

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Образовательная программа «Программная инженерия»

СОГЛАСОВАНО
Научный руководитель, доцент
департамента «Программной инженерии»,
кандидат технических наук

УТВЕРЖДАЮ
Академический руководитель
образовательной программы
«Программная инженерия», кандидат
технических наук

_____ Н.С. Белова
«___» _____ 2023 г.

_____ В.В. Шилов
«___» _____ 2023 г.

**«Мобильное приложение для онлайн соревнования
в решении кроссвордов»**

Пояснительная записка

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.05.15-01 81 01-1-ЛУ

Исполнитель:
студент группы БПИ219

_____ / Фирсов Ф.А. /
«___» _____ 2023 г.

Москва 2023

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	RU.17701729.05.15-01 81 01-1

УТВЕРЖДЕНО
RU.17701729.05.15-01 81 01-1-ЛУ

**«Мобильное приложение для онлайн соревнования
в решении кроссвордов»**

**Пояснительная записка
RU.17701729.05.15-01 81 01-1**

Листов 33

Инв. № подл	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
RU.17701729.05.15-01 81 01-1				

Москва 2023

СОДЕРЖАНИЕ

АННОТАЦИЯ	4
ВВЕДЕНИЕ	5
1.1. НАИМЕНОВАНИЕ ПРОГРАММЫ	5
1.2. ДОКУМЕНТ, НА ОСНОВАНИИ КОТОРОГО ВЕДЕТСЯ РАЗРАБОТКА	5
2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ	6
2.1. НАЗНАЧЕНИЕ ПРОГРАММЫ	6
2.1.1. <i>Функциональное назначение</i>	6
2.1.2. <i>Эксплуатационное назначение</i>	6
2.2. КРАТКАЯ ХАРАКТЕРИСТИКА ОБЛАСТИ ПРИМЕНЕНИЯ	6
3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	7
3.1. ПОСТАНОВКА ЗАДАЧИ НА РАЗРАБОТКУ ПРОГРАММЫ	7
3.2. ОПИСАНИЕ И ОБОСНОВАНИЕ АРХИТЕКТУРЫ ПРОГРАММЫ	8
3.2.1. <i>Описание архитектуры программы</i>	8
3.2.2. <i>Обоснование архитектуры программы</i>	12
3.3. ОПИСАНИЕ АЛГОРИТМА РАБОТЫ ПРОГРАММЫ	13
3.4. ОПИСАНИЕ ВЫБОРА МЕТОДА ОРГАНИЗАЦИИ ВХОДНЫХ И ВЫХОДНЫХ ДАННЫХ	15
3.5. ОПИСАНИЕ И ОБОСНОВАНИЕ ВЫБОРА СОСТАВА ТЕХНИЧЕСКИХ И ПРОГРАММНЫХ СРЕДСТВ	15
3.5.1. <i>Состав технических и программных средств</i>	15
3.5.2. <i>Обоснование выбора технических и программных средств</i>	16
4. ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ	17
4.1. ОРИЕНТИРОВОЧНАЯ ЭКОНОМИЧЕСКАЯ ЭФФЕКТИВНОСТЬ	17
4.2. ПРЕДПОЛАГАЕМАЯ ПОТРЕБНОСТЬ	17
4.3. ЭКОНОМИЧЕСКИЕ ПРЕИМУЩЕСТВА РАЗРАБОТКИ ПО СРАВНЕНИЮ С ОТЕЧЕСТВЕННЫМИ И ЗАРУБЕЖНЫМИ ОБРАЗЦАМИ ИЛИ АНАЛОГАМИ	17
5. ИСТОЧНИКИ, ИСПОЛЬЗОВАННЫЕ ПРИ РАЗРАБОТКЕ	19
ПРИЛОЖЕНИЕ 1	20
ПРИЛОЖЕНИЕ 2	21
ПРИЛОЖЕНИЕ 3	22
ПРИЛОЖЕНИЕ 4	23
ПРИЛОЖЕНИЕ 5	32
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	33

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

АННОТАЦИЯ

Настоящий программный документ представляет собой пояснительную записку к программному проекту «Мобильное приложение для онлайн соревнования в решении кроссвордов».

Раздел «Введение» включает в себя наименование программы и документ, на основании которого ведётся разработка, с указанием организации, утвердившей данный документ.

В разделе «Назначение и область применения» содержатся функциональное и эксплуатационное назначение программы и краткая характеристика области её применения.

В разделе «Технические характеристики» присутствуют следующие подразделы: постановка задачи на разработку программы, описание функционирования программы, описание и обоснование алгоритма работы программы, описание и обоснование математического алгоритма работы программы, описание выбора метода организации входных и выходных данных, описание работы с файловой системой, описание и обоснование выбора состава технических и программных средств.

В разделе «Ожидаемые технико-экономические показатели» указана предполагаемая потребность и экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами.

Программный документ разработан в соответствии с требованиями:

1. ГОСТ 19.101-77 Виды программ и программных документов [1];
2. ГОСТ 19.102-77 Стадии разработки [2];
3. ГОСТ 19.103-77 Обозначения программ и программных документов [3];
4. ГОСТ 19.104-78 Основные надписи [4];
5. ГОСТ 19.105-78 Общие требования к программным документам [5];
6. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом [6];
7. ГОСТ 19.404-79 Пояснительная записка. Требования к содержанию и оформлению [7].

Изменения к Пояснительной записке оформляются согласно ГОСТ 19.603-78 [8], ГОСТ 19.604-78 [9].

Перед прочтением данного документа следует ознакомиться с терминологией, приведенной в Приложении 1.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ВВЕДЕНИЕ

1.1. Наименование программы

Наименование темы разработки: «Мобильное приложение для онлайн соревнования в решении кроссвордов».

