



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**Sistema de automatización de
horarios en una institución de
educación abierta y a distancia**

TESIS

Que para obtener el título de

Ingeniero en Computación

P R E S E N T A N

López Vences Judith Elizabeth

Medina Pérez Miguel Ángel

Méndez Granados Ruth Cristina

Pérez Castillo Lorena Adriana

Ingeniero Eléctrico Electrónico

P R E S E N T A

Hernández González Fabián

DIRECTOR DE TESIS

M.I. Roa Beiza Juan Carlos

Ciudad Universitaria, Cd. Mx., 2016





DEDICATORIAS Y AGRADECIMIENTOS



Agradecimientos

A mis padres Silvia González Medina y Ambrosio Hernández Medina, por su cariño, cuidado, comprensión y confianza que me han brindado todos estos años. Gracias por su apoyo en todo momento.

A mis hermanos Lourdes y Marco Antonio, por brindarme su apoyo en diferentes etapas de mi vida.

A mis sobrinos Ximena Abigail, Esteban y María José, por hacer mis días más alegres y yo estaré con ustedes para apoyarlos en todo momento.

A Haide Aguilar, por esta etapa que iniciamos juntos y brindarme tu apoyo, cariño y confianza.

A todos mis amigos, por todas las experiencias buenas y malas que hemos vivido juntos.

A la UNAM y sobre todo a la Facultad de Ingeniería, por brindar todas las herramientas necesarias para desarrollarme profesionalmente.

A mis compañeros y amigos de tesis, por su dedicación y esfuerzo en este trabajo.

Al M.I. Juan Carlos Roa Beiza, por su apoyo, paciencia y consejos.

Al PAT, por el apoyo brindado para la elaboración de este trabajo de tesis.

Hernández González Fabián.



A mi madre Celia Vences Méndez que es la persona más importante en mi vida, que siempre ha estado a mi lado guiándome por el mejor camino; me enseñaste a ser una mujer fuerte, de carácter firme y a no rendirme, gracias a tu ejemplo del día a día, porque no te importaba estar cansada de hacer las tareas diarias de la casa, siempre encontrabas el tiempo y las fuerzas para estar al pendiente de que hiciera mis labores escolares, explicándome temas que no entendía o sencillamente haciéndome compañía. Has sido mi mayor apoyo en aquellos momentos en los que pensé que no podría levantarme. Simplemente, yo no sería nada si no te tuviera, por lo que soy es gracias a ti. Ahora, después de tanto tiempo puedo retribuirte un poco de todo el esfuerzo que has hecho por mí, cumpliéndote uno de tus mayores sueños, que era el verme titulada como ingeniera.

A mis hermanas Heidi y Wendi, que tuvieron que soportar las locuras de su hermana menor, me han apoyado incondicionalmente en todo lo que he decidido hacer, bríndame sus consejos y ayuda. Nunca se los he dicho pero me siento orgullosa de que sean mis hermanas. También quiero agradecerles, porque me han dado cuatro maravillosas razones por la que debo seguir adelante y no rendirme; que son Loghan, Zaid, Jonathan y Sebastián.

A mi tía Gloria Vences Méndez que ha sido un gran apoyo de mi madre y que me cuidó desde que era pequeña; a la familia Vences Hernández y Vences Ramírez que siempre han estado al pendiente de que estemos bien y que me han apoyado en este largo transcurso.

A mis amigos y compañeros de tesis Lorena Pérez, Ruth Méndez, Miguel Medina y Fabián Hernández, que sin su conocimiento, esfuerzo y dedicación no se habría logrado realizar este trabajo.

A mis amigos SacNicté Guzmán, Aranelly Hernández, Alejandra Flores, Juan Aguilar, Jorge Esquivel, Pablo Sánchez, Raúl Juárez y Eduardo Martínez, a quienes considero como parte de mi familia, que han sido un pilar en los momentos más significativos de



mi vida desde el instante en que los conocí, me han ofrecido un abrazo cuando más lo necesitaba, así también me han aplaudido mis logros. Les agradezco de corazón el tiempo, la confianza y el cariño que me han ofrecido; y porque no dudaron que este día tan importante para mí llegaría.

Al M.I. Juan Carlos Roa por compartirnos su conocimiento durante este tiempo y por ser una parte fundamental para que este proyecto concluyera exitosamente, inyectándonos siempre de su optimismo y confianza de que esto sería posible.

Al PAT por ofrecernos la oportunidad de desarrollar el trabajo de tesis de una manera óptima y concisa.

A la Universidad Nacional Autónoma de México y a la Facultad de Ingeniería por darnos las bases necesarias que necesitamos para desarrollarnos exitosamente en el ámbito profesional.

López Vences Judith Elizabeth



A mi mamá Silvia por estar siempre a mi lado, desde que tengo recuerdos has hecho muchos sacrificios para mostrarme el camino correcto y aunque he tenido tropiezos siempre has estado para mí brindándome tu amor, tu apoyo y tu comprensión, eres una mujer que me llena de mucho orgullo me has dado tanto que no hay nada con lo que te pudiera devolver, pero hoy al ponerle fin a este ciclo solo me queda decir:

Te amo mamá, lo logramos.

A mi abuelito Raúl y a mi Nany's porque cuando mi mamá no estaba presente estuvieron ahí para ser esos padres que me dieron el cariño, amor, tiempo y dedicación y porque con sus valores puedo decir con orgullo que soy el hombre que hoy esta aquí.

A mi hermano Raúl aunque hubo peleas con los años hemos aprendido a llevarnos mejor a confiar en nosotros y saber que siempre podremos contar el uno con el otro pero sobre todo por enseñarme que aun en los momentos difíciles el carácter y el apoyo de la familia nos harán salir adelante.

A mi hermano Isaí por ser un niño especial al que adoro y que me enseñó lo bueno de tener un hermano menor y en el que veo un don muy grande la nobleza.

A Mi tía Adriana ya que ella ha sido un pilar muy importante en mi vida sus consejos y su apoyo me han ayudado a estar aquí y aunque no siempre concordemos siempre podre confiar en ella.

A mi tía María del Carmen porque siempre me brindó su apoyo y cariño y que con su manera muy particular de ser hizo que me esforzara para llegar hasta aquí y sé que por siempre tendrén ella un apoyo incondicional.

A Marisol Sandoval mi primuchis preciosa por siempre escucharme, siempre tener una palabra bonita para mí, por ser mi orgullo y porque cualquiera que sea la situación siempre estaremos ahí para apoyarnos.



A mi tío Arturo por ser el primer ejemplo a quien seguir, mostrarme la responsabilidad, el cariño a la familia pero sobre todo agradecer por el apoyo que me ha brindado en toda mi vida.

A mi tío Álvaro porque siempre tiene un gesto de apoyo para mí, siempre recuerdo momentos de felicidad compartidos y por ser un gran ejemplo de responsabilidad

A Edgar porque siempre ha sido un apoyo no solo para mi sino para mi mamá y mis hermanos especialmente, gracias por estar en todos los momentos en que fuiste necesario.

A mi cuñada Yazmín por compartir muchos momentos y por el amor demostrado a lo largo de los años, a mi sobrino Alex, por venir a dar alegría a la familia y siempre demostrarme su cariño.

A mi primo Christian por ser un amigo, por las convivencias, por los momentos compartidos y porque sé que estará ahí cuando necesite su apoyo.

A Mariana por ser mi prima consentidora y me ayuda siempre que se lo pido.

A Tía Norma por alentarme a continuar y por el apoyo que he recibido durante estos años.

Irving, Denisse, Luis Ángel porque con ellos he convivido toda la vida y son parte importante de mi vida.

A Rafael, Miguel Ángel, Arturo, Luis Edgar, Persy, Memo quienes a lo largo de los años me han demostrado que están ahí en los momentos buenos, malos y los peores, esa sensación de unión que más de amistad es de hermandad. Pero sobre todo agradezco que estén recorriendo este camino a mi lado llamado vida.



A Lupita, Gerardo y Ubaldo porque me enseñaron el trabajo en equipo, me han apoyado con su conocimiento, su compañerismo pero sobre todo con su amistad.

A la UNAM y la Facultad de Ingeniería por darme las herramientas profesionales y personales para desarrollarnos con orgullo no solo en un ámbito profesional, sino también en un ámbito personal haciéndonos excelentes profesionistas pero mejores personas.

A Judith, Ruth, Lorena y Fabián cómplices en este proyecto y que sin el valioso aporte de cada uno de ellos, este proyecto no se vería cristalizado.

Al PAT por brindarnos la oportunidad de cerrar el ciclo universitario.

Al M.I Juan Carlos Roa, porque sin sus conocimientos, aportes y anécdotas no hubiera sido posible dar el paso final a la titulación.

Por último quiero agradecer a todas aquellas personas que han formado parte de mi vida amigos, compañeros de escuela, compañeros de trabajo y que sería imposible de mencionar a cada una de ellas, pero que me han brindado su amistad, sus consejos, su apoyo, porque con sus acciones han dejado una marca en mi persona o simplemente porque en algún momento les ha tocado ser parte de mi vida.

Medina Pérez Miguel Ángel.



Quiero dedicar este logro a mi mami LOURDES, gracias por siempre estar al pendiente de mí, gracias por el sacrificio y esfuerzo que hiciste, brindándome la oportunidad de estudiar, gracias por esos consejos invaluablees y enseñarme a siempre seguir adelante a pesar de las circunstancias. Eres mi mayor ejemplo de fortaleza e inteligencia, eres mi pilar, te amo mamá.

A mi hermanita ANDY, por ser el motivo que me impulsa a seguir adelante dando lo mejor de mí y continuar esforzándome cada día. Gracias por tu alegría y cariño, te quiero tanto hoppy.

A ULI por apoyarme incondicionalmente, por enseñarme cada día a ver la vida de diferente forma, gracias por esos consejos certeros que me hacen reflexionar, gracias por tu cariño y tolerancia. Gracias por siempre estar, te quiero chun.

Al PAT por la oportunidad y el apoyo brindado.

A nuestro director de tesis M.I JUAN CARLOS ROA BEIZA, por dedicarnos tiempo y siempre estar al tanto de nuestros avances, aportando ideas de valor para la realización de esta tesis.

A mis compañeros de tesis, por el apoyo y dedicación para lograr la realización de este trabajo.

A la UNAM por todas las enseñanzas y vivencias, a sus profesores que siempre nos impulsaron a esforzarnos y dar lo mejor de nosotros.

Méndez Granados Ruth Cristina.



AGRADECIMIENTOS

A toda mi familia, por el cariño incondicional. A mi tío Ricardo, por ser un hombre con sabiduría y ser para mí un ejemplo a seguir. A mi tía Coni, por estar siempre ahí, porque a pesar de mis momentos de apatía e inmadurez me apoya y alienta a seguir adelante. A mi tía Basilia, por su sentido del humor y por las llamadas de atención. A mi primo Ricardo, por patrocinar mis proyectos y apoyarme incondicionalmente. A mi prima Panchis, por cuidarme y hacerme reír.

A todos mis amigos, por tanto cariño y apoyo, por llenar mi vida de risas y por creer en mí. A Ari, por ser una luz en mi vida. Le agradezco el siempre tener fe en mí. Porque a pesar de la distancia y el tiempo, ha estado cuando más la necesito. A Lisved, por ser mi hermana por elección, por apoyarme incondicionalmente, por motivarme a concluir esta etapa, por tanta paciencia y cariño y, por hacerme sentir un miembro más de su familia.

A los psiquiatras Alberto, Pamela y Hugo; quienes me enseñaron a dejar el pasado atrás, a reírme de mí misma y a aceptar mi “rareza”.

A la psicóloga Melissa Salas, porque me ha ayudado a trascender la etapa más triste y difícil de mi vida. Le agradezco el hacerme saber que la decisión de estar o no estar aquí no depende de mí; por insistir tanto para que culminara mi proceso de titulación y; sobre todo, por guiarme en el proceso de descubrir y aprender a utilizar las herramientas que necesito para seguir en el camino.

A todos los profesores que me han acompañado a lo largo de mi vida académica, por compartir su conocimiento y experiencia. Me considero afortunada, pues gran parte de los profesores que he conocido, han sido personas con un profundo amor a la docencia, con fe en la juventud y con pasión a su área. Al Dr. José Abel Herrera Camacho, por su apoyo durante mi paso por la Facultad de Ingeniería. A la ingeniera Celia Martínez, por permitirme aprender tanto del área Biomédica, por su apoyo y por compartir su experiencia de vida conmigo.



A mis compañeros y cómplices en este proyecto: Judith Elizabeth López Vences, Ruth Cristina Méndez Granados, Fabián Hernández González y Miguel Ángel Medina Pérez. Les agradezco tanto el entusiasmo, la dedicación, el apoyo y la paciencia. Gracias por haber sido un gran equipo y porque supimos superar los problemas con buen humor.

A mi asesor de tesis, M.I. Juan Carlos Roa Beiza, por el optimismo y la confianza que infundió en nosotros para concluir este proyecto. Por apoyarnos en todo momento y siempre sonreír.

Al PAT, por ser un excelente programa que realmente apoya y motiva a los egresados a concluir esta etapa profesional.

A la Facultad de Ingeniería y a la Universidad Nacional Autónoma de México, por la excelente formación académica que me brindó. Porque pude crecer de manera integral: Académicamente, culturalmente, socialmente, etc. Porque aprendí que nos debemos a una sociedad, que el conocimiento de nada sirve si no se comparte y se utiliza para generar un impacto positivo en el entorno. ¡Gracias UNAM!

Gracias a Dios. Porque a pesar de todas las pruebas que he tenido que superar, he sentido su amor en las personas que ha puesto en mi camino.

AGRADECIMIENTO ESPECIAL

Dr. Carlos de Jesús KramskySteinpreis:

Le agradezco profundamente todo lo que ha hecho por mí. Usted ha sido la segunda persona que más ha influido para que esto sea posible, para que ahora yo tenga una carrera. Gracias por infundir en mí, con su ejemplo, el amor hacia el conocimiento, hacia la cultura y hacia Dios. Gracias por habernos permitido, a mi madre y a mí, ser parte de su vida. Gracias por seguir al pendiente de mí, por continuar apoyándome en todos los sentidos. Las palabras no me son suficientes para poder expresar toda mi gratitud hacia usted. Lo quiero mucho, que Dios lo bendiga siempre.



DEDICATORIA

Cleotilde Castillo López:

¡Lo logramos! He terminado oficialmente la carrera. Este logro, como todos los anteriores y los que vendrán, te lo dedico con todo mi amor. Gracias mamá por enseñarme a dar siempre lo mejor, a no ser conformista, a no tener miedo a triunfar.

Tú mejor que nadie, sabes que no fue fácil; sin embargo, siempre tuviste el temple necesario para motivarme a seguir adelante. Me enseñaste a superar los obstáculos que se presentaran, confiaste siempre en mi fortaleza.

Te agradezco con todo mi ser el haber sido padre, madre, hermana, amiga, todo; gracias por llenar, durante 23 años, mi vida de amor. Aún me duele mucho tu partida, desearía tanto haber compartido estos momentos contigo, el ver en tus ojos la felicidad y el orgullo porque “tu niña” se titula, al fin, como ingeniera.

Aún quedan metas por alcanzar, eres mi motor y mi ejemplo a seguir; lucharé por cumplir mis sueños y procuraré que, en donde quiera que ahora estés, te sigas sintiendo tan orgullosa de mí como yo lo estoy de ti.

Te amo y te extraño.

Pérez Castillo Lorena Adriana



ÍNDICE TEMÁTICO



CAPÍTULO I ANTECEDENTES.	1
1.1 Introducción.	2
1.2 Diagrama funcional de actores y responsabilidades.	5
1.3 Normatividad que rige a la institución en cuanto a espacios.	9
1.4 Descripción de los procedimientos para la asignación de salones, horarios, grupos, etc.	13
1.5 Alcance y acotamiento del problema.	18
CAPÍTULO II MARCO TEÓRICO.	23
2.1 Características, ventajas y desventajas de las bases de datos relacionales y el manejador de bases de datos MySQL.	24
2.2 Características, ventajas y desventajas de la arquitectura cliente-servidor. ..	29
2.3 Características, ventajas y desventajas del lenguaje PHP 5.6.19.	40
2.4 Características, ventajas y desventajas de Apache.	46
2.5 Características, ventajas y desventajas de Java Script y CSS.	50
CAPÍTULO III ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA.	57
3.1 Análisis del problema.	58
3.2 Recopilación y análisis de la información.	62
3.3 Requerimientos generales y particulares del sistema.	68
3.4 Planteamiento de la solución.	75
3.5 Diseño modular de la interfaz de usuario.	84
CAPÍTULO IV DISEÑO Y CONSTRUCCIÓN DEL SISTEMA.	93
4.1 Modelado del sistema.	94
4.1.1 Diagrama de contexto.	94
4.1.2 Diagrama entidad – relación.	95
4.1.3 Diagrama de procesos.	97
4.1.4 Diccionario de datos.	102
4.1.5 Normalización.....	110



4.2	Diseño e implementación de la base de datos.....	118
4.3	Construcción y diseño del front-end.....	130
4.4	Pruebas y mantenimiento.....	141
4.5	Evaluación del sitio empleando diferentes dispositivos y navegadores.....	151
CONCLUSIONES.....		157
BIBLIOGRAFÍA.....		161



CAPÍTULO I

“ANTECEDENTES”

1.1 Introducción.

Debido a la gran demanda que se tiene en la actualidad de alumnos que quieren seguir estudiando y que no se tiene el espacio suficiente en las instituciones educativas y/o que no cuentan con el tiempo para estar inscrito en un plan presencial, se desarrolla la modalidad de educación abierta y a distancia para que la población que se encuentra en esta situación se reduzca y tengan la oportunidad de seguir superándose. El sistema educativo retoma el plan de estudios que se tiene en la modalidad escolarizada, contando con nueve semestres en los que se distribuye 48 asignaturas.

En la figura 1.1.1 se ejemplifica cómo es que funciona el esquema de Educación Abierta y a Distancia.



Figura 1.1.1 - Esquema de Educación Abierta y a Distancia

La institución educativa se ve en la necesidad de realizar una mejor organización en la distribución de espacios designados en los que se presta apoyo educativo, para que se les imparta a los alumnos siete sesiones por materia durante el transcurso del semestre. En dichas sesiones es necesario programar: Encuadre, asesorías y exámenes ordinarios.



Cada profesor debe contar con la asignación de un salón que posea el equipo necesario en un determinado horario, para poder llevar a cabo cada una de las sesiones estipuladas por materia. De ahí la importancia de que se tenga una buena programación de horarios. Actualmente en otras instituciones se cuenta con programas que les facilita la elaboración automática de horarios garantizando la correcta administración de los espacios disponibles para las asesorías.

La escuela elaboraba esta planeación de forma manual, realizando tablas en Excel, ocasionando complicaciones como traslapes o que no se incluyera algún grupo. Para la realización de esta tarea se asignaba a un grupo de cuatro personas, las cuales elaboraban la distribución de horarios. La distribución obtenida se plasma en la página oficial de la institución en un formato PDF, para que sea de fácil acceso para los alumnos y profesores. Pero se corre el riesgo que en el último momento se realizarán modificaciones al archivo y los interesados no lo volvieran a corroborar por lo que se puede generar un descontrol.

Por lo que en el presente documento de tesis se realizará el desarrollo de un sistema en el que se pretende lograr un ahorro significativo de tiempo y esfuerzo para los integrantes de la coordinación administrativa que realizaba la elaboración de horarios, garantizando que no se generen errores logísticos como son traslapes de horario en salones, de igual manera, que todos los grupos asignados en esta modalidad tengan un espacio asignado que cuente con las instalaciones e inmuebles adecuados que le permita tanto al profesor como a los alumnos conseguir la mejor experiencia educativa.

En el primer capítulo se desarrollara el entorno del problema, es decir, se determinará los actores principales con los cuales se va a interactuar y de las responsabilidades que éstos tienen. Así también se especificará la normatividad bajo la cual se rige la institución respecto a la distribución de espacios asignados para las actividades de asesoría en la modalidad abierta y a distancia. La manera en que se realiza la distribución actual de los horarios es manual. Las tablas en las que alojaban la información de horarios, profesores



y salones; se tomarán en cuenta para realizar la delimitación del alcance que tendrá el sistema a desarrollar.

En el segundo capítulo se describe el marco teórico del sistema de automatización, tomando en cuenta las especificaciones de los equipos de cómputo que tiene la institución, como los servidores en los que se alojará la información correspondiente. De igual manera se trabajará con las características, ventajas y desventajas que tiene el software libre con el cual se desarrollará el sistema, se utilizarán PHP, MySQL, Java Script y servidores Apache; este tipo de software nos evita depender de licencias de terceros y nos permite obtener un resultado satisfactorio que cumpla con todos los requerimientos necesarios para que la asignación de horarios se facilite y se logren con los objetivos previstos de ahorro de tiempo, y esfuerzo; evitando así, el traslape en la asignación de los espacios disponibles. También se abordará lo concerniente a redes computacionales para explicar cómo es la arquitectura cliente-servidor con la cual se comunicarán los equipos, en los que se instalará la aplicación, con los servidores en los cuales se alojarán las bases de datos necesarias y se almacenará el archivo PDF para que éste se suba a la página oficial de la institución.

En el tercer capítulo se define el análisis y planteamiento del problema, para lo que se requiere analizar el problema específico y recopilar toda la información necesaria para desarrollar el sistema, sin dejar a un lado lo que la institución educativa utiliza. Por lo que la escuela nos proporcionará la información necesaria para realizar la distribución de salones disponibles, al igual que los profesores y grupos con los que se cuenta para que se realicen las asesorías, y la infraestructura, tanto general como particular, que se debe contar para el buen funcionamiento del sistema. Una vez recopilada y analizada toda la información se procederá a especificar un planteamiento de solución en el que se especifiquen los posibles módulos que contendrá el sistema, y el diseño modular de la interfaz con la que tendrá interacción el usuario final; apegándose siempre a los requerimientos y especificaciones indicadas por la institución y siguiendo una metodología que nos permita tener un desarrollo eficaz y rápido de la aplicación.



En el cuarto y último capítulo se describe el proceso de diseño y construcción del sistema de automatización de horarios. Una vez que ya se tiene especificado cuál es el planteamiento de solución, se desarrollará el modelado adecuado de entidad-relación, de procesos y para la base de datos solicitada por el usuario, conforme a la información obtenida para dicho fin. Serán procesados bajo las herramientas de software libre que se escogieron siguiendo una metodología y modelados específicos que nos ofrezcan una manipulación ágil, fácil y precisa en el desarrollo del sistema. Con esta información se implementará la base de datos en el servidor para que sea manejada desde los módulos especificados en el sistema por la institución. Se desarrollará el diseño y la implementación de la interfaz, con la que el usuario estará interactuando, tratando de que sea lo más amigable posible y sencilla de utilizar. Se mostrarán algunos ejemplos de pruebas que se realicen para la corroboración del adecuado funcionamiento de éste, también se evaluará el comportamiento del sistema en diferentes tipos de navegadores.

1.2 Diagrama funcional de actores y responsabilidades.

El organigrama representa gráficamente la estructura formal de la institución, expresándola de manera clara, entendible y objetiva. Podemos observar la vinculación entre áreas, líneas de autoridad, niveles jerárquicos y líneas de comunicación entre las diferentes áreas para la toma de decisiones. La figura 1.2.1 representa el organigrama general de la institución educativa.

La Dirección es la autoridad máxima encargada de regir todas y cada una de las áreas contenidas, dentro de esta organización se encuentra la Coordinación de Educación Abierta y a Distancia, que es la división en la cual nos enfocaremos para desarrollar el sistema de horarios, debido a que se encuentra bajo el mando de la dirección se debe cumplir la normatividad que rige a la institución educativa.

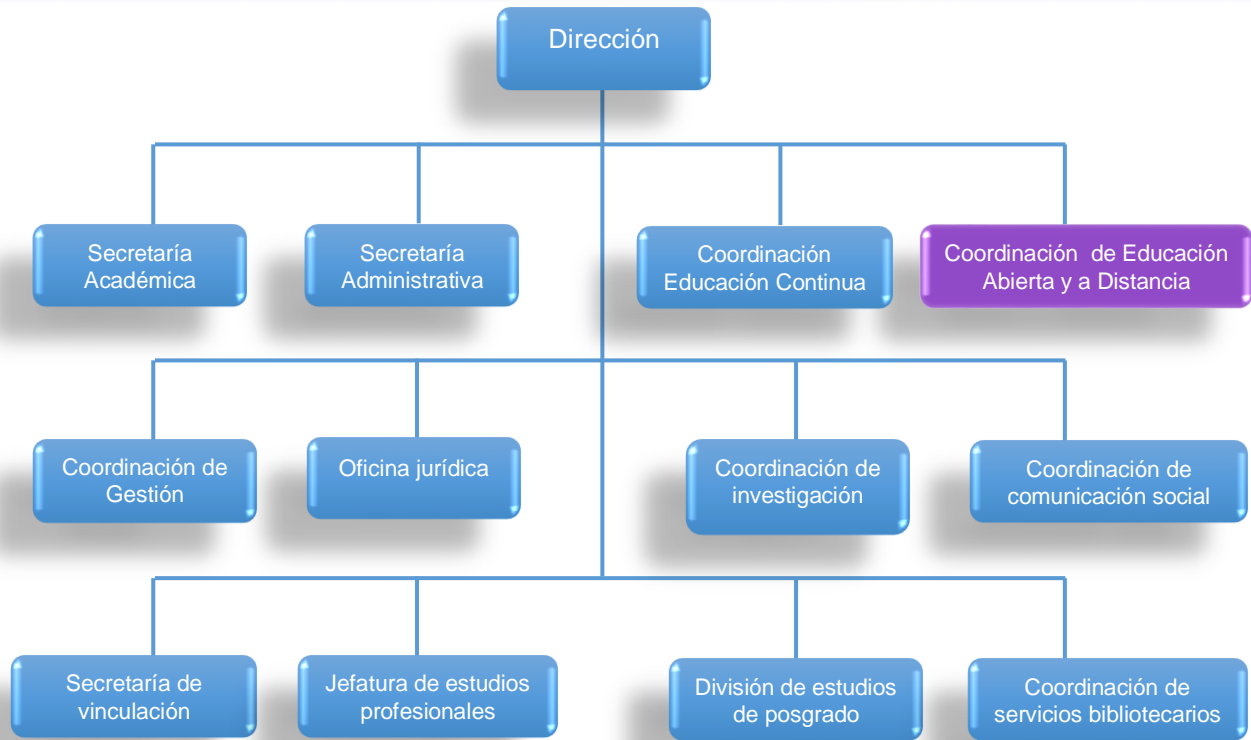


Figura 1.2.1 - Organigrama General de la Institución

La figura 1.2.2 representa el organigrama de la Coordinación de Educación Abierta y a Distancia.

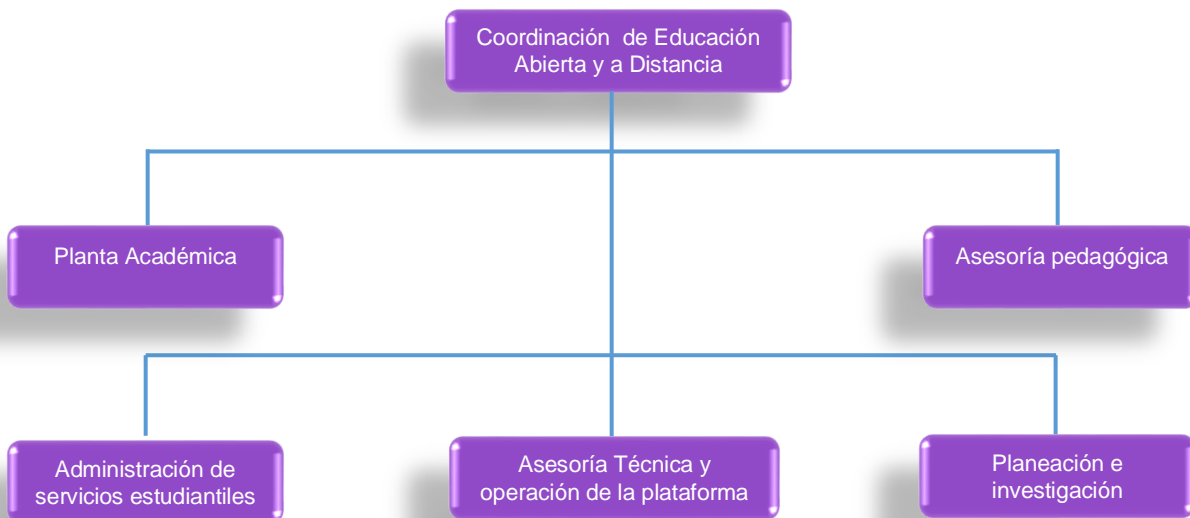


Figura 1.2.2 - Organigrama de la Coordinación de Educación Abierta y a Distancia



La Coordinación de Educación Abierta y a Distancia, es la encargada de fortalecer el modelo de educación a distancia, ofreciendo licenciaturas en esta modalidad educativa, desarrollando recursos que responden a las necesidades particulares de sus estudiantes y poniendo a su disposición los recursos tecnológicos con los que cuenta, logrando llevar la oferta educativa a distintas comunidades más allá del límite de las aulas. Lleva a cabo la planeación y el desarrollo de actividades académicas de calidad relacionadas con la formación, actualización y capacitación en áreas específicas.

La planta académica, se encarga de estructurar y dirigir a los docentes en función de los planes de desarrollo institucional, está conformada por diferentes figuras académicas desde ayudantes de profesor, profesores de asignatura, profesores de carreta, técnicos académicos, entre otros; los cuales deben poseer ciertos requerimientos académicos e institucionales con el fin de garantizar un enfoque integral y de compromiso institucional, adicional a esto se encarga de realizar la revisión de aulas disponibles para la organización de asesorías.

Asesoría pedagógica, tiene como propósito desarrollar materiales claros y entendibles para cada una de las asignaturas que se imparten a través de la modalidad a distancia, estructurándolos de la mejor manera posible y trabajando en conjunto con los profesores de asignatura. También se encarga de dar orientación a profesores para mejorar su desempeño con los alumnos, y así disminuir la deserción. Trabajan en conjunto con el área técnica para el vaciado de contenido en la plataforma online.

Administración de servicios estudiantiles, es la encargada de planear, organizar, difundir y controlar los servicios relacionados con los alumnos, adicionalmente da atención telefónica a cualquier aspecto académico con el fin de ayudar de forma individual y colectiva al estudiante en el proceso enseñanza-aprendizaje y a su formación integral. Adicionalmente coordina la documentación e información necesaria para los programas académicos; prepara y presenta los informes que se requieran por parte de la máxima autoridad en la institución.



Planeación e investigación, se encarga de planear, organizar, dirigir, controlar y evaluar las actividades de planeación, programación, servicios escolares con la normatividad vigente, verifica el cumplimiento de los procedimientos establecidos para la planeación, coordina y supervisa la elaboración de planes de desarrollo institucional y las estrategias a seguir para su ejecución.

Asesoría técnica y operación de la plataforma, esta área es la encargada del proceso informático en la Coordinación, las actividades más destacadas son:

- Administración del servidor que contiene la información de la institución.
- Matriculación de alumnos y profesores en la plataforma.
- Búsqueda, instalación y pruebas de software libre para actividades estudiantiles.
- Actualización de versión para los distintos programas utilizados en la impartición de asignaturas.
- Vaciado de información de cursos en línea realizado por el área pedagógica.
- Distribución de equipos de cómputo para llevar a cabo las asesorías a distancia.
- Transmisiones en vivo de eventos significativos de la escuela.
- Soporte telefónico para resolución de inquietudes o problemas técnicos.
- Actualización de información en el portal oficial de la institución.

La realización de la programación manual de horarios se lleva a cabo entre el área de planta académica y el área de asesoría técnica, trabajan en conjunto para dividir la carga de trabajo que conlleva dicha programación.

Para comenzar con este proyecto es necesario agendar reuniones con el área de planta académica, ya que ellos cuentan con los salones aprobados por la institución para llevar a cabo las asesorías y los profesores asignados para impartir cada una de las materias del plan de estudios, para la programación de grupos y salones.

El área encargada del proyecto para el sistema de horarios es el área de asesoría técnica, ya que es la encargada de dar solución a distintas problemáticas de la institución. Se



trabjará en conjunto con el jefe del departamento de asesoría técnica, para desarrollar una plataforma de uso sencillo y entendible que facilite realizar la automatización de horarios, haciendo la revisión de lo que se tiene actualmente e identificando información que pueda ser de utilidad.

1.3 Normatividad que rige a la institución en cuanto a espacios.

La institución educativa retoma el plan de estudios del sistema presencial: la licenciatura abarca 48 asignaturas teórico-metodológicas. Para la asignación de espacios, se debe considerar que cada asignatura tiene programada:

- Una sesión de encuadre: Se lleva a cabo la presentación del profesor y el temario.
- Cuatro sesiones de asesoría: Con el profesor, ya sea de manera presencial o por algún medio virtual.

Aunque la modalidad es abierta o a distancia, cada profesor debe contar con un aula exclusiva, en un horario definido, en la cual tendrá equipo de cómputo para llevar a cabo las asesorías.

Es importante, tomar en cuenta la programación de exámenes ordinarios en primera y segunda vuelta. Algunas variantes utilizadas, para la realización de los exámenes, son:

- Examen ordinario por videoconferencia: El alumno debe acudir a un Centro a Distancia que cuente con la infraestructura necesaria para realizar la comunicación, en el día y hora acordados para el enlace.
- Examen ordinario vía chat: Tanto el asesor como el alumno ingresan a la plataforma, en el día y hora acordada. El examen es personalizado y en tiempo real.
- Examen ordinario presencial: Se realiza en la fecha y hora acordada entre el alumno y el asesor. Puede ser aplicado por el asesor o por personal acreditado, en el caso de las sedes foráneas.



“De acuerdo a los estatutos del Sistema de Universidad Abierta y Educación a Distancia, para cumplir los objetivos del programa, la Institución educativa puede utilizar, además de sus propias instalaciones, las de otras instituciones de educación superior, las de empresas públicas, privadas, asociaciones, comunidades, sindicatos y otras que se pongan a su disposición”¹.

Para las sedes externas, se solicita que los espacios asignados al programa cuenten con, al menos, las siguientes características:

- Cubículo: Con tres computadoras y mesa de trabajo para cuatro personas.
- Sala de cómputo: Mínimo con siete computadoras y una mesa de trabajo para ocho personas.
- Sala de proyecciones: Destinada a videoconferencias grupales con capacidad para 26 alumnos.

Adicionalmente, contar con los siguientes recursos a disposición de los asesores: Scanner, cámara digital, equipo multifuncional, cámara de vídeo digital, videograbadora, grabador de DVD, reproductor de CD y DVD, audífonos multimedia con micrófono integrado y grabadora de audio.

Actualmente se cuentan con ocho sedes en la República Mexicana: Ciudad de México, Estado de México (Toluca, Tlalnepantla y Chimalhuacán), Puebla, Tlaxcala, Querétaro y Oaxaca.

La distribución de espacios es la siguiente:

- Ciudad de México: 15 salones, 1 centro de cómputo, 1 sala de seminarios y 1 sala de videoconferencia para aula virtual.

¹ Universidad Nacional Autónoma de México (2014). Reglamento del estatuto del Sistema de Universidad Abierta y Educación a Distancia. Recuperado el 20 de abril de 2016, de <http://www.ddu.unam.mx>.



- Chimalhuacán: 8 salones, 1 centro de cómputo, 2 salas de usos múltiples SUAyED, 2 cubículos y 1 sala de videoconferencia, todos para implementación de aula virtual.
- Oaxaca: 2 salones, 2 cubículos, 1 sala de usos múltiples SUAyED y 1 sala de videoconferencia, todos para implementación de aula virtual.
- Puebla: 4 salones, 2 cubículos, 1 centro de cómputo, 1 sala de usos múltiples SUAyED y 1 sala de videoconferencia, todos para implementación de aula virtual.
- Tlalnepantla: 3 salones, 1 cubículo, 1 sala de videoconferencia, todos para implementación de aula virtual; y 1 salón.

En la programación de espacios se llegan a agrupar distintas sedes:

- Toluca – Tlaxcala: 1 salón para aula virtual.
- Puebla – Tlaxcala: 4 cubículos, 1 centro de cómputo, 1 sala de usos múltiples SUAyED y 1 sala de videoconferencia, todos para implementación de aula virtual.
- Chimalhuacán – Oaxaca: 1 salón, 1 cubículo, 1 centro de cómputo y 1 sala de videoconferencia, todos para implementación de aula virtual.
- Oaxaca – Toluca: 1 salón y 1 sala de usos múltiples SUAyED, ambos para implementación de aula virtual.
- Oaxaca – Tlaxcala: 1 cubículo, 1 centro de cómputo y 1 sala de usos múltiples SUAyED, todos para implementación de aula virtual.
- Oaxaca – Querétaro: 1 centro de cómputo y 1 sala de usos múltiples SUAyED, ambos para implementación de aula virtual.
- Toluca – Querétaro: 1 salón y 1 centro de cómputo, ambos para implementación de aula virtual.
- Puebla – Tlaxcala – Oaxaca: 1 sala de usos múltiples SUAyED para implementación de aula virtual.
- Toluca – Puebla – Querétaro: 1 salón y 1 cubículo, ambos para implementación de aula virtual.



