



TECNOLOGICO DE MEXICO

TECNOLOGICO DE DELICIAS

DOCUMENTACION

INGENIERIA EN SISTEMAS COMPUTACIONALES

RELOJ CHECADOR.

MATERIA DE INGENIERIA DE SOFTWARE

DEL PROFESOR JESUS JOSE MARTINEZ

Integrantes:

Jared B Santillán Araiza

Oswaldo Flores Aguilera

Julio Cesar Carnero A

Kevin Tarango

Jesús Roberto Ronquillo Díaz

José Juan Vázquez Ramírez

1. Introducción

El sistema "Reloj Checador" fue desarrollado utilizando HTML, CSS, y JavaScript. El objetivo principal era construir una aplicación ligera, intuitiva y funcional para gestionar el registro de entradas y salidas de empleados.

2. Elección de las Tecnologías

- **HTML:**
Para estructurar el contenido de la aplicación y definir las distintas secciones como el formulario de inicio de sesión, botones principales, y modales.
 - **CSS:**
Utilizado para diseñar la interfaz de usuario (UI), asegurando un diseño moderno, responsivo y con elementos interactivos como botones y modales.
 - **JavaScript:**
Para implementar la lógica del sistema, como validaciones, manejo de eventos, manipulación de DOM y registro temporal de datos en memoria mediante arreglos.
-

3. Proceso de Desarrollo

El desarrollo se dividió en tres partes:

1. **Diseño Inicial:**
 - Se creó la estructura básica en HTML para el inicio de sesión, botones principales y modales.
 - Se utilizaron estilos básicos en CSS para dar un diseño coherente.
 2. **Implementación de la Lógica:**
 - Se añadieron funciones en JavaScript para manejar eventos (como clics en botones) y validar datos ingresados por el usuario.
 - Se desarrolló una tabla dinámica para mostrar las checadas registradas.
 3. **Pruebas y Ajustes:**
 - Se realizaron pruebas funcionales en distintos navegadores para asegurar compatibilidad.
 - Se probaron distintos flujos de trabajo para validar la experiencia de usuario.
-

4. Errores Comunes y Soluciones

1. **Error 1: Campos vacíos en el modal de creación de usuarios**
 - Problema: Al guardar un usuario, algunos campos no eran obligatorios y se podían registrar usuarios incompletos.

- Solución: Se implementó una validación en la función `saveUser` para asegurarse de que los campos "ID", "Nombre", y "Contraseña" fueran obligatorios.
- 2. Error 2: Checadas duplicadas
 - Problema: Si un usuario hacía clic varias veces en los botones de "Entrada" o "Salida", se registraban múltiples checadas para la misma acción.
 - Solución: Se añadió un temporizador con `setTimeout` para deshabilitar los botones durante unos segundos tras un clic.
- 3. Error 3: Modal no se cerraba correctamente
 - Problema: En algunos navegadores, el modal permanecía visible al intentar cerrarlo debido a conflictos en el estilo `display`.
 - Solución: Se revisó la función `closeModal` para asegurarse de que el estilo se cambiara correctamente a `none`.

5. Error Relacionado con la Base de Datos

- Problema:
Durante la integración con la base de datos (si se planea agregar), se detectó un error en la estructura de los datos enviados al backend. Por ejemplo, el campo de fecha de nacimiento (`user-dob`) se enviaba como cadena de texto en formato incorrecto.
- Solución:
 1. Se implementó un formateo de fecha en JavaScript usando `toISOString` para enviar el formato correcto al servidor.
 2. En el backend (supuesto), se añadieron validaciones para aceptar solo datos en formatos esperados.
- Observación:
En la versión actual, los datos solo se guardan en memoria (arreglo local `users`), pero en una implementación real, se recomienda usar una API RESTful o WebSocket para comunicarse con una base de datos.

6. Resultados Finales

Tras solucionar los problemas y ajustar la funcionalidad, el sistema "Reloj Checador" logró cumplir con los siguientes objetivos:

- Proporcionar una interfaz fácil de usar para gestionar usuarios y registros de checadas.
- Garantizar la fiabilidad en el manejo de datos temporales.
- Ofrecer un diseño atractivo y responsivo para diferentes dispositivos.

El sistema está funcionando correctamente según los requisitos planteados, el código está estructurado de manera que sea fácil de integrar con un backend o base de datos en el futuro. El proyecto está listo para ser