



TECNOLÓGICO NACIONAL DE
MÉXICO EN CELAYA



INFORME TÉCNICO DE RESIDENCIAS PROFESIONALES

Mezfer Insider: Renovación digital

13 de julio de 2024 – 13 de diciembre de 2024

IBT Innova Business Technology

Presenta:

Cristhian Alberto Ortega Hernández

20030191

Ingeniería en Sistemas Computacionales

Asesor Externo

Jorge Octavio Luna Medrano

Asesor Interno

Claudia Mayela Alcaraz Avendaño

Celaya Guanajuato, a 13 de enero de 2025

Índice

	Pág.
1 Introducción	1
2 Justificación	2
3 Objetivos	3
3.1 Objetivo General	3
3.2 Objetivos específicos	3
4 Problemas a resolver	4
5 Marco Teórico	5
5.1 Desarrollo Web	5
5.2 Frameworks	6
5.2.1 Frameworks para Frontend	6
5.2.1.1 Next.js	6
5.2.1.2 React.js	6
5.2.2 Frameworks para Backend	7
5.2.2.1 Node.js	7
5.2.2.2 Express.js	7
5.3 Servicios en la nube	8
5.3.1 Beneficios de los servicios en la nube	8
5.3.2 Tipos de servicios en la nube	9
5.3.2.1 Software como servicio (SaaS)	9
5.3.2.2 Plataforma como servicio (PaaS)	9
5.3.2.3 Infraestructura como servicio	9
6 Procedimiento y descripción de las actividades realizadas	11
6.1 Conociendo Mezfer Rewards	11
6.1.1 Aplicación web de Mezfer Rewards	11
6.1.2 Aplicación móvil de Mezfer Rewards	12
6.2 Análisis de la base de datos de Mezfer Rewards	13

6.2.1	Creación de la base de datos de Mezfer Insider	13
6.3	Selección de tecnologías a utilizar	14
6.3.1	Servicios en la nube	14
6.3.2	Tecnologías de frontend	14
6.3.3	Tecnologías de backend	15

1. Introducción

En este informe se documenta el desarrollo y los resultados obtenidos durante el proyecto de residencias profesionales, el cual permitió aplicar y consolidar los conocimientos adquiridos en el ámbito académico, enfrentando problemas reales del entorno profesional y ofreciendo soluciones prácticas y efectivas.

El contenido de este informe abarca desde los fundamentos que justifican el proyecto hasta los resultados y conclusiones finales. Se describen los objetivos específicos, las problemáticas detectadas y el procedimiento que se siguió para resolverlas. Además, se incluyen los resultados obtenidos mediante planos, prototipos, gráficas u otros elementos que permitan mostrar el alcance del trabajo realizado. Por último, se destacan las competencias desarrolladas y se presentan recomendaciones para el futuro del proyecto.

Este trabajo representa una valiosa oportunidad para aplicar habilidades técnicas y profesionales, proporcionando un aprendizaje significativo que contribuye al crecimiento profesional y la solución de problemas.

2. Justificación

Mezfer es una empresa mexicana dedicada a solucionar los problemas del campo mediante la elaboración de productos dirigidos al sector agroalimentario. El creciente número de clientes llevó a Mezfer a crear “Mezfer Rewards”, un programa de recompensas mediante el cual los clientes se inscriben a los diferentes ‘Programas de lealtad’ para canjear puntos que reciben con la compra de productos.

Actualmente, Mezfer Rewards es gestionado mediante 2 aplicaciones diferentes:

- Una aplicación web para que los administradores puedan realizar la gestión del programa.
- Una aplicación móvil para que los clientes puedan realizar todo el proceso para obtener recompensas.

Ahora, Mezfer busca que tanto clientes como administradores puedan realizar sus respectivas actividades en una misma plataforma para así mantener un mejor control. Con este objetivo establecido, comienza el desarrollo de “Mezfer Insider”, un portal para administradores y clientes que unifica las funcionalidades de las aplicaciones anteriores en un solo lugar.

