**Слайд 1 (0:00–0:30)**  
Добрый день, уважаемые члены комиссии! Я, Угольников Даниил - студент 4 курса группы 2‑ИС, представляю дипломный проект на тему «Маркетплейс по продаже товаров для компании ruMarket».

**Слайд 2 (0:30–2:00)**  
Введение и актуальность.  
Тема разработки маркетплейса актуальна, поскольку - Электронная коммерция растёт экспоненциально: в 2024 году объём рынка превысил $6,3 трлн, и более 60 % всех продаж — через маркетплейсы. В России Wildberries показал рост оборота на 70 % год‑к‑году. Исходя из этого был выведен исследовательский аппарат:

**Слайд 3**  
**Проблема**: при таком спросе необходима своя гибкая платформа.  
**Цель работы**: разработать маркетплейс ruMarket.  
**Задачи** - вы можете видеть на экране.

**Слайд 4 (2:00–3:00)**  
Маркетплейс — это цифровая платформа‑посредник между множеством продавцов и покупателей.  
Они квалифицируются по:  
По типу участников: B2C (бизнес для бизнеса), B2B (бизнес для потребителя), C2C (потребитель для потребителя).  
По ассортименту: универсальные и нишевые.  
По модели монетизации: комиссия, подписка, реклама.  
Проведённый анализ типов маркетплейсов позволил выбрать модель **C2C** (потребитель для потребителя) с **универсальным ассортиментом** и **комиссионной схемой монетизации** как наиболее подходящую для целей проекта.

**Слайд 5 (3:00–4:00)**  
Существуют различные способы разработки веб-сайтов:  
конструкторы, cms-системы или с нуля.  
Я выбрал разработку с нуля для максимальной кастомизации и масштабируемости, что позволило гибко реализовать все бизнес‑требования.

**Слайд 6 (4:00–5:00)**  
Глава 2: выбор стека.  
• **Next.js 13+** с App Router для frontend и SSR.  
• **Prisma ORM** + **PostgreSQL** для БД.  
• **NextAuth.js** для аутентификации.  
• **Tailwind CSS** — утилитарный CSS‑фреймворк.  
Такой стек обеспечивает производительность, безопасность и SEO‑оптимизацию.

**Слайд 7 (5:00–6:00)**  
Среды разработки:  
• **Trae IDE** с AI‑ассистентом (GPT‑4o, Claude), поддержкой multimodal.  
• **VS Code**, **WebStorm**, **Sublime**, **Notepad++**.  
Trae IDE ускорила генерацию кода и рефакторинг, интеграция с GitHub облегчила командную работу.

**Слайд 8 (6:00–7:00)**  
Глава 3: архитектура.  
Три роли: гость, покупатель, продавец — с разграничением прав.  
Папка **app/** хранит все страницы, **api/** — серверные эндпоинты (REST: GET, POST, PUT, DELETE).  
Next.js App Router позволил чётко разделить логику и упростить защиту маршрутов.

**Слайд 9 (7:00–8:00)**  
Дизайн и UI.  
Основной цвет — надёжный глубокий синий #4438ca с фиолетовым оттенком.  
Utility‑first Tailwind гарантирует консистентность и позволяет создавать адаптивный интерфейс без лишнего CSS.

**Слайд 10 (8:00–9:30)**  
Демонстрация продукта.  
(Переключаюсь в браузер/демо‑режим)  
Показываю главную страницу, каталог, карточку товара и личный кабинет. Описываю основные сценарии: регистрация, поиск, оформление заказа.

**Слайд 11 (9:30–10:30)**  
Тестирование.  
Функциональное и регрессионное тесты на Pytest+Selenium.  
Speed Insights: среднее время FCP — 1,2 с, LCP — 2,1 с (отлично для e‑commerce).

**Слайд 12 (10:30–12:00)**  
Рекомендации по развитию.  
• Оптимизировать изображения WebP и lazy‑loading.  
• Добавить продвинутые фильтры и сортировку.  
• Внедрить систему рекомендаций на ML‑алгоритмах.  
Полный текст рекомендаций в отдельном документе, кнопки быстрого доступа в UI.

**Слайд 13 (12:00–15:00)**  
Заключение.  
Цель достигнута, все задачи выполнены, проблемы решены.  
Проект готов к дальнейшему сопровождению и масштабированию.  
Благодарю за внимание, готов ответить на ваши вопросы.