

INSTITUTO SUPERIOR TECNOLÓGICO

“SAN GABRIEL”



ESPECIALIDAD INFORMÁTICA

PROYECTO DE TITULACIÓN

PREVIO A LA OBTENCIÓN DEL TÍTULO DE:

TECNÓLOGO EN INFORMÁTICA MENCIÓN ANÁLISIS EN SISTEMAS

TEMA

DISEÑO E IMPLEMENTACIÓN DE UN SITIO WEB Y MÓDULO DE CONTROL PARA LA MOVILIZACIÓN DEL PERSONAL OPERATIVO Y VEHÍCULO DEL CUERPO DE BOMBEROS DEL CANTÓN GUANO DESARROLLADO EN PHP ORIENTADO A OBJETOS Y GESTOR DE BASE DE DATOS MYSQL EN EL

PERÍODO 2018

AUTOR:

VICTOR MARCELO OROZCO FREIRE

TUTOR:

ING. WILLIAM ADRIANO

RIOBAMBA-ECUADOR

2018

CERTIFICACIÓN

Certifico que el Sr. Victor Marcelo Orozco Freire, con el N° de Cédula 060326146-2 ha elaborado bajo mi Asesoría el Trabajo de Investigación titulado:

DISEÑO E IMPLEMENTACIÓN DE UN SITIO WEB Y MÓDULO DE CONTROL PARA LA MOVILIZACIÓN DEL PERSONAL OPERATIVO Y VEHÍCULO DEL CUERPO DE BOMBEROS DEL CANTÓN GUANO DESARROLLADO EN PHP ORIENTADO A OBJETOS Y GESTOR DE BASE DE DATOS MYSQL EN EL PERÍODO 2018

Por tanto, autorizo la presentación para la calificación respectiva.

Ing. William Adriano

TUTOR DE TESIS

“El presente Trabajo de Investigación constituye un requisito previo para la obtención del Título de **Tecnólogo en Informática mención Análisis de Sistema**”



“Yo, Victor Marcelo Orozco Freire con N° de Cédula 060326146-2, declaro que la investigación es absolutamente original, autentica, personal y los resultados y conclusiones a los que se han llegado es de mi absoluta responsabilidad.”

Marcelo Orozco

INSTITUTO TECNOLÓGICO SUPERIOR

“SAN GABRIEL”

ESPECIALIDAD INFORMÁTICA

TRABAJO DE INVESTIGACIÓN

PREVIA A LA OBTENCION DEL TÍTULO DE:

TECNÓLOGO EN INFORMÁTICA MENCION ANÁLISIS DE SISTEMAS

TÍTULO:

DISEÑO E IMPLEMENTACIÓN DE UN SITIO WEB Y MÓDULO DE CONTROL PARA LA MOVILIZACIÓN DEL PERSONAL OPERATIVO Y VEHÍCULO DEL CUERPO DE BOMBEROS DEL CANTÓN GUANO DESARROLLADO EN PHP ORIENTADO A OBJETOS Y GESTOR DE BASE DE DATOS MYSQL EN EL PERÍODO 2018

APROBADO

ASESOR DE TESIS DE GRADO

PRESIDENTE DEL TRIBUNAL

MIEMBRO DEL TRIBUNAL

MIEMBRO DEL TRIBUNAL

FIRMAS DE RESPONSABILIDAD

NOMBRES	FECHA	FIRMA
RECTORA
VICERRECTOR ACADÉMICO
DIRECTOR DE TESIS
MIEMBRO DEL TRIBUNAL
MIEMBRO DEL TRIBUNAL
MIEMBRO DEL TRIBUNAL
NOTA FINAL DE LA TESIS	

DEDICATORIA

Dedico este trabajo a Dios y toda mi familia por todo el apoyo brindado en toda mi trayectoria estudiantil.

Victor Marcelo

AGRADECIMIENTO

Un agradecimiento a Dios, mi familia y los profesores del Instituto Tecnológico Superior San Gabriel quien con su paciencia y conocimiento supieron guiarme de la mejor manera para la culminación de mi carrera estudiantil.

Victor Marcelo

ABREVIATURAS

POO: Programación Orientada a Objetos

PHP: Hypertext Pre-Processor

AT&T: American Telephone and Telegraph

WWW: World Wide Web

HTML: World Wide Web

WEB: “red”, “telaraña” o “malla”

CSS: Cascading Style Sheets

URL: Localizador Uniforme de Recursos

DOM: Modelo de objeto de documento

HIJACKING: Secuestro

AJAX: Asynchronous Javascript and XML

GLOSARIO DE TÉRMINOS

e-commerce: comercio electrónico es un método de compraventa de bienes, productos o servicios valiéndose de internet como medio, es decir, comerciar de manera online

Small talk: es un lenguaje reflexivo de programación, orientado a objetos y con tipado dinámico.

GNU: es un sistema operativo de tipo Unix, lo cual significa que se trata de una colección de muchos programas: aplicaciones, bibliotecas, herramientas de desarrollo y hasta juegos

COOKIES: Las **cookies** son pequeños archivos que algunos sitios web guardan en tu ordenador.

Online: es una palabra inglesa que significa “en línea”.

Sniffing: es una aplicación especial para redes informáticas, que permite como tal capturar los paquetes que viajan por una red.

UNIX: es un sistema operativo, es decir, es una colección de programas que ejecutan otros programas en una computadora

C y C++: lenguaje de programación diseñado en 1979 por Bjarne Stroustrup. La intención de su creación fue extender al lenguaje de programación C mecanismos que permiten la manipulación de objetos

OPEN SOURCE: En general, el código abierto se refiere a cualquier programa cuyo código fuente se pone a disposición para su uso o modificación

Renderizar: es un término usado en para referirse al proceso de generar una imagen desde un modelo.

TABLA DE CONTENIDO

CERTIFICACIÓN.....	II
FIRMAS DE RESPONSABILIDAD	V
DEDICATORIA.....	VI
AGRADECIMIENTO	VII
ABREVIATURAS	VIII
GLOSARIO DE TÉRMINOS	IX
ÍNDICE DE TABLAS.....	XV
ÍNDICE DE GRÁFICOS	XVI
INTRODUCCIÓN.....	XVII
RESUMEN	XIX
SUMMARY	XX
CAPÍTULO I.....	1
1 MARCO REFERENCIAL	1
1.1 FORMULACIÓN GENERAL DEL PROYECTO DE TESIS	2
1.1.1 ANTECEDENTES DEL PROBLEMA DE INVESTIGACIÓN	2
1.1.2 DATOS INFORMATIVOS	3
1.2 DEFINICIÓN DEL PROBLEMA.....	3
1.2.1 JUSTIFICACIÓN.....	3
1.2.2 OBJETIVOS.....	4
1.2.2.1 OBJETIVO GENERAL	4

1.2.2.2	OBJETIVOS ESPECÍFICOS	4
CAPÍTULO II.....		5
2	MARCO TEORICO	5
2.1	ORIENTACION A OBJETOS.....	6
2.1.1	HISTORIA EVOLUTIVA	6
2.1.2	VENTAJAS SOBRE LA PROGRAMACION ESTRUCTURADA	7
2.1.3	CARACTERISTICA de la POO	9
2.1.4	ELEMENTOS DE PROGRAMACION	9
2.1.5	Sitio Web.....	10
2.1.5.1	Clasificación de Sitios Web	10
2.1.5.2	Tipos de Sitios Web	11
2.2	PHP.....	14
2.2.1	DESCRIPCION DE PHP	14
2.2.2	ELEMENTOS DE PHP.....	15
2.3	PROGRAMACION PHP POO	15
2.3.1	VENTAJAS DE la POO.....	15
2.3.2	Variables.....	16
2.3.3	Inclusión de Código PHP POO en una Página HTML	17
2.3.4	Sentencias de control.....	17
2.3.5	ventajas de PHP.....	18
2.3.6	seguridad de PHP	19
2.3.6.1	Protección contra la inyección de código SQL	19

2.3.7	Uso de variables de sesión y no de cookies.....	20
2.3.8	Suplantación de sesiones	20
2.3.9	Parámetros GET/POST explotables	21
2.4	MYSQL	22
2.4.1	QUE ES UN GESTOR DE BASE DE DATOS.....	22
2.4.2	DEFINICION DE MYSQL.....	22
2.4.3	POR QUÉ UTILIZAR MYSQL.....	22
2.4.4	TIPOS DE DATOS	23
2.4.4.1	Tipos de dato numéricos.....	23
2.4.4.2	Tipos de dato con formato fecha	24
2.4.4.3	Tipos de dato texto	25
2.5	compatibilidad entre mysql y php	27
2.6	JAVASCRIPT	29
2.6.1	DEFINICION E HISTORIA	29
2.6.2	DIFERENCIAS con otros lenguajes	29
2.6.3	Elementos para JAVASCRIPT	30
2.7	Librerías JS.....	31
2.7.1	ventajas y desventajas	32
2.7.1.1	ventajas	32
2.7.1.2	desventajas.....	33
2.8	CSS3.....	34
2.8.1	INTRODUCCION AL CSS3	34

2.8.2	Evolución de CSS3.....	34
2.8.3	NOVEDADES DE CSS3	35
CAPÍTULO III		38
3	ANÁLISIS Y DISEÑO	38
3.1	RECOLECCIÓN DE LA INFORMACIÓN	39
3.1.1	Alcance.....	39
3.1.2	ANÁLISIS.....	40
3.1.3	Estudio de factibilidad.....	40
3.1.3.1	Factibilidad operativa.....	40
3.1.3.2	Factibilidad técnica.....	41
3.1.3.3	Factibilidad económica.....	41
3.1.3.4	Factibilidad legal	41
3.1.4	Análisis de los requerimientos	42
3.1.4.1	Requerimientos funcionales del sistema	42
3.1.4.2	Requerimientos no funcionales del sistema	42
3.1.5	Definición de Casos de Uso	43
3.2	DISEÑO	47
3.2.1	Modelo relacional.....	47
3.3	Diccionario de datos.....	48
3.4	Diseño de interfaces	52
CAPÍTULO IV		54
4	IMPLEMENTACIÓN DEL SITIO WEB	54

4.1	CONFIGURACIÓN DE LAS HERRAMIENTAS DE DESARROLLO.....	55
4.2	ARQUITECTURA DEL SISTEMA	57
4.3	IMPLEMENTACIÓN DEL SISTEMA	57
4.3.1	Definición de Módulos.....	57
4.3.2	Desarrollo de la aplicación implementación	58
4.3.3	Implementación de la aplicación.....	58
4.4	PRUEBAS DE APLICACIÓN	58
4.5	CAPACITACIÓN AL PERSONAL	59
4.6	MANTENIMIENTO DE LA APLICACIÓN	60
	CONCLUSIONES.....	61
	RECOMENDACIONES	62
	Bibliografía.....	63
	ANEXOS	66
	ANEXO 1. MANUAL DE USUARIO	67

ÍNDICE DE TABLAS

Tabla N° 1 Ventajas de la POO sobre la Programación estructurada	8
Tabla N° 2 Tipos de Datos Numéricos	24
Tabla N° 3 Tipos de Datos Fecha	25
Tabla N° 4 Tipos de datos tipo texto	26
Tabla N° 5 Cuadro comparativo	27
Tabla N° 6: Cuadro comparativo	28
Tabla N° 7 Tabla comparativa	28
Tabla N° 8. Evolución CSS3	35
Tabla N° 9. DD Detalle	48
Tabla N° 10 DD Implementos	48
Tabla N° 11 DD Lista de victimas	48
Tabla N° 12 DD Lugar	49
Tabla N° 13 DD Parte	49
Tabla N° 14 DD Parte Vehículos	50
Tabla N° 15 DD Personal	50
Tabla N° 16 DD Solicitud	50
Tabla N° 17 DD Tipo de Emergencia	51
Tabla N° 18 DD Usuario	51
Tabla N° 19 DD Vehículo	51
Tabla N° 20 Pruebas de Aplicación	59

ÍNDICE DE GRÁFICOS

Gráfico N° 1 Historia de POO	7
Gráfico N° 2 Sitio Web Institucional	11
Gráfico N° 3 OnePage	11
Gráfico N° 4 Landing Page	11
Gráfico N° 5 Blog.....	12
Gráfico N° 6 Stio e-commerce	13
Gráfico N° 7 LOGO PHP	14
Gráfico N° 9 LOGO MySQL	22
Gráfico N° 10 CU Ingres al Sistema	43
Gráfico N° 11 CU Nuevo Usuario.....	44
Gráfico N° 12 CU Nuevo Parte	45
Gráfico N° 13 CU Solicitud Parte	45
Gráfico N° 14 CU Vehículos.....	46
Gráfico N° 15 Modelo Relacional	47
Gráfico N° 16 Página Principal	52
Gráfico N° 17 Acceso al sistema.....	52
Gráfico N° 18: Pantallas Secundarias.....	53
Gráfico N° 19. Setup XAMPP.....	55
Gráfico N° 20 Configuración para el XAMPP	55
Gráfico N° 21 Finalización de Setup.....	56
Gráfico N° 22 Inicialización del XAMPP	56
Gráfico N° 23 Arquitectura del Sistema.....	57
Gráfico N° 24 Módulos del Sistema.....	58

INTRODUCCIÓN

El Cuerpo de Bomberos del cantón Guano, tiene como finalidad prestar ayuda en diferentes emergencias a los habitantes del cantón, pudiendo ser estas: médicas, de incendios o de otra índole quienes acuden inmediatamente ante las llamadas despachadas por la consola del ECU-911 o/a llamadas privadas de la ciudadanía, en donde el personal operativo al retornar a la compañía de bomberos entrega un reporte de la salida del vehículo destinado para atender dicha emergencia, acción que al ser llevada de forma manual en hojas simples, ha provocado que varios de estos reportes se pierdan por los cambios de turno u otras circunstancias y cuando se requiere de dicha información solicitada principalmente por la fiscalía, seguros u otras autoridades competentes, y al no ser encontradas han causado varios contratiempos.

Ante este requerimiento se propone en el Cuerpo de Bomberos del cantón Guano la utilización de la tecnología para organizar la información y obtener reportes automatizados, que permitan llevar un control y registro cronológico de los vehículos e informes del personal que atendió la emergencia, mismos que deben contener lugar, fecha, hora y descripción detallada de la emergencia atendida, información que será consolidada en una base de datos y posteriormente impresa en formatos PDF para su archivo respectivo, lo que permitirá una mejor organización del trabajo de quienes laboran en esta entidad estatal.

Como miembro de la institución bomberil al evidenciar la falta de un sitio web en el Cuerpo de Bomberos del cantón Guano que agilite procesos, y al estar cursando la Carrera de Informática Mención Análisis de Sistemas en el Instituto Superior Tecnológico “San Gabriel”, se ha creído oportuno aplicar los conocimientos adquiridos durante este proceso de formación y contribuir por medio de este trabajo de investigación al desarrollo de la institución de la que se forma parte.

Para alcanzar lo propuesto, se plantea como objetivo general: Diseñar e implementar un sitio web y módulo de control para la movilización del personal operativo y vehículos del Cuerpo de Bomberos del cantón Guano.

Finalmente se indica que el trabajo de investigación se desarrolla en 5 capítulos que se describen a continuación:

El capítulo primero, contiene el marco referencial, donde se describen los antecedentes y la información referente al Cuerpo de Bomberos del cantón Guano, así como también se determina el planteamiento del problema, la justificación y los objetivos del trabajo de investigación, aspectos que facilitaron la comprensión de las necesidades de la institución y el direccionamiento de la investigación.

