



UNIVERSIDAD TECNOLÓGICA ISRAEL

TRABAJO DE TITULACIÓN

CARRERA: INGENIERÍA EN SISTEMAS

TEMA: SISTEMA DE SEGUIMIENTO DE SYLLABUS

AUTOR/ A: LILY DEL CARMEN FLOR GAVILANES

TUTOR/ A: ING. WILMER VALLE

2014

ÍNDICE CONTENIDOS

ÍNDICE CONTENIDOS	1
ÍNDICE DE GRÁFICOS	3
ÍNDICE DE TABLAS	4
INTRODUCCIÓN	5
PLANTEAMIENTO DEL PROBLEMA	5
OBJETIVO GENERAL	7
OBJETIVOS ESPECIFICOS	7
ALCANCE	7
CAPÍTULO I	9
1. INTRODUCCIÓN	9
1.1. METODOLOGÍA DE INVESTIGACIÓN	9
1.1.1. MÉTODO DE INVESTIGACIÓN	9
1.1.2. TÉCNICA DE INVESTIGACIÓN	11
1.1.3. INSTRUMENTO DE LA INVESTIGACIÓN	11
1.1.4. RESULTADOS DE LA INVESTIGACIÓN	12
1.1.5. RESUMEN DE LA ENTREVISTA APLICADA	12
1.2. DESARROLLO DEL SISTEMA	13
1.2.1. METODOLOGÍA DE DESARROLLO DEL SISTEMA	13
METODOLOGÍA XP	13
PRINCIPIOS	13
CARACTERÍSTICAS	14
ROLES	14
FASES	16
1.2.2. HERRAMIENTAS DE DESARROLLO	21
CAPÍTULO II	23
DESARROLLO	23
2.1. INTRODUCCIÓN	23
2.3. PRUEBAS DE ACEPTACIÓN	26
2.4. MODELO ENTIDAD RELACIÓN	29
2.5. DICCIONARIO DE DATOS	30

3.	CONCLUSIONES	32
4.	RECOMENDACIONES	32
5.	BIBLIOGRAFÍA	33
6.	ANEXOS	35
	INTRODUCCIÓN.....	50
1.	REQUISITOS DE SOFTWARE	51
1.1.	Programador	51
1.2.	Cliente	51
2.	INTRODUCCIÓN.....	51
3.	DIAGRAMA DE LA BASE DE DATOS.....	53
4.	DICCIONARIO DE DATOS.....	54
5.	DIAGRAMA DE RED	55
6.	MANUAL DE PROGRAMADOR	56

ÍNDICE DE GRÁFICOS

Gráfico 1: Modelo Entidad Relación	14
Gráfico 2: Fases de la Metodología XP	25
Gráfico 3: características principales de .NET Framework.....	22

ÍNDICE DE TABLAS

Tabla 1: Asistencia_Alumnos	27
Tabla 2: Parciales	27
Tabla 3: Planificacion_Curriculo	27
Tabla 4: Recursos	27
Tabla 5: Seguimiento_Silabo	28
Tabla 6: Semana_Clases	28

INTRODUCCIÓN

A través del tiempo el ser humano ha venido teniendo cambios radicales en su vida, de tal manera que todo lo que ha aprendido lo ha puesto en práctica para mejorar sus condiciones de vida. Es importante destacar la gran diferencia de modo de vida del ser humano de hace algunos años a los seres humanos de la actualidad, estas diferencias se dan por el avance tecnológico que ha tenido el mundo entero tomando en cuenta que existe una relación muy estrecha entre el ser humano y la tecnología.

Algunos de los países que han contribuido directamente a la evolución de la tecnología son China, Estados Unidos, Japón, Alemania entre otros. La tecnología tiene como principal objetivo el facilitar la vida diaria a los seres humanos.

La tecnología ha tenido un impacto considerable sobre el mundo entero, por tal motivo existen muchas personas que se dedican al desarrollo de aplicaciones, sistemas que ayuden al ser humano en sus labores cotidianas.

Después de mencionar como ha venido evolucionando la tecnología de acuerdo a las necesidades de los seres humanos; se conoce la necesidad de mejorar el control de docentes en la Universidad Tecnológica Israel porque lamentablemente la manera en la que se maneja los procesos es inadecuada y poco veraz; debido a que se realiza el seguimiento de syllabus a través de un sistema Open Source que no tiene un control adecuado de usuarios y planificación.

El proceso de seguimiento de Syllabus se refiere a la planificación que debe llevar el docente de acuerdo al distributivo asignado cada período, tomando en cuenta que el control de asistencia a los alumnos será una manera de controlar que el docente está impartiendo su planificación ingresada al inicio del período académico.

Por tanto el problema principal que se pretende resolver es el inadecuado proceso de seguimiento de syllabus.

PLANTEAMIENTO DEL PROBLEMA

En la actualidad las Universidades Ecuatorianas han experimentado cambios académicos; tomando en cuenta que de acuerdo a la Disposición Transitoria Vigésima de la Constitución de la República del Ecuador que indica que "En el plazo de cinco años a partir de la entrada en vigencia de esta Constitución, todas las instituciones de educación superior, así como sus carreras, programas y postgrados deberán ser evaluados y acreditados conforme a la Ley". (Ecuador, Transitoria Vigésima , 2008)

Además en el Mandato Constituyente 14 de la Asamblea Constituyente de Montecristi dispuso en el segundo inciso de su Disposición Transitoria Primera: "Será obligación que... el Consejo Nacional de Evaluación y Acreditación -CONEA, entregue al CONESUP y a la Función Legislativa, un informe técnico sobre el nivel de desempeño institucional de los establecimientos de educación superior, a fin de garantizar su calidad, proporcionando su depuración y mejoramiento; según lo determinado en el artículo 91 de la Ley Orgánica de Educación Superior". (Ecuador, Transitoria Primera, 2008)

Es importante mencionar que en el artículo 15 del Reglamento General a la Ley Orgánica de Educación Superior indica que: "Todas las universidades o escuelas politécnicas se someterán a la tipología establecida por el CEAACES, la que será tomada en cuenta en los procesos de evaluación, acreditación y categorización. En virtud de la tipología de universidades y escuelas politécnicas, el CEAACES establecerá los tipos de carreras o programas que estas instituciones podrán ofertar, de lo cual notificará al CES para la aprobación de carreras y programas.". (Superior, 2012)

Por tal motivo de acuerdo a los nuevos requerimientos de los organismos de control como son el Consejo de Educación Superior y el Consejo de Evaluación, Acreditación y Aseguramiento de la Calidad de la Educación Superior se comienzan a tener cambios drásticos todas las instituciones educativas tanto públicas como privadas.

Después de las acreditaciones de las instituciones educativas surgen nuevos requerimientos de calidad, como se evidencia en la publicación del Diario El Mercurio "La segunda evaluación y acreditación de las 54 universidades del país revela un mejor nivel académico que hace cuatro años. Si bien no existen universidades cerradas, se define una nueva categorización que ubica solo a tres centros de educación superior en la categoría A. Esta nueva evaluación se realizó con parámetros distintos a los utilizados por el ex CONEA. Algunos de los indicadores advierten que las universidades en su conjunto han mejorado y han apostado por generar conocimiento" Mencionó Francisco Cadena, presidente del Ceaaces, durante la presentación del informe. (Cadena, 2013)

"Las universidades que se mantuvieron en categoría A hicieron un esfuerzo importante de mejora en referencia al sistema y las felicitamos, han mejorado a una velocidad más rápida que el resto. A las de categoría B también las felicitamos porque no significa que están mal, significa que han mejorado, pero no a la velocidad que las de categoría A y eso queremos aclarar para evitar malos entendidos", refirió René Ramírez. (Ramírez, 2013)

Después de las evaluaciones que se han venido dando a todas las instituciones de educación superior la Universidad Tecnológica Israel cambia su modelo de control a los docente, pero lamentablemente la manera en la que se maneja los procesos es inadecuada y poco veraz; porque se realiza el seguimiento de syllabus a través

de un sistema Open Source que no tiene un control adecuado de usuario y planificación.

Por tanto el problema principal que se pretende resolver es el inadecuado proceso de seguimiento de syllabus.

OBJETIVO GENERAL

Mejorar el proceso de seguimiento de syllabus de los docentes de la Universidad Tecnológica Israel.

OBJETIVOS ESPECIFICOS

- Analizar las características de la problemática de la institución en cuanto al seguimiento de Syllabus a los docentes.
- Mejorar el procesamiento de información del seguimiento de Syllabus del docente.
- Automatizar el control de docentes de la Universidad Tecnológica Israel.

ALCANCE

El presente trabajo de titulación tiene como principal objetivo el mejorar el proceso de seguimiento de syllabus de los docentes en la Universidad Tecnológica Israel; para cumplir con dicho objetivo, se plantea desarrollar un sistema en ambiente web de tal forma que sea para el usuario un sistema amigable.

Los principales módulos con los que cuenta el sistema son:

- Planificación del Docente
- Control de Asistencia
- Reporte Asistencia Alumnos
- Reporte Planificación Curricular

- Mantenimiento de Tablas (Usuario Administrador)

CAPÍTULO I

1. INTRODUCCIÓN

En este capítulo se detalla la metodología, el método y el instrumento de la investigación a utilizar para el desarrollo de este trabajo de titulación.

Las metodologías ayudan a tener como referencia un sin número de procedimientos para alcanzar una serie de objetivos propuestos al inicio de una investigación de cualquier tipo. Es importante mencionar que no se puede llamar a cualquier proceso metodología.

El término método, también conocido como técnicas de investigación puede definirse como el camino para alcanzar un fin; en relación con la metodología, consiste en los pasos que deben ejecutarse para dar cumplimiento con lo planteado por ella y conseguir conclusiones verídicas sobre el fenómeno o problema que se analiza.

Los instrumentos de investigación nos sirven para medir las variables que se emplea para analizar el objeto de estudio, así como para recopilar la información que se requiere u observar el comportamiento de dichas variables.

1.1. METODLOGÍA DE INVESTIGACIÓN

1.1.1.MÉTODO DE INVESTIGACIÓN

La investigación científica es la ejecución de varios procesos que conducen al conocimiento por medio de la aplicación de métodos y técnicas, de tal forma es importante tomar en cuenta lo siguiente:

- a) **Descriptivas.**- Su objetivo principal es la descripción de las características fundamentales de los grupos iguales en fenómenos, tomando como herramienta los criterios sistemáticos que permitan mostrar su estructura o comportamiento. Así la obtención de los criterios que aplican a lo estudiado.

b) Explicativas.- Se centra en la determinación de los orígenes o las causas de un conjunto de fenómenos determinado, tomando en cuenta que el objetivo principal es conocer por que suceden hechos particulares a través de delimitar las causas existentes o, por lo menos las condiciones en que se producen.

De tal manera se muestra los siguientes métodos de investigación:

- Método lógico deductivo.
- Método deductivo directo – inferencia o conclusión inmediata.
- Método deductivo indirecto – inferencia o conclusión mediata - formal.
- Método hipotético-deductivo.
- Método lógico inductivo.

Para este trabajo de titulación se utiliza el método Lógico Deductivo.

La deducción va de lo general a lo particular. El método deductivo parte de la información aceptada como verídica, de tal forma llegar a la deducción usando el razonamiento lógico, varias suposiciones, es decir; parte de principios generales establecidos como verdades, para su aplicación en casos individuales y comprobar así su validez.

El razonamiento deductivo es una de los puntales primordiales en la investigación cuantitativa.

Se aplican los principios descubiertos a casos particulares, a partir de un enlace de juicios.

El papel de la deducción consiste en encontrar principios desconocidos, a partir de los conocidos. Una ley o principio puede reducirse a otra más general que la incluya.

1.1.2.TÉCNICA DE INVESTIGACIÓN

Las técnicas son los procedimientos e instrumentos que utilizamos para acceder al conocimiento. Encuestas, entrevistas, observaciones y todo lo que se deriva de ellas.

La entrevista es una de las técnicas preferidas de los partidarios de la investigación cualitativa, pero también es un procedimiento muy usado por los psiquiatras, psicólogos, periodistas, médicos y otros profesionales, que a la postre es una de las modalidades de la interrogación, o sea el hacer preguntas a alguien con el propósito de obtener información específica.

Es una técnica para obtener datos que consisten en un diálogo entre dos personas: El entrevistador "investigador" y el entrevistado; se realiza con el fin de obtener información de parte de este, que es, por lo general, una persona entendida en la materia de la investigación.

De acuerdo a lo que se pretende conseguir con la entrevista, esta puede estar o no estructurada por un cuestionario previamente elaborado. De acuerdo a cuando se aplique la entrevista; las etapas previas de la investigación donde se quiere conocer el objeto a investigar desde una visión externa, sin que se requiera aún la profundización en la esencia del fenómeno, la formulación de preguntas por el entrevistador, se deja a su criterio y experiencia.

1.1.3.INSTRUMENTO DE LA INVESTIGACIÓN

Los instrumentos son las herramientas que se utilizan para llevar a cabo las observaciones. De acuerdo a lo que se desea estudiar, la característica a observar, sus propiedades y factores relacionados como el ambiente, los recursos humanos y económicos, etcétera. Se considera básicamente tres: la observación, la encuesta (que utiliza cuestionarios) y la entrevista.

