

Reporte del Proyecto

HTML

- **Archivo:** admin_dashboard.html
Propósito: Este código HTML es una página web que representa un panel de administrador para la plataforma Nora AI. La página incluye un encabezado con el nombre de la plataforma y un enlace para cerrar sesión, una barra de navegación lateral con opciones como ver Noras, agregar Noras y ver errores, y una sección principal que muestra estadísticas como el número total de Noras activas, errores registrados y la fecha del último deployment. Además, se enlaza un
Rutas: Ninguna
Archivos CSS: {{ url_for('static', filename='css/admin_dashboard.css') }}
Archivos HTML: Ninguno
Tablas y columnas:
Archivos .py usados: Ninguno
- **Archivo:** admin_nora.html
Propósito: Este código HTML es una plantilla de un panel de control para la interfaz de usuario de Nora AI. En la sección ``, se establece la codificación de caracteres, el título de la página y se vinculan estilos CSS y fuentes de Google Fonts. En el cuerpo del documento, se muestra un encabezado con un título dinámico que incluye el nombre visible de la configuración o un nombre predeterminado. A continuación, hay un formulario
Rutas: Ninguna
Archivos CSS: {{ url_for('static', filename='css/panel_noras.css') }}
Archivos HTML: Ninguno
Tablas y columnas:
Archivos .py usados: Ninguno
- **Archivo:** admin_nora_entrenar.html
Propósito: Este código HTML representa una página web que permite entrenar a un sistema de inteligencia artificial llamado "Nora". La página incluye un encabezado con el nombre de la aplicación "Nora AI" y un enlace para cerrar sesión. Además, cuenta con un menú de navegación lateral que permite acceder a diferentes secciones relacionadas con la administración de Nora, como ver Noras existentes, editar Noras y entrenar a Nora. En la sección
Rutas: Ninguna
Archivos CSS: {{ url_for('static', filename='css/admin_dashboard.css') }}
Archivos HTML: Ninguno
Tablas y columnas:
Archivos .py usados: Ninguno
- **Archivo:** admin_nora_nueva.html
Propósito: Este código HTML es una página web que permite al usuario crear una nueva entidad llamada "Nora AI". La página incluye un formulario con campos para ingresar el nombre interno y el nombre visible de la entidad, así como una lista de módulos iniciales que se pueden seleccionar a través de casillas de verificación. Además, la página muestra mensajes de estado que pueden ser mostrados al usuario después de realizar alguna acción. Al final del formulario, hay un botón
Rutas: Ninguna
Archivos CSS: {{ url_for('static', filename='css/panel_admin.css') }}
Archivos HTML: Ninguno
Tablas y columnas:
Archivos .py usados: Ninguno
- **Archivo:** debug_verificacion.html
Propósito: Este código HTML y JavaScript tiene como propósito mostrar una tabla de verificación del sistema en la página web. La tabla incluye información sobre diferentes módulos, sus versiones detectadas y su estado (representado

por emojis como 🟡, 🟠 o advertencia). Si hay un objeto `resultado` disponible, se recorre para mostrar cada módulo con su información en la tabla. De lo contrario, se muestra un mensaje de "Cargando datos...". Ad

Rutas: Ninguna

Archivos CSS: {{ url_for('static', filename='styles/debug_verificacion.css

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** entrena_nora.html

Propósito: Este código HTML representa una página web para entrenar a un asistente virtual llamado Nora. La página incluye un encabezado con el nombre y un enlace para cerrar sesión, una barra de navegación lateral con enlaces a diferentes secciones de administración, y dos secciones principales: una para crear un mensaje de bienvenida personalizado y otra para configurar respuestas automáticas. En la sección de "Entrena tu Nora", los usuarios pueden configurar

Rutas: Ninguna

Archivos CSS: {{ url_for('static', filename='css/entrena_nora.css

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** index.html

Propósito: Este código HTML es la estructura de una página web que muestra un panel de chat de Aura. En la parte superior se define el título de la página y se enlaza un archivo CSS para estilizar el panel de chat. Dentro del cuerpo de la página, se muestra un contenedor con dos secciones: la barra lateral (sidebar) que muestra una lista de contactos, y el área principal del chat. Los contactos se generan dinámicamente con

Rutas: Ninguna

Archivos CSS: /static/aura/panel_chat.css

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** panel_chat.html

Propósito: Este código HTML tiene como propósito principal mostrar una interfaz de chat de Nora. En la parte del encabezado se definen metadatos como el juego de caracteres, la vista inicial en dispositivos móviles, y el título de la página que incluye el nombre de Nora. Luego se incluyen enlaces a hojas de estilo CSS para personalizar el diseño de la página. Dentro de la sección de script, se define un array de contactos que

Rutas: Ninguna

Archivos CSS: {{ url_for('static', filename='css/layout.css, {{ url_for('static', filename='css/sidebar.css, {{ url_for('static', filename='css/chat.css, {{ url_for('static', filename='css/contact_info.css

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** panel_cliente_contactos.html

Propósito: Este código HTML define la estructura de una página web que muestra un panel de contactos. En el encabezado se establece el título de la página y se importan estilos CSS, incluyendo fuentes de Google Fonts. Luego, se define un conjunto de estilos CSS para elementos específicos de la página, como la disposición de elementos en una cuadrícula, el estilo de las secciones y los filtros de búsqueda. En el cuerpo de la página

