МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

(ФГБОУ ВО «ВГУ»)

Факультет Компьютерных наук

Кафедра технологий обработки и защиты информации

Курсовой проект

Многопользовательская мобильная игра «Din.io»

Программирование и информационные технологии

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_ *А.С. Цыбульская, 3 курс, д/о*

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_ *Д.Д. Пащенков, 3 курс, д/о*

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_ *А.А. Введенский, 3 курс, д/о*

Воронеж 2021

Содержание

[Содержание 2](#_Toc66783347)

[Введение 3](#_Toc66783348)

[1 Постановка задачи 4](#_Toc66783349)

[2 Глоссарий 5](#_Toc66783350)

[3 Анализ предметной области 7](#_Toc66783351)

[3.1 Анализ существующих решений 7](#_Toc66783352)

Введение

**Актуальность проекта**. В современном мире компьютерные игры очень популярны. Они помогают людям отвлечься от жизненных проблем, отдохнуть, получить такие эмоции, которые нельзя получить в обычной жизни. Особенно широкое распространение получили мобильные игры, и на это есть несколько причин.

Во-первых, мобильные приложения очень нетребовательны, ведь для их запуска не нужно владеть компьютером с высокими техническими характеристиками, а достаточно иметь обыкновенный смартфон. А смартфон сейчас есть практически у каждого человека.

Во-вторых, порог вхождения в мобильные игры зачастую низкий, ведь играют в них не опытные геймеры, а обычные люди. Это означает, что в таких играх простые и понятные правила, удобный интерфейс и интуитивное управление.

Также важную нишу в сфере игр занимают сетевые игры с мультиплеером, то есть с возможностью взаимодействия с другими игроками с помощью связи по сети. Почему такие игры распространены?

Одна из главных причин этому – то, что с настоящими людьми играть интереснее, чем с программами. Ведь поведение программы установлено заранее, а поведение человека зачастую нельзя предугадать, и эта непредсказуемость увлекает.

Также в подобные игры можно играть не только с незнакомцами, но и со своими друзьями, что привлекает людей еще больше.

Таким образом, спрос на многопользовательские мобильные игры существует. И именно поэтому мы решили разработать одну из таких игр.

# Постановка задачи

Данный проект является многопользовательской мобильной игрой, представляющей из себя Android-приложение.

Целью разработки данного приложения является:

* Создать конкурентоспособное (относительно существующих игр в жанре «io») приложение;
* Создать приложение с уникальными элементами, выделяющееся на фоне других подобных приложений;

Для достижения данной цели были выделены следующие подзадачи:

1. Разработка игровой логики;
2. Разработка Front-end части приложения;
3. Разработка Back-end части приложения;
4. Создание связи между Front-end и Back-end частями приложения;
5. Создание базы данных.

# Глоссарий

# Анализ предметной области

## Анализ существующих решений

Существует множество различных игр жанра «io». Рассмотрим две самые популярные реализации подобной игровой логики:

1. **Agar,io**.

Краткий обзор правил игры: пользователь играет за клетку, которая

может развиваться путём поедания других клеток. Более большие клетки могут съесть тех, кто меньше их по размеру. Когда одна клетка ест другую, она увеличивается в размерах, но становится медленнее. В игре существует таблица лидеров, где выводятся имена игроков, играющих за самые большие клетки.

