

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего образования

**«Южно-Уральский государственный университет  
(национальный исследовательский университет)»**

**Высшая школа электроники и компьютерных наук**

**Кафедра системного программирования**

РАБОТА ПРОВЕРЕНА

Рецензент

Ведущий разработчик

ООО "ВОРТЕКСКОД"

\_\_\_\_\_ П.А. Михайлов

“ \_\_\_\_ ” \_\_\_\_\_ 2018 г.

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой,

д.ф.-м.н., профессор

\_\_\_\_\_ Л.Б. Соколинский

“ \_\_\_\_ ” \_\_\_\_\_ 2018 г.

## **РАЗРАБОТКА ПРОГРАММЫ «ИЗУЧЕНИЕ РУССКОГО ЯЗЫКА КАК ИНОСТРАННОГО» ДЛЯ ОС WINDOWS**

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
ЮУрГУ – 02.03.02.2018.115-080.ВКР

Научный руководитель

к.ф.-м.н., доцент кафедры СП

\_\_\_\_\_ С.А. Иванов

Автор работы,

студент группы КЭ-402

\_\_\_\_\_ С.А. Скориков

Ученый секретарь

(нормоконтролер)

\_\_\_\_\_ О.Н. Иванова

“ \_\_\_\_ ” \_\_\_\_\_ 2018 г.

Челябинск-2018

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

УТВЕРЖДАЮ

Заведующий кафедрой,

д.ф.-м.н., профессор

\_\_\_\_\_  
Л.Б. Соколинский  
09.02.2018

**ЗАДАНИЕ**

**на выполнение выпускной квалификационной работы бакалавра**  
студенту группы КЭ-402  
Скорикову Сергею Андреевичу,  
обучающемуся по направлению  
02.03.02 «Фундаментальная информатика и информационные технологии»

**1. Тема работы** (утверждена приказом ректора от 04.04.2018 № 580)

Разработка программы «Изучение русского языка как иностранного» для ОС Windows.

**2. Срок сдачи студентом законченной работы:** 05.06.2018.

**3. Исходные данные к работе**

3.1. Unity Scripting API. [Electronic resource] URL:

<https://docs.unity3d.com/ScriptReference/> (date of access: 21.05.2018).

3.2. Гальскова Н.Д., Гез Н.И. Теория обучения иностранным языкам. Лингводидактика и методика. – М.: Академия, 2006. – 336 с.

3.3. Хейлмберг А. Язык программирования C#. / Хейлмберг А., Торгерсен М., Вилтамут С., Голд П. – СПб.: Питер, 2012. – 784 с.

**4. Перечень подлежащих разработке вопросов**

4.1. Произвести анализ предметной области.

4.2. Спроектировать программу.

4.3. Реализовать программу.

4.4. Протестировать программу.

**5. Дата выдачи задания: 09.02.2018.**

**Научный руководитель**

**К.ф.-м.н, доцент кафедры СП**

**Задание принял к исполнению**

**С.А. Иванов**

**С.А. Скориков**

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	5
1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ .....	7
1.1. Использование приложений для обучения иностранным языкам .....	7
1.2. Обзор аналогичных проектов .....	8
2. ПРОЕКТИРОВАНИЕ .....	11
2.1. Функциональные требования .....	11
2.2. Нефункциональные требования .....	11
2.3. Описание программы.....	11
2.4. Диаграмма вариантов использования .....	12
2.5. Интерфейс программы.....	13
3. РЕАЛИЗАЦИЯ .....	19
3.1. Средства реализации.....	19
3.2. Структура программы.....	20
3.3. Реализация программы .....	22
4. ТЕСТИРОВАНИЕ .....	30
4.1. Функциональное тестирование.....	30
ЗАКЛЮЧЕНИЕ .....	33
СПИСОК ЛИТЕРАТУРЫ .....	34

## **ВВЕДЕНИЕ**

### **Актуальность темы**

В последние десятилетия иностранные языки являются объектом пристального внимания и изучения не только ученых, но и людей, не связанных с научной деятельностью. При этом отмечается возрастающая роль, которую играют иностранные языки, в осуществлении влияния на сознание и деятельность широких слоев населения [9].

Также в современном мире нельзя недооценивать возрастающее влияние информационных технологий на повседневную жизнь и рабочую среду, где знание иностранных языков необходимо для полноценной и грамотной работы. Знание иностранного языка позволяет общаться с иностранцами, что способствует сотрудничеству и деловым связям, то есть расширению международных связей в целом, их укреплению [12].

В связи с этим, в настоящее время широкое распространение получили программы для изучения иностранных языков.

ОС Windows является наиболее популярной на данный момент [11]. Наличие разнообразных инструментов для разработчиков, поддержка со стороны крупных организаций, большое сообщество разработчиков – основные качества ОС Windows. В соответствии с этим было решено разработать программу для ОС Windows.

### **Цель и задачи**

Целью данной работы является разработка программы «Изучение русского языка как иностранного».

Для достижения поставленной цели необходимо решить следующие задачи:

- 1) произвести анализ предметной области;
- 2) спроектировать программу;
- 3) реализовать программу;
- 4) протестировать программу.

## **Структура и объем работы**

Работа состоит из введения, четырех глав, заключения и библиографии. Объем работы составляет 35 страниц, объем библиографии – 16 источников.

В первой главе «Анализ предметной области» происходит анализ использования приложений для обучения иностранным языкам и обзор аналогичных проектов.

Во второй главе «Проектирование» рассматриваются требования к программе, происходит описание программы, составляется диаграмма вариантов использования, а также проектируется интерфейс программы.

В третьей главе «Реализация» рассматриваются средства реализации, структура программы, а также реализация самой программы.

В четвертой главе «Тестирование» проводится функциональное тестирование разработанной программы.

В заключении сделаны выводы о проделанной работе.

## **1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ**

### **1.1. Использование приложений для обучения иностранным языкам**

Мобильное обучение – это деятельность, осуществляемая регулярно посредством компактных, портативных мобильных устройств и технологий и позволяющая обучающимся стать более продуктивными, общаясь, получая или создавая информацию [1].

В целом, большинство исследователей [5, 2, 10] приходят к выводу, что уникальность мобильного обучения по сравнению с традиционными методами обучения и современными методами, такими как e-learning и смешанное обучение, заключается в том, что обучающиеся прежде всего не привязаны к определенному времени и месту, имея доступ к учебному материалу всегда, в любое удобное время.

На сегодняшний момент в зарубежной педагогической литературе существует несколько определений мобильного обучения, основывающихся на технологических особенностях мобильных устройств и на дидактических возможностях, которые предоставляются этими технологиями. Согласно проекту MoLeNET, мобильное обучение – это использование удобных портативных мобильных устройств и беспроводных, доступных всегда технологий, для облегчения, поддержки, оптимизации и расширения процессов обучения и изучения [4].

В настоящее время пользователям мобильных устройств доступно огромное количество приложений для изучения иностранных языков. Причем, на сегодняшний день существуют мобильные приложения и программы, ориентированные на разные аспекты преподавания иностранного языка.

Мобильные устройства успешно (телефоны, смартфоны, планшетные компьютеры) используются при изучении различных учебных дисциплин, и иностранный язык не является исключением. На сегодняшний день мобильные приложения для изучения иностранных языков приобретают ши-

рокую популярность. Они имеют большой потенциал в повышении эффективности процесса изучения иностранных языков и призваны значительно усовершенствовать процесс иноязычной подготовки широкого круга обучающихся, открыть его новые стороны и превратить из серьезного трудоемкого процесса в увлекательное занятие. Практика показывает, что мобильные приложения имеют немалое преимущество перед традиционными методами обучения: интенсификация самостоятельной деятельности, индивидуализация обучения, повышение познавательной активности и мотивации учения.

## **1.2. Обзор аналогичных проектов**

В ходе выполнения работы прямых аналогов разрабатываемой программы для ОС Windows не было обнаружено, поэтому были рассмотрены приложения по изучению русского языка как иностранного на платформе Android. Из большого списка можно выделить несколько успешных приложений.

### **LinguaLeo**

Приложение состоит из нескольких частей. В «джунглях» доступны тысячи разнообразных текстов, основная задача которых – научить пользователей новым словам. Перевод того или иного слова можно получить, кликнув по нему. Слова можно добавлять в специальные списки для дальнейшего запоминания. В LinguaLeo [14] существуют курсы по грамматике. Однако, доступ к большинству открывается только за дополнительную плату. В наличии также различные тематические словари, упражнения по правильному произношению, запоминанию слов, а также грамматические тесты.

Достоинства:

- 1) большое количество различных упражнений;
- 2) возможность изучать грамматику;
- 3) большое количество статей и обучающих видео.



Недостатки:

- 1) часть контента платная.

### **Duolingo**

Все задания группируются по уровням сложности, начинается процесс с идентификации фотографий и перевода простых слов, а заканчивается длинными предложениями. Приложение исправляет ученика при наличии грамматических ошибок или появлении неправильного перевода. Уроки содержат задания по практике разговорной речи, аудированию, переводу, а также из разнообразных тестов для всестороннего набора навыков [13].

Достоинства:

- 1) весь контент бесплатен;
- 2) большая словесная база.

Недостатки:

- 1) малое количество грамматических упражнений.

### **Anki**

Anki [3] – приложение для запоминания слов, основывающаяся на методике интервального повторения. Интервальное повторение – это техника обучения, при которой повторение материала (новых слов, формул и т.д.) совершается через определенные промежутки времени, таким образом, обеспечивая долговременное запоминание.