Наименование темы разработки на английском языке: «Mobile Application for Online Competitions in Solving Crosswords».

1.2. Документ, на основании которого ведется разработка

Основанием для разработки является учебный план подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденная академическим руководителем тема курсового проекта.

Документ утвержден организацией: Национальный исследовательский университет «Высшая школа экономики», Факультет компьютерных наук, образовательная программа «Программная инженерия».

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

2.1. Назначение программы

2.1.1. Функциональное назначение

Программа позволяет пользователям участвовать в соревнованиях по решению кроссворда на скорость с другими пользователями в различных игровых режимах, а также узнавать свой рейтинг исходя из проведенных рейтинговых игр. Имеет развлекательное назначение.

2.1.2. Эксплуатационное назначение

Предположительными конечными пользователями могут быть как дети школьного возраста, так и взрослые люди, увлеченные решением кроссвордов. Соревновательный эффект в игре должен способствовать расширению аудитории и поддержанию ее интереса к игре.

2.2. Краткая характеристика области применения

«Мобильное приложение для онлайн соревнования в решении кроссвордов» — это игровое приложение, которое позволит не только решать кроссворды в одиночку, но и соревноваться с другими игроками. Пользователи смогут сразиться как с друзьями, так и с равными по силе игроками в рейтинговой игре в борьбе за рейтинговые ранги, что делает игру более захватывающей.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

3.1. Постановка задачи на разработку программы

«Мобильное приложение для онлайн соревнования в решении кроссвордов» должно предоставлять возможность выполнения следующих задач:

1. Предоставление пользователю интерфейса для регистрации в приложении.
2. Предоставление пользователю интерфейса для входа в аккаунт в приложении.
3. Смену пароля или игрового имени в интерфейсе страницы настройки.
4. Предоставление интерфейса с данными о текущем ранге и очках повышения.
5. Доступ начать рейтинговую, обычную или тренировочную игру меню.
6. Свободное перемещение между страницами настройки, главным меню и статистикой.
7. Предоставления интерфейса, отображающего кроссворд – прямоугольное поле, где каждая клетка одного из типов: неактивная клетка (клетка-заглушка, не соответствует буквам и никак не участвует в игре), клетка с загаданной буквой (позволяет игроку ввести букву), клетка с верно отгаданной буквой (отображает букву, недоступна для редактирования). Изначально все клетки или неактивны, или с загаданной буквой. Клетки с загаданными буквами составляют слова по горизонтали или вертикали, которые необходимо отгадать игроку и его сопернику. После того, как игрок, или его соперник отгадывает слово все клетки этого слова становятся клетками с отгаданной буквой.
8. Отображение загадки – текста, объясняющего выбранное слово.
9. Предоставления интерфейса с информацией о количестве отгаданных слов игроком и его соперником.
10. Возможность проверки слова, введенного пользователем в кроссворд.
11. После окончание игры предоставление информации о победе/поражении и опционально о изменении очков повышения и ранга игрока.
12. Отражения информации на странице статистика о:
 - а) Отображение общего количества сыгранных игр.
 - б) Отображение количества сыгранных рейтинговых игр.
 - в) Отображение количества сыгранных обычных игр.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

- d) Отображение общего количества побед.
- e) Отображение количества побед в рейтинговых играх.
- f) Отображение количества побед в обычных играх.
- g) Отображение общего процента побед.
- h) Отображение процента побед в рейтинговых играх.
- i) Отображение процента побед в обычных играх.
- j) Отображение общего количества отгаданных слов.

3.2. Описание и обоснование архитектуры программы

3.2.1. Описание архитектуры программы

«Мобильное приложение для онлайн соревнования в решении кроссвордов» является клиент-серверным мобильным приложением. Мобильное приложение (клиент) написано с использованием среды разработки Android Studio. Android Studio, основанная на программном обеспечении IntelliJ IDEA от компании JetBrains, является официальным инструментом для создания приложений на платформе Android с использованием встроенных библиотек этой платформы. Для написания кода клиента был выбран язык Java, так как он является одним из “классических” языков платформы Android, поддерживается Android Studio и является универсальным и комфортным инструментом. Клиентская часть отвечает за предоставления пользователю интерфейса, описанного выше.

Для создания серверной части был выбран язык Python и его встроенные библиотеки socket и thread. Серверная часть обеспечивает генерацию кроссвордов, ответы на запросы проверки слов, синхронизации игроков и прочую логику взаимодействия клиентов друг с другом и базой данных. Python был выбран из-за его удобства в использовании механизмов связывающих серверную часть с остальными модулями и многопоточной работы, а также лаконичности кода. Так как сервер не решает сложные математические или переборные задачи, а также не реализует сложные алгоритмы, то мощности Python будет достаточно.

Для создания базы данных была выбрана реляционная система управления базами данных MySQL. Эта СУБД была выбрана, так как является одной из самых распространённых баз данных, поддерживает работу с Python, на котором написана серверная часть, и показывает хорошую производительность и эффективность.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

