

Tecnológico de Monterrey

Coder's Tycoon

Documento de Especificación de Requerimientos

Equipo 1

A00835792 Ana Paula Figueroa Tardaguila

A01722533 Jorge Andres Palacios Plascencia

A01254672 Alejandro Barrera Bejarano

A01198630 Fidel Morales Briones

A00837590 Felipe de Jesús Ramos Santos

A01254678 Aldo Leonardo López Ruiz

[ER###]

[v#.##]

[3/mayo/2024]

Control de Versiones

Fecha	Versión	Descripción del cambio	Autor(es)	Autorizó
[3/may/24]	[v1.00]	Versión Inicial	Equipo 1	

Contenido

1. Introducción	4
1.1. Necesidades del negocio	4
1.2. Propósito del producto	4
1.3. Alcance del producto	4
2. Descripción del producto	4
2.1. Perspectiva del producto	4
2.2. Funcionalidades del producto	4
2.3. Características de los usuarios	4
2.4. Limitaciones, supuestos y /o dependencias	5
3. Base de datos	5
WEB	6
Unity	13
4. Requerimientos específicos	15
4.1. Requerimientos funcionales	15
4.2. Requerimientos de interfaces externas	19
4.3. Requerimientos de datos lógicos	20
4.4. Restricciones de diseño	21
4.5. Requerimientos no funcionales	21
5. Glosario	22

1. Introducción

1.1. Necesidades del negocio

Neoris, como empresa, busca alcanzar el mejor desempeño posible en el mercado. Para lograr esto, necesita empleados capacitados adecuadamente en sus respectivas áreas laborales. Por medio del uso continuo de plataformas como Coursera, es posible entrenar a los trabajadores en cualquier tema que requiera para completar satisfactoriamente sus tareas. Neoris reconoció el potencial de este tipo de herramientas, y ahora busca implementar su uso dentro de la empresa de una forma efectiva y fácil de administrar.

1.2. Propósito del producto

El reto que se presenta y que se identificó como equipo frente a esta situación es alentar a quienes están tomando cursos a mantenerse constantes en su uso de la plataforma, con el objetivo de maximizar su aprendizaje. Coder's Tycoon está diseñado con el propósito de incentivar esto, y al mismo tiempo facilitar al administrador la tarea de recopilar los datos recabados de cada empleado acerca de los cursos que van completando.

1.3. Alcance del producto

La aplicación web está dirigida a los trabajadores de Neoris que están en puestos administrativos y de desarrollo de aplicaciones, por esto la aplicación web contará con dos apartados para cada uno de ellos. El apartado de los desarrolladores contará con su perfil y un dashboard que contendrá cada curso y el avance que tienen en ellos, también tendrán un apartado de cursos en donde estarán "paths" profesionales y los cursos que le corresponden a cada path. También una pestaña tendrá un videojuego estilo "clicker" que servirá para incentivar el avance de los cursos. El apartado de los administradores tendrá un dashboard para ver las estadísticas de los trabajadores en los cursos, como el tiempo transcurrido dentro del juego, usuarios diarios dentro de la aplicación, cursos más populares, el promedio de avance en cada curso y las estadísticas de cada desarrollador individualmente. Esta aplicación web solo puede ser accedida desde un navegador web de escritorio.

2. Descripción del producto

2.1. Perspectiva del producto

La aplicación web está diseñada para navegadores web de escritorio, se conectará a una base de datos para obtener información de los usuarios como los datos de guardado del videojuego de los usuarios, logs, avance de los cursos etc. Los dashboards de la aplicación no necesitan de hardware poderoso para ser utilizados ni el videojuego debido a que va utilizar el motor de juego Unity en 2D que es bastante ligero.

2.2. Funcionalidades del producto

La funcionalidad de la aplicación desarrollada es una aplicación tipo juego bajo el concepto de un "clicker" el cual queremos lograr la motivación de los trabajadores de Neoris para completar los cursos y alcanzar los objetivos de aprendizaje. A través de las mecánicas que este juego Coder's Tycoon ofrece, el usuario podrá desbloquear puntos y una variedad de items y bonificaciones dentro del juego a medida en que progresa los cursos educativos. Al igual se ofrece una tabla tipo "leaderboard" para volver el juego más competitivo para que se consiga un progreso a base de competitividad. Al administrador se le permitirá entrar a una página en la cual podrá ver los progresos de todos los jugadores en el juego y en los cursos para asegurar que se esté cumpliendo el objetivo de la aplicación. Se busca la participación activa, el motivar el aprender a través de los cursos y recompensar el esfuerzo y compromiso profesional.

2.3. Características de los usuarios

Los usuarios existentes dentro de la aplicación son 2. Los Administradores y los trabajadores. Los administradores tienen acceso a toda la aplicación, ellos tienen acceso a un página de administrador las cuales le permitirá ver los cursos que los usuarios "trabajador" completan o no han completado. Al mismo tiempo los administradores

tienen acceso a la información dentro del juego el cual permitirá ver el desarrollo de cada uno de los trabajadores.

Los usuarios “trabajadores” tendrán un acceso más limitado, ellos tendrán acceso a sus perfiles, su juego, los cursos y el leaderboard. Esto les permitirá navegar a través de diversas páginas las cuales ocupan interactuar para poder contestar los cursos, mejorar sus personajes de juego, ver su progreso y ver su progreso en comparación al de los demás.

