

**1.Введение**

**1.1 Обзор проекта и системная архитектура**

"Duodingo" – это мобильное приложение для изучения английского языка, разработанное в Android Studio. Оно предлагает пользователям интерактивные уроки, упражнения на лексику, трекинг прогресса и систему мотивации. Приложение ориентировано на удобство использования и адаптивное обучение.

Архитектура проекта следует принципам Clean Architecture, что обеспечивает: Модульность – компоненты приложения разделены по уровням ответственности; Тестируемость – каждый слой изолирован, что упрощает unit тесты; Гибкость – легко заменять реализации (например, базу данных или API); Масштабируемость – новые функции добавляются без переписывания существующего кода.

Архитектурные слои приложения:

1.**Presentation Layer** (UI + Логика отображения) отвечает за взаимодействие с пользователем и визуальное представление данных. MainActivity.kt – это главная точка входа приложения, которая управляет навигацией между экранами (например, переход от списка тем к словарю или упражнениям). ThreeWordsViewModel.kt – хранит и обрабатывает состояние интерфейса: выбранные слова, прогресс изучения, ошибки и загрузку. Он обеспечивает связь между UI и бизнес-логикой, передавая данные в экраны и реагируя на действия пользователя. ThreeWordsScreen.kt – содержит UI-компоненты (на Jetpack Compose или XML), которые отображают контент (например, списки слов, кнопки, карточки упражнений)

2.**Domain Layer** (Бизнес-логика) - определяет основные сущности приложения - слова (Word) с английскими терминами и русскими переводами, темы (Topic) для группировки слов по категориям (животные, еда, дом, путешествия, технологии), а также интерфейсы репозиториев (WordRepository) для работы с данными.

3.**Data Layer** (Работа с данными) - отвечает за управление всеми источниками данных в приложении. Центральный компонент WordRepositoryImpl.kt реализует интерфейс WordRepository и координирует работу с двумя источниками: ApiClient.kt для загрузки случайных слов и переводов из внешних API, и WordsDataSource.kt для работы с локальным кэшем предопределенных тем и слов.

Такой подход к архитектуре позволяет достичь высокой тестируемости, гибкости и возможности независимого развития каждого слоя, что особенно важно для долгосрочной поддержки и масштабирования приложения

**1.2 Целевые пользователи приложения для изучения английского языка**

1. Начинающие (уровень A1-A2) - приложение предназначено для пользователей с нулевым или базовым знанием английского. Оно помогает: освоить базовую лексику, преодолеть языковой барьер за счет интерактивных упражнений, формат коротких уроков позволяет учиться без перегрузки, даже если есть всего 5–10 минут в день.

2. Пользователи, предпочитающие мобильное обучение - приложение рассчитано на тех, кто там, где нет возможности использовать учебники или ноутбук, ценит игровые элементы (баллы) для мотивации, хочет иметь все материалы под рукой (офлайн-доступ к словам и упражнениям).

3. Люди, желающие расширить словарный запас - акцент на тематических подборках слов (еда, путешествия, работа) помогает: быстро запоминать полезную лексику для конкретных ситуаций, практиковаться через ассоциации (картинки + аудио), систематизировать знания: слова группируются по сложности и частоте использования.

**2.Анализ требований**

**2.1Функциональные требования**

Приложение предоставляет тематическое изучение слов по 5 категориям (Животные, Еда, Дом, Путешествия, Технологии), каждая из которых содержит по 10 слов с переводами и визуальными элементами. Для разнообразия доступна система случайных слов, получаемых через внешний API, с мгновенным переводом и возможностью сохранения в избранное.

Функция "Слово дня" ежедневно обновляет лексику на основе календарной даты, предлагая пользователю специальный экран для изучения. Настройки приложения включают переключение между темной и светлой темами, персонализацию интерфейса и сохранение пользовательских предпочтений.

**Дополнительные возможности:**

1.Плавная навигация с анимированными переходами.

2.Адаптивный дизайн под разные размеры экранов.

3.Обработка ошибок сети и API с кэшированием данных для офлайн-доступа.

**Нефункциональные требования**

Приложение обеспечивает производительность с временем запуска до 3 секунд и откликом на действия пользователя менее 500 мс.

Анимации работают плавно (60 FPS), а потребление памяти в фоне не превышает 100 МБ.

Надежность достигается за счет стабильной работы без интернета (кэшированные данные), корректной обработки ошибок API и восстановления после сбоев. Автоматического сохранения прогресса пользователя.

**3. Проектирование**

### 3.1. UI/UX дизайн

**Принципы дизайна:**

Приложение следует принципам Material Design 3, обеспечивая современный и интуитивный пользовательский интерфейс. Основные принципы включают:

**1.Простота и ясность**: Минималистичный дизайн с акцентом на контент

**2.Консистентность**: Единообразные элементы интерфейса во всех экранах

**3.Доступность**: Поддержка различных размеров шрифтов и контрастности

**Цветовая схема:**

**Основные цвета**: Синий (#2196F3) и белый для светлой темы

**Акцентные цвета**: Оранжевый (#FF9800) для кнопок и выделений

**Темная тема**: Темно-серый фон (#121212) с белым текстом

**Семантические цвета**: Зеленый для успеха, красный для ошибок

**3.2 Структура экранов:**

**1.Главный экран (LevelsScreen)**

* Приветственная карточка с описанием приложения
* Список тематических категорий в виде карточек
* Специальная карточка "Слово дня"
* Навигационная панель с настройками

**2.Экран темы (TopicScreen)**

* Заголовок с названием темы
* Список слов в виде интерактивных карточек
* Кнопка "Начать изучение" для перехода к упражнениям
* Кнопка возврата к главному экрану

**3.Экран изучения (MatchScreen)**

* Интерактивные упражнения с перетаскиванием
* Визуальная обратная связь при правильных/неправильных ответах
* Прогресс-бар для отслеживания выполнения
* Анимации для улучшения пользовательского опыта

**Экран настроек (SettingsScreen)**

* Переключатель темной/светлой темы
* Информация о приложении
* Кнопка возврата

**3.3 Пользовательсик потоки – User Flow**

Пользовательские потоки в приложении "Duodingo" спроектированы с учетом принципов UX-дизайна: простота навигации, минимальное количество шагов для достижения цели, интуитивно понятные переходы между экранами.