“Existen dos tipos de aulas virtuales: síncrona y asíncrona. El uso de una u otra depende del asesor. El aula virtual síncrona permite una comunicación en tiempo real, por ello, los participantes deben estar conectados en el mismo momento. En el aula virtual asíncrona la comunicación no se produce en el momento, ya que los participantes no necesitan estar conectados al mismo tiempo”².

Normatividad Web

Además de la normatividad respecto a los espacios, otra normativa que también es importante acatar, es lo correspondiente a la normatividad web, debido a que el proyecto a desarrollar es un sistema web institucional.

En lo general se deben seguir los siguientes lineamientos:

- Para asegurar la velocidad en el despliegue del sistema, se debe cuidar que el peso máximo de una página principal no exceda los 500 KB. En el caso de las páginas internas, el peso ideal es 300 KB.
- Respecto a compatibilidad, el sitio debe ser compatible con los navegadores: Firefox 2.0 en adelante, Google Chrome 2.0 en adelante, Internet Explorer 6 en adelante y Safari 3.0 en adelante.
- El escudo de la institución debe estar desplegado en cada página del sistema, y tener una liga al sitio principal de la Institución educativa.
- Evitar que la funcionalidad importante o indispensable de las páginas sea dependiente de JavaScript, ya que eso puede ocasionar problemas de accesibilidad si el usuario desactiva esta tecnología en su navegador.
- Es importante que exista un mapa del sitio.
- Todos los sistemas web institucionales deben operar ininterrumpidamente los 365 días del año, las 24 horas del día.

² Universidad Nacional Autónoma de México (2012). Coordinación de Universidad Abierta y Educación a Distancia. Recuperado el 20 de abril de 2016, de <http://aulasvirtuales.cuaed.unam.mx>.



- La resolución recomendada es de 1024x768.
- Los sistemas web institucionales deben incorporar el encabezado institucional, el cual está integrado por dos elementos: La imagen con el escudo y el nombre completo de la Universidad a la izquierda del encabezado. Y la imagen a la derecha del encabezado, puede ser el nombre del servicio y el logo de la dependencia.
- Si se utilizan animaciones con Flash, deben ser elaboradas para las versiones más bajas de plug-ins Flash que sea posible, para asegurar la compatibilidad con la mayoría de los usuarios.
- El cierre de sesión de cualquier servicio con autenticación de usuario debe ser claro en cada página.
- El diseño del sitio debe ser sencillo y utilizar en forma moderada elementos decorativos.
- Se deben evitar las animaciones innecesarias.
- “Cuidar en las páginas internas que el color de fondo sea claro y la letra oscura, para aumentar el contraste, obteniendo mayor legibilidad”³.

1.4 Descripción de los procedimientos para la asignación de salones horarios, grupos, etc.

Actualmente la Institución Educativa utiliza un método manual para la elaboración de los calendarios para asignación de grupos, salones y horarios para el Sistema de Educación Abierta y a Distancia. El mismo se designa a la programación y sistematización de actividades educativas durante el semestre correspondiente.

³ Universidad Nacional Autónoma de México (2015). Red de responsables en visibilidad Web UNAM. Recuperado el 21 de abril de 2016, de <http://www.visibilidadweb.unam.mx/catic.html>.



Buscando siempre cumplir estos requisitos:

- “Disponibilidad: La disponibilidad es un servicio que garantiza que los usuarios autorizados tengan acceso a la información”⁴.
- “Exactitud: Abordar un tema con detalles y argumentos sólidos, es decir, sin información ambigua”⁵.
- “Integridad: Es la propiedad que busca mantener los datos libres de modificaciones no autorizadas”⁶.

La calendarización es una tarea principal y primordial dentro del Sistema de y Educación Abierta y a Distancia, es una norma que regula el funcionamiento de la Institución y que de ello depende que el alumnado pueda tener tanto la asesoría presencial así como las herramientas para poder seguir desarrollando una mejor experiencia educativa, esto mediante la correcta asignación al profesor de una aula y horario exclusivo para la impartición de su materia , incluyendo un equipo de cómputo que se utilizará para llevar a cabo las tareas propias de las asesorías.

Esta tarea se debe realizar con antelación ya que se debe tener en consideración un número aproximado del alumnado, que se inscribirá mediante esta modalidad, para poder calcular el número de grupos que se darán de alta en el semestre.

Para la elaboración del calendario en primera instancia se generan las tablas con los profesores y las materias que se impartirán en el semestre en curso. Posteriormente se generan las tablas con los salones y se queda en espera la formación de los grupos para generar el calendario de las asesorías.

⁴ Universidad Nacional Autónoma de México (2012). Fundamentos de seguridad informática. Recuperado el 23 de abril de 2014, de <http://redyseguridad.fi-p.unam.mx/proyectos/seguridad/ServDisponibilidad.php>.

⁵ Pérez Porto, J. Gardey, A. (2008). Definición.de. Recuperado el 23 de abril de 2016, de <http://definicion.de/exactitud>.

⁶ SlideShare (2016). Integridad de la Información. Recuperado el 24 de abril de 2014, de <http://es.slideshare.net/CharlySantana1/integridad-de-la-informacion>.



A continuación se muestra la elaboración manual del calendario de asesorías para los planes de educación abierta y a distancia.

La tabla de profesores se muestra en la figura 1.4.1

	A	B	C	D
1				
2		NOMBRE	CLAVE DE LA ASIGNATURA	MATERIA
3		Roberto Mellado	1111-ED	Teoria Social
4		Ingrid Miranda	1115-ED	Necesidades y Problemas de México
5		Francisco Portillo	1116-ED	Lógica y Epistemología
6		Salvador Alvarado	1114-ED	Análisis del Estado Mexicano
7				

Figura 1.4.1 – Tabla de profesores

Tabla de grupos figura 1.4.2.

	A	B	C	D	E	F	G
1							
2		Grupo	Fecha	Horario	Edificio	Salon	
3		9001	13 de Febrero	08:00-9:00	C		2
4		9001	13 de Febrero	11:00 - 12:30	C		2
5		9001	16 de Febrero	7:00 - 8:30	C		2
6		9001	16 de Febrero	08:30 - 10:00	C		2
7							
8							

Figura 1.4.2 – Tabla de grupos

El calendario final se muestra en la figura 1.4.3.

Nombre	Clave	Materia	Grupo	Fecha	Horario	Edificio	Salón
Roberto Mellado	1111 – ED	Teoría Social	9001	13 de Febrero	8:00 – 9:00	C	2
Ingrid Miranda	1115 – ED	Necesidades y Problemas Sociales	9001	13 de Febrero	11:00 – 12:30	C	2
Francisco Portillo	1116 – ED	Lógica y Epistemología	9001	16 de Abril	7:00 – 8:30	C	2
Salvador Alvarado	1114 – ED	Análisis del Estado Mexicano	9001	16 de Abril	8:30 – 10:00	C	2

Figura 1.4.3 – Calendario de Asesorías

En la figura 1.4.4 se muestra una representación gráfica de la secuencia de operaciones que se realizan para obtener la tabla de calendario de asesorías figura 1.4.3, nos muestra la relación que hay entre las tablas mostradas en las figuras 1.4.1 y 1.4.2, así mismo nos da una idea más clara y detallada sus atributos y de cómo convergen ambas tablas.

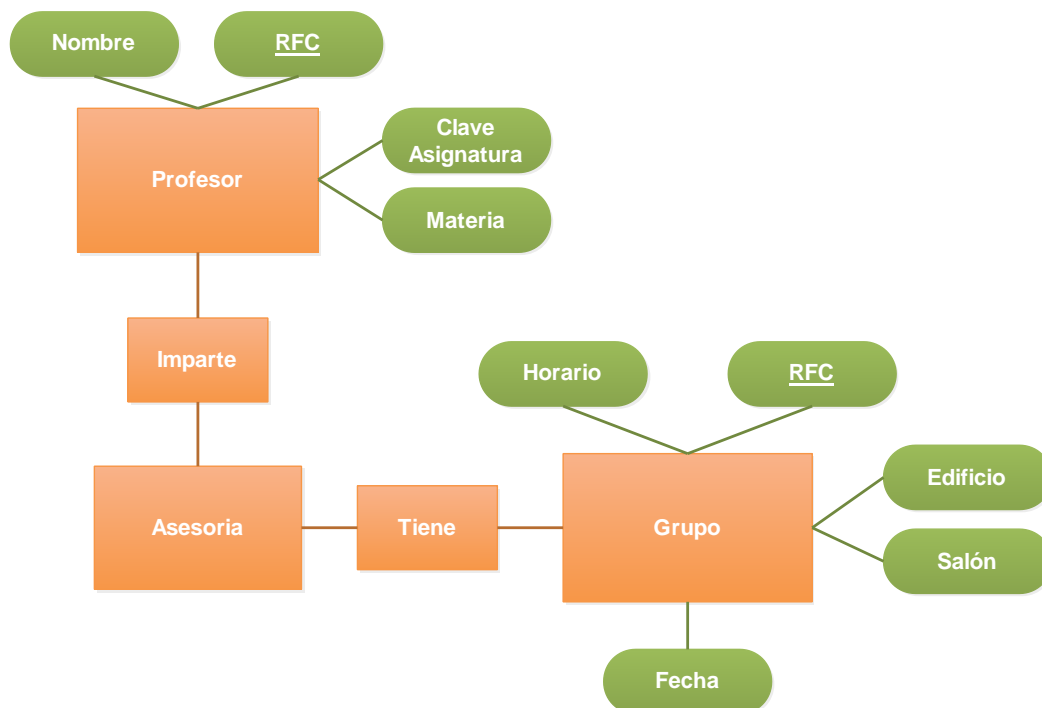


Figura 1.4.4 – Esquema calendario



Este método resulta un tanto anticuado ya que para dicha asignación de grupos, salones y horarios de asesorías, este método lleva a un trabajo ineficiente, repetitivo y en ocasiones erróneo, lo que conlleva a la utilización de más recursos tanto humanos como inversión de tiempo.

Como lo demuestran las tablas anteriores, al ser un procedimiento manual esto implica que en la generación del calendario de asesorías se involucren de 2 a 3 personas lo que nos indica que la información pasa por muchos cambios antes de llegar a su formato final.

Todo este proceso nos indica que se está haciendo uso de recursos humanos que están realizando, en esencia, la misma tarea e involucra el uso de tiempo que se pudiera aprovechar en tareas más productivas.

Esta situación es la principal razón por la cual la información que se está generando debe ser revisada minuciosamente, ya que existe la posibilidad de que la información presentada no sea la correcta, en conclusión, es muy probable el error humano.

El proceso final es subir el archivo a la plataforma de la Institución, aquí nos encontramos con una probable incidencia ya que si por alguna razón la información es errónea se debe actualizar haciendo nuevamente el proceso anteriormente mencionado.

Al final podemos observar que la calendarización manual es un proceso complejo, obsoleto e inseguro, que a pesar de contar con un número de colaboradores importante se podría concluir como un trabajo incapaz de satisfacer las necesidades actuales de contar con una correcta administración de grupos.

En base a la necesidad de contar con un sistema que permita facilitar la correcta administración de grupos, es por lo que se ha pensado en implementar un programa que sea de fácil manejo y también automatice todos los procedimientos que a la fecha de hoy se siguen realizando de forma manual.



Con la implementación del sistema de automatización se busca el tener mayores beneficios en tiempo y esfuerzo, así como garantizar la fiabilidad y seguridad de la información y reducir al mínimo los errores antes mencionados.

1.5 Alcance y acotamiento del problema.

Entre los problemas que se encuentran comúnmente dentro de las instituciones educativas, está el de elaborar horarios que cumplan con indicaciones particulares para cada materia, como lo es el número de horas que se debe impartir; condiciones pedagógicas, tales que como una materia no se debe impartir más de cierto número de horas al día, la disponibilidad de los profesores; así como algunas políticas internas de la institución.

La problemática en particular de la institución que se desea abordar en este trabajo es elaborar horarios de asesorías que se adecue a las necesidades particulares de la institución y de los profesores. Al observar la complejidad que genera la elaboración de la programación de asesorías, surge la necesidad de desarrollar una aplicación web que permita automatizar este proceso.

Se diseñará un sistema que permita automatizar y agilizar la planeación de asesorías de una institución de educación abierta y a distancia, con la finalidad de agilizar la asignación de profesores, salones y horarios que actualmente se realiza en forma manual. Con esto se logrará un ahorro de tiempo considerable, facilitando el proceso y eliminando errores de planificación en la logística interna.

El sistema deberá cumplir con las siguientes características:

- Automatizar los procesos que actualmente se realizan manualmente.
- La interfaz deberá ser entendible y sencilla para los usuarios
- Eliminar los errores de traslape de salones y horarios.



- Desarrollar un sitio seguro en el cual, sólo personal autorizado podrá crear, actualizar y borrar datos.
- Simplificar la búsqueda de horarios dando mayor opción para consultarlos.

Se busca que el sistema cuente con una interfaz amigable y se tenga en cuenta una experiencia de usuario adecuada para el manejo de los horarios, permitiendo una interactividad intuitiva disminuyendo el margen de errores humanos en el manejo de la información. En la figura 1.5.1 mostramos los ciclos por los cuales tendrá que pasar nuestra interfaz.

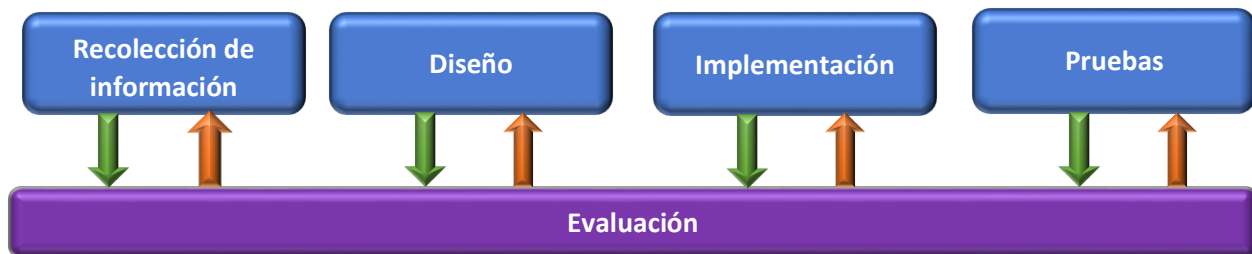


Figura 1.5.1 Desarrollo de la interfaz

Para lograr lo anterior, necesitamos que nuestra interfaz cuente con los siguientes puntos:

- Sencilla. Los elementos están para apoyar, ayudar y guiar, no para confundir. Evitemos la saturación y colocación innecesaria de los mismos.
- Clara. La información debe ser fácilmente localizable, es decir, debe estar organizada ya sea de manera lógica, jerárquica o temática.
- Predecible. A acciones iguales, resultados iguales.
- Flexible. La publicación deber verse y entenderse claramente en la mayoría de los navegadores y plataformas. Debemos pensar siempre en opciones (“botones”, textos) que nos permitan regresar al punto de partida (de ser posible) y a la página principal.
- Consistente. Lograr una semejanza entre las secciones o capítulos. Los elementos una vez organizados y definidos deben permanecer en la misma área, con la



misma función y con el mismo texto o imagen, la única que permite presentar variaciones de la distribución de los elementos es la página principal, pero la representación gráfica y de texto que se haga de las funciones debe permanecer igual en todas las secciones.

- Intuitiva. El usuario se siente más seguro en una publicación en la que no tenga que adivinar ni pensar como ejecutar acciones.
- Coherente. Tanto texto como gráficos, colores y demás elementos utilizados deben corresponder al contenido de la publicación. Apoyados generalmente por una construcción de palabras, frases y elementos visuales.

El desarrollo del sistema estará basado en software libre, disminuyendo los costos inherentes a las licencias, permisos y derechos; y así no depender de terceros. Se elegirán las mejores herramientas de desarrollo libre en el mercado actual, haciendo uso de programación web, garantizando el acceso desde cualquier punto por los usuarios autorizados para ello, además de permitir la escalabilidad del sistema y la actualización de manera modular. De manera general el sistema contará con los aspectos que se muestran en la figura 1.5.2

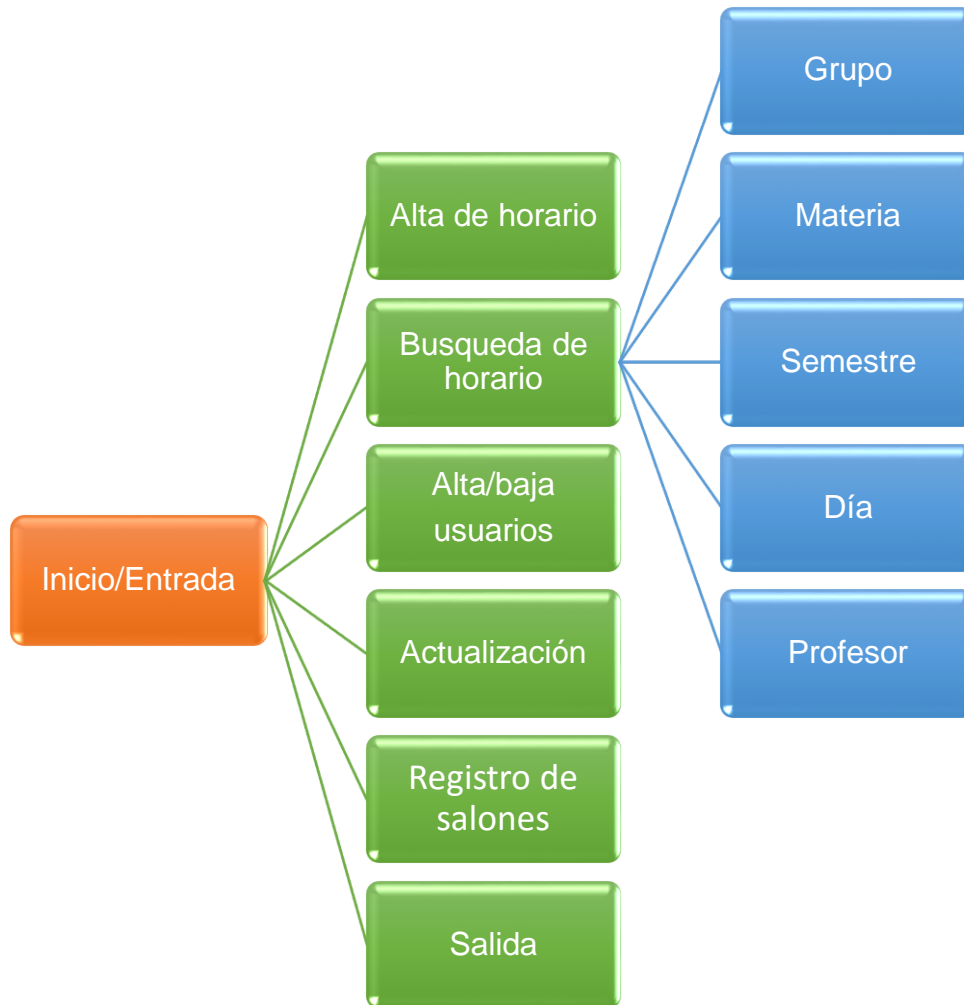


Figura 1.5.2 Diagrama general del sistema.

Como toda aplicación web se tendrá una pantalla de inicio de sesión donde sólo podrá acceder el personal encargado de la programación de horarios, profesores y salones, también se podrán dar de alta y baja usuarios.

Se podrán dar de alta horarios para cada grupo y materia donde se indicará el lugar, día, hora de inicio y fin de sesión, también se podrá realizar la actualización y baja de los mismos.

Es importante considerar que los salones asignados no siempre son los mismos, así que también se contará con un apartado para agregar o eliminar salones. Debido a la



necesidad de tener la información disponible para alumnos y profesores, el sitio dispondrá de una opción para pasar de datos almacenados a un formato PDF.

Se tendrá una sección de búsqueda de horario que podrá ser por grupo, materia, semestre, profesor y día; también se podrá visualizar todos los registros almacenados de una manera sistematizada.



CAPÍTULO II

“MARCO TEÓRICO”

2.1 Características, ventajas y desventajas de las bases de datos relacionales y el manejador de bases de datos MySQL.



Figura 2.1.1 - Logo MySQL

MySQL es un sistema de administración de bases de datos relacionales y estructuradas, es considerada como la base de datos de código abierto más popular del mundo (cualquier persona puede estudiar el código y ajustarlo a sus necesidades) debido a que se asocia con una gran gama de aplicaciones basadas en web. Maneja un lenguaje de consulta estructurado SQL, se puede utilizar en aplicaciones simples y sencillas hasta aplicaciones de alto grado de complejidad y robustas, es multihilo y multiusuario.

Características.

- Disponibilidad en gran cantidad de plataformas y sistemas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.
- Conectividad segura pues ofrece un sistema de contraseñas y privilegios.
- Fiable.
- Veloz y flexible.
- Totalmente ampliable.
- Alto rendimiento.
- Usa tablas almacenadas en disco para búsquedas rápidas.
- Contiene su propio paquete de pruebas de rendimiento.



Ventajas.

- Es Open source.
- Velocidad al realizar operaciones.
- Alto desempeño.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Fácil de usar.
- Fácil búsqueda e indexación.
- Replicación.
- Bajo costo en requerimientos pues puede ser ejecutado en máquinas con escasos recursos sin ningún problema.
- Facilidad de configuración e instalación.
- Soporta gran variedad de Sistemas Operativos.
- Baja probabilidad de que los datos se corrompan.
- Totalmente escalable.

Desventajas.

- Puede provocar problemas de integridad en entornos de alta concurrencia.
- Gran porcentaje de utilidades no están documentadas.
- No es intuitivo.
- Si se abusa de los índices resulta contraproducente pues provoca lentitud.

Bases de datos relacionales

Una base de datos es una colección estructurada de tablas que contienen datos. Una base de datos relacional, archiva datos en tablas separadas en vez de colocar todos los datos en un archivo, esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas, esto hace posible combinar datos de diferentes tablas.

Estructura básica de almacenamiento.

Tabla, objeto de almacenamiento perteneciente a la BD, tiene registros o filas de datos y un nombre único.

Registro, cada una de las filas de una tabla, está compuesto por campos o atributos.

Campo, cada campo tiene un nombre único para la tabla de la cual forma parte, y es de un tipo determinado.

En la tabla 2.1.1 podemos observar la representación lógica, física y el modelo relacional.




Representación lógica	Representación Física	Modelo relacional
 Tabla	Archivo secuencial	Relación
 Fila	Registro	Tupla
 Columna	Campo	Atributo

Tabla 2.1.1 Representación de una base de datos relacional

Llave primaria, campo o combinación de campos que identifica de forma única a cada fila de una tabla.

Llave foránea, es una referencia a una clave de otra tabla, determina la relación que existe entre dos tablas.

Cardinalidad de relaciones.

Relación uno a muchos (1, n). Se crea una relación de uno a muchos si uno de los campos relacionados es una llave primaria.



Relación uno a uno (1, 1). Se crea una relación de este tipo si ambos campos relacionados son llaves primarias.

Relación de muchos a muchos (n, m). En este caso ninguno de los campos relacionados son llaves primarias. En este tipo es necesaria la creación de una tabla intermedia que esté relacionada de uno a muchos.

Características de las Bases de Datos Relacionales.

- La base de datos se compone de varias tablas o relaciones.
- No pueden existir dos tablas con el mismo nombre.
- Cada tabla contiene un conjunto de campos (llamados columnas) y registros (filas).
- La relación entre una tabla padre y una hija se lleva a cabo por medio de claves primarias y claves foráneas.
- Las llaves primarias son la clave principal de un registro dentro de una tabla.
- Las llaves foráneas se colocan en la tabla hija, contiene el mismo valor que la clave primaria en la tabla padre.

Algebra Relacional.

Es un conjunto de operaciones que se realizan sobre las tablas dando como resultado nuevas relaciones, esto da la posibilidad de anidar y combinar operadores para tener un resultado más detallado.

- Selección, consiste en recuperar un conjunto de registros de una tabla o de una relación indicando las condiciones que deben cumplir los registros recuperados.
- Unión, esta operación permite combinar datos de varias relaciones, para poder realizarla es necesario que las tablas a unir tengan las mismas estructuras y que sus campos sean iguales.
- Intersección, esta operación permite identificar filas que son comunes en dos relaciones, dando como resultado los elementos que existan en ambas tablas.
- Diferencia, permite identificar filas que están en una relación y no en otra.



- Producto cartesiano, consiste en la realización de un producto cartesiano entre dos tablas dando como resultado todas las posibles combinaciones entre los registros de ambas tablas.
- Proyección, permite extraer una columna de una tabla dando como resultado un subconjunto vertical de atributos en una relación.
- Combinación, la combinación de dos relaciones se obtiene mediante una selección y una proyección a partir del producto cartesiano de las dos tablas, que tengan algún atributo en común; quitando renglones donde el atributo común no sea igual y eliminando atributos duplicados.

Ventajas de las Bases de Datos Relacionales.

- Posee herramientas que garantizan evitar duplicidad de registros.
- Garantiza la integridad referencial, así al eliminar un registro elimina todos los registros relacionados dependientes.
- Favorece la normalización.
- Independencia física, el modo en el que se almacenan los datos no influye en su manipulación lógica y por tanto usuarios que acceden a estos datos tienen que modificar sus programas por cambios en el almacenamiento físico.
- Independencia lógica, añadir, eliminar o modificar objetos de la base de datos no repercute en programas o usuarios que acceden a subconjuntos parciales de las mismas.
- Flexibilidad, en el sentido de poder presentar a cada usuario los datos de la forma en que este prefiera.
- Uniformidad, las estructuras lógicas de los datos presentan un aspecto uniforme, lo que facilita la manipulación de la base de datos por parte de los usuarios.
- Sencillez, el modelo de datos relacional es fácil de comprender y utilizar por parte del usuario final.

Desventajas de las Bases de Datos Relacionales.

- Presenta deficiencias con datos gráficos, multimedia, y sistemas de información geográfica.
- El personal que maneje la base de datos relacional debe tener cierto nivel de conocimientos para administrarla adecuadamente.

2.2 Características, ventajas y desventajas de la arquitectura cliente-servidor.

“Bajo el protocolo TCP/IP las comunicaciones entre computadoras se rigen básicamente por lo que se llama arquitectura Cliente-Servidor, éste es un modelo que intenta proveer usabilidad, flexibilidad, interoperabilidad y escalabilidad en las comunicaciones”⁷.

Como se observa en la figura 2.2.1 la arquitectura cliente-servidor consiste básicamente en un cliente que realiza peticiones a otro programa que funciona como un servidor que le da respuesta. Aunque esta acción se puede aplicar sobre programas que se ejecutan en una sola computadora, es más ventajoso utilizar un sistema operativo multiusuario distribuido a través de una red de computadoras.

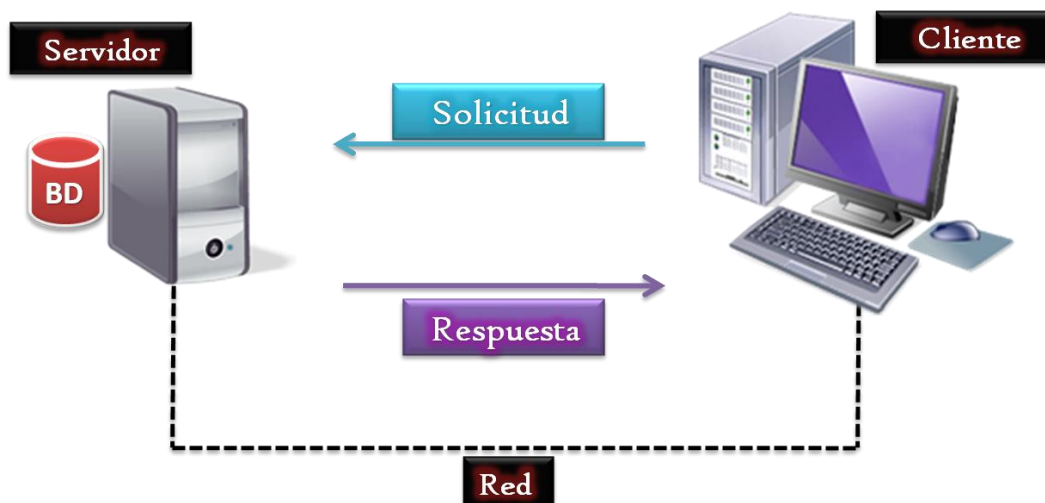


Figura 2.2.1 Arquitectura Cliente-Servidor

⁷ Marquez Avensano, B. (2004). Arquitectura Cliente-Servidor. Recuperado el 26 de abril de 2016, de http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/marquez_a_bm/capitulo5.pdf



“El modelo cliente-servidor es aquel en la que todos los clientes están conectados a un servidor, en el que se centralizan los diversos recursos y aplicaciones con que se cuenta; y que los pone a disposición de los clientes cada vez que estos son solicitados. Esto significa que todas las gestiones que se realizan se concentran en el servidor, de manera que en él se disponen los requerimientos provenientes de los clientes que tienen prioridad, los archivos que son de uso público y los que son de uso restringido, los archivos que son de sólo lectura y los que, por el contrario, pueden ser modificados, etc”⁸.

Generalmente la mayoría del trabajo pesado se hace en el proceso llamado servidor y el o los procesos cliente sólo se ocupan de la interacción con el usuario, aunque esto puede variar. La arquitectura Cliente/Servidor es una extensión de programación modular en la que la base fundamental es separar una gran pieza de software en módulos con el fin de hacer más fácil el desarrollo y mejorar su mantenimiento.

Esta arquitectura permite distribuir físicamente los procesos y los datos en forma más eficiente lo que en computación distribuida afecta directamente al tráfico de la red, reduciéndolo significativamente.

Cliente.

Es el que le permite realizar los requerimientos al servidor, se le conoce con el término front-end. Realiza la manipulación y despliegue de datos, envía una petición al servidor y se queda esperando por una respuesta. Su tiempo de vida es finito una vez que son servidas sus solicitudes, termina el trabajo.

Están diseñadas en plataformas que permiten construir interfaces graficas de usuario (GUI). Las funciones que lleva a cabo el cliente son las siguientes:

- Administrar la interfaz de usuario.

⁸ Leal Castellanos, M. (2015). Taller Cliente-Servidor. Recuperado el 26 de abril de 2016, de <https://basesii.wikispaces.com/file/view/Caracter%C3%ADsticas+de+la+arquitectura+Cliente.pdf>.



- Interactuar con el usuario.
- Procesar la lógica de la aplicación y hacer validaciones locales.
- Generar requerimientos de bases de datos.
- Recibir resultados del servidor.
- Formatear resultados.

Servidor.

Es un programa que ofrece un servicio que se puede obtener en una red. Acepta la petición, realiza el servicio y devuelve el resultado al cliente solicitante. Al ser posible implantarlo como aplicaciones de programas, puede ejecutarse en cualquier sistema donde exista TCP/IP y junto con otros programas de aplicación. El servidor comienza su ejecución antes de comenzar la interacción con el cliente. Su tiempo de vida o de interacción es “interminable”.

Los servidores pueden ejecutar tareas sencillas o complejas. Los servidores sencillos procesan una petición a la vez (son secuenciales o interactivos), por lo que no revisan si ha llegado otra petición antes de enviar la respuesta de la anterior.

Los más complejos trabajan con peticiones concurrentes aun cuando una sola petición lleve mucho tiempo para ser servida, puesto que tienen altos requerimientos de protección y autorización. Pueden leer archivos del sistema, mantenerse en línea y acceder a datos protegidos y a archivos de usuarios. No puede cumplir a ciegas las peticiones de los clientes, deben reforzar el acceso al sistema y las políticas de protección. Los servidores por lo general tienen dos partes:

- Programa o proceso que es responsable de aceptar nuevas peticiones: Maestro o Padre.
- Programas o procesos que deben manejar las peticiones individuales: Esclavos o Hijos.



Middleware.

El middleware es un módulo intermedio que actúa como conductor entre el cliente y servidor permitiendo comunicarse. Permite desarrollar aplicaciones en arquitectura Cliente/Servidor independizando los servidores y clientes, facilitando la interrelación entre ellos y evitando dependencias de tecnologías propietarias

El middleware se estructura en tres niveles:

- Protocolo de transporte.
- Network OperatingSystem (NOS).
- Protocolo específico del servicio.

Las principales características de un middleware son:

- Simplifica el proceso de desarrollo de aplicaciones al independizar los entornos propietarios.
- Permite la interconectividad de los Sistemas de Información del Organismo.
- Proporciona mayor control del negocio al poder contar con información procedente de distintas plataformas sobre el mismo soporte.
- Facilita el desarrollo de sistemas complejos con diferentes tecnologías y arquitecturas.

Características de la Arquitectura Cliente/Servidor.

- Es la unidad básica de diseño. El servidor los proporciona y el cliente los utiliza.
- Tiene recursos compartidos ya que muchos clientes utilizan los mismos servidores y, a través de ellos, comparten tanto recursos lógicos como físicos.
- Sus protocolos son asimétricos; los clientes inician “conversaciones” y los servidores esperan su establecimiento pasivamente.



- Transparencia de localización física de los servidores y clientes. El cliente no tiene por qué saber dónde se encuentra situado el recurso que desea utilizar.
- Independencia de la plataforma HW y SW que se emplee.
- Sistemas débilmente acoplados. Interacción basada en envío de mensajes.
- Encapsulamiento de servicios. Los detalles de la implementación de un servicio son transparentes al cliente.
- Escalabilidad horizontal (añadir clientes) y vertical (ampliar potencia de los servidores).
- La integridad de datos y programas centralizados en servidores facilitan su integridad y mantenimiento.

Ventajas.

- Facilita la integración entre sistemas diferentes y comparte información. De esta manera, se puede integrar PCs con sistemas medianos y grandes, sin necesidad de que todos tengan que utilizar el mismo sistema operativo.
- En el uso de interfaces gráficas para el usuario, presenta la ventaja, con respecto a uno centralizado, de que no siempre es necesario transmitir información gráfica por la red pues ésta puede residir en el cliente, lo cual permite aprovechar mejor el ancho de banda de la red.
- La estructura inherentemente modular facilita además la integración de nuevas tecnologías y el crecimiento de la infraestructura computacional, favoreciendo así la escalabilidad de las soluciones.
- Contribuye, además, a proporcionar a los diferentes departamentos de una organización, soluciones locales, pero permitiendo la integración de la información.

Desventajas.

- El mantenimiento de los sistemas es más difícil pues implica la interacción de diferentes partes de hardware y de software, distribuidas por distintos proveedores, lo cual dificulta el diagnóstico de fallas.



- Cuenta con escasas herramientas para la administración y ajuste del desempeño de los sistemas.
- Es importante que los clientes y los servidores utilicen el mismo mecanismo (por ejemplo sockets o RPC), lo cual implica que se deben tener mecanismos generales que existan en diferentes plataformas.
- Hay que tener estrategias para el manejo de errores y para mantener la consistencia de los datos.
- El desempeño (performance), problemas de este estilo pueden presentarse por congestión en la red, dificultad de tráfico de datos, etc.
- Tiene un costo elevado debido a la complejidad técnica del servidor.
- El servidor es el único eslabón débil en la red de cliente/servidor, debido a que toda la red está construida en torno a él.

Modelos Cliente/Servidor.

“Una de las clasificaciones mejor conocidas de las arquitecturas Cliente/Servidor se basa en la idea de capas, la cual es una variación sobre la división o clasificación por tamaño de componentes. Esto se debe a que se trata de definir el modo en que las prestaciones funcionales de la aplicación serán asignadas, y en qué proporción, tanto al cliente como al servidor. Dichas prestaciones se deben agrupar entre los tres componentes clásicos para Cliente/Servidor: interfaz de usuario, lógica de negocios y los datos compartidos, cada uno de los cuales corresponde a una capa”⁹.

Dentro de esta categoría tenemos las aplicaciones en dos capas, tres capas y multi-capas (distribuido).

- **Arquitectura Cliente – Servidor de 2 capas.**

La arquitectura en 2 capas que se ejemplifica en la figura 2.2.2, se utiliza para describir los sistemas cliente/servidor en donde el cliente solicita recursos y el servidor responde directamente a la solicitud, con sus propios recursos. Esto

⁹ Renaud Paul, E. (1996). Introduction to Client/Server Systems: A Practical Guide for Systems, 2a Ed. New Jersey: John Wiley & Sons.

significa que el servidor no requiere otra aplicación para proporcionar parte del servicio.