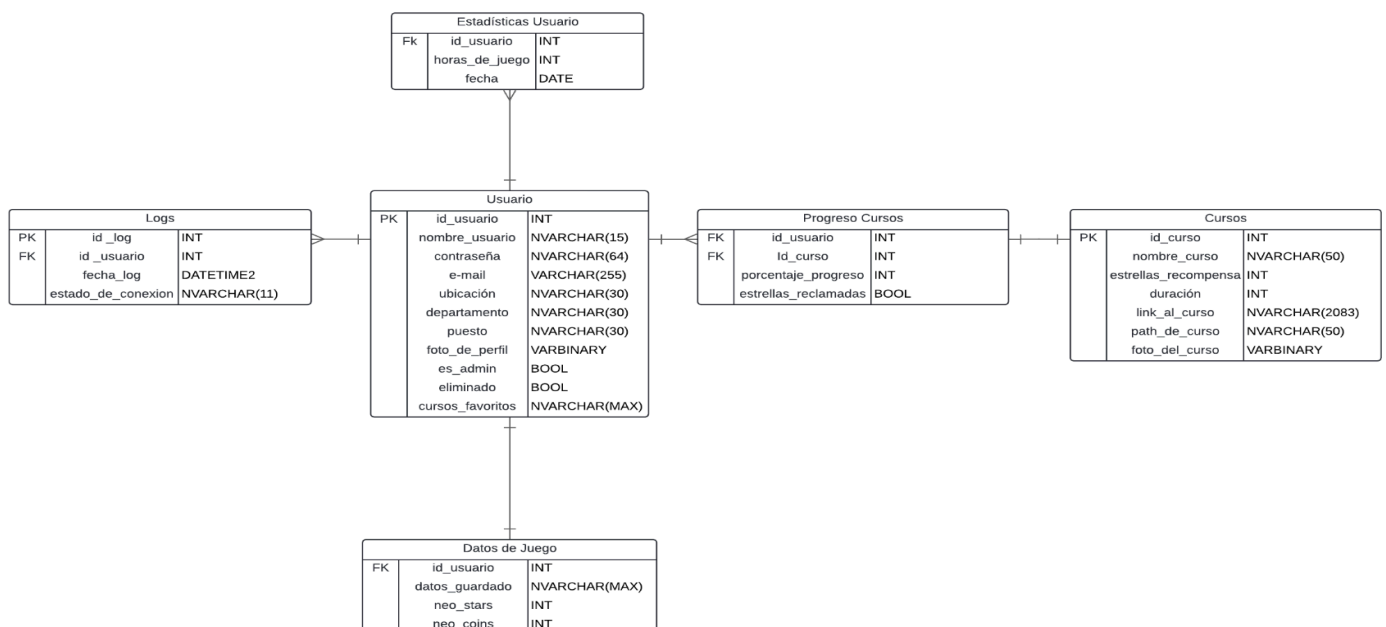
2.4. Limitaciones, supuestos y /o dependencias

- **Limitaciones:** Podría haber restricciones técnicas, como la compatibilidad de navegadores o la capacidad del servidor para manejar múltiples usuarios simultáneamente. Además, las limitaciones presupuestarias podrían afectar la calidad o alcance del proyecto. Hay que considerar también el nivel de habilidad digital de los empleados para interactuar eficientemente con la plataforma y el videojuego.
- **Supuestos:** Se asume que los empleados están motivados para usar la plataforma y participar activamente en los cursos. Otro supuesto es la eficacia del método gamificado para mejorar el aprendizaje y retención de información. Además, se presupone que habrá un acceso constante y sin interrupciones a la plataforma de cursos.
- **Dependencias:** El proyecto depende de la colaboración continua con plataformas de aprendizaje en línea para el contenido de los cursos. El soporte técnico y las actualizaciones continuas del software del juego, desarrollado en Unity 2D, son también cruciales. Además, el éxito del proyecto se vincula estrechamente al monitoreo y análisis de los datos de progreso de los usuarios para ajustar estrategias.

3. Base de datos

1. Esquema de la Base de Datos:

- **Descripción General:** La base de datos de Coder 's Tycoon es un componente crítico diseñado para almacenar y gestionar todos los datos esenciales del sistema. Utilizando Microsoft SQL Server, esta base de datos asegura una alta seguridad, escalabilidad y rendimiento óptimo, adecuados para las necesidades del sistema que soporta tanto la funcionalidad de gamificación como la gestión de cursos educativos.
- **Estructura Organizativa:** La base de datos está organizada en múltiples tablas que están interconectadas para mantener la integridad de los datos y facilitar el acceso eficiente a la información. Cada tabla tiene un propósito claro dentro del sistema, desde almacenar detalles del usuario hasta registrar su progreso en los cursos y su interacción con el juego. Las relaciones entre las tablas se manejan a través de claves primarias y foráneas, asegurando un enlace robusto y coherente.
- **Tablas Principales:** Las principales tablas incluyen: Usuario, Cursos, Progreso _Cursos, Datos_de_Juego y Estadísticas_Usuarios y Logs
- **Diagrama del Esquema:**

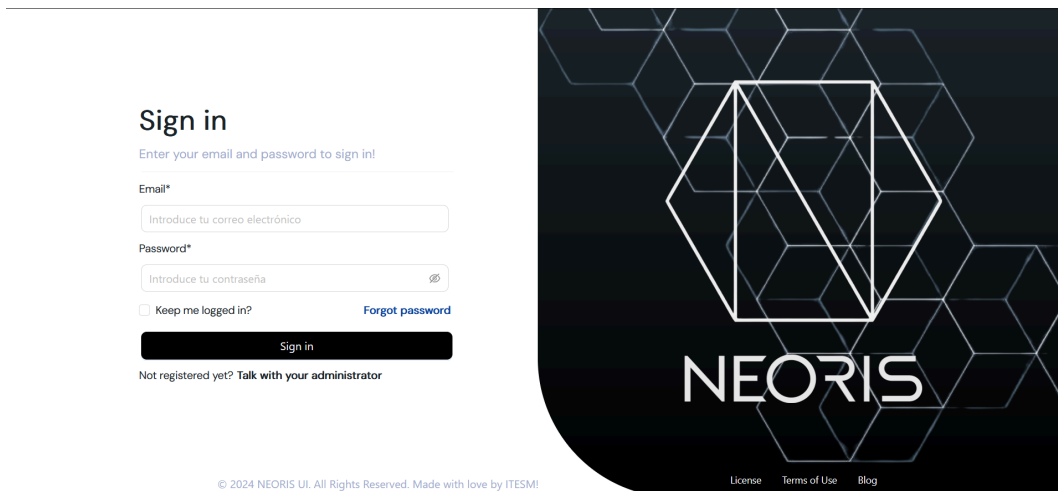


2. Descripción de Tablas:

- **Usuarios:** Esta tabla almacena toda la información esencial de los usuarios del sistema, tanto administradores como usuarios regulares, y es el centro para la autenticación y el manejo de perfiles. Las relaciones que tiene son las siguientes:
 - Estadísticas Usuarios: a través de id_usuario para rastrear estadísticas de juego.
 - Progreso_Cursos: mediante id_usuario para vincular el progreso en cursos
 - Datos_de_Juego: usando id_usuario para guardar datos de juego específicos
 - Logs: vinculada por id_usuario para registrar la actividad de usuario
- **Cursos:** Almacena los detalles de los cursos disponibles para los usuarios, facilitando la gestión de acceso a los recursos educativos. Las relaciones que tiene son las siguientes:
 - Progreso Cursos: a través de id_curso para conectar el curso con el progreso del usuario en dicho curso
- **Progreso Cursos:** Registra el progreso de cada usuario en los cursos específicos, esencial para la funcionalidad de seguimiento y gamificación del aprendizaje. Las relaciones que tiene son las siguientes:
 - Usuarios: mediante id_usuario para identificar al usuario
 - Cursos: a través de id_curso para especificar el curso
- **Datos_de_Juego:** Contiene información relacionada con la interacción del usuario en el componente de juego como las monedas y estrellas obtenidas, crucial para integrar la experiencia del juego con el aprendizaje. Las relaciones que tiene son las siguientes:
 - Usuario: mediante id_usuario para vincular los datos de juego al usuario correspondiente