Достоинства:

- 1) возможность изучения любого языка с помощью этого приложения (слово и его значение может внести сам пользователь);
- 2) пользователь сам может добавить перевод слова.

Недостатки:

- 1) отсутствие упражнений.

Все рассмотренные приложения являются лидерами рейтингов в магазине приложений Google Play и занимают лидирующие позиции уже долгое время.

В главе 1 был проведен анализ использования приложений для обучения иностранным языкам, а также проведен анализ аналогичных проектов.

## **2. ПРОЕКТИРОВАНИЕ**

### **2.1. Функциональные требования**

Приложение должно предоставлять следующие возможности:

- 1) наличие возможности отображать теоретический материал;
- 2) наличие возможности проходить тестирование;
- 3) наличие возможности просмотреть результат прохождения теста;
- 4) наличие возможности информировать игрока о верном или неверном ответе звуком;
- 5) наличие возможности информировать игрока о верном или неверном ответе окраской кнопки выбора;
- 6) наличие возможности управлять настройками.

### **2.2. Нефункциональные требования**

Нефункциональные требования описывают свойства и ограничения, накладываемые на систему. Для реализации приложения были выявлены следующие требования:

- 1) программа должна быть реализована на платформе Unity с использованием интегрированной среды Visual Studio;
- 2) программа должна быть реализована с использованием языка программирования C#;
- 3) программа должна работать на ОС Windows версии 7 и выше.

### **2.3. Описание программы**

Обучающая программа «Russian for foreigner». Пользователю предлагается несколько разделов с темами русского языка для изучения, а также проверки знаний по каждому разделу.

Всего 3 раздела.

Раздел «Lexicology» представляет собой раздел с темами по лексикологии.

Раздел «Orthography» представляет собой раздел с темами по орфографии.

Раздел «Morphology» представляет собой раздел с темами по морфологии.

В каждом из перечисленных разделов имеется возможность также переходить в раздел проверки знаний по пройденному материалу.

Проверка знаний по разделу представляет собой тест с вариантами ответов.

## 2.4. Диаграмма вариантов использования

С помощью языка графического описания для объектного моделирования UML [15], была построена диаграмма вариантов использования. Единственным актером системы, является пользователь. На рис. 1 представлены варианты использования игрового приложения.

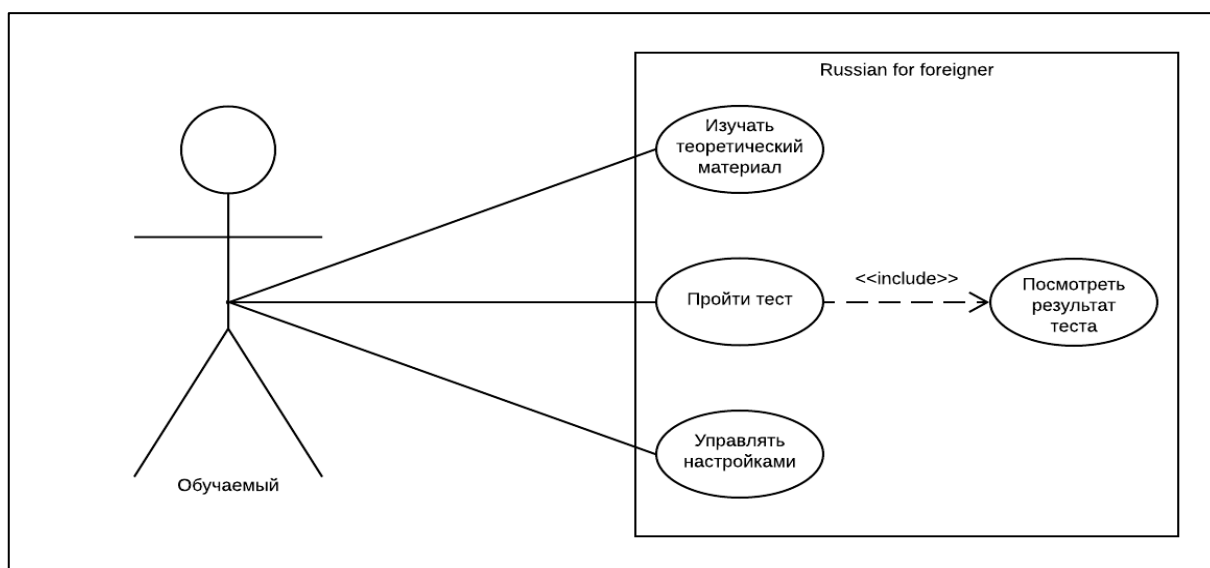


Рис. 1. Диаграмма вариантов использования

### Краткое описание вариантов использования

*Обучаемый* может изучать теоретический материал.

*Обучаемый* может пройти тест.

*Обучаемый* может посмотреть результат теста.

*Обучаемый* может управлять настройками.

## 2.5. Интерфейс программы

Графический интерфейс пользователя – разновидность пользовательского интерфейса, в котором элементы интерфейса (меню, кнопки, значки, списки и т.п.), представленные пользователю на дисплее, исполнены в виде графических изображений.

После загрузки приложения перед пользователем должен отображаться главный экран, представленный на рис. 2. При нажатии кнопки «Лексикология» осуществляется переход в лексикологический раздел. При нажатии кнопки «Орфография» осуществляется переход в орфографический раздел. При нажатии кнопки «Морфология» осуществляется переход в морфологический раздел. При нажатии кнопки настроек осуществляется переход в настройки программы. При нажатии кнопки «выход» осуществляется выход из приложения.

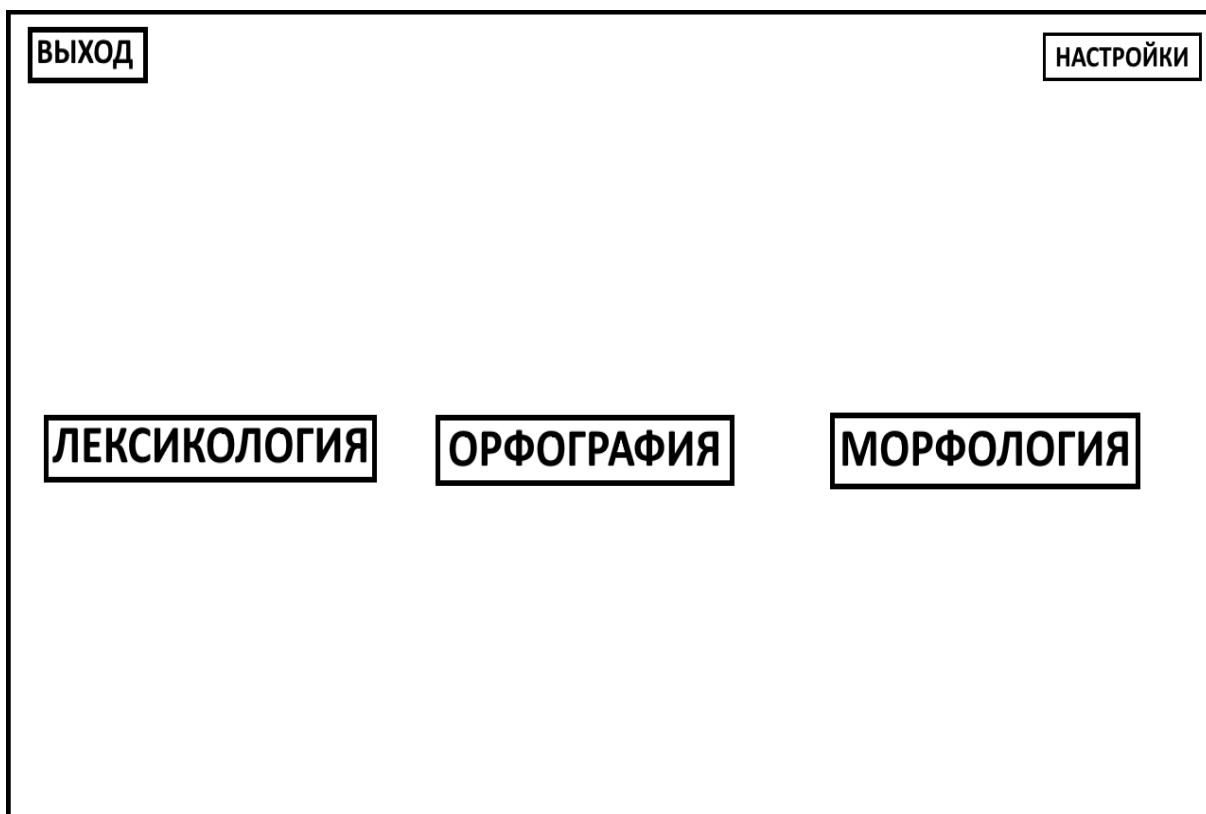


Рис. 2. Макет главного экрана

В меню раздела, представленном на рис. 3, должна содержаться кнопка возврата к предыдущему меню, кнопка настроек программы, кнопка «тео-

рия», кнопка проверки знаний и кнопка возврата в главное меню. При нажатии кнопки «Теория» осуществляется переход к выбору списка тем по теории. При нажатии кнопки проверки знаний осуществляется переход в тестирование по данному разделу. При нажатии кнопки настроек осуществляется переход в настройки программы. При нажатии кнопки возврата к предыдущему меню, осуществляется переход к предыдущему меню. При нажатии кнопки возврата в главное меню, осуществляется переход в главное меню.



