

INSTITUTO SUPERIOR TECNOLÓGICO

VIDA NUEVA



CARRERA:

TECNOLOGÍA EN INFORMÁTICA

TEMA DEL PROYECTO DE APLICACIÓN PRÁCTICA

**DESAROLLAR LA CAPA DE ACCESO A DATOS DE UN SISTEMA DE
GESTIÓN DE TURNOS DIGITALES DE ATENCIÓN ESTUDIANTIL CON
TECNOLOGÍA OPEN SOURCE**

AUTOR:

ALMEIDA FONSECA FAUSTO RENÉ

TUTOR:

ING. ARIAS MARTINEZ DARWIN RENÉ

FECHA:

MARZO 2020

QUITO – ECUADOR

CESIÓN DE DERECHOS DE AUTOR

Yo, ALMEIDA FONSECA FAUSTO RENÉ portador/a de la cédula de ciudadanía 171860939-7, facultado/a de la carrera TECNOLOGÍA EN INFORMÁTICA, autor/a de esta obra certifico y proveo al Instituto Superior Tecnológico Vida Nueva, usar plenamente el contenido del informe con el tema “DESAROLLAR LA CAPA DE ACCESO A DATOS DE UN SISTEMA DE GESTIÓN DE TURNOS DIGITALES DE ATENCIÓN ESTUDIANTIL CON TECNOLOGÍA OPEN SOURCE”, con el objeto de aportar y promover la lectura e investigación, autorizando la publicación de mi proyecto de titulación en la colección digital del repositorio institucional bajo la licencia de Creative Commons: Atribución-NoComercial-SinDerivadas.

En la ciudad de Quito, del mes de Marzo de 2020.

ALMEIDA FONSECA FAUSTO RENÉ

C.I.: 171860939-7

CERTIFICACIÓN DEL TUTOR

En mi calidad de Tutor del Proyecto: “DESAROLLAR LA CAPA DE ACCESO A DATOS DE UN SISTEMA DE GESTIÓN DE TURNOS DIGITALES DE ATENCIÓN ESTUDIANTIL CON TECNOLOGÍA OPEN SOURCE” en la ciudad de Quito, presentado por la ciudadana ALMEIDA FONSECA FAUSTO RENÉ, para optar por el título de Tecnólogo en INFORMÁTICA, certifico, que dicho proyecto ha sido revisado en todas sus partes y considero que reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del tribunal examinador que se designe

En la ciudad de Quito, del mes de Marzo de 2020.

TUTOR: ING. ARIAS MARTÍNEZ DARWIN RENÉ

APROBACIÓN DEL JURADO EXAMINADOR O TRIBUNAL

Los miembros del jurado examinador aprueban el informe de investigación, sobre el tema: “DESAROLLAR LA CAPA DE ACCESO A DATOS DE UN SISTEMA DE GESTIÓN DE TURNOS DIGITALES DE ATENCIÓN ESTUDIANTIL CON TECNOLOGÍA OPEN SOURCE” en la ciudad de Quito”
Del estudiante: ALMEIDA FONSECA FAUSTO RENE de la Carrera en Tecnología en INFORMÁTICA.

Para constancia firman:

C.I.:

C.I.:

C.I.:

C.I.:

DECLARACIÓN DE AUTENTICIDAD

Yo, ALMEIDA FONSECA FAUSTO RENÉ con cedula de ciudadanía 171860939-7 estudiante del Instituto Superior Tecnológico “Vida Nueva”, declaro que he realizado este trabajo de titulación tomando en consideración citas bibliográficas que se nombran en este texto.

El Instituto Superior Tecnológico “Vida Nueva” puede utilizar este trabajo de titulación como una ayuda bibliográfica.

En la ciudad de Quito, del mes de Marzo de 2020.

ALMEIDA FONSECA FAUSTO RENE

C.I.: 171860939-7

ÍNDICE

CESIÓN DE DERECHOS DE AUTOR.....	ii
CERTIFICACIÓN DEL TUTOR.....	iii
APROBACIÓN DEL JURADO EXAMINADOR O TRIBUNAL	iv
DECLARACIÓN DE AUTENTICIDAD.....	v
1. INTRODUCCIÓN	1
2. ANTECEDENTES	2
3. OBJETIVOS.....	4
3.1 Objetivo General.....	4
3.2 Objetivos Específicos	4
4. DESARROLLO.....	5
4.1 MARCO TEÓRICO	5
4.1.1 Internet.....	5
4.1.2 WWW (World Wide Web).....	5
4.1.3 Lenguajes de programación	5
4.1.4 Lenguajes de programación de alto nivel	6
4.1.5 Lenguajes de programación de bajo nivel	7
4.1.6 Lenguaje de Marcas de Hipertexto HTML (HyperText Markup Language).....	7
4.1.7 Lenguaje de programación interpretado JavaScript.....	7
4.1.8 AJAX.....	8
4.1.9 Lenguaje de Programación de alto nivel PHP	8
4.1.10 Bases de datos	9
4.1.11 Sistema Gestor de Base de Datos MySQL.....	9
4.1.12 Servidor local XAMPP	9
4.2 PROCEDIMIENTO / METODOLOGÍA	10
4.2.1 Diseño.....	15

4.2.2	Construcción	21
4.2.2.1	Construcción de base de datos.	21
4.2.2.2	Procedimientos almacenados a nivel de base de datos	23
4.2.2.3	Interfaz de recepción de contenido	23
4.2.2.4	Cabeceras de contenido de tablas desarrolladas en la base de datos. 24	
4.2.2.5	Cabeceras y contenido de tabla historial desarrollada en la base de datos.	25
4.2.2.6	Cabeceras y contenido de tabla módulo desarrollada en la base de datos.	27
4.2.2.7	Consultas a base de datos desde PHP.	28
4.2.2.8	Desarrollo del modelo de transferencia padre e hijo con JavaScript	29
4.2.2.9	Desarrollo HTML	30
4.2.2.10	Desarrollo de módulos a nivel de programación.	31
4.2.2.11	Consumo de datos en tiempo real con Ajax.	32
4.2.3	Implementación	33
4.2.3.1	Instalación del Sistema	34
4.2.3.2	Carga de base de datos	34
4.2.3.3	Software funcionando	35
4.2.3.4	Inserción de información	35
4.2.3.5	Mensajería Móvil	36
4.2.3.6	Consulta de Información	36
4.2.3.7	Asignación de turnos	36
4.2.3.8	Comprobación de turnos.	37
4.2.3.9	Vista de Módulos	37
4.2.3.10	Registro de historial.	37

4.2.3.11 Resultados implementación	38
5. CONCLUSIONES	39
6. RECOMENDACIONES	40
7. Bibliografía	41
8. ANEXOS	43
8.1 Manual de usuario	43

ÍNDICE TABLAS

<i>Tabla 1:</i> Contenido de la tabla ttrámites.....	25
<i>Tabla 2:</i> Contenido de la tabla historial.....	26
<i>Tabla 3:</i> Contenido de la tabla módulo.....	27
<i>Tabla 4:</i> Entendimiento de programación	28
<i>Tabla 5:</i> Entendimiento de consultas	29
<i>Tabla 6:</i> Modelo de transferencia padre e hijo con JS	30
<i>Tabla 7:</i> Entendimiento HTML	31
<i>Tabla 8:</i> Entendimiento de módulos.....	31
<i>Tabla 9:</i> Entendimiento de programación Ajax.....	33
<i>Tabla 10:</i> Requerimientos del sistema de gestión de turnos MITU	33

ÍNDICE DE FIGURAS

Figura 1: Recuperado de “5 beneficios del Software Adaptable”, de Capetillo, J. , (04 Enero 2017). Recuperado de https://www.ceesa.com/noticias/5-beneficios-del-software-adaptable/	10
<i>Figura 2:</i> Metodología ASD.....	14
<i>Figura 3:</i> Diseño de base de datos	15
<i>Figura 4:</i> Procedimientos almacenados en base de datos	15
<i>Figura 5:</i> Tablas creadas en la base de datos	16
<i>Figura 6:</i> Directorio raíz del software	16
Figura 7: Contenido de directorio Ajax	17
<i>Figura 8:</i> Contenido de directorio Backups	17
<i>Figura 9:</i> Contenido de directorio Classes	17

<i>Figura 10:</i> Contenido de directorio Conexión	18
<i>Figura 11:</i> Contenido de directorio CSS.....	18
<i>Figura 12:</i> Contenido de directorio FPDF.....	18
<i>Figura 13:</i> Contenido de directorio Gestión.....	19
<i>Figura 14:</i> Contenido de directorio Historial	19
<i>Figura 15:</i> Contenido de directorio HTML	19
<i>Figura 16:</i> Contenido de directorio IMG	20
<i>Figura 17:</i> Contenido de directorio JS.....	20
<i>Figura 18:</i> Contenido de directorio Vistamodulos.....	20
<i>Figura 19:</i> Contenido de directorio Raíz.....	21
<i>Figura 20:</i> Creación de base de datos	21
<i>Figura 21:</i> Creación de rutinas en PhpMyAdmin	22
<i>Figura 22:</i> Script desarrollado en rutinas.	22
<i>Figura 23:</i> Definición de tablas necesarias para el software	22
<i>Figura 24:</i> Script textual de procedimientos almacenados	23
<i>Figura 25:</i> Modal de recepción de datos	24
<i>Figura 26:</i> Cabeceras de contenido de tablas.....	24
<i>Figura 27:</i> Contenido de tablas, con tipo de valores.	25
<i>Figura 28:</i> Contenido de tabla historia en PhpMyAdmin	26
<i>Figura 29:</i> Estructura de tabla modulo	27
<i>Figura 30:</i> Script de conexión a base de datos	27
<i>Figura 31:</i> Script de consulta base de datos desde PHP	28
<i>Figura 32:</i> Modelo de transferencia Padre e Hijo en JavaScript.....	29
<i>Figura 33:</i> Asignación de atributos HTML	30
<i>Figura 34:</i> Script de módulos del software en HTML.....	31
<i>Figura 35:</i> Script de consumo de datos real Ajax	32
<i>Figura 36:</i> Descompresión de archivo.zip	34
<i>Figura 37:</i> Importación de base de datos en PhpMyAdmin	34
<i>Figura 38:</i> Página de inicio del software en producción	35
<i>Figura 39:</i> Modal de registro de información	35
<i>Figura 40:</i> Mensaje móvil recibido.	36
<i>Figura 41:</i> Información recibida, desde los modales, en tabla de atención ..	36
<i>Figura 42:</i> Asignación de turnos	36

<i>Figura 43: Control de turnos en tiempo real</i>	37
<i>Figura 44: Información de usuario, vista en módulos</i>	37
<i>Figura 45: Histórico de información obtenida diariamente</i>	37
<i>Figura 46: Reporte de turnos en formato PDF</i>	38

INDICE ANEXOS

<i>Anexo 1: Descompresión de software</i>	44
<i>Anexo 2: Creación de base de datos desde PhpMyAdmin</i>	44
<i>Anexo 3: Importación de base de datos en PhpMyAdmin</i>	44
<i>Anexo 4: Imagen de inicio del software en el servidor local</i>	45
<i>Anexo 5: Ingreso de datos al software</i>	45
<i>Anexo 6: Confirmación de recepción de datos</i>	46
<i>Anexo 7: Envió de información</i>	46
<i>Anexo 8: Acceso a información del instituto</i>	46
<i>Anexo 9: Modal de información</i>	47
<i>Anexo 10: Modal de información de opciones de pago</i>	47
<i>Anexo 11: Solicitud de información</i>	48
<i>Anexo 12: Mensaje recibido con enlace de página de información</i>	48
<i>Anexo 13: Página de inicio de información</i>	49
<i>Anexo 14: Página de información general</i>	49
<i>Anexo 15: Modales de información de pago</i>	50
<i>Anexo 16: Página de gestión de turnos</i>	50
<i>Anexo 17: Acciones para la gestión de turnos</i>	51
<i>Anexo 18: Confirmación de turnos atendidos</i>	51
<i>Anexo 19: Atención de turnos, desde interfaz gráfica</i>	51
<i>Anexo 20: Módulos de turnos</i>	52
<i>Anexo 21: Acciones para controlar turnero</i>	52
<i>Anexo 22: Módulos con información asignada</i>	52
<i>Anexo 23: Acción de cerrar el turnero</i>	53
<i>Anexo 24: Botón de historial de información</i>	53
<i>Anexo 25: Consulta de información desde el historial</i>	53
<i>Anexo 26: Botón de búsqueda de información</i>	54
<i>Anexo 27: Obtención de resultados encontrados</i>	54

<i>Anexo 28: Botón de exportación de resultados en Excel</i>	<i>55</i>
<i>Anexo 29: Alojamiento de archivo Excel exportado</i>	<i>55</i>
<i>Anexo 30: Resultado obtenidos del software en formato Excel.....</i>	<i>55</i>
<i>Anexo 31: Impresión o exportación de resultados en formato PDF.....</i>	<i>56</i>
<i>Anexo 32: Archivo de resultados en formato PDF</i>	<i>56</i>

1. INTRODUCCIÓN

El concepto de Open Source, también llamado “Código Abierto” es un término que se utiliza para denominar a cierto tipo de software que se distribuye mediante una licencia que le permite al usuario final, si tiene los conocimientos necesarios, utilizar el código fuente del programa para estudiarlo, modificarlo y realizar mejoras en el mismo, pudiendo incluso hasta redistribuirlo.

La facilidad del manejo de tecnología Open Source se refleja en su lenguaje de programación insignia PHP, que permite embeber pequeños fragmentos de código dentro de una página web común, combinándolos con un sistema gestor de base de datos que permite gestionar archivos llamados de bases de datos. Adicionalmente se cuenta con lenguajes de programación asíncronos como JavaScript, Ajax, NodeJs, entre otros. Los mismos que permiten una mayor interacción entre el software y el usuario final.

Con los antecedentes anteriormente citados se propone el desarrollo de la capa de acceso a datos para un sistema de gestión de turnos digitales de atención estudiantil con tecnología Open Source, para el Instituto Superior Tecnológico Vida Nueva (ITSVN). Con la finalidad de transformar digitalmente un proceso que se realiza de manera manual, a un proceso automatizado, que mejore el flujo de atención a los usuarios, mediante herramientas que ofrece el lenguaje de programación Open Source PHP de parte del servidor, la interactividad que puede brindar el software con lenguaje asíncrono como JavaScript y AJAX, que en su totalidad es gratuito. La implementación de estas herramientas tecnológicas demostrará la flexibilidad para brindar eficiencia en los servicios que brinda el Instituto Superior Tecnológico Vida Nueva, generando así un ambiente confortable para los usuarios al momento de realizar un trámite o solicitud dentro de la institución, donde el sistema desarrollado en código abierto será capaz de realizar respaldos de la información ingresada diariamente, el mantenimiento de la base de datos automáticamente y generación de reportes en Excel y Pdf.