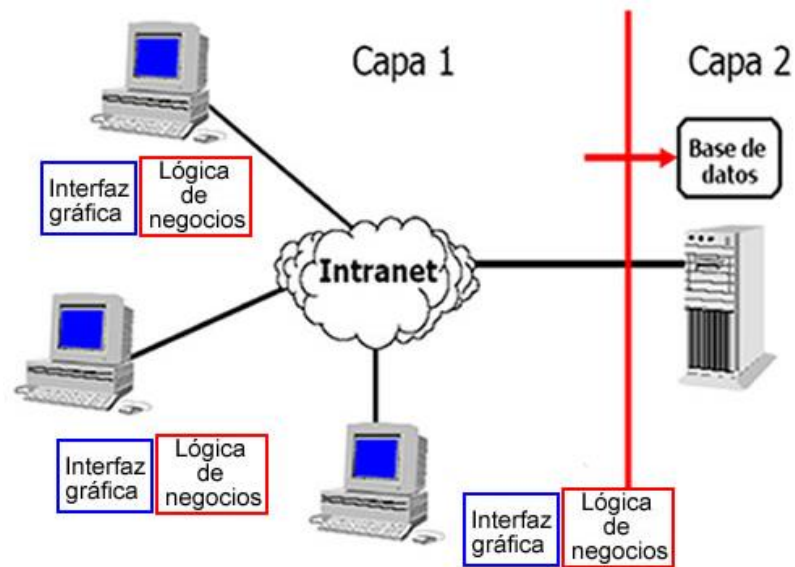


Figura 2.2.2 Cliente –Servidor de 2 capas

Ventajas.

- Presenta una estructura de desarrollo bastante simple ya que el programador maneja un único ambiente de desarrollo.

Desventajas.

- La gran cantidad de información que viaja al cliente congestiona demasiado el tráfico de red, lo que se traduce en bajo rendimiento.
 - Por su bajo rendimiento esta estructura tiene un bajo espectro de aplicación, limitándose a la construcción de sistemas no críticos.
- **Arquitectura Cliente – Servidor de 3 capas.**

Esta estructura se caracteriza por elaborar la aplicación en base a dos capas principales de software, más la capa correspondiente al servidor de base de datos

como se observa en la figura 2.2.3. Al igual que en la arquitectura dos capas, y según las decisiones de diseño que se tomen, se puede balancear la carga de trabajo entre el proceso cliente y el nuevo proceso correspondiente al servidor de aplicación.

En este esquema el cliente envía mensajes directamente al servidor de aplicación, el cual debe administrar y responder todas las solicitudes. Es el servidor, dependiendo del tipo de solicitud, quien accede y se conecta con la base de datos.

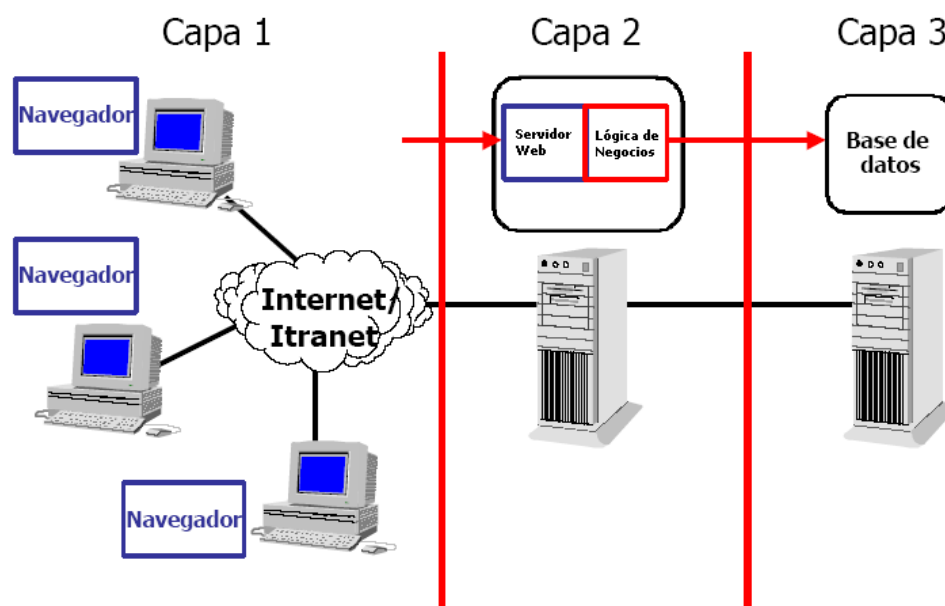


Figura 2.2.3 Cliente –Servidor de 3 capas

Ventajas.

- Reduce el tráfico de información en la red por lo que mejora el rendimiento de los sistemas (especialmente respecto a la estructura en dos planos).
- Brinda una mayor flexibilidad de desarrollo y de elección de plataformas sobre la cual montar las aplicaciones. Provee escalabilidad horizontal y vertical.



- Se mantiene la independencia entre el código de la aplicación (reglas y conocimiento del negocio) y los datos, mejorando la portabilidad de las aplicaciones.
- Los lenguajes sobre los cuales se desarrollan las aplicaciones son estándares lo que aumenta la portabilidad de las aplicaciones entre plataformas.
- Dado que mejora el rendimiento al optimizar el flujo de información entre componentes, permite construir sistemas críticos de alta fiabilidad.
- El mismo hecho de localizar las reglas del negocio en su propio ambiente, en vez de distribuirlos en la capa de interfaz de usuario, permite reducir el impacto de hacer mantenimiento, cambios urgentes de última hora o mejoras al sistema.
- Disminuye el número de usuarios (licencias) conectados a la base de datos.

Desventajas.

- Dependiendo de la elección de los lenguajes de desarrollo, puede presentar mayor complejidad en comparación con Cliente/Servidor de dos capas.
 - Existen pocos proveedores de herramientas integradas de desarrollo con relación al modelo Cliente/Servidor de dos capas, y normalmente son de alto costo.
- **Arquitectura Cliente – Servidor de n capas.**

En la arquitectura en 3 niveles, cada servidor (nivel 2 y 3) realiza una tarea especializada (un servicio). Por lo tanto, un servidor puede utilizar los servicios de otros servidores para proporcionar su propio servicio. Por consiguiente, la arquitectura en 3 niveles es potencialmente una arquitectura en N-niveles como se muestra en la figura 2.2.4.

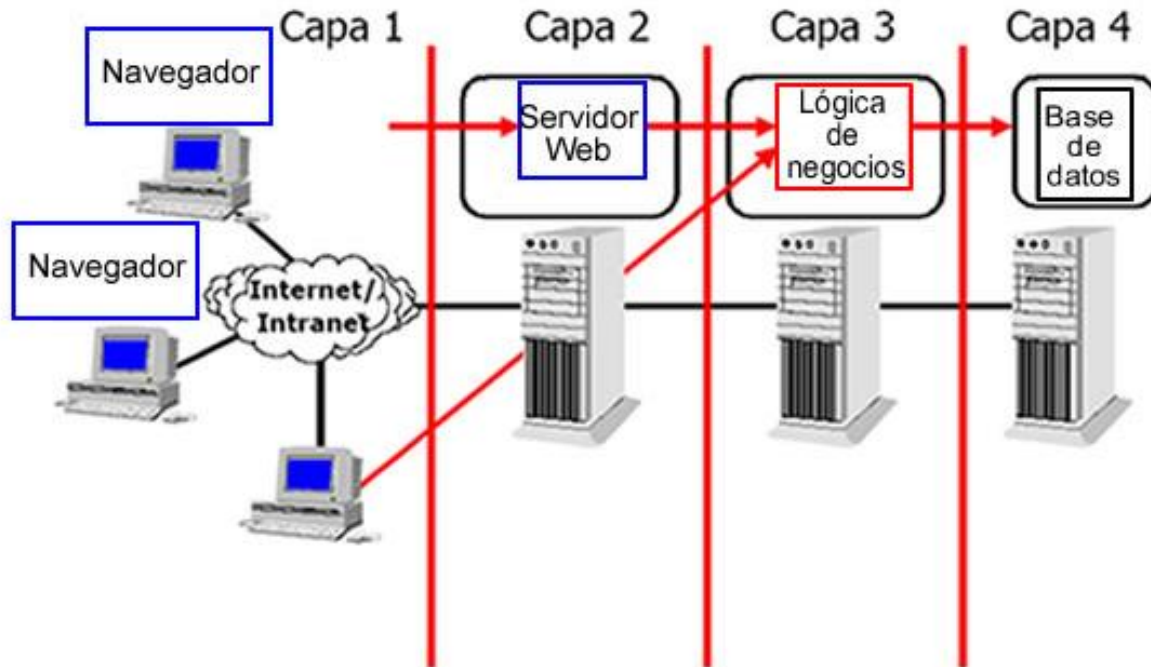


Figura 2.2.4 Cliente –Servidor de n capas

Ventajas.

- La ventaja fundamental de una arquitectura n-capas comparado con una arquitectura de dos niveles (o una tres-capas con una de dos niveles) es que separa hacia fuera el proceso, eso ocurre para mejorar el balance la carga en los diversos servidores; es más escalable.

Desventajas.

- Pone más carga en la red, debido a una mayor cantidad de tráfico de la red.
- Es mucho más difícil programar y probar el software que en arquitectura de dos niveles porque tienen que comunicarse más dispositivos para terminar la transacción de un usuario.



Servidor Flaco "cliente flaco".

Un cliente liviano o cliente ligero es una computadora cliente o un software de cliente en una arquitectura cliente-servidor que depende del servidor central para las tareas de procesamiento, y principalmente se enfoca en transportar la entrada y la salida entre el usuario y el servidor remoto.

Muchos dispositivos de cliente liviano corrían solamente navegadores web o programas de escritorio remoto, lo que significaba que todo el procesamiento significativo ocurría en el servidor. Algunos clientes livianos también son llamados "terminales de acceso".

Servidor gordo "cliente gordo".

Un servidor de cliente pesado no requiere tan alto nivel de desempeño como un servidor de cliente liviano (puesto que los clientes pesados por sí mismos hacen mucho del procesamiento de la aplicación). Esto puede resultar en servidores más baratos, aunque en la práctica, muchos servidores de cliente liviano son en realidad equivalentes a los servidores de archivos en cuanto a las especificaciones pero con memoria adicional.

Los clientes pesados tienen ventajas, en aplicaciones ricas en multimedia, que serían intensivas en ancho de banda si estuvieran completamente residentes en los servidores.

Los clientes livianos pueden ser inusualmente lentos, o muy frustrantes para usar, sobre una conexión de red de alta latencia. Por otra parte, no trabajan en absoluto cuando la red está caída. Con un cliente pesado, puede ser posible trabajar fuera de línea, aunque la manera orientada a red en la que mucha gente trabaja hoy en día, significa que el uso del cliente pesado también puede ser restringido si la red está caída.

2.3 Características, ventajas y desventajas del lenguaje PHP 5.6.19.



Figura 2.3.1 - Logo PHP

“PHP (PHP HypertextPreprocessor) es uno de los lenguajes de programación más utilizados en la actualidad (se utiliza mayormente para desarrollo de sitios web pero para muchos es ya un lenguaje de propósito general) y puede ser incrustado en HTML”¹⁰.

Esto se debe a múltiples factores, entre los cuales podemos mencionar los siguientes:

- Es libre y gratuito. Al igual que otros muchos lenguajes y entornos de programación, está amparado bajo el movimiento open source (código abierto), que permite a los programadores sacar provecho de sus beneficios de manera totalmente gratuita.
- Disponibilidad. Al momento de contratar un servicio de alojamiento, con seguridad, contaremos con el soporte necesario para empezar a programar nuestras aplicaciones.
- Evolución. Las mejoras continuas y avances dentro del lenguaje son gracias a una gran comunidad de desarrolladores que contribuyen y opinan acerca de cuáles podrían ser los avances incluidos en las próximas versiones y que elementos deberían ser corregidas en los siguientes lanzamientos.

¹⁰ Minera Francisco (2010). PHP 6: Sitios dinámicos con el lenguaje más robusto. Argentina: Gradi.



- Facilidad de aprendizaje. A diferencia de otros lenguajes, PHP se caracteriza por su simpleza: la curva de aprendizaje nos indica que aprender los fundamentos requiere mucho menos tiempo en comparación con otras tecnologías, y que profundizar sobre algunas cuestiones específicas no es tan difícil si contamos con una base teórica y una cierta experiencia, ya que la mayor parte de su sintaxis es similar a C, Java y Perl.

Su interpretación y ejecución se da en el servidor, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web generada por un script, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que genera el contenido de manera dinámica, pudiendo modificar el contenido a enviar y regresar el resultado al servidor, el cual se encarga de regresárselo al cliente. En la figura 2.3.2 se muestra un diagrama general del funcionamiento de PHP.

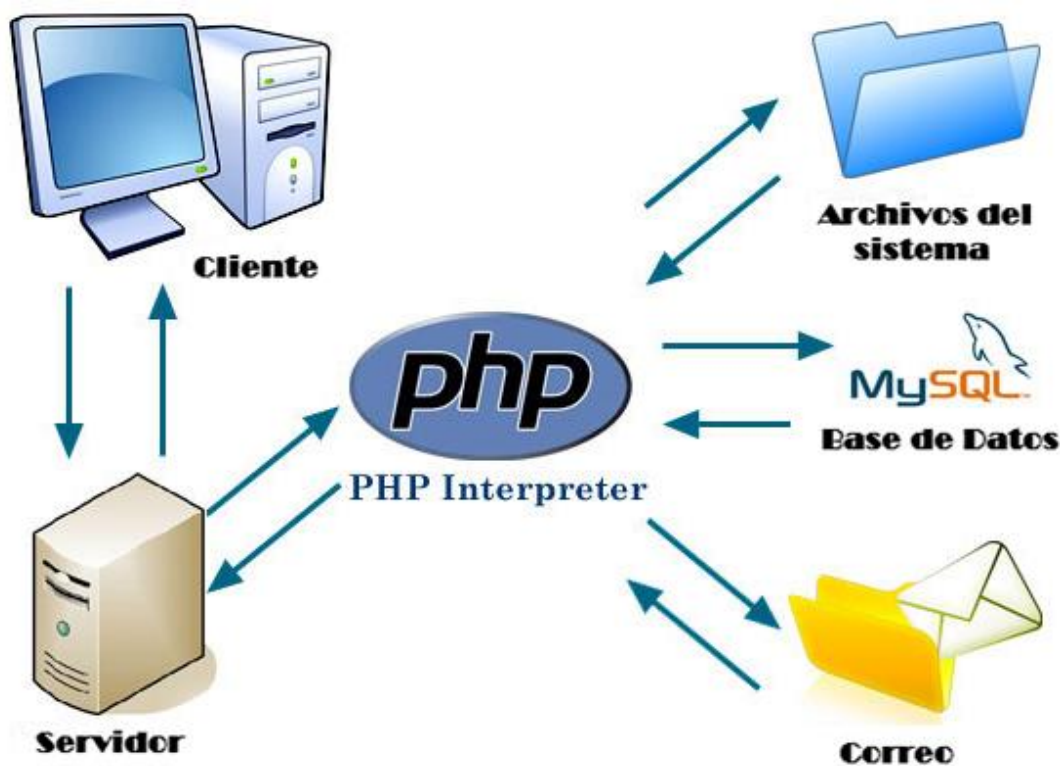


Figura 2.3.2 Diagrama de funcionamiento PHP.



“PHP brinda al desarrollador una gran libertad al momento de desarrollar aplicaciones con este lenguaje, algo que lo diferencia claramente de otros que son en sentido más restringidos, estructurados e inflexibles”¹¹.

Esta cualidad posiciona a PHP como un medio entre el desarrollador y la idea que persigue, es decir, el objetivo que desea plasmar a través del lenguaje. Un medio que como en otros casos, no es obstáculo, sino una ayuda, un puente que nos permite llevar acabo de manera sencilla tareas que en un principio parecen complejas.

PHP tiene una sintaxis simple, intuitiva y a la vez eficaz, que nos permite aplicar soluciones funcionales, con pocas líneas de código y en un tiempo muy reducido. Esto último es muy valorado por:

- Los desarrolladores, que pueden hacer más en menos tiempo.
- Las empresas, que pueden abarcar más proyectos simultáneamente.
- Los clientes, que pueden obtener respuestas inmediatas a sus pedidos.

PHP se encuentra disponible para los siguientes sistemas operativos:

- Mac OS.
- Microsoft Windows.
- Unix.
- Unix / HP-UX.
- Unix / Linux.
- Unix / Mac OS X.
- Unix / Open BSD.
- Unix / Solaris.

¹¹ Eslava, J., Muñoz, V. (2013). El nuevo PHP paso a paso. España: Bubok.



Las bases de datos que soporta PHP son las siguientes:

- DBase.
- Informix.
- Interbase / Firebird.
- Microsoft SQL Server.
- msql.
- MySQL.
- Oracle.
- PostgreSQL.
- SQLite.
- Sybase.

Características.

- Integración con varias bibliotecas externas, permite desde generar documentos en PDF hasta analizar código XML.
- Es un lenguaje Multiplataforma. Funciona en toda máquina que sea capaz de compilar su código.
- “XFORMS define una variable de los webforms tradicionales, los cuales permiten ser usados por una gran variedad de plataformas y navegadores”¹².
- Ofrece una solución simple y universal para las paginaciones dinámicas de la web de fácil programación.
- Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- El código escrito en PHP es invisible al navegador y al cliente ya que el servidor es el encargado de ejecutar el código y enviar el resultado HTML al navegador, esto hace que la programación sea segura y confiable.

¹² The PHP Group (2016). Manejo de XfForms. Recuperado el 28 de abril de 2016, de <http://php.net/manual/es/features.xforms.php>.



- Posee una amplia documentación en el sitio oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un archivo de ayuda.
- Soporta cookies HTTP de forma transparente; las cookies son un mecanismo por el que se almacenan datos en el navegador remoto y poder monitorear o identificar usuarios que regresan al sitio.
- Permite aplicar técnicas de programación orientada a objetos.
- No necesita definición de tipos de variables, aunque sus variables pueden ser evaluadas también por el tipo que estén manejando en tiempo de ejecución.
- Tiene manejo de excepciones.
- No tiene una metodología de programación, el desarrollador puede aplicar cualquiera, puede escribir el código ordenado, estructurado y manejable.
- Soporta ocho tipos de datos primitivos: boolean, integer, double, string, array, object, resource y null.

Ventajas.

- Lenguaje totalmente libre y abierto.
- Los entornos de desarrollo son de rápida y fácil configuración.
- Es un lenguaje Multiplataforma.
- Excelente soporte de acceso a base de datos.
- Es de fácil despliegue: paquetes totalmente autoinstalables que integran PHP.
- Cuenta con una gran comunidad para alguna contingencia que pudiera presentarse.
- Corre en casi cualquier plataforma utilizando el mismo código fuente, pudiendo ser compilando y ejecutado.
- El código fuente escrito es invisible al navegador y al cliente.
- Tiene la posibilidad de extender su potencial utilizando la enorme cantidad de extensiones que pueden ser utilizadas para mejorar el rendimiento en el desarrollo del trabajo que se esté realizando.



- Cuenta con documentación muy amplia en la red.
- Posee versatilidad y sencillez, es un lenguaje con sintaxis sencilla fácil de entender.
- El lenguaje es de uso común en la red, una gran cantidad de desarrolladores que utilizan el programa se encuentran constantemente trabajando para la mejora del motor de PHP, a medida que transcurre el tiempo y aumenta la versión del programa, se consigue mayor seguridad y estabilidad en el mismo.

Desventajas.

- Promueve creación de código desordenado y con un mantenimiento complejo.
- Todo el trabajo lo realiza el servidor. Por tanto puede ser ineficiente a medida que las solicitudes vayan en aumento.
- El sistema necesita contar con actualización de archivos del sistema operativo para evitar vulnerabilidades en la red.
- No cuenta con códigos prediseñados, esto en el momento de estar desarrollando es un pequeño inconveniente.
- No posee manejo adecuado de Unicode.
- Es difícil de optimizar.

2.4 Características, ventajas y desventajas de Apache



Figura 2.4.1 – Logo del Servidor Apache

Apache es un poderoso servidor web, en la figura 2.4.1 puede verse su logotipo, cuyo nombre proviene de la frase inglesa “a patchy server” es decir un servidor construido con código preexistente, piezas y parches de código y es completamente libre, ya que es un software Open Source y con licencia GPL.

“Es un esfuerzo por desarrollar y mantener un servidor HTTP de código abierto para sistemas operativos modernos, incluyendo UNIX y Windows. El objetivo de este proyecto es proporcionar un servidor seguro, eficiente y extensible que proporcione servicios HTTP en sincronización con los estándares actuales”¹³.

Desde hace años, más del 60% de los servidores web de Internet emplean Apache, es el ejemplo de software libre de mayor éxito.

“Su misión es crítica, ya que es el encargado de aceptar las peticiones de páginas (o recursos en general) que provienen de los visitantes que acceden a un sitio web y gestionar su entrega o denegación, de acuerdo a las políticas de seguridad establecida. El funcionamiento de apache se muestra en la figura 2.4.2”¹⁴.

¹³ The apache software foundation (2016). About Apache. Recuperado el 11 de mayo de 2016, de <http://www.apache.org>.

¹⁴ Apache HTTP Server (2015). Servidor Web. Recuperado el 30 de abril de 2016, de <http://culturacion.com/que-es-apache>.

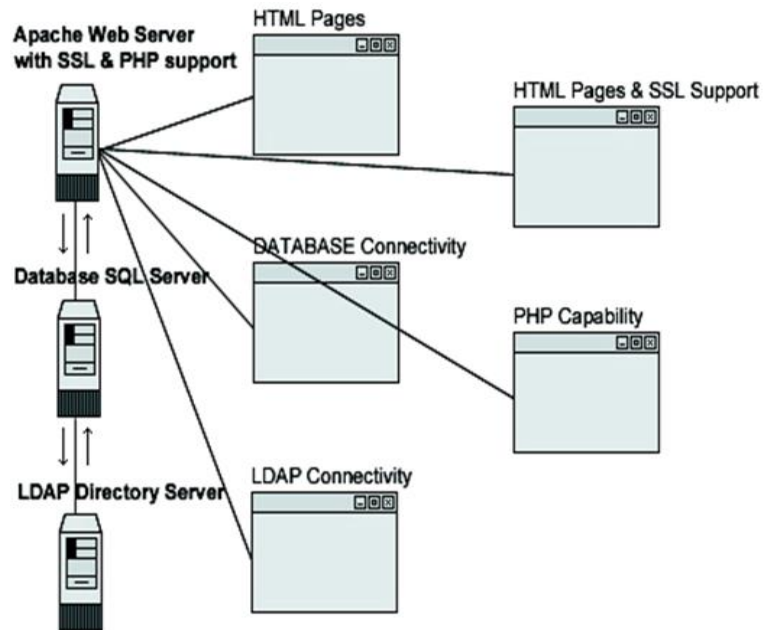


Figura 2.4.2 – Esquema Apache

Características.

- Soporte de seguridad SSL y TLS.
- Apache ofrece tecnología libre y de código abierto, otorgándole transparencia y dando la posibilidad de conocer qué es lo que realmente estamos instalando.
- Puede realizar autenticación de datos utilizando SGDB.
- Puede dar soporte a diferentes lenguajes como Perl, PHP, Python y tcl.r.
- Trabaja en conjunto con gran cantidad de Lenguajes de Programación interpretados como PHP (PHP Hypertext Pre-processor), Perl, soporte con CGI (Common Gateway Interface), Java, JSP (Java Server Pages) y otros lenguajes de script, es el complemento ideal para los sitios web dinámicos que vemos en la actualidad.
- “Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos”¹⁵.

¹⁵ Apache HTTP Server (2015). Características Servidor Apache. Recuperado el 30 de abril de 2014, de <https://es.opensuse.org/Apache>.



- Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.
- Es posible configurar y personalizar cada uno de los mensajes de error que se pueden producir por la utilización del servidor.
- Atender de manera eficiente, ya que puede recibir un gran número de peticiones HTTP, incluyendo una ejecución multitarea ya que pueden darse peticiones simultáneas.
- Restricciones de acceso a los ficheros que no se quieran ‘exponer’, gestión de autenticaciones de usuarios o filtrado de peticiones según el origen de éstas
- Manejar los errores por páginas no encontradas, informando al visitante y/o redirigiendo a páginas predeterminadas.
- Gestión de la información a transmitir en función de su formato e informar adecuadamente al navegador que está solicitando dicho recurso.
- Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para la solución de los mismos
- Módulos de autenticación: mod_access, mod_auth y mod_digest.
- Gestión de logs; es decir, almacena las peticiones recibidas, errores que se han producido y en general toda aquella información que puede ser registrada y analizada posteriormente para obtener las estadísticas de acceso al sitio web.
- Permite la configuración de mensajes de error personalizados y negociación de contenido.
- Contar con los archivos Log, en donde se registra una gran cantidad de información global del sistema, errores producidos en un determinado tiempo. Estos archivos son de gran importancia para los administradores de sistemas y pueden influenciar de alguna manera las políticas de seguridad.



Ventajas.

- Instalación/Configuración. Software de código abierto.
- “Coste. El servidor web Apache es completamente gratuito”¹⁶.
- Funcional y Soporte. Alta aceptación en la red y muy popular, esto hace que muchos programadores de todo el mundo contribuyen constantemente con mejoras, que están disponibles para cualquier persona que use el servidor web y que Apache se actualice constantemente.
- Multi-plataforma. Se puede instalar en muchos sistemas operativos, es compatible con Windows, Linux y MacOS.
- Rendimiento. Capacidad de manejar más de un millón de visitas/día.
- Soporte Apache Web Server cuenta con una gran comunidad de usuarios de soporte.
- Es altamente configurable.
- Al momento de la instalación permite elegir sobre que servidor web se va a correr.
- Es extensible.

Desventajas.

- Falta de integración.
- Posee formatos de configuración NO estándar.
- No posee un buen panel de configuración.
- Se requiere tener conocimientos.
- El servidor es aproximadamente un 20% más lento en su arranque debido a la sobrecarga que la resolución representa para el cargador (loader).
- El servidor es aproximadamente un 5% más lento en su ejecución bajo algunas plataformas porque el PIC (Position IndependentCode, posición de código independiente) necesita maniobras complicadas para direccionamiento dinámico, que no es necesariamente tan rápido como el direccionamiento absoluto.

¹⁶ Apache HTTP Server (2015). Características, instalación y configuración. Recuperado el 1 de mayo de 2016, de http://ldc.usb.ve/~emilio/Portafolio/Software/REDES3/G5b/Presenta_Site/caracteristicas_pyc.htm.



2.5 Características, ventajas y desventajas de JavaScript y CSS

JavaScript.

JavaScript es un lenguaje de programación interpretado, es decir, no es necesario compilar los programas para poder ejecutarlos. Los programas escritos en JavaScript se pueden probar en cualquier navegador sin necesidad de procesos intermedios: El navegador web descarga el código de una página web, y al encontrar en esta página código JavaScript lo ejecuta en la computadora del cliente.

JavaScript es utilizado principalmente para crear páginas web dinámicas: permite incorporar efectos como texto que aparece y desaparece, animaciones que se activan al pulsar botones y ventanas emergentes con mensajes para el usuario.

Debido a que JavaScript trabaja del lado del cliente, se puede utilizar para validar las entradas de los usuarios durante el envío de formularios y modificar el contenido de éstos.

JavaScript también se puede utilizar con CSS (Cascading Style Sheets) para hacer DHTML (DynamicHyper Text MarkupLanguage), el cual permite la aparición y desaparición de las páginas web. JavaScript se ejecuta solamente en la página web que se está visualizando en la ventana del navegador. Esto significa que las secuencias de comandos que se ejecutan en una página se detienen inmediatamente al momento en que el usuario deja de ver la página. La única excepción a esto son las APIs y cookies que pueden ser utilizadas por varias páginas para acumular y transmitir información entre ellos, del lado del cliente. Esto puede ocurrir incluso después del cierre de las páginas.

La principal característica de JavaScript es que está basado en scripts. Un script es una secuencia de órdenes que puede ser ejecutado por un cliente desde su navegador y puede ser visualizado el documento en el que está contenido.



Las sentencias de JavaScript se escriben con una etiqueta <SCRIPT> que informa al navegador que las líneas siguientes son el código JavaScript. El código es incrustado en HTML para ser interpretado y ejecutado por el navegador del cliente.

JavaScript se basa en el navegador para funcionar, aunque también puede utilizarse en otros contextos distintos de un navegador Web.

A pesar de su nombre, JavaScript no tiene relación directa con el lenguaje de programación Java. Java es un lenguaje de programación que puede ser utilizado para crear aplicaciones de todo tipo, mientras que JavaScript solo funciona dentro de una página HTML.

Características.

- JavaScript es un lenguaje sensible al uso de letras mayúsculas y minúsculas, esto se aplica a todos los aspectos del lenguaje: palabras clave, operadores, identificadores de variables, manejadores de eventos, propiedades de objetos, etc.
- Para JavaScript cualquier secuencia excesiva de caracteres de espacio en blanco es ignorada.
- Se pueden utilizar comentarios de una línea o comentarios de bloque. Se utiliza la secuencia de dos barras inclinadas consecutivas (//) para el comentario de una línea. En tanto que se utiliza la secuencia de una barra inclinada y un asterisco para abrir un comentario de bloque y un carácter asterisco seguido de una barra inclinada para cerrarlo.
- JavaScript permite trabajar con variables sin necesidad de declarar de qué tipo son. El tipo de dato es deducido del contexto en que es utilizada, esto es, en el momento en que se le asigna un valor.
- Las instrucciones se pueden terminar con un carácter de punto y coma (;) o con un ENTER. A pesar de que pueden utilizarse uno u otro, es conveniente que se use el punto y coma.



Ventajas.

- El lenguaje de scripts es seguro y fiable, puede ser filtrado debido a que debe ser interpretado.
- Utiliza poca memoria.
- Cuenta con cientos de aplicaciones disponibles para su uso.
- Representa una excelente solución para realizar la validación de datos de un formulario en el lado del cliente.
- El código JavaScript se ejecuta del lado del cliente por lo que el servidor no es solicitado más de lo debido.
- Es un lenguaje de programación muy liviano.
- Es muy útil para el desarrollo de páginas web dinámicas.
- Los scripts pueden desarrollarse en un período relativamente corto de tiempo, debido a que no requieren tiempo de compilación, aunado a esto es posible hacer correcciones de manera rápida.
- Sobresale radicalmente en la creación de efectos dinámicos tales como imágenes dinámicas y presentaciones de diapositivas.
- Las referencias entre los objetos se verifican en tiempo de ejecución.
- Por seguridad no es posible escribir en el disco del cliente.
- Compatible con la mayoría de los navegadores modernos.
- Es un lenguaje con amplia documentación en la web.
- Se puede utilizar para cambiar el aspecto de la pantalla en el dispositivo del usuario después que la página ha sido enviada por el servidor, mejorando la experiencia.

Desventajas.

- Por cuestiones de seguridad, los scripts tienen capacidades limitadas, por lo que es necesario usar JavaScript conjuntamente con otros lenguajes de programación, más evolucionados y seguros.



- Aunque el código esté protegido con las leyes de copyright, el código es visible y puede ser leído por cualquiera.
- Para poder ejecutarse, el código del script debe descargarse completamente. Esto representa un problema si los datos que el script utiliza son muchos.
- El lenguaje es muy intolerante con errores, basta un caracter adicional en una librería o declaración, para que 1000 ó 2000 líneas dejen de funcionar.
- Si el usuario desactiva el complemento de JavaScript en su navegador, la integridad de la página web puede estar totalmente comprometida.
- No es tan extenso en recursos como Flash.
- Las opciones de 3D son limitadas.
- Sistemas complejos pueden no funcionar tan bien como Flash.
- Puede introducir una cantidad enorme de fragmentos de código en los sitios web, comprometiendo la legibilidad y limpieza del código HTML.
- La tipología débil de JavaScript obliga a tener muchas precauciones, ya que puede conllevar ambigüedades potenciales, sobre todo en el uso de operadores.

CSS (Cascading Style Sheets).

Las Hojas de Estilo en Cascada permiten controlar el aspecto de una página web, al ser presentada en pantalla, de forma impresa o cómo será proporcionada la información a través de distintos dispositivos de lectura. La principal característica es la separación entre el contenido y el estilo del archivo HTML. Usando CSS es posible personalizar aspectos del documento HTML como la fuente que se desea en la página, el tamaño del texto, las columnas, si el texto se mostrará en negrita o cursiva, el fondo, los colores de los hipervínculos, los márgenes, la distribución de los objetos en la página, etc.

CSS también permite una descarga más rápida de páginas web, lo cual es muy útil en los equipos más antiguos y los módems.

La codificación de las reglas de estilo CSS se puede hacer en tres lugares:



Directamente en la etiqueta HTML. No es la mejor manera de aplicar las hojas de estilo, de hecho, esta forma es similar al modo usado en el código HTML; la diferencia radica en que con CSS el conjunto de propiedades que se pueden aplicar es mayor y que los estilos especificados mediante CSS tienen preferencia sobre los definidos por HTML.

Hoja de estilo interna. CSS está codificado en el principio de un documento HTML, dentro de la sección definida por las etiquetas `<head></ head>`. El inicio y fin de la hoja de estilos está determinado por las etiquetas `<style type = "text / css"></ style>`. Mediante este método se aplica la hoja de estilo solo a la página que la contiene. Aunque no es tan práctico como usar una CSS externa, resulta útil cuando en el sitio se tienen pocas páginas que usan un formato distinto al resto.

Hoja de estilos externa. Este es un archivo aparte con la extensión `.css` que sirve como referencia para múltiples páginas HTML, cada archivo HTML contiene un enlace que señala a los navegadores donde buscar los estilos. Usando CSS externas es posible cambiar los estilos sin tocar el código HTML y cada CSS puede ser compartida por varias páginas, lo que hace que este método sea el más recomendable porque permite mayor flexibilidad.

Por las leyes de cascada de las CSS, en caso de existir una misma regla (una en la hoja externa y otra en la hoja interna) con distintos valores, tiene preferencia la definida en la hoja interna.

Ventajas.

- Ofrece herramientas de composición más potentes que las implementadas con HTML.
- CSS permite posicionar con precisión cualquier elemento.
- El lenguaje de CSS, es relativamente sencillo y fácil de aprender.
- Los documentos HTML que usan CSS, suelen ser más compactos.



- Las CSS pueden combinarse formando una cascada de estilos con la información de cada una.
- Es posible combinar JavaScript con CSS para obtener efectos dinámicos en las páginas.
- Se puede especificar la CSS de acuerdo al navegador y al dispositivo que se esté utilizando.
- El código HTML es más legible, al tener las CSS en documentos externos.
- El mantenimiento del sitio se facilita debido a que si se requiere una modificación a la presentación, sólo es necesario modificar el archivo que contiene la CSS.

Desventajas.

- Las CSS tienen soporte irregular por parte de los navegadores: Ciertas propiedades no funcionan en todos; también puede haber discrepancias en un mismo navegador dependiendo del Sistema Operativo o de la versión.
- Si hay problemas de compatibilidad con el navegador, es posible que no se visualice la página con la CSS diseñada sino con el formato predeterminado por el navegador.
- Las propiedades de las CSS relacionadas con la posición o visibilidad pueden provocar que el contenido de la página sea inaccesible, parcial o totalmente; y es ilegible, desde ciertos navegadores, si existe incompatibilidad.





CAPÍTULO III

“ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA”



3.1 Análisis del Problema

El problema inicial radica en que la elaboración para la programación de asesorías se realiza de manera obsoleta y a la vez se ha quedado rezagada en el uso y aprovechamiento de la tecnología. El proceso manual con el que se realiza este procedimiento da como resultado múltiples fallas que se enunciarán a continuación:

- Duplicidad de información.
- Cambios de horarios.
- Traspapelado de información.
- Olvido de asignación de grupos.
- Pérdida de información.
- Excesivo consumo de tiempo.
- Poca o nula comunicación entre el personal asignado al proceso.
- Poca eficiencia.

Basado en el análisis de estas fallas se pretende generar una herramienta que sea capaz de involucrar a la Institución Educativa en los avances tecnológicos, principalmente en aquellos que le permitan estar a la vanguardia. Otro de los objetivos es ayudar a realizar las labores del personal de la manera más eficiente, rápida y segura.

En la figura 3.1.1 se muestra el proceso manual mediante el cual se lleva a cabo la programación de asesorías.

