

Инженерная школа информационных технологий и робототехники
 Направление подготовки: 09.03.04 «Программная инженерия»
 Отделение информационных технологий

БАКАЛАВРСКАЯ РАБОТА

Тема работы
Проектирование AR-приложения мгновенного перевода

УДК 004.912:81'322.4

Студенты

Группа	ФИО	Подпись	Дата
8K51	Каракулов Александр Андреевич		
8K51	Тогызбаев Тимур Нургалиевич		
8K51	Сухарев Семен Игоревич		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР ТПУ	Чердынцев Евгений Сергеевич	к.т.н.		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСТН ШБИП ТПУ	Подопригора Игнат Валерьевич	к.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД ШБИП ТПУ	Винокурова Галина Федоровна	к.т.н.		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР ТПУ	Чердынцев Евгений Сергеевич	к.т.н.		

**Планируемые результаты обучения по направлению 09.03.04
«Программная инженерия»**

Код результата	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	Использовать базовые и специальные знания в области проектного менеджмента для ведения комплексной инженерной деятельности.
P8	Владеть иностранным языком на уровне, позволяющем работать в иноязычной среде, разрабатывать документацию, презентовать и защищать результаты комплексной инженерной деятельности.
P9	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.
P10	Демонстрировать знания правовых, социальных, экономических и культурных аспектов комплексной инженерной деятельности.
P11	Демонстрировать способность к самостоятельному обучению в течение всей жизни и непрерывному самосовершенствованию в инженерной профессии.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа информационных технологий и робототехники
Направление подготовки: 09.03.04 «Программная инженерия»
Отделение информационных технологий

УТВЕРЖДАЮ:
Руководитель ООП

(Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

В форме:

бакалаврской работ

Студентам:

Группа	ФИО
8К51	Каракулову Александру Андреевичу
8К51	Тогызбаеву Тимуру Нургалиевичу
8К51	Сухареву Семену Игоревичу

Тема работы:

Алгоритмическое и программное обеспечение поиска и анализа данных социальных сетей	
Утверждена приказом директора (дата, номер)	№1513/с от 26.02.2019

Срок сдачи студентом выполненной работы:

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

Исходные данные к работе <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>	Объектом проектирования в данной работе является AR-приложение мгновенного перевода; Режим работы: непрерывный; Особые требования к продукту: кроссплатформенность;
Перечень подлежащих исследованию, проектированию и разработке вопросов <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>	<ol style="list-style-type: none">1. Исследование предметной области;2. Аналитический обзор существующих средств для перевода текста;3. Проектирование информационной системы AR-перевода;4. Разработка информационной системы для перевода в дополненной реальности;5. Финансовый менеджмент;6. Социальная ответственность.

<p>Перечень графического материала <i>(с точным указанием обязательных чертежей)</i></p>	<ol style="list-style-type: none"> 1. Модель работы программы (Диаграмма в нотации IDEF0) 2. Диаграмма EPC 3. Диаграмма BPMN процесса обработки и визуализации данных 4. Диаграмма вариантов использования 5. Рисунки, демонстрирующие результаты 6. Диаграмма Ганта
---	--

Консультанты по разделам выпускной квалификационной работы
(с указанием разделов)

Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Подопригора Игнат Валерьевич
Социальная ответственность	Винокурова Галина Федоровна

Названия разделов, которые должны быть написаны на русском и иностранном языках:

--	--

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

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР ТПУ	Чердынцев Евгений Сергеевич	к.т.н.		

Задание приняли к исполнению студенты:

Группа	ФИО	Подпись	Дата
8К51	Каракулов Александр Андреевич		
8К51	Тогызбаев Тимур Нургалиевич		
8К51	Сухарев Семен Игоревич		

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа информационных технологий и робототехники
Направление подготовки: 09.03.04 «Программная инженерия»
Уровень образования бакалавриат
Отделение информационных технологий
Период выполнения осенний/весенний семестр 2018/2019 учебного года

Форма представления работы:

бакалаврская работа

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	
--	--

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
	Основная часть	60
	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	20
	Социальная ответственность	20

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Чердынцев Евгений Сергеевич	к.т.н.		

СОГЛАСОВАНО:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР ТПУ	Чердынцев Евгений Сергеевич	к.т.н.		

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

Студенту:

Группа	ФИО
8К51	Каракулову Александру Андреевичу
8К51	Тогызбаеву Тимуру Нурғалиевичу
8К51	Сухареву Семену Игоревичу

Школа	ИШИТР	Отделение школы (НОЦ)	ОИТ
Уровень образования	Бакалавриат	Направление/специальность	09.03.04 Программная инженерия

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

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	Амортизационные затраты на спецоборудование – 14 394 рубля;
2. Нормы и нормативы расходования ресурсов	Затраты на основную и дополнительную з/п – 387949 + 58192,4 рубля;
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	Затраты на отчисление во внебюджетные фонды – 124921 рубль; Накладные расходы – 93673 рубля.

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения	Описание потенциальных потребителей; Анализ технических конкурентных решений; SWOT-анализ.
2. Планирование и формирование бюджета научных исследований	Структура работ в рамках научного исследования Определение трудоемкости выполнения работ и разработка графика проведения научного исследования Бюджет проекта
3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования	Определение интегрального финансового показателя разработки Определение интегрального показателя ресурсоэффективности разработки Определение интегрального показателя эффективности

Перечень графического материала (с точным указанием обязательных чертежей):

1. Оценка конкурентоспособности технических решений
2. Матрица SWOT
3. График проведения и бюджет НИ
4. Оценка ресурсной, финансовой и экономической эффективности НИ

Дата выдачи задания для раздела по линейному графику

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСТН ШБИП ТПУ	Подопригора Игнат Валерьевич	Кандидат экономических наук		

Задание принял к исполнению студенты:

Группа	ФИО	Подпись	Дата
8K51	Каракулов Александр Андреевич		
8K51	Тогызбаев Тимур Нургалиевич		
8K51	Сухарев Семен Игоревич		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студентам:

Группа	ФИО
8К51	Каракулову Александру Андреевичу
8К51	Тогызбаеву Тимуру Нурғалиевичу
8К51	Сухареву Семену Игоревичу

Школа	ИШИТР	Отделение (НОЦ)	ОИТ
Уровень образования	Бакалавриат	Направление/специальность	09.03.04 Программная инженерия

Тема ВКР:

Проектирование AR-приложения мгновенного перевода	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Объект исследования – приложение для перевода текста в дополненной реальности. Рабочее место – рабочий стол с персональным компьютером в учебной аудитории.
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Правовые и организационные вопросы обеспечения безопасности: <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<ul style="list-style-type: none"> – Рабочее место при выполнении работ сидя регулируется ГОСТом 12.2.032 –78 – Организация рабочих мест с электронно-вычислительными машинами регулируется СанПиНом 2.2.2/2.4.1340 – 03 – Использование персональных данных пользователей регулируется Федеральным законом №152 «О персональных данных»
2. Производственная безопасность: 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия	<ul style="list-style-type: none"> – Отклонение показателей микроклимата – Отсутствие или недостаток естественного света – Недостаточная освещенность рабочей зоны – Повышенный уровень электромагнитных излучений
3. Экологическая безопасность:	Анализ негативного воздействия на окружающую природную среду: утилизация компьютеров, смартфонов, оргтехники.
4. Безопасность в чрезвычайных ситуациях:	Возможные чрезвычайные ситуации: <ul style="list-style-type: none"> ● Пожар ● Террористический акт
Дата выдачи задания для раздела по линейному графику	

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД ШБИП ТПУ	Винокурова Галина Федоровна	Кандидат технических наук, доцент		

Задание приняли к исполнению студенты:

Группа	ФИО	Подпись	Дата
8K51	Каракулов Александр Андреевич		
8K51	Тогызбаев Тимур Нургалиевич		
8K51	Сухарев Семен Игоревич		

Реферат

Выпускная квалификационная работа: 98 страниц, 33 рисунка, 18 таблиц, 5 источников, 2 приложения.

Ключевые слова: дополненная реальность, кроссплатформенность, перевод, JavaScript.

Объектом исследования является – применение инструментов дополненной реальности в переводе текста.

Цель работы – разработка приложения перевода текста в дополненной реальности с возможностью дальнейшего обучения.

В процессе исследования проводился аналитический обзор существующих средств перевода текста с помощью программных средств, были выявлены и визуализированы результаты.

В результате исследования было выявлено, что большинство существующих программных средств и инструментов не обладают возможностью перевода в дополненной реальности, а также не имеют возможности дальнейшего изучения переведенных слов.

Экономическая эффективность работы в актуальности поставленной задачи, а также в новом подходе к переводу, при помощи программных средств.

В дальнейшем планируется расширить возможности обучения, посредством разработки и добавления новых заданий, а также добавить другие языки перевода в приложение.

Список терминов и сокращений

В данной работе используются следующие термины:

Дополненная реальность – воспринимаемая смешанная реальность, создаваемая с помощью компьютера с использованием «дополненных» элементов воспринимаемой реальности, когда реальные объекты монтируются в поле восприятия[1].

Кроссплатформенность – способность программного обеспечения работать с двумя и более аппаратными платформами и (или) операционными системами.

Методология IDEF0 – метод описания и формализации бизнес-процессов.

Проектирование информационной системы – это поиск способа, удовлетворяющего требованиям функциональности системы имеющимися средствами с учетом заданных ограничений.

Событийная цепочка процессов (EPC) – тип блок-схемы, используемой для бизнес-моделирования.

BPMN (The Business Process Modeling/Management Notation) – система условных обозначений для построения схемы протекания бизнес-процессов.

Нормативные ссылки

В данной работе используются ссылки на следующие стандарты:

ГОСТ 12.0.003-74. ССБТ. Опасные и вредные производственные факторы. Классификация.

ГОСТ 12.1.045–84 ССБТ. Электростатические поля. Допустимые уровни на рабочих местах и требования к проведению контроля.

ГОСТ 12.1.038–82 ССБТ. Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов.

ГОСТ 12.1.004-91 ССБТ. Пожарная безопасность. Общие требования.

ГОСТ 12.1.010-76 ССБТ. Взрывобезопасность. Общие требования.

ГОСТ 19.402-78. ЕСПД. Описание программы.

Содержание

Введение	15
Глава 1. Исследование предметной области	17
1.1. Историческая справка	17
1.2. Описание проблемы	22
1.3. Сравнение с аналогами	25
Глава 2. Проектирование информационной системы	27
2.1. Проектирование потоков данных.	31
2.2. Проектирование процессов внутри ИС.....	32
Глава 3. Разработка информационной системы перевода в дополненной реальности в режиме реального времени.....	35
3.1. Выявление нефункциональных требований к информационной системе ..	35
3.2. Выявление ролей и вариантов использования для информационной системы. 35	
3.3. Выбор средств разработки.....	36
3.4. Системные требования к информационной системе.	38
3.5. Распределение ответственности.....	39
3.6. Разработка приложения	40
3.6.1. Алгоритмические решения.....	40
3.6.2. Решения при работе с данными	44
3.6.3. Визуальная часть, интерфейс	50
3.7. Демонстрация работы приложения	57
Глава 4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение ...	66
4.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения	66

4.1.1. Потенциальные потребители результатов исследования	66
4.1.2. Анализ конкурентных технических решений	67
4.1.3. SWOT-анализ	69
4.2. Структура работ в рамках научного исследования	70
4.2.1. Определение трудоемкости выполнения работ	72
4.2.2. Разработка графика проведения научного исследования	76
4.3. Бюджет научно-технического исследования (НТИ)	76
4.3.1. Расчет затрат на специальное оборудование для научных (экспериментальных) работ	76
4.3.2. Основная заработная плата исполнителей темы	77
4.3.3. Дополнительная заработная плата исполнителей темы	78
4.3.4. Отчисления во внебюджетные фонды (страховые отчисления)	79
4.3.5. Накладные расходы	79
4.3.6. Формирование бюджета затрат научно-исследовательского проекта ..	79
4.4. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования	80
Глава 5. Социальная ответственность	83
5.1. Правовые и организационные вопросы обеспечения безопасности	84
5.1.1. Организационные мероприятия обеспечения безопасности	84
5.1.2. Особенности законодательного регулирования проектных решений ...	86
5.2. Производственная безопасность	87
5.2.1. Анализ опасных и вредных производственных факторов	88
5.2.2. Обоснование мероприятий по снижению уровней воздействия опасных и вредных факторов на исследователя (работающего)	89
5.3. Экологическая безопасность	90

5.4. Безопасность в чрезвычайных ситуациях	91
5.4.1. Пожар.....	91
5.4.2. Теракт	92
Вывод по разделу «Социальная ответственность».....	92
Заключение.....	93
Список используемых источников	94
Приложение А. Листинг программы.....	95
Приложение Б. Отчет по покрытию типами проекта	96

Введение

В современном мире информационные технологии получили широкое применение в различных сферах деятельности человека. Однако использование телефона с течением времени стало не только производственной необходимостью, но и бытовой.

Разработка переводчиков стала актуальным направлением мобильной разработки в связи с широкой распространенностью иностранных слов.

В наши дни существует множество разнообразных переводчиков, с помощью которых можно получить перевод, например:

- Google Translate;
- Promt;
- Spharus;
- Multitran;
- ABBY Lingua

Сферы применения электронных переводчиков различны, начиная от бытовых переводов заканчивая сложными системными переводами технических текстов. Далее приведен ряд сфер деятельности, где наиболее часто становится необходимым обращение к электронным системам перевода:

- IT;
- Лингвистика;
- Переводоведение;
- Производство;
- Медицина;

Целью данной работы является создание информационной системы для перевода текста в дополненной реальности в реальном времени.

Для достижения данной цели требуется выполнить несколько задач:

- Проанализировать процесс создания перевода стандартным способом.
- Выявить этапы, которые можно провести автоматически.

- Разработать архитектуру приложения
- Провести разработку информационной системы.

Глава 1. Исследование предметной области

1.1. Историческая справка

Машинный перевод – процесс перевода текстов (письменных, а в идеале и устных) с одного естественного языка на другой с помощью специальной компьютерной программы. Так же называется направление научных исследований, связанных с построением подобных систем.

История машинного перевода начинается в 1933 году. Советский ученый Пётр Троянский обращается в Академию наук СССР с изобретённой им «машиной для подбора и печатания слов при переводе с одного языка на другой». Машина была крайне проста: большой стол, печатная машинка с лентой и плёночный фотоаппарат. На столе лежали карточки со словами и их переводами на четырёх языках. Оператор брал первое слово из текста, находил карточку с ним, фотографировал её, а на печатной машинке набирал его морфологическую информацию: существительное, множественное число, родительный падеж.

Клавиши машины были модифицированы для удобства, каждая однозначно кодировала одно из свойств. Лента печатной машинки и плёнка камеры подавались параллельно, на выходе формируя набор кадров со словами и их морфологией.

Полученная лента отдавалась знающим конкретные языки лингвистам, которые превращали набор фотографий в связный литературный текст. Чтобы переводить тексты, как оператору, так и лингвистам требовалось знать только свой родной язык.

Идеи машинного перевода на основе правил начали появляться ещё в 1970-х годах. Учёные анализировали работу лингвистов-переводчиков и пытались запрограммировать компьютеры делать то же самое. Их системы состояли из:

Двуязычного англо-русского словаря.

Набора лингвистических правил под каждый язык (существительные женского рода, оканчивающиеся на -а/-я).

По желанию они дополнялись списками имён, корректорами орфографии и транслитераторами.

Системы дословного перевода (Direct Machine Translation)

Это самый простой способ машинного перевода, понятный любому человеку: Деление текста по словам, перевод каждого слова, исправление морфологии, согласование падежей, окончаний и остального синтаксиса.

Специально обученные лингвисты пишут правила под каждое слово. На выходе получается перевод. В современных системах подход не используется вообще.

Трансферные системы (Transfer-based Machine Translation)

Трансферные системы предусматривают предварительную подготовку. Суть алгоритма: разбор текста на подлежащее, сказуемое, поиск определения и остальных синтаксических конструкций. После этого система оперирует не отдельными словами, а целыми конструкциями.

В теории возможна даже конвертация порядка слов в языках. На практике всё ещё перевод получается фактически дословный. С одной стороны, проще: система позволяет задать общие правила согласования по роду и падежу. С другой – сложнее: сочетаний слов намного больше, чем самих слов. Каждый вариант невозможно учесть вручную.

Интерлингвистические системы (Interlingua Machine Translation)

Исходное предложение полностью конвертируется в некое промежуточное представление, единое для всех языков мира – интерлингва (interlingua), упомянутую ранее Декартом. Интерлингва – специальный метаязык, правила которого едины и покрывают все языки мира, тем самым превращая перевод в техническую задачу.

Специальные парсеры затем конвертируют interlingua в нужный язык и получается сингулярность. Часто interlingua путают с трансферными системами, ведь там тоже есть конвертация. Однако в трансферных системах правила конвертации пишутся под два конкретных языка, а в интерлингвистической между каждым языком и interlingua. Добавив в интерлингвистическую систему третий язык, существует возможность переводить между всеми тремя, а в трансферной – нет.

Создать универсальную интерлингву вручную оказалось крайне сложно. Некоторые учёные потратили всю жизнь пытаясь это сделать. Ничего не получилось,

однако благодаря им появились методы морфологического, синтаксического и иногда семантического анализа. Но сама идея промежуточного языка еще вернётся позже. А именно через лет. Как можно заметить, все RBMT непрактичны и необучаемы, поэтому сейчас редко используются. Разве что в специфических местах вроде перевода метеосводок. Среди плюсов RBMT отмечают морфологическую точность (не путает слова), воспроизводимость (все переводчики получают одинаковый результат) и возможность специализации под предметную область (обучить специальным терминам экономистов или программистов).

Даже если представить, что учёным удалось бы создать идеальную RBMT, а лингвистам заложить в неё все правила правописания, проект все равно бы ждал провал ввиду такого лингвистического явления как исключения. Неправильные глаголы в английском, плавающие приставки в немецком, суффиксы в русском, и просто ситуации, когда правила говорения отличаются от устойчивых языковых норм.

Статистический машинный перевод – Statistical Machine Translation (SMT)

На рубеже 1990 года в исследовательском центре IBM впервые показали систему машинного перевода, которая не взаимодействовала с правилами и лингвистикой. Учёные загрузили в компьютер очень много одинаковых текстов на двух языках и заставили его разбираться в закономерностях самому.

Помимо улучшения точности, перевод по фразам дал больше свободы в поиске двуязычных текстов для обучения. Для Word-based перевода было очень важно точное соответствие переводов, что исключало любые литературные или вольные переводы. Phrase-based прекрасно обучался даже на них.

С 2006 года этот подход начали использовать все. Google, «Яндекс», Bing и другие качественные онлайн-переводчики работали именно как Phrase-based до 2016-го года.

Если Rule-based подход стабильно давал хоть и ужасный, но предсказуемый результат, то статистические методы иногда удивляли и озадачивали. Одно время

Google Translate, когда он переводил «three hundred» как «300». Эту ошибку назвали статистическими аномалиями.

Phrase-based перевод стал настолько популярным, что когда вы слышите «статистический машинный перевод», скорее всего, имеется в виду именно он. Вплоть до 2016 года во всех исследованиях Phrase-based перевод называли the state-of-art.

Статистический перевод на основе синтаксиса – Syntax-based SMT

До прихода нейросетей про синтаксический перевод многие годы говорили как про «будущее переводчиков», но достичь успеха он так и не успел.

Будущее заключалось в объединении подходов SMT и старого трансферного перевода по правилам. Нужно было научиться делать достаточно точный синтаксический разбор предложения – определять подлежащее, сказуемое, зависимые члены, а затем построить дерево.

Имея такое дерево, можно обучить машину правильно конвертировать фигуры одного языка в фигуры другого, выполняя остальной перевод по словам или фразам с помощью машинного обучения. В теории это решило бы проблему порядка слов навсегда.

Трудность в том, что хоть человечество и считает проблему синтаксического разбора давно решённой (для многих языков есть готовые библиотеки), по факту он работает весьма плохо.

Нейронный машинный перевод – Neural Machine Translation (NMT)

В 2016 году Google анонсировал идею применения глубокого машинного обучения в переводах. Идея была похожа на перенос стиля между фотографиями: первая нейросеть умеет только кодировать предложение в набор цифр-характеристик, а вторая – только декодировать их обратно в текст. Обе «не знают» друг о друге, каждая знает только свой язык.

Поиск характеристик осуществлялся при помощи технологий глубокого обучения. Главное отличие глубокого обучения от классических нейросетей как раз

и было в том, что его сети обучаются находить характерные свойства объектов, не понимая их природы.

Возник вопрос определения типа нейросети используемой в кодере и декодере. Для картинок отлично подходят свёрточные нейросети (CNN), потому что они работают с независимыми блоками пикселей. Но в тексте не бывает независимых блоков, каждое следующее слово зависит от предыдущих и последующих. Текст, речь и музыка всегда последовательны.

Для такой обработки подходят рекуррентные нейросети (RNN), они помнят предыдущий результат. В нашем случае это предыдущие слова в предложении. RNN сейчас применяют в распознавание речи в Siri, в подсказках слов на клавиатуре, генерации музыки, в чат-ботах.

За два года нейросети превзошли всё, что было придумано в переводе за последние 20 лет.[2] Нейронный перевод делал на 50% меньше ошибок в порядке слов, на 17% меньше лексических и на 19% грамматических ошибок. Нейросети даже научались сами согласовывать род и падежи в разных языках, хотя никто их этому не учил.

