Министерство просвещения Российской Федерации

Управление образования и науки Тамбовской области

Комитет образования администрации города Тамбова Тамбовской области

Муниципальное автономное общеобразовательное учреждение «Лицей №14

имени Заслуженного учителя Российской Федерации А.М. Кузьмина»

**Индивидуальный проект**

**Игра "Длинные нарды"**

Автор работы:

Ученик 10К класса

Макаров Артём  
Научный руководитель:  
Вязовов С.М., заведующий кафедрой информатики, учитель информатики

**ТАМБОВ 2025**

Оглавление

[Введение 3](#_Toc194051719)

[Ход работы 4](#_Toc194051720)

[Выбор программных инструментов для разработки игры и нейронной сети 5](#_Toc194051721)

[Разработка игры «Длинные нарды» 6](#_Toc194051722)

[Версия v1.0(man vs man) 6](#_Toc194051723)

[Геймплей 10](#_Toc194051724)

[Проблемы, возникшие при разработке версии v1.0 11](#_Toc194051725)

[Версия v2.0(man vs bot) 11](#_Toc194051726)

[Проблемы, возникшие при разработке версии v2.0 15](#_Toc194051727)

[Результаты 16](#_Toc194051728)

[Заключение 16](#_Toc194051729)

[Список использованных источников 16](#_Toc194051730)

# **Введение**

Игра длинные нарды известна, если не всем, то многим. Немного о самой игре: это настольная игра для двоих игроков, в которой используются специальные доска и фишки. Цель игры заключается в том, чтобы переместить все свои фишки в доске и сначала убрать их. Игроки поочередно бросают два кубика, чтобы определить количество ходов. Длинные нарды требуют стратегии и навыков, прежде чем предугадывать действия соперника, что делает игру интересной и продуманной.

Хоть игра и появилась более 3500 лет назад, у неё до сих пор множество поклонников, а если человек не умеет играть, то научиться можно достаточно просто и быстро. Поэтому можно сказать, что длинные нарды актуальны сейчас и будут актуальны ещё долгое время. Однако не всегда есть возможность поиграть с другом, так как например не у всех людей есть игральная доска, а иногда нет и друга, с которым можно поиграть рядом нет. Эти две проблемы решаются с помощью моего приложения, в котором реализованы два режима игры: для игры с другом, и с «компьютером» (под компьютером понимается некоторый алгоритм, с помощью которого будет выбираться ход «виртуального» противника).

В последние годы нейронные сети, представляющие собой вид машинного обучения, стали особенно популярны. Этот тип обучения, известный как глубокое обучение, использует двойные узлы или нейроны, организованные в составную структуру, образующую человеческий мозг. Поскольку динамичная область программирования является инновационной и исследовательской, было решено создать собственную нейронную сеть и внедрить ее в приложение в формате «против компьютера».

# **Ход работы**

Работа над проектом была поделена на несколько задач:

1. Выбрать программные инструменты для разработки игры и нейронной сети
2. Изучение материала для разработки приложения
3. Разработка приложения: UI –пользовательский интерфейс и режим игры «человек против человека»
4. Изучение материала и разработка нейронной сети для режима «человек против компьютера»
5. Обучение нейронной сети
6. Исправление ошибок и доработка приложения

# **Выбор программных инструментов для разработки игры и нейронной сети**

Первоначально стоял вопрос о выборе языка программирования, который должен был соответствовать определённым требованиям: быть легко читаемым и компактным, иметь библиотеки для создания нейронных сетей и обеспечивать эффективную работу кода. Два языка удовлетворяли этим критериям: Python и Kotlin. Однако, поскольку я уже был знаком с основами Python и этот язык мне нравился, я решил выбрать именно его.

Далее мне нужно было выбрать программу для разработки приложения, которая должна была иметь удобный редактор кода и обеспечивать предварительную проверку приложения на ошибки. Программа PyCharm Community Edition полностью удовлетворяла этим требованиям, к тому же я пользовался ей раньше, поэтому я выбрал её.

# **Разработка игры «Длинные нарды»**

Я начал с изучения модуля PyQt5 в языке программирования Python, на котором и будут написаны «Длинные нарды». Изучение модуля проходило по документации модуля, параллельно написанию игры: я придумывал, что сделать, после чего искал в документации нужный материал, чтобы реализовать придуманную идею. Со временем я запоминал всё больше материала, который использовался для реализации функций в приложении, так сказать «набивал руку», что сильно сокращало время разработки, так как мне всё меньше надо было искать материал в документации. Однако без проблем и багов не обошлось.

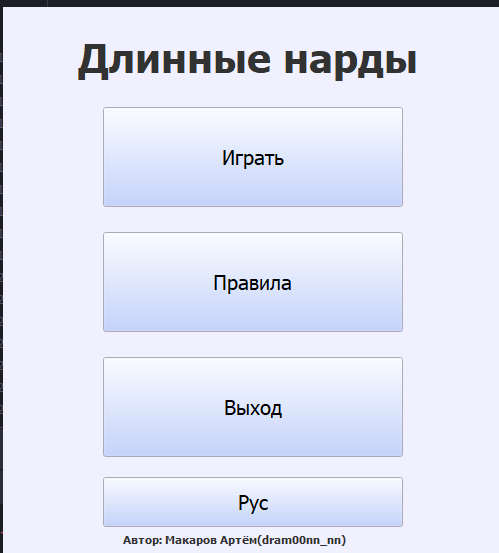
Была разработана версия игры v1.0

# **Версия v1.0(man vs man)**

Эта версия является самой первой, которой можно пользоваться. В ней реализован режим игры «человек против человека», а именно два игрока ходят поочерёдно до победы одного из них (в классических длинных нардах исходом не может являться ничья, что понятно из правил). Также был реализован UI, который включает в себя следующие элементы и функционал:

1. Главное меню игры (рис. 1 и 2)

Рис. 1

Рис. 2

* 1. Кнопка Eng (Рус) - переключение языка игры между Русским и Английским
  2. Кнопка Exit (Выход) - переход игрока в окно, где можно либо выйти из игры, если эта кнопка была нажата намеренно, либо вернуться в меню, если кнопка была нажата случайно (рис.3 и 4).