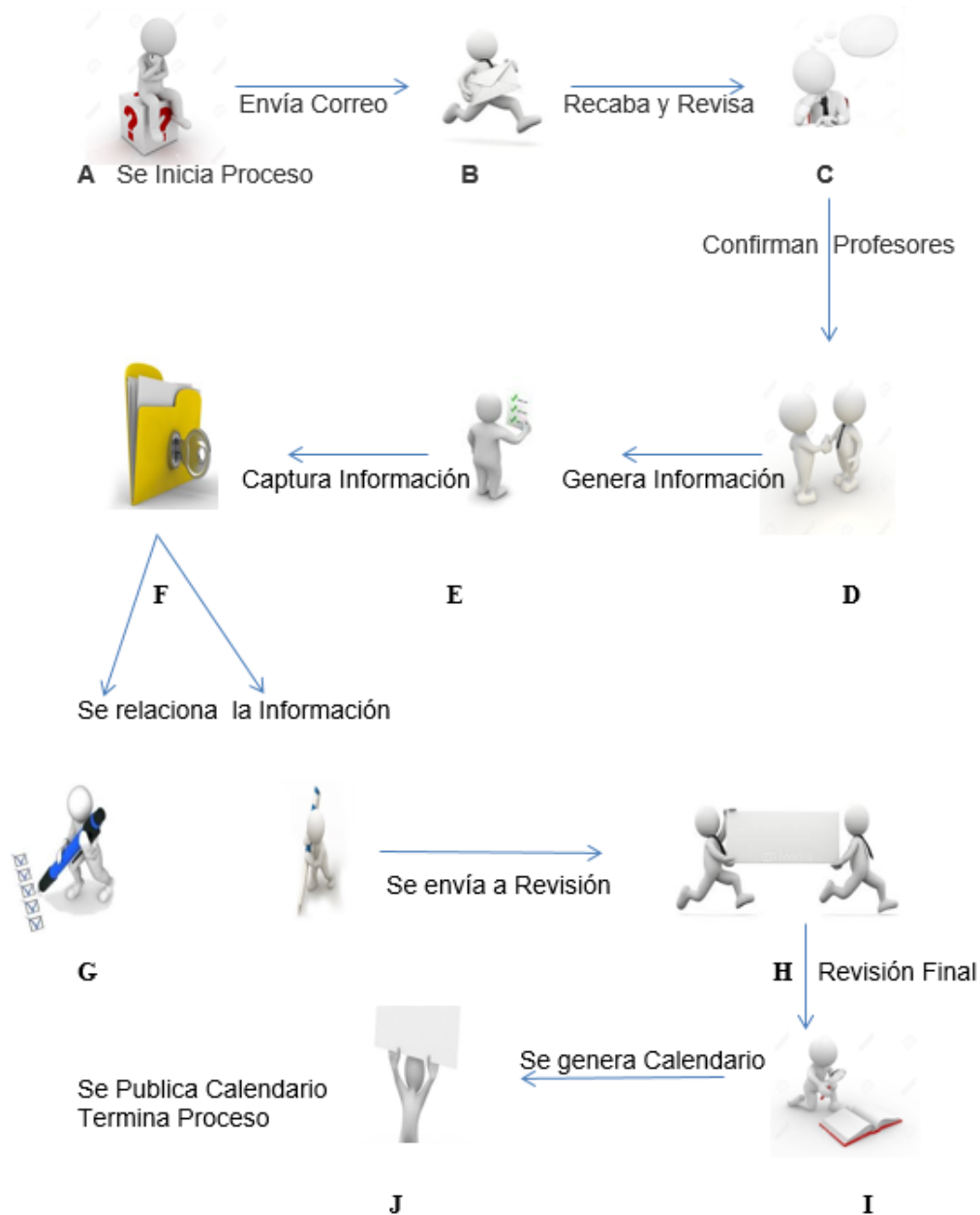


Figura 3.1.1 - Programación de Asesorías Proceso Manual

A.- El proceso inicia con una planeación anterior, ya que se debe de considerar el número aproximado de alumnos a inscribirse en esta modalidad.

B.- Es enviado un correo para confirmar la participación del profesor en el semestre venidero, así como la materia que será impartida por él.



C.- El profesor analiza sus opciones y procede a informar a la Institución Educativa su decisión.

D.- Confirmado el registro de los profesores y las materias, se procede a enviar la información al personal encargado de realizar las tablas con toda la información requerida por la Institución

E.- Se procede a la generación de las tablas de Profesores, Grupos con toda la información detallada para la generación de los horarios de asesorías. De aquí depende que la información proporcionada haya sido revisada a detalle, debido a que es el punto de inicio y en muchas ocasiones la omisión de errores por descuido ocasionaría un retraso en la generación del calendario y, en un caso más fatalista, la validación de toda la información trayendo en consecuencia un derroche innecesario de tiempo.

F.- La información es procesada en tablas de Excel. La información como tal no tiene mayor seguridad y la veracidad de la información proporcionada en ellas no hay forma de validarla.

G.- Con la información proporcionada al personal de la Institución Educativa estas se encargan de relacionar las tablas y a su vez generar los horarios para las asesorías. En este punto del proceso se tiene un mayor riesgo de cometer errores, ya que por un lado se tiene a 2 personas realizando la misma tarea, no hay un orden ni secuencia en el llenado de los formatos. Otra situación para tener en cuenta es que si la información generada trae errores y no se está haciendo una validación correcta ya sea por descuido o error humano es el mismo caso que con las tablas de Excel, se tendría que validar toda la información de nueva cuenta.

H.- El calendario previo es enviado al personal encargado de realizar la revisión final.

I.- El encargado de la revisión final debe de realizar una tarea minuciosa ya que el documento puede acarrear errores tales como duplicidad de información, cambios de



horarios o grupos, faltas de ortografía, (errores humanos, etc.), la tarea principal radica en revisar todas las incidencias que pudieran darse.

J.- Por último ya con el V.ºB.º de la persona encargada de la revisión final.El calendario será publicado en el portal de la Institución Educativa. Si en este paso se detecta algún error en el calendario, el proceso tendría que realizarse nuevamente a totalidad.

En la figura 3.1.2 se muestra el proceso mediante el cual se lleva a cabo la programación de asesorías de manera automatizada.

A – E



F
Se Relaciona Información



G



Se Genera Calendario



H
Se Publica Calendario
Termina Proceso

Figura 3.1.2 - Programación de Asesorías mediante Proceso Automatizado



De la figura 3.1.1 tenemos que los pasos realizados A – E, en el Proceso Automatizado figura 3.1.2, son ejecutados en forma directa en el Sistema, éste contiene la información de forma correcta, con lo cual el administrador sólo debe encargarse de generar la asignación de grupos, salones, profesores, horarios, etc.

G.-El administrador utilizando el aplicativo se dedica a generar los horarios en base a la información contenida en el Sistema.

H.- Por último ya con el V.ºB.º del administrador, el calendario será publicado en el portal de la Institución Educativa.

Las ventajas que se aprecian utilizando el Sistema se enumeran a continuación:

- “Optimización de recursos humanos”¹⁷.
- Eliminación de duplicidad de información.
- Reducción de tiempo para la generación del calendario.
- Menor personal involucrado en el proceso.
- Minimizar los errores humanos.
- Mayor seguridad en el control de la información.
- Disponibilidad de la información.
- Mayor eficiencia.
- “Obtener solo la información necesaria”¹⁸.
- Maneja grandes volúmenes de datos.

3.2 Recopilación y análisis de la información.

La Coordinación de Educación Abierta y a Distancia, realiza la programación de asesorías manualmente en un documento de Excel en el cual capturan y organizan los datos de las asesorías tales como:

¹⁷ SlideShare (2016). Beneficios de un sistema de Información. Recuperado el 6 de mayo de 2016, de <http://es.slideshare.net/sylem123/beneficios-de-los-sistemas-de-informacion>.

¹⁸ Con la tecnología de Blogger (2015). Sistema de Información Computarizado. Recuperado el 6 de mayo de 2016, de <http://sistemadeinformacioncomputarizado.blogspot.mx>.



Capítulo III “Análisis y planteamiento del problema”



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
362	QUINTO SEMESTRE D.F.-CHIMALHUACÁN-QUERÉTARO-ORIZTLÁN-TOLUCA-OAXACA															
363																
364																
365	9005	1411-ED	Problemática urbana	López Escalante Gustavo Alberto	Edif. C	26-ene 9:00 a 3:40	Aula virtual	Edif. C	14-feb 16:00 a 17:00	09-mar 9:00 a 10:00	13-abr 9:00 a 10:00	11-may 9:00 a 10:00	A distancia	27-may	03-jun	
366								Salón 1	Edif. C-s1							
367	9005	1511-ED	Desarrollo Regional	Vázquez Hernández Silvia Andrea		26-ene 3:40 a 10:20			14-feb 17:00 a 18:00	09-mar 10:00 a 11:00	13-abr 10:00 a 11:00	11-may 10:00 a 11:00	A distancia	28-may	04-jun	
368									Edif. C-s1							
369																
370	9005	1512-ED	Salud Pública	Zamora Carrillo Edgar		26-ene 10:20 a 11:00			14-feb 18:00 a 19:00	09-mar 11:00 a 12:00	13-abr 11:00 a 12:00	11-may 11:00 a 12:00	A distancia	29-may	05-jun	
371									Edif. C-s1							
372																
373	9005	1616-ED	Evaluación de proyectos sociales	Hernández Ibarra José Agustín		26-ene 11:00 a 11:40			14-feb 19:00 a 20:00	09-mar 12:00 a 13:00	13-abr 12:00 a 13:00	11-may 12:00 a 13:00	A distancia	27-may	03-jun	
374									Edif. C-s1							
375																
376	9005	1517-ED	Práctica comunitaria II	Contreras María de Jesús		26-ene 11:40 a 12:20			07-feb 18:00 a 18:00	16-mar 9:00 a 11:00	20-abr 9:00 a 11:00	18-may 9:00 a 11:00	A distancia	30-may	06-jun	
377									Edif. C-s1							
378																
379	9015	1517-ED	Práctica comunitaria II	León Silva Agustina	Aula virtual Sala VC	29-ene 16:00 a 18:00	Aula virtual Aula 3 Posgrado		07-feb 18:00 a 20:00	16-mar 9:00 a 11:00	20-abr 9:00 a 11:00	18-may 9:00 a 11:00				
380									Edif. C-s1							
381																

Figura 3.2.3 – Asesoría de grupo 9005 de quinto semestre.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
476	OCTAVO SEMESTRE CHIMALHUACÁN-OAXACA-TABASCO															
477	9128	1414-ED	Psicología del desarrollo humano	Rodríguez Rodríguez María de la Luz	Aula virtual Sala VC	31-ene 16:00 a 17:00	Aula virtual Aula 8 Posgrado		15-feb 16:00 a 17:00	09-mar 9:00 a 10:00	13-abr 9:00 a 10:00	11-may 9:00 a 10:00	A distancia	27-may	03-jun	
478									Edif. C-s2							
479																
480	9128	1610-ED	Situación Jurídica de la familia	De Villa López Ernesto		31-ene 17:00 a 18:00			15-feb 17:00 a 18:00	09-mar 10:00 a 11:00	13-abr 10:00 a 11:00	11-may 10:00 a 11:00	A distancia	28-may	04-jun	
481									Edif. C-s2							
482																
483	9128	1614-ED	Comunicación social	Cortes Osorno María Guadalupe		31-ene 18:00 a 19:00			15-feb 18:00 a 19:00	09-mar 11:00 a 12:00	13-abr 11:00 a 12:00	11-may 11:00 a 12:00	A distancia	29-may	05-jun	
484									Edif. C-s2							
485																
486	9128	1817-ED	Práctica de especialización I	Contreras Tébar Amelia Margarita		31-ene 19:00 a 20:00			15-feb 19:00 a 20:00	16-mar 9:00 a 11:00	20-abr 9:00 a 11:00	18-may 9:00 a 11:00	A distancia	30-may	06-jun	
487									Edif. C-s2							
488																
489	NOVENO SEMESTRE D.F.-OAXACA															
490																
491	9009	1513-ED	Identidad y cultura	Salazar Hernández Enriqueta	Aula Virtual Edif C	31-ene 17:00 a 18:00	Aula virtual Aula 3 Posgrado		15-feb 17:00 a 18:00	09-mar 9:00 a 10:00	13-abr 9:00 a 10:00	11-may 9:00 a 10:00	A distancia	27-may	03-jun	
492									Edif. C-s1							
493																
494	9009	1615-ED	Análisis institucional	Contreras Tébar Amelia Margarita		31-ene 18:00 a 19:00			15-feb 18:00 a 19:00	09-mar 10:00 a 11:00	13-abr 10:00 a 11:00	11-may 10:00 a 11:00	A distancia	28-may	04-jun	
495									Edif. C-s1							
496																
497	9009	1917-ED	Práctica de especialización II	Tellez Martínez María Elena		31-ene 19:00 a 20:00			15-feb 19:00 a 20:00	09-mar 11:00 a 12:00	13-abr 11:00 a 12:00	11-may 11:00 a 12:00	A distancia	29-may	05-jun	
498									Edif. C-s2							
499																

Figura 3.2.4 – Asesoría de grupos de octavo y noveno semestre.

La complejidad para realizar este documento se incrementa cuando la demanda de alumnos es mayor ya que se deben crear más grupos, lo que aumenta la probabilidad de cometer errores, adicional a esto, se debe tener en cuenta que al tratarse de una institución que está conformada por alumnos que toman clases de forma presencial y a distancia, se deben considerar factores para la asignación de aulas, pues para las asesorías a distancia, el aula debe tener acceso a la red inalámbrica, clavija de energía



eléctrica, y que esté ubicada en un área apropiada para permitir que la calidad del audio sea óptima.

La cantidad de grupos en cada semestre es variable, en la tabla 3.2.1 tenemos la relación de grupos en el semestre en curso y la cantidad de asesorías que se deben programar, donde se observa que los grupos varían por semestre.

Semestre	Grupos	Asesorías a programar por semestre
1º	9001 ,9121, 9281, 9321	196
2º	9002, 9012, 9122, 9152	196
3º	9003, 9123, 9153, 9263	196
4º	9004, 9014, 9024, 9124, 9224, 9264	210
5º	9005, 9015	70
6º	9006, 9016, 9156, 9186	140
7º	9007, 9127	70
8º	9008, 9018, 9128	84
9º	9009	21
	Total	1183

Tabla 3.2.1 –Relación de grupos y total de asesorías a programar por semestre.

En este caso se tiene un total de 30 grupos, y cada grupo tiene alrededor de 6 materias por semestre, dando un total de 180 registros, adicional a esto cada materia debe cumplir 7 asesorías obligatorias durante el ciclo escolar, dando como resultado 1183 asesorías que deben ser adaptadas al calendario del ciclo escolar.

Cada semestre se efectúa un proceso exhaustivo y desgastante para organizar las asesorías, los puntos a considerar para llevar a cabo este procedimiento son:



- Profesores, se requiere tener confirmación de la planta académica aprobada por la institución.
- Aulas, considerando que también varían los espacios asignados por semestres es importante tener bien definido los espacios asignados.
- Grupos, es necesario conocer la cantidad de grupos creados para el nuevo semestre.
- Fechas, se deben revisar el calendario escolar para delimitar fecha de inicio y fin de semestre.

Este exhaustivo procedimiento se lleva alrededor de 4 a 6 semanas, y se empieza a trabajar dos meses antes de que dé inicio el siguiente ciclo escolar, prediciendo la cantidad de grupos que se abrirán el siguiente ciclo escolar y los profesores que impartirán las asignaturas.

El procedimiento manual para generar la programación de horarios, tiene un sinnúmero de contrariedades, algunos ejemplos son:

- Pronósticos erróneos, difícilmente se puede hacer un pronóstico acertado, la mayoría de las ocasiones es necesario realizar modificaciones, lo que conlleva doble esfuerzo para el personal implicado en el proceso.
- Cambios de última hora, habitualmente se modifica la programación ya sea para agregar grupos, cambio de profesores, cambio de aulas, etc. Esto afecta a las personas que no revisan constantemente el documento y se quedan con la primera versión de éste.
- Pérdida de tiempo, para la elaboración de horarios intervienen varias personas de distintas áreas, como, planta académica, servicios estudiantiles y el área de operación de la plataforma, que deben trabajar ordenadamente y en conjunto, esto implica dedicarle varias horas del día por cada parte y al final hacer la revisión detallada.



Capítulo III “Análisis y planteamiento del problema”



- Posibles errores, al trabajar distintas personas en la elaboración de horarios, aumenta la probabilidad de cometer errores, desde fechas, horarios, profesores hasta grupos.
- Modificaciones constantes, una vez que se tiene listo el documento en Excel, se copia a formato Word para aplicarle la plantilla institucional, se convierte a formato PDF y finalmente, se sube a la plataforma escolar. Cuando se modifica el documento se tiene que hacer nuevamente todo el proceso y así sucesivamente por cada modificación.
- Tensión en las personas involucradas, al tener el documento inicial con los pronósticos, se tiene que realizar modificaciones con premura ya que cuando se definen los grupos, aulas y profesores, es muy próximo al inicio del semestre, esto ocasiona tensión, estrés y aumenta la probabilidad de que se cometan errores.

En la figura 3.2.5 se muestra un error de repetición de salón y horario para distintos semestres.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
OCTAVO SEMESTRE CHIMALHUACÁN-OAXACA-TABASCO																
476																
477	9128	1414-ED	Psicología del desarrollo Humano	Rodríguez Rodríguez María de la Luz	Aula virtual Sala VC	31enero 16:00 a 17:00		Aula virtual Aula 8 Posgrado	15-feb 16:00 a 17:00	09-mar 9:00 a 10:00	13-abr 9:00 a 10:00	11-may 9:00 a 10:00		A distancia	27-may	03-jun
478									Edif. C-s2							
479																
480	9128	1610-ED	Situación Jurídica de la familia	De Villa López Ernesto		31enero 17:00 a 18:00			15-feb 17:00 a 18:00	09-mar 10:00 a 11:00	13-abr 10:00 a 11:00	11-may 10:00 a 11:00		A distancia	28-may	04-jun
481									Edif. C-s2							
482																
483	9128	1614-ED	Comunicación social	Cones Osorno María Guadalupe		31enero 18:00 a 19:00			15-feb 18:00 a 19:00	09-mar 11:00 a 12:00	13-abr 11:00 a 12:00	11-may 11:00 a 12:00		A distancia	29-may	05-jun
484									Edif. C-s2							
485																
486	9128	1817-ED	Práctica de especialización I	Contreras Tébar Amelía Margarita		31enero 19:00 a 20:00			15-feb 19:00 a 20:00	16-mar 9:00 a 11:00	20-abr 9:00 a 11:00	18-may 9:00 a 11:00		A distancia	30-may	06-jun
487									Edif. C-s2							
488																
489																
NOVENO SEMESTRE D.F.-OAXACA																
490																
491	9009	1513-ED	Identidad y cultura	Salazar Hernández Enriqueta	Aula Virtual Edif C	31enero 17:00 a 18:00		Aula virtual Aula 3 Posgrado	15-feb 17:00 a 18:00	09-mar 9:00 a 10:00	13-abr 9:00 a 10:00	11-may 9:00 a 10:00		A distancia	27-may	03-jun
492									Edif. C-s1							
493																
494	9009	1615-ED	Análisis institucional	Contreras Tébar Amelía Margarita		31enero 18:00 a 19:00			15-feb 18:00 a 19:00	09-mar 10:00 a 11:00	13-abr 10:00 a 11:00	11-may 10:00 a 11:00		A distancia	28-may	04-jun
495									Edif. C-s1							
496																
497	9009	1917-ED	Práctica de especialización II	Tellez Martínez María Elena		31enero 19:00 a 20:00			15-feb 19:00 a 20:00	09-mar 11:00 a 12:00	13-abr 11:00 a 12:00	11-may 11:00 a 12:00		A distancia	29-may	05-jun
498									Edif. C-s2							
499																

Figura 3.2.5 – Error en la programación.



Estos errores provocan descontentos por parte de profesores y alumnos pues si se quedaron con la primera versión del documento y no se percatan que sufrió modificaciones, se generan afectaciones en el desarrollo de las sesiones.

Al ver toda la problemática que genera continuar realizando manualmente la programación, decidimos desarrollar un programa que brinde mejoras en:

- Ahorro de tiempo y esfuerzo al personal implicado, con el sistema se disminuirá considerablemente el tiempo.
- Evitar molestas y disgustos tanto a profesores y alumnos.
- Minimizar errores de traslape de horarios, salones o grupos, el desarrollo tendrá reglas para evitar traslapes.
- Simplificar búsqueda de horarios, se desplegarán horarios por grupo, horario, etc.
- Automatizar todo el proceso, una vez generada la programación se pasará automáticamente a formato PDF para evitar una posible alteración posterior a su publicación.

3.3 Requerimientos generales y particulares del problema

“Los requerimientos especifican qué es lo que el sistema debe hacer (sus funciones) y sus propiedades esenciales y deseables. La captura de los requerimientos tiene como objetivo principal la comprensión de lo que los clientes y los usuarios esperan que haga el sistema. Un requerimiento expresa el propósito del sistema sin considerar como se va a implantar. En otras palabras, los requerimientos identifican el qué del sistema, mientras que el diseño establece el cómo del sistema”¹⁹.

Los requerimientos cumplen un papel primordial en el proceso de producción de software, ya que enfoca un área fundamental: La definición de lo que se desea producir y marcan el punto de partida para actividades como la planeación. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades,

¹⁹ UAM Cuajimalpa (2015). Análisis de requerimiento. Recuperado el 8 de mayo de 2016, de http://web.cua.uam.mx/publicaciones/Notas_Analisis_Requerimiento.pdf.



en forma consistente y compacta, el comportamiento del sistema; de esta manera, se pretende minimizar los problemas relacionados al desarrollo de sistemas y verificar si se alcanzaron o no los objetivos establecidos en el proyecto ya que éstos son un reflejo detallado de las necesidades de los usuarios del sistema y es contra lo que se va a estar validando si se cumplen las metas trazadas.

Se sugiere un conjunto de principios directrices para la ingeniería de requerimientos como se muestra en la figura 3.3.1.



Figura 3.3.1 – Directrices para recolección de requerimientos.

Los principales beneficios que se obtienen de una correcta especificación de requerimientos son:

- Permite gestionar las necesidades del proyecto en forma estructurada.
- Mejora la capacidad de predecir cronogramas de proyectos, así como sus resultados.
- Asegura al cliente que el desarrollador entiende sus necesidades y que está respondiendo a ellas.
- Una oportunidad para una retroalimentación entre el cliente y el proveedor.
- Un método para que el cliente y el proveedor, o desarrollador, puedan identificar tempranamente problemas y malentendidos mientras los costos son relativamente bajos.
- Una base para la calificación y calidad del sistema, para establecer que éste cumple con las necesidades del cliente.



- Protección para el proveedor, dando una línea base para las capacidades del sistema y una base que determine cuando la construcción del sistema está completa.
- Soporte para el desarrollador en la planificación, diseño y desarrollo del programa.
- Ayuda en la evaluación de los efectos causados por los cambios en los requerimientos.

Características de un requerimiento

Los requerimientos bien formulados deben satisfacer varias características. Si no lo hacen, deben ser reformulados hasta hacerlo.

- **Necesario:** Lo que pida un requisito debe ser necesario para el producto.
- **No ambiguo:** El texto debe ser claro, preciso y tener una única interpretación posible.
- **Conciso:** Debe redactarse en un lenguaje comprensible por los inversores en lugar de uno de tipo técnico y especializado, aunque aún así debe referenciar los aspectos importantes.
- **Consistente:** Ningún requisito debe entrar en conflicto con otro requisito diferente, ni con parte de otro. Asimismo, el lenguaje empleado entre los distintos requisitos debe ser consistente también.
- **Completo:** Los requisitos deben contener en sí mismos toda la información necesaria, y no remitir a otras fuentes externas que los expliquen con más detalle.
- **Alcanzable:** Un requisito debe ser un objetivo realista, posible de ser alcanzado con el dinero, el tiempo y los recursos disponibles.
- **Verificable:** Se debe poder verificar con absoluta certeza, si el requisito fue satisfecho o no. Esta verificación puede lograrse mediante inspección, análisis, demostración o testeo.

Estas características suelen ser subjetivas, es decir, no pueden ser calculadas de forma automática por ningún sistema. Por ello, se tiende a utilizar métricas o



indicadores que sí pueden ser calculados de forma automática y que, de algún modo, pueden contribuir a ponderar las anteriores características.

Dificultades para definir los requerimientos

Durante la etapa de especificación de requerimientos se pueden presentar muchos inconvenientes los cuales son importantes de identificar y prevenir, a continuación se presenta un listado con los problemas más comunes en este proceso:

- Los requerimientos no son obvios y vienen de muchas fuentes.
- Son difíciles de expresar en palabras (el lenguaje es ambiguo).
- Existen muchos tipos de requerimientos y diferentes niveles de detalle.
- La cantidad de requerimientos en un proyecto puede ser difícil de manejar.
- Nunca son iguales. Algunos son más difíciles, más riesgosos, más importantes o más estables que otros.
- Los requerimientos están relacionados unos con otros, y a su vez se relacionan con otras partes del proceso.
- Cada requerimiento tiene propiedades únicas y abarcan áreas funcionales específicas.
- Un requerimiento puede cambiar a lo largo del ciclo de desarrollo.
- Son difíciles de cuantificar, ya que cada conjunto de requerimientos es particular para cada proyecto.

Tipos de requerimientos

La especificación debe contener los requerimientos del sistema, la IEEE-830, 1998 divide los requerimientos en funcionales y no funcionales, a continuación se describe en qué consiste cada uno de estos grupos de requerimientos.

- **Requerimientos Funcionales.**
“Son declaraciones de los servicios que proveerá el sistema, de la manera en que éste reaccionará a entradas particulares. En algunos casos, los requerimientos funcionales de los sistemas también declaran explícitamente lo que el sistema no



debe hacer. Los requerimientos funcionales de un sistema describen la funcionalidad o los servicios que se espera que éste provea. Estos dependen del tipo de software y del sistema que se desarrolle y de los posibles usuarios del software. Cuando se expresan como requerimientos del usuario, habitualmente se describen de forma general mientras que los requerimientos funcionales del sistema describen con detalle la función de éste, sus entradas y salidas, excepciones, etc.”²⁰.

- **Requerimientos No Funcionales.**

Describen una restricción sobre el sistema que limita nuestras elecciones en la construcción de una solución al problema. Restringen los servicios o funciones ofrecidas por el sistema. Incluyen restricciones de tiempo, el tipo de proceso de desarrollo a utilizar, fiabilidad, tiempo de respuesta, capacidad de almacenamiento. Los requerimientos no funcionales ponen límites y restricciones al sistema. Éstos se dividen en tres grupos generales:

- **Requerimientos del Producto:** Especifican el comportamiento del producto.
Ejemplos: Rapidez de la ejecución, capacidad de memoria, fiabilidad, etc.
- **Requerimientos Organizacionales:** Derivan de políticas y procedimientos existentes en la organización del cliente y del desarrollador.
Ejemplos: Estándares de procesos, métodos de diseño, lenguajes de programación, métodos de entrega, etc.
- **Requerimientos Externos:** Se derivan de factores externos al sistema y de sus procesos de desarrollo.
Ejemplos: Requisitos de interoperatividad, legislativos, éticos, etc.

Requerimientos Generales

- Se requiere que la información se guarde en una base de datos robusta y confiable.
- Que se ejecute en un ambiente cliente - servidor.

²⁰ Con la tecnología de Blogger (2015). Tipos de requerimiento. Recuperado el 3 de mayo de 2016, de <http://fundamentos-ing-requisitos.blogspot.mx/2012/02/normal-0-21-false-false-false-es-ve-x.html>.



- La base de datos debe estar alojada en el servidor central.
- Esta base de datos solo podrá ser administrada por el personal autorizado.
- La información que contenga debe ser verdadera, estar al día y debe ser necesaria para realizar los procesos que dependen de ella.
- El sistema será desarrollado utilizando software libre para que no se dependa de licencias de terceros.
- Se deben seguir los lineamientos establecidos por la Institución Educativa.
- La aplicación deberá contar con una interfaz gráfica que sea amigable, cómoda, eficiente, fácil de manejar, con elementos como ventanas, menús, botones y listas desplegables.
- Los equipos deben de estar habilitados con Windows, desde la versión XP o posterior, es por eso que resulta esencial utilizar el ambiente de pantallas, botones, iconos, etc. elementos característico de este sistema operativo.
- Se debe de contar con mouse y teclado que faciliten el manejo del entorno grafico del sistema, por ser los que se encuentran disponibles para los usuarios.
- Las ventanas de la aplicación deben tener un balance de colores adecuados, ya que el cliente no desea que el aspecto visual de dichas ventanas pudiera significar un distractor para los usuarios finales.
- Se desea controlar el acceso mediante un usuario y una contraseña, los cuales serán proporcionados y gestionados únicamente por el Administrador de la gestión de horarios.
- Soportado por los principales navegadores web.
- Capacidad para dar servicio a múltiples usuarios.
- Acceso vía web sin necesidad de instalar la aplicación en cada una de las terminales.

Requerimientos Particulares

- Proporcionar información oportuna, íntegra y confiable a los alumnos y profesores que se encuentran en esta modalidad.



- Satisfacer las necesidades de reducción de tiempo y errores logísticos por parte de la coordinación administrativa en la realización de horarios para la modalidad de educación abierta y a distancia.
- Contar con un módulo de seguridad para el control de accesos al sistema.
- Permitir alta, baja y actualización de horarios por grupo y materia.
- Realizar alta y baja de los espacios disponibles para las sesiones.
- Realizar alta y baja de los usuarios que tendrán acceso a la aplicación.
- Cuando se realice una alta o baja de horario, salones o usuarios se muestre una pantalla con el mensaje de que la acción solicitada se realizó correctamente en las bases de datos.
- Si se requiere realizar un alta por grupo, salón o usuario y si éste ya existe en las bases de datos que se muestre en pantalla un mensaje de error por duplicidad de información.
- Realizar búsquedas de horarios por grupo, materia, profesor semestre y día.
- Al realizarse una búsqueda de horario se debe de informar los siguientes datos:
 - Grupo
 - Clave
 - Asignatura
 - Profesor
 - Fecha y hora de las sesiones
- Se genere un archivo PDF con los horarios para que se pueda subir a la página oficial de la Institución Educativa.
- Se muestre un mensaje de error cuando dos o más sesiones se intentan dar de alta con la misma fecha.
- Se muestre un error informativo de que la hora inicial debe ser menor a la hora final al realizar alta de un horario.
- Se genere un cuadro emergente cuando la información que se solicita insertar en la base de datos es errónea o hace falta que se completen campos del formulario.
- Se genere un cuadro emergente para confirmación de que se va a realizar una baja.



- Acceso a los catálogos de profesores y materias con los que se cuenta.

3.4 Planteamiento de la solución.

Es fundamental llevar a cabo un análisis de las herramientas existentes en el mercado para poder elegir la mejor tecnología que permita cumplir las expectativas del usuario final y se ajuste a los requerimientos del sistema que se desea desarrollar. Algunos factores que se deben considerar son el tamaño y el tipo de aplicación, el número de usuarios concurrentes, el equipo y software disponible y el tiempo de entrega del sistema. En base a estos factores se contrastan algunas opciones de solución y posteriormente se seleccionará la más viable.

En el desarrollo web, técnicamente las tecnologías se dividen en dos: front-end y back-end.

- En el back-end se encuentran los lenguajes que se encargan de interactuar con la base de datos, verificar el manejo de sesiones de usuarios, montar la página en un servidor y desde éste proporcionar todas las vistas que el front-end crea.
- Front-end, son aquellos lenguajes que se encargan de la presentación del sitio web, es decir, aquellas aplicaciones que correrán en el navegador del cliente y que tienen como principal función presentar al usuario el sitio web.

PostgreSQL (vea figura 3.4.1).



Figure 3.4.1 – Logo PostgreSQL.

“Es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Utiliza un modelo cliente-servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema seguirá funcionando”²¹.

MySQL (vea la figura 3.4.2).



Figura 3.4.2 – Logo MySQL.

“Es muy conocido y ampliamente usado por su simplicidad y notable rendimiento. Aunque carece de algunas características avanzadas disponibles en otros SGBD del mercado”²². Es una buena opción tanto para aplicaciones comerciales para entretenimiento, precisamente por su facilidad de uso y tiempo reducido de puesta en marcha.

Firebird (vea la figura 3.4.3).



Figura 3.4.3 – Logo Firebird.

“Es un software de administración de una base de datos relacional. Tiene dos componentes principales, el servidor de bases de datos que vive en la misma máquina que la base de datos, y la interfaz de aplicación comúnmente referida como la librería del cliente”²³.

²¹ PostgreSQL-es (2013). Sobre PostgreSQL. Recuperado el 8 de mayo de 2016, de http://www.postgresql.org.es/sobre_postgresql.

²² The world's most popular open source database (2016). Manual de referencia. Recuperado el 8 de mayo de 2016, de <http://dev.mysql.com/doc/refman/5.7/en>.

²³ True universal open source database (2016). About Firebird. Recuperado el 8 de mayo de 2016, de <http://www.firebirdsql.org/en/about-firebird>.



En la tabla 3.4.1 mostramos algunas características de los 3 tipos de bases de datos mencionados.

Características	PostgreSQL	MySQL	Firebird
Licenciamiento gratuito	Sí	Dual	Sí
Limite por base de datos	Ilimitado	Ilimitado	Limitado
Limite por tabla	32 Tb	16 Tb en Solaris	Limitado
Número de índices por tabla	Ilimitado	64	Limitado
Límites numéricos	Precisión de 1000 dígitos	Enteros 64 bits, 38 dígitos para numéricos y decimales	Enteros 64 bits
Límites de cadenas	1Gb	65,532 bytes	32,765 bytes
Soporte de Unicode (UTF-8)	Sí	Sí	Sí
Precisión en el tipo TIMESTAMP	14 dígitos	6 dígitos	4 dígitos
Tipos BINARY y VARBINARY	Sólo binary	Sí	No
Transacciones (conformidad ACID)	Sí	Sí	Sí
Candados a nivel renglón de tabla (conurrencia)	Modelo MVCC	Modelo MVCC, solo para tablas InnoDB	Modelo MVCC
Funcionalidad TIMEOUT	Sí	No	No
Soporte de triggers	Sí	Sí, con restricciones	Sí
Estabilidad/Soporte	Sí	Sí	Solo en el Classic Server
Balaneo de Cargas	Sí	Parcial	No
Replicación	Sí	Sí	Sí

Tabla 3.4.1. – Comparativo de bases de datos (Parte I).

Características	PostgreSQL	MySQL	Firebird
Función TRUNCATE	Sí	Sí	No
Función CAST	Sí	No	Sí
Valores por defecto	CREATE	CREATE UPDATE INSERT	CREATE
Declaración MERGE	Sí	No	Sí
Otorgamiento de privilegios a nivel columna	No	Sí	No
Querys recursivos	Sí	No	Sí

Tabla 3.4.1. – Comparativo de bases de datos (Parte II).

PHP (vea la figura 3.4.4).



Figura 3.4.4 – Logo PHP.

“Es un lenguaje de uso general de script, ejecutado en el servidor originalmente diseñado para el desarrollo web de contenido dinámico”²⁴. Puede ser usado en la mayoría de los servidores web al igual que en todos los sistemas operativos y plataformas.

²⁴ The PHP Group (2016). Documentación. Recuperado el 9 de mayo de 2016, de <http://php.net>.



HTML 5 (vea la figura 3.4.5).



Figura 3.4.5 – Logo HTML 5.

“HTML, Lenguaje de Marcado para Hipertextos (HyperTextMarkupLanguage), se usa para crear y representar visualmente una página web. Determina el contenido de la página web, pero no su funcionalidad. El lenguaje HTML basa su filosofía de desarrollo en la diferenciación. Para añadir un elemento externo a la página (imagen, vídeo, script, entre otros.), este no se incrusta directamente en el código de la página, sino que hace una referencia a la ubicación de dicho elemento mediante texto”²⁵. De este modo, la página web contiene solamente texto mientras que recae en el navegador web (interpretador del código) la tarea de unir todos los elementos y visualizar la página final.

Ruby (vea la figura 3.4.6).



Figura 3.4.6 – Logo Ruby.

“Un lenguaje de programación dinámico y de código abierto enfocado en la simplicidad y productividad. Su elegante sintaxis se siente natural al leerla y fácil al escribirla”²⁶. Es un lenguaje orientado a objetos multiplataforma.

²⁵ HTML.net (2016). Introducción. Recuperado el 11 de mayo de 2016, de <http://html.net/tutorials/html5/introduction.php>.

²⁶ Ruby, el mejor amigo de un desarrollador (2016). Ruby es. Recuperado el 11 de mayo de 2016, de <https://www.ruby-lang.org/es>.



En la tabla 3.4.2 mostramos algunas características de los 3 tipos de lenguaje de programación mencionados.

Características	PHP	HTML 5	RUBY
Programación	Orientada a objetos	Estructurada	Orientada a objetos
Lenguaje libre	Sí	Sí	Sí
Conectividad a todas las bases de datos	Sí	No	Sí
Herramienta de generación de SQL	Sí	No	Sí
Gestión de excepciones	Sí	No	Sí
Plataforma	Cualquiera	Cualquiera	Cualquiera
Aprendizaje	Fácil aprendizaje con formación previa en programación el lenguaje C	Dificultad en el aprendizaje	Conocimiento previo en lenguajes orientados a objetos
Manejo de excepciones	Sí	Sí	Sí
Herramienta de generación de SQL	Sí	No	Sí
Manejo adecuado de Unicode	No	Sí	No
Hilos de ejecución simultáneos	Sí	Sí	Sí

Tabla 3.4.2. – Comparativo de lenguajes de programación.



Apache (vea la figura 3.4.7).



Figura 3.4.7 – Logo Apache.

“Es un servidor web HTTP de código abierto para la creación de páginas y servicios web. Es un servidor multiplataforma, gratuita, robusto y que se destaca por su seguridad y rendimiento”²⁷.

Cherokee (vea la figura 3.4.8).



Figura 3.4.8 – Logo Cherokee.

Es un servidor web multiplataforma que tiene como principal objetivo ser rápido y completamente funcional, sin dejar de ser liviano comparado con otros servidores.²⁸

²⁷ Apache HTTP Server (2015). Características Servidor Apache. Recuperado el 30 de abril de 2014, de <https://es.opensuse.org/Apache>.

