Como crear una nueva AI

Posicionarse el la carpeta raiz de la aplicacio e ir a la carpeta ai

En esta carpeta, crear una nueva carpeta llevando por nombre el nombre de la AI

Dentro de esta carpeta se deben de crear la carpete « info » y el fichero « run.py »

Carpeta info :

Debe contener la descripcion de la ai en un fichiero de texto llamado « description.en »

Debe contener los requerimientos de la ai en in fichiero llamado « requirements.json » (ver requirements structure)

Requirements structure :

El fichiero requirements.json debe contener un objeto json estructurado de la siguiente manera :

{

"images":  {},

"medical\_information": {},

"skin\_lesion\_characteristics": {}

}

El objeto « imagen » va a contener sub objetos con los tipos de imaganes requeridas por la ai conteniendo el valor minimo y maximo de imagenes ej :

"images":  {

            "id" : {

                            "min": 1,

                            "max":5

                            },

            }

*El id del objeto va a ser utilizado a la hora de pasar la informacion a la ai*

*Ejemplo de un objeto image :*

    "dermoscopy" : {

                "min": 1,

                "max":2

                },

Los objetos « medical\_information » y «skin\_lesion\_characteristics » van a contener sub objetos con la informacion necesaria del paciente y de la lesion de la piel, cada uno de estos objetos debe ser estructurado de la siguiente manera

"id" : {

                "name":"",

                "type":"",

"items": [] # optional

                "scale":"" # optional

            }

}

*El id de cada sub objeto va a ser utilizado a la hora de pasarle la informacion a la ai*

El contenido del nombre debe ser un string informativo para el paciente

El contenido del typo debe ser uno de los siguiente valores :

« options » : crea una lista de opciones

« int » : crea un especio numerico

« float » : crea un especio numerico con decimales

« text » : crea un espacio de texto

« bool » : crea la opcion yes/no

« date » : crea un espacio de fechas

El contenido de items debe ser una lista de elementos y debe ser colocado unicamente si se elije el tipo options :

[« element 1 », « element 2 », ….]

El contenido de scale debe ser colocado solo si se elige la opcion int o float y debe ser uno de los siguientes valores

« length »

« time »

« volume »

« weight »

Ejemplos de objetos :

"hair\_color" : {

                "name":"Hair color",

                "type":"options",

                "items":["Black","Brown","Blonde","Grey","White"]

                },

"weight" : {

                "name":"Weight",

                "type":"float",

                "scale":"weight"

            }

run.py structure

el alchivo run.py va a contener el metodo run que va a ser utilizado como puente de comunicacion entre la aplicaion grafica y la ai

el metodo run va a recibir 2 parametros:

def run(images, info):

parametro « images » :

el primero va a contener un diccionario con todas las imagenes de la lesion, para accéder a la lista de imagenes de un tipo de imagen se utiliza el su id, ejemplo :

images["dermoscopy"]

esto va devolver una lista de string con la ubicacion de todas las imagenes

parametro « info » :

va a contener toda la informacion necesaria del paciente y de la lesion :

para accéder a la informacion basica del paciente (edad, sexo) utilice el codigo siguiente :

info["basic\_information"]

para accéder a la informacion medical del paciente utilice el codigo siguiente :

info["medical\_information"]

para accéder a las caracteristicas de la lesion de la piel utilice el codigo siguiente :

info["skin\_lesion\_characteristics"]

return :

luego de haber hecho todo el tratamiento el resultado debe ser devuelto en forma de diccionario “clave”: valor, el parametron “risk” en el diccionario es aconsejado para indicar el tipo de riesgo de la lesion y solo los valores « malignant », « benign » y « indeterminate » son aceptados.