2. ANTECEDENTES

El desarrollo de las tecnologías, en la última década, ha dado un impulso notable a nuevos medios de comunicación, los cuales hace pocos años no pasaban de ser realizados de forma manual, con un radio de acción limitado. En este sentido, el gran desarrollo de nuevos lenguajes de programación para la interacción del computador con el usuario final ha permitido transformar digitalmente procesos manuales, como lo indican en sus estudios realizados los estudiantes Raúl Andrés Landázuri y Richard Vinicio Siza, en la Pontificia Universidad Católica del Ecuador, a través de su tesis: “Diseño, desarrollo e implementación del sistema de gestión de turnos programados para el Sub-centro de Salud Carapungo”, se realiza el estudio de cómo implementar las nuevas tecnologías en el Sub-centro de Salud Carapungo con el propósito de mejorar la atención al usuario. El estudio contempla una investigación del área de acceso de los usuarios hacia el Sub-centro que ya por varios años la manera para acceder a un turno, los usuarios tendrían que madrugar a la unidad médica para alcanzar un turno en medicina general, obstetricia y odontología especialmente, tomando en cuenta la situación de riesgo de ser asaltado. En ese sentido los Estudiantes Raúl Andrés Landázuri y Richard Vinicio Siza, tuvieron como objetivo ofrecer una solución tecnológica desarrollada con Tecnología OPEN SOURCE, que buscaría mejorar la disponibilidad de los turnos para pacientes con patologías prevalentes disminuyendo el tiempo de espera y facilitar la obtención de turnos para los estudios complementarios. Detallando el estudio anterior, los procesos de observación tienen una relación directa con el tema de tesis que se está desarrollando, buscando también la facilidad de turnos y disminución de tiempo de espera en los usuarios.

Un segundo trabajo correspondiente a los Autores Manzanilla Álvarez Cristóbal Fernando y Olivo Vega Julio César en el año 2015, quienes realizaron el: “DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA PARA LA GENERACIÓN DE TURNOS DE LOS SERVICIOS DEL CUERPO DE BOMBEROS DEL GAD MUNICIPAL DEL CANTÓN DE SANTO DOMINGO DE LOS COLORADOS”, para la obtención del título de Ingeniero

de Sistemas y Computación en la PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR SEDE SANTO DOMINGO, detallan en este trabajo la expansión urbana en las últimas décadas en el Cantón de Santo Domingo de los Colorados, la aglomeración de personas en el Cuerpo de Bomberos, y el sistema de atención a los ciudadanos es considerado poco sofisticado en relación al empleo del tiempo y calidad en el servicio ofrecido, por esta razón se desarrolla e implementa el sistema informático denominado Turnos+, propuesta orientada a resolver a las necesidades que tiene el Cuerpo de Bomberos, administrando la atención de servicios mediante la entrega organizada de turnos. El talento humano que labora en la entidad requiere de recursos y herramientas tecnológicas que ayuden a un mejor uso del tiempo y acceso a un ambiente de trabajo que desarrollo su potencial garantizando la ejecución de un trabajo de eficiencia y eficacia, para dar una respuesta a estas necesidades actuales se plantea el desarrollo e implementación de un sistema de información funcional. Para esto se emplea TECNOLOGÍA OPEN SOURCE, que es muy versátil y ayudara a cumplir con los requerimientos de la institución. Este trabajo se relaciona con la investigación en curso, ya que propone el estudio de aglomeración de personas y como se deberá gestionar los turnos de manera correcta de acuerdo a los servicios que brinda la institución y necesidades de los encargados de administrar el sistema.

3. OBJETIVOS

3.1 Objetivo General

- Desarrollar la capa de acceso a datos de un sistema de gestión de turnos digitales de atención estudiantil con Tecnología Open Source.

3.2 Objetivos Específicos

- Configurar el modelo padre e hijo de JavaScript con Ajax.
- Configurar los módulos de turnos en diferentes tablas con PHP y HTML5.
- Desarrollar el Back End utilizando PHP, JavaScript, Ajax con MySQL.
- Crear procedimientos almacenados en MySQL y enlazarlos con PHP.
- Implementar el consumo de datos en tiempo real con Ajax.

4. DESARROLLO

4.1 0MARCO TEÓRICO

4.1.1 Internet

El Internet, algunas veces llamado simplemente "La Red", es un sistema mundial de redes de computadoras, un conjunto integrado por las diferentes redes de cada país del mundo, por medio del cual un usuario en cualquier computadora puede, en caso de contar con los permisos apropiados, acceder a información de otra computadora y poder tener comunicación directa con otros usuarios en otras computadoras. Es el mayor medio de información accesibles a cualquier persona que la visite desde cualquier parte del mundo (Marker, 2019).

4.1.2 WWW (World Wide Web)

WEB (World Wide Web, o www), es un conjunto de documentos (webs) interconectados por enlaces de hipertexto, disponibles en Internet que se pueden comunicar a través de la tecnología digital. Se entiende por "hipertexto" la mezcla de textos, gráficos y archivos de todo tipo, en un mismo documento. Web no son sinónimo de Internet; Internet es la red de redes donde reside toda la información, siendo un entorno de aprendizaje abierto, más allá de las instituciones educativas formales. La web es un subconjunto de Internet que contiene información a la que se puede acceder usando un navegador. Tanto el correo electrónico, como facebook, twiter, wikis, blogs, juegos, etc. son parte de Internet, pero no la web (Latorre, 2018).

4.1.3 Lenguajes de programación

Un lenguaje de programación es un lenguaje formal diseñado para realizar procesos que pueden ser llevados a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana. Está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el

cual se escribe, se prueba, se depura, se compila (de ser necesario) y se mantiene el código fuente de un programa informático se le llama programación (Gervancio, 2018).

Un lenguaje de programación proporciona los elementos de lenguaje necesarios que son necesarios para traducir los pasos de un pseudocódigo en formato comprensible de la máquina. En otras palabras, el lenguaje de programación proporciona el puente para hacer la transición de pseudocódigo legible por humano a instrucciones legibles para máquina. Las dos clasificaciones principales de lenguajes de programación son: bajo nivel de lenguaje de programación y lenguaje de programación de alto nivel (Gervancio, 2018).

4.1.4 Lenguajes de programación de alto nivel

Los lenguajes de programación de alto nivel son lenguajes de programación en los cuales las instrucciones que se envían para que el ordenador ejecute algunas órdenes son parecidas al lenguaje humano. Debido a que el ordenador no puede ser capaz de reconocer estas órdenes, se necesita el uso de un guía que traduzca dicho lenguaje de alto nivel a un lenguaje de bajo nivel el cual el sistema pueda entender (Alfonsina, 2018).

El lenguaje de programación de alto nivel usa palabras similares al inglés, así como símbolos, signos de puntuación y aritméticos de manera que permite el desarrollo de programas. Estos lenguajes son los más utilizados como lenguajes de programación, a pesar de que no son esencialmente declarativos, estos lenguajes nos permiten que los algoritmos se puedan expresar en un solo nivel y estilo de escritura que es fácilmente legible y comprensible por otros programadores. Además de esto, los lenguajes de alto nivel frecuentan tener la característica de “transportabilidad”; esto quiere decir que están implementados sobre varias máquinas, de manera que un programa puede “transportado” (transferido) fácilmente de una máquina a otra sin ninguna revisión sustancial. En esta orientación, se llaman “independientes de la máquina” (Alfonsina, 2018).

4.1.5 Lenguajes de programación de bajo nivel

Un lenguaje de programación de características bajo nivel es aquel en el que sus instrucciones ejercen un control directo sobre el hardware y están condicionados por la estructura física de la computadora que lo soporta. El uso de la palabra bajo en su denominación no implica que el lenguaje sea inferior a un lenguaje de alto nivel, sino que se refiere a la reducida abstracción entre el lenguaje y el hardware. Estos tipos de lenguajes de programación se utilizan para programar tareas críticas de los Sistemas Operativos, de aplicaciones en tiempo real o controladores de dispositivos (Correa, 2015).

Este lenguaje es muy simple o nada complicado, pero estructurar programas a ese nivel es muy difícil. Dado que este lenguaje viene dado por las especificaciones técnicas del hardware, no permite una abstracción fuera de lo estipulado para el microprocesador de un ordenador. Consecuentemente, es fácilmente trasladado a lenguaje de máquina (Correa, 2015).

4.1.6 Lenguaje de Marcas de Hipertexto HTML (HyperText Markup Language).

HTML no es un lenguaje de programación, básicamente este lenguaje se escribe en su totalidad con elementos, estos elementos están constituidos por etiquetas, contenido y atributos. Esto lo hace por medio de las marcas de hipertexto las cuales son etiquetas conocidas en inglés como tags. Cabe destacar que el HTML permite ciertos códigos que se conocen como scripts, los cuales brindan instrucciones específicas a los navegadores que se encargan de procesar el lenguaje. Entre los scripts que pueden agregarse, los más conocidos y utilizados son JavaScript y PHP. El marcado estructural de HTML es el que estipula la finalidad del texto, aunque no define cómo se verá el elemento (Reyes, 2018).

4.1.7 Lenguaje de programación interpretado JavaScript

JavaScript es un lenguaje de programación, al igual que PHP, si bien tiene diferencias importantes con éste. JavaScript se utiliza principalmente del lado del cliente (es decir, se ejecuta en nuestro ordenador, no en el servidor) permitiendo crear efectos atractivos y dinámicos en las páginas web. Los

navegadores modernos interpretan el código JavaScript integrado en las páginas web (B G. , 2019).

La ventaja de JavaScript es que al estar alojado en el ordenador del usuario los efectos son muy rápidos y dinámicos. Al ser un lenguaje de programación permite toda la potencia de la programación como uso de variables, condicionales, bucles, entre otros (B G. , 2019).

4.1.8 AJAX

Ajax no es un lenguaje de programación nuevo, sino una técnica para desarrollar software mejor y más rápido, y una aplicación más interactiva de las aplicaciones web. Con Ajax, su Javascript puede comunicarse directamente con el usuario, usando el objeto XMLHttpRequest del JavaScript. Con este objeto, Javascript puede negociar datos con un servidor web sin necesidad de recargar la página (Arias, 2015).

Ajax usa la transferencia de datos asíncrona (peticiones del HTTP) entre el navegador y el servidor web, permitiendo que las páginas webs envíen pedazos de pequeñas informaciones del usuario en vez de enviar las páginas enteras. La técnica de Ajax hace que las aplicaciones de Internet sean más pequeñas, más rápidas y más userfriendly (amigable para el usuario) (Arias, 2015).

4.1.9 Lenguaje de Programación de alto nivel PHP

PHP es un lenguaje de scripting de código abierto, del lado del servidor, con programación HTML integrada que se utiliza para crear páginas web dinámicas. Las ventajas de PHP son su flexibilidad y su alta compatibilidad con otras bases de datos. Además, PHP es considerado como un lenguaje fácil de aprender. Las funciones de PHP se relacionan con los scripts del lado del servidor. PHP puede realizar cualquier tarea que cualquier programa CGI (Common Gateway Interface) puede hacer y maneja el intercambio de datos entre el servidor y el software. Por lo tanto, PHP puede recopilar datos o crear webs de contenido dinámico (COLABORADORES, 2016).

4.1.10 Bases de datos

Una Base de Datos es un conjunto de datos estructurado y almacenado de forma sistemática con objeto de facilitar su posterior utilización. Una base de datos puede, por tanto, constituirse con cualquier tipo de datos, incluyendo los de tipo puramente espacial (geometrías, entre otros.) tales como los que se utilizan en un sistema de información, así como, por supuesto, datos numéricos y alfanuméricos como los que constituyen la componente temática de la información tradicional. Los elementos clave de la base de datos son esa estructuración y sistematicidad, pues ambas son las responsables de las características que hacen de la base de datos un enfoque superior a la hora de gestionar datos. Aunque las particularidades de los datos tradicionales con los que trabajamos en un sistema de información puedan no ser interpretados directamente, se puede manipular la base de datos a nivel de programación (Olaya, 2016).

4.1.11 Sistema Gestor de Base de Datos MySQL

MySQL es un sistema de gestión de bases de datos relacionales de código abierto (RDBMS, por sus siglas en inglés) con un modelo cliente-servidor. Es un software o servicio utilizado para crear y administrar bases de datos basadas en un modelo relacional. Se adapta un modelo cliente-servidor las computadoras que tienen instalado y ejecutan el software de MySQL se llaman clientes. Siempre que necesitan acceder a los datos, se conectan al servidor Sistema Gestor de Base de Datos. Esa es la parte “cliente-servidor” (Luis, 2017).

4.1.12 Servidor local XAMPP

XAMPP es un servidor independiente de plataforma, software libre, que consiste principalmente en la base de datos MySQL, el servidor Web Apache y los intérpretes para lenguajes de script: PHP y Perl. El nombre proviene del acrónimo de X (para cualquiera de los diferentes sistemas operativos), Apache, MySQL, PHP, Perl. El programa está liberado bajo la licencia GNU y actúa como un servidor Web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris, y MacOS X (A, 2016).

XAMPP solamente requiere descargar y ejecutar un archivo .zip, .tar, o .exe, con unas pequeñas configuraciones en alguno de sus componentes que el servidor Web necesitará. XAMPP se actualiza regularmente para incorporar las últimas versiones de Apache/MySQL/PHP y Perl. También incluye otros módulos como OpenSSL y phpMyAdmin. Para instalar XAMPP se requiere solamente una pequeña fracción del tiempo necesario para descargar y configurar los programas por separado (A, 2016).

4.2 PROCEDIMIENTO / METODOLOGÍA

En este apartado se muestra la metodología aplicada en el Desarrollo de la Capa de Acceso a Datos de un Sistema de Gestión de Tunos Digitales de Atención Estudiantil con tecnología Open Source. La metodología que se utiliza en este tema es ASD (Adaptive Software Development).

El método ágil ASD (Adaptive Software Development) traducido en español significa Desarrollo Adaptable de Software es un modelo de implementación de patrones ágiles para desarrollo de software, que hace énfasis en aplicar las ideas que se originaron en el mundo de los sistemas complejos, adaptación continua del proceso al trabajo.

Las actividades realizadas en esta metodología son las siguientes: Especular, Colaborar y Aprender.

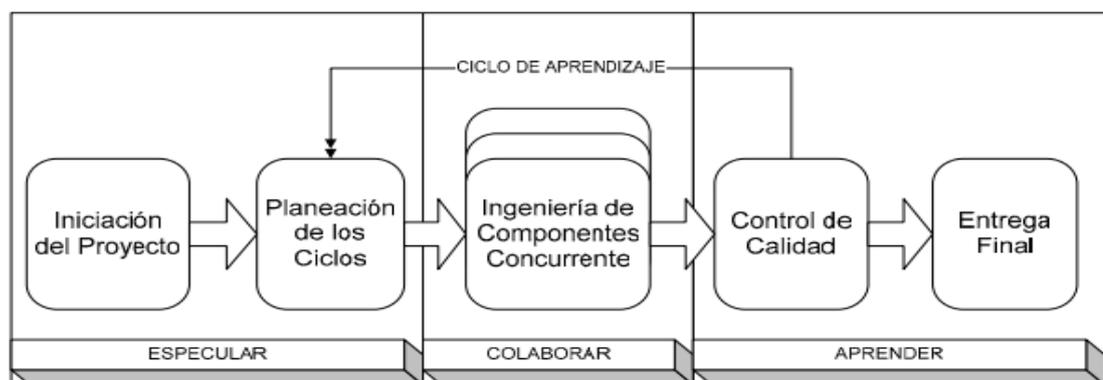


Figura 1: Recuperado de “5 beneficios del Software Adaptable”, de Capetillo, J. , (04 Enero 2017).
Recuperado de <https://www.ceesa.com/noticias/5-beneficios-del-software-adaptable/>

Para la elaboración del proyecto propuesto se utilizó el método ASD (Desarrollo Adaptable de Software), ya que permite obtener la información de necesidades del cliente en un inicio, requerimientos adicionales durante

el periodo de entregas de avances, además el desarrollador opta por las herramientas tecnológicas a utilizarse en el software, sin embargo el mismo está sujeto a las recomendaciones de desarrolladores externos para una mejor construcción del software, con la finalidad de obtener un sistema funcional que cumpla con las necesidades del cliente, el desarrollador y que de cierta forma el sistema finalizado este sujeto al modelo de mejora continua.