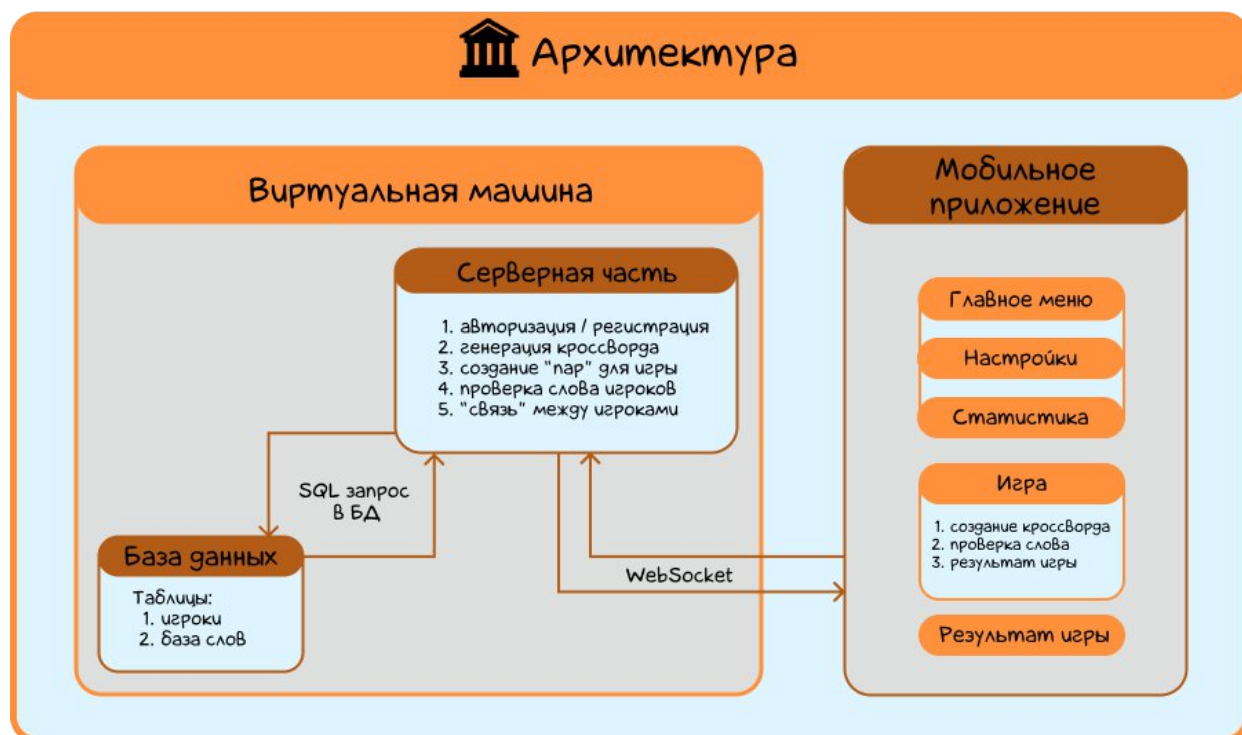


Рисунок 1. Общая архитектура.

Как и любое клиент-серверное приложение, «Мобильное приложение для онлайн соревнования в решении кроссвордов», состоит из двух основных частей расположенных на различных вычислительных устройствах. Первая часть – мобильное приложение, которое представляет удобный и приятный для пользователя интерфейс, а также обрабатывает действия пользователя и создает запросы в серверную часть. Вторая часть – серверное приложение, которое обеспечивает выполнение логики игры, такой как генерация кроссворда, проверка слова на правильность, подбор пар игроков для игры в соответствии с их игровым рейтингом и прочее, что будет подробно описано ниже. Также для хранения больших данных, таких как базы слов с загадками и базы игроков, серверному приложению необходима база данных.

Для связи сервера с клиентами был выбран механизм создания WebSocket так как он является современным стандартом сетевого взаимодействия. Кроме того работа с WebSocket удобно предусмотрена как в JAVA, на котором написано мобильное приложение, так и в Python на серверной части.

Вначале рассмотрим схему взаимодействия элементов системы. Взаимодействие между компонентами можно разбить на следующие фазы:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

1. Пользователь совершает действие в мобильном приложении, требующее запроса на сервер.
2. Мобильное приложение обрабатывает действие пользователя и создает объект запроса на сервер.
3. Мобильное записывает в WebSocket запрос и отправляет его серверу.
4. Сервер распознает тип запроса.
5. (Опционально в зависимости от типа запроса) Сервер подключается и создает запрос в базу данных.
6. Сервер генерирует ответ на запрос клиента и записывает результат в WebSocket. Отправляет назад клиенту.
7. Мобильное приложение обрабатывает результат и выводит результат действия пользователю.

Теперь давайте более подробно рассмотрим внутреннее устройство описанных компонентов.

- 1. Пользователь совершает действие в мобильном приложении, требующее запроса на сервер.**

Запросы на сервер мобильного приложения				
событие	действие	страница	тип запроса	название метода
вход в систему	Нажатии кнопки "войти"	страница входа	login	setOnClickListener
регистрация нового пользователя	Нажатии кнопки "регистрация"	страница регистрации	register	setOnClickListener
отмена поиска игры	Нажатии кнопки "отмена"	ожидание игры	cancel queue	setOnClickListener
проверка нахождения противника	регулярно	ожидание игры	check queue	check_queue
начать новую игру	Нажатии кнопки "Поехали!"	главная страница	add to queue	setOnClickListener
отгадал ли противник новые слова	регулярно	игра	game status	gameStatus
проверка слова	нажатие кнопки с галочкой	игра	check word	setOnClickListener

Рисунок 2. Таблица "Запросы на сервер мобильного приложения"

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

При желании совершить действие указанное в первом столбце таблицы на рисунке 2 пользователь совершает действие указанное в одноименном столбце. После чего запрос обрабатывается на этапе 2.

2. Мобильное приложение обрабатывает действие пользователя и создает объект запроса на сервер.

Действие пользователя для каждого события обрабатывается в методе, указанном в столбце “название метода” таблицы на рисунке. Для запроса создается JSON объект содержащий тип из таблицы и конкретные для каждого типа запроса данные.

3. Мобильное записывает в WebSocket запрос и отправляет его серверу.