Рис. 3. Макет меню раздела

В меню выбора темы для изучения, представленном на рис. 4, должен содержаться список тем для теоретического изучения, кнопка настроек программы, кнопка возврата к предыдущему меню и кнопка возврата в главное меню. При нажатии на тему для теоретического изучения осуществляется переход в соответствующую тему. При нажатии кнопки настроек осуществляется переход в настройки программы. При нажатии кнопки возврата к предыдущему меню, осуществляется переход к предыдущему меню. При нажатии кнопки возврата в главное меню, осуществляется переход в главное меню.

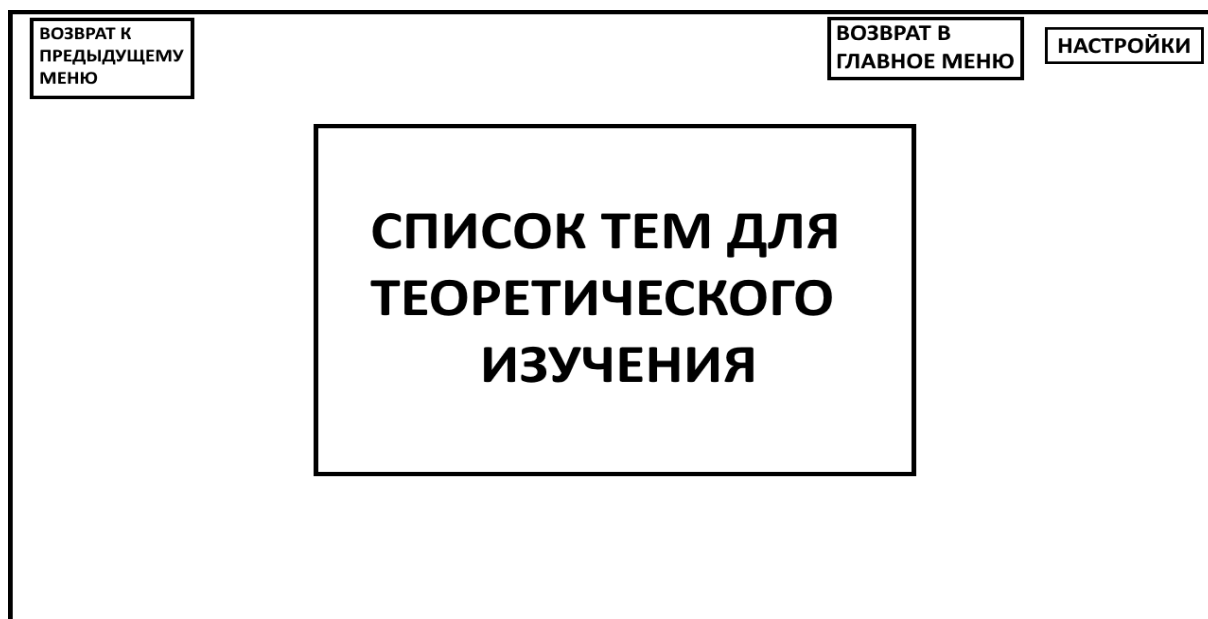


Рис. 4. Макет меню выбора темы для изучения теории

Отображение теоретической части, представленное на рис. 5, должно содержать блок содержащий теорию для изучения, кнопку настроек программы, кнопку возврата к предыдущему меню и кнопку возврата в главное меню. При нажатии кнопки настроек осуществляется переход в настройки программы. При нажатии кнопки возврата к предыдущему меню, осуществляется переход к предыдущему меню. При нажатии кнопки возврата в главное меню, осуществляется переход в главное меню.

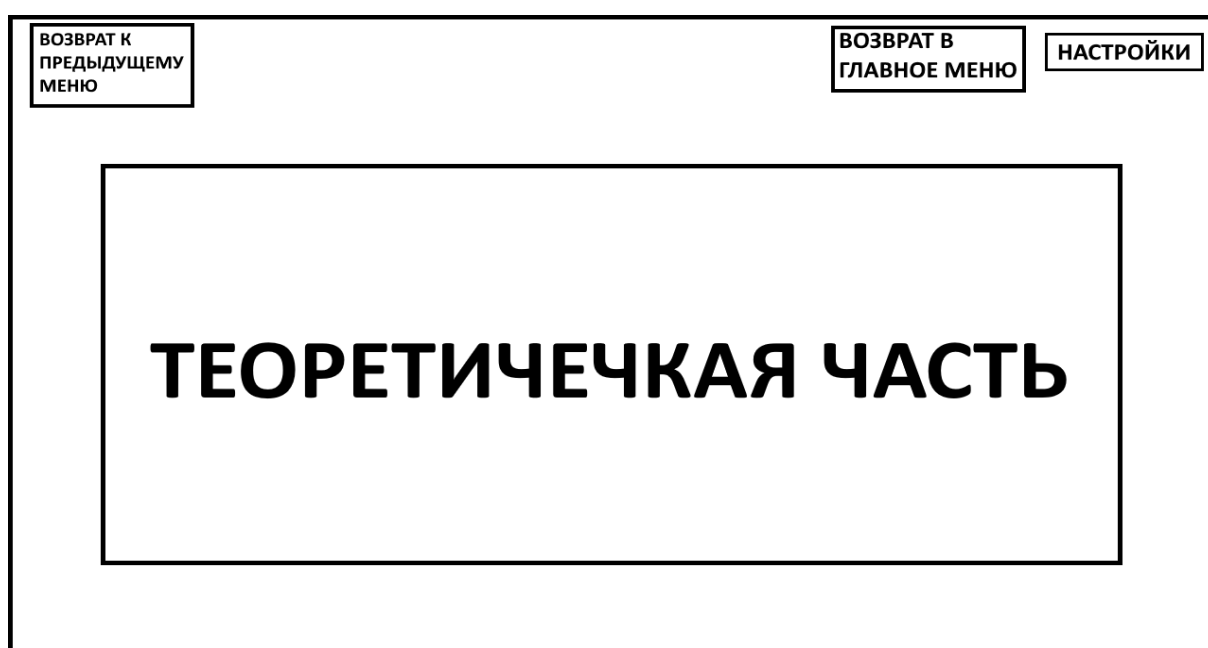


Рис. 5. Макет отображения теоретической части

Проверка знаний, представленная на рис. 6, должна содержать блок с отображаемым вопросом, блок с вариантами ответов на вопрос, кнопку настроек программы, кнопку возврата к предыдущему меню и кнопку возврата в главное меню. При нажатии на один из вариантов ответов осуществляется переход к следующему вопросу. После последнего вопроса осуществляется переход к результатам проверки знаний. При нажатии кнопки настроек осуществляется переход в настройки программы. При нажатии кнопки возврата к предыдущему меню, осуществляется переход к предыдущему меню. При нажатии кнопки возврата в главное меню, осуществляется переход в главное меню.



Рис. 6. Макет проверки знаний

Результат проверки знаний, представленный на рис. 7, должен содержать блок с отображаемым результатом по итогу прохождения теста, кнопку настроек программы, кнопку возврата к предыдущему меню, кнопку возврата в главное меню и кнопку «начать заново». При нажатии кнопки настроек осуществляется переход в настройки программы. При нажатии кнопки возврата к предыдущему меню, осуществляется переход к предыдущему меню. При нажатии кнопки возврата в главное меню, осуществляется переход в главное меню. При нажатии кнопки «начать заново» тест начинается заново.



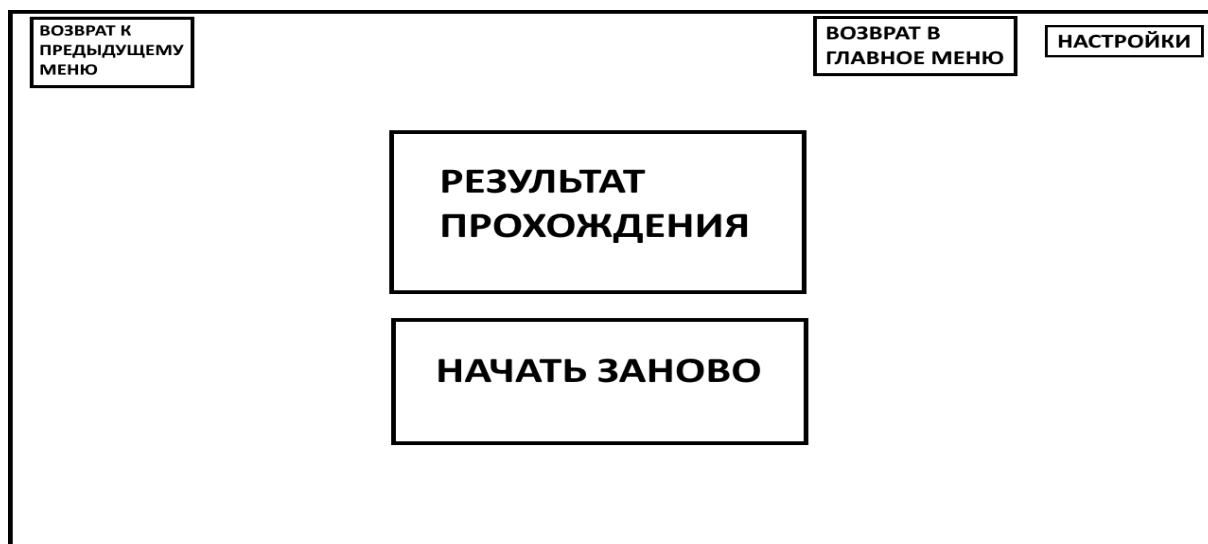


Рис. 7. Макет результата проверки знаний

Меню настроек программы, представленное на рис. 8, должно содержать кнопку управления музыкой в программе, кнопку управления звуком в программе, кнопку «о приложении» и кнопку возврата к предыдущему меню. При нажатии на кнопку управления музыкой включается или выключается музыка в программе, в зависимости от того, включена ли она сейчас. При нажатии на кнопку управления звуком включается или выключается звук в программе, в зависимости от того, включен ли он сейчас. При нажатии кнопки «о приложении» осуществляется переход в отображение информации о приложении. При нажатии кнопки возврата к предыдущему меню, осуществляется переход к предыдущему меню.

