Приложение разработано на ASP.NET MVC.

Выполнены все функциональные требования согласно заданию

На бэке:

* база SQLite в корне репозитория (папочка \_db), как и было предложено в задании. Для работы с базой использован EntityFramework через паттерн Репозиторий, UnitOfWork не используется, так как нет нужды (по фунционалу программы) объединять операции с БД в одну транзакцию. Для генерации и обновления схемы – миграции.
* Для удобства использования и просмотра приложения выполнена генерация тестовых данных, для генерации использовался нугет Bogus.
* Сервисный слой отсутствует ввиду отсутствия сложно бизнес-логики, репозитории инжектятся в контроллеры
* Авторизация/регистрация выполнена упрощено (в том функционале, какого требовало ТЗ) с помощью ASP.NET Identity, схема аутентификации на основе cookie

На фронте:

* классичеcкая схема MVC с SSR
* для UI используется Bootstrap, местами есть inline стили (осознанно, для упрощения и ускорения)
* работа с динамикой и ajax c помощью библиотеки htmx
* работа с домашней страницей полностью динамическая (фильтры, сортировка, подгрузка контента (paging))
* изменение фильтров или сортировки обнуляет пейджинг
* создание, обновление, удаление задач и комментов через формы в модалках также через ajax

тесты:

* использован xUnit
* тесты приведены интеграционные
* реализована общая функциональность по конфигурации тестового хоста, где я мокаю базу (in memory), аутентификацию юзера (чтобы тестировать Authorize контроллеры), антифогери
* тест-кейсы особо не расписаны, так как не было времени, приведены пару для примера

По функциональным штукам:

* Главная страница – Authorize. Можно залогинятся/зарегаться или зайти под существующим (сгенерированным Bogus) – пароль у всех 12345, login можно посмотреть в базе (поле UserName)
* работа с домашней страницей полностью динамическая (фильтры, сортировка, подгрузка контента (paging))
* изменение фильтров или сортировки обнуляет пейджинг
* При создании задачи статус назначается автоматом Created и дата создания
* При закрытии задачи – статус Done – дата закрытия выставляется автоматом
* Кнопки Update/Delete на странице задачи, видны только для автора задачи
* Комментарии – любые пользователи к любым задачам