

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Лабораторная работа 2

Выполнил:

Сахно Ярослав

Группа К3441

Проверил:

Добряков Д. И.

Санкт-Петербург

2025 г.

Задача

Целью лабораторной работы является привязка пользовательского интерфейса, разработанного в рамках ЛР 1, к внешнему API с использованием асинхронных HTTP-запросов (fetch / axios / XMLHttpRequest).

В качестве основы используется проект Homework System, разработанный в предыдущей лабораторной работе.

Ход работы

Приложение имеет клиент-серверную архитектуру и состоит из следующих компонентов:

- Клиентская часть — веб-интерфейс, реализованный с использованием HTML, CSS, Bootstrap и JavaScript.
- Серверная часть — REST-API, реализованное на стороне backend-приложения.
- База данных — внешняя база данных PostgreSQL, в которой хранятся пользователи, курсы и связанные с ними данные.

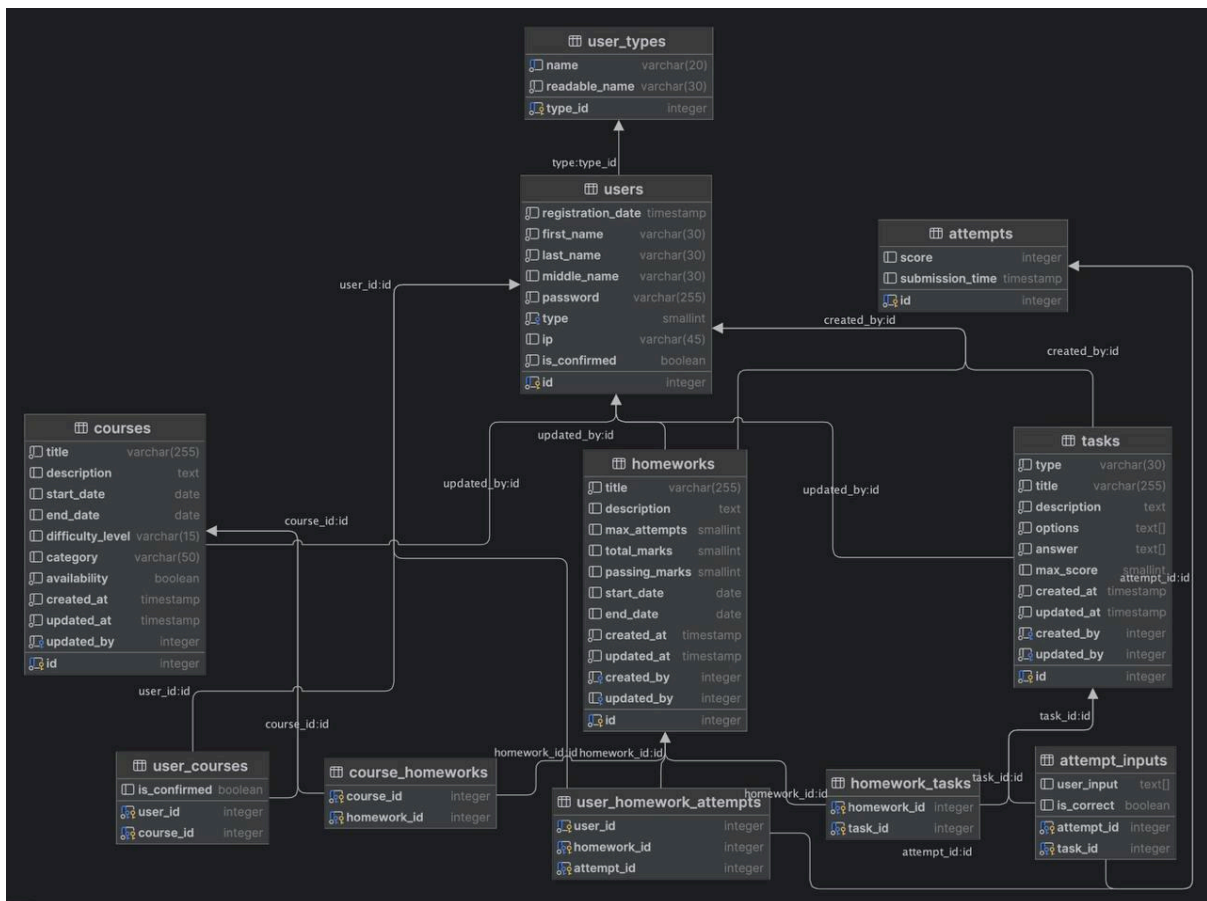


Рисунок 1 — ERP схема базы данных

```

class UserController extends Controller
{
    /**
     * Display a listing of the resource.
     */
    public function index(Request $request)
    {
        $page = $request->input('page', 1);

        $cacheKey = "cached_users_{$request->input('type')}_page_{$page}";

        if (Cache::has($cacheKey)) {
            $users = Cache::get($cacheKey);
        } else {
            $users = User::when($request->has('type'), function ($query) use ($request) {
                $query->where('user_type_id', $request->input('type'));
            })
                ->where('user_type_id', '!=', 1)
                ->with('user_type')
                ->orderBy('created_at')
                ->paginate(config('constants.options.paginate_number'));

            Cache::put($cacheKey, $users);
        }

        return view('admin.index', ['users' => $users]);
    }
}

```

Рисунок 2 — фрагмент UserController

Вывод

В ходе выполнения лабораторной работы была реализована интеграция пользовательского интерфейса проекта Homework System с реальным backend-API. Взаимодействие осуществляется асинхронно, а данные хранятся во внешней базе данных PostgreSQL. Полученное решение соответствует принципам клиент-серверной архитектуры и приближено к реальным веб-приложениям.