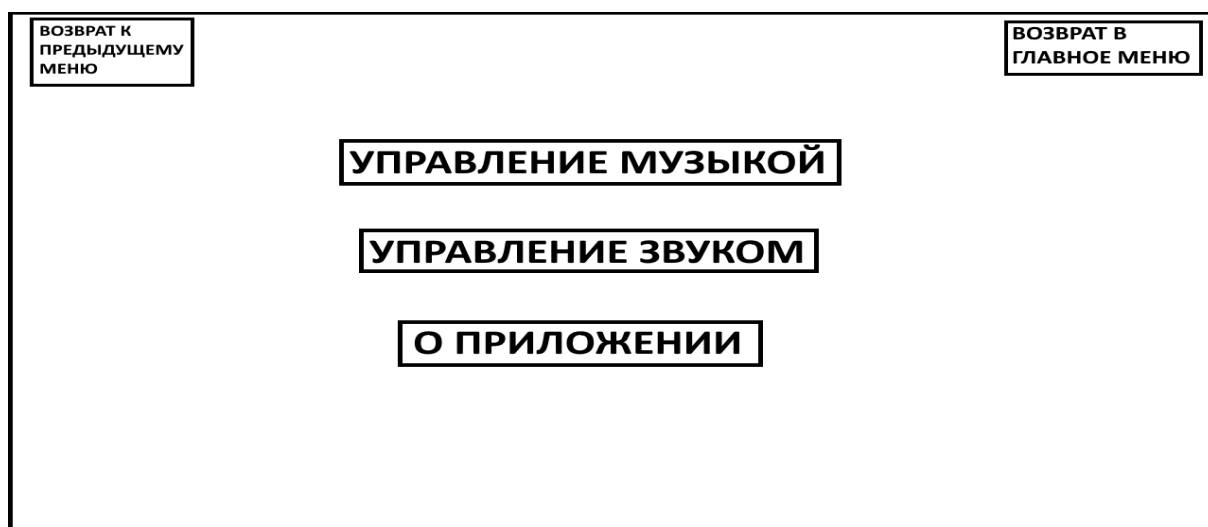


Рис. 8. Макет меню настроек программы

Отображение информации о приложении, представленное на рис. 9, должно содержать блок представления информации о приложении, кнопку возврата к предыдущему меню и кнопку возврата в главное меню. При нажатии кнопки возврата к предыдущему меню, осуществляется переход к предыдущему меню. При нажатии кнопки возврата в главное меню, осуществляется переход в главное меню.

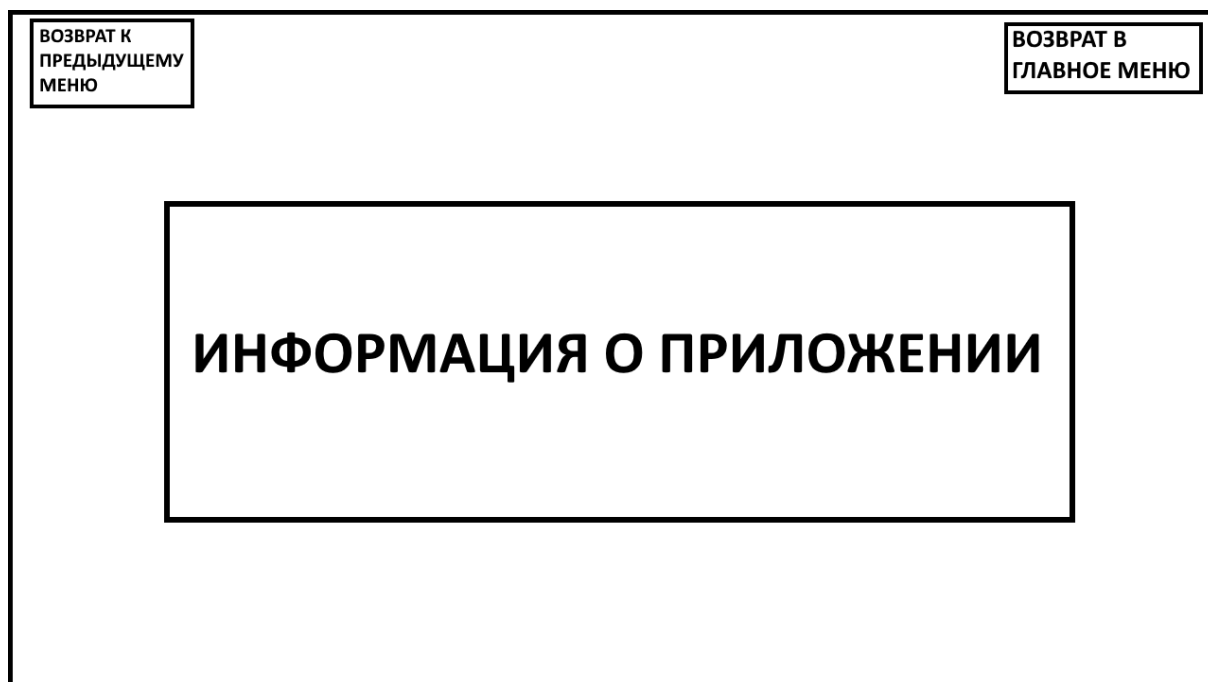


Рис. 9. Макет отображения информации о приложении

В главе 2 была спроектирована программа. В рамках проектирования программы были определены функциональные и нефункциональные требования. Разработана диаграмма вариантов использования, которая определяет одного актера и его основные доступные действия. Было спроектировано восемь макетов интерфейса.

### **3. РЕАЛИЗАЦИЯ**

#### **3.1. Средства реализации**

В наше время существует огромное количество как платных, так и бесплатных инструментов по разработке приложений, начиная от простых библиотек для известных языков программирования, заканчивая крупными редакторами с обширным функционалом.

Для создания качественной программы необходимо использовать качественные инструменты. Поэтому было принято решение использовать готовый инструмент для разработки.

Для реализации программы использованы среда разработки Unity, интегрированная среда программирования Visual Studio [8] и язык программирования C# [16].

Unity [6] – это среда разработки приложений и игр, работающая под операционными системами Windows и OS X. Unity позволяет разрабатывать как двумерные, так и трехмерные программы и игры. Среда позволяет компилировать проект под множество устройств. В этот список попадают устройства с ОС для ПК и MAC: Windows, OS X, Linux, с ОС для смартфонов: Android, iOS, Windows Phone, и даже под игровые консоли: PlayStation, Wii, Xbox. Кроме всего прочего, проект может быть запущен и в браузере, но только с помощью специального плагина от компании разработчиков Unity – UWP (unity web player). Среда имеет распространенный в среде редакторов Drag&Drop-интерфейс, который состоит из различных окон, которые, в свою очередь, очень просто настраиваются по высоте, ширине и размерам. Также Unity предоставляет API для доступа к средствам ввода и настройкам Android и iOS.

Основная используемая парадигма программирования в Unity – компонентно-ориентированное программирование. Данная парадигма опирается на понятие компонент.

Компонент – независимый модуль исходного кода программы, предназначенный для повторного использования и реализующийся в виде объединенного множества языковых конструкций, например, классов.

В Unity объекты создаются путем добавления к ним компонентов, реализующих какое-либо поведение. К любому объекту можно добавить любые другие компоненты из стандартного набора, либо реализованные собственными силами. Кроме того, программные объекты могут образовывать иерархию объектов. Программные объекты, являющиеся дочерними, имеют фиксированное относительно родительских объектов положение. Таким образом удобно добавлять различное поведение только частям объекта, при этом сохраняя общую структуру.

Компоненты, реализуемые собственными силами, называются скрипты. Скрипт является классом, реализованным на языке программирования C# или JavaScript, который наследует класс `MonoBehaviour`. Этот класс позволяет использовать скрипт в качестве компонента. При этом скрипт может в своей работе использовать обычные классы. Для реализации данной работы использован язык программирования C#.

Сцена является основным контейнером объектов. Все действия в игре происходят в какой-либо сцене. Условно каждую сцену можно назвать игровым уровнем, однако отдельной сценой можно сделать и главное меню игры. Переходы между сценами реализует разработчик. При этом, если объект не отмечен как «неуничтожаемый», все игровые объекты, принадлежащие данной сцене, уничтожаются, а вместо них появляются новые, находящиеся на загружаемой сцене. Объекты, отмеченные «неуничтожаемыми», переносятся в другую сцену неизменными. Вне сцены игровые объекты существовать не могут [7].

### **3.2. Структура программы**

На рис. 10 представлена диаграмма классов.

