



TECNOLOGICO NACIONAL DE MEXICO

INSTITUTO TECNOLÓGICO DE OAXACA

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN

CARRERA:

INGENIERÍA EN SISTEMAS COMPUTACIONALES

NOMBRE DEL PROYECTO:

**DESARROLLO DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE
PROYECTOS DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E
INVESTIGACIÓN DEL INSTITUTO TECNOLÓGICO DE OAXACA.**

PRESENTAN:

Luján Lugos Héctor Ángel – 15161353

Chávez Cruz Dagoberto – 14161212

Contenido

CAPITULO I. MARCO CONTEXTUAL.	5
1.1 Nombre del proyecto.	5
1.2 Giro de la empresa.	5
1.3 Misión.	5
1.4 Visión.	5
1.5 Croquis de ubicación.	6
1.6 Organigrama de la empresa.	7
1.7 Breve descripción de los procesos de la empresa.	8
1.8 Planteamiento del problema.	9
1.9 Objetivos.	11
1.9.1 Objetivo general.	11
1.9.2 Objetivos específicos.	12
1.10 Justificación.	12
1.11 Alcances y limitaciones.	14
1.11.1 Alcances.	14
1.11.2 Limitaciones.	16
CAPITULO II. MARCO TEORICO.	17
2.1 Gestión de proyectos de software.	17
2.2 Metodologías de desarrollo.	17
2.3 Metodologías ágiles.	18
2.4 Metodología XP.	20
2.4.1 Etapas.	20
2.5 Aplicación web.	22
2.6 Internet.	22
2.7 Servidor web.	22
2.8 Lenguajes de programación.	23
2.8.1 JavaScript.	23
2.9 Node.js.	24
2.10 Lenguajes de marcado.	24
2.10.1 HTML.	25
2.11 CSS.	25
2.12 Bases de datos.	25

2.13 Gestores de bases de datos.....	26
2.13.1 MySQL.....	27
2.14 Frameworks de programación.....	27
2.14.1 Express.js.	28
2.14.2 Bootstrap.....	28
2.15 Git.	29
2.16 GitHub.	29
CAPITULO III. DESARROLLO METODOLOGICO.	29
3.1 Descripción de las actividades en las fases del modelo de desarrollo.....	30
3.2 Especificación del modelo.....	30
3.3 Lista maestra de iteraciones (Iteration Plan).	30
3.4 Plan de reuniones.....	31
3.5 Primera iteración.	32
3.5.1 Etapa de planeación.	33
3.5.2 Etapa de diseño.	44
3.5.3 Etapa de codificación.....	52
3.5.4 Etapa de pruebas.	64
3.6 Segunda iteración.	71
3.6.1 Etapa de planeación.	71
3.6.2 Etapa de diseño.	72
3.6.3 Etapa de codificación.....	78
3.6.4 Etapa de pruebas.	80
3.7 Tercera iteración.....	86
3.7.1 Etapa de planeación.	86
3.7.2 Etapa de diseño.	86
3.7.3 Etapa de codificación.....	89
3.7.4 Etapa de pruebas.	91
CAPITULO IV. CONCLUSIONES Y RECOMENDACIONES.....	94
4.1 Conclusiones generales.	94
4.2 Conclusiones personales.....	94
4.3 Recomendaciones.....	95
REFERENCIAS BIBLIOGRAFICAS.....	95

CAPITULO I. MARCO CONTEXTUAL.

1.1 Nombre del proyecto.

Desarrollo de una aplicación web para la gestión de proyectos de la División de Estudios de Posgrado e Investigación del Instituto Tecnológico de Oaxaca.

1.2 Giro de la empresa.

Institución educativa.

1.3 Misión.

Ofertar educación superior tecnológica y posgrado, de calidad a nivel nacional e internacional; para participar en el desarrollo científico y tecnológico de las personas y la comunidad.

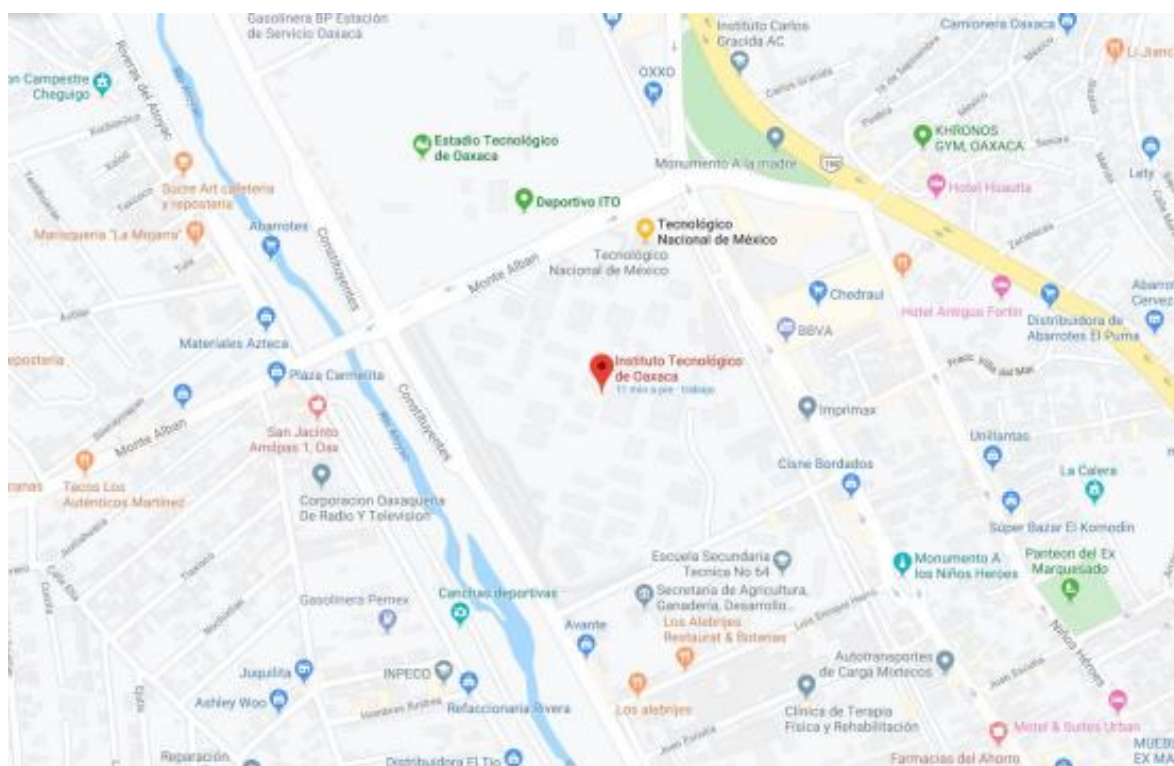
1.4 Visión.

Para el año 2025 el Tecnológico Nacional de MÉXICO/Instituto Tecnológico de Oaxaca, seremos reconocidos como una institución académica de alto impacto a la sociedad, contribuyendo significativamente al desarrollo de Oaxaca y del Sureste de México, gracias a la aportación de las personas que se forman en nuestra institución mediante planes y programas de licenciatura acreditados a nivel internacional, así como la formación de posgrado con registro en PNPC del CONACyT, como resultado de la aplicación de los sistemas funcionales y

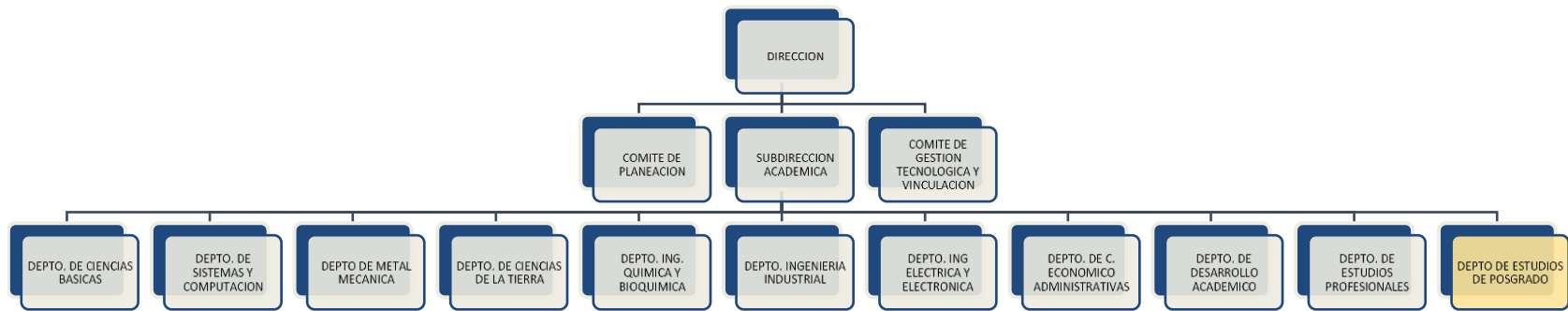
amigables que contribuyen a la integración y compromiso de las personas que trabajan en nuestro nombre.

1.5 Croquis de ubicación.

Avenida Ing. Víctor Bravo Ahuja No. 125 esquina Calzada Tecnológico, C.P. 68030.



1.6 Organigrama de la empresa.



1.7 Breve descripción de los procesos de la empresa.

Dentro del área de posgrado se realizan diferentes actividades, lo que compete al control de la información referente a los proyectos de investigación se lleva a cabo los siguientes procesos:

Una vez aprobado un proyecto ante el TecNM el coordinador del área de posgrado registra la información del financiamiento obtenido, esta información se guarda en hojas de cálculo.

En conjunto con el financiamiento también se registra la información del proyecto una vez que esté aprobado y el coordinador asigna de manera manual el financiamiento que corresponde a un proyecto mediante una clave de asignación única.

El financiamiento de un proyecto lleva consigo algunas partidas presupuestales y el coordinador se encarga de guardar las partidas que corresponden a determinado proyecto, así también las subpartidas para controlar la justificación de los recursos obtenidos.

El coordinador recibe, almacena y digitaliza la información del líder, sus colaboradores y los estudiantes que están involucrados en un proyecto en particular relacionándolo con la información del proyecto.

El coordinador recibe de parte del líder de proyecto, las fechas preliminares de los informes y la entrega final del proyecto, éste las guarda para ir viendo el proceso de avance.

El líder recibe su registro de proyecto en la plataforma del TecNM, teniendo la responsabilidad de subir su información relevante al sistema, el coordinador se encarga de verificar la información y si en dado caso que se requiera realiza la transcripción de la información del proyecto ya que la plataforma del TecNM solo permite descargarlo en formato de imagen.

Conforme se avanza en el proyecto el líder realiza el registro de avance en su investigación y el coordinador se encarga de verificar la información, y darle seguimiento de manera manual.

El coordinador de posgrado realiza revisión del reporte final obtenido de una investigación, y esta información se sube a la plataforma del TecNM, en conjunto con el producto final que se obtiene esta se accede a través de un repositorio.

El coordinador mantiene la información en hojas de cálculo, para posibles reportes o aclaraciones.

1.8 Planteamiento del problema.

El Tecnológico Nacional de México (TecNM) se caracteriza por formar profesionales en diversas áreas de estudio, de igual manera, estudiantes de posgrado en las distintas especializaciones con las que cuenta.

El departamento de posgrado se encarga de gestionar ante el TecNM los proyectos de investigación que se basan en convocatorias que se realiza cada año, este proceso de manera interna en el Tecnológico De Oaxaca inicia cuando un proyecto ya ha sido aprobado y obtenido un financiamiento, con la cual se inicia el proceso

de registro referente a los datos del financiamiento y los datos del proyecto aprobado, aunado a esta se basa en partidas presupuestales bajo las cuales se rige el uso del recurso aprobado, este proceso manual conlleva varios problemas de integridad de los datos, la falta de una estructura adecuada para almacenar y organizar la información ya que todo esto se guarda en hojas de cálculo de Excel, llevando a que esto se tarde en realizarse ya que cada proyecto lleva información relevante la cual se tiene que estar verificando para no tener errores en la redacción.

En cuanto se refiere al proyecto el coordinador de posgrado tiene que recibir por algún medio información del líder, los colaboradores involucrados en un proyecto en particular, así como también recibe las fechas preliminares de los informes y la fecha del reporte final lo que genera retrasos en el envío de información y no hay un seguimiento adecuado en cuanto a los informes debido a que no se cumplen las fechas de entrega.

Una vez que se recibe la autorización por parte del TecNM, el líder sube la información de su proyecto a la plataforma, sin embargo, en ocasiones el líder desconoce las partidas y subpartidas correspondientes para que el financiamiento sea adecuado, y al momento que el coordinador verifica se percata que hay información inconsistente; para la verificación tiene que descargar el archivo que se encuentra en un formato de imagen, aun cuando la información esté o no correcta, tiene que transcribirse generando un nuevo documento de texto, lo que provoca pérdida de tiempo y cargas de trabajo.

Una vez enviadas las revisiones del proyecto y los reportes por el líder, el coordinador revisa el reporte final para que el líder del proyecto lo cargue a la plataforma del TecNM; de todo el proceso se obtiene un archivo en formato PDF como producto final la cual se accede a través de un repositorio; esta información es impresa y archivada de forma física, por lo que al momento de que la información sea solicitada se tiene que realizar la búsqueda de forma manual o bien la búsqueda por nombre de archivo lo que provoca pérdida de tiempo.

Por lo anterior se identificaron los siguientes problemas:

1. Falta de un control y una estructura adecuada para el almacenamiento de información.
2. No se mantienen de manera íntegra los datos.
3. Falta de consistencia en la información.
4. No existe un seguimiento adecuado y genera retrasos en el avance del proyecto.
5. Pérdida de tiempo en la búsqueda específica de algún proyecto.
6. Consumo de tiempo en la transcripción de la información.

1.9 Objetivos.

1.9.1 Objetivo general.

Desarrollar una aplicación web para la gestión de los proyectos de investigación, para la División de Estudios de Posgrado e Investigación del Instituto Tecnológico de Oaxaca.

1.9.2 Objetivos específicos.

- Identificar los procesos de gestión en el área de posgrado.
- Aplicar la metodología de desarrollo de software XP.
- Realizar plan de reuniones.
- Generar plan de iteraciones.
- Realizar historias de usuario.
- Identificar requerimientos funcionales y no funcionales.
- Definir roles y privilegios de los usuarios para acceso al sistema.
- Generar el diagrama entidad-relación.
- Diseñar el modelo relacional.
- Crear la base de datos utilizando el manejador de base de datos de MySQL.
- Diseñar de las interfaces gráficas de usuario.
- Codificar los módulos del sistema mediante el lenguaje JavaScript.
- Probar los módulos del sistema.
- Realizar reportes del sistema.
- Realizar la documentación del sistema.

1.10 Justificación.

Con el desarrollo de una aplicación web para la gestión de los proyectos de investigación, para la División de Estudios de Posgrado e Investigación del Instituto Tecnológico de Oaxaca traerá los siguientes beneficios:

El sistema contará con una base de datos utilizando el manejador de base de datos MySQL para el control adecuado de los registros de cada uno de los proyectos, financiamientos, y usuarios, evitando así los problemas de la falta de integridad de los datos y la falta de una estructura para almacenar y organizar la información, lo que permitirá eficientar los procesos de búsqueda, y que la información esté disponible en cualquier momento.