En el capítulo segundo, se desarrolla el marco teórico que contiene conceptos esenciales sobre el lenguaje de programación PHP y base de datos MySQL, herramientas que facilitan el almacenamiento de los datos que se manejan en el Cuerpo de Bomberos del cantón Guano y a su vez la creación de la interfaz, tablas, y demás formularios del sistema web.

En el capítulo tercero, se desarrolla el análisis y diseño del sitio web, así como también el módulo de control para la movilización del personal operativo y vehículos del cuerpo de Bomberos, en donde se describe como se recolectaron los datos para su realización, factibilidad, casos de uso, requerimientos de funcionalidad y la delimitación de la interfaz, llegando a diseñar un sitio web y un módulo de control del sistema sencillo y de fácil manejo para el usuario.

El capítulo cuarto, contiene el proceso de implementación del sitio web y del módulo de control, donde se determinan aspectos como la configuración de las herramientas de desarrollo, la arquitectura del sistema, la definición de los módulos y el desarrollo e implementación propia del sistema, junto con las pruebas de aplicación e interrelación entre el usuario y el administrador.

En el capítulo quinto, se establecen las conclusiones y recomendaciones de la investigación de acuerdo a los objetivos propuestos que han sido alcanzados, acompañado de dos anexos: el manual de usuario y el manual técnico

RESUMEN

La investigación denominada, DISEÑO E IMPLEMENTACIÓN DE UN SITIO WEB Y MÓDULO DE CONTROL PARA LA MOVILIZACIÓN DEL PERSONAL OPERATIVO Y DE VEHÍCULOS DESARROLLADO EN PHP ORIENTADO A OBJETOS Y GESTOR DE BASE DE DATOS MYSQL se propone para el cuerpo de bomberos del cantón Guano durante el periodo 2018. El proceso metodológico seguido en esta investigación, partió del análisis de las necesidades de la institución, para lo cual, se realizaron entrevistas al representante legal de la institución y sobre la base de sus resultados se propone el diseño e implementación del sitio web y del módulo de control con las especificaciones solicitadas por el usuario. El desarrollo del sitio web se efectuó con la ayuda de la programación orientada a objetos PHP, mientras que en el diseño se utilizó el lenguaje de etiquetado HTML 5 y hojas de estilo CSS con lo que se definió colores, tipografía entre otros. Para el almacenamiento de la información se utilizó el gestor de base de datos MYSQL. Una vez desarrollada la aplicación se realizó las pruebas respectivas de funcionalidad quedando en óptimas condiciones. Los resultados obtenidos con la implementación del sitio web fueron: dar a conocer los antecedentes e historia de la institución, cómo usar un extintor en caso de emergencia, y la publicación de fotografías de acontecimientos importantes con las que se promociona los servicios que brinda la institución, mientras que con la implementación del módulo de control para la movilización del personal operativo y de los vehículos se obtienen reportes tanto de la movilización del personal, de los vehículos como también de la atención de las diversas emergencias, los cuales luego de ser almacenados en la base de datos, pueden ser impresos en PDF para su respectivo archivo.

SUMMARY

The investigation called, DESIGN AND IMPLEMENTATION OF A WEB SITE AND CONTROL MODULE FOR THE MOBILIZATION OF OPERATIONAL AND VEHICLE PERSONNEL DEVELOPED IN OBJECT-ORIENTED PHP AND DATABASE MANAGER MYSQL is proposed for the Guano fire department during the period 2018. The methodological process followed in this investigation, started from the analysis of the needs of the institution, for which, interviews were conducted with the legal representative of the institution and based on its results, the design and implementation of the website is proposed and of the control module with the specifications requested by the user. The development of the website was carried out with the help of PHP object-oriented programming, while the design used the HTML 5 tagging language and CSS style sheets, which defined colors, typography, among others. The MYSQL database manager was used to store the information. Once the application was developed, the respective functionality tests were carried out, remaining in optimal conditions. The results obtained with the implementation of the website were: to publicize the institution's background and history, how to use a fire extinguisher in case of emergency, and the publication of photographs of important events with which the services offered by the institution are promoted. , while with the implementation of the control module for the mobilization of operating personnel and vehicles, reports are obtained both of the mobilization of personnel, of vehicles and also of the attention to various emergencies, which after being stored in the database can be printed in PDF for their respective file.

CAPÍTULO I

1 MARCO REFERENCIAL

1.1 FORMULACIÓN GENERAL DEL PROYECTO DE TESIS

1.1.1 ANTECEDENTES DEL PROBLEMA DE INVESTIGACIÓN

En el año 2000, el 2 de enero, cuando se encontraba como alcalde el Lic. Oswaldo Estrada, se toma la decisión de la creación de un cuerpo de bomberos para el cantón Guano, a cargo del Teniente Coronel Marco y 3 bomberos los Señores Emilio Estrada, Patricio Castillo y Armando Muñoz.

Dotado de un taquero de marca Ford para el combate de incendios, el cuerpo de bomberos en sus inicios no contaba con los recursos básicos, para su labor. En el año 2007 se hace el primer llamado a la ciudadanía para que forme parte de la institución.

En el año 2013 se construye, una segunda compañía en San Andrés conocida como X2, con el propósito de disminuir tiempos y distancias entre emergencias. La institución realiza recates, atiende emergencias y apaga incendios. Con la misión se siempre estar alertas cuando requiera la población.

Con el crecimiento de la institución surge la necesidad de manejar de mejor manera los recursos existentes por tal motivo, la investigación tiene como la finalidad crear un Sitio web que permita un mejor control del patrimonio y servicios del cuerpo de bomberos mediante el uso del lenguaje de Programación PHP Orientado a Objetos y un gestor de BD MySQL entre otros componentes que determine la entrega de un sitio en óptimas condiciones

1.1.2 DATOS INFORMATIVOS

Misión

Salvar vidas y proteger bienes inmuebles con acciones oportunas y eficientes, en la lucha contra el fuego, rescate, salvamento y prevención, de emergencias.

1.2 DEFINICIÓN DEL PROBLEMA

1.2.1 JUSTIFICACIÓN

Con la realización del proyecto se pretende automatizar procesos, que en la actualidad se realizan de forma manual, siendo estos obsoletos y susceptibles a daños físicos. Esto conlleva a la pérdida de información importante y que genera problemas en la entrega de informes.

En la actualidad con el avance tecnológico permite al ser humano, apoyarse en estas formas de tecnología para el control de la información en forma segura, y confiable reduciendo así equivocaciones, y tiempos de respuesta en su vida cotidiana.

Gracias a esto, se generan o crean sistemas informáticos basados en WWW, siendo estos accesibles en cualquier parte, y en cualquier dispositivo.

Las necesidades del Cuerpo de Bomberos Guano es llevar un control mejorado de los reportes de las emergencias para lo cual se apoyará en las tecnologías actuales, que les permita realizar informes más precisos y sin riesgo a daños.

1.2.2 OBJETIVOS

1.2.2.1 OBJETIVO GENERAL

Diseñar e implementar un sitio web y módulo de control para la movilización del personal operativo y vehículo del Cuerpo de Bomberos del cantón Guano.

1.2.2.2 OBJETIVOS ESPECÍFICOS

- Determinar las ventajas que posee la utilización del lenguaje de programación PHP y base de datos MYSQL en la elaboración de un sitio web.
- Realizar el análisis y diseño para determinar las necesidades y requerimientos a la hora de controlar la movilización del personal operativo de vehículos de cuerpo de bomberos de guano.
- Implementar el sitio web y módulo de control para la movilización del personal operativo y vehículo del cuerpo de bomberos del Cantón Guano desarrollado en PHP orientado a objetos y gestor de base de datos MYSQL.

CAPÍTULO II

2 MARCO TEORICO

2.1 ORIENTACION A OBJETOS

La P.O.O. es un paradigma o modelo de programación, que permite una manera de programación que permite tratar a muchos de los lenguajes que usamos ser más competitivos, al igual que permite soportar diseños bajo ese modelo (The Fricky, 2007).

2.1.1 HISTORIA EVOLUTIVA

La Programación Orientación a Objetos (P.O.O.) surge en 1967 con Simula 67 desarrollado por Krinsten Nygaard y Ole-Johan Dahl, en este lenguaje aparece por primera vez clases, corrutinas y subclases.

Por el mismo año la POO en Europa paso sin dar mucha importancia por parte de los programadores. En los años 70 el inconveniente que existía algunos programadores que no terminaban sus desarrollos y muy pocos lo hacían, pero con solo requisitos iniciales lo que conllevaba un gran problema para el desarrollo de los mismos.

Palo Alto Xerox inventaron el lenguaje Small talk que permitió resolver problemas presentados anteriormente, este lenguaje fue el primero en permitir la Orientación a Objetos.

Gracias a D. Parnas aparece la idea de encapsular variables globales en un solo módulo, al igual que el resto de los módulos (objetos) podían acceder a las variables sólo de forma indirecta mediante las operaciones diseñadas para tal efecto.

Bjarne Stroustrup de AT&T Labs. En los ochenta, amplió el lenguaje C para crear C++ que soporta la programación Orientada a Objetos.

En los 90's se consolida la Orientación a Objetos como una de las mejores maneras de programación. Sin esperar a que los requerimientos iniciales estén totalmente precisos.

En la actualidad la orientación a objetos es la mejor manera de programar, no obstante, no es la solución a todos los problemas. Ya que trata de eliminar la crisis del software (Bobarin, 2008).

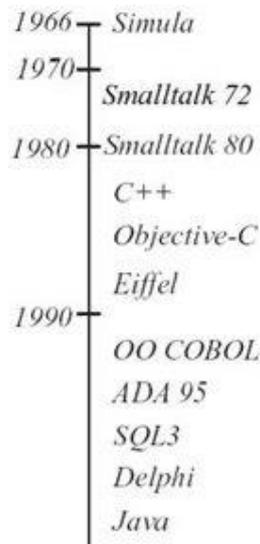


Gráfico N° 1 Historia de POO

Fuente: (Bobarin, 2008)

2.1.2 VENTAJAS SOBRE LA PROGRAMACION ESTRUCTURADA

CARACTERÍSTICA	PROGRAMACIÓN ESTRUCTURADA	VENTAJAS DE LA POO
VARIABLES	Son locales por definición esto evita la programación se mezcle.	Existen variables globales las mismas que permiten que puedan ser usadas después sin problemas y con las mismas características
Funciones y Datos	Tienen acceso ilimitado a los datos lo que permite ser	Mediante esta programación permite

	<p>más vulnerable para corromper.</p> <p>Trata a estos como independientes.</p>	<p>tratar a estos datos como uno solo, y su acceso limita a los datos a un solo tipo,</p> <p>Esto impide que sean corrompidos con facilidad.</p>
Código	<p>No permite la reutilización del código con facilidad ya sea mediante librerías o funciones.</p>	<p>Permite la reutilización de programación como de funciones, que permite una programación más sencilla y cómoda.</p>
Precio y Tiempo	<p>Es más costosa en tiempo y precio ya que una misma función se tiene que repetir varias veces de acuerdo a como se desee utilizar.</p>	<p>Mediante esta programación nos permite ahorrar tiempo y costos por que funciones y datos se pueden llamar cuantas veces lo necesite el programador.</p>

Tabla N° 1 Ventajas de la POO sobre la Programación estructurada

Fuente: (Arduino, 2017)

2.1.3 CARACTERISTICA DE LA POO

- Es un paradigma de lenguaje de programación.
- Utiliza objetos como elementos fundamentales para la solución es problemas de programación.
- Emplea herencia, cohesión, abstracción, polimorfismo, acoplamiento y encapsulamiento.
- Se diferencia de la programación estructurada ya que los datos y los procedimientos están separados sin relación, esto sucede porque el procesamiento de unos datos de entrada se obtiene de otros de salida.
- Los programadores que emplean POO, definen primero los objetos para luego enviarles mensajes solicitándoles que realicen sus métodos por sí mismos (Alegsa, 2015)

2.1.4 ELEMENTOS DE PROGRAMACION

La POO es el manejo de clases organizadas para generar un programa, estas tienen atributos básicos y pueden generar objetos con métodos y atributos específicos.

Atributo: Los atributos son las propiedades o estados de un elemento Clase u objeto, los atributos se declaran como variables del elemento y ayudan a estructurar el objeto de la clase.

Método: Los métodos son el conjunto de funciones que pueden tener los elementos de un POO clase u objeto, pueden ser funciones aritméticas, comparación, medición, etc. Estos métodos pueden ser privados o públicos.

Clase: Una clase en POO es una plantilla para la creación de objetos, esta plantilla es un elemento genérico o básico que tiene las características generales, comportamientos, atributos del objeto que se quiera crear.

Objeto e instancia: Es una unidad de programa que consta de atributos o propiedades y de funciones o métodos, esta unidad de programa se desarrolla a partir de una plantilla o clase, el desarrollo del objeto a partir de una clase (plantilla) se llama instancia (Cisneros, 2017).

2.1.5 SITIO WEB

Es un grupo de páginas desarrolladas con código HTML, que se encuentran en un dominio de Internet que tienen como objetivo su visualización en WWW, que permita a los usuarios la interactividad y el acceso a los mismos en distintas partes del mundo.

2.1.5.1 CLASIFICACIÓN DE SITIOS WEB

Existen 2 tipos de Sitios web

- **Sitios Web Estáticos:** Son aquellos que no necesitan ingresar o contienen una base de datos para obtener resultados. Este sitio es usado por propietarios de empresas que no requieren un cambio en la información.
- **Sitios Web Dinámicos:** Estos sitios acceden a un BD para obtener contenidos y reflejar la información contenida en una base de datos. Y así permitir al usuario del sistema modificar, eliminar y agregar contenidos en el sitio.

El tener un Sitio Web puede incidir positivamente en los resultados de todas las áreas funcionales de la empresa como marketing y comercial, ventas, compras, recursos humanos, y administración entre otras; ofreciendo al navegante todo tipo de información y servicios de valor agregado para sus clientes, tales como:

- Información institucional, servicios ofrecidos, catálogo de productos, actividades, fotos, videos, formas de contacto, etc.
- Promocionar producto o servicios mediante cupones de ventas online, banners, publicidad, sponsors, inscripciones, cursos.

- y toda información necesaria para el interesado, que sean relevantes para la persona que está navegando.

2.1.5.2 TIPOS DE SITIOS WEB

- **Institucionales**



Gráfico N° 2 Sitio Web Institucional

Estos sitios web son los más sencillos ya que contienen información básica de la empresa, con poca información, como:

- Home o página Principal
- Acerca de (Misión, visión, valores, objetivos)
- Servicios (Detalle de cada servicio ofrecidos por la empresa)
- Ubicación (mapa de ubicación de la oficina comercial y sucursales)
- Contacto (dirección, teléfonos y formulario de contacto)

- **OnePage o Landing Page**



Gráfico N° 3 OnePage



Gráfico N° 4 Landing Page

Son sitios que solo tienen una página de información y el usuario va accediendo al contenido a medidas que va desplazando hacia abajo con el mouse o el teclado. Es un concepto

Una Landing Page es un concepto relativamente nuevo, con bajo costo. La información contenida suele ser bastante reducida pero no deja de ser efectiva si está bien lograda y segmentada la información.

- **Blogs**



Gráfico N° 5 Blog

Son sitios de carácter personal, que permiten que otros usuarios se beneficien de las actualizaciones dinámicas y continuas. Estos sitios permiten almacenar artículos escritos y se consideran sitios dinámicos.

