



**TECNOLÓGICO DE ESTUDIOS
SUPERIORES DE COACALCO**

**INFORME TÉCNICO DE ESTANCIAS
DENTRO DE LA EMPRESA SIENNOIS**

ESTANCIAS

**PARA OBTENER EL TITULO DE:
INGENIERO EN TECNOLOGÍAS DE LA
INFORMACIÓN Y COMUNICACIONES
CON LA ESPECIALIDAD EN:
TECNOLOGIAS MOVÍLES**

PRESENTA:

MANCERA NIETO EDUARDO

Matrícula No. 201820303

ASESOR:

M.C.C JOSÉ ALEJANDRO PINEDA AGUILLÓN

COACALCO DE BERRIOZABAL, MÉX. NOV., 2024

AGRADECIMIENTOS

A mis padres, Luz María Nieto Olvera y Jaime Mancera Galindo

La vida me dio la fortuna de tener unos excelentes padres. Quiero agradecer todas sus enseñanzas, regaños, consejos y aliento en el momento adecuado, ya que, gracias a todas sus lecciones, soy la persona que soy hoy en día. A diario trato de mostrar mi mejor versión como persona y como profesionista porque eso es algo que me enseñaron ustedes, que a pesar de que la situación no es la mejor, siempre debemos ver el lado positivo y seguir adelante para sobrepasar todas las adversidades.

Se que esto solo es uno de mis pasos como estudiante, pero lo que más me llena de orgullo es que en todo momento han estado ustedes para guiarme por el camino correcto y darme lo mejor de cada uno.

A mis profesores

A lo largo de la carrera, tuve excelentes profesores que su único objetivo fue el poder dar lo mejor de sí mismos para ayudar a la formación de profesionales que sepan como explotar sus habilidades en el campo laboral. A veces podemos diferir con sus opiniones, pero siempre van a querer lo mejor para cada uno de sus estudiantes.

RESUMEN

El objetivo de este informe es el poder demostrar el conocimiento adquirido durante un año laboral dentro de la empresa “SIENNOIS” desarrollando el proyecto “Dimueble” el cual es un registro a nivel nacional orientado principalmente a estudiantes universitarios, profesionales del área de mobiliario y una categoría especial de “Elektra”. Se podrán observar las diferentes etapas por las cuales pasa un proyecto para ser puesto en producción.

Para el primer capítulo, se podrán encontrar los aspectos generales e información detallada de la empresa en la cual se ha desarrollado este proyecto, esto con el fin tener un mejor conocimiento sobre la empresa y poder comprender de mejor manera su misión, visión y filosofía. Dentro del segundo capítulo, se encontrará el marco teórico el cual ayudará a comprender los diversos conceptos que se abordaran a lo largo de este documento, como lo pueden ser tecnicismos, lenguajes de programación, plataformas de desarrollo y demás.

En el tercer capítulo, se podrá conocer más a fondo como es que fue desarrollado este proyecto partiendo desde las plantillas echas por el equipo de front-end y poder hacer la lógica necesaria para que toda la página pueda ser administrable y sea adecuada para todos los participantes. Para el cuarto capítulo, se podrá ver el resultado final del sitio, después de diversos cambios y funcionalidades que fueron añadidas para una mejor experiencia de usuario y facilitar el registro de cada participante. Como última parte se tienen los apartados para las competencias desarrolladas, aprendizaje que se haya tenido y fuentes de información las cuales sirvieron como sustento teórico para el desarrollo de este informe.

INDICE

CAPITULO I. GENERALIDADES DEL PROYECTO	12
1.1 Introducción	12
1.2 Descripción de la empresa	13
1.2.1 Giro.....	13
1.2.2 Servicios a sus clientes	13
1.2.3 Misión.....	13
1.2.4 Visión	13
1.2.5 Filosofía.....	13
1.2.6 Ubicación.....	13
1.3 Descripción de la participación en proyectos.....	14
1.4 Justificación	14
1.5 OBJETIVOS	15
1.5.1 Objetivo general	15
1.5.2 Objetivos específicos.....	15
1.6 Alcances y limitaciones	15
1.6.1 Alcances	15
1.6.2 Limitaciones	16
CAPITULO II. MARCO TEÓRICO	17
2.1 Historia del Internet	17
2.2 ¿Qué es la World Wide Web?	18
2.3 Lenguajes de programación	19
2.3.1 JavaScript	20
2.3.2 Python.....	20
2.3.3 Java.....	21
2.3.4 C#	22

2.3.5 PHP	23
2.4 Intérprete y compilador ¿Qué son?	24
2.4.1 ¿Qué es un intérprete?	24
2.4.2 ¿Qué es un compilador?	24
2.4.3 Diferencias entre un intérprete y un compilador	25
2.5 Paradigmas de programación	25
2.5.1 Programación declarativa	25
2.5.2 Programación imperativa.....	26
2.5.3 Programación lógica	26
2.5.4 Programación estructurada	26
2.6 Bases de datos	26
2.6.1 ¿Qué es una base de datos?	26
2.6.2 Sistemas gestores de bases de datos	27
CAPITULO III DESARROLLO	29
3.1 Desarrollo de base de datos.....	29
3.1.1 Tabla de campus	30
3.1.2 Tabla de carreras.....	30
3.1.3 Tabla de contenidos	30
3.1.4 Tabla de administradores.....	31
3.1.5 Tabla para los registros individuales	32
3.1.6 Tabla para primera fase de calificaciones	34
3.1.7 Tabla para segunda fase de evaluaciones	34
3.2 Vistas.....	35
3.2.1 Home	35
3.2.2 Tipo de registro.....	38

3.2.3 Registros de cuentas	39
3.2.4 Login para usuarios	41
3.2.5 Datos del participante	42
3.2.6 Registro de documentación	44
3.2.7 Datos universitarios	46
3.2.8 Información profesional	47
3.2.9 Documentación de proyectos	47
3.2.10 Pantalla de agradecimiento.....	49
3.3 Controladores	50
3.3.1 Site.....	50
3.3.2 Forms.....	55
3.4 Modelos.....	63
CAPITULO IV RESULTADOS	70
Bibliografía	85

INDICE DE IMÁGENES

Ilustración 1 Ubicación geográfica de las oficinas de la empresa	14
Ilustración 2 Mapa en semi- producción del Internet, febrero 1982.....	18
Ilustración 3 Tablas de nuestra base de datos	29
Ilustración 4 Etiqueta head de nuestra página de inicio.....	36
Ilustración 5 Primeras etiquetas de nuestro template de head	36
Ilustración 6 Colores administrables para el sitio	37
Ilustración 7 Títulos para nuestro home	37
Ilustración 8 Categorías habilitadas para los registros.....	38
Ilustración 9 Carrusel para los aliados estratégicos dentro de la página.....	38
Ilustración 10 Preguntas frecuentes para los participantes	38
Ilustración 11 Título para el tipo de registro.....	39
Ilustración 12 Botones para la categoría de estudiantes	39
Ilustración 13 Botones para la categoría profesional.....	39
Ilustración 14 Botones para la categoría Elektra	39
Ilustración 15 Título de la página para crear la cuenta	40
Ilustración 16 Apertura de formulario para el participante.....	40
Ilustración 17 Campos para la creación de la cuenta.....	40
Ilustración 18 Asignación de clave para captcha.....	41
Ilustración 19 Scripts para validar cuenta y recaptcha.....	41
Ilustración 20 Campos para el inicio de sesión para los participantes.....	42
Ilustración 21 Secciones del registro	42
Ilustración 22 Primera parte de la información básica del participante.....	43
Ilustración 23 Campos para conocer la ubicación del participante	43
Ilustración 24 Consumo del API.....	44
Ilustración 25 Documentación para el participante individual	44
Ilustración 26 Documentos para el primer integrante del equipo	45
Ilustración 27 Validador para el tipo de archivo.....	45
Ilustración 28 Script para validar el nombre y mostrar previsualización del archivo	46
Ilustración 29 Formulario para los datos universitarios del participante	46

Ilustración 30 Script para datos universitarios.....	47
Ilustración 31 Formulario para la empresa de la categoría profesional	47
Ilustración 32 Campos para los renders del proyecto 1	48
Ilustración 33 Campo para la memoria conceptual de proyecto 1.....	49
Ilustración 34 Validación para archivos pdf y peso máximo.....	49
Ilustración 35 Pantalla de agradecimiento	50
Ilustración 36 Modelo para mostrar la información recuperada de la base de datos	50
Ilustración 37 Función index de nuestra página	50
Ilustración 38 Información de nuestro index y recepción del formulario.....	51
Ilustración 39 Validación del formulario y manejo de datos.....	51
Ilustración 40 Acceso al portal del registro o negar acceso	52
Ilustración 41 Cierre de sesión del participante.....	52
Ilustración 42 Información para mostrar en la vista del registro de los estudiantes	52
Ilustración 43 Comprobar campos ingresados.....	53
Ilustración 44 Validación errónea de los campos del formulario	53
Ilustración 45 Creación de arreglo para la información del estudiante.....	54
Ilustración 46 Envío de correo al usuario	54
Ilustración 47 Redirección al usuario	55
Ilustración 48 Confirmación de la cuenta	55
Ilustración 49 Información para llenar nuestro primer formulario	56
Ilustración 50 Nombre del estado	56
Ilustración 51 Validación de los campos del formulario	56
Ilustración 52 Actualizado de los datos del participante	57
Ilustración 53 Datos del equipo del participante	57
Ilustración 54 Carga de documentos escolares	58
Ilustración 55 Validación de archivos profesionales	59
Ilustración 56 Documento de identificación para categoría Elektra	59
Ilustración 57 Validar formulario	60
Ilustración 58 Validar los demás campos del formulario.	60
Ilustración 59 Actualización de los datos universitarios.....	61
Ilustración 60 Actualizar los datos de la empresa del participante	61

Ilustración 61 Cargar los archivos de los proyectos	62
Ilustración 62 Creando arreglo para enviarlo a la base de datos	62
Ilustración 63 Envío de correo al usuario confirmando su registro completo	63
Ilustración 64 Actualizar datos del participante	63
Ilustración 65 Función para traer un registro de ci_content	64
Ilustración 66 Consulta para el template de la página	65
Ilustración 67 Consulta para los participantes	66
Ilustración 68 Consulta para los integrantes del equipo	67
Ilustración 69 Consulta para conocer estados mediante un Id.....	68
Ilustración 70 Función para actualizar los participantes.....	68
Ilustración 71 Actualizar información del participante	68
Ilustración 72 Pantalla de bienvenida para el participante	70
Ilustración 73 Tipo de registro para el participante	71
Ilustración 74 Formulario para crear cuenta	71
Ilustración 75 Generar error al crear cuenta	72
Ilustración 76 Creación de cuenta completada	72
Ilustración 77 Correo para activación de cuenta.....	73
Ilustración 78 Iniciar sesión dentro del portal	73
Ilustración 79 Primer formulario para empezar con el registro de los proyectos	74
Ilustración 80 Información básica del usuario	74
Ilustración 81 Sección para cargar documentación oficial del participante	75
Ilustración 82 Carga de archivo con nombre erróneo.....	75
Ilustración 83 Previsualización de los archivos cargados.....	76
Ilustración 84 Universidad registrada en la base de datos	77
Ilustración 85 Universidad sin registro en el concurso.....	77
Ilustración 86 Carga de los proyectos del participante.....	78
Ilustración 87 Registro completo.....	78
Ilustración 88 Ingreso de información de la empresa	79
Ilustración 89 Documentación del participante	79
Ilustración 90 Reestablecer contraseña del registro	80
Ilustración 91 Correo de restablecimiento	80

Ilustración 92 Nueva contraseña.....	81
Ilustración 93 Ingreso con la nueva contraseña	81

INDICE DE TABLAS

Tabla 1.- Diferencias entre intérprete y compilador	25
Tabla 2 Campus para las universidades	30
Tabla 3 Carreras.....	30
Tabla 4 ci_content.....	31
Tabla 5 Tabla para los administradores del sitio	31
Tabla 6 Registro para los participantes individuales	32
Tabla 7 Primer fase para las calificaciones de los participantes	34
Tabla 8 Evaluaciones de la segunda fase del concurso.....	34

CAPITULO I. GENERALIDADES DEL PROYECTO

1.1 Introducción

Actualmente las diversas tecnologías de la información, están teniendo un impacto importante en el día a día dentro de las actividades cotidianas de la sociedad, como lo pueden ser relaciones laborales, entretenimiento, divulgación científica, etc. Es por eso que cada vez, son más las empresas que buscan un medio masivo de comunicación para poder llegar a más público y sin hacer un esfuerzo extra o costos demasiados alto con relación a un beneficio general dentro de la empresa.

Dentro del área de exposiciones, congresos y eventos masivos, el principal objetivo de los organizadores, es el poder llegar a más gente año con año con el único objetivo de que su evento sea más grande. Una manera de llegar a mucha más gente hoy en día es con el desarrollo de una página web, ya que, en la actualidad, el acceso a internet es un servicio que la gran mayoría de personas tienen y con un desarrollo optimo, la página podrá ser vista alrededor de todo el mundo.

La asociación “AFAMJAL” (Asociación de Fabricantes de Muebles de Jalisco, A.C.) realiza un concurso a nivel nacional llamado “Dimueble”. El objetivo de este concurso es el poder conocer las diversas ideas de su público en las categorías: estudiantil, profesional y Elektra. Para los organizadores del concurso es muy importante el poder contar con un registro transparente y sobre todo que sea fácil de usar para todos los participantes. Esto con el fin de que año con año la cantidad de registros que se tengan sean mayores, pero sin dejar a un lado la seguridad con la cual se puede garantizar una mejor calificación impartida por los jueces del concurso.

Se desarrollará un portal, el cual ayudará al participante a tener un registro más completo y seguro en comparación al método anterior, contando con diversos métodos de seguridad y autenticación de usuarios, esto evitará que personas ajenas al público objetivo puedan entorpecer el proceso de registro para las diversas categorías presentada y de igual manera tener una etapa de calificación mucho más justa para todos los participantes.

1.2 Descripción de la empresa

1.2.1 Giro

Desarrollo de eCommerce, páginas web y marketing digital

1.2.2 Servicios a sus clientes

Dentro de la empresa “Sienois” se ofrecen 4 principales servicios a todos sus clientes:

Marketing digital: Ayuda a tu marca a dar un impulso con medios y tecnologías, haciendo uso de campañas digitales en Google ADS.

Optimización de SEO: Poder ganar más terreno de una manera orgánica ante los diversos navegadores y poder ser mostrado a más clientes potenciales con un problema u objetivo en general.

Desarrollo web: Se crean sitios webs, los cuales se adaptan a las diversas necesidades de los clientes, sin dejar de lado la UX, la cual es un aspecto importante dentro de la navegación del sitio.

Desarrollo de eCommerce: El poder implementar un sistema de manera correcta que se adapte a las necesidades del cliente, hará que su negocio llegue a más lados y pueda servir de una mejor manera.

1.2.3 Misión

Desarrollar herramientas digitales para nuestros clientes, ofreciéndoles la implementación de plataformas productivas y que respondan a cada necesidad.

1.2.4 Visión

Ser la empresa de desarrollo digital líder y la mejor opción para las pequeñas y medianas empresas en su camino al éxito.

1.2.5 Filosofía

Trabajar con profesionalismo y pasión, esforzándonos para entregar siempre resultados que superen las expectativas de cada uno de nuestros clientes.

1.2.6 Ubicación

Las oficinas de la empresa se encuentran ubicadas en la Avenida 16 de septiembre 101, colonia Necapa, código postal 54800, Cuautitlán, México

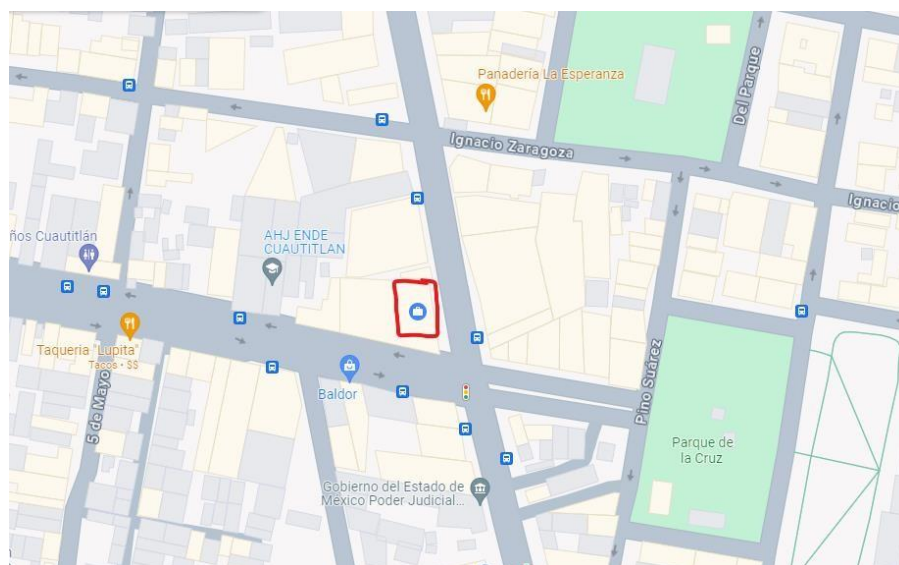


Ilustración 1 Ubicación geográfica de las oficinas de la empresa

1.3 Descripción de la participación en proyectos

La principal actividad que se tuvo en el desarrollo de este proyecto, fue la parte back-end, lo cual implica el poder crear la base de datos con las diferentes tablas que se va a usar, desarrollar la lógica necesaria para que cada uno de los participantes pueda generar su registro de manera exitosa, poder crear reportes con información puntual de cada uno de los participantes y el desarrollar un sistema para la obtención de todas las calificaciones para los registros que hayan sido parte de la fase final del concurso.

1.4 Justificación

Con el fin de poder optimizar el proceso de registro para los diferentes participantes del concurso “Dimueble”, la asociación “AFAMJAL” acudió a nosotros para el desarrollo de un sitio web el cual tendrá como objetivo ser más amigable con el participante y tener una mejor etapa de calificaciones. De igual manera se buscó mejorar la experiencia de usuario durante el registro y evitar que existieran problemas o confusiones a la hora de realizar su respectivo registro.

El desarrollo de esta página no solo ayudará a los participantes a tener un registro más cómodo y amigable, sino que también dará una mejor reputación a los organizadores del

concurso al demostrar una innovación y eficacia. La implementación de este sitio web es una inversión estratégica por parte de los organizadores el cual podrá traer diferentes beneficios como administrativos, estadísticos e incluso llegar a un mayor número de participantes a nivel nacional.

1.5 OBJETIVOS

1.5.1 Objetivo general

Desarrollar un sitio web funcional e intuitivo para el registro de los concursantes dentro del concurso “Dimueble”

1.5.2 Objetivos específicos

Desarrollar una base de datos para poder almacenar y gestionar a los participantes del concurso.

Implementar diferentes métodos para mostrar información a los administradores del sitio con el objetivo de tener un proceso de calificación más claro y justo.

Generar métodos para las calificaciones de los participantes para la primera y segunda etapa de evaluaciones

Crear interfaces amigables con el concursante con el fin de evitar confusión a la hora de cargar su información correspondiente

1.6 Alcances y limitaciones

1.6.1 Alcances

Poder desarrollar un sistema de autenticación de usuario para evitar que cualquier persona pueda crear un registro

Contar con un sistema administrador en el cual se mostrarán todos los concursantes registrados y poder tener etapas de evaluación más justas

Integrar análisis y reportes dentro del mismo administrador para que los jueces puedan conocer más a fondo las estadísticas de los participantes

Contar con un diseño responsivo, esto para garantizar que el usuario pueda acceder desde su computadora, tablet o celular.

1.6.2 Limitaciones

Cambios de último momento por parte del cliente pueden provocar errores en el sitio cuando ya se encuentra en producción, esto puede afectar la experiencia del usuario de forma negativa.

Debido a que el sitio está diseñado exclusivamente para participantes mexicanos, no se ha incluido soporte para diversos idiomas. En caso de que haya participantes que no hablen español, tendrán que adaptarse al idioma predeterminado del sitio

CAPITULO II. MARCO TEÓRICO

2.1 Historia del Internet

El internet no nació siendo una red de computadoras con el fin de conectar a la gente dentro de un mismo círculo. Si no que se creó en el año 1959 siendo un proyecto para el Departamento de Defensa de los Estados Unidos de América, con el fin de poder comunicar las zonas militares y académicas del país. Su principal objetivo fue el desarrollar un sistema descentralizado para un correcto flujo de información.

“La historia del internet empezó en la gran etapa de la ciencia norteamericana en los años 50s, dentro de la gran batalla que se estaba disputando entre Estados Unidos y la Unión Soviética en la llama “Guerra fría”. Tras el lanzamiento del satélite “Sputnik” en el año de 1957, por parte de la administración de los soviéticos, el presidente en turno Dwight Eisenhower ordenó crear una agencia para realizar estudios acerca de material y comunicaciones bélicas” (Sain 2015).

Como primeros pasos para el desarrollo de este proyecto para el Departamento de Defensa, fue asignar un responsable de llevar este proyecto, el elegido fue Jack Licklider, quien fue parte de la creación de la ARPA (Advanced Research Projects Agency), la misión principal para Jack fue mejorar el uso de la tecnología de ejército para la creación de una red para las computadoras militares, esto con el fin de tener la capacidad de publicar algunos trabajos científicos.

“En el transcurso del año 1965 se logró conectar dos ordenadores, uno de ellos se encontraba en Massachusetts y el otro en California, esto se logró mediante una línea telefónica, su funcionalidad no fue tan alta como se esperaba. Sin embargo, para el año 1969, Jack junto a Michael Eli, se logró conectar un ordenador de la Universidad de California con otro del Instituto de Investigación de Stanford, para que tiempo después se unieran cuatro universidades más y con esto fue que nació la ARPANET. Usando ideas desarrolladas en ALOHAnet, la ARPANET fue inaugurada en 1972 y tuvo un crecimiento acelerado hasta 1981. Tuvo un total de 213 host con un promedio de uno nuevo cada 20 días” (Telefónica 2023) .

La primera conexión ARPANET fuera de Estados Unidos se realizó en NORSAR, Noruega en el año de 1973. Para 1979 Tom Truscott y Jim Ellis, hicieron una propuesta para hacer uso de scripts simples en Bourne Shell, esto para poder transferir noticias y mensajes entre la universidad de Duke y su vecina la Universidad de Carolina del Norte. Posteriormente que el software salió para el dominio público, la red UUCP tuvo un desarrollo notable ya que sería usado para noticias, después de este crecimiento, se cambió de nombre a UUCPnet. Antes de 1983, la cantidad de host en UUPC era de 550 pero tan solo para el siguiente año, habían cerrado con la cantidad de 940 usuarios.

Para ese entonces, la ASPANET empezó a fusionarse con NSFNet, lo cual daría pie al termino “Internet”, se puede definir “Internet” como cualquier red que use el protocolo TCP/IP. Hasta esos momentos, “internet” e “internetwork” se habían usado indistintamente y “protocolo de internet” se refería a todos sistemas de redes como lo fue Xerox Network Services.