Самые заметные улучшения были там, где никогда не существовало прямого перевода. Методы статистического перевода всегда работали через английский язык. Если переводили, например, с русского на немецкий, машина сначала переводила текст на английский, а только потом переводила на немецкий. Двойные потери. Нейронному переводу это не нужно. Впервые стало возможно напрямую переводить между языками, у которых не было ни одного общего словаря. [3]

«Яндекс.Переводчик» (2017)

«Яндекс» запустил свой нейросетевой перевод в 2017 году. Главным отличием была заявлена гибридность. Переводчик «Яндекса» переводит предложение сразу двумя методами – статистическим и нейросетевым, а потом с помощью алгоритма CatBoost находит наиболее подходящий.

Дело в том, что нейронный перевод плохо справляется с короткими фразами. Когда необходимо перевести словосочетание вроде «синяя бетономешалка»,

нейросети могут допустить ошибку, а простой статистический перевод найдёт оба слова быстро и без проблем.

1.2. Описание проблемы

Одним из этапов создания переводчика с наличием возможности перевода в режиме реального времени является наполнение базы данных словаря. Это трудоёмкий процесс, требующий слаженной работы специалистов разного профиля. Данный процесс можно разрабатывать самостоятельно, а можно воспользоваться API сервисом, что я и сделал в процессе написания информационной системы для перевода текста в дополненной реальности в режиме реального времени, так как основной проблемой, которую я решал являлась не скудная наполненность словаря, а большие временные затраты на перевод с использованием систем электронного перевода, а также неоправданно большое необходимое количество выполняемых действий. Эти две большие проблемы являются следствием и совокупностью других, менее значительных по отдельности, проблем:

1. Необходимость понимания структуры исходного текста.

Очень часто случается так, что слова в связке друг с другом приобретают несколько иное значение, чем то, что они имеют в отсутствии контекста. Например, слово look является фразовым глаголом, что означает, что в английском языке к нему требуется предлог. Таким образом, добавление к слову look предлога for, меняет изначальный смысл слова (смотреть) на новый (искать).

2. Затрачиваются большие временные ресурсы на внесение текста в переводчик.

Данная проблема обуславливается не только не оптимизированным и непонятным для пользователя дизайном, но и ошибками самого пользователя. Например, расположенные на клавиатуре рядом стоящие буквы «f» и «g». Очень часто встречающиеся взаимные опечатки этих букв, чему способствует маленькие размеры экрана телефона, могут приводить к непоправимым ошибкам, вроде слов

«fun» и «gun». Казалось бы, дружелюбное «I have a fun» (я веселюсь), в диалоге с носителем английского языка может превратиться в угрожающее «I have a gun» (у меня есть пушка).

Также, проблемой для пользователя, находящегося в незнакомой лингвистической среде может быть выбор нужного языка перевода. Например, во французском и английских языках используются стандартные буквы латинского алфавита и человек, который не разбирается в различиях этих двух языков романо-германской группы, вряд ли сможет правильно идентифицировать язык, особенно в отсутствии контекста.

3. Интерфейс многих переводчиков сложен и требует нескольких действий для получения перевода.

Например, для перевода в Promt Mobile прежде чем получить перевод требуется произвести несколько действий:

1. Выбор исходного языка
2. Выбор конечного языка
3. Выбор поля для ввода
4. Ручной ввод нужного текста
5. Нажатие кнопки «перевести»

Все эти проблемы наглядно демонстрирует диаграмма Исикавы т.н. диаграмма «рыбьей кости» (англ. Fishbone Diagram), или «причинно-следственная» диаграмма (англ. Cause and Effect Diagram), известная также как диаграмма «анализа корневых причин». Подробный анализ, представленный в виде этой самой диаграммы представлен ниже:

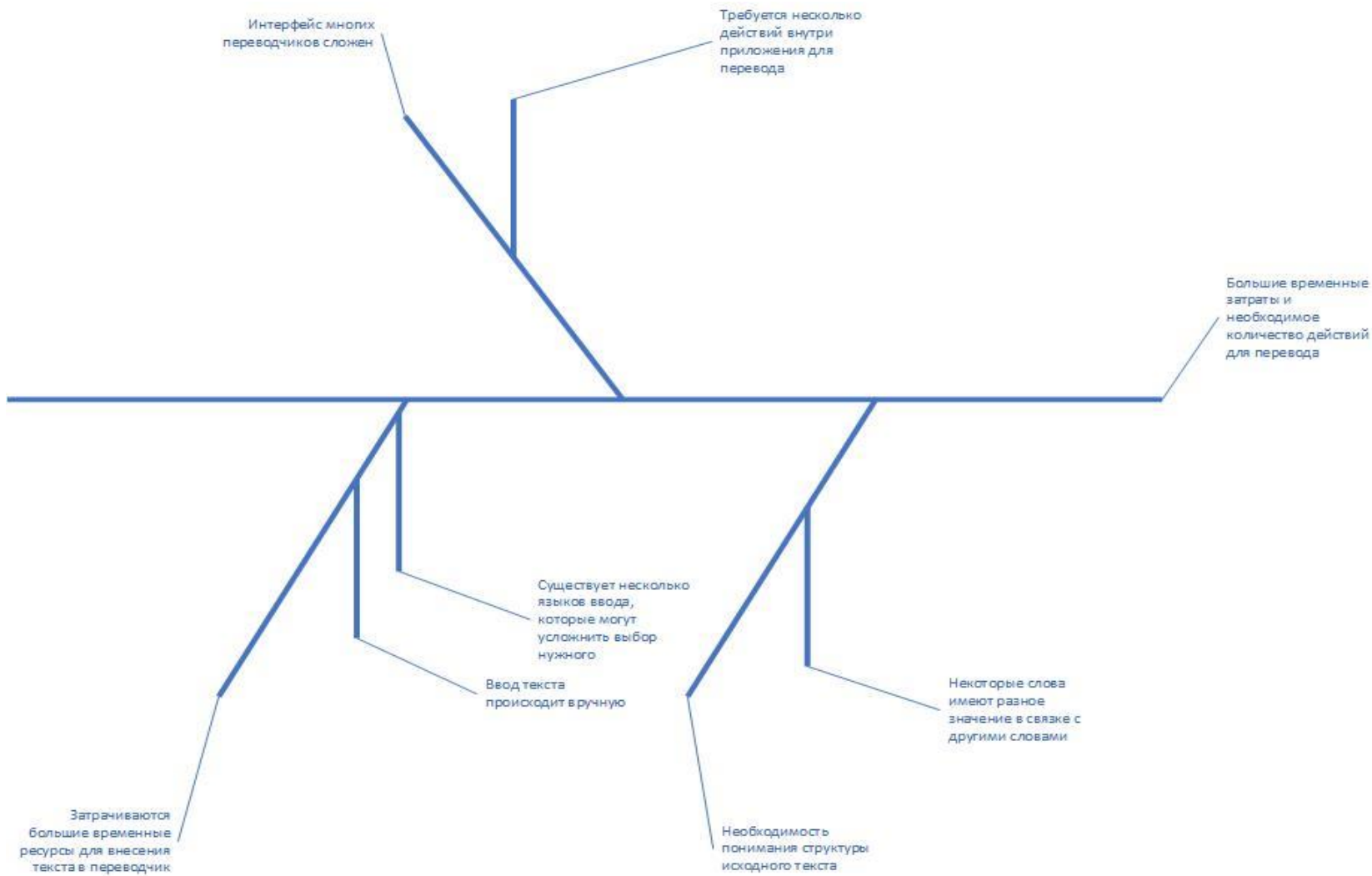


Рисунок 1. Диаграмма FishBone

1.3. Сравнение с аналогами

Так как тема AR не особо развита на рынке переводчиков, то конкурентов в данном сегменте не так много. Но, тем не менее, для наиболее популярных из них следует произвести сравнительный анализ.

Таблица 1

Сравнительный анализ продуктов

Название	Prompt	Google	Multitran
Возможность AR	Нет	Да	Нет
Оценка AR		6/10	
Возможность остановки перевода в режиме реального времени		Да	
Возможность сохранения изображения		Да	
Возможность включения вспышки		Да	
Возможность ZOOM		Отсутствует	
Субъективная оценка	Не является конкурентом	Является серьезным конкурентом	Не является конкурентом

Как видно из приведенной таблицы, на рынке приложений по переводу текста видна явная нехватка AR-перевода в режиме реального времени. Более того, у главного конкурента (GOOGLE) отсутствует функция zoom, которая может стать решающей, при ситуациях, в которых камера не может захватить текст с текущего расстояния.

Проектирование системы подразумевает под собой выявление характеристик различных процессов и создание модели их совместного поведения.

Для планирования потоков процессов часто используют диаграмму EPC. Она позволяет показать структуру потока управления, а именно последовательность решений, функции, события и другие элементы какого-либо бизнес-процесса.

Глава 2. Проектирование информационной системы

Стандартного подхода к процессу перевода в дополненной реальности пока что нет, однако можно выделить основные этапы, каждый из которых можно разбить на составляющие части. Наглядно продемонстрировать процесс перевода в дополненной реальности можно с помощью диаграмм IDEF0 и IDEF3.

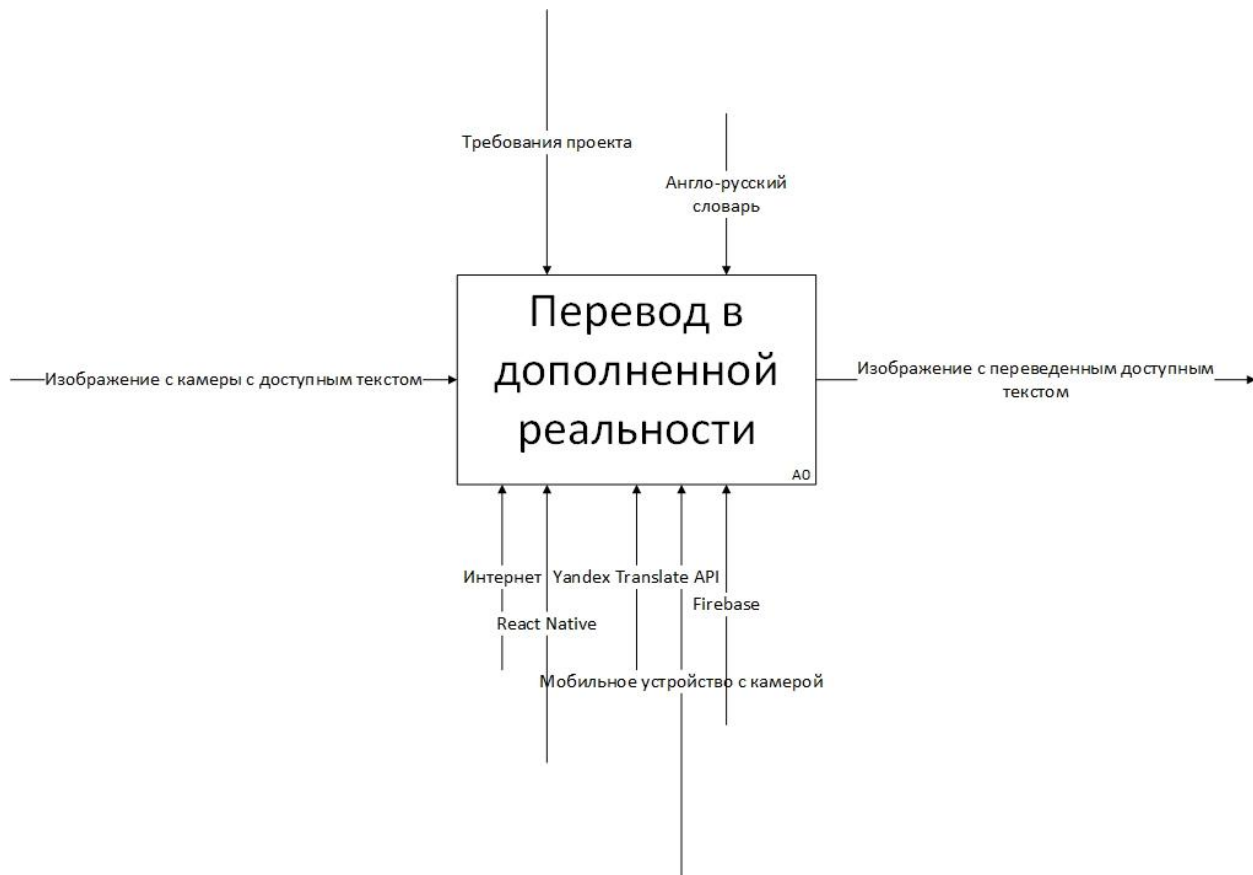


Рисунок 2. Диаграмма IDEF0.

На данной диаграмме представлен процесс перевода в дополненной реальности. Для получения изображения с переведенным текстом нам потребуется мобильное устройство с камерой.

Также для перевода, как такового, нам потребуется Yandex Translate api и интернет для функционирования этого api.

Распознавание текста обеспечивает еще один инструмент Firebase.

В качестве контролирующих документов выступают требования к проекту, в которых прописаны основы взаимодействия и технические характеристики компонентов, а также англо-русский словарь, который позволяет проконтролировать правильность перевода текста.

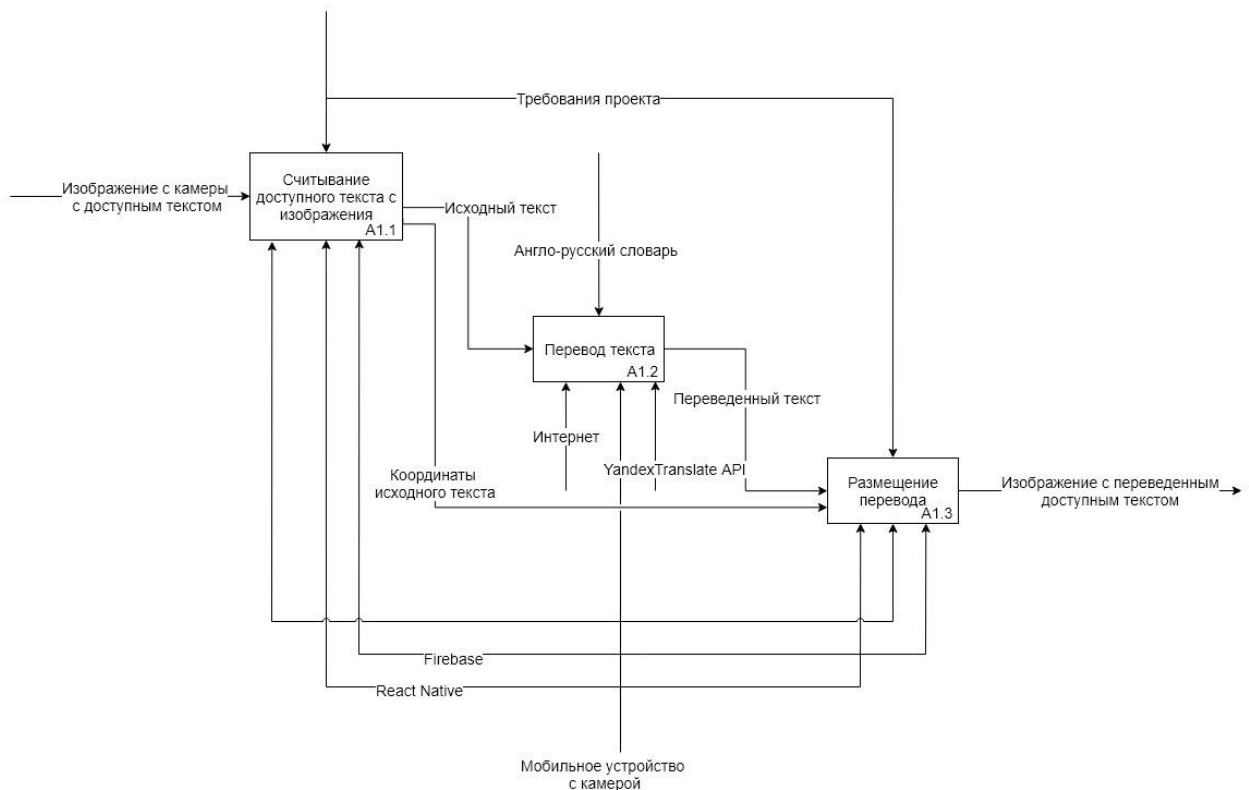


Рисунок 3. Диаграмма IDEF0 (первый уровень декомпозиции).

Процесс перевода текста в дополненной реальности можно разделить на 3 основных процесса: считывание доступного текста с изображения, поступившего с камеры, непосредственно перевод текста, и размещение перевода обратно на плоскости, с выводом изменений на экран устройства.

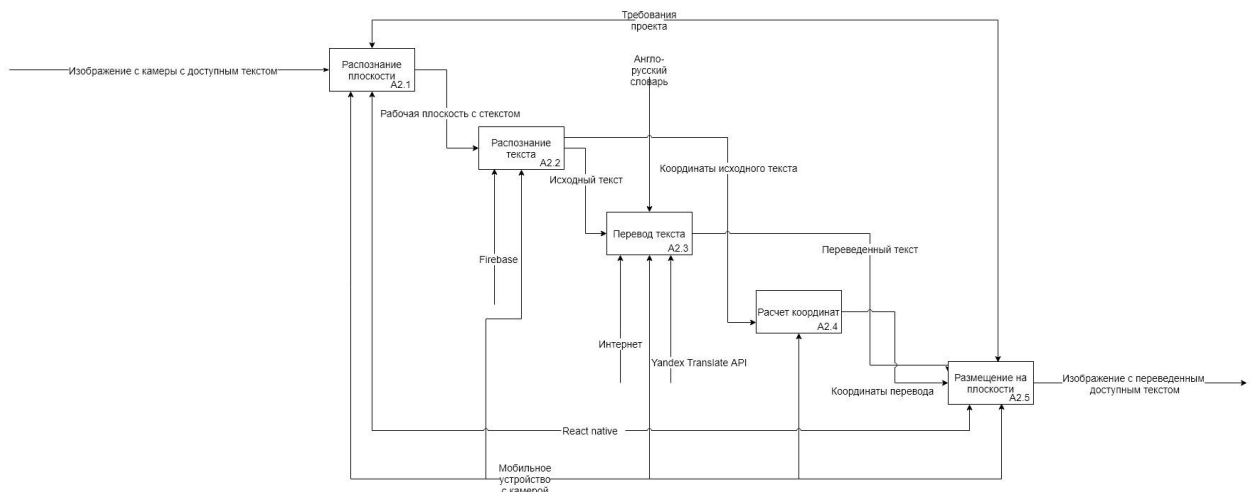


Рисунок 4. Диаграмма IDEF0 (второй уровень декомпозиции).

Процессы считывания доступного текста с изображения и размещения перевода можно разделить на четыре подпроцесса: распознавание плоскости, распознавание текста, перевод текста, расчет координат перевода, размещение перевода на плоскости, с выводом изменений на экран устройства.

распознавание текста на этой плоскости, расчет координат переведенного текста, и, наконец, размещение текста обратно на плоскость.

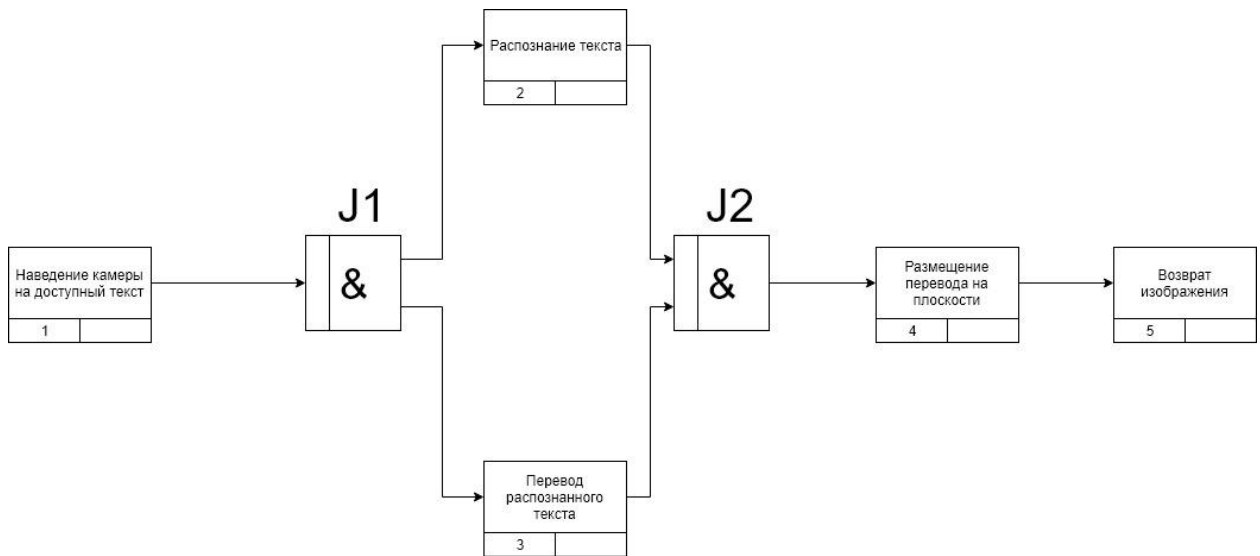


Рисунок 5. Диаграмма IDEF3.