- **Sitios e-commerce**

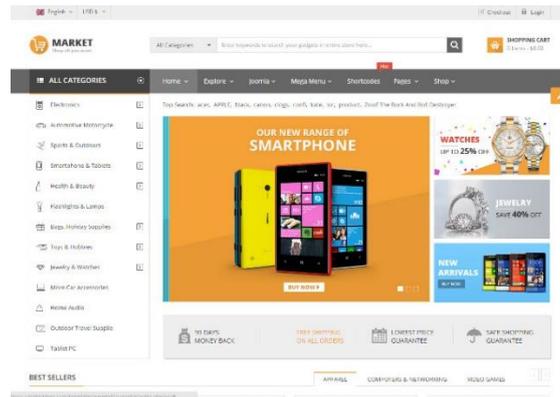


Gráfico N° 6 Stio e-commerce

Mediante estos sitios se realiza comercio electrónico que contienen carritos de compras o ventas. Estos sitios web permiten realizar transacciones entre comprador y vendedor permitiendo:

- Realizar compra-venta mediante tarjeta de crédito
- Realizar un pedido online
- Reserva de producto o servicio (codedimension, 2018).

2.2 PHP



Gráfico N° 7 LOGO PHP

2.2.1 DESCRIPCION DE PHP

“PHP es un lenguaje de código abierto, del lado del servidor e incorporado en HTML que es compatible con todos los principales servidores web (más notablemente Apache). PHP le permite insertar código fragmentos en páginas HTML normales, código que se interpreta como sus páginas se sirven hasta usuarios. PHP también sirve como un lenguaje "pegamento", por lo que es fácil conectar sus páginas web a bases de datos del lado del servidor” (Morgan, 2004)

“El lenguaje PHP es un lenguaje de programación de estilo clásico, es decir, es un lenguaje de programación con variables, sentencias condicionales, ciclos (bucles), funciones.... No es un lenguaje de marcado como podría ser HTML, XML o WML. Está más cercano a JavaScript o a C, para aquellos que conocen estos lenguajes” (Gonzalez, 2005).

PHP es un lenguaje de código abierto, en el cual podemos crear páginas WEB dinámicas, el código de PHP es invisible para el usuario, definiendo así la interacción, desarrollo y aplicación de páginas creadas en este lenguaje son por completo transformadas para que el usuario solo pueda visualizar imágenes, multimedia y formatos que permitan interactuar añadiendo o descargando información.

2.2.2 ELEMENTOS DE PHP

El lenguaje PHP presenta cuatro grandes características:

1. **Velocidad:** PHP no requiere de grandes recursos tanto Hardware como Software esto permite que sea rápido para su ejecución por lo tanto esto no genera retrasos. PHP se integra muy bien junto a otras aplicaciones, especialmente bajo ambientes Unix.
2. **Estabilidad:** Al utilizar su propio sistema de administración de recursos PHP se convierte en un sistema robusto y estable que posee un sofisticado método de manejo de variables.
3. **Seguridad:** PHP maneja distintos niveles de seguridad, estos pueden ser configurados desde el archivo **.ini**
4. **Simplicidad:** Para poder programar en PHP no es necesario ser expertos o tener experiencia, los usuarios que conocen o tienen experiencia en C y C++ podrán utilizar PHP rápidamente. Para lo cual también PHP dispone de una amplia gama de librerías, y gracias a su constante actualización existe una gran variedad de documentación en la WEB (Red Gráfica Latinoamérica, 2013).

2.3 PROGRAMACION PHP POO

La POO es un paradigma que permite utilizar objetos, para diseñar aplicaciones y programas informáticos. Esta manera de programar contiene técnicas, como herencia, cohesión, abstracción, polimorfismo, acoplamiento y encapsulamiento. Este tipo de programación se publicó y apareció en 1990.

2.3.1 VENTAJAS DE LA POO

- Fomenta la reutilización y extensión del código.
- Permite crear sistemas más complejos.

- Relacionar el sistema al mundo real.
- Facilita la creación de programas visuales.
- Construcción de prototipos
- Agiliza el desarrollo de software
- Facilita el trabajo en equipo
- Facilita el mantenimiento del software

2.3.2 VARIABLES

Las variables son identificadores que representa un espacio de memoria que contiene información. La información es del tipo con que se declaró la variable.

Hay 2 categorías de tipos para variables, los tipos primitivos (como int, short, byte, long, char, boolean y double) y las referencias a objetos (como String, Array y otros objetos).

Sintaxis

SET

```
public function setIdUsuario($valor) {
    $this->idUsuario = $valor;
}
```

GET

```
public function getIdUsuario() {
    return $this->idUsuario;
}
```

2.3.3 INCLUSIÓN DE CÓDIGO PHP POO EN UNA PÁGINA HTML

PHP a pesar de ser una herramienta de libre código, siempre deberemos identificar la estructura de en donde el código se visualizará, al incluir código PHP se deberá colocarlo dentro de la etiqueta:

Etiqueta de apertura	<code><?php,</code> Código de programación
Etiqueta de cierre	<code>?>.</code>

Una vez que las etiquetas de apertura y cierre estén colocadas se debe proceder con el código de programación, la misma que si el servidor web se ha configurado bien, este detectara dicho código y se presentara al cliente la página WEB(el diseño, y no en código fuente) (Palomo, 2014).

2.3.4 SENTENCIAS DE CONTROL

Son aquellas que permiten realizar una programación más controlada. Estas sentencias finalizaran siempre con un punto y coma, también están agrupadas como sentencias encapsuladas entre corchetes

- **IF.-** Permite la ejecución condicional de fragmentos de código, esta expresión es evaluada a su valor booleano, si la expresión se evalúa como TRUE, se ejecutará la sentencia y si se evalúa como FALSE la ignorará.

Sintaxis

```
if (expr)
{
    sentencia
}
```

- **ELSE.-** Se utiliza cuando se desea extender una sentencia if , en caso de que esta se evalué como FALSE.

Sintaxis

```
if (expr)
{
    sentencia
}
else
{
    sentencia
}
```

- **ELSEIF/ELSE IF.**- A diferencia de IF ELSE esta sentencia es una expresión alternativa que solo se ejecutara si ELSE IF se evalúe como TRUE.

Sintaxis

```
if (expr)
{
    sentencia
}
elseif
{
    sentencia
}
else
{
    sentencia
}
```

2.3.5 VENTAJAS DE PHP

- Es un lenguaje multiplataforma.
- Orientado al desarrollo de aplicaciones web dinámicas que tienen la información en el BD.
- El código en PHP es invisible tanto para el navegador y el cliente, dado que el servidor es el encargado de ejecutar el código y regresarlo en forma de HTML.

- La conexión en los motores de la BD destaca PHP con MySQL
- Utiliza módulos para expandir el potencial la capacidad (Aguayo, 2016).

2.3.6 SEGURIDAD DE PHP

2.3.6.1 PROTECCIÓN CONTRA LA INYECCIÓN DE CÓDIGO SQL

Cuando se trabaja con MySQL las consultas dependen de parámetros que llegan desde GET o POST, esto permite que los usuarios de la web modifiquen por así decirlo el código.

Permitiendo así a un usuario construir código malicioso para dañar, reescribir y obtener información del sitio.

Si tenemos la información básica pero primordial guardada en una BD, esto podemos protegerlo con mediante parámetros como POST y variables como `$_POST['usuario']` y `$_POST['clave']`. Tras ello realizamos la siguiente consulta SQL:

`mysql_query('select id from usuariosWeb where usuario = "'.$_POST['usuario'].'" and password = "'.$_POST['clave'].'"');` Si al implementar este código el usuario malicioso obtendría acceso al sistema. Pero si se implementa el siguiente código `$_POST['clave'] = "m' or 1=1"`, podemos observar que nos devuelve un valor válido para identificarse conociendo sólo el nombre de usuario.

Para solucionar este problema se implementa varios métodos el más común es substituir las comillas simples y dobles de todos los parámetros POST mediante el siguiente

código:

```
foreach($_POST as $key => $value) {
    $_POST[$key] = str_replace("'", "", $value);
    $_POST[$key] = str_replace('"', "", $value);
}
```

2.3.7 USO DE VARIABLES DE SESIÓN Y NO DE COOKIES

Mediante el uso de las cookies se puede realizar una práctica muy insegura ya que, si lo hiciéramos, esta información es transmitida mediante la red en texto plano y podría ser leída por cualquier persona que realice un sniffing de red o que lea las cookies guardadas en nuestro ordenador mediante un virus o troyano.

Por ello lo mejor es hacer uso de las variables de sesión. Estas variables se guardan en el servidor por lo que, en cualquier caso, la información transmitida sólo es sensible la primera vez que se envía y no las sucesivas veces (por tanto, esto sería evitable haciendo uso del protocolo HTTPS en vez de HTTP).

2.3.8 SUPLANTACIÓN DE SESIONES

En casos donde un usuario, malicioso intenta suplantar al cliente en el servidor. Este caso se trata cuando iniciamos una sesión, la cookie obtiene un identificador único y mandase esa cookie al servidor.

Si estos cambian significaría que es un intento de session hijacking y por tanto es un usuario que realiza un ataque. En el siguiente ejemplo tenemos dos archivos, en el que en el primer archivo realizamos un login satisfactorio y guardamos los datos iniciales de dirección IP y tipo del navegador con el que lo hemos hecho:

```
session_start();
$_SESSION['REMOTE_ADDR'] = $_SERVER['REMOTE_ADDR'];
$_SESSION['HTTP_USER_AGENT'] = $_SERVER['HTTP_USER_AGENT'];
```

En pantallas posteriores, comprobaremos si en cada pantalla los datos actuales de dirección IP y datos del navegador se corresponden con los que se realizó el login como usuario en la web:

```
session_start();
if($_SESSION['REMOTE_ADDR'] != $_SERVER['REMOTE_ADDR'] ||
```

```
$_SESSION['HTTP_USER_AGENT'] != $_SERVER['HTTP_USER_AGENT']) {  
    exit();  
}
```

2.3.9 PARÁMETROS GET/POST EXPLOTABLES

Cuando utilizamos en los parámetros GET y POST se verifica siempre que se correspondan con el usuario actual o si tiene permisos.

Por ejemplo, si tenemos un archivo que edita productos llamado editar Producto.php y recibe el parámetro GET idproducto, que se corresponde con el identificador único del producto a editar y llamamos al archivo de la siguiente manera:

Deberemos comprobar en cualquier caso que en primer lugar seamos un usuario registrado y logueado y después que el producto de identificador único 15 corresponde a uno de los productos del usuario logueado (imaginanet, 2010).

2.4 MYSQL



Gráfico N° 8 LOGO MySQL

2.4.1 QUE ES UN GESTOR DE BASE DE DATOS

Un gestor de BD es un software invisible, que es la unión de varios el lenguaje como lenguaje de definición de datos, de manipulación y consulta los mismos que pueden trabajar en distintos niveles.

El SGBD es la colección de datos relacionados, estructurados y organizados que se encuentran en un conjunto de programas que acceden y facilitan su gestión (Power Data, 2016).

2.4.2 DEFINICION DE MYSQL

MySQL es de código abierto basado en un lenguaje estructurado, mediante un gestor de base de datos. Se ejecuta en las plataformas Linux, UNIX y Windows, este gestor de BD se asocia con aplicaciones WEB y publicaciones en línea de código abierto al igual que Linux.

Tiene como servidor web a Apache, MySQL y se adapta a diferentes lenguajes de programación como son PHP, Perl o Python (Rouse, 2015).

2.4.3 POR QUÉ UTILIZAR MYSQL

- MySQL es un sistema Open Source.
- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.

- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Facilidad de configuración e instalación.
- Soporta gran variedad de Sistemas Operativos
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Su conectividad, velocidad, y seguridad hacen de MySQL Server altamente apropiado para acceder bases de datos en Internet
- El software MySQL usa la licencia GPL (Perez, 2017).

2.4.4 TIPOS DE DATOS

Al crear una tabla la elección correcta de un formato de dato para cada columna de la tabla hará que nuestra BBDD tenga un rendimiento óptimo a medio largo plazo.

Repasando la documentación oficial podríamos dividir en 3 grandes grupos estos datos:

2.4.4.1 TIPOS DE DATO NUMÉRICOS

Listado de cada uno de los tipos de dato numéricos en MySQL, su ocupación en disco y valores.

DATOS	DESCRIPCIÓN
INT (INTEGER)	Ocupación de 4 bytes con valores entre -2147483648 y 2147483647 o entre 0 y 4294967295.
SMALLINT	Ocupación de 2 bytes con valores entre -32768 y 32767 o entre 0 y 65535

TINYINT	Ocupación de 1 bytes con valores entre -128 y 127 o entre 0 y 255
MEDIUMINT	Ocupación de 3 bytes con valores entre -8388608 y 8388607 o entre 0 y 16777215
BIGINT	Ocupación de 8 bytes con valores entre -8388608 y 8388607 o entre 0 y 16777215
DECIMAL (NUMERIC):	Almacena los números de coma flotante como cadenas o string
FLOAT (m,d)	Almacena números de coma flotante , donde ‘m’ es el número de dígitos de la parte entera y ‘d’ el número de decimales
DOUBLE (REAL):	Almacena número de coma flotante con precisión doble. Igual que FLOAT, la diferencia es el rango de valores posibles
BIT (BOOL, BOOLEAN)	Número entero con valor 0 o 1

Tabla N° 2 Tipos de Datos Numéricos

Fuente: (Barrena, 20915)

2.4.4.2 TIPOS DE DATO CON FORMATO FECHA

Listado de cada uno de los tipos de dato con formato fecha en MySQL, su ocupación en disco y valores.

DATOS	DESCIPCION
DATE	Válido para almacenar una fecha con año, mes y día, su rango oscila entre ‘1000-01-01’ y ‘9999-12-31’

DATE TIME	Almacena una fecha (año-mes-día) y una hora (horas-minutos-segundos), su rango oscila entre '1000-01-01 00:00:00' y '9999-12-31 23:59:59'
TIME	Válido para almacenar una hora (horas-minutos-segundos). Su rango de horas oscila entre -838-59-59 y 838-59-59. El formato almacenado es 'HH:MM:SS'
TIMESTAMP	Almacena una fecha y hora UTC. El rango de valores oscila entre '1970-01-01 00:00:01' y '2038-01-19 03:14:07'
YEAR	Almacena un año dado con 2 o 4 dígitos de longitud, por defecto son 4. El rango de valores oscila entre 1901 y 2155 con 4 dígitos. Mientras que con 2 dígitos el rango es desde 1970 a 2069 (70-69)

Tabla N° 3 Tipos de Datos Fecha

Fuente: (Barrena, 20915)

2.4.4.3 TIPOS DE DATO TEXTO

Listado de cada uno de los tipos de dato con formato string en MySQL, su ocupación en disco y valores.