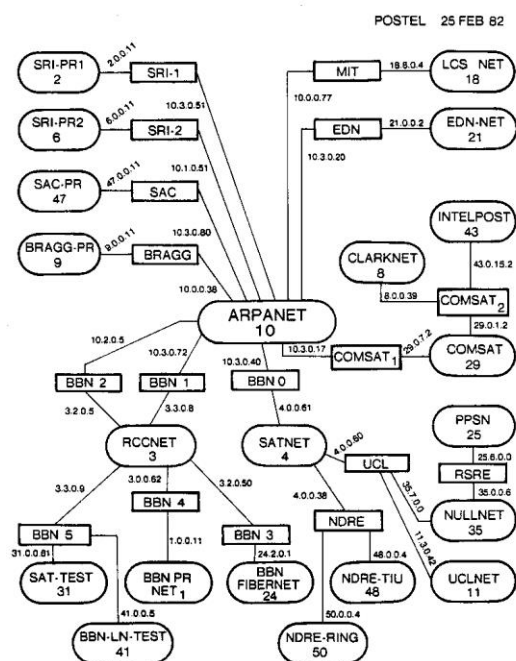


Ilustración 2 Mapa en semi- producción del Internet, febrero 1982

2.2 ¿Qué es la World Wide Web?

Desde la creación de la World Wide Web (WWW) en el año de 1991, se han tenido una infinidad de sitios webs con objetivos totalmente distintos como lo pueden ser:

entretenimiento, informativos, divulgación, entre otros. En la actualidad para la gran mayoría de los usuarios en la web, los términos “Internet” y “World Wide Web” son un claro ejemplo de sinónimos, pero es importante recalcar que gracias al uso del internet se pudo constituir la WWW. Para el científico Tim Berners-Lee “su principal fuente de inspiración para la creación de la World Wide Web, fueron diversos científicos de todo el mundo reuniéndose en los edificios del CERN (Centro Europeo de Física Nuclear), esto para debatir sobre diversos temas” (Carles Bellver 1995).

Para los primeros años de la WWW se tenía pensado como una gran red de difusión para información científica de investigadores ubicados de manera dispersa, para ser más concretos, era dirigida a la comunidad del CERN, esto con el fin de que todos los recursos electrónicos que se tuvieran en los diferentes ordenadores pudieran ser accesibles en todo momento y sin la necesidad de tener que aprender a usar nuevas tecnologías.

Y es con el paso del tiempo, la WWW fue evolucionando poco a poco hasta llegar a ser lo que hoy en día conocemos, una red con infinitud de sitios los cuales puedes ser usados por gente de todo el mundo para diversos objetivos sin la necesidad de pertenecer a un grupo selecto de personas.

2.3 Lenguajes de programación

“En términos generales, un lenguaje de programación es una poderosa herramienta la cual permite a un experto el poder desarrollar software como pueden ser páginas en internet, programas de escritorio o aplicaciones móviles. Los lenguajes de programación son encargados de definir o administrar el comportamiento que llegan a tener los dispositivos lógicos y físicos en una computadora o dispositivos móviles. Lo que se menciona se puede lograr creando e implementando algoritmos los cuales se utilizan para entablar una comunicación humana – computadora” (Ceballos Sierra 2009).

Un lenguaje de programación está formado por una serie de símbolos, reglas de sintaxis y semántica, las cuales son la estructura principal para el lenguaje y de pueden dar un significado a todos sus elementos y expresiones. La programación es el proceso de análisis, diseño, implementación, prueba y depuración de un algoritmo, el lenguaje se encarga de poder hacer la interpretación y compilación de este mismo, generando un código fuente el cual será ejecutado en el dispositivo de desarrollo.

2.3.1 JavaScript

“JavaScript es un lenguaje de programación o de secuencias de comandos el cual permite interpretar funciones complejas en páginas web, cada vez que alguien entra a una página y se muestra algún tipo de información estática, actualización de contenido, mapas interactivos, gráficos en 2D o 3D, reproductores de video, entre otros, es altamente probable que el responsable de esto sea este lenguaje. Es la tercera capa dentro de las tecnologías web, teniendo como primer nivel a HTML, después a CSS y por último a JavaScript” (¿Qué es JavaScript? 2023).

Empezando la década de los años 90, la gran mayoría de los usuarios que se encontraban navegando por Internet hacían uso de módems con una velocidad máxima de 28.8 kbps. El problema viene cuando se desarrollaron las primera páginas web y estas comenzaron a implementar formularios con un procesamiento más complejo. Con aplicaciones web de mayor complejidad y una velocidad de navegación inadecuada para los requerimientos de las páginas, surgió la necesidad de crear un lenguaje que se pudiera ejecutar en el navegador del cliente, esto con el fin de que si en algún momento el usuario no hacía un llenado correcto del formulario no se le haría esperar mucho tiempo hasta que el servidor respondiera con los errores que se detectaron.

“El programador Brendan Eich trabajaba para Netscape, pensó en que podría darle una solución a este problema haciendo uso de tecnologías que ya existían, como lo era el caso de ScriptEase. Principalmente Eich nombre el lenguaje como LiveScript, posteriormente, Netscape alcanzó una importante alianza con Sun Microsystems para el desarrollo del nuevo lenguaje de programación, justo antes del lanzamiento, Netscape decidió hacer un cambio al nombre oficial para el lenguaje, lo cambiara a JavaScript esto con fines de marketing ya que la palabra Java se encontraba de moda en todo el mundo” (Eguíluz Perez 2019).

2.3.2 Python

“Python es un lenguaje de programación el cual puede ser ocupado para diversas aplicaciones web, desarrollo de software, ciencia de datos y para machine learning. Los programadores suelen utilizar Python porque es eficiente y muy fácil de aprender, además de

que ofrece la ventaja de que se puede ejecutar en diversas plataformas, algunas de las ventajas que tiene Python sobre otros lenguajes son:

- Los desarrolladores pueden leer y comprender de mejor manera el lenguaje gracias a su sintaxis básica
- Aumenta la productividad de los programadores ya que pueden crear software con menos líneas de código.
- La alta compatibilidad de Python con otros lenguajes lo hace muy amigable con diversas plataformas” (¿Qué es Python? 2022)

Todo nació gracias a Guido Van Rossum, un programador de Países Bajos, todo comenzó en el año 1989 en el CWI (Centrum Wiskunde & Informatica), este lenguaje nació como un proyecto de ocio para que Guido se mantuviera ocupado en alguna actividad durante las navidades de ese año. El nombre del lenguaje se inspiró gracias al programa que era transmitido en la BBC llamado “Monty Python’s Flying Circus” ya que Guido era gran fan de ver el programa. Las versiones lanzadas de Python son:

Se publica la primera versión del código Python (versión 0.9.0) en el año 1991, en esta versión se hacían una gran corrección de errores, algunos tipos de datos y demás características potentes.

- Para la versión 1.0 se ha hecho un procesamiento de datos más efectivo, así como la asignación, filtrado y la reducción de los mismo.
- El 16 de octubre del 2000 fue lanzado Python 2.0, agregó una compatibilidad para caracteres Unicode y una optimización para el recorrido de listas.
- En la versión lanzada para el 3 de diciembre del 2008, Python 3.0 incluyó características como la función de impresión, soporte para la división de números y gestión de errores.

2.3.3 Java

Java es un lenguaje de programación el cual es usado para poder codificar aplicaciones web, ha sido una de las principales opciones para los programadores durante los últimos 20 años, eso se puede corroborar con la gran cantidad de aplicaciones desarrolladas con este lenguaje. Este lenguaje es multiplataforma, orientado a objetos y se encuentra

centrado a una red la cual se puede utilizar como una plataforma en sí mismo. También es un lenguaje rápido, seguro y confiable para desarrollar aplicaciones móviles, hasta software empresarial.

Para el año 1995, Sun Microsystems lanzó Java por primera vez como un lenguaje de programación y también como una plataforma informática. Desde sus primeros años, ha crecido de gran manera permitiéndole ser un gran impulso para miles de plataformas, servicios y aplicaciones. En 2010, la marca Oracle compraría Java a Sun Microsystems

Java tiene dos propiedades las cuales pueden determinar que tareas se van a poder realizar con este lenguaje:

Es un lenguaje de Programación orientada a objetos (POO): “Toda la interacción que se tiene con el lenguaje es a través de objetos, estas entidades se describen en el código y se les enseña a poder interactuar entre sí. Como resultado, tenemos un programa de estilo orientado a objetos el cual consta de bloques separados que son fácilmente extensible y escalables.” (Tripleten 2024)

Java toma lo mejor de los lenguajes compilados e interpretados: Un lenguaje de programación es la forma en la que el desarrollador se puede comunicar con el procesador y pueden ejecutar comandos y tareas. Pero el procesador no está obligado a conocer todos los lenguajes en los cuales se pueden escribir comandos. Pero esto no sucede con java ya que solo es necesario el uso de una JVM (Java Virtual Machine) para poder ejecutar el código desarrollado. La máquina virtual java proporciona la principal ventaja del lenguaje: el ser multiplataforma. En java un programador no escribe el código para los sistemas operativos, sino que lo hace para la maquina Java y él mismo adapta el código al hardware y los sistemas operativos.

2.3.4 C#

“C# es uno de los principales lenguajes de programación ya que hace parte integral de la plataforma Microsoft .NET, este lenguaje es seguro y completamente orientado a objetos. Este lenguaje es usado actualmente para desarrollar aplicaciones web basadas en ASP.NET, formularios de Windows y aplicaciones de escritorio basadas en WPF (Windows Presentation Foundation). De igual manera con la llega de los smartphones C# llego a ser

utilizado para desarrollar aplicaciones para teléfonos con sistema operativo de Windows.”
(Dimes 2016)

El objetivo de C# y de .NET es poder crear aplicaciones y software sólido de manera fácil. Ya que este lenguaje está diseñado para poder ser usado en el framework de .NET, ya que se basan en el mismo paradigma y esto es gracias a las siguientes características:

- C# da una sintaxis mucho más breve e intuitiva en comparación a C++.
- Hace la incorporación de diferentes elementos de diversos lenguajes que han crecido con los años.
- C# incorpora mecanismos de seguridad para que el acceso a los datos se lleve de la mejor manera posible.
- El lenguaje es fuertemente tipado ya que todas las variables y constantes tienen un tipo. Además de que todos los datos obtenidos son guardados para un uso posterior.
- La compatibilidad de C# hace referencia que su sintaxis es muy parecida a otros lenguajes como C, C++, Java, entre otros

Actualmente, este lenguaje es uno de los que más se usa a nivel mundial ya que es muy útil para dar diversas soluciones a problemas específicos que otros lenguajes no pueden hacer. Cabe mencionar que todos los productos de la actualidad de Microsoft, incluida la nube Azure, son altamente compatibles con C#.

2.3.5 PHP

“PHP es un acrónimo recursivo para “Hypertext Preprocessor”, pero originalmente era “Personal Home Page”. Este es un lenguaje interpretado libre usado originalmente para el desarrollo de aplicaciones que actúan del lado del servidor, capaces de poder generar contenido dinámico en la WWW. Es uno de los principales lenguajes que figuran para la inserción de documentos en HTML. El código es interpretado del lado del servidor gracias al módulo PHP, el cual también genera la página web para que el cliente pueda visualizarla.”
(Arias 2013)

El lenguaje ha pasado por diferentes evoluciones, actualmente puede ofrecer funcionalidades para línea de comandos y ha agregado características adicionales que ayudan

a un mejor desarrollo. Es posible hacer la instalación en la gran mayoría de sistemas operativos. Siendo un competidor directo de la tecnología ASP de Microsoft, PHP es utilizado en diversas aplicaciones como MediWiki, Facebook, WordPress, entre otras. PHP es un software de uso libre el cual se encuentra bajo la licencia de “PHP License” la cual es una licencia incompatible con la GNU General Public License esto debido a los términos de uso del lenguaje.

En la actualidad PHP puede ser ejecutado por una gran cantidad de servidores, como lo pueden ser: Apache, Xitami, Oreilly, IIS, Sambar, PWS, entre otros. Todos estos servidores, al menos los que son web, deben estar instalados en un equipo remoto el cual va a ser el encargado de recibir las peticiones y nos va a devolver la respuesta. Por esta razón no es necesario que el ordenador del cliente tenga instalados los servidores o que tenga configuración para base de datos.

2.4 Intérprete y compilador ¿Qué son?

2.4.1 ¿Qué es un intérprete?

Un intérprete es un programa el cual tiene la función de procesar el código fuente de un proyecto durante el tiempo que se encuentre en ejecución, en otras palabras, mientras el software se encuentre en ejecución el intérprete actúa como interfaz entre el proyecto y el procesador. El intérprete siempre procesa línea por línea, de este modo puede leer, analizar y preparar cada una de las secuencias para el procesador.

Para realizar el procesamiento del código fuente del software, el intérprete recurre a sus propias bibliotecas internas, en el momento que una línea de código se ha traducido de manera correcta a los correspondientes comandos legibles por la máquina, este es enviado directamente al procesador. El proceso de conversión no finaliza hasta que se ha interpretado el código en su totalidad. Puede existir alguna interrupción, en el caso de que se encuentre de manera anticipada alguna falla durante el proceso, esto ayuda a la resolución de errores ya que la línea correspondiente al fallo es identificada después del fallo.

2.4.2 ¿Qué es un compilador?

Es un programa informático el cual traduce todo el código fuente de un proyecto de software a código máquina antes de ser ejecutado, solo hasta ese momento el procesador

puede ejecutar el programa, teniendo todas las instrucciones en código. De esta manera, el procesador tiene en su poder todos los componentes necesarios para hacer la ejecución del software, hacer el procesamiento de las diversas entradas y poder generar los resultados esperados.

En muchos casos, durante el proceso de la compilación, se tiene lugar a un paso intermedio el cual es fundamental: antes de poder hacer la generación de la traducción final en código máquina, la gran mayoría de los compiladores, hacen la tarea de convertir el código fuente en un código intermedio, también conocido como código objeto, que en su gran mayoría de veces es compatible con diversas plataformas y también puede ser usado como interprete.

2.4.3 Diferencias entre un intérprete y un compilador

Tabla 1.- Diferencias entre intérprete y compilador

	Intérprete	Compilador
Momento en el cual se hace la traducción del código fuente	Durante la ejecución del software	Antes de ejecutar el software
Procedimiento de traducción	Línea por línea	Siempre en todo el código
Presentación de errores	Después de cada línea	En conjunto, después de toda la compilación
Velocidad de traducción	Alta	Baja
Eficiencia de traducción	Baja	Alta
Coste de desarrollo	Bajo	Alto
Lenguajes típicos	PHP, Perl, Python, Ruby	C, C++, Pascal

(Compilador e intérprete: definición y diferencias 2020)

2.5 Paradigmas de programación

2.5.1 Programación declarativa

La programación declarativa no determina el cómo se va a hacer el programa o aplicación, sino que su funcionamiento se ve en un nivel de abstracción más alto que el de la programación imperativa. Este paradigma nos da un mejor software el cual está mejor preparado para cambios futuros ya que no tiene la necesidad de establecer el procedimiento por el cual se va a alcanzar un objetivo.

2.5.2 Programación imperativa

Este paradigma es el más simple. La principal función es el poder hacer la ejecución secuencial de acciones. Varios lenguajes se adhieren al paradigma imperativa y cuentan con muchas variantes. Antes, los programadores solían escribir en lenguajes procedimentales, pero con el paso del tiempo, este tipo de programación ha ido quedando obsoleta.

2.5.3 Programación lógica

La programación lógica expresa los objetivos del programa como una colección de afirmaciones o reglas las cuales están basadas en los diferentes resultados y restricciones de la lógica matemática, ya que el problema a resolver se debe modelar con enunciados de lógica de primer orden.

2.5.4 Programación estructurada

Este tipo de programación es uno de los más comunes de utilizar ya que con ella, los programadores pueden aprender a crear código, dando instrucciones a la computadora de una manera secuencial, siguiendo un orden hasta que todas las indicaciones se hayan completado.

2.5.5 Programación orientada a objetos

La programación orientada a objetos es un paradigma el cual proporciona clases y objetos. Este tipo de programación se emplea principalmente para estructurar software en piezas simple y poder reutilizar los planos de código para crear diferentes instancias individuales de objetos

2.6 Bases de datos

2.6.1 ¿Qué es una base de datos?

Una base de datos es una recopilación definida de información o datos estructurados, los cuales son almacenados de forma electrónica en algún software o sistema informático. Por lo general, una base de datos es manejada por un sistema gestor de base de datos (SGBD). Los datos, juntos al SGBD y las diferentes aplicaciones o programas asociados entre ellos, reciben el nombre de “sistema de bases de datos” pero simplemente es abreviado a base de datos.

Antes de que las bases de datos existieran, se hacía uso de los sistemas de ficheros. Este sistema surgió al informatizar la administración de los archivadores manuales esto con el fin de poder ofrecer un acceso más eficiente a los datos que fueran almacenados. Un sistema de ficheros sigue a un modelo descentralizado, en el cual, cada departamento de una empresa puede almacenar y gestionar todos los informes o reportes que fueran necesarios mediante una serie de programas los cuales están hechos especialmente para ese tipo de tareas.

Cuando una empresa tiene implementado un sistema de ficheros, cada uno de los departamentos no deben compartir información ni aplicaciones, esto hace que los datos que se lleguen a ocupar en varios departamentos deberán ser duplicados. Esto puede generar inconsistencias cuando se llegan a duplicar documentos y los datos no llegan a tener coincidencia, ya que, si dos copias se encuentran en departamentos diferentes y solo a uno de ellos se le ha informado algún cambio para algún fichero, los demás departamentos no tendrán esta actualización.

Una base de datos se puede ver como un gran almacén donde se pueden guardar grandes cantidades de datos, que se define y se crea una sola vez, también puede ser utilizada por diferentes usuarios al mismo tiempo. “Una ventaja que ofrecen las bases de datos es que los datos que son integrados cuentan con una mínima cantidad de duplicidad, esto nos quiere decir que no pertenece a un solo departamento o área, sino que es compartida para toda la empresa. Además, la base de datos no solo puede almacenar datos de la organización, sino que también descripción de estos.” (Marqués 2011)

2.6.2 Sistemas gestores de bases de datos

Un sistema gestor de base de datos o abreviado por sus iniciales SGBD, es una aplicación la cual permite a los usuarios el poder definir, crear y darles mantenimiento a las bases de datos además de que ofrecen un acceso controlado al sistema para cada uno de los empleados. Se le denomina “sistema de bases de datos” a todo el conjunto de bases de datos, el SGBD que se esté utilizando y a los programas de aplicación que dan el servicio a la empresa u organización.

El modelo que siguen los sistemas de base de datos tiene una similitud muy grande con el proceso que se sigue actualmente para el desarrollo de programas que usan algún lenguaje con programación orientada a objetos, en donde una implementación interna de un objeto y una especificación externa separada. Una de las ventajas que tiene este modelo, que es mejor conocido como abstracción de datos, es que se pueden cambiar las implementaciones internas que tenga algún objeto sin la necesidad de afectar a los usuarios ya que la especificación externa no se verá afectada. Los sistemas gestores de bases de datos se organizan en tres niveles:

Nivel interior: Son los encargados de la manipulación física de los datos. El usuario pierde de vista como y en donde es que se almacenan los datos, aquí es donde el SGBD se encarga de todas las tareas.

Nivel externo: Aquí está la visión del usuario es la manipulación lógica de los datos que se van a guardar. El usuario maneja la base a través de un entorno organizado mediante una interfaz que sirve de guía sin la necesidad de tener conocimientos sobre la estructura de la base.

Nivel medio: En este nivel, se hace la descripción de la base de datos mediante un lenguaje conceptual el cual permite una visión abstracta del sistema. Se entiende que el lenguaje conceptual, está formado por los comandos de fácil memorización los cuales permiten el ejecutar todas las tareas para el acceso y el manejo de los datos, así como las diferentes tareas de mantenimiento a todo el sistema de gestión.

CAPITULO III DESARROLLO

Es importante mencionar que todas las pruebas realizadas en este informe serán realizadas de manera local creando un ambiente de desarrollo con “XAMPP”, esto con el objetivo de no entorpecer los registros que se tienen en producción, para consultas directas, podrán visitar la página: <https://registro.dimueble.com.mx>

3.1 Desarrollo de base de datos

Como primer parte del desarrollo de la página, se empezará con la creación de la base de datos para poder almacenar a los participantes del concurso y también se deberán crear demás tablas para el administrador de la página.

Servidor: 127.0.0.1 » Base de datos: 2023_dinimable											
Estructura	SQL	Buscar	Generar una consulta	Exportar	Importar	Operaciones	Privilegios	Rutinas	Eventos	Dispara	
☐ ci_category	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	3 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ ci_category_post	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	2 InnoDB utf8_general_ci	16.0 KB	-	
☐ ci_content	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	27 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ ci_events	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 InnoDB utf8_general_ci	16.0 KB	-	
☐ ci_gallery_content	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ ci_gallery_product	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ ci_mails_contact	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ ci_mails_newsletter	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 MyISAM utf8_spanish_ci	1.0 KB	-	
☐ ci_posts	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	1 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ ci_products	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	4 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ ci_sections	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	3 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ ci_sm	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	1 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ ci_social	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	7 InnoDB utf8_general_ci	16.0 KB	-	
☐ ci_structure	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ ci_subcategory	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 InnoDB utf8_general_ci	16.0 KB	-	
☐ ci_users_admin	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	15 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ directorio	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ estados	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	32 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ evaluaciones	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	5 InnoDB utf8mb4_general_ci	16.0 KB	-	
☐ eventos_registro	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 MyISAM latin1_swedish_ci	1.0 KB	-	
☐ formulario_directorio	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 InnoDB utf8_general_ci	16.0 KB	-	
☐ idiomas	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ municipios	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	2,457 InnoDB utf8_spanish_ci	144.0 KB	-	
☐ onlytext	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0 InnoDB utf8_spanish_ci	16.0 KB	-	
☐ payments	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	6 InnoDB utf8_unicode_ci	16.0 KB	-	
☐ primer_fase	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	28 InnoDB utf8mb4_general_ci	16.0 KB	-	
☐ registros_equipo	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	155 InnoDB utf8mb4_general_ci	80.0 KB	-	
☐ registros_individuales	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	258 InnoDB utf8mb4_general_ci	176.0 KB	-	

Ilustración 3 Tablas de nuestra base de datos

En los diversos desarrollos webs, se considera un aproximado de 20 tablas base para todos los proyectos, pero en este caso, no se hará uso de todas las tablas debido al tipo de proyecto, las cuales se explicarán a continuación y se dará la estructura que se tiene.

3.1.1 Tabla de campus

Esta tabla ayudará para almacenar todos los campus de las carreras que se tengan registradas junto a su ciudad y estado.