El sistema permitirá el seguimiento de cada una de las etapas, avances y registros del proyecto con base en los periodos establecidos, manteniendo la interacción entre el líder y el coordinador a través de notificaciones, evitando así los retrasos en las entregas del proyecto, lo que permitirá aumentar la productividad y eficiencia en los procesos para el envío de la información.

El sistema mostrará de forma intuitiva las interfaces gráficas que le permitirán a los usuarios navegar en el sistema de manera más sencilla y amigable para la captura, registro, y envío de información.

Otro beneficio importante del sistema es que les permitirá a los usuarios descargar la información a través de formatos PDF para su revisión o verificación, reduciendo los tiempos de transcripción de los archivos y/o de búsqueda.

El sistema mostrará una lista desplegable de las subpartidas correspondientes a cada una de las partidas de los financiamientos para evitar que los usuarios elijan de manera errónea una subpartida que no pertenezca a dicha partida, evitando así la inconsistencia de la información.

Con este sistema se eficientará, desde las tareas pequeñas y sencillas hasta las tareas manuales y tediosas, siendo una herramienta útil que brindara grandes beneficios como el ahorro de tiempo, integridad y respaldo de los proyectos, manteniendo así la disponibilidad de la información en tiempo y forma, para el departamento de la División de Estudios de Posgrado e Investigación.

1.11 Alcances y limitaciones.

1.11.1 Alcances.

A. El sistema web contará con los siguientes módulos:

1. Usuarios.

En este módulo se llevará a cabo el registro de los usuarios que van a acceder al sistema, a través de un nombre de usuario y una contraseña. Además, permitirá la autenticación, con sus roles y privilegios, mediante notificaciones vía correo electrónico.

2. Financiamiento.

Este módulo permitirá, la alta, baja y actualización de financiamientos para proyectos de investigación, con datos correspondientes de los financiamientos, información sobre partidas y subpartidas.

3. Proyectos de investigación

En este módulo se lleva a cabo la creación, edición y actualización de proyectos de investigación pertenecientes a los financiamientos

creados, añadiendo planeación de fechas de los informes.

4. Búsqueda.

Este módulo permitirá a través de una interfaz intuitiva la búsqueda rápida de proyectos y búsqueda de usuarios, mediante los nombres de los usuarios o identificadores que tengan asignados.

5. Reportes.

Este módulo permitirá generar reportes con base en la información obtenida de los proyectos de investigación, y la descarga de los archivos en formatos PDF.

6. Seguimiento de proyectos.

Este módulo permitirá conocer el estado actual de los proyectos de investigación mediante un semáforo de colores, que puede ser proyectos en tiempo y forma, retrasados, pendientes y/o finalizados.

B. El sistema web contara con las siguientes características:

1. El sistema mostrará una lista con las partidas y subpartidas correspondientes al financiamiento de cada uno de los proyectos.

2. El sistema contará con notificaciones para los usuarios para los tiempos establecidos del envío de información.
3. El sistema contará con un apartado para guardar proyectos anteriores a la creación del sistema, en formato PDF.

1.11.2 Limitaciones.

- El proyecto está sujeto a las fechas establecidas con base en las convocatorias realizadas por el TecNM para el registro de los proyectos.
- La autorización y financiamiento del proyecto le corresponde únicamente al TecNM.
- La duración del proyecto estará basada en las fechas asignadas para el financiamiento dado por el TecNM.
- El tamaño de los archivos a subir al sistema estará sujeto dependiendo del tamaño permitido por la plataforma del TecNM.

CAPITULO II. MARCO TEORICO.

2.1 Gestión de proyectos de software.

La gestión de proyectos de software se encarga de planificar el proceso de desarrollo de un producto, en este caso de un software. Realizando un seguimiento del trabajo y llevando a cumplir los estándares establecidos mediante metodologías de desarrollo bien definidas, repetitivas y confiables.

“La administración del proyecto involucra planificación, monitoreo y control del personal, procesos y acciones que ocurren conforme el software evoluciona desde un concepto preliminar hasta su despliegue operativo completo” (Roger S. Pressman, 2007, p. 554).

Es importante ya que el construir software es una labor compleja, sobre todo cuando trabajan varias personas por un tiempo prolongado, la gestión de software evita errores que se acumularían con el tiempo de no tener una planeación.

2.2 Metodologías de desarrollo.

Las definiciones de metodología de desarrollo de software son variadas, sin embargo, todas convergen en que se consideran como una estructura utilizada para planificar y controlar el procedimiento de creación de un sistema.

Objetivos:

- Establecer los requisitos de un sistema.
- Proporcionar un método sistemático de desarrollo que se pueda controlar.
- Construir software dentro de tiempos y costos aceptables.

- Identificar cualquier cambio lo antes posible.
- Cubrir el ciclo de vida completo del software.

2.3 Metodologías ágiles.

Por definición las metodologías ágiles son aquellas que permiten adaptar la forma de trabajo a las condiciones del proyecto, consiguiendo flexibilidad e inmediatez en la respuesta para amoldar el proyecto y su desarrollo a las circunstancias específicas del entorno. Cabrera, Figueroa y Solís (2007) afirman “Estas metodologías ponen de relevancia la capacidad de respuesta a un cambio es más importante que el seguimiento estricto de un plan” (p. 5). En esencia, las empresas que apuestan por esta metodología consiguen gestionar sus proyectos de forma flexible, autónoma y eficaz reduciendo los costes e incrementando su productividad.

Ventajas:

- Mejorar la satisfacción del cliente: Este se involucrará y comprometerá a lo largo de todo el proyecto.
- Mejorar la motivación e implicación del equipo de desarrollo: Las metodologías ágiles permiten a todos los miembros del equipo conocer el estado del proyecto en cualquier momento, así, los compromisos son negociados y aceptados por todos los miembros del equipo.
- Ahorrar tiempo y costos: El desarrollo ágil trabaja de un modo más eficiente y rápido, y con ello, se cumple de forma estricta el presupuesto y los plazos pactados dentro de un proyecto.

Metodologías ágiles más utilizadas:

Extreme Programming (XP): Esta herramienta es muy útil sobre todo para startups o empresas que están en proceso de consolidación, puesto que su principal objetivo es ayudar en las relaciones entre los empleados y clientes. La clave del éxito del XP es potenciar las relaciones personales, a través, del trabajo en equipo, fomentando la comunicación y eliminando los tiempos muertos.

SCRUM: Se caracteriza por ser la “metodología del caos” que se basa en una estructura de desarrollo incremental, esto es, cualquier ciclo de desarrollo del producto y/o servicio se desgrana en “pequeños proyectos” divididos en distintas etapas: análisis, desarrollo y testing.

SCRUM	XP
Es una metodología de desarrollo ágil basada en la administración del proyecto.	Es una metodología de desarrollo que está más centrada en la programación o creación del producto.
Cada miembro de del equipo trabaja de forma individual.	Los miembro del equipo programan en parejas.
Las ietraciones de entrega son de 1 a 4 semanas.	Las ietraciones de entrega son de 1 a 3 semanas
Al finalizar un Sprint, las tareas del Sprint Backlog que se hayan realizado y que el Product Owner (propietario del producto) haya mostrado su conformidad ya no se retoca. Si funciona y está bien, se aparta y a otra cosa.	Las tareas se van terminando aunque son susceptibles de ser modificadas durante el transcurso de proyecto, incluso, después de que funcionen correctamente.
Trata de seguir el orden de prioridades que marca el Product Owner en el Sprint Backlog pero puede cambiarlo si es mejor para el desarrollo de la tareas.	El equipo de desarrollo sigue estrictamente el orden de prioridad de las tareas definido por el cliente.

Ilustración 1. Tabla comparativa entre SCRUM y XP. (S.f). Recuperado de:
https://islavisual.com/articulos/desarrollo_web/diferencias-entre-scrum-y-xp.php

2.4 Metodología XP.

La Metodología XP “Extreme Programming” o “Programación Extrema” diseñada por Kent Beck es una de las metodologías ágiles de desarrollo de software más conocidas. XP está diseñada para entregar el software que los clientes necesitan en el momento en que lo necesitan. XP alienta a los desarrolladores a responder a los requerimientos cambiantes de los clientes, aún en fases tardías del ciclo de vida del desarrollo.

Según Roger S. Pressman (2007) “Beck define un conjunto de cinco valores que establecen el fundamento para todo trabajo realizado como parte de XP: comunicación, simplicidad, retroalimentación, valentía y respeto.” (p. 61).

2.4.1 Etapas.

Roger S. Pressman (2007) señala que “la programación extrema engloba un conjunto de reglas y prácticas que ocurren en el contexto de cuatro actividades estructurales: planeación, diseño, codificación y pruebas.” (p. 62)

2.4.1.1 Planeación.

La planeación comienza escuchando para recabar requerimientos que permite que los miembros del equipo XP entiendan el contexto del negocio para el software y adquieran la sensibilidad, características principales y funcionalidad que se requieren. Escuchar lleva a la creación de algunas “historias” (también llamadas

historias del usuario) que describen la salida necesaria, características y funcionalidad del software que se va a elaborar.

2.4.1.2 Diseño.

Roger S. Pressman (2007) afirma que “el diseño XP sigue rigurosamente el principio MS (mantenlo sencillo). Un diseño sencillo siempre se prefiere sobre una representación más compleja. Además, el diseño guía la implementación de una historia conforme se escribe.” (p. 63)

2.4.1.3 Codificación.

Un concepto clave durante la actividad de codificación es la programación por parejas. XP recomienda que dos personas trabajen juntas en una estación de trabajo con el objeto de crear código para una historia. Esto da un mecanismo para la solución de problemas en tiempo real y asegura la calidad también en tiempo real.

2.4.1.4 Pruebas.

La creación de pruebas unitarias antes de la codificación es un elemento clave del enfoque de XP. Las pruebas unitarias que se crean deben implementarse con el uso de una estructura que permita automatizarlas (de modo que puedan ejecutarse en repetidas veces y con facilidad). Esto estimula una estrategia de pruebas de regresión siempre que se modifique el código.

2.5 Aplicación web.

Es llamada aplicación web a todas aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador.

2.6 Internet.

Internet es un neologismo del inglés que significa red informática descentralizada de alcance global. Se trata de un sistema de redes interconectadas mediante distintos protocolos que ofrece una gran diversidad de servicios y recursos, como, por ejemplo, el acceso a archivos de hipertexto a través de la web.

2.7 Servidor web.

En informática, un servidor web o Servidor HTTP es una pieza de software de comunicaciones que intermedia entre el servidor en el que están alojados los datos solicitados y el computador del cliente, permitiendo conexiones bidireccionales o unidireccionales, síncronas o asíncronas, con cualquier aplicación del cliente, incluso con los navegadores que traducen un código traducible (renderizable) a una página web determinada. O sea, se trata de programas que median entre el usuario de Internet y el servidor en donde está la información que solicita.

2.8 Lenguajes de programación.

En informática, se conoce como lenguaje de programación a un programa destinado a la construcción de otros programas informáticos. Su nombre se debe a que comprende un lenguaje formal que está diseñado para organizar algoritmos y procesos lógicos que serán luego llevados a cabo por un ordenador o sistema informático, permitiendo controlar así su comportamiento físico, lógico y su comunicación con el usuario humano. Dicho lenguaje está compuesto por símbolos y reglas sintácticas y semánticas, expresadas en forma de instrucciones y relaciones lógicas, mediante las cuales se construye el código fuente de una aplicación o pieza de software determinado.

2.8.1 JavaScript.

JavaScript (JS) es un lenguaje de programación ligero, interpretado, o compilado justo a tiempo (just-in-time) con funciones de primera clase. Si bien es más conocido como un lenguaje de scripting (secuencias de comandos) para páginas web, y es usado en muchos entornos fuera del navegador, tal como Node.js, Apache CouchDB y Adobe Acrobat.

JavaScript es un lenguaje de programación basada en prototipos, multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa.

2.9 Node.js

Node.js es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor, basado en el lenguaje de programación JavaScript, asíncrono, con entrada y salida de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como, por ejemplo, servidores web.

2.10 Lenguajes de marcado.

Un lenguaje de marcado o lenguaje de marcas es una forma de codificar un documento que, junto con el texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o su presentación.

El lenguaje de marcas más extendido es el HTML (HyperText Markup Language, lenguaje de marcado de hipertexto), fundamento del World Wide Web (entramado de comunicación de alcance mundial).

Los lenguajes de marcado suelen confundirse con lenguajes de programación. Sin embargo, no son lo mismo, ya que el lenguaje de marcado no tiene funciones aritméticas o variables, como poseen los lenguajes de programación. Históricamente, el marcado se usaba y se usa en la industria editorial y de la comunicación, así como entre autores, editores e impresores.

2.10.1 HTML.

Lenguaje de Marcado de Hipertextos (HyperText Markup Language), es la pieza más básica para la construcción de la web y se usa para definir el sentido y estructura del contenido en una página web.

2.11 CSS.

Hojas de Estilo en Cascada (Cascading Style Sheets) o CSS es el lenguaje utilizado para describir la presentación de documentos HTML o XML, esto incluye varios lenguajes basados en XML como son XHTML o SVG.

CSS describe como debe ser renderizado el elemento estructurado en pantalla, en papel, hablado o en otros medios. CSS es uno de los lenguajes base de la Open Web y posee una especificación estandarizada por parte del W3C.

De acuerdo con Schulz (2021) afirma “CSS contiene muchas funciones y métodos con los que proporcionar funciones ampliadas a sus visitantes. Por ejemplo, elementos ópticos de navegación o un tipo de representación alternativa conforme a los deseos y necesidades de sus visitantes grupo de visitantes” (p. 5).

2.12 Bases de datos.

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido; una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Actualmente,

y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital, siendo este un componente electrónico, por tanto, se ha desarrollado y se ofrece un amplio rango de soluciones al problema del almacenamiento de datos.

2.13 Gestores de bases de datos.

Un Sistema Gestor de Base de Datos (SGBD) o DataBase Management System (DBMS) es un sistema que permite la creación, gestión y administración de bases de datos, así como la elección y manejo de las estructuras necesarias para el almacenamiento y búsqueda de información del modo más eficiente posible.

Hueso (2006) afirma “La principal función de un SGBD es permitir a los usuarios realizar las cuatro operaciones fundamentales posibles, tanto sobre las estructuras de datos como sobre los datos que albergan, es decir operaciones de inserción o creación, consulta, actualización y borrado” (p. 11).

Los principales Sistemas gestores de bases de datos relacionales (SGBD SQL) actualmente son:

- MySQL.
- MariaDB.
- PostgreSQL.

2.13.1 MySQL.

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, todo para entornos de desarrollo web.

Características:

- Amplio subconjunto del lenguaje SQL.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Posibilidad de selección de mecanismos de almacenamiento que ofrecen diferentes velocidades de operación, soporte físico, capacidad.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.