DATOS	DEFINICION
CHAR	Ocupación fija cuya longitud comprende de 1 a 255 caracteres
VARCHAR	Ocupación variable cuya longitud comprende de 1 a 255 caracteres
TINYBLOB	Una longitud máxima de 255 caracteres. Válido para objetos binarios como son un fichero de texto, imágenes, ficheros de audio o vídeo. No distingue entre minúsculas y mayúsculas

BLOB	Una longitud máxima de 65.535 caracteres. Válido para objetos binarios como son un fichero de texto, imágenes, ficheros de audio o vídeo. No distingue entre minúsculas y mayúsculas
MEDIUMBLOB	Una longitud máxima de 16.777.215 caracteres. Válido para objetos binarios como son un fichero de texto, imágenes, ficheros de audio o vídeo. No distingue entre minúsculas y mayúsculas
LOB	Una longitud máxima de 4.294.967.298 caracteres. Válido para objetos binarios como son un fichero de texto, imágenes, ficheros de audio o vídeo. No distingue entre minúsculas y mayúsculas
SET	Almacena 0, uno o varios valores una lista con un máximo de 64 posibles valores
ENUM	Igual que SET pero solo puede almacenar un valor
TINYTEXT	Una longitud máxima de 255 caracteres. Sirve para almacenar texto plano sin formato. Distingue entre minúsculas y mayúsculas
TEXT	Una longitud máxima de 65.535 caracteres. Sirve para almacenar texto plano sin formato. Distingue entre minúsculas y mayúsculas
MEDIUMTEXT	Una longitud máxima de 16.777.215 caracteres. Sirve para almacenar texto plano sin formato. Distingue entre minúsculas y mayúsculas
LONGTEXT	Una longitud máxima de 4.294.967.298 caracteres. Sirve para almacenar texto plano sin formato. Distingue entre minúsculas y mayúsculas

Tabla N° 4 Tipos de datos tipo texto

Fuente: (Barrena, 20915)

2.5 COMPATIBILIDAD ENTRE MYSQL Y PHP

CARACTERISTICAS	MYSQL	PHP
Dispone una amplia cantidad de documentación y ayuda en línea.	✓	✓
Velocidad para realizar las operaciones	✓	✓
Bajo costo para el desarrollo de proyectos	✓	✓
Facilidad de configuración e instalación	✓	✓
Conectividad segura y confiable	✓	✓
Integración completa	PHP	MYSQL
Programación web	✓	✓

Tabla N° 5 Cuadro comparativo

Fuente: (Guzman, 2016)

CARACTERISTICAS	POSTGRESQL	PHP
Dispone una amplia cantidad de documentación y ayuda en línea.	✓	✓
Velocidad para realizar las operaciones	✓	✓
Bajo costo para el desarrollo de proyectos	✓	✓
Facilidad de configuración e instalación	X	✓
Conectividad segura y confiable	✓	✓
Integración completa	Con algunas versiones de PHP	

Tabla N° 6: Cuadro comparativo

Fuente: Marcelo Orozco

CARACTERISTICAS	SQL SERVER	PHP
Dispone una amplia cantidad de documentación y ayuda en línea.	X	✓
Velocidad para realizar las operaciones	✓	✓
Bajo costo para el desarrollo de proyectos	X	✓
Facilidad de configuración e instalación	X	✓
Conectividad segura y confiable	✓	✓
Integración completa	Con algunas versiones de PHP	

Tabla N° 7 Tabla comparativa

Fuente: Marcelo Orozco

Luego del análisis de herramientas se determina que entre las arquitecturas. Por tal razón para el desarrollo e implementación del sitio web y módulo de control para la movilización del personal operativo y vehículo del cuerpo de bomberos del cantón Guano, se utilizó PHP y MYSQL, ya que existe una compatibilidad del 100%, que permitirá el desarrollo óptimo de la aplicación.

2.6 JAVASCRIPT

2.6.1 DEFINICION E HISTORIA

En una tecnología que permite la interactividad, y mediante estos realizar efectos especiales para el desarrollo y así dar una mejor imagen en la página. Mostrando páginas con mapas interactivos o gráficos animados, video, o sonidos.

En los años 90, con aplicaciones cada vez que solicitaban mejoras en el desarrollo visual, por la tecnología que comenzaba a desarrollar, se vio en la necesidad de crear un lenguaje que se ejecute en el lado del usuario.

Brendan Eich, creo la primera versión que lo llamo Live Script, que soportaba las nuevas tecnologías. Después mediante una alianza con Sun Microsystems, crearon un nuevo lenguaje y por motivos de que Java era un nombre más (Eguiluz, 2012)

2.6.2 DIFERENCIAS CON OTROS LENGUAJES

Podemos realizar los efectos con diferentes lenguajes como son HTML, Javascript, o CSS. Pero que lenguaje podemos utilizar para realizar es una pregunta difícil de contestar.

- a) **Motivos históricos:** para el desarrollo de algunas páginas que fueron creadas antes de este lenguaje permite hacer las cosas a la antigua.
- b) **Motivos de independencia de tecnologías:** HTML y JavaScript estos lenguajes al ser diferentes sería imposible que se mezclaran, pero tomando en cuenta que JavaScript puede incrustarse en HTML, pero JavaScript tiene como característica de potenciar la independencia del programa.
- c) **Motivos de políticas de desarrollo y estándares:** A pesar de que detrás de estos lenguajes, existe una gran variedad de personas individuales, las empresas, y

comunidades de desarrollo tienen que aprobar estas actualizaciones. (Rodríguez, 2015).

2.6.3 ELEMENTOS PARA JAVASCRIPT

En el siguiente listado se definirán las librerías que se utilizan para el desarrollo de un proyecto web:

- **Express.js:** permite al cliente desarrollar aplicaciones con Node.js. Tiene una curva de aprendizaje sólida, estable y fácil de aprender.

Entre algunas de sus características más destacadas tenemos:

- Permite la creación de APIs
- Ofrece Router de URL.
- Incorporar librerías JavaScript como jQuery y Backbone.js.
- **Angular.js:** utilizando MVC en el lado del cliente, se utiliza para el desarrollo de aplicaciones, es un framework estable y sólido para el desarrollo. Sus características fundamentales son:
 - El Modelo y la Vista siempre se actualizan de forma sincronizada con cualquier cambio en los datos.
 - Tiene patrón de Inyección de Dependencias.
 - Compatible con varios navegadores. El JavaScript se adapta a cada uno de ellos.
- **Ember.js:** crea aplicaciones de una sola página del lado del cliente. Algunas de las características fundamentales son:
 - Modificación automática entre los objetos JavaScript y el DOM.
 - Clases y procedimiento para renderizar de forma sencilla una app.

- Dispone de un sistema de plantillas para programar el HTML de la aplicación.
- **Knockout.js:** Librería que contiene una gran cantidad tutoriales y documentación.
Algunas de sus características principales:
 - Facilita el desarrollo de **bindings** declarativos.
 - Seguimiento de dependencias: **permite actualizar los cambios en la Vista y el Modelo-Vista en todos los elementos y objetos independientes.**
 - Sistema de plantillas.
 - Actualización automática de la interfaz de usuario **en función de los cambios que el desarrollador acometa en el Modelo-Vista.**

2.7 LIBRERÍAS JS

- **JQuery:** Esta librería permite la interacción con código HTML, permite realizar todo tipo de animaciones, con pocas líneas de código. Algunas de sus características esenciales son:
 - **Permite modificar el árbol del DOM.**
 - **Soporte para CSS3.**
 - **Manipulación total de las hojas de estilos CSS.**
 - **Creación de eventos.**
 - **Desarrollo de animaciones complejas.**
 - **Soporta extensiones.**
 - **Compatible con todos los navegadores**
- **Backbone.js:** El uso en esta librería de JavaScript y AJAX son esenciales para que las apps tengan una funcionalidad integra. Sus características:

- **Facilita el trabajo intenso con datos.**
- **Creación de eventos.**
- **Enlace con la tecnología por el lado del servidor: PHP, Node.js...**
- **Uso de la API para incorporar funcionalidades.**
- **Utilización simplificada de vistas.**
- **Kendo UI:** más que una librería es un paquete de librerías, los que permiten al cliente crear aplicaciones dinámicas, este paquete utiliza widgets. Características principales:
 - **Funciona con la mayoría de navegadores y sistemas operativos.**
 - **Permite visualización de datos.**
 - **Data-binding.**
- **Modernizr:** esta librería crea aplicaciones que se adapten a las características del navegador. (BBVA, 2015)

2.7.1 VENTAJAS Y DESVENTAJAS

2.7.1.1 VENTAJAS

JavaScript tiene ventajas que podemos destacar como:

- Es un lenguaje muy sencillo.
- Es rápido, por lo tanto, tiende a ejecutar las funciones inmediatamente.
- Cuenta con múltiples opciones de efectos visuales.
- Es soportado por los navegadores más populares y es compatible con los dispositivos más modernos, incluyendo iPhone, móviles y PS3.

- Es muy versátil, puesto que es muy útil para desarrollar páginas dinámicas y aplicaciones web.
- Es una buena solución para poner en práctica la validación de datos en un formulario.
- Es multiplataforma, puede ser ejecutado de manera híbrida en cualquier sistema operativo móvil.
- Es el único lenguaje que permite trabajar modo FullStack en cualquier tipo de desarrollo de programación.

2.7.1.2 DESVENTAJAS

Al igual que contiene ventajas también contiene varias desventajas como:

- En el FrontEnd sus códigos son visibles, por lo tanto pueden ser leídos por cualquier usuario.
- Tiende a introducir gran cantidad de fragmentos de código en los sitios web.
- Sus opciones 3D son limitadas, si se quiere utilizar este lenguaje de programación para crear un juego, deben emplearse otras herramientas.
- No es compatible en todos los navegadores de manera uniforme.
- Los usuarios tienen la opción de desactivar JavaScript desde su navegador.
- Sus scripts son limitados por razones de seguridad y no es posible realizar todo con JavaScript, por lo tanto, es necesario complementarlo con otros lenguajes evolucionados y más seguros. Esta es una de las características de JavaScript que algunos expertos lo contemplan como una ventaja y otros como una desventaja (Duarte, 2012).

2.8 CSS3

2.8.1 INTRODUCCION AL CSS3

Es utilizado para crear páginas web con aspecto para diferentes dispositivos. Que cuando se junta con HTML entra en contraste, siendo que CSS es un lenguaje renderizado.

En la versión CSS3 este añade nuevas capacidades. A pesar de que su proceso de estandarización no es el adecuado, y su camino por recorrer es amplio, se ha creado junto con HTML5 que es uno de los mayores adelantos en diseño web (Campos, 2011).

2.8.2 EVOLUCIÓN DE CSS3

Versión	Año	Características
CSS	A inicios de 1996	<ul style="list-style-type: none">• Creación de varios niveles y perfiles.• Añade funciones a los anteriores• Los perfiles son parte de 1 o N niveles de CSS que definen algún dispositivo en particular.• Uso de perfiles móviles.
CSS1	Mediados de 1998	<ul style="list-style-type: none">• Color de texto, fondos, bordes u otros elementos.• Atributos del texto, como espaciado entre palabras, letras, líneas, etcétera.• Alineación de textos, imágenes, tablas u otros.• Propiedades de caja, como margen, borde, relleno o espaciado.• Propiedades de identificación y presentación de

		listas.
CSS2	Finales de 1998 y finaliza en el 2007	<ul style="list-style-type: none"> • Las funcionalidades propias de las capas como de posicionamiento relativo, niveles, etcétera. • El concepto de “media types“. • Soporte para las hojas de estilo auditivas • Texto bidireccional, sombras, etcétera.
CSS3	Inicios 2008 hasta la actualidad	<ul style="list-style-type: none"> • Se divide en varios documentos separados, llamados “módulos”. • Cada módulo añade nuevas funcionalidades. • Su modularidad contiene diferentes módulos pueden que encontrarse en diferentes estados de su desarrollo, • Estos módulos son: Fondos y colores, Consultas de medios o Diseños multicolumna • Esta versión es considerada la más estable y puede ser implementada en varios navegadores

Tabla N° 8. Evolución CSS3

Fuente: (WordPress, 2015).

2.8.3 NOVEDADES DE CSS3

CSS3 viene con una infinidad de novedades las más importantes son:

- **Bordes**
 - Colores múltiples de borde en un mismo lado
 - Imágenes de borde
 - Bordes redondeados

- **Fondos**

- Fondos Múltiples pueden ser añadidos al mismo elemento como capas
- Posicionamiento del fondo con mayor precisión
- Pueden ser redimensionados

- **Color**

- Opacidad
- Gradientes
- Valores de color: HSL

- **Text**

- Sombras
- Desbordamiento
- Ajuste de línea

- **Transformaciones**

- Escalar
- Sesgar
- Mover
- Rotar en 2D o 3D

- **Transiciones**

- Transición sencilla de estilos

- **Cajas**

- Sombras
- Cajas redimensionables
- Overflow separado en vertical u horizontal
- Compensación entre contorno y borde

- Modelos para especificar altura y anchura
- **Contenido**
- Los estilos pueden añadir contenido a los elementos
- **Opacidad**
- Los elementos pueden ser transparentes
- **Fuentes Web**
- Capacidad de añadir fuentes en vivo a los documentos mejorada
- Tomando en cuenta, que algunos navegadores no soportan el mismo nivel de CSS3 al igual que sus módulos, ni siquiera están implementados (Campos, 2011).

CAPÍTULO III

3 ANÁLISIS Y DISEÑO

3.1 RECOLECCIÓN DE LA INFORMACIÓN

En el cantón Guano en el que está asentado el cuerpo de bomberos, el mismo que brinda atención a la ciudadanía de este sector, se ve la necesidad de crear un sistema que permite automatizar los procesos que en la actualidad se los lleva en papel los mismos que son susceptible de perdida, daño físico por agua o fuego. Al finalizar la entrevista con el Comandante Marco Vinicio Avilés Castillo, dando como resultado la necesidad de implementar un sistema que permita el de control para la movilización del personal operativo y vehículo del cuerpo de bomberos. Con lo cual se recolectó toda la información requerida y todos los requerimientos para empezar con el proceso de creación del Sitio Web.

3.1.1 ALCANCE

En el Cuerpo de Bomberos Guano atienden varias emergencias diarias y a cualquier hora por lo que el presente proyecto cubrirá los requerimientos que conciernen al proceso vehicular y emergencias solicitado.

Para lo cual se implementará un sistema que permita automatizar los procesos, mediante una aplicación web, la aplicación será colocada en un hosting que permitirá el acceso al mismo desde cualquier lugar en que se encuentren tratando las emergencias. Las mismas opciones que serán:

- Acceso al sistema.
- Registro de datos de los voluntarios.
- Registro del siniestro a ser atendido.
- Entrega de partes de los incidentes.
- Entrega de reportes de acuerdo a la necesidad.
- Ingreso de los vehículos existentes

- Ingreso de siniestros.

3.1.2 ANÁLISIS

Una fase de análisis bien diseñada permite la creación de un Sitio Web que, mediante la automatización de los requerimientos, y el buen uso del Sitio Web para automatizar el sistema manual.

El análisis de las funciones será detallado a continuación:

- Ingreso al sistema mediante un usuario y un password que serán validados para dar su acceso.
- Mediante el almacenamiento de un base de datos la información será resguarda.
- Impresión de partes, o reportes.
- Ingreso de información vehicular, y personal del cuerpo de bomberos.

3.1.3 ESTUDIO DE FACTIBILIDAD

La factibilidad de un proyecto permite crear un sistema automatizado que permitirá, dar resultados efectivos que permitirá:

- Mejorar el sistema actual.
- Reducir el tiempo en encontrar y entregar partes solicitados.
- Actualizar y mejorar el servicio de ingreso de información de emergencias.
- Automatización de los procedimientos manuales en relación al manejo de emergencias, e información.

3.1.3.1 FACTIBILIDAD OPERATIVA

El sitio web y módulo de control para la movilización del personal operativo y vehículo del cuerpo de bomberos del cantón Guano, cuenta con el personal capacitado, al igual

que el desarrollador del sitio gracias a los conocimientos adquiridos durante la carrera, presenta un sistema de calidad y operativamente en funcionamiento.

3.1.3.2 FACTIBILIDAD TÉCNICA

El Sitio Web del Cuerpo de bomberos de Guano es considerado factible en la parte técnica, dado que el proponente cuenta con las herramientas

En la institución se cuenta con equipos de computación, que se encuentran con conexión a internet, lo que permitirá un óptimo desarrollo, e implementación del mismo.

3.1.3.3 FACTIBILIDAD ECONÓMICA

- **Online**

Económicamente el proyecto es factible ya que el proponente utilizara software bajo licencia GNU, conocidos también como software libre de desarrollo que permite tener costos bajos en desarrollo e implementación, al igual que el costo de Hosting e investigación serán cubiertos por el proponente del proyecto.