Tabla 2 Campus para las universidades

Campo	Definición
Id	Es un identificador único para cada registro en la tabla
id_universidad	Nos servirá para hacer la relación a la universidad correspondiente
nombre universidad	Nombre completo de la universidad
campus	Nombre completo del campus ingresado
ciudad	Ciudad correspondiente al campus
estado	Estado en el cual se encuentra el plantel

3.1.2 Tabla de carreras

Servirá para poder almacenar las diferentes carreras que los administradores deseen poner

Tabla 3 Carreras

Campo	Definición
Id	Nos ayudara como clave unica del registro
nombre	Nombre de la carrera que se registra

3.1.3 Tabla de contenidos

En particular, esta tabla se podrá encontrar en todos los proyectos que se desarrollen en “Sienois” ya que es la responsable de almacenar los contenidos que se encuentren en el administrador de la página. En este caso, no pondremos todos los campos, solo estaremos incluyendo los que se utilizaron en el administrador

Tabla 4 ci_content

Campo	Definición
Id	Nos ayudara como clave uncia del registro
module	Nos indica si es un registro agregable y a que grupo pertenece
section	En que sección la encontraremos
position	Nos servirá para acomodar los registros que se tengan de menos a mayor
title	podrá ser llenado con un texto simple
subtitle	Al igual que title, será un texto simple
subtitle2	De igual manera será un texto simple
description1	Aquí podremos encontrar un editor de texto más completo
description2	El mismo funcionamiento que description1
image1	Almacenará el nombre de una imagen.

3.1.4 Tabla de administradores

Esta tabla, almacenará los diferentes usuarios que se den de alta para poder acceder al administrador

Tabla 5 Tabla para los administradores del sitio

Campo	Definición
Id	Nos ayudara como clave uncia del registro
user	Es el nombre del administrador
email	Correo para iniciar sesión
password	Contraseña para iniciar sesión
hash_key	Clave única para reestablecer contraseña
hash_expiry	Fecha límite para reestablecer contraseña
permission	Permisos para el administrador
evento	A que secciones puede acceder

3.1.5 Tabla para los registros individuales

Esta tabla será la más grande que se tenga en la base, ya que está almacenando todos los registros e información de los participantes.

Tabla 6 Registro para los participantes individuales

Campo	Definición
Id	Nos ayudara como clave uncia del registro
nombre	Nombre del participante
apellido	Apellidos del participante
correo	Correo electrónico del participante
contraseña	Contraseña para iniciar sesión la cual estará encriptada
id_tipo_registro	Identificador para el tipo de registro al que pertenezca
nacimiento	Fecha de nacimiento
genero	Sexo del participante
profesion	Profesión de los participantes profesionales
celular	Número de teléfono
fijo	Teléfono fijo
cp	Código postal del participante donde radica
ciudad	Ciudad donde radica
archivo1	Aquí se guarda el nombre del archivo para el INE
archivo2	Se ocupa para guardar el nombre del comprobante de estudios
universidad	Id de la universidad del participante
nombre_universidad	Nombre de la universidad
campus	Campus correspondiente
carrera	Id de la carrera que estudia el participante
nombre_carrera	Carrera que estudia el participante
ciudad_universidad	Ciudad del campus de la universidad
estado_universidad	Estado del campus de la universidad

activo	Estado para determinar si puede entrar el participante a su perfil
token	Token único para activación de cuenta
fecha_registro	Fecha del registro del participante
especialidad	Especialidad que curso el participante profesional
estado	Id del estado donde radica el participante
nombre_estado	Nombre del estado
pago	Nombre del archivo del comprobante de pago
nombre_empresa	Nombre de la empresa para el participante profesional
ciudad_empresa	Ciudad de la empresa para el participante profesional
estado_empresa	Estado de la empresa para el participante profesional
fundación_empresa	Fecha de la fundación de la empresa
segmento	Segmento que atiende la empresa
proyecto1	Nombre del proyecto número 1
proyecto1_url	Url para el proyecto 1
archivo1_1	Nombre del archivo del render para el proyecto 1
archivo1_2	Nombre del archivo de la memoria conceptual del proyecto 1
video_proyecto1	Url para el video del proyecto 1
proyecto2	Nombre para el proyecto 2
proyecto2_url	Url para el proyecto 2
archivo2_1	Nombre del archivo del render para el proyecto 1
archivo2_2	Nombre del archivo de la memoria conceptual del proyecto 2
video_proyecto2	Url para el video del proyecto 1
proyecto3	Nombre del proyecto número 3
proyecto3_url	Url para el proyecto 3
archivo3_1	Nombre del archivo del render para el proyecto 3

archivo3_2	Nombre del archivo de la memoria conceptual del proyecto 3
video_proyecto3	Nombre del proyecto número 3
completo	Campo para determinar si el registro se encuentra completo
primero	Saber si es su primer ingreso a la cuenta
hash_key	Clave para reestablecer contraseña
hash_expiry	Fecha de expiración para reestablecer contraseña

3.1.6 Tabla para primera fase de calificaciones

Esta tabla será la responsable de almacenar las primeras calificaciones que se tengan para los participantes y algunas observaciones por juez.

Tabla 7 Primer fase para las calificaciones de los participantes

Campo	Definición
Id	Identificador único para el registro
id_juez	Id del juez que hizo la calificación
proyecto	Nombre del proyecto calificado
proyecto_url	Url del proyecto calificado
archivo_1	Nombre del render del proyecto
archivo_2	Nombre de la memoria conceptual del proyecto
observaciones	Observaciones realizadas por el juez
pasa	Calificación de si o no para el participante y su proyecto

3.1.7 Tabla para segunda fase de evaluaciones

Dentro de esta tabla, se almacenan las calificaciones por juez para la segunda etapa de calificaciones.

Tabla 8 Evaluaciones de la segunda fase del concurso

Campo	Definición
-------	------------

Id	Nos ayudara como clave uncia del registro
representante	Id del representante del proyecto
proyecto	Nombre del proyecto
jurado	Id del jurado que hizo la calificación
originalidad	Originalidad del proyecto
fidelidad	Fidelidad del proyecto
estética	Estética del proyecto
ergonomía	Ergonomía del proyecto
innovación	Innovación del proyecto
calidad	Calidad del proyecto
solidez	Solidez del proyecto
factibilidad	Factibilidad del proyecto
viabilidad	Viabilidad del proyecto
funcionalidad	Funcionalidad del proyecto
valor_sustentable	Valor sustentable del proyecto
puntuación	Puntuación final del proyecto
observaciones_coordinador	Observaciones del coordinador al proyecto
observaciones_jurado	Observaciones del jurado al proyecto

Estas 7 tablas serán las principales que se usarán para el almacenamiento de los datos para todos los participantes que se registren en el concurso. Una vez que ya se han creado las tablas correspondientes para el manejo de toda la información que estará llegando para los registros se podrá empezar con la primera parte para este sitio web, el cual va a ser el registro para todos los participantes.

3.2 Vistas

3.2.1 Home

En la parte de código, al usar el framework de Codeigniter, podremos crear diferentes vistas que pueden ayudar como base o templates para las páginas, solo se deberán cargar con

la sintaxis de php para llamar a elementos ubicados en el proyecto. Se estará cargando dentro de la etiqueta head la hoja de estilos y el template para el head.

```
<!DOCTYPE HTML>
<html lang="es-MX">
<head>
    <?php $this->load->view('pages/template/head',$favicon); ?>
    <style type="text/css" media="screen">
        <?php $this->load->view('pages/template/preloader'); ?>
        .bg-images{background-image: url('<?php base_url();?> img/UniMenu42/<?php print_r($banner['image1']); ?>')}

        @media screen and (max-width:960px){
            .bg-images{height: 30vh}
        }
    </style>
</head>
```

Ilustración 4 Etiqueta head de nuestra página de inicio

Dentro de la vista para el head, se tendrán las etiquetas de meta y link, las cuales ayudará para la decodificación UTF-8 y las medidas de la página.

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<?php
    $host= $_SERVER["HTTP_HOST"];
    $url= $_SERVER["REQUEST_URI"];
    $url_completa = "https://" . $host . $url;
?>
<link rel="alternate" hreflang="es-mx" href="<?php echo $url_completa; ?>" />

<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
<!--[if lt IE 9]>
```

Ilustración 5 Primeras etiquetas de nuestro template de head

Dentro del mismo archivo, se tendrá una etiqueta de style, la cual va a ser muy importante para el proyecto, ya que una petición del cliente fue que los colores principales y secundarios de la página pudieran ser administrables. Siguiendo con las etiquetas, se tendrá el logo de la pestaña de la página y la parte de recursos como estilos adicionales para la página de home.

```

<style>
:root {
  /* PRIMARY COLORS */
  --primary: <?= $colores['subtitle'] ?> !important;
  --hover-primary: <?= $colores['subtitle2'] ?> !important;
  --focus-primary: <?= $colores['txt1'] ?> !important;
  /* SECONDARY COLORS */
  --secondary: <?= $colores['txt2'] ?> !important;
  --hover-secondary: <?= $colores['txt3'] ?> !important;
  --focus-secondary: <?= $colores['txt4'] ?> !important;
}
</style>

<link type="image/x-icon" href="<?= base_url() ?>img/Content/<?= $favicon[0]['image2'] ?>" rel="icon" />

<link rel="stylesheet" href="<?php print base_url(); ?>resources/frontend/css/global.css<?php echo date('Y-m-d H:i:s'); ?>">

```

Ilustración 6 Colores administrables para el sitio

Siguiendo con la página de home, se tendrán diversas variables para poder cambiar el logo que se encuentra en la parte superior de la página y de igual manera los dos textos simples que se tiene debajo con las leyendas “Gracias por participar, registro cerrado” y “Dimueble 17ª edición”

```

<div class="mt-1-md full-width-xl">
  <figure class="logo-home m-auto-md">
    " width="400" height="92">
  </figure>
  <div class="my-2">
    <h1 class="title-3 extra-bold">
      <?= $Contenido[0]['title'] ?>
      <span class="c-primary"><?= $Contenido[1]['title'] ?></span>
    </h1>
    <h2 class="title-small-1 bold"><?= $Contenido[1]['subtitle'] ?></h2>
  </div>
  <p class="fs-size20">Por favor, elige cuidadosamente tu categoria</p>

```

Ilustración 7 Títulos para nuestro home

Como siguiente parte, se tendrá el apartado para los botones de las diferentes categorías, es importante mencionar que en el caso de que una de las categorías fuera dada de baja, se tendría que desactivar desde el administrador y esto evitaría que la gente pueda acceder al registro.

```

<div class="row evenly-xl mt-2 px-2">

  <?php if($Botones[0]['enable'] == 'on') {?>
    <a href="<?php print base_url()>tipo-registro-estudiante" title="Tipo de registro" class="btn-medium btn-full-md my-05 btn-secondary bold fsize20 py-1">
      <?php print $Botones[0]['title'] ?>
    </a>
  <?php } ?>

  <?php if($Botones[1]['enable'] == 'on') {?>
    <a href="<?php print base_url()>tipo-registro-profesional" title="Tipo de registro" class="btn-medium btn-full-md my-05 btn-secondary bold fsize20 py-1">
      <?php print $Botones[1]['title'] ?>
    </a>
  <?php } ?>

  <?php if($Botones[2]['enable'] == 'on') {?>
    <a href="<?php print base_url()>tipo-registro-elektra" title="Tipo de registro" class="btn-medium btn-full-md my-05 btn-secondary bold fsize20 py-1">
      <?php print $Botones[2]['title'] ?>
    </a>
  <?php } ?>

</div>

```

Ilustración 8 Categorías habilitadas para los registros

Siguiente a los botones, se tiene una sección para un carrusel de imágenes, el cual va a contener algunas marcas patrocinadoras dentro del concurso. Conforme se vayan agregando más aliados, se podrá hacer un slide que se desliza de forma automática.

```

<div class="col-xl-12 col-md-12 mt-2">
  <h2 class="uppercase c-primary"><strong>Aliados estratégicos</strong></h2>
  <div class="carousel-cat flex-slide my-1 text-web">
    <?php $totalp = count($Patrocinadores);
    for($i=0; $i<$totalp; $i++) {
      <figure class="text-center pt-1 justify">
        <picture>
          
          <?php print_r($Patrocinadores[$i]['image1']); ?>
          src="<?php print base_url()>img/UniMenu45/">
          <?php print_r($Patrocinadores[$i]['image1']); ?>
          class="lazyload"
          loading="lazy"
          alt="<?php print_r($Patrocinadores[$i]['alt1']); ?>" width="229" height="77">
        </picture>
      </figure>
    }
  </div>
</div>

```

Ilustración 9 Carrusel para los aliados estratégicos dentro de la página

Como última parte del home, se muestra una sección para hacer un redirect a una página donde se encuentran preguntas frecuentes, poder enviar un correo con dudas más específicas o poder iniciar sesión en el sitio.

```

<div>
  <div>
    <a href="https://dimueble.com.mx/faqs" target="_blank" class="c-primary">Preguntas frecuentes</a>
    <p> Para obtener asistencia técnica con el registro, haga clic <a href="mailto:informes@dimueble.com.mx" target="_blank" class="c-primary">aquí</a>
    </p>
    <p>¿Ya tienes una cuenta? <a href="<?php print base_url()>iniciar-sesion"> Inicia sesión</a>
  </div>
</div>

```

Ilustración 10 Preguntas frecuentes para los participantes

3.2.2 Tipo de registro

Dentro de la plataforma para el tipo de registro (individual o grupal), primero se estará mostrando el título para el registro que haya seleccionado el participante, se tendrán 3 textos los cuales dirán: Registro estudiante, Registro profesional y para la categoría “Elektra” se mostrará el logo de la empresa.

```
<?php if($this->uri->segment(1) == 'tipo-registro-estudiante') {?>
    <h2 class="title-1 text-center extra-bold mt-1 mb-2 uppercase">Registro estudiante</h2>
<?php } else if($this->uri->segment(1) == 'tipo-registro-profesional') { ?>
    <h2 class="title-1 text-center extra-bold mt-1 mb-2 uppercase">Registro profesional</h2>
<?php } else if('tipo-registro-elektra') { ?>
    <h2 class="title-1 text-center extra-bold my-2 uppercase">Registro
        
    </h2>
<?php } ?>
```

Ilustración 11 Título para el tipo de registro

Dependiendo el caso del participante (estudiante, profesional o Elektra) se le mostraran dos botones con las leyendas “Individual” o “Equipo”, lo que va a diferenciar entre cada botón es la página a la que se va a dirigir.

```
<?php if($this->uri->segment(1) == 'tipo-registro-estudiante') {?>
    <div class="col-xl-6 col-md-12 mt-2 mt-1-md d-flex-md center-md text-right">
        <a href="<?php print(base_url()) ?>registro-individual-estudiantes" class="btn-medium btn-primary extra-bold py-1 px-2 btn-full-md">Individual</a>
    </div>
    <div class="col-xl-6 col-md-12 mt-2 mt-1-md text-left">
        <a href="<?php print(base_url()) ?>registro-grupal-estudiantes" class="btn-medium btn-primary extra-bold py-1 px-2 btn-full-md">Equipo</a>
    </div>
```

Ilustración 12 Botones para la categoría de estudiantes

```
<?php } else if($this->uri->segment(1) == 'tipo-registro-profesional') { ?>
    <div class="col-xl-6 col-md-12 mt-2 mt-1-md d-flex-md center-md text-right">
        <a href="<?php print(base_url()) ?>registro-individual-profesional" class="btn-medium btn-primary extra-bold py-1 px-2 btn-full-md">Individual</a>
    </div>
    <div class="col-xl-6 col-md-12 mt-2 mt-1-md text-left">
        <a href="<?php print(base_url()) ?>registro-grupal-profesional" class="btn-medium btn-primary extra-bold py-1 px-2 btn-full-md">Equipo</a>
    </div>
```

Ilustración 13 Botones para la categoría profesional

```
<?php } else if($this->uri->segment(1) == 'tipo-registro-elektra') { ?>
    <div class="col-xl-6 col-md-12 mt-2 mt-1-md d-flex-md center-md text-right">
        <a href="<?php print(base_url()) ?>registro-individual-elektra" class="btn-medium btn-primary extra-bold py-1 px-2 btn-full-md">Individual</a>
    </div>
    <div class="col-xl-6 col-md-12 mt-2 mt-1-md text-left">
        <a href="<?php print(base_url()) ?>registro-grupal-elektra" class="btn-medium btn-primary extra-bold py-1 px-2 btn-full-md">Equipo</a>
    </div>
<?php } ?>
```

Ilustración 14 Botones para la categoría Elektra

3.2.3 Registros de cuentas

Una vez que el participante ha seleccionado el tipo de registro, deberá crear una cuenta con información básica como lo es correo, nombre, apellidos y contraseña para iniciar

sesión. Dentro de esta página, se mostrará el título para que el usuario pueda crear su cuenta y dentro del bloque de código php se estará almacenando una variable la cual nos podrá informar algún problema que se tenga a la hora de hacer el registro.

```
<h1 class="title-3 text-center extra-bold c-secondary mt-2"><?= $Contenido['title']?></h1>
<p class="title-small-3 text-center">Es necesario crees una cuenta para realizar tu registro.</p>
<?php if(empty($error2)){ ?>
    <h2 class="alert-log text-center mt-1">
        <?php print $error2 ?>
    </h2>
<?php } ?>
```

Ilustración 15 Título de la página para crear la cuenta

Para el formulario de la cuenta, se tendrá que abrir la etiqueta form, a la cual se le dará la ruta donde se hará el proceso de la información introducida por el participante. Deberemos usar el método “post” para que se envíen los datos de manera correcta.

```
<!--Validamos la seccion de donde viene el usuario-->
<?php if($Contenido['id']=='23') { ?>
    <form action="<?php print base_url() ?>registro-individual-estudiantes" method="post"
<?php } else if($Contenido['id']=='24') { ?>
    <form action="<?php print base_url() ?>registro-individual-profesional" method="post"
<?php } else if($Contenido['id']=='25') { ?>
    <form action="<?php print base_url() ?>registro-individual-elektra" method="post" clas
<?php } ?>
```

Ilustración 16 Apertura de formulario para el participante

Solo se pedirán 4 campos para crear la cuenta. Pero se estarán agregando dos campos extras, el primero servirá para validar el correo y el segundo para validar la contraseña, la cual servirá para iniciar sesión dentro de la plataforma.

```
<div class="col-xl-6 col-md-12 my-1">
    <input type="text" name="nombre" id="nombre" placeholder="Nombre" required class="input-control" <?php if(empty(form_error('nombre'))) { print 'invalid-input'; } ?>
    <?php if(isset($error)) { print form_error('nombre'); } ?>
</div>
<div class="col-xl-6 col-md-12 my-1 mt-1">
    <input type="text" name="apellidos" id="apellidos" placeholder="Apellidos" required class="input-control" <?php if(empty(form_error('apellidos'))) { print 'invalid-input'; } ?>
    <?php if(isset($error)) { print form_error('apellidos'); } ?>
</div>
<div class="col-xl-6 col-md-12 my-1">
    <input type="email" name="correo" id="correo" placeholder="Correo electrónico" required class="input-control" <?php if(empty(form_error('correo'))) { print 'invalid-input'; } ?>
    <?php if(isset($error)) { print form_error('correo'); } ?>
</div>
<div class="col-xl-6 col-md-12 my-1">
    <input type="email" id="correoCon" placeholder="Confirma tu correo electrónico" class="input-control">
</div>
<div class="col-xl-6 col-md-12 my-1">
    <input type="password" name="contrasena" id="contrasena" placeholder="Contraseña" required class="input-control" <?php if(empty(form_error('contrasena'))) { print 'invalid-input'; } ?>
    <?php if(isset($error)) { print form_error('contrasena'); } ?>
</div>
<div class="col-xl-6 col-md-12 my-1">
    <input type="password" id="contraCon" placeholder="Confirma tu contraseña" class="input-control">
</div>
```

Ilustración 17 Campos para la creación de la cuenta

Como parte del sistema de seguridad, se le estará agregando un recaptcha a la página para evitar la creación de cuentas robots y tener la mayor cantidad de registros reales.

```
<div class="col-xl-6 col-md-12 mt-2 mt-1-md d-flex-md center-md">
  <div class="g-recaptcha" data-sitekey="XXXXXXXXXXXXXXXXXXXXXXXXXXXX" data-callback="enabledSubmit"></div>
</div>
```

Ilustración 18 Asignación de clave para captcha

Para la última parte de esta sección, se agregará un script el cual ayudará para la validación del correo y de la contraseña. En caso de que ambos valores para el correo sean iguales y lo mismo para las contraseñas, se podrá seguir con el registro. Dentro de este script de igual manera se agrega otra función para habilitar el botón de enviado una vez que el usuario ha marcado la casilla del recaptcha.

```
<script>
  function validarFormulario() {
    var correo = document.getElementById("correo").value;
    var correoConfir = document.getElementById("correoCon").value;
    var contra = document.getElementById("contrasena").value;
    var contraConfir = document.getElementById("contraCon").value;

    // const upperCaseRegex = /[A-Z]/;
    // const specialCharRegex = /[!@#$%^&*()_+[\]\{\};':"\|/,.<>/?]+/;
    // const numberRegex = /\d/;

    if (correo === correoConfir) {
      if (contra === contraConfir) {
        return true;
      } else {
        alert("Recuerda que las contraseñas ingresadas deben ser las mismas");
        return false;
      }
    } else {
      alert("Recuerda que los correos ingresados deben ser los mismos");
      return false;
    }
  }

  function enabledSubmit(response) {
    document.getElementById("contacto").disabled = false;
  }
</script>
```

Ilustración 19 Scripts para validar cuenta y recaptcha

3.2.4 Login para usuarios

Para el inicio de sesión, se va a requerir el correo y contraseña que se ocuparon dentro del registro anterior, también se va a usar el recaptcha para que se eviten los accesos no autorizados. En caso de que el usuario haya perdido u olvidado su contraseña podrá recuperarla vía correo electrónico.

```

<form action="{<?php base_url()>}iniciar-sesion" method="post" class="row mt-3">
  <div class="col-xl-6 col-md-12">
    <input class="input-control" type="email" name="usuario" placeholder="Correo electrónico">
  </div>
  <div class="col-xl-6 col-md-12 mt-1-md">
    <input class="input-control" type="password" name="contrasena" placeholder="Contraseña">
  </div>
  <div class="col-xl-6 col-md-12 mt-3 mt-1-md d-flex-md center-md">
    <div class="g-recaptcha" data-sitekey="6LeHvMepAAAAIzphv45DEl5l1lechalQsyhZx60S" ></div>
  </div>
  <div class="col-xl-6 col-md-12 mt-3 mt-1-md text-right">
    <button name="login" id="entrar" value="success" class="btn-medium btn-primary extra-bold py-1 px-2 btn-full-md">Iniciar sesión</button>
  </div>
  <div class="col-xl-12 text-center mt-1">
    <a class="c-text" href="{<?php base_url()>}recuperar" title="Deseo recuperar mi contraseña">Deseo recuperar mi contraseña</a>
  </div>
</form>

```

Ilustración 20 Campos para el inicio de sesión para los participantes

3.2.5 Datos del participante

Dentro de la primera sección en el registro, se podrán encontrar los campos necesarios para la información básica del participante y en caso de haber hecho registro por equipo, se le pedirá la información de cada participante. Se empezará con una sección de botones donde se mostrarán las diferentes secciones en caso de que el participante quiera actualizar información en específica.

