**Задание 42.6.1 (HW-04) Исправления**

**Ментор afonick@mail.ru  
Замечания:**

1. UUID используется для уникальной идентификации записей в таблицах базы данных. Другими словами UUID нужно использовать вместо, а не вместе с автоинкрементым ID.
2. Не реализован эндпойнт получения код-сниппета по UUID.
3. Дублирование - присутствует два разных эндпойнта для добавления (регистрации) пользователя.
4. Эндпойнт удаления код-сниппета не работает, т.к. для операции db.delete(snippet) в асинхронной функции не был применен await.

**✅ План исправления ошибок, найденных супервайзером**

Я внимательно проанализировал код проекта и выявил решения для каждого замечания супервайзера.  
Ниже представлен **подробный план исправления ошибок**.

**🔹 Ошибка 1: UUID используется вместе с автоинкрементным ID**

**📌 Проблема**

* В модели Snippet (models.py) **используются и id (автоинкремент), и uuid**.
* **Нужно оставить только uuid**, а id удалить.

**✅ План исправления**

1. **Удалить поле id из модели Snippet в models.py**.
2. **Указать uuid как primary\_key** (он будет основным ключом).
3. **Создать новую миграцию (alembic revision) и применить её**.
4. **Обновить все эндпоинты, использующие id → заменить на uuid**:
   * GET /snippets/{id}
   * PUT /snippets/{id}
   * DELETE /snippets/{id}
5. **Обновить схемы SnippetResponse, SnippetCreate, убрав id**.

**🔹 Ошибка 2: Не реализован эндпойнт получения код-сниппета по UUID**

**📌 Проблема**

* В API **нет эндпоинта**, который позволяет **получить сниппет по uuid**.
* Пользователь должен **по uuid получать сниппет**, даже без авторизации.

**✅ План исправления**

1. **Добавить эндпоинт GET /snippets/uuid/{uuid} в routes.py**.
2. **Этот эндпоинт должен работать без авторизации (get\_current\_user не нужен)**.
3. **Эндпоинт должен возвращать 404, если сниппет не найден**.
4. **Обновить Swagger UI, чтобы эндпоинт был задокументирован**.

**🔹 Ошибка 3: Дублирование эндпоинтов регистрации пользователей**

**📌 Проблема**

* В routes.py и auth\_routes.py есть **два разных эндпоинта регистрации пользователей**.
* **Нужно оставить только один эндпоинт** в auth\_routes.py.

**✅ План исправления**

1. **Удалить дублирующий эндпоинт регистрации из routes.py** (POST /users/).
2. **Оставить эндпоинт регистрации только в auth\_routes.py (POST /auth/register)**.
3. **Проверить Swagger UI, чтобы в /users/ не было регистрации**.

**🔹 Ошибка 4: Эндпоинт удаления код-сниппета (DELETE /snippets/{id}) не работает**

**📌 Проблема**

* В routes.py используется **db.delete(snippet), но нет await**.
* **Без await асинхронный код не удаляет запись**.

**✅ План исправления**

1. **Исправить DELETE /snippets/{id} → добавить await db.delete(snippet)**.
2. **Проверить, что удаление работает (DELETE /snippets/{uuid}).**
3. **Обновить тесты, чтобы убедиться, что сниппет реально удаляется.**

**🎯 Итоговый список исправлений**

✅ **Удалить id из models.py, оставить только uuid.**  
✅ **Создать эндпоинт GET /snippets/uuid/{uuid}.**  
✅ **Удалить дублирующий эндпоинт регистрации.**  
✅ **Добавить await в db.delete(snippet) для корректного удаления.**

**1. Ошибка: UUID используется вместе с автоинкрементным ID**

| **Проблема** | **Исправление** | **Статус** |
| --- | --- | --- |
| В models.py использовались **оба id и uuid**, что неверно. | **Удалили id, теперь uuid является primary\_key**. | ✅ Исправлено |
| В schemas.py и routes.py использовался id. | **Обновили все эндпоинты, теперь uuid используется везде**. | ✅ Исправлено |
| База данных содержала id. | **Пересоздали БД, применили миграции (alembic upgrade head)**. | ✅ Исправлено |

✅ **Теперь вся система работает на UUID.**

**📌 2. Ошибка: Нет эндпоинта получения сниппета по UUID**

| **Проблема** | **Исправление** | **Статус** |
| --- | --- | --- |
| Не было эндпоинта GET /snippets/uuid/{uuid}. | **Добавили в routes.py новый эндпоинт**. | ✅ Исправлено |
| API не позволял получать сниппеты без авторизации. | **Теперь GET /snippets/uuid/{uuid} доступен без авторизации**. | ✅ Исправлено |
| Не было тестов для получения сниппета по UUID. | **Протестировано в Swagger UI — работает!** | ✅ Исправлено |

✅ **Теперь любой пользователь может получить сниппет по UUID без авторизации.**

**📌 3. Ошибка: Дублирующийся эндпоинт регистрации пользователей**

| **Проблема** | **Исправление** | **Статус** |
| --- | --- | --- |
| Было два разных эндпоинта регистрации (POST /users/ и POST /auth/register). | **Удалили POST /users/, оставили POST /auth/register**. | ✅ Исправлено |
| routes.py содержал дубликат регистрации. | **Очистили routes.py, регистрация теперь только в auth\_routes.py**. | ✅ Исправлено |
| Нужно было протестировать регистрацию. | **Тестирование прошло успешно, новый пользователь создаётся**. | ✅ Исправлено |

✅ **Теперь POST /auth/register — единственный эндпоинт для регистрации.**

**📌 4. Ошибка: DELETE /snippets/{id} без await**

| **Проблема** | **Исправление** | **Статус** |
| --- | --- | --- |
| DELETE /snippets/{uuid} не работал, потому что db.delete(snippet) был без await. | **Добавили await db.delete(snippet), теперь удаление работает.** | ✅ Исправлено |
| Сниппеты не удалялись из базы данных. | **Теперь удаляются и не возвращаются при повторном запросе.** | ✅ Исправлено |
| Нужно было протестировать удаление. | **Тестирование в Swagger UI показало, что удаление работает корректно.** | ✅ Исправлено |

✅ **Теперь удаление сниппетов работает корректно.**

**Итог**

✅ **Я исправил все ошибки, найденные супервайзером!**  
✅ **Все исправления протестированы в Swagger UI.**  
✅ **База данных, авторизация, WebSockets, регистрация, удаление работают корректно.**

**Проект полностью исправлен и соответствует требованиям!**