

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

ДОПУСТИТЬ К ЗАЩИТЕ
Заведующий кафедрой ЭВМ
_____ Д. В. Топольский
«___» _____ 2022 г.

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ ТРЕНИРОВКИ
МЕЛКОЙ МОТОРИКИ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ
ЮУРГУ-090301.2022.209 ПЗ ВКР

Руководитель работы,
к.т.н., доцент каф. ЭВМ
_____ Е. С. Ярош
«___» _____ 2022 г.

Автор работы,
студент группы КЭ-405
_____ Р. Г. Гильманова
«___» _____ 2022 г.

Нормоконтролёр,
доцент. каф. ЭВМ
_____ М. А. Алтухова
«___» _____ 2022 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

УТВЕРЖДАЮ

Заведующий кафедрой ЭВМ

_____ Д. В. Топольский

«___»_____ 2022 г.

ЗАДАНИЕ

на выпускную квалификационную работу бакалавра

студенту группы КЭ-405

Гильмановой Розалии Галиевне,

обучающемуся по направлению

09.03.01 «Информатика и вычислительная техника»

1) **Тема работы:** «Разработка мобильного приложения для тренировки мелкой моторики» утверждена приказом по университету от 12 декабря 2021 г. № 308/141

2) Срок сдачи студентом законченной работы: 1 июня 2022 г.

3) **Исходные данные к работе:**

- приложение должно устанавливаться на ОС Android 4.0;
- приложение должно разрабатываться на языке C#;
- при разработке должна использоваться платформа Unity 2020.3.14f1;
- приложение должно содержать разделы: “пальчиковая гимнастика” и “графические упражнения”;
- пользователь должен иметь возможность открыть подсказки по использованию приложения;
- пользователь должен иметь возможность открыть любую информацию в приложении нажав не более чем на четыре кнопки;

– интерфейс приложения должен быть адаптивен для мобильных устройств с разными форматами экранов;

– интерфейс приложения должен быть реализован на русском языке;

4) Перечень подлежащих разработке вопросов:

– рассмотрение существующих приложений и проектов, ориентированных на помощь в развитии мелкой моторики;

– анализ современных программных технологий на возможность их применения в приложении по развитию мелкой моторики;

– разработка собственного приложения для развития мелкой моторики;

– проведение тестов разработанного приложения.

5) Дата выдачи задания: 1 декабря 2021 г.

Руководитель работы _____ /*Е. С. Ярош*/

Студент _____ /*Р. Г. Гильманова*/

КАЛЕНДАРНЫЙ ПЛАН

Этап	Срок сдачи	Подпись руководителя
Введение и обзор литературы	10.03.2022	
Рассмотрение существующих приложений и проектов, ориентированных на помощь в развитии мелкой моторики	15.03.2022	
Анализ современных программных технологий на возможность их применения в приложении по развитию мелкой моторики	21.03.2022	
Разработка собственного приложения для развития мелкой моторики	04.04.2022	
Тестирование и отладка разработанного приложения	25.05.2022	
Компоновка текста работы и сдача на нормоконтроль	16.05.2022	
Подготовка презентации и доклада	24.05.2022	

Руководитель работы _____ /*Е. С. Ярош*/

Студент _____ /*Р. Г. Гильманова*/

АННОТАЦИЯ

Р. Г. Гильманова. Разработка мобильного приложения для тренировки мелкой моторики. – Челябинск: ФГАО ВО “ЮУрГУ (НИУ)”, ВШ ЭКН; 2022, 49 с., 35 ил., библиогр. список – 13 наим.

В рамках выпускной квалификационной работы описывается разработка мобильного приложения для развития мелкой моторики. Производится анализ аналогов приложения, в ходе которого выявляются их достоинства и недостатки. Полученные при анализе сведения учитываются в создаваемом приложении.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	6
1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ	9
2 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ	22
2.1 Нефункциональные требования	22
2.2 Функциональные требования	22
3 РАЗРАБОТКА МОДЕЛИ И ПРОЕКТИРОВАНИЕ.....	24
4 РЕАЛИЗАЦИЯ	28
4.1 Структура папок.....	28
4.2 Скрипты.....	29
4.2.1 Скрипт Timer	29
4.2.2 Скрипты FileName и FileManager	31
4.3 Интерфейс	32
5 ТЕСТИРОВАНИЕ	37
ЗАКЛЮЧЕНИЕ	42
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	43
ПРИЛОЖЕНИЕ А Листинги.....	45

ВВЕДЕНИЕ

Мелкая моторика чрезвычайно важна в повседневной жизни, так как она необходима для совершения множества обычных дел. Например, завязывания шнурков, застегивания молний, пуговиц, письма и многого другого. Также имеется связь моторики с другими функциями организма, в частности, развитием речи [1].

Можно выделить две основные группы людей, у которых встречаются проблемы с мелкой моторикой – это дети дошкольного или раннего школьного возраста и взрослые перенесшие травмы рук, получившие инсульты и так далее. Для коррекции моторики и у детей, и у взрослых используются схожие методы. В основном это физические упражнения, основанные на постановке пальцев в определенные положения, тактильная стимуляция, массаж, графические упражнения [2].

Проблема состоит в том, что в основном занятия проводятся в присутствии педагога. Это подразумевает трату времени на то, чтобы добраться до места проведения занятия, а также оплату сеанса. Таким образом часть людей, которым нужны занятия, могут быть не в состоянии их посещать. Например, возможны проблемы с организацией личного присутствия занимающегося. Маломобильный человек или ребенок может быть не в состоянии самостоятельно добраться на занятия. Также возможны финансовые трудности или банальное отсутствие специалистов в малых или отдаленных населенных пунктах. Поэтому задачу создания альтернативы очным занятиям по развитию мелкой моторики можно считать актуальной.

Цель данной работы – создать мобильное приложение, которое бы предоставляло возможность пользователю самостоятельно выполнять упражнения для развития мелкой моторики. В приложении предполагается использовать методики физических и графических упражнений. Тип мобильного приложения выбран из-за того, что смартфоны достаточно распространены и имеются у большинства людей. Также в них присутствует

мобильная камера, которую планируется использовать в функционале приложения. Дети быстро осваивают принципы использования этих устройств. Платформа Android выбрана, так как она является самой распространённой операционной системой для мобильных устройств [3].

В силу того, что для развития мелкой моторики детей примерно до 5 лет в основном используются иные способы развития, нежели выбранные для реализации, целевой аудиторией будут дети дошкольного и младшего школьного возраста, а также взрослые, поскольку упражнения для реабилитации имеют аналогичную форму.

Для достижения поставленной цели, необходимо решить следующие задачи:

- рассмотреть существующие приложения и проекты, ориентированные на помощь в развитии мелкой моторики;
- провести анализ современных программных технологий на возможность их применения в приложении по развитию мелкой моторики;
- разработать собственное приложение для развития мелкой моторики на основе проведенного анализа;
- протестировать разработанное приложение.

Работа состоит из введения, пяти разделов, заключения и списка использованных источников. Объем работы составляет 49 страниц, количество использованных источников равно 13.

В введении приведена общая формулировка темы, обоснована актуальность выбранной темы, сформулированы цель и задачи исследования, приведены структура и объем работы.

В первой главе “Анализ предметной области” проводится анализ существующих приложений, направленных на развитие мелкой моторики.

Во второй главе “Определение требований” определяется необходимый функционал приложения.

В третьей главе “Проектирование” описывается архитектура предлагаемого решения и алгоритмы решения поставленных в введении задач.

В четвертой главе “Реализация” описывается реализация созданного мобильного приложения.

В пятой главе “Тестирование” производится оценка работоспособности разработанного приложения.

В заключении приводятся краткие выводы, описывается новизна и практическая значимость работы.

1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

В данном разделе будет произведен обзор аналогов разрабатываемого приложения. Планируется рассмотреть следующие приложения, которые позиционируются как приложения, помогающие улучшить мелкую моторику:

- 1) Рисование для Детей – Рисуем животных. Рисовалка [4];
- 2) РИСОВАЛКА! Раскраски Детские Игры для Детей 2 лет [5];
- 3) Fine Motor Skills Practice [6];
- 4) Соедини точки – Динозавры [7];
- 5) Антипарезис [8];
- 6) Dexteria Fine/Rehab Aid [9];
- 7) ReHand, Rehabilitación de Mano [10];
- 8) Indigo [11].

Также рассмотрены два приложения, не нацеленные на развитие мелкой моторики, но реализующие функции, которые можно применить в разрабатываемом приложении. Это приложения:

- 9) SketchAR [12];
- 10) Quiver [13].