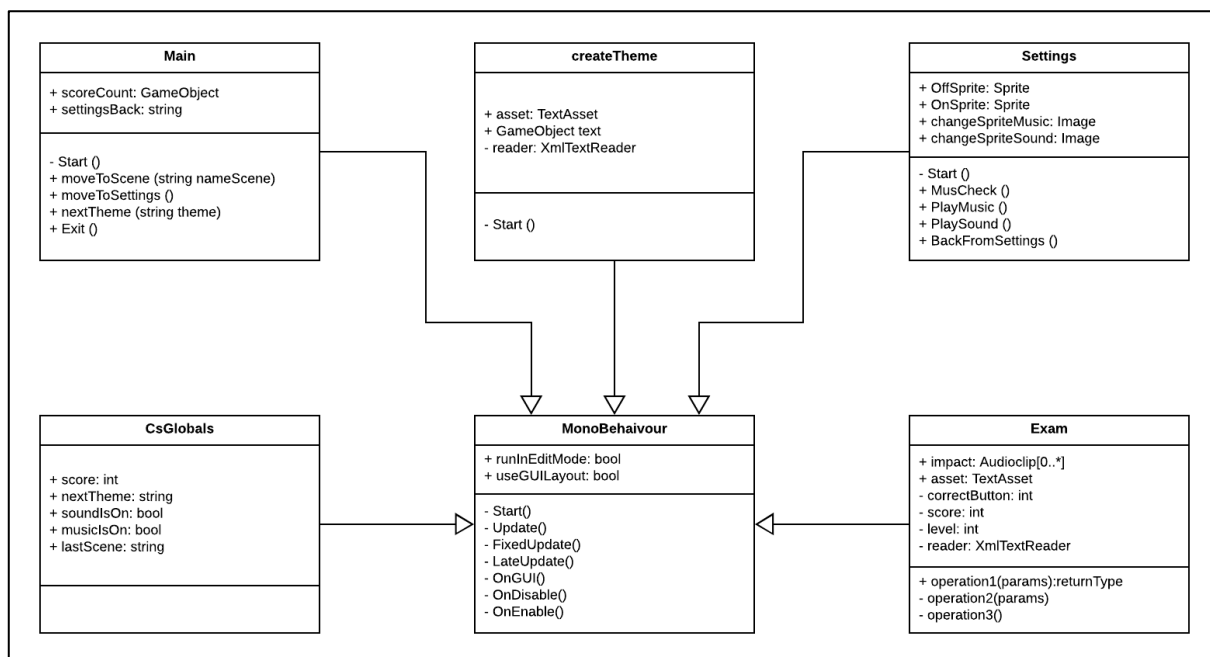


Рис. 10. Диаграмма классов

Класс `MonoBehaviour` – это базовый класс, от которого наследуются все классы (скрипты).

Класс `Main` отвечает за описание поведения приложения в зависимости от действий пользователя во всей программе.

Класс `Exam` отвечает за описание поведения приложения в зависимости от действий пользователя в проверке знаний всех разделов.

Класс `Settings` отвечает за описание поведения приложения в зависимости от действий пользователя в настройках.

Класс `createTheme` отвечает за описание поведения приложения в отображении теоретической части.

Класс `CsGlobal` хранит в себе статические переменные для дальнейшей работы с остальными классами.

В разработанном проекте содержится ряд каталогов, которые хранят в себе:

- 1) шрифты;
- 2) звуки;
- 3) скрипты;

- 4) изображения пользовательского интерфейса и иконки;
- 5) базы с вопросами и ответами;
- 6) базы с теорией.

Файловая структура представлена на рис. 11.

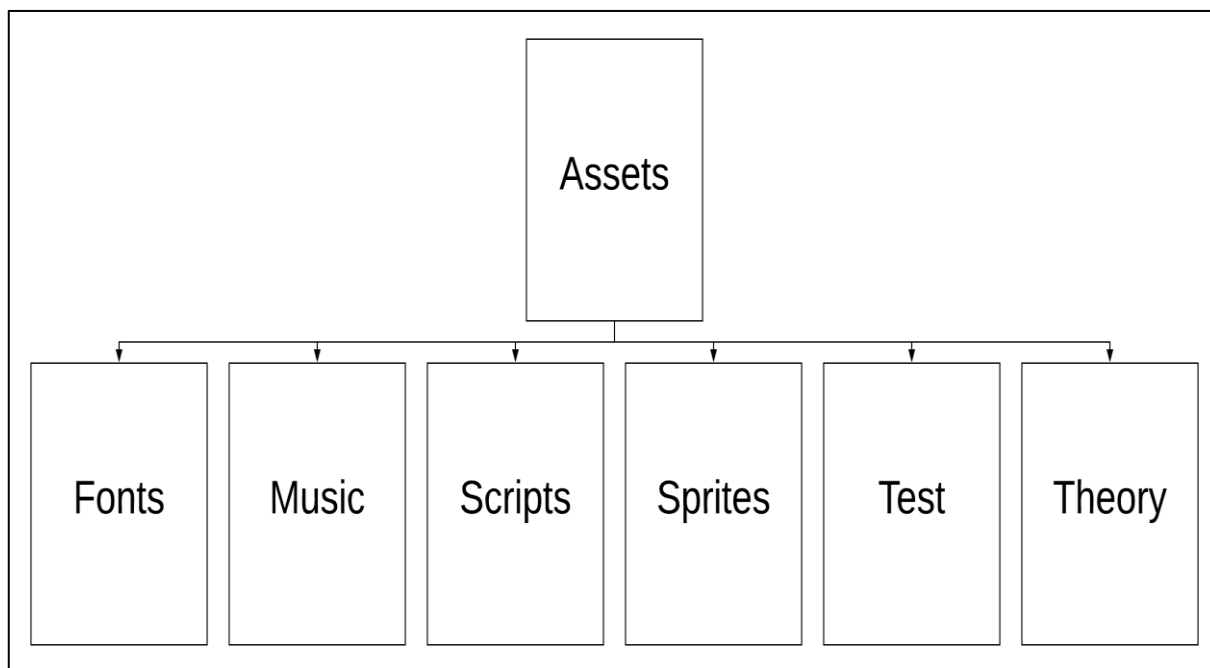


Рис. 11. Файловая структура программы

В директории Fonts содержатся шрифты, применяемые в программе. В директории Music содержатся звуки, которые применяются в программе. В директории Scripts содержатся скрипты с описанием всех классов и взаимодействий. В директории Sprites содержатся элементы всего пользовательского интерфейса. В директории Test хранятся базы с вопросами и ответами по каждому разделу. В директории Theory хранятся базы с теорией по каждому разделу.

### 3.3. Реализация программы

На рис. 15 представлен главный экран программы. Присутствуют кнопки «Лексикология» («Lexicology»), «Орфография» («Orthography»), «Морфология» («Morphology»), кнопка выхода из приложения («Exit»), а также кнопка настроек.



Рис. 15. Главный экран

На рис. 16 представлено меню раздела для изучения. Присутствует кнопка возврата к предыдущему меню («Back»), кнопка настроек программы, кнопка возврата в главное меню («Home»), кнопка «теория» (Theory) и кнопка проверки знаний («Exam»).

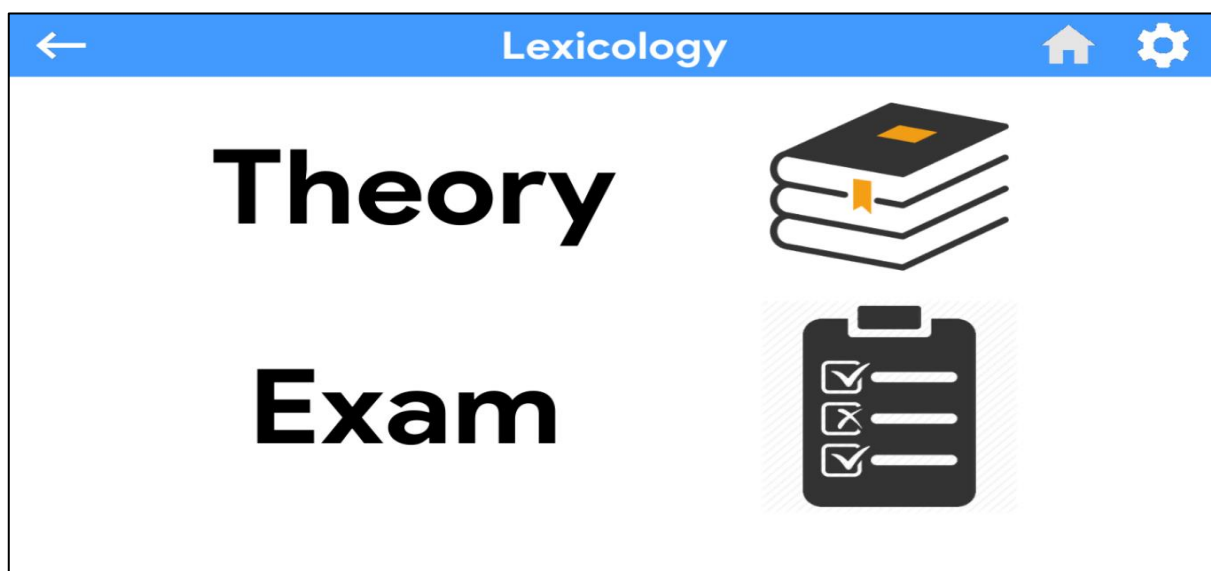


Рис. 16. Меню раздела для изучения лексикологии

На рис. 17 представлено меню выбора темы для изучения. Присутствует список тем для теоретического изучения, кнопка настроек программы, кнопка возврата в главное меню («Home») и кнопка возврата к предыдущему меню («Back»).

