

Инженерная школа информационных технологий и робототехники  
 Направление: «Программная инженерия»  
 Отделение: «Информационные технологии»

### БАКАЛАВРСКАЯ РАБОТА

Тема работы
Разработка сервиса для изучения иностранных языков

УДК 004.75:004.455:004.85:81'243

Студент

Группа	ФИО	Подпись	Дата
8K81	Бойко Григорий Дмитриевич		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Фадеев Александр Сергеевич	Кандидат технических наук		

### КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор	Гасанов Магеррам Али оглы	Доктор экономических наук		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Мезенцева Ирина Леонидовна			

### ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Чердынцев Евгений Сергеевич	Кандидат технических наук		

## Планируемые результаты обучения по ООП

Код результатов	Результат обучения (выпускник должен быть готов)
P1	Применять базовые и специальные естественнонаучные и математические знания в области информатики и вычислительной техники, достаточные для комплексной инженерной деятельности.
P2	Применять базовые и специальные знания в области современных информационных технологий для решения инженерных задач.
P3	Ставить и решать задачи комплексного анализа, связанные с созданием аппаратно-программных средств информационных и автоматизированных систем, с использованием базовых и специальных знаний, современных аналитических методов и моделей.
P4	Разрабатывать программные и аппаратные средства (системы, устройства, блоки, программы, базы данных и т. п.) в соответствии с техническим заданием и с использованием средств автоматизации проектирования.
P5	Проводить теоретические и экспериментальные исследования, включающие поиск и изучение необходимой научно-технической информации, математическое моделирование, проведение эксперимента, анализ и интерпретация полученных данных, в области создания аппаратных и программных средств информационных и автоматизированных систем.
P6	Внедрять, эксплуатировать и обслуживать современные программно-аппаратные комплексы, обеспечивать их высокую эффективность, соблюдать правила охраны здоровья, безопасность труда, выполнять требования по защите окружающей среды.
P7	Универсальные компетенции
P8	Использовать базовые и специальные знания в области проектного менеджмента для ведения комплексной инженерной деятельности.
P9	Владеть иностранным языком на уровне, позволяющем работать в иноязычной среде, разрабатывать документацию, презентовать и защищать результаты комплексной инженерной деятельности.
P10	Эффективно работать индивидуально и в качестве члена группы, состоящей из специалистов различных направлений и квалификаций, демонстрировать ответственность за результаты работы и готовность следовать корпоративной культуре организации.
P11	Демонстрировать знания правовых, социальных, экономических и культурных аспектов комплексной инженерной деятельности.

Инженерная школа информационных технологий и робототехники

Направление: «Программная инженерия»

Отделение: «Информационные технологии»

**УТВЕРЖДАЮ:**  
 Руководитель ООП

\_\_\_\_\_  
 (Подпись)    (Дата)    (Ф.И.О.)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

Бакалаврской работы

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8K81	Бойко Григорию Дмитриевичу

Тема работы:

Разработка сервиса для изучения иностранных языков	
Утверждена приказом директора (дата, номер)	09.02.2022 №40-51/с

Срок сдачи студентом выполненной работы:	06.06.2022
--	------------

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<p><b>Исходные данные к работе</b></p> <p><i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i></p>	<p>Объект исследования: мобильное Flutter приложение для изучения иностранных языков.</p> <p>Предмет исследования: подходы к изучению иностранных языков, архитектура информационной системы для изучения иностранных языков.</p> <p>Режим работы: непрерывный</p>
---	--

<p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b></p> <p><i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<ol style="list-style-type: none"> <li>1. Средства разработки и мобильные приложения для изучения языков</li> <li>2. Проектирование платформы</li> <li>3. Реализация мобильного приложения</li> <li>4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</li> <li>5. Социальная ответственность</li> </ol>
<p><b>Перечень графического материала</b></p> <p><i>(с точным указанием обязательных чертежей)</i></p>	<p>Презентация формата *.pptx</p>
<p><b>Консультанты по разделам выпускной квалификационной работы</b></p> <p><i>(с указанием разделов)</i></p>	
<p style="text-align: center;"><b>Раздел</b></p>	<p style="text-align: center;"><b>Консультант</b></p>
<p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p>Гасанов Магеррам Али оглы</p>
<p>Социальная ответственность</p>	<p>Мезенцева Ирина Леонидовна</p>
<p><b>Названия разделов, которые должны быть написаны на русском и иностранном языках:</b></p>	
<p> </p>	

<p><b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b></p>	<p> </p>
--	----------

**Задание выдал руководитель / консультант (при наличии):**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Фадеев Александр Сергеевич	Кандидат технических наук		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8K81	Бойко Григорий Дмитриевич		

Инженерная школа информационных технологий и робототехники  
 Направление: «Программная инженерия»  
 Отделение: «Информационные технологии»  
 Период выполнения: (осенний / весенний семестр 2021 /2022 учебного года)

Форма представления работы:

бакалаврская работа

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

### КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	06.06.2021
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
	Основная часть	75
	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
	Социальная ответственность	10

**СОСТАВИЛ:**

**Руководитель ВКР**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Фадеев Александр Сергеевич	Кандидат технических наук		

**СОГЛАСОВАНО:**

**Руководитель ООП**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Чердынцев Евгений Сергеевич	Кандидат технических наук		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И  
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

<b>Группа</b>	<b>ФИО</b>
8К81	Бойко Григорию Дмитриевичу

<b>Школа</b>	ИШИТР	<b>Отделение школы (НОЦ)</b>	Отделение информационных технологий
<b>Уровень образования</b>	Бакалавриат	<b>Направление/специальность</b>	Программная инженерия

**Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:**

<i>1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	<i>Оклад инженера – 21760 руб. Оклад руководителя – 33664 руб.</i>
<i>2. Нормы и нормативы расходования ресурсов</i>	<i>Премимальный коэффициент 30%; Коэффициент доплат и надбавок 20%; Районный коэффициент 30%; Коэффициент дополнительной заработной платы 12%; Накладные расходы 16%.</i>
<i>3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	<i>Коэффициент отчислений на уплату во внебюджетные фонды 27,1%</i>

**Перечень вопросов, подлежащих исследованию, проектированию и разработке:**

<i>1. Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	<i>Анализ конкурентных технических решений</i>
<i>2. Планирование и формирование бюджета научных исследований</i>	<i>Формирование плана и графика разработки: -определение структуры и трудоемкости работ; -разработка графика Гантта. Формирование бюджета затрат: -затраты на специальное оборудование; -заработная плата (основная и дополнительная); -отчисления на социальные цели; -накладные расходы.</i>
<i>3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	<i>Определение потенциального эффекта исследования</i>

**Перечень графического материала (с точным указанием обязательных чертежей):**

<ol style="list-style-type: none"> <li>1. Оценка конкурентоспособности технических решений</li> <li>2. Матрица SWOT</li> <li>3. Альтернативы проведения НИ</li> <li>4. График проведения и бюджет НИ</li> <li>5. Оценка ресурсной, финансовой и экономической эффективности НИ</li> </ol>
---

**Дата выдачи задания для раздела по линейному графику**

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Профессор ОСГН	Гасанов М.А.	д.э.н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8K81	Бойко Григорий Дмитриевич		

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

<b>Группа</b> 8К81		<b>ФИО</b> Бойко Григорию Дмитриевичу	
<b>Школа</b>	<b>Инженерная школа информационных технологий и робототехники</b>	<b>Отделение (НОЦ)</b>	<b>Отделение информационных технологий</b>
<b>Уровень образования</b>	Бакалавриат	<b>Направление/специальность</b>	09.04.04 Программная инженерия

Тема ВКР:

<b>Разработка сервиса для изучения иностранных языков</b>	
<b>Исходные данные к разделу «Социальная ответственность»:</b>	
<p><b>Введение</b></p> <ul style="list-style-type: none"> <li>- Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика) и области его применения.</li> <li>- Описание рабочей зоны (рабочего места) при разработке проектного решения</li> </ul>	<p><i>Объект исследования:</i> мобильное Flutter приложение для изучения иностранных языков.</p> <p><i>Область применения:</i> предоставление информации и инструментов для изучения иностранных языков онлайн.</p> <p><i>Рабочая зона:</i> офис.</p> <p>Размеры помещения: 5*3 м.</p> <p><i>Количество и наименование оборудования рабочей зоны:</i> ноутбук.</p> <p><i>Рабочие процессы, связанные с объектом исследования, осуществляемые в рабочей зоне:</i> проектирование, разработка программного обеспечения.</p>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p><b>1. Правовые и организационные вопросы обеспечения безопасности при разработке проектного решения:</b></p> <ul style="list-style-type: none"> <li>- специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</li> <li>- организационные мероприятия при компоновке рабочей зоны.</li> </ul>	<p>«Трудовой кодекс Российской Федерации» от 30.12.2001 N 197-ФЗ (ред. от 25.02.2022) ГОСТ 21889-76 Система «человек-машина». Кресло человека-оператора. Общие эргономические требования. ГОСТ Р 50923-96 Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения</p>
<p><b>2. Производственная безопасность при разработке проектного решения:</b></p> <ul style="list-style-type: none"> <li>- Анализ выявленных вредных и опасных производственных факторов</li> </ul>	<p><b>Вредные факторы:</b></p> <ul style="list-style-type: none"> <li>• Статические физические перегрузки, связанные с рабочей позой.</li> <li>• Умственное перенапряжение, в том числе вызванное информационной нагрузкой.</li> <li>• Перенапряжение анализаторов, в том числе вызванное информационной нагрузкой.</li> <li>• Монотонность труда, вызывающая монотонию.</li> <li>• Отсутствие или недостаток необходимого естественного освещения.</li> <li>• Отсутствие или недостаток необходимого искусственного освещения.</li> </ul> <p><b>Опасные факторы:</b></p> <p>1. Опасность поражения электрическим током</p> <p><b>Требуемые средства коллективной защиты от выявленных факторов:</b> системы естественного освещения, приборы искусственного освещения, очки защитные со спектральными фильтрами ЛС и НСФ., устройства защитного отключения (УЗО), заземление, зануление корпусов электрооборудования.</p>
<p><b>3. Экологическая безопасность при разработке проектного решения</b></p>	<p>Анализ негативного воздействия на окружающую природную среду: утилизация компьютеров и другой оргтехники</p>

	Негативное воздействие на гидросферу и атмосферу совершается посредством производства различной оргтехники Воздействие на литосферу в результате образования отходов при поломке оргтехники
<b>4. Безопасность в чрезвычайных ситуациях при разработке проектного решения</b>	Возможные ЧС: Природные катастрофы (наводнения, цунами, ураган и т.д.); Геологические воздействия (землетрясения, оползни, обвалы, провалы территории); Техногенные аварии (пожар, сбои в электроснабжении)  Наиболее типичная ЧС (пожар)
Дата выдачи задания для раздела по линейному графику	

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Мезенцева Ирина Леонидовна			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8К81	Бойко Григорий Дмитриевич		

## РЕФЕРАТ

Учебно-исследовательская работа содержит 98 страниц, 51 рисунок, 17 таблиц и 32 источника.

Ключевые слова: проектирование, информационная система, мобильное приложение, иностранный язык, обучение.

Цель работы — исследование, проектирование и разработка мобильного приложения для изучения иностранных языков.

Область применения: предоставление информации и инструментов для изучения иностранных языков онлайн.

В первой главе анализируются способы изучения новых слов, приложения-аналоги, средства разработки. Выбраны технологии для реализации серверной части, база данных, API получения перевода. Описаны используемые средства разработки.

Вторая глава содержит проектирование интерфейса и способа взаимодействия в приложении, алгоритм повтора слов и упражнений, а также физическую модель базы данных.

В третьей главе описан процесс создания мобильного приложения, настройка серверной части, аутентификация пользователей, взаимодействие с сервером, перевод слов и результаты реализации приложения.

Четвертая глава представляет собой выполненное задание по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение», в котором отображены потенциал, планирование и эффективность решения.

Пятая глава представляет собой выполненное задание по разделу «Социальная ответственность», в котором рассмотрены организационноправовые, производственные и экологические аспекты безопасности, а также безопасность в чрезвычайных ситуациях.

## ОГЛАВЛЕНИЕ

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ.....	14
ВВЕДЕНИЕ .....	15
1 СРЕДСТВА РАЗРАБОТКИ И МОБИЛЬНЫЕ ПРИЛОЖЕНИЯ ДЛЯ ИЗУЧЕНИЯ ЯЗЫКОВ.....	18
1.1 Решение проблем в изучении иностранных языков.....	18
1.1.1 Забывание новых слов и теории.....	18
1.1.2 Сложность в говорении (speaking).....	20
1.1.3 Выводы .....	20
1.2 Анализ приложений-аналогов .....	21
1.3 Сравнение средств разработки.....	23
1.4 Принцип построения интерфейса в Flutter.....	27
1.5 Выбор серверной части.....	29
1.6 Выбор базы данных .....	30
1.7 Выбор API получения перевода .....	31
1.8 Средства разработки.....	31
2 ПРОЕКТИРОВАНИЕ .....	33
2.1 Разработка UI и способа взаимодействия в приложении .....	33
2.1.1 Основная навигация .....	33
2.1.2 Словарь .....	33
2.1.3 Занятия .....	37
2.1.4 Чтение .....	41
2.2 Алгоритм повтора слов и упражнений .....	43
2.3 Проектирование физической схемы базы данных .....	44
3 РЕАЛИЗАЦИЯ МОБИЛЬНОГО ПРИЛОЖЕНИЯ .....	47
3.1 Реализация аутентификации.....	47

3.2 Создание экранов авторизации и регистрации .....	47
3.2.2 Подключение проекта к Firebase.....	49
3.2.3 Настройка взаимодействия с системой аутентификации Firebase..	50
3.3 Создание базы данных .....	53
3.4 Работа с сервером .....	55
3.5 Реализация перевода по нажатию .....	57
3.5.1 Распознавание слов.....	57
3.5.2 Получение перевода .....	60
Вывод по главе .....	63
<b>4 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ .....</b>	<b>64</b>
4.1 Потенциальные потребители результатов исследования .....	64
4.2 Анализ конкурентных технических решений.....	65
4.3 Технология QuaD.....	67
4.4 SWOT-анализ .....	68
4.5 Планирование научно-исследовательских работ .....	70
4.5.1 Структура работ в рамках научного исследования .....	70
4.5.2 Определение трудоемкости выполнения работ.....	71
4.5.3 Разработка графика проведения научного исследования.....	72
4.6 Бюджет научно-технического исследования .....	73
4.6.1 Расчет материальных затрат НТИ.....	74
4.6.2 Расчет затрат на специальное оборудование для научных (экспериментальных) целей.....	74
4.6.3 Основная заработная плата исполнителей темы .....	75
4.6.4 Дополнительная заработная плата исполнителей темы.....	77
4.6.5 Отчисления во внебюджетные фонды (страховые отчисления).....	77
4.6.6 Накладные расходы .....	78

4.6.7 Формирование бюджета затрат научно-исследовательского проекта .....	79
4.7 Определение потенциального эффекта исследования .....	79
Вывод по главе .....	81
5 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ.....	82
5.1 Правовые и организационные вопросы обеспечения безопасности .....	82
5.1.1 Правовые нормы трудового законодательства .....	82
5.1.2 Эргономические требования к правильному расположению и компоновке рабочей зоны.....	83
5.2 Производственная безопасность .....	85
5.2.1 Статические физические перегрузки .....	86
5.2.2 Умственное перенапряжение и перенапряжение анализаторов, монотонность труда.....	86
5.2.3 Отсутствие или недостаток необходимого естественного освещения .....	87
5.2.4 Отсутствие или недостаток необходимого искусственного освещения.....	88
5.2.5 Опасность поражения электрическим током .....	89
5.3 Экологическая безопасность .....	90
5.4 Безопасность в чрезвычайных ситуациях .....	91
Выводы по главе .....	92
ЗАКЛЮЧЕНИЕ.....	94
СПИСОК ЛИТЕРАТУРЫ .....	95

## ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

1. UI (англ. User Interface) — Пользовательский интерфейс.
2. Мнемотехника — запоминание на основе визуального мышления.
3. ОС – операционная система.
4. SQL (англ. Structured Query Language) — Декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных.
  5. API (Application Programming Interface) — описание способов (набор классов, процедур, функций, структур или констант), которыми одна компьютерная программа может взаимодействовать с другой программой.
  6. БД – база данных.
  4. RSS канал — Семейство XML-форматов, предназначенных для описания лент новостей, анонсов статей.
  5. XML (eXtensible Markup Language) — расширяемый язык разметки.
  8. Фреймворк (с англ. framework — «каркас, структура») — заготовка, готовая модель в IT для быстрой разработки, на основе которой можно дописать собственный код.
  7. SWOT – Strengths Weaknesses Opportunities Threats.
  8. ГОСТ – государственный стандарт.

## ВВЕДЕНИЕ

В настоящее время люди в России считают важным знать иностранные языки, и чем более населенный город, тем больше людей владеют ими [1]. К сожалению, далеко не каждый может позволить себе занятия с преподавателем или в языковой школе, да и время выделить на изучение бывает сложно. Одна из самых долгих частей в изучении языка — это пополнение словарного запаса, слова забываются или долго вспоминаются и их приходится перечислять. К тому же забывается теоретический материал, а если даже человек помнит теорию, грамматику, то далеко не всегда он может применить ее на практике в реальной речи, он может все знать, все понимать, но с процессом построения предложений в реальном времени иметь проблемы, так как навык говорения сложный для освоения, что и показало исследование студентов изучающих английский как иностранный язык [2].

Существуют следующие серьезные проблемы при изучении иностранных языков:

- забывание новых слов и теоретического материала;
- сложность в говорении (*speaking*).

В первом случае дело в том, что человек забывает до 60% новой информации уже через час [3]. В примерной рабочей программе основного общего образования по английскому языку, одобренную решением федерального учебно-методического объединения по общему образованию [4] сказано, что на этапе основного общего образования минимально допустимое количество учебных часов, выделяемых на изучение первого иностранного языка, — 3 часа в неделю, что составляет по 102 учебных часа на каждом году обучения с 5 по 9 класс. Это означает, что расписание занятий не может предусматривать ежедневные уроки иностранного языка, между ними есть перерывы в один или несколько дней, в которых ученики могут забыть новую информацию, если они ее не будут ее повторять [3].

Вторая проблема связана с малым количеством практики. В разделе предметных результатов примерной рабочей программе основного общего

образования по английскому языку [4] для каждого класса указаны необходимые знания, умений и навыки, которые должны овладеть школьники, чтобы соответствовать предполагаемым результатам обучения по различным стандартам. Если учитывать, что школа будет согласно ч. 7 ст. 12 Федерального закона № 273-ФЗ [5] соблюдать общие положения о создании необходимых условий для освоения каждым обучающимся образовательной программы и поделит среднестатистический класс 5-9 классов из 20 человек [6] на 2 группы поровну, то в каждой группе будет по 13 учеников, урок длится 45 минут, а таких всего 3 в неделю, времени на каждого ученика будет только 3,5 минуты, это не учитывая то, что учитель может рассказать тему, проверить знания за предыдущий урок, задать домашнее задания и т.д., то реального времени на конкретного учащегося будет еще меньше. Помимо навыка говорения в предметных результатах много других умений, это означает, что каждый ученик в течении урока может практиковать этот навык совсем незначительное время.

Исследования [7] показывают, что учащемуся нужны не только факты, но и понимание контекста, в котором эта информация имеет смысл. Для приобретения какого-либо навыка нужен опыт, и даже необязательно знать теорию (правила), с помощью примеров человек может сам извлечь необходимые правила и применить их в дальнейшем. Благодаря примерам и приобретению опыта, ученик получает новые нейронные связи, основываясь на прошлом опыте, которые он может распознавать в дальнейшем (например, водитель понимает, когда переключить передачу, ориентируясь на звук двигателя, а не на спидометр). Опытный водитель, безусловно, с большей вероятностью безопасно преодолет поворот, чем компетентный водитель, который тратит много время анализируя дополнительные параметры, а в говорении очень важна скорость для того, чтобы добиться ее, нужно получить как можно больше разнообразного опыта, доведя этот навык до автоматизма [7]. Таким образом времени урока для каждого ученика в классе недостаточно,

чтобы получить много опыта в говорении, а перерывы между уроками усугубляют этот процесс, так как информация забывается.

Для того, чтобы решить проблему повторения и запоминания иностранных слов, теоретического материала, а также малой практики, в настоящей работе была поставлена цель спроектировать и создать мобильное приложение для изучения иностранных языков.

Для достижения этой цели были поставлены следующие основные задачи проекта:

**Задача 1.** Найти решение проблем в изучении иностранных языков.

**Задача 2.** Проанализировать приложения-аналоги.

**Задача 3.** Проработать UI и способ взаимодействия в приложении.

**Задача 4.** Спроектировать физическую схему базы данных.

**Задача 5.** Реализовать авторизацию в приложении.

**Задача 6.** Настроить работу с сервером.

**Задача 7.** Реализовать перевод слов.

Объектом исследования является мобильное Flutter приложение для изучения иностранных языков.

Предметом исследования являются подходы к изучению иностранных языков, архитектура информационной системы для изучения иностранных языков.

# 1 СРЕДСТВА РАЗРАБОТКИ И МОБИЛЬНЫЕ ПРИЛОЖЕНИЯ ДЛЯ ИЗУЧЕНИЯ ЯЗЫКОВ

## 1.1 Решение проблем в изучении иностранных языков

### 1.1.1 Забывание новых слов и теории

Существует много способов запоминания иностранных слов, популярными из которых являются [8]:

- карточки;
- группировка слов;
- мнемотехника;
- интервальные повторения;
- метод историй.

Карточки представляют собой иностранное слово на одной стороне, а перевод на другой. Слова, которые хуже помнятся, или новые – выше в колоде всех слов и попадают чаще.

Группировка слов может быть по различным признакам для лучшего запоминания, так как эти слова что-то объединяет, например, группировка по смыслу, по тематике, по грамматическим признакам (например, неправильные глаголы в английском языке).

Мнемотехника представляет собой визуализацию образа, связывающего иностранное слово и его перевод. Например, слово *top* (швабра) звучит как «мопс», можно представить образ большого мопса, который, пританцовывая под музыку, энергично моет пол в квартире шваброй.

Метод интервальных повторения основан на графике забывания Эббингауза (рисунок 1). Эббингауз проводил эксперименты, в которых выяснил, что без повторений в первые пол часа мы забываем около 50% новой информации, через 10 часов после заучивания в памяти остаётся около 35% от изученного. Далее процесс забывания протекает медленно, и через 6 дней в памяти остаётся около 20% от общего числа первоначально выученной информации, столько же остаётся в памяти и через месяц [3].