3. Objetivos

3.1 Objetivo General

- Desarrollar e implementar la aplicación “Mezfer Insider” para centralizar y controlar las actividades de administradores y clientes.

3.2 Objetivos específicos

- Analizar la base de datos anterior para identificar áreas de mejora.
- Diseñar la nueva base de datos para Mezfer Insider.
- Integrar Mezfer Insider a Control Manager Siso (CMS) para que los administradores puedan iniciar sesión.
- Crear el inicio de sesión para los clientes de Mezfer.
- Crear el dashboard para los clientes.
- Integrar la funcionalidad de canje de puntos para los clientes.

4. Problemas a resolver

- Fragmentación de plataformas: Actualmente, el programa Mezfer Rewards es gestionado mediante dos aplicaciones separadas: una aplicación web para los administradores y una aplicación móvil para los clientes. Esta separación implica un manejo dividido e ineficiente de las funcionalidades.
- Descentralización: El hecho de que las actividades estén dispersas en diferentes plataformas dificulta la capacidad de mantener una visión centralizada y controlada de las operaciones.
- Falta de integración: Las actividades de los clientes y los administradores están divididas en distintos sistemas, lo que hace que la gestión de tareas sea poco eficiente.
- Falta de adaptación de funcionalidades: Se requiere unificar las funcionalidades de las aplicaciones web y móvil en una misma plataforma. Es importante que estas funcionalidades mantengan las características clave que ofrecen, como la gestión del programa y el acceso a las recompensas.
- Mejora de la experiencia del usuario: El usuario es una pieza fundamental en esta nueva plataforma, por lo que es importante asegurar que tanto administradores como clientes puedan realizar sus actividades eficientemente y sin problemas para ofrecer una experiencia agradable.

5. Marco Teórico

5.1 Desarrollo Web

El desarrollo web es el proceso de crear y mantener sitios web y abarca una gran variedad de acciones que van desde la creación de códigos y diseños, hasta la administración de contenidos y servidores. Para el desarrollo web se utilizan diferentes lenguajes de programación que ayudan a crear el código que hace que las páginas funcionen.

Entre los lenguajes de programación más comunes para el desarrollo web se encuentran:

- HyperText Markup Language (HTML): Este lenguaje es el componente más básico de la web, y ayuda a definir el significado y la estructura del contenido web.
- Cascading Style Sheet (CSS): Este lenguaje de estilos es utilizado para describir la presentación de documentos HTML o XML.
- Hypertext Preprocessor (PHP): Es un lenguaje de programación del lado del servidor que puede integrarse en HTML para crear aplicaciones y sitios web dinámicos.
- JavaScript: Es un lenguaje de programación que se ejecuta del lado del cliente de la web y es utilizado para programar cómo se comportan las páginas web cuando ocurre un evento. También puede ser ejecutado de lado del servidor, lo que permite que se puedan generar páginas web dinámicas.

El desarrollo web se puede dividir en tres categorías: Frontend, Backend y Full Stack.

El desarrollo Frontend es el responsable de la parte del cliente, es decir, lo que el usuario ve en pantalla cuando ingresa al sitio.

El desarrollo Backend se encarga de aspectos como servidores, bases de datos y lenguajes de programación. En esta parte se procesan las solicitudes del Frontend que contienen los datos para la base de datos u otros sistemas.

El desarrollo Full Stack es la combinación del Frontend y Backend, por lo que esta categoría es responsable tanto de la parte del cliente como de la parte del servidor.

5.2 Frameworks

En el mundo del desarrollo (y en muchas otras áreas) existen marcos de trabajo o frameworks que ofrecen una estructura base para elaborar un proyecto con objetivos específicos; es una especie de plantilla que sirve como punto de partida. El uso de un framework puede ayudar a realizar un proyecto en menos tiempo gracias a la reutilización de herramientas y módulos, y en la programación, también ayuda a tener un código más limpio y consistente.

Existen una gran diversidad de frameworks que son utilizados tanto para el desarrollo Frontend como el desarrollo Backend.