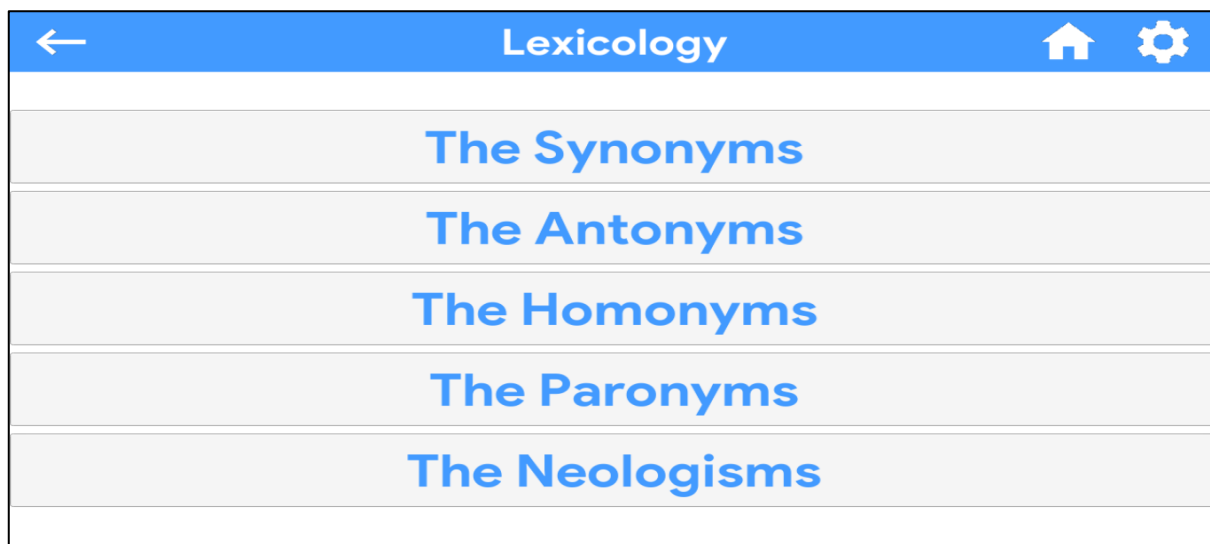


Рис. 17. Меню выбора темы для изучения лексикологии

На рис. 18 представлено отображение теоретической части. Присутствует блок содержащий теорию для изучения, кнопка настроек программы, кнопка возврата в главное меню («Home»), и кнопка возврата к предыдущему меню («Back»).

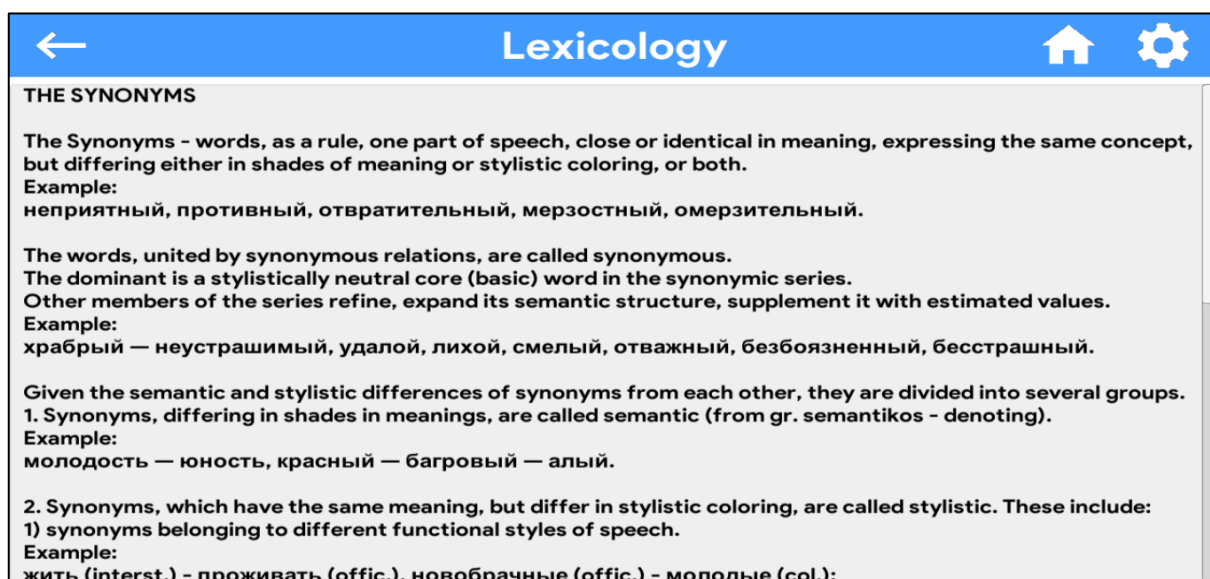


Рис. 18. Отображение теоретической части лексикологии

На рис. 19 представлена проверка знаний. Присутствует блок с отображаемым вопросом, блок с вариантами ответов на вопрос, кнопка настроек программы, кнопка возврата в главное меню («Home»), и кнопка возврата к предыдущему меню («Back»).



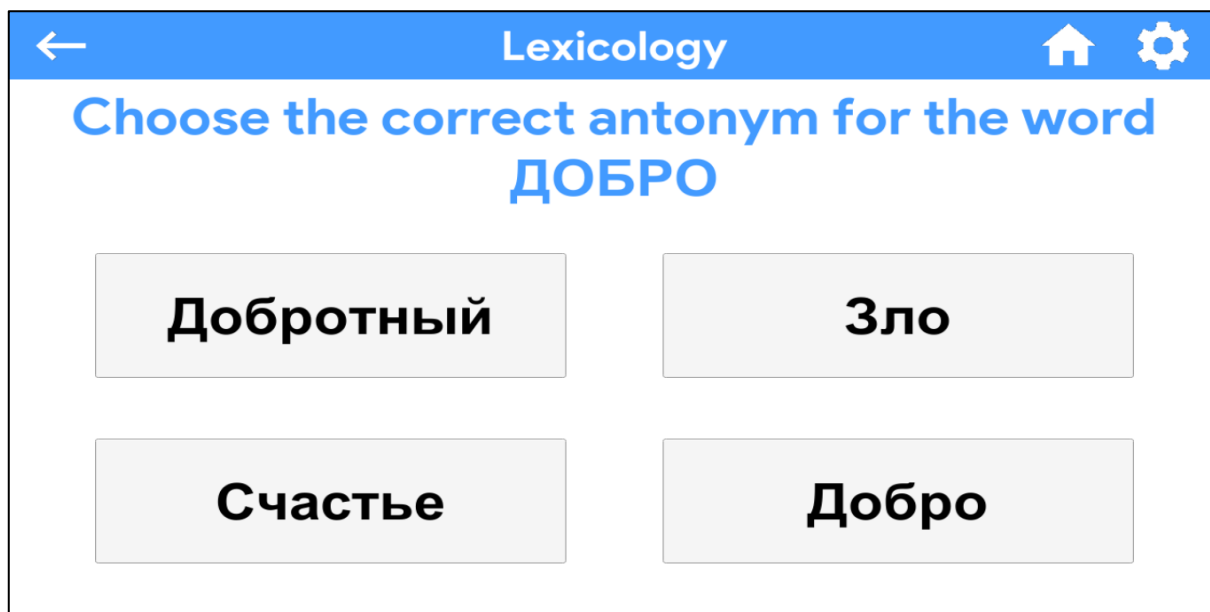


Рис. 19. Проверка знаний по лексикологии

Базы вопросов и ответов для проверки знаний хранятся в .xml файле, часть листинга которого представлен на рис. 20.

```
<?xml version="1.0" encoding="utf-8"?>
<catalog>
  <book
    question ="Choose the correct synonym for the word МОЛОДОСТЬ"
    correctAnswer ="Юность"
    incorrectAnswer1 ="Молодость"
    incorrectAnswer2 ="Молодой"
    incorrectAnswer3 ="Старость"/>
  <book
    question ="Choose the correct antonym for the word ДОБРО"
    correctAnswer ="Зло"
    incorrectAnswer1 ="Добро"
    incorrectAnswer2 ="Добротный"
    incorrectAnswer3 ="Счастье"/>
  <book
    question ="Words БЛИЗКО and ДАЛЕКО are?"
    correctAnswer ="The Antonyms"
    incorrectAnswer1 ="The Synonyms"
    incorrectAnswer2 ="The Paronyms"
    incorrectAnswer3 ="The Homonyms"/>
</catalog>
```