Целевой аудиторией приложения **Рисование для Детей – Рисуем животных** являются дети от трех лет [4]. В приложении пользователю предлагается выбрать рисунок и поэтапно раскрасить его. Скриншот окна выбора рисунка представлен на рисунке 1.



Рисунок 1 – Окно выбора рисунка

Скриншот поэтапного раскрашивания представлен на рисунке 2.



Рисунок 2 – Скриншот интерфейса раскрашивания

Для раскрашивания необходимо водить пальцем по сенсорному экрану в пределах фигуры. После того как рисунок раскрашен, пользователь может с ним взаимодействовать, нажав на одну из предложенных кнопок или на любое место на экране. При нажатии кнопок рисунок выполняет различные движения. Скриншот интерфейса для взаимодействия с рисунком представлен на рисунке 3.



Рисунок 3 – Скриншот интерфейса для взаимодействия с рисунком

Достоинством приложения можно считать использование крупных кнопок и выбор с помощью пролистывания. Такое решение делает удобным использование приложения людьми с плохо развитой моторикой. Также действия озвучиваются, что полезно для детей, не умеющих читать. Есть возможность отключить звуковое сопровождение. Также приложение может работать без доступа к интернету.

К недостаткам можно отнести отсутствие комментариев на русском языке, и необходимость платить за выбор некоторых рисунков.

Функционал приложения **РИСОВАЛКА! Раскраски Детские Игры для Детей 2 лет** аналогичен предыдущему приложению. Из положительных отличий стоит отметить большее количество бесплатных шаблонов рисунков. При этом рисунки сгруппированы по темам. Скриншот меню выбора рисунков приведен на рисунке 4.



Рисунок 4 – Меню выбора темы рисунков

Выбор так же, как в предыдущем приложении, осуществляется пролистыванием. Скриншот выбора рисунка приведен на рисунке 5. Также следует отметить, что имеется озвучка на русском языке.

Недостатками приложения является наличие большого количества рекламы и маленькие кнопки. Пример рекламы и кнопок также представлен на рисунке 5.



Рисунок 5 – Меню выбора рисунка

В приложении **Fine Motor Skills Practice** надо обводить различные узоры, линии, фигуры и буквы. Примеры обводимых фигур представлены на рисунке 6 [6].



Рисунок 6 – Примеры обводимых фигур [6]

Приложение рассчитано на детей от трех лет. Скриншот интерфейса приложения представлен на рисунке 7.

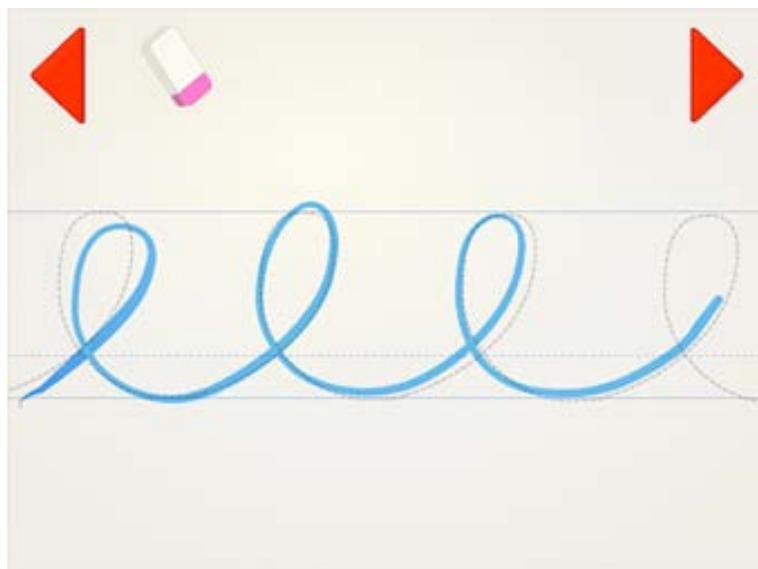


Рисунок 7 – Скриншот интерфейса приложения [6]

Преимущество данного приложения заключается в том, что в его создании участвовали учителя. Также в нем нет рекламы [6].

Недостатки – приложение полностью платное, нет перевода на русский язык.

В приложении **Соедини точки – Динозавры** пользователю предлагается соединить точки и раскрасить получившийся рисунок. Скриншот соединения точек представлен на рисунке 8.



Рисунок 8 – Скриншот соединения точек

Скриншот раскрашивания представлен на рисунке 9. Приложение рассчитано на детей до 12 лет [7].



Рисунок 9 – Скриншот раскрашивания

Преимуществом приложения является большое количество настраиваемых параметров, что делает возможность настроить сложность игры. Скриншот настроек приведен на рисунке 10.



Рисунок 10 – Скриншот настроек

Недостаток – развитие моторики не является основным направлением данного приложения.

Приложение **Антипарезис** представлено в качестве бизнес идеи на конкурсе “Я бизнесмен 2016”. Приложение позиционируется как инструмент для реабилитации после инсультов [8]. В нем предлагается стилусом обводить различные буквы, цифры и фигуры. Скриншот презентационного видео приведен на рисунке 11.



Рисунок 11 – Скриншот презентационного видео [8]

Весомым преимуществом данного приложения является то, что оно разработано специалистом по лечебной физкультуре [8].

К недостаткам можно отнести необходимость наличия планшета и невозможность приобретения приложения обычным пользователем.

Dexterity Fine/Rehab Aid – приложение, нацеленное на развитие мелкой моторики. В приложении есть три игры. Пример главного экрана приведен на рисунке 12.



Рисунок 12 – Пример главного экрана приложения [9]

В первой игре надо нажимать на фигуры определенным пальцем. Скриншот приведен на рисунке 13.

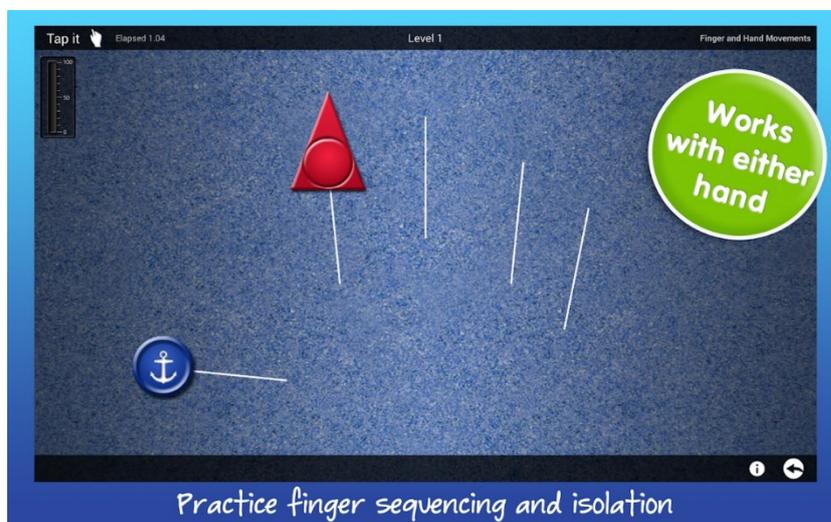


Рисунок 13 – Скриншот интерфейса игры по нажиманию на фигуры [9]

Во второй игре предлагается двумя пальцами “зажимать” движущиеся объекты, в третьей обводить буквы или цифры. Пример написания букв приведен на рисунке 14.

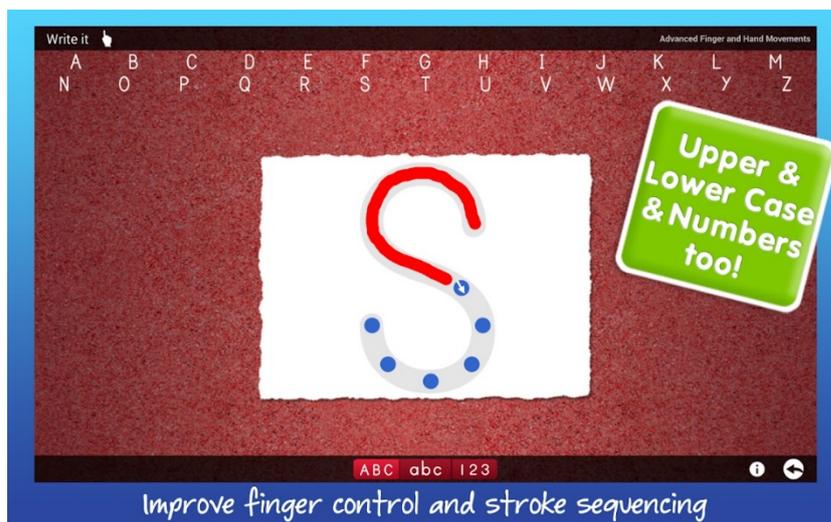


Рисунок 14 – Скриншот игры по написанию букв [9]

Преимущество приложения в том, что при его разработке участвовали профильные специалисты. Также оно предназначено как для детей, так и для взрослых, имеет функцию отслеживания прогресса. Данные о выполнении упражнений можно автоматически пересылать врачу [9].