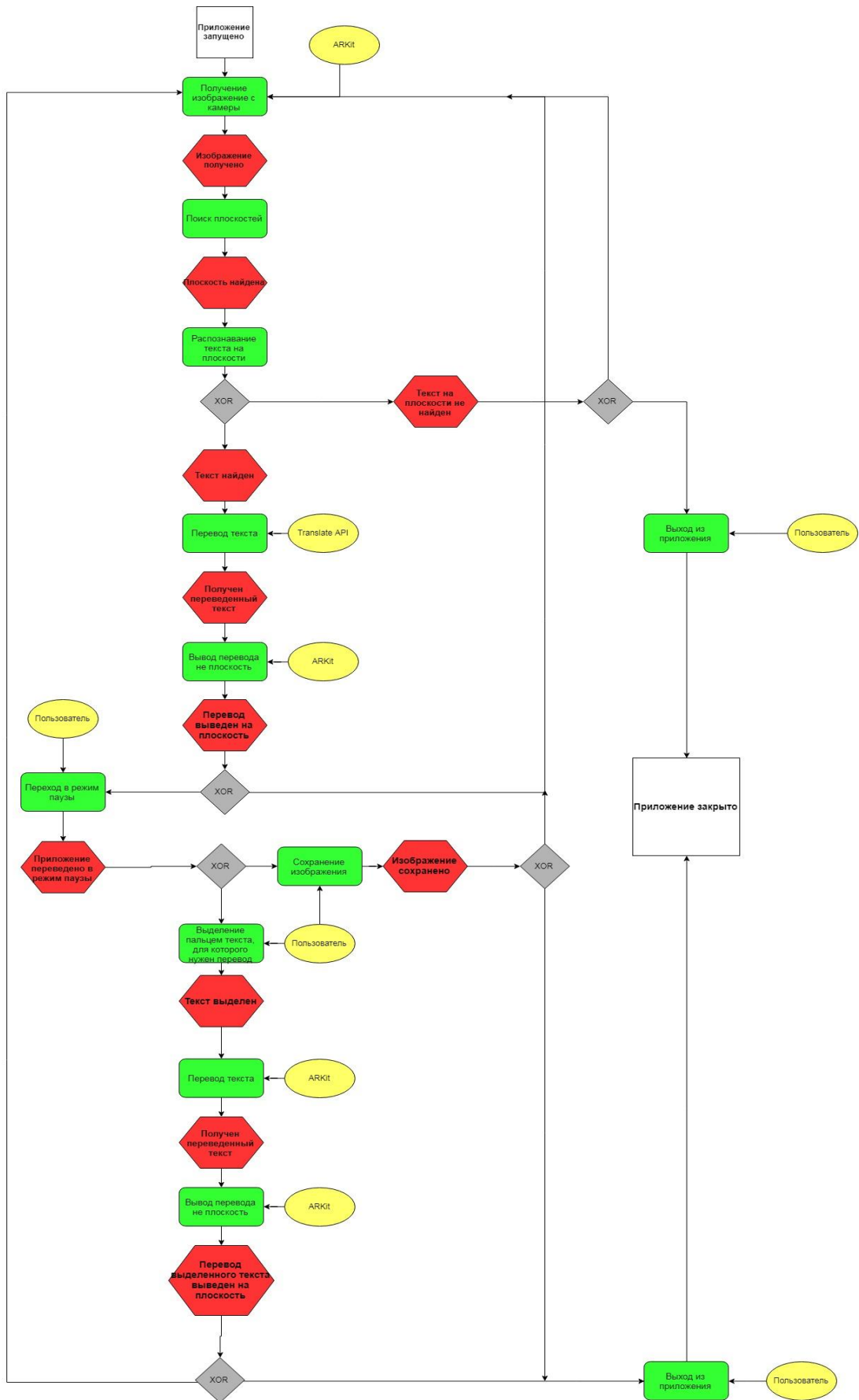


Рисунок 6. Диаграмма EPC

На приведенной диаграмме можно отметить лишь 3 процесса, в которые вовлечен пользователь. Этими процессами являются переход в режим паузы, сохранение изображения, и выделение пальцем текста, для которого необходим перевод.

2.1. Проектирование потоков данных.

Данный процесс также полезно представить с помощью DFD диаграммы, так как разрабатываемая информационная система работает с большим объемом различных данных.



Рисунок 7. Диаграмма DFD

На приведенной диаграмме можно увидеть, что работа с информационной системой заключается в общении пользователя с переводчиком с помощью камеры.

При рассмотрении процесса перевода с помощью информационной системы в дополненной реальности в режиме реального времени можно выделить основные шаги:

- Обработка изображения с помощью приложения;
- Распознавание текста с помощью Firebase;
- Перевод текста с помощью Yandex Translate API;
- Размещение текста с помощью React Native;
- Возврат изображения с переводом пользователю.

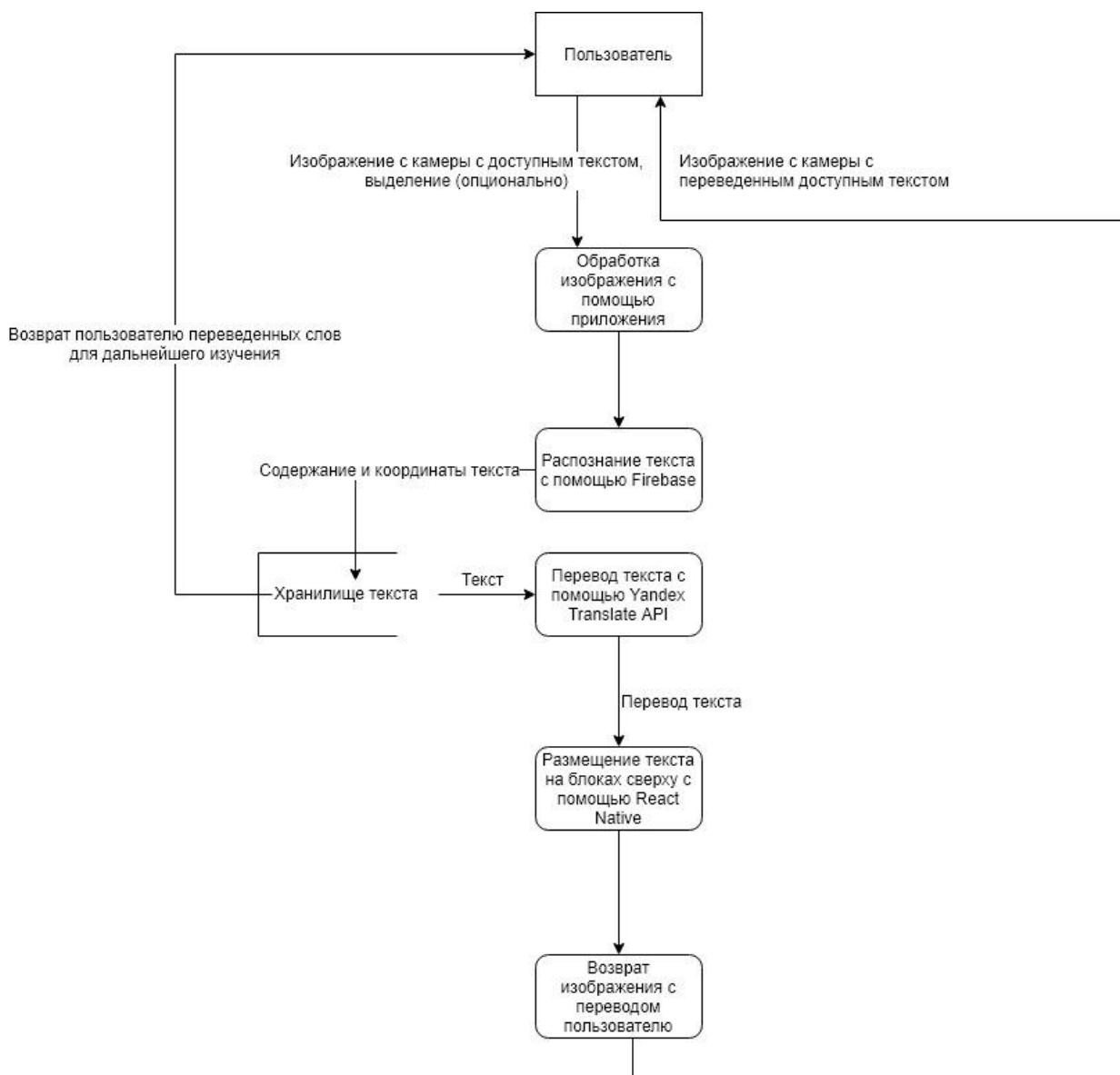


Рисунок 8. Диаграмма DFD (первый уровень декомпозиции)

2.2. Проектирование процессов внутри ИС.

Заключительной частью проектирования процесса решено было сделать BPMN диаграмму. Она ориентирована как на технических специалистов, так и на бизнес-пользователей. Для этого язык использует базовый набор интуитивно понятных элементов, которые позволяют определять сложные семантические конструкции. Кроме того, спецификация BPMN определяет, как диаграммы, описывающие бизнес-процесс, могут быть трансформированы в исполняемые модели.

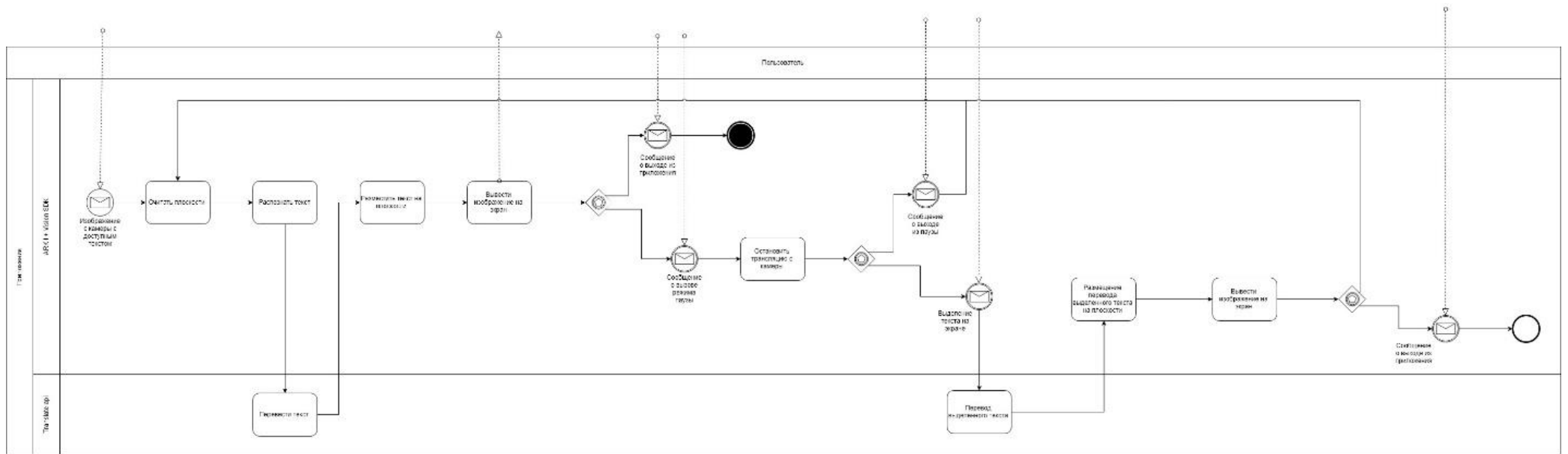


Рисунок 9. Диаграмма BPMN

Эта диаграмма наглядно демонстрирует процесс работы пользователя и разработанного приложения. Большую часть работы приложения выполняет без участия пользователя, только интересуясь о некоторых деталях, касающихся объектов перевода.

Глава 3. Разработка информационной системы перевода в дополненной реальности в режиме реального времени.

3.1. Выявление нефункциональных требований к информационной системе

Для решения проблем, связанных с большими временными затратами, а также большим количеством действий, необходимых для перевода текста, было решено создать собственный инструмент для перевода в дополненной реальности. Для этого необходимо выделить ключевые моменты разрабатываемого ПО:

1. Шрифт и разметка веб-страниц должна адаптивно изменяться в зависимости от используемого устройства. Текст должен быть одинаково хорошо читаемым, изображения – ясно различимыми, но не загромождающими интерактивные элементы или текст;

2. Во время ожидания ответа от сервера клиента необходимо не оповещать о том, что запрашиваемая операция находится в процессе обработки;

3. Максимально допустимое время ожидания ответа от сервера – 5 секунд;

4. Использование не анимированных взаимодействий с интерактивными элементами должно быть минимально возможным;

5. Приложение должно полностью функционировать при использовании следующих операционных систем:

- IOS;
- Android.

3.2. Выявление ролей и вариантов использования для информационной системы.

Проект рассчитан на использование любыми пользователями в быту.

Предложенное ПО имеет несколько различных вариантов использования. Для удобства они представлены на Use-Case диаграмме.

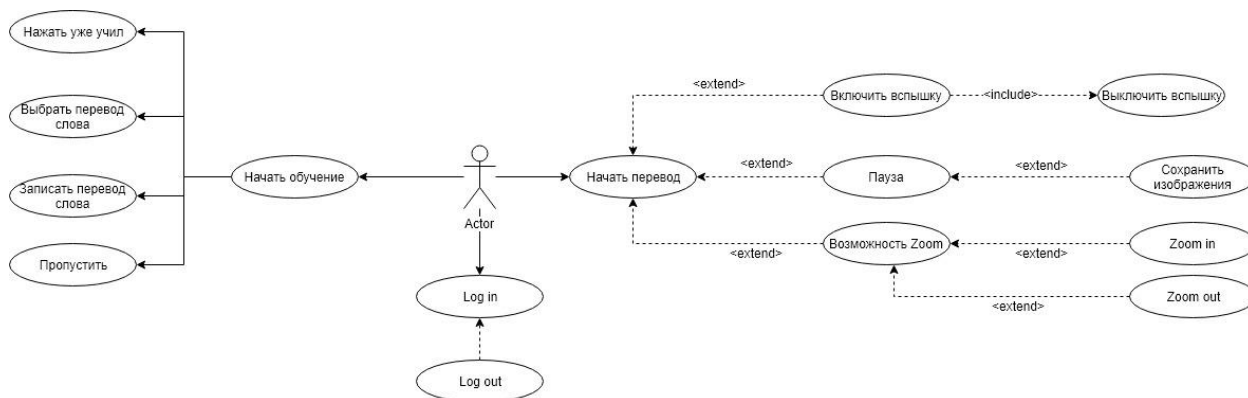


Рисунок 10. Use-Case диаграмма

3.3. Выбор средств разработки.

Выбор средств разработки был в большей степени обусловлен требованиями кроссплатформенности, поэтому языком написания основной части стал JavaScript. Также для реализации всех запланированных возможностей использования были задействованы множество расширений, Фреймворков, сборщиков проектов и API.

В качестве среды разработки был выбран продукт студии JetBrains – WebStorm на основе IntelliJ Idea. JetBrains WebStorm – интегрированная среда разработки на JavaScript, CSS & HTML от компании JetBrains, разработанная на основе платформы IntelliJ IDEA.

WebStorm обеспечивает автодополнение, анализ кода на лету, навигацию по коду, рефакторинг, отладку, и интеграцию с системами управления версиями. Важным преимуществом интегрированной среды разработки WebStorm является работа с проектами (в том числе, рефакторинг кода JavaScript, находящегося в разных файлах и папках проекта, а также вложенного в HTML). Поддерживается множественная вложенность (когда в документ на HTML вложен скрипт на Javascript, в который вложен другой код HTML, внутри которого вложен Javascript) – то есть в таких конструкциях поддерживается корректный рефакторинг.[2]

Немаловажной причиной выбора именно этой среды разработки стало поддержка WebStorm Node.js. WebStorm поддерживает отладку приложений в node.js. Также поддерживается полный набор функций редактирования приложений на javascript – как для исполнения на сервере, так и в браузере: автодополнение,

навигация по коду, рефакторинг и проверка на ошибки. Для node.js поддерживается также вывод сообщений node.js на отдельную вкладку в IDE.

Список используемых инструментов с кратким их описанием:

- Node или Node.js – программная платформа, основанная на движке V8 (транслирующем JavaScript в машинный код), превращающая JavaScript из узкоспециализированного языка в язык общего назначения. Node.js добавляет возможность JavaScript взаимодействовать с устройствами ввода-вывода через свой API (написанный на C++), подключать другие внешние библиотеки, написанные на разных языках, обеспечивая вызовы к ним из JavaScript-кода. Node.js применяется преимущественно на сервере, выполняя роль веб-сервера, но есть возможность разрабатывать на Node.js и десктопные оконные приложения (при помощи NW.js, AppJS или Electron для Linux, Windows и macOS) и даже программировать микроконтроллеры (например, tessel и espruino).[5]

- Для непосредственно перевода текста в приложении было выбрано Translate API от Yandex, которое распространяется на свободной основе.

- Как веб-сервер было принято решение использовать набор технологий, представленный web-фреймворком Express.

- Для тестирования работы с API был выбран сервис Postman.

- Библиотека Mongoose необходима в данном проекте для доступа и работы с не реляционной базой данных Mongo DB, а Mongo DB Atlas предоставляет облачное хранилище для базы данных.

- Технология распознавания объектов в дополненной реальности обеспечивается библиотекой Firebase.

- Подключаемая библиотека MobX обеспечивает контроль состояния приложения и делает невозможность его инконсистентности.

- В качестве системы контроля версий был выбран продукт Git, отвечающий главному требованию – безвозмездности использования.

- Babel JS – используемый в данном приложении компилятор языка JavaScript значительно упрощающий работу с функциями, особенно стрелочными.

- Eslint также является одним из инструментов для работы с кодом, который позволяет выявить распространенные ошибки до компиляции, а также привести уже написанный код к единому стилю.

- В качестве статического анализатора типов в языке Java Script было решено использовать анализатор FlowJS. К сожалению, написание нативных, кроссплатформенных приложений на языке Java Script без анализатора типов практически не представляется возможным.

- Для контроля максимальной длины строки, которая различается при написании приложения под разные платформы использовался инструмент Prettier.

- Для автоматического внесения изменений при написании кода в мобильное устройство использовалась подключаемая библиотека Watchman.

- VscryptJS – подключаемая библиотека для генерации хэша. При разработке приложения безопасность данных пользователей становится во главе, поэтому все данные на сервер передаются в зашифрованном виде. А генерация хэш-функций ложится на данную библиотеку.

- Body Parser – инструмент для взаимодействия вышеупомянутого фреймворка Express с формами в кодировке URL.

- Jsonwebtoken – является стандартом передачи данных между клиентом и сервером, а также инструментом для генерации пакетов этих данных.

- Одним из главных компонентов является React Native. React Native - это платформа мобильных приложений с открытым исходным кодом, созданная Facebook. Он используется для разработки приложений для Android, iOS и UWP, позволяя разработчикам использовать React вместе с собственными возможностями платформы.

3.4. Системные требования к информационной системе.

1. Устройство с операционной системой iOS11 и выше или Android 6.0 и выше;
2. Наличие подключения к интернету;

3. Размер свободной памяти на устройстве более 500 мб;
4. Наличие камеры.

3.5. Распределение ответственности

Таблица 2

Распределение ответственности

Исполнитель	Задание	Описание
Сухарев Семен Игоревич	Проектирование и разработка серверной части приложения.	Выбор стека используемых для разработки серверной части приложения технологий. Выбор базы данных. Проектирование базы данных. Проектирование архитектуры серверной части приложения и ее дальнейшая реализация. Реализация регистрации, авторизации и идентификации клиента. Проектирование и разработка API с учетом постулатов архитектурного стиля REST. Выбор способов размещения и само размещение серверной части приложения. Тестирование серверной части приложения.
Тогызбаев Тимур Нургалиевич	Разработка алгоритма распознавания слов и добавления их в обучающую выборку. Проектирование всего приложения.	Проектирование приложения в целом. Написание базового класса модели. Описание алгоритма распознавания текста и перевода. Описание алгоритма расчета размера шрифта для выводимого перевода. Описание структуры проекта и взаимодействия между элементами. Внедрение инструментов проверки качества кода. Внедрение типизации. Настройка офлайн работы и онлайн синхронизации. Реализация вспомогательных функций

		для «Dependency Injection» реализация «Inversion of control».
Каракулов Александр Андреевич	Разработка интерфейса и интеграция с сервисами	Разработка интерфейса приложения, доступного и на «IOS» и на «ANDROID». Разработка сервиса генерации упражнений, разработка сервиса проверки перевода, разработка сервиса авторизации и регистрации. Разработка масштабируемого интерфейса. Написание тестов для сервисов. Написание типов для кодовой базы. Создание дизайн системы. Описание классов Типографики, Отступ, Цвет и т.д., Реализация маршрутизации приложения.

3.6. Разработка приложения

3.6.1. Алгоритмические решения

При разработке приложения были использованы несколько алгоритмов для работы с массивами, одним из которых стал алгоритм Фишера-Ейтса.

Данный алгоритм смешивает элементы массива. В приложении алгоритм использовался для упражнения, где имеется 4 варианта ответа, в том числе один правильный. Чтобы правильный вариант не находился всегда на одной и той же позиции, мы используем алгоритм смешивания Фишера-Ейтса.

```
function shuffleArray(array) {
  let counter = array.length;
  const newArr = [...array];
  // While there are elements in the array
  while (counter > 0) {
    // Pick a random index
    const index = Math.floor(Math.random() * counter);
    // Decrease counter by 1
    counter -= 1;
    // And swap the last element with it
    const temp = newArr[counter];
```



```

        newArr[counter] = newArr[index];
        newArr[index] = temp;
    }
    return newArr;
}

```

При разработке системы считывания в дополненной реальности использовался алгоритм первичной обработки входящих пар слов. Данный алгоритм использовался чтобы отфильтровать входящий массив. Для этого все слова приводятся к единому регистру, затем исключаются те пары слов, которые на исходном и конечном языке выглядят одинаково, например, «dddfff». Далее необходимо было убрать те пары слов, которые уже встречались. На этом первичная обработка элементов заканчивается после чего начинается серверная обработка.

```

class TextFilterService {
    filterTranslationList(translationList: ?(TWordPair[])) {
        if (!translationList) {
            return [];
        }
        return translationList
            .map(this.valueToLower)
            .filter(this.checkNotEqual)
            .reduce(this.removeDuplicates, []);
    }

    checkNotEqual([word1, word2]: TWordPair) {
        return word1.value !== word2.value;
    }

    valueToLower([word1, word2]: TWordPair) {
        return [{ ...word1, value: word1.value.toLowerCase() }, { ...word2,
value: word2.value.toLowerCase() }];
    }

    existIn(wordList: TWordPair[], [word1ToCheck, word2ToCheck]: TWordPair) {
        return wordList.some(
            ([word1, word2]) => word1ToCheck.value === word1.value &&
word2ToCheck.value === word2.value
        );
    }

    removeDuplicates = (result: TWordPair[], wordPair: TWordPair) => {
        const alreadyExist = this.existIn(result, wordPair);
        if (alreadyExist) {

```

```
        return result;
    }
    return [...result, wordPair];
};
}
```

Также использовался алгоритм, который решал проблему подбора заданий для упражнения, так как есть ряд критериев и особенностей для их создания [Приложение А].