2.14 Frameworks de programación.

La traducción literal del término, marco de referencia nos da un primer indicio de su significado, ya que se trata de un esquema o patrón que permite se puede estructurar de una mejor forma el código, es decir al utilizar un framework se programa escribiendo menos líneas de código, lo que por supuesto es una ventaja que los programadores no desaprovechan. Podría decirse entonces que un

framework es una aplicación genérica incompleta que se puede configurar para añadirle las líneas de código que sean necesarias para programar una determinada aplicación; una analogía de ello podría ser un rompecabezas al que se le pueden agregar nuevas piezas para hacerlo más completo y detallado o agregarle nuevas funciones. Este marco de referencia o marco de trabajo nos provee un conjunto de código que se puede utilizar y reutilizar en cualquier sistema, bien sea simple o complejo; ofreciéndonos una forma estándar para trabajar en un lenguaje de programación determinado; es decir empleando un paradigma ya existente; siendo el Modelo-Vista-Controlador (MVC) el más empleado en la actualidad.

2.14.1 Express.js.

Express.js, o simplemente Express, es un marco de aplicación web back-end para Node.js, lanzado como software gratuito y de código abierto bajo la licencia MIT. Está diseñado para crear aplicaciones web y API. Se le ha llamado el estándar de facto marco servidor para Node.js.

2.14.2 Bootstrap.

Bootstrap es un framework CSS y JavaScript diseñado para la creación de interfaces limpias y con un diseño responsive. Además, ofrece un amplio abanico de herramientas y funciones, de manera que los usuarios pueden crear prácticamente cualquier tipo de sitio web haciendo uso de los mismos.

Actualmente, Bootstrap es una de las alternativas más populares a la hora de desarrollar tanto sitios webs como aplicaciones. Una de las principales ventajas que ofrece es que permite la creación de sitios y apps 100% adaptables a cualquier tipo de dispositivo.

2.15 Git.

Git es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Su propósito es llevar registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos.

2.16 GitHub.

Git es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Su propósito es llevar registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos.

CAPITULO III. DESARROLLO METODOLOGICO.

3.1 Descripción de las actividades en las fases del modelo de desarrollo.

Este capítulo contiene la descripción de todas las actividades del proceso de desarrollo del sistema.

3.2 Especificación del modelo.

Para este proyecto se va a utilizar el modelo de desarrollo de Software ágil en programación extrema (eXtreme Programming, XP), del autor Kent Beck del libro Ingeniería de Software, Roger Pressman.

Sus etapas son:

- Planeación
- Diseño
- Codificación
- Pruebas

Las actividades a desarrollar son:

3.3 Lista maestra de iteraciones (Iteration Plan).

No.	Nombre de la historia de usuario.	Iteración.
1	Gestión de financiamientos.	1ra. Iteración.
2	Gestión de proyectos.	
3	Gestión de usuarios.	
4	Login.	

5	Partidas y subpartidas.	2da. Iteración.
6	Búsqueda.	
7	Reportes.	
8	Planificaciones.	
9	Contenido de proyectos.	
10	Observaciones.	
11.	Seguimiento de proyectos.	3ra. Iteración.
12.	Semaforización.	
13.	Repositorio.	
14.	Notificaciones.	
15.	Restablecimiento de contraseñas.	

3.4 Plan de reuniones.

Actividad	Responsable	Fecha
Recolección de requerimientos del sistema	Dr. Marco Antonio Sánchez Medina.	18 de noviembre del 2020.
Recolección de requerimientos y propuesta del funcionamiento del sistema.	Dr. Marco Antonio Sánchez Medina.	23 de noviembre del 2020.
Reunión para dudas sobre los requerimientos analizados.	Dr. Marco Antonio Sánchez Medina.	30 de noviembre del 2020.
Primera revisión del anteproyecto	M.E.T. Eva Rafael Pérez.	03 de diciembre del 2020.
Segunda revisión del anteproyecto.	M.E.T. Eva Rafael Pérez.	11 de diciembre del 2020.

Revisión de la documentación.	M.E.T. Eva Rafael Pérez.	14 de diciembre del 2020.
Primera revisión del sistema.	M.E.T. Eva Rafael Pérez.	16 de diciembre del 2020.
Revisión de avances del sistema.	M.E.T. Eva Rafael Pérez.	13 de enero del 2021.
Sugerencias de cambios para el sistema.	M.E.T. Eva Rafael Pérez.	14 de enero del 2021.
Revisión de los cambios sugeridos.	M.E.T. Eva Rafael Pérez.	18 de enero del 2021.
Revisión de avances del sistema.	M.E.T. Eva Rafael Pérez.	26 de enero del 2021.
Muestra del avance y dudas sobre el comportamiento de los financiamientos.	Dr. Marco Antonio Sánchez Medina.	10 de febrero del 2021.
Revisión de avances del sistema.	M.E.T. Eva Rafael Pérez.	23 de febrero del 2021.
Revisión de avances del sistema.	M.E.T. Eva Rafael Pérez.	03 de marzo del 2021
Reunión para la delimitación y validación de campos.	Dr. Marco Antonio Sánchez Medina.	04 de marzo del 2021.
Reunión para las partidas y subpartidas autorizadas por TecNM.	Dr. Marco Antonio Sánchez Medina.	09 de marzo del 2021.
Muestra del flujo del sistema.	M.E.T. Eva Rafael Pérez.	17 de marzo del 2021.
Muestra del comportamiento final del sistema.	M.E.T. Eva Rafael Pérez.	26 de marzo del 2021.

3.5 Primera iteración.

3.5.1 Etapa de planeación.

3.5.1.1 Historias de usuario.

Tablas de explicación para prioridades en el negocio.

Tipo de prioridad en el negocio.	Explicación.
Alta.	La prioridad de la historia de usuario o módulo redactado es urgente, dependen de ella futuras funcionalidades del sistema.
Media.	La prioridad de esta historia de usuario depende de una historia de usuario o módulo anteriormente creado.
Baja.	El desarrollo de esta historia de usuario es independiente y funciona como un complemento a otros módulos o historias de usuario por lo tanto su prioridad es baja.

Tabla de explicación para riesgo en desarrollo.

Nivel de riesgo en desarrollo.	Explicación.
Alto.	El desarrollo de la historia de usuario implica el uso de múltiples características del sistema y de ser mal desarrollada desencadenará problemas inmediatos.
Medio.	El mal desarrollo de la historia de usuario causará problemas de comprensión dentro del sistema.
Bajo.	La funcionalidad de esta historia de usuario no afecta a otras funcionalidades del sistema.

Historia de usuario	
Número: 1	Usuario: Administrador
Nombre de la historia: Gestión de financiamientos	

Prioridad en negocio: Alta.	Riesgo en desarrollo: Alta
Puntos estimados: 2	Iteración asignada: 1
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto.	
Descripción: Yo como usuario administrador, quiero que el sistema web permita agregar y modificar financiamientos para los proyectos de investigación que han sido aceptados por el TecNM.	
Validación:	

Historia de usuario	
Número: 2	Usuario: Líder de proyecto
Nombre de la historia: Gestión de proyectos	
Prioridad en negocio: Alta.	Riesgo en desarrollo: Alta
Puntos estimados: 2	Iteración asignada: 1
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario líder de proyecto, quiero que el sistema me permita agregar, modificar y eliminar proyectos de investigación para un financiamiento en particular.	
Validación: Los proyectos solo pueden ser asignados a un financiamiento.	

Historia de usuario	
Número: 3	Usuario: Administrador
Nombre de la historia: Gestión de usuarios	
Prioridad en negocio: Alta.	Riesgo en desarrollo: Alta
Puntos estimados: 2	Iteración asignada: 1
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema permita agregar, modificar y eliminar usuarios responsables de los proyectos de investigación para que reporten información.	
Validación:	

Historia de usuario	
Número: 4	Usuario: Administrador
Nombre de la historia: Login.	
Prioridad en negocio: Media.	Riesgo en desarrollo: Bajo.
Puntos estimados: 2	Iteración asignada: 1
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema permita el acceso únicamente a usuarios creados por mi, para mantener el control de acceso al sistema.	

Validación:

Historia de usuario	
Número: 5	Usuario: Administrador
Nombre de la historia: Partidas y subpartidas.	
Prioridad en negocio: Media.	Riesgo en desarrollo: Medio
Puntos estimados: 2	Iteración asignada: 2
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema tenga un apartado con una lista de partidas con sus subpartidas para poder revisar sus datos y asegurar la información.	
Validación:	

Historia de usuario	
Número: 6	Usuario: Administrador
Nombre de la historia: Búsqueda.	
Prioridad en negocio: Media.	Riesgo en desarrollo: Medio
Puntos estimados: 2	Iteración asignada: 2

Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto
Descripción: Yo como usuario administrador, quiero que el sistema permita realizar búsqueda de proyectos, usuarios y financiamientos para poder revisar sus datos.
Validación:

Historia de usuario	
Número: 7	Usuario: Administrador
Nombre de la historia: Reportes.	
Prioridad en negocio: Media.	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 2
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema permita realizar reportes los proyectos de investigación en cualquier momento para poder imprimirlos.	
Validación:	

Historia de usuario	
Número: 8	Usuario: Administrador
Nombre de la historia: Planificaciones.	

Prioridad en negocio: Media.	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 2
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema permita modificar las fechas de entregas predeterminadas al proyecto de investigación, para mayor control sobre los proyectos.	
Validación:	

Historia de usuario	
Número: 9	Usuario: Líder de proyecto
Nombre de la historia: Contenido de proyectos.	
Prioridad en negocio: Media.	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 2
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario líder de proyecto, quiero que el sistema permita agregar archivos en formato PDF para protocolo e informes, así como registrar los entregables, y los colaboradores de mi proyecto.	
Validación:	

Historia de usuario

Número: 10	Usuario: Administrador
Nombre de la historia: Observaciones.	
Prioridad en negocio: Media.	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 2
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema permita visualizar los archivos de un proyecto de investigación en específico, para realizar observaciones y sugerir correcciones a sus responsables.	
Validación:	

Historia de usuario	
Número: 11	Usuario: Administrador
Nombre de la historia: Seguimiento de proyectos.	
Prioridad en negocio: Media.	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 3
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema permita visualizar el avance de un proyecto de investigación en específico, para detectar de manera más sencilla problemas de tiempo y entregas en ellos.	

Validación:

Historia de usuario	
Número: 12	Usuario: Administrador
Nombre de la historia: Semaforización.	
Prioridad en negocio: Media.	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 3
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema permita visualizar mediante colores los proyectos de investigación, para detectar de manera más sencilla problemas de tiempo y entregas en ellos.	
Validación:	

Historia de usuario	
Número: 13	Usuario: Administrador
Nombre de la historia: Repositorio.	
Prioridad en negocio: Media.	Riesgo en desarrollo: Medio

Puntos estimados: 3	Iteración asignada: 3
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema pueda guardar los proyectos de investigación finalizados de periodos pasados en formato PDF para tenerlos a disposición.	
Validación:	

Historia de usuario	
Número: 14	Usuario: Administrador
Nombre de la historia: Notificaciones.	
Prioridad en negocio: Media.	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 3
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema envíe notificaciones sobre las fechas de entrega de los proyectos de investigación para mantener a los responsables de proyectos atentos a sus entregas.	
Validación:	

Historia de usuario

Número: 15	Usuario: Administrador
Nombre de la historia: Restablecimiento de contraseñas.	
Prioridad en negocio: Baja.	Riesgo en desarrollo: Medio.
Puntos estimados: 1	Iteración asignada: 3
Programadores responsables: Lujan Lugos Hector Angel, Chávez Cruz Dagoberto	
Descripción: Yo como usuario administrador, quiero que el sistema permita cambiar las contraseñas a los usuarios.	
Validación:	

Historias de usuario a realizar.

Iteración.	No. de historia de usuario.	Historia de usuario.
1ra iteración.	1	Gestión de financiamientos.
	2	Gestión de proyectos.
	3	Gestión de usuarios.
	4	Login.

3.5.1.2 Requerimientos funcionales y no funcionales.

A. Funcionales.

- Los usuarios recibirán mensajes de notificación al momento de agregar, actualizar o eliminar datos importantes.
- El sistema enviara un correo electrónico con datos de acceso al nuevo usuario cuando se le asigne una cuenta.

- El sistema permitirá a los líderes de proyectos autorizados crear proyectos de investigación con base a una clave de financiamiento.
- El administrador podrá ver todos los proyectos que han sido creados en el sistema.
- El administrador podrá finalizar los proyectos en cualquier momento.
- El administrador se encarga del acceso a los líderes de proyectos.
- Los líderes de proyectos entregaran los datos en tiempo y forma.
- El administrador podrá cambiar los tiempos de entrega de los proyectos.
- Los lideres de proyectos dejaran de ver los detalles de su proyecto una vez sea finalizado por el administrador.
- Los proyectos históricos, anteriores a la creación del sistema serán agregados por el administrador.
- El avance de los proyectos será visible para ambos usuarios.
- El avance de los proyectos será representado mediante una línea que cambiará de color con el estado del proyecto con iconos de los elementos por realizar.

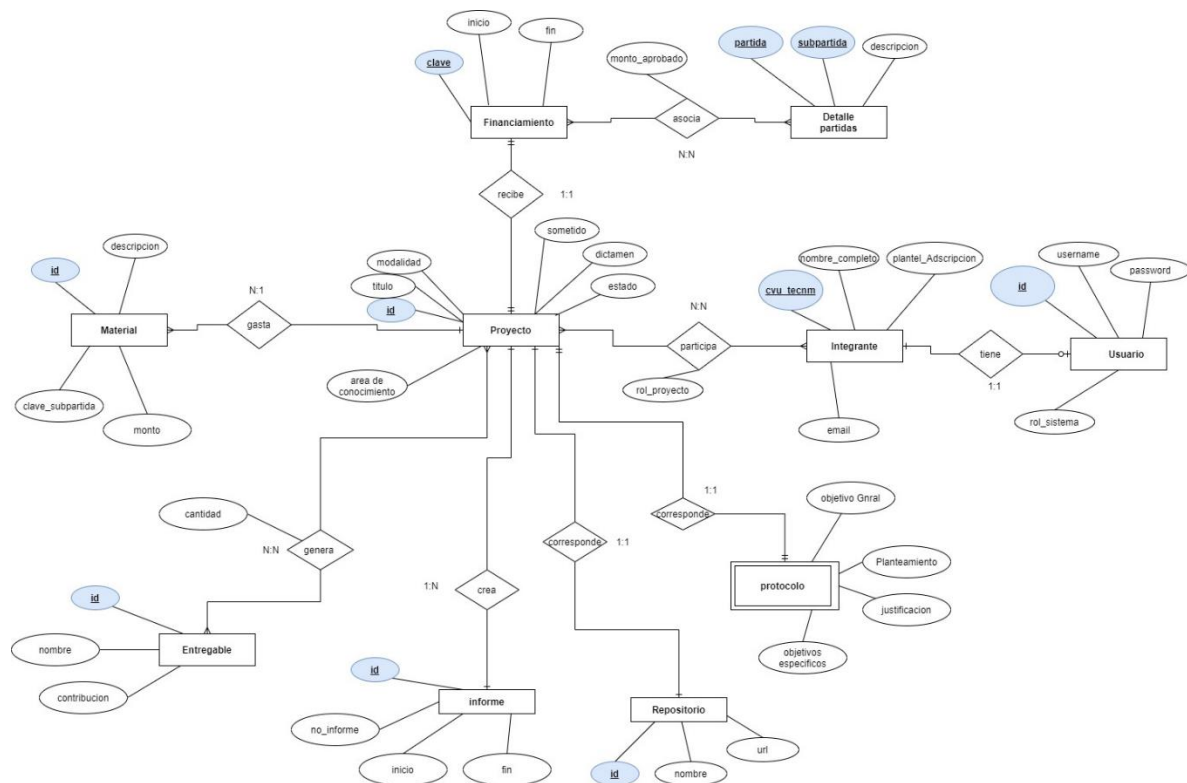
B. No funcionales.

