|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра информационных технологий в атомной энергетике

**ОТЧЕТ ПО ПРОЕКТУ**

по дисциплине «Разработка приложений на языке Котлин»

**Студенты группы** ИКБО-50-23 Петруничев А.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись студента)

**Руководитель проектной работы** Золотухин С.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Работа представлена «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_ 2025 г.

Допущен к работе «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2025 г.

Москва 2025

**ОГЛАВЛЕНИЕ**

[ВВЕДЕНИЕ 5](#_Toc183740821)

[1. ПЛАНИРОВАНИЕ И ПОДГОТОВКА 6](#_Toc183740822)

[1.1 Техническое задание 6](#_Toc183740823)

[1.1.1 Общие сведения 6](#_Toc183740824)

[1.1.2 Цели и назначение автоматизированной системы 6](#_Toc183740825)

[1.1.3 Характеристика объектов автоматизации 6](#_Toc183740826)

[1.1.4 Требования к системе 6](#_Toc183740827)

[1.1.5 Состав и содержание работ 6](#_Toc183740828)

[1.1.6 Порядок разработки и приемки 6](#_Toc183740829)

[1.1.7 Требования к подготовке объекта к вводу системы в действие 6](#_Toc183740830)

[1.1.8 Требования к документированию 6](#_Toc183740831)

[1.1.9 Источники разработки 7](#_Toc183740832)

[1.2 Диаграмма вариантов использования 7](#_Toc183740833)

[1.2.1 Обоснование сценариев использования 7](#_Toc183740834)

[1.2.2 Схематическое представление (Use Case Diagram) 7](#_Toc183740835)

[1.3 Создание репозитория 8](#_Toc183740836)

[2. ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА 8](#_Toc183740837)

[2.1 Макеты экранов 8](#_Toc183740838)

[3. РАЗРАБОТКА ПРИЛОЖЕНИЯ 9](#_Toc183740839)

[3.1 Архитектура приложения 9](#_Toc183740840)

[3.2 Реализация ключевых компонентов 10](#_Toc183740841)

[3.2.1 Frontend 10](#_Toc183740843)

[3.2.2 Backend 10](#_Toc183740844)

[3.2.3 База данных (опционально) 11](#_Toc183740845)

[3.2.4 Особенности реализации отдельных компонентов (опционально) 11](#_Toc183740846)

[3.3 Тестирование приложения 11](#_Toc183740847)

[3.4 Документирование кода 11](#_Toc183740848)

[ЗАКЛЮЧЕНИЕ 12](#_Toc183740849)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 13](#_Toc183740850)

# ВВЕДЕНИЕ

В процессе обучения может возникать необходимость организации интерактивов с группой. Проект «Своя игра» на языке Kotlin направлен на разработку платформы, обеспечивающей функциональность запуска, функционирования, создания и редактирования игр, созданных или скачанных. Основной целью является создание простого в использовании приложения с высокой производительностью, понятным интерфейсом и возможностью быстро создавать «свои игры» быстро и без затруднений.

Необходимо реализовать систему, работающую по раундам, в каждом из которых могут быть свои правила. Также требуется создать обработку правильных/неправильных ответов и действий, следующих за ними. Для отслеживания лидеров игры нужна система подсчёта очков, учитывающая всех условия: раунд, стоимость правильного ответа, тип вопроса.

При выполнении задания на проект необходимо сделать простой редактор игр, позволяющий настраивать количество раундов, тем, вопросов, количество участников, стоимость правильных ответов, создавать и изменять вопросы, указывать правильные ответы, вести подсчет очков. Реализация функции сохранения созданной игры в файл ускорит и упростит создание новой игры.

Проект актуален, как демонстрация владения современным языком Kotlin, позволяющим писать производительные кроссплатформенные приложения. В проекте также будут использованы современные архитектурные паттерны, такие как MVVM или MVI.

