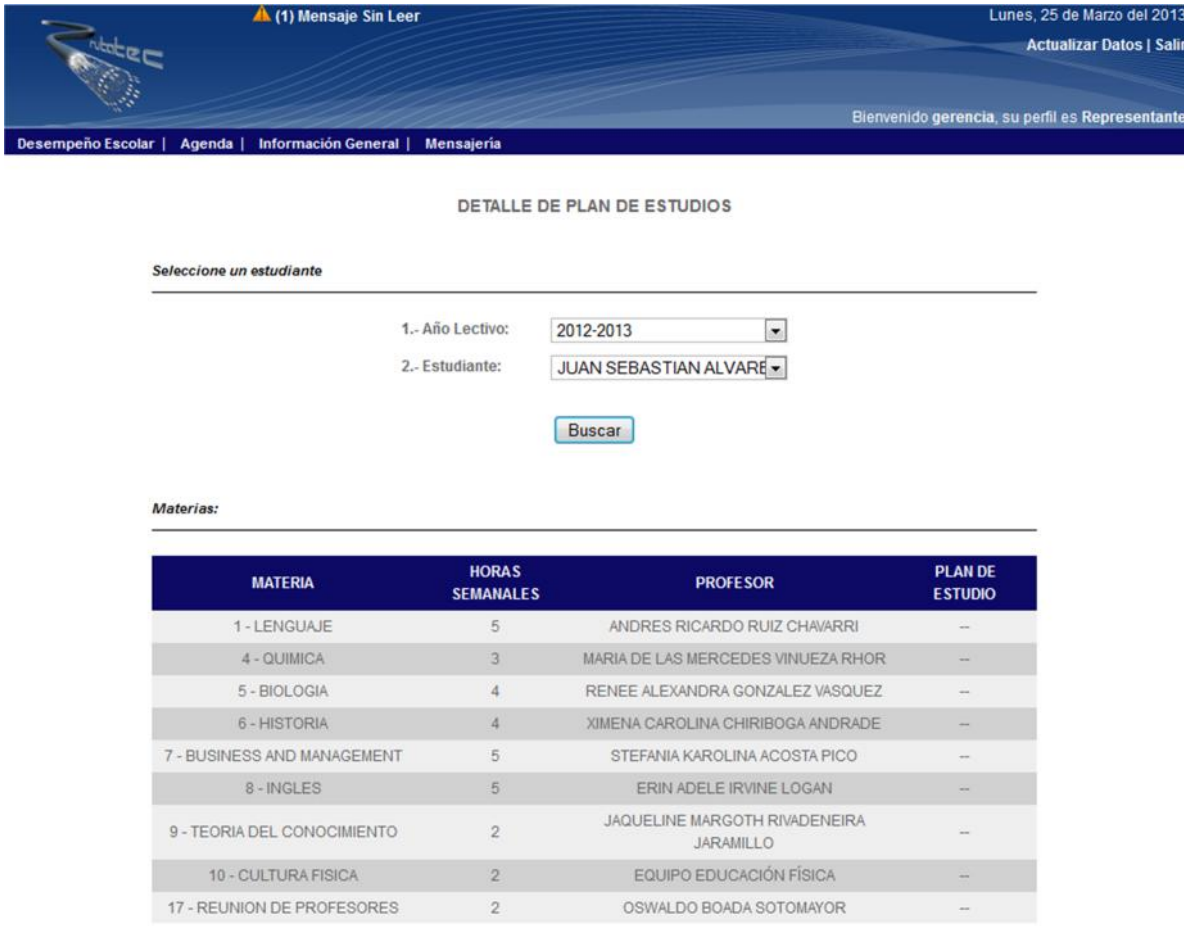
5. Módulo de representantes.

Los representantes tienen acceso a todas las opciones a las que accede cada estudiante al que representa.

Adicionalmente tiene acceso a su estado de cuenta. Al inicio del año tienen la opción actualizar los datos personales de cada representante, actualizar los datos personales y ficha médica de los estudiantes a quién representa y de seleccionar los rubros para cada uno de los estudiantes a quienes representa.



Título: Ilustración 7
Descripción: Captura de Pantalla de la interfaz de Detalle de Actividades RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez



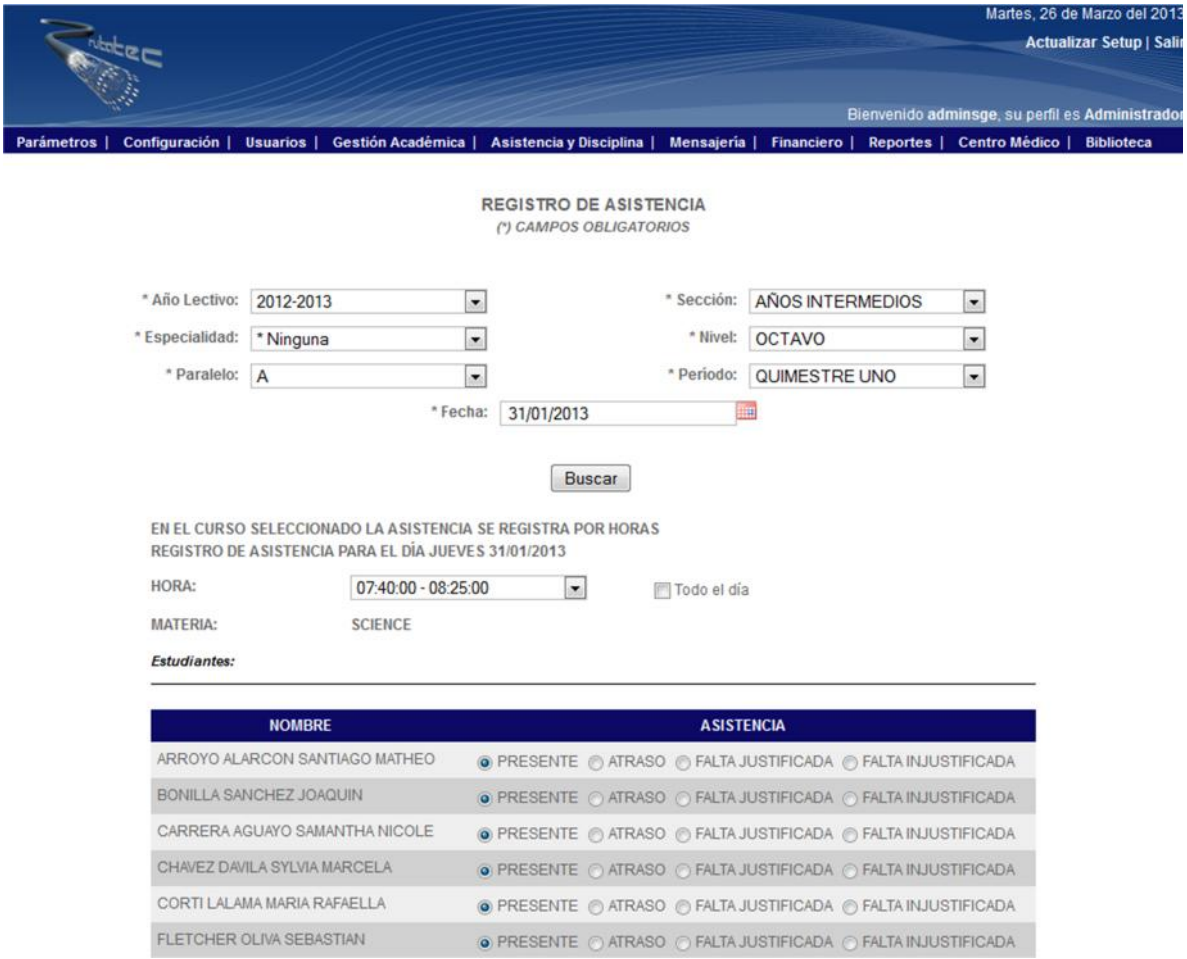
Título: Ilustración 8
Descripción: Captura de Pantalla de la interfaz de Detalle de Plan de Estudios RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

6. Módulo de admisiones.

Una vez abierto el periodo de admisiones, el sistema permite a usuarios internos del sistema (representantes) pre-inscribir a nuevos estudiantes y de igual manera permite que personas externas a la institución apliquen para ser aceptados en la misma (todo esto a través de internet). Esto activa automáticamente el proceso de aprobación y selección de los estudiantes que serán admitidos.

7. Módulo de asistencia y disciplina.

El sistema permite registrar la asistencia por cada hora de clase o por día, la disciplina puede ser registrada por cada profesor o por un departamento que se encargue de ese tema, además dentro de las configuraciones iniciales se permite definir quién registra disciplina y cómo se calcula el promedio por cada nivel; esto puede ser que solo califiquen los profesores y/o el inspector y/o maestro tutor.



Título: Ilustración 9
Descripción: Captura de Pantalla de la interfaz Registro de Asistencia RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez



Título: Ilustración 10
Descripción: Captura de Pantalla de la interfaz de Registro de Asistencia RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

8. Módulo de mantenimiento (bodega).

Se tiene un almacén con materiales para dar mantenimiento físico a la escuela, uso pedagógico o para uso en general, se registra el inventario de cada artículo, incluyendo la categoría Ej. Materiales de limpieza, materiales didácticos, etc., el destino de uso Ej. Área de inglés, Departamento de matemáticas, mantenimiento de aulas, etc., el proveedor, el costo unitario, la persona que solicitó la compra de los materiales, la salida de bodega, la persona responsable, etc.

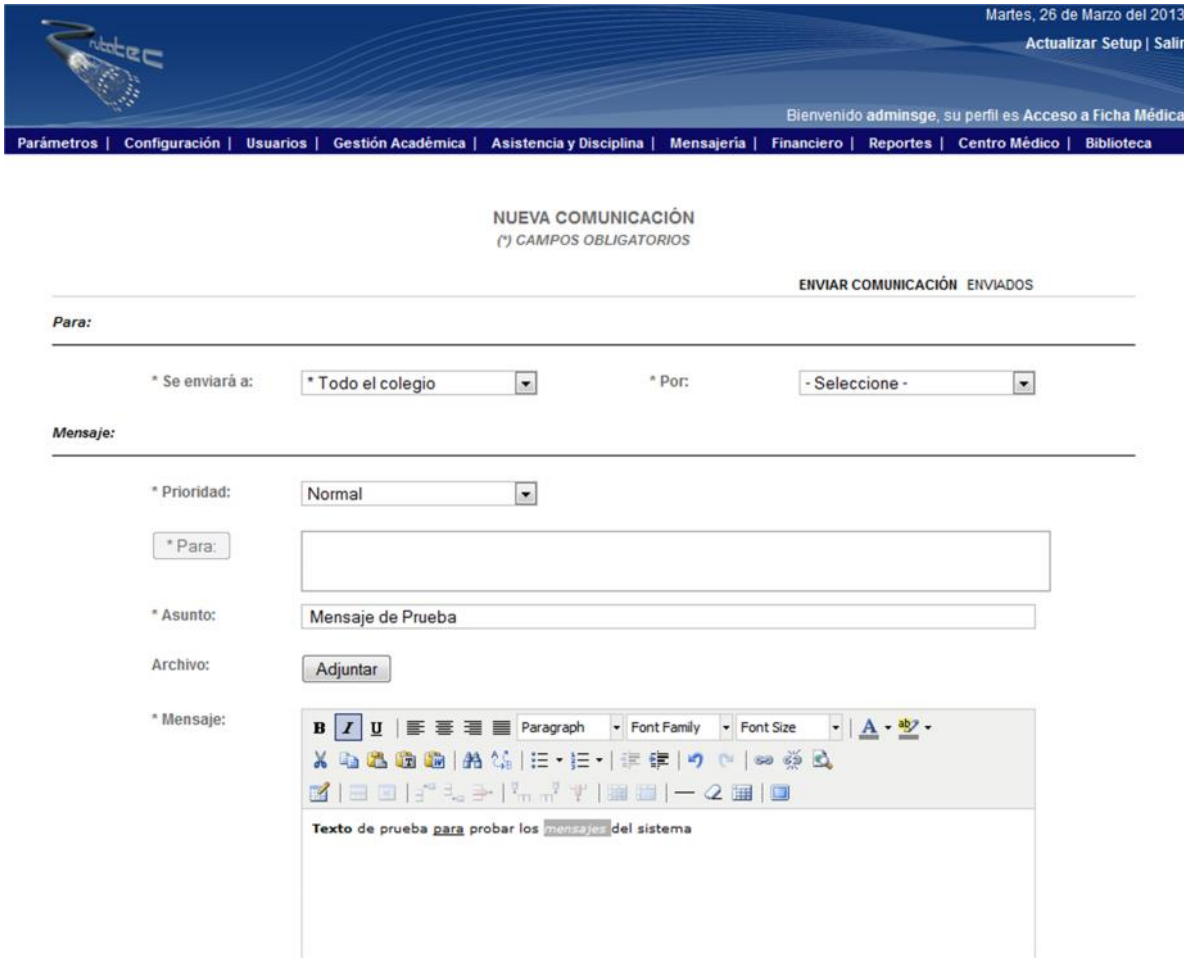
9. Módulo de comunicación y mensajería.

La institución educativa puede registrar eventos en un calendario general de actividades el cual podrá ser leído por cualquier usuario del sistema.

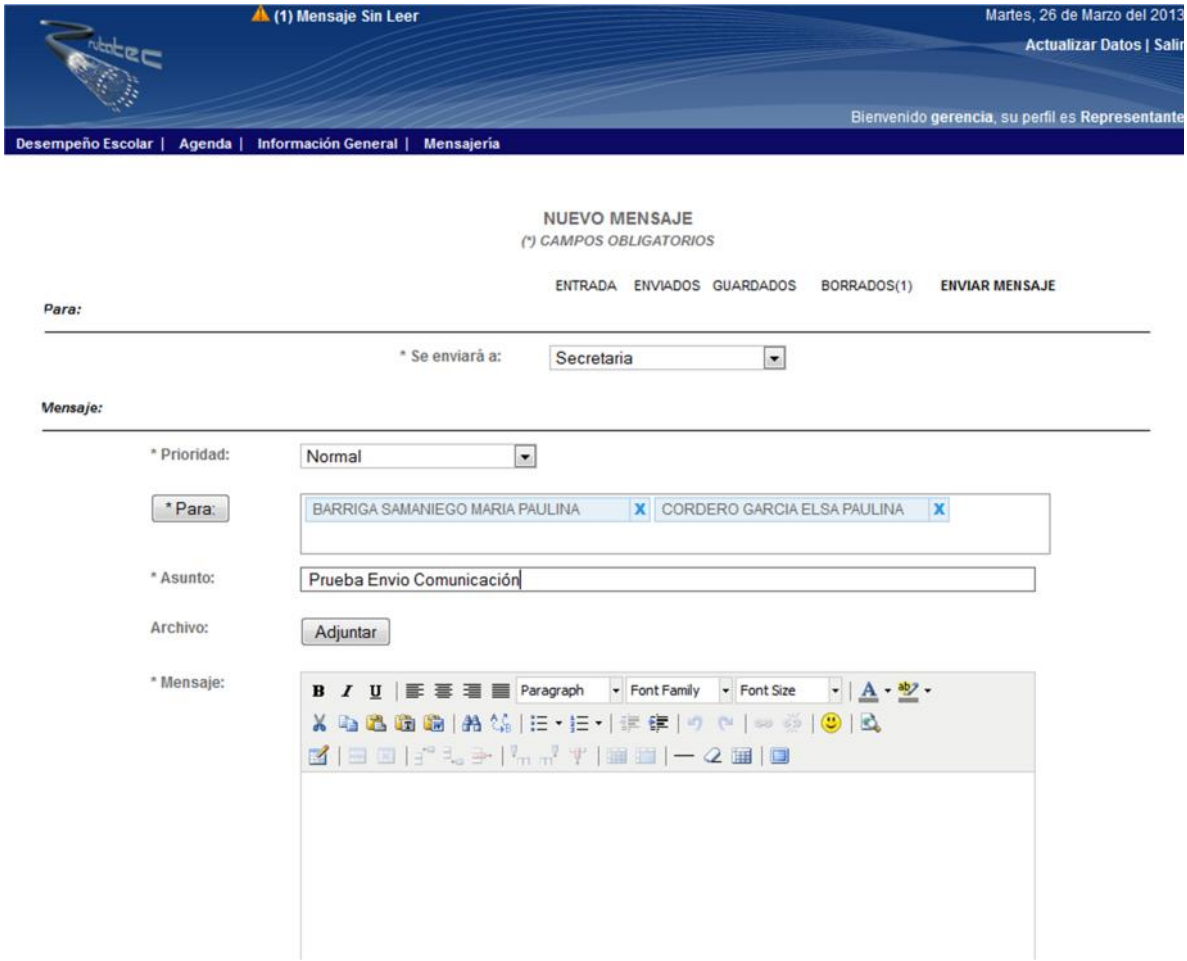
Mediante el centro de mensajes, la institución podrá enviar comunicados globales, para cierto grupo de usuarios o mensajes personales. Toda comunicación entre alumno y profesor o coordinador es replicada al representante del estudiante. Un profesor puede enviar mensajes a su coordinador, alumnos o padres de alumnos.

Cada institución tiene la posibilidad de definir quién puede comunicarse con quién de acuerdo a las políticas de cada organización.

Si los usuarios tienen configurada una dirección de correo electrónico cada mensaje será enviado al sistema para que lo puedan visualizar cada vez que ingresen al mismo y una copia al correo electrónico configurado.



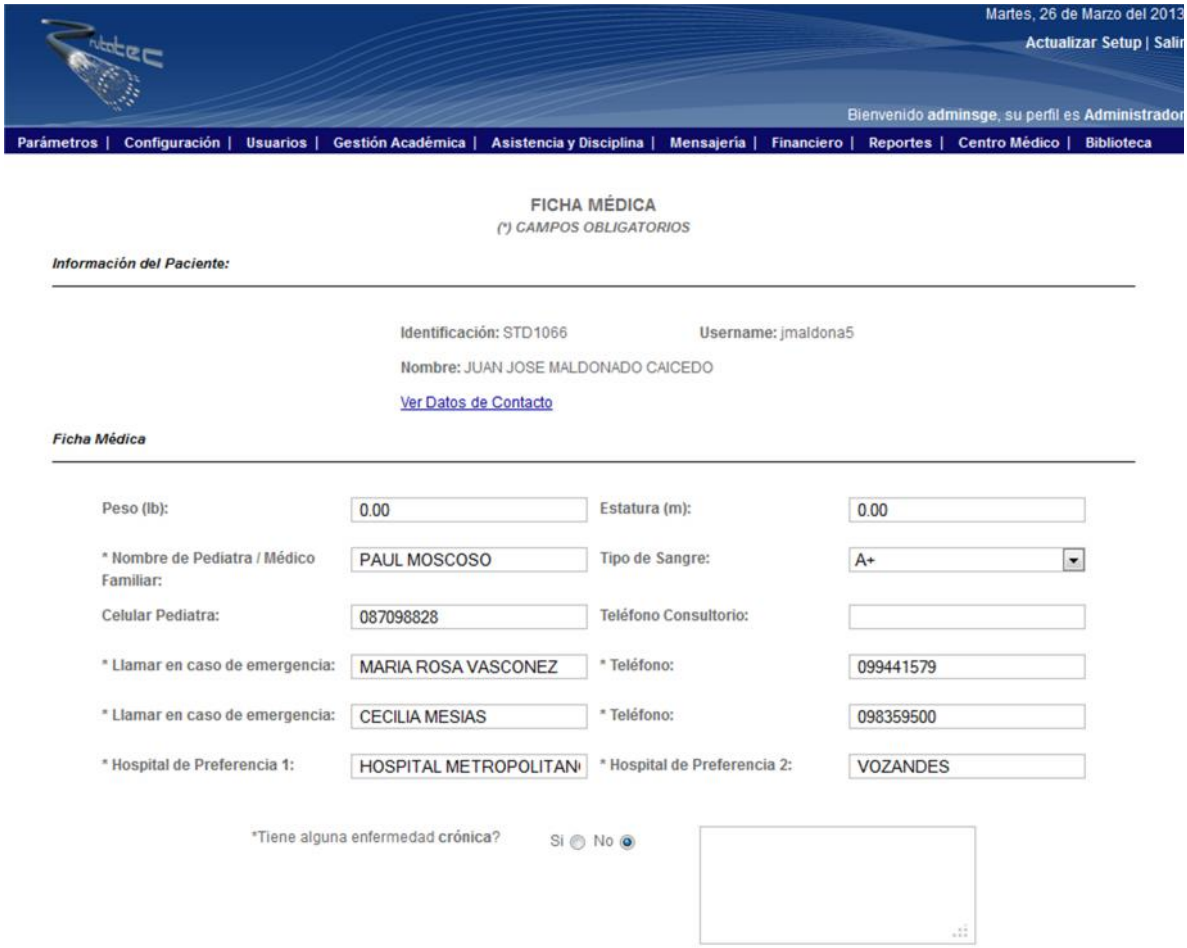
Título: Ilustración 11
Descripción: Captura de Pantalla de la interfaz Envío de Comunicados RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez



Título: Ilustración 12
Descripción: Captura de Pantalla de la interfaz de Envío de Mensajes RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

10. Módulo de departamento médico.

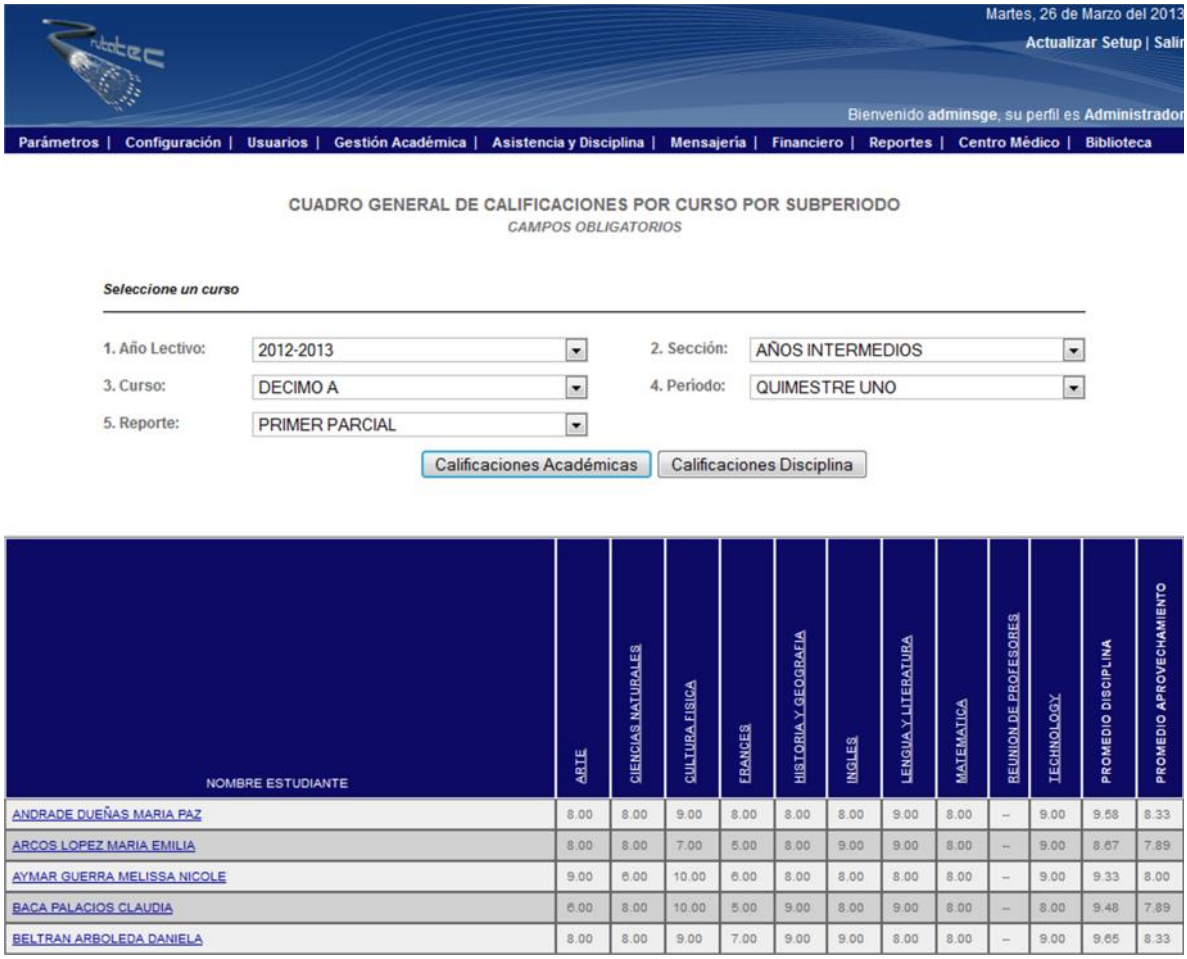
El sistema permite registrar información médica muy importante de los usuarios del sistema por ejemplo peso, talla, alergias, enfermedades (importantes) y centros médicos de preferencia. Adicionalmente permite llevar una ficha médica por cada uno de los usuarios donde se registra toda la información correspondiente a las visitas al departamento médico.



Título: Ilustración 13
Descripción: Captura de Pantalla de la interfaz de Ficha Médica RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

11. Módulo de reportes.

El sistema permite imprimir todos los reportes solicitados por el Ministerio de Educación del Ecuador y muchos reportes internos tales como lista de alumnos, reportes de caja, lista de representantes, reportes de calificaciones, usuarios por bus, asistencia, cuadros de evaluación, calificaciones por materia, actividades enviadas por profesor, etc.