Создание WebSocket и записи и *public static JSONObject executePost(JSONObject json)* класса *LoginMenu*. В качестве единственного параметра принимается JSON объект, сгенерированный на стадии 2.

4. Сервер распознает тип запроса.

Обработка происходит в методе *request_to_client(data: str)* файла *Requester.py*. В качестве входного параметра принимается записанный в строку JSON объект. По полю *type* определяется тип входящего запроса

5. Сервер подключается и создает запрос в базу данных.

Если для получения результата запроса нужно обратиться в базу данных вызывается один из методов *select_from_db*, *insert_to_db* из файла *DbWork.py* для получения или вставки данных.

6. Сервер генерирует ответ на запрос клиента и записывает результат в WebSocket. Отправляет назад клиенту.

В методе *request_to_client(data: str)* из файла *Requester.py* генерируется ответ в виде JSON объекта с полем *status* соответствующим значению коду состояния HTTP и конкретной для каждого запроса информацией ответа. Метод *client_thread* записывает его в WebSocket и отправляет клиенту.

7. Мобильное приложение обрабатывает результат и выводит результат действия пользователю.

Обработка происходит в том же методе, где инициализирован запрос.

Так как все запросы действуют аналогично по этой схеме то подробно разберем работу только проверки статуса игры, как самую интересную.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

При нажатии кнопки «Поехали» в главном меню игрок перемещается на страницу ожидания соперника. С программной стороны это выглядит так:

В мобильном приложении создается Json Object содержащий информацию о запросе, в нашем случае тип запроса - «game status», что видно в таблице. Также для проверки нахождения соперника понадобится идентификатор пользователя, чтобы сервер мог определить кто ожидает соперника. Этот идентификатор передается в поле «uid», как целое число. Больше дополнительной информации не требуется, а значит этот объект готов к отправки на сервер. Мобильное приложение записывает Json Object преобразованный в строку и отправляет его серверу. Теперь мобильное приложение ожидает когда в WebSocket появится ответ сервера и передаст информацию о кроссворде и противнике в GameActivity отвечающее за процесс игры.

Теперь рассмотрим происходящее на сервере. Получив запрос сервер пытается декодировать его из строки в Json Object. У нас корректный запрос с правильным типом «game status», а значит серверу это удастся и он передает запрос в метод game_status(request) отвечающий за ответ на запросы такого типа. Запрос смотрит в данных очередей найден ли противник и в случае, если противник найден генерирует кроссворд. Для защиты от нечестных игроков, которые могут попытаться изменить файлы мобильного приложения, мы не будем передавать сами слова клиенту, а передадим только поле кроссворда, и необходимую информацию о словах - загадки, координаты начала слова, ориентацию и длину. Все это запаковывается в формат json и возвращается клиенту вместе с кодом успеха - 200.

3.2.2. Обоснование архитектуры программы

При разработке «Мобильное приложение для онлайн соревнования в решении кроссвордов» было принято несколько важных архитектурных решений. Объясним каждое из них:

1. Разделение на клиент (мобильное приложение) и сервер. В первую очередь «Мобильное приложение для онлайн соревнования в решении кроссвордов» представляет собой мобильное приложение для платформы Android, поэтому было необходимо выбрать платформу, которая предоставляет возможности по разработке графического приложения. Для этих целей было выбрано Android Studio как рекомендованное Google, создателем платформы Android. На ее основе довольно просто написать необходимые интерфейсы, а также прописать логику и работу со сторонними системами. Общение клиентов друг с другом должен кто-то «регулировать», поэтому появилась необходимость в серверном приложении. Кроме того, генерация кроссворда обязывает к хранению базы данных содержащие все слова,

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

а хранить ее на телефоне не желательно, как в целях безопасности, так и не эффективна. Так как сервер не реализует приложение и минимально контактирует с операционной системой, то жесткой привязки к JAVA и Android Studio уже нет. Наиболее подходящим для задач сервера был выбран язык Python, поскольку вместе с ним можно использовать удобные библиотеки для работы с базой данных и WebSocket и его код лаконичен.

2. В процессе написания серверной обработки запросов я использовал принципы "слоевой" архитектуры. Это способствовало правильной декомпозиции кода, что в свою очередь улучшило читаемость, облегчило тестирование и обнаружение ошибок. Кроме того, такая архитектура обеспечивает возможность добавления новой логики в будущем (если это будет необходимо), делая код более гибким и адаптируемым к изменениям.

3.3. Описание алгоритма работы программы

Разберем алгоритм работы «Мобильного приложения для онлайн соревнования в решении кроссвордов» по шагам:

1. На первом этапе пользователь запускает приложение системы, в результате чего загружается страница входа;
2. Пользователь переходит на страницу регистрации с помощью соответствующей кнопки или переходит к пункту 4;
3. Пользователь создает новый аккаунт указав логин, пароль и игровое имя.
4. Пользователь входит в приложение используя указанный при регистрации логин и пароль.
5. Пользователь может перемещаться между страницами статистика, главное меню и настройки.
6. С помощью кнопок влево и вправо в главном меню можно изменить режим игры.
7. Пользователь нажимает кнопку «поехали!» в главном меню и перемещается на страницу ожидания игры.
8. Ожидание подбора соперника и генерации кроссворда.
9. Пользователь способен переключаться между клетками кроссворда нажатием на клетку. Повторное нажатие меняет ориентацию выбранного участка.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

10. С помощью клавиш пользователь вводит слово отвечающие загадке в выбранные клетки.
11. Пользователь проверяет верность слова при нажатии кнопки с галочкой. Если слово отгадано верно оно блокируется, показывается сопернику, а игрок получает победное очко.
12. Когда все слова отгаданы пользователь перемещается на страницу результатов.
13. Если у игрока очков больше, чем у соперника, то игра выиграна. В противном случае - проигрыш.
14. В случае выигрыша в рейтинговой игре пользователь повышает свой ранг.