Готовое приложение можно адаптировать для использования в учебных заведениях для проведения викторин, в компаниях для корпоративных мероприятий или просто для игры с друзьями.

Для достижения цели необходимо выполнить следующие задачи:

* Задача 1: Разработка интерфейса,
* Задача 2: Разработка игровой логики,
* Задача 3: Разработка редактора игр,
* Задача 4: Создание системы сохранения игр.

Состав команды и распределение ролей:

* Петруничев А.А.: Разработчик, Менеджер проекта
* Лысачева М.М.: Разработчик, Тестировщик

# 1. ПЛАНИРОВАНИЕ И ПОДГОТОВКА

## **1.1 Техническое задание**

### 1.1.1 Общие сведения

Полное наименование: Система для создания «Своей игры».

Краткое наименование: Система.

### 1.1.2 Цели и назначение автоматизированной системы

Назначение: Система предназначена для обеспечения простого способа реализации интерактива в формате «Своей игры».

Цель: создать Систему, упрощающую и ускоряющую процесс создания и использования «Своей игры».

### 1.1.3 Характеристика объектов автоматизации

Объектом автоматизации Системы является процесс «Реализация Своей игры». Данный процесс подразумевает:

* Разработку банка вопросов на заданное количество различных тем;
* Подготовку ответов на составленные вопросы;
* Распределение вопросов по стоимости правильного ответа в соответствии со сложностью вопроса;
* Визуальную реализацию в виде таблицы;
* Реализация интерактивного интерфейса: открытие выбранного вопроса, раскрытие правильного ответа, возврат к главной странице, отображение пройденных и не пройденных вопросов;
* Подсчет очков.

### 1.1.4 Требования к системе

Система должна обеспечивать следующий функционал:

* Создание игры:
  + Выбор количества игроков
  + Выбор количества раундов
  + Выбор количества тем
  + Выбор количества вопросов
  + Создание банка вопросов
  + Настройка правильных ответов
  + Настройка системы подсчета очков
* Использование игры:
  + Настройка количества экранов
  + Запуск игры с отображением правил
  + Объявление раунда и его тем
  + Табличное представление тем и количества вопросов с указанием стоимости правильного ответа
  + Переход на выбранный вопрос с отображением его формулировки
  + Демонстрация правильного ответа
  + Увеличение количества очков ответившим правильно
  + Возврат к списку тем и вопросов
  + Различное отображение пройденных и не пройденных вопросов
  + Переход к следующему раунду
  + Демонстрация результатов

### 1.1.5 Состав и содержание работ

Разработка Системы осуществляется поэтапно в течение семестра в соответствии с заданием, выданным преподавателем, в установленные им сроки.

### 1.1.6 Порядок разработки и приемки

Разработка Системы выполняется в соответствии с описанием процесса «Создания Своей игры» в сроки, установленные преподавателем для каждого этапа.

По окончании разработки предоставляется следующая документация:

* Техническое задание на разработку;
* Исходный код;
* Руководство пользователя.

Прием проекта осуществляется преподавателем.

### 1.1.7 Требования к подготовке объекта к вводу системы в действие

Ввод информации в Систему осуществляется в электронном виде.

### 1.1.8 Требования к документированию

В состав отчетной технической документации входят:

* Техническое задание на разработку;
* Исходный код;
* Руководство пользователя.

Документация к проекту составляется на русском языке.

### 1.1.9 Источники разработки

Источники разработки…...

## **1.2 Диаграмма вариантов использования**

### 1.2.1 Обоснование сценариев использования

Пользователь создал свою игру.

Пользователь запустил игру.

Пользователь проводит игру в качестве ведущего.

Пользователь участвует в игре как игрок.

Пользователь решил изменить созданную им до этого игру.

Пользователь хочет передать созданную им игру другу или переместить на другое устройство.