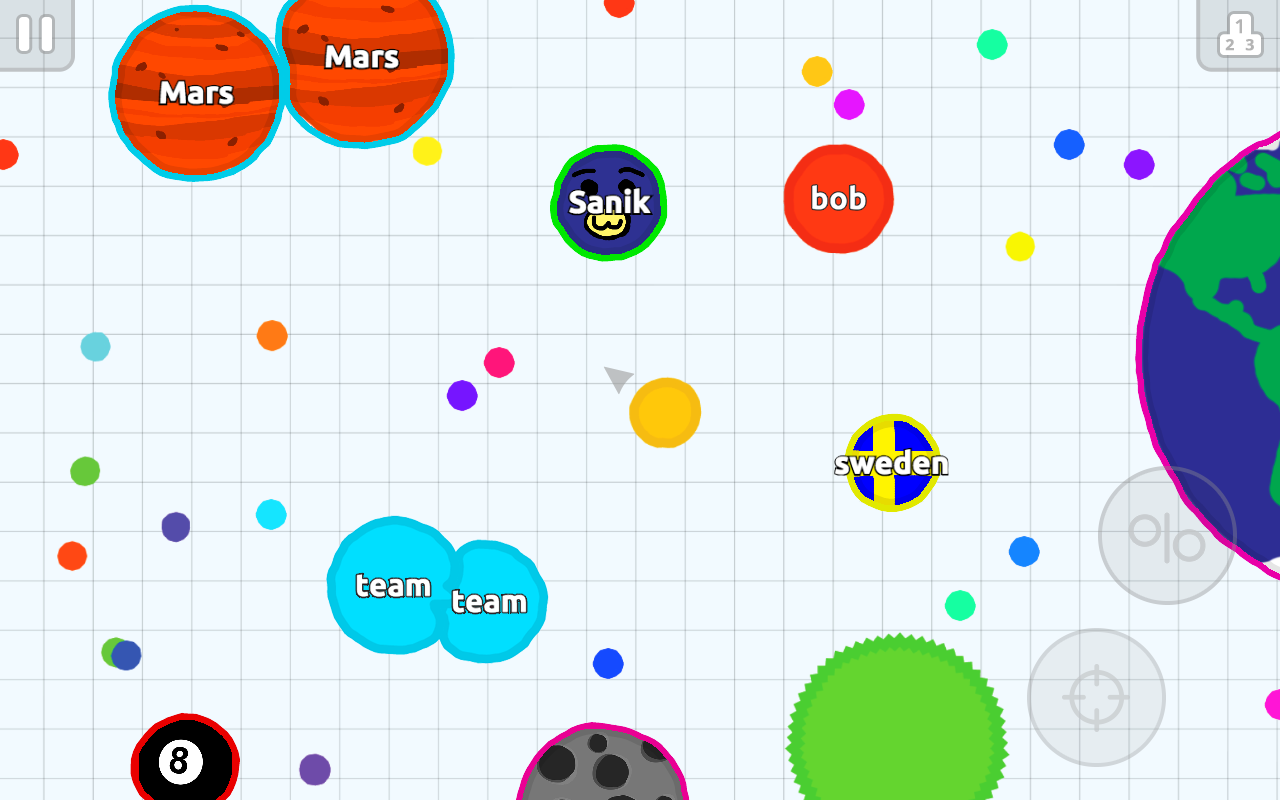


Рисунок 1. Приложение Agar.io

Основными недостатками этой игры мы считаем следующие пункты:

* Однообразие геймплея. Развитие игрового персонажа – это только его увеличение в размерах. Никакого выбора улучшения не существует. Таким образов, от игрока мало что зависит. Такой игровой сюжет быстро надоедает.
* Непобедимость «лидеров» игры. Если игровой персонаж большого размера и занимает лидирующие позиции в таблице победителей, его практически невозможно съесть. Таким образом, приблизиться к вершине таблицы лидеров становится сложно. Такая условная невозможность победы тоже может надоесть игроку.

1. **Diep.io**. Эта игра по своей игровой логике похожа на описанную в предыдущем пункте. В данном случае пользователь играет за модель танка, которая может стрелять в других игроков или игровые элементы, при этом набирая очки. За очки игроку даётся выбор из нескольких улучшений характеристик (например, дальность и скорость стрельбы, дополнительное дуло и так далее).

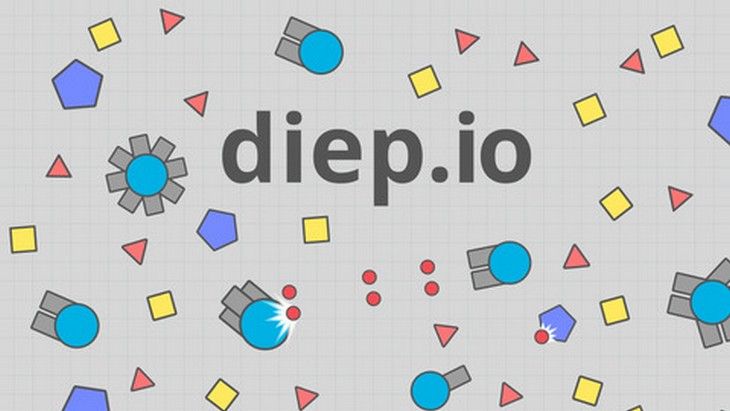


Рисунок 2. Приложение «Diep.io»

Игровой процесс этой игры достаточно разнообразен. Ведь в отличие от Agar.io существует несколько возможных вариантов развития персонажа, а значит, от пользователя напрямую зависит его количество набранных очков.