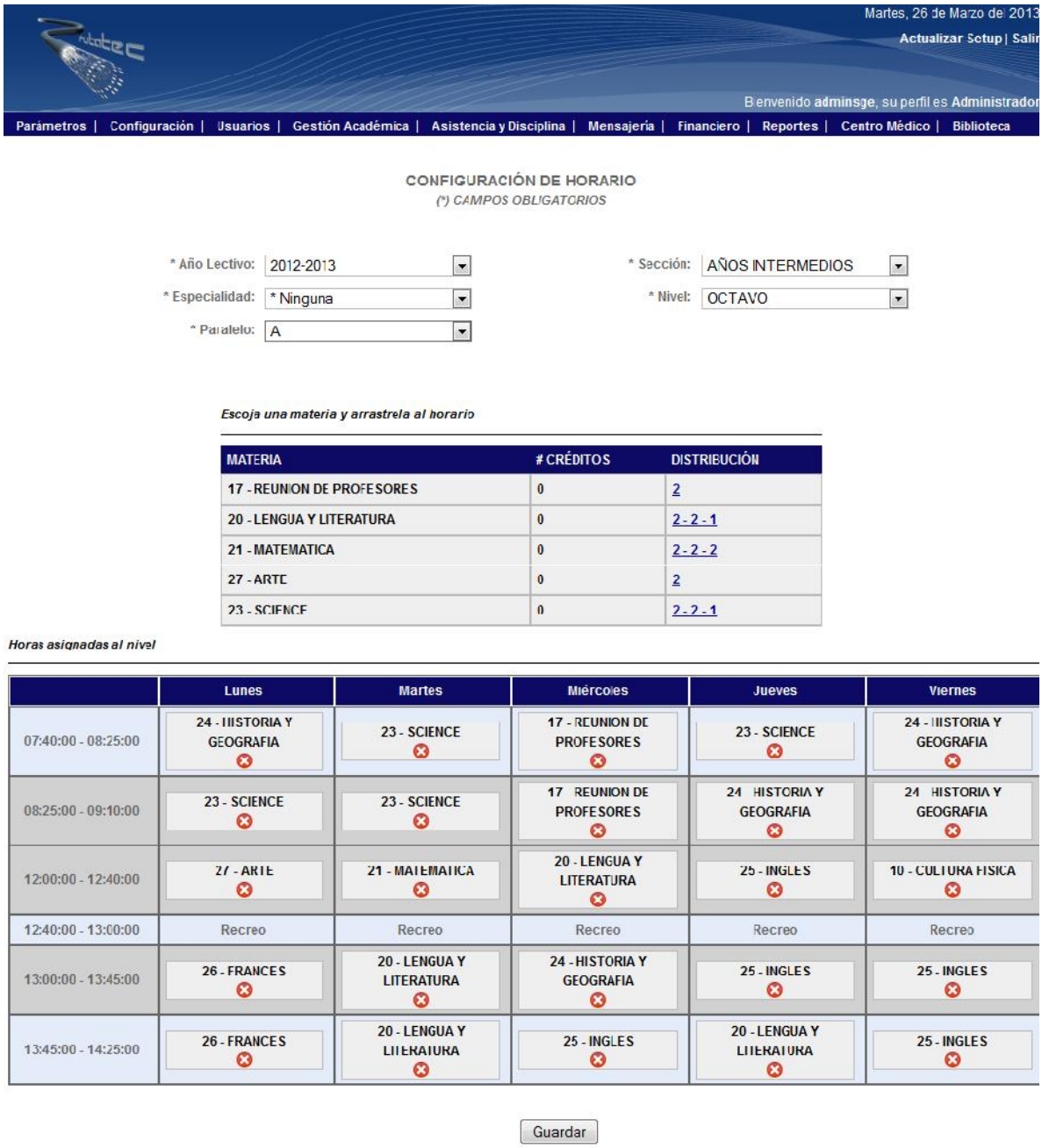


NOMBRE ESTUDIANTE	ARTE	CIENCIAS NATURALES	CULTURA FISICA	FRANCES	HISTORIA Y GEOGRAFIA	INGLES	LENGUA Y LITERATURA	MATEMATICA	REUNION DE PROFESORES	TECHNOLOGY	PROMEDIO DISCIPLINA	PROMEDIO APROVECHAMIENTO
ANDRADE DUEÑAS MARIA PAZ	8.00	8.00	9.00	8.00	8.00	8.00	9.00	8.00	--	9.00	9.58	8.33
ARCOS LOPEZ MARIA EMILIA	8.00	8.00	7.00	5.00	8.00	9.00	9.00	8.00	--	9.00	8.67	7.89
AYMAR GUERRA MELISSA NICOLE	9.00	6.00	10.00	6.00	8.00	8.00	8.00	8.00	--	9.00	9.33	8.00
BACA PALACIOS CLAUDIA	6.00	8.00	10.00	5.00	9.00	8.00	9.00	8.00	--	8.00	9.48	7.89
BELTRAN ARBOLEDA DANIELA	8.00	8.00	9.00	7.00	9.00	9.00	8.00	8.00	--	9.00	9.65	8.33

Título: Ilustración 14
Descripción: Captura de Pantalla de la interfaz Reporte de Calificaciones RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

12. Módulo de Horarios.

El sistema permite registrar horarios de clase de acuerdo a la carga horaria de cada materia para cada curso y la distribución (horas de clase por día) de cada una.



Título: Ilustración 15
Descripción: Captura de Pantalla de la interfaz de Configuración de Horarios RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

13. Módulo de Biblioteca

Capacidad de controlar préstamos de libros y equipos (entradas y salidas de ítems).



Título: Ilustración 16
Descripción: Captura de Pantalla de la interfaz de Préstamo de Libros RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

14. Módulo facturación y cuentas por cobrar.

El módulo de facturación permite la creación de conceptos de pagos (rubros) opcionales u obligatorios, mensuales o anuales, asignación de rubros a niveles, asignación de rubros a estudiantes, impresión de facturas, avisos y reportes de morosidad, venta de productos y manejo de inventario de artículos existentes para la venta.



Título: Ilustración 17
Descripción: Captura de Pantalla de la interfaz de Cobros de Rubros RUTADEMIC
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

1.2.3 OPCIONES DE INSTALACIÓN DE RUTADEMIC

Se cuenta con dos opciones de instalación:

- Instalar el sistema en un servidor propiedad de la institución.
- Instalar el sistema en un servidor externo, contratando servicio de hosting[8].

1.2.4 REQUERIMIENTOS DE INSTALACIÓN DE RUTADEMIC

1. Requerimientos de conectividad

- IP Pública (si se quiere tener acceso al sistema desde Internet)
- Ancho de banda necesaria para tener un servidor web y de aplicaciones dentro de las instalaciones del colegio (esto depende de la cantidad de usuarios que existan en cada institución, y es necesario solamente si se quiere tener acceso al sistema desde Internet).

2. Requerimientos de Software

- Sistema Operativo: Windows Server[9] o Linux.

3. Requerimientos de Hardware

- Procesador C2D (Core 2 Duo) 2,8 GHz o Superior.
- Memoria RAM 3 GB.
- Disco Duro 80 GB.
- Para instituciones educativas de gran tamaño (1.000 estudiantes o más) es recomendable instalar al menos 4 GB de memoria RAM.
- Las características recomendadas podrán variar de acuerdo al número de alumnos de cada institución.

1.3 JUSTIFICACIÓN

El presente trabajo de disertación de grado ayudará a complementar la funcionalidad del sistema de gestión académica RUTADEMIC por medio de la implementación de un módulo de transporte con aplicación de herramientas de geolocalización para así poder gestionar todo lo relacionado con el servicio de transporte dentro de las instituciones educativas en las que funciona el sistema.

[8] El **alojamiento web** (en inglés *web hosting*) es el servicio que provee a los usuarios de Internet un sistema para poder almacenar información, imágenes, vídeo, o cualquier contenido accesible vía web.

[9] **Windows Server** es una marca que abarca una línea de productos servidor de Microsoft Corporation. El nombre se ha utilizado en las versiones de Microsoft Windows para servidores.

1.4 DEFINICIÓN DEL MÓDULO DE TRANSPORTES

El módulo de transportes que se desea implementar, maneja en manera de catálogos[10] los diferentes componentes del mismo, éstos son:

Rutas, Turnos, Buses y Conductores.

Además tiene una interfaz para la configuración de recorridos haciendo uso de los diferentes elementos ingresados anteriormente en los catálogos; y cuenta también con una interfaz para hacer la respectiva asignación a los usuarios de estos recorridos. Todo esto en una interfaz intuitiva y amigable para el usuario final; basándose en los diseños proporcionados por la empresa RUTATEC, ya que estos diseños están elaborados con los requerimientos previamente levantados con los clientes finales.

Además debe contar con una pantalla a manera de reporte en la cual los padres de familia puedan ver los datos generados por el respectivo GPS de los buses, y mostrar estos datos con herramientas de geolocalización.

1.4.1 QUÉ ES GEOLOCALIZACIÓN

Geolocalización es un concepto relativamente nuevo, que se ha popularizado en los últimos años también es denominada como georeferenciación, y se refiere al posicionamiento que define la localización de un objeto en un sistema de coordenadas determinado.

Este proceso era empleado exclusivamente por los sistemas de información geográfica (SIG[11]) para obtener la ubicación de algún cuerpo en un sistema de coordenadas; existen varias alternativas para conocer esta ubicación, una de la más usada es el GPS[12], que es un dispositivo que gracias a su conexión con un satélite en la órbita terrestre nos permite saber nuestra ubicación en coordenadas geográficas con una precisión de pocos metros de variación. Además ahora con el boom de los smartphones[13] podemos encontrar nuestra ubicación de una manera más rápida y sencilla por medio de la red celular en combinación de la red de los GPS.

---

[10] Se conoce como **catálogo** al conjunto de interfaces que manejan la inserción en base de datos de varios elementos o componentes que nos sirven como datos iniciales para hacer uso en cualquier otra pantalla de configuración.

[11] Un **Sistema de Información Geográfica** (**SIG** o **GIS**, en su acrónimo inglés Geographic Information System) es una integración organizada de *hardware*, *software* y datos geográficos diseñada para capturar, almacenar, manipular, analizar y desplegar en todas sus formas la información geográficamente referenciada con el fin de **resolver problemas complejos de planificación y gestión geográfica**.

[12] El **SPG** o **GPS** (**Global Positioning System**: **sistema de posicionamiento global**) o **NAVSTAR-GPS** es un sistema global de navegación por satélite (GNSS) que permite determinar en todo el mundo la posición de un objeto, una persona o un vehículo con una precisión hasta de centímetros (si se utiliza GPS diferencial), aunque lo habitual son unos pocos metros de precisión.

[13] Un **teléfono inteligente**, también denominado **teléfono táctil** (*smartphone* en inglés) es un teléfono móvil construido sobre una plataforma informática móvil, con una mayor capacidad de almacenar datos y realizar actividades semejantes a una mini computadora y conectividad que un teléfono móvil convencional.

Otra alternativa ampliamente usada y a disposición de aquellos que no poseen celulares como los antes mencionados u otros equipos portátiles, es la del Google Earth[14], que consiste en un programa informático, similar al Sistema de Información Geográfica, que permite obtener imágenes del planeta en tecnología 3D en combinación con imágenes de satélite, mapas y el motor de búsqueda de Google y así facilita la visualización de imágenes a escala del lugar del planeta tierra que se pretenda descubrir o identificar.

Gracias a herramientas de uso masificado como las antes mencionadas, la Geolocalización se ha transformado en una herramienta usada a diario por millones de personas alrededor de todo el mundo, posiblemente sin saberlo, y ya no está limitada para los especialistas de las geociencias y los sistemas de información geográfica (SIG); tanto se ha popularizado su uso que se ha convertido en un fenómeno social, por ejemplo las redes sociales se han conectado con sistemas de geolocalización para que sus usuarios puedan saber la ubicación de las personas con las cuales interactúan, y además nos permite compartir esta información con quién nosotros deseemos.

Otra de las ramas en las cuales se ha popularizado la geolocalización es en el marketing; en base a esto ha nacido el geomarketing que es la ciencia de ofertar productos y publicitarlos en las regiones donde sea más óptima su demanda; esto se ha obtenido gracias a las páginas web 2.0[15] que usan herramientas de geolocalización y así pueden determinar la ubicación de la persona que la están usando y basándose en esto las empresas nos promocionan o anuncias productos en nuestra región o país.

1.5 HERRAMIENTA DE DESARROLLO

Al ser un módulo para un sistema ya en funcionamiento se debe manejar con las mismas herramientas y tecnologías que el Sistema de Gestión Académica RUTADEMIC.

Las herramientas de desarrollo son:

Smarty 2.0[16]

PHP 5.3[17]

Javascript 1.3.2[18]

---

[14] **Google Earth** es un programa informático similar a un sistema de información geográfica (SIG), creado por la empresa Keyhole Inc., que permite visualizar imágenes del planeta, combinando imágenes de satélite, mapas y el motor de búsqueda de Google que permite ver imágenes a escala de un lugar específico del planeta.

[15] El término **Web 2.0** comprende aquellos sitios web que facilitan el compartir información, la interoperabilidad, el diseño centrado en el usuario y la colaboración en la World Wide Web.

[16] **Smarty** es un motor de plantillas para PHP, es decir, separa el código PHP, como lógica de negocios, del código HTML, como lógica de presentación, y genera contenidos web mediante la colocación de *etiquetas Smarty* en un documento.

[17] **PHP** es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos.

18 **JavaScript** es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Mysql 5.0[19]

Además se utilizará como IDE[20] Netbeans 7.2.1[21] con soporte para PHP y para manejar Subversionamiento[22] se usará tortoiseSVN 1.7.11[23].

1.6 DEFINICIÓN DE METODOLOGÍA

La metodología a utilizar para el desarrollo va a ser SCRUM, esto debido al tamaño del módulo a desarrollar.

1.6.1 QUÉ ES SCRUM

Scrum es un marco de trabajo para la gestión y desarrollo de software basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software.

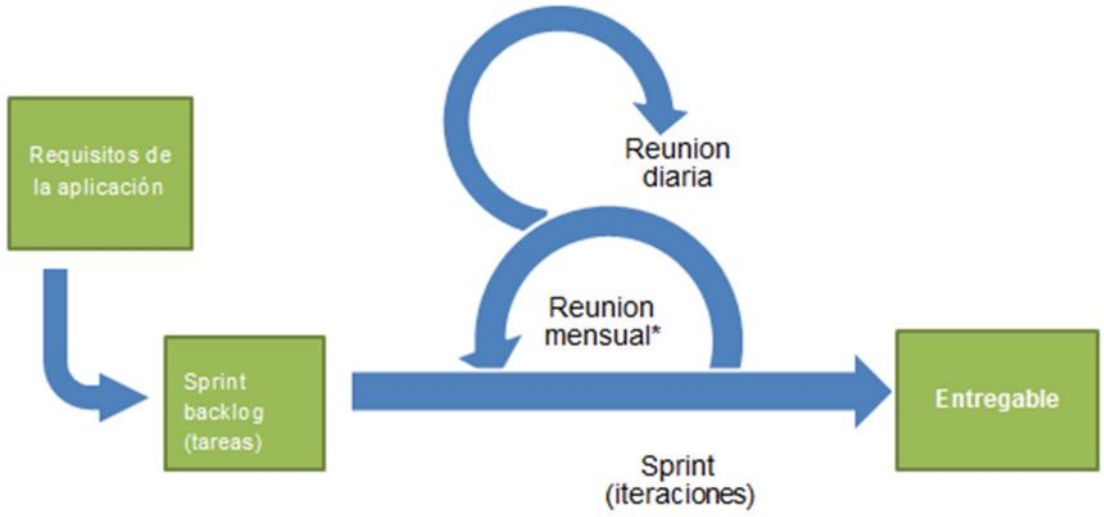
SCRUM controla de forma empírica y adaptable la evolución del proyecto, empleando las siguientes prácticas de la gestión ágil:

Revisión de las Iteraciones
Al finalizar cada iteración (normalmente 30 días) se lleva a cabo una revisión con todas las personas implicadas en el proyecto. Este es el período máximo que se tarda en reconducir una desviación en el proyecto o en las circunstancias del producto

Desarrollo incremental
Durante el proyecto, las personas implicadas no trabajan con diseños o abstracciones. El desarrollo incremental implica que al final de cada iteración se dispone de una parte del producto operativa que se puede inspeccionar y evaluar.

Desarrollo evolutivo
Los modelos de gestión ágil se emplean para trabajar en entornos de incertidumbre e inestabilidad de requisitos. Intentar predecir en las fases iniciales cómo será el producto final, y sobre dicha predicción desarrollar el diseño y la arquitectura del producto no es realista, porque las circunstancias obligarán a remodelarlo muchas veces.

[19] **MySQL** es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones.
[20] Un **entorno de desarrollo integrado**, llamado también **IDE** (sigla en inglés de *integrated development environment*), es un programa informático compuesto por un conjunto de herramientas de programación.
[21] **NetBeans** es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE es un producto libre y gratuito sin restricciones de uso.
[22] **Revision Control System** o **RCS** es una implementación en software del control de versiones que automatiza las tareas de guardar, recuperar, registrar, identificar y mezclar versiones de archivos.
[23] **TortoiseSVN** es un cliente Subversion, implementado como una extensión al shell de Windows. Es software libre liberado bajo la licencia GNU GPL.



Título: Ilustración 18
Descripción: Diagrama explicativo del proceso SCRUM
Fecha: Enero 2013
Fuente: Realizado por Maxie Ayala- Wikipedia

Los elementos que conforman el desarrollo SCRUM son:

Las reuniones

- **Planificación de sprint:** Jornada de trabajo previa al inicio de cada sprint en la que se determina cuál va a ser el trabajo y los objetivos que se deben cumplir en esa iteración.
- **Reunión diaria:** Breve revisión del equipo del trabajo realizado hasta la fecha y la previsión para el día siguiente.
- **Revisión de sprint:** Análisis y revisión del incremento generado.

Los elementos

- **Pila del producto:** lista de requisitos de usuario que se origina con la visión inicial del producto y va creciendo y evolucionando durante el desarrollo.
- **Pila del sprint:** Lista de los trabajos que debe realizar el equipo durante el sprint para generar el incremento previsto.
- **Incremento:** Resultado de cada sprint del modelo SCRUM

Los roles

SCRUM clasifica a todas las personas que intervienen o tienen interés en el desarrollo del proyecto en: propietario del producto, equipo, gestor de SCRUM (también SCRUM Manager o SCRUM Master) y “otros interesados”.
Los tres primeros grupos (propietario, equipo y gestor) son los responsables del proyecto, los que según la comparación siguiente serían los “involucrados”; mientras que el resto de interesados serían los interesados.

Por lo que a continuación se definirá de acuerdo al rol definido en SCRUM los actores de acuerdo al proyecto actual presentado:

Rol "Involucrado"		Rol "Interesado"	
Product Owner	RUTATEC	**Usuarios**	Usuario final
Scrum Master	Alex Chávez	**Stakeholders**	RUTATEC
Equipo	Alex Chávez	**Managers**	Alex Chávez

Título: Tabla 1
Descripción: Cuadro de Asignación de Roles
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

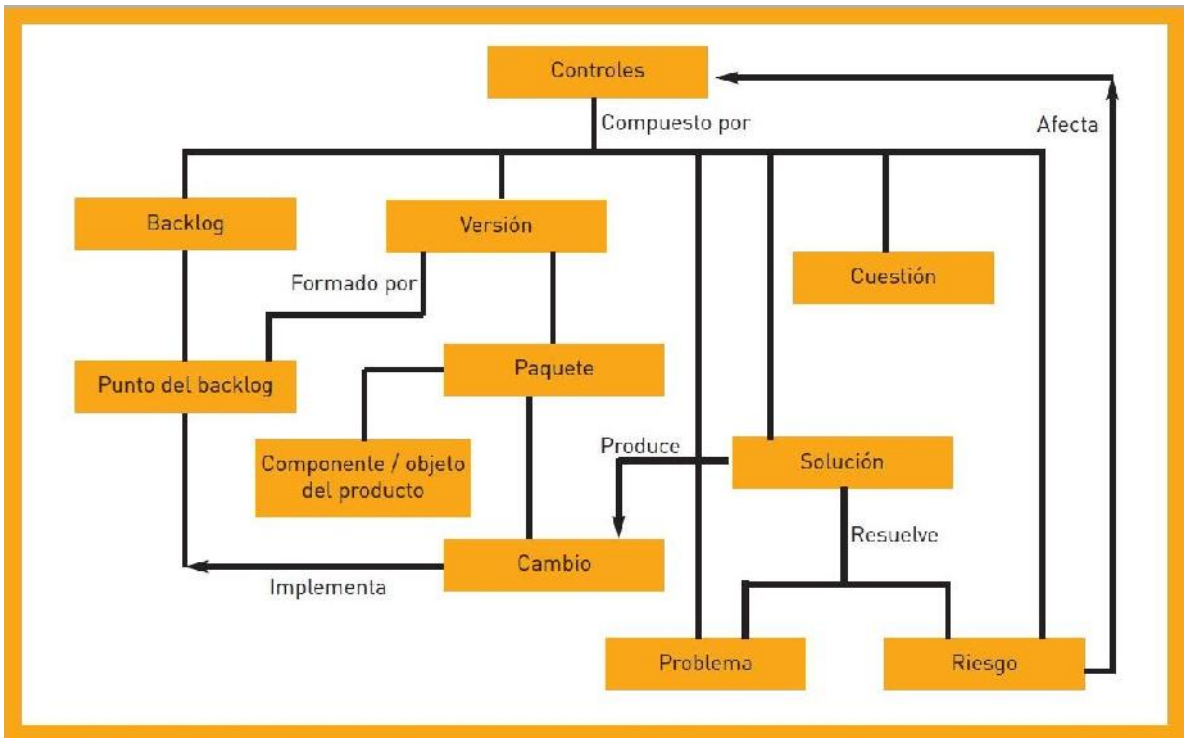
Proceso Control y seguimiento del proyecto basado en objetivos

El proyecto se ejecutará en iteraciones incrementales con una demostración del producto al finalizar cada iteración para tener una retroalimentación de cada desarrollo. De esta manera se podrá conocer de forma objetiva el estado del proyecto (si el desarrollo de los requisitos cumple con las expectativas del cliente, si la calidad es la esperada o si hay retrasos), con lo que el cliente podrá tomar decisiones informadas.

Los requisitos se desarrollarán priorizados por el valor aportado al cliente, de modo que en las primeras iteraciones se obtendrán los objetivos más importantes del proyecto y se podrán realizar ajustes al respecto con la suficiente antelación.

El control y seguimiento del proyecto se basará en los requisitos completados en cada iteración. Se entenderá un requisito como completado si incluye todos los entregables asociados realizados (documentación) e integrados con los entregables de las iteraciones anteriores, de manera que el producto sea susceptible de ser entregado al cliente con el mínimo esfuerzo. Por esta razón, cada requisito deberá ser:

- Independiente del resto de requisitos, en la medida de lo posible.
- Demostrable, de manera que sea claro cómo comprobar con el cliente que el requisito está completado y que se cumplen sus expectativas.
- De un grado de esfuerzo para ser completado semejante al del resto de requisitos, de manera que el cliente pueda realizar una extrapolación del progreso del proyecto.



Título: Ilustración 19
Descripción: Diagrama de los Controles del Proceso SCRUM
Fecha: Marzo 2012
Fuente: Realizado por José Luis Cayo - RUTATEC

1.7 ARQUITECTURA DE LA SOLUCIÓN

Smarty es un motor de plantillas para PHP, esta herramienta facilita la manera de separar la aplicación lógica y el contenido en la presentación. La mejor descripción a esto es en una situación donde la aplicación del programador y la plantilla del diseñador juegan diferentes roles, o en la mayoría de los casos no son la misma persona.

Smarty no intenta separar completamente la lógica de la plantilla. No hay problema entre la lógica y su plantilla bajo la condición que esta lógica sea estrictamente para presentación. Se aconseja mantener la aplicación lógica fuera de la plantilla, y la presentación fuera de la aplicación lógica. Esto tiene como finalidad tener un objeto más manipulable y escalable para un futuro próximo.

Smarty utiliza el patrón de abstracción **modelo vista controlador**[24] **(MVC)** el cual se divide en tres componentes:

COMPONENTE CAPA VISTA

Es la interfaz de usuario en un formato adecuado para interactuar con el sistema. Esta se subdivide a la vez en:

[24] **Modelo Vista Controlador (MVC)** es un patrón o modelo de abstracción de desarrollo de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos (modelo, vista y controlador).

- **Front end**.- en esta se encuentran plantillas que interactúan con la capa controlador; en este desarrollo son archivos del tipo **.tpl,** los mismos que contienen código HTML y código de Smarty, estos son compilados por Smarty y finalmente se muestran al usuario como interfaz.

- **Back end**.- en esta se encuentran archivos que interactúan con el Front end, mostrando la información solicitada. Para nuestro caso de estudio son archivos **.PHP,** estos contienen código PHP, los cuales interactúan con los archivos que se encuentran en el Front end asignando la información correspondiente.

COMPONENTE CAPA CONTROLADOR

Este responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y, probablemente, a la vista. Esta maneja archivos de JavaScript, JQuery[25] y AJAX[26].

COMPONENTE CAPA MODELO

Esta es la representación específica de la información con la cual el sistema opera. En resumen, el modelo se limita a lo relativo de la vista y su controlador facilitando las presentaciones visuales complejas. En la capa de modelo también se encuentran las reglas del negocio y las clases de persistencia que se comunican con la base de datos. Los elementos que forman parte de la capa modelo en el desarrollo de la presente disertación son:

Clases de persistencia.- son clases creadas con código PHP, las cuales se encargan de interactuar con la base de datos.

Reglas de negocio.- archivos en código PHP que contienen procesos propios del negocio. Se comunican con las clases de persistencia si necesitan interactuar con la base de datos.

Archivos remotos.- es una comunicación entre procesos, que permite hacer una llamada a una subrutina o procedimiento desde otra interfaz; cuando se utiliza principios orientados a objetos, se conoce como RPC a la invocación de métodos remotos.

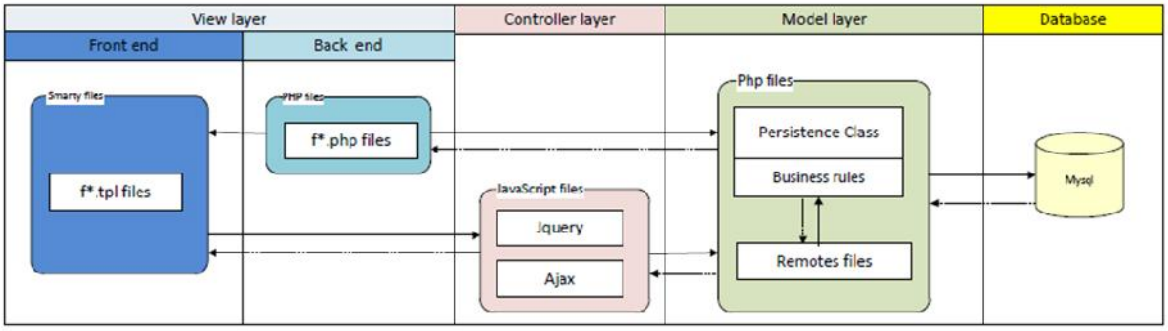
BASE DE DATOS

MySQL.- La base de datos MySQL se ha convertido en la base de datos de código abierto más popular debido a su alto rendimiento, alta fiabilidad y facilidad de uso. También es la base de datos de elección para una nueva generación de aplicaciones basadas en la pila LAMP (Linux, Apache, MySQL, PHP / Perl / Python). Muchas de las organizaciones más grandes y de más rápido crecimiento del mundo, incluyendo Facebook, Google, Adobe, Alcatel Lucent y Zappos se basan en MySQL para ahorrar tiempo y dinero.