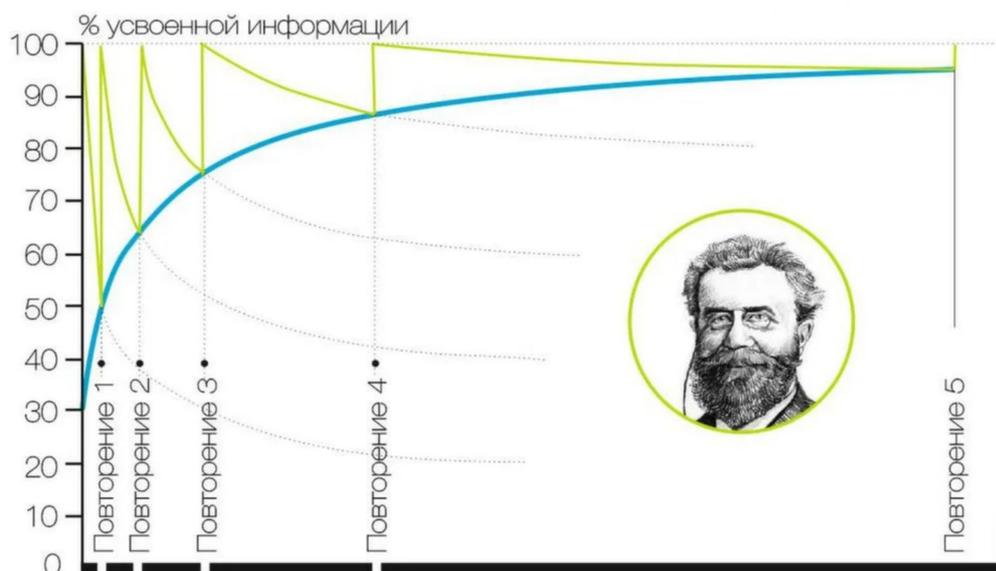


Рисунок 1 – Кривая забывания Эббингауза

Метод историй заключается в том, чтобы составить из незнакомых слов яркий и необычный рассказ. Вспоминая эту историю, будут вспоминаться и новые слова, которые были в ней использованы.

На основе анализа достоинств и недостатков каждого метода было принято решение использовать комбинацию всех перечисленных методов. Для хранения слов будут использоваться карточки, которые можно группировать в различные наборы по усмотрению пользователя, а для лучшего запоминания изучающему будет предоставляться возможность написать историю или другую необходимую информацию в заметках к карточке.

Для того, чтобы уменьшить скорость забывания — было решено использовать систему интервальных повторений, когда каждое правильно вспомненное слово необходимо повторять через увеличенный интервал времени, а если слово пользователь не помнит, то весь прогресс начинается сначала [9]. То же самое и с теоретическим материалом, его нужно также периодически повторять, только будет повторяться не сам материал, а связанные с ним упражнения, это выглядит более логично и практично, так как с практикой она лучше запоминается [7].

Помимо этого, карточки будут применяться для получения практики, вместо слов будут примеры, закрепляющие относящуюся к ним теорию, это

позволит увеличить скорость приобретения опыта и выделит приложения на фоне конкурентов. В этих карточках также будет использоваться метод интервальных повторений, о котором говорилось выше.

### **1.1.2 Сложность в говорении (speaking)**

Решение второй проблемы заключается в том, чтобы увеличить скорость говорения. И речь идёт не про скорость произношения слов, а про скорость вспоминания их в голове, что позволяет вспомнить нужную конструкцию/правило/слово в нужный момент. То есть нужно довести навык воспроизведения той или иной фразы/конструкции до автоматизма и тогда ее уже будет трудно забыть, соответственно и сказать что-то в различных ситуациях, по аналогии с катанием на велосипеде или плаванием (очень сложно разучиться кататься на велосипеде, если это происходит не задумываясь).

Для реализации этого было предложено увеличить количество упражнений и уменьшить скорость для их выполнения, то есть сделать так, чтоб человек не тратил время на ввод ответа в упражнение, а просто воспроизводил его в своей голове, а потом проверял при необходимости. Если к тому же будут показываться чаще ошибочные упражнения, то можно добиться эффективности даже больше, чем в языковых школах и на занятиях с носителем.

### **1.1.3 Выводы**

Проанализировав вышеперечисленные проблемы, было принято решение сделать такой повтор слов, который занимает мало времени и сил пользователя, но имеет высокую эффективность, а в качестве инструмента для изучения новых слов и повторения старых использовать «карточки», что доказало их эффективность по сравнению с традиционным изучением слов на бумаге [10]. Для решения проблем в говорении, было решено сократить время ответа на упражнения и использовать карточки с интервальными повторениям.

## 1.2 Анализ приложений-аналогов

Конкурентов на рынке мобильных приложений довольно много.

Существуют десятки приложений, главными из которых являются:

- Duolingo.
- Lingualeo.
- Bussu.
- Ewa.
- Memrise.
- Simpler.
- Puzzle English.

Было рассмотрено приложение Duolingo [11]. Основной режим в нем – последовательные уроки с разными режимами сложности. Ключевой недостаток этого сервиса - отсутствие возможности повторять и добавлять свои слова. Также занятия происходят медленно, из-за того, что часто необходимо вручную вводить ответ и в новой версии появились «жизни», которые можно потерять и не иметь возможности заниматься дальше в этот день, либо необходимо платно подписаться (рисунок 2). Это сильно замедляет прогресс в изучении.

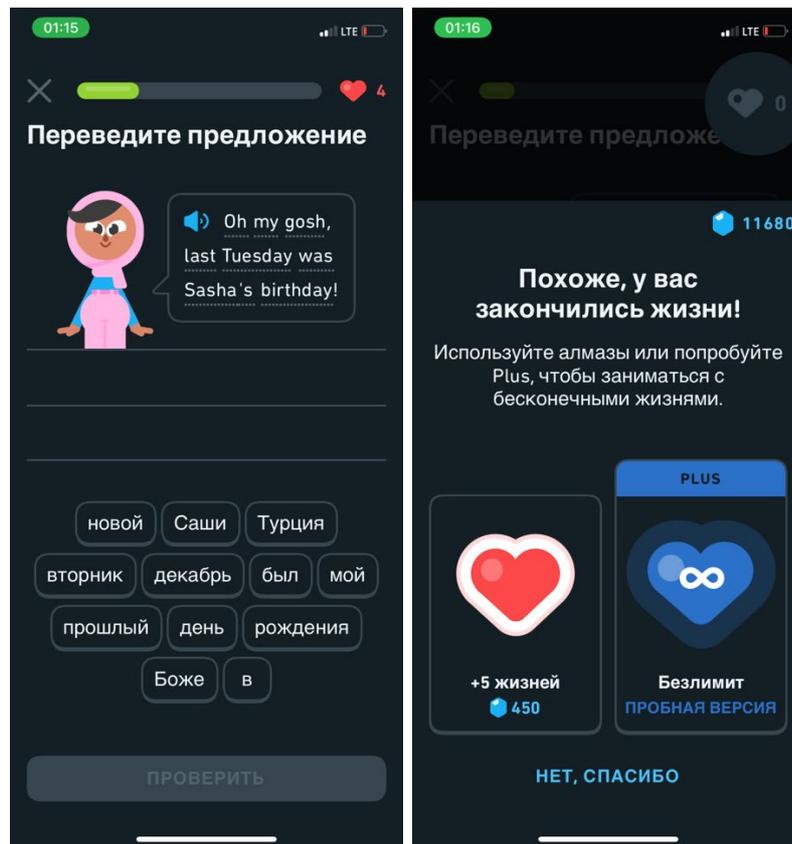


Рисунок 2 – Duolingo. Ввод ответа в упражнении и платная подписка

В Lingualeo [12] есть возможность повторять и добавлять свои слова, но реализация весьма сомнительная, так как в новой версии доступно только повторение в виде спринта, нужно нажать верно-неверно, совсем неэффективный формат, так как можно повторить только новые слова, остальные режимы по подписке (рисунок 3).

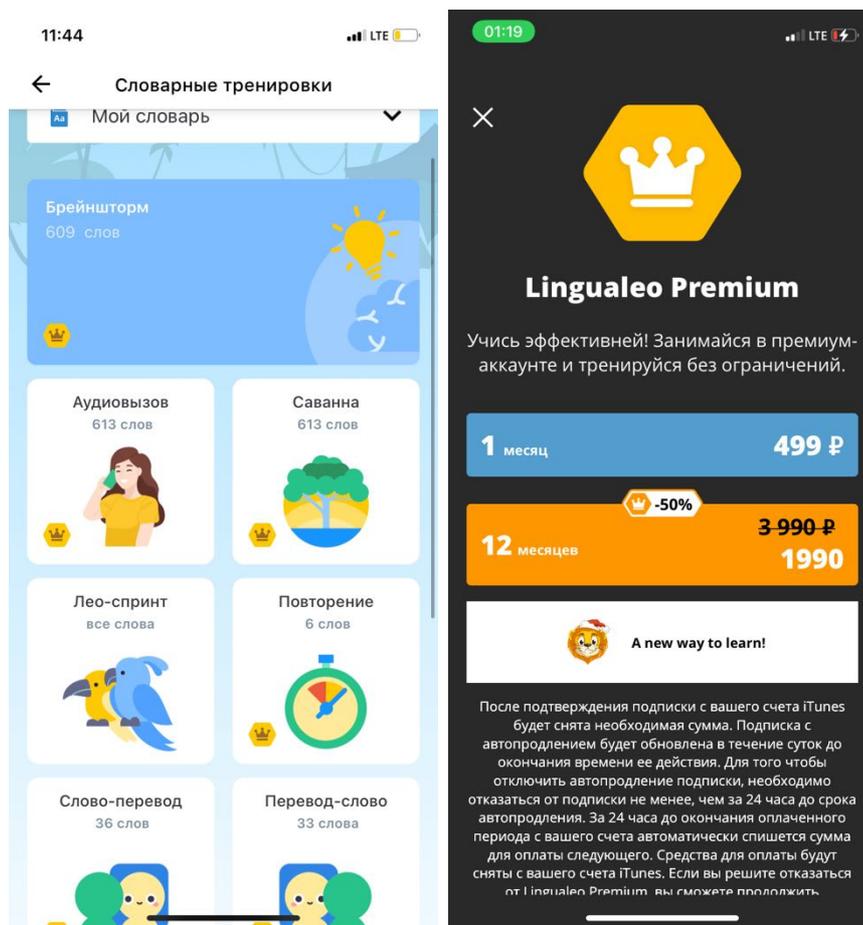


Рисунок 3 – Lingualeo. Ограничение функциональности, платная подписка

У всех остальных конкурентов также имеется недостаток в эффективности практики – процесс слишком медленный, за такое же время можно выполнять большее количество упражнений, следовательно делать это эффективнее.

Таким образом, были найдены недостатки текущих решений и для того, чтобы выделиться на фоне остальных приложений, необходимо сделать удобный, быстрый и понятный для пользователя способ изучения слов и практики, которого нет в существующих приложениях.

### 1.3 Сравнение средств разработки

На рынке смартфонов лидируют 2 операционные системы Android и iOS с долей 71.6% и 27.7% соответственно [13], поэтому разрабатывать под одну ОС или кроме основных двух нецелесообразно.

Сегодня существует множество вариантов разработать мобильное приложение на две ОС, они делятся на два типа:

1. Нативная разработка под конкретную ОС
2. Кроссплатформенная разработка сразу под две платформы

Оба варианта имеют свои плюсы и минусы.

Нативная разработка имеет следующие достоинства:

**Лучшая производительность.** Разработка под каждую из платформ позволяет добиться максимально возможной производительности, так как нет никаких лишних прослоек, которые, как правило, дают кроссплатформенные решения.

**Функциональность.** Нативная разработка открывает всю доступную функциональность под каждую ОС, самые последние функции и полный контроль над ними.

Минусы нативной разработки:

**Дорогая стоимость разработки.** Разработка под обе платформы подразумевает иметь в штате разработчиков как минимум двоих по конкретную ОС.

**Более высокая стоимость обслуживания.** Так как в штате больше людей, то платить им в сумме приходится больше, следовательно дальнейшего обслуживание ПО также дороже.

**Сложность внедрения нового функционала.** Из-за того, что новые функции разрабатываются отдельно для каждой версии приложения, возникают проблемы внедрения их в готовый продукт, так как одна из версий может сильно опаздывать, из-за проблем или особенностей платформы, и нельзя перенести готовый вариант из одной ОС на другую.

В свою очередь, кроссплатформенная разработка обладает следующими преимуществами:

**Дешевизна разработки.** При том же количестве разработчиком, как и при нативной разработке, сроки реализации будут меньше, либо можно уменьшить штат и в таком случае расходы будут меньше.

**Скорость внедрения нового функционала.** Из-за одной кодовой базы можно легко поставлять новый функционал сразу в две версии приложения.

Недостатки кроссплатформенной разработки:

**Потеря производительности.** Такие приложения нередко имеют проблемы с производительностью, либо не полное раскрытие потенциала производительности устройства и ОС, в виду лишней обертки над стандартной реализацией или собственной.

**Функциональность.** В отличие от нативной разработки, кроссплатформенная часто не предоставляет доступ к последним нововведениям конкретной ОС, либо такая возможность открывается только посредством прямого обращения к коду отдельной платформы, что нивелирует плюсы кроссплатформенной разработки.

В связи с малым бюджетом и одним разработчиком в проекте, выбор остановился на кроссплатформенной разработке, остается только решить какой из множества фреймворков использовать.

Рассмотрим три самых популярных фреймворка для кроссплатформенной разработки, а именно: React Native, Flutter и Xamarin.

Если взять статистику популярности запросов в поисковой системе Google по всему миру за последние 5 лет, то можно заметить, что Xamarin становится все менее популярным, значит и востребованным на рынке, следовательно его можно исключить из потенциальных кандидатов на рассмотрение (рисунок 4) [14].

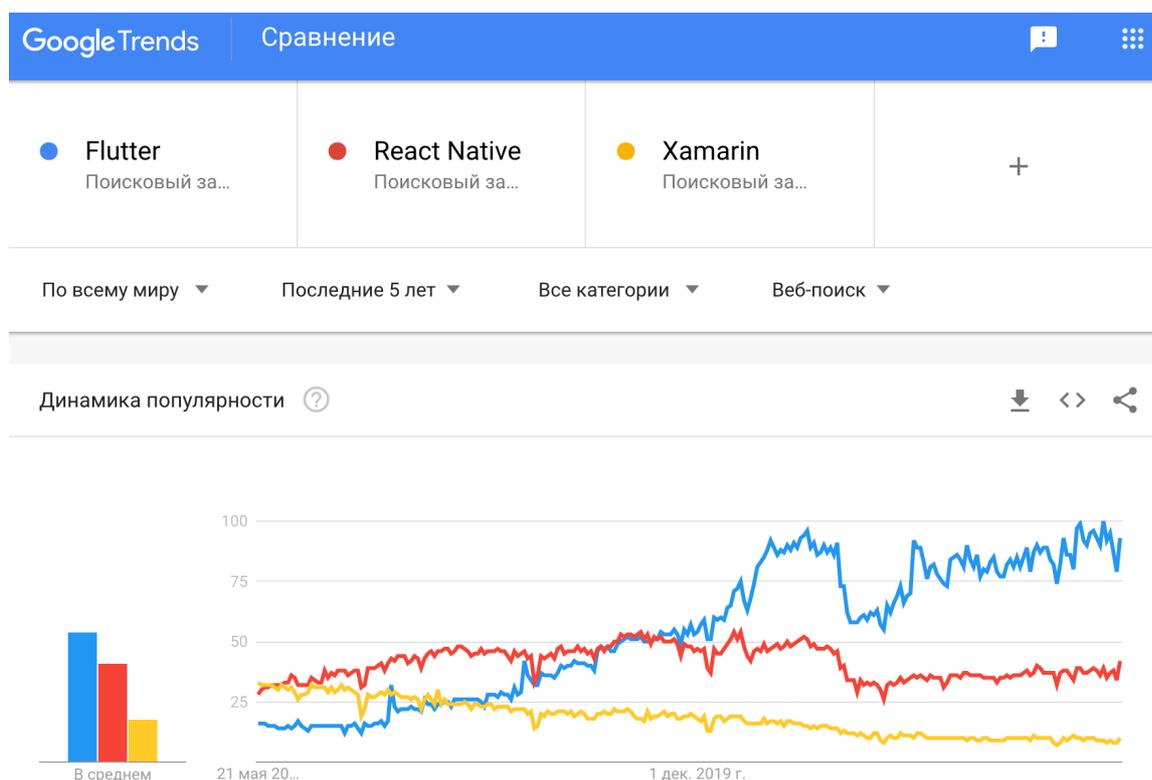


Рисунок 4 – Сравнение поисковых запросов кроссплатформенных фреймворков

Будем сравнивать два востребованных фреймворка Flutter и React Native.

**Flutter** разрабатываемый компанией Google был представлен в 2017 году с открытым исходным кодом, использует язык программирования Dart, позволяет создавать приложения с единой кодовой базой для мобильных платформ, Web и десктопа.

**React Native** был впервые выпущен Facebook в качестве проекта с открытым исходным кодом в 2015 году и построен на основе React - библиотеки JavaScript. Фреймворк позволяет создавать приложения для различных платформ, используя один исходный код.

#### **Наличие сторонних библиотек.**

Оба фреймворка обладают обширной библиотекой сторонних компонентов, но React Native в этом плане превосходит оппонента, из-за молодости второго (стабильная версия Flutter была выпущена в декабре 2018 года).

### **Производительность.**

React Native использует родные виджеты платформы и передает события через JavaScript. Это влияет на производительность уровня представления, однако 60 кадров в секунду все еще достижимы, хотя производительность зависит от версии ОС и самого устройства.

Flutter, с другой стороны, отображает весь интерфейс, используя собственный 2D-движок Skia, избегая какого-либо специального соединения между виджетами и другим кодом. Это делает рендеринг невероятно быстрым.

Нет однозначного ответа что лучше, оба варианта имеют свои плюсы и минусы, но из-за более высокой популярности и производительности, было решено остановиться на решении от компании Google – Flutter [15].

#### **1.4 Принцип построения интерфейса в Flutter**

Flutter спроектирован как расширяемая многоуровневая система. Он существует как серия отдельных библиотек, каждая из которых зависит от нижележащего уровня, а на вершине иерархии находится фреймворк Dart с самыми используемыми библиотеками Widgets и наследуемые от нее Material и Cupertino (рисунок 5). Именно с ними чаще всего приходится взаимодействовать разработчику.

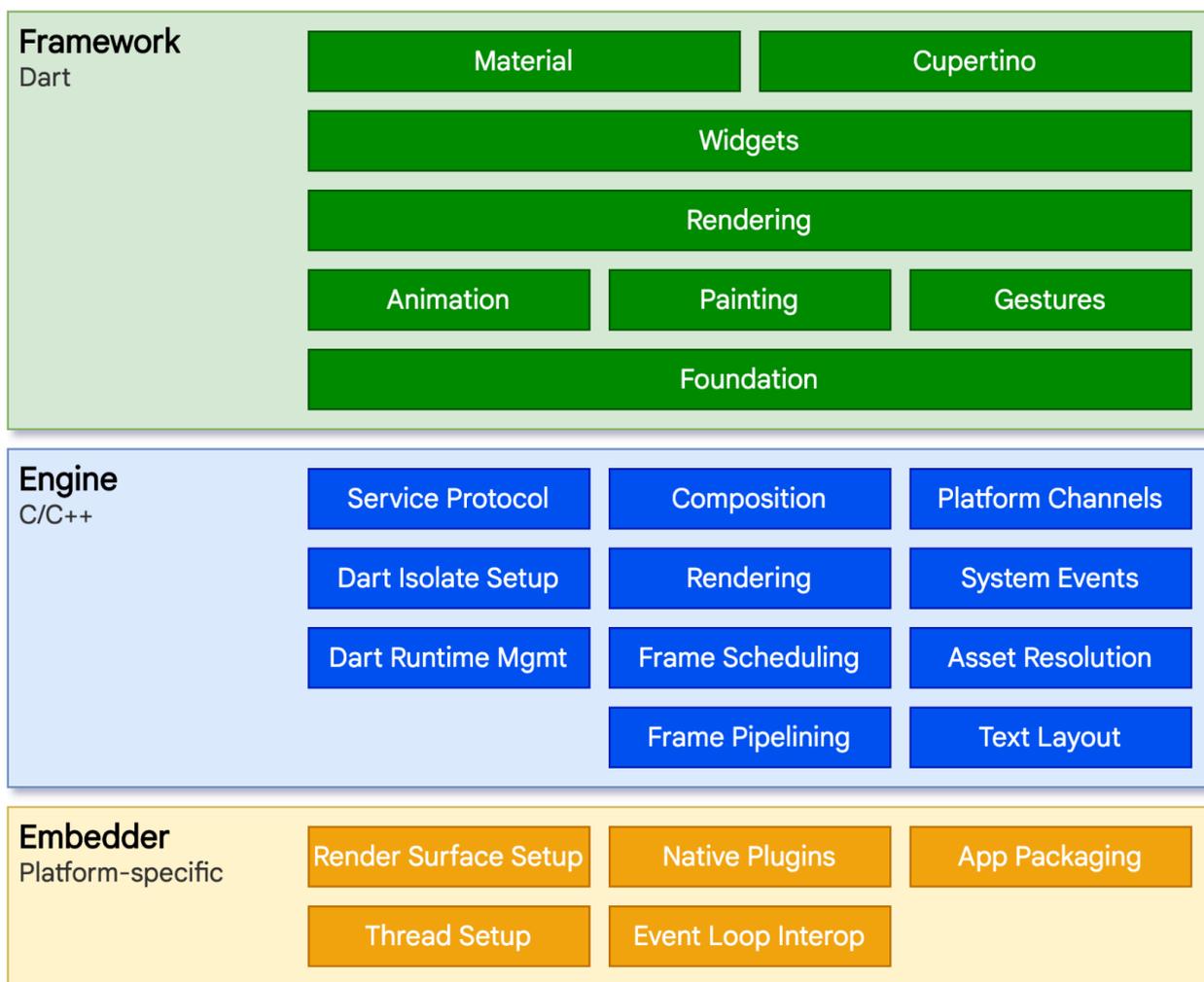


Рисунок 5 – Архитектура Flutter

В большинстве традиционных UI-фреймворков начальное состояние пользовательского интерфейса описывается один раз, а затем отдельно обновляется пользовательским кодом во время выполнения в ответ на события. Одна из проблем этого подхода заключается в том, что по мере усложнения приложения разработчик должен знать, как связаны меняются состояния во всем пользовательском интерфейсе.

Flutter, наряду с другими реактивными фреймворками, использует альтернативный подход к этой проблеме, явно отделяя пользовательский интерфейс от его основного состояния.

Виджеты из библиотеки Widgets представляют собой строительные блоки пользовательского интерфейса, все что пользователь видит на экране – это виджеты. Библиотеки Material и Cupertino делают из виджетов готовые компоненты похожие на стандартные элементы пользовательского

интерфейса ОС Android и iOS соответственно, такие как кнопки, текстовые поля, всплывающие сообщения и т.д.

У каждого виджета есть метод `build`, который возвращает дерево элементов для построения интерфейса и вызывается в каждом визуализируемом кадре воссоздавая только те части пользовательского интерфейса, в которых изменилось состояние.

Существуют виджеты с отслеживанием состояния (`stateful`) и без (`stateless`), например переключатель может иметь 2 состояния – включен или выключен, это будет `stateful` виджет. Обычный текст – это `stateless` виджет `Text`. Менять состояние виджета нужно в методе `setState`, который в свою очередь вызовет метод `build` и перестроит дерево виджетов.