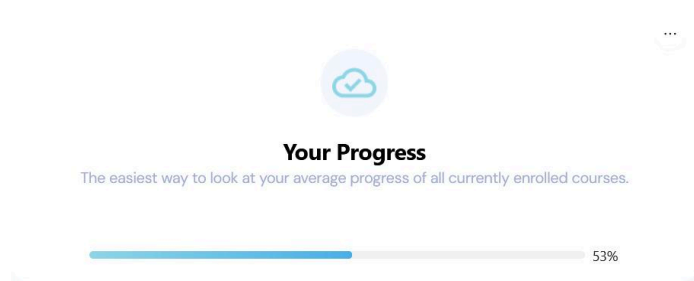
3. Procedimientos Almacenados y Triggers:

WEB



- **verificar_usuario:** Autentica a los usuarios al verificar sus credenciales. Este procedimiento toma como parámetros el 'e-mail' y 'contraseña' del usuario, y comprueba si las credenciales proporcionadas coinciden con las almacenadas en la base de datos. Si son correctas y el usuario no está marcado como eliminado, devuelve los detalles del usuario y un indicador de validación; de lo contrario, devuelve un indicador de validación negativo.

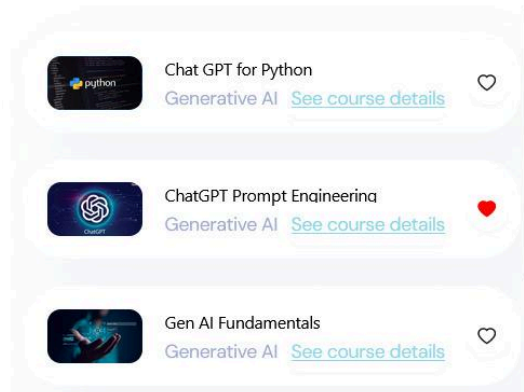
- **promedio_avance:** Calcula el promedio de progreso en todos los cursos para un usuario específico. Acepta el 'id_usuario' como parámetro y calcula el promedio del campo 'porcentaje_progreso' en la tabla 'Progreso_Cursos' para el usuario dado. Este dato es crucial para monitorear el desempeño y la participación de usuario en los cursos



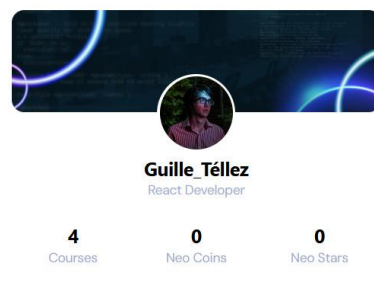
- **listado_cursos_web_card:** Proporciona un listado de cursos juntos con el progreso del usuario para mostrar en tarjetas en la interfaz web. Toma 'id_usuario' y retorna una lista de cursos en los que el usuario está inscrito, incluyendo detalles como el nombre del curso, el progreso y el enlace al curso. Esencial para interfaces de usuario dinámicas donde se muestra información resumida de cursos

Your Courses

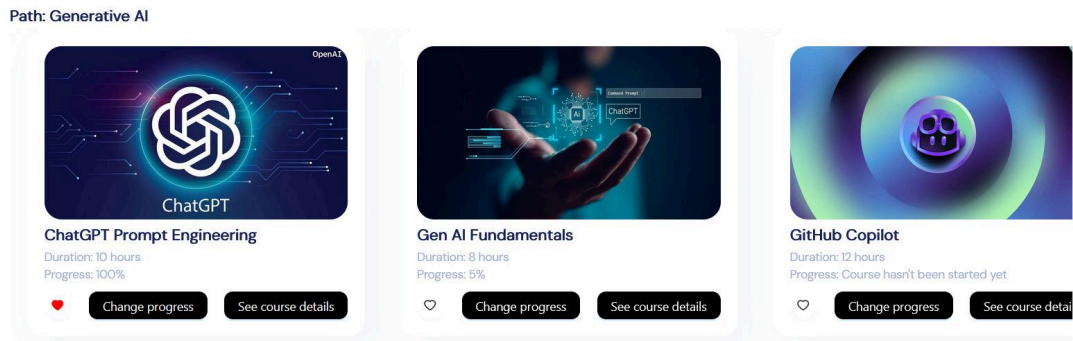
Here you can find more details about your courses. Keep yourself engaged by providing meaningful information.



- **profile_card:** Consigue la información del usuario para mostrarla dentro de su perfil. Esta información incluye las Neo Coins, Neo Stars y la cantidad de cursos inscritos por el usuario. El procedimiento toma el 'id_usuario' y busca dentro de la tabla 'Datos_de_Juego' y 'Progreso_Cursos' el id del usuario para regresar las Neo Coins, Neo Stars y utiliza COUNT() para conseguir la cantidad de cursos inscritos del usuario.



- **cursoes_inscritos_usuario:** Esta función se utiliza para conseguir la toda la información de un curso a base del id del usuario; para mostrarla en la pestaña de cursos. Utiliza 'id_usuario' y busca dentro de la tabla 'Progreso_Cursos' y 'Cursos' por el id del curso, el path, nombre, link, la duración y el porcentaje de progreso y lo muestra dentro de la pestaña de cursos.



- **actualizar_progreso_curso:** Actualiza el progreso de un usuario en un curso específico. Recibe 'id_usuario', 'id_curso', y 'nuevo_porcentaje' para actualizar el campo 'porcentaje_progreso' en la tabla 'Progreso_Cursos'. Fundamental para mantener el registro actualizado del avance del usuario en sus cursos.