```

<ul class="row my-2 center-xl">
  <li class="col-xl-3 col-md-12 my-05-md">
    <a href="{<?php base_url()>}registro-participante/{<?php print_r($this->uri->segment(2));?}"
      class="btn btn-full-xl btn-outline-secondary p-1 active">
      <?php if($Expositor['id_tipo_registro'] == '23') { ?>
        Datos del participante
      <?php } else { ?>
        Datos del líder y participantes
      <?php } ?>
    </a>
  </li>
  <li class="col-xl-3 col-md-12 my-05-md">
    <a href="{<?php base_url()>}documentacion/{<?php print_r($this->uri->segment(2));?}"
      class="btn btn-full-xl btn-outline-secondary p-1">
      <?php if($Expositor['id_tipo_registro'] == '23') { ?>
        Documentación del participante
      <?php } else { ?>
        Documentación del líder y participantes
      <?php } ?>
    </a>
  </li>
  <li class="col-xl-3 col-md-12 my-05-md">
    <a href="{<?php base_url()>}universidad/{<?php print_r($this->uri->segment(2));?}"
      class="btn btn-full-xl btn-outline-secondary p-1">
      Datos de la universidad
    </a>
  </li>
  <li class="col-xl-3 col-md-12 my-05-md">
    <a href="{<?php base_url()>}proyecto/{<?php print_r($this->uri->segment(2));?}"
      class="btn btn-full-xl btn-outline-secondary p-1">
      Datos del proyecto
    </a>
  </li>
</ul>

```

Ilustración 21 Secciones del registro

Los primeros campos que se le pedirán al usuario serán: su nombre, apellidos y fecha de nacimiento, el correo solo se mostrará como información complementaria.

```

<div class="col-xl-6 col-md-12 my-1">
  <label class="for-input"><span class="c-primary">*</span>Nombre</label>
  <input class="input-control" type="text" name="nombre" readonly value="<?php if(isset($Expositor['nombre'])) { echo $Expositor['nombre']; } ?>">
  <?php if(isset($error)) { print form_error('Nombre'); } ?>
</div>
<div class="col-xl-6 col-md-12 my-1 mt-1-md">
  <label class="for-input"><span class="c-primary">*</span>Apellidos</label>
  <input class="input-control" type="text" name="apellido" readonly value="<?php if(isset($Expositor['apellido'])) { echo $Expositor['apellido']; } ?>">
  <?php if(isset($error)) { print form_error('Apellido'); } ?>
</div>
<div class="col-xl-12 my-1">
  <label class="for-input"><span class="c-primary">*</span>Correo electrónico</label>
  <input class="input-control" type="email" name="correo" readonly value="<?php if(isset($Expositor['correo'])) { echo $Expositor['correo']; } ?>">
  <?php if(isset($error)) { print form_error('Apellido'); } ?>
</div>
<div class="col-xl-4 col-md-12 my-1">
  <label class="for-input"><span class="c-primary">*</span>Año de nacimiento</label>
  <input class="input-control" type="date" name="nacimiento" required value="<?php if(isset($Expositor['nacimiento'])) { echo $Expositor['nacimiento']; } ?>">
  <?php if(isset($error)) { print form_error('Nacimiento'); } ?>
</div>

```

Ilustración 22 Primera parte de la información básica del participante

Siguiendo con el llenado de la información básica, se solicitará el sexo del participante, teniendo las opciones de: Masculino, Femenino y Otro, para el caso de otro, se mostrará un campo adicional para que se pueda llenar a preferencia del participante.

```

<div class="col-xl-8 col-md-12 my-1">
  <label class="for-input">Género</label>
  <div class="d-flex-xl flex-wrap-md">
    <label class="radio-checkbox-custom mb-05-md">
      <input class="rc-input masculino" type="radio" name="genero" value="Masculino" <?php echo ($Expositor['genero'] == 'Masculino') ? 'checked' : ''; ?>
      <span class="input-checkbox">Masculino</span>
    </label>
    <label class="radio-checkbox-custom mb-05-md">
      <input class="rc-input femenino" type="radio" name="genero" value="Femenino" <?php echo ($Expositor['genero'] == 'Femenino') ? 'checked' : ''; ?>
      <span class="input-checkbox">Femenino</span>
    </label>
    <label class="radio-checkbox-custom mb-05-md">
      <input class="rc-input otro" type="radio" value="Otro" <?php echo ($Expositor['genero'] != null && $Expositor['genero'] != 'Femenino' && $Expositor['genero'] != 'Masculino') ? 'checked' : ''; ?>
      <span class="input-checkbox">Otro</span>
    </label>
    <input class="input-control specify" type="text" style="display:none;" placeholder="Especificar" value="<?php if(empty($Expositor['genero'])) { echo $Expositor['genero']; } ?>">
  </div>
</div>

```

Después de pedir el género, seguirá una parte para conocer sobre la localización del participante, se le pedirá el código postal, en el momento que se le dé clic al botón de buscar, mediante el uso de JavaScript se hará un consumo a un api para poder tener los nombres de estado y ciudad.

```

<div class="col-xl-6 col-md-12 my-1">
  <label class="for-input"><span class="c-primary">*</span>Código Postal</label>
  <div class="md flex-xl">
    <div class="tooltip-help">
      <input class="input-control" type="number" required id="codigoPostal" name="cp" maxlength="5" value="<?php if(isset($Expositor['cp'])) { echo $Expositor['cp']; } ?>">
      <label title="Ingresa tu código postal y da clic en buscar."><i class="icon-help"></i></label>
    </div>
    <button class="btn btn-primary btn-pill mx-05" type="button" id="buscaCodigo">Buscar</button>
  </div>
</div>
<div class="col-xl-6 col-md-12 my-1" id="DivCiudadInp">
  <label class="for-input"><span class="c-primary">*</span>Ciudad</label>
  <input class="input-control" type="text" name="ciudad" id="inputCiudad" value="<?php if(isset($Expositor['ciudad'])) { echo $Expositor['ciudad']; } ?>">
  <?php if(isset($error)) { print form_error('Ciudad'); } ?>
</div>
<div class="col-xl-6 col-md-12 my-1" style="display:none;" id="DivCiudadAPI">
  <label class="for-input"><span class="c-primary">*</span>Ciudad</label>
  <select class="input-control" id="selectCiudad">
    <option selected disabled>*Ciudad:</option>
  </select>
  <?php if(isset($error)) { print form_error('Ciudad'); } ?>
</div>
<div class="col-xl-6 col-md-12 my-1" id="DivEstadoInp">
  <label class="for-input"><span class="c-primary">*</span>Estado</label>
  <input class="input-control" type="text" name="estado" id="inputEstado" value="<?php if(isset($Expositor['estado'])) { echo $Expositor['estado']; } ?>">
  <?php if(isset($error)) { print form_error('Estado'); } ?>
</div>
<div class="col-xl-6 col-md-12 my-1" style="display:none;" id="DivEstadoAPI">
  <label class="for-input"><span class="c-primary">*</span>Estado</label>
  <select class="input-control" id="selectEstado">
    <option selected disabled>*Estado:</option>
  </select>
  <?php if(isset($error)) { print form_error('Estado'); } ?>
</div>

```

Ilustración 23 Campos para conocer la ubicación del participante

Dentro de una etiqueta script, vamos a hacer el consumo al api para poder conocer el estado y ciudad del usuario que esté haciendo el llenado del formulario.

```
e.preventDefault();
var codigoPostal = $('#codigoPostal').val();

$.ajax({
  url: "https://secure.geonames.org/postalCodeSearchJSON?postalcode=" + codigoPostal + "&country=MX&maxRows=15&username=r3gistr0_dim3ble",
  type: "POST",
  dataType: "json",
  success: function (data) {
    if (data.length === 0) {
      resultadosDiv.append('<p>Recuerda seleccionar tu país para encontrar tu ciudad y estado</p>');
    } else {
      ciudad.empty();
      estado.empty();
      var ciudadesUnicas = new Set();
      var estadosUnicos = new Set();

      $.each(data.postalCodes, function (index, item) {
        if (!ciudadesUnicas.has(item.adminName2)) {
          ciudad.append('<option>', {
            value: item.adminName2,
            text: item.adminName2,
          });
          ciudadesUnicas.add(item.adminName2);
        }

        if (item.adminName1 == "Distrito Federal") {
          item.adminName1 = "Ciudad de México";
        }

        if (item.adminName1 == "México") {
          item.adminName1 = "Estado de México";
        }

        if (!estadosUnicos.has(item.adminName1)) {
          estado.append('<option>', {
            value: item.adminCode1,
            text: item.adminName1,
          });
          estadosUnicos.add(item.adminName1);
        }
      });
    }
  }
});
```

Ilustración 24 Consumo del API

3.2.6 Registro de documentación

Para el apartado de la documentación, se le estará solicitando su INE, comprobante de estudios y en caso de ser registro profesional el pago correspondiente a la inscripción. Por esta parte será toda la información solicitada.

```
<div class="p-2 b-1 bg-light-gray full-width-xl">
  <div class="box relative mb-1">
    <input type="file" name="image2" id="input_file2" <?php if(empty($Expositor['archivo2']))(echo 'required'); ?>
    <label for="image2">
      <div>
        <i class="icon-paperclip"></i>
        <p>Adjunta tu archivo - <strong>JPG o PDF</strong></p>
      </div>
      <span class="none1024">o arrastra aquí.</span></span></br>
      <span><strong>Recuerda nombrar tu archivo: <br> E_DU_APELLIDOCONCURSANTE_NOMBREPROYECTO(S)</strong></span>
    </label>
    <p class="error error_size tooltip-invalid semibold"></p>
  </div>
  <div class="fs18 paragraph" id="Uni">
    <p>Copia de uno de los siguientes documentos que acredite el año que cursa el participante.</p>
    <ul>
      <li>Carta de estudios expedida por la Universidad.</li>
      <li>Boleta de calificaciones</li>
      <li>Pago de colegiatura.</li>
      <li>Credencial de estudiante vigente</li>
    </ul>
    <p>legibles</p>
    <p>No volteadas</p>
    <!-- <p>Revisa <a href="#" target="_blank">aquí</a> las especificaciones</p> -->
  </div>
</div>
```

Ilustración 25 Documentación para el participante individual

En el caso de que el registro haya sido por equipos, se le pedirán los mismos documentos para los demás integrantes del equipo, en caso de ser registro por equipo, el representante será el único que deberá subir el comprobante de pago.

```
<div class="col-xl-12 my-1">
  <p class="fsize20 bold mb-1">Coloca los datos solicitados de TODOS los integrantes del equipo, exceptuando del lider.</p>
  <p class="fsize20 bold c-secondary">Integrante 1</p>
</div>

<div class="col-xl-6 col-md-12 d-flex-xl flex-wrap-xl my-1">
  <div class="p-2 b-1 bg-light-gray full-width-xl">
    <div class="box relative mb-1">
      <input type="file" name="image3" id="input_file3" class="inputfile input-file inputfile-4" accept="image/jpeg, image/jpg, application/pdf">
      <label for="image3">
        <div>
          <i class="icon-paperclip"></i>
          <p>Adjunta tu archivo - <strong>JPG o PDF</strong></p>
        </div>
        <span><span class="none1024">o arrastra aquí.</span></span></div>
        <span><strong>Recuerda nombrar tu archivo: </br> E_ID_APELLIDOCONCURSANTE_NOMBREPROYECTO(S)</strong></span>
      </label>
      <p class="error error_size tooltip-invalid semibold"></p>
    </div>
    <div class="fsize18 paragraph id="INE3">
      <p>Copia digital de identificación oficial INE (ambos lados) o pasaporte.</p>
      <p>Legibles</p>
      <p>No volteadas</p>
      <!-- <p>Revisa <a href="#" target="_blank">aquí</a> las especificaciones</p> -->
    </div>
  </div>
</div>
```

Ilustración 26 Documentos para el primer integrante del equipo

Dentro de esta sección, se tendrán dos diferentes scripts para validar el tipo de documento que se sube a cada tipo de input y validar el peso del mismo.

```
function fileValidate(input) {
  input.on("change", (e) => {
    const file = $(e.target)[0].files[0];
    let fileName = file.name;
    var extension = fileName.split(".").slice(-1);
    extension = extension[0];
    let allowedExtensions = ["jpg", "pdf", "jpeg"];
    var error = input.nextAll('.error').first();
    error.text('');

    if (allowedExtensions.indexOf(extension) === -1) {
      error.text('Solo se permiten archivos JPG o PDF');
    } else {
      const max_file_size = 10000000;

      if (file.size > 0 && file.size > max_file_size) {
        const sizeMB = max_file_size / 1000000;
        error.text(`El tamaño máximo es ${sizeMB} MB`);
      } else {
        // Validación pasada. Puedes enviar el formulario o realizar otras acciones.
      }
    }
  });
}
```

Ilustración 27 Validador para el tipo de archivo

Como segundo script, se va a mostrar una sección para poder mostrar en tiempo real una previsualización de cada documento que suba el participante y también hará la validación para que se haya subido el documento con el nombre correcto que se le muestra en el formulario.

```
function muestraINE() {
    var input = document.getElementById('input_file');
    var file = input.files[0];

    if (file) {
        let nameFile = file.name;
        if(nameFile.includes("E_ID_")){
            $("#arch1").remove();
            var fileReader = new FileReader();
            fileReader.onload = function(e) {
                var visor = document.getElementById('INE');
                visor.innerHTML = '<iframe src="' + e.target.result + '" width="100%" height="500px"></iframe>';
            };
            fileReader.readAsDataURL(file);
        } else {
            input.value='';
            alert("Recuerda usar el nombre adecuado para este archivo");
        }
    }
}
```

Ilustración 28 Script para validar el nombre y mostrar previsualización del archivo

3.2.7 Datos universitarios

En caso de que el usuario haya sido de la categoría “Estudiante”, deberá de llenar la información respectiva a la universidad que haya sido dada de alta por los administradores del sitio.

```
<form action="{?php base_url(). 'universidad/'. print_r($id); }?" method="post" class="row mt-2 mt-1-md has-validation">
    <div class="col-xl-12">
        <p>Los campos marcados con un <span class="c-primary semibold">*</span> deben ser llenados obligatoriamente.</p>
    </div>
    <div class="col-xl-12 col-md-12 my-1">
        <label class="for-input"><span class="c-primary">*</span>Universidad</label>
        <select class="select-universidad" id="universidad" names="universidad" required>
            <option></option>
            <?php for($i=0;$i<count($Universidades);$i++) { ?>
            <option class="id-uni" value="{?php print_r($Universidades[$i]['Id']); }?" <?php if($Expositor['universidad'] == $Universidades[$i]['Id']){echo __("selected");} ?>>
            <?php } ?>
            <option value="999" <?php if($Expositor['universidad'] == '999'){echo __("selected");} ?>>Otra</option>
        </select>
        <?php if(isset($error)) { print form_error('universidad'); } ?>
    </div>
    <div class="col-xl-12 col-md-12 my-1 universidad"> <?php if($Expositor['universidad'] == '999'){echo "style=display:block";}else{echo("style=display:none");} ?>>
    <input class="input-control input-universidad" <?php if($Expositor['universidad'] == '999'){echo __("name=nombre_universidad");} ?> type="text" placeholder="Universidad"
    </div>
```

Ilustración 29 Formulario para los datos universitarios del participante

Se tendrá un script para poder validar el nombre de la universidad, campus, estado y carrera, en el caso de que la universidad del estudiante no aparezca en el listado final, pueda ingresar los datos de manera manual.

```

$(document).ready(function(){
    $('#select-universidad').on('change', function() {
        uni = document.getElementById("universidad").value;
        if(uni == '999'){
            $('#universidad').css("display", "block");
            $('#input-universidad').attr("name","nombre_universidad");
        } else {
            $('#universidad').css("display", "none");
            $('#input-universidad').removeAttr("name");
        }
    });
});

```

Ilustración 30 Script para datos universitarios

3.2.8 Información profesional

Para la categoría “Profesional” se le estará solicitando al participante, información básica sobre la empresa en la que trabaja. Será información como el nombre de la empresa, año de fundación, estado, entre otros campos.

```

<div class="col-xl-12">
    <p>Los campos marcados con un <span class="c-primary semibold">*</span> deben ser llenados obligatoriamente.</p>
</div>
<div class="col-xl-6 col-md-12 my-1">
    <label class="for-input"><span class="c-primary">*</span>Nombre comercial</label>
    <input class="input-control" type="text" name="nombre_empresa" value="<?php if(isset($Expositor['nombre_empresa'])) { echo $Expositor['nombre_empresa']; } ?>" />
    <?php if(isset($error)) { print form_error('Nombre empresa'); } ?>
</div>
<div class="col-xl-6 col-md-12 my-1 mt-1-md">
    <label class="for-input"><span class="c-primary">*</span>Ciudad</label>
    <input class="input-control" type="text" name="ciudad_empresa" value="<?php if(isset($Expositor['ciudad_empresa'])) { echo $Expositor['ciudad_empresa']; } ?>" />
    <?php if(isset($error)) { print form_error('Ciudad empresa'); } ?>
</div>
<div class="col-xl-6 my-1">
    <label class="for-input"><span class="c-primary">*</span>Estado</label>
    <input class="input-control" type="text" name="estado_empresa" value="<?php if(isset($Expositor['estado_empresa'])) { echo $Expositor['estado_empresa']; } ?>" />
    <?php if(isset($error)) { print form_error('Estado empresa'); } ?>
</div>
<div class="col-xl-6 col-md-12 my-1">
    <label class="for-input"><span class="c-primary">*</span>Fecha de fundación</label>
    <input class="input-control" type="date" name="fundacion_empresa" value="<?php if(isset($Expositor['fundacion_empresa'])) { echo $Expositor['fundacion_empresa']; } ?>" />
    <?php if(isset($error)) { print form_error('Fecha fundacion'); } ?>
</div>

```

Ilustración 31 Formulario para la empresa de la categoría profesional

3.2.9 Documentación de proyectos

Como última parte en la sección del registro de la información para el participante, se tendrá el apartado para cargar los documentos sobre los proyectos que se deseen subir, serán un máximo de 3 proyectos por participante o equipo.

El primer apartado de esta sección, se mostrará un link a una página en específico para mostrar los ejemplos del template para los documentos que se deseen cargar para los proyectos.

```
<?php if($Expositor['id_tipo_registro'] == '23' || $Expositor['id_tipo_registro'] == '36') { ?>
    <div class="mb-1">
        <p class="fsizel8 bold uppercase">Descarga <a class="c-secondary" target="_blank" href="https://dimueble.com.mx/categorias#formatos-e">aquí</a> lo
    </div>
</php ?>
else if($Expositor['id_tipo_registro'] == '24' || $Expositor['id_tipo_registro'] == '37') { ?>
    <div class="mb-1">
        <p class="fsizel8 bold uppercase">Descarga <a class="c-secondary" target="_blank" href="https://dimueble.com.mx/categorias#formatos-p">aquí</a> lo
    </div>
</php ?>
else if($Expositor['id_tipo_registro'] == '25' || $Expositor['id_tipo_registro'] == '38') { ?>
    <div class="mb-1">
        <p class="fsizel8 bold uppercase">Descarga <a class="c-secondary" target="_blank" href="https://dimueble.com.mx/categoria-elektro#formatos-elk">aquí</a> lo
    </div>
</php ?>
<div class="mt-1">
    <p class="fsizel8 bold">*Los renders deben contener 3 laminas de tu proyecto en formato pdf, peso no mayor a 10 MB.</p>
    <!-- <p class="fsizel8 bold">Revisa <a href="" target="_blank">aquí</a> las especificaciones</p> -->
</div>
<div class="row py-1">
```

Para la sección de los documentos, se deberán cargar dos archivos pdf, uno será para mostrar el modelo y armado del proyecto, dando una pequeña descripción del mismo y como segundo archivo deberá ser una memoria conceptual donde se den más detalles sobre el proyecto. Dependiendo la categoría del participante, serán los nombres que se le pidan para los archivos.

```
<div class="col-xl-12 mb-1">
    <label class="for-input"><span class="c-primary">*</span>Nombre del proyecto 1:</label>
    <input class="input-control" type="text" name="proyecto1" required value="<?php if(isset($Expositor['proyecto1'])) { echo $Expositor['proyecto1']; } ?>
    <?php if(isset($error)) { print form_error('Proyecto 1'); } ?>
</div>
<div class="col-xl-6 col-md-12">
    <div class="box relative mb-1">
        <input type="file" name="image1" id="proyecto1_1" class="inputfile input-file inputfile-4" accept="application/pdf" onchange="proyecto11()"
        <label for="image1">
            <div>
                <i class="icon-paperclip"></i>
                <p>Adjunta renders en formato - <strong>PDF</strong></p>
            </div>
            <span class="none1024">o arrastra aquí.</span></span></div>
            <?php if($Expositor['id_tipo_registro'] == '23' || $Expositor['id_tipo_registro'] == '36') { ?>
                <span><strong>Usar la siguiente estructura:</strong><br> E_R_*NOMBRE_DEL_PROYECTO*</strong></span>
            <?php } else if($Expositor['id_tipo_registro'] == '24' || $Expositor['id_tipo_registro'] == '37') { ?>
                <span><strong>Usar la siguiente estructura:</strong><br> P_R_*NOMBRE_DEL_PROYECTO*</strong></span>
            <?php } else if($Expositor['id_tipo_registro'] == '25' || $Expositor['id_tipo_registro'] == '38') { ?>
                <span><strong>Usar la siguiente estructura:</strong><br> ELK_R_*NOMBRE_DEL_PROYECTO*</strong></span>
            <?php } ?>
        </label>
        <p class="error error_size tooltip-invalid semibold"></p>
    </div>
```

Ilustración 32 Campos para los renders del proyecto 1

```

<div class="col-xl-6 col-md-12 mt-1-md">
  <div class="box relative mb-1 full-width-xl">
    <input type="file" name="image2" id="proyecto1_2" class="inputfile input-file inputfile-4" onchange="proyecto12()" accept="application/pdf"
    <label for="image2">
      <div>
        <i class="icon-paperclip"></i>
        <p>Adjunta memoria conceptual en formato - <strong>PDF</strong></p>
      </div>
      <span><span class="none1024">o arrastra aquí.</span></span></div>
      <?php if($Expositor['id_tipo_registro'] == '23' || $Expositor['id_tipo_registro'] == '36') { ?>
        <span><strong>Usar la siguiente estructura:</br> E_MC_*NOMBRE_DEL_PROYECTO*</strong></span>
      <?php } else if($Expositor['id_tipo_registro'] == '24' || $Expositor['id_tipo_registro'] == '37') { ?>
        <span><strong>Usar la siguiente estructura:</br> P_MC_*NOMBRE_DEL_PROYECTO*</strong></span>
      <?php } else if($Expositor['id_tipo_registro'] == '25' || $Expositor['id_tipo_registro'] == '38') { ?>
        <span><strong>Usar la siguiente estructura:</br> ELK_MC_*NOMBRE_DEL_PROYECTO*</strong></span>
      <?php } ?>
    </label>
    <p class="error error_size tooltip-invalid semibold"></p>
  </div>

```

Ilustración 33 Campo para la memoria conceptual de proyecto 1

Al igual con la sección de la documentación básica, se podrá validar que los documentos subidos no sobrepasen un peso específico y que solo se puedan subir archivos pdf.