- Sera necesaria la clave de financiamiento dada por el TecNM para comenzar un proyecto.
- El proceso de subida de documentos debe apegarse al proceso predefinido de los proyectos.

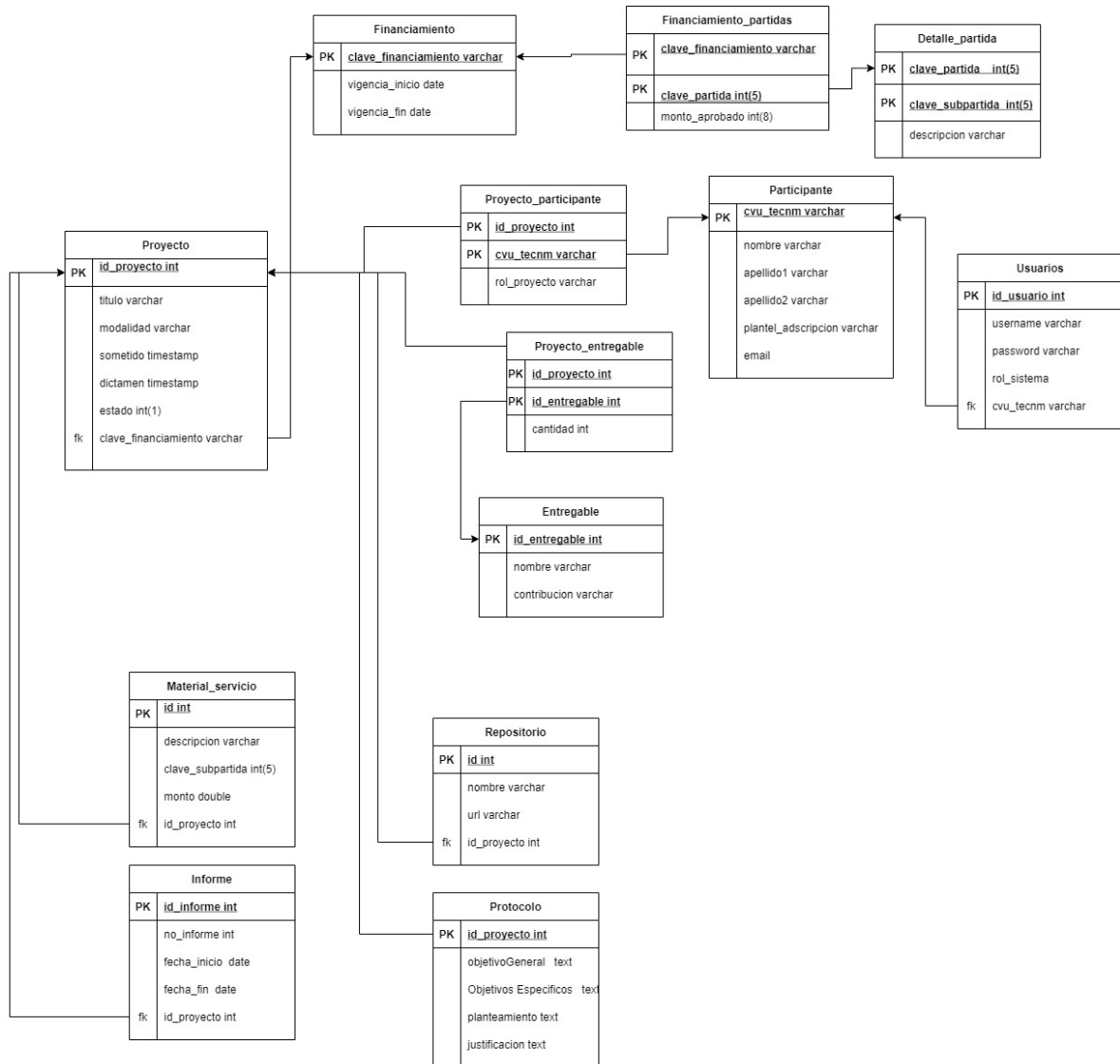
- El tamaño de los documentos será limitado la plataforma del TecNM

3.5.2 Etapa de diseño.

Diagrama E-R.



Modelo relacional.



Interfaces graficas de usuario.

Departamento de Estudios de Posgrado e Investigación.

Plataforma de Gestión de Proyectos de Investigación.



Bienvenida: tiene la finalidad de servir como recibidor, antes de ingresar al sistema.



Nombre de usuario

Contraseña

 Iniciar sesión

Login: permite el ingreso al sistema mediante un usuario y una contraseña.



Dashboard: las características principales del sistema se encuentran en esta ventana.

Crear Nuevo Participante.

* Campo obligatorio

No. CVU TechM

Nombre(s)

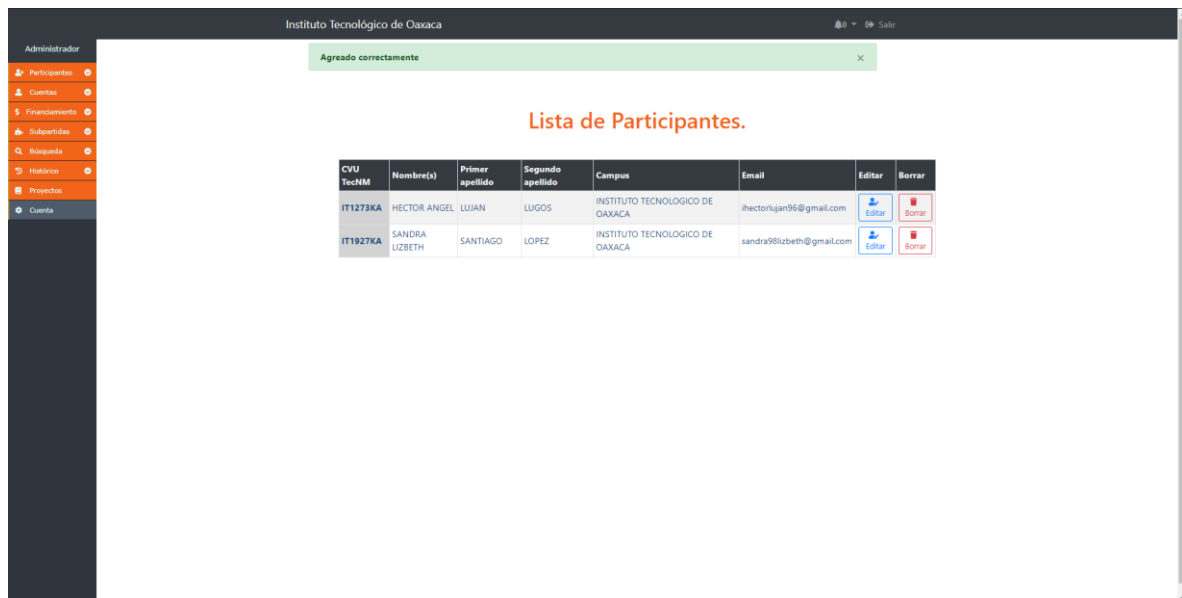
Primer apellido

Segundo apellido

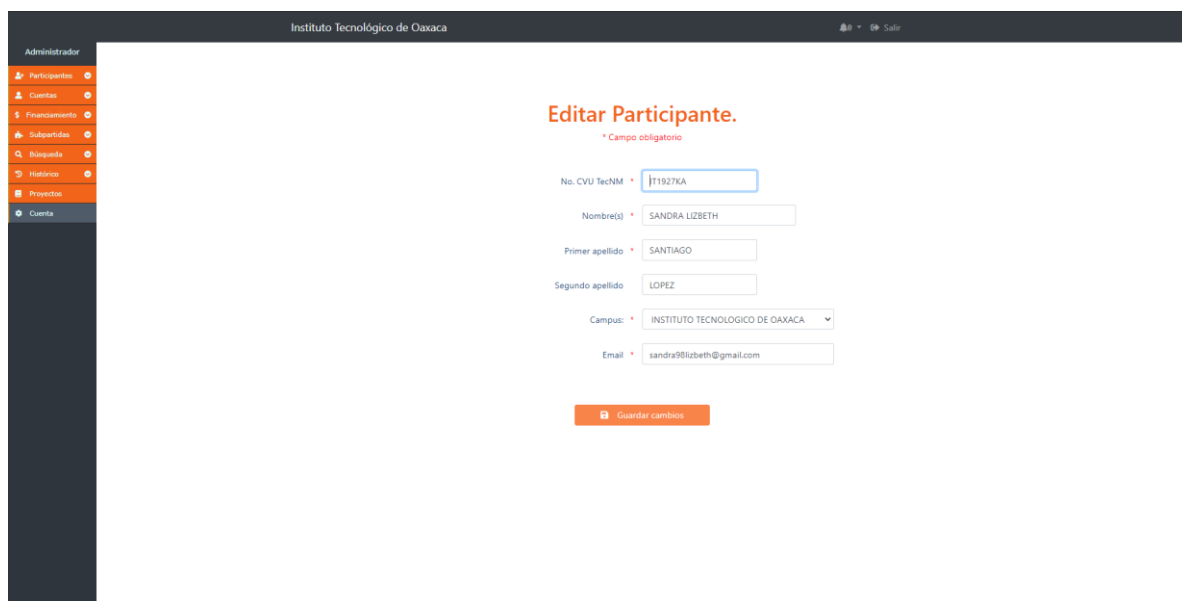
Campus

Email

Crear participante: permite registrar participantes al sistema para posteriormente darles acceso.



Lista de participantes: muestra la lista de participantes registrados en el sistema.



Editar participante: permite realizar cambios a los datos del usuario.

Responsable

Instituto Tecnológico de Oaxaca

Proyecto

Cuenta

Agregar Proyecto de Investigación.

Nombre del proyecto: Desarrollo de Sistema Web DEPI

Fecha de sometido: 01/03/2021

Fecha de dictaminado: 02/03/2021

Clave de financiamiento: 3131.03-P

+ Agregar

Agregar proyecto de investigación: permite agregar proyectos de investigación con base a una clave de financiamiento.

Administrador

Instituto Tecnológico de Oaxaca

Participantes

Cuentas

Financiamiento

Subproyectos

Búsqueda

Historia

Proyecto

Cuenta

Crear Cuenta a Participante.

Lista de participantes registrados.

SANDRA LIZBETH SANTIAGO LOPEZ

Si el usuario no se encuentra registrado haga click aquí

Correo electrónico: sandra98lzbeth@gmail.com

Contraseña generada automáticamente:

Seleccione un rol para el participante.

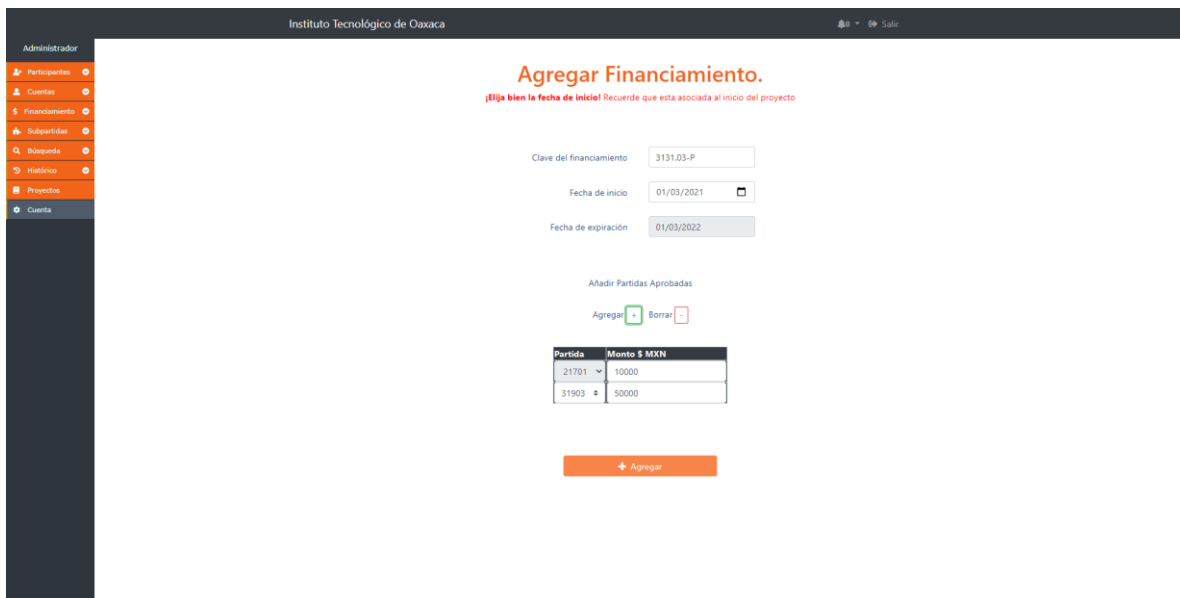
Lider de proyecto

Registrar

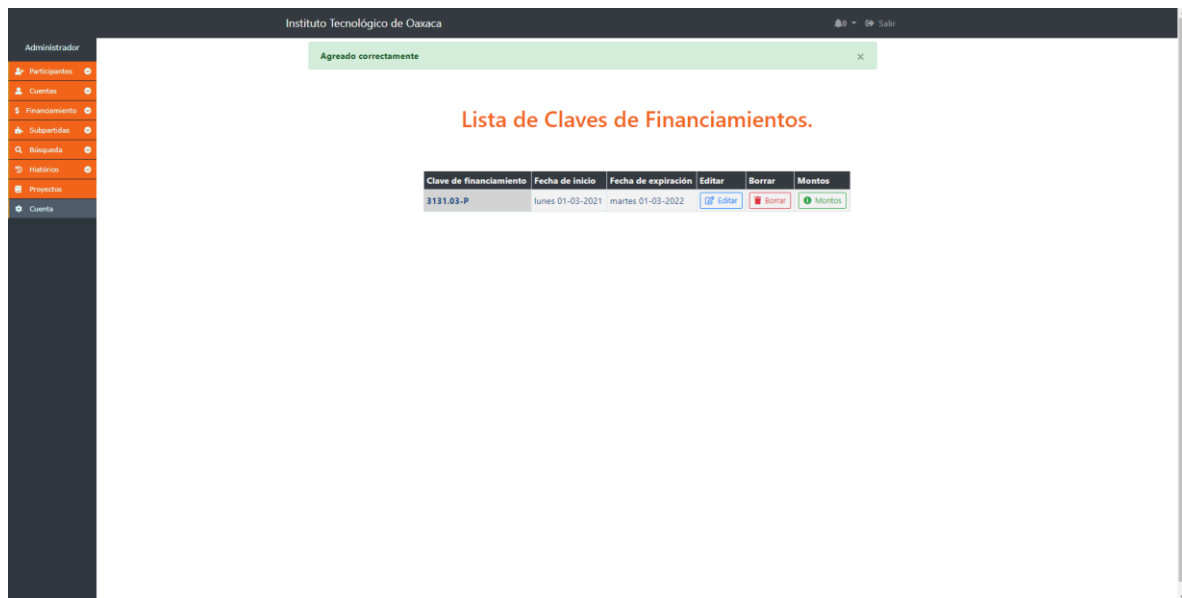
Crear cuenta de acceso a participante: permite dar acceso a los participantes previamente registrados.