Например, в рамках одного упражнения не должно быть повторяющихся слов, то есть заданий. Также в обучающую выборку попадают только те слов рейтинг которых является меньше максимума (в нашем случае 100 у.е.). В рамках алгоритма использовался генератор (итератор, реализованный в виде сопрограммы) для того чтобы не заводить отдельных счетчиков и массивов.

При разработке приложения был использован принцип инверсии зависимостей. Если в стандартном подходе сервисы зависят от моделей, то, когда возникнет необходимость переиспользовать сервисы или что-то изменить, возникнет проблема прямой зависимости классов от модели. Также не будет возможности соответствовать микросервисной архитектуре и модульному подходу, так как все переплетено. При использовании принципа инверсии контроля, вместо импорта классов, эти классы внедряются в целевой класс с помощью инъекций зависимостей и в итоге получается, что класс не напрямую зависит от какого-то другого класса, а зависит от интерфейса класса, что делает систему масштабируемой.

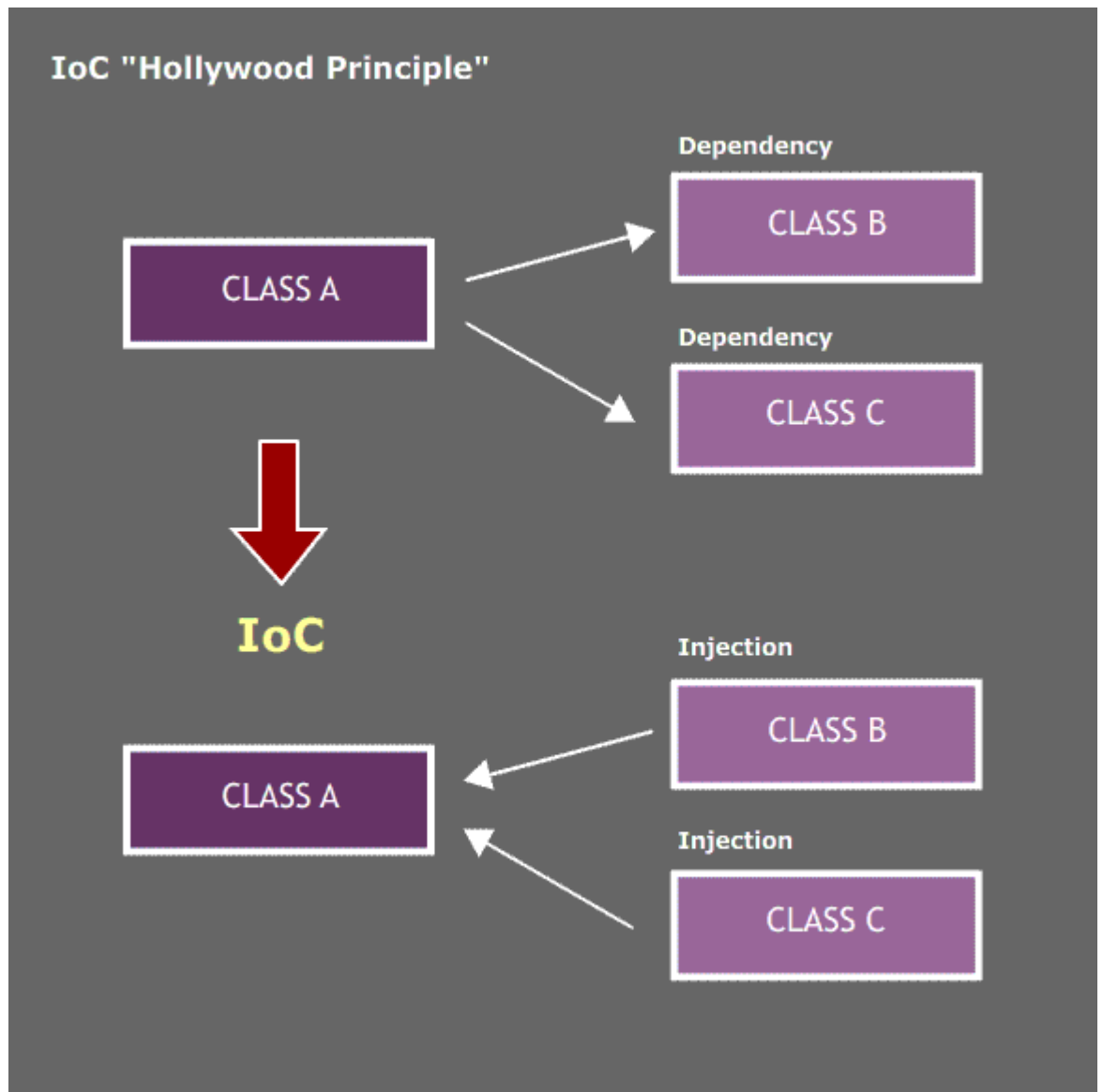


Рисунок 11 – Принцип инверсии зависимостей

Так как JavaScript является нетипизированным языком, для достижения большей безопасности существует ряд инструментов. Самые популярные из них – это FlowJs и TypeScript. Нами был выбран Flowjs из-за более легкой интеграции в готовую кодовую базу. Благодаря системе типов существует возможность избежать ошибок приведении типов или обращения к несуществующим методом. Во время разработки нам удалось покрыть большую часть приложения системой типов и описать какие методы, классы и интерфейсы используются.

Появилась необходимость в инструменте контроля, который показывал состояние системы типов на проекте. Консольная утилита Flow-Coverage-Report собирает данные со всех файлов, анализирует эти данные и формирует отчет в виде

html-страницы [Приложение Б], где можно отслеживать, в каком состоянии сейчас находится проект. В нашем случае покрытие проекта составило 75 процентов, что связано с изучением системы типов и некоторыми ошибками которые исправить не получилось, также некоторые ошибки связаны с ошибками внутри системы Flowjs.

3.6.2. Решения при работе с данными

Серверная часть приложения была написана на Node JS с использованием Express. Express - это гибкий и минималистичный веб-фреймворк для приложений, написанных на Node.js. Фреймворк предоставляет обширный набор функций для мобильных и веб-приложений. Express предоставляет множество служебных HTTP методов и промежуточных обработчиков, с помощью которых, создать надежный API можно быстро и легко. В качестве базы данных выбор пал на Mongo DB. Для работы с базой из приложения используется библиотека Mongoose. Mongoose - объектно-документный отображитель, это означает, что Mongoose позволяет решить проблемы с валидацией, определяя объекты со строго-типизированной схемой, которые соответствуют документу MongoDB.

Для описания API использовался Swagger Editor. Swagger Editor - это редактор с открытым исходным кодом для разработки, определения и документирования RESTful API. Исходный код редактора Swagger можно найти на сайте GitHub.com. Данный редактор имеет очень понятный интерфейс, позволяющий не только задокументировать формат API, но и наглядно показать разработчику, который собирается им пользоваться, как нужно работать с API. Рассмотрим описание API создания переводов.

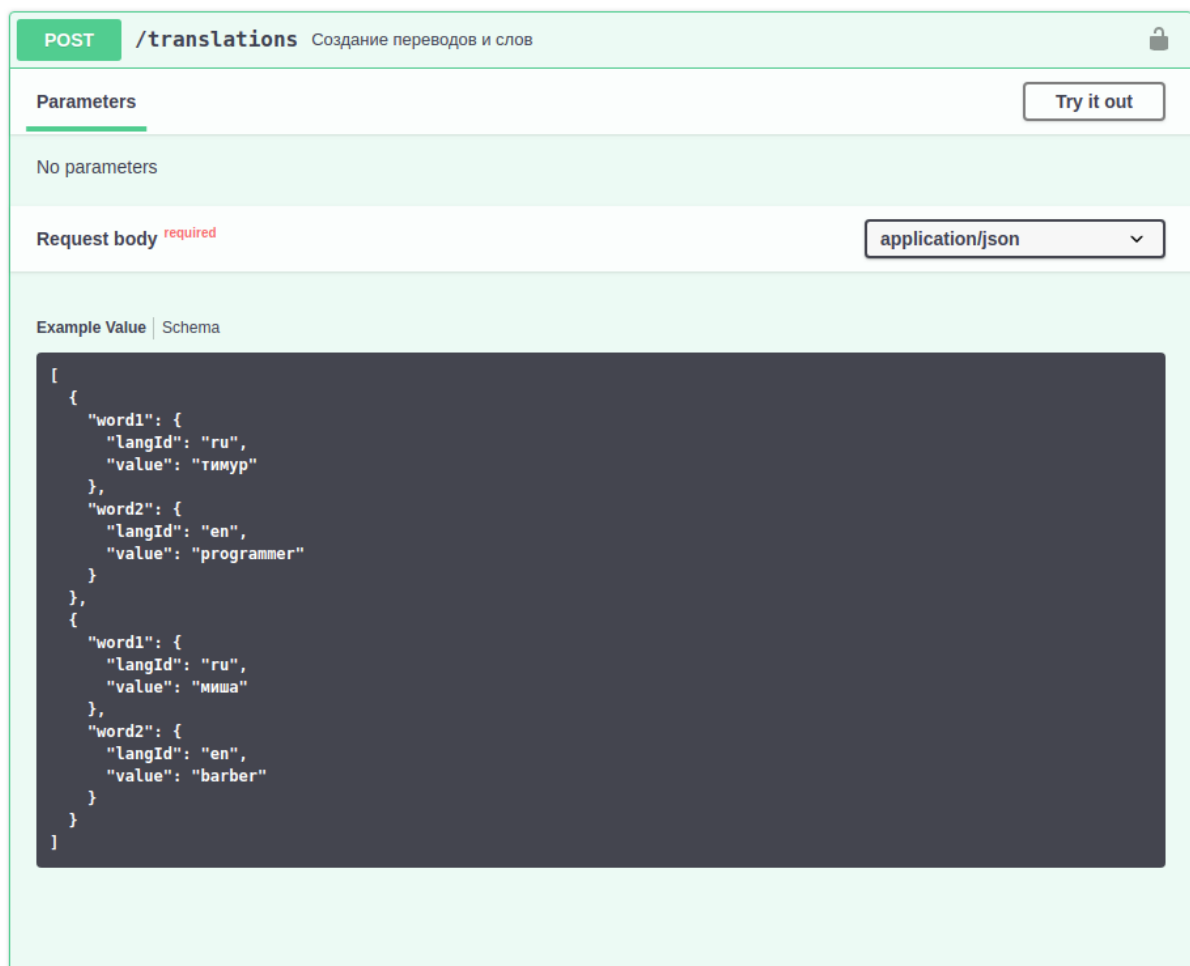


Рисунок 12 – Описание маршрутизации API и тела запроса с помощью Swagger Editor

Swagger Editor предоставляет замечательный инструмент для описания API. Данный скриншот показывает пример тела запроса, из чего мы можем понять формат входных данных.

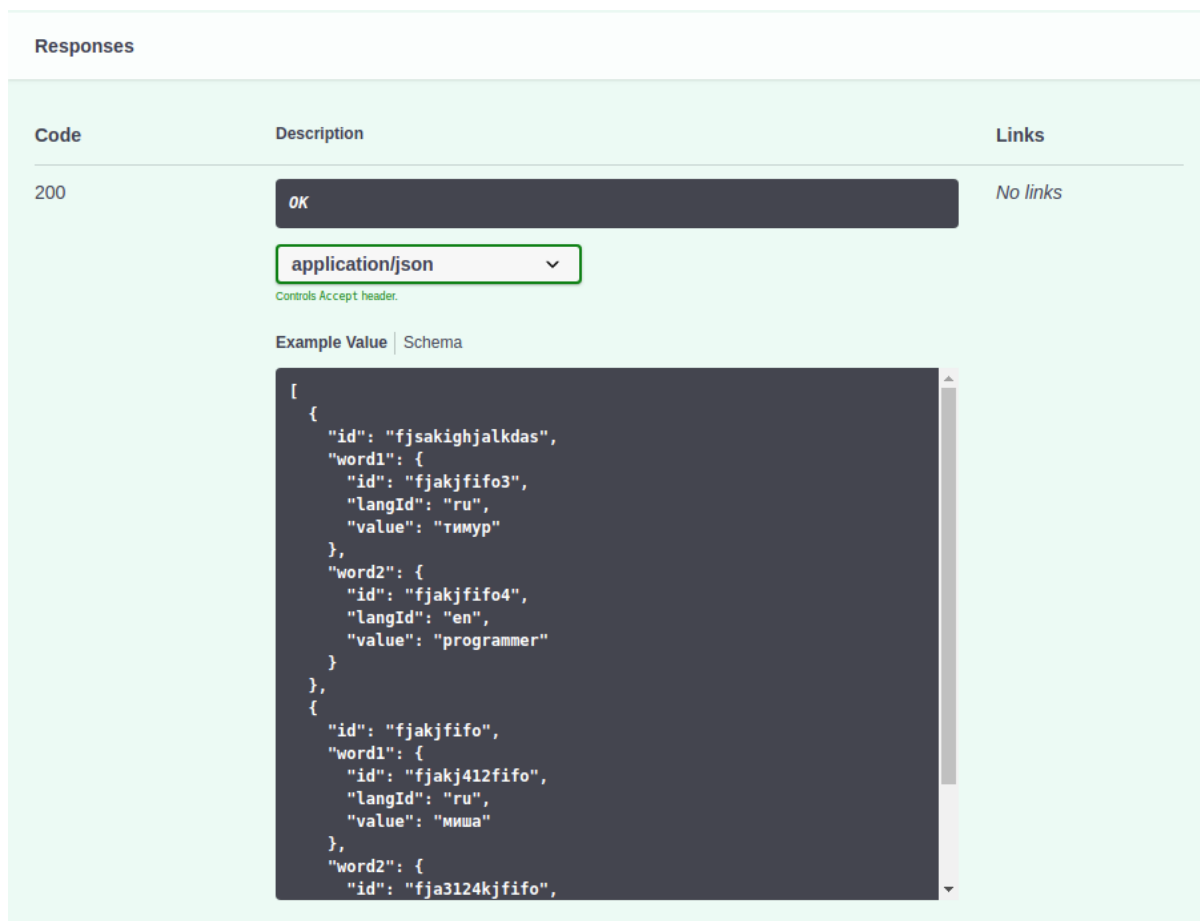


Рисунок 13 – Описание ответа API с помощью Swagger Editor

Сразу после примера тела запроса, описывается пример ответа, чтобы пользователь API мог корректно описать МОСК и обработчик ответа, если, например, сама API еще не написана. Как видно из ответа, API сразу создает записи соответствующих сущностей в БД и возвращает ID каждого.

На все запросы по маршруту translations реагирует контроллер, написанный и подключенный самостоятельно.

```
app.use('/translations', require('./translations/translation.controller'))
```

На POST запрос по маршруту translations срабатывает функция-контроллер «create», которая обрабатывает и передает тело запроса в функцию-обработчик «create», передает ответ функции-обработчика на клиента производит выборку ошибок, передавая их дальше по потоку.

```
router.post('/', create);
```

```
function create(req, res, next) {
  translationService.create(req)
    .then((response) => res.send(response))
}
```

```

        .catch(err => next(err));
    }

```

Функция - обработчик «create» обрабатывает полученный массив, формат которого был описан в Swagger, и создает сразу несколько записей в БД, а именно: «Word» (Слово), «Translation» (перевод), «userTranslationLink» (связь переводов с пользователем), «userWordLink» (связь слов с пользователем). Переводы проходят предварительную фильтрацию на предмет уже существования таких связей слов в БД. При добавлении записи «Word» в БД идет предварительная проверка на предмет существования такой записи. Если она есть, то возвращается «Id» существующей записи и не создается связь-запись «UserWordLink».

```

async function create({body: translationParamList, user: {sub: currentUserId}}) {
    let result = [];

    for(let i=0; i < translationParamList.length; i++) {
        const {word1: wordParam1, word2: wordParam2} = translationParamList[i];
        const wordRecord1 = await createWordRecord(wordParam1, currentUserId);
        const wordRecord2 = await createWordRecord(wordParam2, currentUserId);
        const translationParam = {
            wordId1: wordRecord1.id,
            wordId2: wordRecord2.id,
        };
        const translationEntity = new Translation(translationParam);
        const translationRecord = await translationEntity.save();

        await userTranslationLinkService.create({
            userId: currentUserId,
            translationId: translationRecord.id,
        });
        const translation = {
            id: translationRecord.id,
            word1: wordRecord1,
            word2: wordRecord2,
        };

        result[i] = translation;
    }

    return result;
}

async function createWordRecord(wordParam, userId) {
    const [wordRecord] = await Word.find(wordParam);

    if (!wordRecord) {
        const newWordRecord = await wordService.create(wordParam);
        await userWordLinkService.create({
            userId: userId,
            wordId: newWordRecord.id,
        });
        return newWordRecord;
    }

    return wordRecord;
}

```

}

Связки «userWordLink» и «userTranslationLink» нужны для того, чтобы отдавать пользователю только его слова и переводы. Идентификация пользователя происходит с помощью JWT токенов, которые создаются при авторизации и отправляются клиенту.

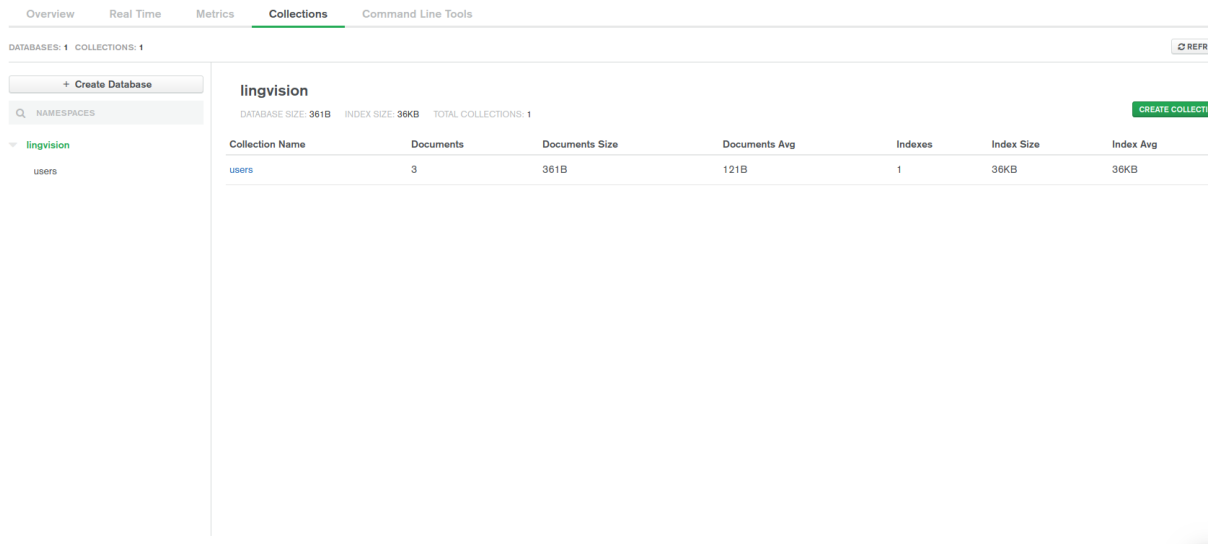


Рисунок 14 – Просмотр текущих сущностей БД с помощью оболочки MongoDB – Atlas

Согласно рисунку, на данный момент в базе находятся записи о трех пользователях. Информация о них не может быть продемонстрирована, потому что она конфиденциальна.

