

Pablo Robles Mansilla. Memoria

Tecnologías

Las tecnologías utilizadas en el examen (lenguajes, bibliotecas, frameworks, base de datos, etc.). Para la base de datos vamos a utilizar MongoDB Atlas. Para construir la API usaré el framework Flask de Python. Usaré pymongo para conectarnos a la base de datos. Como ejemplos de pruebas ejecutables se ha realizado un conjunto de pruebas en Postman.

Entidades

1. Colección tareas

Esta colección representa las tareas que se pueden gestionar en la aplicación. Cada tarea tiene un responsable, una descripción breve, un conjunto de habilidades requeridas, y una estimación de tiempo.

Estructura:

```
{
  "_id": "ObjectId",
  "responsable": "correo@ejemplo.com",
  "descripcion": "Descripción de la tarea",
  "habilidades": ["habilidad1", "habilidad2"],
  "segmentos": 2,
  "colaboradores": ["ObjectId", "ObjectId"]
}
```

Atributos:

- **_id**: Identificador único de la tarea. Es de tipo ObjectId.
- **responsable**: Dirección de correo electrónico del usuario que creó la tarea. Es de tipo string y debe cumplir con el formato de email.
- **descripcion**: Título o breve descripción de la tarea. Es de tipo string y tiene un máximo de 50 caracteres.
- **habilidades**: Lista de habilidades necesarias para realizar la tarea. Es de tipo array y cada elemento es un string que describe una habilidad.
- **segmentos**: Tiempo estimado para completar la tarea en segmentos de una hora. Es de tipo int y su valor mínimo es 1.
- **colaboradores**: Lista de identificadores (ObjectId) de los colaboradores asignados a esta tarea. Es opcional y permite establecer relaciones con documentos en la colección colaboradores.

2. Colección colaboradores

Esta colección representa a los usuarios que pueden colaborar en las tareas. Cada colaborador tiene habilidades que lo califican para ciertas tareas.

Estructura:

```
{
  "_id": "ObjectId",
  "email": "correo@ejemplo.com",
  "nombre": "Nombre del colaborador",
  "habilidades": ["habilidad1", "habilidad2"]
}
```

Atributos:

- **_id**: Identificador único del colaborador. Es de tipo ObjectId.
- **email**: Dirección de correo electrónico del colaborador. Es de tipo string y debe cumplir con el formato de email.
- **nombre**: Nombre completo del colaborador. Es de tipo string.
- **habilidades**: Lista de habilidades que posee el colaborador. Es de tipo array y cada elemento es un string que describe una habilidad.

Instrucciones de instalación.

Para instalar y probar este examen, será necesario lanzar la imagen en Docker, usando el comando "docker compose up --build". Para conectarse a la base de datos, la URI se encuentra en el archivo .env.

Esta es: "mongodb+srv://admin:admin@cluster0.auditv.mongodb.net/"

El puerto de despliegue del servidor es el 5000, también hallado en el archivo .env.