

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МО ЭВМ**

**ИНДИВИДУАЛЬНОЕ ДОМАШНЕЕ ЗАДАНИЕ**  
**по дисциплине «Разработка приложений для мобильных платформ»**  
**Тема: Hotseat игра по типу Dixit**

Студент гр. 1304

\_\_\_\_\_

Мусаев А.И.

Студентка гр. 1304

\_\_\_\_\_

Нго Тхи Йен.

Студент гр. 1304

\_\_\_\_\_

Поршнеv Р.А.

Преподаватель

\_\_\_\_\_

Заславский М. М.

Санкт-Петербург

2025

## ЗАДАНИЕ

Студент Мусаев А.И.

Студентка Нго Тхи Йен

Студент Поршнеv Р.А.

Группа 1304

Тема: Hotseat игра по типу Dixit

Содержание пояснительной записки:

«Содержание»

«Введение»

«Сценарии использования»

«Пользовательский интерфейс»

«Разработанное приложение»

«Выводы»

«Приложения»

«Список использованных источников»

Предполагаемый объем пояснительной записки:

Не менее 15 страниц.

Дата выдачи задания: 03.02.2025

Дата сдачи реферата: 27.03.2025

Дата защиты реферата: 27.03.2025

Студент	_____	Мусаев А. И.
Студентка	_____	Нго Тхи Йен
Студент	_____	Поршнеv Р. А.
Преподаватель	_____	Заславский М. М.

## АННОТАЦИЯ

В рамках индивидуального домашнего задания был разработан макет мобильного приложения для игры в режиме Hotseat, основанной на механике игры Dixit. Макет включает в себя основные экраны приложения, такие как главный экран, экран выбора игроков, экран игры и экран итогов. Также был разработан пользовательский интерфейс (UI) на заглушках с использованием языка программирования Kotlin в среде разработки Android Studio. Навигация между экранами реализована с использованием стандартных инструментов Android. Данный UI является прототипом, предназначенным для демонстрации логики работы приложения до полноценной реализации игровых механик.

Исходный код доступен по ссылке:  
<https://github.com/moevm/adfmp1h25-tixid>.

## SUMMARY

As part of an individual assignment, a mobile application mockup was developed for a Hotseat game based on the mechanics of Dixit. The mockup includes the main screens of the application, such as the main screen, player selection screen, game screen, and results screen. Additionally, a placeholder-based user interface (UI) was implemented using the Kotlin programming language in the Android Studio development environment. Navigation between screens was implemented using standard Android tools. This UI serves as a prototype designed to demonstrate the application's logic before the full implementation of game mechanics.

The source code is available at the link:  
<https://github.com/moevm/adfmp1h25-tixid>.

## СОДЕРЖАНИЕ

1.	Введение	7
1.1.	Актуальность проблемы	7
1.2.	Постановка задачи	7
1.3.	Предлагаемое решение	8
1.4.	Качественные требования к решению	8
2.	Сценарии использования	9
2.1.	Запустить игру	9
2.2.	Добавить игрока	9
2.3.	Передать следующему	10
2.4.	Остальные игроки выбирают карту	11
2.5.	Голосование	11
2.6.	Подсчёт очков и завершение раунда	12
2.7.	Просмотр карты с фишками и начало нового раунда	12
2.8.	Просмотр статистики	13
2.9.	Постановка игры на паузу	13
2.10.	Завершение игры	14
2.11.	Новая игра (из любого состояния приложения)	14
3.	Пользовательский интерфейс	16

3.1.	Макет интерфейса с графом переходов	16
3.2.	Целевые устройства, обоснование требований и максимально подробные характеристики	16
4.	Разработанное приложение	17
4.1.	Краткое описание приложения	17
4.2.	Схема архитектуры	17
4.3.	Использованные технологии (внешние)	17
4.4	Использованные модули/системные библиотеки платформы	17
5.	Выводы	18
5.1.	Достигнутые результаты	18
5.2.	Недостатки и пути для улучшения полученного решения	18
5.3.	Будущее развитие решения	18
6.	Приложения	20
6.1.	Инструкция для пользователя	20
6.2.	Снимки экрана приложения	20
	Список использованных источников	22