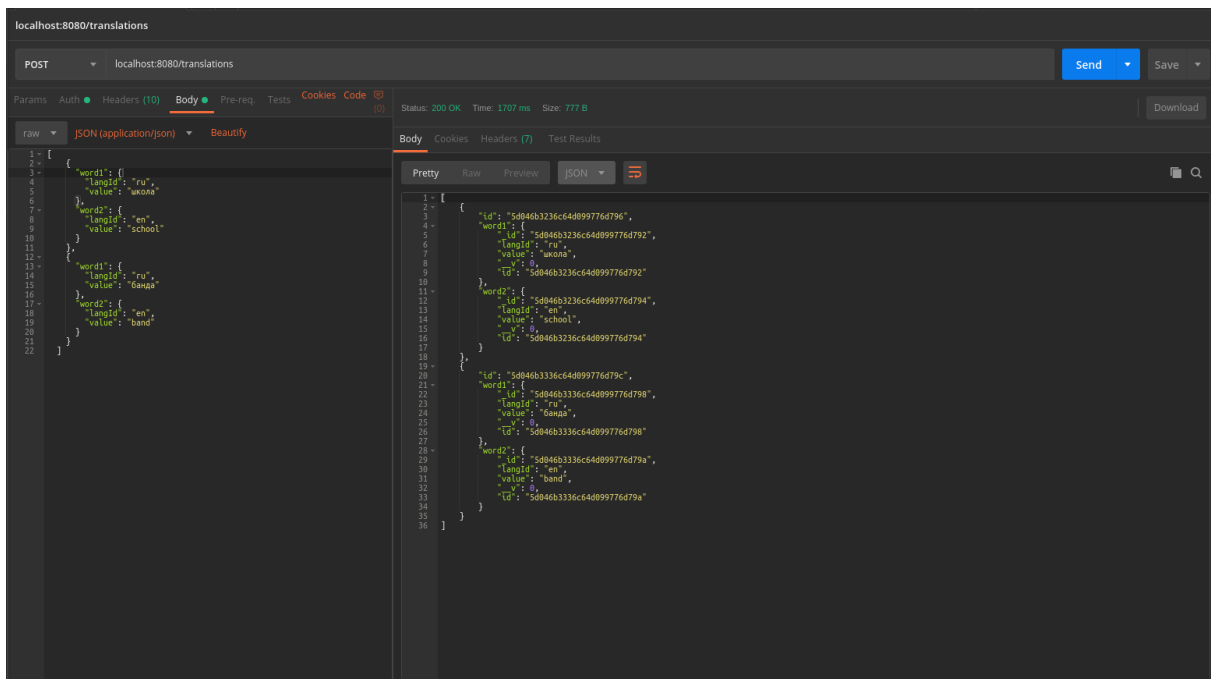
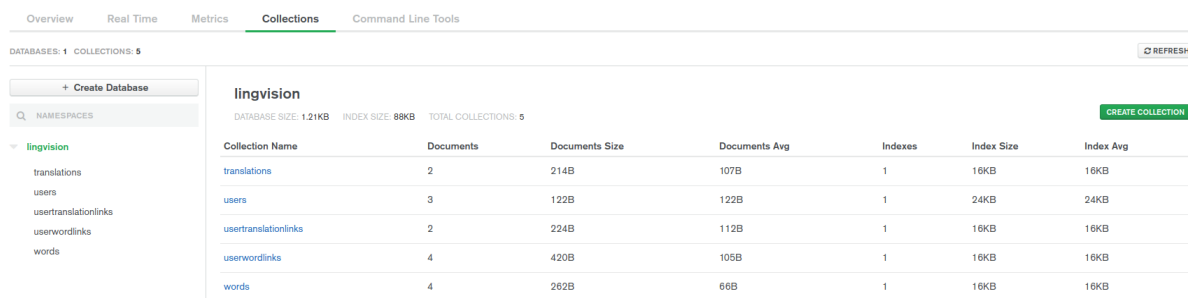


Рисунок 15 – Ответ сервера на отправленный запрос в программе Postman

После отправки запроса сервер отвечает нам в формате, описанном в Swagger.



The screenshot shows the MongoDB Atlas 'Collections' page for the 'lingvision' database. The database size is 1.21KB and the index size is 88KB. There are 5 collections in total. A table lists the collections with their respective document counts, sizes, and index information.

Collection Name	Documents	Documents Size	Documents Avg	Indexes	Index Size	Index Avg
translations	2	214B	107B	1	16KB	16KB
users	3	122B	122B	1	24KB	24KB
usertranslationlinks	2	224B	112B	1	16KB	16KB
userwordlinks	4	420B	105B	1	16KB	16KB
words	4	262B	66B	1	16KB	16KB

Рисунок 16 – Просмотр текущих сущностей БД с помощью оболочки MongoDB – Atlas

Как можем заметить на рисунке 15, в БД появились новые записи, а именно: Два перевода, две связи перевода с пользователем, 4 слова и 4 связи слова с пользователем. Если мы отправим запрос еще раз, то новые записи сделаны не будут.

Валидация реализована инструментами Mongoose. Наиболее нагляден пример модели пользователя. Описывая эту модель, указывается, что у пользователя имеются поля «email» и «hash», также указывается, что эти поля обязательны. При отправке запроса без указания поля «email», функция-обработчик выкинет ошибку, которую обработает функция-контроллер и отправит дальше по потоку. Для обработки ошибок, возникающих в результате запроса клиента, был написан обработчик ошибок. Он подключается после подключения контроллера всех маршрутов и обрабатывает все ошибки.

```
app.use(errorHandler);

module.exports = errorHandler;

function errorHandler(err, req, res, next) {
  if (typeof (err) === 'string') {
    return res.status(400).json({ message: err });
  }

  if (err.name === 'ValidationError') { // Обработка ошибок валидации
    return res.status(400).json({ message: err.message });
  }

  if (err.name === 'UnauthorizedError') {
    return res.status(401).json({ message: 'Invalid Token' });
  }

  return res.status(500).json({ message: err.message });
}
```

Таким образом, при отправке запроса без поля «email» клиент получает 400ю ошибку с сообщением об отсутствие какого-либо поля.

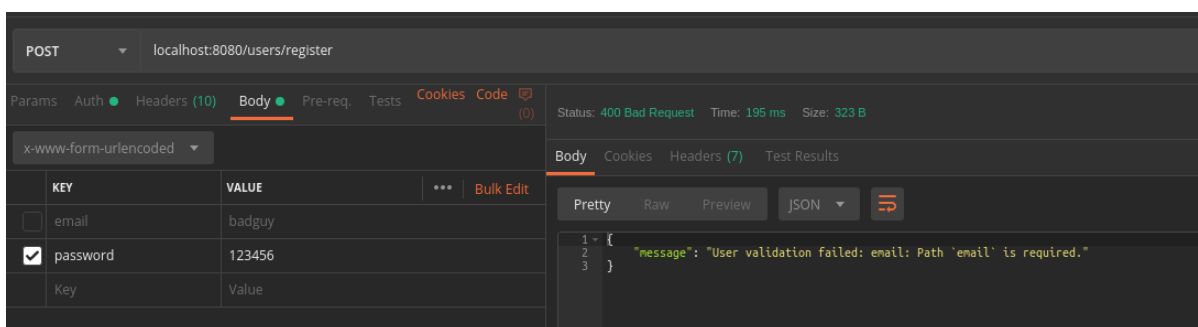


Рисунок 17 – Тело запроса и ответ сервера на отправленный запрос в программе Postman

3.6.3. Визуальная часть, интерфейс

Кроссплатформенность подразумевает запуск и выполнение приложения в разных операционных системах. В нашем случае такими системами стали IOS и Android. Для достижения задач кроссплатформенности был использован фреймворк React Native для работы с языком JavaScript. Данный инструмент позволяет унифицировать код визуальной части. Таким образом единожды написав код мы автоматически получаем два готовых приложения для разных операционных систем, учитывая особенности дизайна для каждой. Это достигается за счет “моста” между нативным кодом и JavaScript.

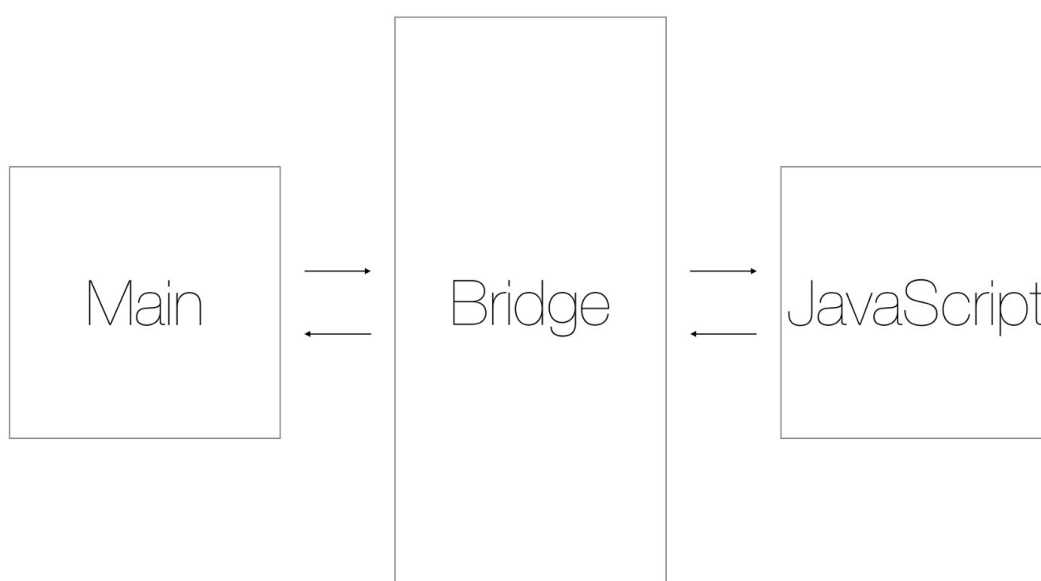


Рисунок 18 – «Мост» между нативным кодом и JavaScript

Между двумя потоками существует «мост», который является основой React Native. Данный мост имеет три важные характеристики:

- Асинхронность – «мост» включает асинхронное общение между потоками, данным методом достигается уверенность в том, что потоки не будут блокировать друг друга.
- Пакетированность – «мост» передает сообщения между потоками оптимизируя их.
- Сериализуемость – два потока никогда не общаются между собой одинаковыми данными, вместо этого данные сериализуются с помощью «моста», благодаря чему достигается скорость работы в каждом из потоков.

В данном примере связывается отображение с сервисом, что позволяет перерисовать отображение, как только что-то изменится в сервисе.

Функция «useService» используется, чтобы достать сервис из контейнера сервисов. После этого в переменную записывается последовательность заданий для упражнений. Данная последовательность является мемоизированной, то есть не изменяемый в рамках одного компонента.

После этого используется функция «useState» для сохранения состояния компонента. Инициализируется состояние с первым заданием упражнения в начале. После этого функция возвращает два параметра. Первый параметр – это текущее задание, второй параметр – это функция смены задания.

```
function useTaskSet(params: TGeneratorProps) {
  const taskGenerator = useService('taskGeneratorService');
  const taskSet = useMemo(() => taskGenerator.generateTaskSet(params), []);

  const [currentTask, setCurrentTask] = useState(() => taskSet.next());
  function nextTask() {
    setCurrentTask(taskSet.next());
  }
  return [currentTask, nextTask];
}
```

Таким образом данная функция используется снаружи:

```
const [task, nextTask] = useTaskSet({
  exerciseType,
  langFrom,
  langTo
});
```

Для корректной работы в качестве аргументов функции передается три параметра: тип упражнения, язык с которого переводим и язык, на который мы переводим. И следующим образом используется для отображения карточки слова:

```
<WordCard
  translationAnswer={task.answer.value}
  exerciseState={exerciseState}
  rating={task && task.ratingEntity ? task.ratingEntity.rating : 0}
  string={task ? task.translation.value : ''}
/>
```

В каждом типе упражнения есть действие, по которому запускается проверка упражнения. В упражнении типа текстового ввода есть кнопка «submit», по которой запускается проверка того, что ввел пользователь. В упражнении, где 4 слова на выбор, проверка осуществляется, когда пользователь непосредственно нажал на слово.

Так как мы собираемся поддерживать большое количество типов упражнений, необходимо было решить проблемы с дублированием кода. Данную проблема решена следующим способом: при вызове компонента упражнения внутрь передается класс, который отвечает за игровую механику и отображение.

```
<Exercise ActionComponent={TextExercise} />
```

И следующие параметры будут переданы в класс механики упражнения. Состояния упражнения: в процессе, ошибка, успешно; «task» – слово для перевода, «answer» – ответ на данной слово, «lang» – язык на который переводится и «check» – функция проверки упражнения.

```
const actionComponentProps = {
  exerciseState,
  task: task.translation,
  answer: task.answer,
  lang: langTo,
  check: onSubmit
};
```

Со стороны дизайна было принято решение не каждый раз писать строку цвета, отступа или шрифта, а создать класс, который бы хранил в себе данные стили. Таким образом, решается сразу несколько проблем. Первая – это уменьшение человеческого фактора, легко ошибиться в написании HEX, а при использовании переменных, ошибиться в названии уже тяжелее.

Вторая проблема – это изменения, как только принимается решение поменять оттенок цвета или шрифт во всем приложении, больше нет необходимости искать во всем приложении места использования, достаточно будет в классе поменять цвет шрифт и другое.

Пример класса отступов:

```
class Spacing {
    static s1: number = PlatformHelpers.isIOS ? 3 : 4;
    static s2: number = PlatformHelpers.isIOS ? 6 : 8;
    static s3: number = PlatformHelpers.isIOS ? 9 : 12;
    static s4: number = PlatformHelpers.isIOS ? 12 : 16;
    static s5: number = PlatformHelpers.isIOS ? 15 : 20;
    static s6: number = PlatformHelpers.isIOS ? 18 : 24;
    static s7: number = PlatformHelpers.isIOS ? 21 : 28;
    static s8: number = PlatformHelpers.isIOS ? 24 : 32;
    static s9: number = PlatformHelpers.isIOS ? 27 : 36;
    static s10: number = PlatformHelpers.isIOS ? 30 : 40;
}
```

В данном классе было принято решение использовать 10 вариаций отступов, так как этого достаточно для потребностей дизайна. Также заранее было уменьшены отступы для IOS устройств.

Пример константы цветов

```
const Color = {
    WHITE: '#ffffff',
    BLACK: '#333333',
    BLUE_10: '#1852E7',
    BLUE_20: '#1342B9',
    BLUE_30: '#0E318B',
```

Цвета были выбраны каждый по 5 оттенков, чтобы была возможность указывать состояния различных элементов. Нажатие – один цвет, кнопка заблокирована – другой цвет и т.д.

Следующий пример из класса типографики. «text60» – условное название для переменной которая содержит в себе стили: размер шрифта – 20 условных единиц (расчитываются в зависимости от устройства, не являются физическими пикселями). Вес шрифта 300, высота линии рассчитывается как размерность, умноженная на 1.2. Что касается самого семейства шрифтов, выбор пал на «Rubik», так как понравился внешне, не являлся гротескным и поддерживал большинство семейств языков, включая кириллицу.

```
text60 = {  
  fontSize: 20,  
  fontWeight: '300',  
  lineHeight: Math.floor(20 * 1.2),  
  fontFamily: 'Rubik-Regular'  
};
```

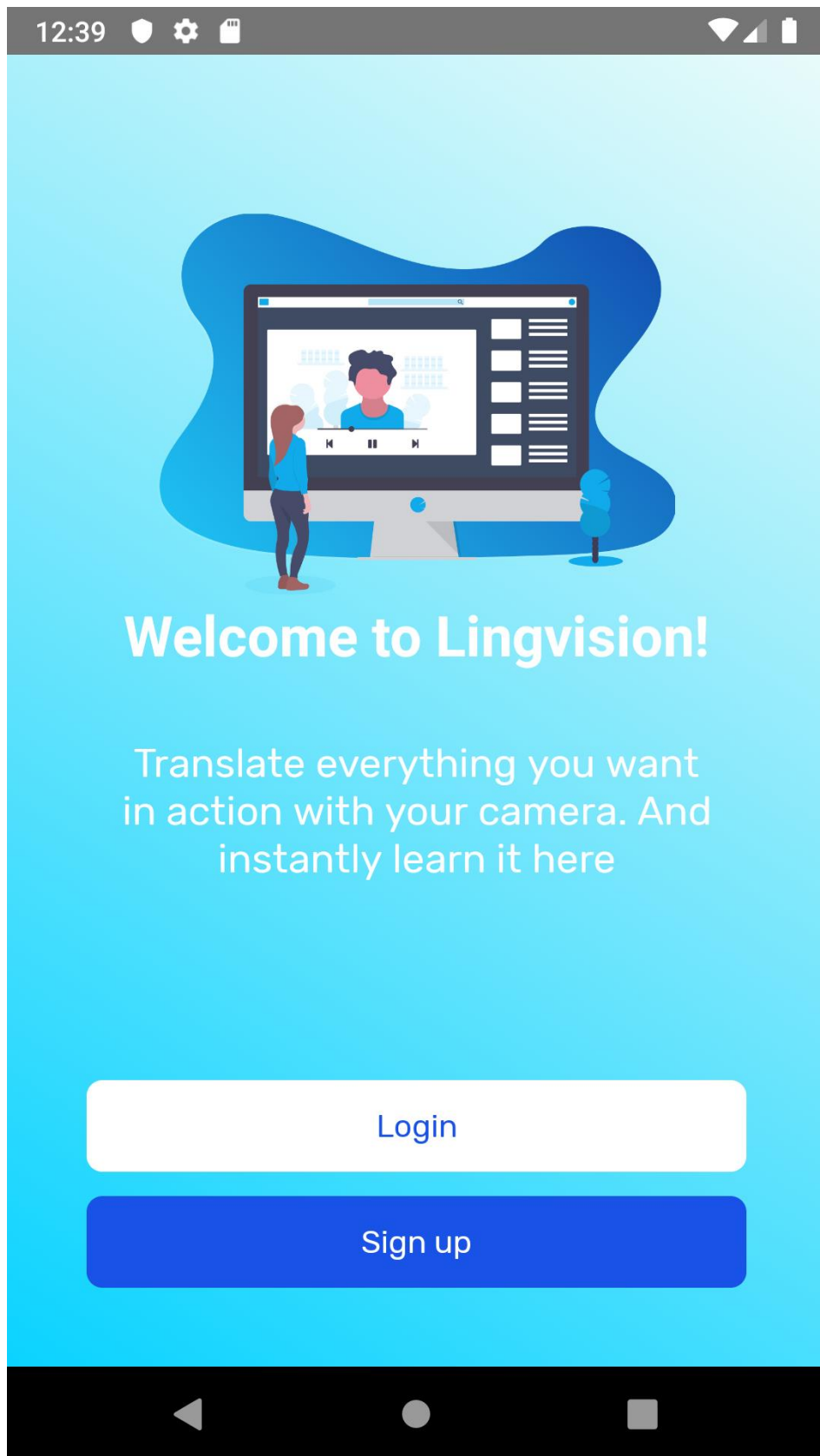
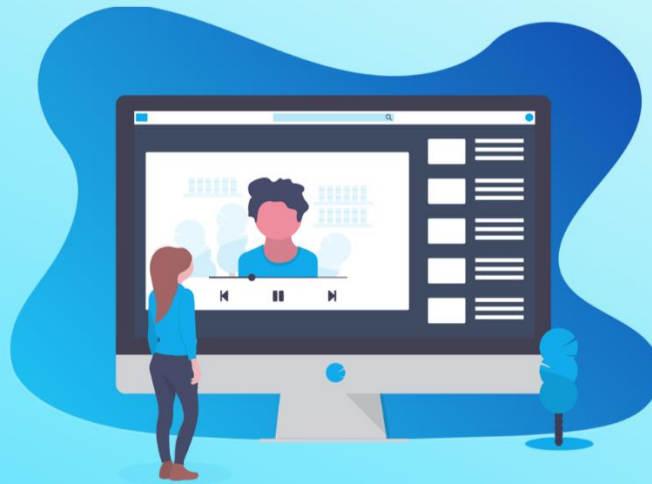


Рисунок 19 – Экран приветствия (Android)

12:43



Welcome to Lingvision!

Translate everything you want in
action with your camera. And
instantly learn it here

Login

Sign up

Рисунок 20 – Экран приветствия (IOS)

3.7. Демонстрация работы приложения

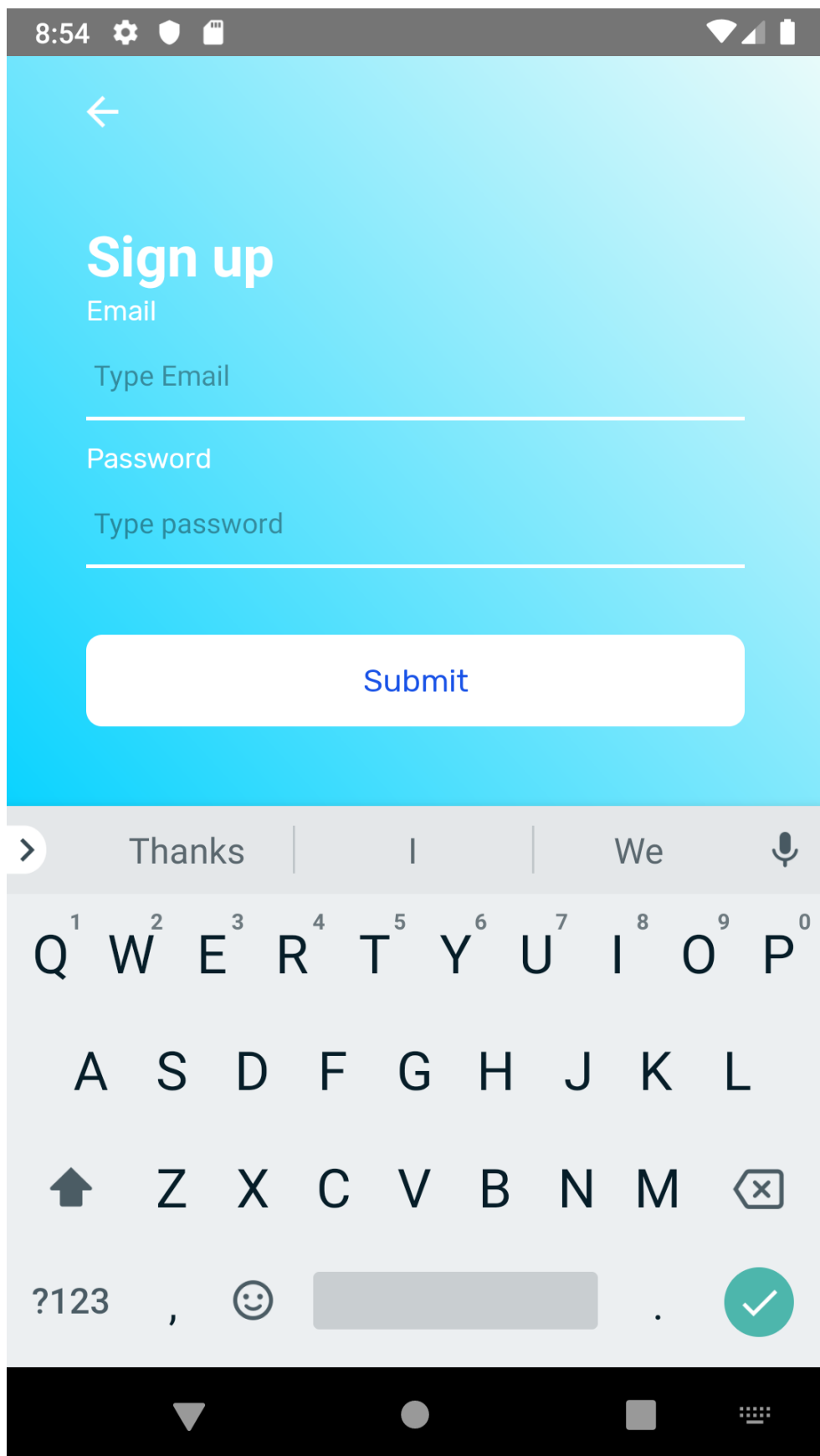


Рисунок 21 – Экран регистрации (Android)

На рисунке 21 изображен экран регистрации, позволяющий пользователю зарегистрироваться при помощи электронной почты. Стоит отметить, что дизайн данного экрана соответствует основам Material Design. Также стоит отметить, что обмен данными клиентской и серверной части осуществляется при помощи токенов, в зашифрованном виде, что обеспечивает безопасность персональных данных пользователей.