Change progress for ChatGPT Prompt Engineering

100

Aceptar

- **crear_progreso_curso:** Inscribe a un usuario en un nuevo curso y establece su progreso inicial. Inserta un nuevo registro en 'Progreso_Cursos' con el 'id_usuario', 'id_curso', y 'porcentaje_progreso' inicial, lo cual es vital para comenzar el seguimiento del avance en un nuevo curso.

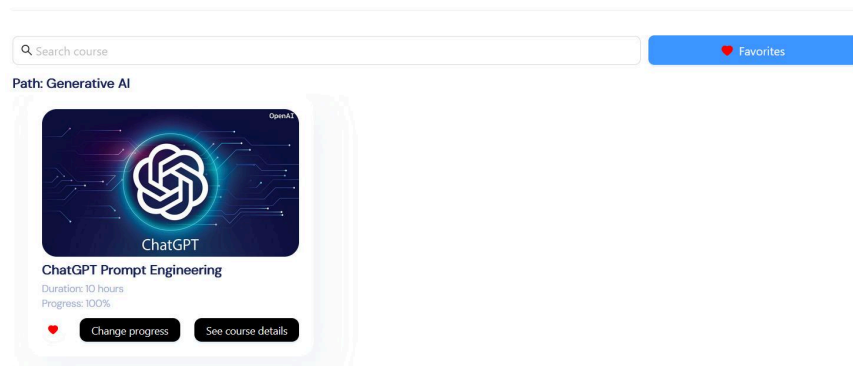
Change progress for GitHub Copilot

Aceptar

- **anadir_curso_favorito y quitar_curso_favorito:** Gestionan la lista de cursos favoritos de un usuario. Modifican el campo 'cursos_favoritos' en la tabla 'Usuarios', agregando o quitando identificadores de cursos. Estos procedimientos son importantes para personalizar y mejorar la experiencia del usuario en la plataforma.



- **favorite_courses:** Muestra un listado de todos los cursos favoritos del usuario dentro de la sección de 'cursos_favoritos' utilizando 'id_usuario' para buscar los id 's de sus cursos marcados como favoritos.



- **creacion_usuario:** Inserta dentro de la tabla de 'Usuarios' la información conseguida desde la web para crear un nuevo id dentro de la base de datos para que el usuario tenga una nueva cuenta. La información que se inserta es: nombre_usuario, e mail, contraseña, ubicación, departamento, puesto, foto_de_perfil, es_admin.

×

Create User

Name

Enter your name

Email

Enter your email

Location

Enter your location

Department

Enter your department

Job Position

Enter your job position

Password

Enter your password

Create Account

- **modificar_usuario:** Consigue información nueva del usuario para actualizar la base de datos y la información de dicho usuario a través de 'id_usuario'. al conseguir la nueva información, utilizamos UPDATE en la tabla usuario y comparamos la información actual con la información antigua, si hay un cambio se actualiza, si no se queda igual.

Editar Usuario

×

Name

Enter your name

Email

Enter your email

Location

Enter your location

Department

Enter your department

Job Position

Enter your job position

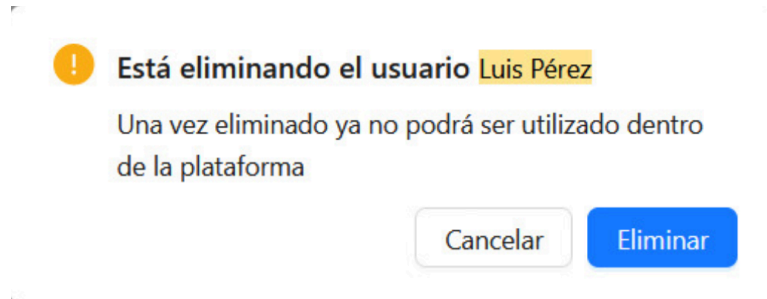
Password

Enter your password

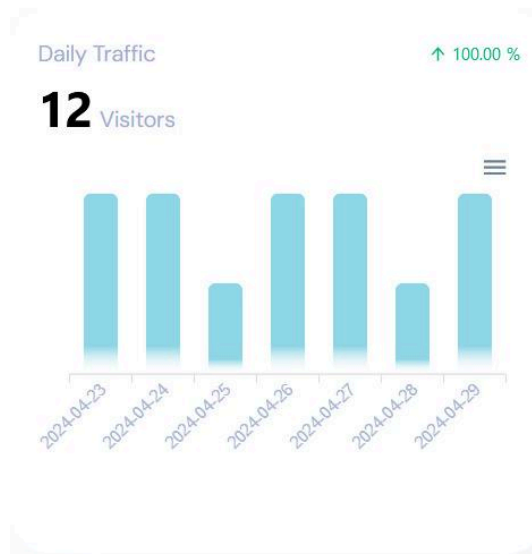
Cancel

OK

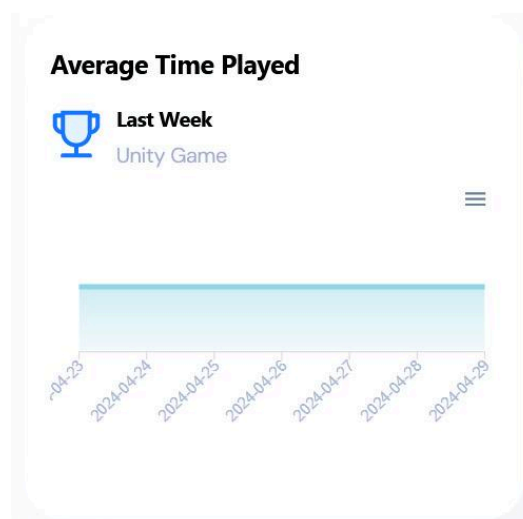
- **eliminar_usuario:** Cambia a un usuario a que este sea “eliminado”. Utilizando ‘id_usuario’ se actualiza el arreglo de eliminado de falso a verdadero. Lo cual convierte a esta información de usuario inaccesible.