# **1. ВВЕДЕНИЕ**

## **1.1. Актуальность проблемы**

Разработка мобильного приложения для игры в режиме Hotseat, основанной на механике настольной игры Dixit, является актуальной задачей по нескольким причинам. Во-первых, настольные игры, особенно ассоциативные, такие как Dixit, пользуются большой популярностью среди различных возрастных групп, так как они способствуют развитию креативного мышления, воображения и коммуникативных навыков. Однако не всегда есть возможность собрать компанию друзей или семьи за физической игрой, поэтому цифровая версия с возможностью игры на одном устройстве становится удобной альтернативой. Во-вторых, мобильные платформы представляют собой удобный способ распространения подобных игр благодаря доступности смартфонов и планшетов. В условиях роста мобильного гейминга и увеличения времени, проводимого пользователями в приложениях, разработка цифровой версии Dixit с локальным многопользовательским режимом (Hotseat) позволит удовлетворить спрос на интерактивные и социальные игры, которые можно запустить в любом месте без необходимости подключения к интернету.

## **1.2. Постановка задачи.**

Задача проекта заключается в разработке мобильного приложения для игры в режиме Hotseat, которое позволяет пользователям:

- Выбирать количество игроков и задавать их имена перед началом игры;
- Просматривать и выбирать карточки с изображениями в соответствии с механикой игры Dixit;
- Голосовать за карты других игроков, пытаясь угадать карту рассказчика;
- Подсчитывать очки по установленным правилам и отображать текущий рейтинг игроков;
- Просмотреть итоговые результаты игры после завершения раунда;

- Переключаться между экранами игры с помощью удобной навигации;
- Использовать игру в офлайн-режиме без необходимости подключения к интернету;
- Запускать приложение на мобильных устройствах под управлением Android.

### **1.3. Предлагаемое решение.**

Для реализации проекта должно быть создано мобильное приложение на языке Kotlin с использованием Android Studio. Интерфейс будет реализован с помощью XML-разметки, а игровая логика — в соответствии с механикой Dixit.

### **1.4. Почему решение необходимо реализовывать как мобильное приложение**

Реализация решения в формате мобильного приложения обусловлена несколькими ключевыми факторами. Во-первых, мобильные устройства являются наиболее удобной платформой для казуальных игр, так как они всегда под рукой и позволяют играть в любом месте без необходимости наличия настольной версии игры. Во-вторых, формат мобильного приложения с режимом Hotseat позволяет нескольким игрокам взаимодействовать на одном устройстве, что делает игру доступной даже без интернета и сетевых подключений. В-третьих, использование Android-платформы упрощает распространение игры через Google Play, а также обеспечивает удобство использования благодаря сенсорному управлению. Кроме того, мобильное приложение позволяет гибко настраивать игровой процесс, добавлять анимации и улучшать пользовательский опыт, делая игру более захватывающей и доступной.



## **2. СЦЕНАРИИ ИСПОЛЬЗОВАНИЯ**

### **2.1. Запустить игру**

Действующие лица: Пользователь.

Предусловия:

- Приложение запущено.
- Игроки готовы начать новую игру.

Основной сценарий:

1. Пользователь открывает приложение.
2. Нажимает кнопку «Новая игра».
3. Вводит (или редактирует) имена всех участников (от 3-ёх до 6-ти).
4. Запускает игру.

Результат основного сценария:

- Игра успешно начинается с указанными участниками.

### **2.2 Добавить игрока**

Действующие лица: Текущий игрок (инициатор добавления), Текущий игрок (инициатор добавления).

Предусловия:

- Приложение запущено.
- Существует активная сессия игры и количество участников не более 6.

Основной сценарий:

1. Текущий игрок нажимает кнопку или выбирает пункт меню «Добавить игрока».
2. На экране появляется форма для ввода имени нового игрока.
3. Текущий игрок или сам новый игрок вводит требуемые данные.
4. Система проверяет корректность данных (имя не пустое и не совпадает с уже существующими именами).

5. Система добавляет нового игрока в список участников текущей игры.

Результат основного сценария:

- В списке участников появляется новый игрок, который может участвовать в последующих раундах игры.

Альтернативный сценарий 1 (имя занято или пустое):

1. При вводе данных система обнаруживает, что имя игрока уже используется или оставлено пустым.

2. Отображается сообщение об ошибке, пользователь должен ввести другое имя.

3. Возврат к шагу 3 основного сценария.

Альтернативный сценарий 2 (игрока не нужно добавлять):

1. На экране добавления есть кнопка "Отменить".

2. Нажатие кнопки приводит к возврату к списку игроков без сохранения изменений.

Альтернативный сценарий 3 (игроков больше 6 или меньше 3):

1. На экране добавления выйдет уведомление.

