

MEMORIA BACK-END Tecnologías de Desarrollo para la Web (TDW)

Se tratará de llevar un orden cronológico respecto de los errores encontrados pero no hay una certeza del 100%.

1. Relaciones (métodos GET y PUT) para Entidades y Personas

- Dudas sobre la estructura de las rutas y controladores:
 - Al principio, surgieron dudas sobre cómo estructurar correctamente los endpoints para añadir y eliminar relaciones entre entidades y personas.
 - Errores en la sincronización bidireccional
 - Finalmente y tras revisar los controladores del elemento productos, así como toda su lógica estructural, fue sencillo llevar a cabo métodos como `operationPerson()`, `getPersons()` etc... Finalmente se trataba simplemente de mantener la misma estructura pero haciendo referencia a otros elementos según el método.

2. Implementación de las operaciones para la gestión de Asociaciones (incluyendo pruebas)

- Relaciones ManyToMany y sincronización (Asociaciones y resto de entidades)
 - Al implementar las relaciones ManyToMany (por ejemplo, entre entidades y asociaciones), aparecieron problemas de sincronización bidireccional: al eliminar una relación en un lado, no siempre se reflejaba en el otro.
 - Se revisaron y ajustaron los métodos `addAssociation`, `removeAssociation`, `addEntity`, `removeEntity` para que siempre actualizaran ambas colecciones. Se detectó que en algunos casos se añadía dos veces la misma asociación, lo que causaba que la eliminación no funcionara correctamente. Se corrigió añadiendo comprobaciones para evitar duplicados.
- Aparecieron errores como `Class not found` en los tests, especialmente con controladores de relaciones.

- Algunos archivos no tenían la extensión .php, lo que impedía que Composer los cargara correctamente.
- Se renombraron los archivos afectados para que tuvieran la extensión .php y se ejecutó composer dump-autoload para regenerar el autoload.
- Los tests fallaban porque, tras eliminar una relación, la clave correspondiente (associations, persons, etc.) no aparecía en la respuesta JSON si la colección estaba vacía.
 - Se modificó el método jsonSerialize() de las entidades para que siempre incluyera las claves de las relaciones, aunque fueran arrays vacíos, cumpliendo así con la estructura esperada por los tests.
- Se detectaron errores por llamadas a métodos inexistentes (por ejemplo, getAssociationsTag() en vez de getEntitiesTag()).
 - Se unificaron los nombres de los métodos y se corrigieron las llamadas en los tests y controladores para mantener la coherencia.
- Al cambiar el generador de nombres de Faker (company() por name()), algunos tests dejaron de encontrar coincidencias en las búsquedas parciales.
 - Se ajustaron los tests para buscar por el nombre completo generado, asegurando que los datos de prueba coincidieran con los datos insertados.

3. Completar la especificación OpenAPI para la gestión de Asociaciones, Entidades y Personas.

- Falta de endpoints para Entidades, Personas y Asociaciones.
 - El fichero openapi.yaml original solo documentaba de forma completa los endpoints para usuarios y productos, pero no incluía rutas ni operaciones para los recursos de entidades, personas y asociaciones, ni sus relaciones.
 - Se han definido y documentado los endpoints REST necesarios para la gestión CRUD de entidades (/entities), personas (/persons) y asociaciones (/associations), siguiendo la misma estructura que los recursos ya existentes.
- Completar parámetros y esquemas

- Faltaban parámetros como associationId en la sección de componentes, y los esquemas de los recursos no reflejaban correctamente las relaciones entre ellos, especialmente para asociaciones.
- Se han ampliado los esquemas de Entity, Person y Association para reflejar correctamente las relaciones y propiedades de cada uno, incluyendo arrays de identificadores relacionados.

Además, durante todo el proceso se ha utilizado un sistema de control de versiones (GitHub). Se adjunta aquí el enlace al repositorio del mismo, donde podrán verse los distintos commits realizados así como las modificaciones respectivas a los mismos:

<https://github.com/RafaelAguilar64/TDWBack-end>