Resumen de Requisitos Implementados:

1. Gestión de Trabajadores

Descripción: Sistema para administrar la información de los empleados.

Tecnologías:

1. Django (Python): Framework web para el backend.

2. Django ORM: Para la gestión de la base de datos.

3. JavaScript: Para interacciones dinámicas en el frontend.

4. Tailwind CSS: Para el diseño y estilizado de la interfaz.

2. Gestión de Criterios de Evaluación

Descripción: Permite crear, editar y eliminar criterios de evaluación para los empleados.

Tecnologías:

1. Django: Para la lógica del backend y manejo de formularios.

2. JavaScript: Para operaciones AJAX en la interfaz de usuario.

3. Tailwind CSS: Para el diseño de la interfaz.

3. Gestión de Evaluaciones

Descripción: Sistema para crear, editar y eliminar evaluaciones de empleados.

Tecnologías:

1. Django: Para la lógica del backend, incluyendo transacciones para garantizar la integridad de los datos.

2. JavaScript: Para la interactividad en el frontend, especialmente en la matriz de evaluación.

3. Tailwind CSS: Para el diseño responsivo de la interfaz.

4. API REST

Descripción: Proporciona endpoints para acceder y manipular datos de empleados, departamentos, roles y evaluaciones.

Tecnologías:

1. Django Rest Framework: Para crear la API REST.

2. Serializers: Para la serialización/deserialización de datos.

5. Autenticación y Autorización

Descripción: Sistema de login y logout, con protección de rutas para usuarios autenticados.

Tecnologías:

1. Django Authentication System: Para manejar la autenticación de usuarios.

2. Decoradores de Django: Para proteger vistas y asegurar que solo usuarios autenticados puedan acceder.

6. Interfaz de Usuario Responsiva

Descripción: Diseño adaptable que funciona en diferentes tamaños de pantalla.

Tecnologías:

1. Tailwind CSS: Framework de CSS utilizado para crear un diseño responsivo y estilizado.

2. HTML5: Para la estructura de las páginas.

7. Manejo de Sesiones

Descripción: Gestión de sesiones de usuario para mantener el estado de autenticación.

Tecnologías:

1. Django Sessions: Para manejar las sesiones de usuario.

8. Validación de Datos

Descripción: Validación tanto en el cliente como en el servidor para asegurar la integridad de los datos.

Tecnologías:

1. Django Forms: Para la validación en el servidor.

2. JavaScript: Para validaciones en el lado del cliente.

9. Manejo de Errores y Feedback al Usuario

Descripción: Sistema para mostrar mensajes de éxito o error al usuario después de realizar operaciones.

Tecnologías:

1. Django Messages Framework: Para mensajes del servidor.

2. JavaScript: Para mostrar mensajes dinámicos en el cliente.

10. Base de Datos Relacional

Descripción: Almacenamiento estructurado de datos con relaciones entre entidades.

Tecnologías:

1. SQLite (desarrollo) / PostgreSQL (producción recomendada): Sistemas de gestión de bases de datos.

2. Django ORM: Para interactuar con la base de datos de manera abstracta.

Tecnologías Utilizadas (Descripción detallada):

1. Django (Python):

Framework web de alto nivel que fomenta el desarrollo rápido y el diseño limpio y pragmático. Utilizado para la lógica del servidor, manejo de rutas, vistas, y modelos de datos.

2. Django Rest Framework:

Potente y flexible toolkit para construir APIs Web. Utilizado para crear la API REST del sistema.

3. JavaScript:

Lenguaje de programación del lado del cliente, utilizado para crear interacciones dinámicas en la interfaz de usuario, validaciones en tiempo real, y peticiones AJAX.

4. Tailwind CSS:

Framework de CSS de utilidad-first para crear diseños personalizados rápidamente. Utilizado para estilizar la interfaz de usuario de manera eficiente y responsiva.

5. HTML5:

Lenguaje de marcado utilizado para estructurar y presentar el contenido en la web.

6. SQLite / PostgreSQL:

Sistemas de gestión de bases de datos relacionales. SQLite se usa típicamente en desarrollo por su ligereza, mientras que PostgreSQL es recomendado para producción debido a su robustez y capacidad de manejo de concurrencia.

7. AJAX (Asynchronous JavaScript and XML):

Técnica de desarrollo web para crear aplicaciones interactivas. Utilizado para realizar peticiones asíncronas al servidor sin recargar la página completa.

Este resumen proporciona una visión general de los requisitos implementados y las tecnologías utilizadas en el sistema de gestión de empleados y evaluaciones. Cada componente trabaja en conjunto para crear una aplicación web robusta, segura y fácil de usar.