Тем не менее, пункт из предыдущих недостатков игры про непобедимость лидеров остаётся и здесь.

# Анализ задачи

## Описание правил игры

Игровые персонажи пользователей и игровые объекты пищи располагаются на карте. Новые игровые объекты пищи в процессе игры добавляются на карту.

Игровой персонаж имеет тело, состоящее из элементов, имеющих определенный уровень. Элемент может быть обычным элементом, элементом атаки или элементом питания. У каждого типа игрового персонажа изначально есть определенный набор обычных элементов и элемент питания.

У каждого игрового персонажа есть показатели очков здоровья, голода, опыта, победных очков.

Количество очков сытости убывает в процессе игры. При столкновении элемента питания игрового персонажа и игрового объекта пищи и соответствии типа персонажа типу пищи, игровой объект пищи удаляется с поля, прибавляя игроку очки сытости, здоровья, опыта и победные очки.

При столкновении элемента атаки одного персонажа с обычным элементом, элементом питания или элементом атаки более низкого уровня второго персонажа второй игровой персонаж отталкивается от первого, теряет очки здоровья, атаковавший персонаж при этом получает победные очки и очки опыта.

Рассмотрим уникальные действия для этих типов персонажей:

* Травоядный игровой персонаж может есть игровые объекты растений, но не может получать очки сытости за счёт нападения на других игровых персонажей.
* У плотоядного игрового персонажа элемент питания является атакующим элементом. Плотоядный игровой персонаж может заполнять шкалу сытости за счёт нападения на других игроков. Также он может есть игровые объекты – куски мяса.

Теперь рассмотрим, что общее есть у игровых персонажей этих типов.

* Любой игровой персонаж вне зависимости от своего типа может атаковать другого для получения победных очков
* Если игровой персонаж ест, но его шкала сытости заполнена полностью, значение шкалы здоровья увеличивается
* Когда персонаж набирает некоторое количество очков, он переходит на следующий уровень и игрок может выбирать себе одно из нескольких возможных улучшений.
* При полном опустошении шкалы сытости либо шкалы здоровья, персонаж умирает, его очки теряются, он имеет возможность вернуться в ту же игру, либо выйти в главное меню

## Варианты использования приложения

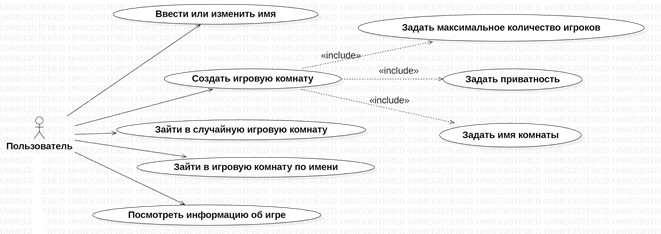


Рисунок 3. Часть диаграммы прецедентов с актором Пользователь

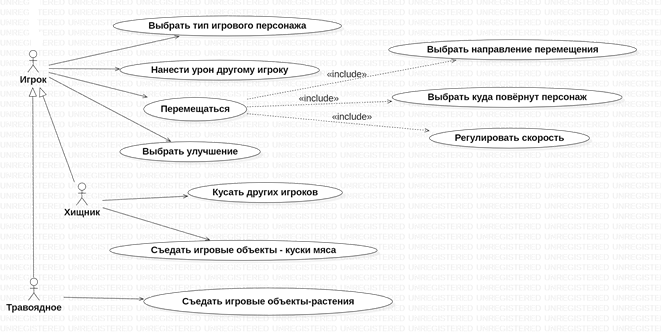


Рисунок 4. Часть диаграммы прецедентов с акторами Игрок, Хищник, Травоядное

При взаимодействии с приложением выделяются четыре роли: пользователь, игрок, травоядное, хищник. Их роли наглядно изображены на рисунке.

Пользователь:

* Ввести или изменить имя
* Создать игровую комнату
* Задать максимальное количество игроков комнаты
* Задать приватность комнаты
* Задать имя комнаты
* Зайти в случайную игровую комнату
* Зайти в комнату по имени
* Посмотреть информацию об игре

После того, как Пользователь начинает участие в игре, он становится Игроком.

Игрок:

* Выбрать тип игрового персонажа
* Нанести урон другому игроку
* Перемещаться
* Выбрать улучшение
* Выбрать направление перемещения
* Выбрать, куда повёрнут персонаж
* Выбрать скорость перемещения

После того, как Игрок выбирает себе тип игрового персонажа, он становится либо Хищником, либо Травоядным.