Главным недостатком является недоступность приложения на территории Российской Федерации, а также то, что приложение полностью платное [9].

Приложение **ReHand, Rehabilitación de Mano** предназначено для помощи в реабилитации запястья, кисти и пальцев. Для реабилитации используются гимнастические упражнения. Конкретное упражнение можно выбрать в меню. Внешний вид меню приведен на рисунке 15.



Рисунок 15 – Внешний вид меню приложения ReHand [10]

Для выполнения упражнений предлагается положить ладонь на экран планшета и перемещать пальцы в соответствии с выводимыми подсказками. Пример выполнения упражнения приведен на рисунке 16.



Рисунок 16 – Пример выполнения упражнения [10]

Преимуществом данного приложения является то, что оно разработано с участием хирургов, физиотерапевтов, терапевтов и эрготерапевтов. Также следует отметить возможность отслеживания прогресса и отправки данных о прогрессе врачу [10].

К недостаткам можно отнести отсутствие перевода на русский, платный доступ к приложению, необходимость иметь планшет для использования приложения [10].

Indigo – приложение для раннего развития детей до пяти лет. Целевой аудиторией являются родители детей. Перед использованием приложения пользователь отвечает на тестовые вопросы. После чего приложение подбирает упражнения на основе теста, пола и возраста ребенка. Скриншот главного меню приложения приведен на рисунке 17.

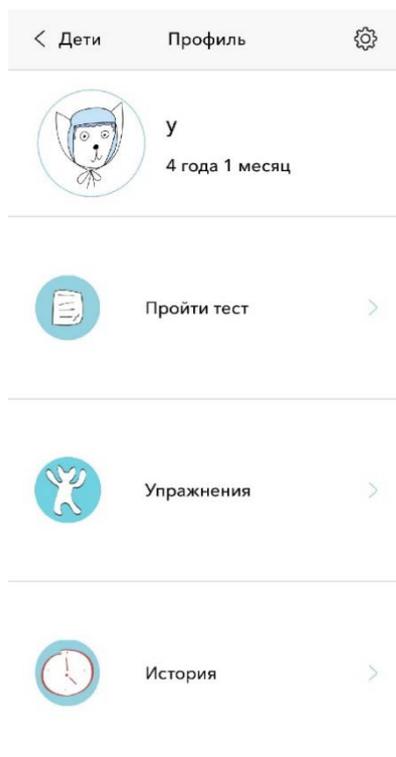


Рисунок 17 – Скриншот главного меню

Кроме мелкой моторики присутствуют упражнения и другой направленности. Скриншот меню выбора упражнений приведен на рисунке 18.



Рисунок 18 – Скриншот меню выбора упражнений

Сами упражнения представлены в виде описания действий, которые необходимо выполнить. Пример описания упражнения приведен на рисунке 19.



Рисунок 19 – Пример описания упражнения

Преимуществом данного приложения является учет особенностей занимающегося. К недостаткам можно отнести то, что упражнения ограничиваются простым описанием и то, что упражнения для развития мелкой моторики доступны только в платном режиме.

Следующие два приложения не нацелены на развитие мелкой моторики, но реализуют функции, которые можно применить в разрабатываемом приложении.

Приложение **Quiver** рассчитано на использование детьми и напоминает по функционалу первые два разобранных приложения. Отличие состоит в использовании AR технологий. Пользователь раскрашивает рисунок на бумаге и наводит на него камеру. В результате на экране телефона появляется 3D модель. Пример работы приложения приведен на рисунке 20.



Рисунок 20 – Пример работы приложения

К преимуществам можно отнести, то что пользователь взаимодействует с реальными физическими предметами и игровой формат.

Из недостатков – необходимость распечатывать раскраски.

SketchAR – это приложение для рисования. В режиме “Рисование с AR” контур рисунка выводится поверх изображения камеры, и пользователю предлагается его обвести. Скриншот данного режима приведен на рисунке 21.



Рисунок 21 – Скриншот интерфейса приложения в режиме “Рисование с AR”

Использование аналогичной функции может позволить выполнять графические упражнения без распечатки прописей.

Вывод. На основе проведенного анализа решено разработать приложение, в котором будут указания для выполнения гимнастических упражнений как в приложении “Indigo”, но тип упражнений будет использоваться как в приложении “ReHand”. Функционал с прикладыванием ладони не используется, чтобы приложением можно было пользоваться и на смартфоне. Графические упражнения решено реализовать в виде заданий по обведению контуров наподобие представленных в первых семи рассмотренных приложениях. Так как тактильные взаимодействия полезны для развития моторики, пользователь будет рисовать на бумаге. Чтобы не распечатывать контуры для обводки, будет использоваться способ наложения рисунка на видеоряд камеры как в приложении “SketchAR”. Для добавления развлекательного момента будет реализован дополнительный режим с рисованием контура по клеткам. В нем при сканировании рисунка камерой будет появляться изображение как в “Quiver”.

2 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ

2.1 Нефункциональные требования

Требования к эргономике и технической эстетике. Для организации взаимодействия Приложения с пользователем должен применяться графический интерфейс пользователя (GUI).

Интерфейс Приложения для конечного пользователя должен удовлетворять следующим требованиям.

- интерфейс должен быть адаптивен для разных устройств с разными форматами экранов;
- интерфейс должен быть реализован на русском языке;
- должен использоваться шрифт не менее чем 12 пт;
- на экране не должно располагаться больше чем 6 пунктов меню;
- между блоками текста должны быть отступы шириной не менее ширины одной строки;
- пользователь должен иметь возможность открыть любую информацию в приложении нажав не более чем на четыре кнопки.

Требования к патентной чистоте. В отношении Российской Федерации должна быть обеспечена патентная чистота разрабатываемого приложения и его частей. По всем программным средствам, применяемым в приложении, должны соблюдаться условия лицензионных соглашений и обеспечиваться патентная чистота.

2.2 Функциональные требования

Функциональные требования к разрабатываемому приложению оформлены в виде диаграммы вариантов языка UML. Диаграмма приведена на рисунке 22.

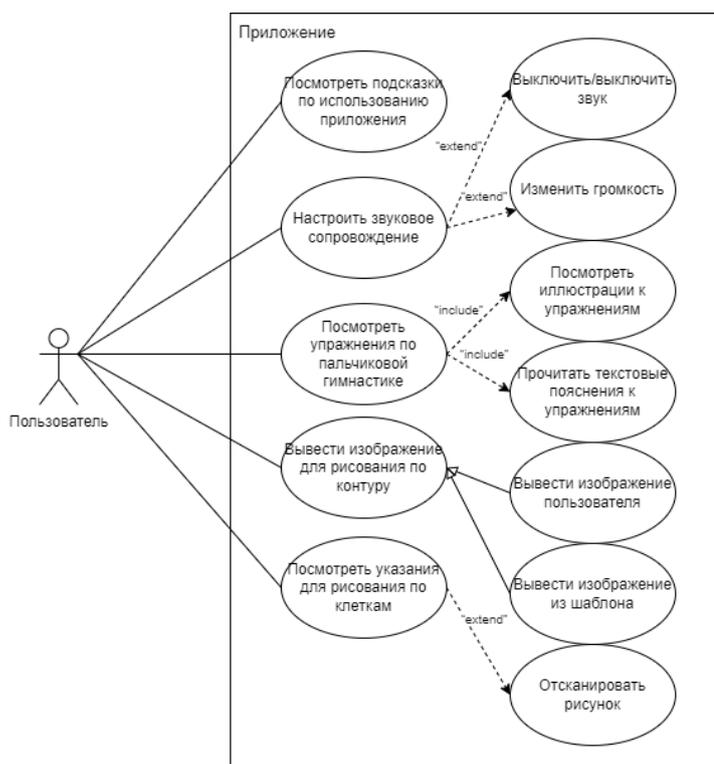


Рисунок 22 – Диаграмма вариантов использования приложения

Рассмотрим прецеденты использования разрабатываемого приложения.

Пользователь может открыть справку с подсказками по использованию приложения.