3.1.3.4 FACTIBILIDAD LEGAL

Cuando se pone en marcha un proyecto se debe tomar en cuenta que muchos sistemas cuentan con licencias que el costo de desarrollo es alto, a lo cual se tomó la decisión de realizar el proyecto con SW GNU con el cual no se requiere de autorización alguna, por ser libre, lo que se hace factible que la aplicación sea puesta en marcha, evitando tramites de tipo jurídico o legal.

Al utilizar este tipo de SW nuestra propuesta de automatización los procesos en el Cuerpo de Bomberos de Guano, será de bajo costo y sin problemas de tipo legal.

3.1.4 ANÁLISIS DE LOS REQUERIMIENTOS

3.1.4.1 REQUERIMIENTOS FUNCIONALES DEL SISTEMA

Los parámetros de requerimientos entregados para la creación del sistema se definen de cursado a las necesidades del usuario.

- Control del personal que sale a emergencias
- Los datos tienen que contar con una integridad y seguridad.
- Un usuario autorizado debe estar constando en la base de datos caso contrario este no tendrá acceso al sistema.
- La información ingresada podrá ser actualizada, eliminada e ingresada como lo desee el usuario del sistema.
- Se entregará reportes y certificados de partes de lo ingresado en el sistema, tanto de vehículos, emergencias.

3.1.4.2 REQUERIMIENTOS NO FUNCIONALES DEL SISTEMA

- **Disponibilidad:** Al ser un sistema Web este estará disponible en un 100%, para su utilización.
- **Escalabilidad:** Todo sistema web desarrollado, permitirá que el sistema crezca o la unión de nuevos módulos al mismo.
- **Seguridad:** Este sistema no permitirá acceso de terceros usuarios, ya que cuenta con scripts de control, tanto de ingreso como de información errónea.
- **Eficiencia:** La página web de los Bomberos del Cantón Guano, superará en todos los aspectos al sistema manual de esta entidad.
- **Funcionalidad:** los campos tendrán controles para que sean ingresados correctamente.

3.1.5 DEFINICIÓN DE CASOS DE USO

Actores

Usuarios: encargados de manipular todo el sistema.

Civiles: solo podrán realizar peticiones de partes.

Caso de uso Ingreso al Sistema

Objetivo: Permitir la validación a los Usuarios permitidos con sus respectivas restricciones

Flujo Principal

1. Ingreso al Sistema

2. Presenta Pantalla

Flujo Secundario

1.1 Ingreso de Credenciales
1.2 Consulta en Base de Datos
1.3 Entrega Resultados

2.1 Acceso Permitido
2.2 Acceso Negado

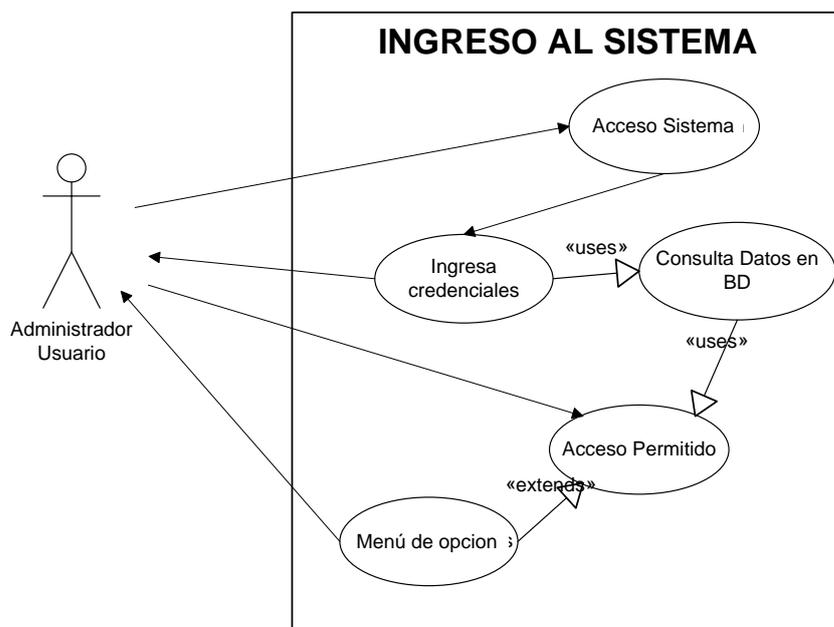


Gráfico N° 9 CU Ingres al Sistema

Fuente: Marcelo Orozco

Caso de uso Ingreso de nuevo usuario (Bombero)

Objetivo: Crear y permitir a nuevos usuarios el ingreso al sistema

Flujo Principal

1. Ingreso nuevo usuario

Flujo Secundario

- 1.1 Ingreso de Información
- 1.2 Validar Datos
- 1.3 Almacenar en Base de datos

2. Presenta
Principal

Página

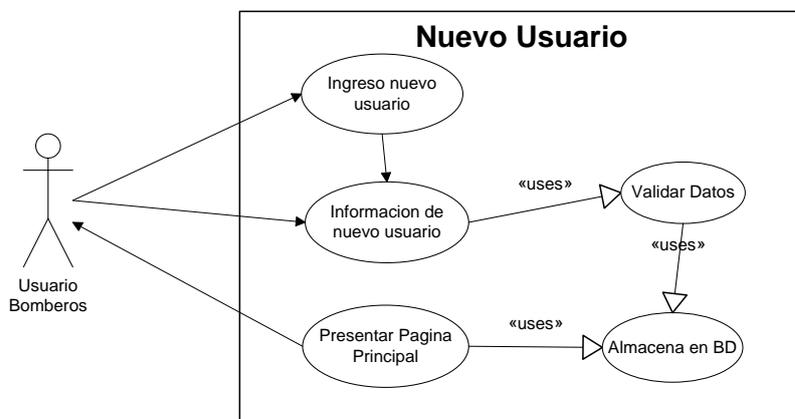


Gráfico N° 10 CU Nuevo Usuario

Fuente: Marcelo Orozco

Caso de uso nuevo parte

Objetivo: Realizar el ingreso de la información del nuevo parte por parte de los usuarios.

Flujo Principal

1. Opción de Parte

Flujo Secundario

- 1.1 Ingreso de Información
- 1.2 Validar Datos
- 1.3 Almacenar en Base de datos

2. Presenta Solicitud

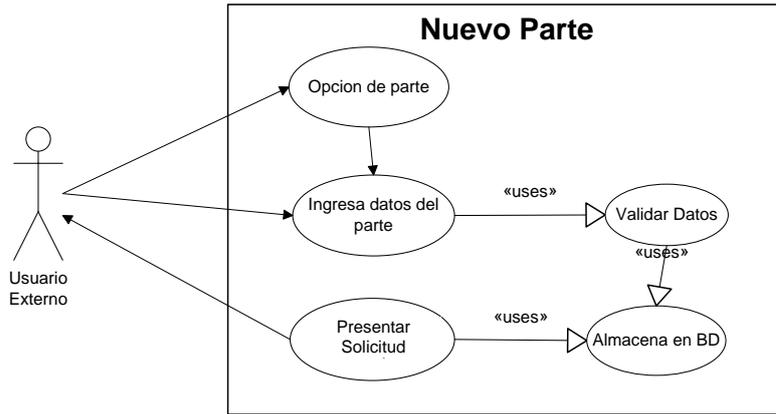


Gráfico N° 11 CU Nuevo Parte

Fuente: Marcelo Orozco

Caso de Uso de Reportes y Partes

Objetivo: Permite al usuario del Sitio web la impresión de reportes, y partes de acuerdo a lo requerido

Flujo Principal

1. Selección de Opción

Flujo Secundario

1.1 Verificar Solicitud de Parte

1.2 Verifica en BD

1.3 Genera parte Solicitado

2. Imprime Parte.

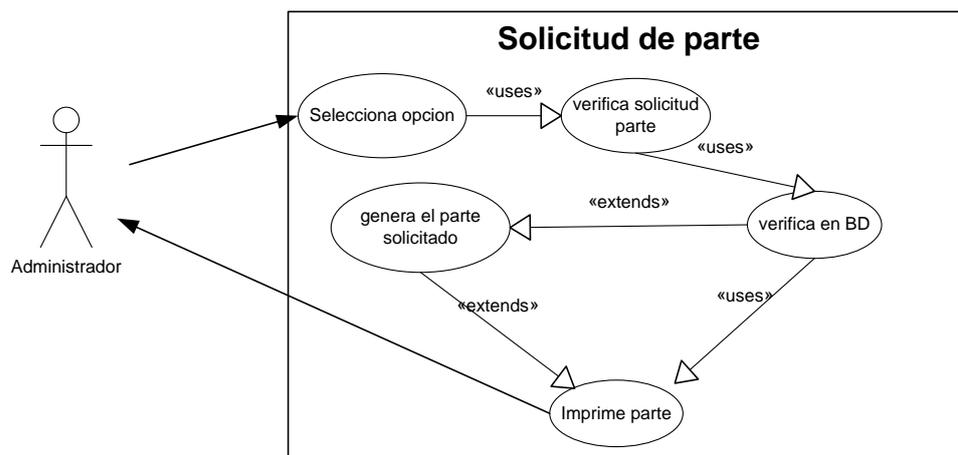


Gráfico N° 12 CU Solicitud Parte

Fuente: Marcelo Orozco

Caso de Uso Registro de Vehículos

Objetivo: Permite al usuario del Sitio web la impresión de reportes, y partes de acuerdo a lo requerido

Flujo Principal

Flujo Secundario

1. Selección de Opción

1.1 Ingresa Información

1.2 Guarda en la BD

2. Regresa al menú Principal.

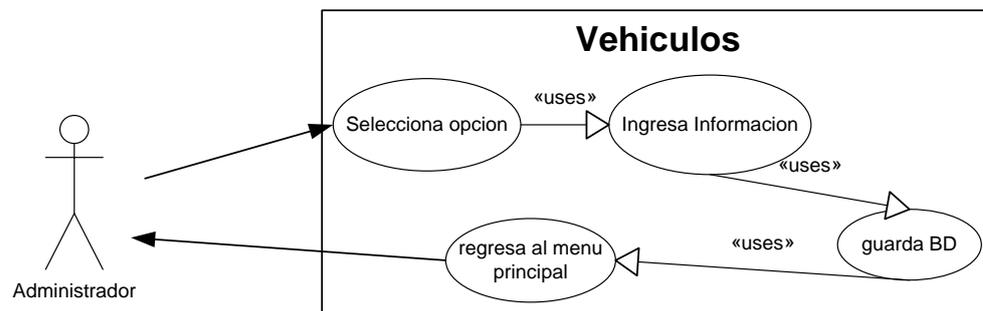


Gráfico N° 13 CU Vehículos

Fuente: Marcelo Orozco

3.2 DISEÑO

3.2.1 MODELO RELACIONAL

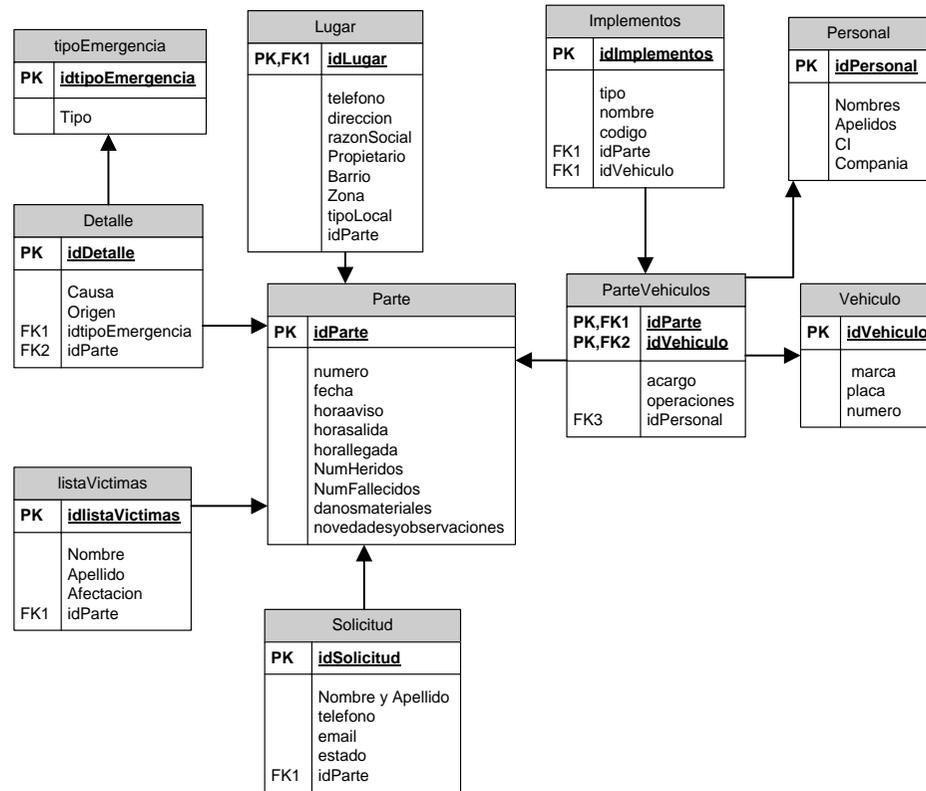


Gráfico N° 14 Modelo Relacional

Fuente: Marcelo Orozco

3.3 DICCIONARIO DE DATOS

Detalle

Columna	Tipo
idDetalle (Primaria)	int(11)
Causa	varchar(50)
Origen	varchar(50)
idtipoEmergencia	int(11)
idParte	int(11)
Estado	tinyint(1)

Tabla N° 9. DD Detalle

Fuente: Marcelo Orozco

Implementos

Columna	Tipo
idImplementos (Primaria)	int(11)
Tipo	varchar(50)
Nombre	varchar(50)
Código	varchar(50)
idParte	int(11)
pkVehiculo	int(11)
Estado	tinyint(1)

Tabla N° 10 DD Implementos

Fuente: Marcelo Orozco

Listavictimas

Columna	Tipo
idlistaVictimas (Primaria)	int(11)
Nombre	varchar(50)
Apellidos	varchar(50)
Afectacion	varchar(50)
idParte	int(11)
Estado	tinyint(1)

Tabla N° 11 DD Lista de victimas

Fuente: Marcelo Orozco

Lugar

Columna	Tipo
idLugar (Primaria)	int(11)
Teléfono	int(11)
Dirección	varchar(100)
razonSocial	varchar(50)
Propiedades	varchar(50)
Barrio	varchar(50)
Zona	varchar(50)
tipoLocal	int(11)
idParte	int(11)
Estado	tinyint(1)

Tabla N° 12 DD Lugar

Fuente: Marcelo Orozco

Parte

Columna	Tipo
idParte (Primaria)	int(11)
Numero	int(11)
Fecha	date
Horaaviso	varchar(5)
Horasalida	varchar(5)
Horallegada	varchar(5)
NumHeridos	int(11)
Numfallecidos	int(11)
Danosmateriales	varchar(200)
novedadesyobservaciones	varchar(200)
Estado	tinyint(1)

Tabla N° 13 DD Parte

Fuente: Marcelo Orozco

Partevehiculos

Columna	Tipo
idParte (Primaria)	int(11)
pkVehiculo (Primaria)	int(11)
Acargo	varchar(50)
Operaciones	varchar(50)
idPersonal	int(11)
Estado	tinyint(1)

Tabla N° 14 DD Parte Vehículos

Fuente: Marcelo Orozco

Personal

Columna	Tipo
idPersonal (Primaria)	int(11)
Nombres	varchar(50)
Apellidos	varchar(50)
CI	varchar(10)
Compania	varchar(50)
Estado	tinyint(1)

Tabla N° 15 DD Personal

Fuente: Marcelo Orozco

Solicitud

Columna	Tipo
idsolicitud (Primaria)	int(11)
NombreyApellido	varchar(100)
Teléfono	int(11)
Email	varchar(50)
Estado	varchar(10)
idParte	int(11)

Tabla N° 16 DD Solicitud

Fuente: Marcelo Orozco

Tipoemergencia

Columna	Tipo
idtipoEmergencia (Primaria)	int(11)
Tipo	varchar(50)
Estado	tinyint(1)

Tabla N° 17 DD Tipo de Emergencia

Fuente: Marcelo Orozco

Usuario

Columna	Tipo
pkusuario (Primaria)	int(11)
Nombre	varchar(200)
Nick	varchar(20)
Clave	varchar(100)
Estado	tinyint(1)
Email	varchar(100)

Tabla N° 18 DD Usuario

Fuente: Marcelo Orozco

Vehículo

Columna	Tipo
pkvehiculo (Primaria)	int(11)
Marca	varchar(100)
Placa	varchar(7)
Numero	int(11)
Estado	tinyint(1)

Tabla N° 19 DD Vehículo

Fuente: Marcelo Orozco

3.4 DISEÑO DE INTERFACES

Esquema tentativo de pantallas del Sitio Web de Bomberos de Guano

Página Principal:



Gráfico N° 15 Página Principal

Fuente: Marcelo Orozco

Acceso al sistema: Login y Contraseña

El formulario de acceso al sistema contiene los siguientes elementos:

- Etiqueta 'USUARIO' seguida de un campo de entrada de texto.
- Etiqueta 'PASSWORD' seguida de un campo de entrada de texto.
- Botón 'INGRESAR' con fondo rojo y texto blanco.
- Botón 'SALIR' con fondo rojo y texto blanco.