Травоядное:

* Съесть игровой объект-растение

Хищник:

* Съесть игровой объект – кусок мяса
* Укусить другого игрока

## Взаимодействие компонентов системы

## Взаимодействие в системе

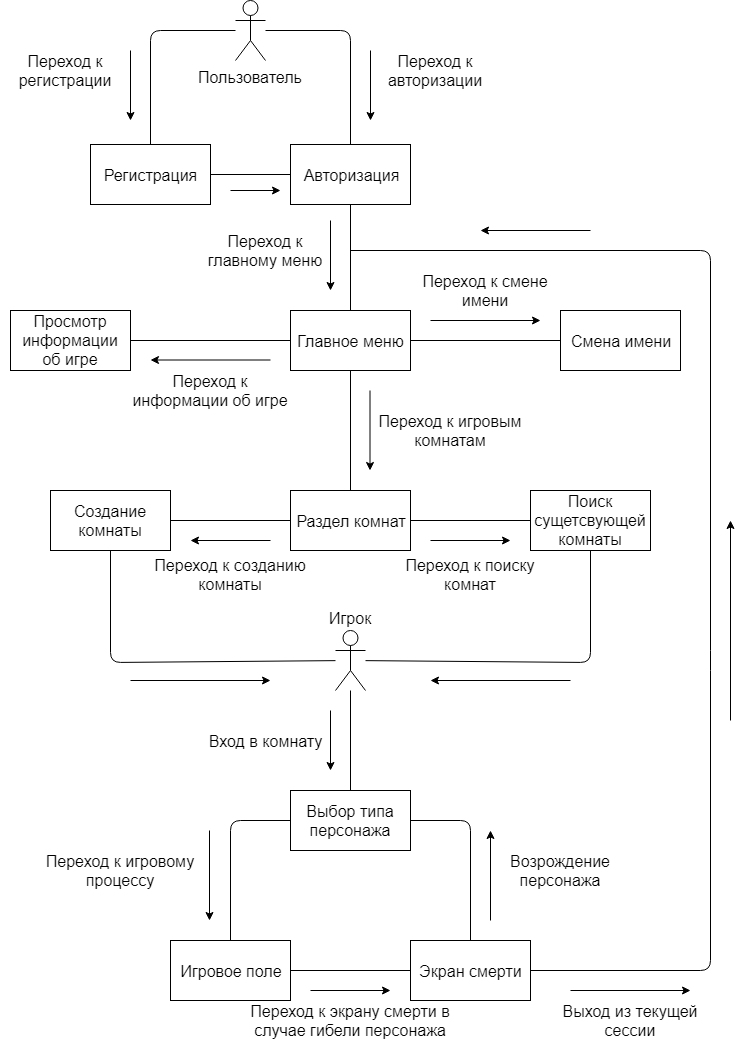


Рисунок 5. Диаграмма взаимодействия.

На рисунке изображена диаграмма, взаимодействие между объектами системы.