2. Возврат в меню игроков.

Результат альтернативных сценариев:

- Либо игрок успешно добавлен после устранения ошибок, либо добавление игрока отменяется.

### **2.3 Передать следующему**

Действующие лица: Текущий игрок, Следующий игрок.

Предусловия:

- Игра запущена.
- Следующий игрок должен положить свою карту либо выбрать карту ведущего.

Основной сценарий:

1. Текущий игрок (например, ведущий) завершает свой ход (выбрал

карту, задал подсказку, подтвердил действие и т.д.).

2. Нажимает кнопку «Передать следующему».
3. На экране появляется специальный экран с кнопкой «Следующий игрок».
4. Следующий игрок берет устройство и нажимает кнопку.
5. Отображаются его карты или действия, доступные именно ему.
6. Игровой процесс продолжается.

Результат основного сценария:

- Передача устройства безопасна, предыдущий игрок не видит чужие карты, а следующий игрок получает свой интерфейс.

## **2.4 Остальные игроки выбирают карты**

Действующие лица: Игроки (не ведущий).

Предусловия:

- Ведущий уже задал подсказку и выбрал свою карту.
- Наступил ход остальных игроков.

Основной сценарий:

1. По очереди (через передачу устройства) каждый игрок видит свою руку карт.
2. Игрок выбирает одну карту, которая, по его мнению, наиболее подходит к подсказке ведущего.
3. Подтверждает выбор.
4. После подтверждения последним из игроков система переходит к следующему шагу — «Голосование».

Результат основного сценария:

- Каждый игрок добавил по одной карте в общий «пул» (кроме ведущего, который уже выбрал карту).

## **2.5 Голосование**

Действующие лица: Все игроки, кроме ведущего.

Предусловия:

- Все карты собраны в общий пул.
- Игроки готовы голосовать, какая из карт принадлежит ведущему.

Основной сценарий:

1. Система случайным образом раскладывает на экране все выбранные карты (включая карту ведущего), не показывая, кому они принадлежат.
2. По очереди (через передачу устройства) каждый игрок просматривает список карт и выбирает ту, которая, по его мнению, является картой ведущего.
3. Игрок подтверждает голос.
4. После голосования последнего игрока система переходит к подсчету результатов.

Результат основного сценария:

- Система получает выбор (голос) от каждого игрока.

## **2.6 Подсчёт очков и завершение раунда**

Действующие лица: Игроки.

Предусловия:

- Все игроки проголосовали.
- Игра не поставлена на паузу.

Основной сценарий:

1. Система вскрывает карты, показывает, какая карта принадлежала ведущему.
2. Система сравнивает результаты голосования с реальной картой ведущего и начисляет очки согласно правилам (например, ведущему нужны не все голоса, но и не 0).
3. Выводится сводка:
  - a. Карта ведущего
  - b. Сколько голосов она получила
  - c. Карты остальных игроков и кто за них проголосовал

- d. Начисленные очки
- 4. Игроки просматривают результаты, нажимают «Далее».

Результат основного сценария:

- Очки обновлены, игроки видят, кто и сколько заработал в текущем раунде.

## **2.7 Просмотр карты с фишками и начало нового раунда**

Действующие лица: Игроки.

Предусловия:

- Завершён подсчёт очков за предыдущий раунд.
- Игра продолжается.

Основной сценарий:

1. После завершения раунда система автоматически отображает карту с фишками игроков (игровое поле, где видно текущее положение).
2. Игроки могут увидеть своё продвижение или количество очков на поле.
3. Ведущий (или любой игрок) нажимает кнопку «Начать раунд».
4. Игра переходит к следующему раунду, ведущий меняется по кругу или остаётся тот же — согласно правилам.

Результат основного сценария:

- Игроки видят своё текущее положение и могут продолжить игру.

## **2.8 Просмотр статистики**

Действующие лица: Игрок.

Предусловия:

- Игра находится на начальном экране (или в любом месте, где доступна кнопка «Статистика»).

Основной сценарий:

1. Пользователь нажимает кнопку «Статистика игроков».

2. Система показывает список участников и их очки (например, общее количество очков или подробный счёт по раундам).

Результат основного сценария:

- Игрок получает информацию о текущем счёте.

Альтернативный сценарий:

- Статистика недоступна (например, нет активной игры или ни одного завершённого раунда).
- Система уведомляет пользователя об отсутствии данных и предлагает начать игру.

## **2.9 Постановка игры на паузу**

Действующие лица: Игрок.