---

[25] **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.

[26] **AJAX**, acrónimo de ***A****synchronous* ***J****avaScript* ***A****nd* ***X****ML* (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (*Rich Internet Applications*). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, la interactividad, velocidad y usabilidad en las aplicaciones.



Título: Ilustración 20
Descripción: Arquitectura de Smarty con PHP
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

2. CAPÍTULO 2

En este Capítulo se estudia la situación del nuevo módulo que se desea implementar y se hace un levantamiento de requerimientos; esta parte es fundamental ya que en función de los requerimientos que se encuentren en el respectivo documentos se basará toda la etapa de desarrollo; además con estos requerimientos podremos diseñar los diagramas UML los cuales nos harán más claro el entendimiento de cómo funciona el módulo del sistema.

2.1 DOCUMENTO DE REQUERIMIENTOS

Especificación de Requerimientos de Software

2.1.1 Introducción

Este documento tiene como propósito el definir en un documento formal las necesidades del cliente, en este caso los usuarios del sistema de gestión académica RUTADEMIC, con el proyecto de **Desarrollo del Módulo de Transportes para el Sistema de Gestión Académica RUTADEMIC.**

2.1.2 Propósito

El propósito de este documento es el de brindar un entorno de trabajo amigable y bien definido para el proyecto, de esta manera se podrá garantizar la entrega de todos los requerimientos para tener un producto de calidad y con las funcionalidades que se necesita.

2.1.3 Alcance

El alcance de este proyecto estará dado por todos los requerimientos funcionales y no funcionales, dejando claro que luego de la validación de estos requerimientos se cumplirá de manera estricta lo validado y cualquier cambio no estará cubierto, a menos que se haga un documento de control de cambios, para los cambios requeridos fuera de este documento, queden registrados correctamente.

2.1.4 Definiciones, acrónimos y abreviaturas

- **Requerimiento funcional:** toda actividad que cualquier usuario realiza sobre el módulo del sistema
- **Requerimiento no funcional:** restricciones para que un usuario realice las actividades funcionales

2.1.5 Referencias

En este documento se hará referencia al documento de casos de uso y descripción de los mismos.

2.1.6 Visión General

En las siguientes secciones se explicará de manera más detallada las funcionalidades y restricciones del módulo del sistema, para de esta manera dejar definidos todas las acciones que serán posibles sobre el mismo.

2.1.7 Descripción General

2.1.7.1 Perspectiva del Producto

Interfaces del sistema

El módulo del sistema contará con interfaces como la administración general de catálogos de Rutas, Turnos, Buses, Conductores; interfaz de Configuración de Recorridos; interfaz de Asignación de Recorridos a Estudiantes; interfaz para consulta de rutas con ayuda de los datos generados por el GPS en Mapas de Google.

Interfaces de usuario

El sistema contará con interfaces gráficas para los usuarios, que contendrá principalmente menús, cuadro de texto, los cuales serán muy intuitivos y amigables con el usuario para garantizar una fácil usabilidad del sistema muy aparte de la funcionalidad del mismo.

2.1.8 Requerimientos específicos

A continuación se listará los requerimientos específicos:

2.1.8.1 Funcionales

1. Crear Rutas con un código y descripción.
2. Buscar y Modificar Rutas.
3. Crear Turnos con un nombre y descripción.
4. Buscar y Modificar Turnos.
5. Crear Buses con código, número de placa, capacidad y descripción.
6. Buscar y Modificar Buses.
7. Crear Conductores con cédula de identidad, nombre, apellido, dos números de celular, tres números de contacto, dirección domicilio, tipo de licencia de conductor, fecha de caducidad de la licencia.
8. Buscar y Modificar Conductores.
9. Configurar Recorridos usando los datos de Rutas, Turnos, Buses y Conductores y además definir un conductor de Reemplazo.
10. Asignar Recorridos a Usuarios.
11. Visualizar la Ruta de un Bus en Particular con ayuda de los datos del GPS del mismo y desplegarlos en un mapa generado por Mapas de Google.

2.1.8.2 Detalle de Requerimientos Funcionales

Número de requisito	RF01
Nombre de requisito	Crear Rutas con un código y descripción
Tipo	Requisito
Descripción	El módulo del sistema deberá contar con una pantalla donde el administrador del sistema o el usuario con permisos puedan ingresar rutas nuevas para la institución educativas según las requiera.
Prioridad del requisito	Alta

Número de requisito	RF02
Nombre de requisito	Buscar y Modificar Rutas
Tipo	Requisito
Descripción	El módulo deberá tener una pantalla para modificar las rutas ingresadas al sistema en el requisito RF01.
Prioridad del requisito	Alta

Número de requisito	RF03
Nombre de requisito	Crear Turnos con un nombre y descripción
Tipo	Requisito
Descripción	El módulo del sistema deberá contar con una pantalla donde el administrador del sistema o el usuario con permisos puedan ingresar turnos nuevos para las rutas configuradas.
Prioridad del requisito	Alta

Número de requisito	RF04
Nombre de requisito	Buscar y Modificar Turnos
Tipo	Requisito
Descripción	El módulo deberá tener una pantalla para modificar los turnos ingresados al sistema en el requisito RF03.
Prioridad del requisito	Alta

Número de requisito	RF05
Nombre de requisito	Crear Buses con código, número de placa, capacidad y descripción
Tipo	Requisito
Descripción	El módulo del sistema deberá contar con una pantalla donde el administrador del sistema o el usuario con permisos puedan ingresar buses con la información solicitada.
Prioridad del requisito	Alta

Número de requisito	RF06
Nombre de requisito	Buscar y Modificar Buses
Tipo	Requisito
Descripción	El módulo deberá tener una pantalla para modificar los buses ingresados al sistema en el requisito RF05.
Prioridad del requisito	Alta

Número de requisito	RF07
Nombre de requisito	Crear Conductores con cédula de identidad, nombre, apellido, dos números de celular, tres números de contacto, dirección de domicilio, tipo de licencia de conductor, fecha de caducidad de la licencia.
Tipo	Requisito
Descripción	El módulo del sistema deberá contar con una pantalla donde el administrador del sistema o el usuario con permisos puedan ingresar conductores de bus con todos los datos solicitados.
Prioridad del requisito	Alta

Número de requisito	RF08
Nombre de requisito	Buscar y Modificar Conductores
Tipo	Requisito
Descripción	El módulo deberá tener una pantalla para modificar los conductores ingresados al sistema en el requisito RF07.

Prioridad del requisito	Alta

Número de requisito	RF09
Nombre de requisito	Configurar Recorridos usando los datos de Rutas, Turnos, Buses y Conductores y además definir un conductor de reemplazo.
Tipo	Requisito
Descripción	El módulo del sistema deberá contar con una pantalla donde el administrador del sistema o el usuario con permisos puedan configurar los recorridos basándose en los datos de rutas, turnos, buses y conductores ingresados anteriormente en los requisitos RF01, RF03, RF05, RF07.
Prioridad del requisito	Alta

Número de requisito	RF10
Nombre de requisito	Asignar Recorridos a Usuarios.
Tipo	Requisito
Descripción	El módulo del sistema deberá contar con una pantalla donde el administrador del sistema o el usuario con permisos puedan asignar a los usuarios existentes en el sistema (estudiantes, empleados) a los diferentes recorridos configurados en el requerimiento RF09.
Prioridad del requisito	Alta

Número de requisito	RF11
Nombre de requisito	Visualizar la Ruta de un Bus en Particular con ayuda de los datos del GPS del mismo y desplegarlos en un mapa generado por Mapas de Google.
Tipo	Requisito
Descripción	El módulo del sistema deberá contar con una pantalla donde los representantes puedan ver la ruta del recorrido al que sus representados están asignados y además representar en un mapa los datos del GPS del bus.
Prioridad del requisito	Alta

2.1.8.3 Especificación de casos de uso

Hacer referencia al documento de casos de uso.

2.1.8.4 Requerimientos no funcionales

Rendimiento

El Módulo de Transporte del Sistema de Gestión Académica RUTADEMIC debe tener en el 90% de los casos un tiempo de respuesta menor a 2 segundos. Estos tiempos pueden variar dependiendo tanto de las características y disponibilidad del servidor como del estado y capacidad del canal de comunicación.

Seguridad

La seguridad del módulo de Transporte seguirá los mismos lineamentos que el Sistema de Gestión Académica RUTADEMIC, es decir debe ser con perfiles de acceso para los usuarios que ingresen al sistema, de esta forma cada usuario tendrá acceso únicamente a lo que corresponda al perfil que tiene.

Adicionalmente a nivel de base de datos las contraseñas de todos los usuarios estarán encriptadas.

Fiabilidad

El Módulo de Transporte del Sistema de Gestión Académica RUTADEMIC deberá tener una fiabilidad del 100% ya que de este dependerá la gestión del transporte interno de los centros académicos que usen el sistema.

Disponibilidad

La disponibilidad del Módulo de Transporte deberá ser igual a la del Sistema de Gestión Académica RUTADEMIC esto es 24/7/365, es decir siempre se podrá acceder a la aplicación vía web, si el hardware, software y elementos externos a la aplicación también estén disponibles. Se entiende que no se encontrará disponible cuando se haga cambio de versiones o esté en mantenimiento con previo aviso a las diferentes partes que hacen uso de él.

Mantenibilidad

El código de la aplicación estará debidamente comentado de tal manera que personas capacitadas en las herramientas utilizadas para el desarrollo puedan comprender el código y dar el mantenimiento correspondiente.

Portabilidad

El Módulo de Transporte al igual que el Sistema de Gestión Académica RUTADEMIC deberá ser portable tanto a nivel de hardware como a nivel de sistema operativo. La aplicación podrá ser instalada al menos sobre los sistemas operativos Windows y Linux de 32 o 64 bits.

2.1.9 Otros requisitos

El Módulo de Transporte del Sistema de Gestión Académica RUTADEMIC deberá visualizarse correctamente en las últimas versiones de los siguientes navegadores: Mozilla Firefox, Internet Explorer, Google Chrome y Safari.

Para el documento completo ver el ***Documento de Especificaciones y Requerimientos de Software*** *en el Anexo 1*

2.2 MODELADO DE CASOS DE USO

Modelado de Diagramas de Caso de Uso

A continuación vamos a mostrar los diagramas de caso de uso de manera grupal e individual para cada acción con sus respectivos actores. Un caso de uso es una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso. Los personajes o entidades que participarán en un caso de uso se denominan actores.

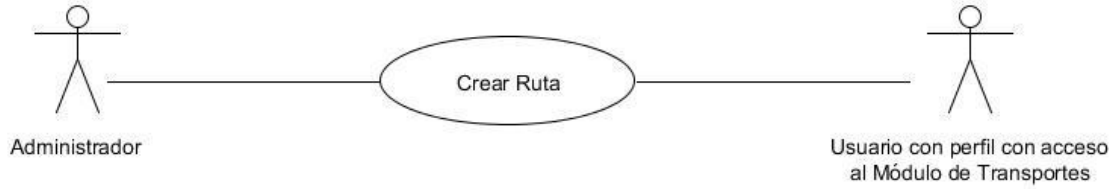
2.2.1 Caso de Uso General Simplificado



Título: Ilustración 21
Descripción: Diagrama General de Casos de Uso
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

2.2.2 Descripción de Casos de Uso

2.2.2.1 Caso de Uso CUS-01



Título: Ilustración 22
Descripción: Diagrama de Casos de Uso CUS-01
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-01
Nombre	Crear Ruta.
Actores	• Administrador del Sistema. • Usuario con perfil con acceso al módulo de Transportes.
Precondición	• Tener permisos de acceso al módulo de transportes.
Flujo básico	1. Ingresar al sistema como administrador o con un perfil con acceso al módulo de transportes. 2. Ingresar a Parámetros. 3. Ingresar a Transporte. 4. Ingresar a Ruta. 5. Ingresar a Nueva Ruta.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Ruta Creada en la Base de Datos.
Resultado	Visualización de la Ruta dentro del sistema para su utilización en Configuración de Transporte.

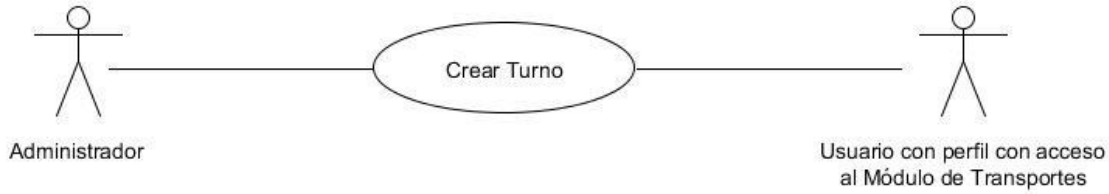
2.2.2.2 Caso de Uso CUS-02



Título: Ilustración 23
Descripción: Diagrama de Casos de Uso CUS-02
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-02
Nombre	Buscar y Modificar Ruta.
Actores	• Administrador del Sistema. • Usuario con perfil con acceso al módulo de Transportes.
Precondición	• Tener permisos de acceso al módulo de transportes.
Flujo básico	1. Ingresar al sistema como administrador o con un perfil con acceso al módulo de transportes. 2. Ingresar a Parámetros. 3. Ingresar a Transporte. 4. Ingresar a Ruta. 5. Ingresar a Modificar Ruta.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Ruta Modificada en la Base de Datos.
Resultado	Visualización de la Ruta con los cambios necesarios dentro del sistema para su utilización en Configuración de Transporte.

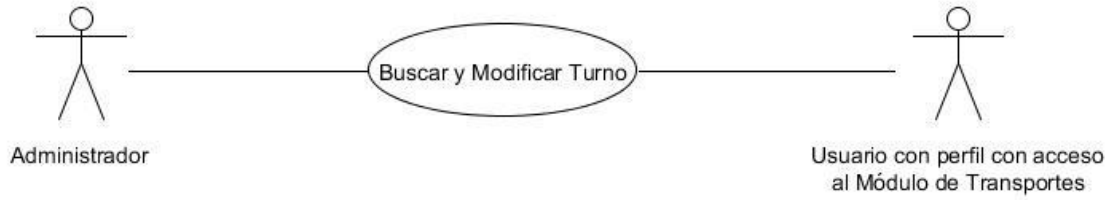
2.2.2.3 Caso de Uso CUS-03



Título: Ilustración 24
Descripción: Diagrama de Casos de Uso CUS-03
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-03
Nombre	Crear Turno.
Actores	• Administrador del Sistema. • Usuario con perfil con acceso al módulo de Transportes.
Precondición	• Tener permisos de acceso al módulo de transportes.
Flujo básico	1. Ingresar al sistema como administrador o con un perfil con acceso al módulo de transportes. 2. Ingresar a Parámetros. 3. Ingresar a Transporte. 4. Ingresar a Turno. 5. Ingresar a Nueva Turno.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Turno Creado en la Base de Datos.
Resultado	Visualización del Turno dentro del sistema para su utilización en Configuración de Transporte.

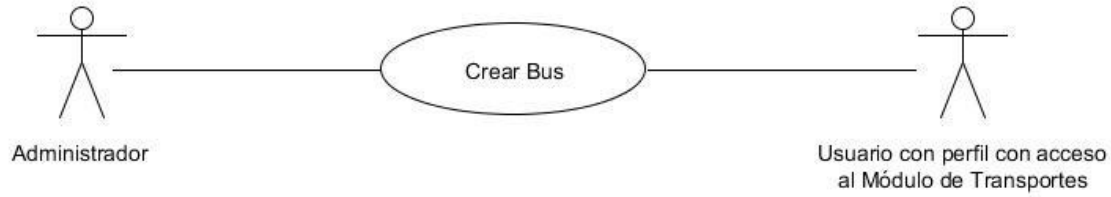
2.2.2.4 Caso de Uso CUS-04



Título: Ilustración 25
Descripción: Diagrama de Casos de Uso CUS-04
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-04
Nombre	Buscar y Modificar Turno.
Actores	• Administrador del Sistema. • Usuario con perfil con acceso al módulo de Transportes.
Precondición	• Tener permisos de acceso al módulo de transportes.
Flujo básico	1. Ingresar al sistema como administrador o con un perfil con acceso al módulo de transportes. 2. Ingresar a Parámetros. 3. Ingresar a Transporte. 4. Ingresar a Turno. 5. Ingresar a Modificar Turno.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Turno Modificado en la Base de Datos.
Resultado	Visualización del Turno con los cambios necesarios dentro del sistema para su utilización en Configuración de Transporte.

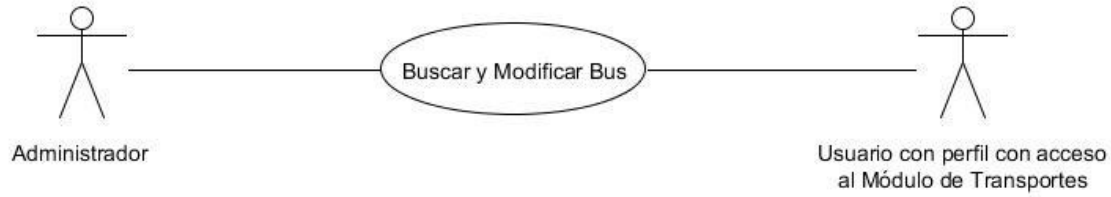
2.2.2.5 Caso de Uso CUS-05



Título: Ilustración 26
Descripción: Diagrama de Casos de Uso CUS-05
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-05
Nombre	Crear Bus.
Actores	• Administrador del Sistema. • Usuario con perfil con acceso al módulo de Transportes.
Precondición	• Tener permisos de acceso al módulo de transportes.
Flujo básico	1. Ingresar al sistema como administrador o con un perfil con acceso al módulo de transportes. 2. Ingresar a Parámetros. 3. Ingresar a Transporte. 4. Ingresar a Bus. 5. Ingresar a Nuevo Bus.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Bus Creado en la Base de Datos.
Resultado	Visualización del Bus dentro del sistema para su utilización en Configuración de Transporte.

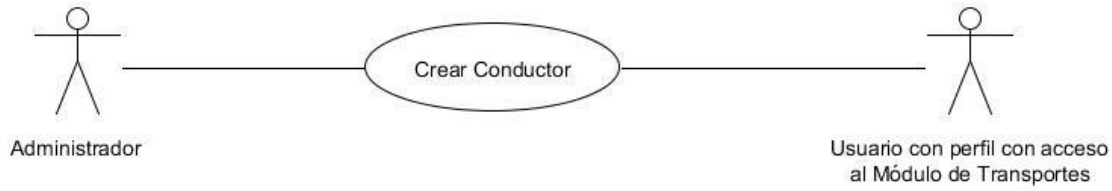
2.2.2.6 Caso de Uso CUS-06



Título: Ilustración 27
Descripción: Diagrama de Casos de Uso CUS-06
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-06
Nombre	Buscar y Modificar Bus
Actores	• Administrador del Sistema. • Usuario con perfil con acceso al módulo de Transportes.
Precondición	• Tener permisos de acceso al módulo de transportes.
Flujo básico	1. Ingresar al sistema como administrador o con un perfil con acceso al módulo de transportes. 2. Ingresar a Parámetros. 3. Ingresar a Transporte. 4. Ingresar a Bus. 5. Ingresar a Modificar Bus.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Bus Modificado en la Base de Datos.
Resultado	Visualización del Bus con los cambios necesarios dentro del sistema para su utilización en Configuración de Transporte.

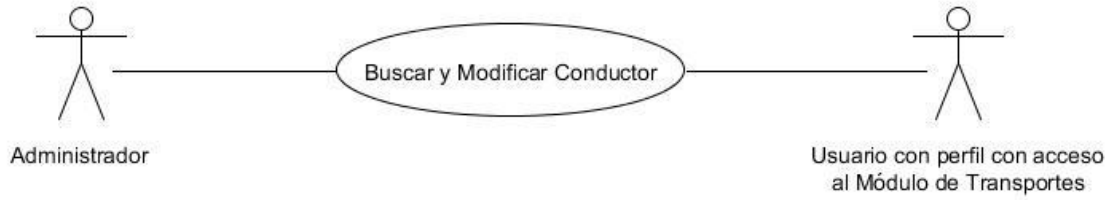
2.2.2.7 Caso de Uso CUS-07



Título: Ilustración 28
Descripción: Diagrama de Casos de Uso CUS-07
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-07
Nombre	Crear Conductor
Actores	• Administrador del Sistema. • Usuario con perfil con acceso al módulo de Transportes.
Precondición	• Tener permisos de acceso al módulo de transportes.
Flujo básico	1. Ingresar al sistema como administrador o con un perfil con acceso al módulo de transportes. 2. Ingresar a Parámetros. 3. Ingresar a Transporte. 4. Ingresar a Conductor. 5. Ingresar a Nuevo Conductor.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Conductor Creado en la Base de Datos.
Resultado	Visualización del Conductor dentro del sistema para su utilización en Configuración de Transporte.

2.2.2.8 Caso de Uso CUS-08



Título: Ilustración 29
Descripción: Diagrama de Casos de Uso CUS-08
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-08
Nombre	Buscar y Modificar Conductor.
Actores	• Administrador del Sistema. • Usuario con perfil con acceso al módulo de Transportes.
Precondición	• Tener permisos de acceso al módulo de transportes.
Flujo básico	6. Ingresar al sistema como administrador o con un perfil con acceso al módulo de transportes. 7. Ingresar a Parámetros. 8. Ingresar a Transporte. 9. Ingresar a Conductor. 10. Ingresar a Modificar Conductor.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Conductor Modificado en la Base de Datos.
Resultado	Visualización del Conductor con los cambios necesarios.

2.2.2.9 Caso de Uso CUS-09



Título: Ilustración 30
Descripción: Diagrama de Casos de Uso CUS-09
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-09
Nombre	Configurar Recorridos.
Actores	• Administrador del Sistema. • Usuario con perfil con acceso al módulo de Transportes.
Precondición	• Tener permisos de acceso al módulo de transportes. • Tener un año lectivo credo en el sistema. • Tener rutas, turnos, buses y conductores ingresados en el sistema.
Flujo básico	1. Ingresar al sistema como administrador o con un perfil con acceso al módulo de transportes. 2. Ingresar a Configuración. 3. Ingresar a Transporte. 4. Ingresar a Iniciar/Continuar.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Recorrido Configurado en la Base de Datos.
Resultado	Visualización del Recorrido en el sistema para su utilización en la asignación de usuarios a recorridos.

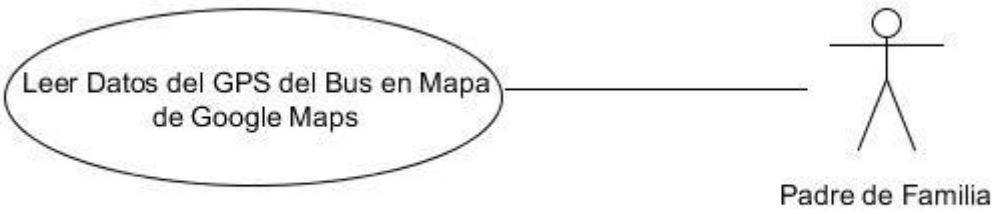
2.2.2.10 Caso de Uso CUS-10



Título: Ilustración 31
Descripción: Diagrama de Casos de Uso CUS-10
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-10
Nombre	Asignar Usuarios a Recorridos
Actores	• Administrador del Sistema. • Usuario con perfil con acceso al módulo de Transportes.
Precondición	• Tener permisos de acceso al módulo de transportes. • Tener un recorrido configurado en el sistema. • Tener usuarios registrados en el sistema.
Flujo básico	1. Ingresar al sistema como administrador o con un perfil con acceso al módulo de transportes. 2. Ingresar a Configuración. 3. Ingresar a Transporte. 4. Ingresar a Asignar Usuarios.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Usuarios Asignados a Recorrido.
Resultado	Usuario asignado a ruta para poder obtener reportes y consultas desde el sistema.

2.2.2.11 Caso de Uso CUS-11



Título: Ilustración 32
Descripción: Diagrama de Casos de Uso CUS-11
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

Código	CUS-11
Nombre	Leer Datos del GPS del Bus en Mapa de Google Maps.
Actores	• Padre de Familia.
Precondición	• Tener permisos de acceso al reporte de datos del GPS del módulo de transportes. • Tener un hijo que haga uso del transporte dentro del sistema.
Flujo básico	1. Ingresar al sistema como Padre de Familia. 2. Ingresar a Reportes.

	3. Ingresar a Transporte. 4. Ingresar a Ver Rutas de Transporte.
Flujos alternativos / Excepciones	No aplica.
Post condiciones	Visualizar Ruta del Bus con datos del GPS en Google Maps embebido en el sistema.
Resultado	Reporte en Pantalla y Mapa en Pantalla.