Gráfico N° 16 Acceso al sistema

Fuente: Marcelo Orozco

Partes, Usuarios: Actualización, eliminación y nuevos

TITULO

BUSQUEDA

Campos									
								Actualizar	Eliminar

NUEVO

Gráfico N° 17: Pantallas Secundarias

Fuente: Marcelo Orozco

CAPÍTULO IV

4 IMPLEMENTACIÓN DEL SITIO WEB

4.1 CONFIGURACIÓN DE LAS HERRAMIENTAS DE DESARROLLO

Requiere un servidor web para desarrollar el entorno anteriormente instalado, por tal motivo se procederá a la instalación de XAMPP

1. Descarga del instalador desde la página web:

<https://www.apachefriends.org/es/download.html>

2. Una vez descargado lo ejecutamos

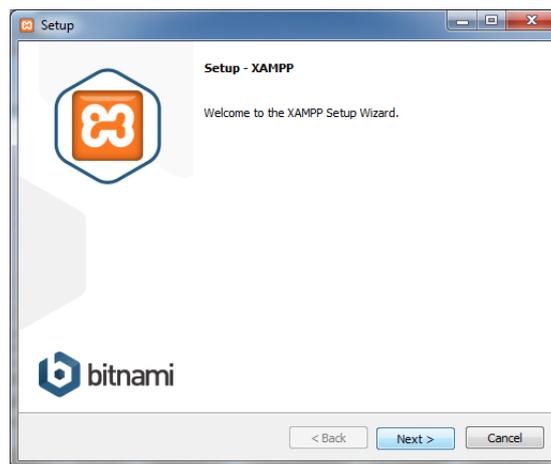


Gráfico N° 18. Setup XAMPP

Fuente: Marcelo Orozco

3. Y continuamos con la instalación dado siguiente y colocando las opciones requeridas para iniciar el proyecto.

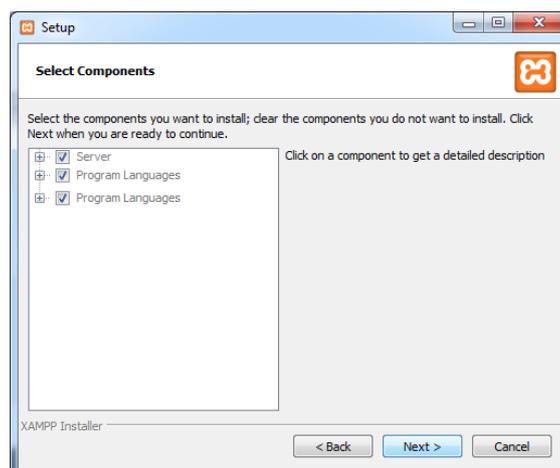


Gráfico N° 19 Configuración para el XAMPP

Fuente: Marcelo Orozco

4. A continuación, aparece la pantalla de confirmación, avanzamos a la siguiente pantalla para dar inicio a la instalación

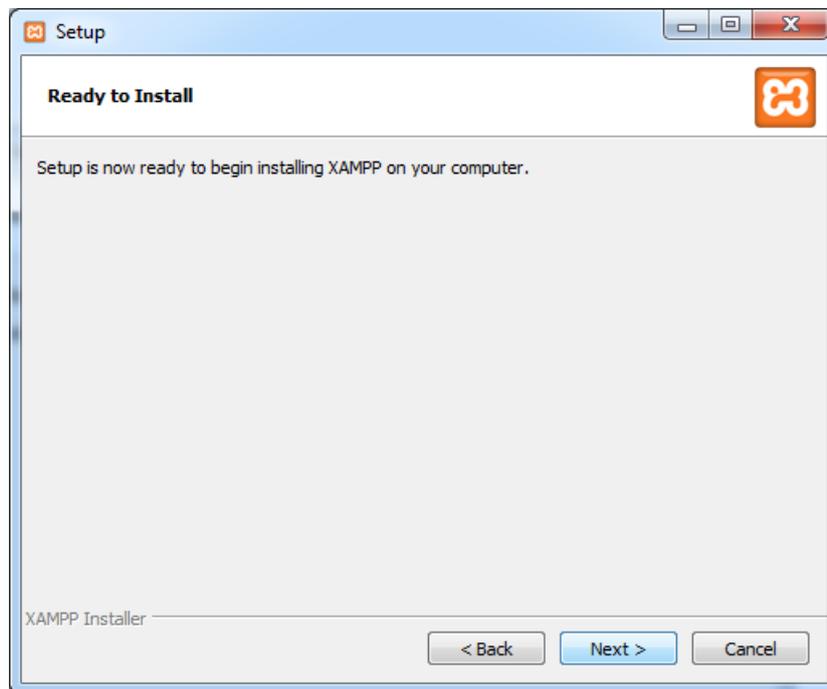


Gráfico N° 20 Finalización de Setup

Fuente: Marcelo Orozco

5. Una vez que finalice la instalación, se puede ejecutar el XAMPP Control Panel, desde el menú Inicio.

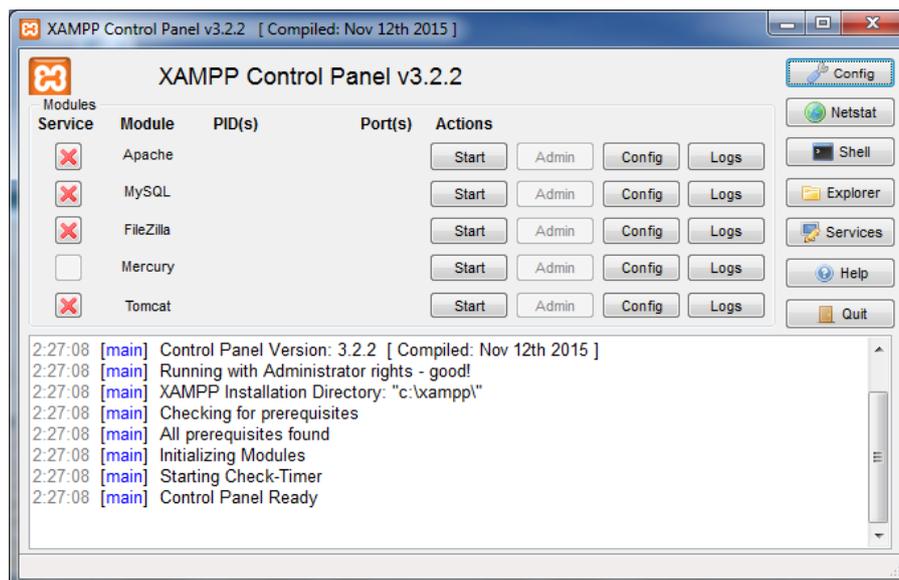


Gráfico N° 21 Inicialización del XAMPP

Fuente: Marcelo Orozco

6. Finalmente se inicia con la creación el Sitio Web

4.2 ARQUITECTURA DEL SISTEMA

Se detalla la arquitectura cliente servidor que permitirá al usuario realizar las tareas diseñadas en el sistema. Es así que se detallan el flujo operativo del sitio:

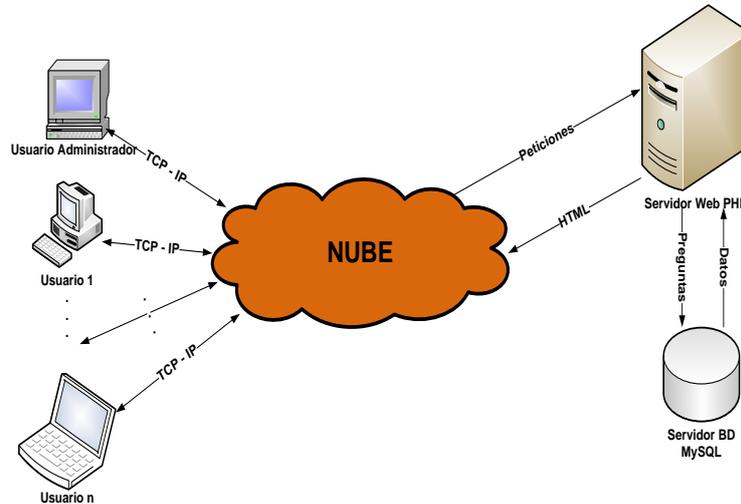


Gráfico N° 22 Arquitectura del Sistema

Fuente: Marcelo Orozco

El Sitio Web del cuerpo de bomberos de Guano accede al servidor para realizar consultas, requerimientos, el mismo que se encuentra en el hosting, adquirido por el investigador recibe esas peticiones y de acuerdo a las credenciales accede o niega las funciones del sitio al usuario.

4.3 IMPLEMENTACIÓN DEL SISTEMA

4.3.1 DEFINICIÓN DE MÓDULOS

Los módulos de los que se encuentra formado el sitio, los mismos que se encuentran en varios componentes con relación y navegabilidad directa

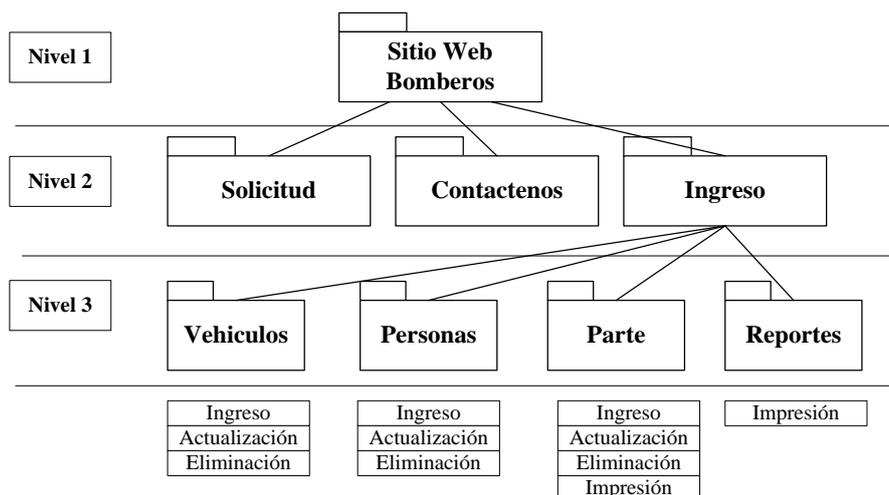


Gráfico N° 23 Módulos del Sistema

Fuente: Marcelo Orozco

4.3.2 DESARROLLO DE LA APLICACIÓN IMPLEMENTACIÓN

Una vez finalizadas las etapas de análisis, diseño instalada las herramientas de desarrollo, se procede el desarrollo de la aplicación, que posteriormente se subirá a la web.

4.3.3 IMPLEMENTACIÓN DE LA APLICACIÓN

Este sitio fue desarrollado con herramientas de licencia GNU y subido al hosting.

4.4 PRUEBAS DE APLICACIÓN

Para comprobar la funcionalidad del sistema se realizaron las siguientes pruebas:

PRUEBAS	SI	NO
Presenta página informativa	✓	
Ingresar, listar, elimina vehículos	✓	
Ingresar, eliminar y actualizar a los usuarios	✓	
Imprimir partes y reportes	✓	
Permitir acceso a usuarios	✓	
Funcionalidad		
Dispositivos Móviles	X	

Navegadores Chrome, FireFox e Internet Explorer	✓	
Ingreso Correcto a la BD	✓	
Conexión de Base de Datos	✓	
Validación de cedulas	✓	
Generación de Solicitud de Datos	✓	
Reducción de tiempo al generar reportes	✓	

Tabla N° 20 Pruebas de Aplicación

Fuente: Marcelo Orozco

4.5 CAPACITACIÓN AL PERSONAL

- **Personal**

Impartidor: Marcelo Orozco, desarrollador del sistema

Destinatarios: Personal administrativo y general del cuerpo de bomberos de Guano

- **Equipos y herramientas tecnológicas**

Equipos y herramientas tecnológicas necesarias para la realización de la capacitación fueron:

Equipos electrónicos

- Computador
- Proyector de pantalla

Herramientas tecnológicas

- Power Point (diapositivas)
- Firefox
- Internet

- **Tiempo**

Tiempo estimado: 1 mes

4.6 MANTENIMIENTO DE LA APLICACIÓN

Se los realizara anualmente con la finalidad de verificar y realizar mejoras permitiendo así que el sistema sea confiable y seguro para los usuarios.

El diseño, desarrollo e implementación del Sitio está sujeta a los requerimientos del usuario, el mantenimiento que se entregara es con la finalidad de corregir o implementar nuevos módulos en el sistema.

Permitiendo así entregar un servicio de calidad y confiabilidad a los usuarios del sistema, que beneficie a la institución.

CONCLUSIONES

- El lenguaje de programación PHP es una de las herramientas más versátiles y ágiles que existen para la creación de sitios web dinámicos, además que su licencia es gratuita es compatible con el sistema gestor de base de datos MYSQL permitiendo que el sitio web aparte de ser eficiente permita una gran cantidad de almacenamiento de información, resultados de gran importancia para las empresas o establecimientos que manejan datos.
- La estructuración de un análisis adecuado permitió el desarrollo de un diseño del sitio web acorde a los requerimientos del cliente mediante la implantación de interfaces gráficas y la estructuración de una base de datos sólida que permita soportar la información que pueda ser almacenada desde la aplicación.
- El desarrollo de un sitio web en el cuerpo de bomberos de Guano permitió así otorgar una herramienta tecnológica a la institución para el buen manejo y organización del personal y de su patrimonio vehicular promoviendo el desarrollo del establecimiento y la entrega de un servicio de calidad a la comunidad.