```

function fileValidate(input) {
  input.on("change", (e) => {
    const file = $(e.target)[0].files[0];
    let fileName = file.name;
    var extension = fileName.split(".").slice(-1);
    extension = extension[0];
    let allowedExtensions = ["pdf"];
    var error = input.nextAll('.error').first();
    error.text('');

    if (allowedExtensions.indexOf(extension) === -1) {
      error.text('Solo se permiten archivos PDF');
    } else {
      const max_file_size = 10000000;

      if (file.size > 0 && file.size > max_file_size) {
        const tamañoEnMb = maxSize / 1000000;
        alert(`El tamaño máximo es ${tamañoEnMb} MB`);
      } else {
        // Validación pasada. Puedes enviar el formulario o realizar otras acciones.
      }
    }
  });
}

```

Ilustración 34 Validación para archivos pdf y peso máximo

3.2.10 Pantalla de agradecimiento

Una vez que se ha terminado el registro del usuario, se le mostrará una vista donde se le da la fecha límite para poder hacer modificaciones al proyecto o demás información que no haya registrado o cambios adicionales que se le pidan.

```

<body class="overflowHidden">
  <main class="row px-1 px-mg-0 py-2 home-intro middle-xl">
    <div class="container">
      <h1 class="title-3 text-center extra-bold c-secondary">Haz completado el registro profesional</h1>
      <p class="title-small-1 text-center">Puedes modificar tu informacion hasta la fecha: <?= $Limite['subtitle']?></p>
    </div>
  </main>

```

Ilustración 35 Pantalla de agradecimiento

3.3 Controladores

3.3.1 Site

Dentro de la función constructora, se mandará a llamar al modelo “Page_model” el cual se estará ocupando en la mayoría de este controlador, ya que es el responsable del manejo de la mayoría de la información para la página.

```

public function __construct()
{
    parent::__construct();
    $this->load->model('Page_model');
}

```

Ilustración 36 Modelo para mostrar la información recuperada de la base de datos

En la función index, solo se estará mostrando información ya que en la primera vista solo se cuenta con los títulos de la página, botones para el tipo de registro. Toda esta parte se encuentra totalmente administrable y se pueden cambiar los colores, favicon, activar/desactivar categorías, entre otras funciones.

```

0 references | 0 overrides
public function index() {
    $Info['favicon'] = $this->Page_model->Consulta_Template();
    $Info['Contenido'] = $this->Page_model->consulta_Contenido('1');
    $Info['Botones'] = $this->Page_model->consulta_module('16');
    $Info['Patrocinadores'] = $this->Page_model->consulta_Sliders_Special('45');
    $Info['logo'] = $this->Page_model->consulta_x_id2('44');
    $Info['banner'] = $this->Page_model->consulta_x_id2('43');
    $Info['colores'] = $this->Page_model->consulta_x_id2('48');
    $this->load->view('pages/home',$Info);
}

```

Ilustración 37 Función index de nuestra página

Para el login, se empezará con el procesamiento y validación de los datos que hayan sido ingresados por el usuario para entrar a su cuenta. Los dos primeros elementos del arreglo

“info” tendrán los colores principales de la página, el logo favicon y la imagen del banner. Seguido de esta información se tiene nuestra condicional para recibir el formulario de inicio de sesión, el cual estará validando la contraseña y usuario del participante.

```
0 references | 0 overrides
public function login() {
    $info['colores'] = $this->Page_model->consulta_x_id2('48');
    $info['favicon'] = $this->Page_model->Consulta_Template();
    $info['error2']='';
    $info['banner'] = $this->Page_model->consulta_x_id2('44');
    if ($this->input->post('login')== 'success'){
        $datos = array('usuario' => $this->input->post('usuario'),
            'contrasena' => $this->input->post('contrasena'));
        $this->form_validation->set_rules('usuario', 'Usuario', 'trim|required|xss_clean');
        $this->form_validation->set_rules('contrasena', 'Contraseña', 'trim|required|xss_clean');
```

Ilustración 38 Información de nuestro index y recepción del formulario

En caso de que algún dato no haya sido validado de manera correcta, se estarán creando dos elementos más para nuestro arreglo “info” los cuales tendrán información del error generado, si la validación fue correcta, cargaremos el modelo “LoUser” para enviar las credenciales y hacer codificación de las mismas.

```
if($this->form_validation->run() == FALSE){
    $info['error'] = validation_errors();
    $info['datos'] = $datos;
    $this->load->view('login',$info);
} else {
    $entonces='';
    $this->load->model('LoUser');
    $entonces = $this->LoUser->login_participantes($datos);
```

Ilustración 39 Validación del formulario y manejo de datos

Cuando el modelo ha regresado la respuesta para el tipo de usuario, se redireccionará a la página correspondiente. Pero en caso de que haya encontrado algún problema con la cuenta, se le regresará al participante a la pantalla de login junto con un mensaje de error para que pueda corregir sus datos de ingreso.

```

if($sentonces['id_tipo_registro'] == '23' || $sentonces['id_tipo_registro'] == '36'){
    redirect('registro-participante/'.$sentonces['Id']);
}else if($sentonces['id_tipo_registro'] == '24' || $sentonces['id_tipo_registro'] == '37'){
    redirect('profesional/'.$sentonces['Id']);
}else if($sentonces['id_tipo_registro'] == '25' || $sentonces['id_tipo_registro'] == '38'){
    redirect('elektra/'.$sentonces['Id']);
}else{
    if($sentonces['id_tipo_registro']=='Sin cuenta'){
        $info['error2'] = 'El usuario ingresado no existe, favor de crear una cuenta';
    }
    else if($sentonces['id_tipo_registro']=='Incorrecto'){
        $info['error2'] = 'El usuario o la contraseña introducida es incorrecta.';
    }
    else if($sentonces['id_tipo_registro']=='Sin activar'){
        $info['error2'] = 'Gracias por participar, espera noticias en nuestros canales oficiales';
    }
    $this->load->view('pages/login',$info);
}

```

Ilustración 40 Acceso al portal del registro o negar acceso

Cuando se el participante haya terminado de llenar su registro o actualizar información de su perfil, podrá cerrar su sesión

```

0 references | 0 overrides
public function logout(){
    $this->session->sess_destroy();
    // session_destroy();
    redirect('iniciar-sesion','refresh');
}

```

Ilustración 41 Cierre de sesión del participante

La siguiente función que se a usar, va a ser la responsable de mostrar la información correspondiente el registro de los estudiantes de manera individual, es importante mencionar que toda esta información será administrable.

```

public function registro_estudiante() {
    $Info['colores'] = $this->Page_model->consulta_x_id2('48');
    $Info['favicon'] = $this->Page_model->Consulta_Template();
    $Info['Contenido'] = $this->Page_model->consulta_x_id2('23');
    $Info['aviso_privacidad'] = $this->Page_model->consulta_x_id2('51');
    $Info['error2']='';
    $Info['banner'] = $this->Page_model->consulta_x_id2('44');
}

```

Ilustración 42 Información para mostrar en la vista del registro de los estudiantes

Dentro de la misma función para el registro del estudiante, se va a validar que el formulario que se haya enviado, tenga el método de envío “POST”, en caso de ser verdadero seguirá con la validación de los demás campos que se hayan ingresado.

```
if($this->input->post('contacto-enviar')== 'ok-registro'){  
    $this->form_validation->set_rules('nombre', 'Nombre', 'trim|required|xss_clean');  
    $this->form_validation->set_rules('apellidos', 'Apellidos', 'trim|required|xss_clean');  
    $this->form_validation->set_rules('correo', 'Correo', 'trim|required|xss_clean|valid_email');  
    $this->form_validation->set_rules('contrasena', 'Contrasena', 'trim|required|xss_clean');  
    $this->form_validation->set_rules('g-recaptcha-response', 'Captcha', 'trim|required|xss_clean');
```

Ilustración 43 Comprobar campos ingresados

En caso de que algún dato no haya sido validado de manera correcta, nos devolverá a la vista con el login del participante y nos mostrará los datos que se hayan enviado para volver a tener que escribir todo.

```
if(($this->form_validation->run() == false)){  
    $Info['error'] = validation_errors();  
    $Info['datos'] = $this->input->post();  
    $Info['error'];  
    $this->load->view('pages/sections/registro-individual', $Info);
```

Ilustración 44 Validación errónea de los campos del formulario

Si la validación pasada fue hecha de manera correcta, se cargará la librería “Password”, la cual nos va a ayudar para encriptar la contraseña, se generará un token único el cual será para futuras validaciones de la cuenta. Después de esto, todos los valores guardados en el input, serán almacenados en un array, el cual será enviado a la base de datos para ser guardados.

```

$this->load->library('Password');
$Pass = $this->password->crypt_pass($this->input->post('contrasena'));
$Token = md5(uniqid(mt_rand(), false)); //Generamos un token para validar la cuenta

$datos['nombre'] = $this->input->post('nombre');
$datos['apellido'] = $this->input->post('apellidos');
$datos['correo'] = $this->input->post('correo');
$datos['contrasena'] = $Pass;
$datos['id_tipo_registro'] = $Info['Contenido']['id'];
$datos['fecha_registro'] = date("Y-m-d");
$datos['token'] = $Token;
$datos['activo'] = false;
$datos['completo'] = false;

$participante = $this->Page_model->busca_participante($datos);

```

Ilustración 45 Creación de arreglo para la información del estudiante.

Una vez que el registro haya sido añadido a la base de datos, al participante se le estará enviando un correo de confirmación de que ha sido registrado de manera exitosa al concurso.

```

// *** CORREO ***
//email
$asunto = $correos['subtitle'];
$para = $datos['correo'];
$emisor = $correos['title'];
$copiaOcultaPara = '';
$mensaje = $this->load->view('emails/correo_contacto', $datos, TRUE);

$this->load->library('phpmailer_lib'); // Load PHPMailer library
$mail = $this->phpmailer_lib->load(); // PHPMailer object

// SMTP configuration
$mail->isSMTP();
$mail->Host = '*****'; // Host de conexión SMTP
$mail->SMTPAuth = true;
$mail->Mailer = "*****";
$mail->Username = $emisor;
$mail->Password = '*****';
$mail->SMTPSecure = '*****';
$mail->Port = *****;
$mail->CharSet = 'UTF-8'; // SE AGREGO ESTA LINEA PARA LA CORRECTA CODIFICACIÓN DE LOS TEXTOS
$mail->setFrom($emisor,$asunto);
// $mail->addReplyTo('info@example.com', 'CodexWorld');

```

Ilustración 46 Envío de correo al usuario

Cuando el correo haya sido enviado, se le redirigirá al usuario a una pantalla de agradecimiento. En caso de que la validación de correo haya sido errónea, se nos devolverá a la pantalla del registro.

```
        redirect('gracias-individual');
    } else {
        $Info['error2'] = "El correo introducido ya existe para una cuenta creada";
        $this->load->view('pages/sections/registro-individual',$Info);
    }
}
```

Ilustración 47 Redirección al usuario

Para los otros tipos de registro (Profesional y Elektra) el funcionamiento para el registro, será el mismo al que hemos visto. Por lo cual se omitirá esta parte para no repetir los mismos pasos. Para que el usuario pueda acceder a su cuenta, dentro del correo se le enviará un link el cual contendrá el token único que se generó a la hora de crear el registro.

```
0 references | 0 overrides
public function confirmar_cuenta() {
    $token = $this->input->get('token');//Obtenemos el valor del token de la URL
    $activa = $this->Page_model->confirma_cuenta($token);

    if($activa){
        redirect('iniciar-sesion');
    }else {
        $this->load->view('pages/sections/sin-cuenta');
    }
}
```

Ilustración 48 Confirmación de la cuenta

3.3.2 Forms

La primera función que se va a usar, será la responsable para llenar el primer formulario de llenado para la información del participante, tendremos las variables las cuales van a llenar la página con las imágenes, fechas, colores y de más información.

```

0 references | 0 overrides
public function completar_registro_participante() {
    $Info['colores'] = $this->Page_model->consulta_x_id2('48');
    $Info['favicon'] = $this->Page_model->Consulta_Template();
    $Info['Expositor'] = $this->Page_model->consulta_participante($this->uri->segment(2));
    $Info['Estudiante'] = $this->Page_model->consulta_x_id2($Info['Expositor']['id_tipo_registro']);
    $Info['Llenado'] = $this->Page_model->consulta_x_id2('35');
    $Info['Equipo'] = $this->Page_model->consulta_equipo($this->uri->segment(2));
    $Info['Estado'] = $this->Page_model->estado_x_id($Info['Expositor']['estado']);
    $Info['banner'] = $this->Page_model->consulta_x_id2('44');
}

```

Ilustración 49 Información para llenar nuestro primer formulario

Cuando el formulario haya sido llenado por el participante, primero validaremos si el campo de estado es completo, podremos tomar el valor que se haya introducido, pero en caso de que no exista el estado, de hará una consulta a la base de datos para obtener el nombre del estado en base a su id

```

if ($this->input->post('registro-personal')== 'ok-participante'){
    if(!empty($this->input->post('estado'))){
        $estado = $this->input->post('estado');
        $nombre = $this->Page_model->estado_x_id($this->input->post('estado'));
        $nombre_estado = $nombre['nombre'];
    } else {
        $nombre = $this->Page_model->estado_x_nombre($this->input->post('estadoInput'));
        $estado = $nombre['id'];
        $nombre_estado = $this->input->post('estadoInput');
    }
}

```

Ilustración 50 Nombre del estado

Una vez que se ha seleccionado el estado, se hará validación del lado del servidor para los principales datos del participante.

```

$this->form_validation->set_rules('nombre', 'Nombre', 'trim|required|xss_clean');
$this->form_validation->set_rules('apellido', 'Apellido', 'trim|required|xss_clean');
$this->form_validation->set_rules('correo', 'Correo', 'trim|required|xss_clean');
$this->form_validation->set_rules('nacimiento', 'Nacimiento', 'trim|required|xss_clean');
$this->form_validation->set_rules('genero', 'Genero', 'trim|required|xss_clean');
$this->form_validation->set_rules('celular', 'Celular', 'trim|required|xss_clean');
$this->form_validation->set_rules('fijo', 'Fijo', 'trim|required|xss_clean');
$this->form_validation->set_rules('cp', 'Codigo Postal', 'trim|required|xss_clean');

```

Ilustración 51 Validación de los campos del formulario

Una vez que se ha hecho la validación de los campos, con los valores que hayan sido introducidos, se creará un arreglo el cual será enviado al modelo para poder actualizar los

datos que hayan sido colocados por el participante. Se tomará como id el segmento 2 de la URL para actualizar los datos.

```
$data = ['nombre' => $this->input->post('nombre'),
        'apellido' => $this->input->post('apellido'),
        'correo' => $this->input->post('correo'),
        'nacimiento' => $this->input->post('nacimiento'),
        'genero' => $this->input->post('genero'),
        'celular' => $this->input->post('celular'),
        'fijo' => $this->input->post('fijo'),
        'cp' => $this->input->post('cp'),
        'ciudad' => $this->input->post('ciudad'),
        'estado' => $estado,
        'nombre_estado' => $nombre_estado];

$this->Page_model->participante_update($this->uri->segment(2), $data);
```

Ilustración 52 Actualizado de los datos del participante

En caso de que el tipo de registro haya sido para equipo se deberán llenar 4 campos para los diversos participantes del equipo (nombre, apellidos, celular y carrera). En caso de que no se tenga registro para los participantes del equipo, creará los datos, pero si ya existen participantes solo hará la actualización de la información

```
if($Info['Expositor']['id_tipo_registro']== '36'){
    $busca_equipo = $this->Page_model->busca_equipo($this->uri->segment(2));
    $equipo = ['nombre_int1' => $this->input->post('nombre_int1'),
              'apellido_int1' => $this->input->post('apellido_int1'),
              'celular_int1' => $this->input->post('celular_int1'),
              'carrera_int1' => $this->input->post('carrera_int1'),
              'nombre_int2' => $this->input->post('nombre_int2'),
              'apellido_int2' => $this->input->post('apellido_int2'),
              'celular_int2' => $this->input->post('celular_int2'),
              'carrera_int2' => $this->input->post('carrera_int2'),
              'nombre_int3' => $this->input->post('nombre_int3'),
              'apellido_int3' => $this->input->post('apellido_int3'),
              'celular_int3' => $this->input->post('celular_int3'),
              'carrera_int3' => $this->input->post('carrera_int3'),
              'nombre_int4' => $this->input->post('nombre_int4'),
              'apellido_int4' => $this->input->post('apellido_int4'),
              'celular_int4' => $this->input->post('celular_int4'),
              'carrera_int4' => $this->input->post('carrera_int4'),
              'id_representante' => $this->uri->segment(2)];

    if($busca_equipo){
        $this->Page_model->actualiza_equipo($this->uri->segment(2),$equipo);
    } else{
        $this->Page_model->inserta_participantes($equipo);
    }
}
```

Ilustración 53 Datos del equipo del participante

Cuando la actualización haya sido completada, se nos redirigirá a la sección de documentación, en donde serán subidos documentos de identificación del participante y de sus compañeros en caso de que haya sido seleccionado el registro por equipo

```
redirect('documentacion/' . $this->uri->segment(2));
```

Dentro de la función para la carga de documentación, primero se validará que el formulario haya sido enviado con el método “Post” y que el participante haya sido estudiante en cualquiera de sus ramas (individual o por equipo). Porque para los estudiantes, se deberán cargar dos documentos, el comprobante de estudios y el INE. En caso de que el participante no haya subido algún documento se tomará como valor el nombre del archivo anterior.

```
if ($this->input->post('documentacion')=='ok-participante'){  
    if($Info['Expositor']['id_tipo_registro'] == '23' || $Info['Expositor']['id_tipo_registro'] == '36'){  
        if(empty($FILES['imagen1']['name'])){  
            $file_ext = pathinfo($FILES['imagen1']['name'], PATHINFO_EXTENSION);  
            if($file_ext == 'pdf'){  
                $archivo1=$this->storage->setPdf($FILES['imagen1']['name'],'file',$this->input->post('archivo1'),'imagen1');  
            } else {  
                $archivo1=$this->storage->setImage($FILES['imagen1']['name'],'identificaciones','file',$this->input->post('archivo1'),'imagen1');  
            }  
        } else {  
            $archivo1= $this->input->post('archivo1');  
        }  
    }  
}
```

Ilustración 54 Carga de documentos escolares

En el caso de que el participante sea parte de la categoría profesional, se le estarán solicitando tres archivos diferentes, los cuales serán: Identificación oficial, comprobante de estudios terminados y comprobante de pago, al igual que con los estudiantes, si alguno de los tres documentos no fue modificado, el archivo seguirá siendo el mismo

```

} else if($Info['Expositor']['id_tipo_registro'] == '24' || $Info['Expositor']['id_tipo_registro'] == '37'){

    if(!empty($_FILES['image1']['name'])){
        $file_ext = pathinfo($_FILES['image1']['name'], PATHINFO_EXTENSION);
        if($file_ext == 'pdf'){
            $archivo1=$this->storage->setPdf($_FILES['image1']['name'],'file',$this->input->post('archivo1'),'image1');
        } else {
            $archivo1=$this->storage->setImage($_FILES['image1']['name'],'identificaciones','file',$this->input->post('archivo1'),'image1');
        }
    }
    $archivo1= $this->input->post('archivo1');
}

    if(!empty($_FILES['image2']['name'])){
        $completo = true;
        $file_ext = pathinfo($_FILES['image2']['name'], PATHINFO_EXTENSION);
        if($file_ext == 'pdf'){
            $pago=$this->storage->setPdf($_FILES['image2']['name'],'file',$this->input->post('pago'),'image2');
        } else {
            $pago=$this->storage->setImage($_FILES['image2']['name'],'identificaciones','file',$this->input->post('pago'),'image2');
        }
    }
    $pago= $this->input->post('archivo2');
    $completo = $this->input->post('completo');
}

    if(!empty($_FILES['image3']['name'])){
        $file_ext = pathinfo($_FILES['image3']['name'], PATHINFO_EXTENSION);
        if($file_ext == 'pdf'){
            $archivo2=$this->storage->setPdf($_FILES['image3']['name'],'file',$this->input->post('archivo2'),'image3');
        } else {
            $archivo2=$this->storage->setImage($_FILES['image3']['name'],'identificaciones','file',$this->input->post('archivo2'),'image3');
        }
    }
    $archivo2= $this->input->post('archivo2');
    $completo = $this->input->post('completo');
}
}

```

Ilustración 55 Validación de archivos profesionales

Por último, tenemos al participante de la categoría Elektra al cual solo se le estará pidiendo su identificación oficial.

```

} else if($Info['Expositor']['id_tipo_registro'] == '25' || $Info['Expositor']['id_tipo_registro'] == '38'){

    if(!empty($_FILES['image1']['name'])){
        $completo = true;
        $file_ext = pathinfo($_FILES['image1']['name'], PATHINFO_EXTENSION);
        if($file_ext == 'pdf'){
            $archivo1=$this->storage->setPdf($_FILES['image1']['name'],'file',$this->input->post('archivo1'),'image1');
        } else {
            $archivo1=$this->storage->setImage($_FILES['image1']['name'],'identificaciones','file',$this->input->post('archivo1'),'image1');
        }
    }
    $archivo1= $this->input->post('archivo1');
    $completo = $this->input->post('completo');
}

    $data = array (
        'archivo1' => $archivo1,
        'completo' => $completo
    );
}

```

Ilustración 56 Documento de identificación para categoría Elektra

Cuando el usuario haya subido sus archivos, dependiendo la categoría perteneciente, se le redirigirá a otra vista con un proceso determinado, si es estudiante, se le estará pidiendo datos de la universidad, pero si es profesional, se le solicitará información de la empresa en la que trabaja.

Para los estudiantes, se les estará pidiendo el nombre, ciudad y estado de la escuela y nombre de la carrera en la que se encuentran estudiando. Primero se validará que los datos estén llegando mediante Post. Después se hará la validación de que tipo de dato está llegando para saber la universidad del participante.

```
if ($this->input->post('registro-universidad')=='ok-participante'){  
    if($this->input->post('universidad') != '999'){  
        $universidad = $this->Page_model->universidad_x_id($this->input->post('universidad'));  
    } else {  
        $universidad['nombre'] = $this->input->post('nombre_universidad');  
    }  
}
```

Ilustración 57 Validar formulario

Cuando se haya determinado la escuela, se validará el nombre de la carrera. Una vez que se han hecho esas dos primeras validaciones, se harán las respectivas validaciones de los demás campos para validar el llenado correcto.

```
if($this->input->post('carrera') != '99'){  
    $carrera = $this->Page_model->carrera_x_id($this->input->post('carrera'));  
} else {  
    $carrera['nombre'] = $this->input->post('nombre_carrera');  
}  
  
$this->form_validation->set_rules('universidad', 'Universidad', 'trim|required|xss_clean');  
$this->form_validation->set_rules('campus', 'Campus', 'trim|required|xss_clean');  
$this->form_validation->set_rules('carrera', 'Carrera', 'trim|required|xss_clean');  
$this->form_validation->set_rules('estado_universidad', 'Estado universidad', 'trim|required|xss_clean');  
$this->form_validation->set_rules('ciudad_universidad', 'Ciudad universidad', 'trim|required|xss_clean');
```

Ilustración 58 Validar los demás campos del formulario.

Si las validaciones de la escuela han sido correctas, se creará el arreglo con todos los datos universitarios y serán enviado a la base de datos para que se pueda agregar el registro. Posteriormente se enviará al último apartado del registro, a la sección del proyecto

```

$data = array('universidad' => $this->input->post('universidad'),
    'nombre_universidad' => $universidad['Nombre'],
    'campus' => $this->input->post('campus'),
    'carrera' => $this->input->post('carrera'),
    'nombre_carrera' => $carrera['nombre'],
    'estado_universidad' => $this->input->post('estado_universidad'),
    'ciudad_universidad' => $this->input->post('ciudad_universidad'));

$this->Page_model->participante_update($this->uri->segment(2), $data);
redirect('proyecto/'.$this->uri->segment(2));

```

Ilustración 59 Actualización de los datos universitarios.

Para los participantes profesionales, solo se le estará pidiendo el nombre, ciudad, estado, fundación y segmento al que corresponda la empresa en la cual laboran. Una vez que se valide que los datos han sido llenados, se crea el arreglo para ser enviado a la base de datos.