*Para el documento completo ver el **Documento de** **Modelado de Casos de Uso** en el Anexo 1.*

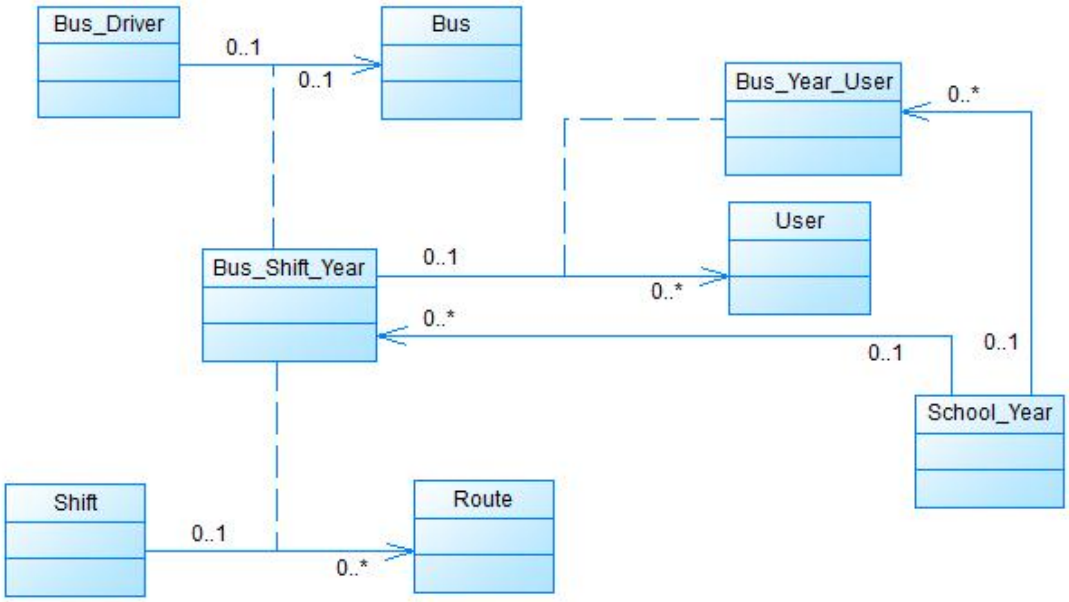
2.3 Diagrama de Clases

El presente tema de disertación es un módulo para un sistema ya en funcionamiento; por este motivo solo representamos en diagramas las clases nuevas y las clases que se utilizan para tener una funcionalidad completa por separado.



Título: Ilustración 33
Descripción: Diagrama de Clases
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

2.4 Diagrama de Clases Conceptual Con Asociaciones y Multiplicidad

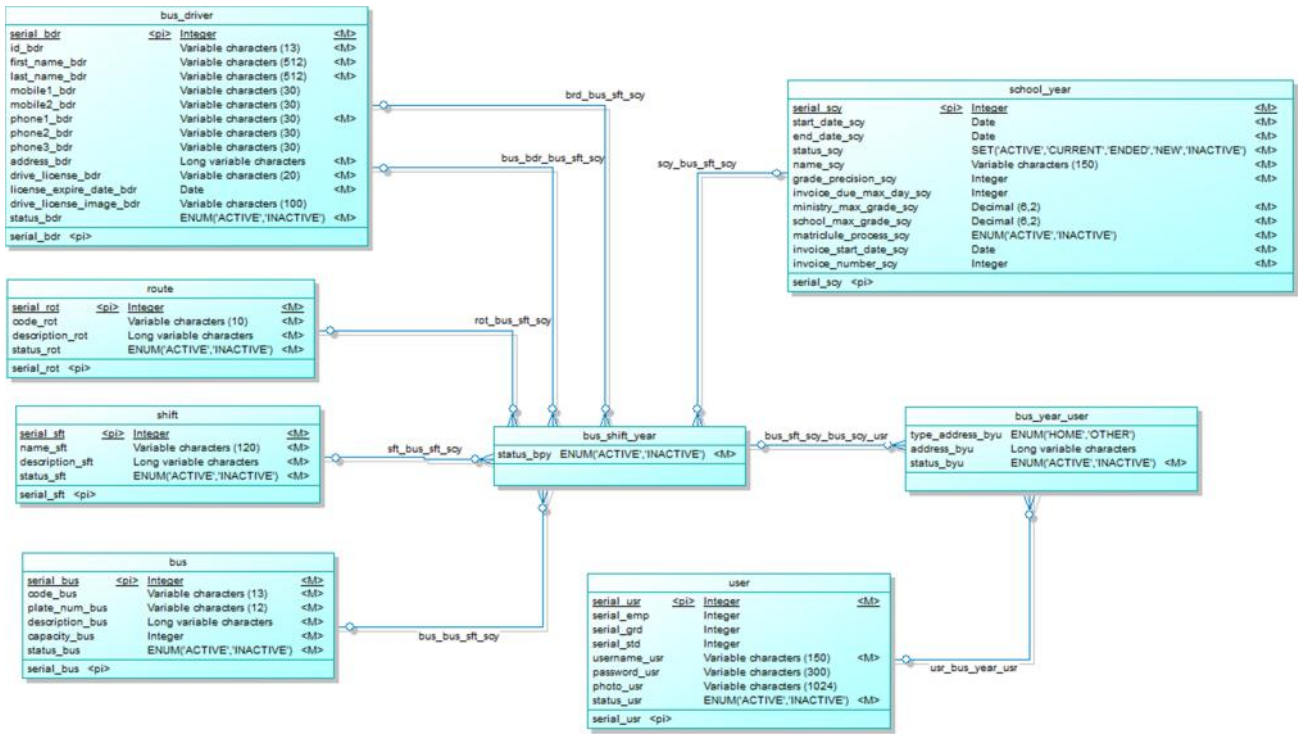


Título: Ilustración 34
Descripción: Diagrama de Clases Conceptual
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

2.5 MODELADO DE BASE DE DATOS

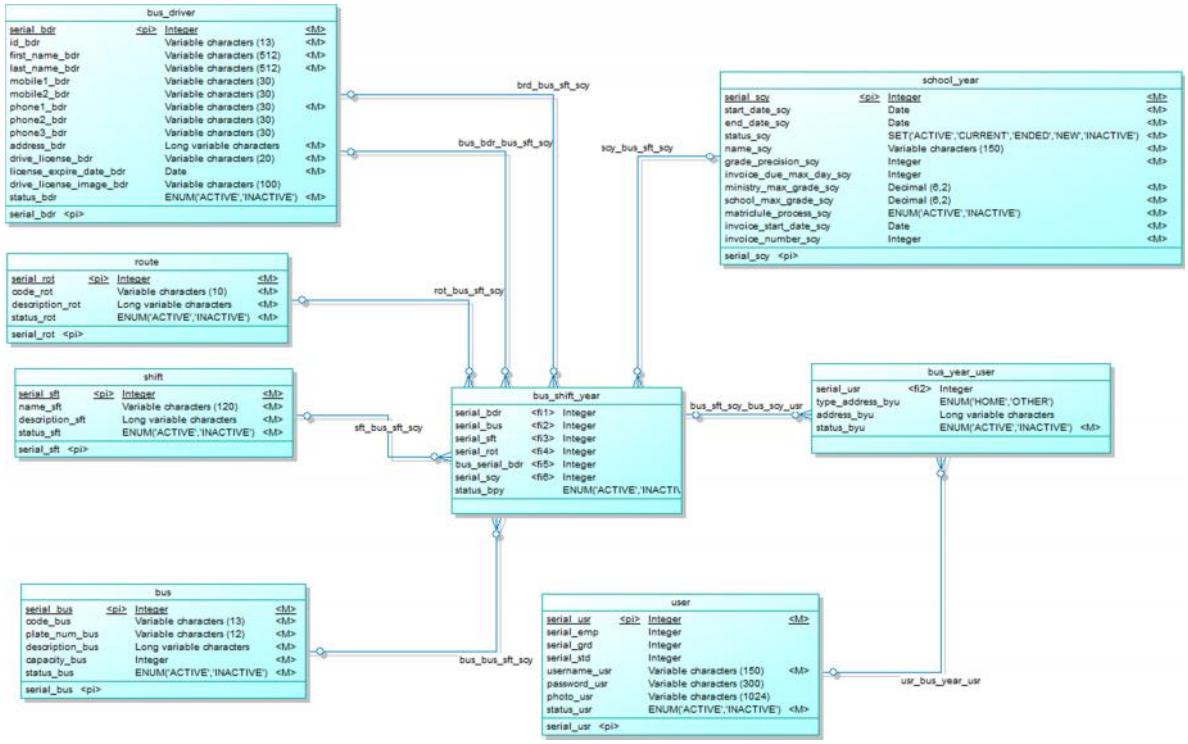
A continuación se presenta el los diversos modelos de las tablas de nuestra base, esto ayuda a explicar y conocer un poco más a fondo el modo de almacenar, organizar y manipular los datos del módulo que se desarrolla.

2.5.1 Modelo Conceptual de la Base de Datos



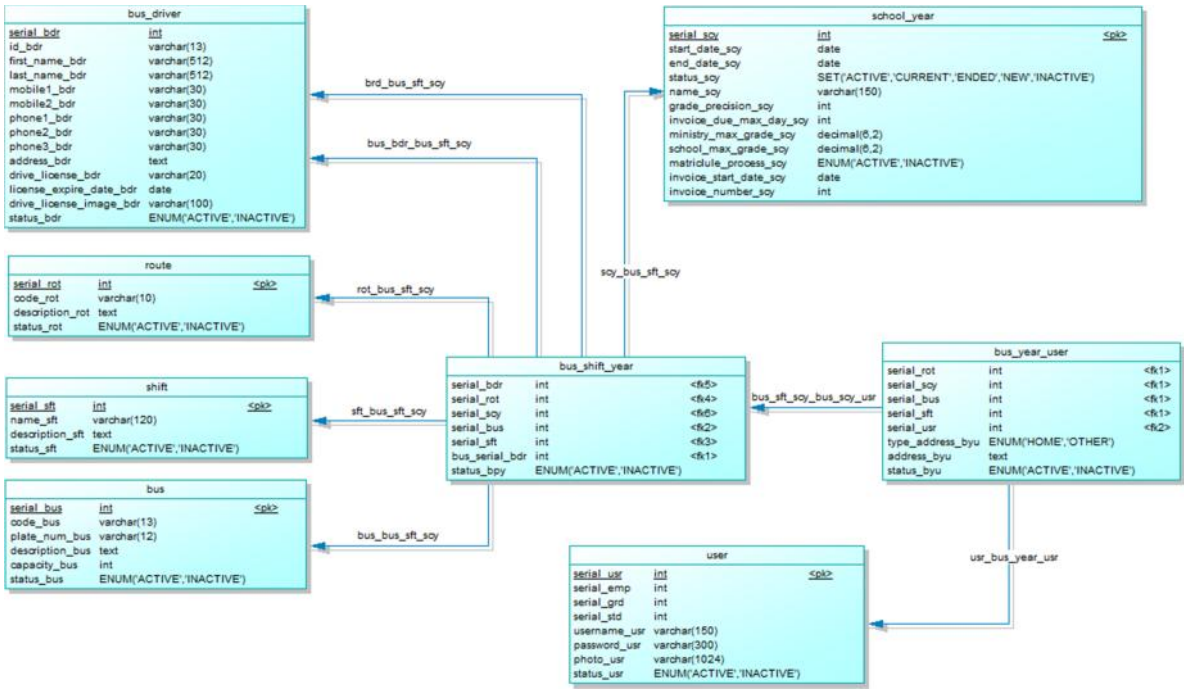
Título: Ilustración 35
Descripción: Modelo Conceptual de la Base de Datos
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

2.5.2 Modelo Lógico de la Base de Datos



Título: Ilustración 36
Descripción: Modelo Lógico de la Base de Datos
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

2.5.3 Modelo Físico de la Base de Datos



Título: Ilustración 37
Descripción: Modelo Físico de la Base de Datos
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

3. CAPÍTULO 3

En este tercer capítulo, se realizará la implementación del módulo de transportes en el Sistema de Gestión Académica; en primer lugar, se implementará las pantallas para hacer el manejo de los catálogos que son las interfaces más básicas e importantes ya que sin ellas no se podría hacer el manejo del transporte.

Cada uno de las interfaces como ya explicamos anteriormente en la arquitectura del módulo consta de tres partes: la parte del backend donde manejamos datos de entrada, la parte del proceso donde nos conectamos con la base de datos y la interfaz gráfica o frontEnd que se despliega al usuario ***(ver ilustración 20)***. Además cada una de las pantallas implementadas, como la metodología lo dice tiene que ser sometida a un proceso de retroalimentación; en el cual se sigue los siguientes pásos: implementación, pruebas unitarias e integración.

Al finalizar el desarrollo de las interfaces del módulo de transportes se realizará las pruebas integradas; esto es, se probará el correcto funcionamiento de las pantallas unitariamente y en conjunto en los dos grandes procesos: *crear recorridos y asignar usuarios* que completan el funcionamiento del módulo.

Otro punto que se deberá tener en cuenta al momento de implementar este módulo es la parte de comentarios en el código; es decir manejar el formato estándar para comentar código haciendo uso de *phpDocs*, esto se lo hace para seguir la nomenclatura impuesta por la empresa. *(Para mayor detalle ver anexo 2)*

Toda prueba realizada sobre este módulo será basada en los requerimientos y en los casos de pruebas planteados en la etapa de diseño así nos aseguramos que el módulo este bajo los requerimientos del cliente y que cumpla con los estándares de calidad solicitados.

Como se puede ver en el documento de requerimientos, el módulo de transportes está estructurado por varios catálogos, es por ésto que en la implementación y por fines explicativos solo se va a detallar la codificación de un catálogo y de una administración.

El catálogo que se va a detallar en la implementación es el catálogo de conductores; mientras que la administración que se detalla es la de generación de mapas con datos del GPS.

Es importante hacer énfasis que las demás implementaciones están en el anexo número 3.

3.1 IMPLEMENTACIÓN CATÁLOGO DE CONDUCTORES

El catálogo de conductores como su nombre lo dice es el encargado de la gestión de personas habilitadas para conducir un bus dentro del sistema, se lo ha elegido para ejemplificar la implementación ya que cuenta con varios campos que debemos validar y además tiene un manejo de imágenes.

Basándonos en los requerimientos del cliente sabemos que tenemos que validar cuidadosamente los siguientes parámetros: número de cédula (ecuatoriana y que no se repita), que tenga una licencia profesional (con archivo adjunto) y datos de contacto como son direcciones y números telefónicos (residencia y celular).

Este catálogo está dividido en dos partes fundamentales que son el Ingreso de un nuevo conductor y la actualización de datos del conductor.

3.1.1 INGRESAR NUEVO CONDUCTOR

Esta parte es la más importante ya que sin la implementación de la misma no se podría continuar con las siguientes implementaciones, porque no tendríamos datos en la base para trabajar con la actualización.
Como se detalla en la parte del diseño, esta y todas las pantallas del sistema trabajan con la arquitectura de 3 capas, aplicando esto dividimos la implementación en 5 archivos que nos ayudan a hacer el manejo de 3 capas, estos son:

Nombre Archivo	Descripción
Bus_Driver.class.php	Es una clase de php en cuál se maneja todos los atributos que puede tener un conductor y además todos los métodos que se requieren para que la interfaz trabaje de una forma adecuada junto con la base de datos. Un punto que se debe tener en cuenta es que esta clase es la conexión que se tiene entre las interfaces y la base de datos, específicamente con la tabla bus_driver.
fNewBusDriver.php	Es un archivo php que junto con la clase antes descrita ayuda a cargar la información necesaria para cargar la interfaz gráfica web previa a la inserción.
fNewBusDriver.js	Es un archivo JavaScript, que su principal función es dar funcionalidad a la interfaz gráfica web y controlar las distintas validaciones que se requiera, hace uso de librerías como JQuery y Ajax.
fNewBusDriver.es.tpl	Es un archivo del tipo Smarty (.tpl) que se encarga fundamentalmente de presentar una plantilla o interfaz gráfica para que los usuarios puedan interactuar con el sistema, además dentro de éste, se identifica al archivo del proceso que es llamado al presionar el botón de “Registrar”.
pNewBusDriver.php	Es un archivo php de proceso, se encarga de recopilar los datos ingresados previamente en la interfaz visual, y de ingresarlos a la base de datos con ayuda de la clase Bus_Driver.class.php.

Título: Tabla 2
Descripción: Archivos Implementación Interfaz de Ingreso de Nuevo Conductor
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

A continuación se presenta el código fuente de los archivos antes descritos:

Bus_Driver.class.php

```php
<?php

/*
 File: Bus_Driver.class
 Author: Alex Chavez
 Creation Date: 06/02/2013
*/

Class Bus_Driver {

  private $db;

  /**
   * Bus Driver Serial
   * @var uInt
   */
  private $serial_bdr;

  /**
   * Bus Driver ID
   * @var String
   */
  private $id_bdr;

  /**
   * Bus Driver First Name
   * @var String
   */
  private $first_name_bdr;

  /**
   * Bus Driver Last Name
   * @var String
   */
  private $last_name_bdr;

  /**
   * Bus Driver Mobile 1
   * @var String
   */
  private $mobile1_bdr;

  /**
   * Bus Driver Mobile 2
   * @var String
   */
  private $mobile2_bdr;

  /**
   * Bus Driver Phone 1
   * @var String
   */
  private $phone1_bdr;

  /**
   * Bus Driver Phone 2
   * @var String
   */
  private $phone2_bdr;

  /**
   * Bus Driver Phone 3
   * @var String
   */
  private $phone3_bdr;

  /**
   * Bus Driver Address
   * @var String
   */
  private $address_bdr;

  /**
   * Bus Driver Driver License
   * @var String
   */
  private $drive_license_bdr;

  /**
   * Bus Driver Driver License
   * @var String
   */
  private $drive_license_image_bdr;

  /**
   * Driver License Expiration Date
   * @var String
   */
  private $license_expire_date_bdr;

  /**
   * Bus Driver status
   * @var Enum
   */
  private $status_bdr;
  public static $status = array('ACTIVE' => 'Activo',
'INACTIVE' => 'Inactivo');

  public function __construct($db, $serial_bdr = 0) {
    $this->db = $db;
    $this->serial_bdr = $serial_bdr;
    if ($serial_bdr)
      $this->load();
  }

  ###### SET FUNCTIONS ######

  /**
   * @param uInt $serial_bdr
   */
  public function setSerialBdr($serial_bdr) {
    $this->serial_bdr = $serial_bdr;
  }

  /**
   * @param String $id_bdr
   */
  public function setIdBdr($id_bdr) {
    $this->id_bdr = $id_bdr;
  }

  /**
   * @param String $first_name_bdr
   */
  public function setFirstNameBdr($first_name_bdr) {
    $this->first_name_bdr = $first_name_bdr;
  }

  /**
   * @param String $last_name_bdr
   */
  public function setLastNameBdr($last_name_bdr) {
```

```
        $this->last_name_bdr = $last_name_bdr;
    }

    /**
     * @param String $mobile1_bdr
     */
    public function setMobile1Bdr($mobile1_bdr) {
        $this->mobile1_bdr = $mobile1_bdr;
    }

    /**
     * @param String $mobile1_bdr
     */
    public function setMobile2Bdr($mobile2_bdr) {
        $this->mobile2_bdr = $mobile2_bdr;
    }

    /**
     * @param String $phone1_bdr
     */
    public function setPhone1Bdr($phone1_bdr) {
        $this->phone1_bdr = $phone1_bdr;
    }

    /**
     * @param String $phone2_bdr
     */
    public function setPhone2Bdr($phone2_bdr) {
        $this->phone2_bdr = $phone2_bdr;
    }

    /**
     * @param String $phone1_bdr
     */
    public function setPhone3Bdr($phone3_bdr) {
        $this->phone3_bdr = $phone3_bdr;
    }

    /**
     * @param String $address_bdr
     */
    public function setAddressBdr($address_bdr) {
        $this->address_bdr = $address_bdr;
    }

    /**
     * @param String $drive_license_bdr
     */
    public function setDriveLicenseBdr($drive_license_bdr) {
        $this->drive_license_bdr = $drive_license_bdr;
    }

    /**
     * @param String $drive_license_bdr
     */
    public function
setDriveLicenseImageBdr($drive_license_image_bdr) {
        $this->drive_license_image_bdr =
$drive_license_image_bdr;
    }

    /**
     * @param String $license_expire_date_bdr
     */
    public function
setLicenseExpireDateBdr($license_expire_date_bdr) {
        $this->license_expire_date_bdr =
$license_expire_date_bdr;
    }

    /**
     * @param ENUM $status_emp
     */
    public function setStatusBdr($status_bdr) {
        if (!array_key_exists($status_bdr, self::$status))
            throw new
Exception('INVALID_BUS_DRIVER_STATUS');

        $this->status_bdr = $status_bdr;
    }

    ###### GET FUNCTIONS ######
    /**
     * @return uInt
     */

    public function getSerialBdr() {
        return $this->serial_bdr;
    }

    /**
     * @return uInt
     */
    public function getIdBdr() {
        return $this->id_bdr;
    }

    /**
     * @return String
     */
    public function getFirstNameBdr() {
        return $this->first_name_bdr;
    }

    /**
     * @return String
     */
    public function getLastNameBdr() {
        return $this->last_name_bdr;
    }

    /**
     * @return String
     */
    public function getMobile1Bdr() {
        return $this->mobile1_bdr;
    }

    /**
     * @return String
     */
    public function getMobile2Bdr() {
        return $this->mobile2_bdr;
    }

    /**
     * @return String
     */
    public function getPhone1Bdr() {
        return $this->phone1_bdr;
    }

    /**
     * @return String
     */
    public function getPhone2Bdr() {
        return $this->phone2_bdr;
    }

    /**
     * @return String
     */
```

```
  public function getPhone3Bdr() {
    return $this->phone3_bdr;
  }

  /**
   * @return String
   */
  public function getAddressBdr() {
    return $this->address_bdr;
  }

  /**
   * @return String
   */
  public function getDriveLicenseBdr() {
    return $this->drive_license_bdr;
  }

  /**
   * @return String
   */
  public function getDriveLicenseImageBdr() {
    return $this->drive_license_image_bdr;
  }

  /**
   * @return Date
   */
  public function getLicenseExpireDateBdr() {
    return $this->license_expire_date_bdr;
  }

  /**
   * @return ENUM
   */
  public function getStatusBdr() {
    return $this->status_bdr;
  }

  ###### PRIVATE FUNCTIONS ######

  /**
   * Get object data from DB
   * @throws invalid BusDirver
   */
  private function load() {
    $sql = "SELECT serial_bdr,
                id_bdr,
                first_name_bdr,
                last_name_bdr,
                mobile1_bdr,
                mobile2_bdr,
                phone1_bdr,
                phone2_bdr,
                phone3_bdr,
                address_bdr,
                drive_license_bdr,
                drive_license_image_bdr,
                DATE_FORMAT(license_expire_date_bdr,
'%d/%m/%Y') as license_expire_date_bdr, status_bdr
                FROM bus_driver
                WHERE serial_bdr = {$this->getSerialBdr()}";

    $result = $this->db->getRow($sql);

    if ($result === FALSE)
      throw new Exception("INVALID_BUS_DRIVER");

    $this->id_bdr = $result['id_bdr'];
    $this->first_name_bdr = $result['first_name_bdr'];
    $this->last_name_bdr = $result['last_name_bdr'];
    $this->mobile1_bdr = $result['mobile1_bdr'];
    $this->mobile2_bdr = $result['mobile2_bdr'];
    $this->phone1_bdr = $result['phone1_bdr'];
    $this->phone2_bdr = $result['phone2_bdr'];
    $this->phone3_bdr = $result['phone3_bdr'];
    $this->address_bdr = $result['address_bdr'];
    $this->drive_license_bdr = $result['drive_license_bdr'];
    $this->drive_license_image_bdr =
$result['drive_license_image_bdr'];
    $this->license_expire_date_bdr =
$result['license_expire_date_bdr'];
    $this->status_bdr = $result['status_bdr'];
  }

  ###### PUBLIC FUNCTIONS ######
  /**
   * Inserts or Updates a Bus Driver in the DB
   * @throws No Update Bus Driver or No Insert Bus Driver
   */

  public function save() {

    if ($this->getSerialBdr()) {
      //UPDATE
      $sql = "UPDATE bus_driver
          SET
          id_bdr = '{$this->getIdBdr()}',
          first_name_bdr = '" . addslashes(trim($this-
>getFirstNameBdr())) . "',
          last_name_bdr = '" . addslashes(trim($this-
>getLastNameBdr())) . "',
          mobile1_bdr = '{$this->getMobile1Bdr()}',
          mobile2_bdr = '{$this->getMobile2Bdr()}',
          phone1_bdr = '{$this->getPhone1Bdr()}',
          phone2_bdr = '{$this->getPhone2Bdr()}',
          phone3_bdr = '{$this->getPhone3Bdr()}',
          address_bdr = '" . addslashes(trim($this-
>getAddressBdr())) . "',
          drive_license_bdr = '" . addslashes(trim($this-
>getDriveLicenseBdr())) . "',
                drive_license_image_bdr = '" .
addslashes(trim($this->getDriveLicenseImageBdr())) . "',
          license_expire_date_bdr =
STR_TO_DATE('{$this-
>getLicenseExpireDateBdr()}','%d/%m/%Y'),
          status_bdr = '{$this->getStatusBdr()}'";
      $sql .= "WHERE serial_bdr = {$this->getSerialBdr()}";

      $result = $this->db->Execute($sql);
      if ($result === false)
        throw new
Exception('NO_UPDATE_BUS_DRIVER');

      $this->load();
    }else {
      //INSERT
      $sql = "INSERT INTO bus_driver
          (serial_bdr,
          id_bdr,
          first_name_bdr,
          last_name_bdr,
          mobile1_bdr,
          mobile2_bdr,
          phone1_bdr,
          phone2_bdr,
          phone3_bdr,
          address_bdr,
          drive_license_bdr,
                drive_license_image_bdr,
          license_expire_date_bdr,
          status_bdr
```

```
            )
            VALUES
            (
              NULL,";
        $sql .= "'{$this->getIdBdr()}',
              '" . addslashes(trim($this-
>getFirstNameBdr())) . "',
              '" . addslashes(trim($this->getLastNameBdr()))
. "',
              '{$this->getMobile1Bdr()}',
              '{$this->getMobile2Bdr()}',
              '{$this->getPhone1Bdr()}',
              '{$this->getPhone2Bdr()}',
              '{$this->getPhone3Bdr()}',
              '" . addslashes(trim($this->getAddressBdr())) .
"',
              '" . addslashes(trim($this-
>getDriveLicenseBdr())) . "',
                '" . addslashes(trim($this-
>getDriveLicenseImageBdr())) . "',
              STR_TO_DATE('{$this-
>getLicenseExpireDateBdr()}','%d/%m/%Y'),
              'ACTIVE')";

        $result = $this->db->Execute($sql);
        if ($result == true) {
          $this->serial_bdr = $this->db->insert_ID();
          $this->load();
        }else
          throw new Exception('NO_INSERT_BUS_DRIVER');
      }
    }

    /**
     * check if the bus driver exists
     * @param String $id_bdr->Driver Id, uInt $serial_bdr-
>serial of the Driver
     * @return uInt|Bool FALSE if the  bus driver doesn't exist
yet
     */
    function checkBusDriver($id_bdr, $serial_bdr = NULL) {
      if ($serial_bdr) {
        $sql = "SELECT serial_bdr
              FROM bus_driver
              WHERE serial_bdr <> {$serial_bdr}
                AND LOWER(id_bdr)= _utf8'" .
utf8_encode($id_bdr) . "' collate utf8_bin";
      } else {
        $sql = "SELECT serial_bdr
              FROM bus_driver
              WHERE LOWER(id_bdr)= _utf8'" .
utf8_encode($id_bdr) . "' collate utf8_bin";
      }

      $result = $this->db->getOne($sql);

      if ($result) {
        return $result;
      } else {
        return FALSE;
      }
    }

    /**
     * get an array of posible bus drivers
     * @paramN/A
     * @return true if there are any bus drivers
     */
    function existBusDrivers() {
      $sql = "SELECT serial_bdr
          FROM bus_driver";
      $result = $this->db->getAll($sql);
      if ($result) {
        return true;
      } else {
        return false;
      }
    }

    /**
     * get an array of posible bus drivers from a string
entered by the user
     * @param String $data (Search Pattern)
     * @return bus drivers array
     */
    function getBusDriversAutocompleter($data) {
      $sql = "SELECT serial_bdr, first_name_bdr,
last_name_bdr, id_bdr
          FROM bus_driver
          WHERE (LOWER(first_name_bdr) LIKE _utf8'%" .
utf8_encode($data) . "%' collate utf8_bin
          OR LOWER(last_name_bdr) LIKE _utf8'%" .
utf8_encode($data) . "%' collate utf8_bin
          OR LOWER(id_bdr) LIKE _utf8'%" .
utf8_encode($data) . "%' collate utf8_bin)
          LIMIT 10";
      //die($sql);
      $result = $this->db->getAll($sql);
      if ($result) {
        return $result;
      } else {
        return false;
      }
    }

    /**
     * return array data
     * @return Array
     */
    function getArray() {
      $data = array();
      $data['serial_bdr'] = $this->serial_bdr;
      $data['id_bdr'] = $this->id_bdr;
      $data['first_name_bdr'] = $this->first_name_bdr;
      $data['last_name_bdr'] = $this->last_name_bdr;
      $data['mobile1_bdr'] = $this->mobile1_bdr;
      $data['mobile2_bdr'] = $this->mobile2_bdr;
      $data['phone1_bdr'] = $this->phone1_bdr;
      $data['phone2_bdr'] = $this->phone2_bdr;
      $data['phone3_bdr'] = $this->phone3_bdr;
      $data['address_bdr'] = $this->address_bdr;
      $data['drive_license_bdr'] = $this->drive_license_bdr;
      $data['drive_license_image_bdr'] = $this-
>drive_license_image_bdr;
      $data['license_expire_date_bdr'] = $this-
>license_expire_date_bdr;
      $data['status_bdr'] = $this->status_bdr;

      return $data;
    }
    /**
     * get an array of available bus drivers
     * @param: $serial_bdr->serial of the Driver (Optional
parameter)
     * @return true if there are some bus drivers; false if
there are no drivers
     */
    function getAvailableBusDrivers($serial_bdr = FALSE) {
      $sql = "SELECT
serial_bdr,id_bdr,CONCAT(last_name_bdr,'
',first_name_bdr) as
```

```
full_name_bdr,DATE_FORMAT(license_expire_date_bdr,
'%d/%m/%Y') as license_expire_date_bdr
            FROM bus_driver
            WHERE status_bdr='ACTIVE'";
      $sql.=$serial_bdr ? " AND serial_bdr NOT IN
({$serial_bdr})" : "";
      $sql.=" ORDER BY last_name_bdr, first_name_bdr";

//  echo $sql;
      $result = $this->db->getAll($sql);
      if ($result) {
         return $result;
      } else {
         return false;
      }
   }

   /**
    * Retrieve true if the bus_driver is already used in the
selected shift
    * @param: $serial_scy->School Year Serial; $serial_sft-
>Shift Serial
    * @return Array
    */
   function getUsedBusDriverByShift($serial_scy,
$serial_sft) {
      $sql = "SELECT
bdr.serial_bdr,id_bdr,CONCAT(first_name_bdr,'
',last_name_bdr) as
full_name_bdr,DATE_FORMAT(license_expire_date_bdr,
'%d/%m/%Y') as license_expire_date_bdr,bsy.serial_rot
            FROM bus_driver bdr
            JOIN bus_shift_year bsy ON
bsy.serial_bdr=bdr.serial_bdr
            WHERE bdr.status_bdr='ACTIVE'
            AND bsy.serial_scy='{$serial_scy}'
            AND bsy.serial_sft='{$serial_sft}'
            AND bsy.status_bpy= 'ACTIVE'";

      $result = $this->db->getAll($sql);
      if ($result) {
         return $result;
      } else {
         return false;
      }
   }
   /**
    * Retrieve true if the  replacement bus_driver is already
used in the selected shift
    * @param: $serial_scy->School Year Serial; $serial_sft-
>Shift Serial
    * @return Array
    */
   function
getUsedReplacementBusDriverByShift($serial_scy,
$serial_sft) {
      $sql = "SELECT
bdr.serial_bdr,id_bdr,CONCAT(first_name_bdr,'
',last_name_bdr) as
full_name_bdr,DATE_FORMAT(license_expire_date_bdr,
'%d/%m/%Y') as license_expire_date_bdr,bsy.serial_rot
            FROM bus_driver bdr
            JOIN bus_shift_year bsy ON
bsy.bus_serial_bdr=bdr.serial_bdr
            WHERE bdr.status_bdr='ACTIVE'
            AND bsy.serial_scy='{$serial_scy}'
            AND bsy.serial_sft='{$serial_sft}'
            AND bsy.status_bpy= 'ACTIVE'";

      $result = $this->db->getAll($sql);
      if ($result) {
         return $result;
      } else {
         return false;
      }
   }

}

?>
```