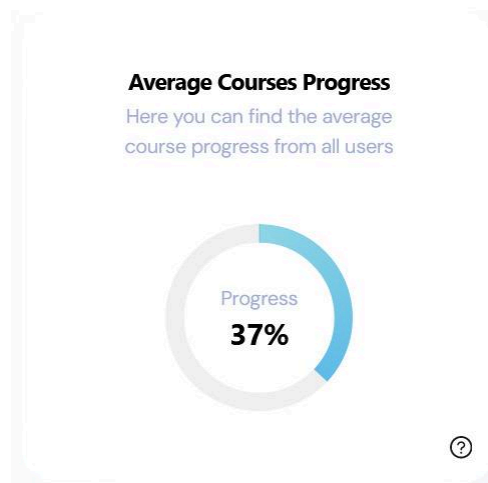
- **conexion_por_fecha:** Proporciona el tráfico de la red de los últimos 7 días. Selecciona el TOP 7 de la tabla ‘Logs’ y consigue el COUNT de usuarios donde el ‘estado_de_conexion’ sea conexión para contarlo como vista al juego.



- **minutos_juego_por_fecha:** Nos devuelve el promedio de los minutos de juego ordenados por fecha. Seleccionamos la fecha y el AVG de los minutos_de_juego de todos los usuarios dentro de la tabla ‘Estadisticas_Usuarios’.



- **promedio_total_progreso:** Nos proporciona el promedio total de todos los cursos que son actualmente cursados por todos los usuarios. De la tabla 'Progreso_Cursos' conseguimos el promedio/AVG del progreso 'porcentaje_progreso' de todos los usuarios.



- **usuarios_no_admins:** Nos devuelve todos los usuarios que no son administradores y no han sido eliminados de la base de datos. Selecciona el nombre, foto de perfil, fecha de conexión de 'Usuarios' y asegura que administrador y eliminado sean falsos.

Your Team

- Guille Téllez**
2 days ago
- Danny_C**
2 days ago
- Nicolas_A**
44 minutes ago
- Deborah_M**
44 minutes ago










→ View All

- **cursos_populares:** Se nos devuelve la información de los cursos más populares. Consiguiendo el nombre del curso, la cantidad de usuarios inscritos, el promedio del progreso y la suma de la cantidad de usuarios que hayan completado el curso se consigue la popularidad de cada curso.

Most Visited Courses			
Course Name	Enrolled Users	Average Progress	Completed
Chat GPT for Python	1	60%	0
ChatGPT Prompt Engineering	1	100%	1
Gen AI Fundamentals	2	10%	0
GitHub Advanced Security	2	35%	0

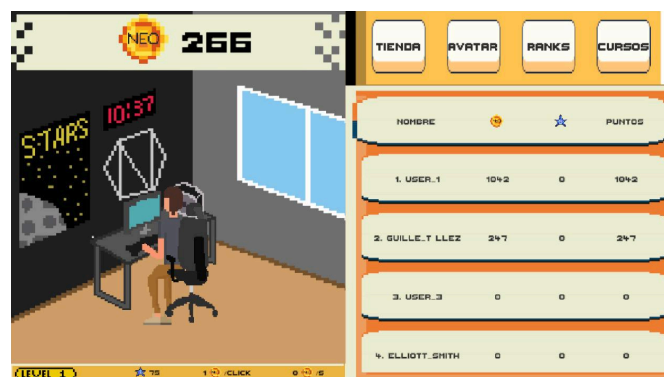
→ View All

- **listar_usuarios:** Nos devuelve un listado de todos los usuarios que no son administradores. Se selecciona toda la información relevante del usuario como id, nombre, e-mail, ubicación, departamento, puesto y foto de perfil de la tabla 'Usuarios' y se muestra en forma de lista.

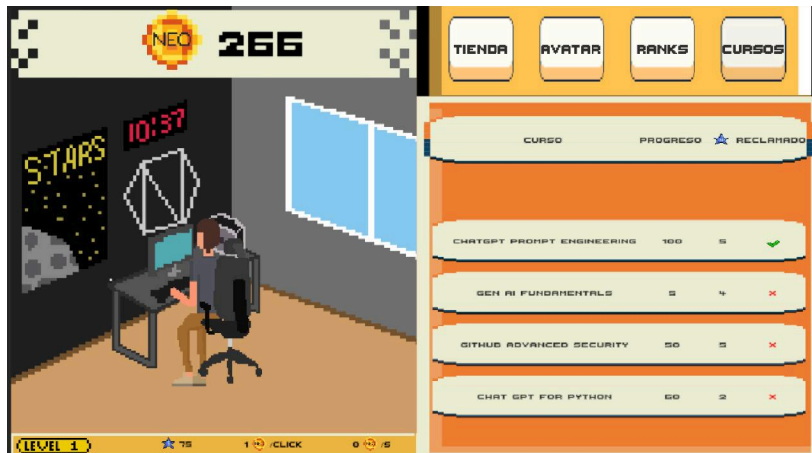
Usuarios			+ Agregar Usuarios	
USUARIO	NOMBRE	ACCIONES		
	Luis Pérez			
	Checo Pérez			
	Ana Pérez			
1 resultados < 1 >				

Unity

- **actualizar_datos_juego:** Actualiza los datos utilizados guardados del juego a la base de datos. Consigue los datos del juego (id_usuario, neo_stars, neo_coins y datos_guardado y se actualizan dentro de la tabla 'Datos_de_Juego'.
- **conseguir_datos_juego:** Regresa los datos guardados de un usuario. Utilizando 'id_usuario' dentro de la tabla de 'Datos_de_Juego' nos regresa un json ('datos_guardados') del jugador para cargar en el juego.
- **datos_leaderboard:** Nos presenta los datos utilizados dentro de la leaderboard. El nombre del usuario, Neo Coins, Neo Stars y los puntos del jugador. Nos regresa el TOP 5 de los nombre_usuario, neo_coins, neo_stars y los puntos utilizando puntos = (neo_coins + neo_stars * 3) y se ordenan por puntuación.



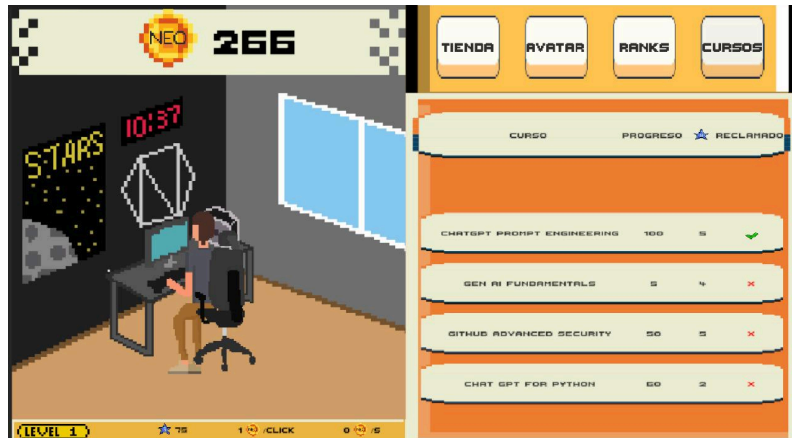
- **listado_cursos_unity:** Se recibe 'id_usuario' y devuelve información sobre los cursos que el usuario está realizando, esta información incluye el nombre del curso, el progreso, y las recompensas obtenidas.