Especulación

Es el primer paso de iniciación para establecer y planificar las características del sistema que se va a desarrollar con el usuario, obteniendo información mediante técnicas como la entrevista, encuesta entre otros, esta fase es la más importante y no puede ser excluida ya que es la clave para el desarrollo del sistema, obteniendo la información necesaria (funcionalidades, administradores, consumidor final, alcance del sistema) para una mejor construcción y funcionalidad del sistema, cumpliendo así con las expectativas el cliente.

Definición de las necesidades que tiene el cliente.

Caso práctico: el cliente, necesita tener reportes de la información receptada dentro del sistema, para lo cual se implementó la opción de descarga de información en formato Excel y PDF.

Implementación

- Inicialización del proyecto: definición del sistema que el cliente necesita para la satisfacer sus necesidades.
- Borrador del sistema: se determina brevemente cual es la finalidad del sistema propuesto por cliente al desarrollador.
- Características del sistema: se detalla el alcance que tendrá el sistema.
- Planeación de ciclos: se crea una periodicidad de tiempo para la entrega de avances del desarrollo del sistema, al cliente.

- Necesidades del cliente: se entrega nuevas ideas de cambio al desarrollador.

Colaboración

Es este paso es donde se desarrollan las características del sistema, después del análisis de la información obtenido en la especificación, para posteriormente tener en cuenta cuáles serán las funciones primordiales para el funcionamiento adecuado del sistema que cubra todas sus necesidades. Definiendo así cuáles serán los detalles de cada proceso para permitir su interpretación y realización física con la aplicación de lenguajes de programación y sistema gestor de base de datos.

Implementación

- Desarrolladores internos: serán los encargados de escoger las herramientas tecnológicas que se utilizar para desarrollar el sistema y que cumplan las necesidades del cliente.
- Herramientas a utilizar: es el tipo de tecnología que se utilizara para el desarrollo del proyecto.
- Desarrolladores externos: se abre el paso a recomendaciones técnicas, de otros desarrolladores que no están relacionados con el proyecto abriendo el paso a mejoras en la construcción del software.

Búsqueda de herramientas tecnológicas que cumplan con las necesidades del cliente.

Caso práctico: el cliente necesita un software no tan costoso, ni que tenga que pagar por periodos por la utilización del software, para cubrir esta necesidad los desarrolladores implementan tecnología Open Source en su totalidad, utilizando: lenguajes de programación como: PHP, AJAX, JavaScript y como motor de base de datos MySQL, que constantemente son actualizados, teniendo como finalidad ser adaptable a diferentes tipos de plataformas web.

Aprendizaje

Se realiza pruebas del funcionamiento del sistema mediante posibles fallas del sistema, para garantizar que el sistema cuenta con planes de contingencia hacia posibles errores; es decir se revisa la calidad del sistema como cual y que esté libre de errores.

En la etapa se identifica tres tipos de aprendizaje

- Calidad del producto desde un punto de vista del cliente: Se realiza pruebas de funcionamiento, dando a conocer la funcionalidad del sistema al cliente.
- Calidad del producto desde un punto de vista de los desarrolladores. Se trata de la evaluación de la calidad de los productos desde un punto de vista técnico.
- La gestión del rendimiento. Es un proceso de evaluación al cliente para saber si el sistema desarrollado cumple sus expectativas al momento de realizar un proceso.

Implementación

- Funcionamiento del sistema: una vez concluido el sistema y después de haber pasado por la Planeación de ciclos se procederá a verificar que el sistema cumpla con las necesidades planteadas inicialmente.
- Punto de vista de calidad, desde la perspectiva del cliente: finalizado el software se entrega el producto final al cliente y que el mismo observe la funcionalidad del mismo, a nivel estético y de procesos.
- Punto de vista de calidad, desde la perspectiva de desarrolladores internos y externos: una vez finalizado el software se realiza la funcionalidad y lectura de código por parte de desarrolladores externos hacia los desarrolladores internos constatando que el software cumple con su funcionamiento correctamente, es adaptable, y está sujeto al modelo continua.
- Planes de contingencia: son pruebas que realiza el cliente al sistema en caso de información errónea midiendo la capacidad de respuesta del mismo.

- Rendimiento del sistema: una vez finalizado el software se realiza las pruebas de contingencia, de funcionalidad correcta, que cumplan normas de calidad y sobre todo las necesidades finales del cliente.

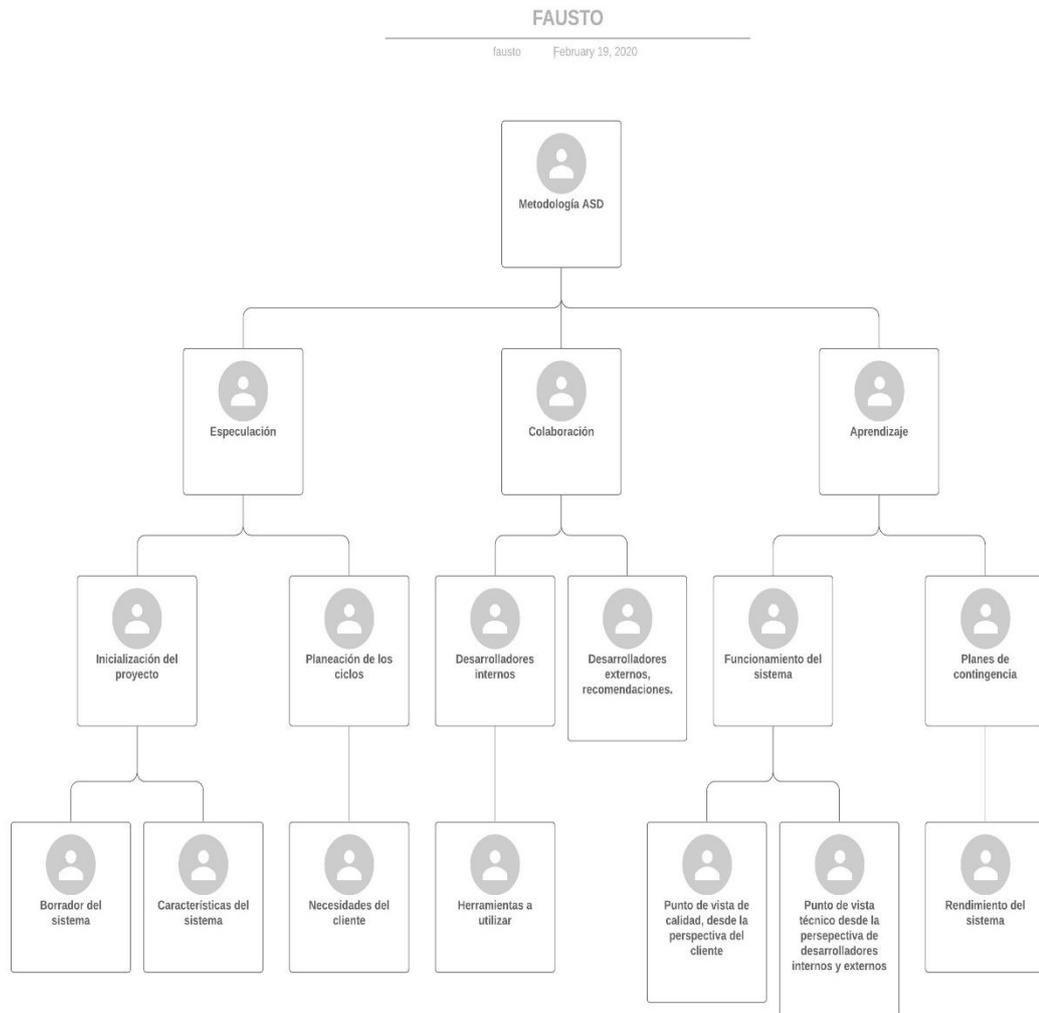


Figura 2: Metodología ASD
Fuente: Fausto Almeida

Conclusión de implementación

Una vez implementado las fases de la metodología ASD, se puede concluir que utilizando este método, se obtendrán resultado favorables al momento de desarrollar un software, ya que para una fase inicial nos permite recopilar información directa del cliente al desarrollador, de igual manera al momento de exponer la factibilidad técnica podremos expresarle al cliente el porqué de nuestra elección, teniendo como resultado final un software final de calidad,

que cumpla con las necesidades del cliente y el mismo este sujeto al modelo de mejora continua.

4.2.1 Diseño

El Desarrollo de la Capa de Acceso a Datos de un Sistema de Gestión de Turnos Digitales de Atención Estudiantil con Tecnología Open Source, se ha realizado de la siguiente manera.

Diseño de procedimientos almacenados en base de datos.

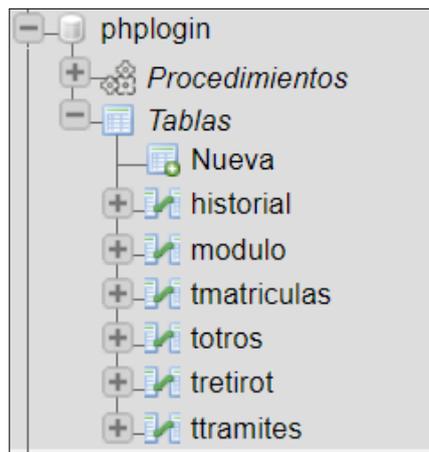


Figura 3: Diseño de base de datos
Fuente: Los investigadores

Script de rutinas realizadas desde PhpMyAdmin

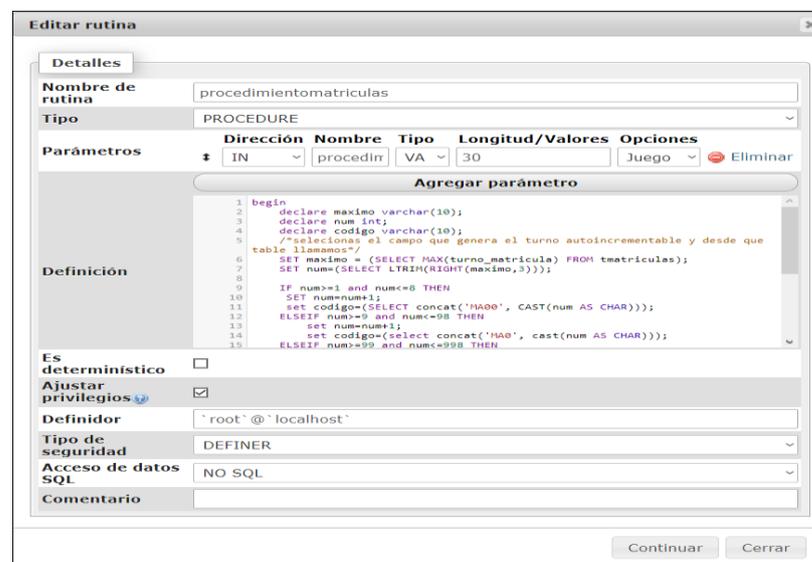


Figura 4: Procedimientos almacenados en base de datos
Fuente: Los investigadores

Estructuras de Base de Datos, no se definen relación, ya que los procedimientos realizan un JOIN de datos para la inserción de datos.

Nombre de Tabla	Columnas y Tipos de Datos
phplogin ttramites	turno_tramites : varchar(10) usuario : varchar(64) hora_tramites : timestamp aten_tra : int(1) estado : varchar(20)
phplogin tretiro	turno_retiro : varchar(10) usuario : varchar(64) hora_retiro : timestamp aten_re : int(1) estado : varchar(20)
phplogin modulo	codigo_r : varchar(10)
phplogin tmatriculas	turno_matricula : varchar(10) usuario : varchar(64) hora_matricula : timestamp aten_mat : int(1) estado : varchar(15)
phplogin totros	turno_otros : varchar(10) usuario : varchar(64) hora_otros : timestamp aten_ot : int(1) estado : varchar(20)
phplogin historial	idHistorial : int(11) tipo : varchar(100) turno : varchar(100) usuario : varchar(100) fecha : datetime aten_mat : int(1) estado : varchar(20)

Figura 5: Tablas creadas en la base de datos
Fuente: Los investigadores

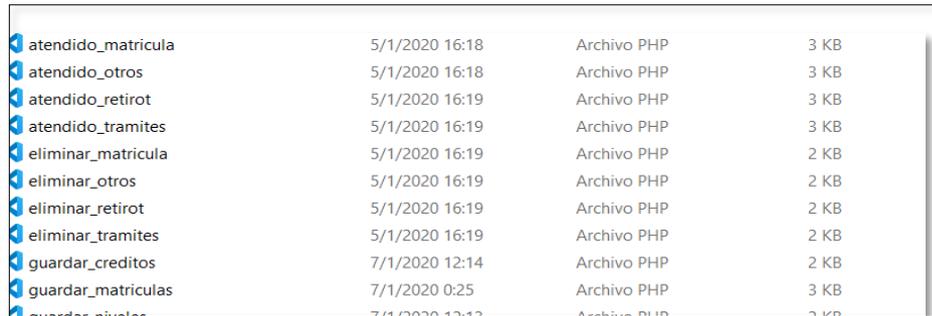
Repositorio de carpeta contenedora del software.

Nombre	Fecha de modificación	Tipo	Tamaño
ajax	7/1/2020 0:24	Carpeta de archivos	
Backups	7/1/2020 0:24	Carpeta de archivos	
Classes	7/1/2020 0:24	Carpeta de archivos	
conexion	7/1/2020 0:24	Carpeta de archivos	
css	7/1/2020 16:07	Carpeta de archivos	
fpdf	7/1/2020 0:24	Carpeta de archivos	
Gestion	7/1/2020 0:24	Carpeta de archivos	
historial	7/1/2020 18:01	Carpeta de archivos	
html	7/1/2020 18:00	Carpeta de archivos	
img	7/1/2020 18:02	Carpeta de archivos	
js	7/1/2020 13:15	Carpeta de archivos	
vistamodulos	9/1/2020 16:07	Carpeta de archivos	
conexion	5/1/2020 16:18	Archivo PHP	1 KB
index	7/1/2020 18:53	Archivo PHP	4 KB
infor	7/1/2020 19:21	Chrome HTML Do...	9 KB
License	5/1/2020 16:18	Archivo	2 KB
turno_vn	6/1/2020 13:30	Archivo PHP	6 KB

Figura 6: Directorio raíz del software
Fuente: Los investigadores

Directorio Ajax

Podemos encontrar las acciones del software.