RECOMENDACIONES

- Se tenga conocimientos sólidos en la programación orientada a objetos el lenguaje PHP para posibles modificaciones del sistema.
- Utilizar las técnicas de recolección adecuadas, realizar cuestionarios con preguntas que ayuden a la identificación de requerimientos y necesidades para el desarrollo del sistema
- Se recomienda al cuerpo de bomberos realizar mantenimiento preventivo y de corrección una vez que el sitio web se encuentre en la nube.
- Se sugiere al Cuerpo de Bomberos Guano adquirir y mantener el hosting y renovarlo cada año.
- Capacitar al personal que estará a cargo del sistema informático, sobre el funcionamiento de la plataforma web del cuerpo de bomberos del cantón Guano

BIBLIOGRAFÍA

- Aguayo, J. (10 de Octubre de 2016). Lenguaje PHP. Obtenido de <https://es.slideshare.net/JessAlfredoGonzlez/lenguaje-php-66879919>
- Alegsa. (1 de Enero de 2015). Características de la programación orientada a objetos. Obtenido de <http://www.alegsa.com.ar/Diccionario/C/26204.php>
- Arduino. (15 de Diciembre de 2017). Gestionando las librerías | Tienda y Tutoriales Arduino. Obtenido de <https://www.prometec.net/librerias/>
- Barrena, A. (1 de Agosto de 2015). Tipos de dato en una base de datos MySQL. Obtenido de <https://www.anerbarrena.com/tipos-dato-mysql-5024/>
- BBVA. (8 de Octubre de 2015). Herramientas BAsicas para programar en JavaScript. Obtenido de <https://bbvaopen4u.com/es/actualidad/conoce-mejor-tus-clientes-gracias-las-apps-de-big-data>
- Bobarin, G. (Septiembre de 2008). Lenguajes Orientados a Objetos: HISTORIA DE LOS LENGUAJES DE PROGRAMACION ORIENTADA A OBJETOS. Obtenido de http://sis324loo.blogspot.com/2008/09/historia-de-los-lenguajes-de_29.html
- Campos, O. (25 de Junio de 2011). Breve introducción a CSS3. Obtenido de <https://www.genbetadev.com/desarrollo-web/breve-introduccion-a-css3>
- Cisneros, A. (17 de Octubre de 2017). Gestionando las librerías | Tienda y Tutoriales Arduino. Obtenido de <https://www.prometec.net/librerias/>
- codedimension. (12 de Octubre de 2018). ¿Qué es y para que sirve un sitio web? Obtenido de <https://www.codedimension.com.ar/noticias-sobre-tecnologia/noticias/que-es-y-para-que-sirve-un-sitio-web/1>

- Cuesta, M. (12 de Febrero de 2018). Trabajando con objetos. Obtenido de https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Trabajando_con_objetos
- Duarte, E. (10 de Octubre de 2012). ¿Qué Es JavaScript ? Ventajas Y Desventajas. Obtenido de <http://blog.capacityacademy.com/2012/10/19/que-es-javascript-ventajas-y-desventajas/>
- Gonzalez, J. (26 de Abril de 2005). Desarrollo Web con PHP y MySQL. Obtenido de <http://educagratis.cl/moodle/mod/resource/view.php?id=4105&usg=AOvVaw2V9ryQz7Xw2ijt2jvVZOKD>
- Guzman, L. (1 de Abril de 2016). Gestores de bases de datos cuadros comparativos. Obtenido de <https://es.slideshare.net/LuisGuzmnEspinoza/gestores-de-bases-de-datos-cuadros-comparativos>
- Hill, M.-G. (30 de Octubre de 2008). MySQL: orígenes e historia. Obtenido de <http://www.mailxmail.com/curso-mysql-informatica/mysql-origenes-historia>
- IIEMD. (9 de Octubre de 2015). QUE ES MYSQL - Definición y características - IIEMD. Obtenido de <https://iiemd.com/mysql/que-es-mysql>
- imaginanet. (10 de Marzo de 2010). Seguridad en PHP básica. Obtenido de <https://www.imaginanet.com/blog/seguridad-en-php-basica-escribiendo-aplicaciones-web-seguras.html>
- Morgan, C. (10 de Julio de 2004). PHP5 and MySQL Bible. Obtenido de https://www.amazon.com/PHP5-MySQL-Bible-Tim-Converse/dp/0764557467&usg=AOvVaw3jPzqnw_eipJF_zU5_wTIy
- Palomo, M. (9 de Julio de 2014). Obtenido de http://servicio.uca.es/softwarelibre/publicaciones/apuntes_php

Perez, M. (15 de Noviembre de 2017). ventajas y desventajas de MySQL, oracle, visual foxpro y access. Obtenido de <https://mape309site.wordpress.com/2017/11/15/ventajas-y-desventajas-de-mysql-oracle-visual-foxpro-y-access/>

Power Data. (26 de Mayo de 2016). ¿Qué es un gestor de datos y para qué sirve? Obtenido de <https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/que-es-un-gestor-de-datos-y-para-que-sirve>

Red Gráfica Latinoamérica. (11 de Marzo de 2013). El Lenguaje de Programación PHP. Obtenido de <http://redgrafica.com/El-lenguaje-de-programacion-PHP>

Rodriguez, A. (9 de Octubre de 2015). Diferencias Entre JavaScript y Java, HTML, CSS, PHP... Obtenido de <https://es.scribd.com/document/284162656/Diferencias-Entre-JavaScript-y-Java-HTML-CSS-PHP>

Rouse, M. (Enero de 2015). ¿Qué es MySQL? - Definición en WhatIs.com. Obtenido de <https://searchdatacenter.techtarget.com/es/definicion/MySQL>

The Fricky. (28 de Diciembre de 2007). Programación Orientada a Objetos – ¿Qué es POO? Obtenido de <https://thefricky.wordpress.com/2007/12/28/programacion-orientada-a-objetos-%C2%BFque-es-poo/>

WordPress. (2 de Junio de 2015). HISTORIA Y EVOLUCIÓN DE CSS. Obtenido de <https://css3seoanehighlandsblog.wordpress.com/2015/06/02/historia-y-evolucion-de-css/>

ANEXOS

ANEXO 1.

MANUAL DE USUARIO

MANUAL DEL USUARIO



SITIO WEB

CUERPO DE BOMBEROS DEL

CANTON GUANO

WEBOMBEROS

VERSIÓN 1.0

REALIZADO POR:

VÍCTOR MARCELO OROZCO FREIRE

Derechos Reservados 2020

CONTENIDO

1	IMPLEMENTACIÓN DEL SISTEMA.	70
1.1	REQUERIMIENTOS MINIMOS DE HARDWARE.....	70
1.2	REQUERIMIENTOS SOFTWARE	70
2	INGRESAR AL SISTEMA	70
3	USUARIOS DEL SISTEMA	71
4	FUNCIONALIDAD GENERAL	71
4.1	BOTON RECOMENDACIONES	71
4.2	SERVICIOS	71
4.3	CONTACTANOS	73
4.4	HISTORIA	73
4.5	INGRESO DE USUARIOS	74
4.5.1	MENÚ USUARIO	74

5 IMPLEMENTACIÓN DEL SISTEMA.

5.1 REQUERIMIENTOS MINIMOS DE HARDWARE

- 6 Computador Personal con acceso a internet con características básicas como:
- 7 Intel Pentium 4
- 8 Memoria con 512 Mb
- 9 RAM 2 Gb
- 10 Resolución recomendada 1266 x 778.

10.1 REQUERIMIENTOS SOFTWARE

- 11 Navegador Mozilla Firefox Quantum [versión 3.0 o superior], Chrome [versión 4.9 o superior], Opera [versión 3.6 o superior], Safari [versión 9.1 o superior].
- 12 Acrobat Reader.
- 13 Obtener una APIKEY de Google para la utilización de Google Maps.

14 INGRESAR AL SISTEMA

Dentro de su navegador, teclee la siguiente dirección electrónica:

<http://bomberosguano.com>



Gráfico N° 24. Pantalla Principal

Elaborado por: Marcelo Orozco

Usuarios del sistema

Los usuarios para este sistema web se definen como:

15 Administrador

16 FUNCIONALIDAD GENERAL

16.1 BOTON RECOMENDACIONES

Este botón mostrará información relacionada a la empresa, que será actualizada, para todos los usuarios.

17 Información Adicional



Gráfico N° 25 Información Adicional

Elaborado por: Marcelo Orozco

17.1 SERVICIOS

En esta parte tenemos un menú de navegabilidad el cual nos permite acceder a las diferentes páginas del sitio web



Gráfico N° 26. Barra de Menú

Elaborado por: Marcelo Orozco

18 Reportar Emergencia

En caso de emergencia un usuario cualquiera puede reportar una emergencia en tiempo real. Dando click en el menú superior en la parte de reportar emergencia



Gráfico N° 27. Datos de Solicitud

Elaborado por: Marcelo Orozco

Una vez que los campos estén llenos los cuales están controlados mediante algoritmos, de control que no permiten ingresar datos erróneos o caracteres especiales. Podemos utilizar los botones de navegación.

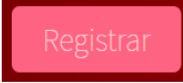
	Guarda la solicitud.
	Borra todos los campos.
	Regresa a la pantalla principal.

Gráfico N° 28. Botones de Navegación

Elaborado por: Marcelo Orozco

18.1 CONTACTANOS

La pestaña de contáctenos se encuentra en el menú superior al dar click en ella se muestra la información de los números de contacto, así como la ubicación a través del mapa del Cuerpo de Bomberos del Cantón Guano.

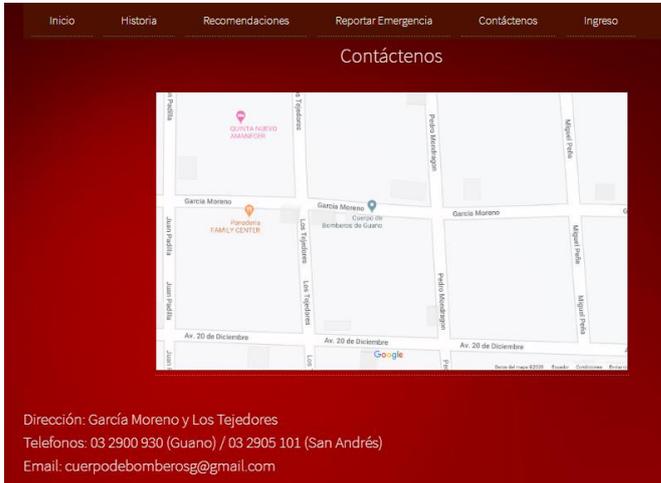


Gráfico N° 29 Contactenos

Elaborado por: Marcelo Orozco

18.2 HISTORIA

En el menú principal tenemos la opción de historia el cual nos muestra la información, de la historia del cuerpo de bomberos



Gráfico N° 30 Historia

Elaborado por: Marcelo Orozco

18.3 INGRESO DE USUARIOS

Permite el ingreso de los usuarios del sistema, los cuales tienen que tener un usuario y su contraseña de lo contrario no podrá acceder a las opciones del usuario.



Gráfico N° 31. Ingresar al sistema

Elaborado por: Marcelo Orozco

Tabla N° 21 Botones y controles de ingreso

	Ingreso al sistema
	Borra la información de los campos.
	Regresa a la pantalla principal

Elaborado por: Marcelo Orozco

18.3.1 MENÚ USUARIO

A continuación, una vez ingresado al usuario nos muestra el menú, el que nos muestra varias opciones como:

- Inicio
- Vehículos
- Personal
- Parte
- Solicitud
- Salir



Gráfico N° 32 Menú Usuario

Elaborado por: Marcelo Orozco

➤ **Registrar nuevo vehículo**

En esta opción el usuario, debe ingresar los datos de los vehículos existentes y los q se van adquirir.



Gráfico N° 33. Registro de nuevo vehículo

Elaborado por: Marcelo Orozco

Tabla N° 22. Operaciones

	Permite corregir errores de ingreso o actualizar datos de cada Vehículo
	Elimina vehículos
	Agregamos nuevo vehículo
	Mostramos un reporte de todos los vehículos en el cuerpo de bomberos(ver grafico 11)

Elaborado por: Marcelo Orozco



CUERPO DE BOMBEROS DEL CANTÓN GUANO



Reporte

Guano, 18 de febrero de 2020

Vehiculos

Marca	Placa	Número
Mack	HHH0000	10
Mercedes Benz	TBA9876	4
Hino	HAA1234	2

Firma

Gráfico N° 34 Reporte de vehículos

Elaborado por: Marcelo Orozco

➤ **Registrar nuevo Personal**

El cuerpo de bomberos, en su constante crecimiento, debe ingresar los datos del personal. En esta opción le permite realizar varias operaciones una vez que los campos estén llenos.



Gráfico N° 35. Registro de nuevo personal

Elaborado por: Marcelo Orozco

Tabla N° 23. Operaciones

	Permite corregir errores de ingreso o actualizar datos de cada personal
	Elimina personal
	Agregamos nuevo personal
	Mostramos un reporte de todos los personal en el cuerpo de bomberos (ver Gráfico 13)

Elaborado por: Marcelo Orozco



CUERPO DE BOMBEROS DEL CANTÓN GUANO

Reporte

Guano, 18 de febrero de 2020

Personal

Nombres Marcelo Daniel Eduardo	Apellidos Caceres Velarde Lopez	CI 0605340011 1234567897	Compañía Matriz Guano Central Brigada
---	--	---------------------------------------	--

Firma

Gráfico N° 36 Reporte de personal

Elaborado por: Marcelo Orozco

➤ Registrar nuevo Parte

Los partes deben ser ingresado para tener a mano en caso de ser entregados, lo cuales pueden ser modificados o eliminados.



Nro	Fecha	H. Aviso	Observaciones	Lugar	Reportado por	Modificar
100	2019-04-16	05:05	Accidente en carretera se perdió pista	Av Leopoldo Freire	Carlos garcia	Detalles Eliminar
121	2019-05-01	18:23	Accidente	Colombi y Carabobo	Carlos Castro	Detalles Eliminar
0	2019-	22:49	Inundacion	Calle principal	Carlos	Detalles Eliminar

Gráfico N° 37. Registro de nuevo personal

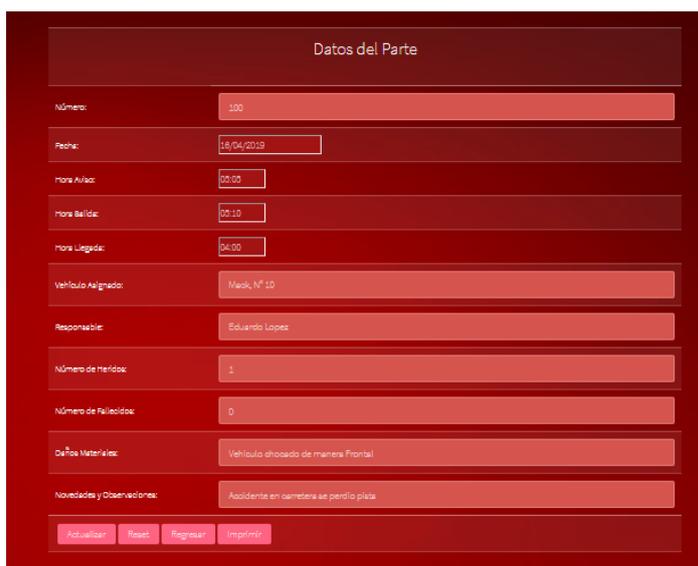
Elaborado por: Marcelo Orozco

Tabla N° 24 Operaciones

	Muestra con más detalle los partes de las emergencias atendidas.
---	--

Elaborado por: Marcelo Orozco

Los datos deben ser ingresados en orden y claros para que los partes sean precisos, y claros.



Datos del Parte

Número:

Fecha:

Hora Aviso:

Hora Salida:

Hora Llegada:

Vehículo Asignado:

Responsable:

Número de Heridos:

Número de Fallecidos:

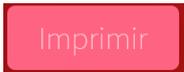
Daños Materiales:

Noticias y Observaciones:

Gráfico N° 38. Datos del Parte

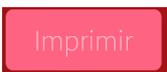
Elaborado por: Marcelo Orozco

Tabla N° 25. Botones de navegación.