5.2.1 Frameworks para Frontend

Estos frameworks proporcionan una base para el desarrollo de la interfaz de usuario de las aplicaciones o páginas web. Hay una gran diversidad de frameworks frontend disponibles, lo que permite que se puedan crear soluciones de muy diferentes escalas y objetivos.

A continuación, se mencionan algunos de los que han sido utilizados para este proyecto:

5.2.1.1 Next.js

Next.js es un framework JavaScript ligero y de código abierto creado sobre React que permite construir sitios y aplicaciones web rápidos y sencillos de usar. Se utilizan los componentes de React para construir la interfaz de usuario mientras que Next.js se encarga de algunas características adicionales y optimizaciones.

Next.js también abstrae y configura automáticamente las herramientas que se requieren para React, como agrupación, compilación, entre otras cosas. Esto permite que los desarrolladores se enfoquen más en el desarrollo de la aplicación en lugar de gastar tiempo en configuraciones.

5.2.1.2 React.js

Para algunas personas, React es un framework pero en realidad, de acuerdo al sitio oficial, es una librería de JavaScript. Aunque no es un framework, se describirá en esta sección por ser una herramienta de gran importancia para el proyecto.

React.js es una librería de JavaScript de código abierto desarrollada por Facebook cuyo objetivo es simplificar el proceso de construir interfaces de usuario interactivas. Permite desarrollar las aplicaciones con la creación de componentes reutilizables; estos componentes son piezas individuales de una interfaz final.

El rol principal de React es encargarse de la capa de vista de una aplicación proveyendo la mejor y más eficiente ejecución de renderizado. Además, motiva a los desarrolladores a separar las interfaces en componentes individuales y reutilizables, de esta manera, React combina la velocidad y eficiencia de JavaScript con un método más eficiente de manipulación del DOM para renderizar las páginas web de manera más rápida.

5.2.2 Frameworks para Backend

Estos frameworks permiten simplificar algunos aspectos del proceso de desarrollo web, haciéndolo más fácil y rápido; ayudan al desarrollador a construir la arquitectura de su sitio web.

A continuación, se mencionan algunos de los que han sido utilizados para este proyecto:

5.2.2.1 Node.js

Node.js no es un framework, si no un entorno de ejecución para JavaScript, A pesar de esto, se describirá en esta sección debido a su importancia para el desarrollo del proyecto.

Node.js es un entorno de ejecución de JavaScript de código abierto y multiplataforma que permite a los desarrolladores crear servidores, aplicaciones web, herramientas para la línea de comandos y scripts.

Está basado en el motor de código abierto V8 de Google, el cual se actualiza constantemente y ofrece una gran rapidez, es por ello que se recomienda para el desarrollo de aplicaciones en tiempo real, las cuales se caracterizan por proporcionar información de manera instantánea.

5.2.2.2 Express.js

Express.js es un framework de backend para Node.js que proporciona características y herramientas robustas para desarrollar aplicaciones de backend escalables.

Este framework proporciona un conjunto de herramientas para aplicaciones web, peticiones y respuestas HTTP, enrutamiento y middleware para construir y desplegar aplicaciones a gran escala. Es utilizado para una gran variedad de cosas en el ecosistema JavaScript/Node.js; se pueden desarrollar aplicaciones, endpoints de APIs, sistemas de enrutamiento y frameworks.

Express.js es un framework no dogmático, es decir, un framework que no impone demasiadas restricciones sobre la mejor manera de unir componentes para alcanzar un objetivo. Es por esto que permite insertar casi cualquier middleware compatible dentro de la cadena del manejo de la petición, en cualquier orden; además, permite definir una estructura de directorios propia, por lo que se pueden crear tantas carpetas y archivos como se desee.

5.3 Servicios en la nube

Los servicios en la nube son recursos para aplicaciones e infraestructura que existen en internet, y permiten a los clientes aprovechar recursos de computación potentes sin la necesidad de adquirir o mantener hardware o software.