Рис. 20. Листинг базы вопросов и ответов по лексикологии

Обработкой XML-документа занимается XML-парсер, листинг которого представлен на рис. 21.

```
struct Quest
{
    public string question;
    public string correctAnswer;
    public string incorrectAnswer1;
    public string incorrectAnswer2;
    public string incorrectAnswer3;
    public Quest(string question, string correctAnswer, string incorrectAnswer1, string incorrectAnswer2, string incorrectAnswer3)

    {
        this.question = question;

        this.correctAnswer = correctAnswer;
        this.incorrectAnswer1 = incorrectAnswer1;
        this.incorrectAnswer2 = incorrectAnswer2;
        this.incorrectAnswer3 = incorrectAnswer3;
    }
}

private XmlTextReader reader;

void Start () {

    reader = new XmlTextReader(new StringReader(asset.text));
    while (reader.Read ()) {
        if (reader.Name == "book") {
            listQuest.Add (new Quest (
                reader.GetAttribute ("question"),

                reader.GetAttribute ("correctAnswer"),
                reader.GetAttribute ("incorrectAnswer1"),
                reader.GetAttribute ("incorrectAnswer2"),
                reader.GetAttribute ("incorrectAnswer3")));
        }
    }
}
```

Рис. 21. Листинг XML-парсера базы вопросов и ответов

Вывод вопросов и ответов на экран осуществляется с помощью метода `getQuestionsAnswers`. Листинг метода представлен на рис. 22.

```
IEnumerator getQuestionsAnswers(float f)
{
    yield return new WaitForSeconds(f);

    int numberQuest = Random.Range (0, listQuest.Count);

    string newQuestion = listQuest [numberQuest].question;
    string correctAnswer = listQuest [numberQuest].correctAnswer;

    listWord.Add (listQuest [numberQuest].incorrectAnswer1);
    listWord.Add (listQuest [numberQuest].incorrectAnswer2);
    listWord.Add (listQuest [numberQuest].incorrectAnswer3);

    List<GameObject> tmpwords = new List<GameObject>();

    tmpwords.AddRange (words);
    correctButton = Random.Range (0, tmpwords.Count);
    tmpwords [correctButton].GetComponentInChildren<Text> ().text = cor-
rectAnswer;
    tmpwords.RemoveAt (correctButton);
    question.GetComponentInChildren<Text> ().text = newQuestion;

    for (int i = 0; i < tmpwords.Count; i++) {
        int numberAns = Random.Range (0, listWord.Count);
        tmpwords [i].GetComponentInChildren<Text> ().text = listWord
[numberAns];

        listWord.RemoveAt (numberAns);
    }
}
```

Рис. 22. Листинг метода `getAnswersQuestions`

На рис. 23 представлен результат проверки знаний. Присутствует блок с отображаемым результатом по итогу прохождения теста, кнопка возврата в главное меню («Home»), кнопка настроек программы, кнопка «начать заново» («Restart») и кнопка возврата к предыдущему меню.

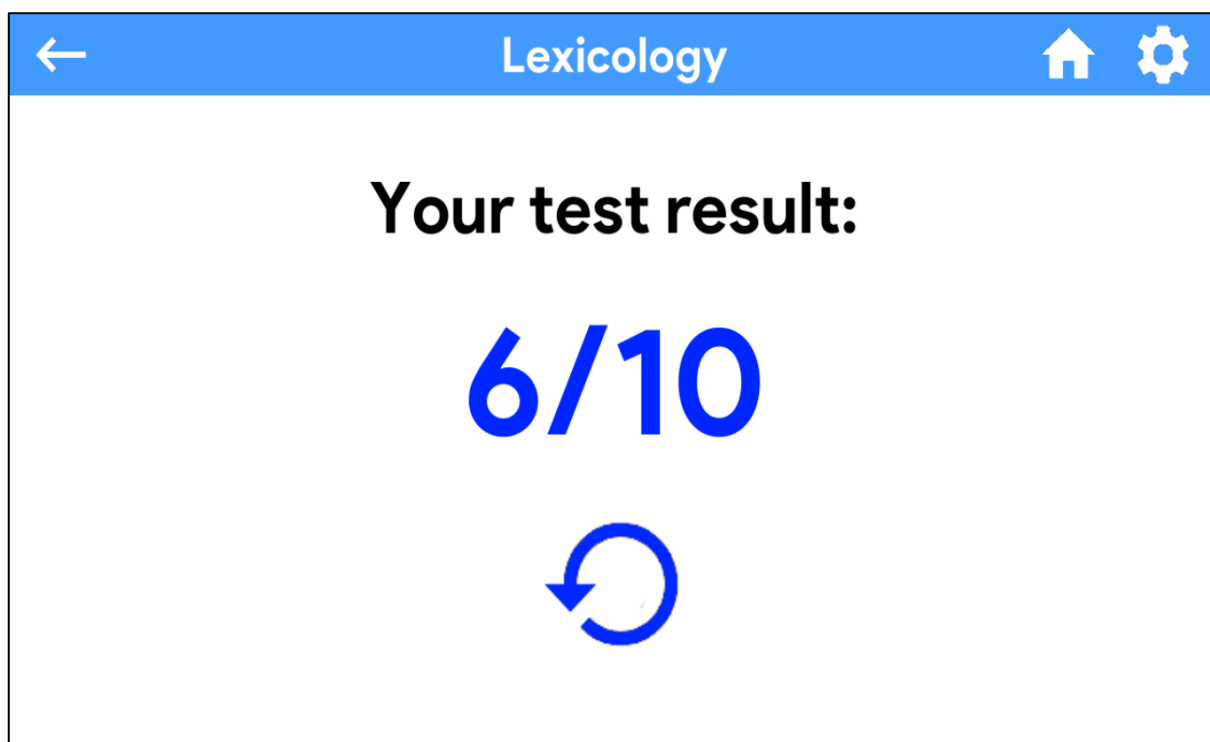


Рис. 23. Результат проверки знаний по лексикологии

На рис. 24 представлено меню настроек программы. Присутствует кнопка управления музыкой в программе («Music»), кнопка управления звуком в программе («Sound»), кнопка «о приложении» («About») и кнопка возврата к предыдущему меню («Back»).

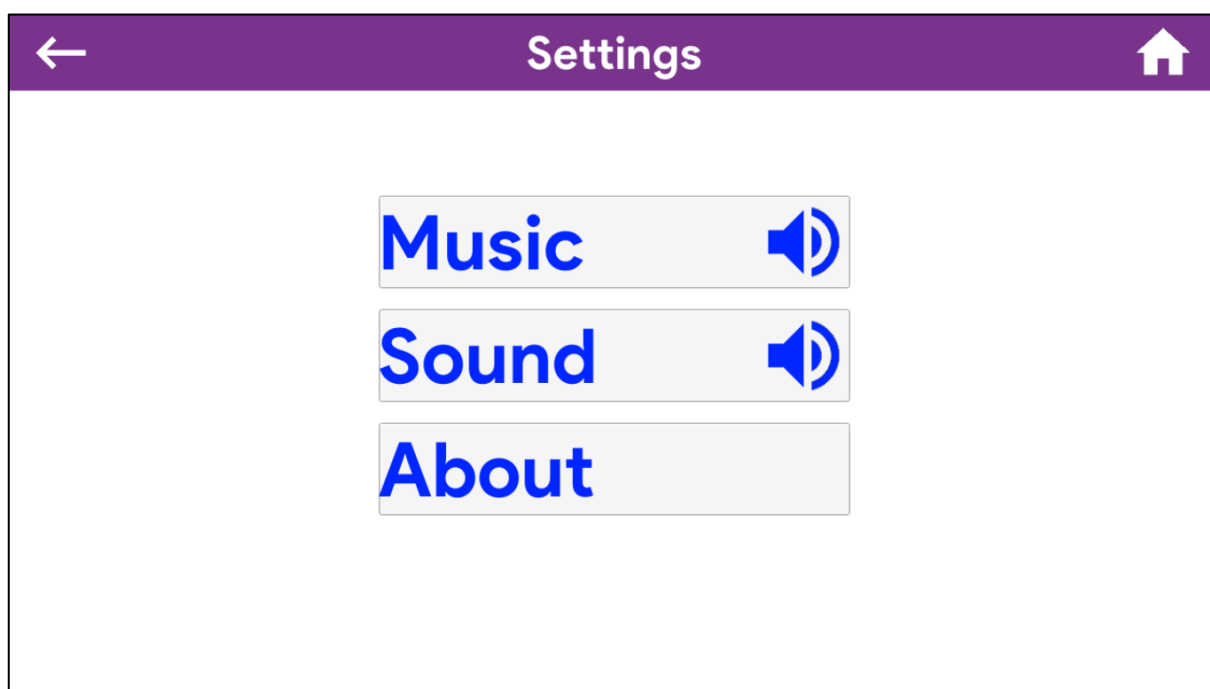


Рис. 24. Меню настроек программы

На рис. 25 представлено отображение информации о приложении. Присутствует блок представления информации о приложении, кнопка возврата в главное меню («Home»), и кнопка возврата к предыдущему меню («Back»).