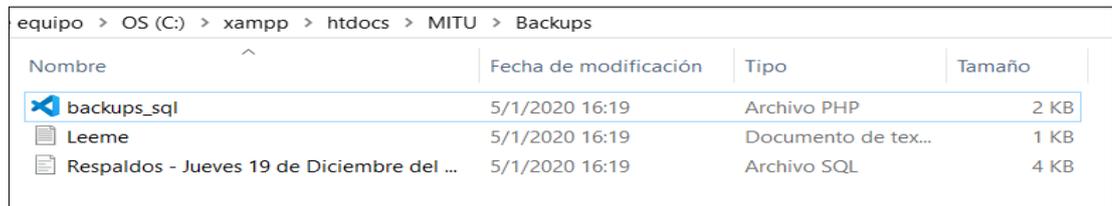


Nombre	Fecha de modificación	Tipo	Tamaño
atendido_matricula	5/1/2020 16:18	Archivo PHP	3 KB
atendido_otros	5/1/2020 16:18	Archivo PHP	3 KB
atendido_retiro	5/1/2020 16:19	Archivo PHP	3 KB
atendido_tramites	5/1/2020 16:19	Archivo PHP	3 KB
eliminar_matricula	5/1/2020 16:19	Archivo PHP	2 KB
eliminar_otros	5/1/2020 16:19	Archivo PHP	2 KB
eliminar_retiro	5/1/2020 16:19	Archivo PHP	2 KB
eliminar_tramites	5/1/2020 16:19	Archivo PHP	2 KB
guardar_creditos	7/1/2020 12:14	Archivo PHP	2 KB
guardar_matriculas	7/1/2020 0:25	Archivo PHP	3 KB
guardar_niveles	7/1/2020 12:12	Archivo PHP	2 KB

Figura 7: Contenido de directorio Ajax
Fuente: Los investigadores

Directorio Backups

Podemos encontrar los respaldos automáticos que realizara el sistema diariamente

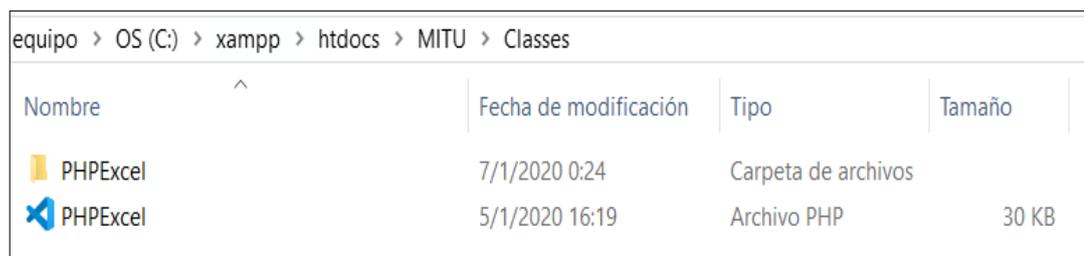


Nombre	Fecha de modificación	Tipo	Tamaño
backups_sql	5/1/2020 16:19	Archivo PHP	2 KB
Leeme	5/1/2020 16:19	Documento de texto	1 KB
Respaldos - Jueves 19 de Diciembre del ...	5/1/2020 16:19	Archivo SQL	4 KB

Figura 8: Contenido de directorio Backups
Fuente: Los investigadores

Directorio Classes

Podemos encontrar la clase que nos permitirá exportar las consultas de datos a formato Excel.



Nombre	Fecha de modificación	Tipo	Tamaño
PHPExcel	7/1/2020 0:24	Carpeta de archivos	
PHPExcel	5/1/2020 16:19	Archivo PHP	30 KB

Figura 9: Contenido de directorio Classes
Fuente: Los investigadores

Directorio Conexión

Encontramos el script de conexión entre la base de datos con el PHP.

Nombre	Fecha de modificación	Tipo	Tamaño
conexionexcel	5/1/2020 16:19	Archivo PHP	2 KB
conexionguardar	5/1/2020 16:19	Archivo PHP	1 KB

Figura 10: Contenido de directorio Conexión

Fuente: Los investigadores

Directorio CSS

Podremos encontrar los estilos de diseño estético de nuestro software.

Nombre	Fecha de modificación	Tipo	Tamaño
custom	5/1/2020 16:19	Documento de hoj...	6 KB
informacion	7/1/2020 18:48	Documento de hoj...	4 KB
informacion.min	7/1/2020 16:07	Documento de hoj...	13 KB

Figura 11: Contenido de directorio CSS

Fuente: Los investigadores

Directorio FPDF

Podemos encontrar la clase para poder exportar las consultas de base de datos en formato .PDF*.

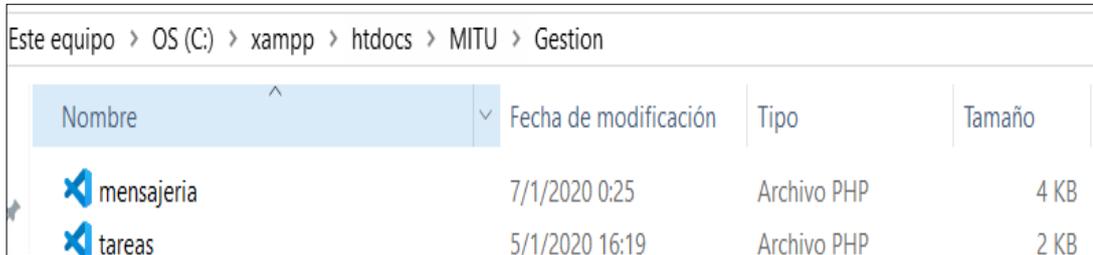
Nombre	Fecha de modificación	Tipo	Tamaño
font	7/1/2020 0:24	Carpeta de archivos	
makefont	7/1/2020 0:24	Carpeta de archivos	
fpdf	5/1/2020 16:19	Archivo PHP	49 KB

Figura 12: Contenido de directorio FPDF

Fuente: Los investigadores

Directorio Gestión

Podemos encontrar los script para mensajería móvil y las tareas automáticas del sistema.



Nombre	Fecha de modificación	Tipo	Tamaño
mensajería	7/1/2020 0:25	Archivo PHP	4 KB
tareas	5/1/2020 16:19	Archivo PHP	2 KB

Figura 13: Contenido de directorio Gestión
Fuente: Los investigadores

Directorio Historial

Podremos encontrar los scripts de la venta historial y funciones que incluye la misma dentro del sistema.

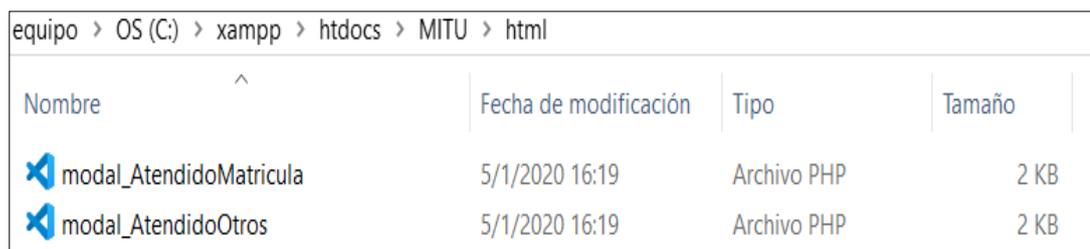


Nombre	Fecha de modificación	Tipo	Tamaño
excelautomatico	5/1/2020 16:19	Archivo PHP	2 KB
exportarExcel	5/1/2020 16:19	Archivo PHP	6 KB
historial	6/1/2020 13:31	Archivo PHP	5 KB
historialDatos	5/1/2020 16:19	Archivo PHP	1 KB
index	5/1/2020 16:19	Archivo PHP	2 KB
pdf	5/1/2020 16:19	Archivo PHP	6 KB

Figura 14: Contenido de directorio Historial
Fuente: Los investigadores

Directorio HTML

Encontraremos los modales de inserción de datos del sistema.



Nombre	Fecha de modificación	Tipo	Tamaño
modal_AtendidoMatricula	5/1/2020 16:19	Archivo PHP	2 KB
modal_AtendidoOtros	5/1/2020 16:19	Archivo PHP	2 KB

Figura 15: Contenido de directorio HTML
Fuente: Los investigadores

Directorio IMG

Podremos encontrar las imágenes utilizadas en la interfaz gráfica del sistema.

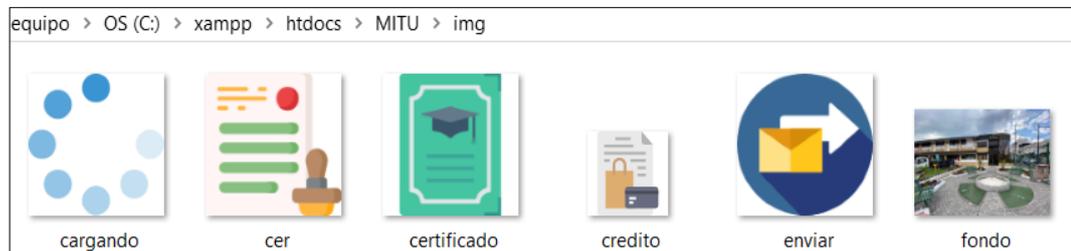


Figura 16: Contenido de directorio IMG
Fuente: Los investigadores

Directorio JS

Podremos encontrar los archivos JavaScript desarrollados para el funcionamiento del sistema.

Nombre	Fecha de modificación	Tipo	Tamaño
main	10/1/2020 15:09	Archivo JavaScript	7 KB
script	10/1/2020 12:20	Archivo JavaScript	5 KB
scripthistorial	10/1/2020 18:05	Archivo JavaScript	6 KB
scriptmatricula	10/1/2020 12:21	Archivo JavaScript	4 KB
scriptotros	10/1/2020 12:21	Archivo JavaScript	3 KB
scriptretiro	10/1/2020 12:21	Archivo JavaScript	3 KB
scriptsmodals	10/1/2020 12:21	Archivo JavaScript	7 KB
scripttramite	10/1/2020 12:21	Archivo JavaScript	3 KB

Figura 17: Contenido de directorio JS
Fuente: Los investigadores

Directorio Vistamodulos

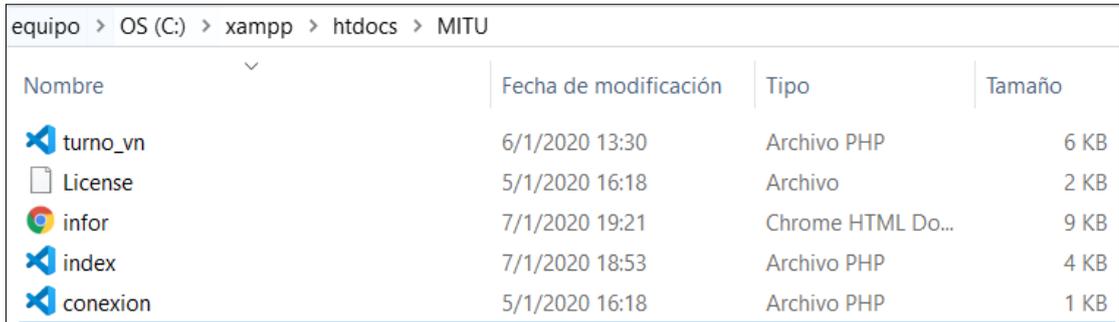
Podremos encontrar las ventanas necesarias para que el sistema, interactúe con su interfaz gráficamente.

Nombre	Fecha de modificación	Tipo	Tamaño
conexion	5/1/2020 16:19	Archivo PHP	1 KB
index	9/1/2020 16:44	Archivo PHP	3 KB
vista	5/1/2020 16:19	Archivo PHP	3 KB
vistar	9/1/2020 16:04	Archivo PHP	4 KB
vistar_t	9/1/2020 15:45	Archivo PHP	6 KB

Figura 18: Contenido de directorio Vistamodulos
Fuente: Los investigadores

Directorio Raíz

Archivo de inicio, gráfico para la www.



Nombre	Fecha de modificación	Tipo	Tamaño
turno_vn	6/1/2020 13:30	Archivo PHP	6 KB
License	5/1/2020 16:18	Archivo	2 KB
infor	7/1/2020 19:21	Chrome HTML Do...	9 KB
index	7/1/2020 18:53	Archivo PHP	4 KB
conexion	5/1/2020 16:18	Archivo PHP	1 KB

Figura 19: Contenido de directorio Raíz
Fuente: Los investigadores

4.2.2 Construcción

Para la parte de la construcción del Desarrollo de la Capa de Acceso a Datos de un Sistema de Gestión de Turnos Digitales de Atención con tecnología Open Source, se ha realizado de la siguiente manera.

Definición de Software necesario.

- Entorno de programación: Visual Studio Code (Version 1.41.1).
- Servidor local: XAMPP (Version 3.24 Junio 5th 2019).
- Lenguaje de maquetado: HTML (Version 5).
- Leguaje de Programación: PHP (Version 7.2), JavaScript.
- Sistema Gestor de Base de Datos: MySQL (Version 8.0.18).
- Administrador de Base de Datos: PhpMyAdmin (Version 5.0.1).

4.2.2.1 Construcción de base de datos.

Se creó la base de datos desde PhpMyAdmin vía Browser (Navegador de Internet).



Figura 20: Creación de base de datos
Fuente: Los investigadores

Se creó los procedimientos o rutinas necesarios para el software en la base de datos desde PhpMyAdmin.

Nombre	Acción	Tipo	Retorna
<input type="checkbox"/> procedimientomatriculas	Editar Ejecutar Exportar Eliminar	PROCEDURE	
<input type="checkbox"/> procedimientootros	Editar Ejecutar Exportar Eliminar	PROCEDURE	
<input type="checkbox"/> procedimientoretiro	Editar Ejecutar Exportar Eliminar	PROCEDURE	
<input type="checkbox"/> procedimientotramites	Editar Ejecutar Exportar Eliminar	PROCEDURE	

Figura 21: Creación de rutinas en PhpMyAdmin
Fuente: Los investigadores

Definición de rutinas de base de datos MySQL.

Nombre de rutina: procedimientomatriculas
Tipo: PROCEDURE

Parámetros:

Dirección	Nombre	Tipo	Longitud/Valores	Opciones
IN	procedin	VA	30	Juego

Definición:

```

1 begin
2   declare maximo varchar(10);
3   declare num int;
4   declare codigo varchar(10);
5   /*seleccionas el campo que genera el turno autoincrementable y desde que
6   table llamamos*/
7   SET maximo = (SELECT MAX(turno_matricula) FROM tmatriculas);
8   SET num=(SELECT LTRIM(RIGHT(maximo,3)));
9
10  IF num>=1 and num<=8 THEN
11    SET num=num+1;
12    set codigo=(SELECT concat('MA00', CAST(num AS CHAR)));
13  ELSEIF num>=9 and num<=98 THEN
14    set num=num+1;
15    set codigo=(select concat('MA0', cast(num AS CHAR)));
16  ELSEIF num>=99 and num<=998 THEN

```

Figura 22: Script desarrollado en rutinas.
Fuente: Los investigadores

Estructura de tablas necesarias para el software en base de datos.

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> historial	Examinar Estructura Buscar Insertar Vaciar Eliminar	11	InnoDB	latin1_swedish_ci	16 KB	-
<input type="checkbox"/> modulo	Examinar Estructura Buscar Insertar Vaciar Eliminar	2	InnoDB	latin1_swedish_ci	16 KB	-
<input type="checkbox"/> tmatriculas	Examinar Estructura Buscar Insertar Vaciar Eliminar	2	InnoDB	utf8_general_ci	16 KB	-
<input type="checkbox"/> totros	Examinar Estructura Buscar Insertar Vaciar Eliminar	2	InnoDB	utf8_general_ci	16 KB	-
<input type="checkbox"/> tretirot	Examinar Estructura Buscar Insertar Vaciar Eliminar	2	InnoDB	utf8_general_ci	16 KB	-
<input type="checkbox"/> ttramites	Examinar Estructura Buscar Insertar Vaciar Eliminar	4	InnoDB	utf8_general_ci	16 KB	-
6 tablas	Número de filas	23	InnoDB	utf8_general_ci	96 KB	0 B

Figura 23: Definición de tablas necesarias para el software
Fuente: Los investigadores

4.2.2.2 Procedimientos almacenados a nivel de base de datos

Un procedimiento almacenado o rutina de base de datos, permite crear una relación entre tablas ya definidas, es así que se creó una procedimientos para la asignación de turnos, con su distinto modulo y trámite correspondiente.