Предусловия:

- Игра запущена.

Основной сценарий:

1. Игрок нажимает кнопку «Пауза».
2. Экран затемняется или блокируется, показывается кнопка «Убрать паузу».
3. Игроки могут оставить устройство без риска показать карты.
4. При нажатии «Убрать паузу» игра продолжается.

Результат основного сценария:

- Игра безопасно приостанавливается и может быть возобновлена.

## **2.10 Завершение игры**

Действующие лица: Игроки.

Предусловия:

- Игра достигла финального этапа (закончились карты, достигнуто нужное количество очков или раундов).

Основной сценарий:

1. После завершения последнего раунда система автоматически вычисляет итоговые очки всех игроков.

2. На экране отображается сообщение с именем победителя:

«Поздравляем! Выиграл игрок [Имя]!»

3. Игроки могут просмотреть итоговую статистику.

4. Пользователь выбирает одно из действий:

a. Начать новую игру

b. Вернуться в главное меню

c. Выйти из приложения (при необходимости)

Результат основного сценария:

- Игроки видят победителя и могут начать новую игру или вернуться в меню.

## **2.11 Новая игра (из любого состояния приложения)**

Действующие лица: Пользователь.

Предусловия:

- Приложение запущено.

Основной сценарий:

1. Пользователь нажимает кнопку «Новая игра».

2. Пользователь вводит имена игроков (минимум 3, максимум 6).

3. Пользователь подтверждает список игроков и запускает новую игру.

Результат основного сценария:

- Игра запущена.

Альтернативный сценарий 1 (Отмена действия):

1. На экране подтверждения пользователь выбирает «Отменить» или «Назад».

2. Приложение возвращается к предыдущему экрану без изменений в текущем прогрессе.

Результат альтернативного сценария:

- Новая игра не начинается.

### 3. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС

#### 3.1 Макет интерфейса с графом переходов

Макет интерфейса с графом переходов представлен на рисунке 1.