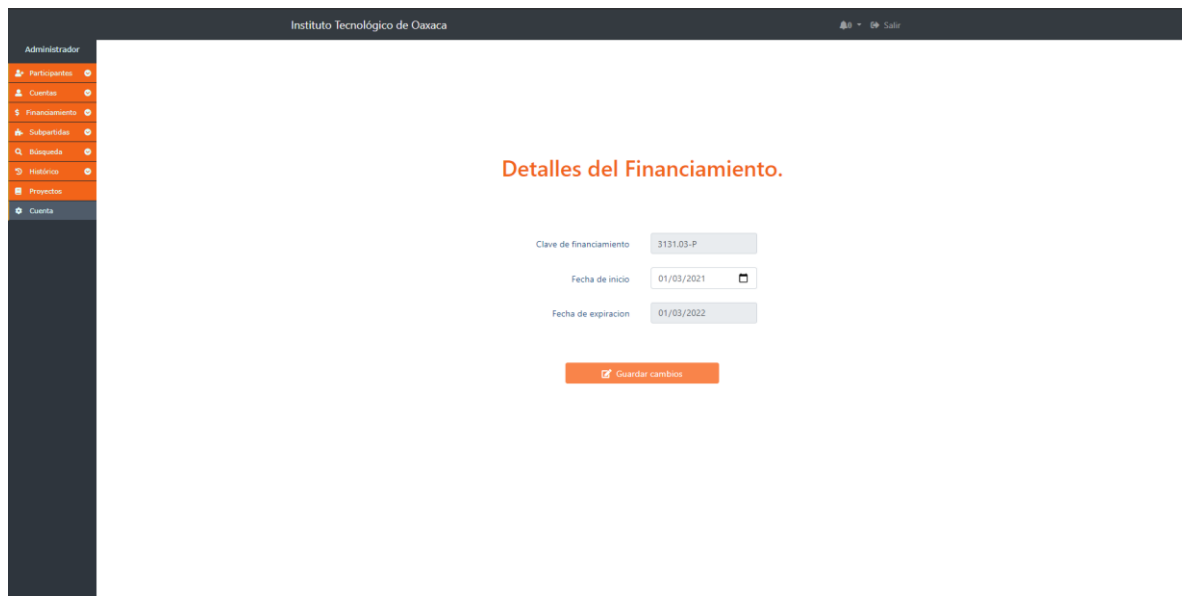
Lista de cuentas: permite identificar las cuentas de acceso activas y suspendidas.



Agregar financiamiento: Crea claves de financiamientos para los proyectos que van a ser creados.



Lista de financiamientos: permite visualizar las claves de financiamiento creadas para los proyectos y sus montos.



Editar financiamiento: permite modificar la fecha inicial del financiamiento:



Montos de financiamiento: permite visualizar los montos asignados a cada subpartida y editarlos.

3.5.3 Etapa de codificación.

3.5.3.1 Tecnologías a utilizar.

Nombre de la tecnología.	Versión de la tecnología.
Node.js	14.15.1
JavaScript	ECMAScript 2016
Express.js	4.17.1
CSS	CSS3
Bootstrap	4.0
Express-Handlebars	5.2.0
MySQL Community	8.0.23
Visual Studio Code	1.53.2
Git	2.29.2

3.5.3.2 Modulo destacado de la primera iteración.

```
const express = require("express");
const moment = require("moment");
const router = express.Router();
const conexion = require("../database");

//validacion con expressValidator
const { check, validationResult } = require("express-validator");

const {
  estaLogueado,
  noEstaLogueado,
  esAdministrador,
} = require("../lib/auth");

router.get("/add", esAdministrador, async (req, res) => {
  var consulta = await conexion.query(
    "select distinct clave_partida from detalle_partida"
  );
  //console.log(consulta);
  res.render("financiamiento/add1", { partidas: consulta });
});

//agrega el financiamiento junto con las partidas que le corresponden al financiamiento
//las partidas existentes son extraidos de detalle_partida
router.post(
  "/add",
  esAdministrador,
  [
    check("clave_financiamiento").notEmpty().toUpperCase(),
    check("clave_partida").notEmpty().withMessage("no se acepta vacio"),
    check("monto_aprobado").notEmpty().withMessage("no se acepta vacio"),
    check("vigencia_fin").notEmpty().withMessage("formato de fecha"),
    check("vigencia_inicio")
      .notEmpty()
      .isDate()
      .withMessage("formato de fecha"),
  ],
  async (req, res) => {
    const errores = validationResult(req);
    //console.log(errores.array());
    if (errores.array().length > 0) {
      return res.status(400).json({ errores: errores.array() });
    }
  }
);
```

```

} else {
  //console.log(req.body);

  const {
    clave_financiamiento,
    vigencia_inicio,
    vigencia_fin,
    clave_partida,
    monto_aprobado,
  } = req.body;
  var final =
    vigencia_fin.substring(6, 10) +
    "-" +
    vigencia_fin.substring(3, 5) +
    "-" +
    vigencia_fin.substring(0, 2);
  //console.log(final);
  const validacion = await conexion.query(
    "select * from financiamiento where clave_financiamiento= ?",
    [clave_financiamiento]
  );
  if (validacion.length > 0) {
    req.flash(
      "message",
      "Ya existe un financiamiento con esa clave intente con una nueva"
    );
    res.redirect("/financiamiento/add");
  } else {
    const newFinanciamiento = {
      clave_financiamiento,
      vigencia_inicio,
      vigencia_fin: final,
    };

    await conexion.query("INSERT INTO financiamiento set ?", [
      newFinanciamiento,
    ]);

    if (clave_partida != null && monto_aprobado != null) {
      if (Array.isArray(clave_partida) && Array.isArray(monto_aprobado)) {
        for (const p in clave_partida) {
          const newPartida = {
            clave_financiamiento,
            clave_partida: clave_partida[p],

```

```

        monto_aprobado: monto_aprobado[p],
    };
    //insertar
    await conexion.query("INSERT INTO financiamiento_partida set ?", [
        newPartida,
    ]);
    }
    } else {
        const newPartida1 = {
            clave_financiamiento,
            clave_partida,
            monto_aprobado,
        };
        await conexion.query("INSERT INTO financiamiento_partida set ?", [
            newPartida1,
        ]);
    }
    }

    req.flash("success", "Agregado correctamente");

    res.redirect("/financiamiento");
    }
    }
    }
    );

//listar un financiamiento especifico con sus partidas correspondientes

router.get(
    "/partidas/:clave_financiamiento",
    esAdministrador,
    async (req, res) => {
        const { clave_financiamiento } = req.params;
        const fin_partidas = await conexion.query(
            "select * from financiamiento natural join financiamiento_partida where c
lave_financiamiento=?",
            [clave_financiamiento]
        );
        res.render("partida/list", {
            partidas: fin_partidas,
            financiamiento: clave_financiamiento,
        });
        console.log(fin_partidas);
    }

```

```

);

//listar de la base de datos
router.get("/", esAdministrador, async (req, res) => {
  const financiamiento = await conexion.query("SELECT * FROM financiamiento ");

  let financiamiento_editado = [];
  let actual = moment();
  for (const p in financiamiento) {
    let i = moment(financiamiento[p].vigencia_inicio);
    let f = moment(financiamiento[p].vigencia_fin);
    let vigencia_inicio = i.format("dddd DD-MM-YYYY");
    let vigencia_fin = f.format(" dddd DD-MM-YYYY");

    const a = {
      clave_financiamiento: financiamiento[p].clave_financiamiento,
      vigencia_inicio,
      vigencia_fin,
    };
    financiamiento_editado.push(a);
  }
  res.render("financiamiento/list", { financiamiento: financiamiento_editado });
});

//ELIMINAR
router.get(
  "/delete/:clave_financiamiento",
  esAdministrador,
  async (req, res) => {
    const { clave_financiamiento } = req.params;
    try {
      await conexion.query(
        "DELETE FROM financiamiento WHERE CLAVE_FINANCIAMIENTO=?",
        [clave_financiamiento]
      );
      //console.log(req.params.id_convocatoria);
      req.flash("success", clave_financiamiento + " Eliminado correctamente");
      res.redirect("/financiamiento");
    } catch (error) {
      req.flash(
        "message",
        clave_financiamiento +
        " Esta asociado a un proyecto no se puede eliminar"
      );
      res.redirect("/financiamiento");
    }
  }
);

```



```

    }
  }
);

router.get("/edit/:clave_financiamiento", esAdministrador, async (req, res) => {
  const { clave_financiamiento } = req.params;

  let validacion = await conexion.query(
    "select * from proyecto where clave_financiamiento=?",
    [clave_financiamiento]
  );
  if (validacion.length <= 0) {
    const nuevo = await conexion.query(
      "SELECT * FROM financiamiento WHERE CLAVE_FINANCIAMIENTO=?",
      [clave_financiamiento]
    );

    //console.log(nuevo[0]);
    var f_i = moment(nuevo[0].vigencia_inicio).format("YYYY-MM-DD");
    var f_f = moment(nuevo[0].vigencia_fin).format("YYYY-MM-DD");

    // console.log(f_i,f_f);
    const aux = {
      clave_financiamiento: nuevo[0].clave_financiamiento,
      vigencia_inicio: f_i,
      vigencia_fin: f_f,
    };

    res.render("financiamiento/edit", { financiamiento: aux });
  } else {
    req.flash(
      "message",
      "Ya no se pueden realizar modificaciones, ya tiene asignado un proyecto"
    );
    res.redirect("/financiamiento");
  }
  //res.send("recibido");
});

router.post(
  "/edit/:clave_financiamiento",
  esAdministrador,
  async (req, res) => {
    const { clave_financiamiento } = req.params;

```

```

const { vigencia_inicio, vigencia_fin } = req.body;
const newFinanciamiento = {
  vigencia_inicio,
  vigencia_fin,
};

await conexion.query(
  "UPDATE financiamiento set ? WHERE CLAVE_FINANCIAMIENTO= ? ",
  [newFinanciamiento, clave_financiamiento]
);
//console.log(newFinanciamiento);
req.flash("success", "Cambios guardados para " + clave_financiamiento);
res.redirect("/financiamiento");
}
);

function findsubstr(str) {
  var substring = str.substr(0, 6);
  s;

  console.log(substring);
}

module.exports = router;

```

3.5.3.3 Script de la base de datos.

```

CREATE DATABASE residenciap_depi;
USE residenciap_depi;

create table convocatoria(
id_convocatoria int(5) auto_increment primary key,
nombre_convocatoria varchar(200) not null,
anio int(4) not null,
fecha_cierre timestamp
);

create table financiamiento(
clave_financiamiento varchar(10) primary key,
vigencia_inicio date not null,
vigencia_fin date not null
);

create table detalle_partida(
clave_partida int(5) not null,

```

```

clave_subpartida int(5) not null ,
descripcion text not null,
primary key(clave_partida, clave_subpartida)
);

describe detalle_partida;

create table financiamiento_partida(
clave_financiamiento varchar(10) not null,
clave_partida int(5) not null,
monto_aprobado int(8) not null,
FOREIGN KEY (clave_financiamiento) REFERENCES financiamiento(clave_financiamient
o),
FOREIGN KEY (clave_partida) REFERENCES detalle_partida(clave_partida),
primary key(clave_financiamiento,clave_partida)
);

CREATE TABLE participante(
cvu_tecnm varchar(10) not null primary key,
nombre varchar(50) not null,
apellido1 varchar(50) not null,
apellido2 varchar(50) ,
plantel_adscripcion varchar(100) not null,
email varchar(100) not null unique
);

create table users(
id_usuario int(5) auto_increment primary key,
username varchar(60) not null unique,
password varchar(200) not null unique,
cvu_tecnm varchar(10) unique,
rol_sistema varchar(15) ,
estado int(1) not null,
foreign key (cvu_tecnm) references participante(cvu_tecnm)
);

describe users;

create table proyecto(
id_proyecto int(10) not null primary key auto_increment,
titulo varchar(500) not null,
fecha_sometido timestamp not null,
fecha_dictamen timestamp not null,
clave_financiamiento varchar(10) unique,

```

```

FOREIGN KEY (clave_financiamiento) REFERENCES financiamiento(clave_financiamient
o),
estado int(1) not null,
creado date not null
);

describe proyecto;

create table proyecto_participante (
id_proyecto int(10) not null,
cvu_tecnm varchar(10) not null,
FOREIGN KEY (cvu_tecnm) REFERENCES participante(cvu_tecnm),
FOREIGN KEY (id_proyecto) REFERENCES proyecto(id_proyecto),
rol_proyecto varchar(30) not null,
primary key(id_proyecto,cvu_tecnm)
);

create table informe(
id_informe int(5) auto_increment primary key,
no_informe int(1) not null,
fecha_inicio date not null ,
fecha_fin date not null,
id_proyecto int(10) not null,
foreign key (id_proyecto) references proyecto(id_proyecto)
);

describe informe;

create table entregable (
id_entregable int not null auto_increment primary key,
nombre varchar(500) not null,
contribucion varchar(100) not null
);
insert into entregable (nombre,contribucion) values
('Tesis concluidas de licenciatura','Formacion de recursos humanos'),
('Tesis concluidas de doctorado','Formacion de recursos humanos'),
('Tesis en desarrollo de doctorado','Formacion de recursos humanos'),
('Incorporacion de alumnos de licenciatura','Formacion de recursos humanos'),
('Alumnos residentes participantes en el proyecto','Formacion de recursos humano
s'),
('Articulos cientificos enviados en revistas arbitrarias','Productividad academi
ca'),
('Articulos en memoria de congreso enviados','Productividad academica');

create table proyecto_entregable(