El instrumento de la investigación será una entrevista grabada realizada al Vicerrector Académico acerca de la problemática que conlleva el no tener un sistema de control de syllabus adecuado.

1.1.4.RESULTADOS DE LA INVESTIGACIÓN

Como principales resultados de la investigación en el manejo de los procesos de seguimiento de syllabus son:

- Mejoramiento del proceso principal del seguimiento de syllabus.
- Procesamiento adecuado de la información ingresada en los diferentes módulos.
- Proceso dinámico y optimizado del proceso de seguimiento de syllabus.
- Reportes de la información ingresada en los diferentes módulos.

1.1.5.RESUMEN DE LA ENTREVISTA APLICADA

En la entrevista realizada al Señor Vicerrector académico es importante destacar los siguientes puntos:

- Actualmente la Universidad Tecnológica Israel cuenta con un sistema informático Open Source.
- El sistema utilizado actualmente no es óptimo ni eficiente ya que no cuenta con una autenticación confiable.
- Es poco dinámico, porque el usuario debe escoger entre todos los usuarios el suyo además; debe de igual forma buscar la asignatura asignada al período académico de entre todas las vigentes.
- Además es importante mencionar que tener un manejo adecuado de los procesos de seguimiento de syllabus en la Universidad Tecnológica Israel crea un impacto considerable. De tal manera que la institución tendrá la capacidad de garantizar la calidad de educación que se ofrece; además de poder tomar correctivos ante cualquier irregularidad en cuanto a cumplimiento de syllabus.
- Los formatos de seguimiento de syllabus no cambian frecuentemente.

1.2. DESARROLLO DEL SISTEMA

1.2.1.METODOLOGÍA DE DESARROLLO DEL SISTEMA

METODOLOGÍA XP

Es una metodología ágil centrada en obtener un ambiente amigable y cómodo entre el equipo de trabajo como factor importante para el éxito en desarrollo de software, preocupándose por el aprendizaje del equipo de trabajo, y manteniendo un clima de trabajo adecuado. XP tiene como pilar fundamental la retroalimentación permanente entre el cliente y el equipo de trabajo, así como la comunicación continua, simplificar las soluciones implementadas y la perseverancia para enfrentar los cambios. XP es aplicable para proyectos con requerimientos inestables e imprecisos.

PRINCIPIOS

1. **Simplicidad.-** La simplicidad consiste en el desarrollo de un sistema que sea necesario, además de solventar en cada momento las necesidades actuales.
2. **Feedback.-** Una metodología basada en el desarrollo iterativo de pequeñas soluciones, con entregables y pruebas frecuentes y continuas, proporcionando un proceso de retroalimentación importante para la detección de problemas o desviaciones.
3. **Decisión.-** Existen 3 características principales que son:
 - a. Implica tomar decisiones difíciles.
 - b. Reparación de inconsistencias cuando se las detecta.
 - c. Mejorar el código siempre tras el feedback y las sucesivas iteraciones.
4. **Comunicación.-** Algunos problemas tiene origen en que alguien no mencionó puntos importantes durante el desarrollo del proyecto. XP controla este tipo de problemas de comunicación directa y continua entre los clientes y equipo de trabajo.

CARACTERÍSTICAS

- Desarrollo iterativo e incremental: mejoramiento continuo.
- Manejo de pruebas continuas, incluyendo pruebas de regresión.
- Frecuente integración del equipo de programación con el cliente o usuario. Es importante que un cliente esté dispuesto a trabajar junto al equipo de desarrollo.
- Corrección de todos los errores antes de añadir nueva funcionalidad. Hacer entregas frecuentes.

ROLES

- **Programador**
 - Pieza básica en desarrollos XP.
 - Más responsabilidad que en otros modos de desarrollo.
 - Responsable sobre el diseño (Refactorizar, simplicidad).
 - Comunicación clara y directa.
 - Aceptar críticas (código colectivo).
- **Cliente**
 - Pieza básica en desarrollos XP.
 - Define especificaciones.
 - Influye sin controlar.
 - Confía en el grupo de desarrollo.
 - Define pruebas funcionales.

- **Encargado de pruebas (Tester)**
 - Apoya al cliente en la preparación/realización de las pruebas funcionales.
 - Ejecución de pruebas funcionales y publicación los resultados

- **Encargado de seguimiento (Tracker)**
 - Recoger, analizar y publicar información durante el proyecto sin afectar al proceso.
 - Supervisa el cumplimiento de las estimaciones de cada iteración.
 - Informa sobre la iteración en curso.
 - Controla la marcha de las pruebas funcionales, de los errores reportados, de las responsabilidades aceptadas y de las pruebas añadidas por los errores encontrados.

- **Entrenador (Coach)**
 - Experto en XP.
 - Responsable del proceso en su conjunto.
 - Identifica las desviaciones y reclama atención sobre las mismas.
 - Guía al grupo en forma indirecta (Sin dañar su seguridad ni confianza).
 - Interviene directamente si es necesario.

- **Gestor (Big Boss)**
 - Favorece la relación entre usuarios y desarrolladores.

- Confía en el equipo XP.
- Cubre las necesidades del equipo XP.
- Aseguramiento del alcance de sus objetivos.

FASES

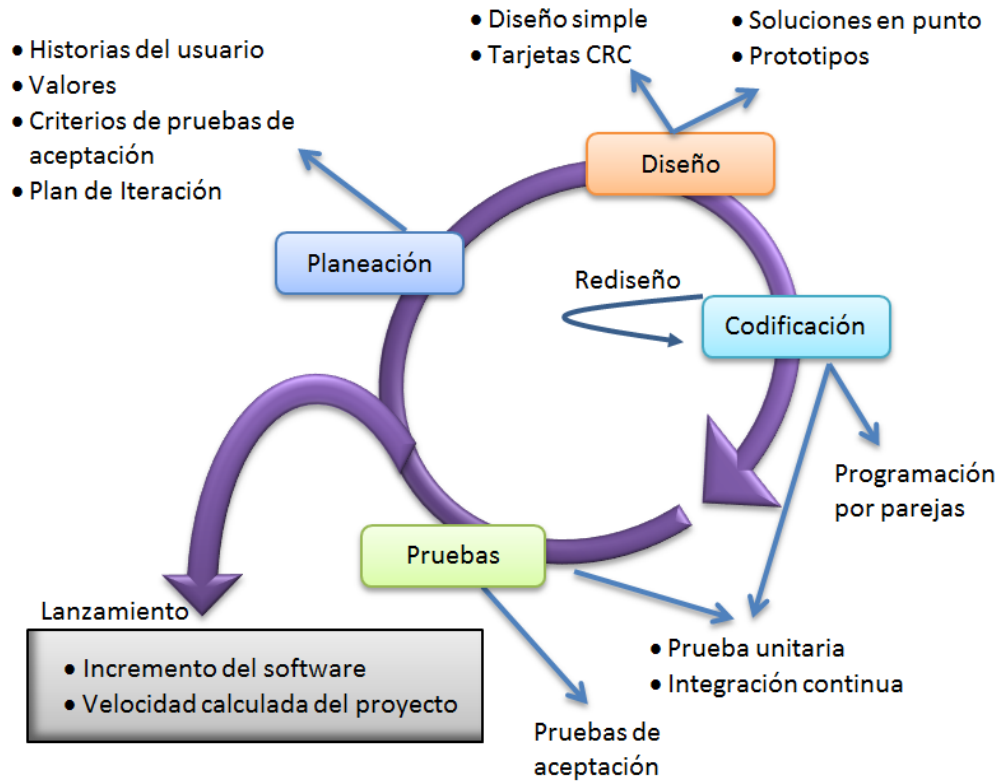


Gráfico1: Fases de la metodología XP

1ª Fase: Planificación del proyecto

Historias de usuario: Lo primero en que se debe realizar usando la metodología XP es la definición de las historias de usuario con el cliente, su objetivo es similar a los casos de uso pero con pequeñas diferencias.

Las historias de usuario son 3 ó 4 líneas escritas en un lenguaje común de manera general de cuáles son las necesidades; no se debe mencionar nada referente a la parte técnica. Son usadas para definir tiempos de desarrollo de los entregables de la aplicación que describen.

También son utilizados en la fase de pruebas, para validar si el programa cumple con lo especificado en la historia de usuario.

En la implementación de una historia de usuario, el cliente y el equipo de trabajo se reúnen para detallar la funcionalidad de dicha historia. El tiempo de desarrollo de una historia de usuario es de 1 a 3 semanas.

Release planning: Una vez definidas las historias de usuario, es imprescindible crear un plan de publicaciones; conocido también como "Release plan", donde se debe indicar las historias de usuario que se desarrollarán por versión y además de fechas de entregas.

Un "Release plan" es donde el equipo de trabajo y clientes coordinan fechas de implementación de las historias de usuario, la importancia de cada una de las historias de usuario desarrolladas e implementadas además de definir cuáles serán implementadas en cada versión.

Iteraciones: El proyecto se debe dividir en iteraciones de al menos 3 semanas de duración. En cada iteración se hace la selección de las historias de usuario definidas en el Plan de Publicaciones para que sean implementadas.

Además se debe seleccionar las historias de usuario que no pasaron las pruebas de aceptación de la iteración anterior. Las historias de usuario se dividen en tareas con una duración de entre 1 y 3 días asignadas al equipo de trabajo.

Velocidad del proyecto: Es una manera de medir la prontitud con la que se va desarrollando el proyecto; la estimación es muy sencilla, para determinar la velocidad se debe contar el número de historias de usuario que se desarrollan e implementan en una iteración; de tal forma, se puede determinar cuántas historias de usuario se pueden desarrollar en cada iteración.

Programación en pareja: La metodología XP aconseja la programación en parejas por la única razón que aumenta tanto la calidad como la productividad del software desarrollado.

Trabajar en pareja involucra a dos desarrolladores trabajando en el mismo equipo; mientras uno de ellos codifica tomando en cuenta los estándares para así implementar una función o método de calidad, mientras el otro analiza si ese método o función es apropiado y está bien diseñado. De esta forma se consigue un código y diseño de gran calidad.

Reuniones diarias: Es importante dentro de esta metodología que los desarrolladores se reúnan diariamente y expongan sus problemas, soluciones e ideas de forma conjunta.

2ª Fase: Diseño

Diseños simples: Se debe conseguir diseños simples y sencillos, para lograr dicho objetivo se procura hacer todo lo menos complicado para conseguir un diseño entendible e implementable que a la larga costará menos tiempo y esfuerzo desarrollar.

Glosarios de términos: Usar glosarios de términos y una correcta especificación de los nombres de métodos y clases ayudará a comprender el diseño y facilitará sus posteriores ampliaciones y la reutilización del código.

Riesgos: En cuanto se generan problemas durante el diseño, se aconseja que una pareja de desarrolladores investiguen y reduzcan el riesgo que se genera en el problema.

Funcionalidad extra: No es recomendable añadir funcionalidad extra al desarrollo a pesar de que se piense que en un futuro será utilizada. Es importante mencionar que el 10% de la misma es utilizada, por tal motivo se convierte en un desperdicio de tiempo y recursos.

Refactorizar: Mejora, modifica la estructura y codificación de código ya creado sin alteraciones a la funcionalidad. Refactorizar es la revisión de códigos para procurar optimizar su funcionamiento. Es muy común la reutilización de códigos ya creados que contienen funcionalidades que no serán usadas y diseños obsoletos; cuando se usa estos códigos ya creados es importante la refactorización para no tener código inestable y mal diseñado.

3ª Fase: Codificación.

Es importante que la codificación se haga bajo estándares de codificación ya creados. Programar bajo estándares mantiene el código consistente y facilita su comprensión y escalabilidad.

XP maneja programación en pareja para obtener un código más eficiente y con una gran calidad. XP maneja repositorios de código para que las parejas de programadores coloquen sus códigos listos para ser puestos a prueba.

De tal manera el resto del equipo de trabajo trabajará con códigos que tengan la última versión. Para asegurar un código consistente esta acción es solo para las parejas de programadores.

XP maneja un modelo de desarrollo en equipo en el cuál los desarrolladores están directamente ligados a las diferentes tareas; de tal forma que el programador haga modificaciones o ampliaciones a una clase o método de otro desarrollador siempre y cuando este suba sus modificaciones al repositorio.

El resto de los desarrolladores pueden hacer modificaciones de códigos que no son suyos porque tienen la ventaja que la publicación de un código primero debe pasar por las pruebas de funcionamiento.

En las pruebas se especifican las funcionalidades que se debe probar de acuerdo al plan de publicaciones.

4ª Fase: Pruebas

Uno de los pilares fundamentales de la metodología XP es el uso de pruebas para comprobar el funcionamiento de los códigos que se vayan implementando.

El uso de las pruebas en XP es el siguiente:

- Crear las aplicaciones que realizarán las pruebas en un entorno de desarrollo específico para pruebas.
- Someter a pruebas las distintas clases del sistema excluyendo los métodos más triviales.
- Crear las pruebas que pasarán los códigos antes de implementarlos.

