**🧾 Requisitos del Proyecto de Registro de Visitantes**

**🎯 Objetivo General**

Desarrollar una aplicación web para registrar, gestionar y controlar el acceso de visitantes a una sede empresarial. Debe incluir autenticación, roles de usuario, control de accesos, registro de visitas con foto y lectura de documento, entre otros.

**📌 Módulos Funcionales**

**1. Autenticación de Usuarios**

**Requisitos:**

* Formulario de login.
* Verificación de credenciales con bcrypt.
* Generación de token JWT.
* Acceso diferenciado según rol (admin o estándar).

**Desarrollo:**

* **Página de Login** (/login)
  + Formulario: correo + contraseña.
  + Al enviar: llamada a POST /api/login.
  + Redirección según rol:
    - admin → Panel de administración.
    - estándar → Panel de autogestión o visita.

**2. Gestión de Usuarios (solo Admin)**

**Requisitos:**

* Listar usuarios pendientes de aprobación (activo = false).
* Aprobar usuarios.
* Crear nuevos usuarios (registro).

**Desarrollo:**

* **Página de Aprobación de Usuarios** (/admin/usuarios)
  + Tabla de usuarios inactivos.
  + Botón “Aprobar”.
  + Uso de token con rol admin.

**3. Registro de Visitantes**

**Requisitos:**

* Captura de nombre, cédula, empresa, motivo.
* Captura de foto desde cámara o archivo.
* Lectura de documento (código de barras o QR).
* Selección de tipo de registro (manual o autogestionado).
* Estado de la visita (activa, finalizada, rechazada).
* Asociación con sede, empleado y usuario que registra.

**Desarrollo:**

* **Página Registro Manual** (/registro)
  + Formulario completo.
  + Campo para captura de foto (input tipo file o cámara con JS).
  + Selector de tipo registro.
  + Enviar a POST /api/visitas.
* **Página de Autogestión (visitante)** (/autoregistro)
  + Igual al formulario anterior, simplificado.
  + Envío sin intervención de usuario estándar.
  + Estado inicia como activa.

**4. Gestión de Visitas**

**Requisitos:**

* Listar visitas activas, finalizadas y rechazadas.
* Filtrar por fecha, estado, visitante o empleado.
* Finalizar visita (agrega fecha\_salida).
* Rechazar visita.

**Desarrollo:**

* **Página de Control de Visitas** (/admin/visitas)
  + Tabla con visitas.
  + Filtros por estado y fechas.
  + Botones para finalizar/rechazar.

**5. Base de Datos**

**Tablas principales:**

* usuarios (admin/estándar)
* visitas
* empleados
* sedes

Se relacionan por claves foráneas. Se valida existencia y coherencia de las relaciones.

**✅ Funcionalidades Complementarias**

**Seguridad**

* Token JWT en todas las rutas protegidas.
* Middleware para verificar token y rol.
* Hash seguro con bcrypt para contraseñas.

**Validaciones**

* Lado backend: campos obligatorios, formatos, longitudes.
* Lado frontend (posterior): validaciones visuales antes de enviar.

**🛠️ Pendientes por Implementar**

* Frontend completo (React):
  + Login
  + Registro visitantes (manual/autogestión)
  + Panel admin (usuarios + visitas)
* Captura de foto en navegador.
* Lectura de documento por cámara o lector QR.
* Estilos y UX amigable.
* Exportar registros en CSV o PDF (opcional).