	Permite corregir errores de ingreso o actualizar datos de los partes
	Borra los campos
	Regresa pantalla anterior
	Mostramos un reporte de todos del parte detallado(ver grafico 16)

Elaborado por: Marcelo Orozco

Tabla N° 26. Botones de navegación.

	Elimina el parte.
	Imprime todos los partes (ver gráfico 16)

Elaborado por: Marcelo Orozco



CUERPO DE BOMBEROS DEL CANTÓN GUANO



REPORTE

Guano, 18 de febrero de 2020

Listado de Partes

Nro	Fecha	H. Aviso	Observaciones	Lugar	Reportado por
100	2019-04-16	05:05	Accidente en carretera se perdio pista	Av Leopoldo Freire	Carlos garcia
121	2019-05-01	18:23	Accidente	Colombi y Carabobo	Carlos Castro
0	2019-05-01	22:49	Inundacion	Calle principal	Carlos Castro
121	2019-05-06	06:17	ninguno		Julio Juanes
0	2019-05-14	05:21	Sin Novedades	fhsadhfasdfasdf	MARIA

Firma

Gráfico N° 39 Reporte de Partes

Elaborado por: Marcelo Orozco

➤ **Registrar nueva solicitud**

Mediante esta opción que se encuentra en el menú superior se puede ver el listado de solicitudes de emergencias, ingresadas las cuales se pueden revisar en detalle, eliminar o ingresar una nueva.



Gráfico N° 40. Registro de nueva solicitud

Elaborado por: Marcelo Orozco

Tabla N° 27. Operaciones

	<p>Muestra con más detalle la solicitud</p>
---	---

Elaborado por: Marcelo Orozco



Gráfico N° 41. Datos del Parte

Elaborado por: Marcelo Orozco

Tabla N° 28. Botones de navegación.

	Permite corregir errores de ingreso o actualizar datos de los partes
	Borra los campos
	Regresa pantalla anterior

Elaborado por: Marcelo Orozco

ANEXO 2.

MANUAL DEL TECNICO.

MANUAL DEL TECNICO



SITIO WEB

CUERPO DE BOMBEROS DEL
CANTON GUANO

WEBOMBEROS

VERSIÓN 1.0

REALIZADO POR:

VICTOR MARCELO OROZCO FREIRE

Derechos Reservados 2020

Contenido

1	REQUERIMIENTO DEL SISTEMA.	84
2	CREAR CONEXIÓN CON LA BASE DE DATOS	84
2.1	CONEXION	84
2.2	DESCONEXION	85
3	OPERACIONES CON LA BASE DE DATOS	85
3.1	GUARDAR	85
3.2	BORRAR	86
3.3	MOSTRAR	86
3.4	ORDENAR REGISTROS	87
3.5	GUARDA UN REGISTRO EN LA TABLA ESPECIFICADA	87
3.6	GUARDA UN REGISTRO EN LA TABLA ESPECIFICADA	89
3.7	BORRA UN REGISTRO EN LA TABLA ESPECIFICADA	90
3.8	DEVUELVE UN REGISTRO EN LA TABLA ESPECIFICADA SEGÚN EL VALOR	91
3.9	DEVUELVE TODOS REGISTROS DE LA TABLA ESPECIFICADA	91
3.10	DEVUELVE TODOS REGISTROS DE LA TABLA ESPECIFICADA POR CAMPO	92
4	CREACION DE UN PDF	92

1 REQUERIMIENTO DEL SISTEMA.

Software utilizado para la creación de la aplicación:

- PHP 4.7.4
- MYSQL 5.1
- XAMPP 7.2
- NetBeans IDE 8.2

2 CREAR CONEXIÓN CON LA BASE DE DATOS

El archivo de conexión se encuentra en el directorio php y se llama **DBLINK.PHP**.

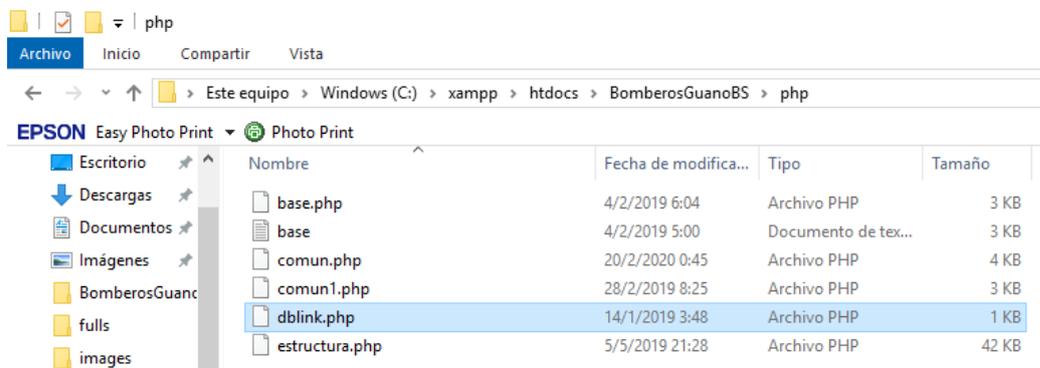


Gráfico N° 42 Carpeta y archivo de conexión

Fuente: Marcelo Orozco

2.1 CONEXION

public static function linkStart()

```
{
//
if ( null == self::$link )
{
try
{
self::$link = new PDO(
"mysql:host=".self::$servidorMysql.";dbname=".self::$nombreBase,
self::$nombreUsuario, self::$claveUsuario);
}
catch(PDOException $e)
{
```

```

        die($e->getMessage());
    }
}
return self::$link;
}

```

2.2 DESCONEXION

```

{
    public static function linkEnd()
    {
        self::$link = null;
    }
}

```

3 OPERACIONES CON LA BASE DE DATOS

El archivo de operaciones con la base de datos se llama ESTRUCTURA.PHP el cual contiene el código fuente de las funcionalidades del sistema.

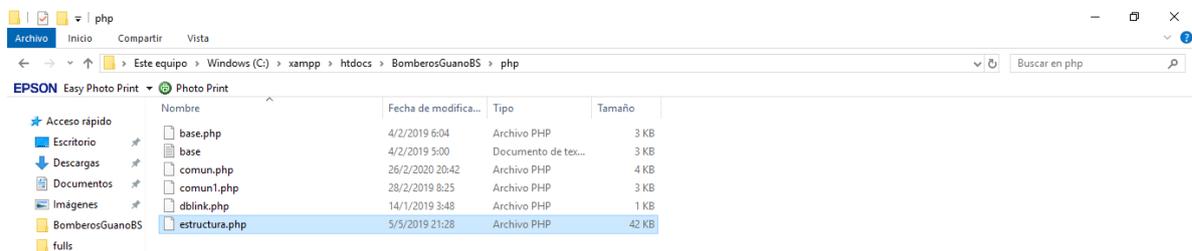


Gráfico N° 43 Dirección de las operaciones

Fuente: Marcelo Orozco

3.1 GUARDAR

```

public function Guardar()
{
    $listaValor = array($this->id,$this->Causa,$this->Origen,$this->IdTipoEmergencia,$this->IdParte,$this->estado);
    GuardarRegistro($this->tabla,$this->pk,$this->listaCampos,$listaValor);
}

```

3.2 BORRAR

```
public function Borrar()
{
    BorrarRegistro($this->tabla,$this->pk,$this->idDetalle);
}

public function DevuelvePorId($valor)
{
    $registros = DevuelveRegistroPorId($this->tabla,$this->pk,$valor);
    foreach ($registros as $filas)
    {
        $this->id = $filas['idDetalle'];
        $this->Causa = $filas['Causa'];
        $this->Origen = $filas['Origen'] ;
        $this->IdTipoEmergencia = $filas['IdTipoEmergencia'];
        $this->IdParte = $filas['idParte'];
        $this->estado = $filas['estado'];
    }
    return $this;
}
```

3.3 MOSTRAR

```
public function DevuelveTodo()
{
    $lista[] = array();
    $registros = DevuelveTodoRegistro($this->tabla);
    $c=0;
    foreach ($registros as $filas)
    {
        $tmp = new Parte();
```

```

$tmp->id = $filas['idDetalle'];
$tmp->Causa = $filas['Causa'];
$tmp->Origen = $filas['Origen'] ;
$tmp->IdTipoEmergencia = $filas['IdTipoEmergencia'];
$tmp->IdParte = $filas['idParte'];
$tmp->estado = $filas['estado'];
$lista[$c] = $tmp;
$c++;
}
return $lista;
}

```

3.4 ORDENAR REGISTROS

function DevuelveOrdenadoUltimoRegistro(\$tabla,\$campoPk)

```

{
$linkBD = dbLink::linkStart();
$SentenciaSql = "select * from $tabla order by $campoPk DESC";
$filas = $linkBD->query($SentenciaSql);
dbLink::linkEnd();
return $filas;
}

```

3.5 GUARDA UN REGISTRO EN LA TABLA ESPECIFICADA

function GuardarRegistro(\$tabla,\$campoPk,\$listaCampo,\$listaValor)

```

{
$linkBD = dbLink::linkStart();
$SentenciaSql = "INSERT INTO $tabla (";
$pos = 0;
foreach($listaCampo as $campo)
{

```

```

        if ($pos != 0)
            $SentenciaSql .= ",";
        $SentenciaSql .= $campo;
        $pos++;
    }
    $SentenciaSql .= ") values(NULL";

    $pos = 0;
    foreach($listaValor as $valor)
    {
        if ($pos != 0)
            $SentenciaSql .= ",".$valor.""";
        $pos++;
    }
    $SentenciaSql .= ")";
    $sqlExiste = "select * from $tabla where $campoPk = '$listaValor[0]'";

    foreach ($linkBD->query($sqlExiste) as $filas)
    {
        $SentenciaSql= "update $tabla set ";
        $pos=0;
        foreach($listaCampo as $campo)
        {
            if ($pos != 0)
            {
                $SentenciaSql .= $campo." = ".$listaValor[$pos].""";

                if ($pos != count($listaValor) - 1)
                    $SentenciaSql .= ", ";
            }
            $pos++;
        }
    }

```

```

        $SentenciaSql .= "where $campoPk = '$listaValor[0]'";

    }

    $linkBD->exec($SentenciaSql);
    dbLink::linkEnd();
}

```

3.6 GUARDA UN REGISTRO EN LA TABLA ESPECIFICADA

function GuardarRegistro2(\$tabla,\$campoPk1,\$campoPk2,\$listaCampo,\$listaValor)

```

{
    $linkBD = dbLink::linkStart();
    $SentenciaSql = "INSERT INTO $tabla (";
    $pos = 0;
    foreach($listaCampo as $campo)
    {
        if ($pos != 0)
            $SentenciaSql .= ",";
        $SentenciaSql .= $campo;
        $pos++;
    }
    $SentenciaSql .= ") values (";

    $pos = 0;
    foreach($listaValor as $valor)
    {
        if ($pos != 0)
            $SentenciaSql .= ",";

        $SentenciaSql .= "".$valor.""";
    }
}

```

```

        $pos++;
    }
    $SentenciaSql .= " ";
    $sqlExiste = "select * from $tabla where $campoPk1 = '$listaValor[0]'
&& $campoPk2 = '$listaValor[1]' ";
    echo '<br>'.$sqlExiste.'<br>';
    foreach ($linkBD->query($sqlExiste) as $filas)
    {
        $SentenciaSql= "update $tabla set ";
        $pos=1;
        foreach($listaCampo as $campo)
        {
            if ($pos != 1)
            {
                $SentenciaSql .= $campo." = '$listaValor[$pos]'. " ";
            }
            if ($pos != count($listaValor) - 1)
                $SentenciaSql .= ", ";
        }
        $pos++;
    }
    $SentenciaSql .= "where $campoPk1 = '$listaValor[0]' &&
$campoPk2 = '$listaValor[1]' ";
    }
    echo '<br>'.$SentenciaSql;
    $linkBD->exec($SentenciaSql);
    dbLink::linkEnd();
}

```

3.7 BORRA UN REGISTRO EN LA TABLA ESPECIFICADA

```
function BorrarRegistro($tabla,$campoPk,$valor)
```

```

{
$linkBD = dbLink::linkStart();
$SentenciaSql = "UPDATE $tabla SET estado = 0 where $campoPk =
'$valor'";
//$SentenciaSql = "DELETE FROM $tabla where $campoPk = '$valor'";
//echo $SentenciaSql;
$linkBD->exec($SentenciaSql);
dbLink::linkEnd();
}

```

3.8 DEVUELVE UN REGISTRO EN LA TABLA ESPECIFICADA SEGÚN EL VALOR

function DevuelveRegistroPorId(\$tabla,\$campoPk,\$valor)

```

{
$linkBD = dbLink::linkStart();
$SentenciaSql = "select * from $tabla where $campoPk = '$valor'";
$filas = $linkBD->query($SentenciaSql);
dbLink::linkEnd();
return $filas;
}

```

3.9 DEVUELVE TODOS REGISTROS DE LA TABLA ESPECIFICADA

function DevuelveTodoRegistro(\$tabla)

```

{
$linkBD = dbLink::linkStart();
$SentenciaSql = "select * from $tabla where estado = 1";
//echo $SentenciaSql;
$lista = $linkBD->query($SentenciaSql);
dbLink::linkEnd();
return $lista;
}

```

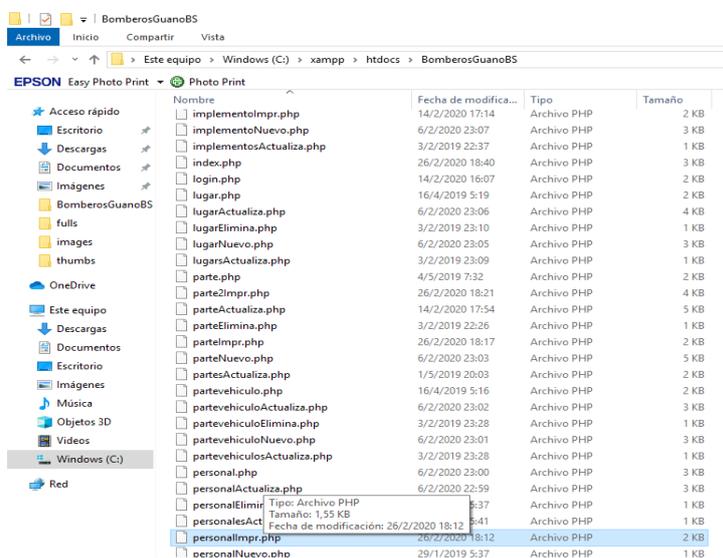
3.10 DEVUELVE TODOS REGISTROS DE LA TABLA ESPECIFICADA POR CAMPO

function DevuelveTodoRegistroPorCampo(\$tabla,\$campos,\$valores)

```
{
    $lista = null;
    $linkBD = dbLink::linkStart();
    $SentenciaSql = "select * from $tabla where estado = 1 ";
    $c=0;
    foreach($campos as $campo)
    {
        $SentenciaSql .= " and ".$campo." = ".$valores[$c] ;
        $c++;
    }
    //echo $SentenciaSql;
    $lista = $linkBD->query($SentenciaSql);
    dbLink::linkEnd();
    return $lista;
}
```

4 CREACION DE UN PDF

En los archivos que contienen personalmpr.php muestra cómo se crea un PDF para presentar las diferentes librerías tanto de creación y generación.



Librería

```
use Spipu\Html2Pdf\Html2Pdf;
```

Generar PDF

```
$mipdf=new HTML2PDF('P','A4','es','true','UTF-8');
```

```
$mipdf->writeHTML($texto);
```

Librería para generar el PDF

```
$mipdf->Output('certificado.pdf');
```