fNewBusDrivers.php

```php
<?php

/*
  File: fNewBusDriver
  Author: Alex Chávez
  Creation Date: 06/02/2013
 */

Request::setInteger('0:success');

$statusList = Bus_Driver::$status;

$smarty->register('statusList');
//check if there are errors to show  and register Error or Success messages
$error_messages = Error_Handler::checkErrors();
if ($error_messages) {
    $smarty->register('error_messages');
} elseif ($success) {
    $smarty->register('success');
}
$smarty->display();
?>
```

fNewBusDrivers.js

```javascript
/*
  File: fNewBusDriver
  Author: Alex Chávez
  Creation Date: 06/02/2013
 */

$(document).ready(function(){
    $('#btnRegister').bind("click", function(){
        $('#messageContainer').html('');
    });
    /*VALIDATION SECTION*/
    $("#frmNewBusDriver").validate({
        errorLabelContainer: "#alerts",
        wrapper: "li",
        onfocusout: false,
        onkeyup: false,
        rules: {
            txtId: {
                required: true,
                alphaNumeric: true,
                remote: {
                    url:
document_root+"rpc/remoteValidation/bus_driver/checkBusDriverId.rpc",
                    type: "POST"
                }
            },
            txtFirstName: {
                required: true,
                textOnly: true
            },
            txtLastName: {
                required: true,
                textOnly: true
            },
            txtMobile1: {
                required: true
            },
            txtPhone1: {
                required: true
            },
            txtAddress: {
                required: true
            },
            txtDriverLicense: {
                required: true,
                alphaNumeric: true
            },
            txtLicenseExpireDate: {
                required: true,
                date: true,
```

```
                dateMoreThan: function(){
                    var curDate=new Date();
                    curDate.setDate(curDate.getDate()-1);
                    return dateFormat(curDate);
                }
            }
        },
        messages: {
            txtId: {
                alphaNumeric: 'El campo "Id" admite s&oacute;lo caracteres alfa-
num&eacute;ricos.',
                remote: 'Ya existe un Conductor de Bus con esa identificaci&oacute;n.'
            },
            txtFirstName: {
                textOnly: 'El campo "Nombre" solo admite texto.'
            },
            txtLastName: {
                textOnly: 'El campo "Apellido" solo admite texto.'
            },
            txtDriverLicense:{
                alphaNumeric: 'El campo "Tipo de Licencia de Conducci&oacute;n" admite
s&oacute;lo caracteres alfa-num&eacute;ricos.'
            },
            txtLicenseExpireDate: {
                date: 'El campo "Fecha de Expiraci&oacute;n de la Licencia" debe tener
el formato dd/mm/YYYY.',
                dateMoreThan: 'La "Fecha de Expiraci&oacute;n de la Licencia" no puede
ser menor a la fecha actual.'
            }
        },
        submitHandler: function(form) {
            $('#btnRegister').attr('disabled', 'true');
            $('#btnRegister').val('Espere por favor...');
            show_ajax_loader();
            form.submit();
        }
    });
    /*END VALIDATION SECTION*/
    //CALENDAR
    setDefaultCalendar($("#txtLicenseExpireDate"),'+0y','+100y','+0d');
});
```

fNewBusDrivers.es.tpl

```
{assign var="title" value="M&Oacute;DULO DE TRANSPORTES - NUEVO CONDUCTOR DE BUS"}
<form name="frmNewBusDriver" id="frmNewBusDriver" method="post"
action="{$document_root}modules/bus_driver/pNewBusDriver">
    <div class="span-24 last title">
        M&Oacute;DULO DE TRANSPORTES - Registro de Nuevo Conductor de Bus<br />
        <label>(*) Campos Obligatorios</label>
    </div>
    {if $error_messages OR $success}
        <div class="append-7 span-10 prepend-7 last " id="messageContainer">
            {if $success}
                <div class="span-10 success" align="center">
                    El Conductor de Bus se ha registrado exitosamente!<br/>
                </div>
            {else}
                <div class="span-10 error" align="center">
                    <ul>
                        {foreach from=$error_messages item="em"}
                            <li>{$em}</li>
                        {/foreach}
                    </ul>
                </div>
            {/if}
        </div>
    {/if}
    <div class="prepend-1 span-22 last line">
        <div class="span-22 last line separator">
            <label>Datos del Conductor:</label>
        </div>
    </div>
    <div class="span-24 last line" id="imageContainer">
        <div class="prepend-1">
            {upload_image type = bus_driver button_name='Elegir Imagen Licencia' default
= 'lic.png'}
```

```
        </div>
    </div>
    <div class="span-24 last line">

        <div class="prepend-2 span-4 label">* Identificaci&oacute;n:</div>
        <div class="span-5 ">
            <input type="text" name="txtId" id="txtId" title='El campo
"Identificaci&oacute;n" es obligatorio.' class="span-5"/>
        </div>
    </div>

    <div class="span-24 last line">
        <div class="prepend-2 span-4 label">* Nombre:</div>
        <div class="span-5 ">
            <input type="text" name="txtFirstName" id="txtFirstName" title='El campo
"Nombre" es obligatorio.' class="span-5"/>
        </div>

        <div class="prepend-2 span-4 label">* Apellido:</div>
        <div class="span-5">
            <input type="text" name="txtLastName" id="txtLastName" title='El campo
"Apellido" es obligatorio.' class="span-5"/>
        </div>
    </div>

    <div class="span-24 last">
        <div class="prepend-2 span-4 line label">* Tel&eacute;fono Celular 1:</div>
        <div class="span-5">
            <input type="text" name="txtMobile1" id="txtMobile1" title='El campo
"Tel&eacute;fono Celular 1" es obligatorio.' class="span-5"/>
        </div>
        <div class="prepend-2 span-4 label">Tel&eacute;fono Celular 2:</div>
        <div class="span-5">
            <input type="text" name="txtMobile2" id="txtMobile2" title='El campo
"Tel&eacute;fono Celular 2" es obligatorio.' class="span-5"/>
        </div>
    </div>

    <div class="span-24 last">
        <div class="prepend-2 span-4 line label">* Tel&eacute;fono 1:</div>
        <div class="span-5">
            <input type="text" name="txtPhone1" id="txtPhone1" title='El campo
"Tel&eacute;fono 1" es obligatorio.' class="span-5"/>
        </div>
        <div class="prepend-2 span-4 label">Tel&eacute;fono 2:</div>
        <div class="span-5">
            <input type="text" name="txtPhone2" id="txtPhone2" title='El campo
"Tel&eacute;fono 2" es obligatorio.' class="span-5"/>
        </div>
    </div>

    <div class="span-24 last line">
        <div class="prepend-2 span-4 label">Tel&eacute;fono 3:</div>
        <div class="span-5">
            <input type="text" name="txtPhone3" id="txtPhone3" title='El campo
"Tel&eacute;fono 3" es obligatorio.' class="span-5"/>
        </div>
        <div class="prepend-2 span-4 label">* Direcci&oacute;n:</div>
        <div class="span-5">
            <textarea class="span-5" name="txtAddress" id="txtAddress" title="El campo
'Direcci&oacute;n' es obligatorio" class="span-5"></textarea>
        </div>
    </div>

    <div class="span-24 ">
        <div class="prepend-2 span-4 label">* Tipo de Licencia de
Conducci&oacute;n:</div>
        <div class="span-5">
            <input type="text" name="txtDriverLicense" id="txtDriverLicense" title='El
campo "Tipo de Licencia de Conducci&oacute;n" es obligatorio.' class="span-5"/>
        </div>
        <div class="prepend-2 span-4 label">* Fecha de Expiraci&oacute;n de la
Licencia:</div>
        <div class="span-6">
```

```
            <input type="text" name="txtLicenseExpireDate" id="txtLicenseExpireDate"
title='El campo "Fecha de Expiraci&oacute;n de la Licencia" es obligatorio.'
class="span-5"/>
        </div>
    </div>
    <div class="span-24  line">
        <div class="prepend-1 span-22 last line">
            <div class="line separator span-22" align="left">
            </div>
        </div>
    </div>
    <div class="span-24 last buttons">
        <input type="submit" name="btnRegister" id="btnRegister" value="Registrar" />
    </div>
</form>

<div class="span-7 span-10 prepend-7 last">
    <ul id="alerts" class="alerts"></ul>
</div>
```

pNewBusDrivers.php

```
<?php

/*
  File: pNewBusDriver
  Author: Alex Chavez
  Creation Date:06/02/2013
 */

Request::setString('txtId,txtFirstName,txtLastName,txtMobile1,txtMobile2,txtPhone1,txtPh
one2,txtPhone3,txtAddress,txtDriverLicense,txtLicenseExpireDate');
try { //creates a new bus driver
    $bus_driver = new Bus_Driver($db);
    $bus_driver->setIdBdr(specialchars($txtId));
    $bus_driver->setFirstNameBdr(specialchars($txtFirstName));
    $bus_driver->setLastNameBdr(specialchars($txtLastName));
    $bus_driver->setMobile1Bdr(specialchars($txtMobile1));
    $bus_driver->setMobile2Bdr(specialchars($txtMobile2));
    $bus_driver->setPhone1Bdr(specialchars($txtPhone1));
    $bus_driver->setPhone2Bdr(specialchars($txtPhone2));
    $bus_driver->setPhone3Bdr(specialchars($txtPhone3));
    $bus_driver->setAddressBdr(specialchars($txtAddress));
    $bus_driver->setDriveLicenseBdr(specialchars($txtDriverLicense));
    $bus_driver-
>setDriveLicenseImageBdr($_SESSION['rutademic_serial_usr']['bus_driver']['image_name']);
    $bus_driver->setLicenseExpireDateBdr(specialchars($txtLicenseExpireDate));
    $bus_driver->setStatusBdr('ACTIVE');
    $bus_driver->save();
    $success = 1;

    unset($_SESSION['rutademic_serial_usr']['bus_driver']['image_name']);
} catch (Exception $e) {
    Error_Handler::registerError($e->getMessage(), 'BUS_DRIVER');
}
$toRedirect = ($success == 1) ? "/$success" : "";
http_redirect('modules/bus_driver/fNewBusDriver' . $toRedirect);
?>
```

checkBusDriverId.rpc

```
<?php
/*
  File: checkBusDriverId.rpc
  Author: Alex Chávez
  Creation Date: 06/02/2013
 */
$bus_driver = new Bus_Driver($db);
Request::setString('serial_bdr');
Request::setString('txtId');
$txtId = trim(strtolower(utf8_decode($txtId)));
if ($bus_driver->checkBusDriver($txtId, $serial_bdr)) {
    echo json_encode(false);
} else {
    echo json_encode(true);
}
?>
```

3.2 IMPLEMENTACIÓN DEL REPORTE DE RUTAS BASADO EN DATOS DEL GPS.

Esta interfaz es usada por los padres de familia para ver un reporte, que basándose en los datos de sus representados despliega información sobre las rutas configuradas para el estudiante seleccionado.

Como complemento, este reporte hace uso de la versión 3 del API[27] de Google Maps[28] para representar los datos del GPS incorporado en cada bus en un mapa que puede ser visto desde un computador con conexión a internet o desde cualquier teléfono celular que soporte navegación web por medio de HTML.

Este reporte funciona obligatoriamente con conexión a internet, ya que todos los datos del mapa son cargados desde los servidores del Google; la ventaja de esta carga remota es que nos permite tener los datos del mapa actualizados y además contar con complementos como navegación en 3D (tres dimensiones) y diversos tipos de mapas como satelital, de elevaciones, etc.

Al igual que la implementación anterior este reporte cuenta con una arquitectura de tres capas, pero además hace uso de *Web Services*[29] para hacer cargas remotas a servidores publicados en internet, los archivos usados son:

Nombre Archivo	Descripción
Bus_Shift_Year.class.php	Es una clase de php en cuál se maneja todos los atributos que puede tener una ruta configurada para un año en específico, además todos los métodos que se requieren para que la interfaz trabaje de una forma adecuada junto con la base de datos. Esta clase es la conexión que se tiene entre las interfaces y la base de datos, específicamente con la tabla bus_shift_year.
Bus_Year_User.class.php	Es una clase de php en la cuál se maneja todos los usuarios que puede tener una ruta configurada para un año en específico, es la conexión que se tiene entre las interfaces y la base de datos, específicamente con la tabla bus_year_year.
fGuardianBusMapReview.js	Este es el archivo JavaScript que maneja todas las validaciones y cargas remotas de la interfaz; además en este caso es el archivo más importante ya que hace la llamada a los métodos del API de Google Maps para hacer la carga del mapa representando los datos obtenidos con el GPS.
fGuardianBusMapReview.php	Es el archivo que nos hace la pre-carga de datos para

[27] **Interfaz de programación de aplicaciones** (**IPA**) o API (del inglés *Application Programming Interface*) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usadas generalmente en las bibliotecas.

[28] **Google Maps** es el nombre de un servicio gratuito de Google. Es un servidor de aplicaciones de mapas en la Web. Ofrece imágenes de mapas desplazables, así como fotos satelitales del mundo e incluso la ruta entre diferentes ubicaciones o imágenes a pie de calle Google Street View.

[29] Un **servicio web** (en inglés, *Web services*) es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet.

	empezar el uso de la interfaz, en este caso se carga los años lectivos disponibles y además valida que el perfil que haya iniciado sesión sea un padre de familia. Una vez que todo esté validado se pasa al ingreso de datos.
fGuardianBusMapReview.es.tpl	Es la interfaz gráfica de la pantalla; en la cual se define los contenedores para los filtros, los datos y para el mapa.
loadDriverInfo.rpc.php	Es un archivo de carga remota para obtener todos los datos del conductor.
loadStudentsBySchoolYear.rpc.php	Es un archivo de carga remota para obtener todos los estudiantes basándose en el año lectivo y en el perfil que inició sesión.
loadTransportationInfo.rpc.php	Es un archivo de carga remota para obtener todos los datos de la ruta configurada.
guardian.es.tpl	Es el contenedor principal para los padres de familia, en este caso se lo hace mención ya que es aqui donde se hace al llamado o conexión con el servidor del API de Google para hacer uso de los mapas.

Título: Tabla 3
Descripción: Archivos Implementación del reporte de rutas basado en datos del GPS
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

A continuación se presenta el código fuente de los archivos antes descritos:

Bus_Shift_Year.class.php

```php
<?php
/*
File: Bus_Shift_Year.class
Created on : 22/02/2013
Author     : Alex Chavez
*/

Class Bus_Shift_Year {
    private $db;
    /**
     * School Year serial
     * @var uInt
     */
    private $serial_scy;
    /**
     * Bus serial
     * @var uInt
     */
    private $serial_bus;
    /**
     * Shift serial
     * @var uInt
     */
    private $serial_sft;
    /**
     * Route serial
     * @var uInt
     */
    private $serial_rot;
    /**
     * Bus Driver Serial
     * @var uInt
     */
    private $serial_bdr;
    /**
     * Bus Driver Replacement Serial
     * @var uInt
     */
    private $bus_serial_bdr;
    /**
     * Bus Shift Year status
     * @var Enum
     */
    private $status_bpy;
    public static
$status=array('ACTIVE'=>'Activo','INACTIVE'=>'Inactivo');

    public function __construct( $db,
$serial_scy = 0, $serial_bus = 0,
$serial_sft = 0, $serial_rot = 0 )
    {
        $this->db = $db;
        $this->serial_scy = $serial_scy;
        $this->serial_bus = $serial_bus;
        $this->serial_sft = $serial_sft;
        $this->serial_rot = $serial_rot;
    }
    ###### SET FUNCTIONS ######
    /**
     * @param uInt $serial_scy
     */
    public function setSerialScy(
$serial_scy )
    {
        $this->serial_scy = $serial_scy;
    }
    /**
     * @param uInt $serial_bus
     */
    public function setSerialBus(
$serial_bus )
    {
        $this->serial_bus = $serial_bus;
```

```
    }
    /**
     * @param uInt $serial_sft
     */
    public function setSerialSft(
$serial_sft )
    {
        $this->serial_sft = $serial_sft;
    }
    /**
     * @param uInt $serial_rot
     */
    public function setSerialRot(
$serial_rot )
    {
        $this->serial_rot = $serial_rot;
    }
    /**
     * @param uInt $serial_bdr
     */
    public function setSerialBdr(
$serial_bdr )
    {
        $this->serial_bdr = $serial_bdr;
    }
    /**
     * @param uInt $bus_serial_bdr
     */
    public function setBusSerialBdr(
$bus_serial_bdr )
    {
        $this->bus_serial_bdr =
$bus_serial_bdr;
    }
    /**
     * @param ENUM $status_bpy
     */
    public function setStatusBpy(
$status_bpy )
    {
        if(
!array_key_exists($status_bpy,
self::$status) )
        throw new Exception(
'INVALID_BUS_SHIFT_YEAR_STATUS' );

        $this->status_bpy = $status_bpy;
    }

       ###### GET FUNCTIONS ######
    /**
     * @return uInt
     */
    public function getSerialScy()
    {
    return $this->serial_scy;
    }
    /**
     * @return uInt
     */
    public function getSerialBus()
    {
    return $this->serial_bus;
    }
    /**
     * @return uInt
     */
    public function getSerialSft()
    {
    return $this->serial_sft;
    }
    /**
     * @return uInt
     */
    public function getSerialRot()
    {
    return $this->serial_rot;
    }
    /**
     * @return uInt
     */
    public function getSerialBdr()
    {
    return $this->serial_bdr;
    }
    /**
     * @return uInt
     */
    public function getBusSerialBdr()
    {
    return $this->bus_serial_bdr;
    }
    /**
     * @return ENUM
     */
    public function getStatusBpy()
    {
    return $this->status_bpy;
    }

    ###### PRIVATE FUNCTIONS ######
    /**
     * load all data from an object
     **/
    private function load()
    {
        $sql= "SELECT status_byu
               FROM bus_shift_year
               WHERE serial_scy='{$this-
>getSerialScy()}'
               AND
serial_serial_rot='{$this-
>getSerialRot()}'
               AND serial_sft='{$this-
>getSerialSft()}'
               AND serial_bus='{$this-
>getSerialBus()}'";

        $result = $this->db->getRow(
$sql );

        if( $result === FALSE )
            throw new Exception(
"INVALID_BUS_SHIFT_YEAR");
    }
    ###### PUBLIC FUNCTIONS ######

    /**
     * saves a bus shift year in
database
     * @throws errors when insert or
update
     */
    public function save()
    {
        //if($this->getSerialScy() &&
$this->getSerialBus() && $this-
>getSerialSft() && $this->getSerialRot()
&& $this->getSerialBdr()){
        if($this->existBusShiftYear()){
            //UPDATE
            $sql = "UPDATE
bus_shift_year
                    SET
                    serial_bdr =
'{$this->getSerialBdr()}',
                    bus_serial_bdr = ";
            $sql.=$this-
>getBusSerialBdr()?"'{$this-
>getBusSerialBdr()}',":"NULL,";
            $sql.=" status_bpy =
'{$this->getStatusBpy()}'
```

```
                    WHERE  serial_scy =
'{$this->getSerialScy()}'
                    AND    serial_sft =
'{$this->getSerialSft()}'
                    AND    serial_rot =
'{$this->getSerialRot()}'
                    AND    serial_bus =
'{$this->getSerialBus()}'";

            $result = $this->db-
>Execute( $sql );
            if ( $result === false )
                throw new
Exception('NO_UPDATE_BUS_SHIFT_YEAR');
        }
        else{
            //INSERT
            $sql1 = "INSERT INTO
bus_shift_year
                        (serial_scy,
                         serial_bus,
                         serial_sft,
                         serial_rot,
                         serial_bdr,
                         bus_serial_bdr,
                         status_bpy)
                        VALUES
                        (
                            '{$this-
>getSerialScy()}',
                            '{$this-
>getSerialBus()}',
                            '{$this-
>getSerialSft()}',
                            '{$this-
>getSerialRot()}',
                            '{$this-
>getSerialBdr()}',";
                $sql1.=$this-
>getBusSerialBdr()?"'{$this-
>getBusSerialBdr()}',":"NULL,";
                $sql1.="    '{$this-
>getStatusBpy()}')";

                $result1 = $this->db-
>Execute( $sql1 );
                if( $result1 == true ){
                    return true;
                }else{
                    throw new
Exception('NO_INSERT_BUS_SHIFT_YEAR');
                }
        }
    }
    /**
     * Retrieve objerct data as array
     * @return Array
     */
    function getArray()
    {
        $data = array();
        $data['serial_scy'] = $this-
>serial_scy;
        $data['serial_bus'] = $this-
>serial_bus;
        $data['serial_sft'] = $this-
>serial_sft;
        $data['serial_rot'] = $this-
>serial_rot;
        $data['serial_bdr'] = $this-
>serial_bdr;
        $data['bus_serial_bdr'] = $this-
>bus_serial_bdr;
        $data['status_bpy'] = $this-
>status_bpy;
        return $data;
    }
```

```
    /**
     * Retrieve true if the bus_shift
exist or false ift it not exist yet
     * @return Array
     */
    function existBusShiftYear()
    {
        $sql="SELECT 1
              FROM bus_shift_year
              WHERE serial_scy='{$this-
>getSerialScy()}'
              AND serial_sft='{$this-
>getSerialSft()}'
              AND serial_rot='{$this-
>getSerialRot()}'";
        $sql.=$this->getSerialBus()?"AND
serial_bus='{$this-
>getSerialBus()}'":"";

        $result = $this->db ->
getOne($sql);
        if($result){
            return true;
        }
        else{
            return false;
        }
    }

    /**
     * Retrieve an array if the
bus_shift exist or false if it not exist
yet
     * @return Array
     */
    function getDataBusShiftYear()
    {
        $sql="SELECT
bsy.serial_bdr,bsy.serial_bus,bsy.bus_se
rial_bdr,CONCAT(bdr.first_name_bdr,'
',bdr.last_name_bdr) as full_name_bdr,
IFNULL(CONCAT(bdr2.first_name_bdr,'
',bdr2.last_name_bdr),'N/A') as
full_name_bdr2, bus.code_bus
              FROM bus_shift_year bsy
              JOIN bus_driver bdr ON
bdr.serial_bdr=bsy.serial_bdr
              LEFT JOIN bus_driver bdr2
ON bdr2.serial_bdr=bsy.bus_serial_bdr
              JOIN bus bus ON
bus.serial_bus=bsy.serial_bus
              WHERE serial_scy='{$this-
>getSerialScy()}'
              AND serial_sft='{$this-
>getSerialSft()}'
              AND serial_rot='{$this-
>getSerialRot()}'
              AND status_bpy='ACTIVE'";
        $sql.=$this->getSerialBus()?"AND
serial_bus='{$this-
>getSerialBus()}'":"";

        $result = $this->db ->
getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }

    /**
     * Retrieve an array with the
serials of the bus, shift, route, and
drivers
     * @return Array
```

```
     */
    function getInfoBusShiftYear()
    {
        $sql="SELECT
serial_scy,serial_bus,serial_sft,serial_
rot,serial_bdr,bus_serial_bdr,status_bpy
                FROM bus_shift_year
                WHERE serial_scy='{$this-
>getSerialScy()}'
                AND serial_sft='{$this-
>getSerialSft()}'
                AND serial_rot='{$this-
>getSerialRot()}'
                AND status_bpy='ACTIVE'";

        $result = $this->db ->
getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }

    /**
     * Retrieve an array if the
bus_driver is already used in the
selected shift
     * @return Array
     */
    function checkUsedBusDriver()
    {
        $sql="SELECT
serial_scy,serial_bus,serial_sft,serial_
rot,serial_bdr,bus_serial_bdr,status_bpy
                FROM bus_shift_year
                WHERE serial_scy='{$this-
>getSerialScy()}'
                AND serial_sft='{$this-
>getSerialSft()}'
                AND (serial_bdr='{$this-
>getSerialBdr()}'
                OR bus_serial_bdr='{$this-
>getSerialBdr()}')
                AND status_bpy= 'ACTIVE'";

        $result = $this->db ->
getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }
    /**
     * Retrieve an array if the bus is
already used in the selected shift
     * @return Array
     */
    function checkUsedBus()
    {
        $sql="SELECT
serial_scy,serial_bus,serial_sft,serial_
rot,serial_bdr,bus_serial_bdr,status_bpy
                FROM bus_shift_year
                WHERE serial_scy='{$this-
>getSerialScy()}'
                AND serial_sft='{$this-
>getSerialSft()}'
                AND serial_bus='{$this-
>getSerialBus()}'
                AND status_bpy= 'ACTIVE'";

        $result = $this->db ->
getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }
    /**
     * Retrieve an array with all
configured routes information of the
selected year
     * @params serial_scy
     * @return Array
     */
    function
getConfiguredTransportation($serial_scy)
{
        $sql="  SELECT
rot.code_rot,sft.name_sft,CONCAT(bdr.las
t_name_bdr,' ',bdr.first_name_bdr) as
full_name_bdr,
IFNULL(CONCAT(bdr2.last_name_bdr,'
',bdr2.first_name_bdr),'N/A') as
full_name_bdr2, bus.code_bus
                FROM bus_shift_year bsy
                JOIN route rot ON
rot.serial_rot= bsy.serial_rot
                JOIN shift sft ON
sft.serial_sft=bsy.serial_sft
                JOIN bus_driver bdr ON
bdr.serial_bdr=bsy.serial_bdr
                LEFT JOIN bus_driver
bdr2 ON
bdr2.serial_bdr=bsy.bus_serial_bdr
                JOIN bus bus ON
bus.serial_bus=bsy.serial_bus
                WHERE bsy.status_bpy =
'ACTIVE'
                AND
bsy.serial_scy='{$serial_scy}'
                ORDER BY  full_name_bdr
ASC, full_name_bdr2 ASC, rot.code_rot
ASC,sft.name_sft ASC";

        $result = $this->db ->
getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }

    /**
     * Gets bus shifts by by year
     * @param $serial_scy
     * @return bus shift array or false
if it doesn't exist
     * @author Santiago Benitez
     */
    function
getBusShiftByYear($serial_scy){
        $sql = "SELECT *
                FROM bus_shift_year
                WHERE
status_bpy='ACTIVE'
                AND
serial_scy=$serial_scy";
        //die($sql);
        $result= $this->db-
>getAll($sql);

        if($result){
            return $result;
        }else{
            return false;
```

```
        }
    }

    /**
     * Deactivates previous
configuration in Qualitative Equivalence
in the DB
     * @param $serial_scy
     * @throws The Qualitative
Equivalence isn't inserted
     * @author Santiago Benitez
     */
    public function
deactivatePreviousConfigurationByYear($s
erial_scy){
        //UPDATE
        $sql = "UPDATE bus_shift_year
                SET status_bpy =
'INACTIVE'
                WHERE status_bpy =
'ACTIVE'
                AND serial_scy =
$serial_scy";

        $result = $this->db->Execute(
$sql );
        if ( $result === false )
            throw new
Exception('NO_UPDATE_BUS_SHIFT_YEAR');
    }

    /**
     * Retrieve all configured
transportation Shifts
     * @params serial_scy
     * @return Array
     */
    function
getConfiguredTransportationForAssingToUs
ers($serial_scy){
        $sql="  SELECT DISTINCT
rot.serial_rot,
                sft.serial_sft,
                rot.code_rot,
                sft.name_sft,

CONCAT(bdr.first_name_bdr,'
',bdr.last_name_bdr) as full_name_bdr,

IFNULL(CONCAT(bdr2.first_name_bdr,'
',bdr2.last_name_bdr),'N/A') as
full_name_bdr2,
                bus.code_bus,
                bus.serial_bus
                FROM bus_shift_year bsy
                JOIN route rot ON
rot.serial_rot= bsy.serial_rot
                JOIN shift sft ON
sft.serial_sft=bsy.serial_sft
                JOIN bus_driver bdr ON
bdr.serial_bdr=bsy.serial_bdr
                LEFT JOIN bus_driver
bdr2 ON
bdr2.serial_bdr=bsy.bus_serial_bdr
                JOIN bus bus ON
bus.serial_bus=bsy.serial_bus
                WHERE bsy.status_bpy =
'ACTIVE'
                AND
bsy.serial_scy='{$serial_scy}'
                ORDER BY  rot.code_rot
ASC,sft.name_sft ASC,full_name_bdr ASC";



        $result = $this->db ->
getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }

    /**
     * Retrieve all data from configured
transportation Shifts
     * @param $serial_scy->School Year
serial; $serial_rot->Route Serial;
$serial_sft->Shift Serial; $serial_bus-
>Bus Serial
     * @return Array
     */
    function
getAllDataFromConfiguredTransportation($
serial_scy, $serial_rot, $serial_sft,
$serial_bus){
        $sql="  SELECT  rot.serial_rot,
sft.serial_sft,bus.serial_bus,
bdr.serial_bdr, bsy.bus_serial_bdr,
rot.code_rot,sft.name_sft,CONCAT(bdr.fir
st_name_bdr,' ',bdr.last_name_bdr) as
full_name_bdr,
IFNULL(CONCAT(bdr2.first_name_bdr,'
',bdr2.last_name_bdr),'N/A') as
full_name_bdr2, bus.code_bus
                FROM bus_shift_year bsy
                JOIN route rot ON
rot.serial_rot= bsy.serial_rot
                JOIN shift sft ON
sft.serial_sft=bsy.serial_sft
                JOIN bus_driver bdr ON
bdr.serial_bdr=bsy.serial_bdr
                LEFT JOIN bus_driver
bdr2 ON
bdr2.serial_bdr=bsy.bus_serial_bdr
                JOIN bus bus ON
bus.serial_bus=bsy.serial_bus
                WHERE bsy.status_bpy =
'ACTIVE'
                AND
bsy.serial_scy='{$serial_scy}'
                AND
rot.serial_rot='{$serial_rot}'
                AND
sft.serial_sft='{$serial_sft}'
                AND
bus.serial_bus='{$serial_bus}'
                ORDER BY  rot.code_rot
ASC,sft.name_sft ASC,full_name_bdr ASC";

        $result = $this->db ->
getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }
    /**
     * Retrieve true if the status of
the Route is changed succesfully. If not
retrieve false
     * @return Array
     */
    function
deactivateAllInfoByRouteAndShift(){
        $sql= "UPDATE bus_shift_year
                SET status_bpy =
'INACTIVE'
                WHERE serial_scy =
'{$this->getSerialScy()}'
```

```
                AND serial_sft = '{$this->getSerialSft()}'
                AND serial_rot = '{$this->getSerialRot()}'";

        $result = $this->db -> Execute($sql);
        if($result){
            return true;
        }
        else{
            return false;
        }

    }
    /**
     * Returns all shifts for the selected Route in the selected Year
     * @return Array
     */
    function getShiftsByRouteByYear(){
        $sql= " SELECT sft.serial_sft,sft.name_sft
                FROM bus_shift_year bsy
                JOIN shift sft ON sft.serial_sft=bsy.serial_sft AND sft.status_sft='ACTIVE'
                WHERE bsy.serial_scy={$this->getSerialScy()}
                AND bsy.serial_rot={$this->getSerialRot()}";

        $result = $this->db -> getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }
}
?>
```

Bus_Year_User.class.php

```
<?php
/*
File: Bus_Year_User.class
Created on : 22/02/2013
Author     : Alex Chavez
*/

Class Bus_Year_User {
    private $db;
    /**
     * User serial
     * @var uInt
     */
    private $serial_usr;
    /**
     * School Year serial
     * @var uInt
     */
    private $serial_scy;
    /**
     * Bus serial
     * @var uInt
     */
    private $serial_bus;
    /**
     * Shift serial
     * @var uInt
     */
    private $serial_sft;
    /**
     * Route serial
     * @var uInt
     */
    private $serial_rot;
    /**
     * Bus Shift Year status
     * @var Enum
     */
    private $status_byu;
    public static $status=array('ACTIVE'=>'Activo','INACTIVE'=>'Inactivo');

    /**
     * Bus Year User Type Address
     * @var Enum
     */
    private $type_address_byu;
    public static $typeAddress=array('HOME'=>'Domicilio','OTHER'=>'Otro');

  /**
     * Bus Year User address
     * @var String
     */
    private $address_byu;

    public function __construct( $db, $serial_usr = 0, $serial_scy = 0, $serial_bus = 0, $serial_sft = 0, $serial_rot = 0)
    {
        $this->db = $db;
        $this->serial_usr = $serial_usr;
        $this->serial_scy = $serial_scy;
        $this->serial_bus = $serial_bus;
        $this->serial_sft = $serial_sft;
        $this->serial_rot = $serial_rot;
        if($serial_usr && $serial_scy && $serial_bus && $serial_sft && $serial_rot)
            $this->load();
    }
    ###### SET FUNCTIONS ######
    /**
     * @param uInt $serial_usr
     */
    public function setSerialUsr( $serial_usr )
    {
        $this->serial_usr = $serial_usr;
    }
    /**
     * @param uInt $serial_scy
     */
    public function setSerialScy( $serial_scy )
    {
        $this->serial_scy = $serial_scy;
    }
    /**
     * @param uInt $serial_bus
     */
    public function setSerialBus( $serial_bus )
    {
        $this->serial_bus = $serial_bus;
```

```
    }
    /**
     * @param uInt $serial_sft
     */
    public function setSerialSft(
$serial_sft )
    {
        $this->serial_sft = $serial_sft;
    }
    /**
     * @param uInt $serial_rot
     */
    public function setSerialRot(
$serial_rot )
    {
        $this->serial_rot = $serial_rot;
    }
    /**
     * @param ENUM $status_bpy
     */
    public function setStatusBpy(
$status_byu )
    {
        if(
!array_key_exists($status_byu,
self::$status) )
        throw new Exception(
'INVALID_BUS_YEAR_USER_STATUS' );

        $this->status_byu = $status_byu;
    }

        /**
     * @param ENUM $type_address_byu
     */
    public function setTypeAddressByu(
$type_address_byu )
    {
        if(
!array_key_exists($type_address_byu,
self::$typeAddress) )
        throw new Exception(
'INVALID_BUS_YEAR_USER_TYPE_ADDRESS' );

        $this->type_address_byu =
$type_address_byu;
    }

     /**
     * @param String $address_byu
     */
    public function
setAddressByu($address_byu) {
        $this->address_byu =
$address_byu;
    }

       ###### GET FUNCTIONS ######
    /**
     * @return uInt
     */
    public function getSerialUsr()
    {
    return $this->serial_usr;
    }
    /**
     * @return uInt
     */
    public function getSerialScy()
    {
    return $this->serial_scy;
    }
    /**
     * @return uInt
     */
    public function getSerialBus()
    {
    return $this->serial_bus;
    }
    /**
     * @return uInt
     */
    public function getSerialSft()
    {
    return $this->serial_sft;
    }
    /**
     * @return uInt
     */
    public function getSerialRot()
    {
    return $this->serial_rot;
    }
    /**
     * @return ENUM
     */
    public function getStatusByu()
    {
    return $this->status_byu;
    }

        /**
     * @return ENUM
     */
    public function getTypeAddressByu()
    {
    return $this->type_address_byu;
    }

      /**
     * @return String
     */
    public function getAddressByu() {
        return $this->address_byu;
    }

    ###### PUBLIC FUNCTIONS ######

    /**
     * saves a bus shift year in
database
     * @throws errors when insert or
update
     */
    public function save()
    {
        if($this->existBusYearUser()){
            //UPDATE
            $sql = "UPDATE bus_year_user
                    SET
                    serial_usr =
'{$this->getSerialUsr()}',
                    serial_scy =
'{$this->getSerialScy()}',
                    serial_bus =
'{$this->getSerialBus()}',
                    serial_sft =
'{$this->getSerialSft()}',
                    serial_rot =
'{$this->getSerialRot()}',
                    status_byu =
'ACTIVE',

type_address_byu = '{$this-
>getTypeAddressByu()}',

address_byu = '" .
addslashes(trim($this->getAddressByu()))
. "'
                    WHERE  serial_scy =
'{$this->getSerialScy()}'
```

```
                    AND    serial_usr =
'{$this->getSerialUsr()}'
                    AND    serial_sft =
'{$this->getSerialSft()}'
                    AND    serial_rot =
'{$this->getSerialRot()}'";

            $result = $this->db-
>Execute( $sql );
            if ( $result === false )
                throw new
Exception('NO_UPDATE_BUS_YEAR_USER');
        }
        else{
            //INSERT
            $sql1 = "INSERT INTO
bus_year_user
                        (serial_usr,
                         serial_scy,
                         serial_bus,
                         serial_sft,
                         serial_rot,
                         status_byu,

type_address_byu,

address_byu)
                        VALUES
                        (
                            '{$this-
>getSerialUsr()}',
                            '{$this-
>getSerialScy()}',
                            '{$this-
>getSerialBus()}',
                            '{$this-
>getSerialSft()}',
                            '{$this-
>getSerialRot()}',
                            'ACTIVE',

'{$this->getTypeAddressByu()}',

'" . addslashes(trim($this-
>getAddressByu())) . "'

)";

                $result1 = $this->db-
>Execute( $sql1 );
                if( $result1 == true ){
                    return true;
                }else{
                    throw new
Exception('NO_INSERT_BUS_YEAR_USER');
                }
        }
    }
    /*
     * Change the status to Inactive
     */
    function delete(){
        $sql = "UPDATE bus_year_user
                    SET
                    Status_byu =
'INACTIVE'
                    WHERE  serial_scy =
'{$this->getSerialScy()}'
                    AND    serial_usr =
'{$this->getSerialUsr()}'
                    AND    serial_sft =
'{$this->getSerialSft()}'
                    AND    serial_rot =
'{$this->getSerialRot()}'";

            $result = $this->db-
>Execute( $sql );
            if ( $result === false )
                throw new
Exception('NO_UPDATE_BUS_YEAR_USER');
    }
    /**
     * Retrieve objerct data as array
     * @return Array
     */
    function getArray()
    {
        $data = array();
        $data['serial_usr'] = $this-
>serial_usr;
        $data['serial_scy'] = $this-
>serial_scy;
        $data['serial_bus'] = $this-
>serial_bus;
        $data['serial_sft'] = $this-
>serial_sft;
        $data['serial_rot'] = $this-
>serial_rot;
        $data['status_byu'] = $this-
>status_byu;

$data['type_address_byu'] = $this-
>type_address_byu;
                $data['address_byu'] =
$this->address_byu;
        return $data;
    }

    /**
     * Retrieve true if the bus_shift
exist or false ift it not exist yet
     * @return Array
     */
    function existActivesBusYearUser()
    {
        $sql="SELECT 1
              FROM bus_year_user
              WHERE serial_scy='{$this-
>getSerialScy()}'
              AND serial_usr='{$this-
>getSerialUsr()}'
              AND serial_sft='{$this-
>getSerialSft()}'
              AND serial_rot='{$this-
>getSerialRot()}'
              AND serial_bus='{$this-
>getSerialBus()}'
              AND status_byu =
'ACTIVE'";
        $result = $this->db ->
getOne($sql);
        if($result){
            return true;
        }
        else{
            return false;
        }
    }
    /**
     * Retrieve true if the bus_shift
exist or false ift it not exist yet
     * @return Array
     */
    function existBusYearUser()
    {
        $sql="SELECT 1
              FROM bus_year_user
              WHERE serial_scy='{$this-
>getSerialScy()}'
              AND serial_usr='{$this-
>getSerialUsr()}'
              AND serial_sft='{$this-
>getSerialSft()}'
```

```
                AND serial_rot='{$this-
>getSerialRot()}'";
        $result = $this->db ->
getOne($sql);
        if($result){
            return true;
        }
        else{
            return false;
        }
    }
    /*
     * load all data from an object
     */
    function load()
    {
        $sql= "SELECT
status_byu,address_byu,type_address_byu
                FROM bus_year_user
                WHERE serial_rot='{$this-
>getSerialRot()}'
                AND serial_scy='{$this-
>getSerialScy()}'
                AND serial_bus='{$this-
>getSerialBus()}'
                AND serial_sft='{$this-
>getSerialSft()}'
                AND serial_usr='{$this-
>getSerialUsr()}'";

        $result = $this->db->getRow(
$sql );

        if( $result === FALSE )
            throw new Exception(
"INVALID_BUS_YEAR_USER");

                $this->address_byu =
$result['address_byu'];
                $this->type_address_byu
= $result['type_address_byu'];

    }
    /**
     * Retrieve true if the bus has
already assigned users
     * @return Array
     */
    function
checkBusHasUsers($serial_bus,$serial_rot
,$serial_sft,$serial_scy){
        $sql="SELECT DISTINCT
byu.serial_bus,byu.serial_rot,byu.serial
_sft,byu.serial_usr, byu.status_byu
              FROM  bus_year_user byu
              WHERE byu.serial_bus =
'{$serial_bus}'
              AND
byu.serial_rot='{$serial_rot}'
              AND
byu.serial_sft='{$serial_sft}'
              AND
byu.serial_scy='{$serial_scy}'
              AND
byu.status_byu='ACTIVE'";

        $result = $this->db->getAll(
$sql );

        if($result){
            return true;
        }
        else{
            return false;
        }
    }
    /*
```

```
     * Retrieve all the Users from a
Route-Shift-Turn.Bus Configuration
     * @return Array
     */
    function
getAllEmployeesUsersFromBusConfiguration
()
    {
        $sql = "SELECT byu.serial_usr,

CONCAT(emp.last_name_emp,' ',
emp.first_name_emp) as full_name_user,
                'N/A' as
full_course_name,
                'Empleado' as user_type,

byu.type_address_byu
                FROM bus_year_user byu
                JOIN user usr ON
byu.serial_usr = usr.serial_usr
                JOIN employee emp ON
usr.serial_emp = emp.serial_emp
                WHERE byu.serial_rot =
'{$this->getSerialRot()}'
                AND byu.serial_scy =
'{$this->getSerialScy()}'
                AND byu.serial_bus =
'{$this->getSerialBus()}'
                AND byu.serial_sft =
'{$this->getSerialSft()}'
                AND byu.status_byu =
'ACTIVE'
                                ORDER BY
emp.last_name_emp, emp.first_name_emp";

        $result = $this->db-
>getAll($sql);

        if($result){
            return $result;
        }
        else{
            return false;
        }
    }

    /*
     * Retrieve all the Users from a
Route-Shift-Turn.Bus Configuration
     * @return Array
     */
    function
getAllStudentsUsersFromBusConfiguration(
)
    {
        $sql = "SELECT
                byu.serial_usr,

CONCAT(std.last_name_std,' ',
std.first_name_std) as full_name_user,
                CONCAT(lvl.name_lvl,'
',clg.name_clg) as course_clg,
                spt.name_spt,
                sec.name_set,
                sec.type_set,
                'Estudiante' as
user_type,

byu.type_address_byu
                FROM bus_year_user byu
                JOIN user usr ON
byu.serial_usr = usr.serial_usr
                JOIN student std ON
usr.serial_std = std.serial_std
                JOIN student_by_course
sbc ON std.serial_std = sbc.serial_std
```

```
                JOIN course crs ON
sbc.serial_crs = crs.serial_crs AND
crs.serial_scy = '{$this-
>getSerialScy()}'
                JOIN class_group clg ON
crs.serial_clg = clg.serial_clg
                JOIN specialty spt ON
crs.serial_spt = spt.serial_spt
                JOIN level lvl ON
crs.serial_lvl = lvl.serial_lvl
                JOIN section sec ON
crs.serial_set = sec.serial_set
                                JOIN
student_year sty ON
std.serial_std=sty.serial_std  AND
sty.serial_scy = '{$this-
>getSerialScy()}' AND sty.status_sty !=
'RETIRED'
                WHERE byu.serial_rot =
'{$this->getSerialRot()}'
                AND byu.serial_scy =
'{$this->getSerialScy()}'
                AND byu.serial_bus =
'{$this->getSerialBus()}'
                AND byu.serial_sft =
'{$this->getSerialSft()}'
                AND byu.status_byu =
'ACTIVE'
                                AND
sbc.status_stc = 'ACTIVE'
                                ORDER BY
std.last_name_std, std.first_name_std";

        $result = $this->db-
>getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }

    /*
     * Return the route assigned to a
student
     * @return Array
     */
    function
getBusRouteByStudentUser($serial_usr,$se
rials)
    {
        $sql="SELECT
byu.serial_scy,scy.name_scy,rot.code_rot
,sft.name_sft,bus.code_bus
                FROM bus_year_user byu
                JOIN user usr ON
byu.serial_usr = usr.serial_usr
                JOIN route rot ON
rot.serial_rot=byu.serial_rot AND
rot.status_rot='ACTIVE'
                JOIN shift sft ON
sft.serial_sft=byu.serial_sft AND
sft.status_sft='ACTIVE'
                JOIN bus bus ON
bus.serial_bus=byu.serial_bus AND
bus.status_bus='ACTIVE'
                JOIN school_year scy ON
scy.serial_scy=byu.serial_scy
                WHERE byu.serial_usr =
{$serial_usr}
                AND byu.serial_scy IN
({$serials})
                AND byu.status_byu =
'ACTIVE'"; //echo $sql;
```

```
        $result = $this->db-
>getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }

    }
     /*
     * Retrieve all the Users from a
Route-Shift-Turn.Bus Configuration
     * @return Array
     */
    function
getBusConfigurationByEmployeeByYear($ser
ial_usr,$serials)
    {
        $sql = "SELECT
byu.serial_usr,CONCAT(emp.first_name_emp
,' ', emp.last_name_emp) as
full_name_user,byu.serial_scy,scy.name_s
cy,rot.code_rot,sft.name_sft,bus.code_bu
s
                FROM bus_year_user byu
                JOIN user usr ON
byu.serial_usr = usr.serial_usr
                JOIN employee emp ON
usr.serial_emp = emp.serial_emp
                JOIN route rot ON
rot.serial_rot=byu.serial_rot AND
rot.status_rot='ACTIVE'
                JOIN shift sft ON
sft.serial_sft=byu.serial_sft AND
sft.status_sft='ACTIVE'
                JOIN bus bus ON
bus.serial_bus=byu.serial_bus AND
bus.status_bus='ACTIVE'
                JOIN school_year scy ON
scy.serial_scy=byu.serial_scy
                WHERE byu.serial_usr =
{$serial_usr}
                AND byu.serial_scy IN
({$serials})
                AND byu.status_byu =
'ACTIVE'";

        $result = $this->db-
>getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }
    /*
     * Retrieve all the Users from a
Route-Shift-Turn.Bus Configuration
     * @return Array
     */
    function
getBusConfigurationByStudentByYear($seri
al_usr,$serials)
    {
        $sql = "SELECT
byu.serial_usr,byu.serial_scy,scy.name_s
cy,rot.serial_rot,rot.code_rot,rot.descr
iption_rot,sft.serial_sft,sft.name_sft,b
us.serial_bus,bus.code_bus,bdr.serial_bd
r,CONCAT(bdr.first_name_bdr,' ',
bdr.last_name_bdr) as
full_name_bus_driver,
byu.type_address_byu,byu.address_byu
                FROM bus_year_user byu
```

```
                JOIN user usr ON
byu.serial_usr = usr.serial_usr
                JOIN student std ON
usr.serial_std = std.serial_std
                JOIN route rot ON
rot.serial_rot=byu.serial_rot AND
rot.status_rot='ACTIVE'
                JOIN shift sft ON
sft.serial_sft=byu.serial_sft AND
sft.status_sft='ACTIVE'
                JOIN bus bus ON
bus.serial_bus=byu.serial_bus AND
bus.status_bus='ACTIVE'
                JOIN bus_shift_year bsy
ON bsy.serial_rot=byu.serial_rot AND
bsy.serial_sft=byu.serial_sft AND
bsy.serial_bus=byu.serial_bus
                JOIN bus_driver bdr ON
bdr.serial_bdr=bsy.serial_bdr
```

```
                JOIN school_year scy ON
scy.serial_scy=byu.serial_scy
                WHERE byu.serial_usr =
{$serial_usr}
                AND byu.serial_scy IN
({$serials})
                AND byu.status_byu =
'ACTIVE'";

        $result = $this->db-
>getAll($sql);
        if($result){
            return $result;
        }
        else{
            return false;
        }
    }
}
?>
```