Un punto importante es crear pruebas que no tengan ninguna dependencia del código fuente que en un futuro será evaluado. Hay que crear las pruebas abstrayéndose del futuro código, de esta forma se asegura la independencia de la prueba respecto al código que evalúa.

El uso de las pruebas es importante para identificar claramente las diferentes refactorizaciones que se hace al código. Las pruebas ayudan a validar que el cambio en la estructura de un código no afecte al funcionamiento.

Al ser las distintas funcionalidades de la aplicación poco extensas, no se realizarán pruebas para analizar las partes de las mismas, sino que las pruebas se realizarán en forma general para validar que se cumpla con la descripción de los requerimientos específicos.

1.2.2. HERRAMIENTAS DE DESARROLLO

Microsoft .NET

“.NET es un framework de Microsoft que hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones. Basado en ella, la empresa intenta desarrollar una estrategia horizontal que integre todos sus productos, desde el sistema operativo hasta las herramientas de mercado.” (Wikiperdia, 2014)

CARACTERISTICAS .NET

- Proporciona un entorno orientado a programación orientada a objetos, de tal manera que los objetos se almacenan y ejecutan de forma local.
- Ejecución de código segura.
- Ejecución de código en un entorno de mejoramiento de rendimiento.
- El entorno permite desarrollar aplicaciones tanto en Windows como en Web.
- La comunicación está basada en estándares de comunicación para asegurar la integración de la aplicación con otros códigos de programación.

“La biblioteca de clases de .NET Framework es una colección de tipos reutilizables que se integran estrechamente con Common Language Runtime. La biblioteca de clases está orientada a objetos, lo que proporciona tipos de los que su propio código administrado puede derivar funciones. Esto ocasiona que los tipos de .NET Framework sean sencillos de utilizar y reduce el tiempo asociado con el aprendizaje de las nuevas características de .NET Framework. Además, los componentes de terceros se pueden integrar sin dificultades con las clases de .NET Framework.” (Microsoft Developer Network, 2014)



Gráfico 3: características principales de .NET Framework

Microsoft SQL SERVER 2005

“Microsoft SQL Server es un sistema para la gestión de bases de datos producido por Microsoft basado en el modelo relacional. Sus lenguajes para consultas son T-SQL y ANSI SQL. Microsoft SQL Server constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como son Oracle, PostgreSQL o MySQL.”. (Wikiperdia, 2014)

Características de Microsoft SQL SERVER 2005

- Soporta varias transacciones a la vez.
- Soporta ejecución de stored procedure.
- Uso de comandos DDL y DML.
- Trabaja en modo cliente-servidor, de tal forma que los datos están disponibles a todos los usuarios.
- Permite la administración de otros servidores de datos.

CAPÍTULO II

DESARROLLO

2.1. INTRODUCCIÓN

El sistema de seguimiento de Syllabus es un sistema orientado a la Web, tomando en cuenta que está desarrollado en C# como lenguaje de programación en la plataforma de desarrollo Visual Studio 2010.

El sistema está dividido en 4 módulos que son:

- Módulo de Planificación Curricular.
- Módulo de Control de Asistencia de Alumnos.
- Módulo de reporte de Planificación Curricular.
- Módulo de reporte de Asistencia de Alumnos.

La base de datos de la aplicación está desarrollada en el DMBS SQLServer 2005.

Se usa la metodología XP porque es una metodología que se ajusta a clientes con cambios frecuentes en los requerimientos del sistema, además que tiene una relación directa entre el cliente y el programador lo que ayuda a la depuración de errores en cada iteración.

En cuanto al levantamiento de requerimientos relacionados con la metodología XP aplicada se especifica historias de usuario a continuación:

2.2. HISTORIAS DE USUARIO

Historias de Usuario	
Número: 1	Usuario: Usuario y administrador
Nombre Historia: Autenticación usuarios	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Medio
Puntos Estimados: Autenticación de 2 tipos de usuario	Iteración Asignada: 1
Programador Responsable: Lily Flor	
Descripción: Permitir acceder a la aplicación una vez registrado y autenticado	
Observaciones:	

Historias de Usuario	
Número: 2	Usuario: Usuario
Nombre Historia: Planificación Curricular	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Medio
Puntos Estimados: Agregar la planificación curricular	Iteración Asignada: 1
Programador Responsable: Lily Flor	
Descripción: Agregar la planificación curricular de acuerdo a la asignatura asignada al docente.	
Observaciones:	

Historias de Usuario	
Número: 3	Usuario: Usuario
Nombre Historia: Asistencia Alumnos	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Medio
Puntos Estimados: Validar que el listado sea de acuerdo a la modalidad y sección de la asignatura	Iteración Asignada: 1
Programador Responsable: Lily Flor	
Descripción: Llevar un control de los alumnos que asistan, llevar el control de acuerdo a la planificación ingresada y los recursos usados.	
Observaciones:	

Historias de Usuario	
Número: 4	Usuario: Usuario
Nombre Historia: Reporte de planificación	
Prioridad en Negocio: Media	Riesgo en Desarrollo: Medio
Puntos Estimados: Reporte de la planificación curricular	Iteración Asignada: 1
Programador Responsable: Lily Flor	
Descripción: Generar un reporte de la planificación curricular de acuerdo a la asignatura	
Observaciones:	

Historias de Usuario	
Número: 5	Usuario: Usuario
Nombre Historia: Reporte de Asistencia	
Prioridad en Negocio: Media	Riesgo en Desarrollo: Medio
Puntos Estimados: Reporte de la asistencia de alumnos	Iteración Asignada: 1
Programador Responsable: Lily Flor	
Descripción: Generar un reporte de la asistencia de los alumnos de acuerdo a la planificación curricular.	
Observaciones:	

Historias de Usuario	
Número: 6	Usuario: Administrador
Nombre Historia: Mantenimiento de tablas	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Alto
Puntos Estimados: Insert, Update, Delete	Iteración Asignada: 1
Programador Responsable: Lily Flor	
Descripción: Insertar, modificar y eliminar las tablas de Recursos, Semanas Clases y Parciales	
Observaciones:	

2.3. PRUEBAS DE ACEPTACIÓN

Caso de Prueba de Aceptación	
Número: 1	Historia de Usuario #1, Autenticación usuarios
Nombre: Probar autenticación correcta de usuarios	
Descripción: Se probará que se cree usuarios y se autentiquen correctamente de acuerdo a lo registrado en la creación de usuarios	
Condiciones de Ejecución: <ul style="list-style-type: none"> • La prueba se realizará en varios ordenadores. • Se valida que se grabe en la base de datos los usuarios creados. • Se valida que no se pueda volver a registrar un usuario. 	
Entrada/Pasos de Ejecución: <ul style="list-style-type: none"> • Creación de usuarios y contraseñas. • Validación de condiciones. • Ingreso a la página luego de la autenticación. 	
Resultado Esperado: Correcta creación y validación de usuarios.	
Evaluación de la Prueba: Positiva	

Caso de Prueba de Aceptación	
Número: 2	Historia de Usuario #2, Planificación Curricular
Nombre: Probar inserción, modificación de planificación curricular	
Descripción: Se probará que se pueda insertar y modificar la planificación curricular de acuerdo a las asignaturas asignadas al docente en el periodo vigente.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Inserción de la planificación de acuerdo a las asignaturas vigentes para el período activo. • Modificación de la planificación. • Validación del registro en las tablas de la base de datos. 	
Entrada/Pasos de Ejecución: <ul style="list-style-type: none"> • Inserción de la planificación curricular de acuerdo a las asignaturas asignadas en el período activo. • Modificación de la planificación curricular agregada anteriormente. 	
Resultado Esperado: Correcta creación y modificación de la planificación curricular.	
Evaluación de la Prueba: Positiva	

Caso de Prueba de Aceptación	
Número: 3	Historia de Usuario #3, Asistencia Alumnos
Nombre: Probar inserción de la asistencia de los alumnos.	
Descripción: Se probará que se pueda insertar la asistencia de los alumnos matriculados en la asignatura de acuerdo a la modalidad y sección.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Inserción de la asistencia de los alumnos. • Inserción del contenido dictado en la clase. • Validación del registro en las tablas de la base de datos. 	
Entrada/Pasos de Ejecución: <ul style="list-style-type: none"> • Inserción de la asistencia de los alumnos de acuerdo a la modalidad y sección. • Inserción de los contenidos de acuerdo a la semana de clases. 	
Resultado Esperado: Correcta inserción de la asistencia y contenidos de acuerdo a la modalidad y sección de la asignatura asignada al docente.	
Evaluación de la Prueba: Positiva	

Caso de Prueba de Aceptación	
Número: 4	Historia de Usuario #3, Reporte de planificación
Nombre: Validación del reporte de planificación.	
Descripción: Se validará que los datos mostrados en el reporte sean los correctos de acuerdo a la base de datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Validación de que el reporte sea de acuerdo a lo insertado en la base de datos. 	
Entrada/Pasos de Ejecución: <ul style="list-style-type: none"> • Validación que la información que se muestra en el reporte sea la misma de la base de datos. 	
Resultado Esperado: Validación de la información de acuerdo a la base de datos.	
Evaluación de la Prueba: Positiva	

Caso de Prueba de Aceptación	
Número: 5	Historia de Usuario #3, Reporte de asistencia
Nombre: Validación del reporte de asistencia.	
Descripción: Validar que los datos mostrados en el reporte sean los correctos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Validación de que el reporte sea de acuerdo a lo insertado en la base de datos. 	
Entrada/Pasos de Ejecución: <ul style="list-style-type: none"> • Validación de que la información que se muestra en el reporte sea la misma de la base de datos de acuerdo al número de contenidos ingresados. 	
Resultado Esperado: Validación de la información de acuerdo a la base de datos.	
Evaluación de la Prueba: Positiva	

Caso de Prueba de Aceptación	
Número: 6	Historia de Usuario #3, Mantenimiento de tablas
Nombre: Inserción, Modificación, Eliminación de las tablas de mantenimiento.	
Descripción: Probar que se inserte, modifique y elimine los datos.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Inserción de información en las tablas de mantenimiento. • Modificación de la información en las tablas de mantenimiento. • Eliminación de la información de las tablas de mantenimiento. 	
Entrada/Pasos de Ejecución: <ul style="list-style-type: none"> • Validación que la información que se inserta, modifica o elimina en las tablas de mantenimiento. 	
Resultado Esperado: La información insertada, modificada y eliminada es correcta.	
Evaluación de la Prueba: Positiva	

2.4. MODELO ENTIDAD RELACIÓN

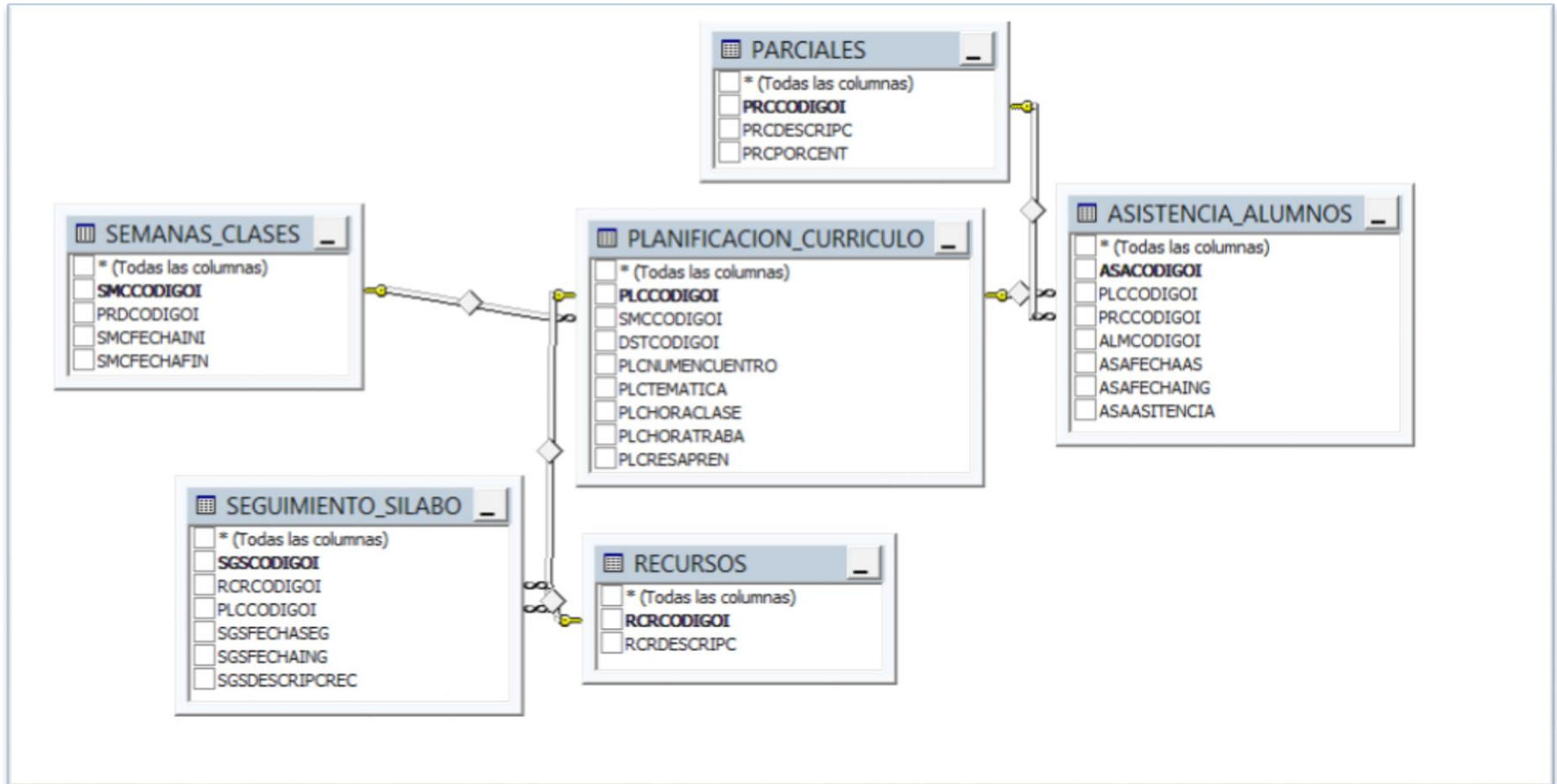


Gráfico 2: Modelo Entidad Relación

2.5. DICCIONARIO DE DATOS

En el diccionario de datos se muestra el tipo y la longitud de los datos de las diferentes tablas que se usan dentro de la base de datos.