En este procedimiento creamos variables globales para la asignación de turnos a sus distintas tablas, contando con un contador ascendente acompañado por una variable.

```
begin
declare maximo varchar(10);
declare num int;
declare codigo varchar(10);
/* Seleccionas el campo que genera el turno autoincrementable y
desde que tabla llamamos */
SET maximo = (SELECT MAX(turno_matricula) FROM
tmatriculas);
SET num=(SELECT LTRIM(RIGHT(maximo,3)));
IF num>=1 and num<=8 THEN
SET num=num+1;
SET codigo=(SELECT concat('MA00', CAST(num AS CHAR)));
ELSEIF num>=9 and num<=98 THEN
set num=num+1;
set codigo=(select concat('MA0', cast(num AS CHAR)));
ELSEIF num>=99 and num<=998 THEN
set num=num+1;
set codigo=(select concat('MA',cast(num AS CHAR)));
```

*Figura 24: Script textual de procedimientos almacenados
Fuente: Los investigadores*

4.2.2.3 Interfaz de recepción de contenido

Para receptar la información obtendremos unos modales para receptar la información de acuerdo a la solicitud del usuario.

Variables

Nombres =X

Teléfono= X

Modal de recepción de datos con el título "Ingrese sus datos". El modal contiene un ícono de documentos con una bandera roja. Hay dos campos de entrada de texto: "Nombres" y "Teléfono". En la parte inferior derecha hay un botón azul que dice "Gestionar Turno".

Figura 25: Modal de recepción de datos
Fuente: Los investigadores

4.2.2.4 Cabeceras de contenido de tablas desarrolladas en la base de datos.

Las tablas tmatriculas, totros, tretiro, ttramites, cuenta con la misma estructura siendo los encargados de recibir la información directamente desde la interfaz gráfica.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1	turno_tramites	varchar(10)	utf8_general_ci	No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	2	usuario	varchar(64)	utf8_general_ci	No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	3	hora_tramites	timestamp		No	current_timestamp()			Cambiar Eliminar Más
<input type="checkbox"/>	4	aten_tra	int(1)		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	5	estado	varchar(20)	utf8_general_ci	No	Ninguna			Cambiar Eliminar Más

Figura 26: Cabeceras de contenido de tablas
Fuente: Los investigadores

Valores	Título	Descripción
Columna 1	turno_tramite	Recepción de número de turno en forma ordenada.

Columna 2	usuario	Información personal del usuario.
Columna 3	hora_tramites	Hora de asignación del turno.
Columna 4	aten_tra	Valor numérico para saber si está el usuario atendido o no.
Columna 5	estado	Estado de turno necesario para reportes en interfaz gráfica.

Tabla 1: Contenido de la tabla trámites
Fuente: Los investigadores

+ Opciones		turno_tramites	usuario	hora_tramites	aten_tra	estado
<input type="checkbox"/>	Editar Copiar Borrar	TR001	RENE FONSECA	2020-01-03 14:03:53	0	NO ATENDIDO
<input type="checkbox"/>	Editar Copiar Borrar	TR002	FAUSTO ALMEIDA	2020-01-08 08:57:38	0	NO ATENDIDO
<input type="checkbox"/>	Editar Copiar Borrar	TR003	JONATHAN ARIAS	2020-01-09 16:30:33	0	NO ATENDIDO
<input type="checkbox"/>	Editar Copiar Borrar	TR004	LUIS EDISON	2020-01-09 16:30:46	0	NO ATENDIDO

Figura 27: Contenido de tablas, con tipo de valores.
Fuente: Los investigadores

4.2.2.5 Cabeceras y contenido de tabla historial desarrollada en la base de datos.

La tabla historial es la encargada de receptor la información de las tablas secundarias, que gestionan los demás trámites, teniendo como propósito receptor información en tiempo real, y llevar un histórico de información.

Valores	Titulo	Descripción
Columna 1	idHistorial	Numeración ascendente única ordenada según los turnos registrados globalmente.
Columna 2	Tipo	Tipo de trámite que se gestiona, en el sistema.
Columna 3	Turno	Recepción de número de turno en forma ordenada, de cualquier turno de acuerdo al ingreso de información.

Columna 4	usuario	Información personal del usuario.
Columna 5	fecha	Hora y fecha exacta de recepción del turno desde la tabla que gestione el mismo.
Columna 6	aten_mat	Valor numérico para saber si está el usuario atendido o no.
Columna 7	estado	Estado de turno necesario para reportes en interfaz gráfica.

Tabla 2: Contenido de la tabla historial
Fuente: Los investigadores

✓ Mostrando filas 0 - 10 (total de 11, La consulta tardó 0,0012 segundos.)

SELECT * FROM `historial`

Perfilando [Editar en línea] [Editar] [E]

Mostrar todo | Número de filas: 25 | Filtrar filas: Ordenar según la clave: Ninguna

Opciones

	idHistorial	tipo	turno	usuario	fecha	aten_mat	estado
<input type="checkbox"/> Editar Copiar Borrar	1	TRAMITES	TR001	RENE FONSECA	2020-01-03 14:03:53	0	NO ATENDIDO
<input type="checkbox"/> Editar Copiar Borrar	2	MATRICULAS	MA001	ESTEFANIA MANCHENO	2020-01-03 14:04:07	0	NO ATENDIDO
<input type="checkbox"/> Editar Copiar Borrar	3	RETIRO TRAMITE	RT001	FERNANDO PINCAY	2020-01-03 14:04:26	0	NO ATENDIDO
<input type="checkbox"/> Editar Copiar Borrar	4	OTROS	OT001	JUAN CARLOS JACHO	2020-01-03 14:04:43	0	NO ATENDIDO
<input type="checkbox"/> Editar Copiar Borrar	5	TRAMITES	TR002	FAUSTO ALMEIDA	2020-01-08 08:57:38	0	NO ATENDIDO
<input type="checkbox"/> Editar Copiar Borrar	6	MATRICULAS	MA001	FONSECA RENE	2020-01-08 08:58:37	0	NO ATENDIDO
<input type="checkbox"/> Editar Copiar Borrar	7	TRAMITES	TR003	JONATHAN ARIAS	2020-01-09 16:30:33	0	NO ATENDIDO
<input type="checkbox"/> Editar Copiar Borrar	8	TRAMITES	TR004	LUIS EDISON	2020-01-09 16:30:46	0	NO ATENDIDO
<input type="checkbox"/> Editar Copiar Borrar	9	MATRICULAS	MA002	ELI LARA	2020-01-10 14:57:36	0	NO ATENDIDO
<input type="checkbox"/> Editar Copiar Borrar	10	RETIRO TRAMITE	RT002	DARWIN GUANO	2020-01-10 14:57:49	0	NO ATENDIDO
<input type="checkbox"/> Editar Copiar Borrar	11	OTROS	OT002	HECTOR GARCIA	2020-01-10 14:58:03	0	NO ATENDIDO

Figura 28: Contenido de tabla historia en PhpMyAdmin
Fuente: Los investigadores

4.2.2.6 Cabeceras y contenido de tabla módulo desarrollada en la base de datos.

La tabla modulo es la encargada de hacer un JOIN con las tablas secundarias, que gestionan los demás trámites, teniendo como propósito indicar el modulo donde el usuario tiene que acceder.

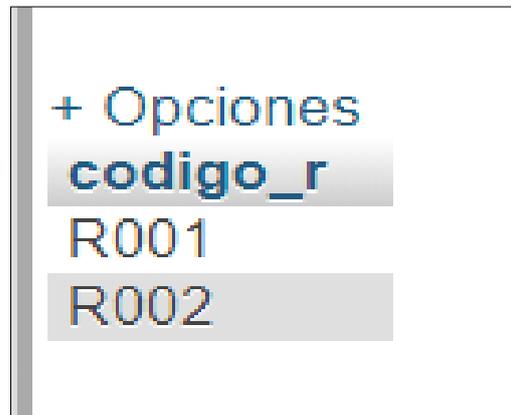


Figura 29: Estructura de tabla modulo
Fuente: Los investigadores

Valor	Titulo	Descripción
Columna 1	codigo_r	Asignación de número de módulos.

Tabla 3: Contenido de la tabla módulo
Fuente: Los investigadores

Script de conexión a la base de datos desde PHP.

```
1 <?php
2 $conexion=new mysqli('localhost','root','','phplogin');
3 if($conexion){
4     echo"";
5 }else{
6     echo"Error de conexion";
7 }
8 ?>
```

Figura 30: Script de conexión a base de datos
Fuente: Los investigadores

Entendimiento de programación.

Valor	Descripción
localhost	Servidor donde está alojado nuestra base de datos, el servidor web puede ser local o desde un servidor de internet.
Root	Usuario con privilegios en la base de datos
'	Contraseña de la base de datos
Phplogin	Nombre de la base de datos, que se afectara

Tabla 4: Entendimiento de programación
Fuente: Los investigadores

4.2.2.7 Consultas a base de datos desde PHP.

Las consultas a la base de datos, se realiza por medio de interfaz gráfica y a nivel de programación, para el software desarrollado se utilizó procedimientos que interactúan entre las tablas directamente.

```

1  <?php
2  require_once("../Gestion/mensajeria.php");
3  require_once("../conexion/conexionguardar.php");
4
5      if($conexion->connect_error){
6          die("La conexion falló:" . $conexion->connect_error);
7      }else{
8          /*LLAMAMOS AL PROCEDIMIENTO QUE REALIZARA NUESTRO GESTIONAMIENTO DE TURNOS Y EL CAMPO DONDE SE HARA LA INSERCCION*/
9          $query =mysql_query($conexion,"CALL procedimientomatriculas('$_POST[turno_matricula]')");
10
11      if ($conexion->query($query)===TRUE){
12
13      }

```

Figura 31: Script de consulta base de datos desde PHP
Fuente: Los investigadores

Entendimiento de consulta

Valores	Descripción
CALL procedimientomatrícula	Invocamos a nuestra rutina creada en la base de datos para la inserción de datos en la tabla correspondiente.
\$_POST[turno_matricula]	Recuperamos la información desde la interfaz gráfica para poder insertarla en el

	campo correcto en la base de datos.
--	-------------------------------------

Tabla 5: Entendimiento de consultas
Fuente: Los investigadores

4.2.2.8 Desarrollo del modelo de transferencia padre e hijo con JavaScript

El modelo de transferencia padre e hijo con JavaScript nos permite controlar una ventana secundaria a partir de una principal.

```

1  (function() {
2
3      /*Declaramos variable*/
4      var VentanaHija;
5      var Historial;
6      var audioElement = document.createElement('audio');
7
8      /*Encerramos las acciones de los botones del html */
9      document.getElementById('btnAbrir').onclick = AbrirVentanaHija;
10     document.getElementById('btnCerrar').onclick = CerrarVentanaHija;
11     document.getElementById('btnActualizar').onclick = ActualizarVentanaHija;
12     document.getElementById('btnHistorial').onclick = AbrirHistorial;
13     document.getElementById('btnCerrarHistorial').onclick = CerrarHistorial;
14     audioElement.setAttribute('src', 'img/uno.mp3');
15
16     /*Creamos las funciones para realizar las acciones Abrir */
17     function AbrirVentanaHija() {
18         if (VentanaHija) {
19             audioElement.play();
20             var toast = document.getElementById("abrirt");
21             toast.className = "mostrar";
22             setTimeout(function() { toast.className = toast.className.replace("mostrar", ""); }, 2500);
23         } else {
24             VentanaHija = window.open('http://localhost/MITU/vistamodulos/');
25         }
26     }
27     /*Creamos las funciones para realizar las acciones Cerrar */
28     function CerrarVentanaHija() {
29         if (!VentanaHija) {
30             var toast = document.getElementById("CerrarTurnero");
31             toast.className = "mostrar";
32             setTimeout(function() { toast.className = toast.className.replace("mostrar", ""); }, 2500);
33         } else {
34             VentanaHija.close();
35             VentanaHija = undefined;
36         }
37     }
38 }

```

Figura 32: Modelo de transferencia Padre e Hijo en JavaScript
Fuente: Los investigadores

Entendimiento de programación de modelo transferencia padre e hijo con JavaScript.

Valores	Descripción
document.getElementById	Este atributo de JavaScript nos permite asignar una acción a un botón al momento

	de hacer clic, por medio de su id asignado en el HTML correspondiente.
function AbrirVentanaHija()	La función AbrirVentanaHija permite crear un punto de partida, asignando la url al botón asignado para realizar esta función, y controlar sus acciones desde el mismo, guardando cambios en tiempo real.
function CerrarVentanaHija()	La función CerrarVentanaHija asignada a un botón desde la misma interfaz de donde se controla la función AbrirVentanaHija permite cerrar la url abierta desde ahí.
window.open	Es la asignación correspondiente para que el botón controle una url.
VentanaHija.close()	Es la asignación correspondiente para que el botón controle la url, asignada por la ventana principal.

Tabla 6: Modelo de transferencia padre e hijo con JS
Fuente: Los investigadores

4.2.2.9 Desarrollo HTML

El lenguaje HTML nos permite otorgar los atributos correspondientes a cada botón para su interacción con JavaScript.

```

<div class="row">
  <div class="col col-xs-2 pull-right" style="padding:0.3px;">
    <a id='btnCerrarHistorial' class="btn btn-danger" title="Cerrar Historial"><i class="material-icons">highlight_off</i></a>
    <a id='btnHistorial' class="btn btn-success" title="Abrir Historial"><i class="material-icons">line_style</i></a>
  </div>
  <div class="col col-xs-2 pull-right" style="padding:0.3px;">
    <a id='btnCerrar' class="btn btn-danger" title="Cerrar Turnero"><i class="material-icons">label_off</i></a>
    <a id='btnActualizar' class="btn btn-primary" title="Actualizar Turnero"><i class="material-icons">restore_page</i></a>
    <a id='btnAbrir' class="btn btn-success" title="Abrir Turnero"><i class="material-icons">label</i></a>
  </div>

```

Figura 33: Asignación de atributos HTML
Fuente: Los investigadores

Entendimiento HTML

Valores	Descripción
id	Etiqueta HTML, que permite asignarle un atributo de concatenación al elemento con JavaScript para su interacción.