Пользователь может настроить звуковое сопровождение. В частности, включить, отключить его или изменить громкость.

Пользователь может посмотреть указания к упражнениям по пальчиковой гимнастике, которые состоят из иллюстраций и поясняющего текста.

Пользователь может вывести поверх видеоряда камеры контурное изображение для обводки. Изображение при этом может быть выбрано из имеющихся в приложении шаблонов или загружено в приложение с устройства пользователя.

Пользователь может посмотреть указания по рисованию изображения по клеткам, выбранного из списка. Далее, пользователь имеет возможность отсканировать созданный им рисунок.

3 РАЗРАБОТКА МОДЕЛИ И ПРОЕКТИРОВАНИЕ

В Приложении предполагается выделить следующие функциональные подсистемы:

1) подсистема упражнений пальчиковой гимнастики, которая предоставляет иллюстрированные методические указания к упражнениям по развитию мелкой моторики, использующим гимнастический метод;

2) подсистема графических упражнений с обведением контуров, которая загружает, обрабатывает и выводит на экран изображения, преобразованные в материал для упражнений по развитию мелкой моторики, использующих графический метод;

3) подсистема графических упражнений с рисованием фигур, которая выводит указания для рисования фигур, производит сканирование рисунка пользователя и последующее отображение иллюстрации.

Подсистема упражнений пальчиковой гимнастики. Перечень функций и задач к подсистеме упражнений пальчиковой гимнастики отображен в таблице 1.

Таблица 1 – Перечень функций и задач к подсистеме упражнений пальчиковой гимнастики

Функция	Задача
Предоставление указаний к упражнению	Вывод иллюстрации к упражнению
	Вывод описания действий
Переход между упражнениями	Отслеживание принудительного перехода
	Отслеживание времени выполнения упражнения
	Переход к следующему или предыдущему упражнению

Временной регламент реализации каждой задачи отображен в таблице 2.

Таблица 2 – Временной регламент задач упражнений пальчиковой гимнастики

Задача	Требования к временному регламенту
Вывод иллюстрации к упражнению	Во время работы подсистемы гимнастики для пальцев
Вывод описания действий	Во время работы подсистемы гимнастики для пальцев
Отслеживание принудительного перехода	Во время работы подсистемы гимнастики для пальцев
Отслеживание времени выполнения упражнения	Во время работы подсистемы гимнастики для пальцев
Переход к следующему или предыдущему упражнению	После истечении времени выполнения упражнения или выбора пользователем опции принудительного перехода

Подсистема графических упражнений с обведением контуров.

Перечень функций и задач к подсистеме графических упражнений с обведением контуров отображен в таблице 3.

Таблица 3 – Перечень функций и задач к подсистеме графических упражнений с обведением контуров

Функция	Задача
Выбор шаблона для упражнения	Загрузка изображения с мобильного устройства в Приложение
	Выбор имеющегося шаблона
Вывод шаблона	Обработка загруженного изображения
	Наложение изображения на видеоряд с камеры

Временной регламент задач подсистемы графических упражнений с обведением контуров отображен в таблице 4.

Таблица 4 – Временной регламент задач к подсистеме графических упражнений с обведением контуров

Задача	Требования к временному регламенту
Загрузка изображения с мобильного устройства в Приложение	Поле выбора пользователем соответствующей опции
Выбор имеющегося шаблона	Поле выбора пользователем соответствующей опции
Обработка загруженного изображения	После загрузки изображения
Наложение изображения на видеоряд с камеры	После обработки изображения иди выбора шаблона

Подсистема графических упражнений с рисованием фигур.

Перечень функций и задач к подсистеме графических упражнений с рисованием фигур отображен в таблице 5.

Таблица 5 – Перечень функций и задач к подсистеме графических упражнений с рисованием фигур

Функция	Задача
Выбор шаблона для упражнения	Выбор имеющегося шаблона
Вывод указаний для рисования	Отображение текстового описания
	Отображение примера готового рисунка
Сканирование рисунка пользователя	Идентификация рисунка
	Отображение иллюстрации

Временной регламент задач подсистемы графических упражнений с рисованием фигур отображен в таблице 6.

Таблица 6 – Временной регламент задач подсистемы графических упражнений с рисованием фигур

Задача	Требования к временному регламенту
Выбор имеющегося шаблона	Поле выбора пользователем соответствующей опции
Отображение последовательности из длин и направлений отрезков рисуемой фигуры	Во время работы подсистемы графических упражнений с рисованием фигур
Отображение примера готового рисунка	Поле выбора пользователем соответствующей опции
Идентификация рисунка	Поле выбора пользователем соответствующей опции
Отображение иллюстрации	После идентификации рисунка

Сведения по обеспечению требований к функциям, выполняемым каждой подсистемой и определяющим её качество приведены в таблице 7.

Таблица 7 – Сведения по обеспечению требований к функциям

Подсистема	Функция	Метод реализации
упражнений гимнастики для пальцев	Предоставление указаний к упражнению	Путем осуществления вывода на экран мобильного устройства текста с описанием действий в упражнении и рисунков, иллюстрирующих действия.
	Переход между упражнениями	Путем введения автоматического переключения по таймеру или при нажатии кнопки перехода.

Продолжение таблицы 7

Подсистема	Функция	Метод реализации
графических упражнений с обведением контуров	Выбор шаблона для упражнения	Путем загрузки готового шаблона из файлов приложения или загрузки изображения с устройства пользователя.
	Вывод шаблона	Путем наложения контуров, выделенных с изображения на видеоряд с камеры.
подсистеме графических упражнений с рисованием фигур	Выбор шаблона для упражнения	Путем загрузки готового шаблона из файлов приложения.
	Вывод указаний для рисования	Путем отображения примера готового рисунка и последовательности из длин и направлений отрезков, составляющих рисуемую фигуру.
	Сканирование рисунка пользователя	Путем использования функций платформы vuforia.

4 РЕАЛИЗАЦИЯ

4.1 Структура папок

В приложении имеются две папки: Assets и Packages. В Packages хранятся пакеты, установленные с помощью Unity Package Manager. Папка Assets содержит 10 вложенных папок. Список папок представлен на рисунке 23.

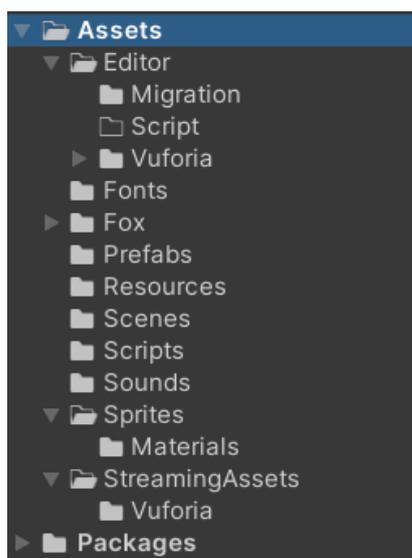


Рисунок 23 – Список папок приложения

В папке Editor расположены файлы, необходимые для работы с платформой Vuforia. Папка Fonts содержит шрифты, используемые в приложении. В папке Fox хранятся все ресурсы, связанные с ассетом лисы из сцены CheckDraw. Ассет был загружен из Unity Asset Store. В Prefabs находятся префабы. Далее, в Resources лежит VuforiaConfiguration.asset. Он предназначен для ввода лицензионного ключа Vuforia и настройки части параметров для ее использования. В Scenes находятся сцены приложения, в Scripts – скрипты для их реализации. Sounds содержит аудиофайлы, Sprites – спрайты и вложенную папку с материалами. В StreamingAssets лежат файлы Vuforia, которые должны храниться в исходных форматах без преобразования во внутренние форматы Unity.

4.2 Скрипты

Для реализации сцен были разработаны скрипты на языке C#. Всего написано 10 скриптов. Информация о них сведена в таблицу 8.

Таблица 8 – Наименования и назначения скриптов

	Наименование скрипта	Назначение
1	Audio	Настройка музыки в приложении.
2	FileName	Создание объектов при просмотре файлов для загрузки.
3	FilesManager	Загрузка изображения с пользовательского устройства.
4	ImageOnOff	Активация и деактивация объектов.
5	LoadScenes	Загрузка сцен.
6	LoopMusic	Проигрывание фоновой музыки.
7	MenuButtons	Отображение материалов, соответствующих выбранной кнопке.
8	SliderContr	Передача значений слайдера в настройки громкости.
9	SnapScrolling	Пролистывание с помощью скроллинга.
10	Timer	Отсчет времени и анимация gif-изображений в разделе “гимнастика для рук”

Далее рассмотрены основные скрипты, отвечающие за загрузку и обработку пользовательского изображения, и скрипт для реализации раздела гимнастических упражнений.