Es importante mencionar que de esta forma el programador tendrá una mejor manipulación de la información de la base de datos.

ASISTENCIA_ALUMNOS			
Column name	Data Type	PK	FK
ASACODIGOI	INT	✓	
PLCCODIGOI	BIGINT		✓
PRCCODIGOI	INT		✓
ALMCODIGOI	BIGINT		✓
ASAFECHAAS	DATETIME		
ASAFECHAING	DATETIME		
ASAASISTENCIA	BIT		

TABLA 1: Asistencia_Alumnos

PARCIALES			
Column name	Data Type	PK	FK
PRCCODIGOI	INT	✓	
PRCDESCRIPC	VARCHAR(50)		
PRCPORCENT	INT		

TABLA 2: Parciales

PLANIFICACIÓN_CURRÍCULO			
Column name	Data Type	PK	FK
PLCCODIGOI	BIGINT	✓	
SMCCODIGOI	INT		✓
DSTCODIGOI	BIGINT		✓
PLCNUMENCUENTRO	INT		
PLCTEMATICA	VARCHAR(150)		
PLCHORACLASE	INT		
PLCHORATRABA	INT		
PLCRESAPREN	VARCHAR(150)		

TABLA 3: Planificacion_Curriculo

RECURSOS			
Column name	Data Type	PK	FK
RRCODIGOI	INT	✓	
RRCDESCRIPC	VARCHAR(50)		

TABLA 4: Recursos

SEGUIMIENTO_SILABO			
Column name	Data Type	PK	FK
SGSCODIGOI	INT	✓	
RCRCODIGOI	INT		✓
PLCCODIGOI	BIGINT		✓
SGSFECHASEG	DATETIME		
SGSFECHAING	DATETIME		
SGSDESCRIPCRC	VARCHAR(60)		

TABLA 5: Seguimiento_Silabo

SEMANAS_CLASES			
Column name	Data Type	PK	FK
SMCCODIGOI	INT	✓	
PRDCODIGOI	INT		✓
SMCFECHAINI	DATE		✓
SMCFECHAFIN	DATE		

TABLA 6: Semanas_Clases

3. CONCLUSIONES

- De acuerdo al análisis de la problemática de la institución se llega a la conclusión que; el sistema con el que trabajan actualmente es poco dinámico, tomando en cuenta que deben escoger tanto el nombre del docente como la asignatura además que no tiene un control en cuanto a los contenidos ingresados.
- En cuanto al mejoramiento del procesamiento de la información se llega a la conclusión que; se procesa la información ingresada de tal forma que se puede generar reportes de lo ingresado en el sistema tanto de la planificación curricular como de la asistencia de los alumnos. Es importante mencionar que el listado de alumnos está atado a la matrícula del estudiante por tanto se tendrá un listado de alumnos actualizado.
- En cuanto a la automatización del proceso de seguimiento de syllabus se llega a la conclusión que; se automatiza de una manera dinámica el proceso de seguimiento de syllabus de tal manera que tiene varios módulos de administración de contenidos curriculares y asistencia de alumnos.
- En cuanto al caso de aplicación en la Universidad Tecnológica Israel se llega a la conclusión que; con la implementación del sistema se optimiza el proceso de seguimiento de syllabus en la institución.

4. RECOMENDACIONES

Se recomienda:

- Realizar respaldos periódicos de los datos y la aplicación.
- Implementar la aplicación con el fin de realizar las pruebas y corregir errores.
- Estandarizar los procesos de la institución.

5. BIBLIOGRAFÍA

REFERENCIAS

- Cadena, F. (28 de Noviembre de 2013). Universidades del Ecuador. Diario el Mercurio.
- Ecuador, C. d. (2008). Transitoria Primera.
- Ecuador, C. d. (2008). Transitoria Vigésima.
- Ramirez, R. (2013 de Noviembre de 2013). Universidades del Ecuador. Diario El Universo
- Superior, R. G. (2012). Artículo 15.
- Wikipedia. (29 de Agosto de 2014). Wikipedia. Obtenido de http://es.wikipedia.org/wiki/Microsoft_SQL_Server
- Wikiperdia. (29 de Agosto de 2014). Wikipedia. Obtenido de http://es.wikipedia.org/wiki/Microsoft_.NET

LIBROS

- Microsoft .NET Enterprise Solutions Patterns & Practices. (Microsoft Corporation) 2008.
- Advanced .NET Debugging. (Microsoft Global Solution Support Team, Student Workbook Curriculum).

WEB

- http://ingenieriadesoftware.mex.tl/52753_XP---Extreme-Programing.html
- <http://ingsoftware072301.obolog.es/metodologia-xp-2012877>

- <http://es.slideshare.net/Piskamen/metodologa-xp>
- <http://es.slideshare.net/LisPater1/metodologias-agiles-xp>
- <http://programacionextrema.tripod.com/fases.htm>
- [http://msdn.microsoft.com/es-es/library/zw4w595w\(v=vs.110\).aspx](http://msdn.microsoft.com/es-es/library/zw4w595w(v=vs.110).aspx)
- <https://www.microsoft.com/latam/technet/productos/servers/sql/2005/overview.msp>

GRÁFICOS

- <http://www.codejobs.biz/es/blog/2013/06/05/programacion-extrema-xp#sthash.3X4Yo7Ki.dpbs>
- [http://msdn.microsoft.com/es-es/library/zw4w595w\(v=vs.110\).aspx](http://msdn.microsoft.com/es-es/library/zw4w595w(v=vs.110).aspx)

6. ANEXOS

ENTREVISTA

1. ¿Cómo se lleva actualmente el seguimiento de syllabus a los docentes?
2. ¿Qué tan confiable es?
3. ¿Con qué frecuencia se cambia los formatos de seguimiento de syllabus?
4. ¿Qué impacto genera llevar un seguimiento de syllabus verdadero?
5. ¿Existe algún sistema informático de seguimiento de syllabus?

MANUAL DE USUARIO

INTRODUCCIÓN

Este manual tiene como principal objetivo explicar de una manera específica y sencilla al usuario cómo funciona el sistema, de esta manera el usuario tendrá la capacidad de manipular correctamente el sistema.

1. ACCESO A LA APLICACIÓN

En la siguiente imagen se muestra el acceso (login) del Sistema, donde se introduce el usuario y contraseña.

El diagrama muestra la interfaz de login con un encabezado azul que dice "LOGIN". Hay dos campos de entrada: "USUARIO:" y "CONTRASEÑA:". El campo de usuario está etiquetado como "Usuario del Sistema" y el campo de contraseña como "Contraseña". Debajo de los campos hay un botón con un icono de un hombre en un traje y una llave, etiquetado como "Botón de Login".

2. REGISTRO DE USUARIOS

Esta es la página principal al presionar el botón "Regístrate Aquí" que se encuentra en la parte superior derecha se mostrará el siguiente formulario:

El diagrama muestra la interfaz de registro con un encabezado azul que dice "REGISTRO". Hay cinco campos de entrada: "CEDULA:", "DOCENTE:", "USUARIO:", "CONTRASEÑA:" y "REPITA LA CONTRASEÑA:". A la derecha de los campos hay un botón azul que dice "REGISTRAR", etiquetado como "Botón de Registro". Debajo de los campos hay un icono de un documento con una llave.

Insertar la cédula del docente a registrar.

Al presionar el botón “REGISTRAR” se desplegará el nombre del docente como se muestra en la siguiente imagen:

REGISTRO

CEDULA: **REGISTRAR**

DOCENTE: PASTOR PASALODOS OSWALDO MIGUEL

USUARIO:

CONTRASEÑA:

REPITA LA CONTRASEÑA:



REGISTRO

CEDULA: **REGISTRAR**

DOCENTE: PASTOR PASALODOS OSWALDO MIGUEL

USUARIO:

CONTRASEÑA:

REPITA LA CONTRASEÑA:

 Botón Guardar

Llenar los campos de “USUARIO”, “CONTRASEÑA” y “REPITA LA CONTRASEÑA”.

3. Organización del Sistema de Seguimiento de Syllabus

[Cerrar Sesión](#)


DATOS GENERALES DOCENTE

DOCENTE: VALLE BASTIDAS WILMER RAMIRO
 CEDULA: 1709343071
 PERIODO: 01/10/2013 - 31/03/2014



	CODIGO	ASIGNATURA	PARALELO	SYLLABUS	CONTROL	PLANIFICACION	ASISTENCIA
SELECCIONAR	159	APLICACIONES DISTRIBUIDAS II	A				
SELECCIONAR	310	LECTURA CRITICA	B				
SELECCIONAR	358	FUNDAMENTOS DE ADMINISTRACION	A				
SELECCIONAR	388	PLATAFORMAS DE DESARROLLO I	A				
SELECCIONAR	389	PROGRAMACION WEB II	A				
SELECCIONAR	452	TALLER DE METODOLOGIA DE LA INVESTIGACION APLICADA II	A				

IMAGEN	DESCRIPCIÓN
	Botón de inserción de planificación curricular.
	Botón de Control de Asistencia.
	Reporte de la Planificación insertada.
	Reporte de la Asistencia de Alumnos.
SELECCIONAR	Botón de Selección de Asignatura.
Cerrar Sesión	Botón para Cerrar Sesión ubicado en la parte superior izquierda.

4. INSERCIÓN DE LA PLANIFICACIÓN CURRICULAR

Al presionar el botón () se desplegará la siguiente página.



DATOS GENERALES DOCENTE	DATOS SYLLABUS
DOCENTE: VALLE BASTIDAS WILMER RAMIRO CEDULA: 1709343071 PERIODO: 01/10/2013 - 31/03/2014 ASIGNATURA: LECTURA CRITICA	SEMANA ACADÉMICA CURRICULAR: 2014-07-28 <--> 2014-08-01 ▼ NÚMERO DE ENCUENTRO: <input type="text"/> TEMATICA A DESARROLLAR: <input type="text"/> NUMERO DE HORAS DE CLASES PRESENCIALES: <input type="text"/> NUMERO DE HORAS DE TRABAJO AUTONOMO: <input type="text"/> RESULTADOS: <input type="text"/>
	

	NUMERO	FECHAS	TEMATICA	CLASES	TRABAJO	RESULTADO
SELECCIONAR	2	2014-07-21 <--> 2014-07-25	PRUEBA 1234	3	3	EL ESTUDIANTE
SELECCIONAR	3	2014-07-28 <--> 2014-08-01	nosenose	3	4	NO SE
SELECCIONAR	4	2014-07-02 <--> 2014-08-03	PRUBAS	4	4	NO SE

La distribución de la página es la siguiente:

4.1. INFORMACIÓN BÁSICA DEL DOCENTE

DATOS GENERALES DOCENTE

DOCENTE: VALLE BASTIDAS WILMER RAMIRO
CEDULA: 1709343071
PERIODO: 01/10/2013 - 31/03/2014
ASIGNATURA: LECTURA CRITICA

4.2. FORMULARIO DE INSERCIÓN DE INFORMACIÓN DE LA PLANIFICACIÓN CURRICULAR

DATOS SYLLABUS

SEMANA ACADÉMICA CURRICULAR: 2014-07-28 <--> 2014-08-01 ▾


NÚMERO DE ENCUENTRO:


TEMATICA A DESARROLLAR:

NUMERO DE HORAS DE CLASES PRESENCIALES:

NUMERO DE HORAS DE TRABAJO AUTONOMO:

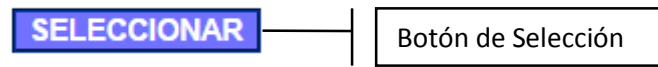
RESULTADOS:

 Botón Agregar

 Botón Modificar

4.3. LISTADO DE CONTENIDOS

	NUMERO	FECHAS	TEMATICA	CLASES	TRABAJO	RESULTADO
SELECCIONAR	2	2014-07-21 <--> 2014-07-25	PRUEBA 1234	3	3	EL ESTUDIANTE
SELECCIONAR	3	2014-07-28 <--> 2014-08-01	nosenose	3	4	NO SE
SELECCIONAR	4	2014-07-02 <--> 2014-08-03	PRUBAS	4	4	NO SE



Al momento de seleccionar un contenido de la planificación curricular se mostrará la siguiente información en la sección 4.2.