```

```

id_proyecto int(10) not null,
id_entregable int not null,
cantidad int not null,
FOREIGN KEY (id_entregable) REFERENCES entregable(id_entregable),
FOREIGN KEY (id_proyecto) REFERENCES proyecto(id_proyecto),
primary key(id_proyecto,id_entregable)
);

create table material_servicio (
id_material_servicio int not null auto_increment primary key,
descripcionms varchar(500) not null,
id_proyecto int(10) not null,
clave_subpartida int(5) not null ,
monto_solicitado int(8) not null,
FOREIGN KEY (id_proyecto) REFERENCES proyecto(id_proyecto)
);

insert into detalle_partida (clave_partida,clave_subpartida,descripcion) values
(21701,21101,"Materiales y útiles de oficina"),
(21701,21201,"Materiales y útiles de impresión y reproducción"),
(21701,21301,"Material estadístico y geográfico"),
(21701,21401,"Materiales y útiles consumibles para el procesamiento en equipos y
bienes informáticos."),
(21701,21501,"Material de apoyo informativo"),
(21701,22201,"Productos alimenticios para animales"),
(21701,23101,"Productos alimenticios, agropecuarios y forestales adquiridos como
materia prima"),
(21701,23301,"Productos de papel, cartón e impresos adquiridos como materia prim
a"),
(21701,23401,"Combustibles, lubricantes, aditivos, carbón y sus derivados adquir
idos como materia prima"),
(21701,23501,"Productos químicos, farmacéuticos y de laboratorio adquiridos como
materia prima"),
(21701,23601,"Productos metálicos y a base de minerales no metálicos adquiridos
como materia prima"),
(21701,23701,"Productos de cuero, piel, plástico y hule adquiridos como materia
prima"),
(21701,23901,"Otros productos adquiridos como materia prima"),
(21701,23902,"Petróleo, gas y sus derivados adquiridos como materia prima"),
(21701,24501,"Vidrio y productos de vidrio"),
(21701,24601,"Material eléctrico y electrónico"),
(21701,24701,"Artículos metálicos para la construcción"),
(21701,24901,"Otros materiales y artículos de construcción y reparación"),
(21701,25101,"Productos químicos básicos"),
(21701,25201,"Plaguicidas, abonos y fertilizantes"),

```

```
(21701,25301,"Medicinas y productos farmacéuticos"),
(21701,25401,"Materiales, accesorios y suministros médicos"),
(21701,25501,"Materiales, accesorios y suministros de laboratorio"),
(21701,25901,"Otros productos químicos"),
(21701,26105,"Combustibles, lubricantes y aditivos para maquinaria, equipo de producción y servicios administrativos"),
(21701,29101,"Herramientas menores"),
(21701,29401,"Refacciones y accesorios para equipo de cómputo y telecomunicaciones"),
(21701,29501,"Refacciones y accesorios menores de equipo e instrumental médico y de laboratorio"),
(21701,29801,"Refacciones y accesorios menores de maquinaria y otros equipos"),
(31903,33301,"Servicios de desarrollo de aplicaciones informáticas"),
(31903,33304,"Servicios de mantenimiento de aplicaciones informáticas"),
(31903,33601,"Servicios relacionados con traducciones"),
(31903,33901,"Subcontratación de servicios con terceros"),
(31903,35301,"Mantenimiento y conservación de bienes informáticos"),
(31903,35401,"Instalación, reparación y mantenimiento de equipo e instrumental médico y de laboratorio"),
(31903,35702,"Mantenimiento y conservación de plantas e instalaciones productivas");
```

```
create table protocolo (
id_protocolo int not null auto_increment primary key,
nombre_archivo varchar(500) not null,
url_archivo varchar(700) not null,
anotaciones text ,
id_proyecto int(10) not null,
FOREIGN KEY (id_proyecto) REFERENCES proyecto(id_proyecto),
revisiones int(1) not null
);
```

```
/*integridad referencial*/
```

```
ALTER TABLE `residenciap_depi`.`users`
DROP FOREIGN KEY `users_ibfk_1`;
ALTER TABLE `residenciap_depi`.`users`
ADD CONSTRAINT `users_ibfk_1`
  FOREIGN KEY (`cvu_tecnm`)
  REFERENCES `residenciap_depi`.`participante` (`cvu_tecnm`)
  ON UPDATE CASCADE;
```

```
ALTER TABLE `residenciap_depi`.`proyecto_participante`
ADD CONSTRAINT `proyecto_participante_ibfk`
  FOREIGN KEY (`cvu_tecnm`)
```

```

REFERENCES `residenciap_depi`.`participante` (`cvu_tecnm`)
ON UPDATE CASCADE;

ALTER TABLE `residenciap_depi`.`proyecto_participante`
DROP FOREIGN KEY `proyecto_participante_ibfk_2`;
ALTER TABLE `residenciap_depi`.`proyecto_participante`
ADD CONSTRAINT `proyecto_participante_ibfk_2`
FOREIGN KEY (`id_proyecto`)
REFERENCES `residenciap_depi`.`proyecto` (`id_proyecto`)
ON DELETE CASCADE
ON UPDATE CASCADE;

create table archivo_informes(
id_proyecto int(10) not null,
id_informe int(5) not null,
url_archivo varchar(500) not null,
anotaciones text,
revisiones int(1) not null,
FOREIGN KEY (id_proyecto) REFERENCES proyecto(id_proyecto),
FOREIGN KEY (id_informe) REFERENCES informe(id_informe),
primary key(id_proyecto,id_informe)
);

ALTER TABLE protocolo
ADD intentos int(1) not null;

alter table archivo_informes
add intentos int(1) not null;

create table historico(
clave_financiamiento varchar(10) not null primary key,
titulo varchar(500) not null,
responsable varchar(200) not null,
anio int(4) not null,
url_archivo varchar(700) not null
);

create table notificaciones(
id_notificacion int not null auto_increment primary key,
destinatario varchar(10) not null,
mensaje varchar(800) not null,
leido int(1) not null
);

```

3.5.4 Etapa de pruebas.

En esta iteración se aplicaron los siguientes tipos de prueba para todos los elementos:

- Prueba unitaria: se focaliza en ejecutar cada módulo
- Prueba de integración: identifica errores introducidos por la combinación de programas probados unitariamente.
- Pruebas de aceptación: el cliente prueba y verifica que cumpla con sus expectativas.

3.5.4.1 Validaciones.

Validaciones para el login.



Nombre de usuario

Contraseña

 Completa este campo

 Iniciar sesion

En caso de tratar de iniciar sesión sin datos.



Nombre de usuario

ihectorlujan96@gmail.com

Contraseña

! Completa este campo

➔ Iniciar sesion

En caso de no ingresar la contraseña.



Nombre de usuario

ihectorlujan9

! Incluye un signo "@" en la dirección de correo electrónico. La dirección "ihectorlujan9" no incluye el signo "@".

.....

➔ Iniciar sesion

En caso de intentar iniciar sesión con un usuario invalido.

Contraseña incorrecta



En caso de iniciar sesión con una contraseña errónea.



Nombre de usuario

ihectorlujan96@gmail.com

Contraseña

.....

➔ Iniciar sesión

Forma correcta para el inicio de sesión.

Bienvenido ihectorlujan96@gmail.com



Departamento de Estudios de Posgrado e Investigación.

Plataforma de Gestión de Proyectos de Investigación.



Inicio de sesión exitoso.

ihectorlujan96@gmail.com
¿desea salir del sistema?

Aceptar Cancelar

Confirmación para cerrar sesión.

Validaciones para la creación de participantes.

Crear Nuevo Participante.

* Campo obligatorio

No. CVU TecNM *


Nombre(s) * ! Completa este campo

Primer apellido *

Segundo apellido

Campus * INSTITUTO TECNOLÓGICO DE OAXACA ▼

Email *

 Crear integrante

Crear Nuevo Participante.

* Campo obligatorio

No. CVU TecNM * IT169827


Nombre(s) * ! Completa este campo

Primer apellido *

Segundo apellido

Campus * INSTITUTO TECNOLÓGICO DE OAXACA ▼

Email *


 Crear integrante

En caso de tratar de agregar el participante sin algún campo obligatorio.

Crear Nuevo Participante.

* Campo obligatorio

No. CVU TecNM *	<input type="text" value="IT169827"/>
Nombre(s) *	<input type="text" value="GERARDO ANTONIO"/>
Primer apellido *	<input type="text" value="LUJAN"/>
Segundo apellido	<input type="text" value="LUGOS"/>
Campus *	<input type="text" value="INSTITUTO TECNOLOGICO DE OAXACA"/>
Email *	<input type="text" value="ihectorlujan97@gmail.com"/>

 Crear integrante

Manera correcta de agregar un nuevo participante.



Agregado correctamente




Mensaje de creación correcta.



Validaciones para la creación de proyectos.


Agregar Proyecto de Investigación.

Nombre del proyecto	<input type="text"/>
Fecha de sometido	<input type="text" value="dd/mm"/>  Completa este campo
Fecha de dictaminado	<input type="text" value="dd/mm/aaaa"/> 
Clave de financiamiento	<input type="text"/>

 Agregar



Agregar Proyecto de Investigación.

Nombre del proyecto	<input type="text" value="Desarrollo de Sistema Web DEPI"/>
Fecha de sometido	<input type="text" value="dd/mm/aaaa"/> 
Fecha de dictaminado	<input type="text" value="dd/mm/aaaa"/>  Completa este campo
Clave de financiamiento	<input type="text" value="3131.03-P"/>

 Agregar

En caso de no rellenar alguno de los campos del proyecto

Agregar Proyecto de Investigación.

Nombre del proyecto	<input type="text" value="Desarrollo de Sistema Web DEPI"/>
Fecha de sometido	<input type="text" value="01/03/2021"/> 
Fecha de dictaminado	<input type="text" value="02/03/2021"/> 
Clave de financiamiento	<input type="text" value="3131.03-P"/>


+ Agregar

Manera correcta de agregar un proyecto de investigación.

Validaciones para la asignación de cuentas.

Crear Cuenta a Participante.

Lista de participantes registrados.





Si el usuario no se encuentra registrado haga click aquí

Correo electrónico

Contraseña generada automáticamente

Seleccione un rol para el participante.



 Registrar

Manera correcta de asignar una cuenta a un participante.

rellene todos los campos

Aceptar

En caso de no rellenar alguno de los campos.

Agregado con éxito



Mensaje por creación de cuenta exitoso.

Validaciones para la creación de financiamientos.

Agregar Financiamiento.

¡Elija bien la fecha de inicio! Recuerde que esta asociada al inicio del proyecto

Clave del financiamiento

Fecha de inicio

Fecha de expiración

Añadir Partidas Aprobadas

Agregar Borrar

Partida	Monto \$ MXN
21701	<input type="text"/>

+ Agregar

Agregar Financiamiento.

¡Elija bien la fecha de inicio! Recuerde que esta asociada al inicio del proyecto

Clave del financiamiento

Fecha de inicio

Fecha de expiración

Añadir Partidas Aprobadas

Agregar Borrar

Partida	Monto \$ MXN
21701	<input type="text"/>

! Completa este campo


+ Agregar

Validaciones en caso de no rellenar campos de texto al agregar financiamiento.

Agregar Financiamiento.

¡Elija bien la fecha de inicio! Recuerde que esta asociada al inicio del proyecto


Clave del financiamiento

Fecha de inicio 

Fecha de expiración

Añadir Partidas Aprobadas

Agregar  Borrar 

Partida	Monto \$ MXN
21701 	<input type="text" value="15000"/>

 Agregar

Forma correcta de agregar un financiamiento.

Agregado correctamente



Mensaje por creación exitosa.

3.6 Segunda iteración.

3.6.1 Etapa de planeación.

Historias de usuario a realizar.

Iteración.	No. de historia de usuario.	Historia de usuario.
2da iteración.	5	Partidas y subpartidas
	6	Búsqueda.
	7	Reportes.

	8	Planificaciones.
	9	Contenido de proyectos.
	10	Observaciones.

3.6.2 Etapa de diseño.

Interfaces graficas de usuario.

Instituto Tecnológico de Oaxaca

Administrador

Participantes

Cuentas

Financiamiento

Subpartidas

Búsqueda

Historio

Proyectos

Cuenta

Agregar Subpartida.

Clave de la partida

Agregar subpartidas.

Agregar Borrar

Clave subpartida	Descripción
<input type="text" value="45007"/>	<input type="text" value="Donaciones"/>

Agregar subpartida: permite agregar una nueva partida y subpartida a las existentes.

Administrador

Participantes

Cuentas

Financiamiento

Subpartidas

Búsqueda

Historio

Proyectos

Cuenta

Instituto Tecnológico de Oaxaca

Salir

Glosario de Subpartidas.

Partida	Subpartida	Descripción	Editar	Borrar
21701	21101	Materiales y útiles de oficina	Guardar cambios	Borrar
21701	21201	Materiales y útiles de impresión y	Guardar cambios	Borrar
21701	21301	Material estadístico y geográfico	Guardar cambios	Borrar
21701	21401	Materiales y útiles consumibles pa	Guardar cambios	Borrar
21701	21501	Material de apoyo informativo	Guardar cambios	Borrar
21701	22201	Productos alimenticios para anima	Guardar cambios	Borrar
21701	23101	Productos alimenticios, agropecua	Guardar cambios	Borrar
21701	23301	Productos de papel, cartón e impr	Guardar cambios	Borrar
21701	23401	Combustibles, lubricantes, aditivo	Guardar cambios	Borrar
21701	23501	Productos químicos, farmacéutico	Guardar cambios	Borrar
21701	23601	Productos metálicos y a base de n	Guardar cambios	Borrar
21701	23701	Productos de cuero, piel, plástico	Guardar cambios	Borrar
21701	23901	Otros productos adquiridos como	Guardar cambios	Borrar
21701	23902	Petróleo, gas y sus derivados adq.	Guardar cambios	Borrar
21701	24501	Vidrio y productos de vidrio	Guardar cambios	Borrar
21701	24601	Material eléctrico y electrónico	Guardar cambios	Borrar
21701	24701	Artículos metálicos para la constr.	Guardar cambios	Borrar
21701	24901	Otros materiales y artículos de cor	Guardar cambios	Borrar
21701	25101	Productos químicos básicos	Guardar cambios	Borrar
21701	25201	Plaguicidas, abonos y fertilizantes	Guardar cambios	Borrar
21701	25301	Medicinas y productos farmacéuti	Guardar cambios	Borrar

Glosario de subpartidas: muestra la lista de todas las partidas existentes.

Administrador

Participantes

Cuentas

Financiamiento

Subpartidas

Búsqueda

Historio

Proyectos

Cuenta

Instituto Tecnológico de Oaxaca

Salir

Búsqueda por Integrantes.