Cuando se utilizan los servicios en la nube, se puede delegar la gestión de la infraestructura y centrarse en utilizarla. El proveedor que se elija podrá proporcionar una amplia gama de actividades que mantendrán un negocio en funcionamiento. Cuando se emplean estos servicios, los usuarios autorizados podrán comunicarse, colaborar y gestionar proyectos, así como analizar, procesar, compartir y almacenar datos sin la necesidad de que alguien más tenga que supervisar, mantener o realizar copias de seguridad de la actividad.

5.3.1 Beneficios de los servicios en la nube

Los servicios en la nube ofrecen una gran variedad de beneficios para las empresas, entre los cuales se encuentran:

- **Escalabilidad:** Proporcionan la capacidad para escalar los recursos vertical y horizontalmente de forma instantánea en respuesta a la demanda.
- **Flexibilidad:** Se puede acceder a recursos informáticos y modificarlos con más agilidad, aumentando su capacidad para responder al uso de las unidades de negocio, los cambios en las circunstancias del mercado y las demandas de los clientes.

- **Rentabilidad:** Los mecanismos de precios con pago por consumo y la ausencia de inversiones iniciales en hardware e infraestructura mejoran la eficiencia de los costes de capital.
- **Seguridad:** Para ofrecer protección frente amenazas e infracciones, los proveedores de nube emplean fuertes características de seguridad como cifrado, límites al acceso y supervisión.

5.3.2 Tipos de servicios en la nube

Existen diferentes tipos de servicios que proporciona la nube, y en cada caso, los proveedores mantienen la infraestructura de la nube subyacente.

5.3.2.1 Software como servicio (SaaS)

En este servicio, los proveedores a los suscriptores el uso de su software ejecutándose sobre la infraestructura de la nube, lo que significa que la aplicación permite una amplia distribución y acceso. El proveedor se ocupa de cuestiones como la gestión y el control de la red, los servidores, sistemas operativos, almacenamiento, virtualización, datos, middleware e incluso capacidades de aplicaciones individuales. Las aplicaciones de SaaS se suelen diseñar para resultar fáciles de usar por un público más amplio.

5.3.2.2 Plataforma como servicio (PaaS)

Con PaaS, los usuarios tienen más control que con SaaS, porque obtienen acceso a un marco que empieza en el sistema operativo. Permite a los usuarios ubicar sus propias aplicaciones en la infraestructura de la nube con lenguajes de programación, bibliotecas, servicios y herramientas que admita el proveedor. El suscriptor dispone de control sobre las aplicaciones implementadas, los datos y, posiblemente, los ajustes de configuración del entorno de hospedaje. Sin embargo, la gestión y control de la red, los servidores, los sistemas operativos y el almacenamiento recaen en el proveedor.

5.3.2.3 Infraestructura como servicio

Con IaaS, los suscriptores pueden diseñar un entorno completo configurando una red virtual separada de otras redes. Los usuarios ejecutan un sistema operativo y aprovisionan el procesamiento, el almacenamiento, las redes y otros recursos informáticos fundamentales para ejecutar software en la infraestructura de la nube. IaaS también proporciona a

los suscriptores un control limitado de determinados de la red, y en ocasiones, tambien ofrecerán servicios como supervisión, automatización, seguridad, equilibrio de carga y resiliencia del almacenamiento.

6. Procedimiento y descripción de las actividades realizadas

6.1 Conociendo Mezfer Rewards

Mezfer Rewards es un programa de recompensas creado por la compañía Mezfer para poder entregar a sus clientes diferentes premios mediante el canje de puntos que obtienen gracias a la compra de productos. Para poder llevar a cabo este programa, Mezfer decidió crear una aplicación web y una aplicación móvil.

Para poder comenzar con el nuevo proyecto “Mezfer Insider” fue de gran importancia conocer ambas aplicaciones con las que funciona el proyecto, pues estas sirven como antecedentes y guías para tener una idea más sólida sobre como se va a estructurar el nuevo proyecto y todas las funciones que se van a mantener o eliminar.