Rutas: Ninguna

Archivos CSS: {{ url_for('static', filename='css/panel_noras.css

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** panel_cliente_envios.html

Propósito: Este código HTML es una plantilla para una página web que muestra una interfaz de usuario para programar

y visualizar envíos programados. La página incluye un encabezado con el nombre de la aplicación y un enlace para cerrar sesión, así como una sección principal que muestra una lista de envíos programados, permite programar nuevos envíos a través de un formulario y ofrece un enlace para volver al panel del cliente. Se utiliza la sintaxis de

Rutas: Ninguna

Archivos CSS: `{{ url_for('static', filename='css/admin_dashboard.css`

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** `panel_cliente_ia.html`

Propósito: Este código HTML es una plantilla para una página web que muestra un panel de control de una Inteligencia Artificial llamada Nora. La página incluye un encabezado con el nombre de Nora y un botón para cerrar sesión. En el cuerpo de la página se muestra el estado actual de la IA, con la posibilidad de activarla o desactivarla a través de un botón. Además, hay un enlace para volver al panel del cliente. La página utiliza

Rutas: Ninguna

Archivos CSS: `{{ url_for('static', filename='css/admin_dashboard.css`

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** `panel_cliente_respuestas.html`

Propósito: Este código HTML y Jinja (un lenguaje de plantillas para Python) se utiliza para crear una interfaz de usuario que permite administrar respuestas automáticas. La página muestra un título con el nombre de usuario, un área para mostrar mensajes flash, un formulario para agregar o editar respuestas automáticas, y una lista de respuestas existentes. El script JavaScript incluido define una función ``editarRespuesta`` que se utiliza para prellenar el formulario de edición de

Rutas: Ninguna

Archivos CSS: `{{ url_for('static', filename='css/panel_admin.css`

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

Otros

- **Archivo:** `admin_noras.html`

Propósito: El código proporciona la estructura de una página web que muestra un administrador de Noras (posiblemente un sistema de inteligencia artificial). La página incluye una lista de Noras activas, cada una representada por una tarjeta con información como el nombre, estado de la inteligencia artificial, fecha de última actualización, módulos asociados y tickets pendientes. Además, se incluyen enlaces para acceder al panel de chat y al dashboard de cada Nora

Rutas: Ninguna

Archivos CSS: `{{ url_for('static', filename='css/panel_noras.css`

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

Etiquetas

- **Archivo:** `admin_nora_dashboard.html`

Propósito: Este código HTML es una plantilla de un panel de control de un sistema de inteligencia artificial llamado Nora AI. La página muestra diferentes secciones con información relevante: 1. ****Estado de IA****: Muestra si la inteligencia artificial está activada o desactivada. 2. ****Módulos activos****: Lista los módulos activos en el sistema. 3. ****Contactos****: Muestra el total de contactos, la cantidad sin IA y sin etiquetas,

Rutas: Ninguna

Archivos CSS: {{ url_for('static', filename='css/admin_dashboard.css

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** admin_nora_editar.html

Propósito: Este código HTML junto con algunas etiquetas de Jinja2 está diseñado para mostrar una página web donde se puede editar la información de una entidad llamada "Nora". La página incluye un formulario con campos para ingresar un nuevo nombre visible para Nora y seleccionar los módulos que se desean activar. También muestra mensajes de estado que pueden aparecer como notificaciones en la parte superior. Además, se incluye un botón para guardar los cambios realizados y otro para

Rutas: Ninguna

Archivos CSS: {{ url_for('static', filename='css/panel_admin.css

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** panel_cliente.html

Propósito: Este código HTML es una plantilla para un panel de cliente personalizado. La página muestra un saludo al usuario con su nombre, el nombre de la sección que está gestionando y una lista de módulos activos. Además, se muestran diferentes secciones de contenido dependiendo de los módulos activos, como la gestión de contactos, etiquetas y envíos programados. Cada sección incluye un título, una descripción y un bot

Rutas: Ninguna

Archivos CSS: {{ url_for('static', filename='css/panel_noras.css

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** panel_cliente_etiquetas.html

Propósito: Este código HTML es una plantilla para una página web que permite a un usuario gestionar etiquetas para sus contactos. La página incluye un encabezado con el título "Etiquetas para {{ nombre_nora }}" donde {{ nombre_nora }} es una variable que se espera ser rellena dinámicamente. Luego, hay un formulario donde se puede agregar una nueva etiqueta introduciendo un nombre en un campo de texto y pulsando un botón "Agregar".

Rutas: Ninguna

Archivos CSS: {{ url_for('static', filename='css/panel_cliente.css

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno

- **Archivo:** panel_envios_programados.html

Propósito: El código proporciona una interfaz web para programar envíos de mensajes a contactos etiquetados. Permite ingresar un mensaje, seleccionar una fecha y hora, elegir etiquetas de contactos a los que se enviará el mensaje y programar el envío. También muestra una lista de contactos agrupados por etiquetas, y un historial de envíos programados con opciones para filtrar por estado (todos, pendientes o cancelados). Además,

Rutas: Ninguna

Archivos CSS: {{ url_for('static', filename='css/panel_envios_programados.css

Archivos HTML: Ninguno

Tablas y columnas:

Archivos .py usados: Ninguno