## Варианты состояния системы

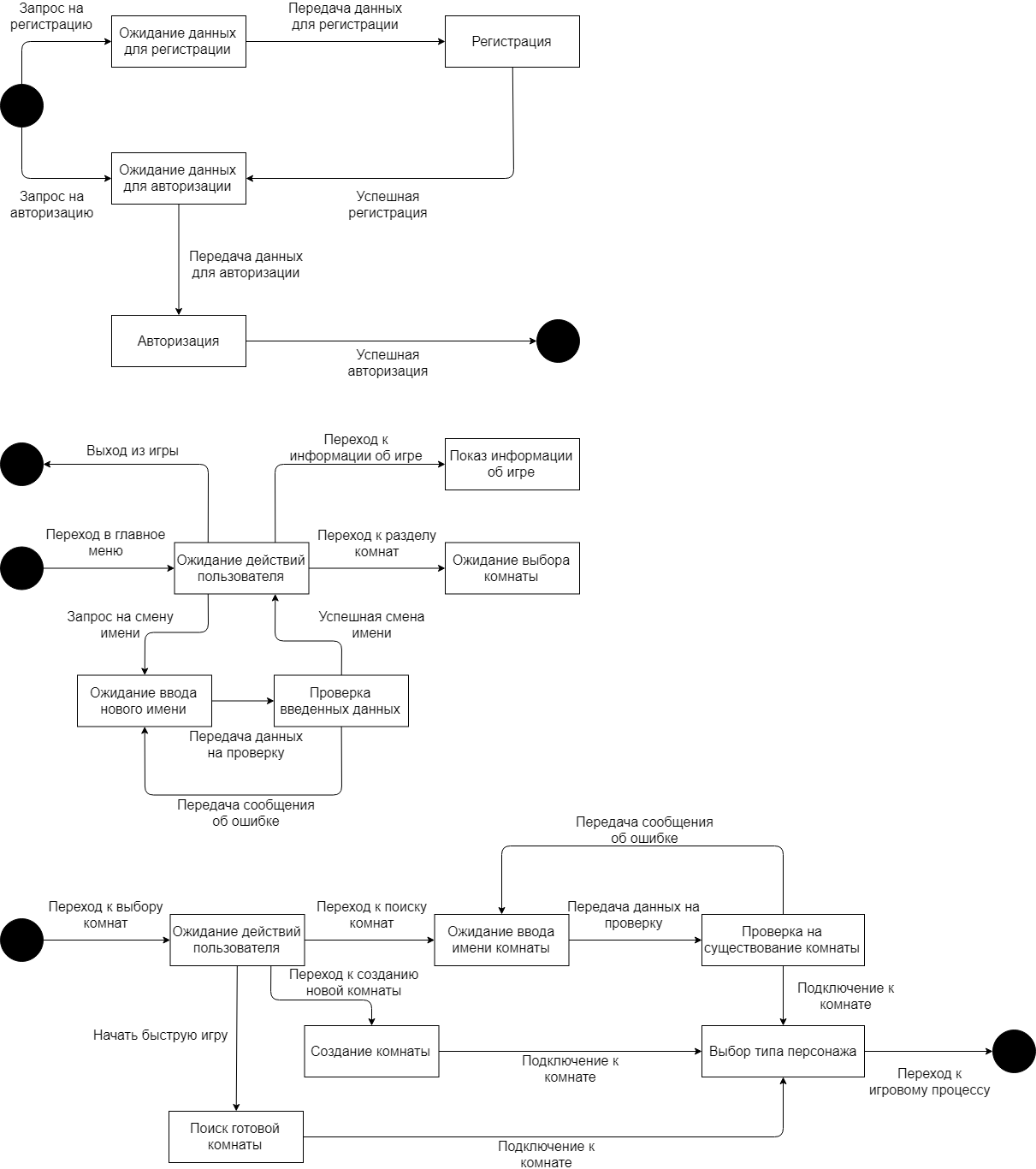


Рисунок 6. Диаграмма состояния системы.

При первом входе в приложение пользователь вводит имя своего игрового персонажа, при последующих входах этого делать не потребуется, так как это имя запомнится.