- **log_conexion:** Registra cada sesión de usuario para mantener un historial de actividad. Se ejecuta para insertar un registro en la tabla 'Logs' cada vez que un usuario se conecta o desconecta, lo que es esencial para la seguridad, auditoría y monitoreo de la actividad del usuario en la plataforma.
- **nuevo_usuario:** Crea datos "default" para nuevos usuarios. Al crear un nuevo usuario, si el usuario no es administrador se le hace insertar dentro de 'Datos_de_Juego' donde se guarda la información nueva por el usuarios como ,Neo Coins, Neo Stars, Auto Coins, etc...



- **UpdateEstrellasReclamadas:** Nos permite actualizar la información sobre el usuario y los cursos que ha completado para evitar que se le den estrellas de más al usuario por completar el mismo curso. Utilizando 'id_usuario' y 'CourseName' buscamos el id del curso para conseguir su nombre y se procede a actualizar 'estrellas_reclamadas' a verdadero.



- **actualizar_horas_de_juego:** nos devuelve los minutos que se jugaron durante la sesión de juego. Cuando se registra una desconexión de sesión de juego. Se calcula los minutos entre la conexión y desconexión de un usuario actualizando las estadísticas en la tabla "Estadísticas_Usuarios". Si ya existía un tiempo para el usuario ese día, se actualiza el tiempo de juego acumulado.

4. Requerimientos específicos

4.1. Requerimientos funcionales

REQUERIMIENTOS:

ID Req.	Requerimiento Funcional
RF01001	Autenticación de Usuarios/Administrador: El sistema debe permitir a los usuarios y administradores iniciar sesión utilizando sus cuentas. Debe asegurar la autenticidad de los usuarios y administradores para acceder a sus respectivos dashboards.
RF01002	Dashboard de Administrador: Debe proporcionar a los administradores una vista integral de las estadísticas de los trabajadores, incluyendo usuarios diarios, cursos más populares, promedio de avance en cada curso, estadísticas individuales de progreso y tiempo dedicado en el juego.
RF01003	Gamificación y Recompensas: El sistema debe incluir un videojuego tipo "clicker" que recompense a los usuarios con puntos, items y bonificaciones basadas en su avance en los cursos
RF01004	Gestion de Perfiles de Usuario: El administrador debe poder crear, editar y eliminar perfiles de usuario dentro de la plataforma.
RF01005	Seguimiento del Progreso del Curso: Los usuarios deben poder actualizar el progreso actual en cada curso, incluyendo cursos completados.

RF01006	Búsqueda y Filtro de Cursos: El sistema debe ofrecer una función de búsqueda que permita a los usuarios localizar cursos específicos utilizando palabras clave, y por selección de cursos favoritos.
RF01007	Registro de Actividad del Usuario: El sistema debe mantener un registro detallado de todas las actividades de los usuarios dentro de la plataforma.

ID Req.	Requerimiento No Funcional
RF02001	Rendimiento: La aplicación debe ser capaz de soportar múltiples usuarios simultáneamente sin degradar la experiencia del usuario. Se espera un tiempo de respuesta rápido en las interacciones del usuario con el sistema.
RF02002	Seguridad: Debe asegurar la protección de los datos de usuario mediante el cifrado de contraseñas y la implementación de medidas de seguridad para la autenticación. Además, debe cumplir con las normativas de protección de datos aplicables.
RF02003	Compatibilidad: La aplicación debe ser compatible con los principales navegadores web y versiones recientes de estos.
RF02004	Estabilidad: La plataforma debe ser capaz de escalar para soportar un aumento en el número de usuarios sin afectar el rendimiento
RF02005	Seguridad de la información: Implementación de protocolos de seguridad robustos para proteger la información personal y el progreso de los usuarios en los cursos
RF02006	Accesibilidad: La plataforma debe cumplir con las directrices de accesibilidad web para asegurar que todos los usuarios puedan interactuar con la aplicación sin barreras.
RF02007	Eficiencia de Recursos: El sistema debe optimizar el uso de recursos del servidor, como la memoria y el procesamiento, para manejar de manera eficiente la carga de usuarios y operaciones simultáneas, asegurando así un rendimiento óptimo.
RF02008	Interoperabilidad: La plataforma debe ser capaz de integrarse de forma efectiva con otras herramientas y plataformas empresariales, permitiendo el intercambio fluido de datos y la calibración entre sistemas sin conflictos de compatibilidad.

CASOS DE USO:

CU1010. Buscar un curso	
Actor principal	Usuario
Precondición	El usuario está autenticado y se encuentra en la pantalla principal de la plataforma.
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario accede al campo de búsqueda ubicado en la parte superior del dashboard. 2. El usuario ingresa palabras clave relacionadas con el curso que desea encontrar. 3. El usuario presiona el botón de búsqueda o presiona "Enter". 4. El sistema procesa la solicitud y muestra una lista de cursos que coinciden con los términos de búsqueda. 5. El usuario visualiza los resultados y puede seleccionar un curso para ver más detalles.