fGuardianBusMapReview.js

```
/*
File: fGuardianBusMapReview.js
Author: Alex Chávez
Creation Date: 22/02/2013
Last Modified:
*/
$(document).ready(function(){
    /*LISTENERS*/
    $("#selSchoolYear").bind("change",
function(){
        if($("#profile").val() == '2'){
            $('#map_container').hide();
            $('#bus_details').hide();

loadStudentsBySchoolYear($(this).val());
        }
        else{
            $('#map_container').hide();
            $('#bus_details').hide();
            setSessionVar($(this).val(),
'serial_scy');
        }
    });

    $("#selStudent").bind("change",
function(){
        if($(this).val()){
            $('#map_container').hide();
            $('#bus_details').hide();
            setSessionVar($(this).val(),
'serial_std');
        }
        else{
            $('#map_container').hide();
            $('#bus_details').hide();
        }
    });

    $("#btnSearch").bind("click",
function(){

if($("#frmBusRouteDetail").valid())
        {

loadTransportationDetails($("#selSchoolY
ear").val(),$("#selStudent").val());
        }
    });

    /*END LISTENERS*/

    $("#frmBusRouteDetail").validate({
        errorLabelContainer: "#alerts",
        wrapper: "li",
```

```
        onfocusout: false,
        onkeyup: false,
        rules: {
            selSchoolYear: {
                required: true
            },
            selStudent: {
                required: true
            }
        }
    });

//    initialize();
});
//FUNCTIONS
/**
 *@Name: setSessionVar
 *@Description: Sets the Session Va
 **/
function setSessionVar(serial, type)
{
    if(serial && type){
        $.ajax({
            url:
document_root+"rpc/remoteValidation/stud
ent_details/set_session.rpc.php",
            type: "post",
            data: ({
                serial: serial,
                type: type
            }),
            success: function(data) {
            }
        });
    }
}
/**
 *@Name: loadStudentsBySchoolYear
 *@Description: Loads the students for
the selected school year
 **/
function
loadStudentsBySchoolYear(serial_scy)
{
    if(serial_scy){
        setSessionVar(serial_scy,
'serial_scy');

$('#studentsContainer').load(document_ro
ot+"rpc/loads/student_details/transporta
tion_detail/loadStudentsBySchoolYear.rpc
",{
            serial_scy: serial_scy
        },function(){
```

```
$("#selStudent").bind("change",
function(){
                if($(this).val()!=''){

$('#map_container').hide();

$('#bus_details').hide();

setSessionVar($(this).val(),
'serial_std');
                }
                else{

$('#map_container').hide();

$('#bus_details').hide();
                }
            });
        });
    }else{

$('#studentsContainer').html('<select
name="selStudent" id="selStudent"
class="span-5" title="El campo
Estudiante es obligatorio"><option
value="">- Seleccione -
</option></select>');
        $('#map_container').hide();
        $('#bus_details').hide();
    }
}
/**
 *@Name: loadTransportationDetails
 *@Description: Loads the transportation
details
 **/
function
loadTransportationDetails(serial_scy,
serial_std)
{
    if(serial_scy, serial_std){

$('#bus_details').load(document_root+"rp
c/loads/student_details/transportation_d
etail/loadTransportationInfo.rpc",{
            serial_scy: serial_scy,
            serial_std: serial_std

        },function(){

$("#driver_details").dialog({
                bgiframe: true,
                autoOpen: false,
                height: 400,
                width: 600,
                maxHeight :700,
                maxWidth :750,
                modal: true,
                hide: 'slide',
                show: 'slide',
                title: 'DATOS CONDUCTOR'
            });


$("#map_details_gps").dialog({
                bgiframe: true,
                autoOpen: false,
                height: 480,
                width: 700,
                maxHeight :480,
                maxWidth :700,
                modal: true,
                hide: 'slide',
                show: 'slide',
                title: 'DATOS GPS'
            });
```

```
$('[id^=bus_driver_detail_]').each(funct
ion(){
            $(this).bind('click',
function(){

$('#driver_details').load(document_root+
"rpc/loads/student_details/transportatio
n_detail/loadDriverInfo.rpc",
                    {
                        serial_bdr:
$(this).attr('serial_bdr')
                    },
                    function(){
                        //Set table size

$('#table_wrapper').attr('class',$('#tab
le_wrapper').attr('class')+' span-14');

                    });

$('#driver_details').dialog('open');
                });
            });


$('[id^=map_details_icon_]').each(functi
on(){
            $(this).bind('click',
function(){

$('#map_details_gps').dialog('open');

$('#map_container').show();
//
alert($(this).attr('serial_rot')+'_'+$(t
his).attr('serial_scy')+'_'+$(this).attr
('serial_bus')+'_'+$(this).attr('serial_
sft'))

initialize($(this).attr('serial_rot'),$(
this).attr('serial_scy'),$(this).attr('s
erial_bus'),$(this).attr('serial_sft'));
                });
            });
            $('#bus_details').show();
        });
    }else{
        $('#bus_details').html('');
        $('#map_container').hide();
    }
}
/**
 *@Name: initialize
 *@Description: initialize the map with
the GPS data
 **/
function initialize(serial_rot,
serial_scy, serial_bus, serial_sft) {
    var map = new
google.maps.Map(document.getElementById(
"map_controls"), {
        mapTypeId:
google.maps.MapTypeId.ROADMAP
    });
    $.ajax({
        type: "GET",
        url:
document_root+"map/"+serial_rot+"-
"+serial_sft+"-"+serial_bus+"-
"+serial_scy+".xml",
        dataType: "xml",
        success: function(xml) {
            var points = [];
            var bounds = new
google.maps.LatLngBounds ();
```

```
$(xml).find("trkpt").each(function() {
                var lat =
$(this).attr("lat");
                var lon =
$(this).attr("lon");
                var p = new
google.maps.LatLng(lat, lon);
                points.push(p);
                bounds.extend(p);
            });
            var poly = new
google.maps.Polyline({
                path: points,
                strokeColor: "#FF00AA",
                strokeOpacity: .7,
                strokeWeight: 4
            });
            poly.setMap(map);
            // fit bounds to track
            map.fitBounds(bounds);
        }
    });
}
```

fGuardianBusMapReview.php

```php
<?php
/*
File:fguardianBusMapReview.php
Author: Alex Chávez
Creation Date: 22/02/2013
*/
if(Global_Functions::getCurrentActiveProfile() == 2){//ONLY FOR GUARDIAN
    $profile = '2';
    $smarty->register('profile');
    if(isset($_SESSION['rutademic_serial_usr']['student_detail_year']) ||
isset($_SESSION['rutademic_serial_usr']['student_detail_id'])){
        $selectedSchoolYear = $_SESSION['rutademic_serial_usr']['student_detail_year'];
        $selectedStudent = $_SESSION['rutademic_serial_usr']['student_detail_id'];
        $user = new User($db, $_SESSION['rutademic_serial_usr']['serial_usr']);
        $serial_grd = $user->getSerialGrd();
        if($serial_grd){
            $studentGuardian = new Student_Guardian($db);
            $studentGuardian->setSerialGrd($serial_grd);
            $studentsList = $studentGuardian-
>getStudentsByGuardianInSchoolYear($selectedSchoolYear);
            if($studentsList){
                $smarty->register('studentsList,selectedSchoolYear,selectedStudent');
            }
        }
    }
    $schoolYear = new School_Year($db);
    $schoolYearList = $schoolYear->getSchoolYearsActiveCurrent();
    if($schoolYearList){
        $smarty->register('schoolYearList');
    }
}
if(Global_Functions::getCurrentActiveProfile() == 3){//ONLY FOR STUDENT
    $profile = '3';
    $user = new User($db, $_SESSION['rutademic_serial_usr']['serial_usr']);
    $loggedStudent = $user->getSerialStd();
    $student = new Student($db, $loggedStudent);
    $nameStudent = $student->getFirstNameStd().' '.$student->getLastNameStd();
    $smarty->register('profile,loggedStudent,nameStudent');
    if(isset($_SESSION['rutademic_serial_usr']['student_detail_year'])){
        $selectedSchoolYear = $_SESSION['rutademic_serial_usr']['student_detail_year'];
        $smarty->register('selectedSchoolYear');
    }
    $schoolYear = new School_Year($db);
    $schoolYearList = $schoolYear->getSchoolYearsActiveCurrent();
    if($schoolYearList){
        $smarty->register('schoolYearList');
    }
}

if($error_messages){
    $smarty->register('error_messages');
}elseif($success){
    $smarty->register('success');
}


$smarty->display();
?>
```

fGuardianBusMapReview.es.tpl

```
<!--/*
File: fGuardianBusMapReview.es.tpl
Author: Alex Chávez
Creation Date: 22/02/2013
Last Modified:
*/-->

{assign var="title" value="M&Oacute;DULO DE TRANSPORTES - DETALLE DE TRANSPORTE"}
<form name="frmBusRouteDetail" id="frmBusRouteDetail" method="post" action="" >
    <div class="span-24  last line title">
        M&Oacute;DULO DE TRANSPORTES - DETALLE DE TRANSPORTE<br /><br />
    </div>
    <div class="span-7 span-10 prepend-7 last">
        <ul id="alerts" class="alerts"></ul>
    </div>
```

```
    <div class="span-24 last line">
        <br />
    </div>

    {if $schoolYearList}
        <div class="prepend-1 span-22 last line">
            <div class="span-22 last line separator">
                <label>Seleccione un estudiante</label>
            </div>
        </div>
        <div class="span-24 last line">
            <div class="prepend-8 span-3 label" style="text-align: left;">1.- A&ntilde;o
Lectivo:</div>
            <div class="span-5">
                <select name="selSchoolYear" id="selSchoolYear" class="span-5" title="El
campo 'A&ntilde;o Lectivo' es obligatorio">
                    <option value="">- Seleccione -</option>
                    {foreach from=$schoolYearList item=l}
                        <option value="{$l.serial_scy}"{if $selectedSchoolYear eq
$l.serial_scy}selected{/if}>{$l.name_scy|escape:'htmlall'}</option>
                    {/foreach}
                </select>
            </div>
        </div>
        <div class="span-24 last line">
            <div class="prepend-8 span-3 label" style="text-align: left;">2.-
Estudiante:</div>
            {if $profile eq '2'}
                <div class="span-5 last" id="studentsContainer">
                    <select name="selStudent" class="span-5"  id="selStudent" title="El
campo 'Estudiante' es obligatorio">
                        <option value="">- Seleccione -</option>
                        {if $studentsList}
                            {foreach from=$studentsList item=l}
                                <option value="{$l.serial_std}"{if $selectedStudent eq
$l.serial_std}selected{/if}>{$l.name_std|escape:'htmlall'}</option>
                            {/foreach}
                        {/if}
                    </select>
                </div>
            {else}
                {if $profile eq '3'}
                    <div class="span-5 last">{$nameStudent}</div>
                    <input type="hidden" name="selStudent" id="selStudent"
value="{$loggedStudent}">
                {/if}
            {/if}
        </div>

        <div class="span-24 last buttons">
            <br/>
            <input type="button" name="btnSearch" id="btnSearch" value="Ver Detalle de
Transportes">
        </div>
        <div class="span-24 last line">
            <br />
        </div>
        <div id="bus_details" class="span-22">
        </div>
        <div id="driver_details" class="span-22">
        </div>
        <div id="map_details_gps" class="span-24" style="display: none">
            <div id="map_container" class="span-24"style="display: none">
                <div class="prepend-1 span-20">
                    <div id="map_controls" class="map">
                        <p>This will be replaced with the Google Map.</p>
                    </div>
                </div>
            </div>
        </div>
        <div class="span-24 last line">
            <br />
        </div>
        <input type="hidden" name="profile" id="profile" value="{$profile}">
```

```
    {else}
        <div class="span-7 span-10 prepend-7 last" align="center">
            <div class="alerts"><label>No existen a&ntilde;os lectivos en
curso.</label></div>
        </div>
    {/if}
</form>
```

loadDriverInfo.rpc.php

```
<?php
/*
    Document   : loadDriverInfo
    Created on : 22/02/2013
    Author     : Alex Chavez
    Description:
*/

Request::setInteger('serial_bdr');

if($serial_bdr){
$busDriver = new Bus_Driver($db, $serial_bdr);
$busDvrdata = $busDriver->getArray();
}

$smarty->register('busDvrdata');
$smarty->assign('container','none');
$smarty-
>display('helpers/loads/student_details/transportation_detail/loadDriverInfo.es.tpl');
?>
```

loadStudentsBySchoolYear.rpc.php

```
<?php
/*
    Document   : loadStudentsBySchoolYear
    Created on : 22/02/2013
    Author     : Alex Chavez
    Description:
        Loads the Students fot the selected School Year
*/
Request::setInteger('serial_scy');

$user = new User($db, $_SESSION['rutademic_serial_usr']['serial_usr']);
$serial_grd = $user->getSerialGrd();
if($serial_grd){
    $studentGuardian = new Student_Guardian($db);
    $studentGuardian->setSerialGrd($serial_grd);
    $studentsList = $studentGuardian->getStudentsByGuardianInSchoolYear($serial_scy);
    if($studentsList){
        $smarty->register('studentsList');
    }
}

$smarty->assign('container','none');
$smarty-
>display('helpers/loads/student_details/transportation_detail/loadStudentsBySchoolYear.e
s.tpl');
?>
```

loadTransportationInfo.rpc.php

```
<?php
/*
    Document   : loadTransportationInfo
  Created on : 22/02/2013
    Author     : Alex Chavez
    Description:
        Load Transportation Information from the Selected Student
*/
Request::setInteger('serial_scy,serial_std');
$student=new Student($db, $serial_std);
$serial_usr=$student->getSerialUsr();
$byu=new Bus_Year_User($db);
$transportationInfo=$byu->getBusConfigurationByStudentByYear($serial_usr, $serial_scy);
$size=sizeOf($transportationInfo);
if($size>1){
```

```
    $cont=0;
    foreach($transportationInfo as $ti){
        $serials_routes[]=$ti['serial_rot'];
        $serials_shifts[]=$ti['serial_sft'];
    }

    $serials_routes=array_unique($serials_routes);
    $serials_shifts=array_unique($serials_shifts);

    $cont=0;
    foreach($serials_routes as $r){
        $route=New Route($db, $r);
        $routes[$cont]['serial_rot']=$r;
        $routes[$cont]['code_rot']=$route->getCodeRot();
        $cont++;
    }
    $smarty->register('routes');
}
$smarty->register('transportationInfo,size');
$smarty->assign('container','none');
$smarty-
>display('helpers/loads/student_details/transportation_detail/loadTransportationInfo.es.
tpl');
?>
```

guardian.es.tpl

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<!--[if IE]>
    <meta http-equiv="X-UA-Compatible" content="IE=EmulateIE7" />
<![endif]-->
<meta http-equiv="Content-Type" content="text/html;  charset=iso-8859-1" />
<link type="image/x-icon" rel="shortcut icon" href="{$document_root}img/favicon.ico"/>

<!--GLOBAL CSS FILES-->
<link rel="stylesheet" href="{$document_root}css/screen.css" type="text/css"
media="screen, projection" />
<link rel="stylesheet" type="text/css" media="all" href="{$document_root}css/jquery-
ui/jquery-ui.css" />
<link rel="stylesheet" type="text/css" media="all" href="{$document_root}css/jquery-
ui/jquery-ui-datepicker.css" />
<link rel="stylesheet" href="{$document_root}css/system_styles.css?{$css_timestamp}"
type="text/css" />
<link rel="stylesheet" href="{$document_root}css/ie_system_styles.css?{$css_timestamp}"
type="text/css" />
<link rel="stylesheet" href="{$document_root}css/jquery.autocomplete.css"
type="text/css" />
<link rel="stylesheet" href="{$document_root}css/paging.css" type="text/css" />
<link rel="stylesheet" href="{$document_root}css/browse_filter.css" type="text/css" />
<link rel="stylesheet" href="{$document_root}css/message_filter.css" type="text/css" />
<link rel="stylesheet" href="{$document_root}css/table_styles.css?{$css_timestamp}"
type="text/css" />
<link rel="stylesheet" href="{$document_root}css/menu_styles.css?{$css_timestamp}"
type="text/css" />
<link rel="stylesheet" media="screen" type="text/css"
href="{$document_root}css/jquery.Jcrop.css" />
<link rel="stylesheet" media="screen" type="text/css"
href="{$document_root}css/tooltip.css" />
<link rel="stylesheet" media="screen" type="text/css"
href="{$document_root}css/demo_table.css" />
<link rel="stylesheet" media="screen" type="text/css"
href="{$document_root}css/TableTools.css" />
<link rel="stylesheet" media="screen" type="text/css"
href="{$document_root}css/TableTools_JUI.css" />
<link rel="stylesheet" media="screen" type="text/css"
href="{$document_root}css/demo_table_jui.css" />

<!--[if IE 6]>
<link rel="stylesheet" type="text/css" media="all" href="{$document_root}css/src/ie.css"
/>
<![endif]-->
{if $css_file}
    <link rel="stylesheet" type="text/css"
href="{$document_root}css/{$css_file}?{$css_timestamp}" />
```

```
{/if}

<!--END CSS FILES-->

<!--GLOBAL JS FILES-->
<script type="text/javascript" src="{$document_root}js/jqueryLibrary/jquery.js"
></script>
<script type="text/javascript" src="{$document_root}js/jqueryLibrary/jquery-
validation.js"></script>
<script type="text/javascript" src="{$document_root}js/jqueryLibrary/jquery-
ui.min.js"></script>
<script type='text/javascript' src='{$document_root}js/jqueryLibrary/jquery-
datepicker.js'></script>
<script type='text/javascript'
src='{$document_root}js/jqueryLibrary/jquery.autocomplete.js'></script>
<script type="text/javascript" src="{$document_root}js/system.js"></script>
<script type='text/javascript' src='{$document_root}js/fullcalendar.js'></script>
<script type="text/javascript" src="{$document_root}js/mbMenu.js"></script>
<script type="text/javascript" src="{$document_root}js/global_vars.js"></script>
<script type="text/javascript" src="{$document_root}js/global_functions.js"></script>
<script type="text/javascript" src="{$document_root}js/qwerty.js"></script>
<script type="text/javascript"
src="{$document_root}js/jqueryLibrary/jquery.hoverIntent.js"></script>
<script type="text/javascript" src="{$document_root}js/paging.js"></script>
<script type="text/javascript" src="{$document_root}js/profileHandler.js"></script>
<script type="text/javascript" src="{$document_root}js/ajaxupload.js"></script>
<script type="text/javascript" src="{$document_root}js/jquery.Jcrop.js"></script>
<script type="text/javascript" src="{$document_root}js/jquery.table-search.js"></script>
<script type="text/javascript" src="{$document_root}js/jquery.tools.min.js"></script>
<script type="text/javascript" src="{$document_root}js/jqueryLibrary/json2.js"></script>
<script type="text/javascript" src="{$document_root}js/jquery.fusioncharts.js"></script>
<script type="text/javascript" src="{$document_root}js/jquery.dataTables.js"></script>
<script type="text/javascript" src="{$document_root}js/TableTools.js"></script>
<script type="text/javascript" src="{$document_root}js/ZeroClipboard.js"></script>
<script type="text/javascript"
src="http://maps.google.com/maps/api/js?sensor=false"></script>

{if $js_file}
    <script type="text/javascript" src="{$document_root}js/{$js_file}"></script>
{/if}
<!--END JS FILES-->
<title>{$title}</title>
</head>


<body class="inner_body container">
    <div class="span-24 last banner">
        <div class="span-4">
            <img src="{$document_root}img/logo/{$setup.logo}" alt="logo" />
        </div>
        <div class="span-20 last">
            <div class="span-20">
                <div class="span-7"> 
                    <div class="prepend-1 span-5" id="user_messages_container"
style="display: none;">
                        <a href="{$document_root}modules/message/fInbox"
style="color:white; text-decoration: none; cursor: pointer;"><img
src="{$document_root}img/icons/alert.png"/> <label id="user_messages_number"
style="cursor: pointer;"></label></a>
                    </div>
                </div>
                <div class="span-13 last" align="right" style="color:white;" >
                    {assign var="weekday" value=$date.weekday}
                     {assign var="month" value=$date.month}
                     {$global_weekDaysNumb.$weekday}, {$date.day} de
{$global_weekMonths.$month} del {$date.year}
                </div>
            </div>
            <div class="span-20" style="margin-top: 5px;">
                <div class="{if $serial_usr neq 1}prepend-14 span-6{else}prepend-15
span-5{/if} last" align="right">
                    {if $serial_usr eq 1}<strong><a
href="{$document_root}modules/fSetup" style="text-decoration:none;
color:white">Actualizar Setup</a></strong> | {/if}<strong>{if $serial_usr neq 1}<a
href="{$document_root}modules/user/fUserAccount" style="text-decoration:none;
```

```
color:white;">Actualizar Datos</a> | {/if}<a href="{$document_root}index"
style="text-decoration:none; color:white;">Salir</a></strong>
                </div>
            </div>

            <div class="span-20" style="margin-top: 40px; color:white;">
                <div class="prepend-11 span-9" align="right">
                     Bienvenido <strong>{$username}</strong>, su perfil es
                     {if $profilesSize gt 1}
                        <select name="selProfilesMain" id="selProfilesMain">
                            {foreach from=$profiles item="p"}
                                <option value="{$p.serial_prf}" {if
$p.in_use}selected{/if}>{$p.name_prf}</option>
                            {/foreach}
                        </select>
                    {else}
                        <strong>{$profiles.0.name_prf}</strong>
                    {/if}
                </div>
            </div>

        </div>
    </div>

    <!--<a href="{$document_root}modules/student_details/fGradesByStudent"> Notas
Estudiante</a>
    <a href="{$document_root}modules/student_details/fClassmateDetail">
Compa&ntilde;eros</a>
    <a href="{$document_root}modules/student_details/fAttendanceDetail"> Asistencia</a>
    <a href="{$document_root}modules/student_details/fScheduleDetail"> Horario</a>
    <a href="{$document_root}modules/student_details/fSyllabusFilesDetail">
P&eacute;nsum de Estudios</a>
    <a href="{$document_root}modules/student_details/fTransportationDetail">
Transporte</a>-->
    {if $menu}
        {get_menu menu=$menu}
    {/if}
<div class="span-24 last">
    <div class="span-24 last contents">
        {$buffer}
    </div>
</div>