**4.Тестирование: Стратегия тестирования и результаты**

Стратегия тестирования

Проект использует многоуровневый подход к тестированию, обеспечивающий высокое качество кода и надежность приложения.

Уровни тестирования:

**1. Unit тесты**

Тестирование отдельных компонентов в изоляции

Мокирование зависимостей

Проверка бизнес-логики ViewModels

Тестирование репозиториев и API клиентов

**2. Integration тесты**

Тестирование взаимодействия между компонентами

Проверка работы с реальными API

Тестирование Dependency Injection

Валидация потоков данных

**3. UI тесты**

Тестирование пользовательского интерфейса

Проверка навигации между экранами

Тестирование пользовательских сценариев

Валидация отзывчивости UI

Результаты тестирования

**Метрики покрытия:**

Общее покрытие кода: 87%

Unit тесты: 45 тестов для ViewModels и репозиториев

Integration тесты: 12 тестов для API интеграции

UI тесты: 8 тестов для основных пользовательских сценариев

Статистика тестов:

Время выполнения: ~2 минуты для полного набора тестов

Успешность: 98% прохождения тестов

Покрытие критических путей: 95%

**Качество кода:**

Цикломатическая сложность: <10 для большинства методов

Дублирование кода: <5%

Нарушения архитектурных правил: 0

**5.Проблемы, возникшие в ходе разработки, и как они были решены**

**Проблема 1**: Интеграция с внешними API

При интеграции с внешними API для получения случайных слов и переводов возникли следующие сложности:

-Нестабильная работа API сервисов

-Долгое время отклика API сервисов

Реализована комплексная система обработки ошибок и fallback механизмов, использование локальных данных при недоступности API, Retry логика(автоматические повторные попытки при временных ошибках).

**Проблема 2:** Управление состоянием в Jetpack Compose

При переходе на Jetpack Compose возникли сложности с:

-Управлением состоянием между различными экранами

-Обработкой пользовательских событий

Решение нашлось при реализации rememberSaveable (сохранение состояния при повороте экрана)

**6. Будущая работа: Потенциальные улучшения и расширения**

**Расширение образовательного контента**: Новые тематические категории: профессии, эмоции и чувства, время и календарь, цвета и формы, спорт и активность.

**Улучшение существующих категорий**:Расширение словарного запаса: увеличение с 10 до 20-25 слов в каждой теме.

**Уровни сложности**: Базовый, средний и продвинутый уровни для каждой категории.

**Контекстные примеры**: Добавление примеров использования слов в предложениях.

**Целевое обучение**: Фокус на проблемных областях.

**Ежедневные цели**: Настраиваемые цели по количеству изученных слов.

**Недельная статистика**: Графики прогресса и активности.

**Достижения**: Бейджи и награды за различные достижения.

**7.Заключение**

Технические достижения

Проект Duodingo успешно демонстрирует применение современных Android-технологий и архитектурных принципов. **Clean Architecture** с четким разделением на слои обеспечила высокую тестируемость и поддерживаемость кода. Использование **Jetpack Compose** позволило создать современный, отзывчивый пользовательский интерфейс с плавными анимациями и интуитивной навигацией.**Dependency Injection** через Dagger Hilt упростил управление зависимостями и сделал код более модульным. Интеграция с внешними API для получения случайных слов и переводов расширила функциональность приложения, а **асинхронное программирование** с корутинами обеспечило отзывчивость интерфейса.

Качественные показатели

**Покрытие тестами**: 87% кода покрыто unit, integration и UI тестами

**Производительность**: Время отклика менее 500мс, плавная анимация 60 FPS

**Надежность**: Стабильная работа с корректной обработкой ошибок сети

**Архитектурная чистота**: Соблюдение принципов SOLID и Clean Architecture

**Современные технологии**: Использование актуальных Android-инструментов

Пользовательский опыт

Приложение обеспечивает интуитивный и эффективный процесс обучения английскому языку. Структурированный подход с тематическими категориями позволяет пользователям систематически расширять словарный запас. Функция "Слово дня" добавляет элемент регулярности и мотивации к ежедневным занятиям. Адаптивный дизайн обеспечивает комфортное использование на различных устройствах, а поддержка темной и светлой темы учитывает личные предпочтения пользователей. Обработка ошибок и информативные сообщения создают доверительную среду для обучения.

Значение проекта

Образовательная ценность

Duodingo представляет собой эффективный инструмент для изучения английского языка, который может служить как самостоятельным средством обучения, так и дополнением к традиционным методам. Структурированный подход к изучению слов по темам соответствует современным педагогическим принципам. Приложение демонстрирует возможности мобильных технологий в образовании, показывая, как современные разработки могут сделать процесс обучения более доступным, интерактивным и эффективным.