Tabla 7: Entendimiento HTML

Fuente: Los investigadores

4.2.2.10 Desarrollo de módulos a nivel de programación.

Para la interacción entre JavaScript y PHP se utiliza Ajax, para tener resultados en tiempo real de la siguiente manera, se realiza etiquetado HTML para invocar el JS.

```

<div class='clearfix'></div>
<hr>
<div id="tramite"></div><!-- Carga de datos ajax aqui -->
<div id="resultados_tramites"></div><!-- Carga de datos ajax aqui -->
<div class='Tramites'></div><!-- Carga de datos ajax aqui -->

<div id="matricula"></div><!-- Carga de datos ajax aqui -->
<div id="resultados_matricula"></div><!-- Carga de datos ajax aqui -->
<div class='Matriculas'></div><!-- Carga de datos ajax aqui -->

<div id="retiro"></div><!-- Carga de datos ajax aqui -->
<div id="resultados"></div><!-- Carga de datos ajax aqui -->
<div class='outer_divRetiroT'></div><!-- Carga de datos ajax aqui -->

<div id="otros"></div><!-- Carga de datos ajax aqui -->
<div id="resultados_otros"></div><!-- Carga de datos ajax aqui -->
<div class='Otro'></div><!-- Carga de datos ajax aqui -->

```

Figura 34: Script de módulos del software en HTML

Fuente: Los investigadores

Entendimiento módulos.

Valores	Descripción
id	Los atributos id, permite que el JavaScript con el que se concadene traer los resultados en tiempo real, y asignarlos en una posición determinada.
div	Segmento de HTML, que crea un contendedor de información, resultante de script PHP o JavaScript.

Tabla 8: Entendimiento de módulos

Fuente: Los investigadores

4.2.2.11 Consumo de datos en tiempo real con Ajax.

Ajax nos permite consumir datos en tiempo real, sin la necesidad de recargar la página para eso necesita interactuar entre un JavaScript y archivo PHP.

```
$(function() {
    loadM(1);
});

function loadM(page, reload_on_return) {
    var query = $("#q").val();
    var per_page = 5;
    var parametros = { "action": "ajax", "page": page, 'query': query, 'per_page': per_page };
    $("#matricula").fadeIn('slow');
    $.ajax({
        url: 'ajax/listar_matriculas.php',
        data: parametros,
        beforeSend: function(objeto) {
            $("#matricula").html("Cargando...");
        },
        success: function(data) {
            $(".Matriculas").html(data).fadeIn('slow');
            $("#matricula").html("");
            if (reload_on_return) {
                setTimeout(
                    function() {
                        location.reload();
                    }, 10);
            }
        }
    })
}
```

Figura 35: Script de consumo de datos real Ajax
Fuente: Los investigadores

Entendimiento de programación

Valores	Descripción
parámetros	Nos permite explicarle al navegador, que acción va a realizar cuando cambiemos datos, en este caso le indicamos que la acción es Ajax, en la misma página, sin recargar la página.
url	Es el archivo PHP que realizara la acción necesaria en segundo plano, sin tener que recargar la página o hacerle una solicitud directa.
data	Es la respuesta que tendrá Ajax, de acuerdo a los parámetros solicitados.

success	Etiqueta de finalización y respuesta que tendrá en el HTML, cuando la acción de Ajax esté terminada.
beforeSend	Envío de datos consumidos por el archivo PHP.

Tabla 9: Entendimiento de programación Ajax
Fuente: Los investigadores

4.2.3 Implementación

Para la implementación del software se ha trabajado con los siguientes requerimientos de software.

Programa	Características Mínimas	Características Recomendadas
Sistema Operativo	Windows 8.1 x64	Windows 10 x64
Servidor Local	WAMPP	XAMPP
Sistema Gestor de Base de Datos	MariaDB 10.1	MySQL
Navegadores Web	Opera Browser	<ul style="list-style-type: none"> • Google Chrome Browser • Opera GX Browser • Mozilla Firefox
PHP	Version 7.0	Version 7.0 o superior
Internet	Internet	Internet

Tabla 10: Requerimientos del sistema de gestión de turnos MITU
Fuente: Los investigadores

4.2.3.1 Instalación del Sistema

El Software viene comprimido en un formato .zip, para instalarlo no basta con dirigirnos a la ruta de nuestro servidor local en nuestro caso XAMPP, C:\xampp\htdocs_y extraerlos en ese directorio.

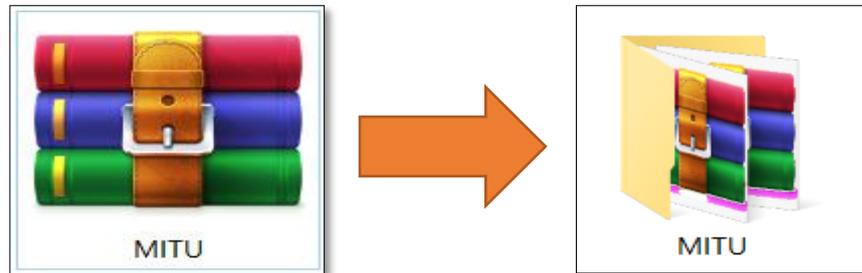


Figura 36: Descompresión de archivo.zip
Fuente: Los investigadores

4.2.3.2 Carga de base de datos

Nos dirigimos a nuestro sitio de administración de base de datos PhpMyAdmin, creamos nuestra base de datos e importamos nuestra base de datos [phplogin.sql](#) que viene en la carpeta extraída en C:\xampp\htdocs\MITU.



Figura 37: Importación de base de datos en PhpMyAdmin
Fuente: Los investigadores

4.2.3.3 Software funcionando

Una vez cargada la base de datos, podemos abrir la url <http://localhost/MITU/> de nuestro proyecto en nuestro navegador. Donde obtendremos la página de inicio como respuesta.



Figura 38: Página de inicio del software en producción
Fuente: Los investigadores

4.2.3.4 Inserción de información

Abrimos un modal de un trámite, donde podemos insertar la información solicitada por el mismo.

Ingrese sus datos

Nombres
FAUSTO ALMEIDA

Teléfono
0995701166

Gestionar Turno

Figura 39: Modal de registro de información
Fuente: Los investigadores

4.2.3.5 Mensajería Móvil

Una vez ingresada la información, nos llegara un mensaje al número ingresado con un detalle de lo que solicitamos.



Figura 40: Mensaje móvil recibido.
Fuente: Los investigadores

4.2.3.6 Consulta de Información

Para la consulta información registrada en el sistema debemos ingresar a la siguiente dirección: http://localhost/MITU/turno_vn.php.

Turno	Nombre	Hora Turno	Turno Atendido	Turno No Atendido
TR001	RENE FONSECA	2020-01-03 14:03:53	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TR002	FAUSTO ALMEIDA	2020-01-08 08:57:38	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TR003	JONATHAN ARIAS	2020-01-09 16:30:33	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TR004	LUIS EDISON	2020-01-09 16:30:46	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Mostrando 1 al 4 de 4 registros

< Anterior 1 Siguiente >

Figura 41: Información recibida, desde los modales, en tabla de atención
Fuente: Los investigadores

4.2.3.7 Asignación de turnos

Para asignar turnos, le damos clic en turnos atendidos o no atendidos y observar en los módulos directamente.

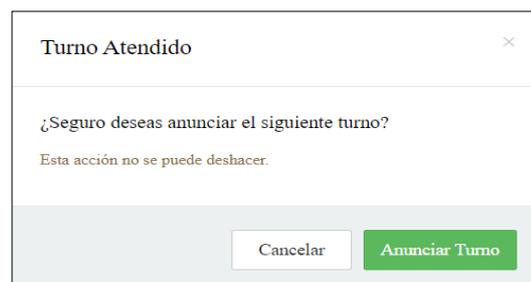


Figura 42: Asignación de turnos
Fuente: Los investigadores

4.2.3.8 Comprobación de turnos.

Para visualizar el estado de los turnos y sus resultados en tiempo real damos clic en el siguiente botón. 



Figura 43: Control de turnos en tiempo real
Fuente: Los investigadores

4.2.3.9 Vista de Módulos

Podemos observar un colero de turnos y la lista de turnos a continuación.



Figura 44: Información de usuario, vista en módulos
Fuente: Los investigadores

4.2.3.10 Registro de historial.

En este apartado podremos ingresar al historial de datos, que se cuenta en la base de datos.



Figura 45: Histórico de información obtenida diariamente
Fuente: Los investigadores

4.2.3.11 Resultados implementación

Con la implementación del software desarrollado el punto de mayor aglomeración de personas, se les indico a los usuarios el uso de la nueva herramienta tecnológica para que accedan a una atención personalizada, facilitando así la atención del personal de ventanilla hacia el usuario al momento de realizar un trámite, causando un ambiente de confort en el usuario al saber en qué puesto de espera se encuentra para su posterior atención, de esta manera los usuarios tuvieron un tiempo de espera, el mismo que fue utilizado para realizar otras actividades, como: completar la documentación necesaria para ser atendidos, llenar formularios, entre otros. Al finalizar la implementación se obtuvo un total de 122 turnos, obteniendo un total de 89 turnos atendidos satisfactoriamente y los 33 restantes no atendidos, pero registrados en el sistema teniendo una gran aceptación el sistema desarrollado por parte de los usuarios atendidos, cumpliendo así con los objetivos planteados inicialmente, y dando solución a la problemática encontrada correctamente.

REPORTE DE TURNOS				
TIPO		TURNOS ATENDIDOS		
MATRICULAS		6		
OTROS		22		
RETIRO TRAMITE		31		
TRAMITES		30		
TOTAL		89		
TIPO		TURNOS NO ATENDIDOS		
MATRICULAS		24		
OTROS		9		
TOTAL		33		
TIPO	TURNOS	SOLICITANTE	FECHA-HORA SOLICITUD	ESTADO
MATRICULAS	MA001	ADRIAN CONTRERAS	2019-12-16 09:00:07	ATENDIDO
TRAMITES	TR001	FABIAN CORTEZ	2019-12-16 09:05:02	ATENDIDO
RETIRO TRAMITE	RT001	JULIAN RODRIGUEZ	2019-12-16 09:08:15	ATENDIDO
OTROS	OT001	ALEX SILVA	2019-12-16 09:08:37	ATENDIDO
MATRICULAS	MA002	JHEAN ALVAREZ	2019-12-16 09:10:00	NO ATENDIDO
TRAMITES	TR002	ELIAN RIVERA	2019-12-16 09:10:05	ATENDIDO
RETIRO TRAMITE	RT002	ALEJANDRO RAMIREZ	2019-12-16 09:10:15	ATENDIDO
OTROS	OT002	ABIGAIL PALOMO	2019-12-16 09:15:37	ATENDIDO
MATRICULAS	MA003	LUCIA LOPEZ	2019-12-16 09:15:52	NO ATENDIDO
TRAMITES	TR003	FABIAN INCA	2019-12-16 09:30:02	ATENDIDO

Figura 46: Reporte de turnos en formato PDF
Fuente: Los investigadores

5. CONCLUSIONES

- El desarrollo del Back End mediante la implementación del modelo padre e hijo con los lenguajes de programación de interacción de código abierto como JavaScript con AJAX, permitieron una favorable obtención de resultados en tiempo real de los procesos que se realizan en el sistema desarrollado.
- Con la combinación entre PHP y JavaScript, se pudo evidenciar un correcto funcionamiento al momento de consumir los datos ingresados mediante la interfaz gráfica, teniendo los mismos resultados en nuestro motor de base de datos MySQL y en los módulos encargados de replicar esa misma información en el sistema desarrollado.
- Mediante la creación de procedimientos almacenados en MySQL, se omitió el paso de crear una relación entre tablas al nivel de base de datos, siendo así el procedimiento el que crea las relaciones internamente, asignando así los datos de una manera correcta y no convencional, gestionando así los datos ingresados de forma eficaz.
- Con el uso de la metodología ASD, se ha desarrollado un sistema que cumple con las expectativas del cliente, implementando en su totalidad tecnología de código abierto (PHP, MySQL, JavaScript, AJAX), que es adaptable a diferentes tipos de plataformas web y móviles.

6. RECOMENDACIONES

Dentro de un proyecto tan ambicioso como fue este, al momento de desarrollar un Software que interactúe con el usuario vía browser, que cuente con estándares de calidad y con principios sólidos de transformación digital, siempre se desea que haya una mejora continua del mismo, por lo tanto, se recomienda:

- Desarrollar un borrador del sistema, para tener en cuenta el alcance que tendrá el sistema una vez finalizado considerando el tipo de herramientas tecnológicas que se utilizará para el desarrollo del sistema, para así cumplir con las necesidades del cliente.
- Para la aplicación del modelo de transferencia padre e hijo desarrollado en JavaScript, se debe considerar que las rutas de intercambio de información serán estáticas, y que para su interacción se debe abrir la ventana hijo desde la ventana padre, para poder tener un perfecto control sobre las acciones que se realicen, si la ventana hijo se abre individualmente, las acciones de la ventana padre no tendrán ninguna afectación sobre la ventana hijo directamente.
- El desarrollo de procedimientos almacenados o rutinas en MySQL, se deben realizar dentro del entorno del motor de base de datos utilizado, para poder controlar así las consultas de inserción, actualización, y eliminación de información, de esta manera se podrá invocar a estos procedimientos almacenados de MySQL desde funciones de consultas realizadas en PHP.
- Para el consumo de datos en tiempo real, es necesario encapsular con los atributos correctos las variables que enviarán la información desde la interfaz gráfica hacia la base de datos (MySQL), para así poder insertar los datos del lado del servidor (PHP), y reflejar la información asincrónicamente al usuario (JavaScript y AJAX), se debe tomar en cuenta de igual manera, que cada variable encapsulada deberá ser igual en el archivo PHP, JavaScript que ejecute la función.