```

if ($this->input->post('registro-empresa')== 'ok-participante'){
    $this->form_validation->set_rules('nombre_empresa', 'Nombre empresa', 'trim|required|xss_clean');
    $this->form_validation->set_rules('ciudad_empresa', 'Ciudad empresa', 'trim|required|xss_clean');
    $this->form_validation->set_rules('estado_empresa', 'Estado empresa', 'trim|required|xss_clean');
    $this->form_validation->set_rules('fundacion_empresa', 'Fundacion empresa', 'trim|required|xss_clean');
    $this->form_validation->set_rules('segmento', 'Segmento', 'trim|required|xss_clean');

    $data = array('nombre_empresa' => $this->input->post('nombre_empresa'),
        'ciudad_empresa' => $this->input->post('ciudad_empresa'),
        'estado_empresa' => $this->input->post('estado_empresa'),
        'fundacion_empresa' => $this->input->post('fundacion_empresa'),
        'segmento' => $this->input->post('segmento'));

    $this->Page_model->participante_update($this->uri->segment(2), $data);
    redirect('proyecto/'.$this->uri->segment(2));
}

```

Ilustración 60 Actualizar los datos de la empresa del participante.

En nuestra función para los proyectos de los participantes, vamos a aceptar dos documentos por proyecto (Hasta un máximo de 3 proyectos por participante), el participante deberá cargar los renders y memorias conceptuales de los proyectos que deseen cargar. Si solo desea actualizar uno o varios documentos, solo será necesario cargar el archivo deseado.

```

if ($this->input->post('registro-proyecto')=='ok-participante'){
    if(empty($_FILES['image1']['name'])){
        $archivo1_1=$this->storage->setPdf($_FILES['image1']['name'],'file',$this->input->post('archivo1_1old'),'image1');
    } else {
        $archivo1_1 = $this->input->post('archivo1_1');
    }

    if(empty($_FILES['image2']['name'])){
        $archivo2_2=$this->storage->setPdf($_FILES['image2']['name'],'file',$this->input->post('archivo2_2old'),'image2');
    } else {
        $archivo2_2 = $this->input->post('archivo2_2');
    }

    if(empty($_FILES['image3']['name'])){
        $archivo3_1=$this->storage->setPdf($_FILES['image3']['name'],'file',$this->input->post('archivo3_1old'),'image3');
    } else {
        $archivo3_1= $this->input->post('archivo3_1');
    }

    if(empty($_FILES['image4']['name'])){
        $archivo2_2=$this->storage->setPdf($_FILES['image4']['name'],'file',$this->input->post('archivo2_2old'),'image4');
    } else {
        $archivo2_2 = $this->input->post('archivo2_2');
    }

    if(empty($_FILES['image5']['name'])){
        $archivo3_1=$this->storage->setPdf($_FILES['image5']['name'],'file',$this->input->post('archivo3_1old'),'image5');
    } else {
        $archivo3_1 = $this->input->post('archivo3_1');
    }

    if(empty($_FILES['image6']['name'])){
        $archivo3_2=$this->storage->setPdf($_FILES['image6']['name'],'file',$this->input->post('archivo3_2old'),'image6');
    } else {
        $archivo3_2 = $this->input->post('archivo3_2');
    }
}

```

Ilustración 61 Cargar los archivos de los proyectos

Al momento de validar los nombres de los proyectos, se va a crear el arreglo para enviar la información de los proyectos del participante.

```

$this->form_validation->set_rules('proyecto1', 'Proyecto 1', 'trim|required|xss_clean');
$this->form_validation->set_rules('proyecto2', 'Proyecto 2', 'trim|required|xss_clean');
$this->form_validation->set_rules('proyecto3', 'Proyecto 3', 'trim|required|xss_clean');

$data = array('proyecto1' => $this->input->post('proyecto1'),
    'proyecto2' => $this->input->post('proyecto2'),
    'proyecto3' => $this->input->post('proyecto3'),
    'proyecto1_url' => $this->LimpiarTexto($this->input->post('proyecto1')),
    'proyecto2_url' => $this->LimpiarTexto($this->input->post('proyecto2')),
    'proyecto3_url' => $this->LimpiarTexto($this->input->post('proyecto3')),
    'archivo1_1' => $archivo1_1,
    'archivo1_2' => $archivo1_2,
    'archivo2_1' => $archivo2_1,
    'archivo2_2' => $archivo2_2,
    'archivo3_1' => $archivo3_1,
    'archivo3_2' => $archivo3_2,
    'video_proyecto1' => $this->input->post('video_proyecto1'),
    'video_proyecto2' => $this->input->post('video_proyecto2'),
    'video_proyecto3' => $this->input->post('video_proyecto3'));

```

Ilustración 62 Creando arreglo para enviarlo a la base de datos

En caso de que haya sido la primera vez de que el participante ingresó a su cuenta, se le estará enviando un correo de confirmación donde se le informe que ha llenado todo su perfil y que podrá hacer alguna actualización a su información antes de la fecha establecida por los administradores.

```
if($Info['Expositor']['primero'] == true){
    $data = array('primero' => false);
    $Info['logo'] = $this->Page_model->consulta_x_id2('44');
    $correos = $this->Page_model->consulta_x_id2('22');
    $asunto = $correos['subtitle'];
    $para = $Info['Expositor']['correo'];
    $emisor = $correos['title'];
    $copiaOcultaPara = '';
    $mensaje = $this->load->view('emails/correo_notificacion', $Info, TRUE);

    $this->load->library('phpmailer_lib'); // Load PHPMailer Library
    $mail = $this->phpmailer_lib->load(); // PHPMailer object

    // SMTP configuration
    $mail->isSMTP();
    $mail->Host = '*****'; // Host de conexión SMTP
    $mail->SMTPAuth = true;
    $mail->Mailer = "*****";
    $mail->Username = $emisor;
    $mail->Password = '*****';
    $mail->SMTPSecure = '*****';
    $mail->Port = ****;
    $mail->CharSet = 'UTF-8'; // SE AGREGO ESTA LINEA PARA LA CORRECTA CODIFICACIÓN DE LOS TEXTOS
    $mail->setFrom($emisor, "Dimueble | Registro completo");
    //$mail->addReplyTo('info@example.com', 'CodexWorld');
```

Ilustración 63 Envío de correo al usuario confirmando su registro completo

Cuando se mandó el correo, se registran los datos en la base y se le enviará al usuario a la pantalla final donde se le muestra la fecha límite para actualizar datos e ingresar de manera libre a su cuenta.

```
$this->Page_model->participante_update($this->uri->segment(2), $data);
redirect('perfil-completo');
```

Ilustración 64 Actualizar datos del participante

3.4 Modelos

Los modelos son los encargados recolectar la información que se encuentre en la base de datos. Para esta página, se estará usando principalmente el modelo “Page_model”. Esta función nos ayudará para traer contenido de la tabla “ci_content”, estará usando el id

del registro y como discriminante que esté habilitado el contenido. Esta consulta solo nos traerá un registro por llamado.

```
public function consulta_x_id2($data){ // SOLO - U N O
    $this->db->where('id', $data);
    //$this->db->where('type', 'Contenido');
    $this->db->where('enable', 'on');
    $query = $this->db->get('ci_content');
    if($query->num_rows() > 0 )
    {
        foreach ($query->result() as $row){
            $info = array(
                'id' => $row->id,
                'module' => $row->module,
                'section' => $row->section,
                'title' => $row->title,
                'subtitle' => $row->subtitle,
                'subtitle2' => $row->subtitle2,
                'description1' => $row->description1,
                'description2' => $row->description2,
                'description3' => $row->description3,
                'url' => $row->url,
                'webp1' => $row->webp1,
                'txt1' => $row->txt1,
                'txt2' => $row->txt2,
                'txt3' => $row->txt3,
                'txt4' => $row->txt4,
                'image1' => $row->image1,
                'alt1' => $row->alt1,
                'image1_m' => $row->image1_m
            );
        }
        return $info;
    }
}
```

Ilustración 65 Función para traer un registro de ci_content

Para poder darle un llenado a la mayoría de las páginas, se usará “Consulta_Template”, la cual servirá para traer todos los registros que pertenezcan al template del sitio, en este caso se estará devolviendo un array multidimensional, ya que va a tener varios registros de la base de datos.

```

public function Consulta_Template()
{
    $this->db->where('module', 'Template');
    $query = $this->db->get('ci_content');
    if($query->num_rows() > 0 )
    {
        foreach ($query->result() as $row){
            $info[] = array(
                'id' => $row->id,
                'section' => $row->section,
                'title' => $row->title,
                'subtitle' => $row->subtitle,
                'subtitle2' => $row->subtitle2,
                'description1' => $row->description1,
                'description2' => $row->description2,
                'description3' => $row->description3,
                'txt1' => $row->txt1,
                'txt2' => $row->txt2,
                'txt3' => $row->txt3,
                'txt4' => $row->txt4,
                'url' => $row->url,
                'image1' => $row->image1,
                'alt1' => $row->alt1,
                'image2' => $row->image2,
                'alt2' => $row->alt2,
                'image3' => $row->image3,
                'alt3' => $row->alt3,
                'module' => $row->module,
                'webp1' => $row->webp1,
                'webp2' => $row->webp2,
                'webp3' => $row->webp3
            );
        } // return $info;
    }

    $info['webinars'] = $this->consulta_Special('38');

    return $info;
}

```

Ilustración 66 Consulta para el template de la página

Para obtener los datos correspondientes al participante que este realizando su registro, usaremos la función llamada “consulta_participante” la cual está recibiendo un parámetro, el cual será el Id del usuario para poder usar ese dato como discriminante dentro de la tabla “registros_individuales”, aquí se estará devolviendo un array sencillo

```

0 references | 0 overrides
public function consulta_participante($data){
    $this->db->where('Id', $data);
    $query = $this->db->get('registros_individuales');
    if($query->num_rows() > 0 )
    {
        foreach ($query->result() as $row){
            $info = array(
                'Id' => $row->Id,
                'nombre' => $row->nombre,
                'apellido' => $row->apellido,
                'correo' => $row->correo,
                'nacimiento' => $row->nacimiento,
                'id_tipo_registro' => $row->id_tipo_registro,
                'genero' => $row->genero
            );
        } return $info;
    }
}

```

Ilustración 67 Consulta para los participantes.

En caso de que el registro del participante haya sido por equipos, se usará la función “consulta_equipo”, en esta consulta se usará como diferenciador, el id del representante del equipo y hará la consulta a la tabla “registros_equipo” para devolver el registro con los integrantes del equipo.

```

0 references | 0 overrides
public function consulta_equipo($data){
    $this->db->where('id_representante',$data);
    $query = $this->db->get('registros_equipo');
    if($query->num_rows() > 0 )
    {
        foreach ($query->result() as $row){
            $info= array(
                'nombre_int1' => $row->nombre_int1,
                'apellido_int1' => $row->apellido_int1,
                'celular_int1' => $row->celular_int1,
                'carrera_int1' => $row->carrera_int1,
                'archivo1_int1' => $row->archivo1_int1,
                'archivo2_int1' => $row->archivo2_int1,
                'nombre_int2' => $row->nombre_int2,
                'apellido_int2' => $row->apellido_int2,
                'celular_int2' => $row->celular_int2,
                'carrera_int2' => $row->carrera_int2,
                'archivo1_int2' => $row->archivo1_int2,
                'archivo2_int2' => $row->archivo2_int2,
                'nombre_int3' => $row->nombre_int3,
                'apellido_int3' => $row->apellido_int3,
                'celular_int3' => $row->celular_int3,
                'carrera_int3' => $row->carrera_int3,
                'archivo1_int3' => $row->archivo1_int3,
                'archivo2_int3' => $row->archivo2_int3,
                'nombre_int4' => $row->nombre_int4,
                'apellido_int4' => $row->apellido_int4,
                'celular_int4' => $row->celular_int4,
                'carrera_int4' => $row->carrera_int4,
                'archivo1_int4' => $row->archivo1_int4,
                'archivo2_int4' => $row->archivo2_int4,
            );
        } return $info;
    }
}

```

Ilustración 68 Consulta para los integrantes del equipo

La función "estado_x_id" ayudará para hacer la búsqueda mediante el id del estado correspondiente esto cuando el usuario no ingrese el nombre del estado donde reside y ponga el nombre del mismo.

```

public function estado_x_id($id)
{
    $this->db->where('id',$id);
    $query = $this->db->get('estados');
    if($query->num_rows() > 0 ){
        foreach ($query->result() as $row){
            $info = array(
                'id'=>$row->id,
                'nombre'=>$row->nombre);
        }
        return $info;
    }
}

```

Ilustración 69 Consulta para conocer estados mediante un Id

Para actualizar la información de los representantes por equipos, se ocupará la función “actualiza_equipo”, la cual estará recibiendo dos parámetros, el primero será el id del registro para el equipo y el segundo será un arreglo con la toda la información de los participantes.

```

0 references | 0 overrides
public function actualiza_equipo ($id, $data){
    $this->db->where('id_representante', $id);
    $this->db->update('registros_equipo', $data);
}

```

Ilustración 70 Función para actualizar los participantes

“participante_update” hará la misma función que “actualiza_equipo” estará recibiendo el id del participante, junto con toda la información que se actualizará del participante.

```

0 references | 0 overrides
public function participante_update($id, $data){
    $this->db->where('Id', $id);
    $this->db->update('registros_individuales', $data);
}

```

Ilustración 71 Actualizar información del participante

“primer_proyecto” solo estará recibiendo información pertinente del primer proyecto y se tendrán dos funciones adicionales para los otros dos proyectos. Primero, vamos a contar la cantidad de datos que está llegando en \$data, para hacer un ciclo for y recorrer toda la información.