²⁸ Cherokee (2013). Evolved Web Infrastructure Software. Recuperado el 11 de mayo de 2016, de <http://cherokee-project.com>.

Nginx (vea la figura 3.4.9).



Figura 3.4.9 – Logo Nginx.

“Es un servidor web, multiplataforma, ligero y de alto rendimiento. Es software libre, liberado bajo la licencia BSD”²⁹.

En la tabla 3.4.3 mostramos algunas características de los 3 tipos de servidores mencionados.

Características	Apache	Cherokee	Nginx
Ejecución de código	Estático/Dinámico	Estático/Dinámico	Estático/Dinámico
Uso de memoria RAM	Mediana	Baja	Baja
Trabajo asíncrono	No	Sí	Sí
SSL/TLS	Sí	Sí	Sí
Virtual Host	Sí	Sí	Sí
Balanceo de cargas	Sí	No	Sí
Reescritura de URL	Sí	No	Sí
Compresión	Sí	Sí	Sí
Control de acceso	Sí	Sí	Sí
Personalización de log	Sí	No	Sí

Tabla 3.4.3. – Comparativo de servidores web (Parte I).

²⁹ NGINX (2015). Flawless Application Delivery for the Modern Web. Recuperado el 11 de mayo de 2016, de <https://www.nginx.com>.



Características	Apache	Cherokee	Nginx
Transmisión de FLV	Sí	Sí	Sí
Ejecución de FastCGI	Sí	Sí	Sí
Estable	Muy estable	Estable	Estable
Ejecuta funciones de servidor proxy	No	No	Sí
Uso de memoria RAM	Muy alta	Baja	Baja

Tabla 3.4.3. – Comparativo de servidores web (Parte II).

Por lo que podemos concluir con respecto al entorno de desarrollo y al manejador de bases de datos, las herramientas que utilizaremos serán PHP, MySQL y Apache, porque son softwares de uso libre y no se requieren licencias para su manejo, mismos que son compatibles entre sí.

Para una mejor justificación, estas son las razones por las que se eligió MySQL:

- La velocidad es muy buena a la hora de realizar las operaciones, lo que le hace uno de los gestores que ofrece mayor rendimiento.
- Su bajo consumo lo hace apto para ser ejecutado en una maquina con escasos recursos sin ningún problema.
- Las utilidades de administración son envidiables para muchos de los gestores comerciales existentes, debido a su gran facilidad de configuración e instalación.
- Tiene una probabilidad reducida de corromper los datos, incluso en los casos en los que los errores no se produzcan en el propio gestor, sino en el sistema en el que se está.
- El conjunto de aplicaciones PHP, MySQL y Apache es uno de los más utilizados en internet en servicios de foro y de buscadores de aplicaciones.



Razones por las que se eligió PHP:

- Independencia de plataforma.
- Portabilidad.
- Facilidad de instalación, configuración y utilización.
- Soporte para gran cantidad de bases de datos.
- Tiene una mejor interacción con MySQL.

Razones por las que se eligió Apache:

- Modular.
- Código abierto.
- Multiplataforma.
- Extensible.
- Ejecuta código PHP.
- Portabilidad.
- Facilidad para conseguir ayuda y soporte.

Para nuestra aplicación utilizaremos XAMPP, ya que es una herramienta muy práctica que nos permite instalar MySQL, PHP y Apache. Esta herramienta es multiplataforma y la podemos usar en diferentes sistemas operativos. Su instalación es sencilla sólo requiere mínimas configuraciones.

3.5 Diseño modular de la interfaz de usuario

De acuerdo a las necesidades que se pretenden cubrir con el sitio web, el sistema se compondrá de cinco bloques principales, los cuales se pueden ver en la figura 3.5.1.

El primer bloque será el correspondiente a la autenticación del usuario, es necesario este bloque para garantizar la seguridad de la información: Solo podrán ingresar usuarios registrados.



Otra funcionalidad importante es lo referente a búsqueda, este módulo deberá permitir realizar búsquedas de acuerdo a los siguientes criterios: Grupo y materia, semestre, grupo, profesor, día y búsqueda general.

El tercer módulo corresponde a las modificaciones a los horarios, se podrá realizar alta, baja y actualización.

Referente al cuarto y quinto módulo, éstos corresponden a la funcionalidad de dar de alta o eliminar usuarios y salones, respectivamente.

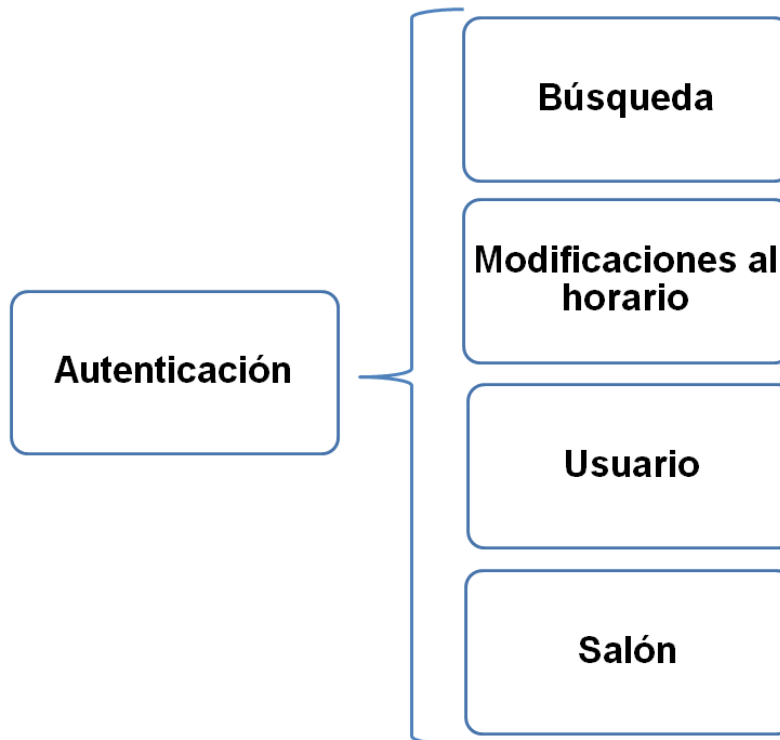
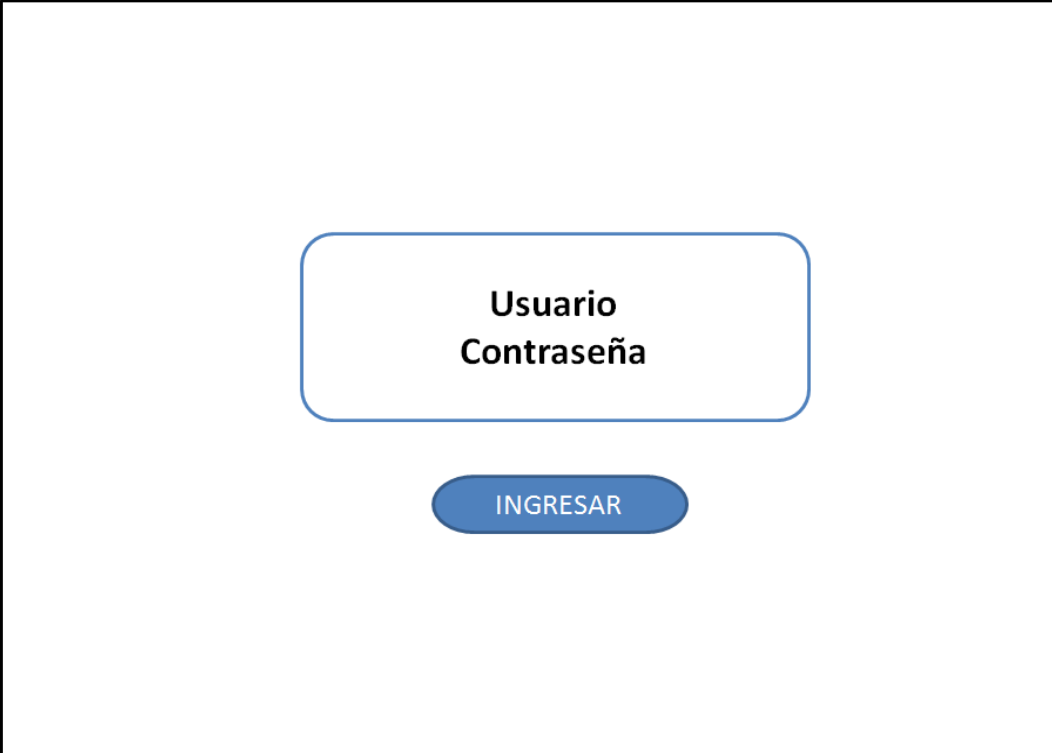


Figura 3.5.1 – Diagrama de bloques del sitio web.

Lo expuesto anteriormente corresponde a la composición general del sitio web, a continuación se mostrará el diseño en particular de las páginas que compondrán cada módulo.

La primer página que verá el usuario será la correspondiente a la autenticación, en ésta se deberán ingresar un nombre de usuario y contraseña válidos. La distribución corresponderá a lo esbozado en la figura 3.5.2.



El diagrama muestra un recuadro rectangular que representa la interfaz de autenticación. En el centro del recuadro hay un campo de entrada con un borde azul y esquinas redondeadas. Dentro de este campo, el texto "Usuario" está en la línea superior y "Contraseña" está en la línea inferior. Debajo del campo de entrada, centrado, hay un botón azul con el texto "INGRESAR" en mayúsculas blancas.

Figura 3.5.2 – Distribución de la página de autenticación.

En la parte central estarán los espacios correspondientes para que el usuario se autentifique y pueda ingresar al sistema de horarios, en esta página no habrá escudos ni gráficos o animaciones.

Una vez que el usuario ingrese correctamente sus datos, accederá completamente al sistema de horarios, la distribución general de la interfaz será como se muestra en la figura 3.5.3.



Figura 3.5.3. Interfaz principal del sistema de horarios.

▪ Encabezado

El encabezado tendrá el logotipo correspondiente del Sistema de Universidad Abierta y a Distancia, dicha imagen contendrá un enlace a la página principal del SUAyED. Para brindar una experiencia amigable al usuario, la opción para cerrar la sesión, mostrará el nombre del usuario activo.

De acuerdo a la sección en la que se encuentre el usuario, en el encabezado también se mostrará el nombre del área en la que se esté trabajando.

▪ Menú

El menú estará localizado en la parte central izquierda, tendrá una distribución vertical y será de dos niveles. La estructura puede observarse en la figura 3.5.4. El menú estará

visible durante toda la navegación por el sitio, es decir, se mantendrá fijo. El despliegue del submenú será hacia la derecha.

La estructura que en general tendrá la distribución de páginas, combinará los tipos jerárquico y red, ya que hay niveles entre las opciones pero también será posible navegar de una opción a otra sin posibilidad de “perderse” o de seguir un orden forzoso.



Figura 3.5.4 Estructura del menú y submenús.

▪ Contenido

Todos los formularios que se utilizarán en la aplicación se desplegarán en la parte central de la página, más adelante se explicará la distribución que tendrán de acuerdo al número de funciones que tenga habilitada cada opción. Ésta será la única sección que irá cambiando de acuerdo a la navegación que haga el usuario.



▪ Pie de página

En esta sección se tendrán los escudos de la Institución Educativa con un enlace respectivo a la página de internet principal de dicha escuela. También se incluirá la leyenda de derechos reservados y la fecha de realización de la página.

Tanto en la sección de encabezado como pie de página deben respetarse las normas revisadas en la sección 1.3 de este trabajo, debido a que el presente sitio web será institucional.

De acuerdo a la función en la que esté trabajando el usuario, existe la posibilidad de que el área de contenido tenga más de un formulario, en la figura 3.5.5 se puede ver la distribución que tendrá el área de contenido al desplegar varios formularios.



Figura 3.5.5 – Distribución del área de contenido con varios formularios.

Este tipo de distribución se visualizará, por ejemplo, con la función de Alta de horario, debido a que se deben ingresar los datos para programar aspectos como grupo y materia, encuadre, asesorías y exámenes; por lo que serán necesarios varios formularios. Es importante cuidar la distribución de cada uno, para evitar confusiones en el usuario.

Como se puede apreciar en la figura 3.5.5, aunque en el encabezado se mostrará la función en la que se esté trabajando; también en el área de contenido se desplegará la opción del menú que se haya seleccionado, esto con el fin de orientar al usuario.

En la figura 3.5.6 se muestra la distribución del área de contenido cuando sólo se tenga un formulario.

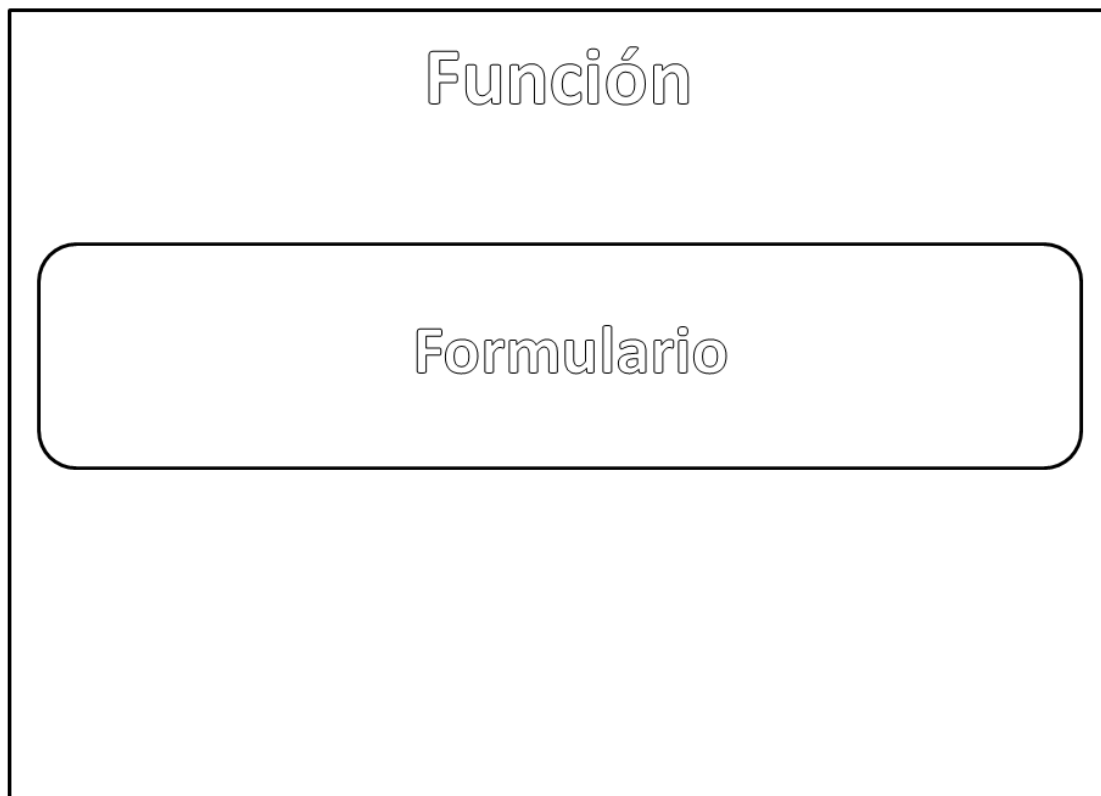


Figura 3.5.6 – Distribución del área de contenido con un formulario.

Este tipo de distribución será el que se utilice en la mayoría de las opciones, debido a que serán pocos datos que se soliciten, por ejemplo, en las opciones de altas y bajas de usuarios, o en las búsquedas, donde sólo se solicitarán los criterios de búsqueda. Para



brindar claridad y limpieza a la página, el formulario estará colocado en la parte central del área de contenido.

Además de las páginas hasta ahora mencionadas, también se integrarán una página de inicio y una de contacto. Ambas respetarán el diseño planteado. En la página de inicio sólo se mostrará, en el área de contenido, un breve mensaje de bienvenida utilizando el nombre del usuario activo en la sesión.

Respecto a la página de contacto, la finalidad de ésta, será la recopilación de dudas y/o comentarios por parte de los usuarios. Se utilizará la distribución mostrada en la figura 3.5.6: sólo contendrá un formulario donde se le soliciten al usuario datos de contacto y un comentario.





CAPÍTULO IV

“DISEÑO Y CONSTRUCCIÓN DEL SISTEMA”

4.1 Modelado del sistema

4.1.1 Diagrama de contexto.

Para desarrollar el sistema es indispensable seguir una metodología con el fin de obtener un software de calidad y que sea una buena solución al problema. El modelado es importante porque con él podemos explicar el comportamiento del sistema, además de tener un mayor control sobre el mismo. Este diagrama, muestra a través de flujos de datos, las interacciones existentes entre los agentes externos y el sistema, sin describir en ningún momento la estructura del sistema.

En la figura 4.1.1.1 se muestra el diagrama general del sistema. Este diagrama permite de una manera global tener un entendimiento general y abstracto de lo que será el sistema.

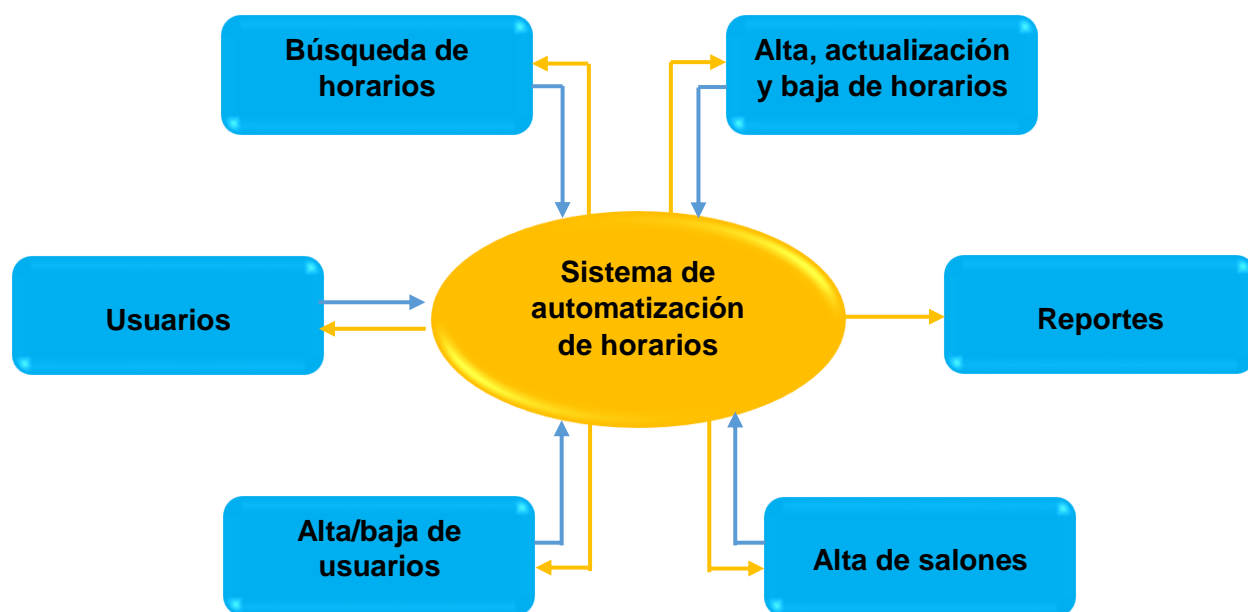


Figura 4.1.1.1 – Diagrama de contexto del sistema.

- Usuarios: Este será el encargado de dar de alta, baja y actualizar los horarios de asesorías. También podrá dar de alta salones para dichas asesorías.



- Alta, actualización y baja: Aquí se asignarán los días, hora de entrada, hora de salida y profesor para las asesorías. También se podrán actualizar y dar de baja.
- Búsqueda de horarios: En esta sección se podrá hacer la búsqueda por grupo, materia, semestre y profesor.
- Alta y baja de usuarios: el usuario que tenga acceso al sistema podrá dar de alta a otros usuarios, así como también los podrá dar de baja.
- Reportes. Estos reportes serán generados en formato PDF y cualquier alumno tendrá acceso a ellos.
- Alta de salones: Es importante considerar que los salones asignados no siempre son los mismos, así que se contará con un apartado para agregar o eliminar salones.

4.1.2 Diagrama Entidad- Relación

El diagrama Entidad - Relación es una herramienta que sirve para realizar el modelado de bases de datos, en el cual visualizamos de forma gráfica los objetos pertenecientes a la base de datos. Se basa en la percepción del mundo real, que está formada por objetos básicos llamados entidades y las relaciones entre estos objetos. Este diseño es útil y práctico para lograr entender desde el nivel más esencial la base de datos pues es lo fundamental para lograr el diseño físico.

El modelo entidad relación contiene los objetos que a continuación se describen:

- Entidad, es el pilar del modelo entidad relación, representa cualquier cosa u objeto del mundo real, que es distinguible del resto pues tiene valores específicos e independientes.
- Atributos, son características que definen o identifican cada entidad en un conjunto de entidades en forma única.
- Conjunto de entidades, es una colección de relaciones de la misma naturaleza que comparten los mismos atributos o características.

- Cardinalidad, describe cierta dependencia entre entidades permitiendorelaciones entre ellas, de éstas se desprenden los siguientes tipos de cardinalidades:
 - Uno a uno (1:1), una instancia de la entidad A se relaciona con una y solamente una de la entidad B.
 - Uno a varios o varios a uno (1:M), cada instancia de la entidad A se relaciona con varias instancias de la entidad B.
 - Varios a varios (M:M), Cualquier instancia de la entidad A se relaciona con cualquier instancia de la entidad B.
- Llave primaria, identifica en forma única cada instancia de la entidad, por este motivo no debe ser nulo y podrá existir en otra entidad como llave foránea.
- Llave foránea, es un identificador de la instancia que determina la relación con otra entidad por medio de su llave primara, con esto se forma la asociación entre ellas.

La figura 4.1.2.1 representa el diagrama entidad relación utilizado para realizar el sistema de horarios.

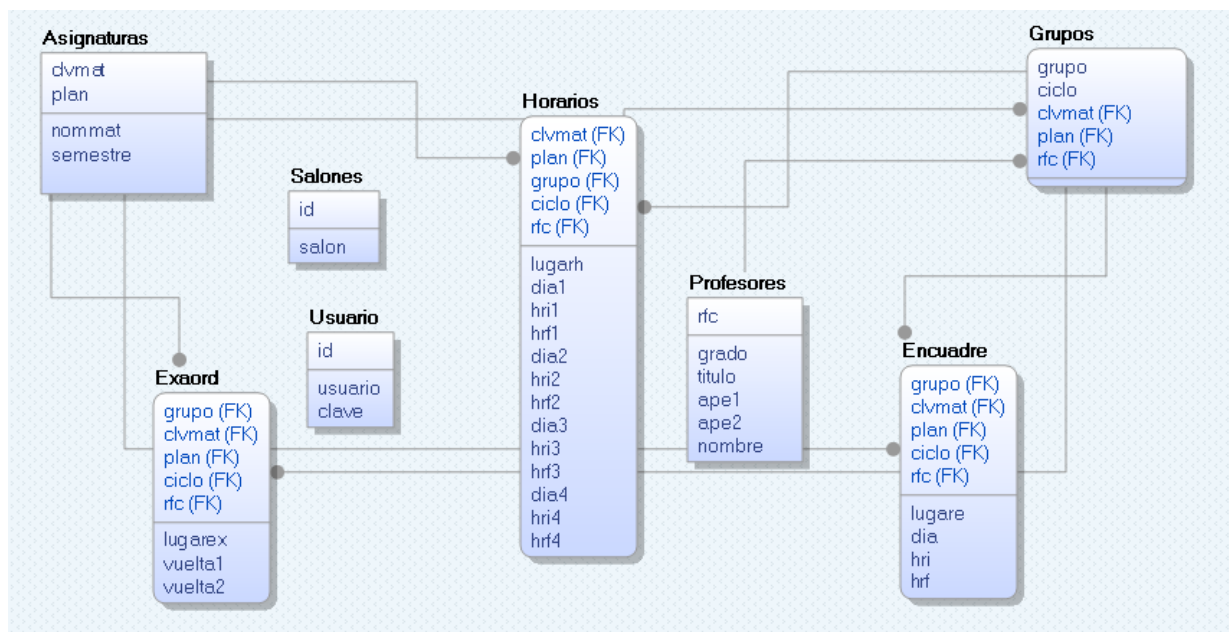


Figura 4.1.2.1 – Diagrama Entidad-Relación.

4.1.3 Diagrama de procesos

El diagrama de procesos representa de manera esquemática a los procesos que interactúan en el sistema, la figura 4.1.3.1 muestra el diagrama general. De acuerdo a los requerimientos solicitados, se tienen los siguientes procesos:

- 1: Autenticación
- 2: Menú principal
- 3: Alta de horario
- 4: Buscar horario
- 5: Actualizar horario
- 6: Baja de horario
- 7: Modificar usuario
- 8: Modificar salones

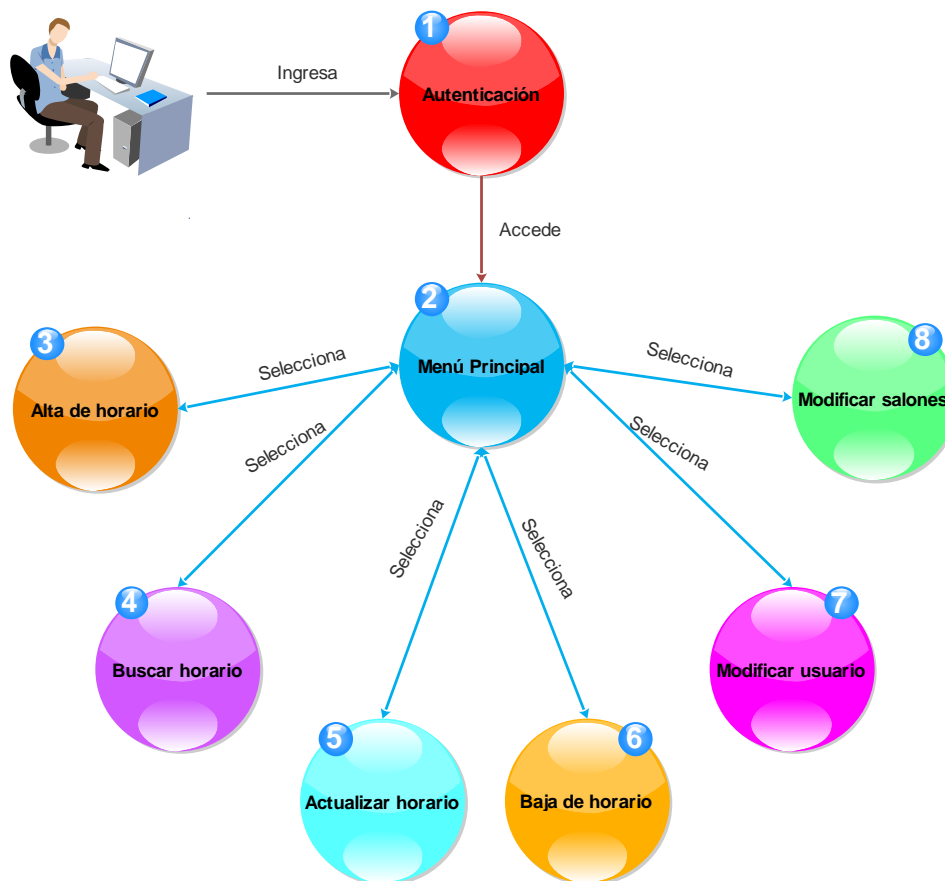


Figura 4.1.3.1 – Diagrama general.

Proceso 1: Autenticación

Es el proceso mediante el cual el usuario tiene que ingresar credenciales válidas para tener acceso al sistema. Al ingresar los datos, se genera una consulta en la base de datos y, si es un usuario registrado, el usuario accede al menú principal. En la figura 4.1.3.2 puede verse el diagrama de este proceso.

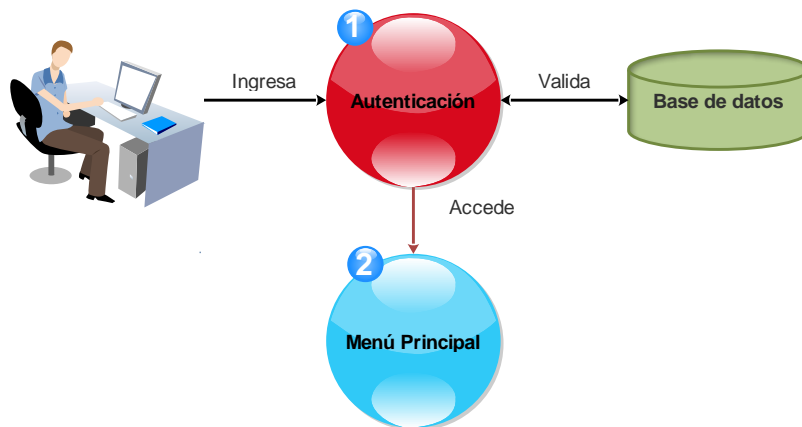


Figura 4.1.3.2 – Proceso 1.1: Autenticación.

Proceso 3: Alta de horario

En este proceso el usuario llena el formulario con los datos correspondientes a: Grupo y materia, encuadre, asesoría y exámenes ordinarios. Hecho esto, se almacenan los datos en la base de datos. Si existe algún empalme con un horario previamente creado, no se permite guardar los datos en la base de datos y se envía un mensaje de error al usuario para que corrija los datos ingresados. En la figura 4.1.3.3 está el diagrama de este proceso.



Figura 4.1.3.3 – Proceso 3.1: Alta de horario.

Proceso 4: Buscar horario

Para buscar un horario determinado, el usuario debe seleccionar el parámetro de búsqueda: Grupo y materia, semestre, grupo, profesor, día y general. El sistema envía una consulta a la base de datos, y si la búsqueda es exitosa, se despliega el horario encontrado en pantalla, también es posible obtener el horario en formato PDF. En la figura 4.1.3.4 puede verse el diagrama de este proceso.

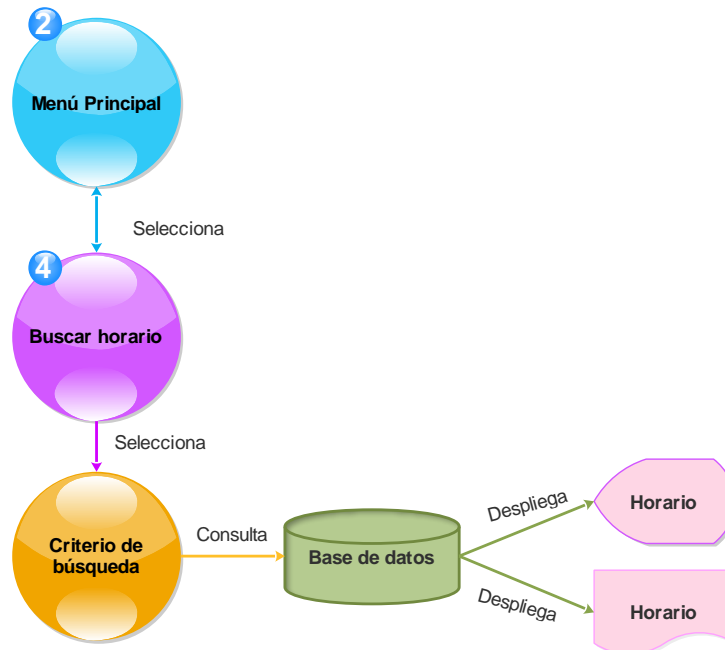


Figura 4.1.3.4 – Proceso 4.1: Buscar horario.

Proceso 5: Actualizar horario

En este proceso, el usuario ingresa grupo y clave de materia, se realiza la consulta en la base de datos y se despliega el formulario con los datos precargados. El usuario realiza la modificación que requiera y se guarda el registro actualizado en la base de datos. En la figura 4.1.3.5 se puede ver el diagrama de este proceso.

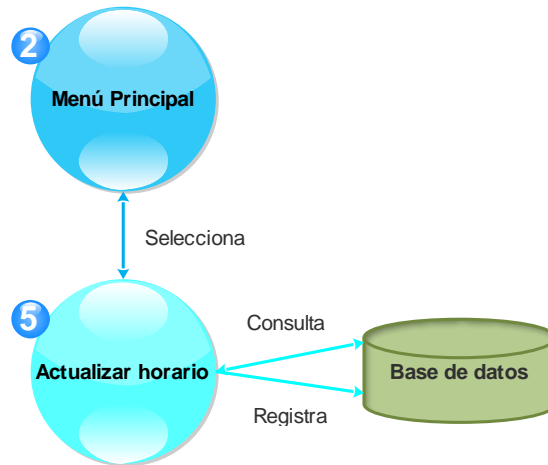


Figura 4.1.3.5 – Proceso 5.1: Actualizar horario.

Proceso 6: Baja de horario

En la figura 4.1.3.6 se puede ver el diagrama de baja de horario. En este proceso, el usuario ingresa grupo y clave de materia, se realiza la consulta en la base de datos y el sistema pide confirmar la baja de la materia para ese grupo. Si el usuario confirma la baja, el registro es eliminado de la base de datos.

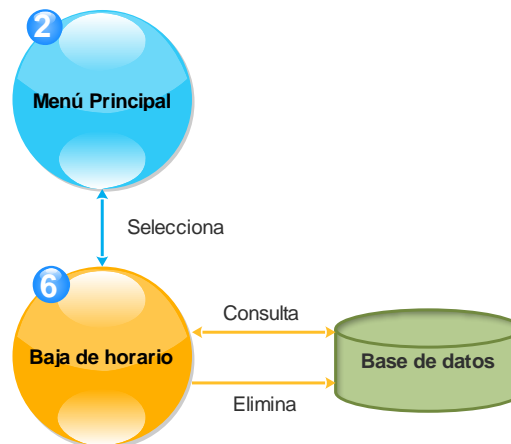


Figura 4.1.3.6 – Proceso 6.1: Baja de horario.

Proceso 7: Modificar usuario.

Este proceso contiene dos subprocesos: Registrar (7.2) y Eliminar (7.3). Si el usuario selecciona Registrar, se despliega el formulario y, una vez completado con los datos del nuevo usuario, se guarda el registro en la base de datos. Para el caso del subproceso Eliminar, se genera una consulta a la base de datos para desplegar a los usuarios registrados, se selecciona el usuario a eliminar y se borra el registro de la base de datos. En la figura 4.1.3.7 se puede ver el diagrama de este proceso y los subprocesos que lo componen.

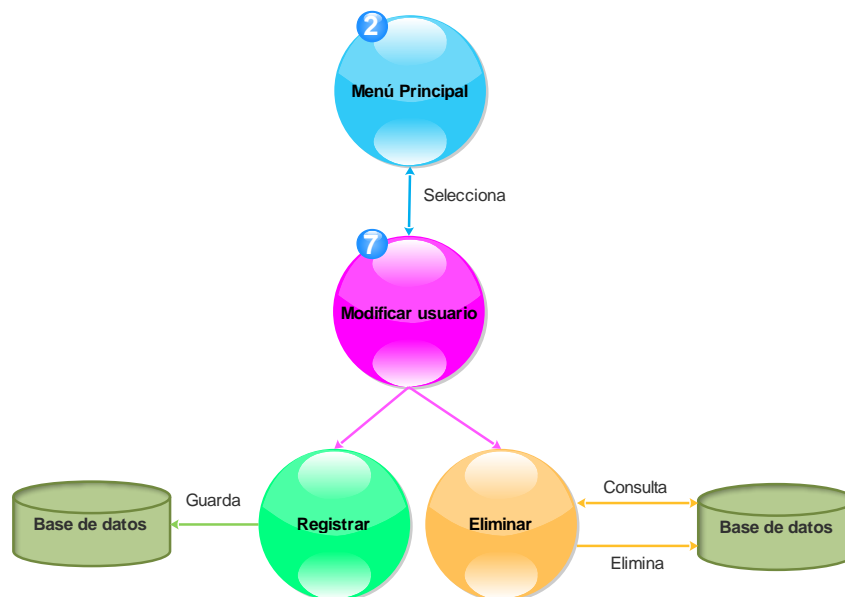


Figura 4.1.3.7 – Proceso 7.1: Modificar usuario.

Proceso 8: Modificar salones.

La figura 4.1.3.8 muestra el diagrama del proceso para modificar los salones registrados. Este proceso contiene dos subprocesos: Registrar (8.2) y Eliminar (8.3). Si el usuario selecciona Registrar, se le solicita al usuario ingresar el nombre o número de salón a dar de alta y se guarda el registro en la base de datos. Para el caso del subproceso Eliminar, se genera una consulta a la base de datos para desplegar a los salones registrados, se selecciona el salón a eliminar y se borra el registro de la base de datos.