Escenarios de excepción	a) No se encuentran cursos que coincidan con los terminos de busqueda <ul style="list-style-type: none"> El sistema no muestra ningún curso que coincida. b) El usuario quiere filtrar los resultados <ul style="list-style-type: none"> El usuario selecciona opciones de filtrado desde un menú desplegable o interfaz similar. c) La conexión a internet es inestable o se pierde durante la búsqueda: <ul style="list-style-type: none"> El sistema detecta el problema y muestra un mensaje de error.
-------------------------	---

CU#####. Ingresar al juego	
Actor principal	Usuario
Precondición	El usuario está autenticado y se encuentra en el dashboard principal de la plataforma.
Escenario Principal	1. El usuario seleccione la opción “Juego” o “Game” desde la barra de navegación o el dashboard principal. 2. El sistema carga la escena del juego que presenta al usuario con una vista de un escritorio y un personaje tecleando la computadora. 3. En el interfaz del juego, el usuario observa el saldo actual de monedas virtuales en la parte superior. 4. El usuario puede hacer clic en diferentes items en la interfaz , como “Tienda”, “Avatar”, “Ranks”, y “Cursos” para navegar por las opciones del juego. 5. Al seleccionar un item del menú, como “Tienda” , el usuario puede usar las monedas virtuales para mejorar su equipamiento en el juego, lo que se refleja en cambios visuales y en el rendimiento del ‘clicker’. 6. El usuario interactúa con el juego haciendo clic en el botón marcado con “Click” para ganar más monedas o mejorar estadísticas dentro del juego. 7. A medida que el usuario progresa, las estadísticas de su personaje y sus capacidades en el juego mejoran, lo que se refleja en la interfaz y la experiencia de juego.
Escenarios de excepción	a) Falta de monedas virtuales para realizar mejoras: <ul style="list-style-type: none"> El usuario puede seguir interactuando con el juego para acumular las monedas necesarias. b) Intento de acceso a una función no implementada <ul style="list-style-type: none"> El usuario debe esperar que dicha funcionalidad esté operativa en futuras actualizaciones de juego. c) Error de carga del juego <ul style="list-style-type: none"> El usuario tiene que recargar la página..

CU#####. Ingresar al Dashboard de Estadísticas	
Actor principal	Administrador
Precondición	El administrador está autenticado y se encuentra en la página principal del panel de administración.
Escenario Principal	1. El administrador hace click en la pestaña “Dashboards” / “Admin” en la barra lateral de navegación.

	<ol style="list-style-type: none"> El sistema presenta el dashboard administrativo con secciones como “Daily Traffic”, “Average Time Played”, “Average Courses Progress” , “Your Team” y “Most Visited Courses”. El administrador visualiza la cantidad total de visitas diarias y el porcentaje de cambio en un gráfico de barras. En la sección “Average Courses Progress” , el administrador revisa el tiempo total promedio dedicado a los cursos por los usuarios. En la sección “Total Course Estimation”, el administrador observa el porcentaje de cursos completados en relación con el total ofrecido. Bajo “Your team” , el administrador puede ver los miembros del equipo y su actividad reciente En “Most Visited Courses”, el administrador analiza los cursos con mayor tráfico y completitud. Si es necesario, el administrador hace clic en “View all” para detalles extendidos en cualquiera de las secciones mencionadas. El administrador puede descargar reportes o estadísticas específicas utilizando los botones o enlaces proporcionados para ese propósito
Escenarios de excepción	<ol style="list-style-type: none"> Los datos no se cargan debido a un error de sistema <ul style="list-style-type: none"> El sistema muestra un mensaje de error. No existen datos registrados <ul style="list-style-type: none"> El sistema presenta un mensaje indicando la ausencia de datos.

CU#####. Iniciar Sesión en la Plataforma	
Actor principal	Usuario/Administrador
Precondición	El usuario o administrador ha llegado la página de inicio de sesión
Escenario Principal	<ol style="list-style-type: none"> El usuario o administrador hace clic en el campo de correo electrónico y escribe su dirección de email. El usuario o administrador hace clic en el campo de contraseña y escribe su contraseña. El usuario o administrador puede seleccionar la opción de “Keep me logged in” si desea permanecer conectado después de cerrar la sesión. El usuario o administrador hace clic en el botón “Sign in” para acceder a su cuenta. El sistema valida las credenciales y , si son correctas, redirige al usuario o administrador a su dashboard respectivo. Si el usuario o administrador elige la opción “Keep me logged in”, el sistema almacena la información de inicio de sesión para futuros accesos.
Escenarios de excepción	<ol style="list-style-type: none"> Credenciales incorrectas <ul style="list-style-type: none"> Si el usuario o administrador ingresa un email o contraseña incorrectos, el sistema muestra un mensaje de error El usuario o administrador tiene la opción de reintentar ingresar sus credenciales Problemas de conectividad

	<ul style="list-style-type: none"> • Si hay un problema de conectividad que impide el inicio de sesión, el sistema muestra un mensaje de error. <p>c) Nueva cuenta</p> <ul style="list-style-type: none"> • Si el usuario aún no tiene una cuenta, se le indica en la ventana de Login que debe ponerse en contacto con su administrador.
--	---

CU#####. Administrar Equipos de trabajo	
Actor principal	Administrador
Precondición	El administrador está autenticado y tiene permisos para gestionar equipos
Escenario Principal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción “Tu equipo” o “Your Team” en el dashboard de administración. 2. El sistema muestra la lista de miembros del equipo con información relevante como nombre, último acceso y progreso reciente. 3. El administrador hace clic en un miembro específico para ver más detalles o realizar acciones como asignar tareas, cambiar roles o actualizar permisos. 4. Para añadir un nuevo miembro, el administrador selecciona la opción “Añadir miembro” y proporciona los detalles necesarios como nombre, email y rol dentro del equipo. 5. Para remover a un miembro, el administrador selecciona la opción “Eliminar” junto al nombre del miembro y confirma la acción. 6. El sistema actualiza la lista de equipo y muestra el estado actualizado al administrador
Escenarios de excepción	<ol style="list-style-type: none"> a) Error al añadir un nuevo miembro <ul style="list-style-type: none"> • El administrador puede corregir la información o intentar la acción nuevamente b) Intento de eliminar un miembro con tareas pendientes <ul style="list-style-type: none"> • El administrador debe gestionar la reasignación de tareas antes de proceder con la eliminación. c) Fallo en la actualización de roles o permisos <ul style="list-style-type: none"> • El administrador puede intentar modificar los permisos nuevamente. d) Acciones Restringidas por permisos <ul style="list-style-type: none"> • El administrador debe solicitar los permisos adecuados a un superior o revisar las políticas de gestión de equipos.