```
0 references | 0 overrides
public function primer_proyecto($data){
    $info = [];
    $cuenta = count($data);
    if($cuenta>0){
        for($i=0;$i<$cuenta;$i++){
            if(!empty($data[$i]['proyecto1_url'])){
                $proyecto1 = $data[$i]['proyecto1_url'];
                $this->db->where('proyecto', $proyecto1);
                $this->db->where('pasa', 'Si');
                $query = $this->db->get('primer_fase');
                if($query->num_rows() > 0 )
                {
                    foreach ($query->result() as $row){
                        $info[$row->id_representante] = $query->num_rows();
                    }
                }
            }
        }
    }
    return $info;
}
```

Esas serán las funciones principales que se usarán dentro del proyecto para el manejo de toda la información de todos los participantes.

CAPITULO IV RESULTADOS

Tenemos nuestra pantalla de inicio, donde se estarán mostrando la edición del concurso, junto con las tres categorías principales para que el participante pueda elegir conforme a su perfil y después tendremos una sección para preguntas frecuentes. En caso de que el participante tenga alguna duda más puntual sobre el proceso podrán enviar un correo con la duda más específica para que pueda ser ayudado por el panel de expertos. Si ya se ha registrado el participante, podrá iniciar sesión directamente desde esta primera pantalla.

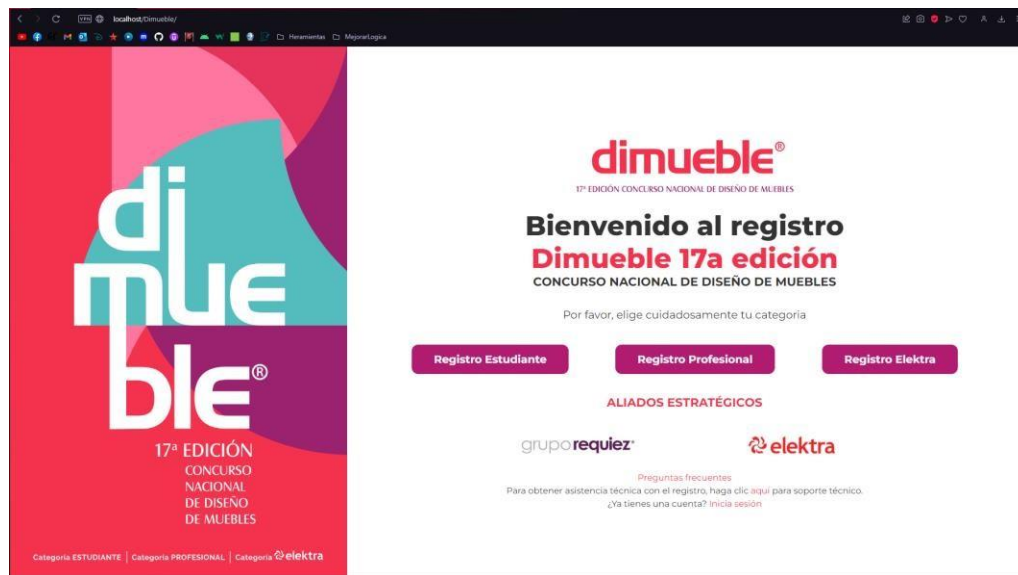


Ilustración 72 Pantalla de bienvenida para el participante

Una vez que se ha seleccionado la categoría correcta, se nos mostrará una segunda pantalla donde se dará la opción para seleccionar el tipo de registro, ya sea para la rama individual. El proceso para cualquier categoría y rama va a ser el mismo, solamente cambiarán algunos campos.

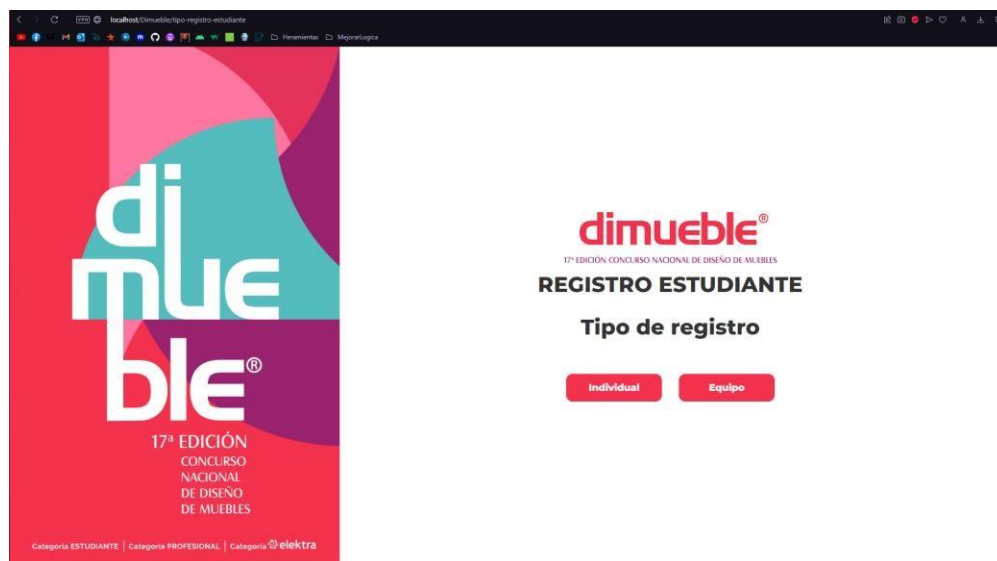


Ilustración 73 Tipo de registro para el participante

Al momento de seleccionar el tipo de registro de la cuenta, se mostrará un formulario sencillo con datos básicos para que se pueda crear la cuenta del participante, se estará solicitando: nombre, apellido, correo y contraseña. En la parte inferior de la página, se mostrará un pequeño aviso de privacidad por parte de la empresa.

Ilustración 74 Formulario para crear cuenta

Se deberá llenar de manera correcta todos los campos correspondientes para que se pueda seguir con el registro, si llegara a ocurrir algún problema con el correo o contraseña de la cuenta, se le estará informado al usuario y poder cambiar el dato que se encuentra mal.

localhost dice
Recuerda que los correos ingresados deben ser los mismos

Registro Estudiante
Es necesario crees una cuenta para realizar tu registro.

Eduardo Mancera
eduardo2@centralinteractiva.com eduardo@centralinteractiva.com
[CAPTCHA] Registrarse

A continuación recibirás una confirmación vía correo electrónico para validar tu cuenta.
Al dar clic sobre el botón "Registrarse", crearás una cuenta en Dimueble. Actualizamos nuestro aviso de privacidad, consúltalo aquí.

AFAMGAL, a través de su encargado de datos personales, con domicilio en Avenida Mariano Otero 133, Colonia Rinconada del Bosque, c.p. 44530, Guadalajara, Jalisco, México, utilizará los datos personales recabados para realizar la prospección y comercialización en físico y/o en línea de memberships de afiliación empresarial; la promoción y venta en físico y/o en línea de los servicios de AFAMGAL, tales como: planeación, organización, promoción y venta de los eventos foráneos EXPO MUEBLE INTERNACIONAL y TECNO MUEBLE INTERNACIONAL, el concurso nacional de diseño de muebles DIMUEBLE, diseño, impresión, promoción y venta de publicidad y suscripciones, de la revista impresa y digital MÓBILAE, la provisión de productos y/o servicios empresariales para el Sector Mueblero; tales como: publicidad impresa y/o digital en la revista especializada, en los directorios empresariales, en el sitio web iFork, y/o en redes sociales; suscripción a la revista y boletín electrónico; envío masivo de correos electrónicos publicitarios; patrocinio de eventos, programas recreativos y sesiones comerciales anuales; y networking; capacitación con cursos, talleres y conferencias; y consultoría estratégica en desarrollo de negocios; conformar y operar la Red Empresarial de Socios de AFAMGAL, con afiliación de empresas, negocios y profesionales del Sector Mueblero; emitir la facturación fiscal; así mismo, realizar mercadotecnia y difusión en físico y/o en línea de sus servicios; enviar correos postales con promocionales, boletines, informes y folletos impresos de los servicios y eventos; enviar correos electrónicos para publicar eventos, cursos y conferencias; cobros, pagos, y para la facturación fiscal.

Ilustración 75 Generar error al crear cuenta

En el momento que todos los datos sean ingresados de manera correcta, se nos dirigirá a la pantalla de agradecimiento donde solo se muestra un texto sencillo. Hasta este momento el participante ya cuenta con una cuenta para el concurso, pero para validar su identidad, se deberá verificar la cuenta de correo que fue ingresada

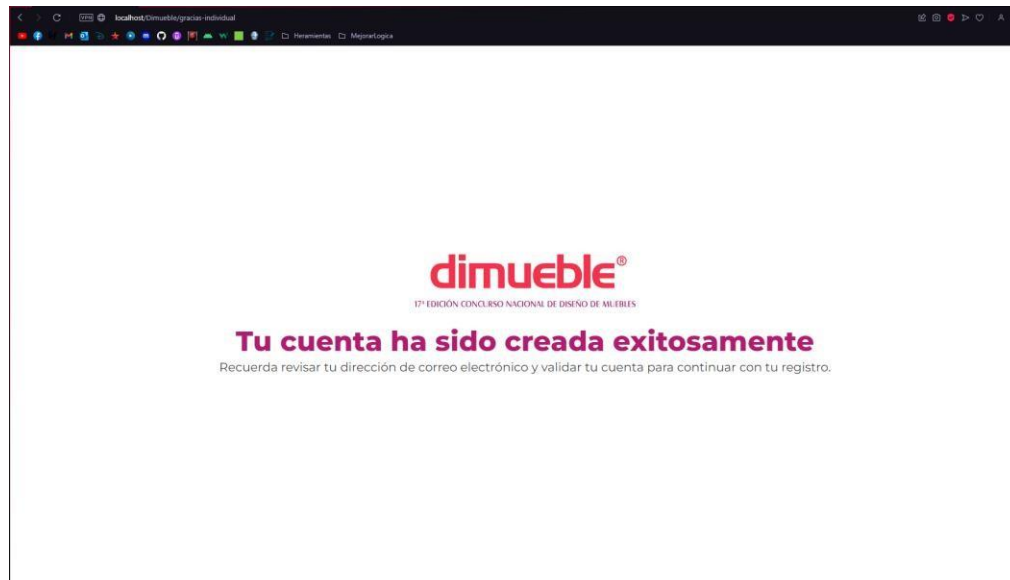


Ilustración 76 Creación de cuenta completada

En base a la dirección de correo que se registró, llegará un correo electrónico para concretar la activación de la cuenta del participante. Bastará con dar clic en el botón que dice

“aquí” para que se mande de nuevo al inicio de sesión. Pero en esta ocasión si se podrá ingresar al panel del participante.



Ilustración 77 Correo para activación de cuenta.

Después de activar la cuenta mediante correo electrónico, se podrá iniciar sesión para empezar a hacer el llenado de la información pertinente para que el perfil sea completo y pueda ser considerado para la primera etapa de evaluación.

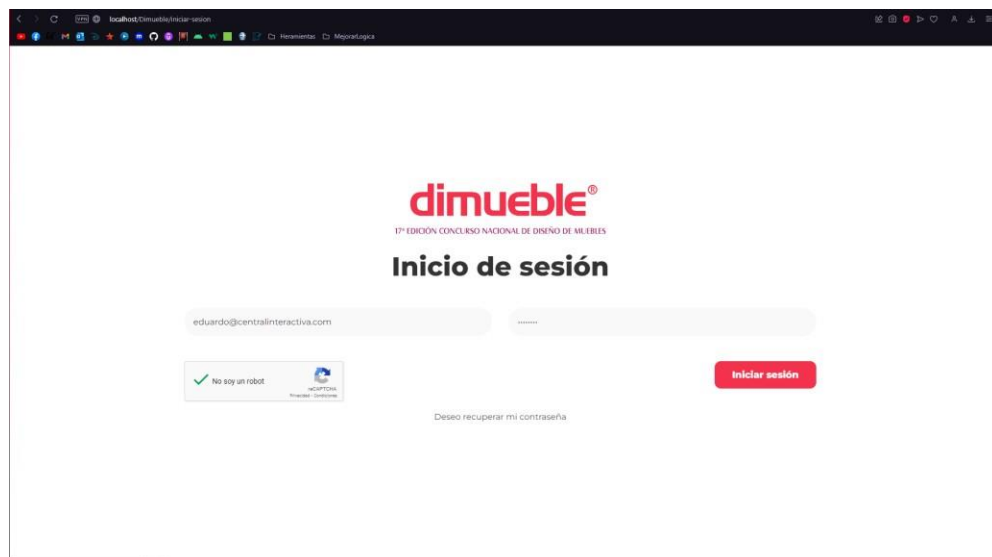


Ilustración 78 Iniciar sesión dentro del portal.

Cuando el inicio de sesión se haya hecho de manera correcta, se nos mostrará la primera parte del registro, donde se estará pidiendo información básica del participante, como lugar donde radica, género o números telefónicos.

The screenshot shows a web browser window with the URL 'localhost:3000/registro-participante/143'. The page features the 'dimueble' logo and the text '17ª EDICIÓN CONCURSO NACIONAL DE DISEÑO DE MUEBLES'. The main heading is 'Registro individual estudiante' with a deadline of 'Fecha limite de llenado: 10/Mayo/2024'. There are four tabs: 'Datos del participante' (selected), 'Documentación del participante', 'Datos de la universidad', and 'Datos del proyecto'. The 'Datos del participante' tab contains the following fields: 'Nombre' (Eduardo), 'Apellidos' (Mancera), 'Correo electrónico' (eduardo@centralinteractiva.com), 'Año de nacimiento' (dd/mm/aaaa), 'Género' (radio buttons for Masculino, Femenino, Otro), 'Teléfono celular', and 'Teléfono fijo'. A note states: 'Los campos marcados con un * deben ser llenados obligatoriamente.'

Ilustración 79 Primer formulario para empezar con el registro de los proyectos.

Una vez que el participante haya llenado sus datos de manera correcta, podrá seguir con el proceso de registro, esta primera parte servirá para la pura información básica. Para la ubicación del participante, solo deberá ingresar su código postal y de manera automática el estado y municipio aparecerán en los campos correspondientes.

This screenshot shows the same 'Datos del participante' form, but with additional fields filled out. The 'Año de nacimiento' is now '19/07/2000'. The 'Teléfono celular' is '5544332211' and 'Teléfono fijo' is '1122334455'. The 'Código Postal' is '54387', and a 'Buscar' button is next to it. The 'Ciudad' dropdown is set to 'Tultepec'. The 'Estado' dropdown is set to 'Estado de México'. A 'Guardar y continuar' button is at the bottom. The same note about mandatory fields is present.

Ilustración 80 Información básica del usuario

La segunda sección del registro, corresponde a los diferentes documentos que tiene el participante para validar matricula en la escuela y de igual manera se le estará solicitando su INE

dimueble®
17ª EDICIÓN CONCURSO NACIONAL DE DISEÑO DE MUEBLES

Registro individual estudiante

Fecha limite de llenado: 10/Mayo/2024

Datos del participante | **Documentación del participante** | Datos de la universidad | Datos del proyecto

Documentación del participante

*Documentación personal Formato pdf/jpg, peso no mayor a 10 MB

Adjunta tu archivo - **JPG o PDF**

o arrastra aquí.

Recuerda nombrar tu archivo:
E.ID_APELLIDCONCURSANTE_NOMBREPROYECTO(S)

Copia digital de identificación oficial INE (ambos lados) o pasaporte.

Legibles

No volteadas

Adjunta tu archivo - **JPG o PDF**

o arrastra aquí.

Recuerda nombrar tu archivo:
E.DU_APELLIDCONCURSANTE_NOMBREPROYECTO(S)

Copia de uno de los siguientes documentos que acredite el año que cursa el participante.

- Carta de estudios expedida por la Universidad.
- Boleta de calificaciones
- Pago de colegiatura.

Ilustración 81 Sección para cargar documentación oficial del participante

En el caso de que el participante haya subido de algún documento mal nombrado, se le estará notificando mediante una alerta que debe cambiar la nomenclatura del nombre del archivo. Ya que el nombre con el que sea guardado es un método para identificar la categoría del participante.

localhost dice
Recuerda usar el nombre adecuado para este archivo

Aceptar

Datos del participante | **Documentación del participante** | Datos de la universidad | Datos del proyecto

Documentación del participante

*Documentación personal Formato pdf/jpg, peso no mayor a 10 MB

Adjunta tu archivo - **JPG o PDF**

E.ID_montesdiaz-silakoa.pdf

Recuerda nombrar tu archivo:
E.ID_APELLIDCONCURSANTE_NOMBREPROYECTO(S)

Adjunta tu archivo - **JPG o PDF**

o arrastra aquí.

Recuerda nombrar tu archivo:
E.DU_APELLIDCONCURSANTE_NOMBREPROYECTO(S)

Ilustración 82 Carga de archivo con nombre erróneo

Una vez que los dos archivos han sido cargados con el nombre correcto, se le estarán mostrando mediante una previsualización al participante para que pueda validar que los archivos que está cargando sean los correctos

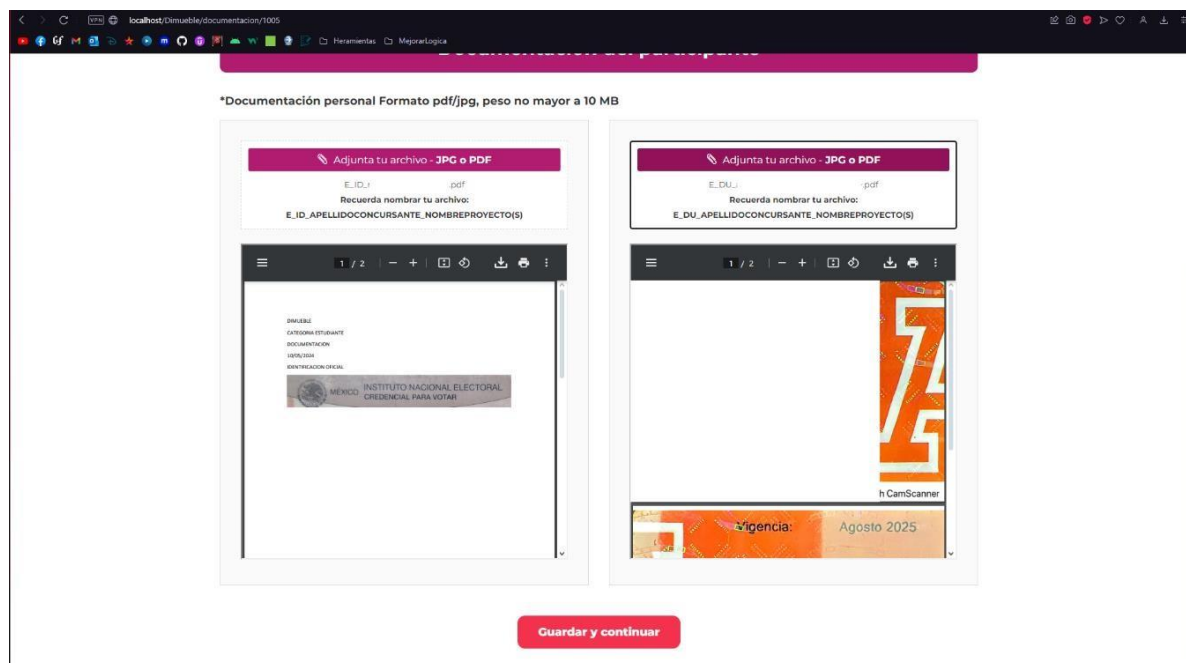


Ilustración 83 Previsualización de los archivos cargados

La siguiente sección será la correspondiente para la información universitaria del participante de la categoría estudiante, los campos que se llenarán estarán limitados a la información que sea ingresada en el administrador, pero en caso de que alguna universidad o plantel no se encuentre registrado, se mostrará un campo nuevo para agregar el campus o universidad.

dimueble®
17ª EDICIÓN CONCURSO NACIONAL DE DISEÑO DE MUEBLES

Registro individual estudiante

Fecha limite de llenado: 10/Mayo/2024

Datos del participante | Documentación del participante | **Datos de la universidad** | Datos del proyecto

Datos de la universidad

Los campos marcados con un * deben ser llenados obligatoriamente.

*Universidad
Universidad de Guadalajara

*Campus
Centro Universitario de Arte, Arquitectura y Diseño

*Carrera
Arquitectura o afines

*Ciudad de la Universidad
Guadalajara

*Estado
Jalisco

Guardar y continuar

Ilustración 84 Universidad registrada en la base de datos

Fecha limite de llenado: 10/Mayo/2024

Datos del participante | Documentación del participante | **Datos de la universidad** | Datos del proyecto

Datos de la universidad

Los campos marcados con un * deben ser llenados obligatoriamente.

*Universidad
Otra

Tesco

*Campus
Otro

Coacalco

*Carrera
Otro

Tics

*Ciudad de la Universidad
Coacalco

*Estado
México

Guardar y continuar

Ilustración 85 Universidad sin registro en el concurso

Cuando la información de la escuela del participante haya sido completada, podrá seguir con el apartado más importante, que es el registro de sus proyectos correspondientes. Cada usuario tendrá hasta un máximo de 3 proyectos para que pueda participar en el registro.

Cada uno de ellos deberá tener archivos de memoria conceptual y renders, los cuales deberán ser cargados en formato pdf

Registro individual estudiante
Fecha limite de llenado: 10/Mayo/2024

Datos del participante | Documentación del participante | Datos de la universidad | **Datos del proyecto**

Datos del proyecto

En caso de que no tengas a la mano los documentos a subir, deberás cerrar sesión e ingresar nuevamente a tu cuenta cuando esten listos.

DESCARGA AQUÍ LOS FORMATOS PARA RENDERS Y MEMORIA CONCEPTUAL

*Los renders deben contener 3 laminas de tu proyecto en formato pdf, peso no mayor a 10 MB.

*Nombre del proyecto l:

Adjunta renders en formato - PDF
E.R.,renders.pdf
Usar la siguiente estructura:
E.R.,NOMBRE_DEL_PROYECTO

Adjunta memoria conceptual en formato - PDF
E.MC,memy.pdf
Usar la siguiente estructura:
E.MC.,NOMBRE_DEL_PROYECTO

Ilustración 86 Carga de los proyectos del participante

En el momento que el participante haya llenado correctamente la información de su proyecto registrado, se le enviará a una última venta donde se le notificará que ha completado su registro y se le muestra la fecha límite para poder hacer actualización a su información.

Haz completado el registro en nuestro portal.
Recuerda que puedes modifica tu información hasta la fecha: 10/Mayo/2024

Guardar y cerrar sesión

Ilustración 87 Registro completo

Para la categoría profesional, se tendrá un apartado diferente al de estudiante, en el cual se tendrá que registrar información básica de la empresa en la cual labora el participante, esto es con el fin de llevar un control más específico sobre las empresas que participan en el concurso.

Ilustración 88 Ingreso de información de la empresa

Dentro de la categoría “Elektra” solo se estarán mostrando 3 apartados, el único que cambia con respecto a los otros dos tipos de registro es el de “Documentación” ya que en esta sección solo se le pedirá el documento de identificación al usuario

Ilustración 89 Documentación del participante

En el caso de que el participante haya olvidado o perdido su contraseña, podrá hacer un restablecimiento de la misma, dentro de la sección para recuperarla se le pedirá el correo con el cual se creó la cuenta, ya que éste será el campo para identificar el registro dentro de la base de datos.

A screenshot of a web browser showing the password recovery page for 'dimueble'. The page has a white background with the 'dimueble' logo in red at the top center. Below the logo, it says '17ª EDICIÓN CONCURSO NACIONAL DE DISEÑO DE MUEBLES'. The main heading is 'Recuperar contraseña' in bold black text. There is a text input field containing the email 'eduardo@centralinteractiva.com'. Below the input field is a red button with the text 'Recuperar' in white.

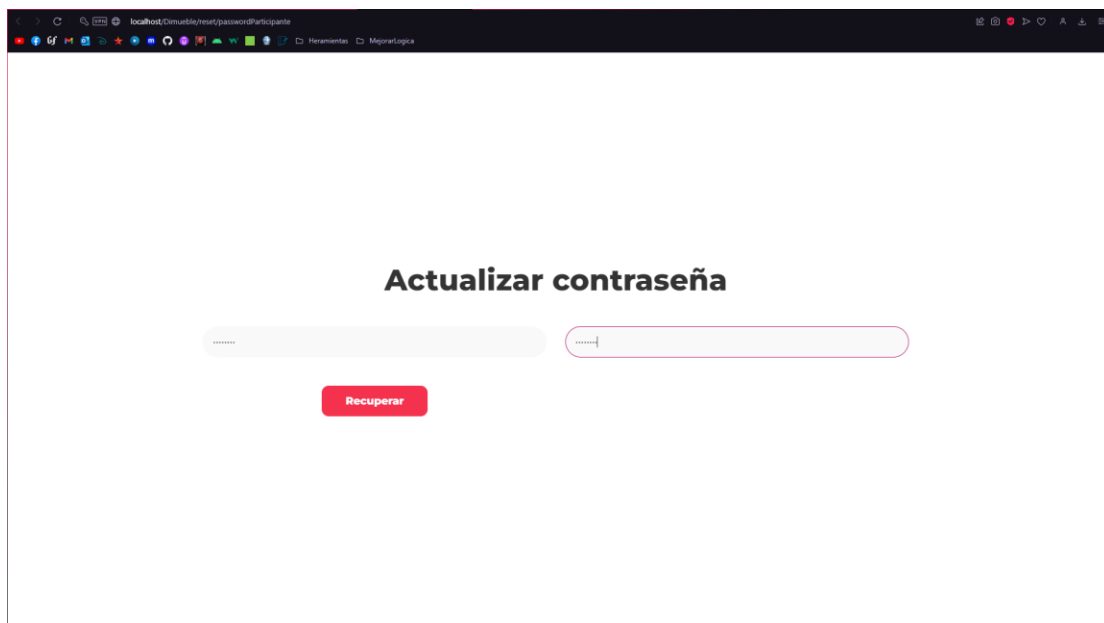
Ilustración 90 Reestablecer contraseña del registro

En caso de que el correo haya sido el correcto, a la misma dirección se le estará enviando un correo de restablecimiento para que pueda acceder de nuevo al portal a cambiar su acceso.

A screenshot of an email interface showing a password reset email from 'dimueble'. The email header includes 'Url de reseteo de contraseña' with a yellow 'Externo' tag and a 'Recibidos x' button. The sender is 'Restablecimiento de contraseña <no-reply@paginadeprueba.com.mx>' with a blue profile icon. The email body features the 'dimueble' logo and the text '17ª EDICIÓN CONCURSO NACIONAL DE DISEÑO DE MUEBLES'. The main heading is 'Correo de recuperación'. Below it, the text says 'Favor de ingresar al siguiente link para reestablecer tu contraseña'. The link for resetting the password is provided: 'http://localhost/Dimueble/reset/passwordParticipante?hash=34ee90de5df980b511f2157c21f679fe0f4c778f784e7b5da392d0d7619ff7b6'. At the bottom, there are two buttons: 'Responder' with a left arrow and 'Reenviar' with a right arrow.

Ilustración 91 Correo de restablecimiento

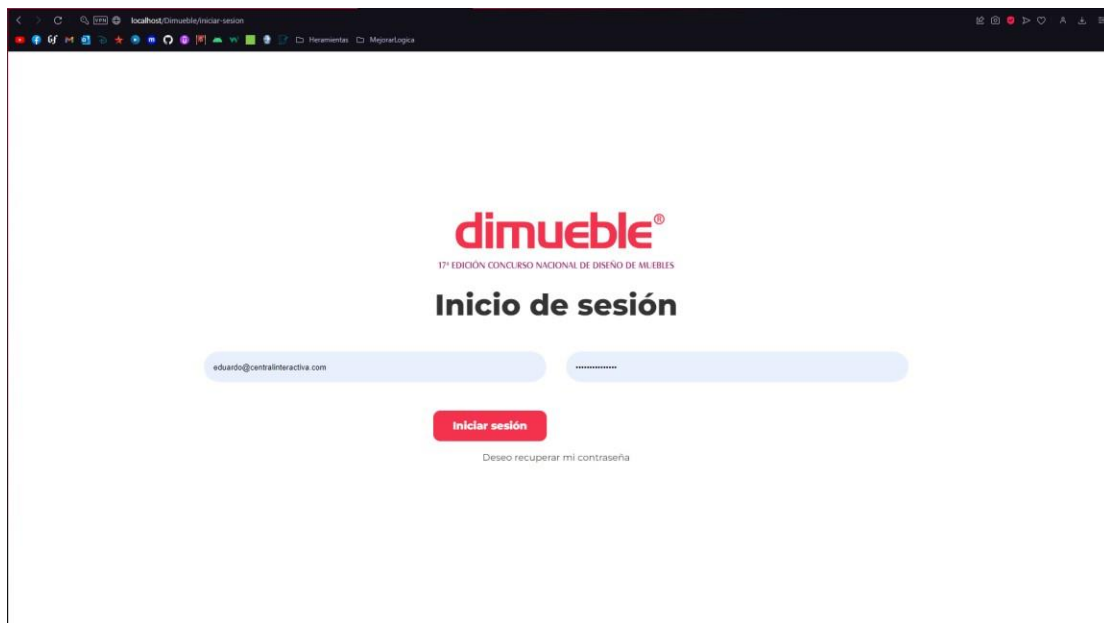
Al momento de ingresar al link, se le redirigirá al usuario a una nueva ventana con dos campos para hacer la actualización de su contraseña



The screenshot shows a web browser window with the address bar displaying 'localhost:3000/actualizar-contrasena'. The page has a white background with a thin red border. In the center, the title 'Actualizar contraseña' is displayed in bold black text. Below the title are two input fields: the first is a light gray rounded rectangle containing a series of dots, and the second is a white rounded rectangle with a red border also containing dots. Below these fields is a red button with the white text 'Recuperar'.

Ilustración 92 Nueva contraseña

Cuando se dé clic en el botón recuperar, se está haciendo el cambio de contraseña y se enviará a la página de inicio de sesión para que pueda seguir con su registro.



The screenshot shows a web browser window with the address bar displaying 'localhost:3000/inicio-sesion'. The page has a white background with a thin red border. In the center, the 'dimueble®' logo is displayed in red, with the text '17ª EDICIÓN CONCURSO NACIONAL DE DISEÑO DE MUEBLES' in small black text below it. Below the logo, the title 'Inicio de sesión' is displayed in bold black text. There are two input fields: the first is a light blue rounded rectangle containing the email 'eduardo@centralinteractiva.com', and the second is a light blue rounded rectangle containing a series of dots. Below these fields is a red button with the white text 'Iniciar sesión'. At the bottom, there is a link in small black text that says 'Deseo recuperar mi contraseña'.

Ilustración 93 Ingreso con la nueva contraseña

CONCLUSIONES

Toda la información expuesta en este informe, tuvo el propósito de exponer todo el conocimiento adquirido a lo largo de un año laboral dentro de la empresa “Sienois” desarrollando la plataforma para el registro “Dimueble”, llevando a cabo practicas de desarrollo de bases de datos, programación web con el lenguaje php, JavaScript y CSS. Además de aplicar los conocimientos bases que se adquirieron a lo largo de la carrera de Tecnologías de la información y comunicaciones.