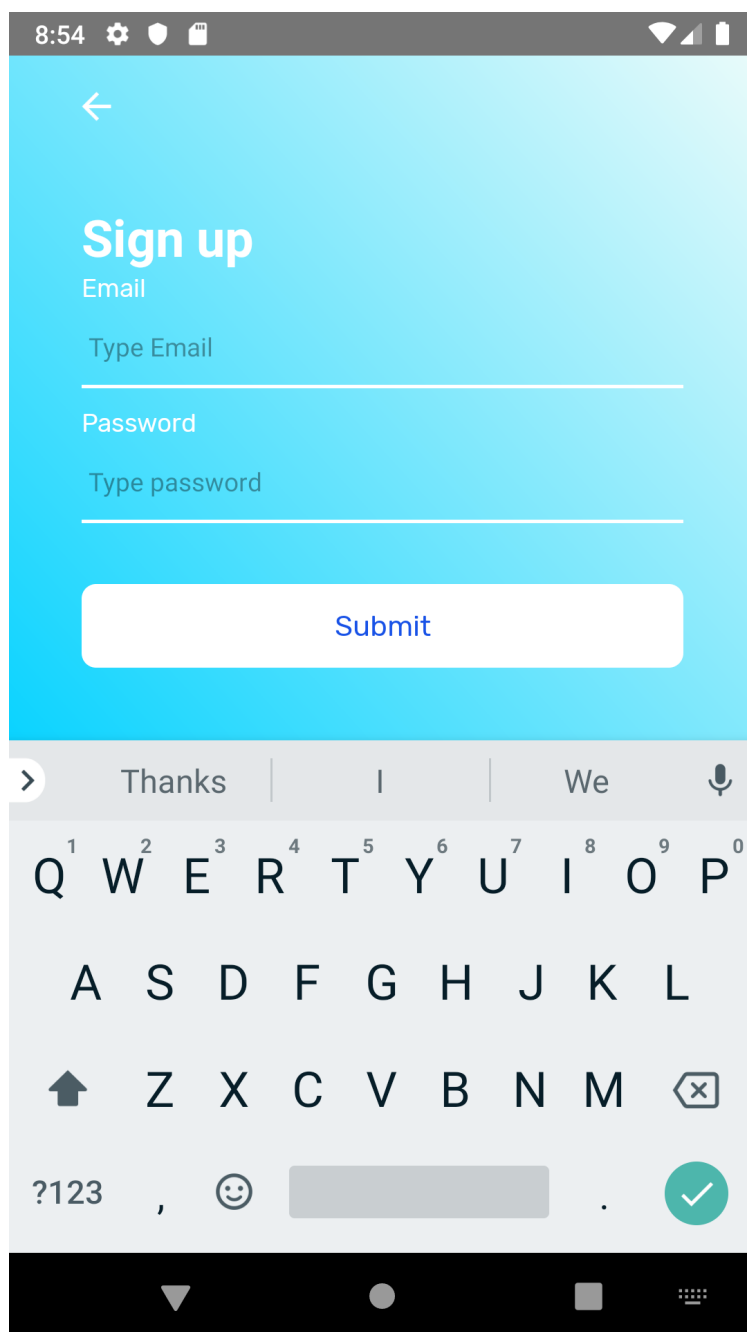


Рисунок 22 – Экран авторизации пользователя (Android)



Рисунок 23 – Демонстрация процесса перевода (до перевода)

На рисунке 23 видно изображение, которое выводится на экран через камеру.

На мониторе изображен английский текст.

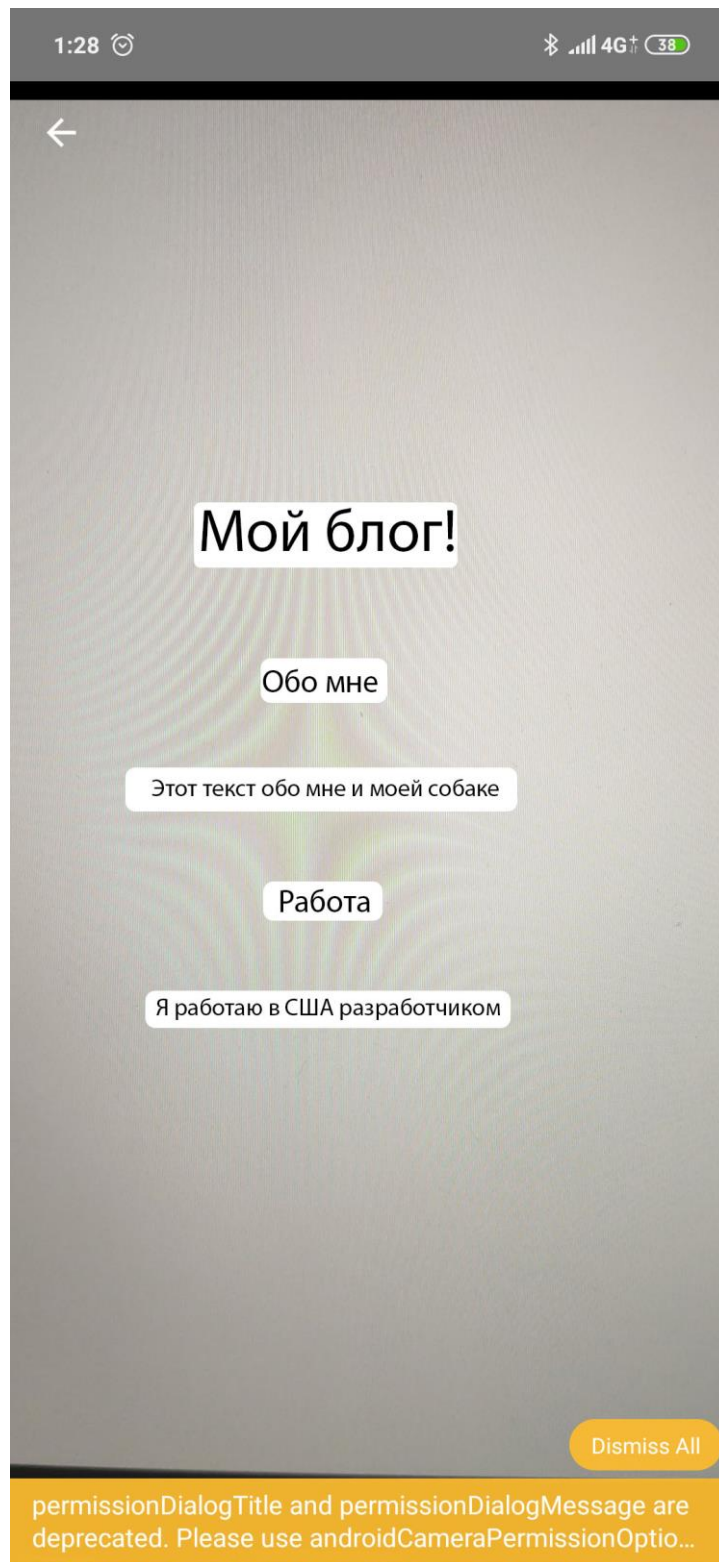


Рисунок 24 – Демонстрация процесса перевода (после перевода)

На рисунке 24 виден тот же текст, однако после процедуры перевода. Построчно на текст наложены блоки с переводом на русский язык. В силу не совершенности и незавершенности программной разработки, текст пока что не может быть переведен не построчно. Также важной проблемой технологии перевода

является слишком частый захват кадров и обновление текстового содержимого изображения, передаваемого с камеры, что негативно сказывается на качестве перевода.

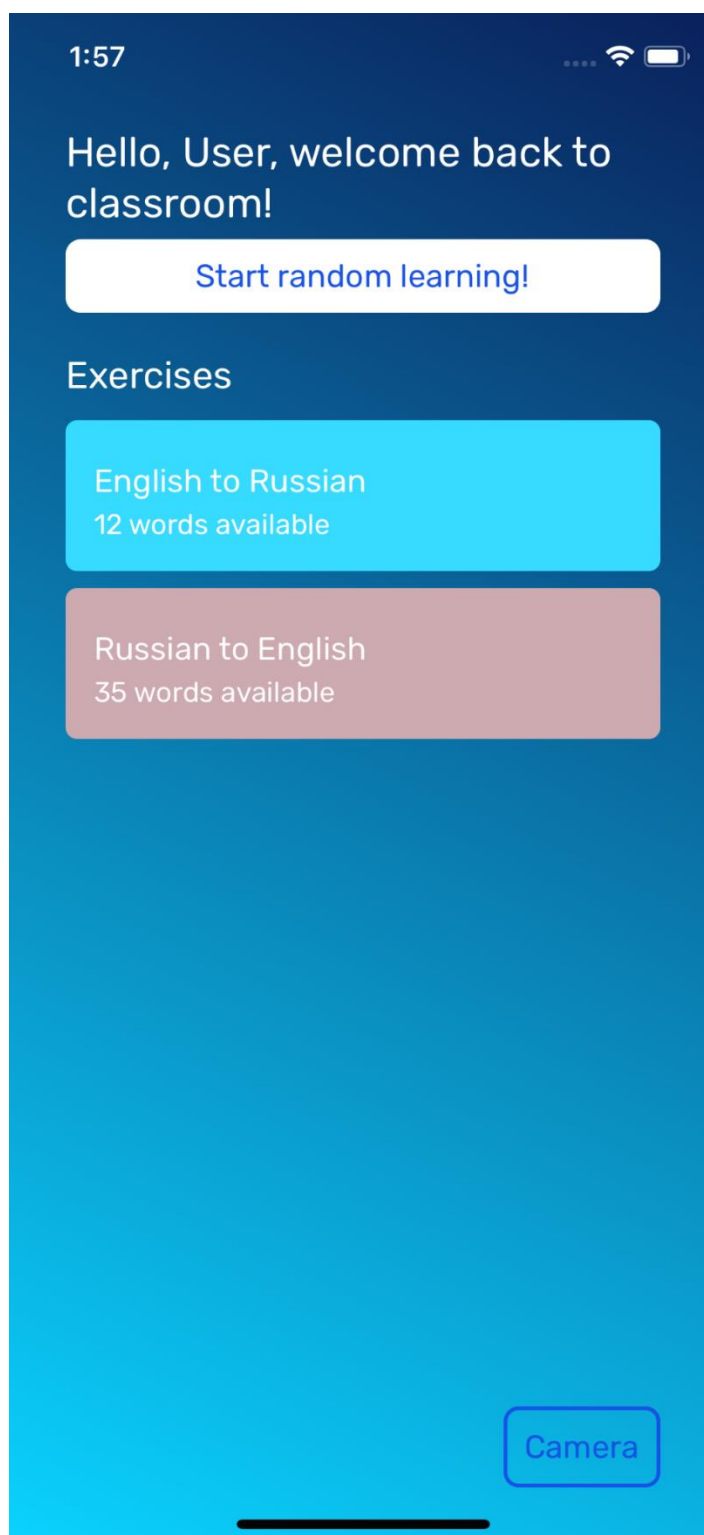


Рисунок 25 – Экран начала обучения(IOS)

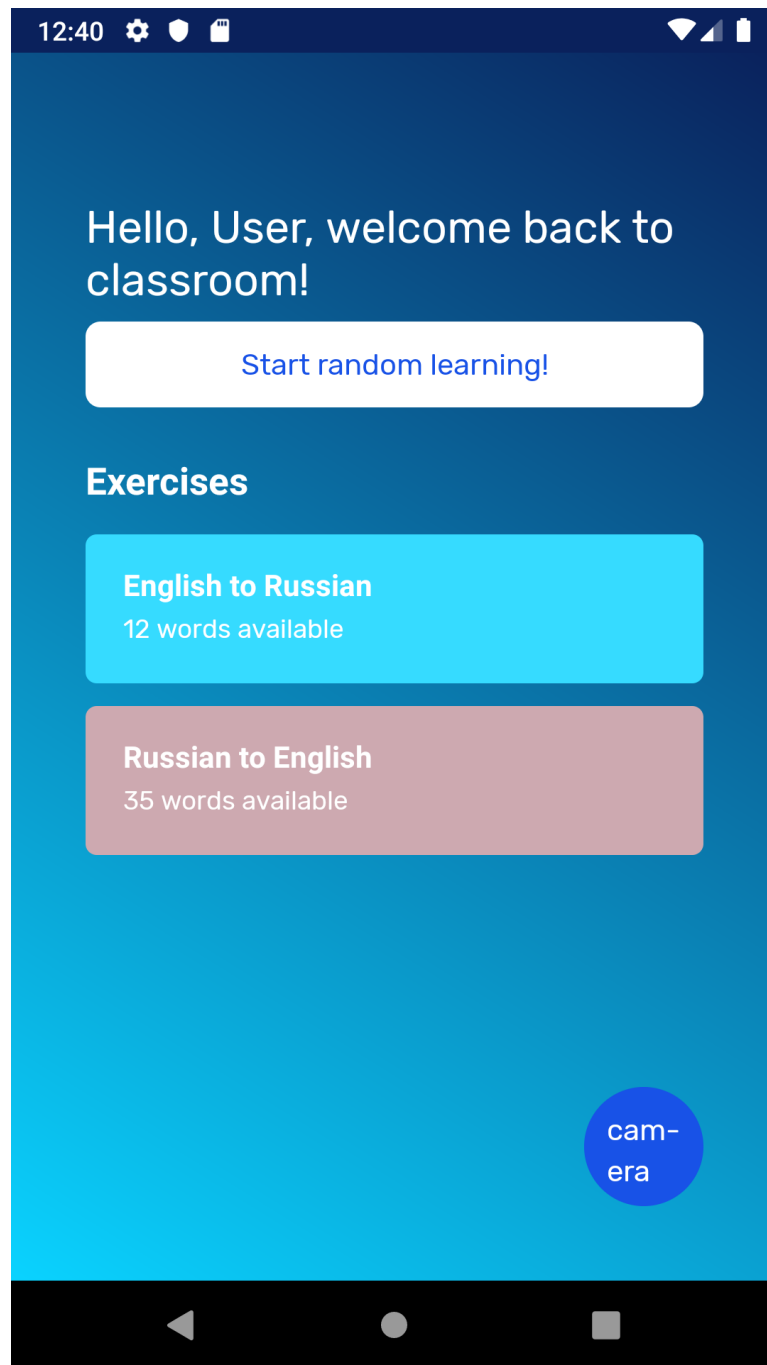


Рисунок 26 – Экран начала обучения (Android)

На данном экране пользователь может выбрать способы изучения новых слов. Была разработана возможность случайного выбора упражнений. Упражнения делятся по направленности перевода: с английского языка на русский язык и с русского языка на английский язык.

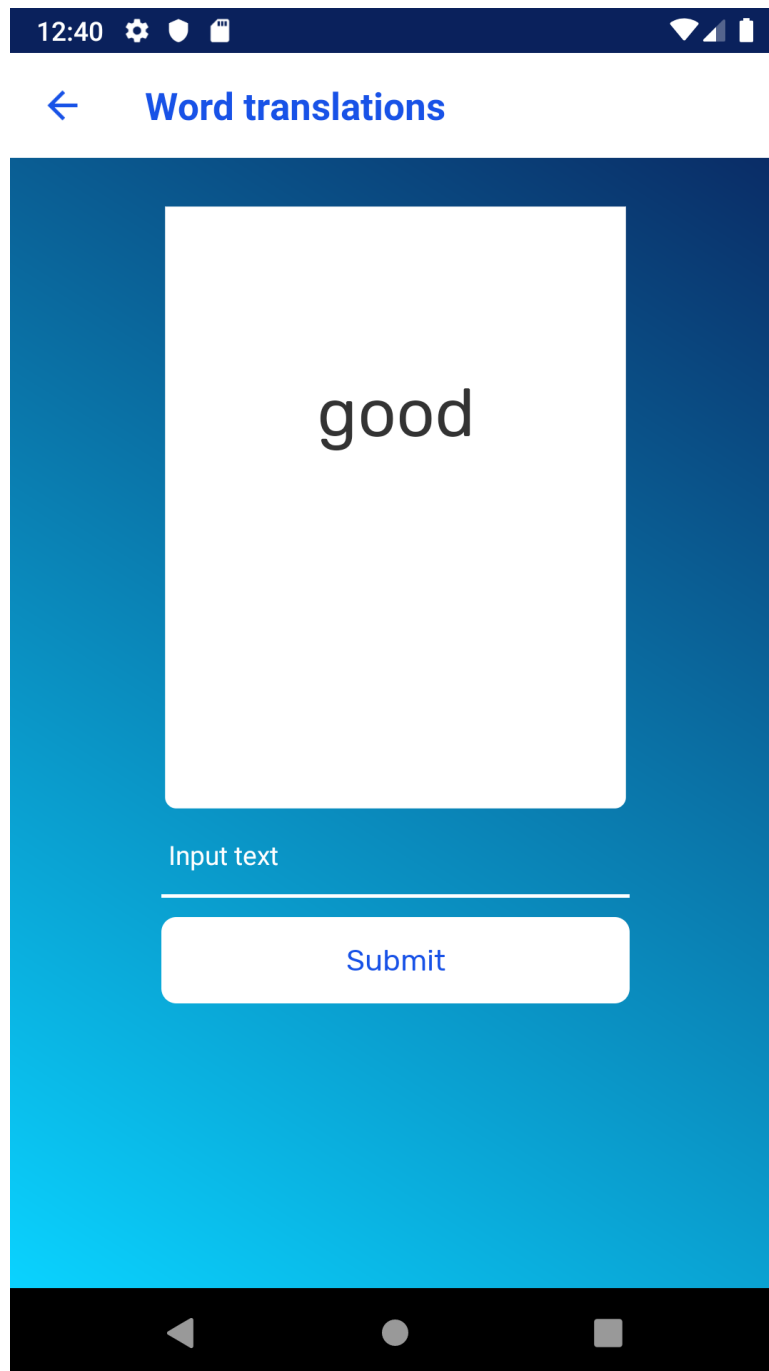


Рисунок 27 – Экран обучения с ручным вводом (Andoid)

На рисунке изображен один из вариантов упражнений, согласно которому пользователю придется ввести перевод предложенного слова самостоятельно.

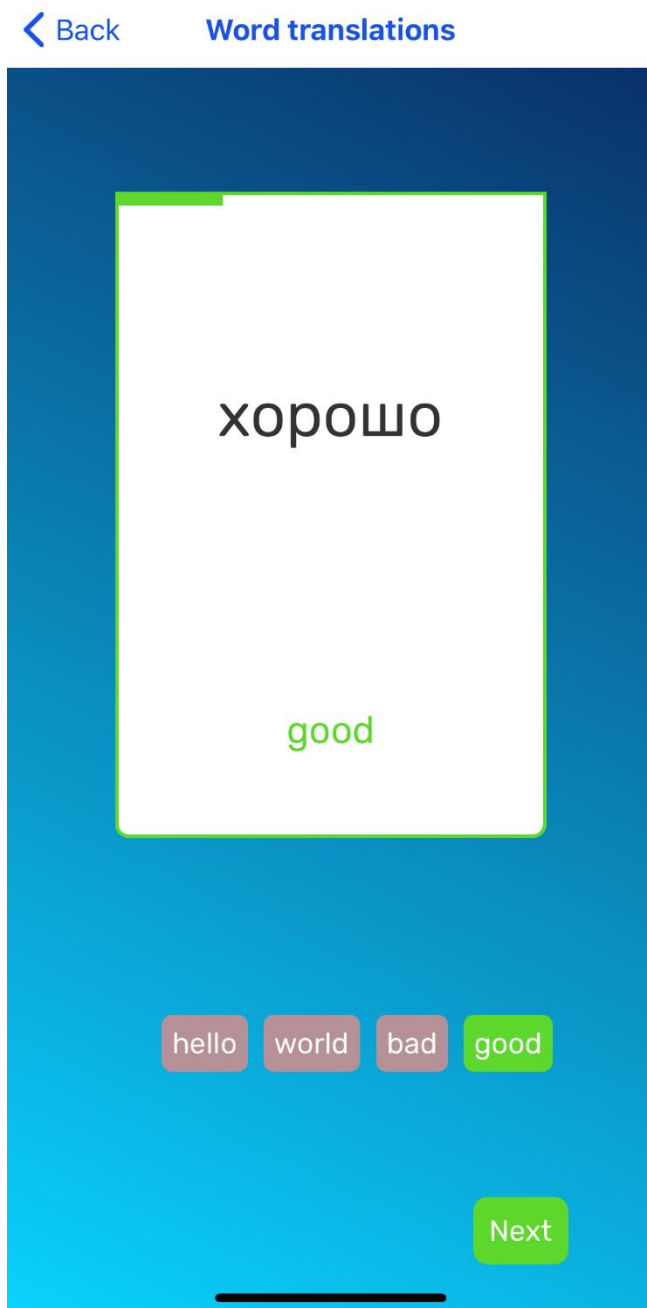


Рисунок 28 – Экран обучения с верно выбранным словом (IOS)

Также существует еще один вариант упражнения – выбор перевода слова из предложенного списка. После сделанного выбора пользователю выводится индикация правильности ответа. Зеленый, если ответ был верным и красный, если ответ был неверным.

