

- `visualization_service` module
 - Imports
 - Funciones
 - `def get_model_folders() -> list`
 - `def get_models(folder: str) -> list`
 - `def update_prediction_service() -> None`
 - `def fetch_data() -> dict`
 - `def update_data(new_data: dict) -> None`

visualization_service module

Servicio de Visualización en Tiempo Real con Streamlit

Este script implementa una aplicación de visualización en tiempo real utilizando Streamlit. Se conecta a servicios de predicción y datos para mostrar resultados en vivo.

Imports

- **streamlit** : Framework para aplicaciones web interactivas.
- **pandas** : Librería para manipulación de datos.
- **requests** : Librería para realizar solicitudes HTTP.
- **os** : Librería para interactuar con el sistema operativo.
- **pygame** : Librería para reproducción de sonido.
- **datetime** : Librería para manejar fechas y horas.
- **matplotlib.pyplot** : Librería para crear gráficas.
- **streamlit_autorefresh** : Plugin de Streamlit para auto refrescar la página.

Funciones

`def get_model_folders() -> list`

Obtiene las carpetas de los modelos.

Returns : : `list`: Lista de nombres de carpetas de modelos.

def get_models(folder: str) -> list

Obtiene los nombres de los modelos dentro de una carpeta.

Args : : folder (str): Nombre de la carpeta que contiene los modelos.

Returns : : list: Lista de nombres de modelos (sin extensión .pkl).

def update_prediction_service() -> None

Actualiza el modelo en el servicio de predicción.

Envía una solicitud POST al servicio de predicción para actualizar el modelo.

def fetch_data() -> dict

Obtiene datos del servicio de datos.

Returns : : dict: Diccionario con los datos obtenidos del servicio de datos.

def update_data(new_data: dict) -> None

Actualiza el estado de los datos en la aplicación.

Args : : new_data (dict): Diccionario con los nuevos datos obtenidos.