

Руководство по выполнению курсовой работы по C#

Общие положения

Формат работы: Командный проект (2-4 человека)

Технологический стек: ASP.NET WebAPI, Entity Framework, SQLite, WPF/Avalonia/Web

Итоговая оценка: Складывается из баллов за этапы + дополнительные баллы

Максимум базовых баллов: 100

Максимум дополнительных баллов: +100

Распределение ролей в команде

Перед началом работы команда должна распределить роли. Например:

- **Backend-разработчик** — ответственный за API, архитектуру, тесты
- **Frontend-разработчик** — клиентское приложение, UI/UX, интеграция с API
- **DevOps/QA** — БД, миграции, Git, тесты, документация

Важно: На защите будет проверяться индивидуальный вклад каждого участника.

Этап 1: Анализ и проектирование

Дедлайн: _____.2025

Максимальный балл: 15

Обязательные задания

- Созданы Use Case диаграммы — 5 балла
- Спроектирована ER-диаграмма БД — 5 балла
- Подготовлены макеты UI в Figma — 5 балла

Опциональные задания

- Интерактивный прототип в Figma (кликаемые переходы) — +2 балла

Что сдать: Ссылка на Figma, файлы диаграмм (.drawio, .png).

Этап 2: Разработка

Дедлайн: _____.2025

Максимальный балл: 65

Обязательные задания

Серверное приложение

- Реализовано WebAPI (CRUD операции) — 8 баллов**
- Настроена Entity Framework — 5 баллов**
- Реализована авторизация (JWT-токен, регистрация, логин, хеширование паролей) — 7 баллов**
- API соответствует REST (правильные HTTP-методы, статусы ответов) — 5 баллов**
- Подключен Swagger с описанием всех эндпоинтов — 3 балла**
- Функционал работы с файлами (загрузка/скачивание, валидация размера/типа) — 4 балла**
- Покрытие тестами NUnit/xUnit/Moq — 3 балла**

Клиентское приложение

- Реализовано клиентское приложение (WPF/Avalonia/Web/Mobile) — 8 баллов**
- Реализована авторизация на клиенте (хранение и обновление токена) — 5 баллов**
- Интеграция всех API-методов — 7 баллов**
- Асинхронные вызовы API (async/await, обработка загрузки) — 3 балла**
- Обработка ошибок (401, 404, сетевые ошибки с UI-уведомлениями) — 3 балла**
- Реализована работа с файлами (загрузка, отображение, скачивание) — 4 балла**

Опциональные задания

Серверное приложение

- Реализована пагинация и фильтрация в API — +2 балла**
- Использован паттерн Repository или Unit of Work — +2 балла**
- Настроен CORS — +1 балл**
- Логирование запросов (Serilog/NLog) — +1 балл**
- Microservices (минимум 3 сервиса) — +10 баллов**
- Identity Server (вместо простого JWT) — +4 баллов**
- Кеширование данных (IMemoryCache/Redis) — +3 балла**

Клиентское приложение

- Реализовано несколько клиентских приложений на разных платформах — +5 баллов
- SignalR Real-time обновления — +4 балла
- Адаптивный дизайн (для веб/мобильного клиента) — +2 балла
- Валидация данных на клиенте — +1 балл
- Реализована пагинация на UI — +1 балл
- Другое (согласовать лично) — +1-5 баллов

Что сдать: Рабочий API, работающее настольное приложение, ссылка на репозиторий с видимой историей коммитов.

Этап 3: Финализация и отчет

Дедлайн: ___. ___. 2025

Максимальный балл: 20

Обязательные задания

- GitHub репозиторий с четкой структурой, README с описанием проекта — 3 балла
- База данных, нормализованная до ЗНФ — 8 баллов
- Код соответствует стилю C# — 3 балла
- Подготовлен отчет (макс 30 страниц, все требования) — 4 балла
- Подготовлена презентация (~15 слайдов) — 2 балла

Опциональные задания

- Docker-контейнеризация (docker-compose для всей системы) — +4 балла
- CI/CD на GitHub Actions (автоматический запуск тестов) — +3 балла
- Обратный прокси-сервер (Nginx) — +3 балла
- Развёртывание на сервере (VDS/облако) — +3 балла
- Performance тесты (нагрузочное тестирование API) — +2 балла

Что сдать: Ссылка на финальный репозиторий, PDF отчета, PPTX презентации, доступ к развернутому приложению (если есть).

Требования к презентации

~15 слайдов, 10 минут демо + 5 минут вопросы

Обязательные слайды:

1. Титульный
 2. Проблема и цель проекта
 3. Распределение ролей в команде
 4. Use Cases
 5. ER-диаграмма
 6. Общая архитектура системы
 7. Стек технологий
 8. Демо API (Swagger)
 9. Демо клиентского приложения
 10. Дополнительная функциональность (если есть)
 11. Личный вклад каждого
-

Итоговая оценка

Сумма баллов	Оценка
0-49	Неудовлетворительно
50-69	Удовлетворительно
70-89	Хорошо
>90	Отлично

0 при:

- Отсутствии GitHub репозитория
 - Плагиате кода или отчета
-

Полезные ресурсы

Удачи! Вопросы задавайте на занятиях или в личных сообщениях.