

Sistema de Reservas del Aula AtecA

Descripción del Proyecto

El objetivo de este proyecto es desarrollar un sistema digital de reservas para el Aula AtecA en el IES Comercio. Este sistema reemplazará el proceso manual actual de creación de horarios en Excel y gestión de reservas a través de correo electrónico. El sistema consta de tres aplicaciones:

1. **Aplicación Web (Angular):** Permite a los profesores reservar el Aula AtecA a través de un calendario interactivo.
2. **Aplicación de Escritorio (WPF):** Gestiona las reservas, permite que un administrador apruebe o rechace solicitudes y modifique la disponibilidad del aula.
3. **API .NET:** Un API Rest que gestiona toda la información con BBDD.

Objetivos

- **Reducir la carga administrativa** mediante la automatización del proceso de reservas.
- **Garantizar precisión** en la planificación de reservas.
- **Implementar autenticación** utilizando Google Authentication para los profesores y IdentityUser para el administrador de WPF.
- **Permitir que los administradores gestionen reservas** a través de una aplicación de escritorio.
- **Evitar reservas en días no lectivos** y sobre reservas existentes.
- **Asegurar el acceso a la API**, permitiendo solo a usuarios autenticados interactuar con el sistema.

Aplicación Web (Angular)

Características

- Los profesores pueden iniciar sesión **únicamente con Google Authentication** utilizando su correo @iescomercio.com, no valen todos los correo, solo el de @iescomercio.com.
- Ver un calendario interactivo con los horarios disponibles: reservas pedidas, reservar hechas, reservar hechas por otros profesores. Para las franjas de reservar se tendrán que respetar los tiempos lectivos, que serán modificables desde WPF.
- Reservar un aula seleccionando una fecha, hora y el grupo con el que quieres ir.
- Evitar reservas duplicadas.
- Deshabilitar la reserva de un día no lectivo.
- Las solicitudes de reserva requieren aprobación antes de confirmarse.
- **Restringir el acceso a la API:** La API backend rechazará cualquier solicitud de usuarios no autenticados con Google.

Aplicación de Escritorio (WPF)

Características

- Los administradores pueden iniciar sesión **usando autenticación IdentityUser**.
- Ver reservas pendientes.
- Aprobar o rechazar solicitudes de reserva.
- Modificar los horarios disponibles para reservas, las franjas de horarios, respetando los 5 minutos entre clase y clase y recreos.
- Marcar días específicos como no lectivos.
- **Restringir el acceso a la API:** Solo los administradores autenticados con IdentityUser pueden modificar reservas.

API

Características

- Gestionará toda la información de BBDD utilizando EF y Code First.
- Todas las peticiones serán autenticadas, las de reservar serán específicas para los usuarios de [Google Authentication](#) y todas las peticiones de WPF serán con IdentityUser, para el usuario de WPF cread un usuario por defecto en el [seed](#).
- Los usuarios de IdentityUser **no** se podrán registrar, solo hacer Login.

Requisitos Funcionales

Roles de Usuario

- **Profesor**
 - Iniciar sesión mediante [Google Authentication](#).
 - Ver horarios disponibles.
 - Enviar solicitudes de reserva.
- **Administrador**
 - Iniciar sesión mediante IdentityUser.
 - Aprobar o rechazar solicitudes de reserva.
 - Modificar horarios disponibles para reservas.
 - Especificar días no lectivos.

Seguridad y Acceso a la API

- La API backend **debe exigir autenticación** para todos los endpoints.
- Los profesores solo pueden reservar aulas **si están autenticados mediante Google Authentication**.
- Los administradores solo pueden modificar reservas **si están autenticados mediante IdentityUser**.
- Las solicitudes a la API realizadas por usuarios no autenticados deben ser **rechazadas con una respuesta 401 Unauthorized**.

Proceso de Reserva

1. Un profesor selecciona un horario disponible en la aplicación Angular.
2. La solicitud de reserva se envía a la API .NET y se almacena como "Pendiente".
3. Un administrador revisa la solicitud en la aplicación WPF. El profesor administrador es notificado.. La mejor solución será enviar un correo electrónico al profesor utilizando un servicio como SendGrid o [SMTP](#).
4. Si se aprueba, la reserva se confirma y se muestra en el calendario.
5. Si se rechaza, el profesor es notificado. La mejor solución será enviar un correo electrónico al profesor utilizando un servicio como SendGrid o [SMTP](#).

Entregables

Dado que todos haréis el mismo proyecto, si se detectan partes del código como copiadas, se pondrá un 0 como resultado de este proyecto. Podeís copiar de los materiales dados en clase, pero no entre vosotros.

- Aplicación web Angular completamente funcional.
- Aplicación de escritorio WPF con gestión de reservas.
- API .NET 9 con autenticación IdentityUser y Google Authentication.
- Documentación del código y guía de despliegue.
- Esquema de base de datos para la gestión de reservas.
- Presentación del proyector.