DATOS SYLLABUS

SEMANA ACADÉMICA CURRICULAR:



NÚMERO DE ENCUENTRO:


TEMATICA A DESARROLLAR:

NUMERO DE HORAS DE CLASES PRESENCIALES:

NUMERO DE HORAS DE TRABAJO AUTONOMO:

RESULTADOS:




Una vez modificada la información que se desea se presionará el botón ().

Para regresar a la página principal se debe presionar el botón “HOME” que tiene la siguiente imagen:



5. CONTROL DE ASISTENCIA A ALUMNOS

Al presionar el botón () en el menú de inicio se desplegará la siguiente información.

El control de asistencia a alumnos es donde se manejará la asistencia y el contenido que se impartirá a los estudiantes.

DATOS GENERALES DOCENTE		DATOS GENERALES CONTENIDOS	
DOCENTE:	VALLE BASTIDAS WILMER RAMIRO	PARCIAL:	EXAMEN PARCIAL 1 ▾
CEDULA:	1709343071	FECHA ACTUAL:	21/09/2014
PERIODO:	01/10/2013 - 31/03/2014	HORARIO:	SIN HORARIO ▾
		FECHA DE ASISTENCIA:	dd/mm/aaaa
		ASIGNATURA:	TALLER DE METODOLOGIA DE LA INVESTIGACION APLICADA II

LISTADO DE ALUMNOS			
MODALIDAD	SECCION	PARALELO	
SELECCIONAR	PRESENCIAL	DIURNA	A

CODIGO	CEDULA	NOMBRES	ASISTENCIA
9224	1720023025	MUÑOZ SANCHEZ VICTOR HUGO	<input checked="" type="checkbox"/>
9506	1722634563	OSORIO LATORRE JONATHAN EDGAR	<input checked="" type="checkbox"/>
12846	1723718134	QUIGUANGO DELGADO PATRICIO JAVIER	<input checked="" type="checkbox"/>
9559	1720640836	SALAZAR JACOME CHRISTIAN FABIAN	<input checked="" type="checkbox"/>
9453	1716154198	SANDOVAL SOLANO MAURICIO RAMIRO	<input checked="" type="checkbox"/>
12249	0503647026	VILLOTA BRACERO CARLOS ISAAC	<input checked="" type="checkbox"/>

CONTENIDOS ASIGNATURAS			
CONTENIDO 1:	programacion ▾	RECURSOS:	Internet ▾  
			Internet ▾ 

A continuación se detallará cada una de las partes que contiene el “CONTROL DE ASISTENCIA A ALUMNOS”:

5.1. DATOS GENERALES DEL DOCENTE

Se muestra los datos generales del docente que impartirá la asignatura

DATOS GENERALES DOCENTE

DOCENTE: VALLE BASTIDAS WILMER RAMIRO
CEDULA: 1709343071
PERIODO: 01/10/2013 - 31/03/2014

5.2. DATOS GENERALES DE CONTENIDOS

Se muestra los datos generales que se aplicará al control de los alumnos

DATOS GENERALES CONTENIDOS




PARCIAL: EXAMEN PARCIAL 1 ▼
FECHA ACTUAL: 21/09/2014
HORARIO: SIN HORARIO ▼
FECHA DE ASISTENCIA: dd/mm/aaaa
ASIGNATURA: TALLER DE METODOLOGIA DE LA INVESTIGACION APLICADA II

5.3. LISTADO DE ALUNNOS

Se muestra de acuerdo a la modalidad y sección que se seleccione presionando el botón (**SELECCIONAR**) el listado de alumnos matriculados

LISTADO DE ALUNNOS			
MODALIDAD	SECCION	PARALELO	
SELECCIONAR	PRESENCIAL	DIURNA	A
CODIGO	CEDULA	NOMBRES	ASISTENCIA
9224	1720023025	MUÑOZ SANCHEZ VICTOR HUGO	<input checked="" type="checkbox"/>
9506	1722634563	OSORIO LATORRE JONATHAN EDGAR	<input checked="" type="checkbox"/>
12846	1723718134	QUIGUANGO DELGADO PATRICIO JAVIER	<input checked="" type="checkbox"/>
9559	1720640836	SALAZAR JACOME CHRISTIAN FABIAN	<input checked="" type="checkbox"/>
9453	1716154198	SANDOVAL SOLANO MAURICIO RAMIRO	<input checked="" type="checkbox"/>
12249	0503647026	VILLOTA BRACERO CARLOS ISAAC	<input checked="" type="checkbox"/>



5.4. CONTENIDOS DE ASIGNATURA


CONTENIDOS ASIGNATURAS			
CONTENIDO 1:	<input type="text" value="programacion"/>	RECURSOS:	<input type="text" value="Internet"/>  <input type="text" value="Internet"/>  

Escoger el contenido:


CONTENIDO 1:

Escoger el recurso:

RECURSOS: una vez escogido el recurso se debe presionar el botón () o en caso contrario quitar del listado de recursos se debe presionar el botón ().

Una vez agregada toda la información se presiona el botón ()

6. REPORTE DE PLANIFICACIÓN CURRICULAR


Al presionar el botón () en el menú de inicio se desplegará la siguiente información.

DATOS GENERALES DOCENTE				
DOCENTE: VALLE BASTIDAS WILMER RAMIRO				
CEDULA: 1709343071				
PERIODO: 01/10/2013 - 31/03/2014				
ASIGNATURA: TALLER DE METODOLOGIA DE LA INVESTIGACION APLICADA II				

ENCUENTRO	TEMATICA	HORAS CLASE	HORAS TRABAJADAS	RESULTADO
1	programacion	4	4	Sera capaz de programar
2	analisis	5	5	sera capaz de analizar no se
3	No se	4	4	No se el estudiante
4	asdada	4	2	dsdadasadasd
5	DDDDDD	3	3	PUEDE
6	NO NO NO	6	6	NO NO NO NO NO NO
7	EEEEEE	3	3	HGGJFFJHFJHFJKJK
8	RRRRRRR	5	5	DSSFSDFSDFSDFSDF
9	Estructura de datos	4	4	sasadasd


Information General of the Teacher

Curricular planning of the subject

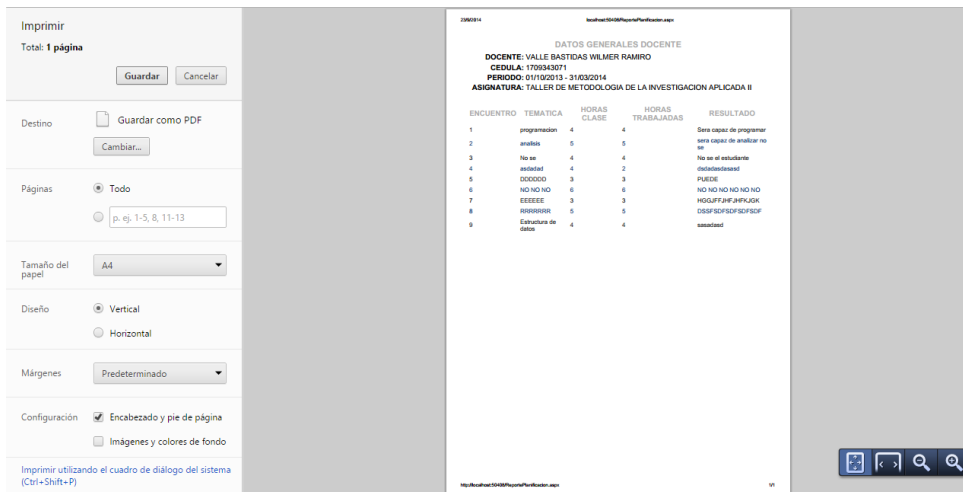
Para imprimir la planificación de la asignatura seleccionada se debe presionar el botón ().

Para regresar a la página principal se debe presionar el botón "HOME" que tiene la siguiente imagen:

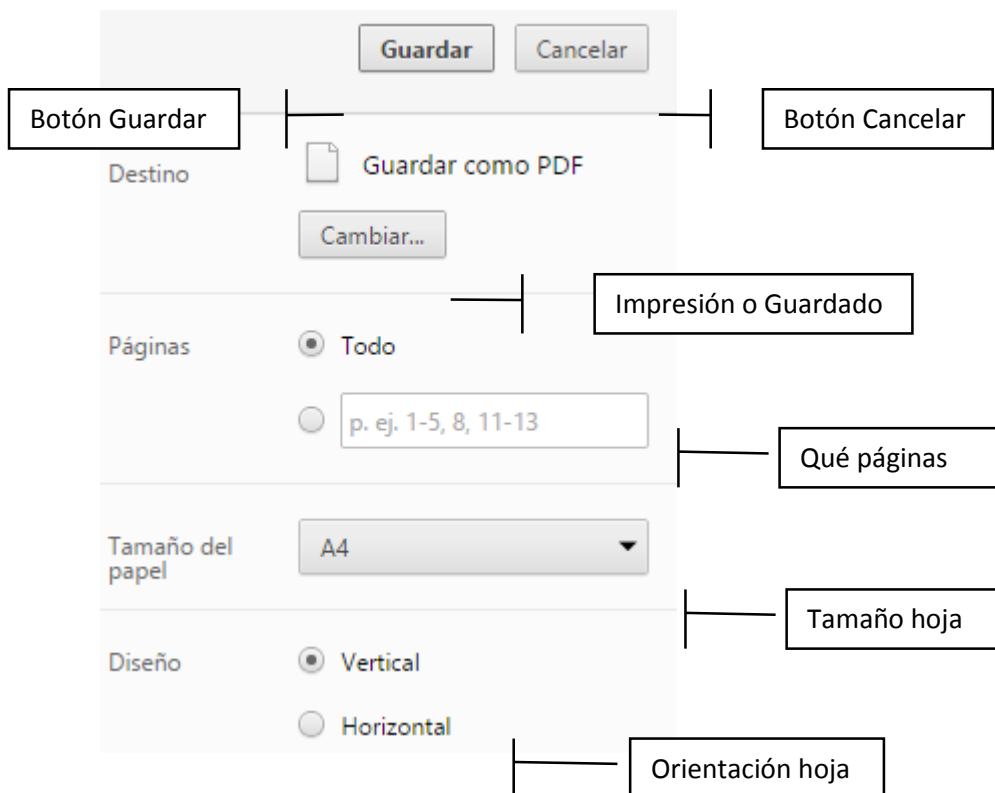


Al presionar el botón ()


Se desplegará la siguiente ventana emergente:



En esta ventana visualizará los siguientes opciones:



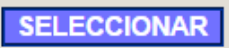
7. REPORTE ASISTENCIA ALUMNOS

Al presionar el botón () en el menú de inicio se desplegará la siguiente información.


Se mostrará la siguiente información:

DATOS GENERALES DOCENTE	
DOCENTE: VALLE BASTIDAS WILMER RAMIRO CEDULA: PERIODO: 01/10/2013 - 31/03/2014 ASIGNATURA: TALLER DE METODOLOGIA DE LA INVESTIGACION APLICADA II	Información General del Docente

MODALIDAD	SECCION	PARALELO
SELECCIONAR	PRESENCIAL	DIURNA A

Se debe presionar el botón () para seleccionar la modalidad y sección para desplegar la siguiente información:

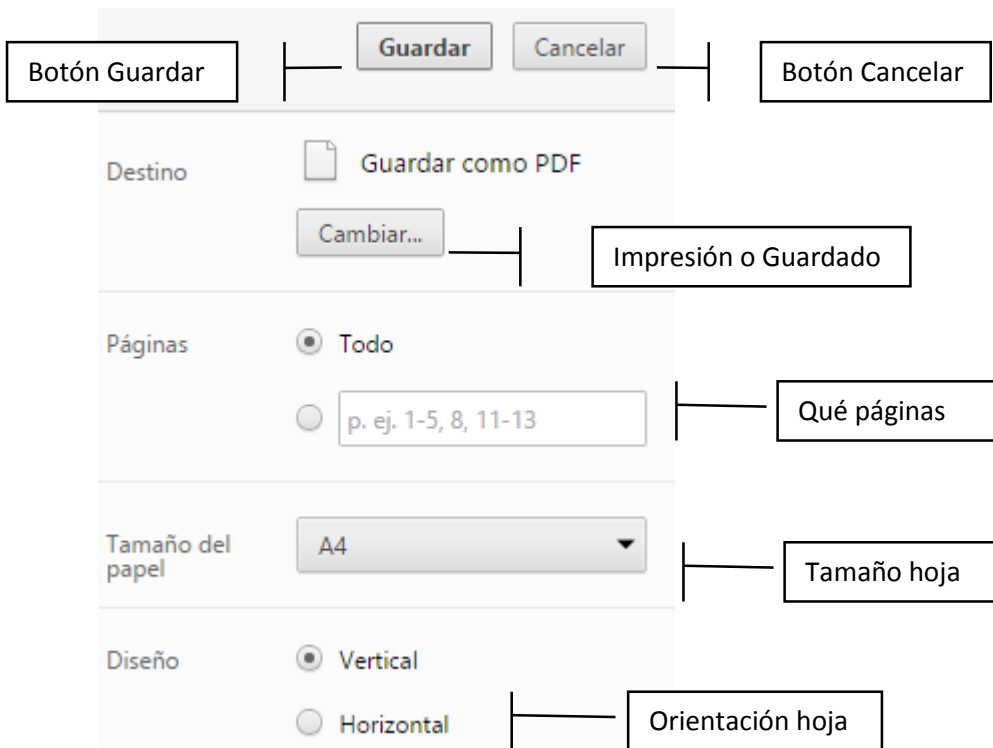
ALUMNO	1	2	3	4	5	6	7	8	9
MUÑOZ SANCHEZ VICTOR HUGO	A	A	A	A	A	A	A	A	A
OSORIO LATORRE JONATHAN EDGAR	A	A	A	A	A	A	A	A	A
QUIGUANGO DELGADO PATRICIO JAVIER	A	A	A	A	A	A	A	A	A
SALAZAR JACOME CHRISTIAN FABIAN	A	A	A	A	A	A	A	A	A
SANDOVAL SOLANO MAURICIO RAMIRO	A	A	A	A	A	A	A	A	A
VILLOTA BRACERO CARLOS ISAAC	A	A	A	A	A	A	A	A	A

Al presionar el botón ()

Se desplegará la siguiente ventana emergente:



En esta ventana visualizará los siguientes opciones:



MANUAL DE PROGRAMADOR

INTRODUCCIÓN

El presente documento tiene como objetivo principal dar a conocer al administrador y programador del sistema las principales funciones especificadas en el código fuente y que tarea hacen cada una de ellas, además que cuenta con el modelo entidad de relación de la base de datos y el diccionario de datos, con esto el programador tendrá la capacidad de solventar cualquier problema con el sistema.