4.2.1 Скрипт Timer

Скрипт Timer предназначен для реализации раздела гимнастических упражнений. В скрипте Timer происходит отсчет времени от заданного значения до нуля. При этом во время отсчета пропорционально изменяется размер индикатора, а над ним отображаются проценты. Анимация иллюстраций происходит за счет смены спрайтов. При окончании отсчета происходит увеличение переменной, хранящей номер упражнения и обновление текста с описанием упражнения. На рисунке 24 приведена блок-схема алгоритма скрипта Timer.

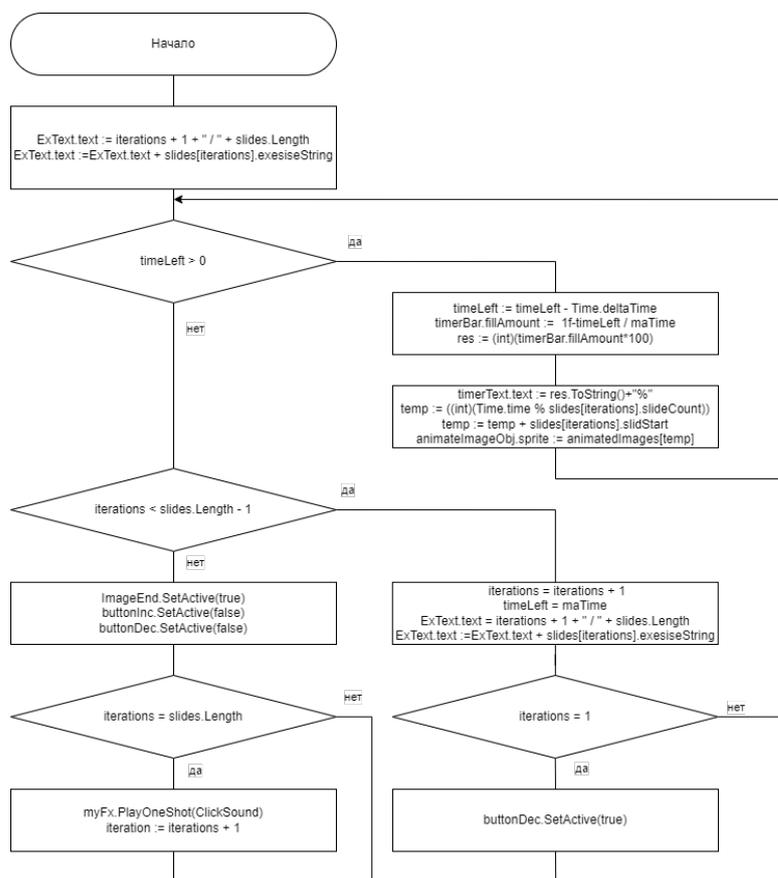


Рисунок 24 – Блок-схема алгоритма скрипта Timer

В скрипте содержится класс Timer, имеющий подкласс Exercise. В таблице 9 описаны назначения данных классов из них.

Таблица 9 – Наименования и назначения классов скрипта Timer

Наименование класса	Назначение
Timer	Отсчет времени, отведенного на упражнение, анимация индикатора времени и иллюстраций, переход между упражнениями.
Exercise	Хранение информации о ресурсах к упражнению.

Далее, на рисунке 25 приведена диаграмма классов.

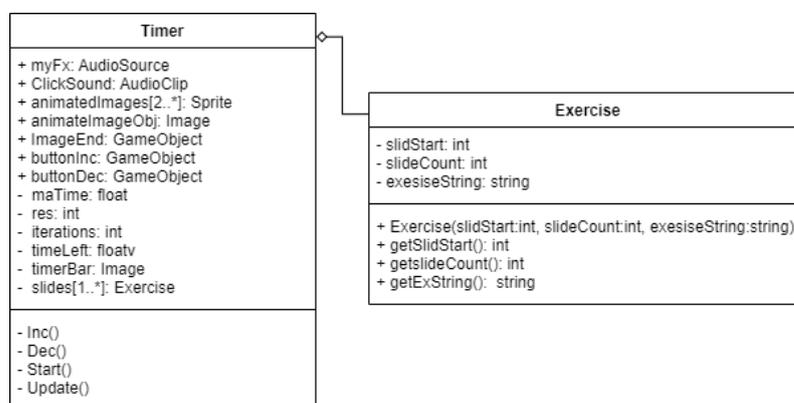


Рисунок 25 – Диаграмма классов для скрипта Timer

В таблице 10 приведено краткое описание назначения каждого метода.

Таблица 10 – Наименования и назначения методов классов Timer и Exercise

Наименование класса	Наименование метода	Назначение
Timer	Inc	Увеличение на единицу переменной, хранящей номер текущего упражнения.
	Dec	Уменьшение на единицу переменной, хранящей номер текущего упражнения.
	Start	Присвоение начальных значений переменным.
	Update	Ведение отсчета времени, анимации изображения и индикатора.
Exercise	Exercise	Конструктор класса Exercise
	getSlidStart	Доступ к полю slideStart.
	getSlidCount	Доступ к полю slideCount.
	getExString	Доступ к полю exerciseString.

Исходный код скрипта Timer приведен на листинге 1 в приложении А.

4.2.2 Скрипты FileName и FileManager

Данные скрипты предназначены для загрузки пользовательского изображения, выделения на нем контуров и вывода обработанного изображения поверх видеоряда с камеры смартфона. Для получения видеоряда используется ARCamera Vuforia. Изображение накладывается как текстура на объект LoadImg. В скриптах содержится несколько классов: FileName, FileManager и SelectConturs. В таблице 11 описано назначение каждого из перечисленных классов.

Таблица 11 – Наименования и назначения классов скриптов FileName и FileManager

Наименование класса	Назначение
FileName	Вызов метода SelectFile с индексом выбранного файла.
FileManager	Загрузка пользовательского изображения.
SelectConturs	Выделение контуров на загруженном изображении.

Далее, на рисунке 26 приведена диаграмма классов для скриптов FileName и FileManager.

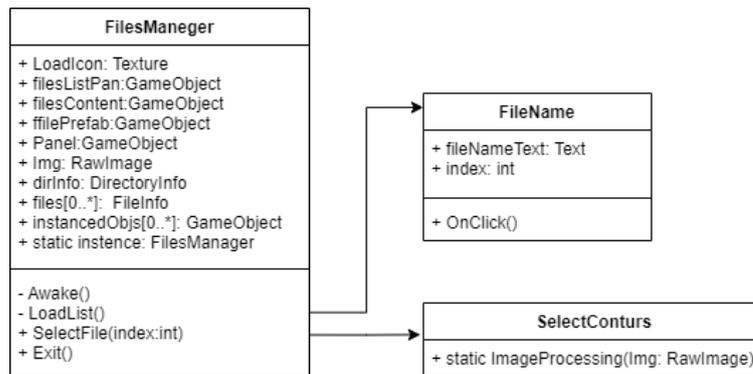


Рисунок 26 – Диаграмма классов для скриптов FileName и FilesManager

В таблице 12 приведено краткое описание назначения каждого метода.

Таблица 12 – Наименования и назначения методов классов FileName, FilesManager и SelectConturs

Наименование класса	Наименование метода	Назначение
FilesManager	Awake	Получение ссылки на класс FilesManager.
	LoadList	Отображение окна с файлами изображений, которые можно загрузить.
	SelectFile	Вывод выбранного изображения.
	Exit	Активация начальной панели и возвращение начальной иконки.
SelectConturs	ImageProcessing	Выделение контуров на изображении.
FileName	OnClick	Вызов метода SelectFile для выбранного файла.

Исходный код скрипта FileName приведен на листинге 2 в приложении А. Исходный код скрипта FilesManager приведен на листинге 3 в приложении А.

4.3 Интерфейс

При запуске приложения пользователь попадает на экран главного меню. Скриншот экрана главного меню представлен на рисунке 27.



Рисунок 27 – Скриншот главного меню приложения

На данном экране присутствуют 3 кнопки для перехода в режимы:

- гимнастика для рук;
- рисование по контурам;
- рисование по клеткам.

Кнопки сделаны крупными, чтобы пользователю с плохо развитой мелкой моторикой было легче их нажимать. Также на кнопках присутствуют иллюстрации. Они добавлены в качестве подсказок на случай, если приложение будет использовать ребенок не умеющий читать.