Подробнее рассмотрим пункт 14 этого списка. Система изменения ранга устроена следующим образом.

Ранг игрока складывается из 3 частей – категории ранга, подкатегории ранга и ранговых очков. Категория ранга обозначается одним из цветов радуги: красным, оранжевым, желтым, зеленым, голубым, синим или фиолетовым. Категории ранга следуют друг за другом именно в этом порядке, то есть при создании аккаунта игрок получает красную категорию и со временем может повысить ее до оранжевой, потом желтой и так далее. Стоит заметить что противник в рейтинговой игре как раз подбирается по категории ранга, то есть «желтые» игроки могут играть только с «желтыми», а красные с «красным». Это максимально поддерживает комфорт и соревновательный азарт игрока, так как он понимает, что противника победить ему по силам, но это не тривиальная задача. Также это защищает «новичков» в игре, которые еще только начинают разгадывать кроссворды от игроков с максимальным рангом, которые не оставили бы шанса новичку.

Каждая категория делится на 3 подкатегории - «С», «В» и «А». Они также линейно следуют друг за другом в порядке обратном алфавитному. Если перед повышением подкатегории, она была равна «А», то подкатегория становится равной «С», а категория игрока увеличивается на 1, например с ранга «Зеленый-А» игрок повышается до ранга «Голубой-С». Пользователь так же может их только «повысить», а потерять уже полученную подкатегорию невозможно. Это поддерживает интерес пользователя к игре в моменты неудач – пользователь видит, что он прогрессирует и оттачивая свои навыки сможет добиться любого ранга.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Для определения повышения подкатегории используются рейтинговые очки. Это счетчик, который принимает одно из 3 целочисленных значений от 0, до 2. При победе в рейтинговой игре значения счетчика увеличивается на 1, при проигрыше уменьшается. При проигрыше с нулевым значением счетчика ничего не меняется, а при победе с значением 2, счетчик обнуляется и происходит повышение ранга. Такая система позволяет удерживать интерес к игре в моменте, то есть начиная проигрывать игрок будет стараться вернуть победу, а не побыстрее закончить игру, чтобы найти противника полегче. Глобально же пользователь не может потерять ранг, а значит будет прогрессировать в ранговой системе.

3.4. Описание выбора метода организации входных и выходных данных

Мобильное и серверное приложение не имеет входных и выходных данных. Все данные хранятся в базе данных. Для корректной работы программы необходима таблица words в датасете WordDuel на текущем MySQL сервере. Таблица состоит из трех столбцов: word_id – уникальное целое число, word – слово(строка), definition – загадка(строка)

3.5. Описание и обоснование выбора состава технических и программных средств

3.5.1. Состав технических и программных средств

Для нормального функционирования мобильного приложения требуется смартфон, оснащенный следующими техническими компонентами:

1. 100 Мбайт свободного пространства на жестком диске;
2. 256 Мбайт оперативной или видео памяти или более;
3. Доступ в локальную или глобальную сеть с запущенным серверным приложением.

Для работы мобильного приложения необходим следующий состав программных средств:

1. ОС Android 3.8 или больше;

Для нормального серверного приложения требуется компьютер/сервер оснащенный пропорционально нагрузке на систему.

Для работы серверного приложения необходим следующий состав программных средств:

1. ОС Windows 7+ или Linux
2. Python 3 с библиотеками threading, mysql, socket

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

3.5.2. Обоснование выбора технических и программных средств

Программе необходимо как минимум 256 мб оперативной памяти, поскольку в иных случаях ресурсов компьютера может не быть недостаточно для корректного отображения приложения.

Для работы серверного приложения необходим компьютер с мощностями пропорциональными одновременному количеству активных пользователей, так как нагрузка, то есть количество одновременных запросов на сервер, растет прямопропорционально и линейно количеству активных пользователей приложения. При малом количестве пользователя функцию сервера может использовать любой домашний персональных компьютер с необходимым программным обеспечением, однако при пиковом количестве игроков сопоставимом с конкурентами понадобятся мощности нескольким серверных компьютеров.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

4. ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

4.1. Ориентировочная экономическая эффективность

В рамках данной работы расчет экономической эффективности не предусмотрен.

4.2. Предполагаемая потребность

«Мобильного приложения для онлайн соревнования в решении кроссвордов» может быть использовано исключительно в развлекательных целях. Использование приложения повышает кругозор, развивает память, а также способствует развитию навыка быстрого подбора слов, что улучшает речь человека.

4.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами

Сравнение «Мобильного приложения для онлайн соревнования в решении кроссвордов» с аналогами представлено в следующей таблице:

Сравнительная таблица					
Название	Кроссворды на русском	Эрудит онлайн	Words of Wonders	Clash royale	Приложение для онлайн соревнования в решении кроссвордов
Сайт	https://play.google.com/store/apps/details?id=com.kodex.kodex	https://play.google.com/store/apps/details?id=com.kodex.kodex	https://play.google.com/store/apps/details?id=com.kodex.kodex	https://play.google.com/store/apps/details?id=com.kodex.kodex	
Отгадывание слов	+	+	+	-	+
Разгадывание классических кроссвордов	+	-	-	-	+
Онлайн соревнование	-	+	+	+	+
Рейтинговая система	-	-	-	+	+
Система друзей	-	+	-	+	+
Изменение сложности игры	+	-	+	+	+
Кроссплатформенность	-	+	+	+	-

Основным преимуществом по сравнению с конкурентами является смешение идей и механик классических приложений с разгадыванием кроссвордов с идеями соревновательных онлайн игр. Из классических кроссвордов будут взяты основы геймплея – угадывание слов, размещенных на поле, при помощи данных описаний, а из соревновательных онлайн игр взяты идеи о противостоянии 2-х игроков в решении одного кроссворда, механики рейтинга и рангов, ограничение по времени, нетривиальный подбор игроков и другое.