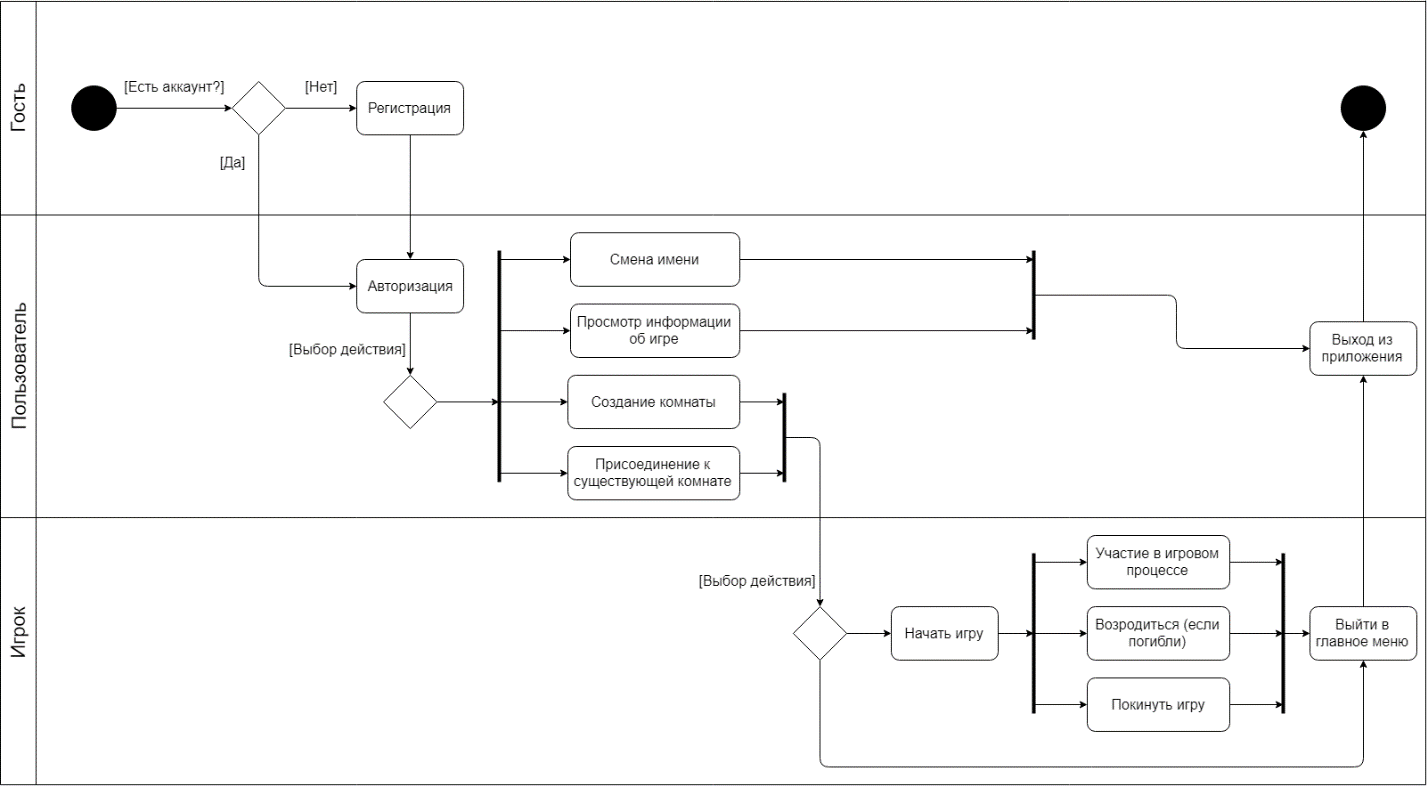
При повторном входе пользователь сразу попадает на главное меню, и имеет возможность изменить имя своего персонажа, посмотреть правила игры или перейти в меню комнат.

В меню комнат пользователь может создать комнату, введя следующие параметры: максимальное количество игроков, название комнаты и приватность комнаты. Если комната приватна, в нее можно попасть только через опцию присоединения к существующей комнате. Если комната публичная, в нее могут попасть пользователи, выбравшие в меню комнат опцию «Быстрая игра».

Также в меню комнат пользователь может присоединиться к уже существующей комнате, введя ее имя. Или же, выбрав опцию «Быстрая игра», попасть в комнату с наибольшим количеством участников, но не заполненную до конца.

Далее пользователь может выбрать из двух типов персонажей – травоядное и плотоядное. После выбора типа персонажа начинается игровой процесс.

## Действия с системой

Рисунок 7. Диаграмма активностей.

На рисунке изображена диаграмма активностей, которая показывает последовательность действий для перехода от одной деятельности к другой.

## Развёртывание

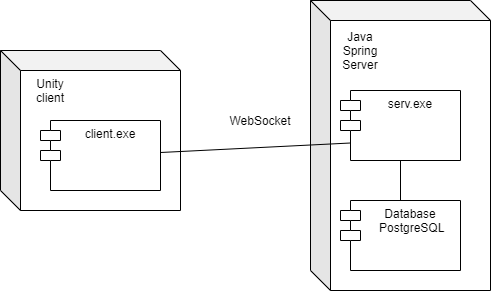


Рисунок 8. Диаграмма развёртываний.

## IDF0

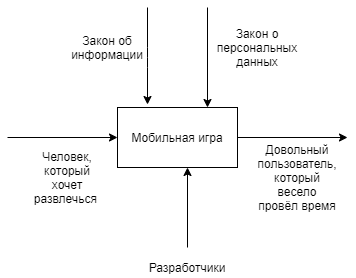


Рисунок 9. IDF0

На рисунке изображена IDEF0 диаграмма. Работу приложения регулируют Закон об информации и Закон о персональных данных, так как при регистрации в приложении пользователи сообщают mac-адреса своих устройств. Работу приложения обеспечивают разработчики. На вход в приложение поступает человек, который хочет развлечься. На выходе приложение выдает пользователя, который весело провёл время.