После выбора в главном меню режима, пользователь переходит в следующее меню, где выбирает изображение для рисования по клеткам или контуру в зависимости от того какой режим выбран. Скриншот меню для выбора картинки приведен на рисунке 28.



Рисунок 28 – Скриншот меню выбора картинки в режиме рисования по клеткам

В режиме “гимнастика для рук” пользователю предлагается повторить действие на gif-изображении. Под изображением идет отсчет времени, выделенного на упражнение. Отсчет выполнен в виде отображения числа процентов прошедшего времени и индикатора выполнения. Также присутствует текстовое описание упражнения. Пользователь может перемещаться по упражнениям с помощью соответствующих кнопок. В качестве примера приведен скриншот одного из упражнений на рисунке 29.



Рисунок 29 – Скриншот вкладки “гимнастика для рук”

На рисунке 30 приведен пример работы приложения в режиме “рисование по контурам”. В данном случае, пользователю предлагается навести камеру на лист бумаги и нарисовать линии в промежутке, указанном выводимым на экран контуром.

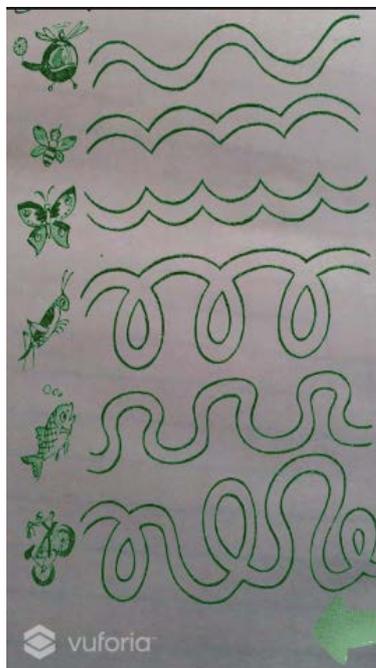
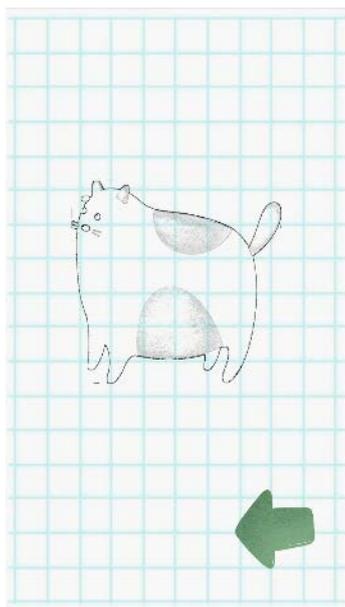


Рисунок 30 – Скриншот одного из вариантов “рисования по контурам”

В режиме “рисование по контурам” пользователь может выбрать картинку для вывода на экран из предложенных в меню или загрузить собственную из памяти устройства. Пример загруженной данным скриптом картинки приведен на рисунке 31 (а).

На рисунке 31 (б) изображен пример работы приложения в режиме “рисование по клеткам”. В нем пользователю предлагается нарисовать определенную фигуру по клеткам и отсканировать ее. Если в рисунке нет ошибок в построении и линии достаточно ровные появится изображение или 3D модель.

а)



б)



Рисунок 31 – Скриншот применения загрузки пользовательской картинке в “рисовании по контурам”

Для реализации данного режима использована платформа Vuforia. В частности, Image Tracking. В качестве Image Targets здесь выступает нарисованный пользователем рисунок. Какой рисунок надо нарисовать, указывается двумя вариантами. Первый вариант – “математический диктант”, т. е. последовательность из пар направлений и количества клеток. Второй вариант – пример готового рисунка.

5 ТЕСТИРОВАНИЕ

Разработанное приложение прошло альфа-тестирование. Тестирование проводилось по методологии модульного тестирования. Для тестирования были использованы AVR Manager в Android Studio и физическое устройство.

Характеристики физического устройства:

- модель – HUAWEI P20 lite;
- версия Android – 9;
- процессор – Hisilicon Kirin 659;
- RAM – 4 ГБ;
- разрешение экрана – 2280 x 1080.

На физическом устройстве проверена корректность работы основных функций приложения. На виртуальных устройствах проверены адаптивность интерфейса под различные разрешения экранов и работа приложения на различных версиях Android. Результаты тестов на физическом устройстве сведены в таблицу 13.

Таблица 13 – Результаты тестов на физическом устройстве

№ теста	действие	ожидаемый результат	полученный результат	итог
1	нажатие на кнопку “гимнастика для рук”	переход в режим “гимнастика для рук”	переход в режим “гимнастика для рук”	пройден
2	нажатие на кнопку “рисование по клеткам”	переход в режим “рисование по клеткам”	переход в режим “рисование по клеткам”	пройден
3	нажатие на кнопку “рисование по контуру”	переход в режим “рисование по контуру”	переход в режим “рисование по контуру”	пройден
4	нажатие на кнопку “вперед” в режиме “гимнастика для рук”	переход к следующему упражнению	переход к следующему упражнению	пройден
5	нажатие на кнопку “назад” в режиме “гимнастика для рук”	переход к предыдущему упражнению	переход к предыдущему упражнению	пройден

Продолжение таблицы 13

№ теста	действие	ожидаемый результат	полученный результат	итог
6	нажатие на кнопку “подсказка” в режиме “рисование по клеткам”	отображение изображения примера готового рисунка	отображение изображения примера готового рисунка	пройден
7	нажатие на кнопку “Готово” в режиме “рисование по клеткам”	переход в режим сканирования	переход в режим сканирования	пройден
8	наведение камеры на “правильный” рисунок в режиме сканирования	отображение 3d модели поверх рисунка	отображение 3d модели поверх рисунка	пройден
9	наведение камеры на “не правильный” рисунок в режиме сканирования	отсутствие отображения иллюстрации поверх рисунка	отсутствие отображения иллюстрации поверх рисунка	пройден
10	наведение камеры в режиме сканирования на рисунок, нарисованный по указаниям во вкладке “лиса”	отображение 3d модели в виде лисы поверх рисунка	отображение 3d модели в виде лисы поверх рисунка	пройден
11	наведение камеры в режиме сканирования на рисунок, нарисованный по указаниям во вкладке “медведь”	отображение иллюстрации в виде медведя поверх рисунка	отображение иллюстрации в виде медведя поверх рисунка	пройден
12	нажатие на кнопку “дорожки” в режиме “рисования по контурам”	отображение контуров изображения поверх видеоряда с камеры устройства	отображение контуров изображения поверх видеоряда с камеры устройства	пройден
13	нажатие на кнопку “своя картинка” в режиме “рисования по контурам”	появление кнопки загрузки пользовательского изображения	появление кнопки загрузки пользовательского изображения	пройден
14	Нажатие на иконку файла при выборе загружаемого пользовательского изображения	отображение контуров выбранного изображения поверх видеоряда с камеры устройства	отображение контуров выбранного изображения поверх видеоряда с камеры устройства	пройден

Результаты тестов на виртуальных устройствах сведены в таблицу 14.

Таблица 14 – Результаты тестов на виртуальных устройствах

№ теста	действие	ожидаемый результат	полученный результат	итог
15	запуск приложения на устройстве с разрешением экрана 480 x 800	отображение окна главного меню, без искажений пропорций кнопок	отображение окна главного меню, без искажений пропорций кнопок	пройден
16	запуск приложения на устройстве с разрешением экрана 1440 x 2560	отображение окна главного меню, без искажений пропорций кнопок	отображение окна главного меню, без искажений пропорций кнопок	пройден
17	запуск приложения на устройстве с разрешением экрана 1080 x 2340	отображение окна главного меню, без искажений пропорций кнопок	отображение окна главного меню, без искажений пропорций кнопок	пройден
18	запуск приложения на устройстве с Android версии 11.0	отображение окна главного меню	отображение окна главного меню	пройден
19	запуск приложения на устройстве с Android версии 9.0	отображение окна главного меню	отображение окна главного меню	пройден
20	запуск приложения на устройстве с Android версии 8.0	отображение окна главного меню	отображение окна главного меню	пройден

Далее, представлена дополнительная информация к некоторым из проведенных тестов.

Тесты 8 и 9. Камера в режиме сканирования была наведена на рисунок, сделанный по указаниям во вкладке “медведь”. Рисунок не имеет ошибок в построении и состоит из почти прямых линий. После сканирования поверх рисунка отобразилась иллюстрация. Скриншот приведен на рисунке 32 (а).