1. Кроссворды на русском

Классический представитель мобильного приложения для решения кроссвордов с большим набором заготовленных заранее заданий.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

2. Эрудит онлайн

Мобильная версия словесной игры Эрудит построенной на составлении слов и выкладывании их на поле по принципу кроссворда. В отличие от классических кроссвордов позволяет соревноваться нескольким игрокам, однако уходит от отгадывания слов к составлению их из данных букв.

3. Words of Wonders

Во многом похожа на Эрудит, но все слова состоят из одних и тех же букв данных изначально и отличаются только длиной и порядком букв.

4. Clash Royale

Один из самых популярных примеров соревновательных онлайн игр на мобильные устройства. Геймплей построен на противостоянии примерно равных по силе игроков в дуэлях по результатам, которые меняется рейтинг, арена (аналог рангу) и открываются новые возможности. Является прекрасным примером реализации классических механик соревновательных онлайн игр таких как: соревновательные игры, рейтинг, система друзей, система кланов.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

5. ИСТОЧНИКИ, ИСПОЛЬЗОВАННЫЕ ПРИ РАЗРАБОТКЕ

1. ГОСТ 19.101-77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
2. ГОСТ 19.102-77 Стадии разработки. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
3. ГОСТ 19.103-77 Обозначения программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
4. ГОСТ 19.104-78 Основные надписи. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
5. ГОСТ 19.105-78 Общие требования к программным документам. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
6. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
7. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
8. ГОСТ 19.603-78 Общие правила внесения изменений. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
9. ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
10. ГОСТ 15150-69 Машины, приборы и другие технические изделия. Исполнения для различных климатических районов. Категории, условия эксплуатации, хранения и транспортирования в части воздействия климатических факторов внешней среды. – М.: Изд-во стандартов, 1997.
11. ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
12. John Torjo «Boost.Asio C++ Network Programming»
13. The Art of Computer Programming, Donald Ervin Knuth
 - a. Volume 1: Fundamental Algorithms. Third Edition (Reading, Massachusetts: Addison-Wesley, 1997)
 - b. Volume 2: Seminumerical Algorithms. Third Edition (Reading, Massachusetts: Addison-Wesley, 1997)
 - c. Volume 4B: Combinatorial Algorithms, Part 2 (Upper Saddle River, New Jersey: Addison-Wesley, 2023)
14. Документация по NumPy [Электронный ресурс] / NumPy community. Режим доступа: <https://numpy.org/>, свободный.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 1

ТЕРМИНОЛОГИЯ

Таблица 1

Термин	Определение
Соревнование, игра	Ограниченный по времени этап игры, когда два участника соревнования решают общий кроссворд на время.
Ранг игрока	Показатель мастерства игрока определяемый исходя из проведенных игроком рейтинговых игр. Состоит из ключевого слова и буквы, например «Зеленый А».
СУБД	Система управления базы данных
Кроссворд	Словесная игра состоящая из поля и набора загадок. На прямоугольном сетчатом поле слова разместили так, чтобы каждая буква находилась в отдельной клеточке, но некоторые клетки входили одновременно в несколько слов. То есть слова могут пересекаться одинаковыми буквами. После этого для каждого слова придумали загадку, и указав для каждого слова его начало начальную клетку, ориентацию и загадку сами буквы стерли. Цель игрока правильно отгадать все слова и записать буквы в клетки поля. Так как некоторые буквы входят одновременно в несколько слов они могут быть помощниками игроку для отгадывания следующий слов.
Загадка	Некоторый текст который должен помочь пользователю отгадать слово в кроссворде. Это может быть синоним, описание, какой-то факт о слове, пословица с пропущенным словом или что либо другое помогающее отгадать слово.
Противник, Соперник	Другой пользователь приложения участвующий в том же соревновании, что и игрок.
Выделенное слово	При выборе клетки игроком (при нажатии на нее) выбирается и слово в которое входит эта буква. Для этого слова показывается загадка и именно его пытаются разгадать игрок.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 2

**ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КОМПОНЕНТОВ
ИНФРАСТРУКТУРЫ СИСТЕМЫ**

Описание и функциональное назначение компонентов инфраструктуры системы.

Таблица 2

Название компонента	Тип	Назначение
Мобильное приложение WordDuel	Мобильное приложение	Основная часть мобильного приложения для онлайн соревнования в решении кроссвордов
Серверное приложение	Скрипты	Запускает серверную часть программы

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 3

**ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ
МОБИЛЬНОГО ПРИЛОЖЕНИЯ**

Описание и функциональное назначение классов.

Таблица 3

Класс	Назначение
Word	Класс, описывающий слово, как часть кроссворда.
Crossword	Класс, описывающий кроссворд. Хранит информацию о позициях всех слов, о текущем статусе (отгадана/не отгадана/пустая) всех клеток.
LoginMenu	Класс, обрабатывающий логику поведения и данные, которые использует окна регистрации.
RegisterMenu	Класс, обрабатывающий логику поведения и данные, которые использует окно регистрации.
StatisticMenu	Класс, обрабатывающий логику поведения и данные, которые использует окно статистики.
WaitGameMenu	Класс, обрабатывающий логику поведения и данные, которые использует окно ожидание игры.
ResultActivityMenu	Класс, обрабатывающий логику поведения и данные, которые использует окно результатов игры.
MainActivity	Класс, обрабатывающий логику поведения и данные, которые использует главного меню.
SettingsMenu	Класс, обрабатывающий логику поведения и данные, которые использует окно настройки.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 4

ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ, МЕТОДОВ И СВОЙСТВ КЛАССОВ WPF-ПРИЛОЖЕНИЯ

Таблица 4.1

Описание полей, методов и свойств класса Word				
Поля				
Наименование	Модификатор доступа	Тип аргумента	Назначение	
map	private	int[][][]	Информация о клетках поля кроссворда.	
words	private	Word[]	Информация о словах загаданных в кроссворде.	
crossword_id	public	int	Уникальный идентификатор кроссворда.	
wordid_to_id	public	JSONObject	Словарь преобразования идентификатора слова в базе данных в локальный идентификатор.	
Методы				
Наименование	Модификатор доступа	Тип	Аргументы	Назначение
getStatus	public	int	int i, int j, boolean d	Получения статуса клетки по координатам в зависимости от ориентации.
getStatus	public	int	int i, int j	Получения статуса клетки по координатам.
getHeight	public	int	-	Получения ширины кроссворда.
getWidth	public	int	-	Получения высоты кроссворда.
getOrientation	public	boolean	int id	Получения ориентации слова по его локальному идентификатору.
getSize	public	int	int id	Получения длины слова по его локальному идентификатору.
getDefinition	public	String	int id	Получения загадки слова по его локальному идентификатору.
getWordId	public	String	int id	Получения глобального идентификатора слова по его локальному идентификатору.
getI	public	int	int id	Получения первой координаты слова по его локальному идентификатору.
getJ	public	int	int id	Получения второй координаты слова по его локальному идентификатору.
setGood	public	int	int id	Устанавливает клутку в статус отгаданной.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Таблица 4.2

Описание полей, методов и свойств класса GameActivity				
Свойства				
Наименование	Модификатор доступа	Тип	Назначение	
selectedI	private	int	Первая координата выбранной клетки	
selectedJ	private	int	Вторая координата выбранной клетки	
selectedId	private	int	Идентификатор слова, в который входит выбранная клетка	
selectedOrientation	private	boolean	Ориентация слова, в который входит выбранная клетка	
crossword	private	Crossword	Информация о кроссворде	
mCurrentVideoPosition	private	int	Текущий момент видео, отображаемого на фоне.	
mMediaPlayer	private	MediaPlayer	Плеер видео на заднем фоне.	
background	private	VideoView	Видео на заднем фоне.	
your_score	public	int	Количество отгаданных противником слов.	
enemy_score	public	int	Количество отгаданных игроком слов.	
crossword_id	private	int	Идентификатора текущего кроссворда.	
width	private	int	Ширина клетки кроссворда.	
used	private	Set<String>	Множество отгаданных слов.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна.
makeGood	private	void	int word_id	Установка слова в статус, отгаданного с блокировкой и перекраской клеток.
setWord	private	void	int word_id	Запись слова в кроссворд, если оно отгадано игроком.
createCrossword	public	void	-	Отрисовка кроссворда.
unmarkWord	private	void	-	Отмена выделения текущего слова.
markWord	private	void	int i, int j, boolean orient	Выделить соответствующее слово.
createKeyboard	private	void	-	Отрисовка клавиатуры.
addSpace	private	void	LinearLayout keyboard	Добавление пустого места в клавиатуру.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

addLine	private	void	LinearLayout ut linearLayout, List<String > letters, int height, int width, int k	Добавление строки в клавиатуру.
selectLetter	private	void	int i, int j, boolean orient	Выбор буквы в кроссворде.
findFirstWord	private	void	-	Поиск не отгаданного слова.
onPause	protected	void	-	Остановка видео при закрытии окна.
onResume	protected	void	-	Продолжение видео при открытии окна.
onDestroy	protected	void	-	Завершение видео при закрытии окна.
gameStatus	private	void	-	Синхронизация с соперником отгаданных слов.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Таблица 4.3

Описание полей, методов и свойств класса LoginMenu

Свойства			
Наименование	Модификатор доступа	Тип	Назначение
mCurrentVideoPosition	private	int	Текущий момент видео, отображаемого на фоне.
background	private	VideoView	Видео на заднем фоне.
mMediaPlayer	private	MediaPlayer	Плеер видео на заднем фоне.

Методы

Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна.
onPause	protected	void	-	Остановка видео при закрытии окна.
onResume	protected	void	-	Продолжение видео при открытии окна.
onDestroy	protected	void	-	Завершение видео при закрытии окна.
executePost	public	JSONObject	JSONObject json	Создание запроса на сервер.
showRegistor	private	void	-	Открыть меню регистрации

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Таблица 4.4

Описание полей, методов и свойств класса RegisterMenu

Свойства			
Наименование	Модификатор доступа	Тип	Назначение
mCurrentVideoPosition	private	int	Текущий момент видео, отображаемого на фоне.
background	private	VideoView	Видео на заднем фоне.
mMediaPlayer	private	MediaPlayer	Плеер видео на заднем фоне.

Продолжение таблицы 4.3

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна.
onPause	protected	void	-	Остановка видео при закрытии окна.
onResume	protected	void	-	Продолжение видео при открытии окна.
onDestroy	protected	void	-	Завершение видео при закрытии окна.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Таблица 4.5

Описание полей, методов и свойств класса ResultActivityMenu			
Свойства			
Наименование	Модификатор доступа	Тип	Назначение
mCurrentVideoPosition	private	int	Текущий момент видео, отображаемого на фоне.
background	private	VideoView	Видео на заднем фоне.
mMediaPlayer	private	MediaPlayer	Плеер видео на заднем фоне.