### 1.2.2 Схематическое представление (Use Case Diagram)

Включает в себя актеров (Пользователь (игрок/ведущий), Система) и сценарии (сценарии перечислены выше, в п. 1.2.1)

Результат создания диаграммы прецедентов представлен на Рисунке 1.

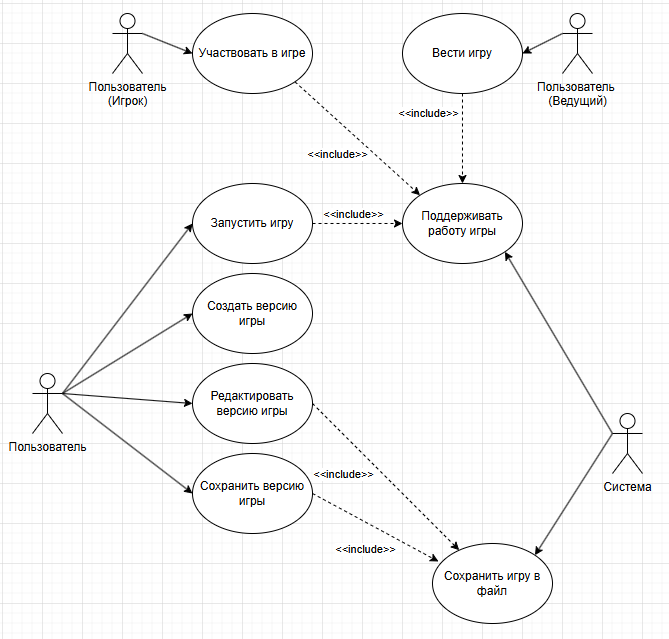


Рисунок 1 — Диаграмма вариантов использования разрабатываемого проекта

## **1.3 Создание репозитория**

На платформе GitHub был создан открытый репозиторий. В участников были добавлены весь состав группы разработки. Результат создания репозитория разрабатываемого проекта представлен на Рисунке 2.