Buscar

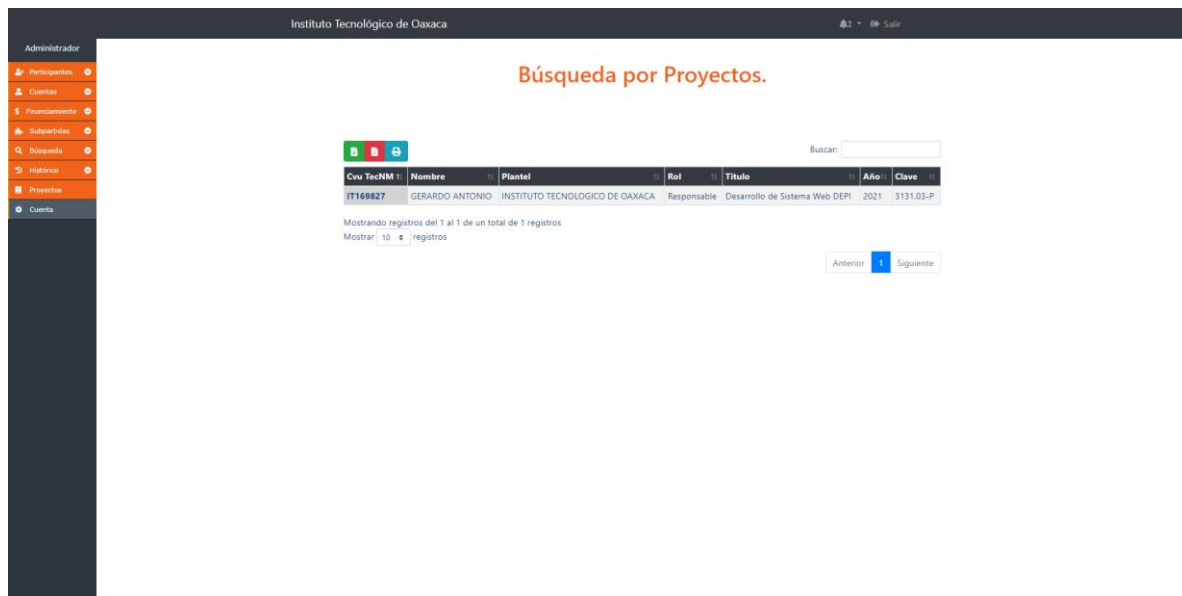
Clave CVU:	Nombre(s)	Primer apellido	Segundo apellido	Campus	Email
IT1273KA	HECTOR ANGEL	LUJAN	LUGOS	INSTITUTO TECNOLÓGICO DE OAXACA	ihectorlujan96@gmail.com
IT169827	GERARDO ANTONIO	LUJAN	LUGOS	INSTITUTO TECNOLÓGICO DE OAXACA	ihectorlujan97@gmail.com
IT1927KA	SANDRA LIZBETH	SANTIAGO	LOPEZ	INSTITUTO TECNOLÓGICO DE OAXACA	sandra98lzbeth@gmail.com

Mostrando registros del 1 al 3 de un total de 3 registros

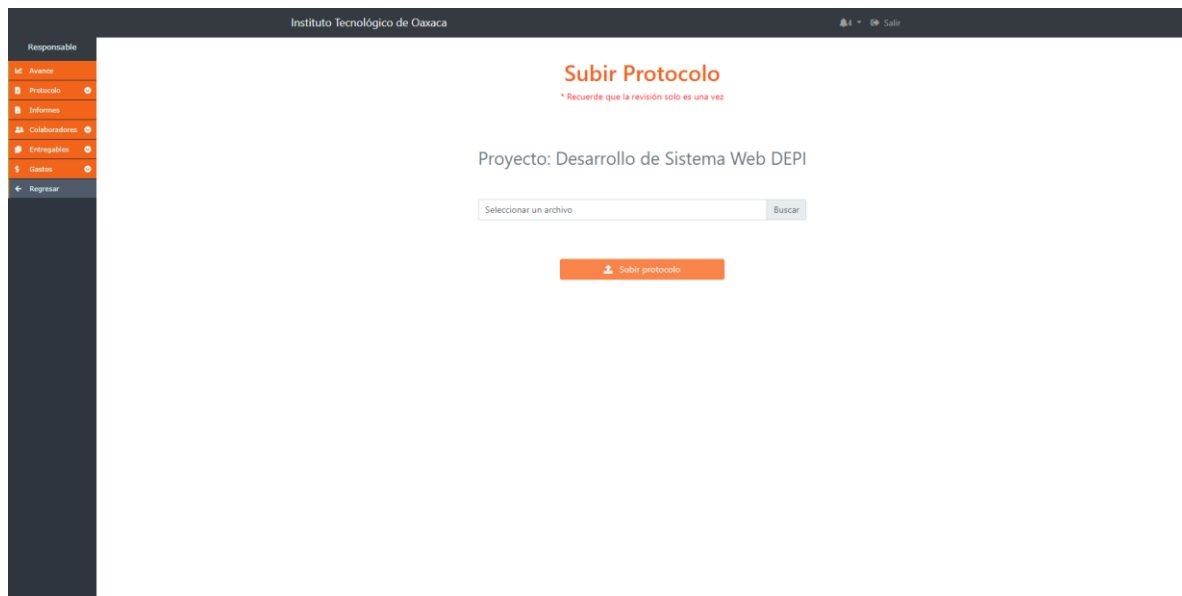
Mostrar 10 registros

Anterior 1 Siguiente

Búsqueda por integrantes: Muestra los integrantes y permite realizar búsquedas por cualquiera de los campos de la tabla.



Búsqueda por proyectos: Muestra los proyectos y permite realizar búsquedas por cualquiera de los campos de la tabla.



Subir protocolo: permite subir el protocolo de investigación de un proyecto específico.

Responsable

Instituto Tecnológico de Oaxaca

🔍 📄 📄 Subir

Responsable

Avance

Protocolo

Informes

Colaboradores

Entregables

Gastos

Regresar

Informes

No.	Fecha de inicio	Fecha final	Subir PDF	Previsualizar PDF
1	2021-05-25	2021-06-01	faltan 54 días para activarse	Previsualizar
2	2021-08-25	2021-09-01	faltan 146 días para activarse	Previsualizar
3	2021-11-24	2021-12-01	faltan 237 días para activarse	Previsualizar
4	2022-03-01	2022-04-01	faltan 334 días para activarse	Previsualizar

Informes: permite subir los cuatro informes de un proyecto de investigación, los botones de subir informe se activan durante una semana llegado el día.

Responsable

Instituto Tecnológico de Oaxaca

🔍 📄 📄 Subir

Responsable

Avance

Protocolo

Informes

Colaboradores

Entregables

Gastos

Regresar

Crear Nuevo Participante.

* Campo obligatorio

No. CVU TecNM *

Nombre(s) *

Primer apellido *

Segundo apellido

Campus *

Elija una opcion

Email *

Crear integrante

Colaboradores: permite crear y agregar nuevo participante al proyecto.

Responsable

Instituto Tecnológico de Oaxaca

8.1 + - Salir

Responsable

Avance

Protocolo

Informes

Colaboradores

Entregables

Gastos

Regresar

Agregar Entregables.

Añadir entregables

Agregar Eliminar

Entregable	Cantidad
Tesis concluidas de licenciatura	1

Entregables: permite detallar los entregables con los que se comprometió al iniciar el proyecto.

Responsable

Instituto Tecnológico de Oaxaca

8.1 + - Salir

Responsable

Avance

Protocolo

Informes

Colaboradores

Entregables

Gastos

Regresar

Lista de Entregables.

Nombre	Cantidad	Contribución
Tesis concluidas de licenciatura	1	Formación de recursos humanos

Lista de entregables: muestra la lista de entregables que tiene el proyecto.

Responsable

Avance

Protocolo

Informes

Colaboradores

Entregables

Gastos

Regresar

Instituto Tecnológico de Oaxaca

Agregar Materiales y Servicios.

Monto restante para justificar.
Clave: 21701 Monto restante \$ 0
Clave: 31903 Monto restante \$ 0

Materiales y Servicios

Agregar Eliminar

Descripcion	Subpartida	Monto \$ MXN
Compra en linea	Materiales y utiles de oficina	\$ 10000
Contratacion	Mantenimiento y conservaci3n de plantas e instalaciones productivas	\$ 50000

Agregar

Gastos: permite desglosar el monto de las subpartidas asignadas al proyecto en materiales y servicios.

Responsable

Avance

Protocolo

Informes

Colaboradores

Entregables

Gastos

Regresar

Instituto Tecnológico de Oaxaca

Lista de Materiales y Servicios.

Total: \$60000.00 MXN

Material / Servicio	Clave subpartida	Monto \$ MXN
Compra en linea	21101	\$ 10000.00 MXN
Contratacion	35702	\$ 50000.00 MXN

Lista de gastos: muestra en que se gasto la totalidad del monto de las subpartidas.

Administrador

Inicio

Protocolo

Informes

Entregables

Gastos

Colaboradores

Regresar

Instituto Tecnológico de Oaxaca

8.1 - 8 - Salir

Agregar Observaciones

* Recuerde que la revisión solo es una vez

Proyecto: Desarrollo de Sistema Web DEPI

Anotaciones:

Agregar observación

Agregar observaciones: permite realizar observaciones a protocolos o informes de proyectos de investigación en específico.

3.6.3 Etapa de codificación.

3.6.3.1 Modulo destacado de la segunda iteración.

```
const express = require("express");
const router = express.Router();
const conexion = require("../database");
const {
  estaLogueado,
  noEstaLogueado,
  esAdministrador,
} = require("../lib/auth");

router.get("/add", esAdministrador, async (req, res) => {
  res.render("subpartida/add");
});

router.post("/add", esAdministrador, async (req, res) => {
  const { clave_partida } = req.body;
  const { clave_subpartida } = req.body;
  const { descripcion } = req.body;

  //console.log(req.body);
```

```

try {
  if (Array.isArray(clave_subpartida) && Array.isArray(descripcion)) {
    for (const p in clave_subpartida) {
      const newSubpartida = {
        clave_partida,
        clave_subpartida: clave_subpartida[p],
        descripcion: descripcion[p],
      };
      //insertar
      await conexion.query("INSERT INTO detalle_partida set ?", [
        newSubpartida,
      ]);
    }
  } else {
    const newSubpartida1 = {
      clave_subpartida,
      clave_partida,
      descripcion,
    };
    await conexion.query("INSERT INTO detalle_partida set ?", [
      newSubpartida1,
    ]);
  }

  req.flash("success", "Agregado correctamente");

  //res.render("subpartida/add");
  res.redirect("/subpartida");
} catch (e) {
  req.flash("success", "Ya existe esa subpartida");

  //res.render("subpartida/add");
  res.redirect("/subpartida/add");
}
});

//listar todas
router.get("/", estaLogueado, async (req, res) => {
  const subpartidas = await conexion.query("SELECT * FROM detalle_partida ");
  console.log(subpartidas);

  res.render("subpartida/list", { subpartidas });
});

//eliminar

```

```

router.get("/delete/:clave_subpartida", esAdministrador, async (req, res) => {
  const { clave_subpartida } = req.params;
  await conexion.query(
    "DELETE FROM detalle_partida WHERE CLAVE_SUBPARTIDA=?",
    [clave_subpartida]
  );
  //console.log(req.params.id_convocatoria);
  req.flash("success", " Eliminado correctamente");
  res.redirect("/subpartida");
});

//editar
router.post("/edit/:clave_subpartida", esAdministrador, async (req, res) => {
  const { clave_subpartida } = req.params;
  const { clave_partida } = req.body;
  const { descripcion } = req.body;

  //console.log(req.body, req.params);
  //res.send('recibido');

  //const {vigencia_inicio,vigencia_fin}= req.body;
  const newSubpartida = {
    clave_subpartida,
    clave_partida,
    descripcion,
  };

  await conexion.query(
    "UPDATE detalle_partida set ? WHERE CLAVE_SUBPARTIDA=?",
    [newSubpartida, clave_subpartida]
  );
  //console.log(newFinanciamiento);
  req.flash("success", "Cambios guardados para ");
  res.redirect("/subpartida");
});

module.exports = router;

```

3.6.4 Etapa de pruebas.

En esta iteración se aplicaron los siguientes tipos de prueba para todos los elementos:

- Prueba unitaria: se focaliza en ejecutar cada módulo
- Prueba de integración: identifica errores introducidos por la combinación de programas probados unitariamente.
- Pruebas de aceptación: el cliente prueba y verifica que cumpla con sus expectativas.

3.6.4.1 Validaciones.

Validaciones para subpartidas.

Agregar Subpartida.

Clave de la partida

! Completa este campo

Agregar subpartidas.

Agregar + Borrar -

Clave subpartida	Descripción
<input type="text"/>	<input type="text"/>

+ Agregar

Agregar Subpartida.

Clave de la partida

Agregar subpartidas.

Agregar + Borrar -

Clave subpartida	Descripción
<input type="text" value="72003"/>	<input type="text"/>

! Completa este campo

+ Agregar

En caso de no rellenar algún campo al agregar una subpartida.

Agregar Subpartida.

Clave de la partida

Agregar subpartidas.

Agregar Borrar

Clave subpartida	Descripción
<input type="text" value="72003"/>	<input type="text" value="Enfermería"/>

Forma correcta de rellenar los campos.

Agreado correctamente



Mensaje de creación exitosa de subpartida.

Validaciones para los archivos del proyecto.

Ya tiene un archivo agregado desea remplazarlo?
En caso de ser así revise su archivo porque solo se permite 1 revision

Aceptar

Cancelar

En caso de querer subir nuevamente algún archivo.


Validaciones para los informes.


No.	Fecha de inicio	Fecha final	Subir PDF	Previsualizar PDF
1	2021-05-25	2021-06-01	faltan 54 días para activarse	
2	2021-08-25	2021-09-01	faltan 146 días para activarse	
3	2021-11-24	2021-12-01	faltan 237 días para activarse	
4	2022-03-01	2022-04-01	faltan 334 días para activarse	

Los botones para subir archivos únicamente permanecen activos durante la última semana del periodo.

Subir Informe

Proyecto: Desarrollo de Sistema Web DEPI

 Selecciona un archivo

 Subir archivo

En caso de dar clic en el botón sin agregar ningún archivo.

Validaciones para colaboradores.

Crear Nuevo Participante.

* Campo obligatorio

No. CVU TecNM *


Nombre(s) * ! Completa este campo

Primer apellido *

Segundo apellido

Campus * INSTITUTO TECNOLOGICO DE OAXACA ▼

Email *

 Crear integrante

Crear Nuevo Participante.

* Campo obligatorio

No. CVU TecNM * IT169827


Nombre(s) * ! Completa este campo

Primer apellido *

Segundo apellido

Campus * INSTITUTO TECNOLOGICO DE OAXACA ▼

Email *

 Crear integrante

En caso de tratar de agregar el participante sin algún campo obligatorio.

Crear Nuevo Participante.

* Campo obligatorio

No. CVU TecNM * IT169827


Nombre(s) * GERARDO ANTONIO

Primer apellido * LUJAN

Segundo apellido LUGOS

Campus * INSTITUTO TECNOLOGICO DE OAXACA ▼

Email * ihectorlujan97@gmail.com

 Crear integrante

Manera correcta de agregar un nuevo participante.

Agregado correctamente



Mensaje de creación correcta.

Validaciones para entregables.

Agregar Entregables.

Añadir entregables

Agregar + Eliminar -

Entregable	Cantidad
Tesis concluidas de doctorado	

Completar este campo

+ Agregar

En caso de tratar de agregar un entregable sin cantidad.

Validaciones para observaciones.

Recuerde que solo tenra una oportunidad de revisarlo

Aceptar Cancelar

Mensaje de alerta para observaciones

Proyecto: Desarrollo de Sistema Web DEPI

Anotaciones:

Completar este campo

Agregar observación

En caso de pulsar agregar observación sin texto.