1. REQUISITOS DE SOFTWARE

Los requisitos de software previos a la instalación del sistema son los siguientes:

1.1. Programador

- SQL Server 2008 o superior
- Visual Studio 2010 o superior

1.2. Cliente

- Mozilla Firefox 7 o superior
- Google Chrome 21 o superior
- Safari 6.x o superior

2. INTRODUCCIÓN

A través del tiempo el ser humano ha venido teniendo cambios radicales en su vida, de tal manera que todo lo que ha aprehendido lo ha puesto en práctica para mejorar sus condiciones de vida. Es importante mencionar la gran diferencia de modo de vida del ser humano de hace algunos años a los seres humanos de la actualidad, estas diferencias se dan por el avance tecnológico que ha tenido el mundo entero tomando en cuenta que existe una relación muy estrecha entre el ser humano y la tecnología.

Algunos de los países que han aportado directamente a la evolución de la tecnología son China, Estados Unidos, Japón, Alemania entre otros. La tecnología tiene como principal objetivo el facilitar la vida diaria a los seres humanos.

La tecnología ha tenido un impacto considerable sobre el mundo entero, por tal motivo existen muchas personas que se dedican al desarrollo de aplicaciones, sistemas que ayuden al ser humano en sus labores cotidianas.

Después de mencionar como ha venido evolucionando la tecnología de acuerdo a las necesidades de los seres humanos; se conoce la necesidad de mejorar el

control de docentes en la Universidad Tecnológica Israel porque lamentablemente la manera en la que se maneja los procesos es inadecuada y poco veraz; debido a que se realiza el seguimiento de syllabus a través de hojas sueltas que corren el riesgo de ser perdidas o adulteradas.

Por tanto el problema principal que se pretende resolver es el inadecuado proceso de seguimiento de syllabus.

3. DIAGRAMA DE LA BASE DE DATOS

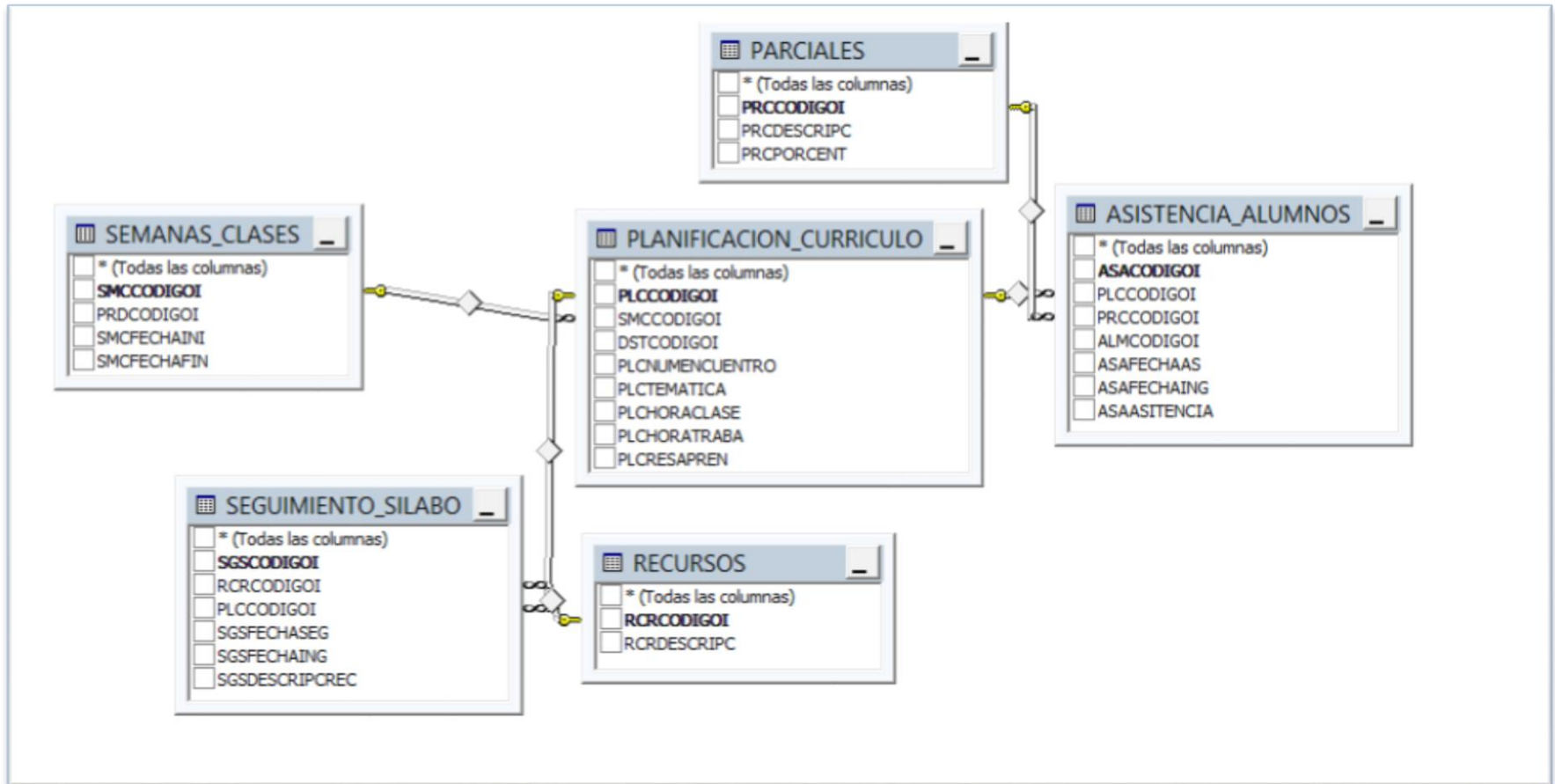


Grafico 1: Modelo Entidad Relación

4. DICCIONARIO DE DATOS

En el diccionario de datos se muestra el tipo y la longitud de los datos de las diferentes tablas que se usan dentro de la base de datos.

Es importante mencionar que de esta forma el programador tendrá una mejor manipulación de los datos de la base de datos.

ASISTENCIA_ALUMNOS			
Column name	Data Type	PK	FK
ASACODIGOI	INT	✓	
PLCCODIGOI	BIGINT		✓
PRCCODIGOI	INT		✓
ALMCCODIGOI	BIGINT		✓
ASAFECHAAS	DATETIME		
ASAFECHAING	DATETIME		
ASAASISTENCIA	BIT		

TABLA 1: Asistencia_Alumnos

PARCIALES			
Column name	Data Type	PK	FK
PRCCODIGOI	INT	✓	
PRCDESCRIPC	VARCHAR(50)		
PRCPORCENT	INT		

TABLA 2: Parciales

PLANIFICACIÓN_CURRÍCULO			
Column name	Data Type	PK	FK
PLCCODIGOI	BIGINT	✓	
SMCCODIGOI	INT		✓
DSTCODIGOI	BIGINT		✓
PLCNUMENCUENTRO	INT		
PLCTEMATICA	VARCHAR(150)		
PLCHORACLASE	INT		
PLCHORATRABA	INT		
PLCRESAPREN	VARCHAR(150)		

TABLA 3: Planificacion_Curriculo

RECURSOS			
Column name	Data Type	PK	FK
RCRCODIGOI	INT	✓	
RCRDESCRIPC	VARCHAR(50)		

TABLA 4: Recursos

SEGUIMIENTO_SILABO			
Column name	Data Type	PK	FK
SGSCODIGOI	INT	✓	
RCRCODIGOI	INT		✓
PLCCODIGOI	BIGINT		✓
SGSFECHASEG	DATETIME		
SGSFECHAING	DATETIME		
SGSDESCRIPCRC	VARCHAR(60)		

TABLA 5: Seguimiento_Silabo

SEMANAS_CLASES			
Column name	Data Type	PK	FK
SMCCODIGOI	INT	✓	
PRDCODIGOI	INT		✓
SMCFECHAINI	DATE		✓
SMCFECHAFIN	DATE		

TABLA 6: Semanas_Clases

5. DIAGRAMA DE RED

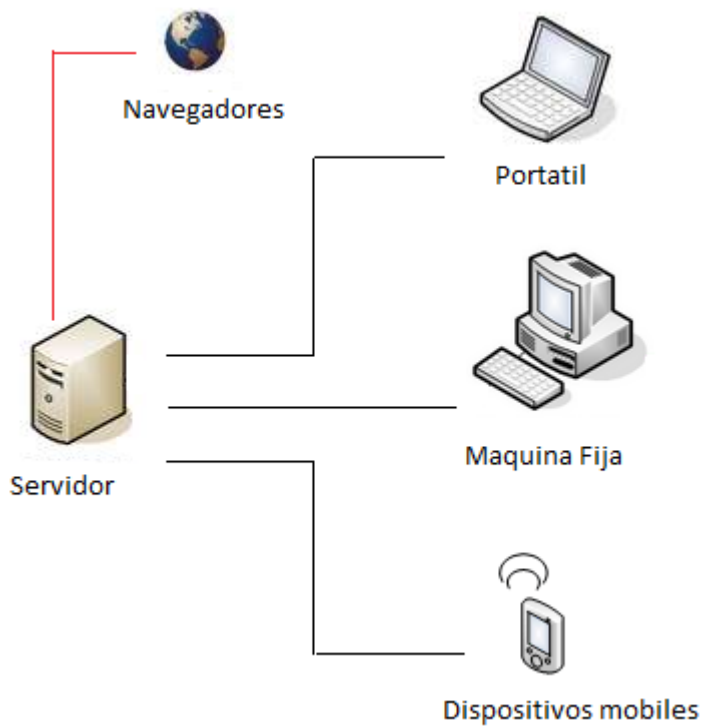


Grafico 2: Diagrama de Red

6. MANUAL DE PROGRAMADOR

Dentro de los formularios del sistema se encuentra dividido por 3 regiones que son:

- Variables, Conexión
- Eventos y Botones
- Funciones

Conexión a la base de datos

```
SqlConnection conection = new SqlConnection("Data Source = SERVIDOR;
Initial Catalog = NOMBRE_BDD ;Integrated Security = True;
MultipleActiveResultSets = true");
```

MultipleActiveResultSets.- Es para que permita múltiples reads dentro de la programación.