Камера в режиме сканирования была наведена на рисунок, сделанный по указаниям во вкладке “медведь”. Рисунок имеет ошибки в построении и кривые линии. После сканирования поверх рисунка не отобразилась иллюстрация. Скриншот приведен на рисунке 32 (б).

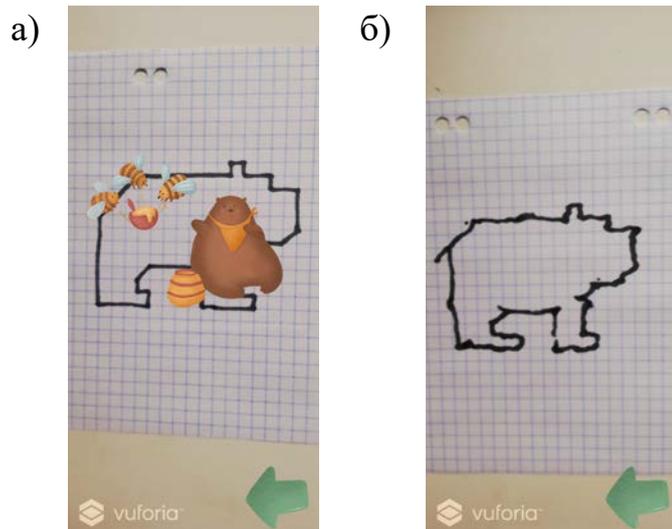


Рисунок 32 – Скриншоты экрана при сканировании рисунков, удовлетворяющего и не удовлетворяющего требованиям

Тесты 10 и 11. Камера в режиме сканирования была наведена на рисунок, сделанный по указаниям во вкладке “лиса”. Рисунок не имеет ошибок в построении и состоит из почти прямых линий. После сканирования поверх рисунка отобразилась модель в виде лисы. Скриншот приведен на рисунке 33 (а).

Камера в режиме сканирования была наведена на рисунок, сделанный по указаниям во вкладке “медведь”. Рисунок не имеет ошибок в построении и состоит из почти прямых линий. После сканирования поверх рисунка отобразилась иллюстрация в виде медведя. Скриншот приведен на рисунке 33 (б).

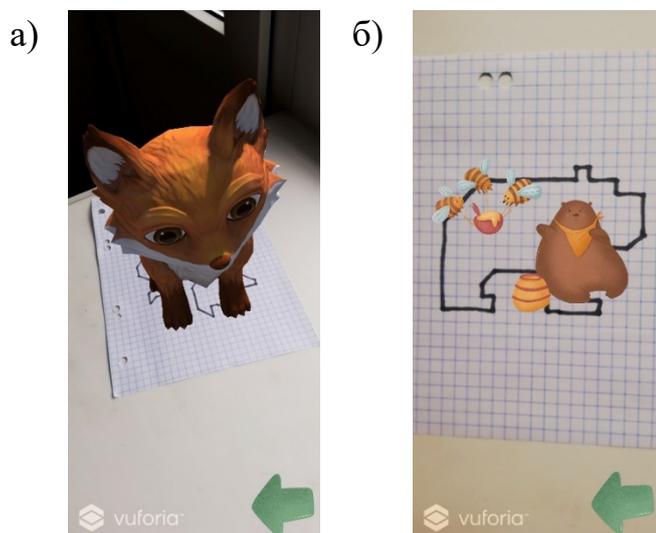


Рисунок 33 – Скриншоты экрана при сканировании разных рисунков

Тест 12. Была нажата кнопка “дорожки” в режиме “рисования по контурам”. После чего произошло отображение контуров изображения поверх видеоряда с камеры устройства. Скриншот приведен на рисунке 34.

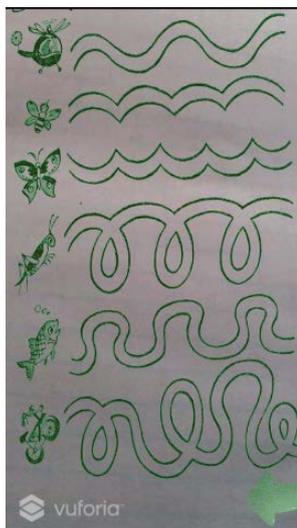


Рисунок 34 – Скриншот экрана в режиме “рисования по контурам”

Тесты с 15 по 17. В тестах с 15 по 17 проверялась корректность отображения интерфейса при различных разрешениях экрана. Для тестирования были использованы виртуальные устройства. На рисунке 35 приведены скриншоты главного меню. Под буквой (а) приведен скриншот экрана устройства с разрешением 480 x 800, под буквой (б) – 1440 x 2560, под буквой (в) – 1080 x 2340.

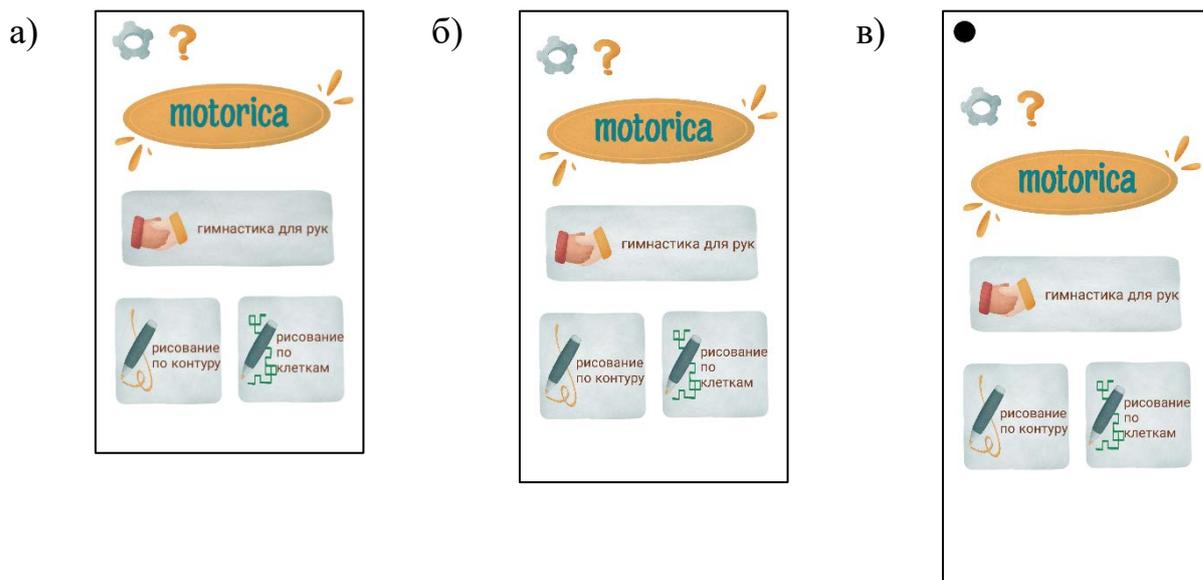


Рисунок 35 – Скриншоты главного меню при разных разрешениях экрана

ЗАКЛЮЧЕНИЕ

В рамках выпускной квалификационной работы разработано мобильное приложение для развития мелкой моторики. Для этого выполнены следующие действия:

- рассмотрены существующие приложения и проекты, ориентированные на помощь в развитии мелкой моторики;
- проведен анализ современных программных технологий на возможность их применения в приложении по развитию мелкой моторики;
- спроектировано и реализовано собственное приложение для развития мелкой моторики на основе проведенного анализа;
- проведено тестирование разработанного приложения.

Полученное приложение может быть полезно использовать для самостоятельных занятий по развитию мелкой моторики детей дошкольного или раннего школьного возраста и взрослых, имеющих проблемы с мелкой моторикой.

При дальнейшем развитии приложения можно увеличить количество упражнений, разнообразить их вариативность.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Маркувинас, И. В. Мелкая моторика как средство развития речи / И. В. Маркувинас. // Наука и реальность/Science & Reality. – 2021. – № 2 (6). – 71-72 с.
2. Чельшева, С. А. Развитие мелкой моторики у детей младшего подросткового возраста / С. А. Чельшева. // Психология и педагогика: методика и проблемы практического применения. – 2012. – №27. – 48-51 с.
3. Как работает Android, часть 1. – Текст : электронный // Хабр : [сайт]. – 3 августа 2017. – URL: <https://habr.com/ru/company/solarsecurity/blog/334796/> (дата обращения 13.12.2021).
4. GoKids : [официальный сайт]. – URL: <https://gokidsmobile.com/ru/> (дата обращения 02.06.2022).
5. Vini Bambini : [официальный сайт]. – URL: <https://teachdraw.com/> (дата обращения 10.03.2022).
6. Abecedaire Education : [официальный сайт]. – URL: <http://abecedaire.co.uk/> (дата обращения 10.03.2022).
7. Kedronic : [официальный сайт]. – URL: <https://kedronic.com/dots> (дата обращения 10.03.2022).
8. Бизнес-идея: мобильное приложения для развития мелкой моторики рук. – Текст. Изображение : электронные // Контур : журнал. – URL: <https://kontur.ru/ib2016/3471> (дата обращения 10.03.2022).
9. BinaryLabs : [официальный сайт]. – URL: <http://www.dexteria.net/> (дата обращения 10.03.2022).
10. ReHand : [официальный сайт]. – URL: <https://rehand.net/en/home/> (дата обращения 10.03.2022).
11. Indigo : [официальный сайт]. – URL: <https://ru.indigoclub.org/> (дата обращения 10.03.2022).
12. SketchAR : [официальный сайт]. – URL: <https://sketchar.io/> (дата обращения 10.03.2022).