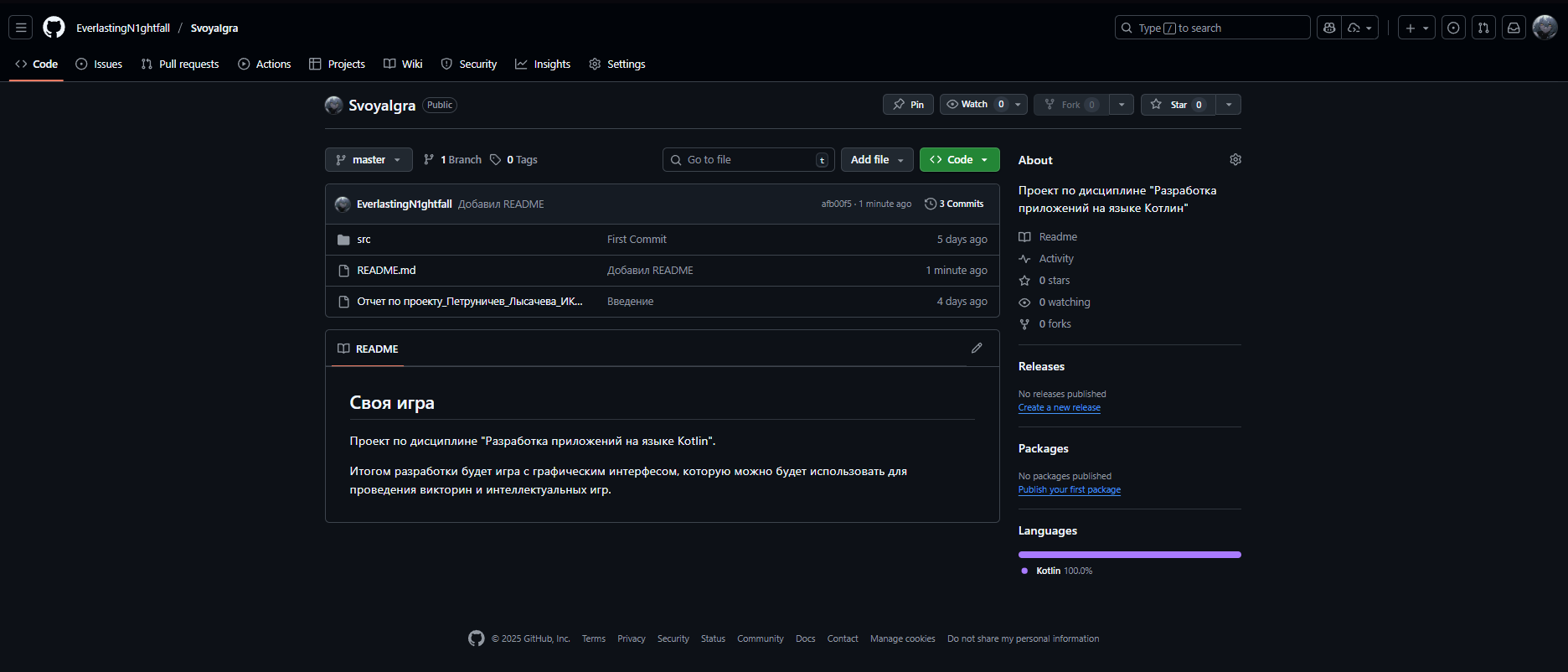


Рисунок 2 — Результат создания репозитория проекта

Ссылка на репозиторий: https://github.com/EverlastingN1ghtfall/SvoyaIgra

# 2. ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА

## **2.1 Макеты экранов**

Описать проектирование интерфейса/экранов (схематичное представление) и т.п.

Макет экрана страницы заказа товара представлен на Рисунке 3.

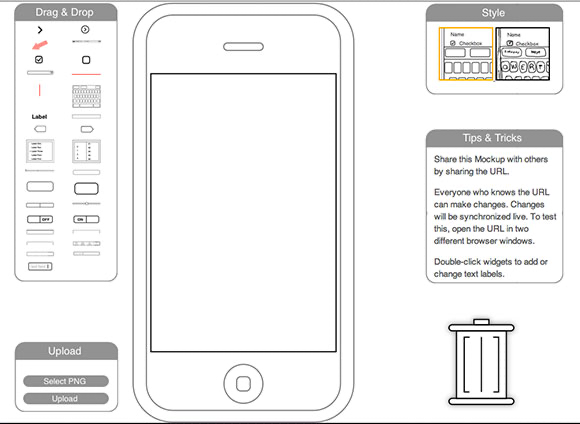


Рисунок 3 — Результат создания макета экрана страницы «О товаре»

# 3. РАЗРАБОТКА ПРИЛОЖЕНИЯ

## **3.1 Архитектура приложения**

Существует множество архитектурных подходов для разработки программных приложений, и выбор зависит от типа проекта, требований к масштабируемости, поддерживаемости и функциональности…MVC, MVVM, MVP, Microservices Architecture, Клиент-серверная и т.п…

Выбрана архитектурная модель такая-то….потому что … для … .

Схематичное изображение архитектуры разрабатываемого проекта представлено на Рисунке 4.