Control.aspx.cs			
Lenguaje:	C#	Acción:	Carga
Método:	Cargarperiodo()	Entrada:	
Descripción:	Carga el periodo habilitado		
<pre>//La funcion sirve para cargar el periodo activo public void cargarperiodo() { conection.Open(); SqlCommand com = new SqlCommand("SELECT * FROM PERIODO WHERE PRDESTADOC = 1", conection); SqlDataReader read = com.ExecuteReader(); if (read.Read()) { DateTime fechaini = Convert.ToDateTime(read["PRDFECINIF"].ToString()); string fechaformatoini = String.Format("{0:d}", fechaini); lblperiodoini.Text = fechaformatoini; DateTime fechafin = Convert.ToDateTime(read["PRDFECFINF"].ToString()); string fechaformatofin = String.Format("{0:d}", fechafin); lblperiodofin.Text = fechaformatofin; lblcodigoi.Text = read["PRDCODIGOI"].ToString(); } conection.Close(); }</pre>			

TABLA 7: Función Cargar periodo academico

Inicio.aspx.cs			
Lenguaje:	C#	Acción:	Carga
Método:	Cargarperiodo()	Entrada:	
Descripción:	Carga el periodo habilitado		
<pre>//La funcion sirve para cargar el periodo activo public void cargarperiodo() { conection.Open(); SqlCommand com = new SqlCommand("SELECT * FROM PERIODO WHERE PRDESTADOC = 1", conection); SqlDataReader read = com.ExecuteReader(); if (read.Read()) { DateTime fechaini = Convert.ToDateTime(read["PRDFECINIF"].ToString()); string fechaformatoini = String.Format("{0:d}", fechaini); lblperiodoini.Text = fechaformatoini; DateTime fechafin = Convert.ToDateTime(read["PRDFECFINF"].ToString()); string fechaformatofin = String.Format("{0:d}", fechafin); lblperiodofin.Text = fechaformatofin; lblcodigoi.Text = read["PRDCODIGOI"].ToString(); } conection.Close(); }</pre>			

TABLA 8: Función Cargar periodo académico

Parcial.aspx.cs			
Lenguaje:	C#	Acción:	Carga
Método:	Cargarperiodo()	Entrada:	
Descripción:	Carga el periodo habilitado		
<pre>//La funcion sirve para cargar el periodo activo public void cargarperiodo() {conection.Open(); SqlCommand com = new SqlCommand("SELECT * FROM PERIODO WHERE PRDESTADOC = 1", conection); SqlDataReader read = com.ExecuteReader(); if (read.Read()) { DateTime fechaini = Convert.ToDateTime(read["PRDFECINIF"].ToString()); string fechaformatoini = String.Format("{0:d}", fechaini); lblperiodoini.Text = fechaformatoini; DateTime fechafin = Convert.ToDateTime(read["PRDFECFINF"].ToString()); string fechaformatofin = String.Format("{0:d}", fechafin); lblperiodofin.Text = fechaformatofin; lblcodigoi.Text = read["PRDCODIGOI"].ToString(); } conection.Close(); }</pre>			

TABLA 9: Función Cargar periodo académico

ReporteAsistencia.aspx.cs			
Lenguaje:	C#	Acción:	Carga
Método:	Cargarperiodo()	Entrada:	
Descripción:	Carga el periodo habilitado		
<pre>//La funcion sirve para cargar el periodo activo public void cargarperiodo() { conection.Open(); SqlCommand com = new SqlCommand("SELECT * FROM PERIODO WHERE PRDESTADOC = 1", conection); SqlDataReader read = com.ExecuteReader(); if (read.Read()) { DateTime fechaini = Convert.ToDateTime(read["PRDFECINIF"].ToString()); string fechaformatoini = String.Format("{0:d}", fechaini); lblperiodoini.Text = fechaformatoini; DateTime fechafin = Convert.ToDateTime(read["PRDFECFINF"].ToString()); string fechaformatofin = String.Format("{0:d}", fechafin); lblperiodofin.Text = fechaformatofin; lblcodigoi.Text = read["PRDCODIGOI"].ToString(); } conection.Close(); } }</pre>			

TABLA 10: Función Cargar periodo académico

ReportePlanificacion.aspx.cs			
Lenguaje:	C#	Acción:	Carga
Método:	Cargarperiodo()	Entrada:	
Descripción:	Carga el periodo habilitado		
<pre>//La funcion sirve para cargar el periodo activo public void cargarperiodo() { conection.Open(); SqlCommand com = new SqlCommand("SELECT * FROM PERIODO WHERE PRDESTADOC = 1", conection); SqlDataReader read = com.ExecuteReader(); if (read.Read()) { DateTime fechaini = Convert.ToDateTime(read["PRDFECINIF"].ToString()); string fechaformatoini = String.Format("{0:d}", fechaini); lblperiodoini.Text = fechaformatoini; DateTime fechafin = Convert.ToDateTime(read["PRDFECFINF"].ToString()); string fechaformatofin = String.Format("{0:d}", fechafin); lblperiodofin.Text = fechaformatofin; lblcodigoi.Text = read["PRDCODIGOI"].ToString(); } conection.Close(); } }</pre>			

TABLA 11: Función Cargar periodo académico

SemanasClases.aspx.cs			
Lenguaje:	C#	Acción:	Carga
Método:	Cargarperiodo()	Entrada:	
Descripción:	Carga el periodo habilitado		
<pre>//La funcion sirve para cargar el periodo activo public void cargarperiodo() { conection.Open(); SqlCommand com = new SqlCommand("SELECT * FROM PERIODO WHERE PRDESTADOC = 1", conection); SqlDataReader read = com.ExecuteReader(); if (read.Read()) { DateTime fechaini = Convert.ToDateTime(read["PRDFECINIF"].ToString()); string fechaformatoini = String.Format("{0:d}", fechaini); lblperiodoini.Text = fechaformatoini; DateTime fechafin = Convert.ToDateTime(read["PRDFECFINF"].ToString()); string fechaformatofin = String.Format("{0:d}", fechafin); lblperiodofin.Text = fechaformatofin; lblcodigoi.Text = read["PRDCODIGOI"].ToString(); } conection.Close(); } }</pre>			

TABLA 12: Función Cargar periodo académico

Syllabus.aspx.cs			
Lenguaje:	C#	Acción:	Carga
Método:	Cargarperiodo()	Entrada:	
Descripción:	Carga el periodo habilitado		
<pre>//La funcion sirve para cargar el periodo activo public void cargarperiodo() { conection.Open(); SqlCommand com = new SqlCommand("SELECT * FROM PERIODO WHERE PRDESTADOC = 1", conection); SqlDataReader read = com.ExecuteReader(); if (read.Read()) { DateTime fechaini = Convert.ToDateTime(read["PRDFECINIF"].ToString()); string fechaformatoini = String.Format("{0:d}", fechaini); lblperiodoini.Text = fechaformatoini; DateTime fechafin = Convert.ToDateTime(read["PRDFECFINF"].ToString()); string fechaformatofin = String.Format("{0:d}", fechafin); lblperiodofin.Text = fechaformatofin; lblcodigoi.Text = read["PRDCODIGOI"].ToString(); } conection.Close(); } }</pre>			

TABLA 13: Función Cargar periodo académico

Control.aspx.cs			
Lenguaje:	C#	Acción:	Carga
Método:	Casignatura()	Entrada:	
Descripción:	Obtiene el nombre y código de la asignatura		
<pre>//La funcion sirve para obtener el codigo de la asignatura public void casignatura() { conection.Open(); SqlCommand com1 = new SqlCommand("SELECT ASIGNATURA.ASGCODIGO AS CODIGO, ASIGNATURA.ASGNOMBREC AS NOMBRE FROM DISTRIBUCION INNER JOIN ASIGNATURA ON DISTRIBUCION.ASGCODIGO = ASIGNATURA.ASGCODIGO INNER JOIN PROFESOR ON DISTRIBUCION.PRFCODIGO = PROFESOR.PRFCODIGO WHERE PROFESOR.DTPCEDULAC = '" + lblcedula.Text + "' AND DISTRIBUCION.PRDCODIGO = '" + lblcodigoi.Text + "'",conection); SqlDataReader read1 = com1.ExecuteReader(); if (read1.Read()) { lblcodasignatura.Text = read1["CODIGO"].ToString(); lblasignatura.Text = read1["NOMBRE"].ToString(); } conection.Close(); } }</pre>			

TABLA 14: Función Obtener código asignatura

Control.aspx.cs			
Lenguaje:	C#	Acción:	Incrementa
Método:	aumentaid()	Entrada:	
Descripción:	Incrementa el código de la tabla ASISTENCIA_ALUMNOS		
<pre>//La funcion sirve para obtner el ultimo id de la tabla ASISTENCIA_ALUMNOS y agregar de acuerdo a esto uno nuevo public void aumentaid() { SqlCommand cmd = new SqlCommand("SELECT MAX(ASACODIGO) AS CODIGO FROM ASISTENCIA_ALUMNOS",conection); SqlDataReader read = cmd.ExecuteReader(); if (read.Read()) { asiscodigoi = Convert.ToInt32(read["CODIGO"].ToString()); if (asiscodigoi == 0) { asiscodigoi = 1; } else { asiscodigoi = asiscodigoi + 1; } } } }</pre>			

TABLA 15: Función Obtener Código ASISTENCIA_ALUMNOS

Syllabus.aspx.cs			
Lenguaje:	C#	Acción:	Incrementa
Método:	aumentaid()	Entrada:	
Descripción:	Incrementa el código de la tabla ASISTENCIA_ALUMNOS		
<pre>//La funcion sirve para obtener el ultimo id de la tabla ASISTENCIA_ALUMNOS y agregar de acuerdo a esto uno nuevo public void aumentaid() { SqlCommand cmd = new SqlCommand("SELECT MAX(ASACODIGO) AS CODIGO FROM ASISTENCIA_ALUMNOS",conection); SqlDataReader read = cmd.ExecuteReader(); if (read.Read()) { asiscodigoi = Convert.ToInt32(read["CODIGO"].ToString()); if (asiscodigoi == 0) { asiscodigoi = 1; } else { asiscodigoi = asiscodigoi + 1; } } }</pre>			

TABLA 16: Función Obtener código ASISTENCIA_ALUMNOS

Control.aspx.cs			
Lenguaje:	C#	Acción:	Incrementa
Método:	aumentaidsilabo()	Entrada:	
Descripción:	Incrementa el código de la tabla SEGUIMIENTO_SILABO		
<pre>//La funcion sirve para obtener el ultimo id de la tabla SEGUIMIENTO_SILABO y agregar de acuerdo a esto uno nuevo public void aumentaidsilabo() { SqlCommand cmd = new SqlCommand("SELECT MAX(SGSCODIGO) AS CODIGO FROM SEGUIMIENTO_SILABO", conection); SqlDataReader read = cmd.ExecuteReader(); if (read.Read()) { silcodigoi = Convert.ToInt32(read["CODIGO"].ToString()); if (silcodigoi == 0) { silcodigoi = 1; } else { silcodigoi = silcodigoi + 1; } } }</pre>			

TABLA 17: Función Obtener código SEGUIMIENTO_SILABO

Login.aspx.cs			
Lenguaje:	C#	Acción:	Encripta
Método:	encripta(string password)	Entrada:	String password
Descripción:	Encripta el usuario y contraseña		
<pre>//Funcion que se usa para encriptar tanto el usuario como la contraseña private string Encripta(string password) { System.Security.Cryptography.HashAlgorithm hashValue = new System.Security.Cryptography.SHA1CryptoServiceProvider(); string pass; byte[] passwordBytes = ASCIIEncoding.ASCII.GetBytes(password); byte[] passwordHash = hashValue.ComputeHash(passwordBytes); pass = Convert.ToBase64String(passwordHash, 0, passwordHash.Length); return (pass); }</pre>			

TABLA 18: Función Encriptar Pasword Usuario

Login.aspx.cs			
Lenguaje:	C#	Acción:	Desencripta
Método:	Desencripta(string password)	Entrada:	String password
Descripción:	Desencripta el usuario y contraseña		
<pre>//Funcion que se usa para desencriptar tanto el usuario como la contraseña private string Desencripta(string password) { System.Security.Cryptography.HashAlgorithm hashValue = new System.Security.Cryptography.SHA1CryptoServiceProvider(); byte[] bytes = System.Text.Encoding.UTF8.GetBytes(password); byte[] byteHash = hashValue.ComputeHash(bytes); hashValue.Clear(); return (Convert.ToBase64String(byteHash)); }</pre>			

TABLA 19: Función Desencriptar Password y Usuario

Parciales.aspx.cs			
Lenguaje:	C#	Acción:	Incrementa
Método:	Aumentacodigoi()	Entrada:	
Descripción:	Incrementa el código de la tabla PARCIALES		
<pre>//Funcion para obtener el ultimo id de la tabla PARCIALES y de este modo agregar //un id nuevo public void aumentacodigoi() {conection.Open(); SqlCommand cmd1 = new SqlCommand("SELECT MAX(PRCCODIGO) AS CODIGO FROM PARCIALES", conection); SqlDataReader read = cmd1.ExecuteReader(); if (read.Read()) { codigoi = Convert.ToInt32(read["CODIGO"].ToString()); if (codigoi == 0) { codigoi = codigoi + 1; lblcodigoi.Text = Convert.ToString(codigoi); } else { codigoi = codigoi + 1; lblcodigoi.Text = Convert.ToString(codigoi); } } conection.Close(); }</pre>			

TABLA 20: Función Obtener código de PARCIALES

Recursos.aspx.cs			
Lenguaje:	C#	Acción:	Incrementa
Método:	Aumentacodigoi()	Entrada:	
Descripción:	Incrementa el código de la tabla RECURSOS		
<pre>//La funcion obtiene el ultimo id de la tabla RECURSOS y lo aumenta para agregar uno nuevo public void aumentacodigoi() {conection.Open(); SqlCommand cmd1 = new SqlCommand("SELECT MAX(RCRCODIGO) AS CODIGO FROM RECURSOS", conection); SqlDataReader read = cmd1.ExecuteReader(); if (read.Read()) { codigoi = Convert.ToInt32(read["CODIGO"].ToString()); if (codigoi == 0) { codigoi = codigoi + 1; lblcodigoi.Text = Convert.ToString(codigoi); } else { codigoi = codigoi + 1; lblcodigoi.Text = Convert.ToString(codigoi); } } conection.Close(); }</pre>			