4.2. Requerimientos de interfaces externas

01. Conexión con Plataformas de Aprendizaje	
ID Req.	Requerimiento de interfaz externa
RI01001	Sincronización de datos de progreso con base de datos internas

RI01002	Integración con API para acceso a cursos y datos de progreso
RI01003	Integración con SQL Server Express 2019 en Google Cloud para la gestión de bases de datos y almacenamiento de datos de usuario y progreso
RI01004	Interfaz de usuario construida con React y Ant Design para una interacción eficiente y atractiva en el frontend
RI01005	Uso de Node.js y Express.js para la creación de servicios web que interactúan con la base de datos y las APIs externas

02. Interacción con Servicios de Gamificación	
ID Req.	Requerimiento de interfaz externa
RI02001	Integración con sistemas de recompensas para el intercambio de puntos y logros
RI02002	Conexión con servidores de juego para actualización de estados en tiempo real
RI02003	API para la recopilación de estadísticas de uso y desempeño de juegos

03. Integración de Juegos y Dashboards	
ID Req.	Requerimiento de interfaz externa
RI03001	Integración con motor de juego Unity para la sincronización de datos de juego y usuario
RI03002	Conexión con dashboard analítico para la visualización del progreso de usuarios en tiempo real
RI03003	Interfaz de usuario para intercambio de puntos del juego por recompensas o mejoras de cursos

4.3. Requerimientos de datos lógicos

01. Datos de Usuarios y Progreso	
ID Req.	Requerimiento de datos lógicos
RD01001	Almacenar detalles de perfil de usuario, incluyendo avance en cursos y logros
RD01002	Registro de interacciones dentro del juego y cursos completados
RD01003	Seguimiento de tiempos de acceso y actividades de aprendizaje

02. Datos de Administración y Monitoreo	
ID Req.	Requerimiento de datos lógicos
RD02001	Datos de uso diario por usuario para informes administrativos
RD02002	Estadísticas detalladas del progreso por curso y usuario
RD02003	Registro de problemas técnicos y errores en el uso de la plataforma

03. Datos de Gamificación y Desempeño	
ID Req.	Requerimiento de datos lógicos
RD32001	Almacenar y gestionar las recompensas obtenidas y los items desbloqueados por el usuario en el juego
RD03002	Registro de puntuaciones, rankings, y logros obtenidos por los usuarios para mostrar en el leaderboard
RD03003	Datos históricos del desempeño del usuario en cursos y su correlación con el engagement en el juego

4.4. Restricciones de diseño

01. Diseño y Usabilidad	
ID Req.	Restricción de diseño
RR01001	Interfaz debe ser compatible con navegadores principales
RR01002	Diseño responsivo que se adapte a diferentes tamaños de pantalla
RR01003	Carga eficiente de contenidos para minimizar el tiempo de respuesta del sistema

02. Seguridad y Confiabilidad	
ID Req.	Restricción de diseño
RR02001	Implementar cifrado de datos en transmisión y en reposo
RR02002	Sistema debe ser capaz de manejar carga de usuarios concurrentes sin degradar rendimiento
RR02003	El sistema debe permitir a los usuarios guardar su progreso
RR02004	El sistema debe ser diseñado considerando la arquitectura de microservicios con Node.js y Express.js para facilitar la escalabilidad y el mantenimiento

03. Diseño Interactivo y Visual	
ID Req.	Restricción de diseño
RR03001	La interfaz del juego debe ser intuitiva y atractiva para mantener la motivación del usuario
RR03002	Los dashboards deben presentar información de forma clara y concisa para facilitar la toma de decisiones administrativas
RR03003	La aplicación debe ser coherente en términos visuales y funcionales a través de diferentes plataformas y dispositivos
RR03004	Los elementos de UI deben ser desarrollados utilizando Ant Design para garantizar consistencia visual en todos los componentes del frontend

4.5. Requerimientos no funcionales

01. Rendimiento y Escalabilidad	
ID Req.	Requerimiento no funcional
RN01001	La aplicación debe mantener una alta disponibilidad y un tiempo de respuesta bajo durante picos de uso
RN01002	Escalabilidad para administrar aumento gradual de usuarios sin necesidad de intervención manual
RN01003	El sistema debe ser diseñado para operar óptimamente en el entorno de Google Cloud, aprovechando sus capacidades de auto-escalado y balanceo de carga

02. Seguridad y Acceso	
ID Req.	Requerimiento no funcional
RN02001	Todos los datos personales deben ser protegidos según las normativas de protección de datos aplicables
RN02002	Accesibilidad en la plataforma para usuarios con discapacidades, cumpliendo con las directrices de accesibilidad web
RN02003	El sistema debe ser diseñado para operar óptimamente en el entorno de Google Cloud, aprovechando sus capacidades de auto-escalado y balanceo de carga

03. Experiencia de Usuario y Usabilidad	
ID Req.	Requerimiento no funcional
RN03001	El juego y los dashboards deben ser de fácil acceso y manejo para usuarios con diferentes niveles de habilidad tecnológica
RN03002	Accesibilidad en la plataforma para usuarios con discapacidades, cumpliendo con las directrices de accesibilidad web
RN03003	Se debe proporcionar documentación y guías de uso para los desarrolladores y administradores del sistema, asegurando una correcta utilización de las tecnologías involucradas.

5. Glosario

Término	Explicación
1. Autenticación	Proceso de verificar identidad de un usuario antes de permitirle acceso a la plataforma
2. Dashboard	Panel de control que muestra resúmenes visuales de datos y estadísticas para administradores
3. Gamificación	Uso de elementos de diseño de juegos en entornos no lúdicos para mejorar la participación del usuario
4. Clicker	Tipo de juego donde los usuarios hacen clic para ganar puntos o avanzar en el juego
5. Leaderboard	Tabla que clasifica a los usuarios según su rendimiento o puntos en el juego
6. Unity 2D	Motor de juego utilizado para desarrollar juegos en dos dimensiones
7. API	Interfaz de programación de aplicaciones que permite la comunicación entre software
8. Criptogranja	En el contexto del juego, se refiere a una mecánica de recompensas basada en criptomonedas
9. GPU	Unidad de procesamiento gráfico, en el contexto del juego puede referirse a una mejora o recompensa
10. GPT	Mejora dentro del juego basada en ChatGPT
11. Paths profesionales	Secuencias de cursos diseñados para mejorar habilidades específicas para el desarrollo profesional
13. Back - end	Parte del sistema que procesa las acciones del usuario sin que estas sean visibles
14. Front - end	Interfaz del usuario donde interactúan con el sistema de manera directa
15. Interoperabilidad	Capacidad del sistema para trabajar con otros productos o sistemas sin restricciones especiales
16. Navegador web de escritorio	Programa de software para recuperar y presentar información en la web, usado en computadoras en escritorio
17. Puntos de restauración	Puntos específicos guardados que representan el estado de un sistema o progreso en un momento dado