3.7 Tercera iteración.

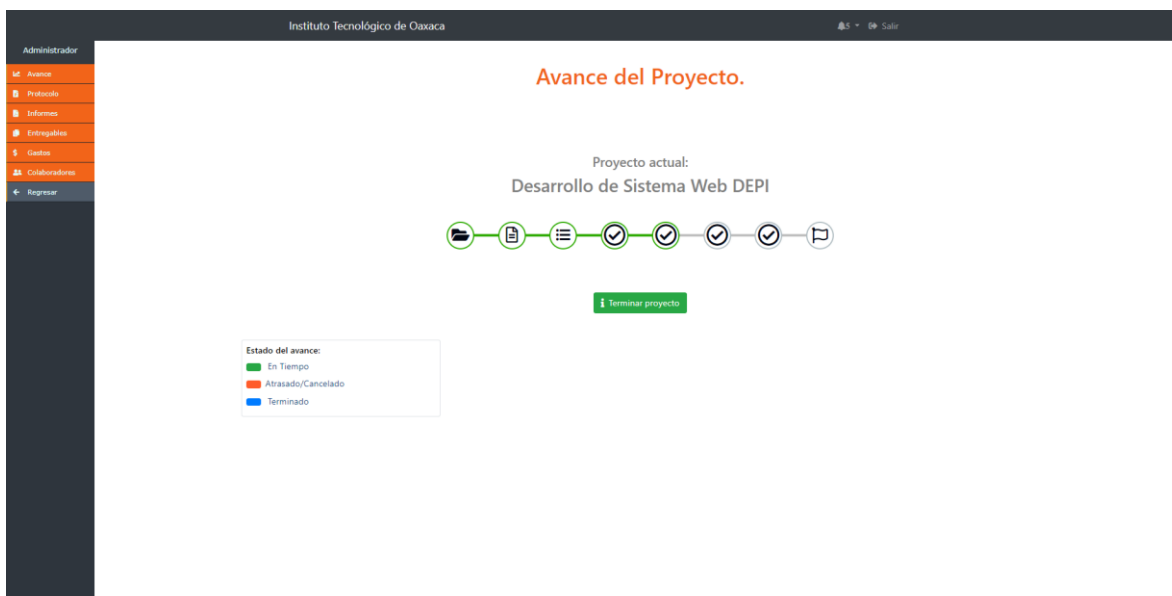
3.7.1 Etapa de planeación.

Historias de usuario a realizar.

Iteración.	No. de historia de usuario.	Historia de usuario.
3ra iteración.	11	Seguimiento de proyectos.
	12	Semaforización.
	13	Repositorio.
	14	Notificaciones.
	15	Restablecimiento de contraseñas.

3.7.2 Etapa de diseño.

Interfaces graficas de usuario.



Avance del proyecto: marca el estado actual del proyecto y el punto en el que se encuentra.

Instituto Tecnológico de Oaxaca

Administrador

Lista de Proyectos.

Filtro por estado:

todos En Tiempo Atrasado Terminado Cancelado

Nombre del proyecto	Fecha sometido	Fecha dictamen	Financiamiento	Responsable	Detalle
Desarrollo de Sistema Web DEPI	lunes 01 de marzo 2021	martes 02 de marzo 2021	3131.05-P	GERARDO ANTONIO	i Detalles

Semaforización: permite visualizar por colores el estado de todos los proyectos.

Instituto Tecnológico de Oaxaca

Administrador

Agregar Proyecto Histórico

Clave de financiamiento: 123123NOW

Nombre del proyecto: Proyecto de desarrollo historico

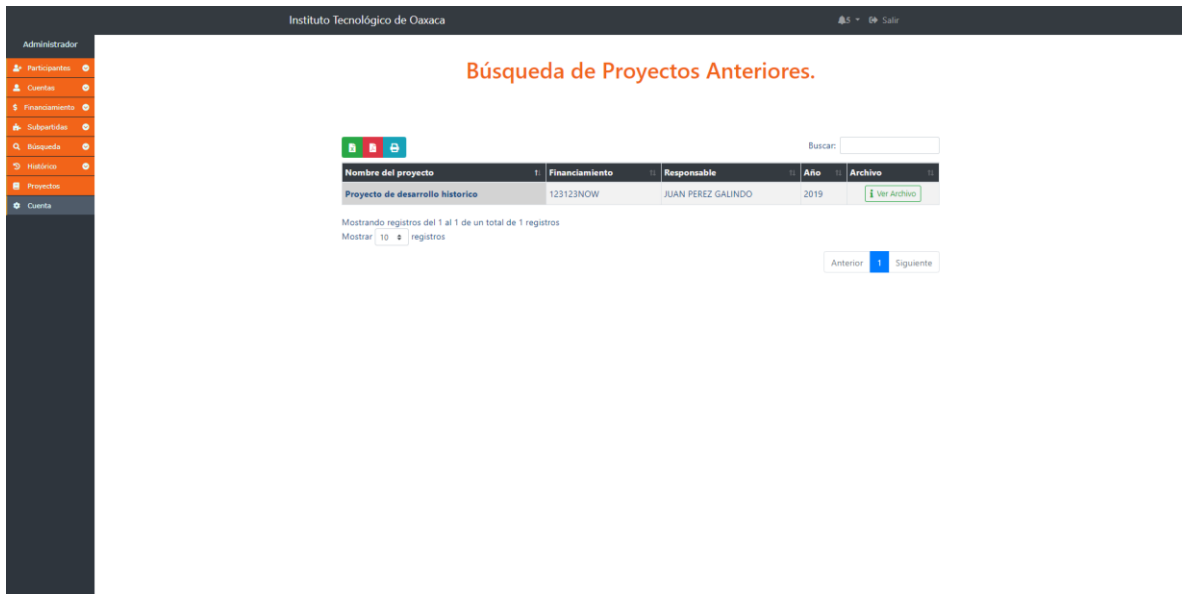
Nombre del responsable: JUAN PEREZ GALINDO

Año de realización: 2019

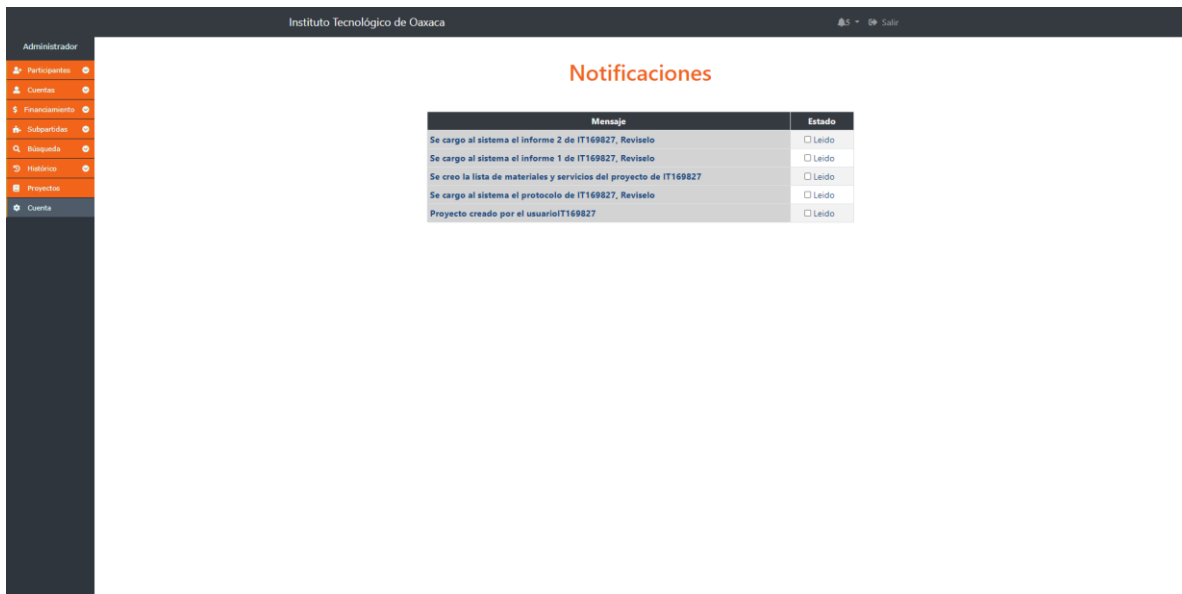
Archivo PDF: Gestion de Proyectos de Investigacion.pdf [Buscar](#)

[+ Agregar](#)

Histórico: permite agregar proyectos anteriores a la creación del sistema.



Lista de proyectos históricos: permite buscar los proyectos del repositorio por los campos mostrados.



Notificaciones: muestra alertas importantes referentes a los proyectos de investigación y permite marcarlos como leídas o conservarlas.

Administrador Instituto Tecnológico de Oaxaca 8.5 - Sair

- Participantes
- Cuentas
- Financiamiento
- Subarbitros
- Búsqueda
- Histórico
- Proyectos
- Cuenta

Cambiar Contraseña.

Correo electrónico

Contraseña actual

Nueva contraseña

Confirme la contraseña

[Cambiar contraseña](#)

Cuenta: permite modificar la contraseña del usuario activo.

3.7.3 Etapa de codificación.

3.7.3.1 Modulo destacado de la tercera iteración.

```
const express = require("express");
const router = express.Router();
const cron = require("node-cron");
const conexion = require("../database");
let cronm = "";
let estado = 0;

//inicio principal
router.get("/", async (req, res) => {
  let notificaciones;
  if (req.user) {
    notificaciones = await conexion.query(
      "select * from notificaciones where destinatario=? and leido=0 order by id_notificacion desc",
      [req.user.cvu_tecnm]
    );
    console.log(notificaciones);
  }
});
```

```

req.app.locals.notificaciones = notificaciones;

if (estado == 0) {
  ejecutarSiempre(req, res);

  cronm.start();
  estado = 1;
} else {
  console.log("es mas de 0 ", estado);
}
} else {
  if (cronm !== "") {
    cronm.stop();
    estado = 0;
  }
}

res.render("../index", { notificaciones: notificaciones });
});

async function ejecutarSiempre(req, res) {
  cronm = cron.schedule(
    "1 * * * * *",
    async function () {
      var usuario = req.user.cvu_tecnm;

      var x = await conexion.query(
        "select * from notificaciones where destinatario=? and leido=0 order by
id_notificacion desc",
        [usuario]
      );

      req.app.locals.notificaciones = x;

      console.log("contenido", req.app.locals.notificaciones);
    },
    {
      scheduled: false,
    }
  );
}

router.get("/notificaciones/list", async (req, res) => {
  let usuarioActual = req.user.cvu_tecnm;
  const notificacionX = await conexion.query(

```

```

    "select * from notificaciones where destinatario=? and leído=0 order by id_n
otificacion desc",
    [usuarioActual]
  );
  req.app.locals.notificaciones = notificacionX;

  res.render("notificaciones/listaNotificaciones", {
    notificaciones: notificacionX,
  });
});

router.get("/notificaciones/update/:id_notificacion", async (req, res) => {
  let { id_notificacion } = req.params;

  await conexion.query(
    "UPDATE notificaciones set ? WHERE id_notificacion=?",
    [{ leído: 1 }, id_notificacion]
  );

  res.redirect("/notificaciones/list");
});

module.exports = router;

```

3.7.4 Etapa de pruebas.

En esta iteración se aplicaron los siguientes tipos de prueba para todos los elementos:

- Prueba unitaria: se focaliza en ejecutar cada módulo
- Prueba de integración: identifica errores introducidos por la combinación de programas probados unitariamente.
- Pruebas de aceptación: el cliente prueba y verifica que cumpla con sus expectativas.

3.7.4.1 Validaciones.

Validaciones de proyectos históricos.

Agregar Proyecto Histórico

Clave de financiamiento

Nombre del proyecto ! Completa este campo

Nombre del responsable

Año de realización

Archivo PDF

Agregar Proyecto Histórico

Clave de financiamiento

Nombre del proyecto

Nombre del responsable

Año de realización

Archivo PDF ! Selecciona un archivo

En caso de no completar alguno de los campos y dar al botón agregar.

Agregar Proyecto Histórico

Clave de financiamiento

Nombre del proyecto

Nombre del responsable

Año de realización


Archivo PDF

Forma correcta de registrar un proyecto histórico.

Validaciones para cambios de contraseña.

Cambiar Contraseña.

Correo electrónico	<input type="text" value="ihectorlujan96@gmail.com"/>
Contraseña actual	<input type="password"/>
Nueva contraseña	<input type="password" value="! Completa este campo"/>
Confirme la contraseña	<input type="password"/>

 Cambiar contraseña

En caso de no completar algún campo dar clic en cambiar contraseña.


No coincide la confirmacion de contraseña

Aceptar

Alerta en caso de que las contraseñas no coincidan.

Cambiar Contraseña.

Correo electrónico	<input type="text" value="ihectorlujan96@gmail.com"/>
Contraseña actual	<input type="password" value="....."/>
Nueva contraseña	<input type="password" value="....."/>
Confirme la contraseña	<input type="password" value="....."/>

 Cambiar contraseña

Manera correcta de cambiar la contraseña.

Cambios guardados para ihectorlujan96@gmail.com

×

Mensaje de cambio de contraseña exitoso.

CAPITULO IV. CONCLUSIONES Y RECOMENDACIONES.

4.1 Conclusiones generales.

El software desarrollado en este proyecto impactara de manera positiva en los tiempos de trabajo del Departamento de Estudios de Posgrado e Investigación (DEPI), debido a la capacidad que tiene el software de integrar todas las funcionalidades requisitadas en características únicas y diferentes, con el fin de aprovechar todas las opciones que se implementaron en él, opciones que van desde el aspecto de presentación amigable al usuario final hasta el aspecto de la eficiencia del manejo de los datos.

Se halló que el uso de este software puede:

- Disminuir la cantidad de trabajo que se realiza.
- Mejorar los tiempos en los que se realiza una operación.
- Facilitar la consulta de los diferentes registros.
- Facilitar la adaptación a nuevos empleados.

Como producto final el sistema desarrollado presenta una idea innovadora para el DEPI. Ya que actualmente no se permiten una gestión fácil y directa a los contenidos.

4.2 Conclusiones personales.

El sistema ofrece mejoras notables desde el primer momento de su uso, todas las características están bien ubicadas para llevar el flujo sin confusiones. El impacto

que se tendrá dentro del Departamento de Estudios de Posgrado e Investigación (DEPI) será comprobable desde el inicio, además de que la comodidad de usar y administrar este sistema será casi perfecta. Sin duda alguna el equipo de desarrolladores queda satisfecho con el resultado y el proceso, es un software hecho a medida sumamente eficiente.

4.3 Recomendaciones.

Se recomienda el uso del sistema de manera seria y constante, hacer caso a cada una de las alertas y notificaciones en tiempo y forma, ya que esto le evitara contratiempos como usuario. De la misma manera, administrar su cuenta es responsabilidad de cada uno de los usuarios, todo esto nos llevará a un uso más cómodo de la plataforma y nos permitirá visualizar todas las características con las que fue diseñada.

REFERENCIAS BIBLIOGRAFICAS.

Ibáñez, L. H., Cabrera, J. L. R., Galindo, L. H. I., & TOME, A. G. (2011). *Administración de sistemas gestores de bases de datos: grado superior*. Alianza Editorial.

Roger S. Pressman, & Pressman, R. S. (2010). *INGENIERIA DE SOFTWARE*. McGraw-Hill Education.

Schulz, R. (2021). *Diseño Web Con Css*. Alfaomega Grupo Editor.