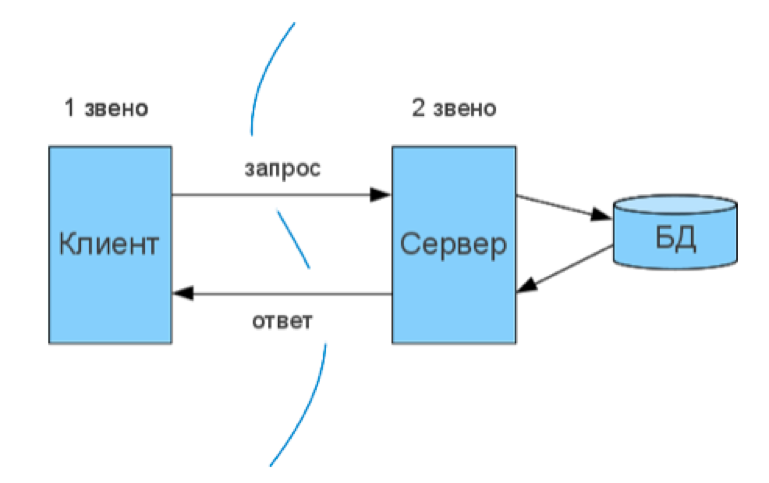


Рисунок 4 — Результат создания схемы архитектуры проекта

## **3.2 Реализация ключевых компонентов**

### Заполняется в зависимости от типа проекта каждым участником команды индивидуально. Пункты frontend, backend и т.п – опционально. Создавайте необходимые пункты под свой проект. Листинги кода обязательны.

### 3.2.1 Frontend

Для реализации клиентской части проекта выбраны технологии/фреймворки…и т.п.

Реализованный функционал карточки товара представлен в Листинге 1.

Листинг 1 – Листинг кода

|  |
| --- |
| код |

### 3.2.2 Backend

Для реализации серверной части проекта выбраны технологии/фреймворки…и т.п.

Реализованный функционал карточки товара представлен в Листинге 2.

Листинг 2 – Листинг кода

|  |
| --- |
| код |

### 3.2.3 База данных (опционально)

Какая СУБД была выбрана, описание структуры БД и т.п.

### 3.2.4 Особенности реализации отдельных компонентов (опционально)

Особенности….

## **3.3 Тестирование приложения**

Для тестирования приложения были написаны Unit тесты для ключевого функционала ….

Или тестирование проекта производилось с использованием ….Selenium…. и т.п

Результат написания Unit тестов представлен в Листинге …

Результат успешного прохождения тестирования представлен на Рисунке …

## **3.4 Документирование кода**

Использован KDoc для описания классов и функций. Генерация HTML-документации производилась … Пример документирования кода ключевого функционала представлен на Рисунке….

# ЗАКЛЮЧЕНИЕ

Разработка интернет-магазина позволила реализовать современное решение для электронной коммерции. Основные задачи проекта выполнены — от проектирования до тестирования и документирования. В будущем возможна интеграция с системами аналитики и добавление новых функций, таких как рекомендации товаров. Разработка проекта также способствовала улучшению навыков программирования на языке Kotlin и развитию командной работы.

Ссылка на GitHub-репозиторий разработанного проекта: ….

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 34.602—2020 ("Техническое задание на создание автоматизированной системы") является стандартом, который определяет общие требования к разработке автоматизированных систем (АС). Он используется в разных отраслях, включая медицину, и применим для разработки автоматизированных информационных систем (АИС).
2. ГОСТ 34.201—2020. Межгосударственный стандарт. Информационные технологии. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем: Приказом Федерального агентства по техническому регулированию и метрологии № 1521-ст от 19 ноября 2021 г.: дата введения 2022-01-01. – М.: Российский институт стандартизации, 2021. – 10 с.
3. Блоштин А.В., Лаптев Д.В. Современные подходы к проектированию клиент-серверных приложений // Программные системы: теория и приложения. 2021. №2. URL: https://elibrary.ru/item.asp?id=46523678 (дата обращения: 26.11.2025)
4. Android Developers — официальный сайт документации по разработке на Kotlin: https://developer.android.com/kotlin (дата обращения: 26.11.2025).
5. Kotlin Documentation — полный справочник по языку Котлин: https://kotlinlang.org/docs (дата обращения: 26.11.2025).
6. Жемеров С., Исакова С. Kotlin в действии. 2-е издание. М.: Питер, 2022. URL: https://www.piter.com/product/kotlin-v-dejstvii (дата обращения: 26.11.2025).
7. Документация JUnit — для тестирования Kotlin-приложений: https://junit.org/junit5 (дата обращения: 26.11.2025).