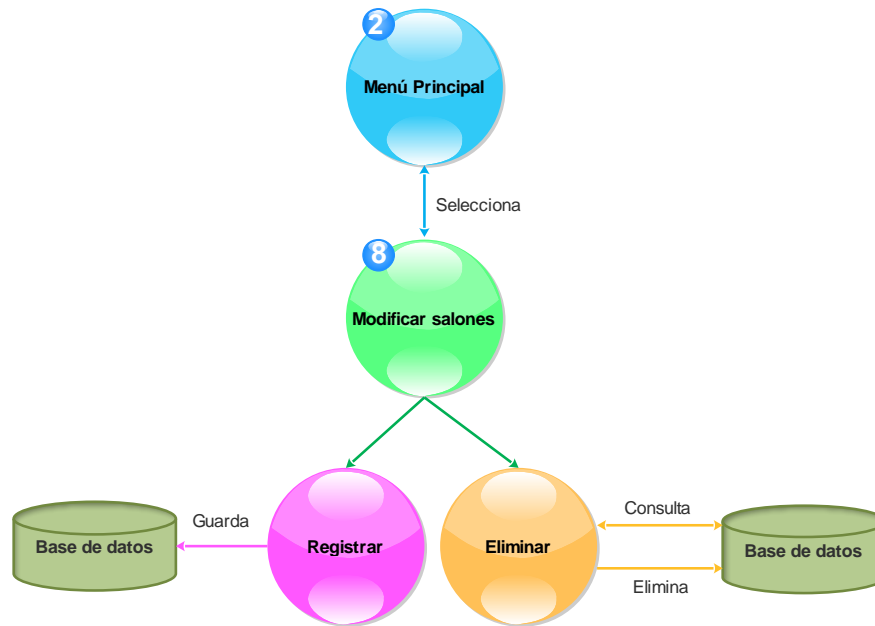


Figura 4.1.3.8 – Proceso 8.1: Modificar salones.

4.1.4 Diccionario de datos.

Son un componente importante del análisis estructurado ya que por sí solos los diagramas de flujo de datos no describen el objeto de la investigación. El diccionario de datos proporciona más información relacionada con el sistema.

“Un diccionario de datos es un conjunto de registros que contiene las características lógicas y puntuales de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización, se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño”³⁰.

El diccionario de datos son datos acerca de los datos que se conocen como metadatos y describen a cada uno de los elementos contenidos en un registro en forma clara y precisa. Si se construye un diccionario de datos completo puede hacerse ingeniería inversa con

³⁰ Kroenke David (2003). Procesamiento de bases de datos. México: Prentice Hall.



alguna herramienta de “CAD” y se pueden generar los diagramas que se deseen. Identifica los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información.

En el diccionario de datos se documenta la información generada por el diseñador de bases de datos, sus principales utilidades son:

- Describir el propósito de la Base de Datos.
- Descripción detallada de cada tabla dentro de la Base de Datos.
- Documentar la estructura interna de cada tabla, incluyendo sus campos y sus tipos de datos con comentarios, todos los índices y todas las vistas.
- Describir reglas, como pueden ser el uso de valores no nulos o valores únicos.

Este diccionario es muy importante pues contiene todos los nombres y características de los atributos de cada objeto de la base de datos del sistema, en resumen contiene metadatos y frecuentemente es utilizado por el administrador de la base de datos para el registro de las decisiones tomadas en cuanto a la estructura y nombre de los objetos.

“El objetivo de un diccionario de datos es dar precisión sobre los datos que se manejan en un sistema, evitando así malas interpretaciones o ambigüedades”³¹.

El contenido de un registro de un diccionario es un conjunto de columnas relacionada a una tabla del sistema, que debe tener la siguiente información:

- Nombre.
- Acrónimo.
- Tipo de dato.
- Longitud.
- Tipo de llave.

³¹ Silberchatz, A., Henry, F.(1998). 5a ed. Fundamentos de las bases de datos. España: McGrall Hill.



- Nulos.
- Tablas relacionadas.
- Descripción.

A continuación se muestran las tablas 4.1.4.1 a 4.1.4.8 que conforman la base de datos.

Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
Clave de la materia	CLVMAT	Smallint(4)	Primaria	No	Encuadre, exámenes ordinarios, grupos y horarios	Clave de la materia
Nombre de la materia	NOMMAT	Varchar(40)		No		Nombre de la materia
Créditos	CREDITOS	Char(2)		No		
Semestre	SEMESTRE	Tinyint(2)		No		
Plan	PLAN	Tinyint(1)	Primaria	No		
Seriación	SERIACION	Smallint(4)		No		
Área	AREA	Char(2)		No		

Tabla 4.1.4.1 – Asignaturas.

Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
Grupo	GRUPO	Varchar(4)	Foránea	No	Asignaturas, exámenes ordinarios, grupos y horarios	Grupo de la materia
Clave de la Materia	CLVMAT	Mediumint(5)	Foránea	No	Exámenes ordinarios, grupos y horarios	Clave de la materia
Lugar de encuadre	LUGARE	Varchar(30)		No		Lugar donde se llevara a cabo el encuadre
Día	DIA	Date		No		Día del encuadre
Hora de inicio	HRI	Varchar(7)		No		Hora de inicio del encuadre
Hora final	HRF	Varchar(7)		No		Hora final del encuadre

Tabla 4.1.4.2 – Encuadre.



Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
Grupo	GRUPO	Varchar(4)	Foránea	No	Encuadre, grupos y horarios	Grupo de la materia
Clave de la materia	CLVMAT	Mediumint(5)	Foránea	No	Asignaturas, encuadre, grupos y horarios	Clave de la materia
Lugar de exámenes ordinarios	LUGAREX	Varchar(30)		No		Lugar donde se realizarán los exámenes ordinarios
Primera vuelta	VUELTA1	Date		No		Fecha del primer examen ordinario
Segunda vuelta	VUELTA2	Date		No		Fecha del segundo examen ordinario

Tabla 4.1.4.3 – Exámenes ordinarios.

Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
Grupo	GRUPO	Varchar(4)	Primaria	No	Encuadre, Exámenes ordinarios y horarios	Grupo de la materia
Clave de la materia	CLVMAT	Mediumint(5)	Foránea	No	Asignaturas, encuadre, exámenes ordinarios y horarios	Clave de la materia
Ciclo escolar	CICLO	Smallint(5)	Primaria	No		Ciclo escolar
RFC	RFC	Varchar(10)	Foránea	No	Profesores	RFC del maestro que impartirá la materia
Hora de entrada día 1	DHE1	Varchar(6)		No		Hora de entrada del primer día (lunes)

Tabla 4.1.4.4 – Grupos (Parte I).



Grupos						
Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
Hora de salida día 1	DHS1	Varchar(6)		No		Hora de salida del primer día (lunes)
Hora de entrada día 2	DHE2	Varchar(6)		No		Hora de entrada del segundo día (martes)
Hora de salida día 2	DHS2	Varchar(6)		No		Hora de salida del segundo día (martes)
Hora de entrada día 3	DHE3	Varchar(6)		No		Hora de entrada del tercer día (miércoles)
Hora de salida día 3	DHS3	Varchar(6)		No		Hora de salida del tercer día (miércoles)
Hora de entrada día 4	DHE4	Varchar(6)		No		Hora de entrada del cuarto día (jueves)
Hora de salida día 4	DHS4	Varchar(6)		No		Hora de salida del cuarto día (jueves)
Hora de entrada día 5	DHE5	Varchar(6)		No		Hora de entrada del cuarto día (viernes)
Hora de salida día 5	DHS5	Varchar(6)		No		Hora de salida del cuarto día (viernes)
Hora de entrada día 6	DHE6	Varchar(6)		No		Hora de entrada del cuarto día (sábado)
Hora de salida día 6	DHS6	Varchar(6)		No		Hora de salida del cuarto día (sábado)

Tabla 4.1.4.4 – Grupos (Parte II).



Grupos						
Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
Hora de entrada día 7	DHE7	Varchar(6)		No		Hora de entrada del cuarto día (domingo)
Hora de salida día 7	DHS7	Varchar(6)		No		Hora de salida del cuarto día (domingo)
Salón	SALON	Char(3)		No	Salones	Salón en el que se imparte la materia
Cupo	CUPO	Tinyint(2)		No		Cupo de alumnos en el grupo
Turno	TURNO	Tinyint(1)		No		Turno en que se imparte la materia
Horas a la semana	HORASEM	Tinyint(2)		No		Total de horas que se imparte la materia a la semana
Institución	INSTITUCION	Varchar(100)		No		Institución en la que se imparte la materia

Tabla 4.1.4.4 – Grupos (Parte III).

Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
Grupo	GRUPO	Varchar(4)	Foránea	No	Encuadre, exámenes ordinarios y grupos	Grupo de la materia
Clave de la materia	CLVMAT	Mediumint(5)	Foránea	No	Asignaturas, encuadre, exámenes ordinarios y grupos	Clave de la materia
Lugar de la asesoría	LUGARH	Varchar(3)		No		Lugar donde se realizará la asesoría

Tabla 4.1.4.5 – Horarios (Parte I).



Horarios						
Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
Día 1	DIA1	Date		No		Día de la primera asesoría
Hora inicial día 1	HRI1	Varchar(7)		No		Hora inicial de la primera asesoría
Hora final día 1	HRF1	Varchar(7)		No		Hora final de la primera asesoría
Día 2	DIA2	Date		No		Día de la segunda asesoría
Hora inicial día 2	HRI2	Varchar(7)		No		Hora inicial de la segunda asesoría
Hora final día 2	HRF2	Varchar(7)		No		Hora final de la segunda asesoría
Día 3	DIA3	Date		No		Día de la tercera asesoría
Hora inicial día 3	HRI3	Varchar(7)		No		Hora inicial de la tercera asesoría
Hora final día 3	HRF3	Varchar(7)		No		Hora final de la tercera asesoría
Día 4	DIA4	Date		No		Día de la cuarta asesoría
Hora inicial día 4	HRI4	Varchar(7)		No		Hora inicial de la cuarta asesoría
Hora final día 4	HRF4	Varchar(7)		No		Hora final de la cuarta asesoría

Tabla 4.1.4.5 – Horarios (Parte II).



Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
RFC	RFC	Varchar(10)	Primaria	No	Grupos	RFC
Homoclave	HOMO	Char(3)		No		Homoclave
Grado	GRADO	Tinyint(1)		No		Grado
Título	TITULO	Varchar(4)		No		Título (licenciatura, maestría o doctorado))
Primer apellido	APE1	Varchar(30)		No		Apellido paterno
Segundo apellido	APE2	Varchar(30)		No		Apellido materno
Nombre	NOMB	Varchar(30)		No		Nombre
Nacionalidad	NAC	Char(1)		No		Nacionalidad
Sexo	SEXO	Char(1)		No		Sexo
Curp	CURP	Varchar(18)		No		CURP

Tabla 4.1.4.6 – Profesores.

Salones						
Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
Identificador	Id	Int(11)	Primaria	No		Identificador de la tabla salones
Salón	SALON	Varchar(35)		No	Grupos	Salón que se dará de alta para las asesorías

Tabla 4.1.4.7 – Salones.

Usuarios						
Nombre	Acrónimo	Tipo de dato y longitud	Llave	Nulo	Tablas relacionadas	Descripción
Identificador	Id	Int(11)	Primaria	No		Identificador de la tabla usuarios
Usuario	usuario	Varchar(15)		No		Nombre de usuario
Clave	clave	Varchar(205)		No		Clave de usuario

Tabla 4.1.4.8 – Usuarios.



4.1.5 Normalización

La normalización es el proceso de organizar la información de la base de datos para pasar de datos complejos a un conjunto de estructuras más simples, nos beneficia para mejorar el rendimiento de la aplicación, eliminar inconsistencias y redundancia de la base de datos, pues si las tablas no están correctamente diseñadas, el rendimiento del sistema es afectado. Cada regla de normalización se basa en la anterior.

La normalización se ejerció pues el procedimiento de almacenar todos los datos en un solo archivo o tabla, era ineficiente y provocaba errores de lógica al momento de manipular los datos.

Algunas ventajas al aplicar las reglas de normalización son:

- Claridad. Debido a que la representación de la información es a través de tablas simples y pequeñas.
- Facilidad de implantación. Las tablas se almacenan físicamente como archivos planos.
- Facilidad de entendimiento. Los datos están agrupados en tablas que identifican claramente un objeto.
- Flexibilidad. La información que necesitan los usuarios se puede obtener de las tablas mediante las operaciones del álgebra relacional.
- Mínima redundancia. La información no estará duplicada innecesariamente
- Precisión. Las interrelaciones entre tablas consiguen mantener la información relacionada con toda exactitud.
- Ahorro de espacio en disco. Una base de datos normalizada ocupa menos espacio en disco que una no normalizada pues hay menos repetición de datos, lo que tiene como consecuencia un mucho menor uso de espacio en disco.
- Seguridad. Consultar o actualizar información es mucho más sencillo de realizar.
- Mejora la independencia de datos. Los programas son independientes de las estructuras, al aumentar la base de datos añadiendo nuevos atributos o nuevas tablas no se afecta a los programas que las manejan.



- Facilidad de gestión. Los lenguajes manipulan la información de forma sencilla al estar los datos basados en el álgebra y cálculo relacional.
- Máximo rendimiento de las aplicaciones. Sólo se trata aquella información que va a ser de utilidad a cada aplicación.
- Evita anomalías en inserciones, modificaciones y borrados.

Existen varios niveles de normalización, pero hemos empleado la Primera (1NF), la Segunda (2NF) y la Tercera Forma Normal (3NF), porque hasta este nivel ha sido suficiente para cumplir con las necesidades de la Base de Datos del sistema. Es importante señalar que para que una tabla esté en un nuevo nivel o forma de normalización, es necesario que haya cumplido con la regla anterior.

A continuación se presenta la información no normalizada en la tabla 4.1.5.1.

Clave materia	Nombre materia	Nombre profesor	RFC	Gru po	Ciclo	Id salón	Nombre salón
1204 1311	Política social Teoría social	Almanza Beltrán Irma	AABI581226	9004	20161	1	Salón 15
1311 1112	Teoría social Teoría económica	Casas Torres Graciela	CATG541226	9011	20161	2	Salón 16
1314	Psicología social	Arguello Zepeda Francisco	AUZF531104	9104	20161	3	Aula virtual
1112	Teoría económica	Barreto Pérez Silvia	BAPE680807	9002	20162	4	Sala videoconferencias
1411	Problemática urbana	Hernández Olivera José Luis	HEOL510919	9021	20161	5	Salón 11

Tabla 4.1.5.1 – Tabla no normalizada (Parte I).



Clave materia	Nombre materia	Nombre profesor	RFC	Grupo	Ciclo	Id salón	Nombre salón
1512	Salud pública	López Alfredo Arcadio	LOAL550109	9005	20162	6	Salón 8
1613	Salud mental	Martínez	MAVA410121	9013	20162	7	Sala de usos múltiples
1512	Salud pública	Verdugo Armando					
1131	Teoría social	Ornelas	OEBJ760330	9011	20162	9	Salón 3
1613	Salud mental	Bernal Jorge					
1413	Derechos humanos	Portillo Téllez Francisco	POTF691203	9009	20162	10	Salón 8

Tabla 4.1.5.1 – Tabla no normalizada (Parte II).

En la figura 4.1.5.1 podemos observar que la tabla no se encuentra normalizada, así que procederemos aplicar las formas normales.

Primera Forma Normal 1FN

Una tabla se encuentra en Primera Forma Normal si y solo si todas las columnas de todas las tablas contienen solo valores atómicos; es decir, un único valor. Establece que las columnas repetidas deben eliminarse y colocarse en tablas separadas.



- Tabla Grupo_Ciclo

Clave materia	Nombre materia	Grupo	Ciclo	Id salón	Nombre salón
1204	Política social	9004	20161	1	Salón 15
1311	Teoría social	9004	20161	1	Salón 3
1112	Teoría económica	9011	20161	2	Salón 16
1314	Psicología social	9104	20161	3	Aula virtual
1411	Problemática urbana	9021	20161	5	Salón 11
1512	Salud pública	9005	20162	6	Salón 8
1613	Salud mental	9013	20162	7	Sala de usos múltiples
1413	Derechos humanos	9009	20162	10	Sala videoconferencias

Tabla 4.1.5.2 – Tabla Grupo_ciclo.

- Tabla profesor

Id	RFC	Nombre profesor
1	AABI581226	Almanza Beltrán Irma
2	CATG541226	Casas Torres Graciela
3	AUZF531104	Arguello Zepeda Francisco
4	BAPE680807	Barreto Pérez Silvia
5	HEOL510919	Hernández Olivera José Luis
6	LOAL550109	López Alfredo Arcadio
7	MAVA410121	Martínez Verdugo Armando
8	OEBJ760330	Ornelas Bernal Jorge
9	POTF691203	Portillo Téllez Francisco

Tabla 4.1.5.3. – Tabla profesor.



Segunda Forma Normal 2FN

La regla de la Segunda Forma Normal establece que todas las dependencias parciales se deben eliminar y separar dentro de sus propias tablas. Una dependencia parcial es un término que describe a aquellos datos que no dependen de la llave primaria de la tabla para identificarlos.

Una vez alcanzado el nivel de la Segunda Forma Normal, se controlan la mayoría de los problemas de lógica. Podemos insertar un registro sin un exceso de datos en la mayoría de las tablas.

- Tabla materia

Clave materia	Nombre materia	Id salón	Nombre salón
1204	Política social	1	Salón 15
1311	Teoría social	1	Salón 3
1112	Teoría económica	2	Salón 16
1314	Psicología social	3	Aula virtual
1411	Problemática urbana	5	Salón 11
1512	Salud pública	6	Salón 8
1613	Salud mental	7	Sala de usos múltiples
1413	Derechos humanos	10	Sala videoconferencias

Tabla 4.1.5.4 –Tabla Materia.



- Tabla Grupo_ciclo

Grupo	Ciclo
9004	20161
9004	20161
9011	20161
9104	20161
9021	20161
9005	20162
9013	20162
9009	20162

Tabla 4.1.5.5 – Tabla Grupo_ciclo.

Tercera Forma Normal 3FN

Una tabla está normalizada en esta forma si todas las columnas, que no son llave, son funcionalmente dependientes por completo de la llave primaria y no hay dependencias transitivas. Una dependencia transitiva es aquella en la cual las columnas que no son llave son dependientes de otras columnas que tampoco son llave.

Cuando las tablas están en la Tercera Forma Normal se previenen errores de lógica cuando se insertan o borran registros. Cada columna en una tabla está identificada de manera única por la llave primaria, y no debe haber datos repetidos. Esto provee un esquema limpio y elegante, que es fácil de trabajar y expandir.

- Tabla materia

Clave materia	Nombre materia
1204	Política social
1311	Teoría social
1112	Teoría económica
1314	Psicología social
1411	Problemática urbana
1512	Salud pública
1613	Salud mental
1413	Derechos humanos

Tabla 5.1.4.6 – Tabla Asignaturas.

- Tabla profesor

Nombre profesor	RFC
Almanza Beltrán Irma	AABI581226
Casas Torres Graciela	CATG541226
Arguello Zepeda Francisco	AUZF531104
Barreto Pérez Silvia	BAPE680807
Hernández Olivera José Luis	HEOL510919
López Alfredo Arcadio	LOAL550109
Martínez Verdugo Armando	MAVA410121
Ornelas Bernal Jorge	OEBJ760330
Portillo Téllez Francisco	POTF691203

Tabla 5.1.4.7 – Tabla Profesores.



- Tabla salones

Id salón	Nombre salón
1	Salón 15
2	Salón 16
3	Aula virtual
4	Sala videoconferencias
5	Salón 11
6	Salón 8
7	Sala de usos múltiples
9	Salón 3

Tabla 5.1.4.8 – Tabla salones.

- Tabla grupo

Grupo	Ciclo
9004	20161
9011	20161
9002	20162
9021	20161
9005	20162
9013	20162
9011	20162
9009	20162

Tabla 5.1.4.9 – Tabla grupo.

4.2 Diseño e Implementación de la Base de datos

El primer paso se inicia con la descarga de la paquetería XAMPP de la siguiente liga:

[“https://sourceforge.net/projects/xampp/files/XAMPP%20Windows/5.6.8/”](https://sourceforge.net/projects/xampp/files/XAMPP%20Windows/5.6.8/)³².

La instalación es totalmente automática, el paquete realizará todas las operaciones necesarias para instalar el software en el equipo con la participación del usuario como se muestra en la figura 4.2.1

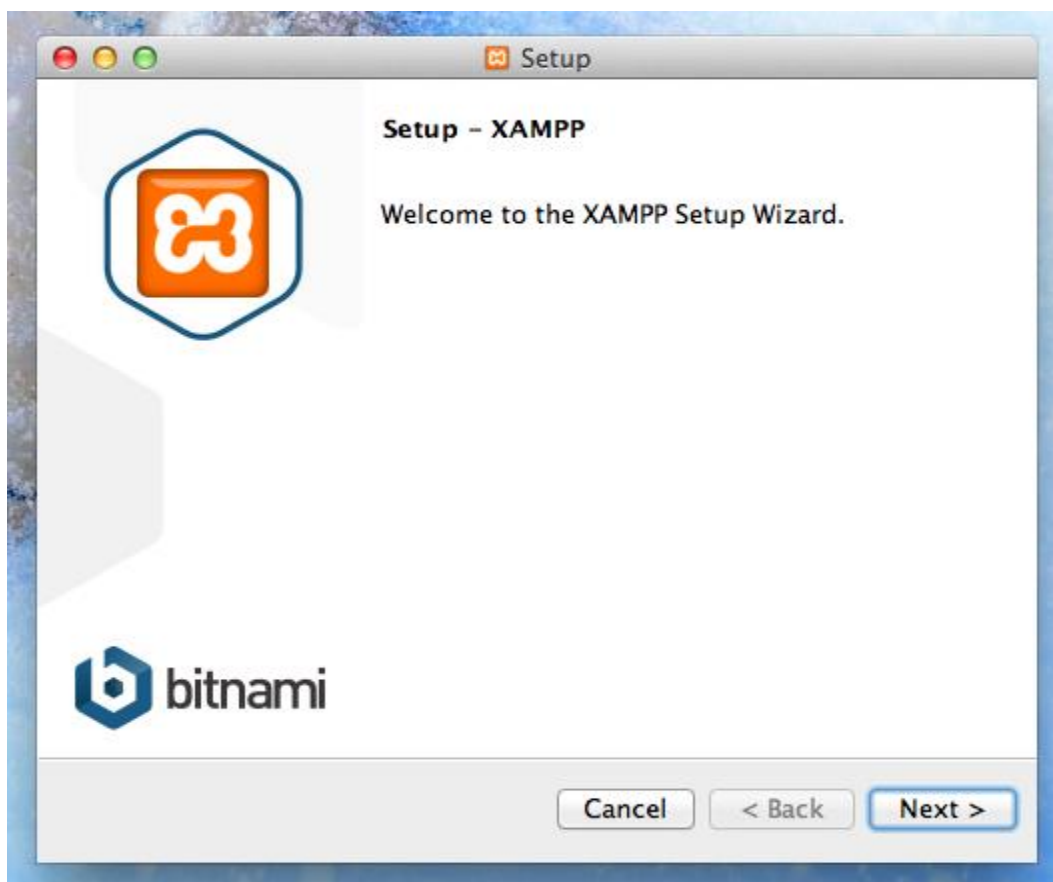


Figura 4.2.1 – XAMPP.

³² Sourceforge (2016). Download. Recuperado el 13 de mayo de 2016, de <https://sourceforge.net/projects/xampp/files/XAMPP%20Windows/5.6.8.>



Al terminar la instalación, se muestra una pantalla donde se indica que el proceso ha finalizado (figura 4.2.2), o en caso contrario, si llegasen a existir anomalías nos mostraría una pantalla con el error.

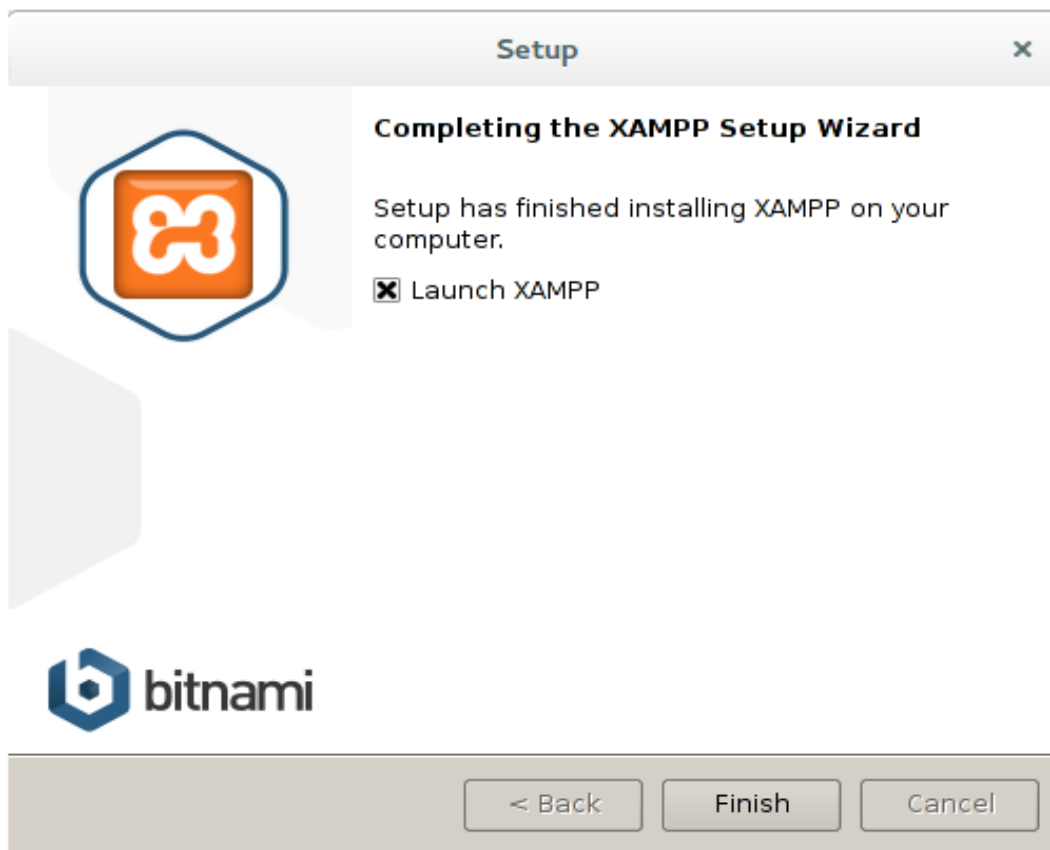


Figura 4.2.2 – Fin de la Instalación.

En la figura 4.2.3 se muestra el panel de control de XAMPP, el cual nos apoya para poder levantar los servicios de Apache y de MySQL así como los puertos habilitados respectivamente para cada servicio.

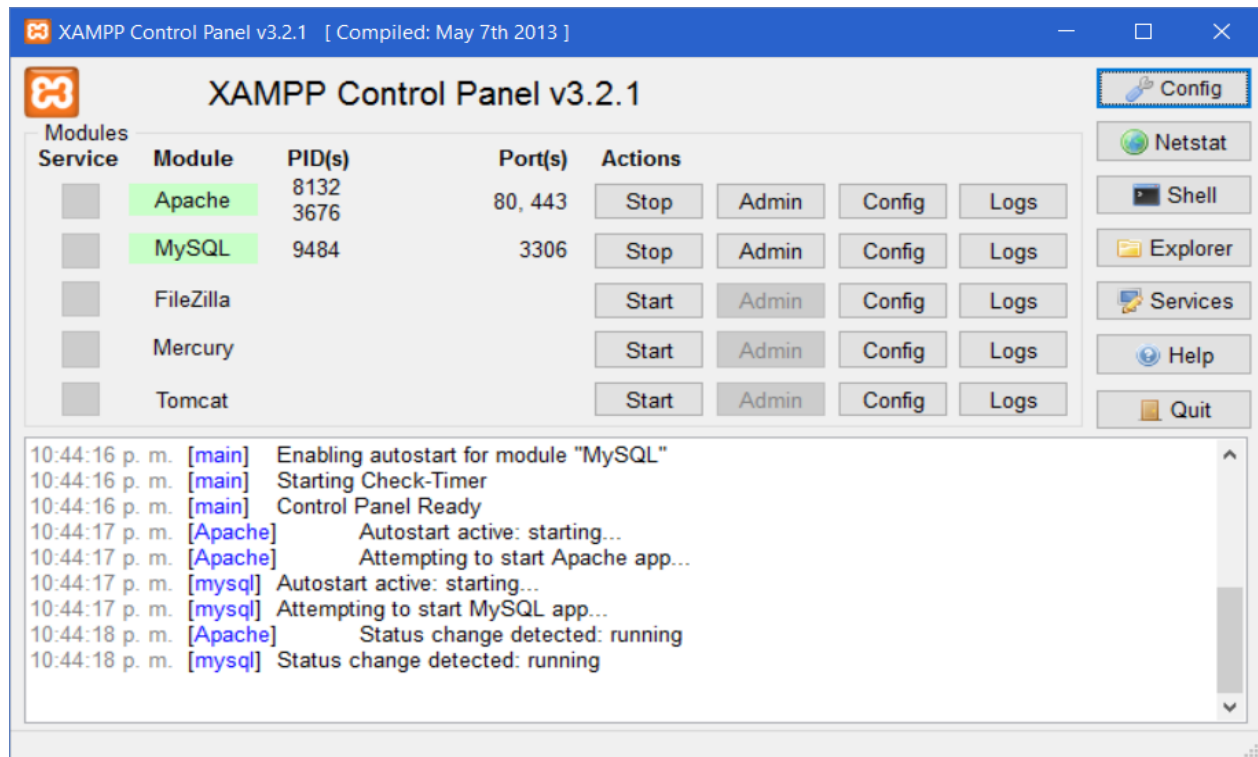


Figura 4.2.3 – Levantar Sesión.

En la figura 4.2.4 podemos comprobar que el servicio de Apache está funcionando correctamente.



Figura 4.2.4 – Apache.



Creación de la Base de Datos

Para la creación de la Base de Datos se ejecuta la herramienta phpMyAdmin la cual nos proveerá tanto el ambiente como las herramientas necesarias para gestionar la Base de Datos.

Esto mediante una interfaz muy sencilla de operar y la cual es muy amigable para el programador de la Base de Datos.

Para la creación de una nueva base de datos en phpMyAdmin escribimos el nombre de la nueva base de datos en el cuadro de texto que aparece abajo **Crear base de datos** en la página principal y pulsar el botón **Crear**.

La opción de **Cotejamiento**, si el valor se deja en blanco tomará el valor por defecto que variará según la configuración de cada servidor.

Como se muestra en la figura 4.2.5

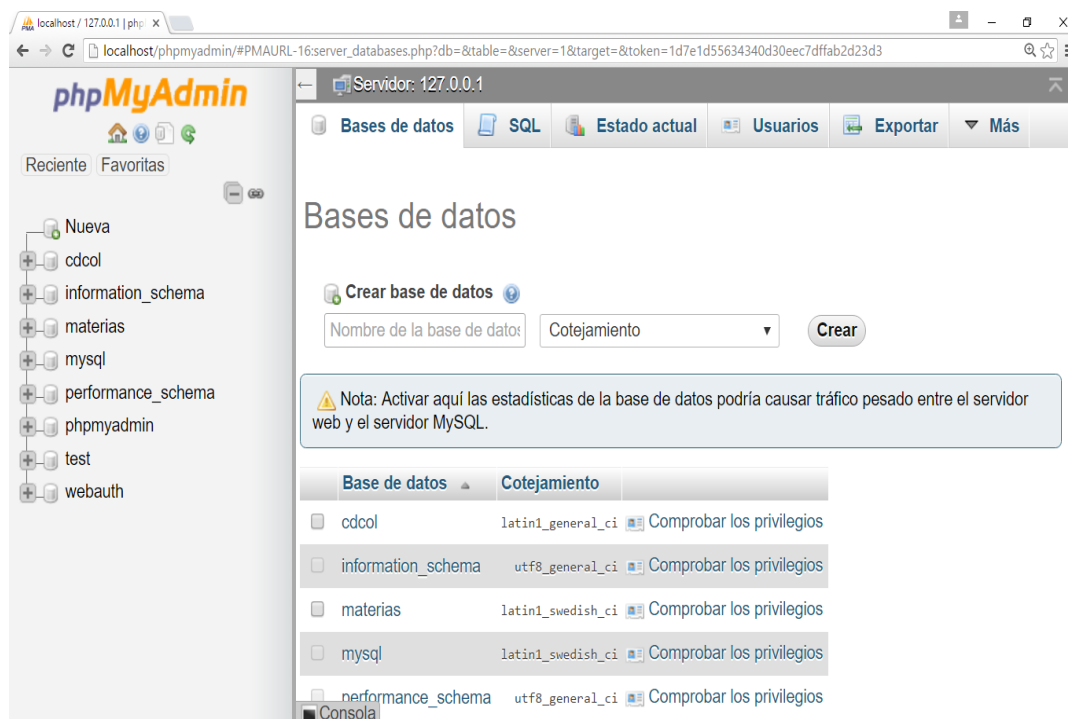


Figura 4.2.5 – Creación Base.



Si se desea crear la base de datos mediante la sintaxis SQL, se emplea el siguiente código básico:

```
CREATE DATABASE nombre de la base  
ON  
(NAME = nombre de la base_data,  
FILENAME='ubicación.mdf',  
SIZE=4  
GO
```

- **CREATE DATABASE.** Se utiliza para crear una nueva base de datos vacía.
- **ON.** Especifica que los archivos de disco utilizados para almacenar las secciones de datos de la base de datos
- **NAME.** Es el nombre lógico para el archivo
- **FILENAME.** Es la ruta de acceso y el nombre de archivo que el sistema utiliza para cuando se crea el archivo.
- **SIZE.** Especifica el tamaño del archivo.

Desde el punto de vista de SQL, una base de datos es sólo un conjunto de relaciones (o tablas), y para organizarlas o distinguirlas se accede a ellas mediante su nombre. A nivel de sistema operativo, cada base de datos se guarda en un directorio diferente.

En el momento de crearla, la base de datos estará vacía, es decir, no contendrá ninguna tabla.

Creación de Tablas

Para crear una tabla lo único que se debe hacer es escribir **Nombre** y especificar **Número de campos**.



Con esa información definida damos click al botón de **Continuar** y se mostrará una pantalla como en las figuras 4.2.6 y 4.2.7

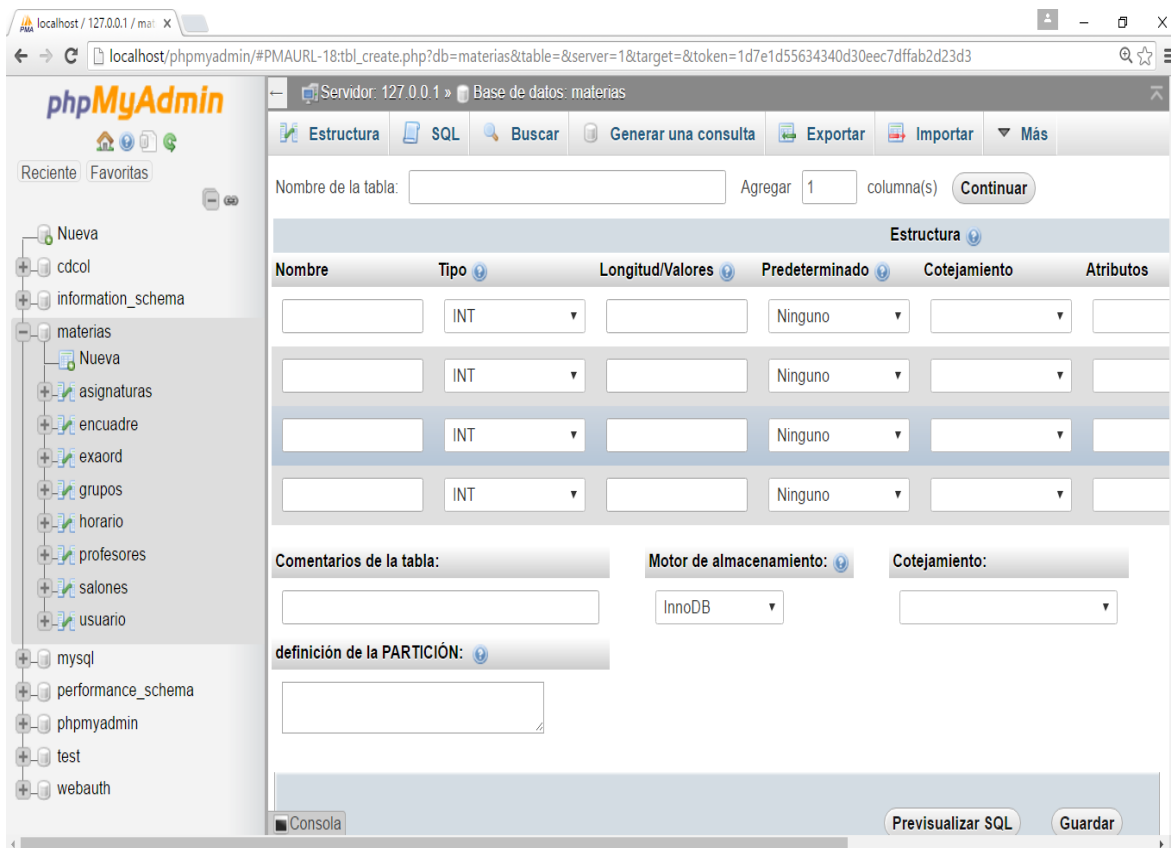


Figura 4.2.6 – Creación Tabla.