Несмотря на относительно наивный подход, он весьма эффективный и позволяет создавать высокопроизводительные интерактивные приложения, к тому же упрощает написание кода, сосредоточив внимание на объявлении того, из чего состоит виджет, а не на сложности перехода пользовательского интерфейса из одного состояния в другое.

### **1.5 Выбор серверной части**

Разработка приложения для изучения иностранных языков подразумевает хранить данные пользователей, теорию и упражнения где-то удаленно и получать к ним доступ посредством интернета. За это отвечает серверная часть, которая безусловно занимает важную часть в проекте. Так как создать и настроить свой сервер весьма трудоемко и дорогостояще, особенно когда в команде один человек, было принято решение воспользоваться готовым решением от компании Google – Firebase.

Firebase – это платформа для разработки мобильных приложений, в которой есть самые различные современные функции и возможности такие как базы данных, аутентификацию, push-уведомления, аналитику, хранение файлов и многое другое [16].

Для того, чтобы начать пользоваться данным сервисом не надо ничего платить, Firebase предлагает бесплатный начальный план, который

называется Spark, он имеет множество функций, которых часто хватает, чтобы начать работать. При увеличении требований к проекту, можно сменить тарифный план. Бесплатный план предлагает безграничные возможности авторизации через социальные сети и почту, а также 1 Гб хранилища в базе данных и 5 Гб для хранения файлов.

Как уже было сказано выше, Firebase также решает проблему авторизации и регистрации, что весьма трудоемко сделать самостоятельно (безопасное хранение паролей, сброс пароля, подтверждение почты, авторизация по номеру телефона и т.д.).

### **1.6 Выбор базы данных**

Firebase предоставляет выбор из двух вариантов NoSQL баз данных: Realtime Database и Cloud Firestore.

**Realtime Database** – это старая и оригинальная база данных, которая предоставит решение с низкой задержкой для мобильных приложений. Она мгновенно синхронизирует данные между клиентами и хранит данные в формате JSON.

**Cloud Firestore** – это новая база данных Firebase, которая используется для создания мобильных приложений. Она основана на успехах Realtime Database с новой, более интуитивно понятной моделью данных, она также имеет лучшую масштабируемость. Кроме того, Cloud Firestore обладает более быстрыми запросами с более богатыми функциями. Данные хранятся в виде набора документов. Небольшие данные легко хранить в документах, которые очень похожи на JSON, с другой стороны, сложные данные, основанные на иерархии, очень легко организовать в масштабе, можно использовать вложенную коллекцию в документах, также Cloud Firestore требует меньшей нормализации.

Так как в проекте не нужно мгновенно синхронизировать данные и нужно структурированно хранить информацию, то выбор однозначно за Cloud Firestore [17].

Для загрузки файлов с теорией и изображений будет использоваться хранилище Firebase Cloud Storage с бесплатным хранилищем 5 Гб.

### **1.7 Выбор API получения перевода**

В качестве источника перевода, рассматривались 2 крупных словаря АBBYU Lingvo и Яндекс.Словарь.

АBBYU Lingvo имеет бесплатный API (до 50 000 символов в день) [18], тогда как у Яндекс.Словаря есть ограничение на количество обращений в размере 10 000 обращений в сутки. Из-за более понятной документации и JSON ответов было принято решение в пользу Яндекс.Словаря [19].

Пользовательское соглашение сервиса «API Яндекс.Словарь» [20] обязует отображать текст «Реализовано с помощью сервиса «Яндекс.Словарь» с активной гиперссылкой на страницу <https://tech.yandex.ru/dictionary>. Размер шрифта ссылки при этом должен быть не менее размера шрифта основного текста, а цвет должен совпадать с цветом шрифта основного текста страницы.

### **1.8 Средства разработки**

После анализа предметной области было принято решение во время проектирования и разработки проекта использовать следующие средства и инструменты:

- Adobe XD – используется для проектирования пользовательского интерфейса, основан на векторной графике.
- Android Studio – интегрированная среда разработки для работы с платформой Android, а также для проектов на Flutter. Является основным инструментом для разработки данного проекта.
- Xcode – интегрированная среда разработки программного обеспечения для платформ iOS, используется для тестирования проекта на симуляторе и реальном устройстве.
- MySQL Workbench – инструмент для визуального проектирования баз данных, интегрирующий проектирование, моделирование, создание и эксплуатацию БД в единое бесшовное окружение для системы баз данных

MySQL. Будет использоваться для проектирования и наглядного представления сущностей базы данных.

– Postman – приложение для работы с API, будет использоваться для проверки ответов сервиса Яндекс.Словарь и RSS каналов.

Разработка и проектирование проекта будет производиться под управлением операционной системы macOS Big Sur 11.5.2.

## 2 ПРОЕКТИРОВАНИЕ

### 2.1 Разработка UI и способа взаимодействия в приложении

Интерфейс и способ взаимодействия в приложении должен быть интуитивно понятным и простым, как это и принято для современных приложений.

На данном этапе не будут рассматриваться все мельчайшие детали интерфейса такие как цвета и шрифты, все возможные состояния экранов, это прежде всего прототип для дальнейшей реализации.

#### 2.1.1 Основная навигация

Основная навигация выполнена в виде нижних вкладок с тремя главными разделами: Словарь, Занятия и Чтение (рисунок 5).

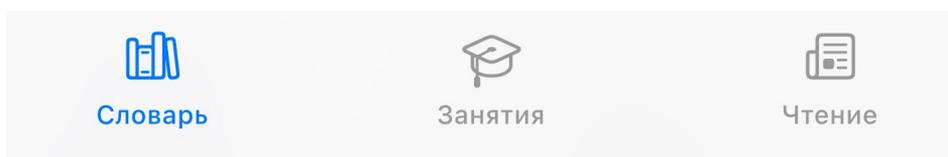


Рисунок 6 – Основная навигация в виде нижних вкладок

#### 2.1.2 Словарь

На первом экране отображаются добавленные слова. Вверху можно зайти в настройки приложения, изменить язык изучения, поменять сортировку, выбрать слова для дальнейших действий и искать их с помощью поиска (рисунок 6). Также можно изменить что будет показываться первым, текст на иностранном или на родном языке путем нажатия на En⇌Ru, поменять все слова на свои наборы, готовые наборы (заранее созданные) или на сложные слова, в которых чаще всего пользователь допускает ошибки (рисунки 7-8).

Во всем приложении отображается статус в виде цвета: красный – нужно выучить, оранжевый – желательно повторить, зеленый – можно не повторять. Есть особый статус «New» у слов и наборов, когда они еще ни разу не были изучены. В наборы можно сохранять выбранные слова и в разных наборах могут быть одни и те же слова, что редко можно увидеть в других приложениях, этим можно получить конкурентное преимущество (рисунок 9).

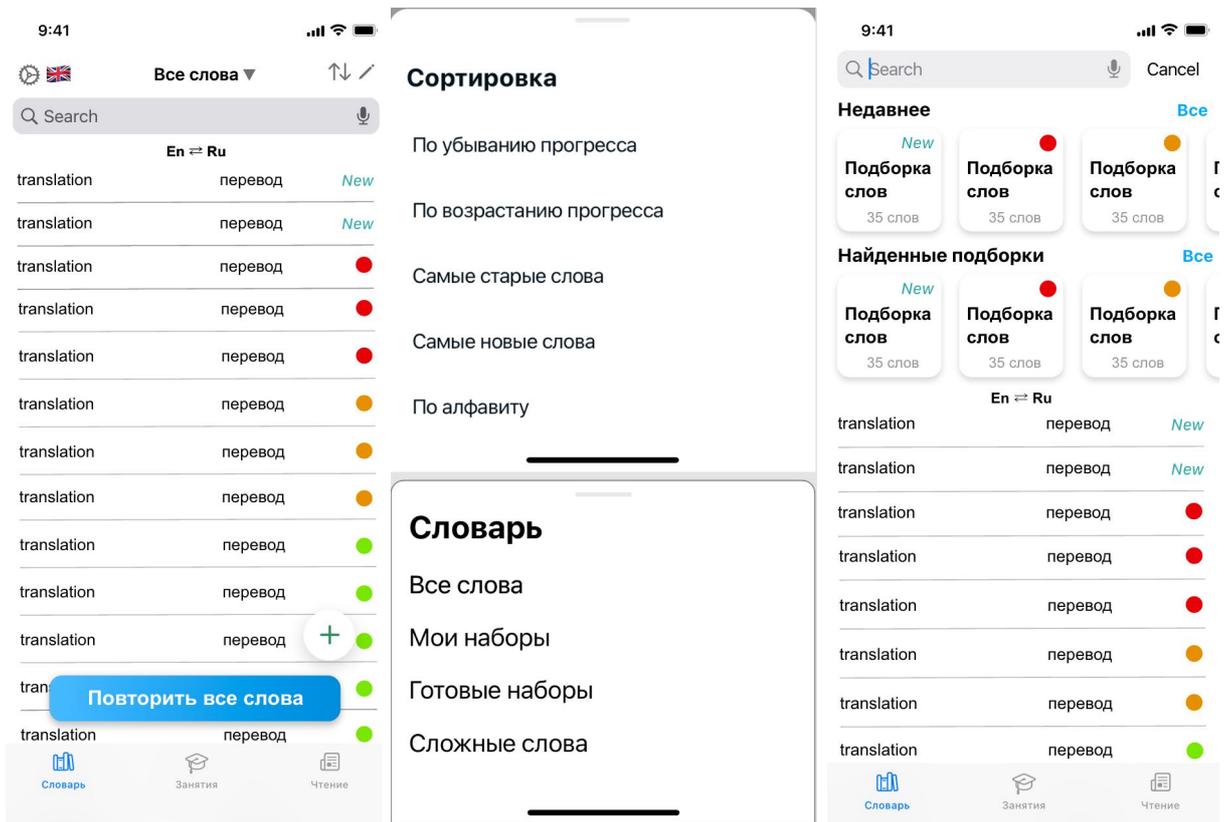


Рисунок 7 – Словарь. Выбор режима и сортировки. Поиск

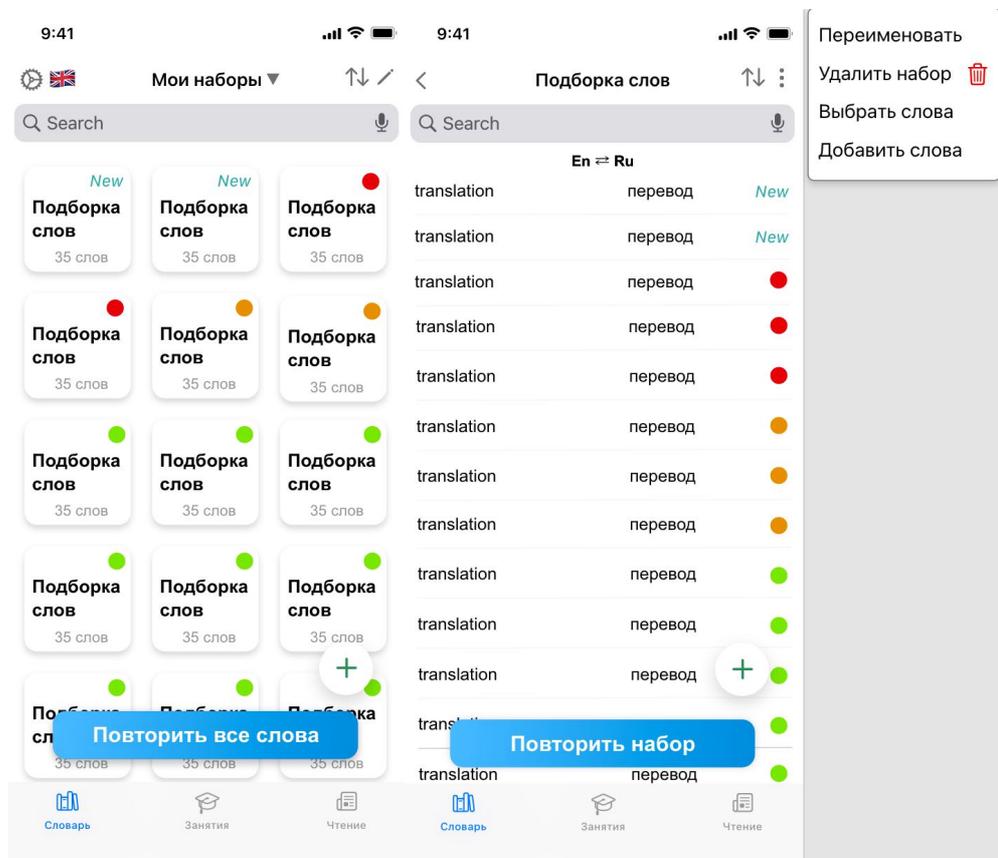


Рисунок 8 – Подборки слов. Контекстное меню для набора

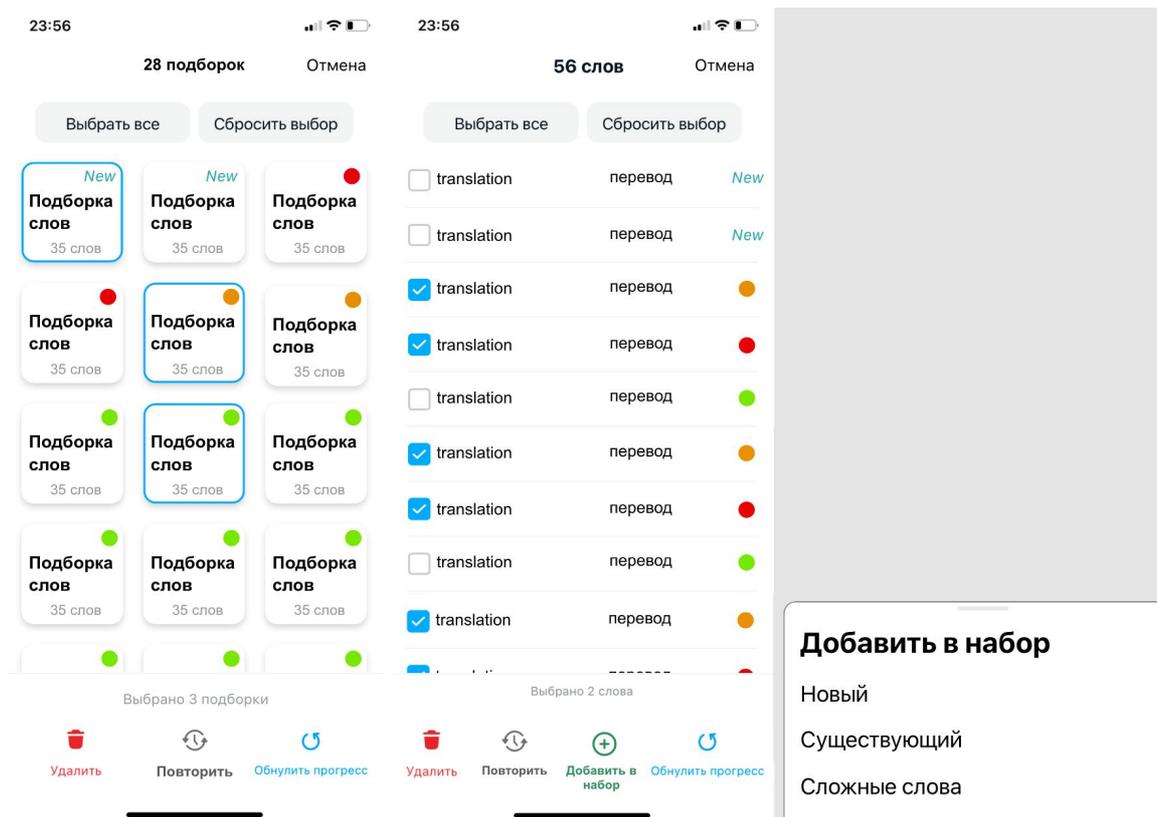


Рисунок 9 – Режим выбора слов и наборов слов. Варианты добавления в набор.

Кнопка плюс позволяет добавить новое слово в словарь. При этом можно добавить транскрипцию, выбрать часть речи, дописать заметки, также будут добавлению слов взятых со стороннего API, будут добавляться примеры в заметки (рисунок 10). Данную информацию о слове можно изменить в любой момент.

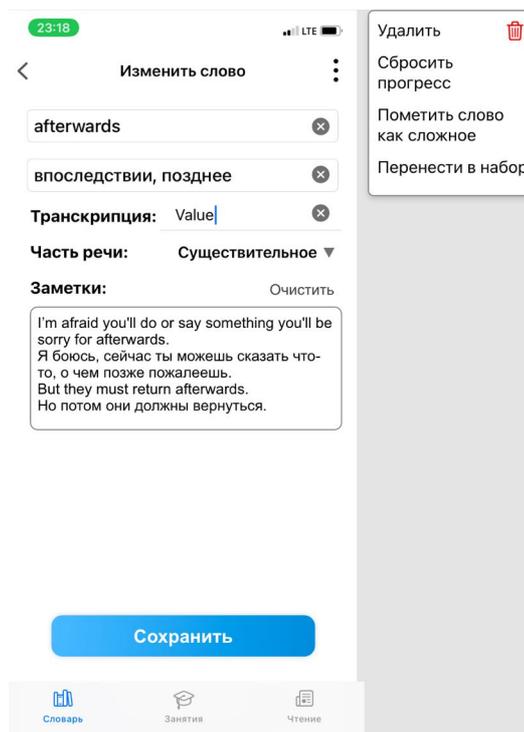


Рисунок 10 – Добавление и редактирование слов

Для того, чтобы реализовать метода интервальных повторений нужно сделать флеш-карточки. Сами карточки выглядят как набор реальных карт, с иностранным словом на лицевой стороне и переводом на обратной (можно изменить порядок) (рисунок 11).

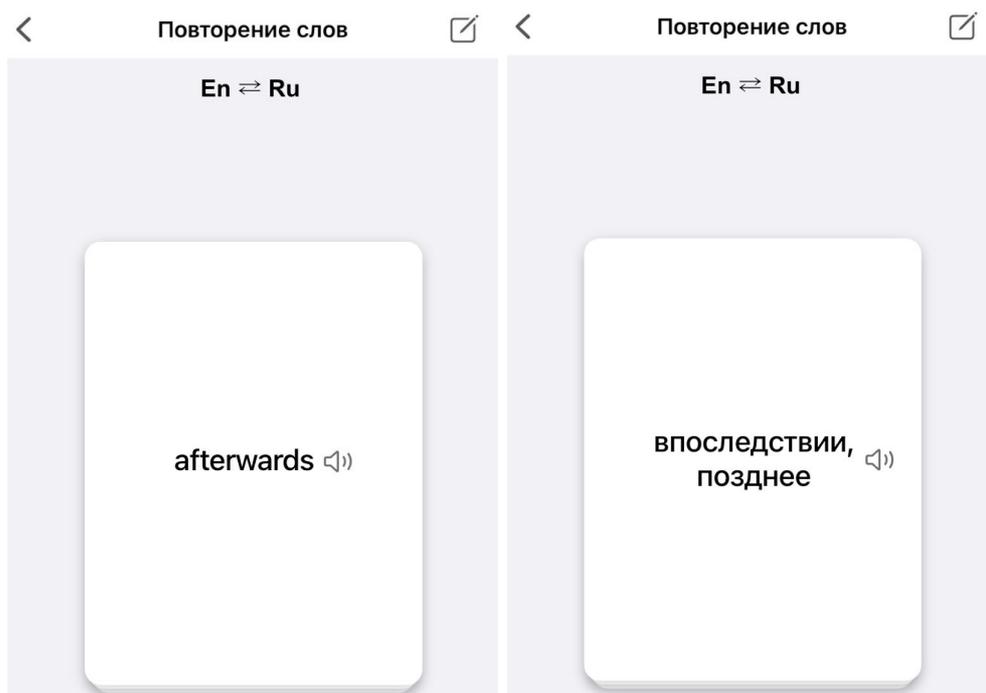


Рисунок 11 – Набор карточек, лицевая и обратная сторона

Пользователь может смахнуть (провести пальцем по экрану) карточку вправо или влево, если он помнит слово или забыл соответственно. Также во время повторений слов можно изменить карточку, нажав в правый верхний угол (рисунок 11).

При нажатии на карточку, срабатывает анимация переворота и отображается перевод слова, если нажать еще раз, то карточка перевернется еще раз и покажется лицевая сторона.

После повторения слов, пользователю отображается мотивирующее сообщение и кнопка с надписью «Повторить еще раз» (рисунок 12).

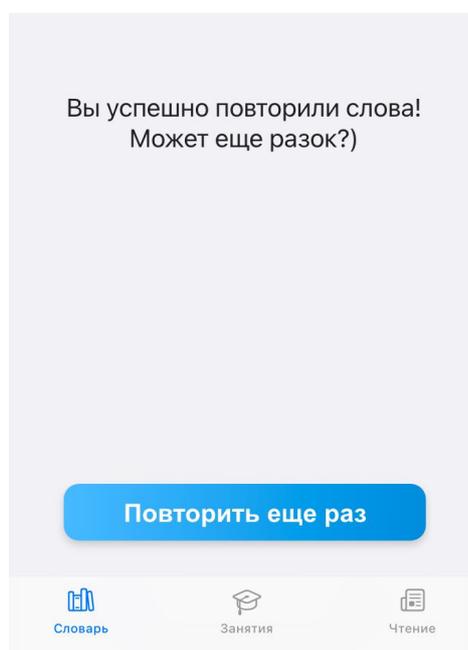


Рисунок 12 – Интерфейс после повторения слов

### 2.1.3 Занятия

Во втором разделе отображаются занятия, а именно последовательные уроки и подборки уроков. В первом случае занятия структурированы по уровню владения языком (можно скрывать блоки, выбирать определенные занятия и искать по названию) (рисунок 13). В подборках отображаются занятия (группа занятий) по разделам: для тебя, для начала, продолжающий и т.д. (рисунок 15). У некоторых занятий есть уровень сложности, который можно выбирать в момент выделения нескольких уроков и менять в процессе изучения (рисунок 14).