7. Bibliografía

- A, H. (9 de Enero de 2016). *XAMPP / APACHE FRIENDS*. Obtenido de EcuRed: <https://www.ecured.cu/XAMPP>
- Alfonsina, C. (15 de Octubre de 2018). *Lenguajes de Programación de Alto Nivel*. Obtenido de Googl Sites: <https://sites.google.com/site/fundamentosprogramacionc6/1-5-lenguajes-de-programacion/1-5-2-lenguaje-de-programacion-de-alto-nivel>
- Arias, A. (2015). *Aprende a Programar Ajax y jQuery*. Lima: Angel Arias.
- B, G. (18 de Enero de 2019). *JavaScript*. Obtenido de Hostinger: <https://www.hostinger.es/tutoriales/que-es-javascript-introduccion-basica/>
- B, G. (13 de Mayo de 2019). *MYSQL / CARACTERISTICAS*. Obtenido de HOSTINGER: <https://www.hostinger.es/tutoriales/que-es-mysql/>
- C.C, R. (19 de Junio de 1987). *El Metodo Comparativo:Ir más allá de las estrategias cualitativas y cuantitativas*. Berkeley: Universidad de California.
- Capetillo, J. (04 de Enero de 2017). *Ilustracion de Actividades del ciclo de vida del desarrollo adaptable del software [Figura1]*. Obtenido de Ceesa: <https://www.ceesa.com/noticias/5-beneficios-del-software-adaptable/>
- COLABORADORES, R. (15 de Septiembre de 2016). *PHP*. Obtenido de RYTE: <https://es.ryte.com/wiki/PHP>
- Contreras, M. M. (20 de Abril de 2011). *Formular correctamente antecedentes de una investigación*. Obtenido de Educapuntos: <http://educapuntos.blogspot.com/2011/04/antecedentes-de-la-investigacion.html>
- Correa, P. (11 de Noviembre de 2015). *Lenguajes de Programación de Bajo Nivel*. Obtenido de ITM201511: <https://itm201511.webnode.es/archivos-del-sistema/lenguajes/lenguajes-de-bajo-nivel/>
- Gavilanez, G. (18 de Abril de 2014). *Que es Open Source*. Obtenido de Tecnología Facil: <https://tecnologia-facil.com/que-es/que-es-open-source/>
- Gervancio, L. O. (23 de Abril de 2018). *Lenguaje de Programación*. Obtenido de Conogasi: <http://conogasi.org/articulos/lenguaje-de-programacion/>

- Landázuri Almeida, R. A. (04 de Abril de 2011). *Diseño, desarrollo e implementación del sistema de gestión de turnos programados para el Sub-centro de Salud Carapungo*. Obtenido de REPOSITORIO PUCE: <http://repositorio.puce.edu.ec/handle/22000/3392>
- Latorre, D. M. (28 de Marzo de 2018). *Historia de las web, 1.0, 2.0, 3.0 y 4.0*. Obtenido de Universidad Marcelino Champagnat: http://umch.edu.pe/arch/hnomarino/74_Historia%20de%20la%20Web.pdf
- Luis, J. (03 de Julio de 2017). *Tecnología MySQL*. Obtenido de Global Soft: <https://es.globalsoftm.com/tecnologia-mysql/>
- Manzanilla Cristobal, O. J. (16 de Agosto de 2016). *Desarrollo e implementación de un sistema para la generación de turnos de los servicios del cuerpo bomberos de Santo Domingo de los Colorados*. Obtenido de ISSUU: https://issuu.com/pucesd/docs/ia4_2016_01_manzanilla_-_olivo
- Marker, G. (27 de Marzo de 2019). *Internet*. Obtenido de Tecnología Informática: <https://tecnologia-informatica.com/que-es-internet-nacio-como-funciona-internet/>
- Olaya, V. (18 de Febrero de 2016). *BASE DE DATOS*. Obtenido de VOLAYA: https://volaya.github.io/libro-sig/chapters/Bases_datos.html
- Reyes, J. J. (06 de Mayo de 2018). *Que es HTML*. Obtenido de DEVCODE: <https://devcode.la/blog/que-es-html/>

8. ANEXOS

8.1 Manual de usuario

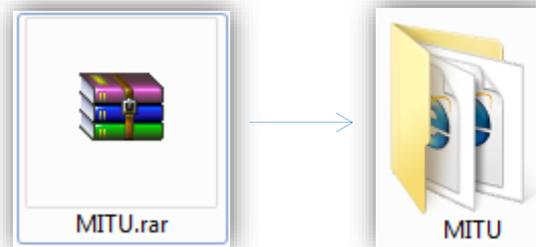
MANUAL DE USUARIO



MITU

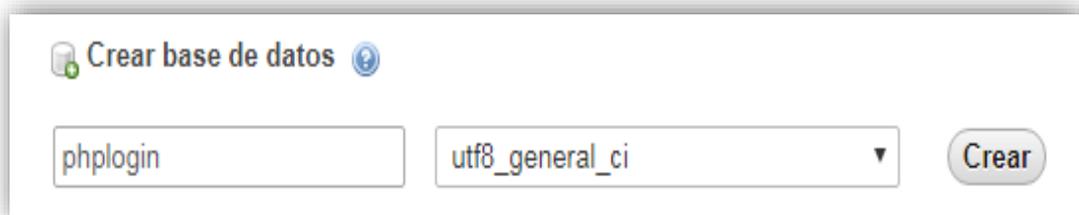
1. Instalación del Sistema

1.1. Dirigirse a la ruta del servidor local XAMPP C:\xampp\htdocs y extraer aquí.

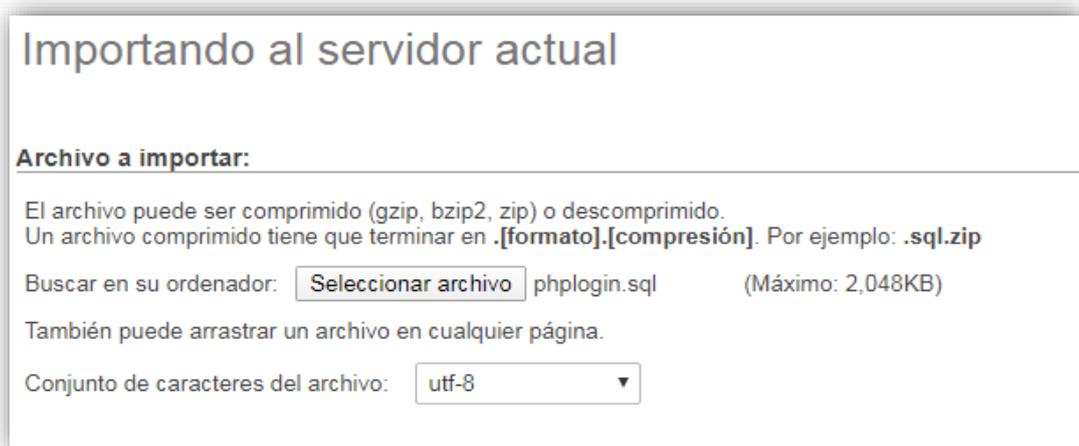


Anexo 1: Descompresión de software
Fuente: Los investigadores

1.2. Para cargar la base de datos ir al sitio de administración de base de datos PhpMyAdmin <http://localhost/phpmyadmin/> e importar la base de datos localizada en el directorio C:\xampp\htdocs\MITU con el nombre phplogin.sql.



Anexo 2: Creación de base de datos desde PhpMyAdmin
Fuente: Los investigadores



Anexo 3: Importación de base de datos en PhpMyAdmin
Fuente: Los investigadores

2. Acceso al Sistema

2.1. Dirigirse a nuestro navegador mediante la siguiente URL: <http://localhost/MITU>. Obtendremos la página de inicio del sistema que contiene.



Anexo 4: Imagen de inicio del software en el servidor local
Fuente: Los investigadores

2.2. En la página de inicio dar clic en trámites, se desplegará el siguiente formulario, llenar los campos con los datos del usuario.

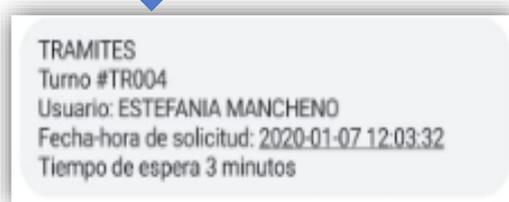
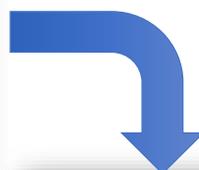
Anexo 5: Ingreso de datos al software
Fuente: Los investigadores

3. Mensajería Móvil

3.1. Después de ingresar la información, dar clic en el botón gestionar turno, mostrará una alerta de confirmación y llegará un mensaje al número de celular registrado con el detalle del turno solicitado.



Anexo 7: Envío de información
Fuente: Los investigadores



Anexo 6: Confirmación de recepción de datos
Fuente: Los investigadores

4. Información matrículas

4.1. En la página de inicio dar clic en el botón para obtener información de Incripción y matrículas.



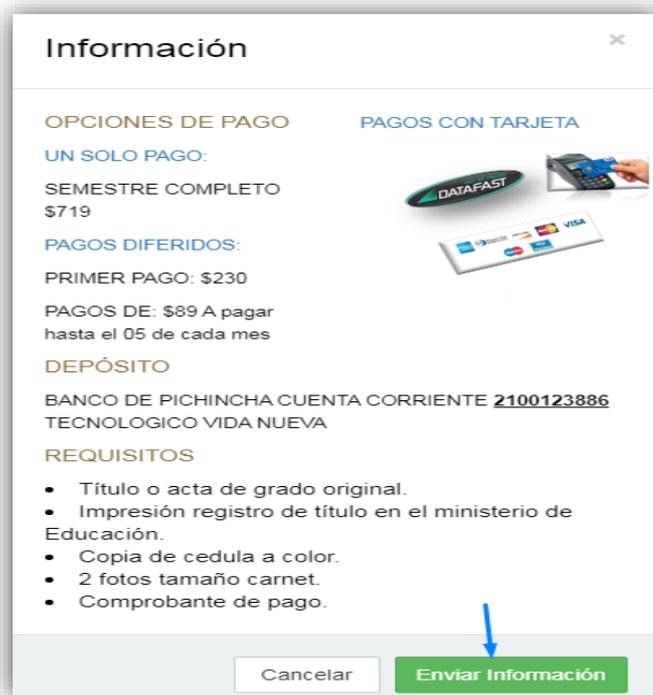
Anexo 8: Acceso a información del instituto
Fuente: Los investigadores

4.2. Se desplegará el siguiente formulario, seleccionar Inscripciones y matrículas para primer nivel.



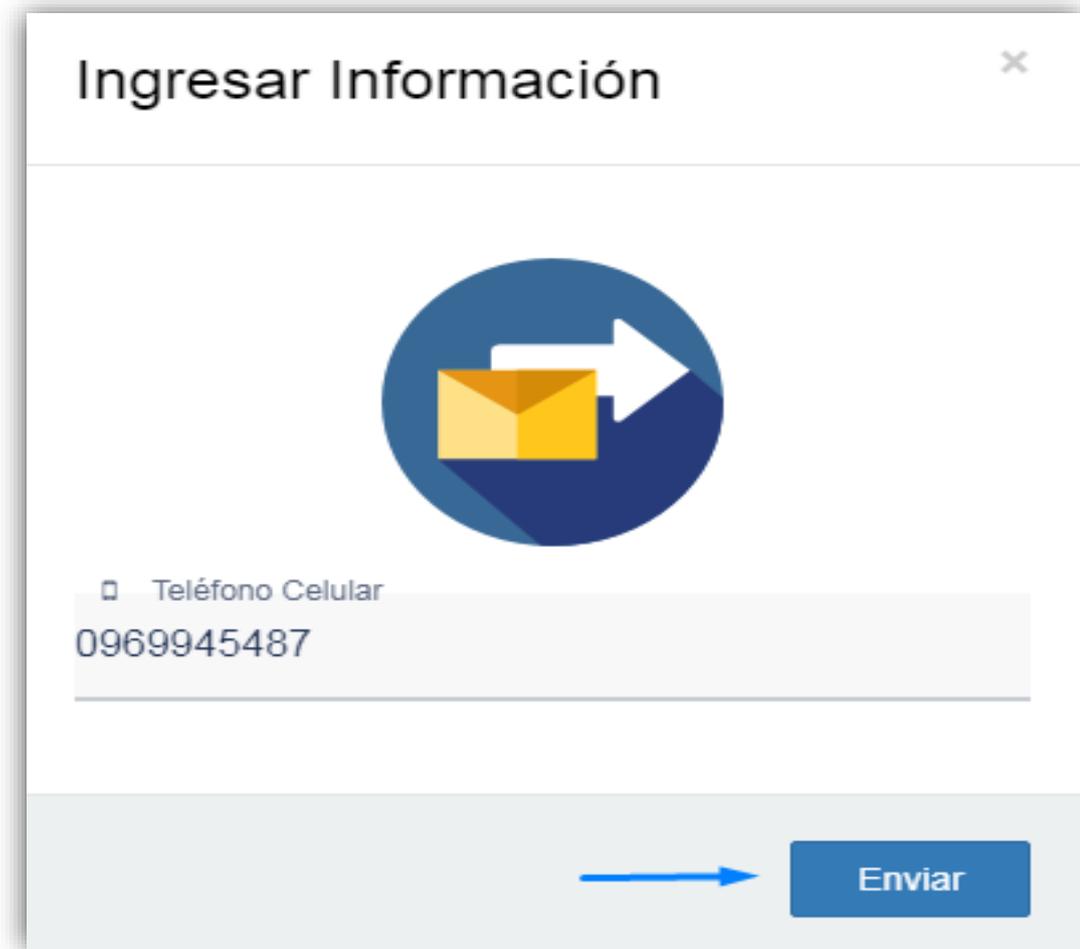
Anexo 9: Modal de información
Fuente: Los investigadores

4.3. Posteriormente en el siguiente formulario mostrará la información solicitada, dar clic en el botón enviar información para completar la acción.



Anexo 10: Modal de información de opciones de pago
Fuente: Los investigadores

4.4. En el formulario que aparece llenar el campo solicitado y al finalizar clic en el botón enviar información.



The image shows a web form titled "Ingresar Información" with a close button (X) in the top right corner. Below the title is a large blue circular icon containing a yellow envelope and a white arrow pointing right. Underneath the icon is a text input field labeled "Teléfono Celular" containing the number "0969945487". At the bottom right of the form is a blue button labeled "Enviar" with a blue arrow pointing towards it.

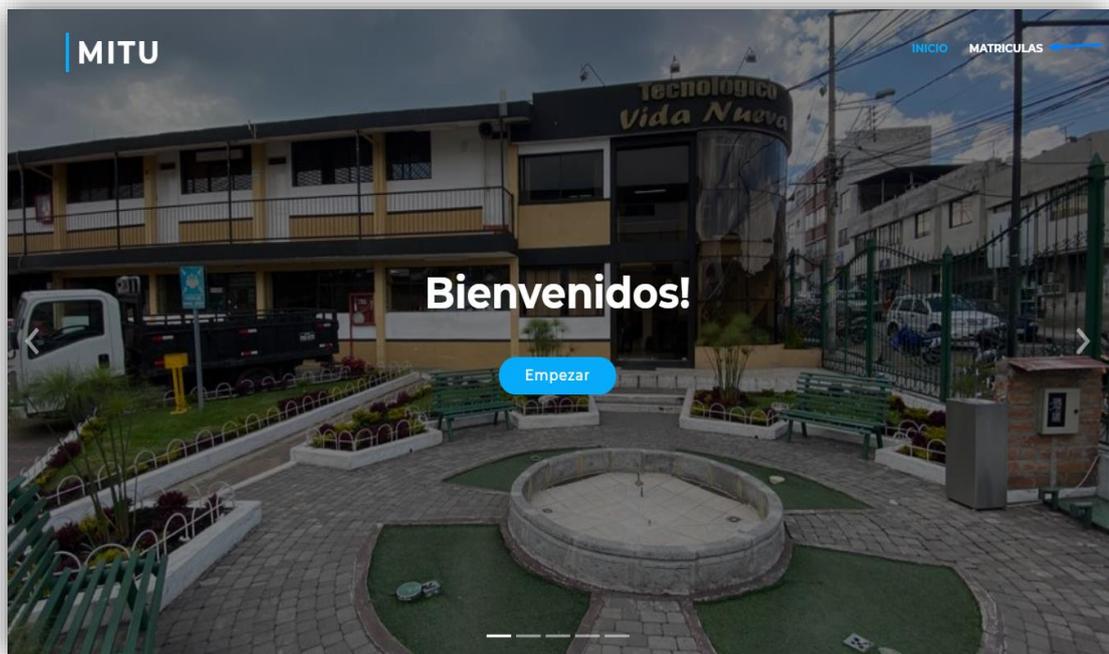
Anexo 11: Solicitud de información
Fuente: Los investigadores

4.5. Una vez enviada la información, nos llegará un mensaje al número registrado detallando lo solicitado, dar clic en el enlace.