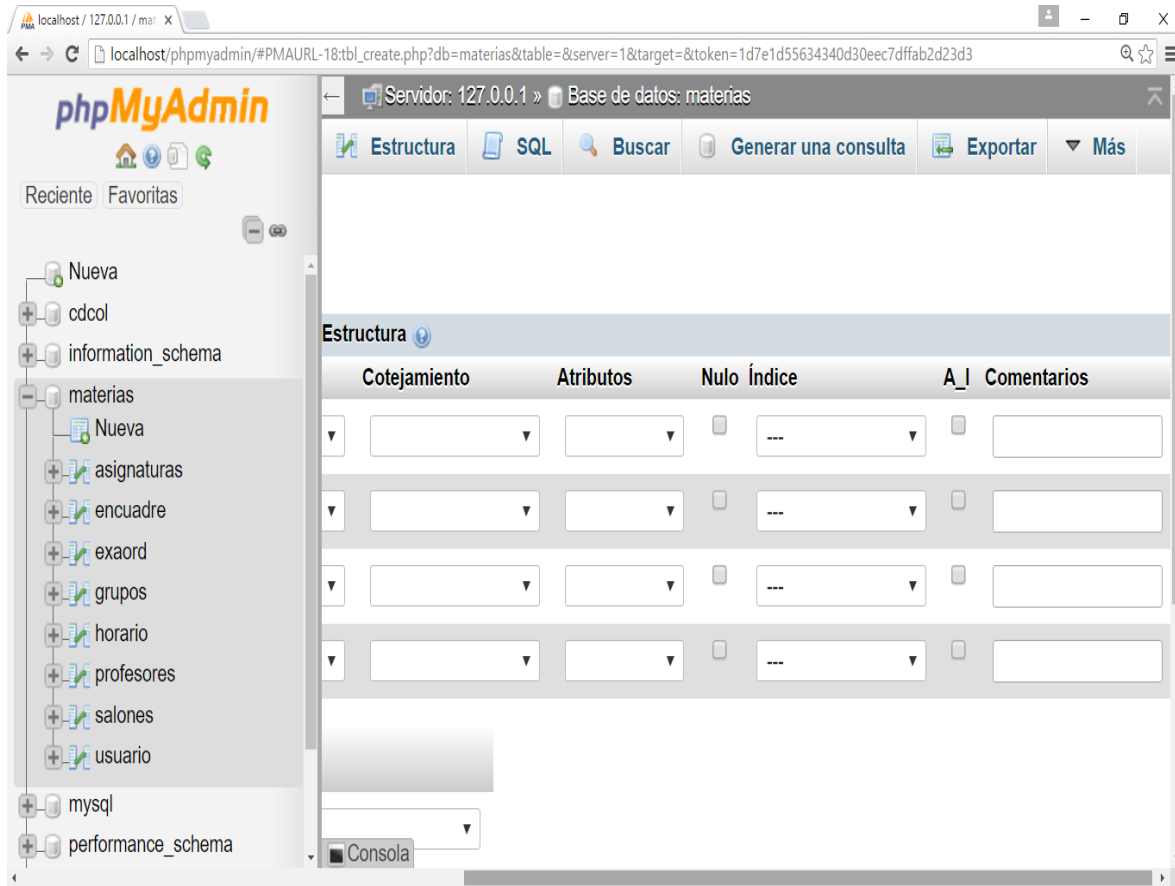


Figura 4.2.7 – Creación Tabla.

Aquí se deberá de establecer la configuración de cada uno de los campos de la tabla.

A continuación, se detalla cada uno de los datos requeridos para la creación de la Tabla:

- **“Campo:** El nombre del campo”³³.
- **Tipo:** El tipo del valor del campo: texto (varchar o text), número entero (int), decimal (decimal), etc.
- **Longitud/Valores Tamaño máximo del campo.** Un texto de tamaño 15 sólo podrá tener 15 caracteres.

³³ Servidor de pruebas (2012). Introducción a phpMyAdmin. Recuperado el 15 de mayo de 2015, de http://www.aulaclic.es/dreamweaver-cs5/t_20_9.htm.



- **Predeterminado:** Un valor predeterminado para el campo, si al crear el registro no se le asigna otro valor.
- **Cotejamiento:** Juego de caracteres que utilizará el campo.
- **Atributos:** Dependientes del tipo de dato. Por ejemplo, que un numérico solo sea positivo.
- **Nulo:** Si el campo permite valores nulos, o debe llenarse obligatoriamente.
- **Índice:** Indica si el campo está indexado. Puede ser llave principal, valores únicos, etc.
- **A_I:** Si el campo es un auto-incremento, que se establecerá automáticamente al crear nuevos registros.
- **Comentarios:** Nos permite agregar comentarios descriptivos del campo o su contenido.

Para la creación una nueva tabla mediante Código, es requerido especificar, los campos y los atributos para la tabla.

```
CREATE TABLE nombre detabla  
(Idint NOT NULL,  
campo1 atributo1,  
campo2 atributo2,  
... ;  
ALTERTABLE nombre tabla ADD PRIMARY KEY (id);
```

NOTA: Esto va en concordancia con lo especificado en cada una de las tablas del diccionario de datos.

Generación de Consultas

La función de las consultas se puede realizar mediante un modo gráfico (figura 4.2.8). Las consultas son las acciones principales para poder acceder a la información contenida en las tablas de las Bases de datos.

Con las consultas se puede modificar, borrar, mostrar y agregar datos en una base de datos.

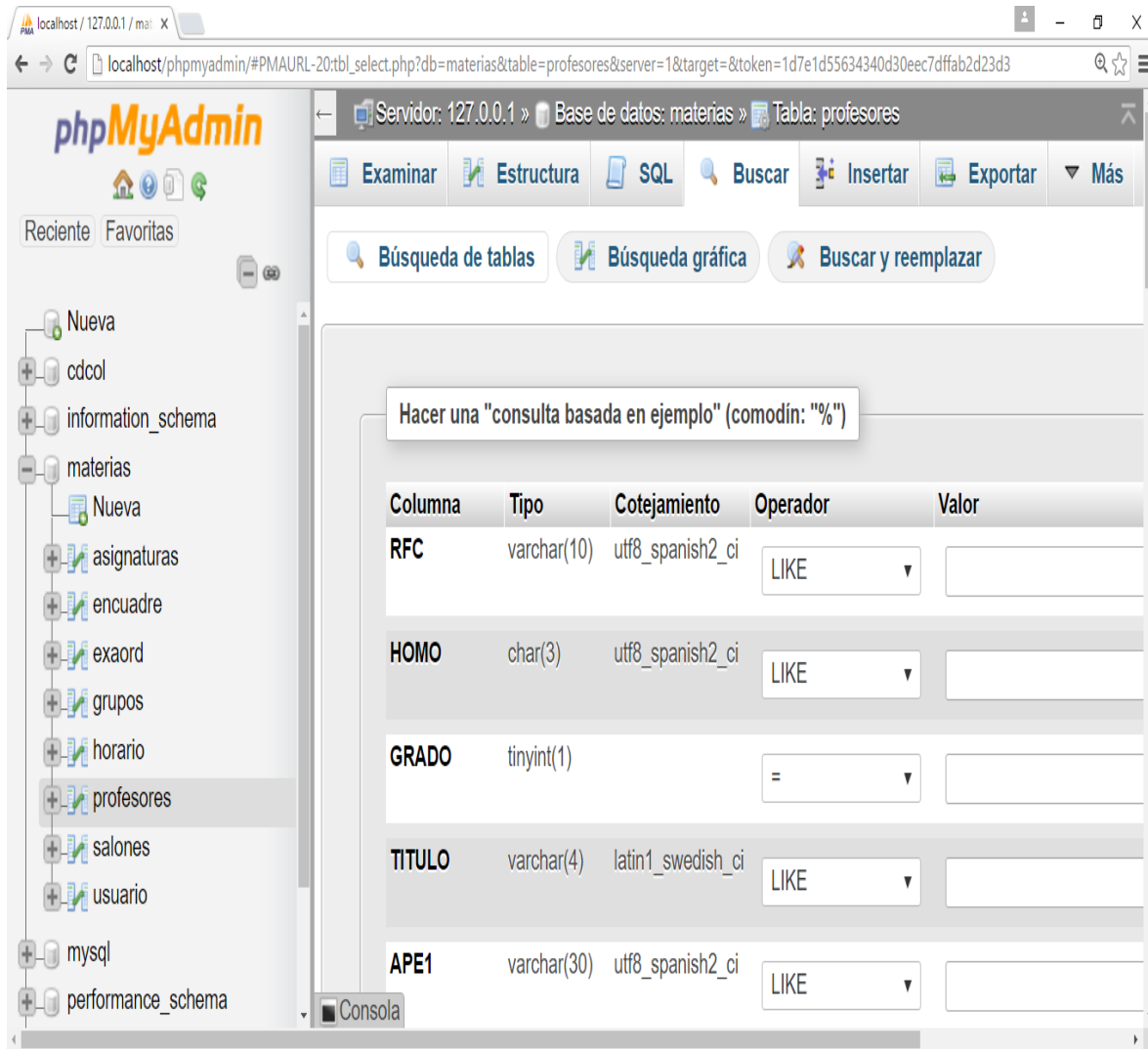


Figura 4.2.8 – Consulta.

Puede usarse la herramienta gráfica y seleccionar los tipos de datos para cada campo (figura 4.2.9).

Se hace clic en el botón SQL directamente desde la vista de la tabla. Esto mostrará una ventana de consultas SQL que se habrá llenado parcialmente.

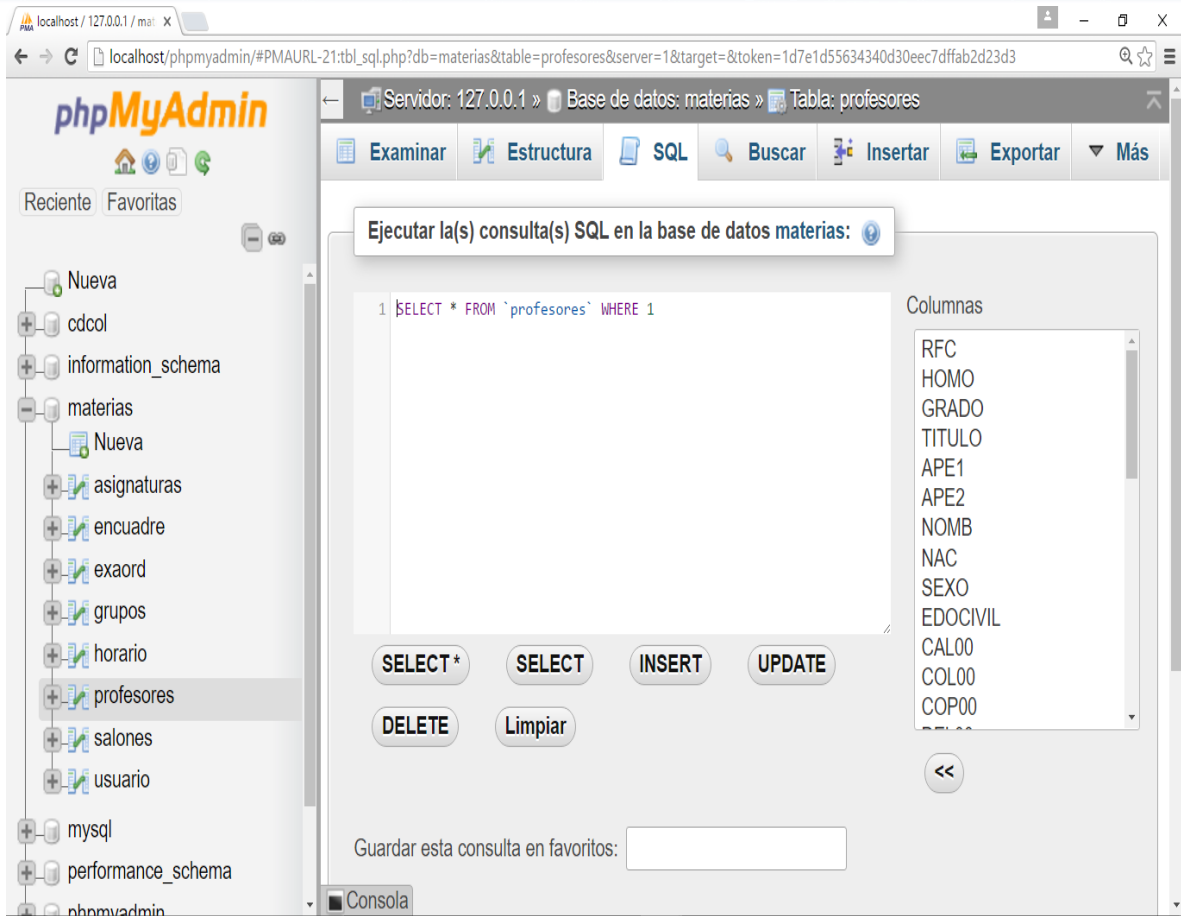


Figura 4.2.9 – Ejecución Consulta.

Mediante código sería de la siguiente forma.

```
SELECT nombre_campo  
FROM nombre_tabla  
WHERE nombre_campo = 'texto_a_buscar'  
ORDER BY nombre_campo
```

Árbol de Tablas

A continuación, en las figuras 4.2.10, 4.2.11, 4.2.12 y 4.2.13, se muestran las tablas generadas para cada uno de los casos mencionados.

Tabla de Grupos

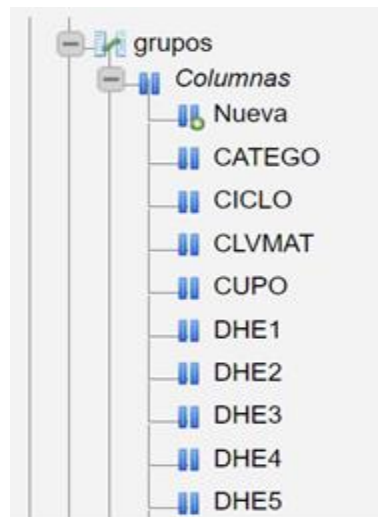


Figura 4.2.10 – Grupos.

Tabla de Horarios



Figura 4.2.11 – Horarios.



Tabla Profesores

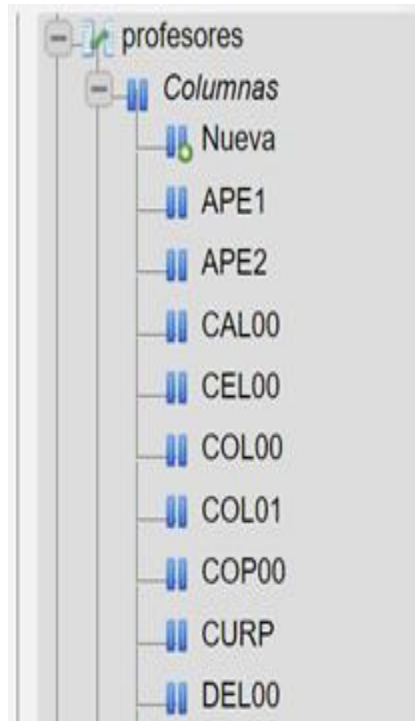


Figura 4.2.12– Profesores.

Tabla Salones

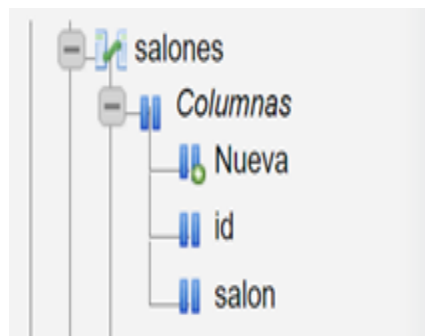


Figura 4.2.13 – Salones.



4.3 Construcción y diseño del front-end.

“La interfaz gráfica de usuario son todos los elementos gráficos que nos ayudan a comunicarnos con un sistema. El diseñador de la interfaz gráfica cumple una parte fundamental para aterrizar la ideal central, materializarla y construir los mensajes que permiten la comunicación usuario sistema. Para ello, necesita partir de una base teórica y profundizar en ella para discernir cada uno de los elementos que la conforman y dar sus propias soluciones. De la capacidad que se tenga, como diseñador de interfaces, para captar, interpretar y traducir, esta información a códigos visuales e incluso auditivos que el usuario reconozca, dependerá el éxito que tenga el proyecto, es decir, si la interacción, facilidad de uso y manejo del mensaje visual, logran la inmersión e interacción del usuario con la interfaz, se cumple el objetivo de la misma”³⁴.

Una interfaz de usuario mal diseñada provocará que los usuarios no puedan explotar el total de las características del software, cometerán errores y sentirán que el sistema en lugar de ayudarles les dificulta realizar su trabajo e impide lograr los objetivos para los que fue creado. “Algunos factores importantes a considerar son los siguientes:

- Reducir la demanda de memoria. Las personas tenemos una memoria limitada a corto plazo, por lo tanto si a los usuarios se les presenta demasiada información al mismo tiempo es posible que no puedan asimilarla.
- Familiaridad del usuario. Los usuarios no deben ser obligados a adaptarse a la interfaz sólo porque sea conveniente implementarla, la interfaz debe utilizar términos familiares para los usuarios, y los objetos que el sistema manipula deben estar directamente relacionados con el entorno de trabajo del usuario.
- Uniformidad. Siempre que sea posible, las operaciones comparables se deben activar de la misma forma.
- Mínima sorpresa. Es apropiado debido a que las personas se irritan demasiado cuando el sistema se comporta de manera inesperada” .³⁵

³⁴ Luna González, L (2004). El diseño de interfaz gráfica de usuario para publicaciones digitales, UNAM. Recuperado el 16 de mayo de 2016, de http://www.revista.unam.mx/vol.5/num7/art44/ago_art44.pdf.

³⁵ Sommerville, Ian. (2005). Ingeniería del Software. México: Pearson Educación.



Generalmente la facilidad de uso de una interfaz gráfica no depende de elementos estéticos o modernos, ésta ocurre cuando la arquitectura de la interfaz corresponde a las necesidades de las personas que la usarán, es por eso que el diseño de la interfaz pretende apegarse a los requerimientos por parte del cliente, principalmente en la definición de las tareas que los usuarios finales deben realizar para el correcto desempeño de sus labores y el contenido que es necesario presentar como parte de la interfaz.

A continuación se dará una explicación de algunas de las pantallas que conforman la interfaz de usuario del sistema planteado, como solución para la problemática detectada en la realización de horarios para la modalidad de educación abierta y a distancia.

El sistema se divide en cinco módulos, como ya se había planteado en el capítulo 3.5 de este trabajo de tesis, los cuales son:

- Autenticación
- Búsqueda
- Modificaciones al horario
- Usuario
- Salón

Cada uno de estos módulos es utilizado por los usuarios finales. La pantalla de inicio corresponde al módulo de autenticación, contiene un formulario típico de inicio de sesión, el cual está conformado por los siguientes elementos básicos como se muestra en la figura 4.3.1:

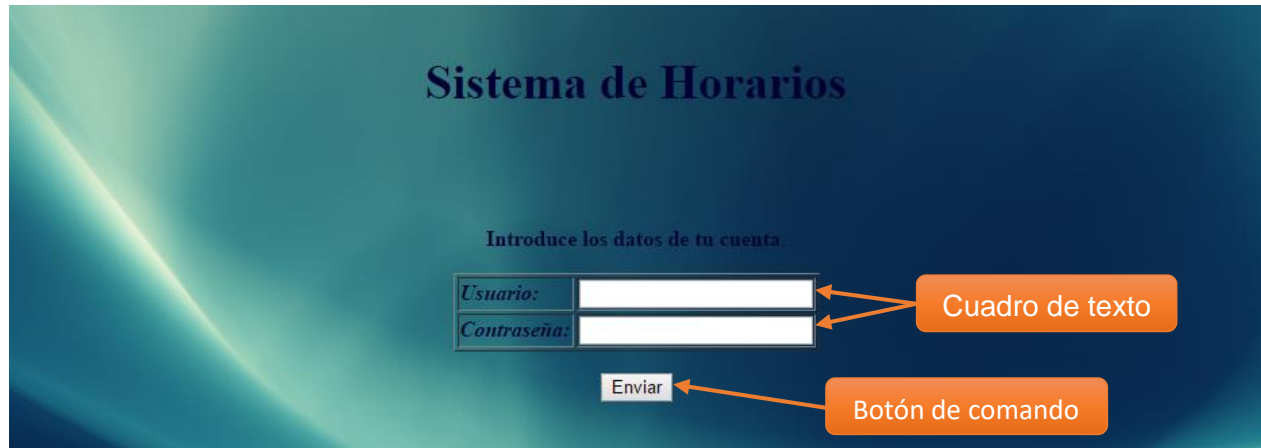


Figura 4.3.1 – Página de inicio.

- Cuadro de texto. Es un control de entrada que permite al usuario escribir texto que será utilizado por el programa.
Al cuadro de texto destinado para introducir la contraseña, se le aplica la propiedad de no visualizar la captura.
- Botón de comando: Está asociado a eventos que se ejecutan al dar clic sobre él, ya sean de tipo procedimientos, tareas o secuencias de programación.

Una vez validado el nombre de usuario y la contraseña se presentará la pantalla principal, la cual es similar en todos los módulos; sin embargo, pueden variar las opciones en el Menú como se muestra en la figura 4.3.2

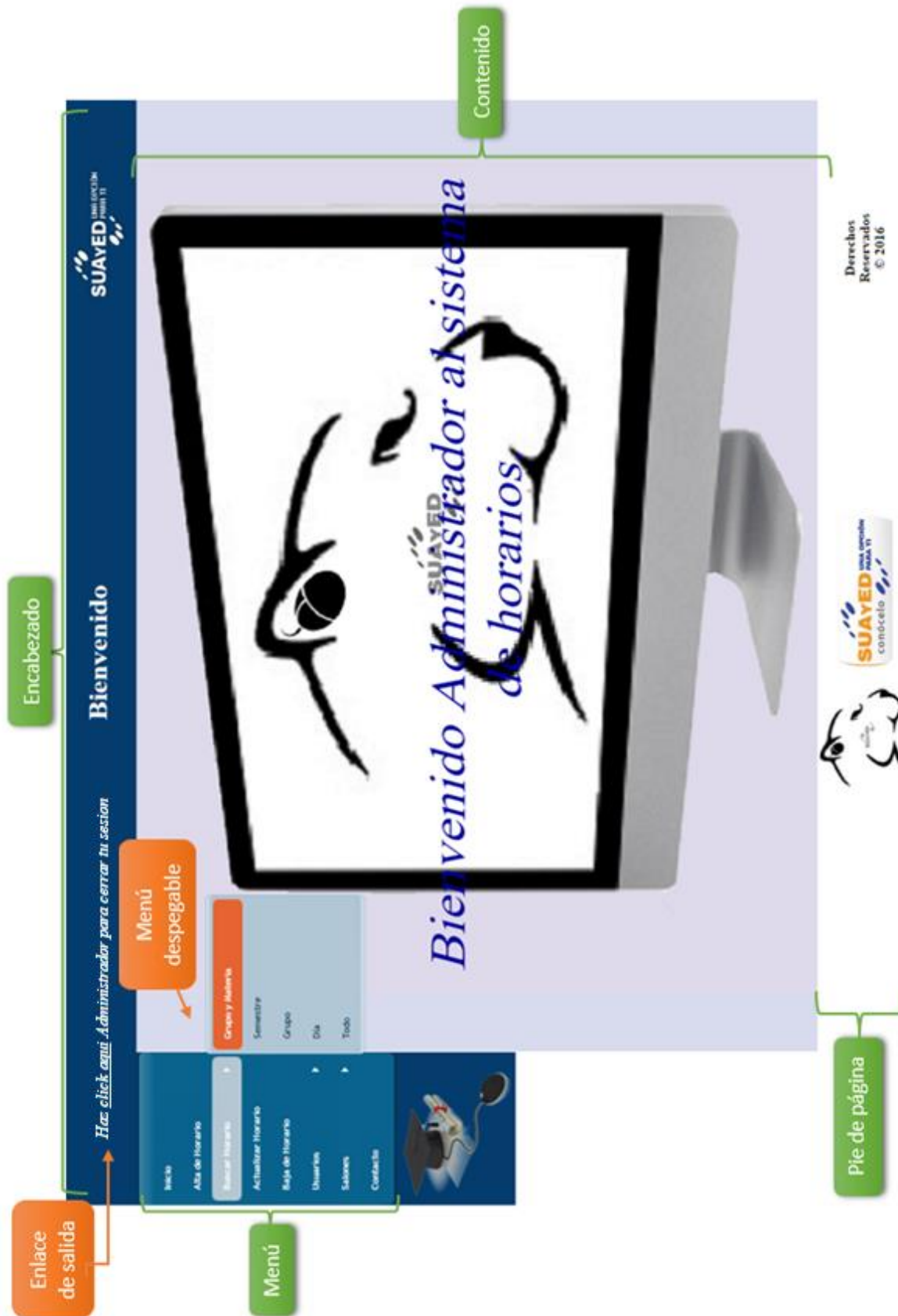


Figura 4.3.2 – Página de principal




En la pantalla principal se encuentran todas las opciones que existen en cada módulo. Dentro de los elementos que conforman la pantalla principal están:

- Encabezado
 - Se tiene un enlace de salida con el cual nos permite salir de la aplicación, cerrando la sesión de usuario y nos regresa a la página de inicio.
 - Se muestra el nombre del usuario con el cual se ingresó.
 - En la parte media del encabezado se muestra el área en la cual se está trabajando para la comodidad del usuario.
 - Se tiene también el logo de la modalidad de educación abierta y a distancia “SUAYED”.

- Contenido

En esta área se desplegarán todos los formularios con los que se cuenta en la aplicación, como son alta, baja o modificación de horarios, salones, usuarios o la búsqueda de algún horario.

- Menú
 - Se muestran las diferentes funciones que se pueden realizar para la elaboración de horarios, como son: Alta, baja o búsqueda de un horario.
 - Se cuenta con menús despegables los cuales nos muestran las diversas opciones que tiene una función, éstas se identifican por el símbolo .

- Pie de página
 - Se observan los logos de la institución educativa y de la modalidad de educación abierta y a distancia.
 - Se muestra los derechos reservados y el año en que se realizó la aplicación.
 - Este módulo no cambiará durante la navegación del sitio web.

Si se requiere realizar el alta de un horario se escoge la opción de alta en el menú para que nos muestra la siguiente pantalla, como se observa en la figura 4.3.3.

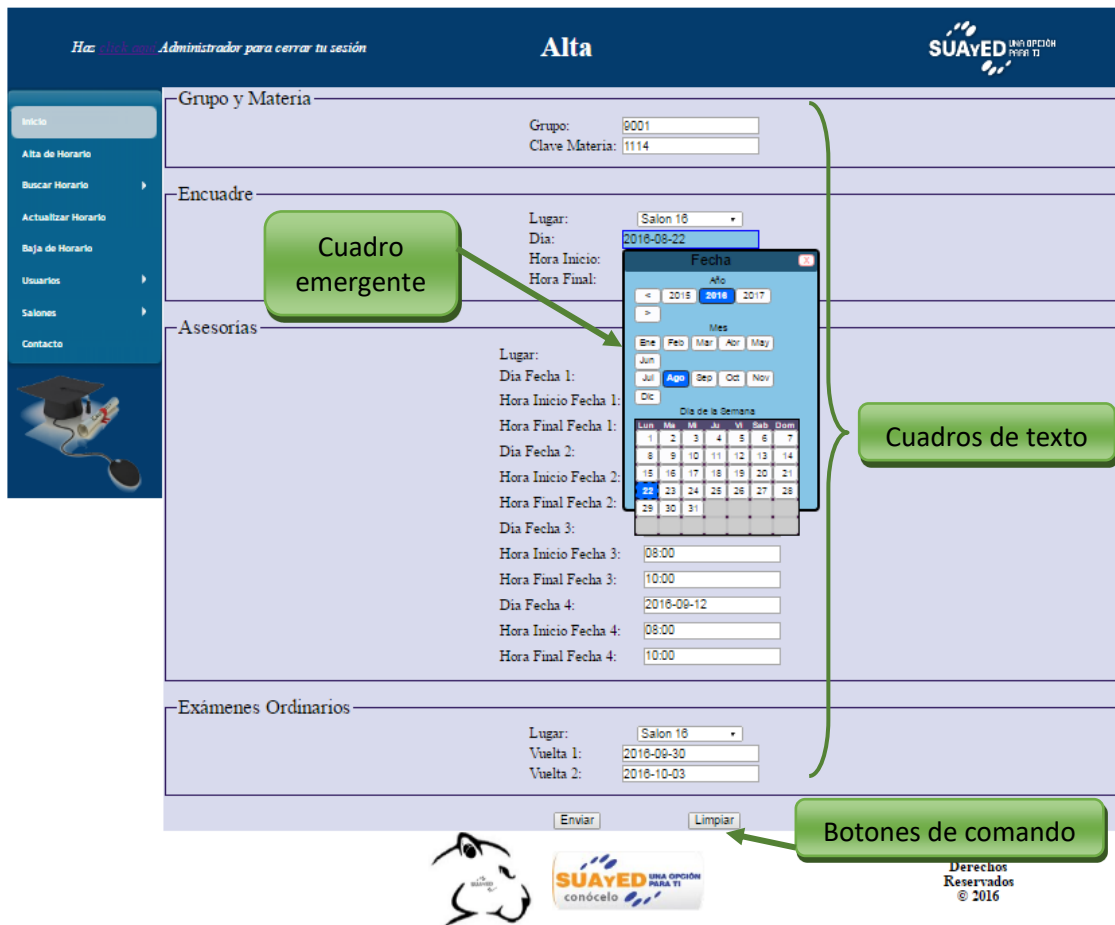


Figura 4.3.3 Pantalla de alta de horarios.

En la pantalla de alta de horarios se cuenta con varios cuadros de texto, en los cuales se debe de capturar los datos necesarios, como son el grupo, materia, salones, horarios y días, con los que se realizará la asignación de horarios en el sistema.

En los campos de salones de desplegará un listado con los salones que están asignados para que se lleven a cabo las sesiones.

Al ser seleccionados los campos de fecha y hora, emerge un cuadro en el que se muestra calendario o reloj, según corresponda, para que se elijan los datos que requiere el usuario para ser dados de alta. Estos campos realizan las siguientes validaciones:

- Que la fecha de una sesión no tenga una fecha menor a la de la sesión anterior.
- Que no se encuentre ya dada de alta una sesión de otro grupo en el mismo salón, día y horario.

Si ocurre un error al efectuarse estas validaciones se mostrara un mensaje de error en la parte superior del segmento.

El cuadro de comando nos permitirá realizar la alta del horario asignado en la base de datos y el de borrará todos los campos de textos en los que se haya proporcionado información.

Una vez que se capturaron todos los datos necesarios y se selecciona el botón de comando “enviar” se muestra la pantalla de que el alta fue exitosa, como se observa en la figura 4.3.4, en el caso de que hubiera un error en el alta, como pudiera ser la duplicidad de grupo y materia, se mostrará el mensaje de error en esta misma pantalla como se muestra en la figura 4.3.5.



Figura 4.3.4 – Pantalla de confirmación de alta.



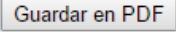
Figura 4.3.5 – Pantalla de error en el alta.

Para realizar búsquedas de horarios se escoge en el menú la opción de búsqueda y se elige el criterio por el cual se quiere realizar como se muestra a continuación en la figura 4.3.6.



Figura 4.3.6 – Pantalla de búsqueda.



Una vez que se eligió el criterio de búsqueda en el módulo de contenido aparecerán cuadros de texto en los que se debe de colocar la información solicitada que será enviada a la base de datos para que muestre la información como se ejemplifica en la figura 4.3.7 y la cual es la forma en la que se plasmará en formato PDF si se elige el botón de comando  como se ve en la figura 4.3.8.

Los módulos de modificación de horarios, usuario y salones se efectúan de la misma manera que los módulos de búsqueda y alta de horarios.

Haz [click aquí](#) Administrador para cerrar tu sesión

Buscar

SUAYED UNA OPCIÓN PARA TI

Grupo	Clave Asignatura	Asignatura	Título	Primer Apellido	Segundo Apellido	Nombre Lugar	Encuadre			Asesorías						Exámenes Ordinarios					
							Inicio	Final	Lugar	1	2	3	4	5	6	7	8	9	10		
9002	2210	DESARROLLO HISTÓRICO DEL TRABAJO SOCIAL	MTRA	SAVONEZ ZEPEDA	ANOREA ABOYA	Salón 15	27	15:00	16:00	03	03	07:00	08:00	10	17	24	24	12:00	14:00	02	08
9002	2222	TEORÍA SOCIAL I	MTRA	GONZALEZ ULLON	RABLO AGUIRRE ABRANDO	Salón 32	22	15:00	16:00	30	30	08:00	09:30	08	23	20	20	10:30	14:30	27	04
9002	2222	TEORÍA ECONOMICA I	LIC	ORTA	NORMEZA ANTONIO	Salón 19	17	09:30	10:30	24	24	13:27	15:30	21	07	14	14	12:30	16:30	21	28
9002	2222	SITUACION INTERNACIONAL CONTEMPORANEA	LIC	VALERO	KAROL AIDA	Salón 15	11	15:52	17:52	18	18	13:30	16:32	25	01	08	08	12:30	16:30	22	29
9002	2214	ANÁLISIS DEL ESTADO MEXICANO	MTRA	SANTIBANES VELEZ	VICTOR MANUEL	Salón 16	12	14:18	17:18	19	19	14:18	17:18	26	04	04	14:18	16:18	16	23	

Regresar **Botón de comando**

Derechos Reservados © 2016

Figura 4.3.7 – Resultado de búsqueda.



Horarios SUAYED

Reporte

Grupo	Cb/Asig	Asig	Título	1er Ape	2o Ape	Nombre	Lugar	Da	Hora Inicio	Hora Final	Lugar	Da 1	Hora Inicio 1	Hora Final 1	Da 2	Hora Inicio 2	Hora Final 2	Da 3	Hora Inicio 3	Hora Final 3	Da 4	Hora Inicio 4	Hora Final 4	Lugar	Fecha 1	Fecha 2
9001	1110	DESAR ROLLO HISTORICO TRABAJO SOCIAL	MTRA SANCHEZ	ZEPEDA A	ANDREA KENYA	Salon 16	27	15:00	16:00	Salon 15	03	07:00	08:00	2016-06-10	09:00	10:00	2016-06-17	11:00	12:00	2016-06-24	13:00	14:00	Salon 15	2016-06-01	2016-07-08	
9001	1111	TEORIA SOCIA L1	MTRG GONZALEZ ULLAO	AGUIRRE NORIEGA	PABLO ARBA NDO	Salon 32	23	15:00	16:00	Salon 15	30	08:00	09:30	2016-06-06	10:00	11:30	2016-06-13	10:30	12:00	2016-06-20	13:00	14:30	Salon 15	2016-06-27	2016-07-04	
9001	1112	TEORIA ECON OMICA I	LIC ORTGA	NORIEGA	JOSE ANTON HO	Salon 15	17	09:30	10:30	Salon 15	24	13:37	15:38	2016-06-31	13:38	16:38	2016-06-07	13:38	16:38	2016-06-14	13:38	16:38	Salon 15	2016-06-21	2016-06-28	
9001	1113	SITUACION INTERNACIONAL DE LA ECONOMIA	LIC OCCHOA	VALERO	KAROL ADRA	Salon 15	11	13:52	17:52	Salon 15	18	13:50	16:51	2016-06-25	13:51	18:51	2016-06-01	13:51	18:51	2016-06-08	13:51	16:51	Salon 16	2016-06-22	2016-06-29	
9001	1114	ANALISIS DEL ESTADO ECONOMICO	MTRG SANTIBANZ	VELEZ	VICTOR RAMANUEL	Salon 16	12	14:19	17:19	Salon 16	19	14:19	17:19	2016-06-26	16:19	17:19	2016-06-02	14:19	16:19	2016-06-09	15:19	18:19	Salon 16	2016-06-16	2016-06-23	

Figura 4.3.8 – Formato PDF de búsqueda.



4.4 Pruebas y mantenimiento

Pruebas de software

Las pruebas de software se definen como "el proceso de analizar un elemento de software para detectar diferencias entre las condiciones existentes y las requeridas y para evaluar las características de los elementos de software".³⁶

- Pruebas de caja blanca

Las pruebas de caja blanca se enfocan en la estructura de control del programa. Este tipo de pruebas realizan un seguimiento del código fuente de manera que se determinan las instrucciones, bloques, etc., en los que existan errores.

- Pruebas de caja negra

El objetivo de las pruebas de caja negra es validar los requerimientos funcionales del programa sin considerar el funcionamiento interno. Se enfocan en el dominio de información del software.

- Pruebas de regresión

Cuando se genera una modificación sobre software que ya fue probado, existe la posibilidad de introducir nuevos defectos. Por esto, es necesario volver a ejecutar los casos de prueba, que fueron diseñados y ejecutados anteriormente, sobre el código modificado.

- Pruebas de integración

Las pruebas de integración consisten en un proceso mediante el cual se construye la arquitectura del software mientras se llevan a cabo pruebas para encontrar errores asociados con la interfaz.