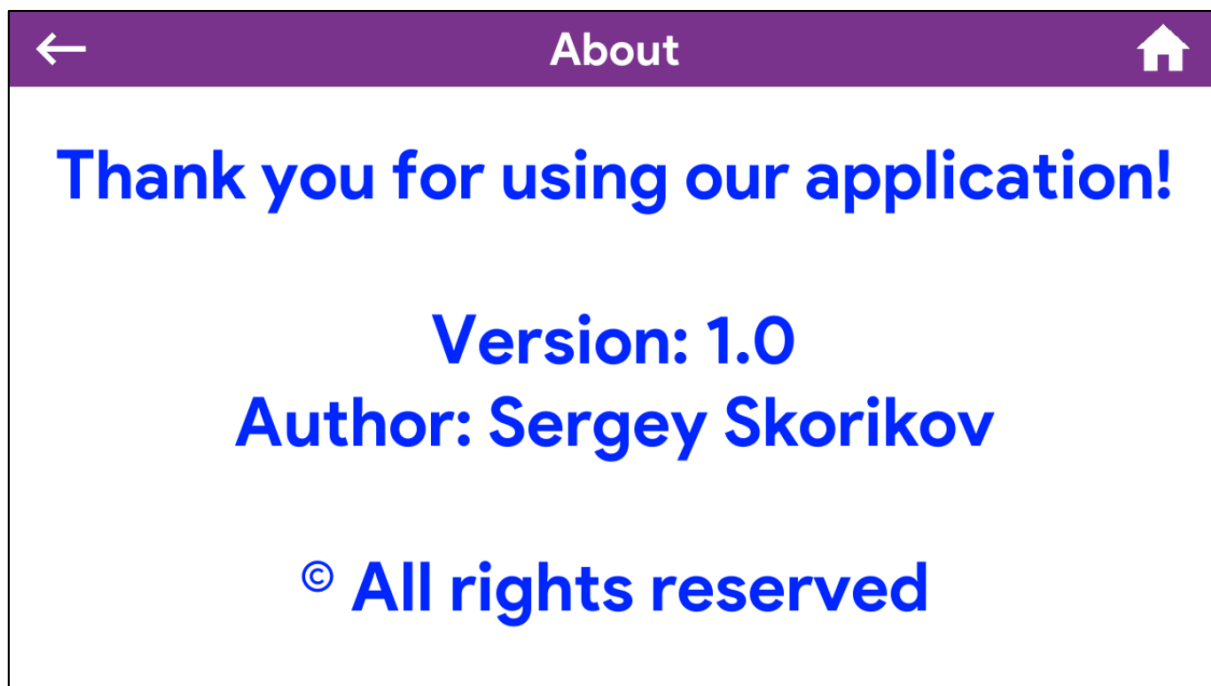


Рис. 25. Отображение информации о приложении

### **Вывод**

В главе 3 были использованы различные средства реализации, продемонстрирована структура программы, а также продемонстрирован интерфейс обучающей программы.

## 4. ТЕСТИРОВАНИЕ

### 4.1. Функциональное тестирование

Функциональное тестирование – это тестирование программного обеспечения в целях проверки реализуемости функциональных требований, то есть способности программного обеспечения в определенных условиях решать задачи, нужные пользователям. Набор тестов на функциональность представлен в табл. 1.

Табл. 1. Результаты функционального тестирования

№	Название теста	Входные данные	Ожидаемый результат	Тест пройден?
1	Отображение главного экрана	Нажать на кнопку «Lexicology»	Переход в раздел лексикологии	Да
2	Отображение главного экрана	Нажать на кнопку «Orthography»	Переход в раздел орфографии	Да
3	Отображение главного экрана	Нажать на кнопку «Morphology»	Переход в раздел морфологии	Да
4	Отображение главного экрана	Нажать на кнопку «Settings»	Переход в меню настроек	Да
5	Отображение всех разделов	Нажать на кнопку «Theory»	Переход в меню выбора темы изучения	Да
6	Отображение всех разделов	Нажать на кнопку «Exam»	Переход в раздел проверки знаний	Да
7	Отображение всех разделов	Нажать на кнопку «Settings»	Переход в меню настроек	Да
8	Отображение всех разделов	Нажать на кнопку «Back»	Переход в предыдущее меню	Да
9	Отображение всех разделов	Нажать на кнопку «Home»	Переход в главное меню	Да
10	Отображение меню выбора темы изучения всех разделов	Нажать на кнопку выбранной темы	Переход в соответствующую тему	Да

№	Название теста	Входные данные	Ожидаемый результат	Тест пройден?
11	Отображение меню выбора темы изучения всех разделов	Нажать на кнопку «Settings»	Переход в меню настроек	Да
12	Отображение меню выбора темы изучения всех разделов	Нажать на кнопку «Back»	Переход в предыдущее меню	Да
13	Отображение меню выбора темы изучения всех разделов	Нажать на кнопку «Home»	Переход в главное меню	Да
14	Отображение темы изучения всех разделов	Нажать на кнопку «Settings»	Переход в меню настроек	Да
15	Отображение темы изучения всех разделов	Нажать на кнопку «Back»	Переход в предыдущее меню	Да
16	Отображение темы изучения всех разделов	Нажать на кнопку «Home»	Переход в главное меню	Да
17	Отображение проверки знаний всех разделов	Нажать на вариант ответа	Переход к следующему вопросу	Да
18	Отображение проверки знаний всех разделов	Нажать на кнопку «Settings»	Переход в меню настроек	Да
19	Отображение проверки знаний всех разделов	Нажать на кнопку «Back»	Переход в предыдущее меню	Да
20	Отображение проверки знаний всех разделов	Нажать на кнопку «Home»	Переход в главное меню	Да

№	Название теста	Входные данные	Ожидаемый результат	Тест пройден?
21	Отображение результата проверки знаний всех разделов	Нажать на кнопку «Settings»	Переход в меню настроек	Да
22	Отображение результата проверки знаний всех разделов	Нажать на кнопку «Back»	Переход в предыдущее меню	Да
23	Отображение результата проверки знаний всех разделов	Нажать на кнопку «Home»	Переход в главное меню	Да
24	Отображение результата проверки знаний всех разделов	Нажать на кнопку «Restart»	Тест начинается заново	Да
25	Отображение меню настроек	Нажать на кнопку «Music»	Выключается/включается музыка в программе	Да
26	Отображение меню настроек	Нажать на кнопку «Sound»	Выключается/включается звук в программе	Да
27	Отображение меню настроек	Нажать на кнопку «About»	Переход к информации о приложении	Да
28	Отображение информации о приложении	Нажать на кнопку «Back»	Переход в предыдущее меню	Да
29	Отображение информации о приложении	Нажать на кнопку «Home»	Переход в главное меню	Да

### Вывод

В главе 4 было выполнено функциональное тестирование программы. Все тесты были выполнены успешно.



## **ЗАКЛЮЧЕНИЕ**

В ходе выполнения проделанной работы были решены следующие задачи:

- 1) произведен анализ предметной области;
- 2) спроектирована программа;
- 3) реализована программа;
- 4) протестирована программа.

## СПИСОК ЛИТЕРАТУРЫ

1. eLearning Guild. Mobile Learning: What it is, why it matters, and how to incorporate it into your learning strategy. [Electronic resource] URL: [http://www.cedma-europe.org/newsletter%20articles/eLearning%20Guild/360%20Report%20on%20Mobile%20Learning%20\(Jul%2008\).pdf](http://www.cedma-europe.org/newsletter%20articles/eLearning%20Guild/360%20Report%20on%20Mobile%20Learning%20(Jul%2008).pdf) (date of access: 21.05.2018).
2. Jarvis H., Achilleos M. From Computer Assisted Language Learning (CALL) to Mobile Assisted Language Use (MALU). // The Electronic Journal for English as a Second Language, 2013. – Vol. 16, – No. 4. – P. 1–12.
3. Official site Anki. [Electronic resource] URL: <https://apps.ankiweb.net/> (date of access: 21.05.2018).
4. Official site MoLeNET. [Electronic resource] URL: <https://www.talpalink.co.uk/molenet-iamlearn/> (date of access: 21.05.2018).
5. Traxler J. Current State of Mobile Learning. // Mobile Learning: Transforming the Delivery of Education and Training, 2009. – P. 9–24.
6. Unity Scripting API. [Electronic resource] URL: <https://docs.unity3d.com/ScriptReference/> (date of access: 21.05.2018).
7. Unity User Manual. [Electronic resource] URL: <https://docs.unity3d.com/Manual/index.html/> (date of access: 21.05.2018).
8. Visual Studio documentation. [Electronic resource] URL: <https://docs.microsoft.com/en-us/visualstudio/ide/> (date of access: 21.05.2018).
9. Актуальность изучения иностранных языков в социально-экономических условиях современной России. [Электронный ресурс] URL: <http://www.ibl.ru/konf/140509/61.html/> (дата обращения: 21.05.2018).
10. Варшавская С.В. Использование мобильных справочных приложений для оптимизации процесса обучения английскому языку. // Новые технологии в обучении иностранным языкам: Материалы научно-практической конференции. – Омск: Омский гос. ун-т им. Ф.М. Достоевского, 2015. – С. 97–103.

11. Деревянко Е.Ю., Бельченко В.Е. Сравнение операционных систем Windows 7/8 и Linux. // Инновационные технологии в науке и образовании, 2015. – № 3. – С. 195-196.
12. Зенина Л.В., Каменева Н.А. Возможности современных компьютерных программ в процессе обучения иностранным языкам в вузе. // Вестник Московского государственного гуманитарного университета им. М.А. Шолохова. – Педагогика и психология, 2013. – № 3 – С. 57–60.
13. Официальный сайт Duolingo. [Электронный ресурс] URL: <https://www.ru.duolingo.com/> (дата обращения: 21.05.2018).
14. Официальный сайт Lingualeo. [Электронный ресурс] URL: <https://www.lingualeo.com/ru/> (дата обращения: 21.05.2018).
15. Фаулер М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования. – СПб.: Символ-Плюс, 2018. – 192 с.
16. Хейлмберг А. Язык программирования C#. / Хейлмберг А., Торгерсен М., Вилтамут С., Голд П. – СПб.: Питер, 2012. – 784 с.