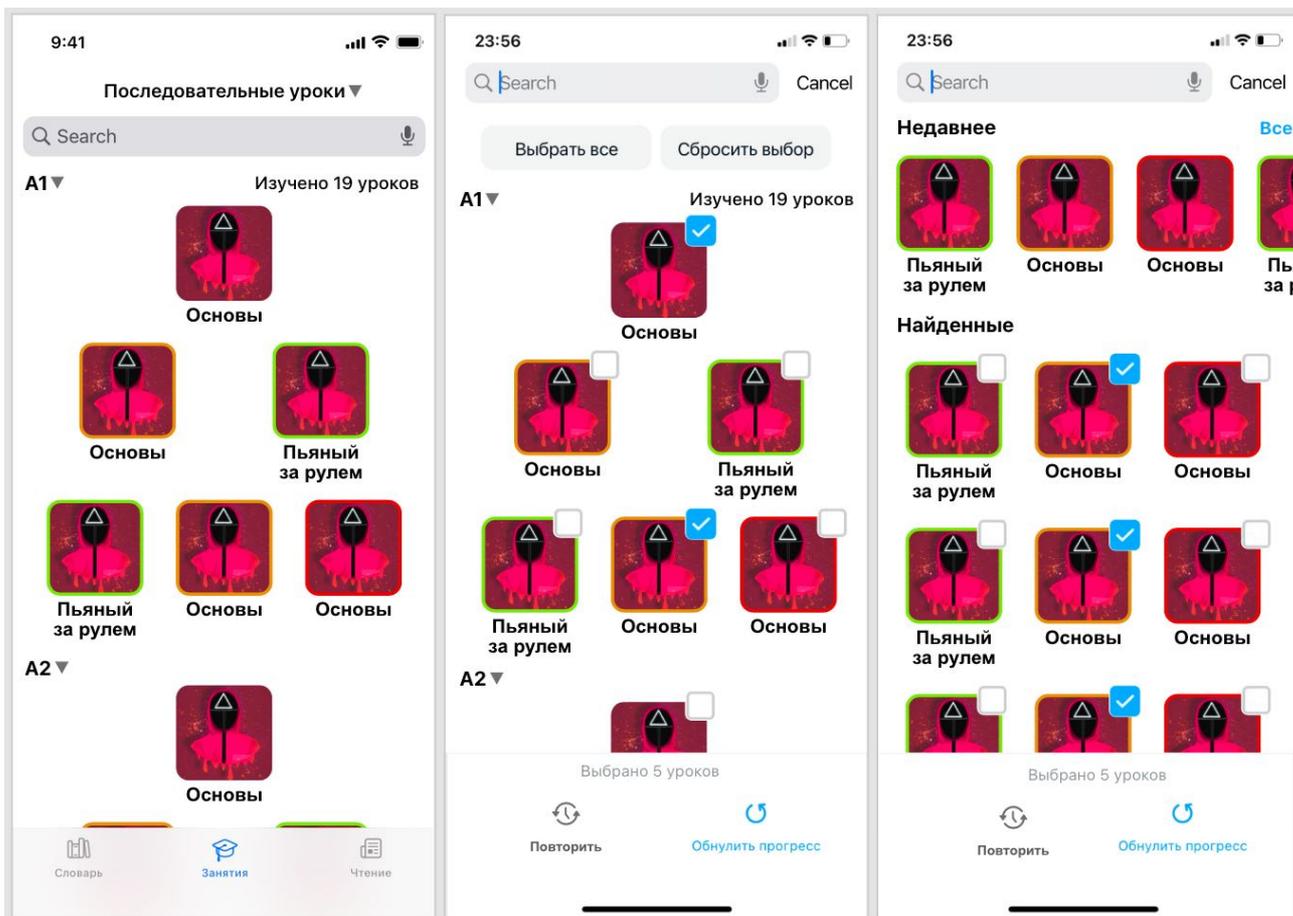


Рисунок 13 – Последовательные уроки. Выбор нескольких уроков.

### Поиск уроков



Рисунок 14 – Выбор занятий, сложности



Рисунок 15 – Подборка занятий

Процесс изучения урока прост и понятен. Пользователю показывается теория, в ней можно нажимать на слова и смотреть их перевод (рисунок 20), в конце которой предлагается посмотреть список необходимых слов, либо сразу приступить к уроку (рисунок 16).

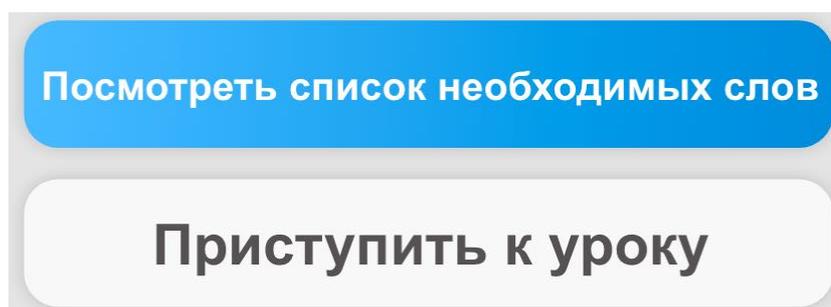


Рисунок 16 – Конец просмотра теории к занятию

При просмотре слов можно выбрать их для добавления в словарь и повторения (рисунок 17).

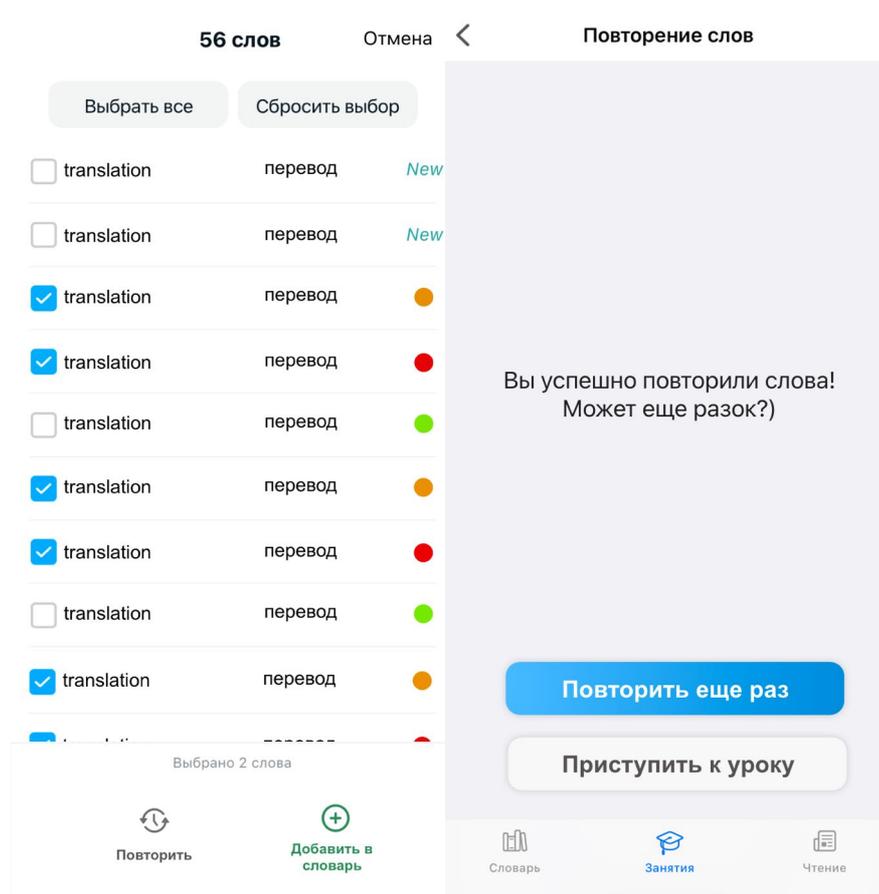


Рисунок 17 – Интерфейс выбора слов и после их повторения

Непосредственный процесс отработки практики выглядит почти также, как и при изучении слов. Можно сменить направление языка, уровень урока и посмотреть теорию. Также у каждого примера есть кнопка вопроса в правом верхнем углу, которая открывает объяснение для каждого приложения (рисунок 18). По нажатию на надпись «Как этим пользоваться?», должна всплывать информация о том, что нужно смахивать карточки максимально быстро и не проверять ответ, если в нем уверен.

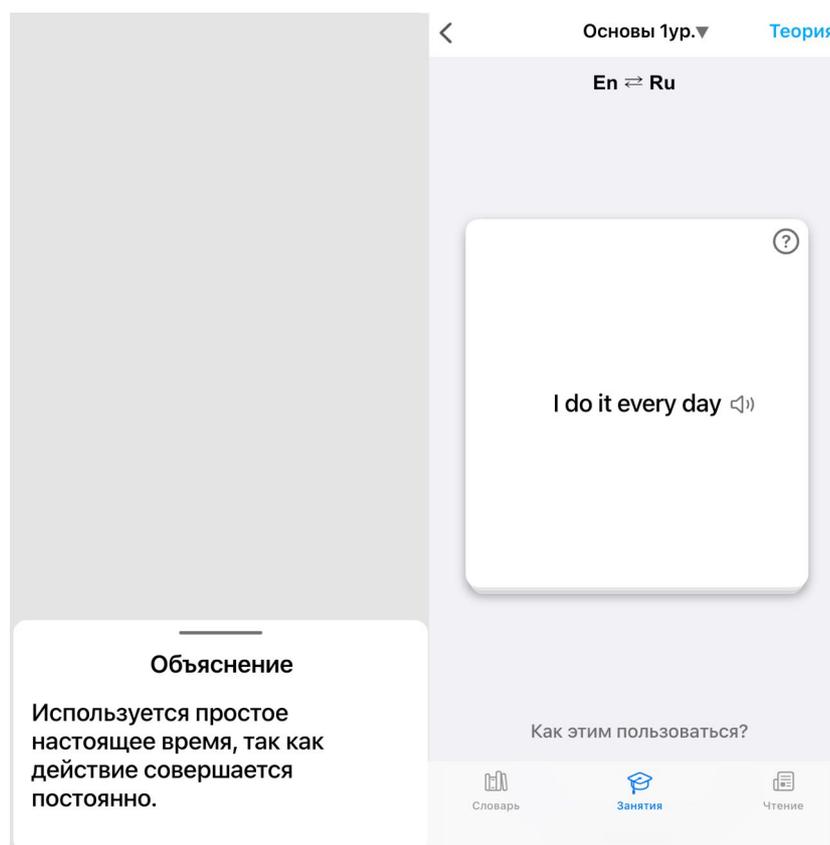


Рисунок 18 – Интерфейс изучения урока и объяснения к примеру

#### 2.1.4 Чтение

В третьем разделе отображаются новости/статьи для пополнения словарного запаса (рисунок 19). Новости будут браться с общедоступных RSS каналов.

Их можно искать, менять темы и сортировать различными способами. При чтении можно легко просматривать перевод новых слов (путем нажатия на них) и добавлять их в словарь. Зеленым цветом указан добавленный перевод, по нажатию на синий текст можно перейти на экран изменения слова, в котором автоматически подставится нужная часть речи, транскрипция, в заметках будет написаны примеры использования (рисунок 20).

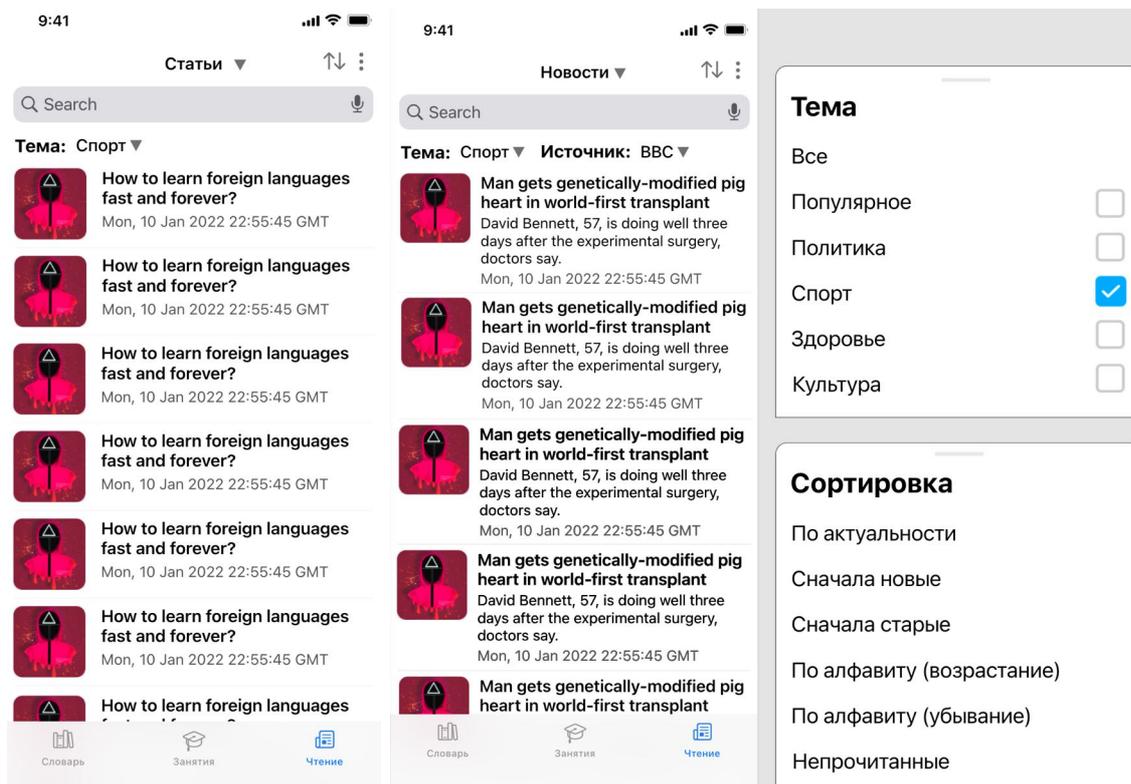


Рисунок 19 – Интерфейс просмотра списка новостей/статей

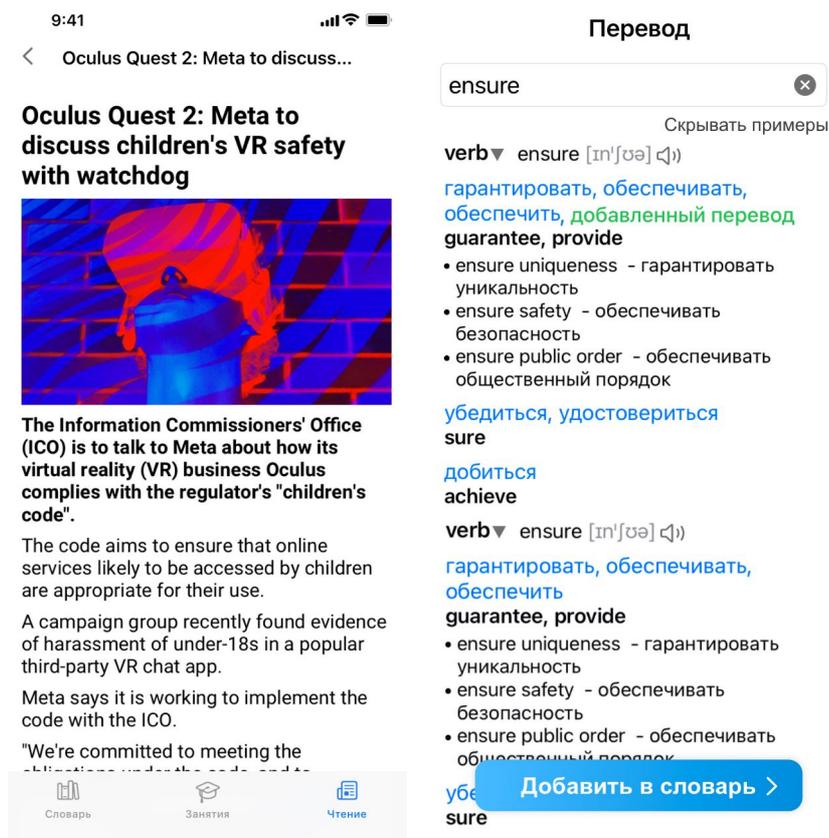


Рисунок 20 – Интерфейс теории/статьи/новости. Перевод слов

## 2.2 Алгоритм повтора слов и упражнений

Для реализации метода интервальных повторений определены следующие параметры для хранения:

`date_created (date_first_repetition)` — дата создания (первого повторения в случае с упражнениями), с ее помощью карты, которые являются обязательными для повторения, будут сортироваться в правильном порядке.

`repeat_interval` – хранит информацию о количестве минут, необходимые для расчета интервала повторения (по умолчанию 360);

`date_next_repetition` – дата следующего повторения, служит для учета карт, которые нужно обязательно повторить;

`date_last_repetition` — дата последнего повторения, с помощью ее можно отсортировать карты в правильном порядке, которые не являются обязательными для повторения.

Для учета сложных слов и упражнений были определены дополнительные параметры: `number_of_mistakes` и `is_difficult`. Первый означает количество ошибок, а второй является ли это слово/упражнение сложным. Слово/упражнение автоматически становится сложным, если количество ошибок больше 15 или пользователь сам изменил этот параметр через контекстное меню. Такие слова и упражнения можно будет посмотреть отдельно от простых, и они будут показываться в два раза чаще обычных.

Сам алгоритм выглядит следующим образом: пользователю показываются 10 карт для повторения, сначала берутся новые карты обязательные для повтора, это те, у которых дата следующего повторения меньше текущей, они сортируются по дате создания (первого просмотра) по убыванию (самые свежие карточки будут показываться в первую очередь), потом берутся необязательные для повтора карточки и сортируются по дате последнего повтора по возрастанию (те карточки, которые давно не повторяли будут показываться первыми).

Когда пользователь смахивает карточку влево – прогресс сбрасывается, и карта помещается назад в конец колоды, если вправо – для обычных карт

дата следующего повторения будет равна текущей дате плюс  $repeat\_interval * 2$ , для сложных карт интервал повторения будет умножаться на 1.5, а не на 2.

Пример работы алгоритма повтора слов можно увидеть на рисунке 21.

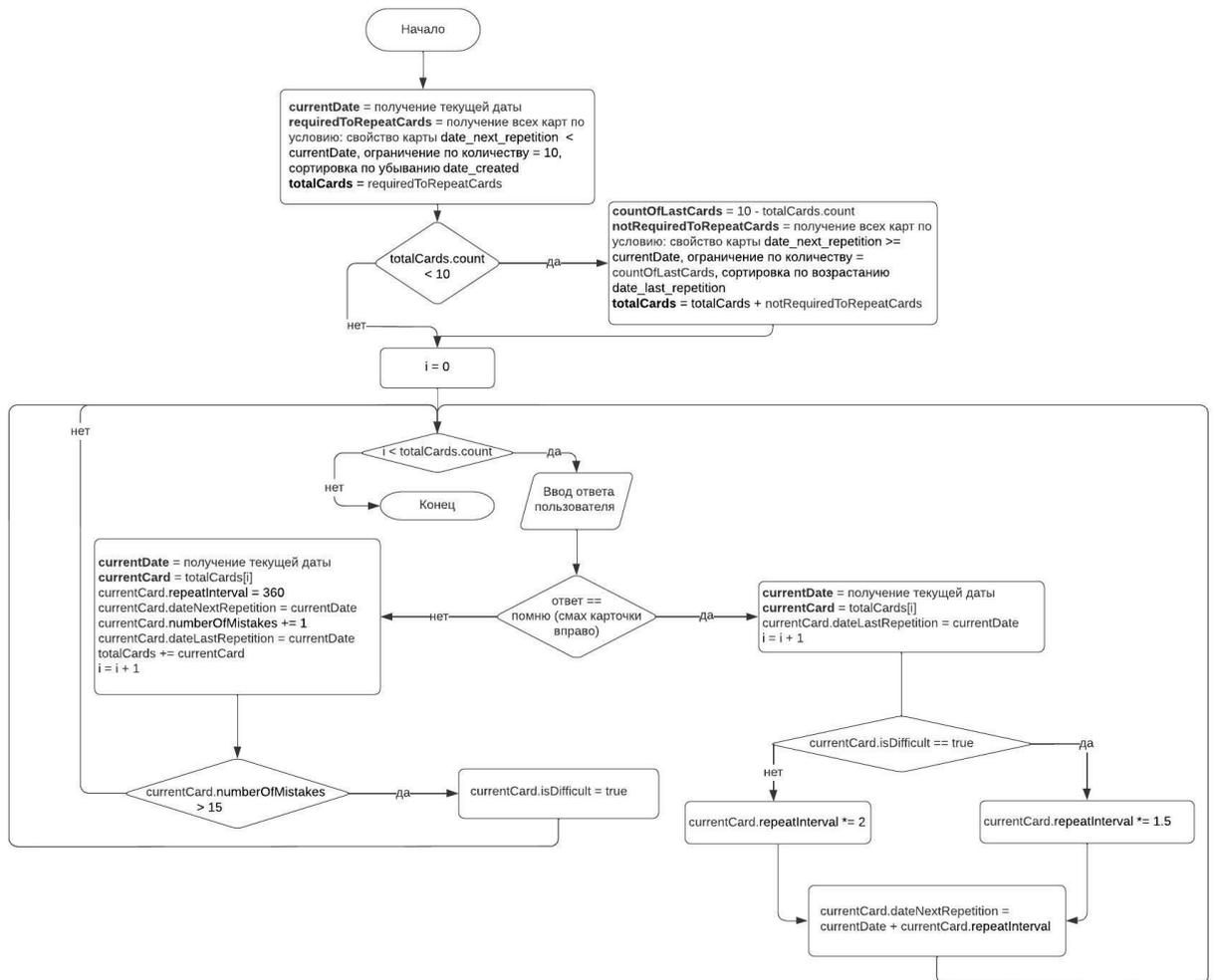


Рисунок 21 - Алгоритм повтора слов

### 2.3 Проектирование физической схемы базы данных

При проектировании не учитывались данные пользователей, так как на первом этапе будет сохраняться только их id, без дополнительных сведений.

Таблицы **part\_of\_speech** хранит различные возможные части речи, такие как существительное, глагол, прилагательное и так далее. Таблица **languages** хранит возможные языки их название в трех формах, полное название (Russian), код языка по стандарту ISO 639-1 (ru) и ISO 639-3 (rus). Текст теорий/объяснений/статей будет храниться в HTML формате.

Одна из главных таблиц **cards** хранит все карточки, созданные кем-либо, и содержит `id` части речи, `id` языка, иностранный текст и перевод, транскрипцию, заметку, а также опциональное поле `created_user_id`, которое содержит пользователя, создавшего карточку, если оно пустое, то эта карточка не имеет конкретного владельца, используется для готовых наборов и слов для изучения к уроку. Таблица **user\_cards** хранит пользовательские карточки слов, их дату создания и изменения, а также данные для учета прогресса (количество ошибок, интервал повторения, дату следующего повтора, дату последнего повторения).

Для наборов карточек есть 2 таблицы: **cards\_in\_groups** и **cards\_group**. Первая хранит карты в группе карт, а вторая – название, язык изучения и опционально имя пользователя (если пустое поле, то это готовый набор или набор слов для изучения к уроку).

Для занятий созданы 12 таблиц, две из которых по аналогии с наборами карточек: **lessons\_in\_group** и **lessons\_groups**. Таблица **lessons** хранит сами занятия, включающие в себя уровень сложности занятия (`lesson_level_id`), уровень сложности владения иностранным языком (`language_level_id`), `id` теории и опционально `id` группы рекомендованных слов. Уровень владения языком хранится в таблице **language\_levels** от 0 до 5 с названием от A1 до C2.

Теория хранится в таблице **theories**, в которой есть только `id`, на него ссылается таблица **localized\_theory**, которая содержит в себе язык теории (для каждого учащегося он свой) и ссылку на HTML файл в хранилище файлов Firebase, таким образом можно добиться локализованной и адаптированной теории к уроку для конкретного родного языка пользователя.

Сами упражнения хранятся в таблицах **lessons\_examples** и **examples**, первая содержит номера упражнений к занятию, а вторая сами примеры с иностранным текстом и `id` объяснения к примеру (таблица **explanations**). Упражнения и их объяснения также локализованы, за это отвечают таблицы **localized\_examples** и **localized\_explanations** соответственно. По аналогии с `user_cards` существует `user_examples`.