- Prueba de módulos

Las pruebas unitarias se centran en el módulo, que es la menor unidad de diseño del software. El objetivo es detectar errores en cada uno de los módulos del software al ser ejecutados de manera independiente del resto de los componentes. Estas pruebas se

³⁶ Isabel Ramos Román, José Javier Dolado Cosín (2007). Técnicas cuantitativas para la gestión en la ingeniería del software. España: Netbiblo.



suelen ejecutar por el desarrollador que construye el módulo. Están centradas en los aspectos estructurales.

- Pruebas de estrés

El objetivo de esta prueba es determinar la solidez de la aplicación en momentos de carga extrema, ayuda a los administradores a determinar si en caso de sobrecarga, la aplicación rendirá lo suficiente. Estas pruebas permiten tomar decisiones sobre configuraciones de hardware, ajustes de software y selección de arquitecturas.

- Pruebas alfa

La prueba alfa se lleva a cabo en un ambiente controlado, el desarrollador está presente y registra los errores y problemas que experimentan los usuarios. El software se usa en un escenario natural, la prueba se lleva a cabo por un grupo representativo de usuarios finales.

- Pruebas beta

La prueba beta no se realiza en un ambiente controlado: el desarrollador no está presente. El cliente registra todos los problemas, ya sean reales o imaginarios, que se encuentran durante la prueba y los reporta periódicamente al desarrollador.

Pruebas aplicadas al sistema

Entre las pruebas que se aplicaron al sistema, y que se documentan en este escrito, se encuentran pruebas de caja negra y de integración. Respecto a las pruebas de caja negra, la primera corresponde a la entrada de autenticación, en la figura 4.4.1 se ve la pantalla inicial que se muestra al ejecutar el sistema.



Figura 4.4.1 – Pantalla inicial del sistema.

Se ingresan datos incorrectos para validar el proceso de autenticación, estos datos pueden verse en la figura 4.4.2.

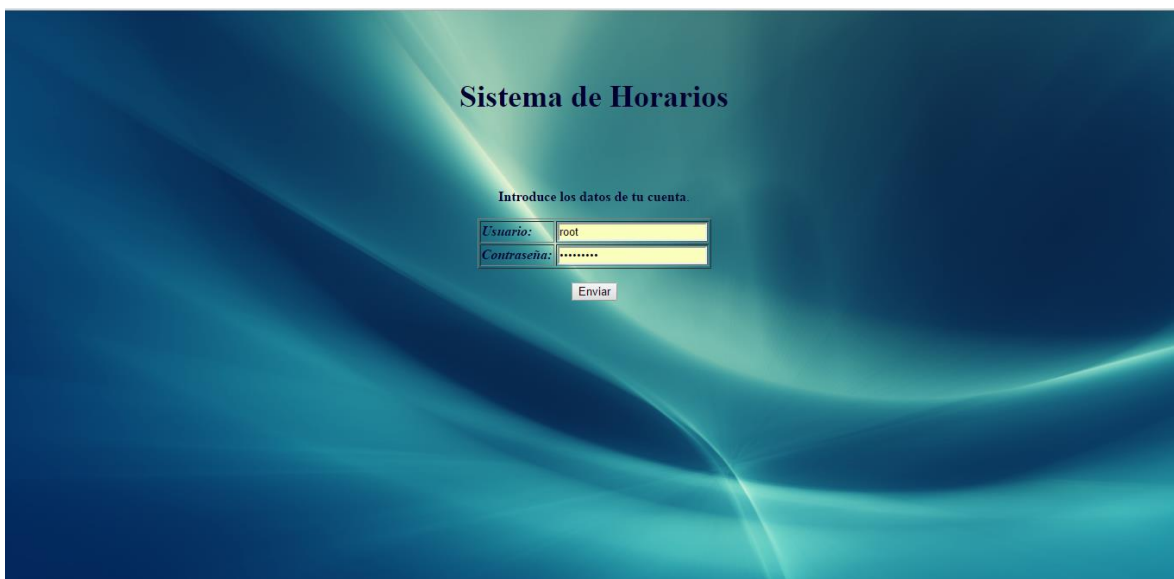


Figura 4.4.2 – Ingreso de usuario y contraseña incorrectos.

Al ingresar datos incorrectos, el sistema regresa al usuario a la pantalla inicial, la cual puede verse en la figura 4.4.1.

Ahora se ingresan datos correctos, un usuario y contraseña registrados (figura 4.4.3), se da click en el botón Enviar y el sistema da acceso al usuario mostrando el menú principal (figura 4.4.4).

Sistema de Horarios

Introduce los datos de tu cuenta

Usuario:	spawn
Contraseña:	*****

Figura 4.4.3 – Ingreso de usuario y contraseña registrados en la base de datos.



Figura 4.4.4 – Despliegue de la pantalla principal del sistema, dando la bienvenida al usuario registrado.

La segunda prueba de caja negra valida el proceso de búsqueda, en el primer caso las entradas son grupo y materia registrados en la base de datos. Se ingresan los datos en



el formulario correspondiente (figura 4.4.5) y al realizarse la búsqueda se obtiene el despliegue en pantalla del horario (figura 4.4.6).



Figura 4.4.5 – Ingreso de los datos para búsqueda.



Figura 4.4.6 – Despliegue del horario encontrado.

En el segundo caso, se ingresan datos erróneos en el formulario de búsqueda (figura 4.4.7), al realizar la búsqueda de horario, se despliega en pantalla un mensaje que le indica al usuario que para esa materia y grupo no hay un horario registrado (figura 4.4.8).



Figura 4.4.7 – Ingreso de datos erróneos.

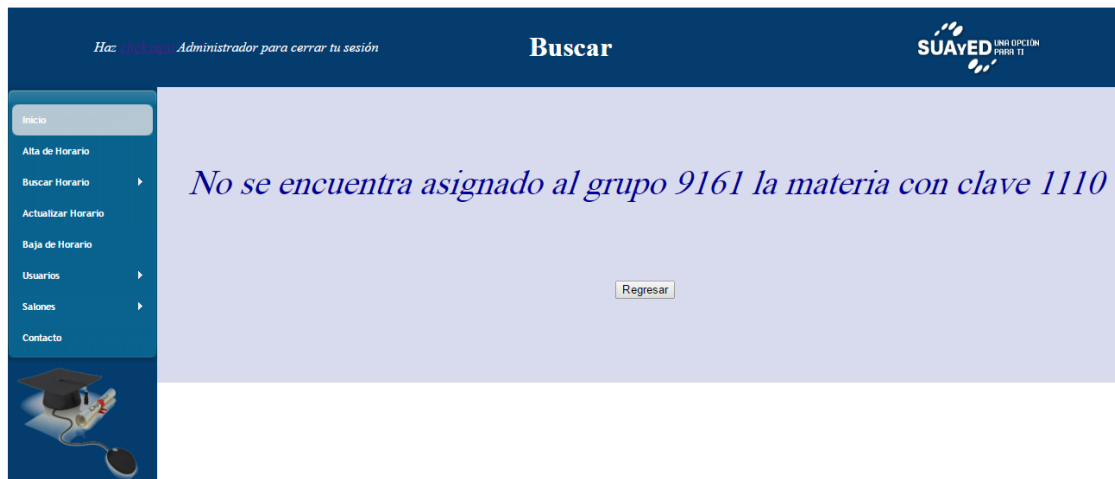


Figura 4.4.8 – Despliegue del mensaje de error.

Otra prueba que se realizó fue una de integración. El proceso fue el siguiente: Se seleccionó un grupo (figura 4.4.9) y se visualizó el horario ya registrado (figura 4.4.10).



Derechos Reservados © 2016

Figura 4.4.9 – Búsqueda por grupo de un horario registrado.



Derechos Reservados © 2016

Figura 4.4.10 – Despliegue del horario.

Se procede a eliminar la materia 1110 del grupo 9001 (figura 4.4.11). El sistema muestra una pantalla de confirmación (figura 4.4.12).

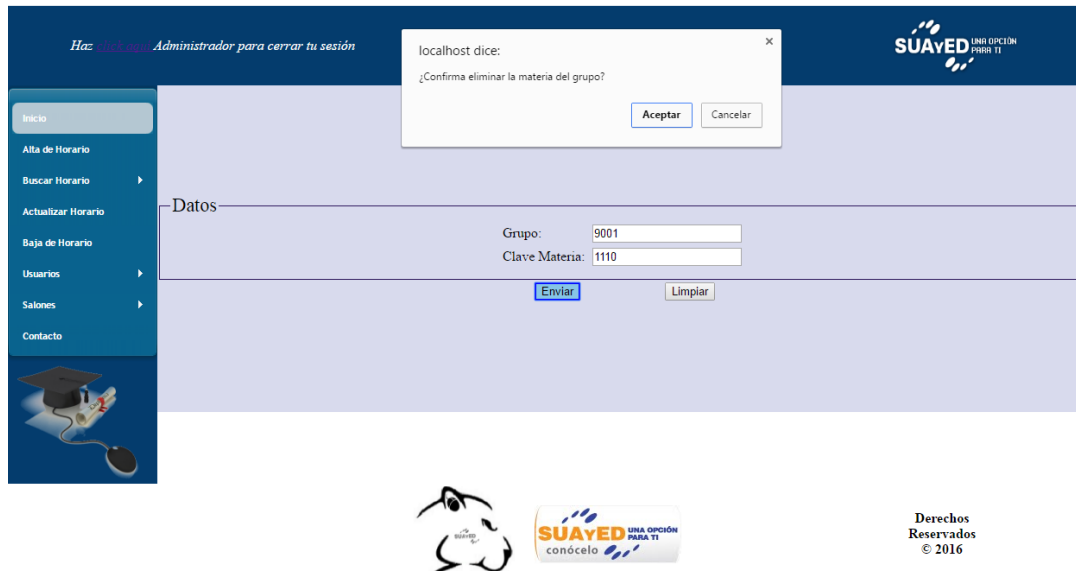


Figura 4.4.11 Baja de la materia en un grupo registrado.



Figura 4.4.12 Mensaje de confirmación del borrado.

Para validar que el borrado se realizó correctamente en la base de datos, se busca nuevamente el horario registrado para el grupo 9001 (figura 4.4.13).



Figura 4.4.13 – Búsqueda del horario del grupo en el que se realizó el borrado.

Se despliega el horario del grupo 9001 y se confirma que la materia 1110 ya no está asignada a este grupo (figura 4.4.14).



Figura 4.4.14 – Despliegue del horario del grupo 9001 donde se confirma que ya no aparece la materia 1110.



Mantenimiento

El mantenimiento es el proceso mediante el cual se mejora y optimiza el software después de su entrega al usuario final, incluye corrección y prevención de defectos.

Existen diversos tipos de mantenimiento de software:

- Mantenimiento Predictivo

Consiste en hacer revisiones periódicas y programadas, para detectar cualquier condición (presente o futura) que pudiera impedir el uso apropiado del software y poder corregirla.

- Mantenimiento Correctivo

Este mantenimiento consiste en corregir los errores, detectados tanto en la fase de pruebas como en la fase de operación del sistema, del software. Estos defectos son reportados generalmente por los usuarios.

- Mantenimiento Perfectivo

Son acciones llevadas a cabo para mejorar la calidad interna del sistema: Reestructuración del código, definición más clara del sistema, y optimización del rendimiento y de la eficiencia.

- Mantenimiento Aumentativo

Consiste en las modificaciones e incorporaciones necesarias en un producto de software para cubrir las necesidades en expansión de los usuarios.

- Mantenimiento Horizontal

Es el mantenimiento al hardware sobre el cual están instaladas las plataformas que soportan la implementación del sistema.

- Mantenimiento Vertical

Es el mantenimiento enfocado al software que conforma la plataforma para la implementación del sistema.



4.5 Evaluación del sitio empleando diferentes dispositivos y navegadores.

Responsive Web Design permite crear sitios web que se adaptan al ancho del dispositivo en que se esté navegando, lo que permite tener un único diseño para PC’s, teléfonos y tabletas. Un aspecto muy interesante de esta tecnología es que no solamente cambia el diseño cuando cambia el ancho del dispositivo, sino que también se ajusta el tamaño de las imágenes.

En la figura 4.5.1 se muestra el sitio web en el navegador Mozilla Firefox, aquí observamos la página principal después de haber validado el nombre de usuario y contraseña. En todas las páginas del sistema se mostrara el encabezado, menú y pie de página, solo cambiará el contenido.

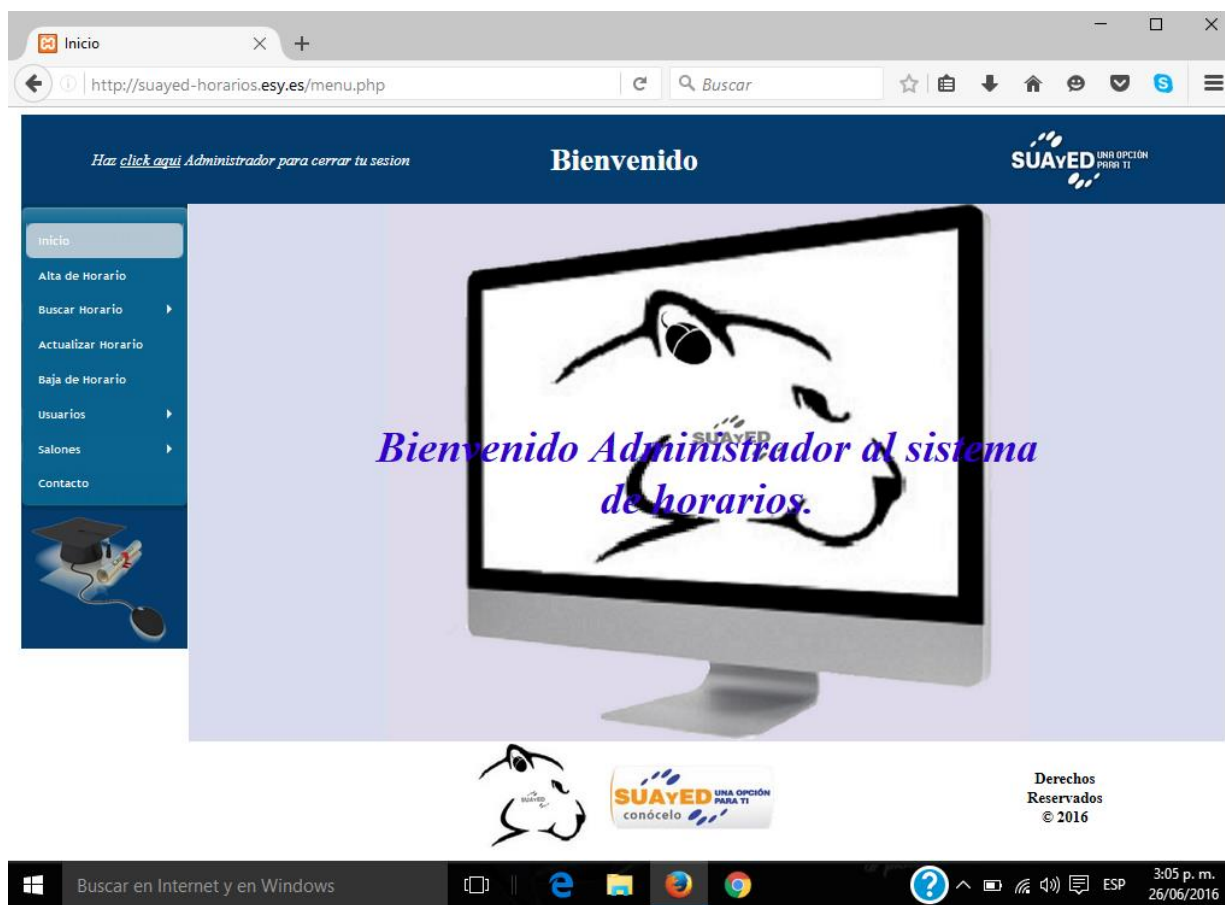


Figura 4.5.1 – Web en navegador Mozilla Firefox.

En la figura 4.5.2 se muestra el sistema web en el navegador Internet Explorer, aquí observamos la página de alta de horario en donde al momento de seleccionar el día nos desplegará un cuadro emergente para seleccionar el día, mes y año en que se desea dar de alta la asesoría.

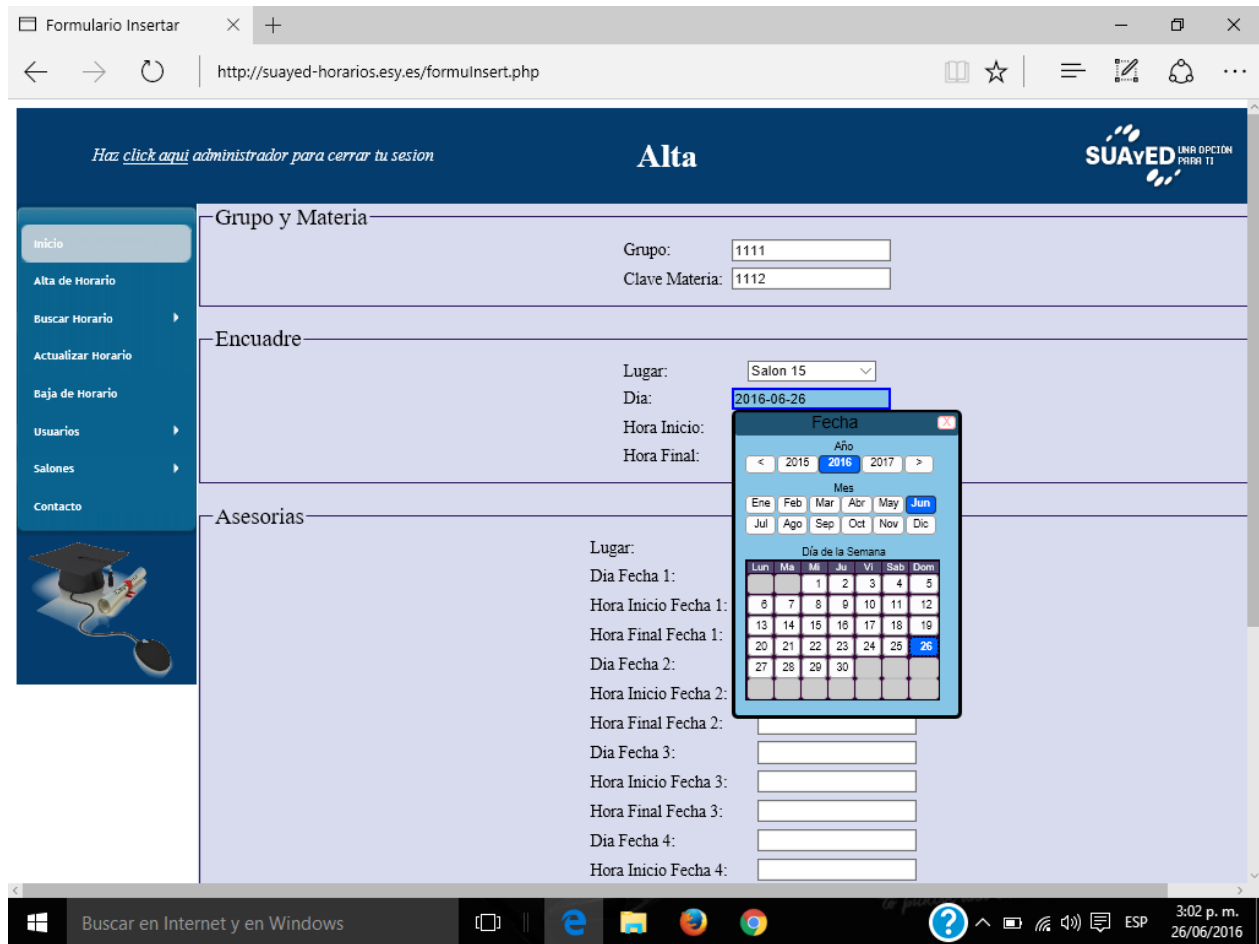


Figura 4.5.2 – Web en navegador Internet Explorer.

En la figura 4.5.3 se muestra el sistema web en el navegador Google Chrome, aquí observamos un submenú al momento de realizar la búsqueda de un horario, esta búsqueda puede ser por grupo y materia, semestre, día o mostrar todos los horarios que se encuentran dados de alta.

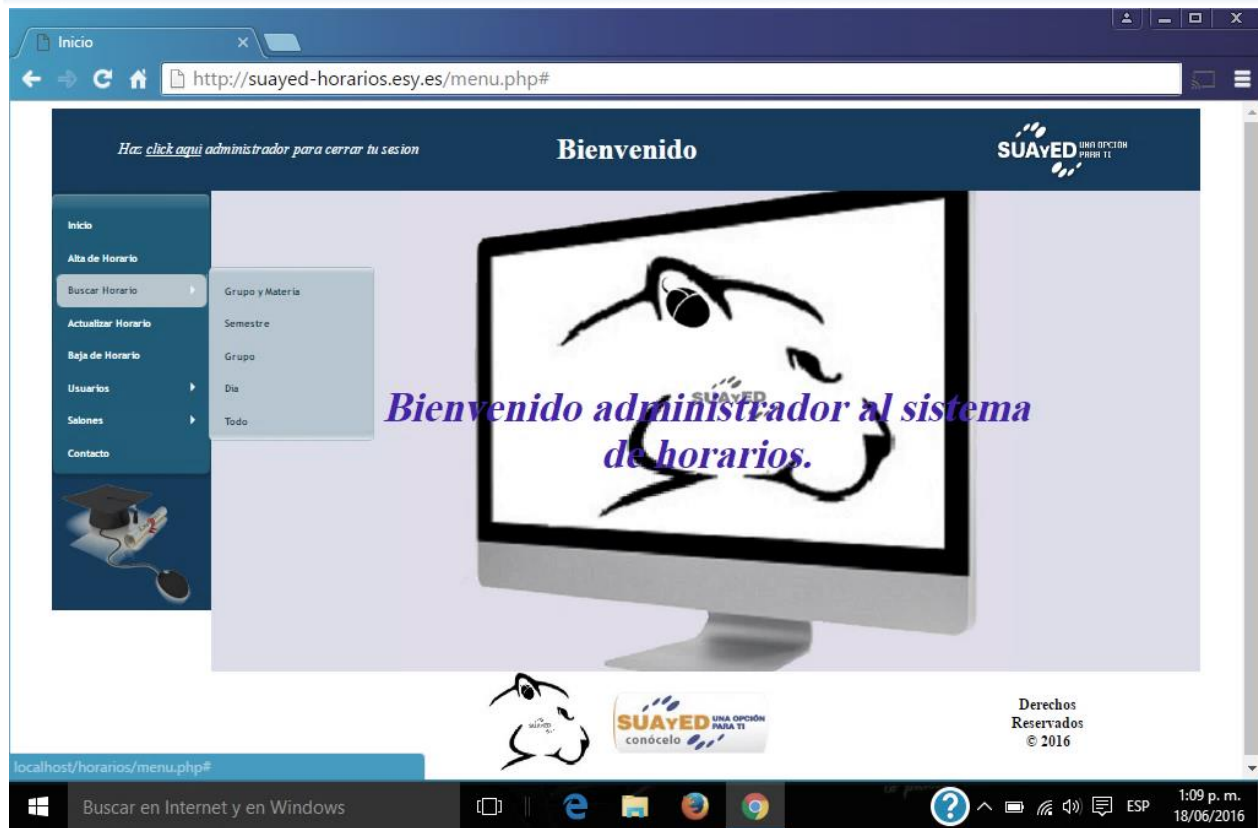


Figura 4.5.3 – Web en navegador Google Chrome.

En la figura 4.5.4 se muestra el sistema web en un dispositivo móvil, aquí observamos una vez más la página de alta de horario en donde al momento de seleccionar la hora de inicio o la hora final de una asesoría nos desplegara un cuadro emergente para seleccionar dichas horas. Si se selecciona una hora final menor que la hora inicial de la asesoría, automáticamente nos mostrará un mensaje de advertencia.

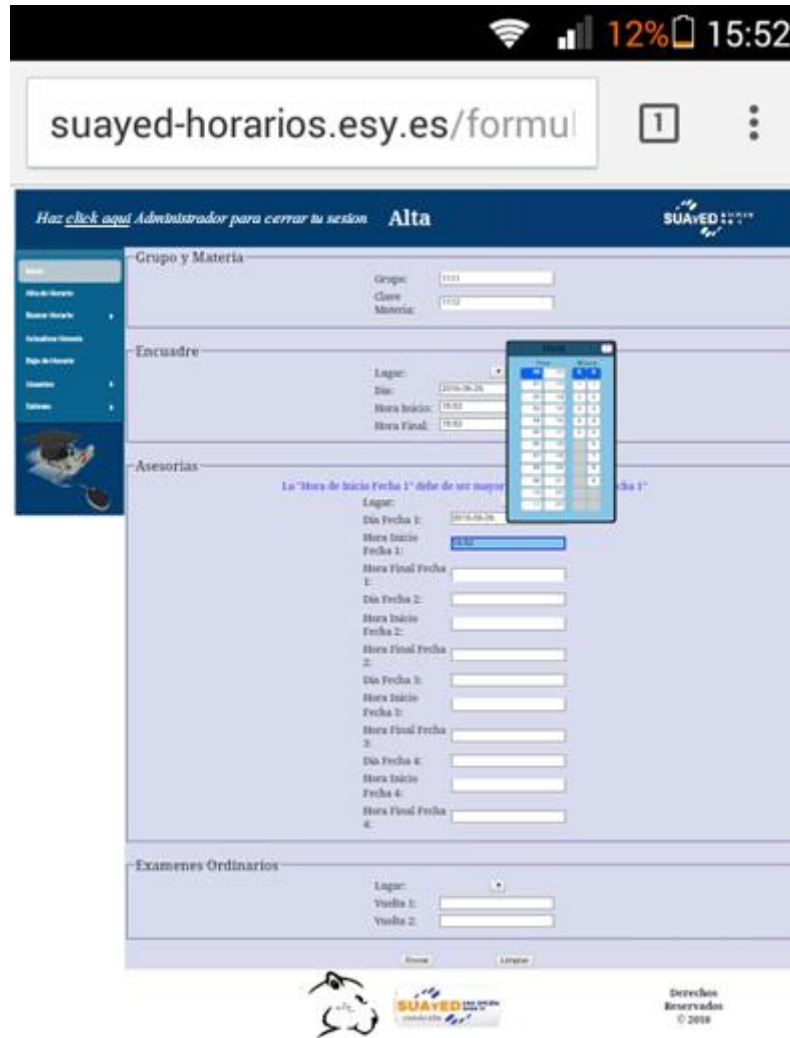


Figura 4.5.4 – Web en un Smartphone.

En la figura 4.5.5 se muestra el sistema web en una Tablet, aquí se observa la página donde se dará de alta un usuario nuevo, donde nos pedirá insertar un nombre de usuario y una contraseña. Este nuevo usuario podrá dar de alta, baja y cambios en los horarios.



Derechos Reservados © 2016

Figura 4.5.5 – Web en una Tablet.





CONCLUSIONES



Conclusiones

- En la actualidad, emplear un sistema automatizado de software permite satisfacer las necesidades del usuario, pues se logran optimizar tiempo y recursos.
- Mejora el flujo de la información, haciendo más eficientes los procesos de la institución.
- Al utilizar software libre se tiene la ventaja de aminorar costos y crear desarrollos de gran calidad, pues el continuo aporte de ideas, evoluciona las características de las herramientas utilizadas.
- Al elaborar esta tesis, podemos concluir que realizamos de manera satisfactoria el análisis, diseño, pruebas e implementación del sistema de automatización en la institución.
- Se logró automatizar los procesos realizados por personal de la institución, logrando disminuir errores en duplicidad de datos, traslape de horarios, asignación errónea de salones, etc.
- Ahorro de tiempo y energía por parte del personal encargado de realizar la programación de horarios manualmente evitando suposiciones y doble esfuerzo.
- La interfaz web del sistema es intuitiva y comprensible para el usuario final.
- La consulta de horarios se realiza de manera más sencilla y práctica, obteniendo la información requerida de manera más eficiente, dando resultados concisos.



- El sistema ofrece disponibilidad y confiabilidad lo que repercute positivamente en el servicio a alumnos y profesores.
- Llevar a cabo este proyecto nos sensibilizó a observar detalladamente las necesidades que se tienen en las instituciones, a las que comúnmente no se les presta la atención debida, ya sea por insuficiencia de fondos para comprar un proyecto o para pagar a un especialista. Se va dejando de lado las necesidades que por mínimas que aparenten ser, cuando se logran llevar a cabo, genera un impacto que nos produce satisfacción, pues ponemos en práctica lo aprendido a lo largo de la carrera.
- También pudimos percatarnos de que no se necesita gran cantidad de recursos financieros para desarrollar un sistema competente, necesario a instituciones que deseen aumentar la eficiencia del control y administración de la información que utilizan, así como mejorar los procesos internos.
- Logramos vislumbrar lo importante que es prestar atención a los requerimientos del usuario final y revisar con él cada avance, para que esté convencido de que el sistema le ayudará a mejorar sus labores diarias.
- Utilizar herramientas informáticas permite al usuario tener un mayor control sobre la información que maneja y llevar de forma ordenada y controlada la asignación de horarios, contribuyendo al mejoramiento de procesos.
- El diseño del sistema se realizó de forma clara y entendible para ayudar al usuario a familiarizarse con él y utilizarlo de forma eficiente.
- Pasar de lo manual a lo automático es una gran ventaja además de optimizar el sistema rudimentario, ayuda en la reducción de tiempos en el proceso de generación horarios.



- El desarrollo del sistema permitirá a la institución dar un gran paso hacia la automatización y optimización de los procesos, así como a la digitalización y modernización de los elementos que intervienen en ello, elevando la confiabilidad de la institución.
- El sistema que hemos desarrollado nos ha dejado conocimientos importantes en el ámbito profesional, ya que como ingenieros nos enfrentamos diariamente a problemas a los que es necesario dar una solución óptima, la cual es posible hallar mediante un buen análisis, metodología y diseño.
- Finalmente externamos nuestro agradecimiento y orgullo que sentimos hacia la Facultad de Ingeniería de la UNAM ya que nos formó profesionalmente para detectar necesidades y proponer soluciones para erradicarlas.



BIBLIOGRAFÍA



- Adrian W. West (2014). Practical PHP and MySQL Website Databases. Apress.
- Apache HTTP Server (2015). Características Servidor Apache. Recuperado el 30 de abril de 2014, de <https://es.opensuse.org/Apache>.
- ----- (2015). Servidor Web. Recuperado el 30 de abril de 2016, de <http://culturacion.com/que-es-apache>.
- ----- (2015). Características, instalación y configuración. Recuperado el 1 de mayo de 2016, de http://ldc.usb.ve/~emilio/Portafolio/Software/REDES3/G5b/Presenta_Site/caracteristicas_pyc.htm.
- Cherokee (2013). Evolved Web Infrastructure Software. Recuperado el 11 de mayo de 2016, de <http://cherokee-project.com>.
- Cobo Ángel (2005). PHP y MySQL: Tecnología para el desarrollo de aplicaciones web. España: Díaz de Santos.
- Con la tecnología de Blogger (2015). Tipos de requerimiento. Recuperado el 3 de mayo de 2016, de <http://fundamentos-ing-requisitos.blogspot.mx/2012/02/normal-0-21-false-false-false-es-ve-x.html>.
- ----- (2015). Sistema de Información Computarizado. Recuperado el 6 de mayo de 2016, de <http://sistemadeinformacioncomputarizado.blogspot.mx>.
- Date C.J. (2001). Introducción a los Sistemas de Bases de Datos, 7ª Ed. México: Pearson Educación.
- Eslava, J., Muñoz, V. (2013). El nuevo PHP paso a paso. España: Bubok.
- Hernan Beati (2001). PHP: creación de páginas web dinámicas. Argentina: Alfaomega.
- HTML.net (2016). Introducción. Recuperado el 11 de mayo de 2016, de <http://html.net/tutorials/html5/introduction.php>.
- Isabel Ramos Román, José Javier Dolado Cosín (2007). Técnicas cuantitativas para la gestión en la ingeniería del software. España: Netbiblo.
- Kroenke David (2003). Procesamiento de bases de datos. México: Prentice Hall.
- Leal Castellanos, M (2015). Taller Cliente-Servidor. Recuperado el 26 de abril de 2016, de



- <https://basesii.wikispaces.com/file/view/Caracter%C3%ADsticas+de+la+arquitectura+Cliente.pdf>.
- Lucas Gómez Ángel (1993). Diseño y gestión de Sistemas de Bases de Datos. España: Paraninfo.
 - Luna González, L (2004). El diseño de interfaz gráfica de usuario para publicaciones digitales, UNAM. Recuperado el 16 de mayo de 2016, de http://www.revista.unam.mx/vol.5/num7/art44/ago_art44.pdf.
 - Marquez Avensaño, B. (2004). Arquitectura Cliente-Servidor. Recuperado el 26 de abril de 2016, de http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/marquez_a_bm.
 - Michael Peacock (2010). PHP 5 Social Networking. Packt Publishing.
 - Minera Francisco (2010). PHP 6: Sitios dinámicos con el lenguaje más robusto. Argentina: Gradi.
 - NGINX (2015). Flawless Application Delivery for the Modern Web. Recuperado el 11 de mayo de 2016, de <https://www.nginx.com>.
 - Olga Pons, Nicolás Marín, Juan Miguel Medina, Silvia Acid (2006). Fundamentos de Bases de Datos, 5ª Ed. España: Thomson.
 - Pérez Porto, J. Gardey, A. (2008). Definición.de. Recuperado el 23 de abril de 2016, de <http://definicion.de/exactitud>.
 - PostgreSQL-es (2013). Sobre PostgreSQL. Recuperado el 8 de mayo de 2016, de http://www.postgresql.org/es/sobre_postgresql.
 - Ramez A. Elmasri & Shamkant B. Navathe (2001). Introducción a los sistemas de bases de datos, 7ª Ed. Boston: Prentice Hall.
 - Renaud Paul, E. (1996). Introduction to Client/Server Systems: A Practical Guide for Systems, 2ª Ed. New Jersey: John Wiley & Sons.
 - Roger, S. Pressman (1998). Ingeniería del software, 3ª Ed. España: McGraw Hill.
 - Ruby, el mejor amigo de un desarrollador (2016). Ruby es. Recuperado el 11 de mayo de 2016, de <https://www.ruby-lang.org/es>.
 - Servidor de pruebas (2012). Introducción a phpMyAdmin. Recuperado el 15 de mayo de 2015, de http://www.aulaclie.es/dreamweaver-cs5/t_20_9.htm.



- Silberchatz, A., Henry, F.(1998). 5ª ed. Fundamentos de las bases de datos. España: McGrall Hill.
- SlideShare (2016). Beneficios de un sistema de Información. Recuperado el 6 de mayo de 2016, de <http://es.slideshare.net/sylem123/beneficios-de-los-sistemas-de-informacion>.
- ----- (2016). Integridad de la Información. Recuperado el 24 de abril de 2014, de <http://es.slideshare.net/CharlySantana1/integridad-de-la-informacion>.
- Sommerville, Ian. (2005). Ingeniería del Software. México: Pearson Educación
- Sourceforge (2016). Download. Recuperado el 13 de mayo de 2016, de <https://sourceforge.net/projects/xampp/files/XAMPP%20Windows/5.6.8>.
- Stoyan Stefanov (2013). JavaScript for PHP Developers. O'Reilly Media.
- The apache software foundation (2016). About Apache. Recuperado el 11 de mayo de 20126, de <http://www.apache.org>.
- ----- (2016). The Number One HTTP Server On The Internet. Recuperado el 5 de mayo de 2015, de <https://www.apache.org>.
- The PHP Group (2016). Documentación. Recuperado el 9 de mayo de 2016, de <http://php.net>.
- ----- (2016). Manejo de XfForms. Recuperado el 28 de abril de 2016, de <http://php.net/manual/es/features.xforms.php>.
- The world's most popular open source database (2016). Manual de referencia. Recuperado el 8 de mayo de 2016, de <http://dev.mysql.com/doc/refman/5.7/en>.
- Troy Dimes (2015). JavaScript Una Guía de Aprendizaje para el Lenguaje de Programación JavaScript. España: Babelcubes books.
- True universal open source database (2016). About Firebird. Recuperado el 8 de mayo de 2016, de <http://www.firebirdsql.org/en/about-firebird>.
- UAM Cuajimalpa (2015). Análisis de requerimiento. Recuperado el 8 de mayo de 2016, de http://web.cua.uam.mx/publicaciones/Notas_Analisis_Requerimiento.pdf.



- Universidad Nacional Autónoma de México (2012). Coordinación de Universidad Abierta y Educación a Distancia. Recuperado el 20 de abril de 2016, de <http://aulasvirtuales.cuaed.unam.mx>.
- ----- (2012). Fundamentos de seguridad informática. Recuperado el 23 de abril de 2014, de <http://redyseguridad.fi-p.unam.mx/proyectos/seguridad/ServDisponibilidad.php>.
- ----- (2014). Reglamento del estatuto del Sistema de Universidad Abierta y Educación a Distancia. Recuperado el 20 de abril de 2016, de <http://www.ddu.unam.mx>.
- ----- (2015). Red de responsables en visibilidad Web UNAM. Recuperado el 21 de abril de 2016, de <http://www.visibilidadweb.unam.mx/catic.html>.