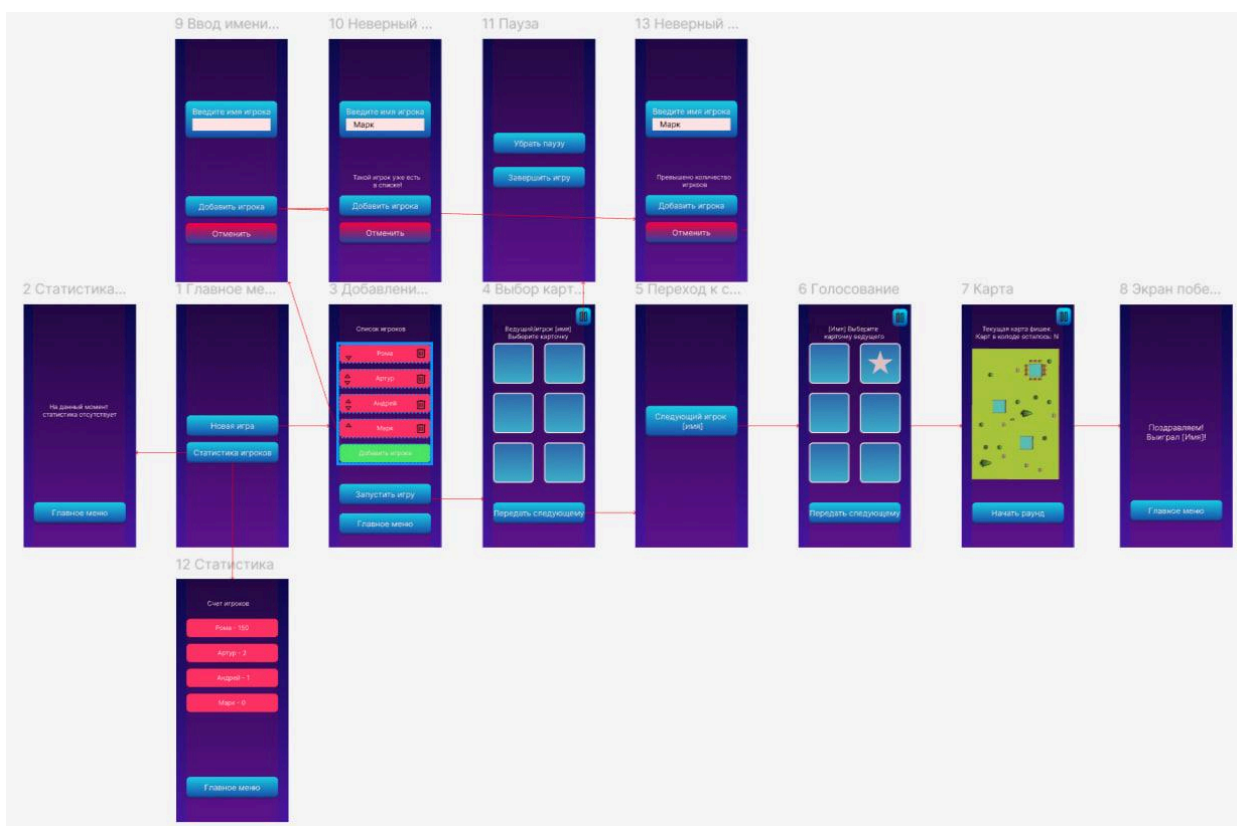


Рисунок 1 - Макет

#### 3.2 Целевые устройства, обоснование требований и максимально подробные характеристики

Целевыми устройствами для разработки являются смартфоны и планшеты на базе Android. Приложение ориентировано на современные устройства с API Level 35 (Android 15) и поддерживает экраны с высоким DPI. Минимальные системные требования включают Android 8.0, 2 ГБ ОЗУ, 100 МБ свободного места и поддержку OpenGL ES 3.0. Рекомендуемые характеристики — Android 12+, 4 ГБ ОЗУ, 500 МБ памяти и Vulkan API. Разработка и тестирование ведутся на эмуляторе Pixel 9 Pro API 35 с разрешением 1280×2856, 480 DPI, 2-ядерным процессором и 11,5 ГБ ОЗУ.



## **4. РАЗРАБОТАННОЕ ПРИЛОЖЕНИЕ**

### **4.1. Краткое описание приложения.**

Проект представляет собой мобильное приложение для игры в режиме Hotseat, основанное на механике настольной игры Dixit. В текущей версии разработан пользовательский интерфейс (UI) на заглушках без реализации игровой логики, с возможностью навигации между экранами.

### **4.2. Схема архитектуры.**

Приложение состоит из нескольких основных экранов, таких как главный экран, экран выбора игроков, игровой экран и экран итогов. Навигация реализована с использованием стандартных инструментов Android. Взаимодействие между компонентами сосредоточено на передаче данных между экранами, а также обработке пользовательских действий.

### **4.3. Используемые технологии (внешние).**

- Kotlin – основной язык разработки приложения.
- Android SDK – набор инструментов для работы с Android.

### **4.4 Используемые модули/системные библиотеки платформы:**

- Jetpack Compose – для построения пользовательского интерфейса.
- Navigation Component – для организации навигации между экранами.
- ViewModel – для управления данными, связанными с UI.
- LiveData – для обновления интерфейса в ответ на изменения данных.

## **5. ВЫВОДЫ**

### **5.1. Достигнутые результаты.**

На текущем этапе разработан пользовательский интерфейс (UI) мобильного приложения для игры в режиме Hotseat, основанного на механике настольной игры Dixit. Реализована навигация между основными экранами, что позволяет пользователям переходить между этапами игры.

### **5.2. Недостатки и пути для улучшения полученного решения.**

На данный момент в приложении отсутствует игровая логика, что делает его неполноценным. Для дальнейшего развития необходимо реализовать основные механики игры, такие как выбор карт, голосование и подсчет очков. Также отсутствует гибкость в настройках, например, возможность изменения правил или количества карт. Добавление таких опций улучшит пользовательский опыт. Игра рассчитана только на локальный режим Hotseat, что ограничивает ее аудиторию. В перспективе можно рассмотреть добавление онлайн-мультиплеера. UI можно оптимизировать за счет улучшения интерактивных элементов и анимаций, что сделает игровой процесс более плавным и привлекательным.

### **5.3. Будущее развитие решения.**

В дальнейших версиях может быть разработана и внедрена игровая логика, позволяющая пользователям полноценно играть в цифровую версию Dixit. Также будет улучшено визуальное оформление, включая анимации и интерактивные элементы. Может быть добавлена возможность настраивать параметры игры, а также сохранение статистики и прогресса. В перспективе возможно расширение функционала за счет сетевого режима, позволяющего играть по интернету или локальной сети. Дополнительное тестирование на реальных устройствах и оптимизация производительности также станут важными этапами развития проекта. Таким образом, приложение имеет

потенциал для масштабирования и превращения в полноценную мобильную игру.