Результат проектирования представлен на рисунке 22.

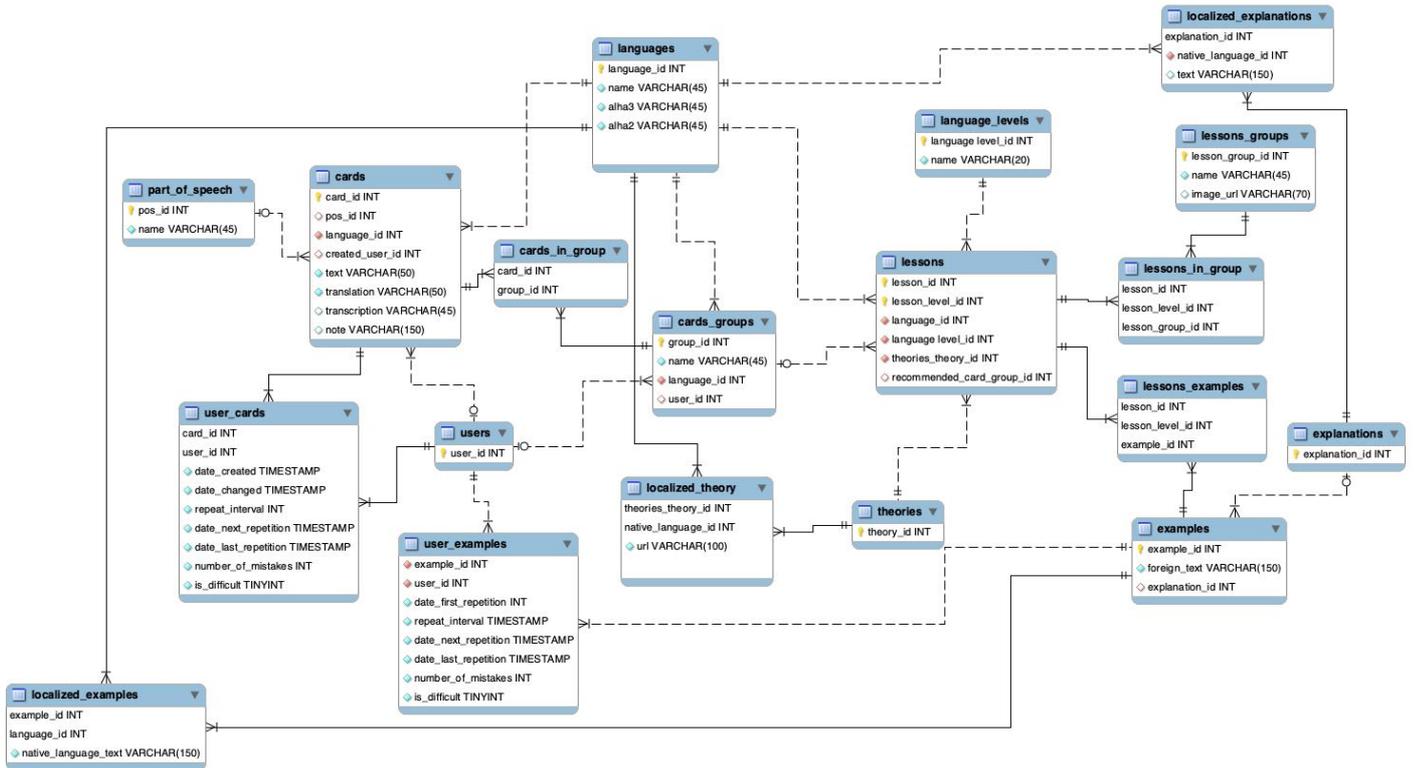


Рисунок 22 – Физическая схема базы данных

## **3 РЕАЛИЗАЦИЯ МОБИЛЬНОГО ПРИЛОЖЕНИЯ**

### **3.1 Реализация аутентификации**

Чтобы реализовать авторизацию и регистрацию в приложение, потребовалось создать 2 экрана для каждого действия, подключить проект к Firebase, предварительно настроив его и наладить взаимодействие.

### **3.2 Создание экранов авторизации и регистрации**

Все экраны начинались с виджета Scaffold, он представляет собой класс, реализующий базовую структуру визуального макета Material Design. В нем находится виджет SafeArea, обеспечивающий расположение дочерних элементов в зоне, не перекрывающей элементы навигации и верхнего информационного бара, дальше располагается ListView, который позволяет расположить дочерние элементы вертикально (по умолчанию) или горизонтально. В нем находятся различные виджеты с названиями, кнопки и текстовые поля.

На рисунке 23 приведен пример кода верстки экрана авторизации на Flutter, а на рисунке 24 представлены сами экраны авторизации и регистрации.

```

24 Widget build(BuildContext context) {
25   const title = Text(...); // Text
32   const subTitle = Text(...); // Text
39   final email = TextFormField(...); // TextFormField
52   final password = TextFormField(...); // TextFormField
65   final loginButton = MaterialButton(...); // MaterialButton
85   final googleLoginButton = MaterialButton(...); // MaterialButton
106  final forgotLabel = TextButton(...); // TextButton
113  final newUserLabel = TextButton(...); // TextButton
123
124  return Scaffold(
125    backgroundColor: Colors.white,
126    body: SafeArea(
127      child: ListView(
128        shrinkWrap: true,
129        padding: const EdgeInsets.only(left: 24.0, right: 24.0),
130        children: <Widget>[
131          SizedBox(height: MediaQuery.of(context).size.height*0.03),
132          title,
133          const SizedBox(height: 8),
134          subTitle,
135          const SizedBox(height: 48,),
136          email,
137          const SizedBox(height: 10,),
138          password,
139          const SizedBox(height: 24.0,),
140          loginButton,

```

Рисунок 23 – Код верстки экран авторизации

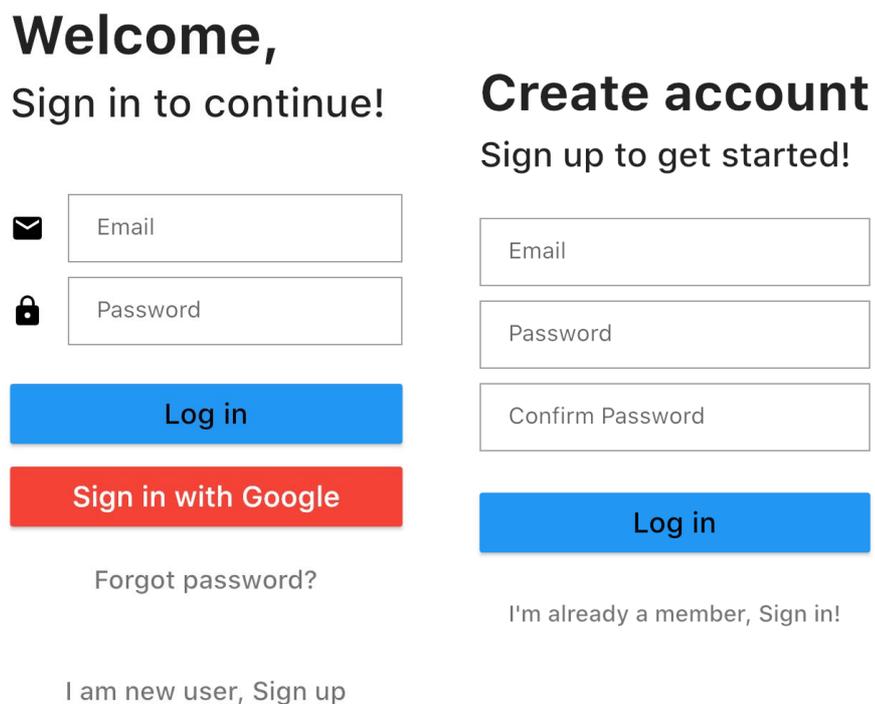


Рисунок 24 – Экраны авторизации и регистрации

### 3.2.2 Подключение проекта к Firebase

Для того, чтобы подключить приложение к Firebase, был создан и настроен проект на сайте фреймворка. Сначала были указаны приложения для разных ОС, таких как Android и iOS, введены данные приложений и скачан конфигурационный файл (рисунок 25). После были произведены изменения в файлах проекта, следуя инструкции на сайте, указаны зависимости.

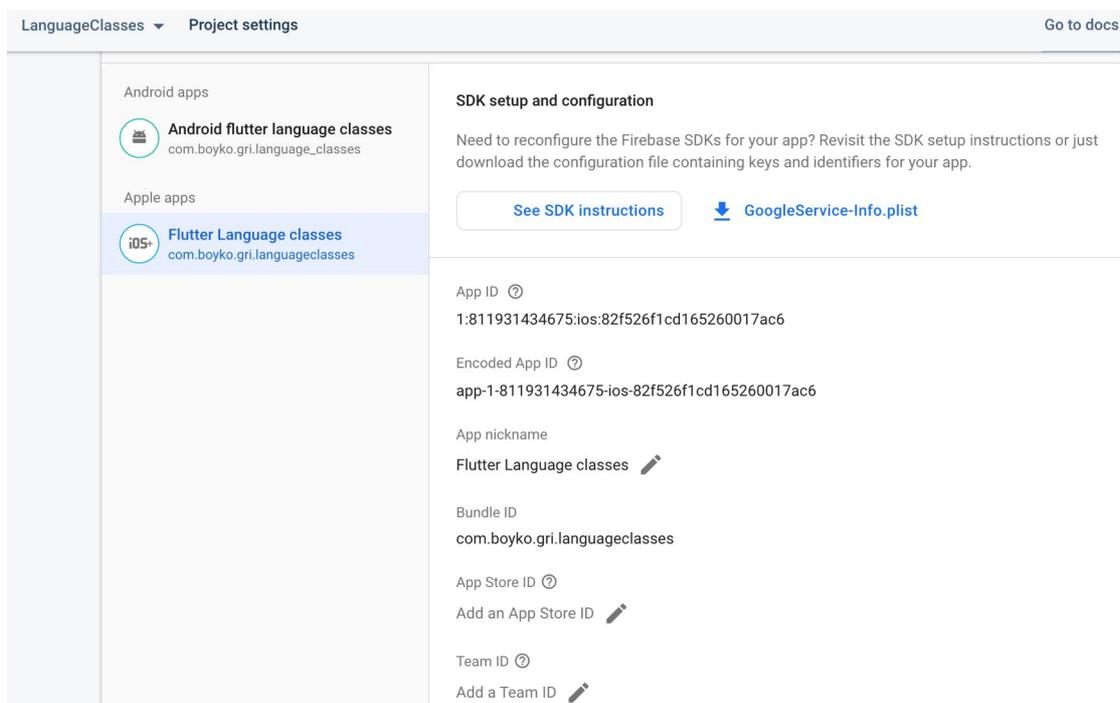


Рисунок 25 – Настройка Firebase

На вкладке Authentication на сайте можно выбрать необходимые способы аутентификации, для данного проекта были использованы почта/пароль и Google (рисунок 26).

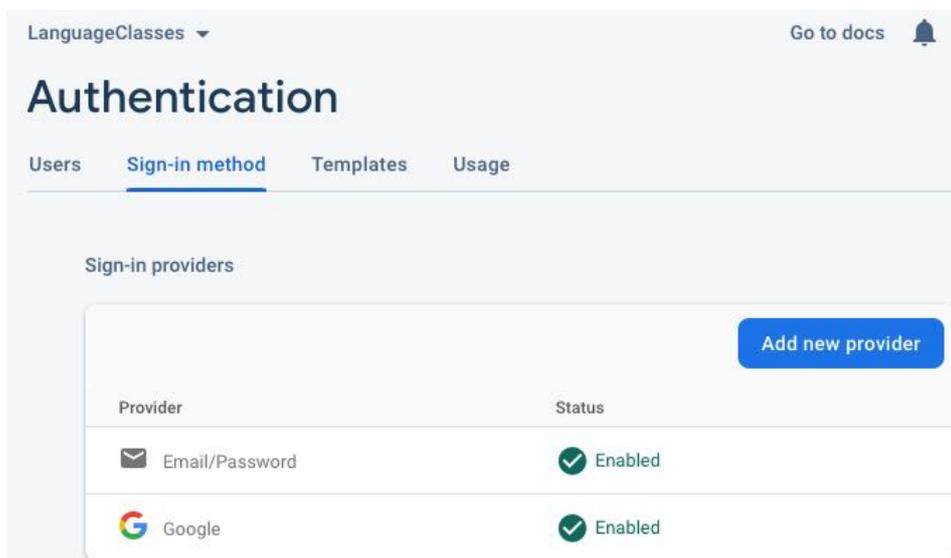


Рисунок 26 – Настройка способов аутентификации Firebase

### 3.2.3 Настройка взаимодействия с системой аутентификации Firebase

Для начала были подключены 2 зависимые библиотеки `firebase_auth` и `google_sign_in` (рисунок 27).

```
dependencies:  
  flutter:  
    sdk: flutter  
  
  firebase_auth: ^3.3.8  
  google_sign_in: ^5.3.1
```

Рисунок 27 – Настройка зависимостей для аутентификации Firebase

Затем был инициализирован Firebase при старте приложения в методе `main` (рисунок 28).

```
main() async {  
  await Firebase.initializeApp();  
  
  runApp(MyApp());  
}
```

Рисунок 28 – Инициализация Firebase

В следующем этапе для реализации архитектуры, которую можно легко поддерживать в дальнейшем, был использован один из главных принципов объектно-ориентированного программирования – принцип единой ответственности, он определяет ответственность класса как причину изменения и заявляет, что у класса должна быть только одна причина для изменения. Исходя из этого был создан класс `AuthService`, который отвечает за аутентификацию пользователей. В нем содержится логика авторизации и регистрации посредством почты и пароля, аккаунта Google, а также выход из аккаунта (рисунок 29).

```

class AuthService {
    final FirebaseAuth _auth = FirebaseAuth.instance;
    final GoogleSignIn _googleSignIn = GoogleSignIn();

    Future<Object?> createUserWithEmailAndPassword({
        required BuildContext context,
        required String email,
        required String password,
    }) async {...}

    Future signInWithGoogle(BuildContext context) async {...}

    Future<Object?> loginWithEmailAndPassword({
        required BuildContext context,
        required String email,
        required String password,
    }) async {...}

    Future signOut(BuildContext context) async {...}

    _userDidChange(BuildContext context) async {...}
}

```

Рисунок 29 – Класс AuthService

В этом классе есть приватный метод `_userDidChange`, который вызывается при изменении состояния авторизации пользователя, он, как и другие методы, принимает контекст типа `BuildContext` и перенаправляет на нужный экран в зависимости от состояния пользователя, на домашнюю страницу или на экран авторизации (рисунок 30). Контекст при этом хранит информацию о расположении виджета относительно других виджетов и нужен для осуществления навигации.

```

_userDidChanged(BuildContext context) async {
  if (_auth.currentUser == null) {
    //User is currently signed out!
    Navigator.of(context).pushNamedAndRemoveUntil(
      LoginPage.routeName,
      (route) => false,
    );
  } else {
    //User is signed in!
    Navigator.of(context).pushNamedAndRemoveUntil(
      MyHomePage.routeName,
      (route) => false,
    );
  }
}

```

Рисунок 30 – Метод `_userDidChanged`

Сама логика навигации содержится в корневом виджете приложения – `MaterialApp`. У него есть несколько свойств, отвечающих за маршрутизацию экранов, главное из которых – `routes` (рисунок 31). Оно содержит словарь с названием пути маршрута и метод, возвращающий какой-либо виджет. Путь к экрану обычно обозначают также как в адресной строке через косую черту (рисунок 32).

```

Widget build(BuildContext context) {
  return MaterialApp(
    title: 'Flutter Demo',
    theme: ThemeData(...), // ThemeData
    home: user != null ? const MyHomePage() : const LoginPage(),
    debugShowCheckedModeBanner: false,
    routes: {
      MyHomePage.routeName: (ctx) => const MyHomePage(),
      LoginPage.routeName: (ctx) => const LoginPage(),
      SignUpPage.routeName: (ctx) => const SignUpPage(),
      CreateEditCardScreen.routeName: (ctx) => CreateEditCardScreen(),
      RepetitionScreen.routeName: (ctx) => RepetitionScreen(),
      NewReader.routeName: (ctx) => NewReader(),
      SettingsPage.routeName: (ctx) => SettingsPage(),
    },
  ),
}

```

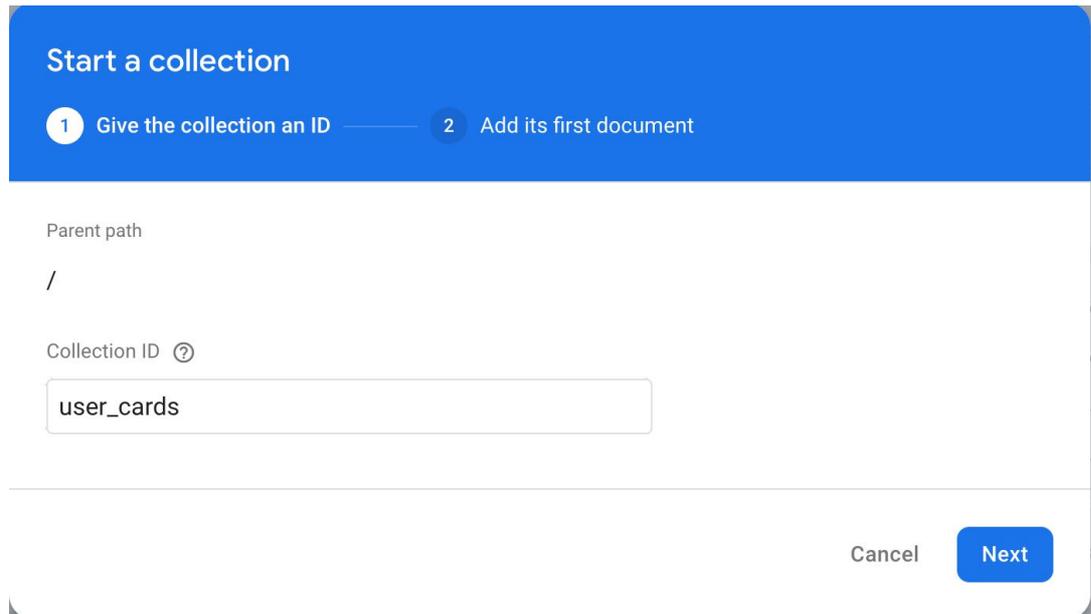
Рисунок 31 – Свойство `routes` виджета `MaterialApp`

```
class LoginPage extends StatefulWidget {  
    static String routeName = "/login";  
}
```

Рисунок 32 – Путь навигации к экрану авторизации

### 3.3 Создание базы данных

Процесс создание коллекций (таблиц) в Cloud Firestore состоит из нескольких шагов. Сначала было указано название (рисунок 33).



Start a collection

1 Give the collection an ID — 2 Add its first document

Parent path  
/

Collection ID ⓘ  
user\_cards

Cancel Next

Рисунок 33 – Создание коллекции в Cloud Firestore

После этого был сгенерирован id документа, добавлены необходимые поля, указан тип и имя (рисунок 34).

Document ID [?](#)

OGQ552By5ZolalxFMvek

Field	Type	Value
cardId	string	oCckWKylssUPhI
userId	string	vllbT9zqggYdAqI
dateCreated	timestamp	

Date

May 24, 2022

Time

00:00:00000

Рисунок 34 – Создание документа в Cloud Firestore

Список всех коллекций и документов в них выглядит следующим образом (рисунок 35).

languageclasses-df688	user_cards	OGQ552By5ZolalxFMvek
+ Start collection	+ Add document	+ Start collection
cards	OGQ552By5ZolalxFMvek	+ Add field
cards_groups		cardId: "oCckWKylssUPhPSD65egrA58tWA3"
cards_in_group		dateChanged: May 24, 2022 at 12:00:00 AM UTC+7
language_levels		dateCreated: May 24, 2022 at 12:00:00 AM UTC+7
languages		dateLastRepetition: May 24, 2022 at 12:00:00 AM UTC+7
lessons		dateNextRepetition: May 25, 2022 at 12:00:00 AM UTC+7
localized_theory		isDifficult: false
parts_of_speech		numberOfMistakes: 0
theories		repeatInterval: 360
user_cards		userId: "vllbT9zqggYdAqhFdK3gi2tCGdL2"

Рисунок 35 – Список коллекций и документов в Cloud Firestore

Для хранения файлов использовалось хранилище Firebase Cloud Storage. В нем можно создавать папки для удобного хранения файлов. Для данного проекта были созданы папки images и theory, для хранения изображений и теории в формате HTML соответственно (рисунок 36).

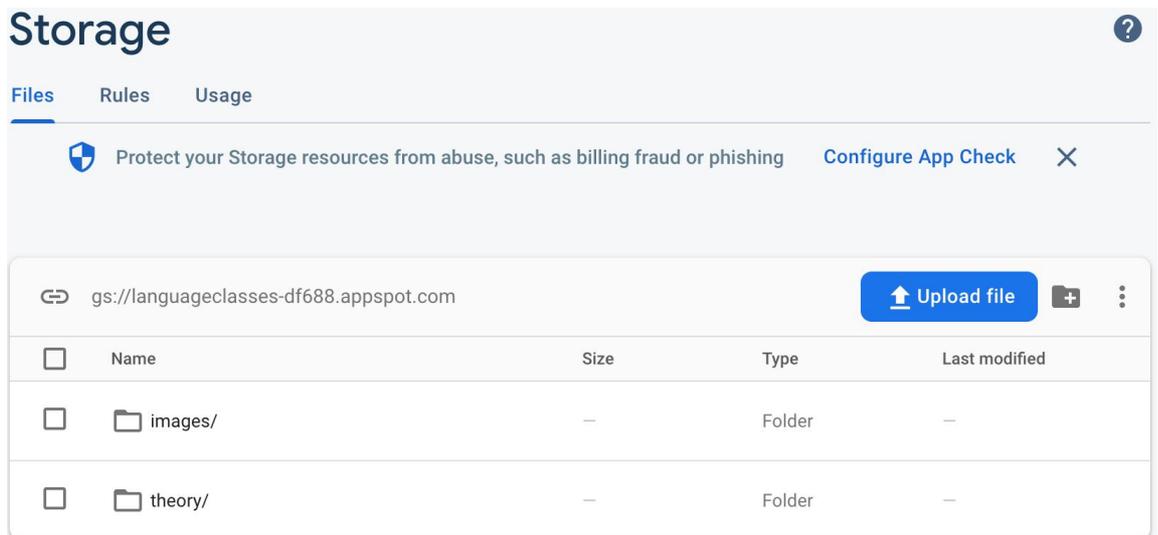


Рисунок 36 – Хранилище Firebase Cloud Storage

База данных и хранилище Firebase имеют возможность настраивать правила безопасности, которые позволяют уберечь данные от нежелательного изменения или просмотра, путем настройки доступа для чтения или записи данных при определенных условиях. Для коллекций были установлены следующие правила, не позволяющие читать или изменять данные, если пользователь не авторизован (рисунок 37).

```
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    match /{document=**} {
      allow read: if (request.auth.uid != null);
      allow write: if (request.auth.uid != null);
    }
  }
}
```

Рисунок 37 – Правила доступа к базе данных и хранилищу Firebase

### 3.4 Работа с сервером

Обычно работа с сервером заключается в отправке HTTP запросов через REST API серверу и получению ответов назад, но так как Firebase и его сервисы тесно связаны в приложении, есть ограничение во времени и ресурсах – создание и настройка API необязательна. Вместо этого были использованы

классы посредники, которые выполняют логику сервера и возвращают/меняет данные в БД.