< Back

Word translations

привет

hello

hello

good

world

bad

Next

Рисунок 29 – Экран с неверно выбранным словом (IOS)

Глава 4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Введение

Осуществляя любую научно-исследовательскую работу, необходимо провести ее экономическое обоснование. Экономическое обоснование включает в себя: определение потенциальных потребителей и сегмента рынка, сравнительный анализ с конкурентами, оценка себестоимости и определение трудоемкости выполнения работ. Это необходимо при поиске источников финансирования и оценке коммерческой ценности проекта.

Данный раздел помогает понять, будет ли продукт востребованным на рынке, сколько он будет стоить, какой срок понадобится для его реализации и так далее. Цель раздела: спроектировать и создать конкурентоспособные технологии, отвечающие современным требованиям в области ресурсоэффективности и ресурсосбережения.

4.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения

4.1.1. Потенциальные потребители результатов исследования

Одним из этапов создания переводчика с наличием возможности перевода в режиме реального времени является наполнение базы данных словаря. Это трудоёмкий процесс, требующий слаженной работы специалистов разного профиля. На данный момент интеграция иностранных языков в нашу жизнь происходит чаще и интенсивнее. Однако уровень владения Английским языком в России достаточно низок. [1] Электронные системы перевода текста являются одними из самых популярных источников значений иностранных слов, а также большинство электронных систем имеет набор возможностей для машинного перевода.

Необходимость понимания иностранной речи и письменности, а также выражения своих собственных мыслей на языке отличном от естественного сегодня возникает не только за границей, но и в родной стране. Перевод становится необходим во всех областях нашей жизни: начиная от производственных

предприятий и медицинских учреждений, заканчивая бытовыми вопросами. Множество продуктов питания, бытовой техники, специального оборудования и технической литературы производятся не в России, а за рубежом. Таким образом возникает необходимость перевода иностранного текста буквально каждый день.

В наши дни, почти у каждого есть смартфон, который позволяет воспользоваться возможностями машинного перевода. Также существует множество коммерческих проектов, разрабатывающих соответствующее программное обеспечение. Однако использование подобных информационных систем не всегда удовлетворяет требованиям современной жизни – постоянному дефициту свободного времени. Множество систем машинного перевода требуют непосредственной вовлеченности пользователя в процесс перевода, что отнимает время и делает возможным совершение множества ошибок, связанных с человеческим фактором. Проект осуществления перевода в дополненной реальности в режиме реального времени сводит к минимуму возможные ошибки, сокращает время необходимое для осуществления перевода, а также легко применим, как на производстве или в профессиональной области, так и в быту. Таким образом целевая аудитория данного проекта представляется довольно обширным количеством человек, имеющим доступ к смартфонам.

4.1.2. Анализ конкурентных технических решений

Конкуренция в сфере информационных технологий сейчас представляет собой напряженную борьбу. Для успешного занятия и удержания выбранной ниши, необходимо регулярно проводить анализ конкурирующих разработок. Оценивая сильные и слабые стороны проекта и конкурентов можно правильно выбрать вектор развития своего проекта.

Оценочная карта, представленная в таблице 1, позволяет наглядно анализировать разработки конкурентов в сравнении с собственным проектом, выделяя наиболее важные факторы.

Оценочная карта для сравнения конкурентных технических решений

№	Факторы конкурентоспособности товаров	Товары конкуренты				Важность b_j	Вес w_j
		Google Translate	Яндекс. Переводчик	Переводчик Translate.Ru	Свой проект		
1	Полнота базы	7/0,84	8/0,96	8/0,96	7/0,84	4	0,12
2	Скорость интерактива	5/0,75	6/0,9	5/0,75	10/1,5	5	0,15
3	Наличие возможности AR перевода	10/1,5	0/0	0/0	10/1,5	5	0,15
4	Возможность перевода по фото	10/1,2	10/1,2	10/1,2	10/1,2	4	0,12
5	Понятность интерфейса	5/0,75	7/1,05	5/0,75	9/1,35	5	0,15
6	Потребление оперативной памяти	5/0,3	5/0,3	9/0,54	5/0,3	2	0,06
7	Потребление сетевых ресурсов	5/0,3	8/0,48	8/0,48	4/0,24	2	0,06
8	Возможность сохранения в буфер обмена	10/1,2	10/1,2	7/0,84	10/1,2	4	0,12
9	Кроссплатформенность	9/0,81	9/0,81	5/0,45	5/0,45	3	0,09
	Итоговая оценка (сумма)	7,65	6,9	5,97	8,58	34	-

4.1.3. 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).

Таблица 4

SWOT-анализ

	<p>Сильные стороны научно-исследовательского проекта:</p> <p>С1. Использование передовых технологий</p> <p>С2. Востребованность продукта</p> <p>С3. Более низкая стоимость по сравнению с аналогами</p> <p>С4. Простота использования</p> <p>С5. Широкая клиентская направленность</p>	<p>Слабые стороны научно-исследовательского проекта:</p> <p>Сл1. Сложность разработки</p> <p>Сл2. Необходимость постоянной поддержки и обновления ПО после внедрения</p> <p>Сл3. Высокие аппаратные требования</p> <p>Сл4. Низкая известность продукта</p>
<p>Возможности:</p> <p>В1. Разработка новых возможностей программного продукта</p> <p>В2. Расширение области</p>	<p>Благодаря высокому спросу на приложения перевода, проект гарантированно будет вызывать интерес пользователей. Применение технологий моментального перевода возможно не только в быту, но и в условиях производства. А возможность разработки технологии перевода для определенных</p>	<p>Основными слабыми сторонами является сложность реализации проекта, которая включает не только трудоемкую разработку, но и постоянную поддержку проекта. К тому же, конкуренты в выбранной</p>

применения технологии В3. Монетизация посредством рекламы В4. Развитие иностранной клиентской базы	сфер деятельности повышает конкурентоспособность проекта.	сфере так же не стоят на месте и имеют больше финансирования и средств разработки.
Угрозы: У1. Выход на рынок новых конкурентов У2. Развитие старых конкурентов У3. Запрет на использование технологий дополненной реальности У4. Новые налоговые нагрузки в области IT	Большой угрозой можно считать развивающихся конкурентов, как новых, так и уже существующих. Технологии дополненной реальности являются одними из самых перспективных и имеют высокий потенциал.	Немаловажную роль в будущем проекта играют запретительные тенденции в законотворчестве Российской Федерации. При запрете использования иностранных технологий существование проекта в целом невозможно.

1. Планирование научно-исследовательских работ

4.2. Структура работ в рамках научного исследования

Для планирования комплекса предполагаемых работ необходимо выполнить следующие задачи:

1. Определить структуры работ в рамках научного исследования
2. Определить участников каждой работы

3. Установить продолжительность работ

4. Построить график проведения научных исследований

Таблица 5

Перечень этапов, работ и распределение исполнителей

№ работы	Наименование работы	Исполнители работы
1	Выбор научного руководителя бакалаврской работы	Каракулов А.А., Сухарев С.И., Тогызбаев Т.Н.
2	Составление и утверждение темы бакалаврской работы	Чердынцев Е.С., Каракулов А.А., Сухарев С.И., Тогызбаев Т.Н.
3	Составление календарного плана-графика выполнения бакалаврской работы	Чердынцев Евгений Сергеевич
4	Подбор и изучение литературы по теме бакалаврской работы	Каракулов А.А., Сухарев С.И., Тогызбаев Т.Н.
5	Анализ предметной области	Каракулов А.А., Сухарев С.И., Тогызбаев Т.Н.
6	Проектирование мобильного приложения	Каракулов А.А., Сухарев С.И., Тогызбаев Т.Н.
7	Разработка мобильного приложения	Каракулов А.А., Сухарев С.И., Тогызбаев Т.Н.
8	Тестирование мобильного приложения	Каракулов А.А., Сухарев С.И., Тогызбаев Т.Н.
9	Согласование выполненной работы с научным руководителем	Чердынцев Е.С., Каракулов А.А., Сухарев С.И., Тогызбаев Т.Н.
10	Выполнение других частей работы (финансовый менеджмент, социальная ответственность)	Каракулов А.А., Сухарев С.И., Тогызбаев Т.Н.
11	Подведение итогов, оформление работы	Каракулов А.А., Сухарев С.И., Тогызбаев Т.Н.

4.2.1. Определение трудоемкости выполнения работ

Трудовые затраты в большинстве случаев образуют основную часть стоимости разработки, поэтому важным моментом является определение трудоемкости работ каждого из участников научного исследования.

Для определения ожидаемого значения трудоемкости используется следующая формула:

$$t_{ож\ i} = \frac{3t_{min\ i} + 2t_{max\ i}}{5},$$

где $t_{ож\ i}$ – ожидаемая трудоемкость выполнения i -ой работы чел.-дн. ;

$t_{min\ i}$ – минимально возможная трудоемкость выполнения заданной i -ой работы, чел.-дн.;

$t_{max\ i}$ – максимально возможная трудоемкость выполнения заданной i -ой работы

Исходя из этого, определяется продолжительность каждой работы в рабочих днях T_p , учитывающая параллельность выполнения работ несколькими исполнителями.

$$T_{pi} = \frac{t_{ож\ i}}{Ч_i},$$

где T_{pi} – продолжительность одной работы, раб. дн;

$t_{ож\ i}$ – ожидаемая трудоемкость выполнения одной работы, чел.-дн.;

$Ч_i$ – численность исполнителей, выполняющих одновременно одну и ту же работу на данном этапе, чел.

Таблица 6

Временные показатели проведения научного исследования

Наименование работы	Исполнители работы	Трудоемкость работ, чел-дни			Длительность работ, дни	
		tmin	tmax	тож	Тр	Тк

Выбор научного руководителя бакалаврской работы	Каракулов Александр Андреевич	3	6	4,2	5	7
	Тогызбаев Тимур Нургалиевич					
	Сухарев Семен Игоревич					
Составление и утверждение темы бакалаврской работы	Каракулов Александр Андреевич	1	4	2,2	3	4
	Тогызбаев Тимур Нургалиевич					
	Сухарев Семен Игоревич	1	4	2,2	3	4
	Чердынцев Евгений Сергеевич					
Изучение технической литературы по бакалаврской работе	Каракулов Александр Андреевич	3	7	4,6	5	7
	Тогызбаев Тимур Нургалиевич					
	Сухарев Семен Игоревич					
Изучение предметной области	Каракулов Александр Андреевич	3	7	4,6	5	7
	Тогызбаев Тимур Нургалиевич					

	Сухарев Семен Игоревич					
Проектирование приложения	Каракулов Александр Андреевич	2	5	3,2	4	5
	Тогызбаев Тимур Нургалиевич					
	Сухарев Семен Игоревич					
Разработка программного продукта	Каракулов Александр Андреевич	15	20	17	17	21
	Тогызбаев Тимур Нургалиевич					
	Сухарев Семен Игоревич					
Тестирование программного продукта	Каракулов Александр Андреевич	3	6	4,2	5	7
	Тогызбаев Тимур Нургалиевич					
	Сухарев Семен Игоревич					
Экономический анализ работы	Каракулов Александр Андреевич	2	4	2,8	3	4
	Тогызбаев Тимур Нургалиевич					
	Сухарев Семен Игоревич					

Социальный анализ работы	Каракулов Александр Андреевич	2	4	2,8	3	4
	Тогызбаев Тимур Нургалиевич					
	Сухарев Семен Игоревич					
Создание отчета по бакалаврской работе	Каракулов Александр Андреевич	5	9	6,6	7	9
	Тогызбаев Тимур Нургалиевич					
	Сухарев Семен Игоревич					
Оценка результатов проведенной работы	Каракулов Александр Андреевич	2	4	2,8	3	4
	Тогызбаев Тимур Нургалиевич					
	Сухарев Семен Игоревич					
	Чердынцев Евгений Сергеевич					

Продолжительности работ исполнителей

Длительность работ в рабочих днях		Длительность работ в календарных днях	
Руководитель	Лаборант	Руководитель	Лаборант
7	60	7	79

4.2.2. Разработка графика проведения научного исследования



Рисунок 30 – Диаграмма Ганта

4.3. Бюджет научно-технического исследования (НТИ)

4.3.1. Расчет затрат на специальное оборудование для научных (экспериментальных) работ

В процессе работы над проектом планируется использование персонального компьютера и телефона, купленных заранее, поэтому стоит рассчитать их амортизацию. Срок полезного использования составляет 3 года. Время написания ВКР – 5 месяцев. Тогда норма амортизации:

$$A_H = \frac{1}{n} * 100\% = \frac{1}{3} * 100\% = 33,33\%$$

Годовые амортизационные отчисления:

$$A_{\Gamma} = 100000 * 0,33 = 33\ 000 \text{ рублей}$$

Ежемесячные амортизационные отчисления:

$$A_{\text{м}} = \frac{A_{\Gamma}}{12} = \frac{33000}{12} = 2750 \text{ рублей}$$

Таблица 8

Расчет бюджета затрат на приобретение спецоборудования для научных работ

№	Наименование	Кол-во единиц	Цена единицы, руб.	Общая стоимость, тыс. руб.	Амортизационные отчисления
1	Персональный компьютер (собственная сборка)	1	70 000	70 000	9625
2	ПО Microsoft office	1	4 684	4 684	644
3	Телефон Samsung Galaxy s8	1	30 000	30 000	4125
Итого:				104 684	14 394

4.3.2. Основная заработная плата исполнителей темы

Таблица 9

Баланс рабочего времени

Показатели рабочего времени	Студент	Руководитель
Календарные дни	365	
Нерабочие дни (праздники/выходные)	66	
Потери рабочего времени		
-отпуск	56	56
-невыходы по болезни	12	2
Действительный годовой фонд рабочего времени	231	241

Затраты на заработную плату рассчитываются по следующей формуле:

$$Z_n = Z_{осн} + Z_{доп}, \text{ где}$$

$Z_{доп}$ – дополнительная заработная плата, руб;

Зосн – основная заработная плата, руб.

$$Зосн = Здн * Тр * (1 + Кпр + Кд) * Кр, \text{ где}$$

Здн – среднедневная заработная плата, руб.;

Кпр – премиальный коэффициент (0,3);

Кд – коэффициент доплат и надбавок (0,3-0,5);

Кр – районный коэффициент (для Томска 1,3)

Тр – продолжительность работ, выполняемых работником, раб. дни

Среднедневная заработная плата:

$$Здн = \frac{Зм * М}{Fд}$$

Зм – месячный оклад работника, руб.;

М – количество месяцев работы без отпуска в течение года (для 6-дневной рабочей недели М=10,4);

Fд – действительный годовой фонд рабочего времени персонала, раб.дн.

Таблица 10

Расчет основной заработной платы

Исполнители	Здн, руб	Кпр	Кд	Кр	Тр	Зосн
Студент 1	979,7	0,3	0,3	1,3	60	122 266
Студент 2	979,7	0,3	0,3	1,3	60	122 266
Студент 3	979,7	0,3	0,3	1,3	60	122 266
Научный руководитель	1452,7	0,3	0,4	1,3	7	21 151
Итого:						387949

4.3.3. Дополнительная заработная плата исполнителей темы

Таблица 11

Расчет дополнительной заработной платы

Исполнитель	Основная заработная плата, руб.	Коэффициент дополнительной заработной платы	Дополнительная заработная плата, руб
Студент 1	122 266	0,15	18 339,9
Студент 2	122 266		18 339,9

Студент 3	122 266		18 339,9
Руководитель	21 151		3172,7
Итого:			58192,4

4.3.4. Отчисления во внебюджетные фонды (страховые отчисления)

Таблица 12

Расчет отчислений во внебюджетные фонды

Исполнитель	Основная заработная плата + дополнительная, руб.	Коэффициент отчислений во внебюджетные фонды	Сумма отчислений
Студент	140 606	0,28	39 370
Студент	140 606		39 370
Студент	140 606		39 370
Руководитель	40 612,98		6811
Итого:			124 921

4.3.5. Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопия материалов, оплата услуг связи, электроэнергии и т.д.

$\text{Знакл} = (\text{сумма статей 1-4}) * \text{Кнр}$, где

Кнр – коэффициент, учитывающий накладные расходы (16%)

4.3.6. Формирование бюджета затрат научно-исследовательского проекта

Таблица 13

Бюджет затрат НИИ

Наименование статьи	Сумма, руб	Примечание
Амортизационные затраты на спецоборудование	14 394	Таблица 8
Затраты на основную заработную плату	387949	Таблица 10

Затраты на дополнительную заработную плату	58192,4	Таблица 11
Затраты на отчисление во внебюджетные фонды	124921	Таблица 12
Накладные расходы	93673,02	16% от суммы статей 1-4
Бюджет затрат НИИ	679 129	Сумма статей 1-5

4.4. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

Определение эффективности проекта происходит на основе расчета интегрального показателя эффективности научного исследования. Основные показатели следующие:

Интегральный показатель финансовой эффективности получают в ходе оценки бюджета затрат нескольких вариантов выполнения исследования.

Интегральный финансовый показатель определяется так:

$$I_{\text{фин } p}^{\text{исп } i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}},$$

Где $I_{\text{фин } p}^{\text{исп } i}$ – интегральный финансовый показатель разработки;

Φ_{pi} – стоимость i -го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта.

Интегральный показатель ресурсоэффективности определяется следующим образом:

$$I_{pi} = \sum a_i * b_i$$

Где I_{pi} – интегральный показатель ресурсоэффективности для i -го варианта исполнения разработки;

a_i – весовой коэффициент i -го варианта исполнения разработки;

b_i – бальная оценка i -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

n – число параметров сравнения.

Таблица 14

Сравнительная оценка характеристик вариантов исполнения проекта

Критерии \ Объект исследования	Весовой коэффициент параметра	Исп. 1	Исп. 2	Исп. 3
Способствует росту производительности труда пользователя	0,2	5	3	4
Удобство в эксплуатации (соответствует требованиям потребителей)	0,3	4	3	3
Помехоустойчивость	0,1	4	5	2
Энергосбережение	0,1	4	5	3
Надежность	0,2	3	5	5
Материалоемкость	0,1	4	4	5
Итого:	1			

$$I_{p-ucn1} = 5 * 0,2 + 4 * 0,3 + 4 * 0,1 + 4 * 0,1 + 3 * 0,2 + 4 * 0,1 = 4$$

$$I_{p-ucn2} = 3 * 0,2 + 3 * 0,3 + 5 * 0,1 + 5 * 0,1 + 5 * 0,2 + 4 * 0,1 = 3,9$$

$$I_{p-ucn3} = 4 * 0,2 + 3 * 0,3 + 2 * 0,1 + 3 * 0,1 + 5 * 0,2 + 5 * 0,1 = 3,7$$

Интегральный показатель эффективности вариантом использования определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{испi} = \frac{I_{p-испi}}{I_{финр. i}}$$

Сравнительная эффективность проекта:

$$\mathcal{E}_{cp} = \frac{I_{исп1}}{I_{исп2}}$$

Сравнительная эффективность разработки

№	Показатели	Исп. 1	Исп. 2	Исп. 3
1	Интегральный финансовый показатель разработки	1	0,95	0,9
2	Интегральный показатель ресурсоэффективности разработки	4	3,9	3,7
3	Интегральный показатель эффективности	4	4,1	4,11
4	Сравнительная эффективность вариантов исполнения	0,97	0,99	1

В данном разделе были определены финансовый показатель разработки, показатель ресурсоэффективности, интегральный показатель эффективности и, на основании сравнительной эффективности вариантов исполнения, оптимальным был выбран 3 вариант исполнения.

Глава 5. Социальная ответственность

Введение

Научно-исследовательская работа заключается в написании программного обеспечения, осуществляющего распознавание и перевод текста в дополненной реальности, а также создание списков переведенных слов для дальнейшего их изучения. Во время написания программного обеспечения были разработаны специальные алгоритмы распознавания текста на плоскости, перевода посредством интернета, размещения переведенного текста обратно на видимую для пользователя часть, а также для сохранения слов. Прикладная область данного приложения не ограничена: приложения может применяться, как на производстве, так и в быту, оставаясь при этом в равной степени эффективным. Приложение также обладает большими перспективами специализации под различные области применения. Расширяя возможности использования можно сделать приложение полезнее, например, добавив считывание штрих-кода или QR-кода и внедрив данное приложение в складскую систему хранения.

Так как работа, в большей своей части, была связана с персональным компьютером и смартфоном, то стоит отметить несколько вредных факторов при работе с вышеперечисленными приборами:

1. Мерцание монитора, связанное с вертикальным обновлением картинки с частотой 60 Гц, приводящее к повышенной утомляемости глаз;
2. Постоянное электромагнитное излучение от всех составляющих персонального компьютера, включенных в сеть;
3. Постоянное положение сидя, нарушающее кровообращение в шейно-воротниковой и поясничной зонах спины;
4. Коротковолновое излучение от смартфона;
5. Другие факторы.

