2. ВВЕДЕНИЕ В ASP.NET WEB API

2.1. Организация структуры проекта по Clean Architecture

В контексте музыкального онлайн портала, организация структуры проекта по принципам Clean Architecture обеспечивает модульность, чистоту и независимость компонентов приложения. Ниже приведена примерная организация структуры проекта:

Domain Layer (Доменный слой):

В доменном слое содержится бизнес-логика приложения и определяются основные сущности и правила их взаимодействия. Здесь могут находиться следующие компоненты:

- Сущности (Entities): Определяют основные объекты предметной области. Например, модели для треков, альбомов, исполнителей и пользователей.

- Бизнес-правила (Business Rules): Включают в себя правила валидации данных, логику обработки заказов или реализации функций подписки на музыкальные сервисы.

Infrastructure Layer (Инфраструктурный слой):

Инфраструктурный слой содержит реализации абстракций из доменного слоя и обеспечивает взаимодействие приложения с внешними системами и ресурсами. Здесь могут находиться следующие компоненты:

- Репозитории (Repositories): Реализации интерфейсов репозиториев для работы с базой данных или другими источниками данных.

- Сервисы доступа к данным (Data Access Services): Компоненты, обеспечивающие доступ к данным, выполнение запросов к базе данных и работу с ORM.

- Внешние сервисы (External Services): Клиенты для взаимодействия с внешними сервисами, такими как API музыкальных платформ или сервисов оплаты.

Такая структура позволяет создать гибкое и масштабируемое приложение, разделяя бизнес-логику от инфраструктурных деталей.

2.2. Создание моделей приложения

Создание моделей приложения для музыкального онлайн портала включает определение основных объектов данных, которые будут использоваться в приложении. Примеры моделей для такого портала могут включать:

1. Модель Track (Трек):

- Название трека

- Исполнитель

- Альбом

- Продолжительность

- Жанр

- Рейтинг

2. Модель Album (Альбом):

- Название альбома

- Исполнитель

- Год выпуска

- Жанр

3. Модель Artist (Исполнитель):

- Имя исполнителя

- Жанры

- Альбомы

4. Модель User (Пользователь):

- Имя пользователя

- Плейлисты

- Оценки треков

- Отзывы

- Подписки на исполнителей

Эти модели могут быть расширены и дополнены в зависимости от функциональности и требований к приложению.

*Заключение:*

*Организация структуры проекта по принципам Clean Architecture и создание соответствующих моделей приложения позволяют разработать гибкое, масштабируемое и легко поддерживаемое приложение для музыкального онлайн портала. Разделение на доменный и инфраструктурный слои позволяет четко определить бизнес-логику и инфраструктурные компоненты, что способствует лучшей организации кода и облегчает его поддержку и развитие.*