<div id="spinner"  style="display: none;">
    <div id="ajax_loader"></div>
    <div class="ajax_loader_text">Cargando...</div>
</div>
</body>
</html>
```

3.3 PRUEBAS UNITARIAS PANTALLA INGRESO NUEVO CONDUCTOR

En las pruebas unitarias de la pantalla de ingreso de nuevo conductor vamos a evaluar los siguientes casos:

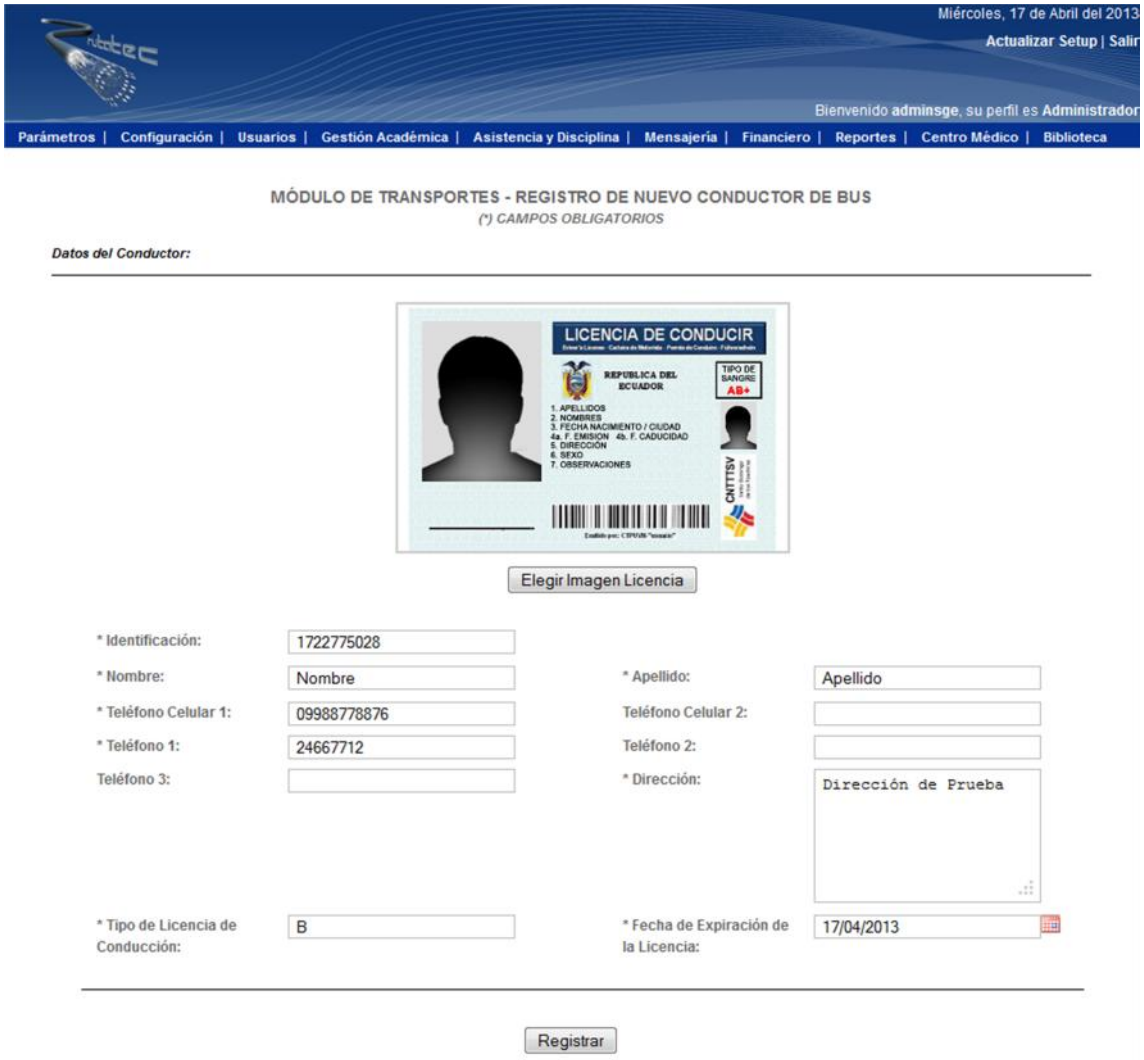
- Omisión de datos mandatorios, tipos de datos y procesos remotos.
- Caso ideal.

En el primer caso se hace el intento de guardar la información en la base de datos sin ingresar ningún campo y sin respetar las validaciones y como se espera el resultado es un mensaje de error.

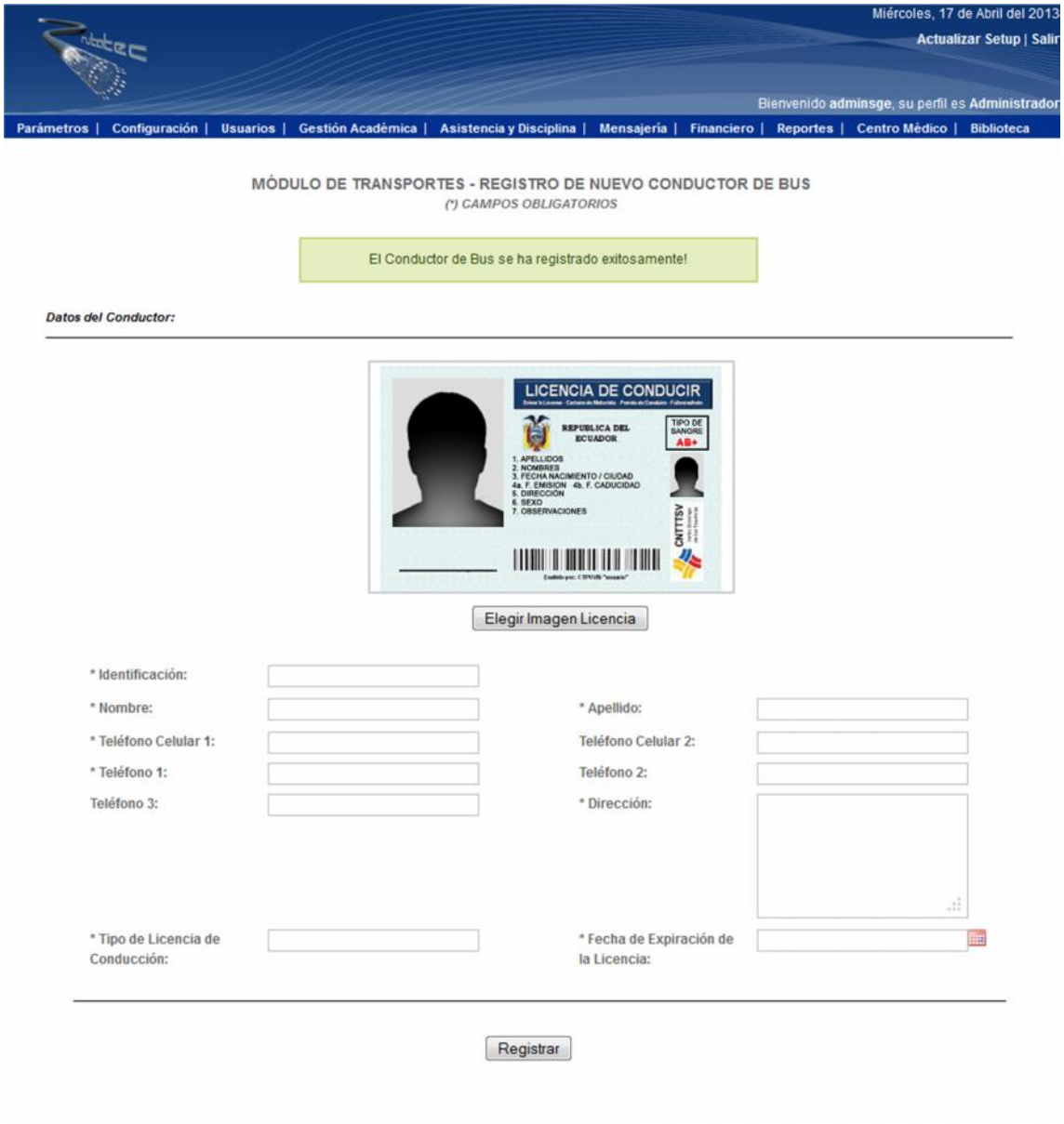


Título: Ilustración 38
Descripción: Captura de Pantalla de la interfaz para el ingreso de un nuevo conductor con mensaje de error al validar.
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

La siguiente prueba es ingresar correctamente todos los datos, es decir el caso ideal. El resultado de este caso de prueba es un mensaje de confirmación y éxito del ingreso del nuevo conductor en la base de datos.



Título: Ilustración 39
Descripción: Captura de Pantalla de la interfaz para el ingreso de un nuevo conductor con datos correctos.
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez



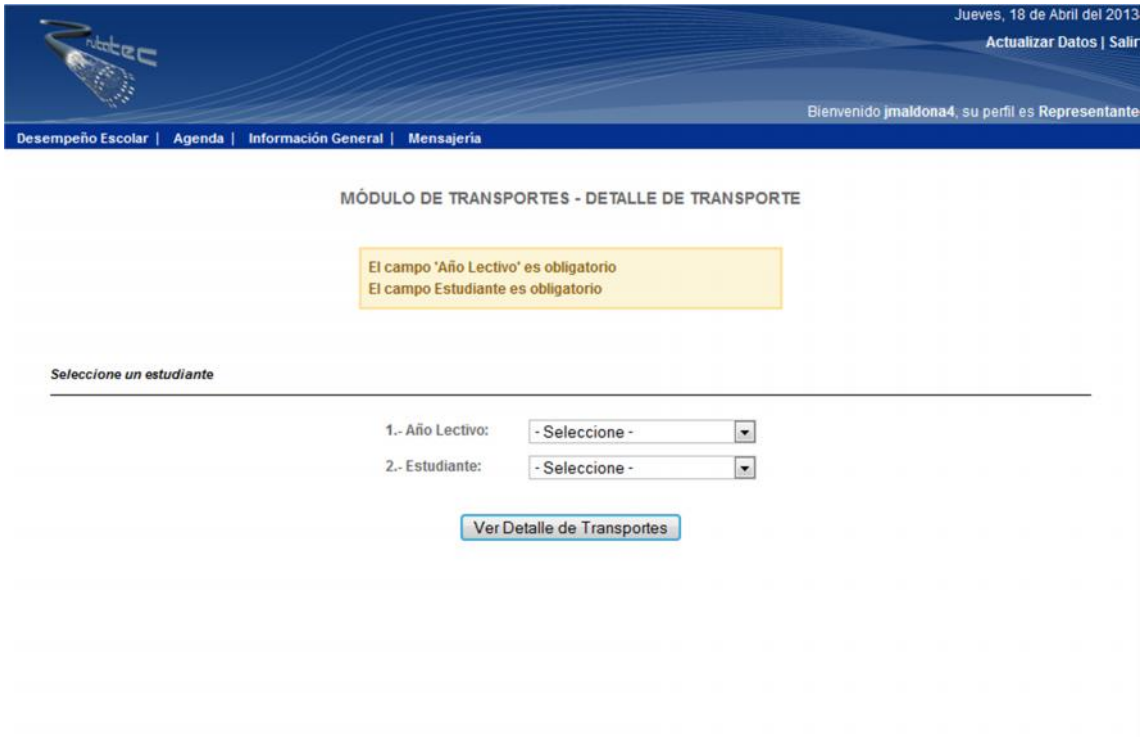
Título: Ilustración 40
Descripción: Captura de Pantalla de la interfaz para el ingreso de un nuevo conducto, ingreso exitoso.
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

3.4 PRUEBAS UNITARIAS REPORTE DE RUTAS BASADO EN DATOS DEL GPS

En las pruebas unitarias del reporte de rutas basado en datos del GPS se manejarán dos casos de prueba:

- Omisión de filtros obligatorios,
- Caso ideal.

Para el primer caso se intentará generar el reporte visual sin elegir los filtros obligatorios y lo que se tiene como resultado es un mensaje de error.

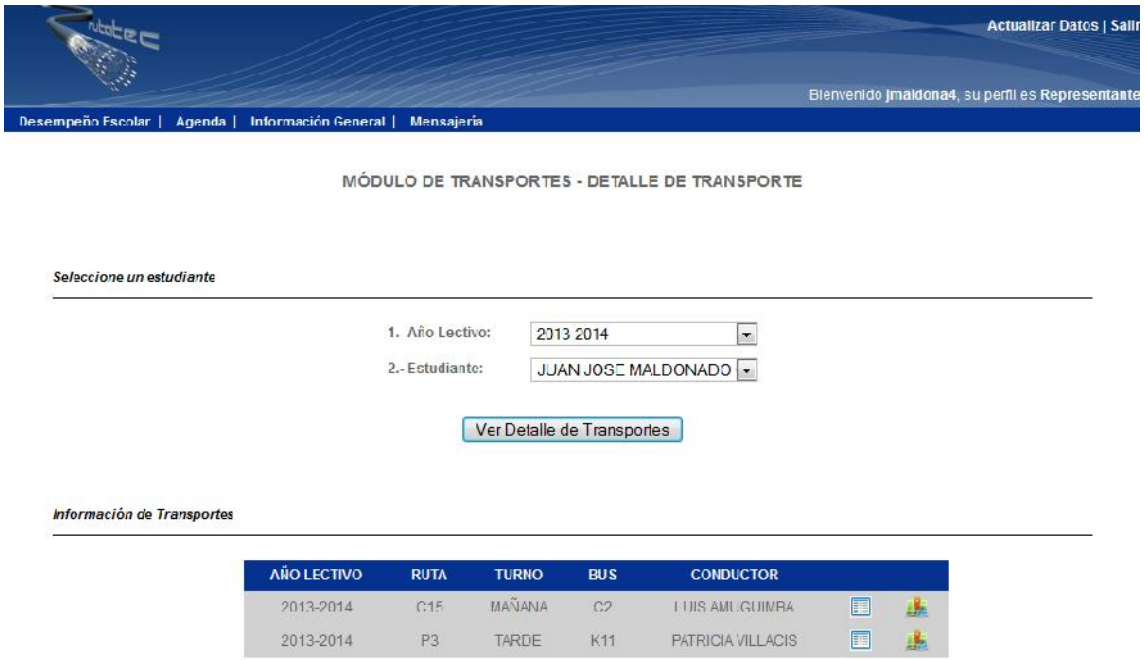


Título: Ilustración 41
Descripción: Captura de Pantalla de reporte de rutas para padres de familia, ingreso erroneo.
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

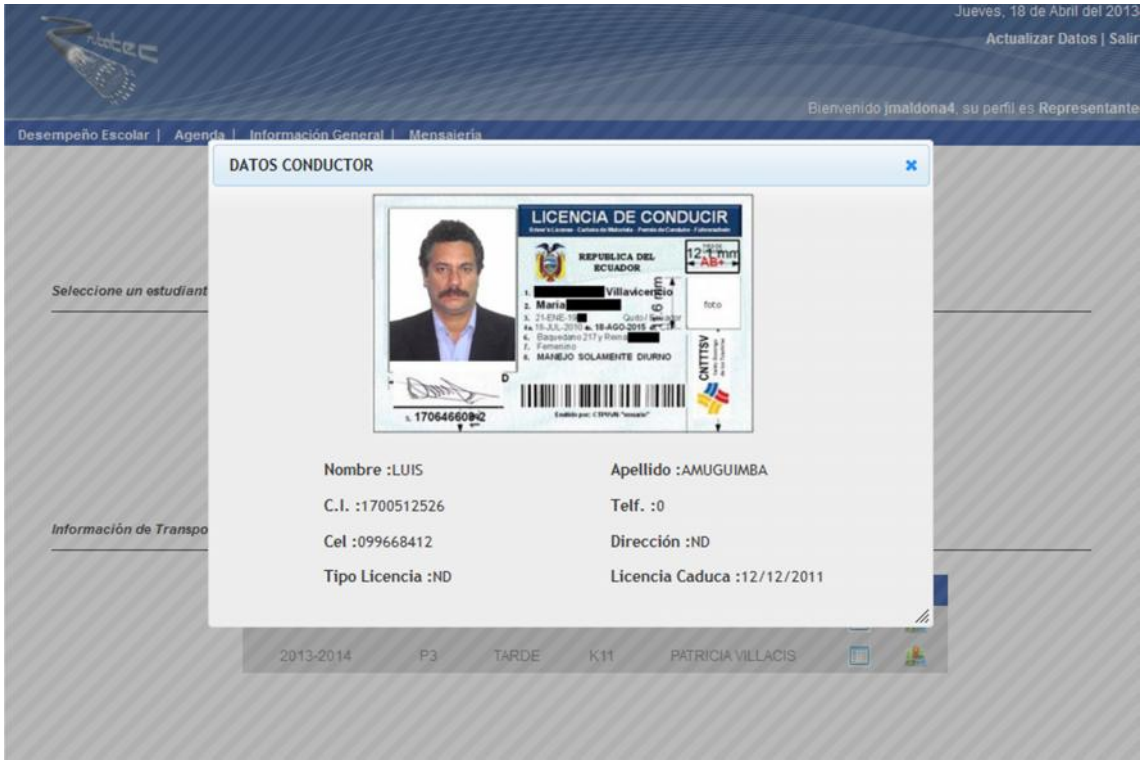
Para el caso ideal se va a seguir el proceso correctamente, es decir con el ingreso o selección de los filtros necesarios. El resultado de este caso de prueba es la obtención de un reporte visual de los datos de la ruta entre los que se encuentran los datos del conductor y además un mapa generado con ayuda de Google Maps con la representación de los datos del GPS del bus seleccionado.



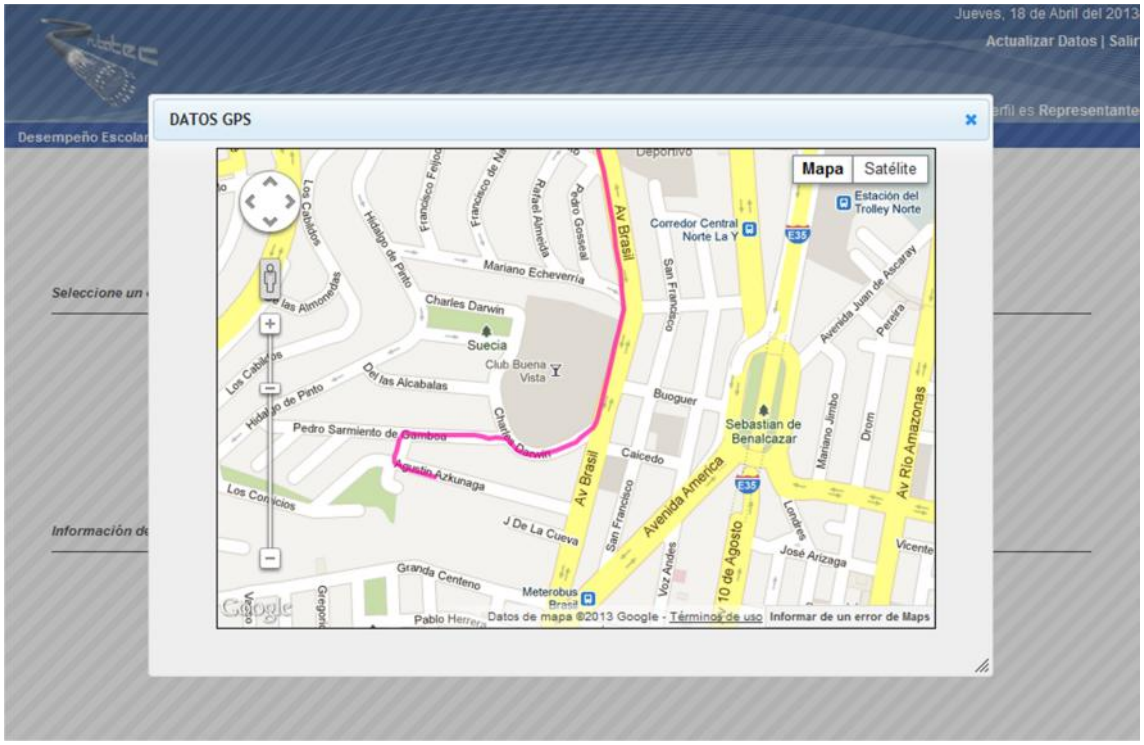
Título: Ilustración 42
Descripción: Captura de Pantalla de reporte de rutas para padres de familia.
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez



Título: Ilustración 43
Descripción: Captura de Pantalla de reporte de rutas para padres de familia, selección correcta de filtros
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez



Título: Ilustración 44
Descripción: Captura de Pantalla de reporte de rutas para padres de familia, consulta datos conductores.
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez



Título: Ilustración 45
Descripción: Captura de Pantalla de reporte de rutas para padres de familia, consulta datos GPS con ayuda de Google Maps.
Fecha: Marzo 2013
Fuente: Realizado por Alex Chávez

4. CAPÍTULO 4

Una vez terminado el presente trabajo y haber hecho las pruebas respectivas, se obtiene las siguientes conclusiones y recomendaciones:

4.1 CONCLUSIONES

- La arquitectura de 3 capas ayuda a tener un sistema mucho más organizado y seguro, ya que no se tiene acceso directo a la base de datos. Además en el caso de que existieran cambios, mejoras o mantenimiento se lo puede hacer de una manera más sencilla y organizada.
- Un buen plan de pruebas ayuda a evitar errores tales como: omitir validaciones importantes, no controlar datos ingresados, no controlar inyección de código y SQL, permitir brechas de seguridad, etc. Estos errores podrían causar pérdidas de información y económicas para la empresa y para el usuario final.
- Es importante mantener una comunicación continua con el cliente al momento de desarrollar software, debido a que esta es la única manera de cumplir con todos los requerimientos y de poder hacer cambios en los mismos, ahorrando tiempo de trabajo y desperdicio de código.
- Usar diagramas UML es una herramienta muy poderosa y eficaz para representar las funcionalidades de un software y los requerimintos para lograr un fácil entendimiento por parte del cliente.
- Una buena práctica al desarrollar para entornos web es la de tener siempre un servidor de pruebas y el servidor de producción ya que si aparece algún problema lo podemos verificar en el servidor de pruebas sin tener que bajar el sitio de producción y así podemos evitar pérdidas de tiempo y dinero al cliente.
- Con la implementación de GPS en los buses, y con el uso del módulo de transportes los coordinadores de transporte de las instituciones que usa en Sistema de Gestión Académico tienen un mayor control sobre las rutas, tiempos, velocidades y paradas no planificadas en los diferentes recorridos.
- Se puede concluir que la tecnología utilizada específicamente en la generación de mapas fue la adecuada, pues se logró lo requerido con buenos resultados gráficos y buenos tiempos de respuesta.
- Algunos de los beneficios de usar el módulo de transporte son: rápido y fácil acceso a los datos generados por el GPS; fácil representación de los datos del GPS en una herramienta de uso cotidiano como es Google Maps; centralización de toda la información importante de los estudiantes en un solo sistema haciendo uso de un solo nombre de usuario y contraseña; obtención de datos estadísticos referenciales para monitorear el labor de los conductores en sus respectivos recorridos.

- La implementación fue sencilla gracias a la gran cantidad de documentación disponible de la herramienta en uso (Google Maps) y a la arquitectura usada. Es por este motivo y por las constantes actualizaciones, que se eligió esta herramienta y no ninguna de sus similares como: Bing Maps, Yahoo Maps, Apple Maps, etc.
- Al momento de trabajar con Google Maps combinado con JavaScript y Ajax es mucho mejor usar archivos del formato XML y no el formato estándar GPX. Esto se debe a que XML es más dinámico al momento de ser leído por una aplicación web ya que existen funciones propias o complemnetos para lograr una fácil manipulación de los datos. Además al generar el XML podemos omitir información que no va a ser usada para obtener archivos más livianos en disco y mejorar la lectura de los datos desde un servidor web.
- Con el uso de la misma tecnología usada en el módulo de transporte se puede generar diversos tipos de sistemas como por ejemplo Sistema de Monitoreo de Datos para Atletas de Alto Rendimiento; esto gracias a que los GPS hoy en día pueden ser sincronizados y configurados junto a otros tipos de sensóres biométricos y asi obtener datos útiles para cualquier tipo de control y monitoreo.

4.2 RECOMENDACIONES

- Como el Módulo de Transportes es un nuevo desarrollo para el Sistema de Gestión Escolar, se debe seguir los estándares de nomenclatura de la empresa; los cuales nos dan lineamentos para poner nombres a clases, archivos y tablas en la base de datos. Con esto tenemos un código más ordenado y entendible para personas que están relacionadas con los estándares usados.
- Antes de implantar el sistema se debe verificar que los servidores tengan la capacidad de almacenamiento necesaria para que cumpla con los requerimientos y que la aplicación funcione correctamente.
- Se recomienda usar NetBeans ya que es un IDE muy útil al momento de trabajar con PHP porque tiene integración completa con el código, su sintaxis y además reconoce todos los tipos de archivos y nos ayuda a organizar de mejor manera el trabajo.
- Se recomienda tener un servidor de respaldos de las bases de datos ya que este es el principal valor del sistema, si se llega a perder la información ésta no se podrá recuperar.
- Se recomienda usar un servidor de versionamiento en la nube para así hacer más dinámico el trabajo y que se lo pueda hacer desde cualquier sitio con una conexión a internet.
- Se recomienda manejar contraseñas cifradas de al menos 12 caracteres numéricos y especiales para evitar intrusiones en el sistema o en la base de datos, lo mismo para el servidor, con esto aseguramos seguridad de los datos.
- También se recomienda trabajar con una VPN cifrada para hacer mantenimiento remoto al servidor debido a que esta es la manera más segura de establecer una conexión y se evita la pérdida de información y las intrusiones.
- Se recomienda elaborar un manual de uso de fácil entendimiento para poder así dar un mejor soporte técnico al sistema.
- Se recomienda que el desarrollo de software sea portable y abierto, esto nos ayuda a que se adapte a varias necesidades.

4.3 BIBLIOGRAFÍA

- **Maldonado Juan Pablo (2013).** *Elaboración de un Plan de Mercadeo para el sistema de Gestión Escolar RUTADEMIC*, Tesis Maestría, Pontificia Universidad Católica del Ecuador.
- **About Moodle.** Consultado el 03 de Fecbrero del 2013, de http://docs.moodle.org/24/en/About_Moodle.
- **Cerda Seguel Diego (2006).** *El Mundo Según Google. Google Earth y la Creación del Dispositivo Geosemántico Global.* Consultado el 05 de Febrero del 2013, de http://escaner.zoomblog.com/archivo/2006/07/12/-El-Mundo-Segun-Google-Google-Earth-Y-.html.
- **Palacio Juan (2006).** *El modelo Scrum.* Consultado 23 de Febrero del 2013, de http://www.navegapolis.net/files/s/NST-010_01.pdf
- **Melo María Luisa (2003).** *Geolocalización: el poder de las coordenadas terrestres.* Consultado 23 de Febrero del 2013, de http://www.computerworld.es/archive/geolocalizacion-el-poder-de-las-coordenadas-terrestres.
- **Microsoft Corporation (2013).** *Visual Studio,* Consultado 25 Febrero 2013, de http://www.microsoft.com/visualstudio/esn/visual-studio-update
- **Microsoft Corporation (2013).** *SQL Server,* Consultado 25 Febrero 2013, de http://www.microsoft.com/es-es/sqlserver/default.aspx
- **Microsoft Corporation (2013).** *Sharepoint,* Consultado 25 Febrero 2013, de http://sharepoint.microsoft.com/es-es/Paginas/default.aspx
- **Microsoft Corporation (2013).** *Windows Server,* Consultado 25 Febrero 2013, de http://www.microsoft.com/en-us/server-cloud/windows-server/
- **Apache Friends (2013).** *XAMPP,* Consultado 02 Marzo 2013, de http://www.apachefriends.org/es/xampp.html
- **Netscape Corporation (2007).** *The SSL Protocol*, Consultado 02 Marzo 2013, de http://home.netscape.com/newsref/std/SSL.html
- **Maguire, D.J., Goodchild M.F., Rhind D.W. (1997).** *Geographic Information Systems: principles, and applications,* Longman Scientific and Technical, Harlow.
- **Pogge Richdard (2008).** *Real-World Relativity: The GPS Navigation System.* Consultado 07 Marzo 2013, de http://www.astronomy.ohio-state.edu/~pogge/Ast162/Unit5/gps.html
- **Google Corporation (2013).** *Google Earth,* Consultado 07 Marzo 2013, de http://www.google.com/earth/index.html
- **PHP (2013),** *What PHP is?*, Consultado 07 Marzo 2013, de http://www.php.net/
- **SMARTY Tenmplate Engine (2013),** *What is Smarty?,* Consultado 07 Marzo 2013, de http://www.smarty.net/about_smarty
- **Mozilla Developer Network (2013),** *JavaScript,* Consultado 07 Marzo 2013, de https://developer.mozilla.org/es/docs/JavaScript
- **Leff, Avraham; James T. Rayfield (September 2001).** *Web-Application Development Using the Model/View/Controller Design Pattern.* IEEE Enterprise Distributed Object Computing Conference. pp. 118–127.