TABLA 21: Función Obtener código RECURSOS

ReporteAsistencia.aspx.cs			
Lenguaje:	C#	Acción:	Obtiene
Método:	asignaturanombre()	Entrada:	
Descripción:	Obtiene el nombre de la asignatura		
<pre>//Carga el nombre de la asignatura seleccionada anteriormente public void asignaturanom() { conection.Open(); SqlCommand com1 = new SqlCommand("SELECT ASIGNATURA.ASGNOMBREC AS NOMBRE FROM DISTRIBUCION INNER JOIN ASIGNATURA ON DISTRIBUCION.ASGCODIGOI = ASIGNATURA.ASGCODIGOI INNER JOIN PROFESOR ON DISTRIBUCION.PRFCODIGO = PROFESOR.PRFCODIGO WHERE PROFESOR.DTPCEDULAC = '" + lblcedula.Text + "' AND DISTRIBUCION.PRDCODIGOI = '" + lblcodigoi.Text + "' AND ASIGNATURA.ASGCODIGOI = " + lblcodasignatura.Text + "'", conection); SqlDataReader read1 = com1.ExecuteReader(); if (read1.Read()) { lblasignatura.Text = read1["NOMBRE"].ToString(); } conection.Close(); } }</pre>			

TABLA 22: Función Cargar nombre asignatura

ReportePlanificacion.aspx.cs			
Lenguaje:	C#	Acción:	Obtiene
Método:	asignaturanombre()	Entrada:	
Descripción:	Obtiene el nombre de la asignatura		
<pre>//Carga el nombre de la asignatura seleccionada anteriormente public void asignaturanom() { conection.Open(); SqlCommand com1 = new SqlCommand("SELECT ASIGNATURA.ASGNOMBREC AS NOMBRE FROM DISTRIBUCION INNER JOIN ASIGNATURA ON DISTRIBUCION.ASGCODIGOI = ASIGNATURA.ASGCODIGOI INNER JOIN PROFESOR ON DISTRIBUCION.PRFCODIGO = PROFESOR.PRFCODIGO WHERE PROFESOR.DTPCEDULAC = '" + lblcedula.Text + "' AND DISTRIBUCION.PRDCODIGOI = '" + lblcodigoi.Text + "' AND ASIGNATURA.ASGCODIGOI = " + lblcodasignatura.Text + "'", conection); SqlDataReader read1 = com1.ExecuteReader(); if (read1.Read()) { lblasignatura.Text = read1["NOMBRE"].ToString(); } conection.Close(); } }</pre>			

TABLA 23: Función Cargar nombre asignatura

Syllabus.aspx.cs			
Lenguaje:	C#	Acción:	Obtiene
Método:	asignaturanombre()	Entrada:	
Descripción:	Obtiene el nombre de la asignatura		
<pre>//Carga el nombre de la asignatura seleccionada anteriormente public void asignaturanom() { conection.Open(); SqlCommand com1 = new SqlCommand("SELECT ASIGNATURA.ASGNOMBREC AS NOMBRE FROM DISTRIBUCION INNER JOIN ASIGNATURA ON DISTRIBUCION.ASGCODIGOI = ASIGNATURA.ASGCODIGOI INNER JOIN PROFESOR ON DISTRIBUCION.PRFCODIGO = PROFESOR.PRFCODIGO WHERE PROFESOR.DTPCEDULAC = '" + lblcedula.Text + "' AND DISTRIBUCION.PRDCODIGOI = '" + lblcodigoi.Text + "' AND ASIGNATURA.ASGCODIGOI = '" + lblcodasignatura.Text + "'", conection); SqlDataReader read1 = com1.ExecuteReader(); if (read1.Read()) { lblasignatura.Text = read1["NOMBRE"].ToString(); } conection.Close(); } }</pre>			

TABLA 24: Función Cargar nombre asignatura

ReporteAsistencia.aspx.cs			
Lenguaje:	C#	Acción:	Carga
Método:	cargaReporte()	Entrada:	
Descripción:	Carga el reporte		
<pre>//Arma el reporte public void cargaReporte() { conection.Open(); DataTable dt = new DataTable(); DataRow dr = dt.NewRow(); SqlCommand cmd = new SqlCommand("SELECT COUNT(PLANIFICACION_CURRICULO.PLCNUMENCUENTRO) as columna FROM dbo.PLANIFICACION_CURRICULO INNER JOIN dbo.DISTRIBUCION ON dbo.PLANIFICACION_CURRICULO.DSTCODIGOI = dbo.DISTRIBUCION.DSTCODIGOI WHERE DISTRIBUCION.ASGCODIGOI = '" + lblcodasignatura.Text + "' AND PLANIFICACION_CURRICULO.DSTCODIGOI = '"+lblcoddst.Text+'"', conection); SqlDataReader read = cmd.ExecuteReader(); if (read.Read()) { numcol = Convert.ToInt32(read["columna"].ToString()); } read.Close(); dt.Columns.Add("ALUMNO"); for (int i1 = 1; i1 <= numcol; i1++) { dt.Columns.Add(i1.ToString()); } }</pre>			

```

        SqlCommand cmd1 = new SqlCommand("SELECT DISTINCT
ALUMNOMATERIAS.ALMCODIGO AS CODIGO, INSCRIPCION.DTPCEDULAC,
RTRIM(DATOSPERSONALES.DTPAPELLIC) + ' ' + RTRIM(DATOSPERSONALES.DTPAPELLIC2) + ' '
+ RTRIM(DATOSPERSONALES.DTPNOMBREC) AS NOMBRES, count(ALUMNOMATERIAS.ALMCODIGO)
FROM MATRICULA INNER JOIN ALUMNOMATERIAS ON MATRICULA.MTRNUMEROI =
ALUMNOMATERIAS.MTRNUMEROI INNER JOIN INSCRIPCION ON MATRICULA.INSCODIGO =
INSCRIPCION.INSCODIGO INNER JOIN DATOSPERSONALES ON INSCRIPCION.DTPCEDULAC =
DATOSPERSONALES.DTPCEDULAC INNER JOIN CARRERA_MODAL ON MATRICULA.CRRMODCODIGO =
CARRERA_MODAL.CRRMODCODIGO LEFT OUTER JOIN ASISTENCIA_ALUMNOS ON
ALUMNOMATERIAS.ALMCODIGO = ASISTENCIA_ALUMNOS.ALMCODIGO INNER JOIN
dbo.PLANIFICACION_CURRICULO ON dbo.ASISTENCIA_ALUMNOS.PLCCODIGO =
dbo.PLANIFICACION_CURRICULO.PLCCODIGO WHERE (ALUMNOMATERIAS.ASGCODIGO = " +
lblcodasignatura.Text + ") AND (MATRICULA.PRDCODIGO = " + lblcodigo.Text + ")
AND (ALUMNOMATERIAS.ALMESTADOMAT = 'N') AND CARRERA_MODAL.MDLCODIGO=" +
lblcodmod.Text + " AND MATRICULA.SCCCODIGO = " + lblcodsec.Text + " AND
ALUMNOMATERIAS.ALMGRUPO = '" + lblcodparal.Text + "' GROUP BY
RTRIM(DATOSPERSONALES.DTPAPELLIC) + ' ' + RTRIM(DATOSPERSONALES.DTPAPELLIC2) + ' '
+ RTRIM(DATOSPERSONALES.DTPNOMBREC),
ALUMNOMATERIAS.ALMCODIGO,INSCRIPCION.DTPCEDULAC having
COUNT(ALUMNOMATERIAS.ALMCODIGO)>1 ORDER BY NOMBRES ", conection);
        SqlDataReader read1 = cmd1.ExecuteReader();
        while (read1.Read())
        {
            dr = dt.NewRow();
            codalm = Convert.ToInt32(read1["CODIGO"].ToString());
            dr["ALUMNO"] = read1["NOMBRES"].ToString();

            SqlCommand cmd2 = new SqlCommand("SELECT
PLANIFICACION_CURRICULO.PLCCODIGO AS PLCCODIGO FROM dbo.PLANIFICACION_CURRICULO
INNER JOIN dbo.DISTRIBUCION ON dbo.PLANIFICACION_CURRICULO.DSTCODIGO =
dbo.DISTRIBUCION.DSTCODIGO WHERE DISTRIBUCION.ASGCODIGO = " +
lblcodasignatura.Text + " AND PLANIFICACION_CURRICULO.DSTCODIGO =
"+lblcoddst.Text+"", conection);
            SqlDataReader read2 = cmd2.ExecuteReader();

            while (read2.Read())
            {
                plccodigo =
Convert.ToInt32(read2["PLCCODIGO"].ToString());

                SqlCommand cmdasis = new SqlCommand("SELECT
ISNULL(ASISTENCIA_ALUMNOS.ASAASITENCIA, 0) AS ASISTENCIA FROM MATRICULA INNER JOIN
ALUMNOMATERIAS ON MATRICULA.MTRNUMEROI = ALUMNOMATERIAS.MTRNUMEROI INNER JOIN
INSCRIPCION ON MATRICULA.INSCODIGO = INSCRIPCION.INSCODIGO INNER JOIN
DATOSPERSONALES ON INSCRIPCION.DTPCEDULAC = DATOSPERSONALES.DTPCEDULAC INNER JOIN
CARRERA_MODAL ON MATRICULA.CRRMODCODIGO = CARRERA_MODAL.CRRMODCODIGO LEFT OUTER
JOIN ASISTENCIA_ALUMNOS ON ALUMNOMATERIAS.ALMCODIGO =
ASISTENCIA_ALUMNOS.ALMCODIGO INNER JOIN dbo.PLANIFICACION_CURRICULO ON
dbo.ASISTENCIA_ALUMNOS.PLCCODIGO = dbo.PLANIFICACION_CURRICULO.PLCCODIGO WHERE
(ALUMNOMATERIAS.ASGCODIGO = " + lblcodasignatura.Text + ") AND
(MATRICULA.PRDCODIGO = " + lblcodigo.Text + ") AND (ALUMNOMATERIAS.ALMESTADOMAT
= 'N') AND CARRERA_MODAL.MDLCODIGO=" + lblcodmod.Text + " AND
MATRICULA.SCCCODIGO = " + lblcodsec.Text + " AND ALUMNOMATERIAS.ALMGRUPO = '" +
lblcodparal.Text + "' AND ASISTENCIA_ALUMNOS.PLCCODIGO= " + plccodigo + " AND
ALUMNOMATERIAS.ALMCODIGO = " + codalm + " ", conection);
                SqlDataReader readasis = cmdasis.ExecuteReader();
                while (readasis.Read())

```

```

        {
            for (int i = 1; i <= numcol; i++)
            {
                string asistencia =
(readasis["ASISTENCIA"].ToString());
                if (asistencia == "True")
                {
                    dr[i.ToString()] = "A";
                }
                else
                {
                    dr[i.ToString()] = "I";
                }
            }
        }
        dt.Rows.Add(dr);
        read2.Close();
    }
    read1.Close();
    gvReporte.DataSource = dt;
    gvReporte.DataBind();
}

```

TABLA 25: Función Armar Reporte

SemanasClases.aspx.cs			
Lenguaje:	C#	Acción:	Incrementa
Método:	aumentacodigoi()	Entrada:	
Descripción:	Incrementa el código de la tabla SEMANAS_CLASES		
<pre> //Obtiene el último id de la tabla SEMANAS_CLASES y aumenta 1 public void aumentacodigoi() { connection.Open(); SqlCommand cmd1 = new SqlCommand("SELECT MAX(SMCCODIGOI) AS CODIGO FROM SEMANAS_CLASES", connection); SqlDataReader read = cmd1.ExecuteReader(); if (read.Read()) { codigoi = Convert.ToInt32(read["CODIGO"].ToString()); if (codigoi == 0) { codigoi = 1; lblcodigoi.Text = Convert.ToString(codigoi); } else { codigoi = codigoi + 1; lblcodigoi.Text = Convert.ToString(codigoi); } } connection.Close(); } </pre>			

TABLA 26: Función Obtener código SEMANAS_CLASES

Syllabus.aspx.cs			
Lenguaje:	C#	Acción:	Incrementa
Método:	aumentacodigoi()	Entrada:	
Descripción:	Aumenta el id de la tabla PLANIFICACION_CURRICULO		
<pre>//Aumenta el id de la tabla PLANIFICACION_CURRICULO public void aumentacodigoi() { conection.Open(); SqlCommand cmd1 = new SqlCommand("SELECT MAX(PLCCODIGO) AS CODIGO FROM PLANIFICACION_CURRICULO", conection); SqlDataReader read = cmd1.ExecuteReader(); if (read.Read()) { codigoi = Convert.ToInt32(read["CODIGO"].ToString()); if (codigoi == 0) { codigoi = 1; } else { codigoi = codigoi + 1; } } read.Close(); conection.Close(); }</pre>			

TABLA 27: Función Obtener código PLANIFICACION_CURRICULO