A continuación, se describen de manera general ambas aplicaciones.

6.1.1 Aplicación web de Mezfer Rewards

La aplicación web fue desarrollada utilizando Laravel, un framework para el lenguaje de programación PHP. Esta aplicación es un panel de administración, por lo que sólo está pensada para ser utilizada por usuarios que sean administradores.

La aplicación está integrada a un sistema más grande llamado “Siso ERP”, un sistema integral de gestión empresarial que controla, facilita y optimiza los procesos operativos de las empresas. Por lo tanto, para poder acceder al panel de administración de Mezfer Rewards, es necesario tener una cuenta en Siso ERP.

Una vez que el administrador ha ingresado, podrá realizar todas las acciones necesarias para gestionar el programa. Entre las acciones más importantes se encuentran:

- Gestión de usuarios: El administrador es capaz de activar o desactivar la cuenta de un usuario, modificar su rol y suscribir o desuscribir al usuario de una campaña.
- Gestión de campañas: El administrador es capaz de modificar toda la información relevante de una campaña como su descripción, los productos participantes y los

premios disponibles.

- Gestión de solicitudes de puntos: El administrador es capaz de aceptar o rechazar una solicitud de puntos que haya enviado un cliente, de acuerdo a si la evidencia que envió es válida o no.

Hay muchas más acciones que puede realizar el administrador y que complementan al programa de recompensas. Si bien son acciones que también son importantes, no es fundamental describirlas para poder comprender mejor la aplicación.

6.1.2 Aplicación móvil de Mezfer Rewards

La aplicación móvil fue desarrollada con Flutter, un framework que permite desarrollar aplicaciones nativas para iOS y Android, y cuyo lenguaje de programación es Dart. Esta fue desarrollada específicamente para los clientes.

Al descargar e iniciar la aplicación, los clientes deberán ingresar su número telefónico y su contraseña para acceder; si no tienen cuenta, tendrán que registrarse.

Cuando el cliente ha iniciado sesión, accederá a la aplicación y podrá ver las campañas disponibles, así como otro contenido que podría ser de su interés. Es importante mencionar que la aplicación no cuenta con una función para inscribirse a la campaña que desee, pero esto no es una falla en el diseño, sino una decisión de la propia empresa; para que el cliente pueda inscribirse tendrá que comunicarse con un encargado de Mezfer y realizar la solicitud.

Si el cliente no está inscrito a ninguna campaña, sólo podrá revisar la información general como la descripción, los productos participantes y los premios disponibles; si el cliente está inscrito a alguna campaña, además de poder ver la información general, podrá realizar dos acciones más:

- Realizar solicitud de puntos: El cliente es capaz de enviar una solicitud de puntos, en donde especifica los productos que compró y la cantidad, así como la evidencia de la compra.
- Canjear puntos disponibles: El cliente es capaz de canjear los puntos que obtuvo con la compra de productos para poder obtener premios que sean de su interés.

Hay otras acciones que puede realizar el cliente y que complementan su experiencia en el uso de la aplicación. Si bien son acciones que también son importantes, no es fundamental describirlas para poder comprender mejor la aplicación.

6.2 Análisis de la base de datos de Mezfer Rewards

Para las aplicaciones anteriores de Mezfer Rewards se utilizó una base de datos relacional creada en PostgreSQL, un Sistema Gestor de Bases de Datos. Esta base de datos fue creada gracias a los servicios en la nube de Amazon Web Services (AWS), específicamente con su servicio “Amazon Relational Database Service” (Amazon RDS), un servicio que web que facilita la configuración, operación y escala de una base de datos relacional.

Esta base de datos contiene un total de 60 tablas, cada una con los atributos necesarios para almacenar la información relevante. El análisis que se realizó sirvió para identificar las tablas y los atributos que debían conservarse o que se debían eliminar, además de identificar áreas de mejora para la normalización de la base de datos.