Рассмотрим взаимодействие с серверов на примере работы с карточками слов. Для этого был создан класс `CardsNetworkManager` для непосредственной работы с БД на сервере, и перечисление `SortType` для сортировки. `CardsNetworkManager` содержит ссылки на базу данных `Firestore`, текущего пользователя, коллекции `cards` и `user_cards`, а также тип сортировки и методы (рисунок 38). Модель карточки у слова содержит поля, совмещенные с двух коллекций – `cards` и `user_cards`, а также преобразованная в строку часть речи (рисунок 39). Созданный менеджер преобразует полученные данные с нескольких коллекций в одну модель данных и возвращает ее пользователю (методы `add`, `update`, `delete`), также и в обратную сторону (методы `getAllCards`, `getCard`, `getCardForSearchText`)

```
enum SortType {
    ascendingProgress, descendingProgress, newFirst, oldFirst, az, za
}

class CardsNetworkManager {
    final db = FirebaseFirestore.instance;
    User? user = FirebaseAuth.instance.currentUser;
    SortType sort = SortType.descendingProgress;
    late var cardsRef = db.collection("cards");
    late var userCardsRef = db.collection("user_cards");

    Future<bool> update(CardModel card) async {...}

    Future<bool> add(CardModel card) async {...}

    Future<bool> delete({required String cardId}) async {...}

    Future<List<CardModel>?> getAllCards() async {...}

    Future<CardModel?> getCard({required String cardId}) async {...}

    Future<List<CardModel>?> getCardsForSearchText(String text) async {...}
}
```

Рисунок 38 – Класс `CardsNetworkManager` и перечисление `SortType`

```

class CardModel {
  String id = "";
  String text;
  String translation;
  String? transcription;
  String? partOfSpeech;
  String? notes;
  int repeatInterval = 360;
  late DateTime dateCreated;
  late DateTime dateChanged;
  late DateTime dateNextRepetition;
  DateTime? dateLastRepetition;
  int numberOfMistakes = 0;
  bool isDifficult = false;
}

```

Рисунок 39 – Класс CardModel

### 3.5 Реализация перевода по нажатию

Для реализации перевода по нажатию, необходимо сделать два действия: распознать слово по нажатию на него и перевести.

#### 3.5.1 Распознавание слов

Так как новостные статьи имеют сложный и стилизованный HTML код, то было принято решение отображать его в том же виде целиком, не меняя внешний вид. Для взаимодействия с HTML нужно использовать JavaScript, для этого пришлось воспользоваться сторонним решением для отображения Web страниц, которое имеет удобный доступ работы со скриптами – flutter\_inappwebview [21].

Данная библиотека предлагает прежде всего виджет InAppWebView, который умеет обрабатывать JavaScript функции.

Сам скрипт представляет собой обработчик нажатия, который активирует функцию returnClickedWord и возвращает нажатое слово (рисунок 39). Чтобы связать виджет с данной функцией, необходимо вызвать window.flutter\_inappwebview.callHandler внутри слушателя события flutterInAppWebViewPlatformReady, которое оповещает, что платформа готова к обработке методов (рисунок 41).

```

function returnClickedWord() {
    // Gets clicked on word (or selected text if text is selected)
    var t = '';
    if (window.getSelection && (sel = window.getSelection()).modify) {
        var s = window.getSelection();
        if (s.isCollapsed) {
            s.modify('move', 'forward', 'character');
            s.modify('move', 'backward', 'word');
            s.modify('extend', 'forward', 'word');
            t = s.toString();
            s.modify('move', 'forward', 'character'); //clear selection
        }
        else {
            t = s.toString();
        }
    }
    else if ((sel = document.selection) && sel.type != "Control") {
        // IE 4+
        var textRange = sel.createRange();
        if (!textRange.text) {
            textRange.expand("word");
        }
        // Remove trailing spaces
        while (/s\$/ .test(textRange.text)) {
            textRange.moveEnd("character", -1);
        }
        t = textRange.text;
    }
    return t.split(",").join("").split(".").join("").trim();
}

```

Рисунок 40 – Функция для распознавания слово по нажатию

```

window.addEventListener("flutterInAppWebViewPlatformReady", function (event) {
    console.log("ready");

    document.addEventListener('click', function () {
        window.flutter_inappwebview.callHandler('clickOnWord', returnClickedWord());
    });
});

```

Рисунок 41 – Вызов метода для распознавания слово по нажатию

Сам скрипт добавляется изначально в HTML файл в свойстве `initialUserScripts` и после создания виджета, устанавливается обработчик нажатия, который получает нажатое слово (рисунок 42).

```

onWebViewCreated: (controller) async {
  print('onWebViewCreated');
  inAppWebViewController = controller;

  controller.addJavaScriptHandler(
    handlerName: 'clickOnWord',
    callback: (args) async {
      if (args.isNotEmpty) {
        String word = args.first as String;
        print('clickOnWord - $word');
        showTranslationsFor(context, word: word);
      }
    });
},
initialUserScripts: UnmodifiableListView<UserScript>([
  UserScript(
    source: TapableWordsScript.script,
    injectionTime: UserScriptInjectionTime.AT_DOCUMENT_END),
]), // UnmodifiableListView

```

Рисунок 42 – Настройка скрипта в виджете

Так как скрипт написан и работает с HTML, нецелесообразно хранить теорию и статьи в других форматах и делать другую логику.

В результате интерфейс просмотра любого HTML текста выглядит следующим образом (рисунок 43).



Рисунок 43 - Экран просмотра текста в формате HTML

### 3.5.2 Получение перевода

После того, как было получено слово, можно его перевести, установив пользовательский ключ, язык перевода и слово в параметрах запроса класса YandexDictionary (рисунок 44).

```
class YandexDictionary {
    final Map<String, String> _defaultParameters = {
        "key" : "dict.1.1.20201201T144656Z.9be9b4db4fdb8683.8c833ba38cf16fcd45cdd",
        "lang" : "en-ru",
    };

    final _url = "dictionary.yandex.net";
    final _path = "/api/v1/dicservice.json/lookup";

    Future<List<Translation>> getTranslationsFromYaDict({required String word
        _defaultParameters['text'] = word;

    try {
        var url = Uri.https(_url, _path, _defaultParameters);
        var response = await http.get(url, headers: _defaultParameters);
```

Рисунок 44 – Класс YandexDictionary

Сервис возвращает ответ в следующем виде (рисунок 45).

```
class Translation {
    String text;
    String? pos; // Часть речи (может отсутствовать)
    String? ts; // Транскрипция
    List<Translation>? tr; // Массив переводов
    String? gen; // Род
    List<Translation>? syn; // Массив синонимов
    List<Translation>? mean; // Массив значений
    List<Translation>? ex; // Массив примеров
}
```

Рисунок 45 – Модель перевода Яндекс.Словаря

После этого происходит массивная обработка ответа для отображения ее на экране (рисунок 46).

```
final word = trn.text;
//Выводим часть речи, если только основные блоки перевода (если есть)
final pos = trn.pos;
//Выводим транскрипцию (если есть)
final ts = trn.ts;

viewModels.add(TranslationTitleViewModel(word: word, pos: pos, transcription: ts));

//Вызываем рекурсию для вывода переводов
if (trn.tr != null) {
    for (var t in trn.tr!) {
        List<WordAndGen> wordAndGenTranslations = [];
        final word = t.text;
        final gen = t.gen;
        wordAndGenTranslations.add(WordAndGen(word, gen));

        //Если есть синонимы – выводим
        final syn = t.syn;
        if (syn != null) {...}

        //Выводим англ значение различных переводов
        List<String>? foreignSynonyms;

        final mean = t.mean;
        if (mean != null) {...}
        List<TranslationExample>? examples;
        //Выводим примеры
        final ex = t.ex;
        if (ex != null) {...}
        viewModels.add(TranslationsViewModel(...));
    }
}
```

Рисунок 46 – Обработка исходной модели перевода

Таким образом, после нажатия на слово и обработки ответа с сервера, экран перевода выглядит следующим образом (рисунок 47).



Рисунок 47 - Экран перевода иностранных слов

Пользователь может нажать на слово, выделенное синим цветом, и перейти на экран создания карточки, в который передается и заполняется вся имеющаяся информация о слове (рисунок 48). Если пользователь нажал на кнопку «Добавить в словарь», то информацию о слове нужно вводить самостоятельно.

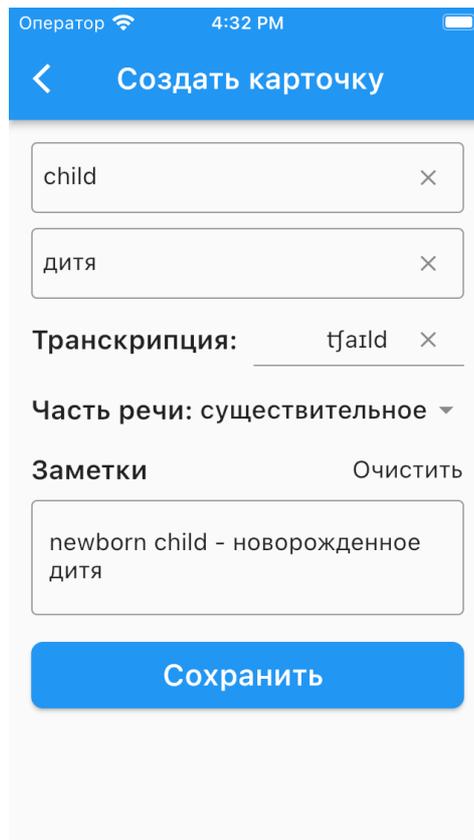


Рисунок 48 - Экран создания карточки

### **Вывод по главе**

В ходе реализации мобильного приложения были сделаны основные экраны (авторизация и регистрации, просмотр всех слов, уроков и новостей, создание и редактирование карточек, повторение в виде карточек, просмотр конкретного новостного или теоретического материала, экран с переводом слова), создана и настроена база данных, налажено взаимодействие с сервером.

Приложение успешно прошло тестирование на симуляторе под управлением ОС Android и iOS, но все еще не готово для выпуска в магазины приложений. Необходимо доработать функционал, протестировать на различных реальных устройствах и создать продуманный языковой курс.

## **4 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ**

Разработка НИ производится группой работников, состоящей из двух человек – руководителя и студента.

Данная выпускная квалификационная работа заключается в проектировании и реализации сервиса для изучения иностранных языков в виде мобильного приложения.

Целью раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» является определение перспективности и успешности НИ, оценка его эффективности, уровня возможных рисков, разработка механизма управления и сопровождения конкретных проектных решений на этапе реализации.

Для достижения обозначенной цели необходимо решить следующие задачи:

1. Оценить коммерческий потенциал и перспективность разработки НИ;
2. Осуществить планирование этапов выполнения исследования;
3. Рассчитать бюджет затрат на исследования;
4. Произвести оценку научно-технического уровня исследования и оценку рисков.

### **4.1 Потенциальные потребители результатов исследования**

**Целевой рынок** – потребители возраста начальных классов и старше, любой национальности, социальной принадлежности и профессии, с любым уровнем дохода, владеющие одним из языков, доступном в приложении и желающие изучать иностранные языки, которые есть в сервисе. На начальном этапе в приложении будут доступны русский и английский языки в качестве родного и иностранного соответственно.

Наиболее значимыми критериями, которыми должен обладать потребитель и на которые будет нацелено приложение, являются: знание русского языка, уровень дохода не позволяющий оплачивать языковые школы и преподавателей.

В будущем, после добавления других языков в качестве родных и иностранных, количество потенциальных потребителей увеличится многократно, главное, чтоб они подходили под доступные языки в приложении.

#### 4.2 Анализ конкурентных технических решений

При рассмотрении различных сервисов для изучения иностранных языков, в качестве главных конкурентных решений с похожими возможностями можно выделить 2 приложения: Ewa (Ke) и Memrise (Km)

**Ewa** имеет возможность выбрать курсы разной сложности, в которых есть интерактивные уроки, больше направленные на аудирование, чем на построение приложений и фраз, режим повторений для слов слишком медленный, а для практики к тому же не столь разнообразный. Набор словарного запаса происходит через статьи, книги, аудиокниги, а также во время прохождения курса и игр, но нельзя добавить собственные слова (рисунок 49). Весь основной функционал бесплатный, пользователю показывается реклама, также ее можно отключить и получить дополнительный функционал за подписку.

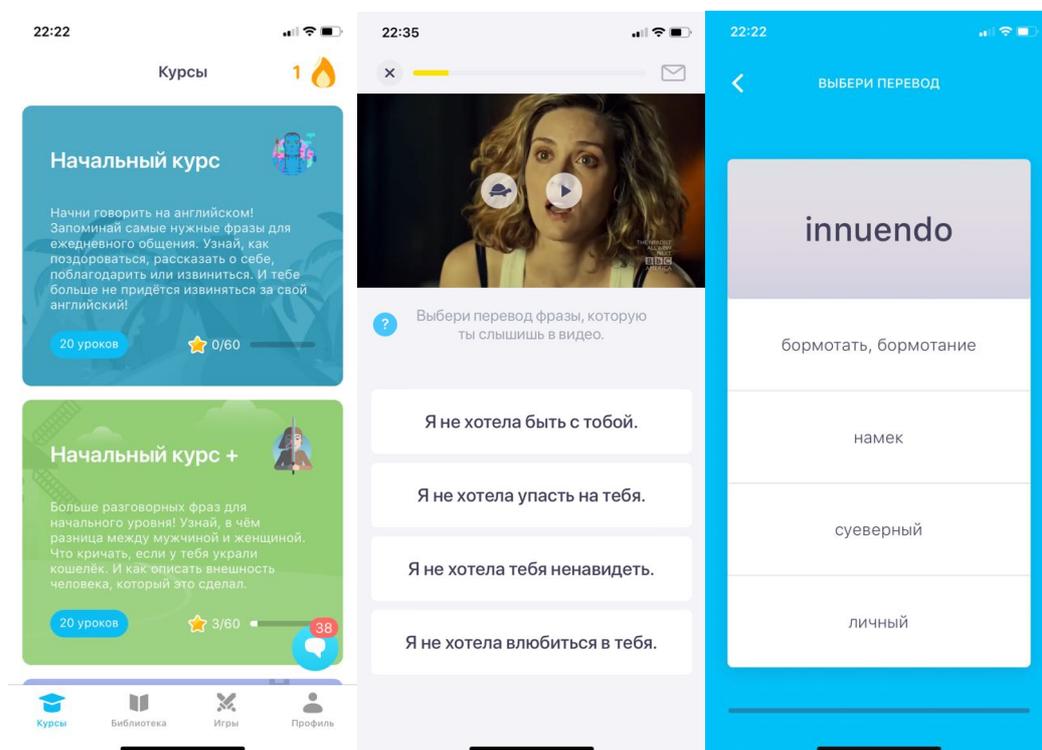


Рисунок 49 – Приложение Ewa

**Memrise** в свою очередь больше ориентирован на изучения слов и фраз, процесс не самый быстрый, но его можно настроить. Пополнять словарный запас через чтение или добавлять собственные слова нельзя. Также значительно ограничен функционал в бесплатной версии и показывается реклама (рисунок 50).

Текущее решение предполагает показывать рекламу пользователям, но никак не ограничивать функционал приложения.

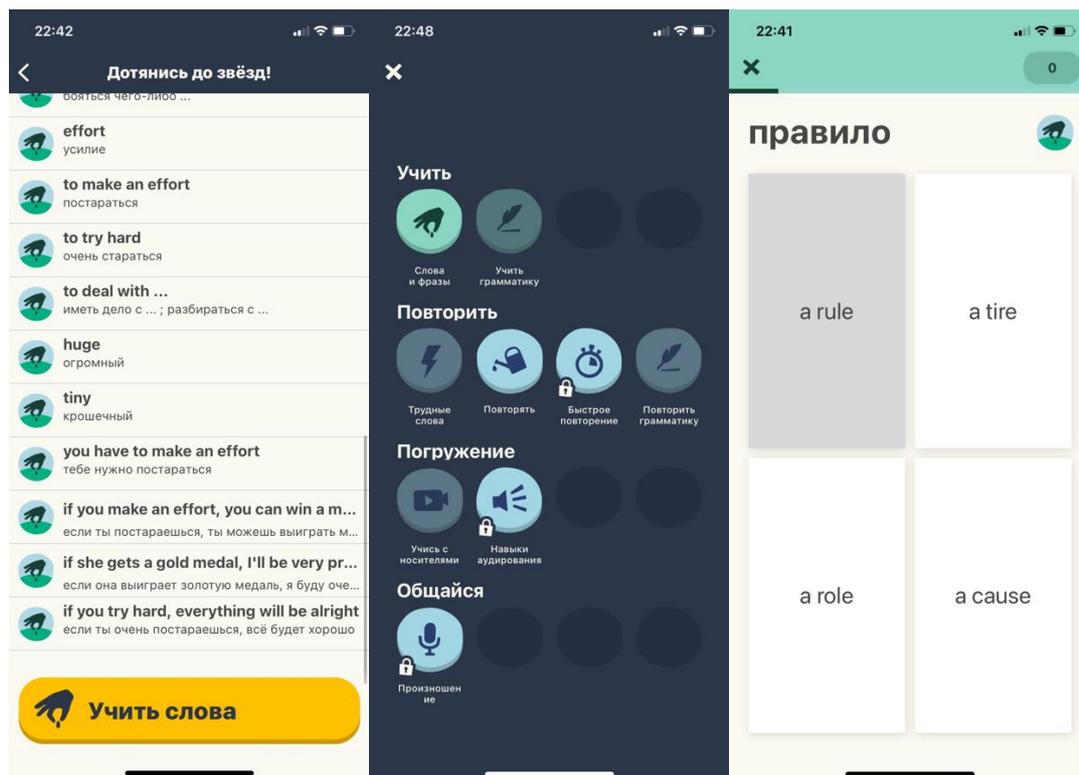


Рисунок 50 – Приложение Memrise

Анализ конкурентных технических решений был проведен с помощью оценочной карты, представленной в таблице 1 (индекс Т отвечает за текущее решение).

Таблица 1. Оценочная карта критериев эффективности конкурентных технических решений (разработок)

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Б <sub>Т</sub>	Б <sub>Ке</sub>	Б <sub>Км</sub>	К <sub>Т</sub>	К <sub>Ке</sub>	К <sub>Км</sub>
<b>Технические критерии оценки ресурсоэффективности</b>							

Простота и удобство в эксплуатации	0,1	5	5	4	0,5	0,5	0,4
Скорость изучения и повтора слов и упражнений	0,2	5	3	4	1	0,6	0,8
Источники для пополнения словарного запаса	0,1	4	5	2	0,4	0,5	0,2
Возможность добавлять свои слова	0,1	5	1	1	0,5	0,1	0,1
Уровни сложности материала	0,1	4	5	3	0,4	0,5	0,3
<b>Экономические критерии оценки эффективности</b>							
Послепродажное обслуживание	0,15	5	3	4	0,75	0,45	0,6
Цена	0,25	5	4	3	1,25	1	0,75
<b>Итого</b>	<b>1</b>	<b>33</b>	<b>26</b>	<b>21</b>	<b>4,8</b>	<b>3,65</b>	<b>3,15</b>

Анализ конкурентных решений определяется по формуле 1:

$$K = \sum V_i \times B_i, \quad (1)$$

где K – конкурентоспособность научной разработки или конкурента;

$V_i$  – вес показателя (в долях единицы);

$B_i$  – балл  $i$ -го показателя.

Основываясь на рассчитанных значениях, можно предположить, что скорость изучения и повтора слов и упражнений, возможность добавлять свои слова и цена формирует более чем конкурентное преимущество. Существующим альтернативам не хватает возможности добавлять свои слова и бесплатности продукта без ограничения функциональности, также послепродажное обслуживание у них дороже, из-за наличия видео материалов, аудиокниг (приложение Ewa).

### 4.3 Технология QuaD

Технология QuaD (QUality ADvisor) представляет собой гибкий инструмент измерения характеристик, описывающих качество новой разработки и ее перспективность на рынке и позволяющие принимать решение

целесообразности вложения денежных средств в научно-исследовательский проект.

Анализ имеющихся конкурентных продуктов необходимо проводить с достаточной регулярностью, так как рынок ИТ находится в постоянном и активном движении на сегодняшний день. Данный анализ позволяет производить изменения текущего внедрения, чтобы сделать его наиболее перспективным и конкурентоспособным в дальнейшем. Результат QuaD-анализа представлен в таблице 2.

Таблица 2. Оценка критериев в соответствии с технологией QuaD

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение	Средневзвешенное значение
Простота и удобство в эксплуатации	0,1	95	100	0,95	9,5
Скорость изучения и повтора слов и упражнений	0,2	90	100	0,9	18
Источники для пополнения словарного запаса	0,1	85	100	0,85	8,5
Возможность добавлять свои слова	0,1	100	100	1	10
Уровни сложности материала	0,1	85	100	0,85	8,5
Послепродажное обслуживание	0,15	90	100	0,9	13,5
Цена	0,25	90	100	0,9	22,5
<b>Итого:</b>					<b>90,5</b>

Средневзвешенное значение показателя качества и перспективности научной разработки равен 90,5 (попадает в промежуток 80–100), то есть такая разработка считается перспективной.

#### 4.4 SWOT-анализ

SWOT-анализ – метод стратегического планирования, заключающийся в выявлении факторов внутренней и внешней среды организации и разделении их на четыре категории:

- Strengths (сильные стороны);
- Weaknesses (слабые стороны);
- Opportunities (возможности);
- Threats (угрозы).

Сильные (S) и слабые (W) стороны являются факторами внутренней среды объекта анализа, (то есть тем, на что сам объект способен повлиять); возможности (O) и угрозы (T) являются факторами внешней среды (то есть тем, что может повлиять на объект извне и при этом не контролируется объектом).

В рамках данного анализа выявлены сильные и слабые стороны проекта, а также его возможности и угрозы. Все параметры SWOT анализа данного проекта представлены в таблице 3.

