

Práctica 2

- Contratos de microservicios

Microservicio	# de contrato	EndPoint	Descripción
Genderize.io	1	Retorno de probabilidad	Retorna la probabilidad en que el nombre le pertenezca a un hombre o a una mujer.
Agify.io	2	Retorno de edad	Retorna la probable edad de la persona basado en su nombre.
Unión de microservicios	3	Unión de respuestas	Une las dos respuestas obtenidas por los microservicios en una única respuesta.

ID: 001	NOMBRE: Retorno de probabilidad	
PRIORIDAD: Alta	HISTORIA DE USUARIO: El usuario administrador quiere utilizar Genderize.io para obtener la probabilidad en que el nombre de una persona le pertenezca a un hombre o una mujer.	
ESTIMADO: 4 puntos		
MÓDULO: Generar respuestas		
CRITERIOS DE ACEPTACIÓN: <ul style="list-style-type: none">• El sistema debe utilizar el microservicio Genderize.io.• El usuario debe ingresar un nombre válido.		
RUTA: /Genderize		
MÉTODO: GET		
FORMATO DE ENTRADA: JSON		
HEADER:		
ATRIBUTO	TIPO	DESCRIPCIÓN
Content_type	header	application/json

BODY:		
ATRIBUTO	TIPO	DESCRIPCIÓN
nombre	cadena	Nombre del que se desea saber la probabilidad.
FORMATO DE SALIDA: JSON CÓDIGO DE RESPUESTA EXITOSA: HTTP 200 SALIDA:		
ATRIBUTO	TIPO	DESCRIPCIÓN
mensaje	cadena	Mensaje que se mostrará como alerta “Se ha ingresado correctamente a Genderize.io”
CÓDIGO DE RESPUESTA FALLIDA:		
CÓDIGO	DESCRIPCIÓN	
400	El nombre ingresado no es válido.	
PARÁMETROS DE ENTRADA	Body: <pre>{ nombre: "<<nombre>>" }</pre>	
PARÁMETROS DE SALIDA EXITOSA	Header: <pre>{ status: 200 }</pre> Body: <pre>{ "count": 1439272, "name": "juan", "gender": "male", "probability": 1 }</pre>	
PARÁMETROS DE SALIDA FALLIDA	Header: <pre>{ status: 400 }</pre> Body: <pre>{ mensaje: "Error" }</pre>	
ID: 002	NOMBRE: Retorno de edad	
PRIORIDAD: Alta	HISTORIA DE USUARIO:	

ESTIMADO: 4 puntos	El usuario administrador quiere utilizar Agify.io para obtener la edad de la persona basado en su nombre.
MÓDULO: Generar respuestas	

CRITERIOS DE ACEPTACIÓN:

- El sistema debe utilizar el microservicio Agify.io.
- El usuario debe ingresar un nombre válido.

RUTA: /Agify
MÉTODO: GET
FORMATO DE ENTRADA: JSON
HEADER:

ATRIBUTO	TIPO	DESCRIPCIÓN
Content_type	header	application/json

BODY:

ATRIBUTO	TIPO	DESCRIPCIÓN
nombre	cadena	Nombre del que se desea saber la probabilidad.

FORMATO DE SALIDA: JSON
CÓDIGO DE RESPUESTA EXITOSA: HTTP 200
SALIDA:

ATRIBUTO	TIPO	DESCRIPCIÓN
mensaje	cadena	Mensaje que se mostrará como alerta “Se ha ingresado correctamente a Agify.io”

CÓDIGO DE RESPUESTA FALLIDA:

CÓDIGO	DESCRIPCIÓN
400	El nombre ingresado no es válido.

PARÁMETROS DE ENTRADA	Body: <div><pre>{ name: "<<nombre>>" }</pre></div>
PARÁMETROS DE SALIDA EXITOSA	Header: <div><pre>{ status: 200 }</pre></div>

	Body: <pre>{ "count": 275260, "name": "juan", "age": 61 }</pre>
PARÁMETROS DE SALIDA FALLIDA	Header: <pre>{ status: 400 }</pre> Body: <pre>{ mensaje: "Error" }</pre>

ID: 003	NOMBRE: Unión de respuestas
PRIORIDAD: Alta	HISTORIA DE USUARIO: El usuario administrador quiere unir las respuestas obtenidas por los microservicios.
ESTIMADO: 4 puntos	
MÓDULO: Unir respuestas	

CRITERIOS DE ACEPTACIÓN:

- El usuario debe obtener correctamente la probabilidad de ser hombre o mujer según el nombre de Genderize.io.
- El usuario debe obtener correctamente la edad de la persona según el nombre de Agify.io.

RUTA: /Union

MÉTODO: GET

FORMATO DE ENTRADA: JSON

HEADER:

ATRIBUTO	TIPO	DESCRIPCIÓN
Content_type	header	application/json

BODY:

ATRIBUTO	TIPO	DESCRIPCIÓN
nombre	cadena	Nombre del que se desea saber la probabilidad.

FORMATO DE SALIDA: JSON
CÓDIGO DE RESPUESTA EXITOSA: HTTP 200
SALIDA:

ATRIBUTO	TIPO	DESCRIPCIÓN
mensaje	cadena	Mensaje que mostrará el nombre, la probabilidad obtenida y la edad obtenida.

CÓDIGO DE RESPUESTA FALLIDA:

CÓDIGO	DESCRIPCIÓN
400	El nombre ingresado no es válido.

PARÁMETROS DE ENTRADA

Body:
{
 nombre: "<<nombre>>"
}

PARÁMETROS DE SALIDA EXITOSA

Header:
{
 status: 200
}
Body:
{
 "name": "juan",
 "genderize": {
 "count": 1439272,
 "name": "juan",
 "gender": "male",
 "probability": 1
 },
 "agify": {
 "count": 275260,
 "name": "juan",
 "age": 61
 }
}

PARÁMETROS DE SALIDA FALLIDA

Header:
{
 status: 400
}

	<pre>} Body: { mensaje: "Error" }</pre>
--	---

Definir kubernetes y sus componentes

Definición: Kubernetes es una plataforma de código abierto que automatiza la implementación, el escalado y la gestión de aplicaciones en contenedores. Permite administrar un gran número de contenedores y servicios de manera eficiente.

Componentes para usar Kubernetes:

- Pods: representación de una sola instancia de contenedor. Pueden contener uno o varios contenedores relacionados que comparten recursos y espacio de red.
- Deployments: definen cómo se deben crear y actualizar los pods.
- Services: exponer aplicaciones en ejecución a otros componentes del sistema.
- Ingress: gestiona el acceso externo a los servicios en un clúster. Permite la configuración de reglas de enrutamiento basadas en HTTP y TLS, lo que facilita el acceso a las aplicaciones desde fuera del clúster.

Definir docker y sus componentes

Definición: Docker es una plataforma de código abierto que permite a los desarrolladores empaquetar, distribuir y ejecutar aplicaciones dentro de contenedores.

Componentes para usar Docker:

- Contenedores: instancias en ejecución de imágenes Docker. Cada contenedor es una entidad que incluye todo lo necesario para ejecutar una aplicación de forma aislada.
- Imágenes: plantillas de solo lectura que contienen el sistema operativo, las bibliotecas, las dependencias y el código de una aplicación.
- Dockerfile: archivo de texto plano que contiene las instrucciones para construir una imagen Docker. Define el entorno y las dependencias necesarias para ejecutar una aplicación dentro de un contenedor.
- Docker Engine: administra los contenedores, las imágenes, las redes y los volúmenes. Proporciona una API para interactuar con el sistema Docker.