Продолжение таблицы 4.3

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна.
onPause	protected	void	-	Остановка видео при закрытии окна.
onResume	protected	void	-	Продолжение видео при открытии окна.
onDestroy	protected	void	-	Завершение видео при закрытии окна.

Таблица 4.6

Описание полей, методов и свойств класса SettingsMenu			
Свойства			
Наименование	Модификатор доступа	Тип	Назначение
mCurrentVideoPosition	private	int	Текущий момент видео, отображаемого на фоне.
background	private	VideoView	Видео на заднем фоне.
mMediaPlayer	private	MediaPlayer	Плеер видео на заднем фоне.

Продолжение таблицы 4.3

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна.
onPause	protected	void	-	Остановка видео при закрытии окна.
onResume	protected	void	-	Продолжение видео при открытии окна.
onDestroy	protected	void	-	Завершение видео при закрытии окна.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Таблица 4.7

Описание полей, методов и свойств класса StatisticMenu			
Свойства			
Наименование	Модификатор доступа	Тип	Назначение
mCurrentVideoPosition	private	int	Текущий момент видео, отображаемого на фоне.
background	private	VideoView	Видео на заднем фоне.
mMediaPlayer	private	MediaPlayer	Плеер видео на заднем фоне.
count_games	public	int	Количество сыгранных игр.
games_wins	public	int	Количество выигранных игр.
words_wins	public	int	Количество отгаданных слов.
rangs_games	public	int	Количество сыгранных игр в рейтинговых матчах.
rangs_wins	public	int	Количество выигранных игр в рейтинговых матчах.
series_rang	public	int	Лучшая серия побед в рейтинговых матчах.
series_rang_cur	public	int	Текущая серия побед в рейтинговых матчах.
series_comm	public	int	Лучшая серия побед в обычных матчах.
series_comm_cur	public	int	Текущая серия побед в обычных матчах.

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна.
onPause	protected	void	-	Остановка видео при закрытии окна.
onResume	protected	void	-	Продолжение видео при открытии окна.
onDestroy	protected	void	-	Завершение видео при закрытии окна.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Таблица 4.8

Описание полей, методов и свойств класса WaitGameMenu			
Свойства			
Наименование	Модификатор доступа	Тип	Назначение
mCurrentVideoPosition	private	int	Текущий момент видео, отображаемого на фоне.
background	private	VideoView	Видео на заднем фоне.
mMediaPlayer	private	MediaPlayer	Плеер видео на заднем фоне.

Продолжение таблицы 4.3

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
onCreate	protected	void	Bundle savedInstanceState	Создание окна.
onPause	protected	void	-	Остановка видео при закрытии окна.
onResume	protected	void	-	Продолжение видео при открытии окна.
onDestroy	protected	void	-	Завершение видео при закрытии окна.

Таблица 4.9

Описание полей, методов и свойств класса Word			
Свойства			
Наименование	Модификатор доступа	Тип	Назначение
i	public	int	Первая координата первой буквы.
j	public	int	Вторая координата первой буквы.
size	public	int	Длина слова
orientation	public	boolean	Ориентация слова
definition	public	String	Загадка для кроссворда.
word_id	public	String	Глобальный идентификатор слова.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 5

**ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ, МЕТОДОВ
СЕРВЕРНОГО ПРИЛОЖЕНИЯ**

Таблица 5.1

Описание полей и методов класса CrosswordGenerator			
Поля			
Наименование			Назначение
word_grid			Поле кроссворда
words			Список слов
character_comparison_grid			Таблица сравнения символов
word_location_info			Информация о положении слов в кроссворде
Методы			
Наименование	Статический	Аргументы	Назначение
find_comparisons	-		Поиск колличества пересечений в кроссворде
compare_words	+	word1: str, word2: str	Возвращает строку из всех символов, которые есть в обоих словах.
create_grid	-	-	Создает поле кроссворда
place_words	-	selected: str, orientation: bool, starting_position : tuple	Размещает выбранное слово в выбранной позиции
test_placement	-	intersection: list, orientation: bool, possible: list, word: str	Тестовое размещение слов в кроссворде для оценки количества пересечений.
test_collision	-	possible_intersections: list, intersection: list, word: str, orientation:bool	Тестовое размещение слова в кроссворде для оценки количества пересечений.
find_intersections	-	key: int, count: list	Поиск пересечений.
empty_column_left	-	-	Проверка на возможность удалить левый столбец.
empty_column_right	-	-	Проверка на возможность удалить правый столбец.
clear_left_column	-	-	Удалить левый столбец.
clear_right_column	-	-	Удалить правый столбец.
get_crossword	-	-	Возвращает информацию о кроссворде.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Таблица 5.2

Описание полей и методов скрипта Requester			
Поля			
Наименование		Назначение	
games		Текущие игры	
crossword_id		Номер следующей создаваемой игры	
queue		Очереди игроков	
users_status		Текущий статус игроков	
queue_semaphore		Семафор для синхронизации работы потоков с очередью	
Методы			
Наименование	Статический	Аргументы	Назначение
generate_crossword	-	-	Генерация нового кроссворда.
login	-	request: dict	Запрос входа в аккаунт.
register	-	request: dict	Запрос регистрации аккаунта.
check_word	-	request: dict	Проверка верности слова.
add_to_queue	-	request: dict	Добавление игрока в очередь ожидания игры.
check_queue	-	request: dict	Проверка очереди ожидания игры на нахождение соперника.
game_status	-	request: dict	Проверка на отгадывание противником новых слов. (Синхронизация с противником)
cansel_queue	-	request: dict	Отмена поиска игры.
request_to_client	-	data: str	Ответ на запрос клиента: получение запроса и перенаправление запроса в соответствующий метод.
update_results	-	request: dict	Обновление дынных после завершения игры.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

[illegible]

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.05.15-01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата