

При разработке приложения использовался язык программирования Java, который является одним из основных языков для создания Android-приложений. Java обеспечил удобное управление логикой приложения, обработку событий, работу с данными и реализацию основных функций взаимодействия между компонентами. Благодаря типизации и широкому набору библиотек Java позволил реализовать надёжное, структурированное и масштабируемое приложение.

Для создания пользовательского интерфейса использовался язык разметки XML. С помощью XML описана структура экранов приложения, размещение элементов управления, таких как кнопки, текстовые поля, списки и осуществлена настройка их внешнего вида.

В качестве основной среды разработки использовалась AndroidStudio – официальная IDE для Android-разработки от Google. AndroidStudio предоставила мощные инструменты для написания, отладки и тестирования кода. В процессе разработки использовалась функция автодополнения кода, визуальные редакторы макетов, система сборки Gradle и встроенные средства профилирования для оптимизации производительности приложения. Благодаря AndroidStudio процесс создания приложения стал значительно более эффективным и удобным.

### Литература

1 Java – сайт о языке программирования : [сайт]. – США, 1995–2025. – URL: <https://www.java.com> (дата обращения: 14.02.2025).

2 AndroidStudio – сайт о среде программирования : [сайт]. – США, 2000–2025. – URL : <https://developer.android.com> (дата обращения: 24.02.2025).

УДК 004.9

*Т. В. Земченко*

### ИСПОЛЬЗОВАНИЕ UNITY ДЛЯ РЕАЛИЗАЦИИ ИГРЫ «НИМ» С ИСПОЛЬЗОВАНИЕМ БЕСПРОВОДНЫХ ПЕРСОНАЛЬНЫХ СЕТЕЙ

*В статье рассматриваются вопросы использования современных информационных технологий для разработки одного из вариантов игры «Ним». Отмечаются возможные режимы игры: игра с ботом, игра через Bluetooth. Приводится описание игры и особенности её реализации. Описывается архитектура проекта и взаимодействие его компонентов. Осуждаются возможные решения реализации программного обеспечения.*

Современные игровые технологии предоставляют разработчикам широкие возможности для создания интерактивных и многопользовательских приложений. Одной из таких технологий является движок Unity – популярная кроссплатформенная среда разработки, позволяющая создавать как двухмерные, так и трёхмерные игры с возможностью подключения к различным внешним сервисам и устройствам [1].

Цель данной статьи – рассмотреть реализацию классической логической игры «Ним» в Unity с поддержкой двух режимов: игры против бота и игры по Bluetooth в рамках беспроводной персональной сети. Особое внимание уделяется интеграции Bluetooth-соединения в игровую механику и построению архитектуры взаимодействия между устройствами.

**Игра «Ним»: описание и особенности реализации.** Игра «Ним» – это классическая математическая игра, относящаяся к жанру абстрактных стратегических игр. Она

имеет простые правила, но требует от игроков логического мышления. Наиболее распространённый вариант предполагает наличие нескольких куч с предметами, игроки поочерёдно убирают любое количество предметов из одной кучи. Проигрывает тот, кто делает последний ход, оставив поле пустым.

Реализация игры в Unity начинается с моделирования игрового поля. Для удобства используется двухмерная сцена с визуализацией куч и предметов (например, камней или фишек), которые можно перемещать или удалять при взаимодействии игрока. Интерфейс включает основные элементы: отображение текущего состояния куч, кнопки для подтверждения хода, информацию о текущем игроке и сообщения о победе или поражении.

**Режим игры с ботом.** Одним из режимов игры является противостояние с ботом. В данном случае бот реализует строго определённый алгоритм, основанный на математических принципах выигрышных и проигрышных позиций. Используется стратегия, основанная на вычислении «ним-суммы» (суммы количества предметов в кучах). Если «ним-сумма» равна нулю, то позиция считается проигрышной, и бот делает случайный ход. В противном случае бот выбирает оптимальный ход, переводящий игру в проигрышное состояние для оппонента.

С точки зрения программной архитектуры, в этом режиме игра представляет собой локальную сессию, в которой все действия обрабатываются на одном устройстве. Пользовательский интерфейс, игровой цикл и логика бота интегрированы в единую систему на сцене Unity. Реализация такого режима позволяет игроку оттачивать свои навыки и изучать выигрышные стратегии.

**Реализация Bluetooth-соединения в Unity.** Второй режим игры – многопользовательская сессия через Bluetooth – требует реализации взаимодействия между двумя физическими устройствами. Поскольку Unity не предоставляет нативной поддержки Bluetooth, необходимо использовать сторонние библиотеки или плагины, позволяющие обеспечить подключение устройств по стандарту WPAN (Wireless Personal Area Network).

Для Android-платформы существует ряд решений, интегрирующих Bluetooth в проекты Unity. Такие плагины, как Android Bluetooth Plugin for Unity, позволяют инициализировать соединение, сканировать доступные устройства, устанавливать соединения и передавать данные между ними.

При разработке Bluetooth-режима один из пользователей выбирает роль хоста (ведущего), другой – клиента (подключающегося). После установления соединения передаётся информация о каждом ходе в формате сериализованных сообщений (например, JSON). Каждое устройство синхронизирует состояние игрового поля, основываясь на полученных данных.

Важным аспектом Bluetooth-соединения является надёжность и минимизация задержек. Разработчик должен предусмотреть обработку обрывов связи, повторные подключения и корректную синхронизацию состояния игры после восстановления соединения.

**Архитектура проекта и взаимодействие компонентов.** Структура проекта в Unity включает несколько основных компонентов:

- GameManager – основной управляющий скрипт, координирующий игровой процесс, переключение ходов, определение победителя;
- BluetoothManager – модуль, обеспечивающий взаимодействие с Bluetooth API, установку соединений и передачу данных;
- BotController – логика ИИ-оппонента (алгоритм без элементов машинного обучения);
- UIManager – отвечает за обновление интерфейса, отображение ходов, информации о победителе, статуса подключения.

В проекте используются стандартные Unity-компоненты (Canvas, Text, Button, Image) и системы событий (EventSystem), позволяющие обрабатывать клики и другие пользовательские действия. Особое внимание уделяется сериализации и десериализации данных: информация о ходе игрока должна быть передана партнёру по Bluetooth в корректной форме и обработана без задержек. Схема программного обеспечения представлена на рисунке 1.

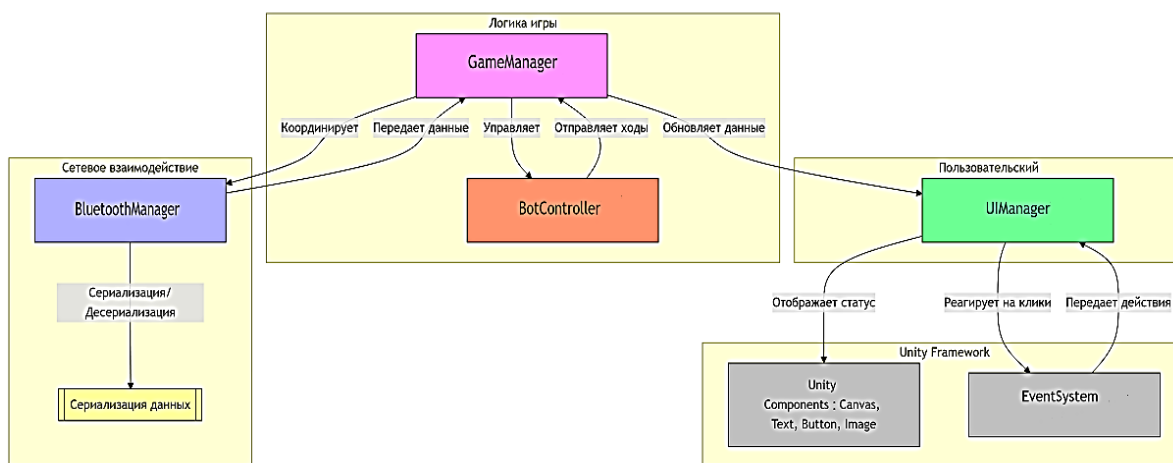


Рисунок 1 – Схема программного обеспечения

**Преимущества и особенности WPAN в мобильных играх.** Использование беспроводных персональных сетей в играх, таких как «Ним», открывает возможности для локального многопользовательского взаимодействия без необходимости в интернет-соединении. Это особенно актуально в условиях ограниченного доступа к сетевым ресурсам или при желании играть в офлайн-режиме.

WPAN-соединения (на базе Bluetooth) обладают рядом преимуществ:

- низкое энергопотребление;
- достаточная пропускная способность для обмена игровыми событиями;
- простота соединения между мобильными устройствами;
- независимость от внешней инфраструктуры (Wi-Fi, мобильная сеть).

Однако также существуют и ограничения: небольшой радиус действия, возможные помехи, необходимость согласования протоколов передачи. При реализации игры важно предусмотреть стабильную работу даже в случае нестабильного сигнала.

Реализация игры «Ним» в Unity с поддержкой как одиночного режима против бота, так и многопользовательского режима через Bluetooth, наглядно демонстрирует потенциал современных инструментов разработки для создания не только развлекательных, но и обучающих приложений. Благодаря использованию технологий беспроводных персональных сетей становится возможным обеспечить быстрое и надёжное локальное взаимодействие между пользователями без необходимости подключения к интернету. Это делает игру доступной и удобной для использования в различных условиях, включая образовательные учреждения, мероприятия или просто игру в компании друзей.

В процессе разработки такого проекта особое внимание уделяется не только точной реализации логики самой игры, но и проработке сетевого взаимодействия, архитектуры подключения, а также пользовательского опыта. Это включает в себя оптимизацию интерфейса, упрощение процесса подключения по Bluetooth, обеспечение надёжной передачи данных между устройствами и предотвращение возможных сбоев при нестабильном соединении.

Такой комплексный подход гарантирует создание устойчивой, логически завершённой и увлекательной игровой среды, которую можно легко адаптировать под различные платформы. Всё это делает игру «Ним» не только отличным учебным примером, но и полноценным продуктом, пригодным для публикации и дальнейшего развития.

## Литература

1 Тикоски, С. Современная разработка игр на Unity / С. Тикоски. – М. : БХВ-Петербург, 2024. – 496 с.