Debido a la constante actualización de las tecnologías y medios de comunicación, hoy en día es de suma importancia, que todas las empresas del mundo tengan algún medio electrónico de información para llegar de una manera mas fácil y eficiente a todo su publico objetivo, ya que a través de una página web, la difusión de información puede ser realizada de una manera mas rápida y con la seguridad que podrá ser vista en cualquier parte del mundo.

La carrera de Tecnologías de la Información y Comunicaciones, actualmente está teniendo un crecimiento exponencial debido a que las personas egresadas de esta carrera pueden desarrollar diversas actividades dentro de una misma empresa, como mantenimiento a los equipos de cómputo, desarrollos de aplicaciones web o móviles, mantenimiento a servidores o redes, entre otras actividades.

En la empresa “Sienois” adquirí conocimiento fundamental para desarrollo de aplicaciones web, con estilos responsivos y que la experiencia de usuario pueda ser la mejor, aparte de trabajar con diversas tecnologías para poder adaptarme a las diferentes necesidades de los clientes, ya que, al ser una empresa de desarrollo, existen empresas que no se dedican al ámbito de las exposiciones, si no que solo quieren su página web informativa o desea, el desarrollo de un eCommerce completo.

Al ingresar al área de back-end tuve la responsabilidad de realizar proyectos desde 0, partiendo desde la idea principal del cliente y aplicando mejorar o editando funcionalidades para que pudiera llevarse acabo la página o el sistema que se deseaba desarrollar. En algunos casos, el cliente llega con necesidades especificas como lo puede el framework de desarrollo, servidor en el cual desplegar la aplicación o una renovación a un sitio ya existente, en esos

casos, nosotros como programadores debemos adaptarnos al 100% de las necesidades presentadas.

En lo personal, me siento muy contento con los resultados obtenidos a lo largo de este primer año laboral, pudiendo conocer más a fondo mis cualidades y virtudes, pero también he podido trabajar en mis debilidades, teniendo la oportunidad de crecer como profesionista, alumno e hijo. Sin duda alguna, el primer año dentro de cualquier empresa puede ser el más difícil de todos, porque estas entrando a un nuevo mundo y más aun si se es recién egresado ya que hoy en día, las empresas quieren chicos jóvenes con experiencia y esto puede provocar problemas para la mayoría de egresados a nivel nacional.

Hace un año que he terminado mis estudios universitarios y puedo agradecer a todos mis profesores por su conocimiento y apoyo a lo largo de estos 4 años. Los cuales se seguirán viendo reflejados a lo largo de toda mi vida laboral, ya que gracias a ellos los fundamentos que tengo dentro de mi carrera son bastante solidos y esto me ayudara a seguir creciendo de manera profesional. Es importante mencionar que jamás se es demasiado tarde para aprender un lenguaje nuevo, una nueva rama para estudiar o salir de nuestra zona de confort, lo importante de la vida es arriesgarse y no tener miedo a lograr cosas nuevas.

COMPETENCIAS DESARROLLADAS

Fundamentos de base de datos

- Habilidad para diseñar y crear una base de datos desde 0, utilizando el sistema SQL, permitiendo almacenamiento de grandes cantidades de datos
- Optimización de consultas para gestión de información precisa

Desarrollo web

- Dominar tecnologías como HTML, CSS Y JavaScript para crear una interfaz intuitiva y funcional para el usuario
- Conocimiento en el desarrollo de servidores utilizando lenguajes como PHP para poder manejar la lógica del sitio.

Seguridad web

- Conocimiento en diversas técnicas para la encriptación de datos, autenticación y autorización de los datos del usuario.
- Implementación de buenas practicas para la seguridad y evitar inyecciones SQL, XSS O CSRF

Diseño de experiencia de usuario

- Capacidad para desarrollar diversas interfaces intuitivas y amigables que permitan una navegación fluida.

Administrador de sistema

- Capacidad para configurar y mantener el servidor en el cual va a vivir el sitio, asegurando la disponibilidad de este.

BIBLIOGRAFÍA

¿Qué es JavaScript? 02 de Agosto de 2023. https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/What_is_JavaScript.

¿Qué es Python? 31 de Julio de 2022. <https://aws.amazon.com/es/what-is/python/>.

Arias, Miguel A. *Introducción a PHP*. IT Campus Academy., 2013.

Ayoze Castillo, Alberto. *Curso de Programación Web: JavaScript, Ajax y jQuery*. IT Campus Academy, 2017.

Carles Bellver, Jordi Adell. «La internet como telaraña: el World Wide Web.» Castellón, 1995.

Ceballos Sierra, Javier. *Enciclopedia del lenguaje C++*. México: Alfaomega, 2009.

Compilador e intérprete: definición y diferencias. 16 de Junio de 2020. <https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/compilador-e-interprete/>.

Dimes, Troy. *Programación en C# para Principiantes*. Babelcube Inc, 2016.

Eguíluz Perez , Javier. «Introduccion a JavaScript.» 2019.

Marqués, Mercedes. *Bases de datos*. Castellón de la Plana: Universitat Jaume I. Servei, 2011.

Montano, Joaquin. *Lifeder*. 24 de Julio de 2020. <https://www.lifeder.com/historia-tics/>.

Sain, Gustavo. *Historia de Internet (I)*. Buenos Aires: Revista pensamiento legal, 2015.

Telefónica. *Telefónica*. 06 de Septiembre de 2023. <https://www.telefonica.com/es/sala-comunicacion/blog/historia-internet-como-nacio-evolucion/>.

Tripletten. *Qué es Java: todo lo que un principiante necesita saber sobre este lenguaje de programación*. 17 de Enero de 2024. <https://tripletten.mx/blog/que-es-java-y-por-que-la-necesito/>.

ANEXOS

Primer recibo de nomina

Recibo de pago de nómina

Emisor

RFC:	SIE171003FK6	Folio fiscal:	0F6931A5-6887-4E78-A291-3D01C026559A
Nombre:	SIENCOIS	No. de serie del CSD:	00001000000510565578
Registro patronal:	C281567510	Lugar, fecha y hora de emisión:	11560 2023-05-12 14:02:41
Régimen fiscal:	General de Ley Personas Morales	Efecto del comprobante:	Nómina
Folio:	001-9-1	RFC Patrón Origen:	
		Serie:	N

Receptor

Nombre:	EDUARDO MANCERA NIETO	No. de Seguridad Social:	01876305952
RFC:	MANE0007194VA	CURP:	MANE000719HMCNTDA2
No. Empleado:	000016	Antigüedad:	P4W
Riesgo puesto:	Clase I	Tipo contrato:	Contrato de trabajo por tiempo determinado
Fecha de inicio relación laboral:	2023-04-16	Periodicidad de pago:	Quincenal
Régimen de contratación:	Sueldos (Incluye ingresos señalados en la fracción I del artículo 94 de LISR)	Salario diario:	219.51
Tipo de jornada:	Diurna	Salario base:	219.51
Clave Entidad Federativa:	MEX	Sindicalizado:	No
Uso CFDI:	Nómina		
Código postal del receptor:	54987		
Régimen fiscal del receptor:	Sueldos y Salarios e Ingresos Asimilados a Salarios		

Datos Generales

Tipo nómina:	Nómina ordinaria	No. de días pagados:	15.000
Fecha pago:	2023-05-15	Fecha final de pago:	2023-05-15
Fecha Inicial de pago:	2023-05-01	Método de pago:	Pago en una sola exhibición
Forma pago:			
Moneda:	Peso Mexicano		
Exportación:	No aplica		

Concepto

Cve del producto/servicio	No. identificación	Cantidad	Clave unidad	Unidad	Descripción	Valor unitario	Importe	Descuento	Objeto impuesto
84111505		1	ACT		Pago de nómina	3138.00	3138.00	138.03	No objeto de impuesto

Percepciones

Tipo de percepción	Clave	Concepto	Importe gravado	Importe exento
Sueldos, Salarios Rayas y Jornales	001	Sueldos y salarios	3138.00	0.00
Total Percepciones			\$ 3,138.00	\$ 0.00

Total percepciones

Total sueldos	Total exento	Total gravado
3138.00	0.00	3138.00

Este documento es una representación impresa de un CFDI

Página 1 de 2

Deducciones

Tipo de deducción	Clave	Concepto	Importe
ISR	098	Retención ISR	59.83
Seguridad social	099	Cuota IMSS	78.20

Total Deducciones **\$138.03**

Total deducciones

Total otras deducciones	Total impuestos retenidos
78.20	59.83

Otros pagos

Tipo otro pago	Clave	Concepto	Importe
Subsidio para el empleo (efectivamente entregado al trabajador).	002	Subsidio para el empleo	0.00

Total Otros Pagos: **\$ 0.00**

Subsidio al empleo

Subsidio causado
0.00

Totales Nómina

Total percepciones	Total otros pagos	Total deducciones
3138.00	0.00	138.03

IMPORTE NETO **\$ 2,999.97**

Sello digital del CFDI:

TsKk6ufb2FrQRHZksMAxnW5cT0A7TEuB8LYxwlrB7CZsvol5ywOszFBJRKEdX7bASVSoVvANc1XZmG8OhYh9ZbWb8OT6cEh87qAL2CMHXcuIXRr7vFFqPBL6ykvCZzShjQy2gSiw
ea+c13CG9um1U+wowxFUYWXcneoERoMZ891VrE25H8dxc5pPcsglStlQUyG2Uaofn+LxVmapr92lxxsqjwWfctB9k7X1r2qu6wsQCKLXy86aO3AfmQe67Z+vUPLJ2oUuJhMuF0+8T6+6
Y44GnStsokQxcg6q+ZSIA1wik9sk9wWbKkzEWieAlHezxSBgtA==

Sello digital del SAT:

xjYv51rR3JZkru2TT393XCP0GMBWIOFq/QN9kyNj8xWhCAAnKkRtm4rzelpmoBa3zrieCduc9Jy276KhvCsbQOvzT8mYO/BCKFK0ACIKQsc7vLDy7zsVW+LGPTGF1WPBLazY4x8+Is6e4+T
01khv5Ce0uYwF+ENN2SHX4ageD4a2sahziPV+AwRd5gMvWb6L0kNldGd7IRGhNqNBIQb9b6ew+Jbhz4IfUCKusn00DRyzN3vOYFWWh2yFB652PQOumltzluCKuFACgbApwbs6kfr05
+s2nJ2CMnIEws07Fy6E8VBMuPFUihvUDgtpkuu7xdKpDc4sbzx8g==

**Cadena Original del complemento de certificación digital del SAT:**

||1.1|JOF6931A5-6887-4E78-A291-3D01C026559A|2023-05-12T14:02:42|MAS980812UK1|TsKk6ufb2FrQRHZksMAxnW5cT0A7TEuB8LYxwlrB7CZsvol5
ywOszFBJRKEdX7bASVSoVvANc1XZmG8OhYh9ZbWb8OT6cEh87qAL2CMHXcuIXRr7vFFqPBL6ykvCZzShjQy2gSiw
ea+c13CG9um1U+wowxFUYWXcneoERoMZ891VrE25H8dxc5pPcsglStlQUyG2Uaofn+LxVmapr92lxxsqjwWfctB9k7X1r2qu6wsQCKLXy86aO3AfmQe67Z+vUPLJ2oUuJhMuF0+8T6+6
Y44GnStsokQxcg6q+ZSIA1wik9sk9wWbKkzEWieAlHezxSBgtA==|00001000000506315610|

RFC del proveedor de certificación: MAS980812UK1 **Fecha y hora de certificación:** 2023-05-12 14:02:42

No. de serie del certificado SAT 00001000000506315610



CFDI NÓMINA

Versión n mina: 1.2 Tipo n mina: O=Ordinaria

Recibo: N001-9-6 Nómina: 001 Periodo: 9

Exportación: 01 No aplica

SIENNOIS

RFC: SIE171003FK6 Registro Patronal: C281567510 Régimen fiscal: 601-General de Ley Personas Morales

Av. Presidente Masaryk 61, Col. Polanco, CP 11560

Ciudad de México

No. Trab.:	000016	Entidad federativa:	MEX
Nombre:	EDUARDO MANCERA NIETO		
RFC:	MONE0007194VA		
Regimen fiscal:	605		
C.P. domicilio fiscal:	54987		
CURP:	MANE000719HMCNTDA2		
No. IMSS:	01-87-63-0595-2		
Inicio Rel. Lab.:	16/04/2023		
Antigüedad:	P56W		
Salario diario:	295.79	Salario diario int:	310

Deplo.:	Sistemas	
Puesto:		
Días trabajados:	15.00	
Periodicidad de pago:	04	
Periodo del:	01/05/2024	al: 15/05/2024
Fecha pago:	15/05/2024	
Tipo Régimen:	02	
Tipo Contrato:	03	
Tipo Jornada:	01	
Moneda:	MXN	

Concepto: ClvProd. 84111505 Caril. 1 Clv. U:ACT	Descr. Pago de nómina	Valor uni.	4,510.80	Imp.	4,510.80	Desccto.	436.97
Objeto impuesto: 01 No objeto de impuesto.							

PERCEPCIONES				DEDUCCIONES			
Tipo	Clave	Concepto	Importe	Tipo	Clave	Concepto	Importe
001	001	Sueldos y salarios	4,141.06	002	098	Retención ISR	326.25
021	041	Prima vacacional	73.95	001	099	Cuota IMSS	110.72
001	042	Vacaciones gozadas	295.79				
Total sueldos:		4,510.80		Total otras deducciones:		110.72	
			Total percepciones:				Total impuestos retenidos:
			4,510.80				326.25
							Total deducciones:
							436.97

OTROS PAGOS			
Tipo	Clave	Concepto	Importe
002	002	Subsidio para el empleo	0.00

Subtotal:	4,510.80	Descuentos:	436.97	Total:	4,073.83
				Cuotas patronales:	0.00
				Importe en especie:	0.00
				NETO A PAGAR:	4,073.83

(CUATRO MIL SETENTA Y TRES PESOS 83/100 M.N.)

(CUATRO MIL SETENTA Y TRES PESOS 83/100 M.N.)

Cuotas patronales:	0.00
Importe en especie:	0.00
NETO A PAGAR:	4,073.83

Recibi de SIENOTIS la cantidad indicada que cubre a la fecha el importe de mi salario, tiempo extra y todas las percepciones y prestaciones a que tengo derecho sin que se me adeude alguna cantidad por otro concepto.

EDUARDO MANCERA NIETO

Folio fiscal (UUID): 2DBE5B65-7936-4C37-9717-3F036E085B09

Lugar de expedición: 11560

Fecha-hora emisión: 15/06/2024 09:47:35

Uso de CFDI: CN01

No. serie certificado CSD: 00001000000510565578

Método de pago: PUE=Pago en una sola exhibición

Versión CFDI: 4.0 Tipo comprobante: N=Nómina

Banco: _____ Cuenta bancaria: _____

Sello Digital del CFDI:

EHL+ds3X4G5dXGTY+G4aRBLuWZ8FkFLHuYdaqQAVJkvJ4g9Cm1UAMH6kgxpms10TK3KPGPGKogvpBo4mG895yEoO/V9Q9HoHMcXIDEI:PrXakwwCkWyes/76agCk0+sDxRsBjKbHuyHuT
 +C8Z9f1EiF5KjUjJl8nBvG91+uYHmT6cYzKc+CuRGQYeneF4d2yee2PNdQNeXhXkVw3sJgDeW6uGfVJLZBfHmKbJvmlKcB8G+AZ7Vjx8PJk54t1dCh7Dzmx08LA0RwmfWdcaawB8j4t26
 bGocC8Z9f1EiF5KjUjJl8nBvG91+uYHmT6cYzKc+CuRGQYeneF4d2yee2PNdQNeXhXkVw3sJgDeW6uGfVJLZBfHmKbJvmlKcB8G+AZ7Vjx8PJk54t1dCh7Dzmx08LA0RwmfWdcaawB8j4t26



Versión Timbre: 1.1 Fecha-Hora Timbrado: 15/05/2024 9:47:42 No. Serie CSD del SAT: 00001000000506315610

Cadena Original del Complemento de Certificación Digital del SAT:

[illegible]

Sello Digital del SAT:

15yH2WQDQXa67bV1tk5008qRpp9bRBFx7wsX1VtQT83WwCz6dntHyW5n6Eq6MXkUjgAM2tQp8MvVFuVn9KcRSB2NhDCK+GLZ4Jms
 dEyoAETH5x0WgwZ2haR2d01TbCmXq0Bw6WwwXAJBpJ4dQ28TjQLULh3nqQMWHvChKQabb5pKaS+Br5b6UJV2aaibWJ3m8A6uLmLg9hmR
 Z2u3PaEYmz6YyVA4250vz5E68U5F08hElzbYzShZJ9dEEHnkB672GG8B9XOLPjGtjVjvYsgOPR5FWdUib+XbRZJPGoTvRjDE0LVPQDuYqkw=

Este documento es una representación impresa de un CFDI



CFDI NÓMINA

Versión nómina: 1.2 Tipo nómina: O=Ordinaria

Recibo: N001-10-6 Nómina: 001 Período: 10

Exportación: 01 No aplica

SIENOS

RFC: SIE171003FK6 Registro Patronal: C281567510 Régimen fiscal: 601-General de Ley Personas Morales

Av. Presidente Masaryk 61, Col. Polanco, CP 11560

Ciudad de México

No. Trab.: 000016 Entidad federativa: MEX
Nombre: EDUARDO MANCERA NIETO
RFC: MANE0007194VA
Régimen fiscal: 605
C.P. domicilio fiscal: 54987
CURP: MANE000719HMCNTDA2
No. IMSS: 01-87-63-0595-2
Inicio Rel. Lab.: 16/04/2023
Antigüedad: P58W
Salario diario: 295.79 Salario diario int.: 310.79

Depto.: Sistemas
Puesto:
Días trabajados: 16.00
Periodicidad de pago: 04
Período del: 16/05/2024 al: 31/05/2024
Fecha pago: 30/05/2024
Tipo Régimen: 02
Tipo Contrato: 03
Tipo Jornada: 01
Moneda: MXN

Concepto: ClvProd. 84111505 Cant. 1 Clv. U.:ACT Descr. Pago de nómina Valor uni. 4,732.64 Imp. 4,732.64 Descro. 476.53
Objeto impuesto: 01 No objeto de impuesto.

PERCEPCIONES			
Tipo	Clave	Concepto	Importe
001	001	Sueldos y salarios	4,732.64
Total sueldos:		4,732.64	Total percepciones: 4,732.64

OTROS PAGOS			
Tipo	Clave	Concepto	Importe
002	002	Subsidio para el empleo	0.00

Subtotal: 4,732.64

DEDUCCIONES			
Tipo	Clave	Concepto	Importe
002	098	Retención ISR	358.43
001	099	Cuota IMSS	118.10
Total otras deducciones:		118.10	Total impuestos retenidos: 358.43
			Total deducciones: 476.53

Descuentos: 476.53

Total: 4,256.11

Cuotas patronales: 0.00

Importe en especie: 0.00

NETO A PAGAR: 4,256.11

(CUATRO MIL DOSCIENTOS CINCUENTA Y SEIS PESOS 11/100 M.N.)

Recibí de SIENOS la cantidad indicada que cubre a la fecha el importe de mi salario, tiempo extra y todas las percepciones y prestaciones a que tengo derecho sin que se me adeude alguna cantidad por otro concepto

EDUARDO MANCERA NIETO

Folio fiscal (UUID): C31BC6FA-6D74-477F-BBB6-A839266CE8AE

Lugar de expedición: 11560

Fecha-hora emisión: 29/05/2024 13:29:55

Uso de CFDI: CN01

No. serie certificado CSD: 00001000000510565578

Método de pago: PUE=Pago en una sola exhibición

Versión CFDI: 4.0 Tipo comprobante: N=Nómina

Banco: Cuenta bancaria:

Sello Digital del CFDI:

VxlaZzRfRlBm4N/zIC2EK01P0OTTIdvDUEpb8ZQoJwqkeL33Anpgv+rwJkxR+/qsjv1QggdSHSGmHeTnqbhAshJrxgnW6D1+SSnoilJUde+dezQLgb59YR0h3TbtRqLLMe3VnONmZHWf+HwQJvJZ2OaQXfBFGK175a6vLyYuU2jmx20cVxHPBhG5PB17xeywBDIRd5EOTB2fyalxuyX7d4F6HW5o+z5LoHoHhXsF8vgYVMp+LjzkuQ39seaYKeFwYcVU5NbAilIKqJouzocmU6jRF21r0yJAJBJPpE/8aHQ1B/W1WacompC28xYXnoeMKc0JsGA==



Versión Timbre: 1.1

Fecha-Hora Timbrado: 29/05/2024 13:30:04 No. Serie CSD del SAT: 00001000000506315610

Cadena Original del Complemento de Certificación Digital del SAT:

[1.1]C31BC6FA-6D74-477F-BBB6-A839266CE8AE[2024-05-29T13:30:04]MAS980812UK1[VxlaZzRfRlBm4N/zIC2EK01P0OTTIdvDUEpb8ZQoJwqkeL33Anpgv+rwJkxR+/qsjv1QggdSHSGmHeTnqbhAshJrxgnW6D1+SSnoilJUde+dezQLgb59YR0h3TbtRqLLMe3VnONmZHWf+HwQJvJZ2OaQXfBFGK175a6vLyYuU2jmx20cVxHPBhG5PB17xeywBDIRd5EOTB2fyalxuyX7d4F6HW5o+z5LoHoHhXsF8vgYVMp+LjzkuQ39seaYKeFwYcVU5NbAilIKqJouzocmU6jRF21r0yJAJBJPpE/8aHQ1B/W1WacompC28xYXnoeMKc0JsGA==][00001000000506315610]

Sello Digital del SAT:

4J45WYsa/E7k97ZHRFvH2ajwjm04nzbaje0bahiZzIR441U29y8LnyeRHEutRdTU8HaGgkxQaOANJa3Rw0RaEKxaczHjdv9Nb6ddp8mS8gqcbSiCyxcQgkSIPPpuxYwPefNavOQuYIYDqWwRae7MM07+DBIxcouej5Db1ne8KGOMWVDg1Pj8K7W0365JXB14HbCg4bQnqzcbhYu9psd9bXKGOZ1AJHMMBCB26S7sp2jndaLqhdOObtjOQpqd1sAHwGPM5gz+eGBnTko+2cyW5+23KGFpetTayYosgzfYDSRLI79vAd+UeNnpSPNcGbw==

Este documento es una representación impresa de un CFDI



☎ 55 8834 9673
✉ contacto@siennois.com.mx
🌐 www.siennois.com.mx

Cuautitlán, México a 30 de Octubre de 2024.

A QUIEN CORRESPONDA:

Por medio del presente hacemos de su conocimiento que el trabajador **Eduardo Mancera Nieto**, presta sus servicios para esta empresa, desempeñando el puesto de **Desarrollador Backend** desde el 16 de abril del 2023 a la fecha, realizando las siguientes actividades.

- Analizar la estructura del sitio web.
- Conocer el flujo de navegación para la propuesta del cliente.
- Realizar pruebas de navegación y usabilidad sobre los módulos que conforman el sitio, en entorno de desarrollo y prueba.
- Implementar el sitio en un entorno de producción.
- Responsabilizarse directamente sobre el funcionamiento, navegación y usabilidad del sitio.
- Generar la documentación de la página web, como accesos, archivos y demás documentación que se utilizó durante su desarrollo.
- Resguardar información de acuerdo a formato.
- Creación y administración de base de datos en SQL
- Optimización de consultas para gestión de información precisa haciendo uso de queries y procedimientos almacenados.
- Desarrollo back-end en lenguajes de programación como lo son C++ y PHP
- Manejo de frameworks tales como .NET CORE, CodeIgniter 3 y 4
- Conocimiento en diversas técnicas para la encriptación de datos, autenticación y autorización de los datos del usuario.



SIENOIS

☎ 55 8834 9673

✉ contacto@sienois.com.mx

🌐 www.sienois.com.mx

- Implementación de buenas prácticas para la seguridad y evitar inyecciones SQL, XSS O CSRF
- Capacidad para configurar y mantener el servidor en el cual va a vivir el sitio, asegurando la disponibilidad de este.

Se extiende la presente constancia para los fines académicos que al interesado convenga.

ATENTAMENTE

NANCY TANIA BOJÓRQUEZ HERNÁNDEZ
GERENTE DE ADMINISTRACIÓN Y FINANZAS