Таблица 3. SWOT-анализ проекта

		Внутренние факторы	
		Сильные стороны проекта: 1. Возможность добавлять свои слова 2. Скорость изучения и повтора слов и упражнений 3. Возможность пользоваться продуктом бесплатно, без потери функциональности	Слабые стороны проекта: 1. Сложность поддержки системы 2. Необходимость длительной поддержки 3. Необходимость в качественном и объемном материале для изучения 4. Недостаточная проработка дизайна приложения системы
Внешние факторы	<b>Возможности:</b> 1. Продажа готового продукта для крупной компании 2. Выведение продукта на мировой рынок 3. Рост спроса на рынке приложений для изучения иностранных языков 4. Приоритетная политика правительства РФ по поддержке малого и среднего бизнеса 5. Приоритетная политика правительства РФ по поддержке IT сектора	1. Востребованность как среди конечных потребителей, так и среди крупных организаций 2. Значительные конкурентные преимущества 3. Новый подход к изучению	1. Ценовая политика конкурентов поможет повысить количество лояльных клиентов и репутацию организации 2. Качество материала приоритетнее объема, а источники информации в виде новостей – безграничны 3. Дизайн всегда можно изменить
	<b>Угрозы:</b> 1. Неокупаемость проекта в связи с низким количеством пользователей 2. Несоответствие ожиданиям клиента 3. Снижение спроса на обучение иностранным языкам	1. Интуитивно понятный дизайн, а также возможность его изменить может повысить посещаемость ресурса 2. Увидев преимущества данного приложения, пользователи будут рассказывать знакомым с такими же интересами, что позволит привлечь новых пользователей	1. Малые вычислительные мощности серверов могут уменьшить посещаемость сайта 2. Неблагоприятные изменения в гос. законах могут повлечь убытки (федеральный закон о персональных данных) 3. Ухудшение отношений между странами могут стать

	4. Заимствование конкурентных преимуществ другими компаниями		причиной ухудшения распространения сервиса и монетизации
--	--	--	--

SWOT-анализ показал, что риски, связанные с устранением несоответствий на всех этапах эксплуатации продукта, должны занимать ключевую роль в расчете выгоды предлагаемой разработки. Общая картина говорит о том, что, несмотря на угрозы и слабые стороны проекта, разработка обладает конкурентным преимуществом и является перспективной.

#### **4.5 Планирование научно-исследовательских работ**

##### **4.5.1 Структура работ в рамках научного исследования**

Важным этапом проведения научно-исследовательских работ является необходимость планирования работ, которое включает в себя определение полного перечня работ, а также их распределение между всеми исполнителями проекта. Исполнителями проекта являются студент и научный руководитель. Научный руководитель определяет цели и задачи для студента, направляет и контролирует работу его работу, оценивает результаты проделанной работы и дает рекомендации студенту. Студент полностью отвечает за выполняемую работу. В таблице 4 представлен перечень работ, а также распределение исполнителей по ним.

Таблица 4. Перечень работ и распределение исполнителей

№ работы	Наименование работы	Исполнители работы
1	Выбор научного руководителя бакалаврской работы	Бойко Г.Д. Фадеев А.С.
2	Составление и утверждение темы бакалаврской работы	Бойко Г.Д. Фадеев А.С.
3	Составление календарного плана-графика выполнения бакалаврской работы	Бойко Г.Д.
4	Подбор и изучение литературы по теме бакалаврской работы	Бойко Г.Д. Фадеев А.С.
5	Анализ предметной области	Бойко Г.Д.
6	Проектирование информационной системы	Бойко Г.Д.
7	Разработка клиентского части приложения	Бойко Г.Д.
8	Настройка серверной части приложения	Бойко Г.Д.
9	Согласование выполненной работы с научным руководителем	Бойко Г.Д. Фадеев А.С.

10	Выполнение других частей работы (финансовый менеджмент, социальная ответственность)	Бойко Г.Д.
11	Подведение итогов, оформление работы	Бойко Г.Д.

#### 4.5.2 Определение трудоемкости выполнения работ

Как правило, трудовые затраты образуют основную часть стоимости исследования, поэтому важным этапом планирования научно-исследовательской деятельности является определение трудоемкости работ.

Определение трудоёмкости выполнения работ осуществляется на основе экспертной оценки ожидаемой трудоёмкости выполнения каждой работы путём расчёта длительности работ в рабочих и календарных днях каждого этапа работ.

Трудоемкость оценивается по следующей формуле 2:

$$t_{ож\ i} = \frac{3t_{мин\ i} + 2t_{макс\ i}}{5}, \quad (2)$$

где  $t_{ож\ i}$  – это ожидаемая трудоемкость  $i$ -ой работы (чел.-дни),

$t_{мин\ i}$  – это минимально возможная трудоемкость выполнения заданной  $i$ -ой работы (оптимистическая оценка) (чел.-дни),

$t_{макс\ i}$  – это максимально возможная трудоемкость выполнения заданной  $i$ -ой работы (пессимистическая оценка) (чел.-дни).

После оценки ожидаемой трудоемкости работ, производится определение продолжительности каждой работы в рабочих днях по формуле 3:

$$T_{pi} = \frac{t_{ож\ i}}{Ч_i}, \quad (3)$$

где  $T_{pi}$  – это продолжительность одной работы (раб. дни),

$t_{ож\ i}$  – это ожидаемая трудоемкость выполнения одной работы (чел.-дни),

$Ч_i$  – это численность исполнителей, выполняющих одновременно одну и ту же работу на этом этапе (чел.).

Для того чтобы в дальнейшем построить график работ с помощью диаграммы Ганта, необходимо также произвести перевод длительности работ из рабочих дней в календарные по формуле 4:

$$T_{ki} = T_{pi} \times k_{\text{кал}}, \quad (4)$$

где  $T_{ki}$  – это продолжительность выполнения  $i$ -й работы в календарных днях;

$T_{pi}$  – это продолжительность выполнения  $i$ -й работы в рабочих днях;

$k_{\text{кал}}$  – это коэффициент календарности, равный 1,48.

Коэффициент календарности  $k_{\text{кал}}$  рассчитывается по формуле 5:

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - T_{\text{вых}} - T_{\text{пр}}}, \quad (5)$$

где  $k_{\text{кал}}$  – коэффициент календарности;

$T_{\text{кал}}$  – количество календарных дней в году;

$T_{\text{вых}}$  – количество выходных дней в году;

$T_{\text{пр}}$  – количество праздничных дней в году.

С учётом того, что календарных дней в 2022 году 365, а сумма выходных и праздничных дней составляет 118 дней, коэффициент календарности равен  $k_{\text{кал}} = 1,48$ .

#### 4.5.3 Разработка графика проведения научного исследования

В таблице 5 приведены временные показатели научного исследования.

Таблица 5. Временные показатели проведения научного исследования

Наименование работы	Исполнители работы	Трудоёмкость работ, чел-дни		Длительность работ, дни		
		tmin	tmax	тож	Тр	Тк
Выбор научного руководителя бакалаврской работы	Бойко Г.Д. Фадеев А.С.	1	2	1,4	1	1
Составление и утверждение темы бакалаврской работы	Бойко Г.Д. Фадеев А.С.	1	3	1,8	1	1
Составление календарного плана-графика выполнения бакалаврской работы	Бойко Г.Д.	1	2	1,4	1	1
Подбор и изучение литературы по теме бакалаврской работы	Бойко Г.Д. Фадеев А.С.	5	10	7	4	6
Анализ предметной области	Бойко Г.Д.	5	10	7	7	10

Проектирование информационной системы	Бойко Г.Д.	8	14	10,4	10	15
Разработка клиентского части приложения	Бойко Г.Д.	20	30	24	24	36
Настройка серверной части приложения	Бойко Г.Д.	10	20	14	14	21
Согласование выполненной работы с научным руководителем	Бойко Г.Д. Фадеев А.С.	2	2	2	1	1
Выполнение других частей работы (финансовый менеджмент, социальная ответственность)	Бойко Г.Д.	5	12	7,8	8	12
Подведение итогов, оформление работы	Бойко Г.Д.	8	14	10,4	10	15
Итого	Бойко Г.Д.	47	93	65,4	81	86
	Фадеев А.С.	9	17	12,2	7	9

На основе данных таблицы 5 был построен календарный план-график представленный на рисунке 51.

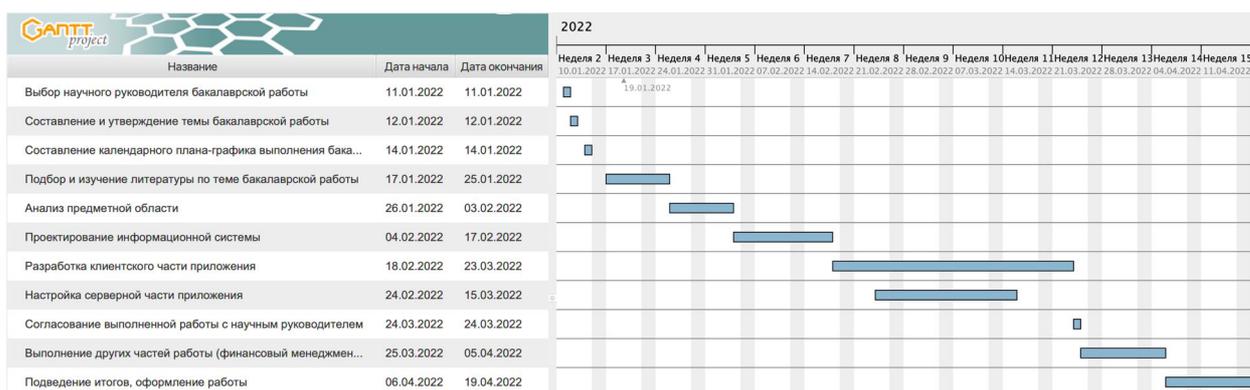


Рисунок 51 – Календарный план-график проведения работ

#### 4.6 Бюджет научно-технического исследования

В состав бюджета входит стоимость всех расходов, необходимых для выполнения работ по проекту. При формировании бюджета используется группировка затрат по следующим статьям:

- расчет материальных затрат НТИ;
- затраты на специальное оборудование;
- основная заработная плата исполнителей;
- дополнительная заработная плата исполнителей;
- отчисления во внебюджетные фонды;
- накладные расходы.

#### 4.6.1 Расчет материальных затрат НИИ

Так как затраты на электроэнергию, услуги связи, канцелярские принадлежности и т.д. учитываются как накладные расходы и не относятся в категорию материальных затрат, то сумма материальных затрат равняется нулю.

#### 4.6.2 Расчет затрат на специальное оборудование для научных (экспериментальных) целей

Данная статья затрат включает в себя затраты на приобретение специального оборудования. Также в эту статью включаются затраты по доставке и монтажу оборудования, равные 15% от его стоимости.

В ходе работы над проектом использовалось оборудование, имеющееся у исполнителей, соответственно необходим расчет его амортизации.

При создании информационной системы был использован один ноутбук, стоимостью 100000 рублей.

Расчет амортизации ноутбука: первоначальная стоимость 100000 рублей; срок полезного использования для машин офисных код 330.28.23.23 составляет 36 месяцев. Планируемое время использования для написания ВКР - 6 месяцев.

Норма амортизация основных средств линейным способом рассчитывается по формуле 6:

$$A_n = \frac{1}{n} \times 100\%, \quad (6)$$

$n$  – установленный срок в месяцах;

$A_m$  – норма амортизации.

Тогда расчет амортизации ПК:

Норма амортизации:

$$A_n = \frac{1}{n} \times 100\% = \frac{1}{36} \times 100\% = 2,78\%$$

Ежемесячные амортизационные отчисления:

$$A_m = 100000 \times 0,0278 = 2780 \text{ рублей}$$

Итоговая сумма амортизации основных средств:

$$A = 2780 \times 6 = 16680 \text{ рублей}$$

Таким образом, сумма затрат на специальное оборудование составляет 16680 рублей, в виде амортизационных отчислений.

#### 4.6.3 Основная заработная плата исполнителей темы

Данная статья затрат включает основную заработную плату, премии и доплаты всех исполнителей проекта. В качестве исполнителей проекта выступают студенты и научный руководитель.

Зарботная плата рассчитывается по формуле 7:

$$Z_{зп} = Z_{осн} + Z_{доп}, \quad (7)$$

где  $Z_{зп}$  – заработная плата исполнителя;

$Z_{осн}$  – основная заработная плата исполнителя;

$Z_{доп}$  – дополнительная заработная плата исполнителя (12%-15% от размера основной заработной платы).

Основную заработную плату можно получить по формуле 8:

$$Z_{осн} = Z_{дн} \times T_p \times (1 + K_{пр} + K_{д}) \times K_p \quad (8)$$

где  $Z_{дн}$  – среднедневная заработная плата, руб.

$K_{пр}$  – премиальный коэффициент (0,3);

$K_{д}$  – коэффициент доплат и надбавок (0,2-0,5);

$K_p$  – районный коэффициент (для Гомска 1,3);

$T_p$  – продолжительность работ, выполняемых работником, раб. дни

где  $Z_{дн}$  – среднедневная заработная плата исполнителя;

$T_p$  – продолжительность работ, выполняемых исполнителем.

Среднедневную заработную плату можно получить по формуле 9:

$$Z_{дн} = \frac{Z_m \times M}{F_d}, \quad (9)$$

где  $Z_m$  – месячный должностной оклад исполнителя, рубли;

$M$  – количество месяцев работы равно:

При отпуске в 24 рабочих дня  $M = 11,2$  месяца, 5-дневная неделя;

При отпуске в 48 рабочих дней  $M = 10,4$  месяца, 6-дневная неделя;

$F_d$  – действительный годовой фонд рабочего времени персонала по разработке.

Должностные оклады исполнителей проекта согласно приказу ТПУ представлены в таблице 6.

Таблица 6. Месячные должностные оклады исполнителей

Исполнитель	Районный коэффициент (для Томска)	Размер месячного должностного оклада без учета коэффициента, рубли
Научный руководитель (должность – доцент, степень – кандидат технических наук)	1,3	33664
Студент (ассистент, без степени)	1,3	21760

Баланс рабочего времени для 6-дневной рабочей недели представлен в таблице 7.

Таблица 7. Баланс рабочего времени (для 6-дневной недели)

Показатели рабочего времени	Дни
Календарные дни	365
Нерабочие дни (праздники/выходные)	118
Действительный годовой фонд рабочего времени	247

На основе формулы 8 и таблиц 6–7 была рассчитана среднедневная заработная плата:

$$Z_{\text{дн}}(\text{студент}) = \frac{21760 \times 10,4}{244} = 927,48 \text{ рубля}$$

$$Z_{\text{дн}}(\text{научный руководитель}) = \frac{33664 \times 10,4}{243} = 1434,86$$

Расчет затрат на основную заработную плату приведен в таблице 8.

Таблица 8. Затраты на основную заработную плату

Исполнители	Здн, руб.	Кпр	Кд	Кр	Гр	Зосн, руб.
Студент	927,48	0,3	0,2	1,3	81	146495,47
Научный руководитель	1434,86	0,3	0,2	1,3	7	19585,84
<b>Итого:</b>						166081,31

Итоговая сумма затрат на основную заработную плату составила 166081,31 руб.

#### 4.6.4 Дополнительная заработная плата исполнителей темы

Данная статья расходов учитывает величину предусмотренных Трудовым кодексом РФ доплат за отклонение от нормальных условий труда и выплат, связанных с обеспечением гарантий и компенсаций.

Расчёт дополнительной заработной платы осуществляется по формуле 10:

$$З_{\text{доп}} = k_{\text{доп}} \times З_{\text{осн}}, \quad (10)$$

где  $З_{\text{доп}}$  – дополнительная заработная плата, рубли;

$k_{\text{доп}}$  – коэффициент дополнительной заработной платы (на стадии проектирования принимается равным 0,12 – 0,2);

$З_{\text{осн}}$  – основная заработная плата, рубли.

Расчет затрат на дополнительную заработную плату приведен в таблице 9.

Таблица 9. Затраты на дополнительную заработную плату

Исполнители	Зосн, руб.	Кдоп	Здоп, руб.
Студент	146495,47	0,12	17579,46
Научный руководитель	19585,84	0,12	2350,30
<b>Итого:</b>			19929,76

Итоговая сумма затрат на дополнительную заработную плату составила 19929,76 руб.

#### 4.6.5 Отчисления во внебюджетные фонды (страховые отчисления)

К отчислениям во внебюджетные фонды относятся отчисления:

- отчисления органам государственного социального страхования (ФСС);
- отчисления в пенсионный фонд (ПФ);
- отчисления медицинского страхования (ФФОМС).

Сумма отчислений во внебюджетные фонды рассчитывается на основе затрат на оплату труда исполнителей и может быть вычислена по формуле 11.

$$З_{\text{внеб}} = k_{\text{внеб}} \times (З_{\text{осн}} + З_{\text{доп}}), \quad (11)$$

где  $k_{внеб}$  – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и др.);

На 2014 г. в соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 30%. На основании пункта 1 ст.58 закона №212-ФЗ для учреждений, осуществляющих образовательную и научную деятельность в 2014 году, водится пониженная ставка – 27,1%.

Расчет затрат на отчисления во внебюджетные фонды приведен в таблице 10.

Таблица 10. Отчисления во внебюджетные фонды

Исполнители	Зосн, руб.	Здоп, руб.	Квнеб	Звнеб, руб.
Студент	146495,47	17579,46	0,271	44464,31
Научный руководитель	19585,84	2350,30	0,271	5944,69
<b>Итого:</b>				50409

Итоговая сумма отчислений во внебюджетные фонды составила 50409 руб.

#### 4.6.6 Накладные расходы

Накладные расходы – расходы на организацию, управление и обслуживание процесса производства товара, оказания услуги; носят комплексный характер. Накладные расходы вычисляются по формуле 12:

$$Z_{нкл} = \text{сумма статей}(1 - 5) \times k_{нр}, \quad (12)$$

где  $k_{нр}$  – коэффициент накладных расходов (16% от суммы затрат, подсчитанных выше).

Расчет накладных расходов приведен в таблице 11.

Таблица 11. Расчет накладных расходов

Статьи затрат	Сумма, руб.
Материальные затраты НТИ	0
Затраты на специальное оборудование	16680
Затраты на основную заработную плату	166081,31
Затраты на дополнительную заработную плату исполнителям проекта	19929,76
Затраты на отчисления во внебюджетные фонды	50409
Коэффициент накладных расходов	0,16
<b>Накладные расходы</b>	<b>40496,01</b>

Итоговая сумма накладных расходов составила 40496,01 руб.

#### **4.6.7 Формирование бюджета затрат научно-исследовательского проекта**

После того, как была подсчитана каждая из статей расходов, можно приступить к формированию общего бюджета затрат проекта. Итоговый бюджет затрат представлен в таблице 12.

Таблица 12. Расчет бюджета затрат НИИ

<b>Наименование статьи</b>	<b>Сумма, руб.</b>	<b>Удельный вес, %</b>
Материальные затраты НИИ	0	0
Затраты на специальное оборудование	16680	5,6
Затраты на основную заработную плату	166081,31	55,5
Затраты на дополнительную заработную плату	19929,76	6,7
Страховые взносы	56175,34	18,8
Накладные расходы	40496,01	13,5
<b>Общий бюджет</b>	<b>299362,42</b>	<b>100</b>

Таким образом, общий бюджет НИИ составляет 299362,42 рублей.

#### **4.7 Определение потенциального эффекта исследования**

Для определения эффективности работы необходимо рассчитать интегральный показатель эффективности научного исследования, нахождение которого связано с определением финансовой эффективности и ресурсоэффективности. При расчёте показателей эффективности оценивались 2 варианта исполнения системы:

1. Клиентская часть реализуется на фреймворке Flutter.
2. Клиентская часть реализуется на фреймворке React Native.

Стоимость разработки в этих двух вариантах является практически одинаковой, поэтому отдельный расчет для второго варианта не производился. Как следствие, интегральный финансовый показатель для обоих вариантов примем равным единице.

Существенным различием (и выгодным отличием первого варианта) является производительность и дальнейший потенциал развития первого варианта в будущем, в то время как второй вариант из-за технических особенностей вынужден будет столкнуться с ограниченной

производительностью, а дальнейшая поддержка, из-за падения интереса к фреймворку, повлечет более высокие затраты на дальнейшее развитие.

Расчет интегральных показателей ресурсоэффективности обоих вариантов исполнения приведен в таблице 13.

Таблица 13 - Сравнительная оценка характеристик вариантов исполнения проекта

Критерии \ Объект исследования	Весовой коэффициент параметра	Исп.1	Исп.2
1. Гибкость настройки	0,2	5	4
2. Понятность и удобство пользовательского интерфейса	0,2	5	5
3. Нагрузка на смартфон пользователя	0,2	4	3
4. Лёгкость масштабирования и дальнейшего развития	0,4	5	3
Итого	1	4,75	3,75

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных. Расчет приведен в таблице ниже:

Таблица 14 – Сравнительная эффективность разработки

№	Показатели	Исп.1	Исп.2
1	Интегральный финансовый показатель разработки	1	1
2	Интегральный показатель ресурсоэффективности разработки	4,75	3,75
3	Интегральный показатель эффективности	4,75	3,75
4	Сравнительная эффективность вариантов исполнения	1	0,79

Сравнив значения интегральных показателей эффективности, можно сделать вывод, что реализация программного обеспечения в первом исполнении является более эффективным вариантом решения задачи,

поставленной в данной работе с позиции финансовой и ресурсной эффективности.

### **Вывод по главе**

В рамках данного раздела был проведен комплексный SWOT-анализ проекта, который позволил выявить его сильные и слабые стороны, а также определить соответствие его характеристик внешним факторам. Более того, были выявлены возможные пути дальнейшего развития и совершенствования проекта для повышения его конкурентоспособности. Также была произведена оценка качества и перспективности данного проекта.

Ещё одним ключевым моментом, рассмотренным в данном разделе, является планирование работ, выполняемых в рамках проекта. Составленный список необходимых работ с распределением исполнителей, а также вычисленная трудоемкость, позволили построить наглядный план-график работ в виде диаграммы Ганта. Это позволило рационально распорядиться временными ресурсами, отведенными на выполнение проекта.

Кроме того, был определен бюджет проекта. Определены затраты на заработные платы исполнителям с учетом страховых отчислений, рассчитаны накладные расходы, а также амортизация оборудования, задействованного в процессе выполнения проекта.

Сравнение интегральных показателей эффективности вариантов исполнения показывает, что наиболее выгодным с позиции финансовой и ресурсной эффективности является первый вариант исполнения (реализация клиентской части на фреймворке Flutter), который и был реализован.

## **5 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ**

### **Введение**

Объектом исследования являются мобильное Flutter приложение для изучения иностранных языков.

Выпускная квалификационная работа представляет собой проектирование и реализацию сервиса для изучения иностранных языков в виде мобильного приложения.

Область применения: предоставление информации и инструментов для изучения иностранных языков онлайн. Проект позволяет решить проблему доступности языковых курсов и школ, и является актуальным, так как обладает большим количеством потенциальных пользователей, которыми являются люди практически любого возраста, желающие изучать иностранные языки, но не обладающими значительными финансовыми средствами.

Целью разработанного программного обеспечения является решение проблем повторения и запоминания иностранных слов, теории, а также малой практики.

Рабочий процесс в основном заключается в пребывании в офисе с размерами помещения: 5\*3 м в положение сидя, занимаясь проектированием и непосредственно разработкой программного обеспечения, взаимодействуя с оборудованием таким как ноутбук.

В данном разделе выпускной квалификационной работы рассмотрены опасные и вредные факторы, оказывающие влияние на производственную деятельность программиста. Также рассмотрены вопросы техники пожарной безопасности, правила поведения во время чрезвычайных ситуаций и способы предотвращения их возникновения.