13. QuiverVision : [официальный сайт]. – URL: <https://quivervision.com/>
(дата обращения 10.03.2022).

ПРИЛОЖЕНИЕ А

Листинги

Листинг 1 – Скрипт Timer

```
using UnityEngine;
using UnityEngine.UI;

public class Timer : MonoBehaviour
{
    public class Exercise
    {
        private int slidStart;
        private int slideCount;
        private string exerciseString;

        public Exercise(int slidStart, int slideCount, string exesiseString)
        {
            this.slidStart = slidStart;
            this.slideCount = slideCount;
            this.exesiseString = exerciseString;
        }
        public int getSlidStart()
        {
            return this.slidStart;
        }
        public int getslideCount()
        {
            return this.slideCount;
        }
        public string getExString()
        {
            return this. exerciseString;
        }
    }

    public AudioSource myFx;
    public AudioClip ClickSound;
    public Sprite[] animatedImages;
    public Image animateImageObj;
    public GameObject ImageEnd, buttonInc, buttonDec;
    [SerializeField] private float maTime = 20f;
    int res;
    int iterations = 0;
    float timeLeft;
    Image timerBar;
```

Продолжение приложения А

```
Exercise[] slides = { new Exercise(0, 2, " \nОдновременно разводите пальцы левой и
правой рук на максимальное расстояние, а после сводите обратно"),
    new Exercise(2, 2, " \nСжимайте одновременно пальцы на левой и правой руках в
кулак и разжимайте до максимального положения"),
    new Exercise(4, 2, " \nКончики пальцев обеих рук прижмите друг к другу,
образуя подобие сферы, затем соедините ладони у запястья. Примите исходное
положение."),
    new Exercise(6, 4, " \nОдновременно кончиками больших пальцев правой и левой
рук поочередно касайтесь кончиков указательных, средних безымянных пальцев и
мизинцев.") };

[SerializeField] private Text timerText;
[SerializeField] private Text ExText;
void Inc()
{
    if (iterations < slides.Length) { timeLeft = 0; }
    else { buttonInc.SetActive(false); }
}

void Dec()
{
    if (iterations > 1) { timeLeft = 0; iterations= iterations - 2;
buttonDec.SetActive(true); }
    else { if (iterations == 1) { iterations = iterations - 2;
buttonDec.SetActive(false);timeLeft = 0;} }
}

void Start()
{
    timerBar = GetComponent<Image>();
    timerText = GameObject.Find("TimeText").GetComponent<Text>();
    timeLeft = maTime;
    ExText = GameObject.Find("ExerciseText").GetComponent<Text>();
    ExText.text = ExText.text = iterations + 1 + " / " + slides.Length +
slides[iterations].getExString();
}
```

```

void Update()
{
    if (timeLeft > 0)
    {
        timeLeft -= Time.deltaTime;
        timerBar.fillAmount = 1f-timeLeft / maTime;
        res = (int)(timerBar.fillAmount*100);
        timerText.text = res.ToString()+"%";
        animateImageObj.sprite = animatedImages[((int)(Time.time %
slides[iterations].getslideCount())) + slides[iterations].getSlidStart()];

    }
    else
    {
        if (iterations < slides.Length - 1)
        {
            iterations++;
            timeLeft = maTime;
            ExText.text = iterations + 1 + " / " + slides.Length +
slides[iterations].getExString();
            if(iterations == 1) buttonDec.SetActive(true);
        }
        else
        {
            ImageEnd.SetActive(true);
            buttonInc.SetActive(false);
            buttonDec.SetActive(false);
            if (iterations == slides.Length)
            { myFx.PlayOneShot(ClickSound); iterations++; };
        }
    }
}
}

```

Листинг 2 – скрипт FileName

```

using UnityEngine;
using UnityEngine.UI;

public class FileName : MonoBehaviour
{
    public Text fileNameText;

    public int index;

    public void OnClick()
    {
        FileManager.instance.SelectFile(index);
    }
}

```

Листинг 3 – скрипт FileManager

```

using UnityEngine;
using UnityEngine.UI;
using System.IO;
using System.Linq;

public class FileManager : MonoBehaviour
{
    public Texture LoadIcon;
    public GameObject filesListPan, filesContent, filePrefab, Panel;
    public RawImage Img;
    private DirectoryInfo dirInfo = new DirectoryInfo("/mnt/sdcard/Pictures/");
    private FileInfo[] files;
    private GameObject[] instancedObjs;
    public static FileManager instance;
    private void Awake()
    {
        instance = this;
    }
    public void LoadList()
    {
        filesListPan.SetActive(true);
        Img.gameObject.SetActive(false);
        files = new string[] { "*.png" }.SelectMany(ext => dirInfo.GetFiles(ext,
SearchOption.AllDirectories)).ToArray();
        instancedObjs = new GameObject[files.Length];
        for (int i = 0; i < files.Length; i++)
        {
            FileName file = Instantiate(filePrefab,
filesContent.transform).GetComponent<FileName>();
            file.fileNameText.text = files[i].Name;
            file.index = i;
            instancedObjs[i] = file.gameObject;
        }
    }

    public void SelectFile(int index)
    {
        WWW www = new WWW("file://" + files[index].FullName);
        Img.texture = www.texture;
        SelectContours.ImageProcessing(Img);

        filesListPan.SetActive(false);
        Panel.SetActive(false);
        Img.gameObject.SetActive(true);
    }
}

```

```

        foreach (GameObject obj in instancedObjs)
        {
            Destroy(obj);
        }
    }
    public void Exit()
    {
        Panel.SetActive(true);
        Img.texture = LoadIcon;
    }
}

public class SelectContours : MonoBehaviour
{
    public static void ImageProcessing(RawImage Img)
    {
        Texture2D texture2 = (Texture2D)Img.texture;
        Color color;
        for (int y = 0; y < texture2.height; y++)
        {
            for (int x = 0; x < texture2.width; x++)
            {
                color = ((Texture2D)Img.texture).GetPixel(x, y);
                float hit = 0.2126f * 1 - color.r + 0.7152f * 1 - color.g + 0.0722f *
1 - color.b;
                color.r = hit;
                texture2.SetPixel(x, y, color);
            }
        }
        texture2.Apply();
        for (int y = 0; y < texture2.height; y++)
        {
            for (int x = 0; x < texture2.width; x++)
            {
                color = texture2.GetPixel(x, y);
                float z1 = texture2.GetPixel(x - 1, y - 1).r;
                float z2 = texture2.GetPixel(x, y - 1).r;
                float z3 = texture2.GetPixel(x + 1, y - 1).r;
                float z7 = texture2.GetPixel(x - 1, y + 1).r;
                float z8 = texture2.GetPixel(x, y + 1).r;
                float z9 = texture2.GetPixel(x + 1, y + 1).r;
                float gx = z1 + z2 + z3 - z7 - z8 - z9;
                float gy = z3 + ((Texture2D)Img.texture).GetPixel(x + 1, y).r + z9 -
z1 - ((Texture2D)Img.texture).GetPixel(x - 1, y).r - z7;
                color.g = Mathf.Pow((Mathf.Pow(gx, 2) + Mathf.Pow(gy, 2)), 0.5f);
                texture2.SetPixel(x, y, color);
            }
        }
        texture2.Apply();
        for (int y = 0; y < texture2.height; y++)
        {
            for (int x = 0; x < texture2.width; x++)
            {
                color = texture2.GetPixel(x, y);
                color.a = color.g;
                color.r = color.g = color.b = 0;
                texture2.SetPixel(x, y, color);
            }
        }
        texture2.Apply();
        Img.texture = (Texture)texture2;
    }
}

```