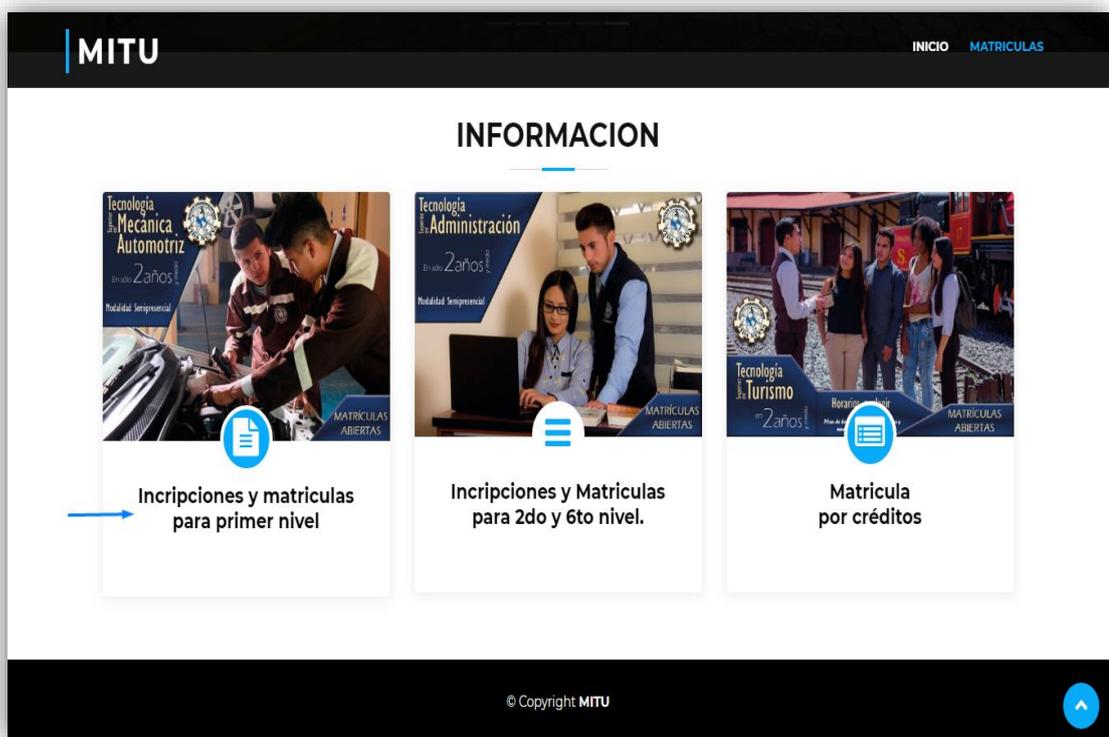
Anexo 12: Mensaje recibido con enlace de página de información
Fuente: Los investigadores

4.6. En nuestro navegador se abrirá la siguiente página web, dar clic en la opción matriculas.



Anexo 13: Página de inicio de información
Fuente: Los investigadores

4.7. Mostrará la siguiente sección de la información solicitada, dar clic en inscripciones y matrículas para primer nivel.



Anexo 14: Página de información general
Fuente: Los investigadores

4.8. Después se desplegará el siguiente formulario con la información correspondiente.

Información
✕

OPCIONES DE PAGO

UN SOLO PAGO:
SEMESTRE COMPLETO \$719

PAGOS DIFERIDOS:
PRIMER PAGO: \$230.
PAGOS DE: \$89 A pagar hasta el 05 de cada mes.

PAGOS CON TARJETA



DEPÓSITO

BANCO DE PICHINCHA
CUENTA CORRIENTE
2100123886 TECNOLÓGICO
VIDA NUEVA

REQUISITOS

- Título o acta de grado original.
- Impresión registro de título en el ministerio de Educación.
- Copia de cedula a color.
- 2 fotos tamaño carnet.
- Comprobante de pago.

Cancelar

Anexo 15: Modales de información de pago
Fuente: Los investigadores

5. Gestión de turnos

5.1. En el navegador ingresar la siguiente URL: http://localhost/MITU/turno_vn.php para realizar la consulta de la información ingresada.

Turnos Registrados

🗨
📄
🗑
☰
⌛

Turno	Nombre	Hora Turno	Turno Atendido	Turno No Atendido
TR000	ISTVN	2020-01-15 11:13:43	🟢	🔴
TR001	ESTEFANIA MANCHENO	2020-01-15 11:14:07	🟢	🔴
TR002	FERNANDO PINCAY	2020-01-15 11:14:33	🟢	🔴
TR003	ABIGAIL PALOMO	2020-01-15 11:14:47	🟢	🔴
TR004	TATHYANA GANCINO	2020-01-15 11:15:00	🟢	🔴

Mostrando 1 al 5 de 5 registros

◀
1
▶

Anexo 16: Página de gestión de turnos
Fuente: Los investigadores

5.2. En la página de gestión de turno dar clic en botón  de turno atendido.

Turno	Nombre	Hora Turno	Turno Atendido	Turno No Atendido
TR000	ISTVN	2020-01-15 11:13:43		
TR001	ESTEFANIA MANCHENO	2020-01-15 11:14:07		
TR002	FERNANDO PINCAY	2020-01-15 11:14:33		
TR003	ABIGAIL PALOMO	2020-01-15 11:14:47		
TR004	TATHYANA GANCINO	2020-01-15 11:15:00		

Mostrando 1 al 5 de 5 registros « Anterior 1 Siguiente »

Anexo 17: Acciones para la gestión de turnos
Fuente: Los investigadores

5.3. Al desplegarse el siguiente formulario dar clic en anunciar turno y mostrará un mensaje de confirmación.

Turno Atendido ✕

¿Seguro deseas anunciar el siguiente turno?
Esta acción no se puede deshacer.

Cancelar
Anunciar Turno

Anexo 18: Confirmación de turnos atendidos
Fuente: Los investigadores

¡Bien hecho! Siguiente turno activado. ✕

Turno	Nombre	Hora Turno	Turno Atendido	Turno No Atendido
TR001	ESTEFANIA MANCHENO	2020-01-15 11:14:07		
TR002	FERNANDO PINCAY	2020-01-15 11:14:33		
TR003	ABIGAIL PALOMO	2020-01-15 11:14:47		
TR004	TATHYANA GANCINO	2020-01-15 11:15:00		

Mostrando 1 al 4 de 4 registros « Anterior 1 Siguiente »

Anexo 19: Atención de turnos, desde interfaz gráfica
Fuente: Los investigadores

6. Vista Módulos

6.1. En la página de gestión de turnos dar clic en  para abrir el turno.



MÓDULO	TURNO	SOLICITANTE
R001	TR001	ESTEFANIA MANCHENO

Anexo 20: Módulos de turnos
Fuente: Los investigadores

6.2. Para realizar la actualización del turno nos dirigimos a la página de gestión de turnos dar clic en el botón 



Anexo 21: Acciones para controlar turno
Fuente: Los investigadores



MÓDULO	TURNO	SOLICITANTE
R001	TR001	ESTEFANIA MANCHENO
R001	M000	ISTVN
R001	RT000	ISTVN

BIENVENIDOS PRÓXIMOS

Copyright © MITU Desarrollado: Fausto Almeida - Estefania Mancheno Tutor: Ing. Darwin Arias

Anexo 22: Módulos con información asignada
Fuente: Los investigadores

6.3. Para cerrar el turnero en la pagina de gestión de turnos dar clic en el botón .



Anexo 23: Acción de cerrar el turnero
Fuente: Los investigadores

7. Historial de Turnos

7.1. Para el historial de turnos dirigirse a la página de gestión de turnos dar clic en botón  se abrirá la siguiente página.



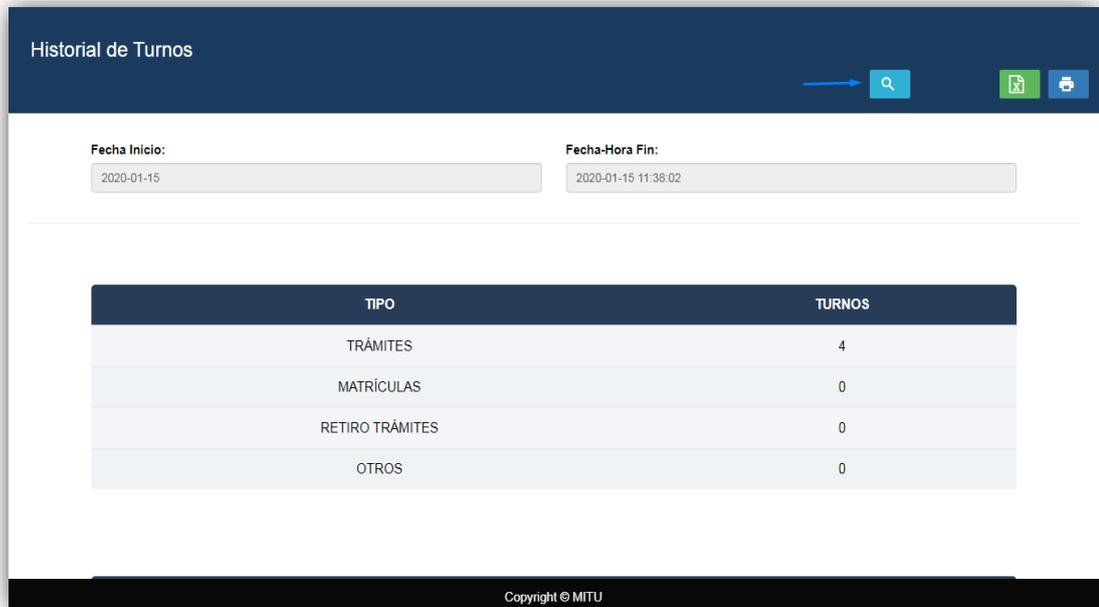
Anexo 24: Botón de historial de información
Fuente: Los investigadores

TIPO	TURNOS
TRÁMITES	
MATRÍCULAS	
RETIRO TRÁMITES	
OTROS	

Copyright © MITU

Anexo 25: Consulta de información desde el historial
Fuente: Los investigadores

7.2. Dar clic en el botón  mostrará los datos obtenidos entre la fecha Inicio y fecha-hora fin del día solicitado.



Historial de Turnos

Fecha Inicio: 2020-01-15

Fecha-Hora Fin: 2020-01-15 11:38:02

TIPO	TORNOS
TRÁMITES	4
MATRÍCULAS	0
RETIRO TRÁMITES	0
OTROS	0

Copyright © MITU

Anexo 26: Botón de búsqueda de información
Fuente: Los investigadores

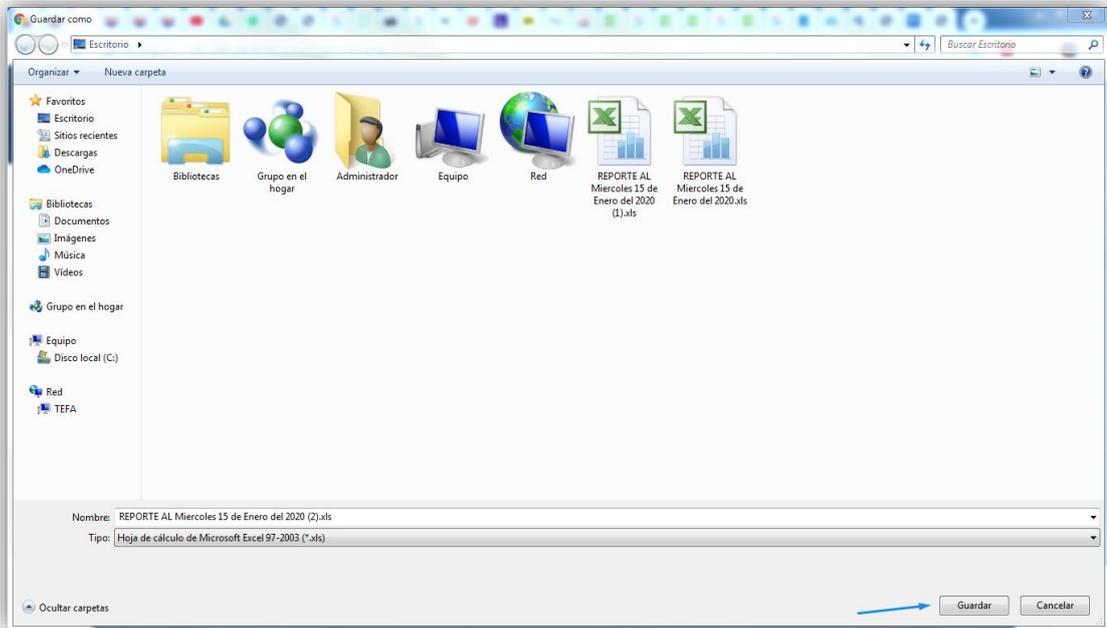
TIPO	TURNO	NOMBRE	FECHA	ESTADO
TRAMITES	TR001	ESTEFANIA MANCHENO	2020-01-15 11:14:07	NO ATENDIDO
TRAMITES	TR002	FERNANDO PINCAY	2020-01-15 11:14:33	NO ATENDIDO
TRAMITES	TR003	ABIGAIL PALOMO	2020-01-15 11:14:47	NO ATENDIDO
TRAMITES	TR004	TATHYANA GANCINO	2020-01-15 11:15:00	NO ATENDIDO

Anexo 27: Obtención de resultados encontrados
Fuente: Los investigadores

7.3. En la página del historial de turnos dar clic en el botón  para exportar el reporte en un documento Excel, seleccionar la ubicación donde desea guardar el documento y clic en guardar.



Anexo 28: Botón de exportación de resultados en Excel
Fuente: Los investigadores



Anexo 29: Alojamiento de archivo Excel exportado
Fuente: Los investigadores

7.4. Dirigirse a la ubicación del reporte abrir el documento Excel mostrará el siguiente formato con los turnos atendidos y no atendidos en el día.

A	B	C	D	E
REPORTE DE TURNOS Miercoles 15 de Enero del 2019				
TIPO	TURNOS	SOLICITANTE	HORA DE SOLICITUD	ESTADO
TRAMITES	TRO01	ESTEFANIA MANCHENO	2020-01-15 11:14:07	NO ATENDIDO
TRAMITES	TRO02	FERNANDO PINCAY	2020-01-15 11:14:33	NO ATENDIDO
TRAMITES	TRO03	ABIGAIL PALOMO	2020-01-15 11:14:47	NO ATENDIDO
TRAMITES	TRO04	TATHYANA GANCINO	2020-01-15 11:15:00	NO ATENDIDO

Anexo 30: Resultado obtenidos del software en formato Excel
Fuente: Los investigadores

7.5. Dar clic en el botón  para imprimir el reporte del historial de turnos se descargará automáticamente y se abrirá en el editor de PDF o navegador, mostrando los turnos atendidos y no atendidos en el día.



Anexo 31: Impresión o exportación de resultados en formato PDF
Fuente: Los investigadores

REPORTE DE TURNOS

TIPO	TURNOS ATENDIDOS
TOTAL	0

TIPO	TURNOS NO ATENDIDOS
TRAMITES	4
TOTAL	4

TIPO	TURNO	SOLICITANTE	FECHA-HORA SOLICITUD	ESTADO
TRAMITES	TR001	ESTEFANIA MANCHENO	2025-01-15 11:14:07	NO ATENDIDO
TRAMITES	TR002	FERNANDO PINCAY	2025-01-15 11:14:33	NO ATENDIDO
TRAMITES	TR003	ABIGAIL PALOMO	2025-01-15 11:14:47	NO ATENDIDO
TRAMITES	TR004	TATHYANA GANCINO	2025-01-15 11:15:00	NO ATENDIDO

MITU
MITU@hotmail.com
+(593)0969945487

Página 1/1

Anexo 32: Archivo de resultados en formato PDF
Fuente: Los investigadores