### **5.1 Правовые и организационные вопросы обеспечения безопасности**

#### **5.1.1 Правовые нормы трудового законодательства**

Трудовые отношения между работодателем и работником регулируются с помощью законодательного акта «Трудовой кодекс

Российской Федерации» от 30.12.2001 N 197-ФЗ (ред. от 25.02.2022) [22]. В нём регламентируются права и обязанности работодателя и работника, вопросы организации труда, режим труда и отдыха, оплата и нормирование труда, компенсации работникам, защита персональных данных работника, урегулирование трудовых споров. Ниже приведены несколько наиболее важных положений:

- Нормальная продолжительность рабочего времени не должна превышать 40 часов в неделю.
- В течение рабочего дня (смены) работнику должен быть предоставлен перерыв для отдыха и питания продолжительностью не более двух часов и не менее 30 минут, который в рабочее время не включается. Правилами внутреннего трудового распорядка или трудовым договором может быть предусмотрено, что указанный перерыв может не предоставляться работнику, если установленная для него продолжительность ежедневной работы (смены) не превышает четырех часов (в ред. Федерального закона от 18.06.2017 N 125-ФЗ).
- Всем работникам предоставляются выходные дни (еженедельный непрерывный отдых).

### **5.1.2 Эргономические требования к правильному расположению и компоновке рабочей зоны**

Основными элементами рабочего места программиста являются: рабочий стол, рабочий стул (кресло), дисплей, клавиатура, мышь; вспомогательными – пюпитр, подставка для ног.

Согласно ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения» [23], рабочее место с дисплеем должно обеспечивать оператору возможность удобного выполнения работ в положении сидя и не создавать перегрузки костно-мышечной системы.

Регулируемая высота рабочей поверхности стола должна изменяться в пределах от 680 до 800 мм, а конструкция рабочего стола должна обеспечивать

возможность размещения на рабочей поверхности необходимого комплекта оборудования и документов с учетом характера выполняемой работы. Размеры рабочей поверхности стола должны быть: глубина - не менее 600 мм, ширина - не менее 1200 мм.

Дисплей на рабочем месте оператора должен располагаться так, чтобы изображение в любой его части было различимо без необходимости поднять или опустить голову. Дисплей на рабочем месте должен быть установлен ниже уровня глаз оператора. Угол наблюдения экрана оператором относительно горизонтальной линии взгляда не должен превышать 60°.

Клавиатура должна иметь возможность свободного перемещения. Клавиатуру следует располагать на поверхности стола на расстоянии от 100 до 300 мм от переднего края, обращенного к оператору, или на специальной регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.

Кресло оператора может быть с профилированными и непрофилированными элементами согласно ГОСТ 21889-76 «Система "Человек-машина". Кресло человека-оператора» [24]. Поверхность сиденья может быть плоской с наклоном 0-5°, или профилированной с углами наклона сиденья. Опорная плоскость сиденья может быть плоской или профилированной с радиусом кривизны поясничной опоры, равным 460 мм, радиусом изгиба для грудного отдела позвоночника, равным 620 мм и другими точками изгиба.

При размещении рабочих мест необходимо исключить возможность прямой засветки экрана источником естественного освещения.

При выполнении выпускной квалификационной работы правовых и организационных нарушений по указанным требованиям не было выявлено, рабочее место было оборудовано согласно всем нормам и правилам.

## 5.2 Производственная безопасность

В данном подразделе приведен анализ вредных и опасных факторов, которые воздействуют на разработчиков программного обеспечения, выполняющих работы на своих рабочих местах.

Все выявленные факторы приведены в таблице 15.

Таблица 15 – Возможные опасные и вредные производственные факторы на рабочем месте «офис»

Факторы (ГОСТ 12.0.003-2015)	Нормативные документы
1. Статические физические перегрузки, связанные с рабочей позой.	ГОСТ 12.0.003-2015 «Система стандартов безопасности труда. Опасные и вредные факторы» [25]
2. Умственное перенапряжение, в том числе вызванное информационной нагрузкой.	МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [26]
3. Перенапряжение анализаторов, в том числе вызванное информационной нагрузкой.	МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [26]
4. Монотонность труда, вызывающая монотонию.	МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [26]
5. Отсутствие или недостаток необходимого естественного освещения.	СП 52.13330.2016 «Естественное и искусственное освещение» [27]
6. Отсутствие или недостаток необходимого искусственного освещения.	СП 52.13330.2016 «Естественное и искусственное освещение» [27]
7. Производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий.	ГОСТ Р 12.1.019-2017 «Система стандартов безопасности труда (ССБТ). Электробезопасность. Общие требования и номенклатура видов защиты» [28]

Из данной таблицы следует, что на разработчиков программного обеспечения в ходе их деятельности воздействуют только физические и

психологические факторы, а химические и биологические факторы отсутствуют.

### **5.2.1 Статические физические перегрузки**

Работа программиста является малоподвижной деятельностью, а значит может вызывать гиподинамию. Как известно, ограничение физической активности приводит к нарушениям в работе опорно-двигательного аппарата, сердечно-сосудистой системы, желудочно-кишечного тракта. Все эти нарушения оказывают значительное влияние на состояние организма работника и негативно сказываются как на его повседневной жизни, так и на качестве его работы.

Для профилактики гиподинамии следует предпринимать следующие меры:

- увеличение физической активности;
- регулярные перерывы в работе для небольшой разминки.

### **5.2.2 Умственное перенапряжение и перенапряжение анализаторов, монотонность труда**

Монотонный труд является частью практически любого процесса разработки, так как каждый специалист выполняет ряд каких-либо действий, связанных с его задачами. Кроме того, программисты по роду своей деятельности проводят много времени, решая сложные и нестандартные задачи и проблемы, из-за чего может возникать умственное перенапряжение (переутомление). Этот негативный фактор играет важную роль в эмоциональном состоянии. При умственном перенапряжении у сотрудника могут наблюдаться следующие негативные последствия:

- нарушение сна или сонливость;
- отсутствие или повышение аппетита;
- приступы тошноты, обмороки;
- головная боль;
- раздражительность, нервозность;
- усталость, апатия;

- медлительность;
- снижение внимательности и работоспособности.

Для снижения уровня умственного перенапряжения разработчика программного обеспечения следует принимать следующие меры:

- во время рабочего дня делать регулярные перерывы;
- вне рабочего времени проводить время на свежем воздухе;
- нормализовать режим сна;
- регулярно и сбалансировано питаться;
- выделять время в течении дня на разминку или на полноценные тренировки;
- стараться организовывать свое рабочее время таким образом, чтобы не заниматься одной задачей на протяжении длительного времени;
- крупные задачи делить на небольшие подзадачи.

Согласно МР 2.2.9.2311-07 [26], меры профилактики стрессовых состояний предусматривают внедрение рациональных режимов труда и отдыха, комплекса оздоровительно-профилактических мероприятий для предупреждения воздействия стресс-факторов на организм работающих.

При постоянном взаимодействии с компьютером (набор текстов, ввод данных и т.п.), при исключении возможности периодического переключения на другие виды трудовой деятельности, не связанные с ПЭВМ, рекомендуется организация перерывов по 10 - 15 мин. через каждые 45 - 60 мин. работы.

Для повышения работоспособности, снижения зрительного утомления рекомендуется использовать очки защитные со спектральными фильтрами ЛС и НСФ, разрешенные Минздравом России для работы с ПЭВМ.

### **5.2.3 Отсутствие или недостаток необходимого естественного освещения**

Нехватка естественного освещения или его полное отсутствие в течение рабочего дня может повлечь болезни психологического характера, невротические расстройства, снизить работоспособность и концентрацию сотрудника.

Источником естественного освещения является солнечный свет, основными причинами его недостатка являются географическое положение здания, а также планировка его внутренних помещений, расположение рабочих мест. Окна в помещениях, где эксплуатируется вычислительная техника, преимущественно должны быть ориентированы на север и северо-восток. При этом столы следует располагать так, чтобы свет от окон падал на рабочую поверхность стола слева.

Разработка программного обеспечения относится к работам высокой точности (наименьший или эквивалентный объект различения 0,30 – 0,50 мм), разряд Б, подразряд 1, относительная продолжительность зрительной работы при направлении зрения на рабочую поверхность не менее 70%. Требования к естественному освещению рабочего помещения согласно СП 52.13330.2016 «Естественное и искусственное освещение» [27] представлены в таблице 16.

Таблица 16 – Требования к естественному освещению рабочего помещения

Коэффициент естественной освещенности, %, при	
верхнем или комбинированном освещении	боковом освещении
3,0	1,0

При недостаточной освещенности помещения может помочь расширение оконных проемов и установка качественных источников искусственного освещения.

#### **5.2.4 Отсутствие или недостаток необходимого искусственного освещения**

Недостаточная освещенность рабочей зоны является вредным производственным фактором, приводящим к быстрому утомлению и снижению работоспособности человека на предприятии. При недостаточной освещенности помещения человек быстрее устает, снижается внимание и концентрация. Продолжительная работа в условиях низкой освещенности приводит к ухудшению зрения.

Нормы естественного, искусственного и совместного освещения регламентируются СП 52.13330.2016 «Естественное и искусственное освещение» [27]. Разработка программного обеспечения относится к работам высокой точности (наименьший или эквивалентный объект различения 0,30 – 0,50 мм), разряд Б, подразряд 1, относительная продолжительность зрительной работы при направлении зрения на рабочую поверхность не менее 70%.

В таблице 17 представлены требования к искусственному освещению рабочего помещения.

Таблица 17 – Требования к искусственному освещению рабочего помещения

Искусственное освещение			
Освещенность на рабочей поверхности от системы общего освещения, лк	Цилиндрическая освещенность, лк	Объединенный показатель UGR, не более	Коэффициент пульсации освещенности, Кп, %, не более
400	100	19	15

Искусственное освещение в помещениях для эксплуатации ПК должно осуществляться системой общего равномерного освещения. В случаях преимущественной работы с документами, следует применять системы комбинированного освещения (к общему освещению дополнительно устанавливаются светильники местного освещения).

Яркий свет в зоне периферийного зрения заметно увеличивает напряжение глаз. Для снижения влияния вредного фактора недостаточной освещенности необходимо, чтобы уровень освещения рабочего пространства приблизительно совпадал с яркостью дисплея. Проблему недостаточной освещенности помещения можно решить при помощи установки дополнительных осветительных приборов.

### 5.2.5 Опасность поражения электрическим током

Работа программиста происходит в непосредственной близости от электрических сетей и приборов, поэтому работник должен с осторожностью

обращаться с электропроводкой и компьютером, а также помнить об опасности поражения электрическим током.

Покрытие полов следует делать из однослойного линолеума, что снизит величины зарядов статического электричества. Несмотря на то, что эти величины безопасны для здоровья человека, вычислительная техника подвергается опасности при воздействии зарядов такого рода.

Поражение электрическим током является опасным производственным фактором. Вероятность поражения повышается при:

- повышенной влажности воздуха в помещении (более 75%);
- высокой температуре воздуха и поверхностей (более 35 °С);
- наличии токопроводящей пыли;
- неверной проектировке рабочего места;
- отсутствии защитных конструкций для проводов;
- наличии посторонних предметов на электроприборах.

Мерами защиты от воздействия электрического тока при неисправности изоляции являются защитное заземление, зануление [28] и использование устройств защитного отключения (УЗО).

### **5.3 Экологическая безопасность**

Большое количество процессов, операций и материалов, используемых при изготовлении электронных средств, являются источниками огромного количества веществ, имеющих неблагоприятное воздействие на человека и биосферу. При изготовлении элементной базы, электронных изделий, при обработке, выращивании полупроводниковых кристаллов, при изготовлении интегральных схем, в процессе гальванического производства утилизация исходных материалов часто происходит с низким коэффициентом использования, огромное количество их идет в отходы, попадая в атмосферу, гидросферу, загрязняя почву. Таким образом, наряду с истощением природных запасов дефицитных материалов происходит загрязнение окружающей среды, что ведет к губительным последствиям для отдельных экосистем и биосферы в целом.

На данном рабочем месте выявлен предполагаемый источник загрязнения окружающей среды, а именно воздействие на литосферу в результате образования отходов при поломке предметов вычислительной техники и оргтехники.

Согласно ГОСТ Р 53692—2009, вышедшее из строя ПЭВМ и сопутствующая оргтехника относится к IV классу опасности и подлежит специальной утилизации.

Первым этапом является утилизация обезвреженных (инертных) отходов. Во время утилизации может быть произведена переработка бракованных или вышедших из употребления видов продукции, изделий, их составных частей и отходов от них путем разборки (разукрупнения), переплавки, использования других технологий с обеспечением рециркуляции (восстановления) органической и неорганической составляющих.

Вторым этапом является безопасное размещение отходов I—IV классов опасности на соответствующих полигонах или уничтожение [29].

#### **5.4 Безопасность в чрезвычайных ситуациях**

Возможные ЧС:

- Техногенные (взрывы, пожары, обрушение помещений).
- Природные (наводнения, ураганы, бури, природные пожары).
- Биологические (эпидемии, пандемии).
- Антропогенные (война, терроризм).

При работе с вычислительной техникой наиболее типичной чрезвычайной ситуацией является пожар, так как в современных ЭВМ очень высокая плотность размещения элементов электронных схем. К причинам электрического характера можно отнести короткое замыкание, искрение, статическое электричество.

Для предотвращения возникновения пожара необходимо:

- Регулярно проводить инструктажи по пожарной безопасности.
- Разместить в помещении план эвакуации и плакаты с краткой информацией о действиях, предпринимаемых при возникновении пожара.

- Соблюдать правила и нормы при монтаже электрических приборов и электрической проводки.

- Оборудовать помещение пожарной сигнализацией, а также средствами тушения пожара.

В случае возникновения пожара каждый сотрудник должен:

- незамедлительно сообщить об этом в пожарную охрану;
- принять меры по эвакуации людей, материальных ценностей согласно плану эвакуации;

- отключить электроэнергию, приступить к тушению пожара первичными средствами пожаротушения.

Возможный пожар на рабочем месте относится к классам А1, А2, Е [30]. Первичные средства пожаротушения являются: огнетушители порошковые переносные с порошками типа АВСЕ, огнетушители углекислотные.

### **Выводы по главе**

В результате работы по главе «Социальная ответственность» были выявлены основные нормативные акты для обеспечения безопасности жизнедеятельности на рабочем месте, рассмотрены наиболее значимые опасные и вредные факторы, возникающие при проектировании и разработке сервиса для изучения иностранных языков, описано влияние процесса разработки программного обеспечения на окружающую среду и меры, необходимые для уменьшения влияния вредных и опасных факторов на организм человека и для сокращения негативного влияния процесса разработки программного обеспечения на окружающую среду.

Согласно пункту 1.1.13 ПУЭ-7 рабочая зона является помещением без повышенной опасности. Согласно «Правилам по охране труда при эксплуатации электроустановок» [31] персонал должен иметь первую группу по электробезопасности.

Работа в офисе относится к категории тяжести труда Ia [32] – работы выполняются при оптимальных условиях внешней производственной среды и

при оптимальной величине физической, умственной и нервно-эмоциональной нагрузки.

Рабочая зона относится к категории В по взрывопожарной и пожарной опасности.

Все необходимые требования и нормы безопасности были соблюдены, нарушений по организации рабочего процесса при выполнении ВКР выявлено не было.

## **ЗАКЛЮЧЕНИЕ**

В ходе выполнения выпускной квалификационной работой были проанализированы существующие платформы и их возможности для конечного клиента, описаны проблемы в изучении иностранных языков и способы их решения, показаны преимущества выбранного фреймворка для разработки разрабатываемого проекта, описаны макеты и способы взаимодействия в приложении, спроектирована физическая схема базы данных, созданы основные экраны приложения, настроена работа с сервером, безопасность доступа к данным, авторизация пользователей.

Стек технологий (Flutter, Firebase), находящийся в основе проекта, позволил разработать приложение с возможностями как горизонтального, так и вертикального масштабирования, что позволит занять определенную нишу на конкурентном рынке мобильных приложений, предназначенных для изучения иностранных языков.

В дальнейшем планируется доработать приложение для выхода на рынок, добавить качественный и продуманный языковой курс по английскому языку, добавить другие языки и новые функции.

## СПИСОК ЛИТЕРАТУРЫ

1. Результаты исследования важности знания иностранных языков в России [Электронный ресурс]. – Режим доступа: <https://iom.anketolog.ru/2018/06/18/inostrannye-yazyki>, свободный (дата обращения: 17.01.2022).
2. An Investigation of the Difficulties Faced by EFL Undergraduates in Speaking Skills [Электронный ресурс]. – Режим доступа: <https://iom.anketolog.ru/2018/06/18/inostrannye-yazyki>, свободный (дата обращения: 17.01.2022).
3. Кривая забывания [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/Кривая\\_забывания](https://ru.wikipedia.org/wiki/Кривая_забывания), свободный (дата обращения: 17.01.2022).
4. Примерная рабочая программа основного общего образования по английскому языку (для 5–9 классов образовательных организаций) – Режим доступа: [https://fgosreestr.ru/uploads/files/6897ed6441a220ff20a73833d4395afa.pdf#:~:text=Учебный%20предмет%20«Иностранный%20\(английский\)%20язык»,с%20%20по%209%20класс](https://fgosreestr.ru/uploads/files/6897ed6441a220ff20a73833d4395afa.pdf#:~:text=Учебный%20предмет%20«Иностранный%20(английский)%20язык»,с%20%20по%209%20класс) (дата обращения 17.01.2022).
5. Федеральный закон от 29 декабря 2012 г. N 273-ФЗ "Об образовании в Российской Федерации" (с изменениями на 16 апреля 2022 года) [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/902389617> (дата обращения 17.01.2022).
6. Average number of primary and secondary education students per class in Russia in school year 2020/2021, by school type [Электронный ресурс]. – Режим доступа: <https://www.statista.com/statistics/1130053/average-school-size-in-russia/#:~:text=Средний%20размер%20класса%20начального%20образования, государственных%20учреждениях%20составляло%2017%20человек> (дата обращения 17.01.2022).

7. The Five-Stage Model of Adult Skill Acquisition [Электронный ресурс]. – Режим доступа: <https://journals.sagepub.com/doi/pdf/10.1177/0270467604264992> (дата обращения 17.01.2022).
8. Способы запоминания иностранных слов [Электронный ресурс]. – Режим доступа: [https://advanceclub.ru/knowledge/articles/kak\\_luchshe\\_vsego\\_zapominat\\_inostrannye\\_slova/](https://advanceclub.ru/knowledge/articles/kak_luchshe_vsego_zapominat_inostrannye_slova/) (дата обращения 26.01.2022).
9. Интервальные повторения [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/Интервальные\\_повторения#:~:text=Интервальные%20повторения%20\(англ.%20spaced%20repetition\),получил%20при%20изучении%20иностраных%20языков, свободный](https://ru.wikipedia.org/wiki/Интервальные_повторения#:~:text=Интервальные%20повторения%20(англ.%20spaced%20repetition),получил%20при%20изучении%20иностраных%20языков, свободный) (дата обращения: 26.01.2022).
10. A Comparison of Undergraduate Students' English Vocabulary Learning: Using Mobile Phones and Flash Cards [Электронный ресурс]. – Режим доступа: <https://eric.ed.gov/?id=EJ898010> (дата обращения 26.01.2022).
11. Duolingo [Электронный ресурс]. – Режим доступа: <https://www.duolingo.com> (дата обращения 26.01.2022).
12. Lingualeo [Электронный ресурс]. – Режим доступа: <https://lingualeo.com> (дата обращения 26.01.2022).
13. Mobile Operating System Market Share Worldwide [Электронный ресурс]. – Режим доступа: <https://gs.statcounter.com/os-market-share/mobile/worldwide> (дата обращения 26.01.2022).
14. Сравнение поисковых запросов кроссплатформенных фреймворков [Электронный ресурс]. – Режим доступа: <https://trends.google.com/trends/explore?date=today%205-y&q=Flutter,React%20Native,Xamarin> (дата обращения 26.01.2022).
15. Flutter [Электронный ресурс]. – Режим доступа: <https://flutter.dev/> (дата обращения 26.01.2022).

16. Firebase [Электронный ресурс]. – Режим доступа: <https://firebase.google.com/> (дата обращения 26.01.2022).
17. Cloud Firestore [Электронный ресурс]. – Режим доступа: <https://firebase.google.com/docs/firestore> (дата обращения 26.01.2022).
18. Документация Lingvo API [Электронный ресурс]. – Режим доступа: <https://developers.lingvolive.com/ru-ru/Help>, свободный (дата обращения: 26.01.2022).
19. API Яндекс Словаря [Электронный ресурс]. – Режим доступа: <https://yandex.ru/dev/dictionary/>, свободный (дата обращения: 26.01.2022).
20. Пользовательское соглашение сервиса «API Яндекс.Словарь» [Электронный ресурс]. – Режим доступа: [https://yandex.ru/legal/dictionary\\_api/](https://yandex.ru/legal/dictionary_api/), свободный (дата обращения: 26.01.2022).
21. Flutter InAppWebView Documentation [Электронный ресурс]. – Режим доступа: <https://inappwebview.dev/docs/> (дата обращения 18.02.2022).
22. «Трудовой кодекс Российской Федерации» от 30.12.2001 N 197-ФЗ (ред. от 25.03.2022).
23. ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200025975> (дата обращения: 25.03.2022).
24. ГОСТ 21889-76 «Система "Человек-машина". Кресло человека-оператора» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200012832> (дата обращения: 25.03.2022).
25. ГОСТ 12.0.003-2015 «Система стандартов безопасности труда. Опасные и вредные факторы» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200136071> (дата обращения: 25.03.2022).
26. МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200072234> (дата обращения: 25.03.2022).

27. СП 52.13330.2016 «Естественное и искусственное освещение» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/456054197> (дата обращения: 25.03.2022).

28. ГОСТ Р 12.1.019-2017 «Система стандартов безопасности труда (ССБТ). Электробезопасность. Общие требования и номенклатура видов защиты» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200161238> (дата обращения: 25.03.2022).

29. ГОСТ Р 53692-2009 «Ресурсосбережение. Обращение с отходами. Этапы технологического цикла отходов» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200081740> (дата обращения: 25.03.2022).

30. Федеральный закон от 22.07.2008 N 123-ФЗ (ред. от 30.04.2021) «Технический регламент о требованиях пожарной безопасности» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/902111644> (дата обращения: 25.03.2022).

31. Приказ Минтруда России от 15 декабря 2020 года N 903н «Об утверждении Правил по охране труда при эксплуатации электроустановок» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/573264184> (дата обращения: 25.03.2022).

32. Приказ Минтруда России от 24.01.2014 N 33н (ред. от 27.04.2020) «Об утверждении Методики проведения специальной оценки условий труда, Классификатора вредных и (или) опасных производственных факторов, формы отчета о проведении специальной оценки условий труда и инструкции по ее заполнению» [Электронный ресурс]. – Режим доступа: [http://www.consultant.ru/document/cons\\_doc\\_LAW\\_158398/](http://www.consultant.ru/document/cons_doc_LAW_158398/) (дата обращения: 25.03.2022).