## 6. ПРИЛОЖЕНИЯ

### 6.1. Инструкция для пользователя.

Для установки приложения скачайте репозиторий с GitHub и откройте его в Android Studio. Запустите эмулятор с Android API 35 или подключите физическое устройство с Android 8.0+ в режиме разработчика. После запуска откроется главный экран, где можно выбрать количество игроков. Далее происходит переход к игровому процессу, в котором пользователи могут взаимодействовать с картами и переходить между экранами. После завершения игры отображается итоговый экран. Управление осуществляется через сенсорные кнопки и жесты.

### 6.2. Снимки экрана приложения.

Экраны приложения представлены на рисунке 1, 2, 3, 4, 5.

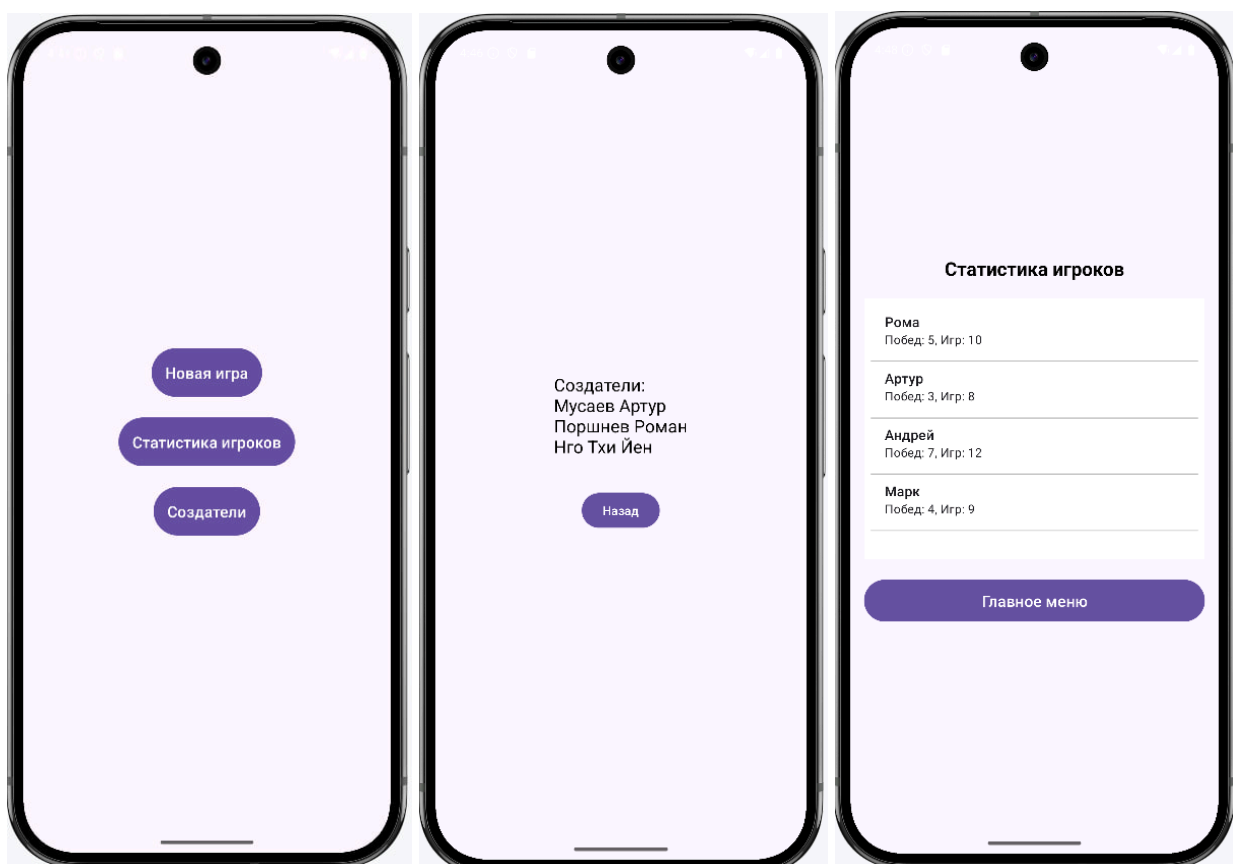


Рисунок 2 - Экраны главного меню, создателей и статистики

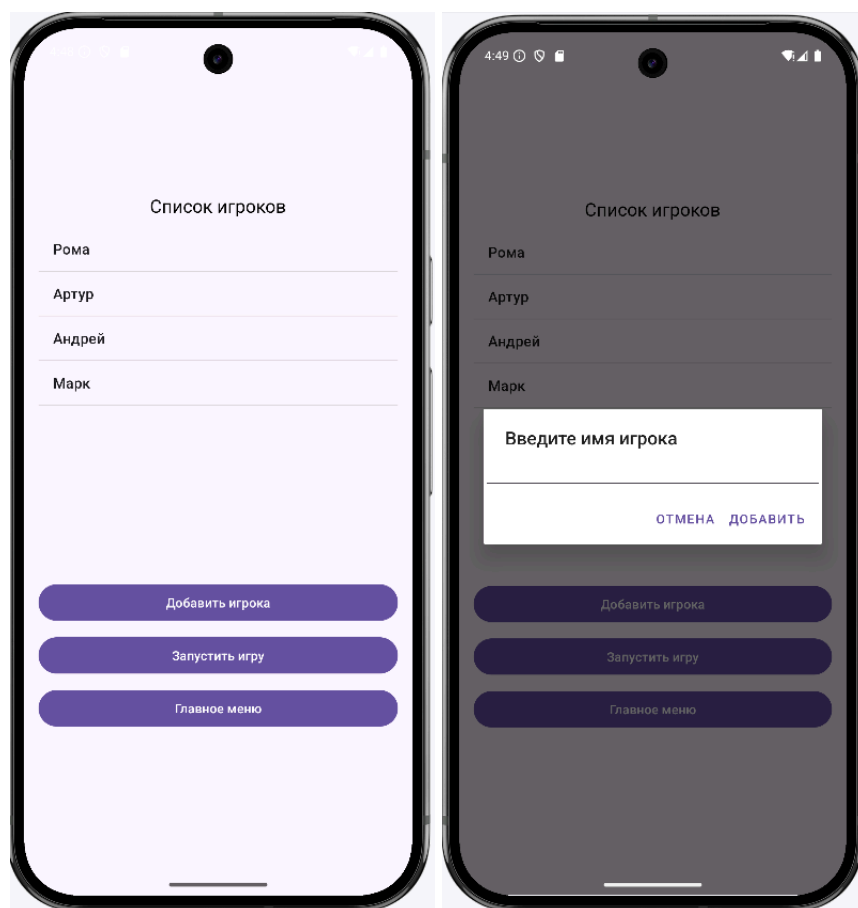


Рисунок 3 - Список игроков и экран добавления игрока

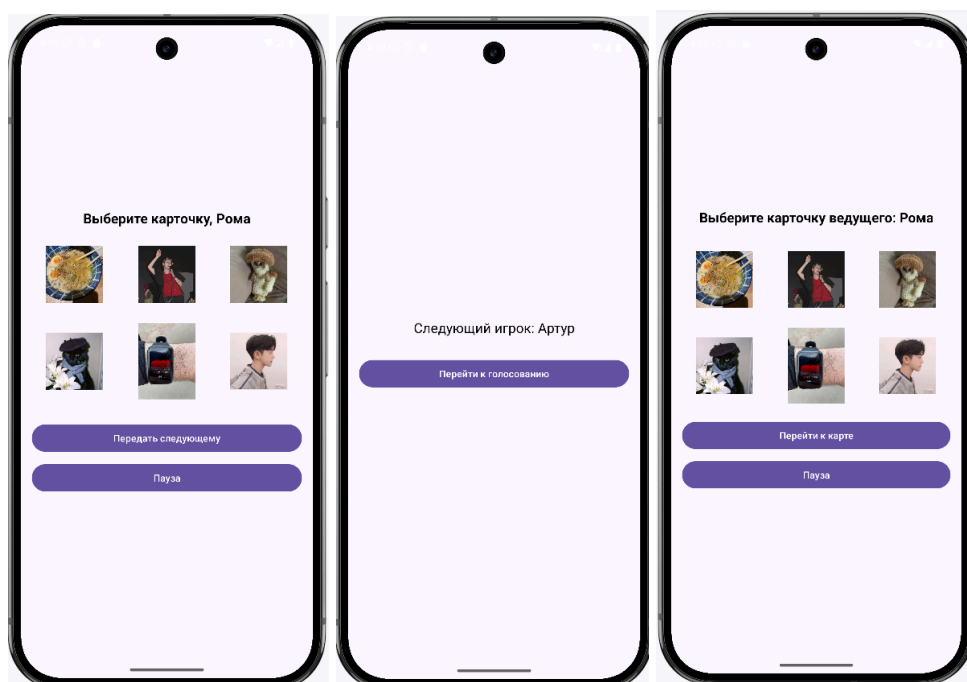