6.2.1 Creación de la base de datos de Mezfer Insider

Para Mezfer Insider se decidió crear una nueva base de datos en la que se aplicarían los cambios que se identificaron en la base de datos anterior. Estos cambios no se realizaron en la antigua base de datos porque no se quería afectar el funcionamiento de las aplicaciones, ya que éstas debían seguir disponibles para los usuarios en lo que se termina el desarrollo de Mezfer Insider; realizar estos cambios implicaría también actualizar el código de las aplicaciones para que siguieran funcionando correctamente, y esto quitaría tiempo de desarrollo para el nuevo proyecto.

Esta nueva base de datos también fue desarrollada como una base de datos relacional utilizando PostgreSQL, y está almacenada en la nube gracias al servicio RDS de Amazon Web Services.

Además de los cambios, una nueva funcionalidad que se quiso añadir a la base de datos de Mezfer Insider fueron las bitácoras. El propósito de las bitácoras es permitir llevar un registro sobre qué usuarios han realizados cambios en la información de ciertas tablas, de esta manera se podrá rastrear más fácilmente a cualquier usuario que haya realizado una

modificación indebida y se podrá recuperar la información que haya sido cambiada.

Hasta el momento no se ha utilizado esta nueva base de datos pues no se ha iniciado el proceso de migración, pero próximamente será la única en utilizarse.

6.3 Selección de tecnologías a utilizar

Como último paso para comenzar con el desarrollo de Mezfer Insider, se analizaron diferentes tecnologías para usar en frontend y backend.

6.3.1 Servicios en la nube

La plataforma que se utilizó para proveer los servicios en la nube fue Amazon Web Services.

De esta plataforma se utilizaron servicios como:

- Amazon Elastic Compute Cloud (EC2): Es un servicio que proporciona capacidad de computación escalable bajo demanda en la nube. Permite lanzar cuantos servidores virtuales sean necesarios. Dos instancias de EC2 fueron utilizadas para lanzar el proyecto de Mezfer Insider y la API.
- Amazon Relational Database Service (RDS): Es un servicio web que facilita la configuración, la operación y la escala de una base de datos relacional en la nube. Se utilizó este servicio para crear la base de datos de Mezfer Insider.
- Amazon Simple Storage Service (S3): Es un servicio de almacenamiento de objetos que ofrece escalabilidad, disponibilidad de datos, seguridad y rendimiento líderes del sector. Este servicio fue utilizado para almacenar recursos importantes de Mezfer Insider, como archivos e imágenes.

6.3.2 Tecnologías de frontend

Para facilitar el desarrollo del frontend, se decidió comprar una plantilla de panel de usuario llamada DashTail, la cual proporciona vistas ya diseñadas o componentes para poder ir construyendo poco a poco la vista para el usuario; esta plantilla ofrece una alta personalización, por lo que es posible realizar cualquier cambio a los componentes.

DashTail fue desarrollada utilizando Next.js y Tailwind CSS.

Next.js es un framework basado en JavaScript y React que permite crear aplicaciones web modernas, dinámicas y escalables. Aunque es un framework full-stack, en DashTail sólo se utiliza para frontend.

Tailwind CSS es un framework de CSS para el diseño de páginas web. Su principal característica es que no genera una serie de clases predefinidas para elementos, en su lugar, crea una lista de clases “de utilidad” que son utilizadas para dar estilos individuales a cada elemento. Es gracias a este framework que los componentes de la plantilla son altamente personalizables.

6.3.3 Tecnologías de backend

El backend consistió en desarrollar una API para que el frontend pudiera comunicarse con esta y realizar peticiones HTTP para interactuar con la base de datos.

La principal tecnología de backend es Node.js, un entorno de ejecución utilizado para poder ejecutar código de JavaScript directamente en los servidores, es decir, sin la necesidad de un navegador web.

Junto a Node.js se utiliza Express.js, un framework de backend para Node.js que proporciona características y herramientas robustas para desarrollar aplicaciones de backend escalables. Sus herramientas como peticiones y respuestas HTTP, enrutamiento y middlewares resultaron muy útiles para el desarrollo de la API.