5.1. Правовые и организационные вопросы обеспечения безопасности

5.1.1. Организационные мероприятия обеспечения безопасности

Рабочее место должно быть организовано с учетом требований ГОСТ 12.2.032-78 «ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования» и СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».

Некоторые, наиболее важные пункты из ГОСТ (нумерация пунктов сохранена в соответствие с ГОСТ 12.2.032-78 «ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования»):

2.4.1. Конструкция регулируемого кресла оператора должна соответствовать требованиям ГОСТ 21889-76.

4.2. Очень часто используемые средства отображения информации, требующие точного и быстрого считывания показаний, следует располагать в вертикальной плоскости под углом $\pm 15^\circ$ от нормальной линии взгляда и в горизонтальной плоскости под углом $\pm 15^\circ$ от сагиттальной плоскости.

СанПиН (нумерация пунктов сохранена в соответствие с СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы»):

3.2. Естественное и искусственное освещение должно соответствовать требованиям действующей нормативной документации. Окна в помещениях, где эксплуатируется вычислительная техника, преимущественно должны быть ориентированы на север и северо-восток.

3.4. Площадь на одно рабочее место пользователей ПЭВМ с ВДТ на базе электронно-лучевой трубки (ЭЛТ) должна составлять не менее 6 м СанПиН 2.2.2/2.4.1340-03, в помещениях культурно-развлекательных учреждений и с ВДТ на базе плоских дискретных экранов (жидкокристаллические, плазменные) - 4,5 м СанПиН 2.2.2/2.4.1340-03.

3.7. Помещения, где размещаются рабочие места с ПЭВМ, должны быть оборудованы защитным заземлением (занулением) в соответствии с техническими требованиями по эксплуатации.

4.2. В производственных помещениях, в которых работа с использованием ПЭВМ является основной (диспетчерские, операторские, расчетные, кабины и посты управления, залы вычислительной техники и др.) и связана с нервно-эмоциональным напряжением, должны обеспечиваться оптимальные параметры микроклимата для категории работ 1а и 1б в соответствии с действующими санитарно-эпидемиологическими нормативами микроклимата производственных помещений. На других рабочих местах следует поддерживать параметры микроклимата на допустимом уровне, соответствующем требованиям указанных выше нормативов.

5.1. В производственных помещениях при выполнении основных или вспомогательных работ с использованием ПЭВМ уровни шума на рабочих местах не должны превышать предельно допустимых значений, установленных для данных видов работ в соответствии с действующими санитарно-эпидемиологическими нормативами.

6.3. Освещенность на поверхности стола в зоне размещения рабочего документа должна быть 300-500 лк. Освещение не должно создавать бликов на поверхности экрана. Освещенность поверхности экрана не должна быть более 300 лк.

9.4. Экран видеомонитора должен находиться от глаз пользователя на расстоянии 600-700 мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов.

10.1. Высота рабочей поверхности стола для взрослых пользователей должна регулироваться в пределах 680-800 мм; при отсутствии такой возможности высота рабочей поверхности стола должна составлять 725 мм.

10.6. Клавиатуру следует располагать на поверхности стола на расстоянии 100-300 мм от края, обращенного к пользователю, или на специальной, регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.

Данная работа была выполнена при помощи персонального компьютера в помещении с естественным и искусственным освещением, оборудованном

компьютерными столом и креслом. При выполнении работы все требования, предусматриваемые ГОСТ 12.2.032-78 были соблюдены. Выполнение выпускной квалификационной работы не несли в себе угрозы для здоровья и жизни сотрудников, а также не представляли опасности для окружающей среды.

5.1.2. Особенности законодательного регулирования проектных решений

На основании федерального закона Российской Федерации «О персональных данных», №152-ФЗ, глава 2, статья 5:

1. Обработка персональных данных должна осуществляться на законной и справедливой основе.

2. Обработка персональных данных должна ограничиваться достижением конкретных, заранее определенных и законных целей. Не допускается обработка персональных данных, несовместимая с целями сбора персональных данных.

3. Не допускается объединение баз данных, содержащих персональные данные, обработка которых осуществляется в целях, несовместимых между собой.

4. Обработке подлежат только персональные данные, которые отвечают целям их обработки.

5. Содержание и объем обрабатываемых персональных данных должны соответствовать заявленным целям обработки. Обрабатываемые персональные данные не должны быть избыточными по отношению к заявленным целям их обработки.

6. При обработке персональных данных должны быть обеспечены точность персональных данных, их достаточность, а в необходимых случаях и актуальность по отношению к целям обработки персональных данных. Оператор должен принимать необходимые меры либо обеспечивать их принятие по удалению или уточнению неполных, или неточных данных.

7. Хранение персональных данных должно осуществляться в форме, позволяющей определить субъекта персональных данных, не дольше, чем этого требуют цели обработки персональных данных, если срок хранения персональных данных не установлен федеральным законом, договором, стороной которого, выгодоприобретателем или поручителем, по которому является субъект персональных данных. Обработываемые персональные данные подлежат уничтожению либо обезличиванию по достижении целей обработки или в случае утраты необходимости в достижении этих целей, если иное не предусмотрено федеральным законом.

Проанализировав данную статью закона, было принято решения отказаться от сбора биометрических данных о пользователе, а также ввести обязательное информирование пользователя о сборе других персональных данных и электронное согласие перед процедурой сбора данных.

5.2. Производственная безопасность

В данном пункте производится анализ вредных и опасных факторов, которые могут возникнуть на одном из этапов выполнения работы.

Отказ от рассмотрения химических и шумовых факторов обусловлен незначительностью их наличия на рабочем месте или их полным отсутствием.

Таблица 16

Возможные опасные и вредные факторы

Факторы (по ГОСТ 12.0.003-74)	Этапы работ			Нормативные документы
	Проектирование	Программирование	Эксплуатация	
Отклонение показателей микроклимата	+	+	+	СанПиН 2.2.2.548-96
Отсутствие или недостаток естественного света	+	+	+	СНиП 23-05-95*

Недостаточная освещенность рабочей зоны	+	+	+	СанПиН 2.2.1/2.1.1.1278-03
Повышенный уровень электромагнитных излучений	+	+	+	СанПиН 2.2.4/2.1.8.055-96.

5.2.1. Анализ опасных и вредных производственных факторов

Таблица 17

Влияние опасных и вредных факторов

Фактор	Источник	Воздействие	Допустимые нормы
Отклонение показателей микроклимата	Кондиционеры/их отсутствие, увлажнители воздуха/их отсутствие	Вялость, усталость, сниженная концентрация и внимание	Таблица 3
Отсутствие или недостаток естественного света	Отсутствие окон	Ухудшение зрения, усталость глаз	КЕО не ниже 1,2%-1,5%
Недостаточная освещенность рабочей зоны	Отсутствие или неподходящие лампы	Ухудшение зрения, усталость глаз	Освещенность на рабочей поверхности от системы общего искусственного освещения 200-300 лк.
Повышенный уровень электромагнитных излучений	Все части компьютера	Возможно возникновение рака	Напряженность электростатического поля не более 20 кВ/м

Для описания допустимых норм микроклимата невозможно ограничиться одним параметром, так как микроклимат включает в себя температуру помещения, температуру поверхностей, влажность воздуха и скорость движения воздуха.

Допустимые величины показателей микроклимата по СанПиН 2.2.2.548-96

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	20,0-21,9	19,0-26,0	15-75	0,1
Теплый	21,0-22,9	20,0-29,0		

5.2.2. Обоснование мероприятий по снижению уровней воздействия опасных и вредных факторов на исследователя (работающего)

Для восстановления и поддержания допустимого микроклимата необходимо придерживаться следующих правил:

- Оборудование помещения системами обогрева, вентилирования и увлажнения
- Защита фасада здания от солнца: шторы, жалюзи, навесы и т.д.
- Рационально размещать рабочие места
- Своевременная влажная уборка помещения

Для решения проблемы отсутствия или недостатка естественного света и плохой освещенности рабочего места подходят следующие пункты:

- Сокращение времени работы
- Своевременная чистка стекол в светопроемах
- Снос деревьев, препятствующих проникновению света в помещение
- Ремонт помещения в светлых тонах
- Установка более мощных ламп или в большем количестве
- Установка ламп в правильном положении

Повышенный уровень электромагнитных излучений можно избежать, если следовать следующим пунктам:

- выключать монитор каждый раз, когда отходите от рабочего места;
- использовать монитор с жидкокристаллическим экраном;

- располагать монитор в углу помещения для того, чтобы стены поглощали излучение;
- выключать компьютер при его неиспользовании;
- сокращать время, проводимое за компьютером.

5.3. Экологическая безопасность

Экологической безопасностью называется комплекс мероприятий по снижению негативных влияний производственной и общественной деятельности человека на окружающую среду и защиту человека от последствий этого влияния.

Буквально вся исследовательская работа выполнена с использованием персонального компьютера и смартфона. Влияние работы персональных компьютеров на экологическое состояние планеты сложно недооценить. Наибольший вред от них в работе – потребление электроэнергии и создание электромагнитного поля. Персональный компьютер, как и смартфон не производят выбросов вредных веществ, не создают излучения, способного нарушить экологическую безопасность природы. Однако их производство и утилизация составляют серьезную проблему: текстолит, используемый при производстве микросхем имеет очень большой срок разложения (более 1000 лет).

Также при производстве смартфонов и персональных компьютеров используются тяжелые, щелочноземельные металлы, ртуть, пластик и стекло, что без должной утилизации по окончании службы попадает в природу и остается в не переработанном виде от века до полутора тысяч лет.

Мероприятия, позволяющие сохранять экологическую безопасность находясь на своем рабочем месте:

- Правильная утилизация персональных компьютеров и смартфонов, а также их комплектующих;
- Использование энергосберегающих ламп;
- Использование аккумуляторов вместо солевых батареек.

5.4. Безопасность в чрезвычайных ситуациях

Чрезвычайная ситуация (ЧС) – обстановка на определенной территории, сложившаяся в результате аварии, опасного природного явления, катастрофы, стихийного или иного бедствия, которая может повлечь за собой человеческие жертвы, а также ущерб здоровью человека или окружающей среде, значительные материальные потери и нарушение условий жизнедеятельности.

5.4.1. Пожар

Научно-исследовательская работа проходила в помещении, подходящем под определение офиса. Наиболее возможная чрезвычайная ситуация, при работе в офисе – пожар. К пожару могут привести неисправности в технических средствах, оргтехнике, а также действия самих сотрудников. Главное во время пожара – не поддаваться панике и действовать согласно правилам поведения при пожаре. Для сотрудника существует порядок действий и правила поведения в подобной чрезвычайной ситуации:

1. Заметив пожар или загорание, необходимо немедленно организовать оповещение об этом всех находящихся в здании людей, независимо от размеров и места пожара или загорания, равно как и при обнаружении хотя бы малейших признаков горения (дыма, запаха гари) и немедленно вызвать пожарную охрану по телефону «01». Очевидно, что быстрота прибытия пожарной помощи, позволит успешнее ликвидировать пожар и быстрее помочь людям, находящимся в опасности.

2. Сообщения о пожаре, как правило, передаются по телефону. Поэтому каждый человек должен хорошо знать места расположения телефонных аппаратов, особенно тех, которые доступны в любое время суток. Следует помнить, что с помощью сотового телефона можно вызвать помощь даже при отсутствии денег на счете или SIM-карты по номеру «112».

3. Каждый работник образовательного учреждения, обнаруживший пожар или его признаки (задымление, запах горения или тления различных материалов, повышение температуры и т.п.) обязан:

4. немедленно сообщить об этом по телефону в пожарную часть (при этом необходимо четко назвать адрес учреждения, место возникновения пожара, а также сообщить свою должность, фамилию и номер своего телефона);

5. задействовать систему оповещения людей о пожаре, приступить самому и привлечь других лиц к эвакуации детей из здания в безопасное место согласно плану эвакуации;

6. известить о пожаре руководителя образовательного учреждения или заменяющего его работника;

7. организовать встречу пожарных подразделений, принять меры по тушению пожара имеющимися в учреждении средствами пожаротушения.

5.4.2. Теракт

В помещении, где выполняется работа возможны акты терроризма, в том числе захват заложников. Оказавшийся в заложниках человек сначала не может поверить в то, что это произошло и адекватно оценить ситуацию. Как показывает анализ поведения лиц, оказавшихся в заложниках, у некоторых из них может возникнуть неуправляемая реакция протеста против совершаемого насилия. Важно не терять самообладание, так как в этой ситуации террористы часто убивают взбунтовавшихся. Террористы, как правило, находятся в состоянии сильнейшего стресса и поэтому крайне агрессивны.

Вывод по разделу «Социальная ответственность»

Подводя итоги можно отметить, что выполнение проекта соответствовало всем заявленным нормам безопасности жизнедеятельности. Рабочее место во время проведения исследовательской работы соответствовало региональным стандартам, а также санитарно-эпидемиологическим правилам и нормам. Особенно стоит отметить соблюдение федерального закона «О персональных данных» и готовность участников к чрезвычайным ситуациям.

Заключение

Результатом проведения научно-исследовательской работы стало проведение анализа существующих средств перевода при помощи программных средств. Было выявлено, что лишь небольшая часть программных средств обладает возможностью перевода в дополненной реальности, и ни один сервис не обладает одновременно и возможностью дальнейшего обучения, и возможностью перевода в дополненной реальности.

Разработанная система не только позволяет сократить время, затрачиваемое пользователем на перевод, но и позволяет в дальнейшем неоднократно возвращаться к уже переведенным словам, с целью дополнительного обучения. Система позволяет изучать новые слова, которые встречались пользователю в реальной жизни.

Также разработанная система является кроссплатформенной, а именно выполняется на платформах IOS и Android, что значительно увеличивает целевую аудиторию пользователей.

Во рамках проведения научно-исследовательской работы был выполнен финансовый анализ системы, который показал конкурентное преимущество, обусловленное расширенными вариантами использования.

Работа проходила в условиях, соответствующих всем нормативным требованиям и не нарушала экологической безопасности.

Список используемых источников

1. Начало работы с NodeJS [Электронный ресурс]. – Режим доступа: <https://medium.com/devschacht/node-hero-chapter-1-239f7afeb1d1> (дата обращения: 26.05.2019);
2. Смольникова, Д.Б. Технологии обработки текста компьютерными системами / Д.Б. Смольникова // Аллея Науки.— 2018.— № 7.— С. 986–991.
3. Машинный перевод от холодной войны до глубокого обучения [Электронный ресурс] – Режим доступа: <https://vc.ru/future/32616-mashinnyu-perevod-ot-holodnoy-voyny-do-glubokogo-obucheniya/> (дата обращения: 26.05.19);
4. WebStorm [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/WebStorm#cite_ref-_8899f4c9bd08a9d0_1-0 (дата обращения: 26.05.2019);
5. Возможности применения технологий дополненной реальности в обучении информационным технологиям [Электронный ресурс] – Режим доступа: <https://elibrary.ru/item.asp?id=29120664/> (дата обращения 26.05.19);

Листинг программы

Алгоритм подбора заданий для упражнения:

```

*generateTaskSet({ exerciseType, langFrom, langTo }: TGeneratorProps):
Generator<TTask, void, any> {
  const translationList = this._translationModel.findMany({ lang: langFrom
}).filter((translation) => {
  const ratingEntity = this._ratingModel.findOne({
    exerciseType,
    translationId: translation.id
  });
  return ratingEntity && ratingEntity.rating < 100;
});

  const taskSetIds = [];
  const taskSetSize = Math.min(TaskGeneratorService.defaultTaskSetSize,
translationList.length);

  while (taskSetIds.length !== taskSetSize) {
    const randomIndex = Math.floor(Math.random() * translationList.length);
    const translation = translationList[randomIndex];

    const alreadyInSet = taskSetIds.some((translationId) => translationId ===
translation.id);
    if (!alreadyInSet) {
      taskSetIds.push(translation.id);
      const ratingEntity = this._ratingModel.findOne({
        exerciseType,
        translationId: translation.id
      });

      yield {
        translation,
        answer: this._translationHelper.findAnswerList({ wordId:
translation.id, lang: langTo })[0],
        ratingEntity
      };
    }
  }
}

```

Отчет по покрытию проекта типами

Flow Coverage Report

Summary

Percent	Total	Covered	Uncovered
75 %	3253	2448	805

Files

Filename ^	Annotation	Percent	Total	Covered	Uncovered
app/App.js	@flow	100 %	2	2	0
app/components/Home/components/ExerciseBlockList/index.js	@flow	83 %	65	54	11
app/components/Home/screens/HomeScreen/index.js	@flow	86 %	46	40	6
app/components/Home/shared/CategoryCard/index.js	@no flow	100 %	0	0	0
app/components/Home/shared/CategoryHeader/index.js	@flow	92 %	14	13	1
app/components/Home/shared/ExerciseBlock/index.js	@flow	88 %	44	39	5
app/components/Learning/components/BlockExercise/index.js	@flow	55 %	87	48	39
app/components/Learning/components/Exercise/index.js	@flow	77 %	126	98	28
app/components/Learning/components/TextExercise/index.js	@flow	33 %	6	2	4
app/components/Learning/components/logic/use-exercise-state.js	@flow	61 %	26	16	10
app/components/Learning/components/logic/use-task-set.js	@flow	100 %	29	29	0
app/components/Learning/components/shared/TextInputExercise/index.js	@flow	79 %	39	31	8
app/components/Learning/screens/LearningScreen/index.js	@flow	79 %	43	34	9
app/components/Learning/shared/StringBlock/index.js	@flow	54 %	44	24	20
app/components/Learning/shared/StringSlicer/index.js	@flow	100 %	4	4	0
app/components/Learning/shared/WordCard/index.js	@flow	59 %	112	67	45

Рисунок 31 – Отчет по покрытию типами текста (часть 1)

app/components/Learning/shared/WordCard/index.js	@flow	59 %	112	67	45
app/components/Recognition/screens/TextRecognizerScreen/index.js	@no flow	100 %	2	2	0
app/components/Recognition/shared/TextRecognizer/index.js	@flow	64 %	391	251	140
app/components/Start/screens/AuthCheckerScreen/index.js	@flow	100 %	22	22	0
app/components/Start/screens/HelloScreen/index.js	@flow	75 %	58	44	14
app/components/Start/screens/SignInScreen/index.js	@flow	97 %	40	39	1
app/components/Start/screens/SignUpScreen/index.js	@flow	97 %	39	38	1
app/components/Start/shared/LoginLayout/index.js	@flow	74 %	35	26	9
app/components/shared/Button/index.js	@flow	64 %	112	72	40
app/components/shared/HomeLayout/index.js	@no flow	60 %	33	20	13
app/components/shared/Label/index.js	@flow	100 %	12	12	0
app/components/shared/Text/index.js	@flow	61 %	85	52	33
app/components/shared/TextInput/index.js	@flow	66 %	48	32	16
app/constants/ExerciseState.js	@no flow	75 %	8	6	2
app/constants/ExerciseType.js	@flow	100 %	7	7	0
app/constants/Languages.js	@flow	100 %	7	7	0
app/di/di.js	@flow	100 %	32	32	0
app/di/models.js	@no flow	89 %	117	105	12
app/di/shared-services.js	@flow	100 %	19	19	0
app/helpers/_tests_/shuffleArray.js	@no flow	68 %	47	32	15
app/helpers/compose.js	@flow	26 %	15	4	11
app/helpers/platform.js	@no flow	55 %	72	40	32
app/helpers/shuffleArray.js	@no flow	42 %	35	15	20
app/logic/decorators/injectModel.js	@flow	41 %	17	7	10
app/logic/decorators/injectSharedService.js	@flow	41 %	17	7	10
app/logic/hocs/injectService.js	@flow	100 %	18	18	0
app/logic/hooks/use-navigation.js	@no flow	50 %	4	2	2
app/logic/hooks/use-service.js	@flow	85 %	7	6	1
app/models/Model.js	@flow	80 %	139	112	27

Рисунок 32 – Отчет по покрытию типами текста (часть 2)

app/models/TranslationRelationModel.js	@flow	100 %	3	3	0
app/routes/AppRouter.js	@no flow	71 %	14	10	4
app/routes/AuthCheckerRouter.js	@no flow	100 %	0	0	0
app/routes/RootRouter.js	@flow	76 %	38	29	9
app/routes/StartRouter.js	@no flow	78 %	14	11	3
app/services/AuthService.js	@flow	82 %	68	56	12
app/services/ExerciseCheckingService.js	@flow	100 %	76	76	0
app/services/TaskGeneratorService.js	@flow	85 %	149	128	21
app/services/TextFilterService.js	@flow	50 %	66	33	33
app/services/TextRecognitionService.js	@flow	62 %	16	10	6
app/services/TranslationHelperService.js	@flow	85 %	60	51	9
app/services/TranslationService.js	@flow	73 %	56	41	15
app/services/__tests__/TextFilterService.js	@no flow	89 %	86	77	9
app/services/rest/AuthApi.js	@flow	73 %	15	11	4
app/services/rest/BaseApi.js	@flow	80 %	36	29	7
app/services/rest/TranslateApi.js	@flow	75 %	33	25	8
app/services/rest/TransportApi.js	@flow	66 %	18	12	6
app/services/rest/__tests__/TranslateApi.test.js	@no flow	81 %	92	75	17
app/services/types.js	@no flow	100 %	2	2	0
app/style/Color.js	@no flow	97 %	41	40	1
app/style/NavigationStyle.js	@no flow	60 %	5	3	2
app/style/Spacing.js	@flow	60 %	51	31	20
app/style/Typography.js	@no flow	85 %	163	139	24
app/types/utils.js	@no flow	100 %	2	2	0

Рисунок 33 – Отчет по покрытию типами текста (часть 3)