Рисунок 4 - Экраны выбора карточки, передачи телефона и голосования

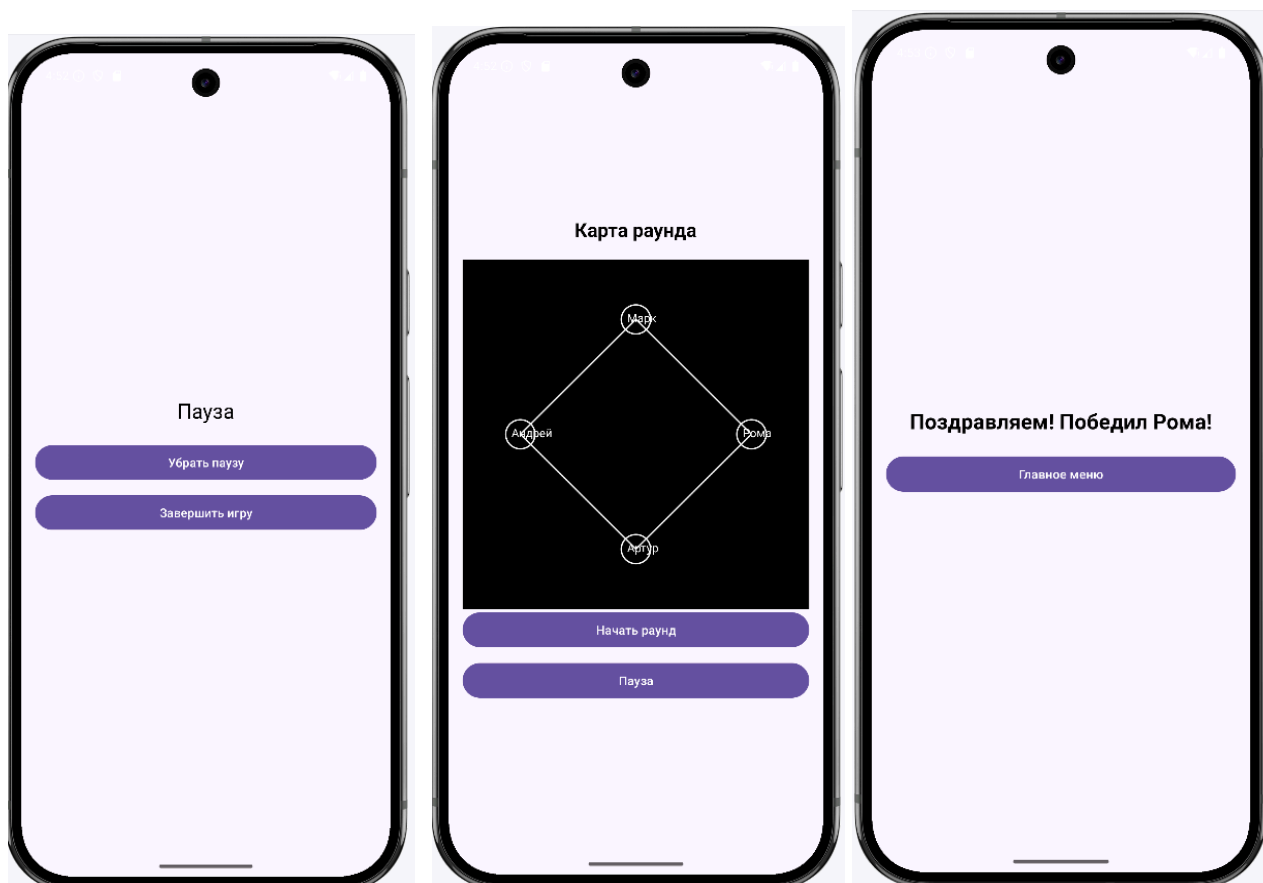


Рисунок 5 - Экраны паузы, карты и победы

## **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Ссылка на код проекта на GitHub. – [Электронный ресурс]. – URL: <https://github.com/moevm/adfmp1h25-tixid>.
2. Правила Dixit. – [Электронный ресурс]. – URL: [https://ru.wikipedia.org/wiki/Dixit\\_\(игра\)](https://ru.wikipedia.org/wiki/Dixit_(игра))