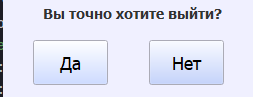
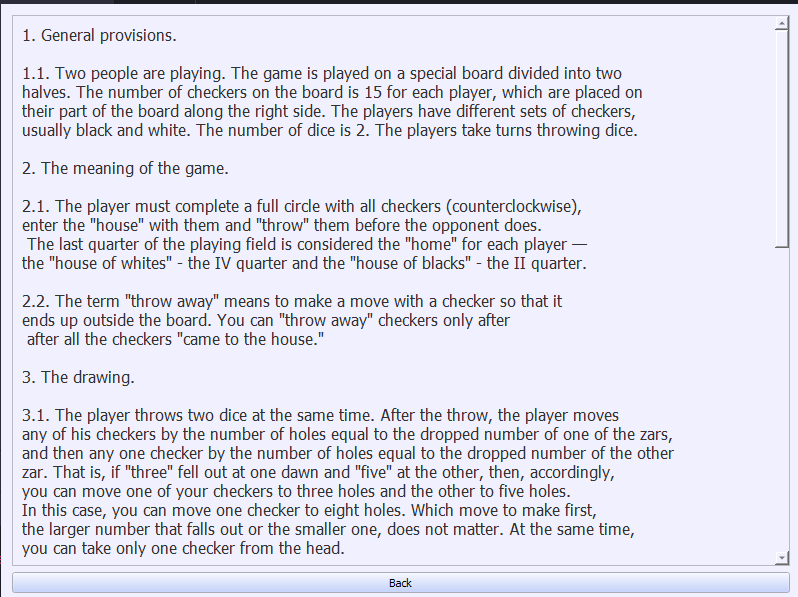
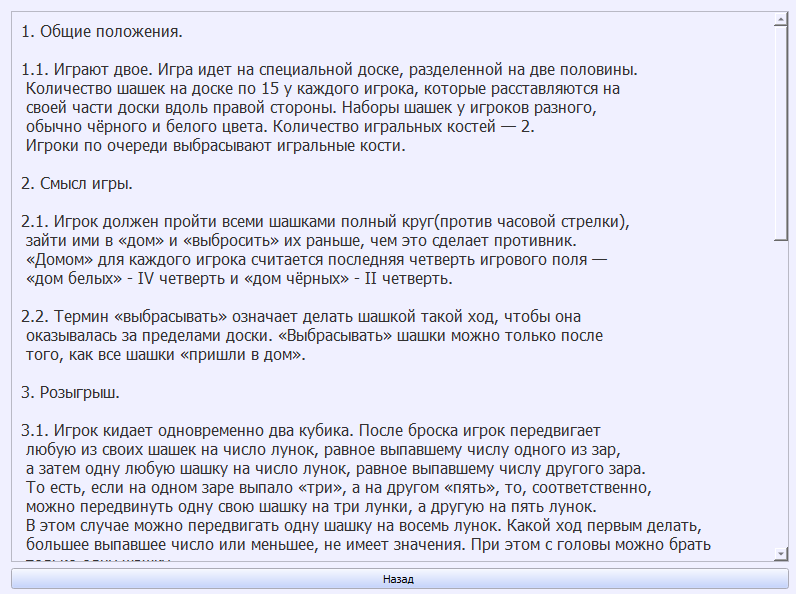
Рис. 4

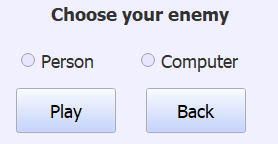
Рис. 3

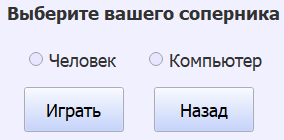
* 1. Кнопка Rules (Правила) - переход игрока в окно с правилами игры. Из-за того сами правила длинные, и при этом они различаются по длине в зависимости от выбранного языка, я реализовал окно с правилами с помощью QScrollArea, привязав длину текста к прокручиваемой области QscrollArea (рис. 5 и 6).

Рис.5

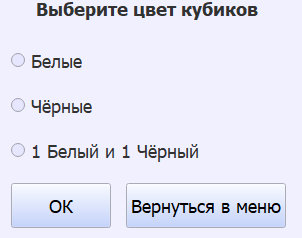
Рис. 6

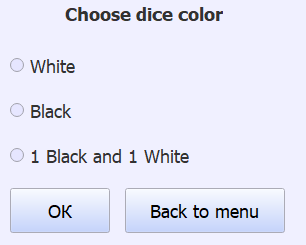
* 1. Кнопка Play (Играть) - переход игрока в окно выбора режима игры, который реализован через QRadioButton: «человек против человека» или «человек против компьютера». Режим игры против компьютера в этой версии не реализован, но такая настройка добавлена на будущие версии, где режим будет реализован. В открывшемся окне реализованы 2 кнопки: Play (Играть) – переход к следующей и последней настройке: выбор цвета кубиков, и кнопка Back (Назад) – возврат в меню (рис. 7 и 8).

Рис. 7

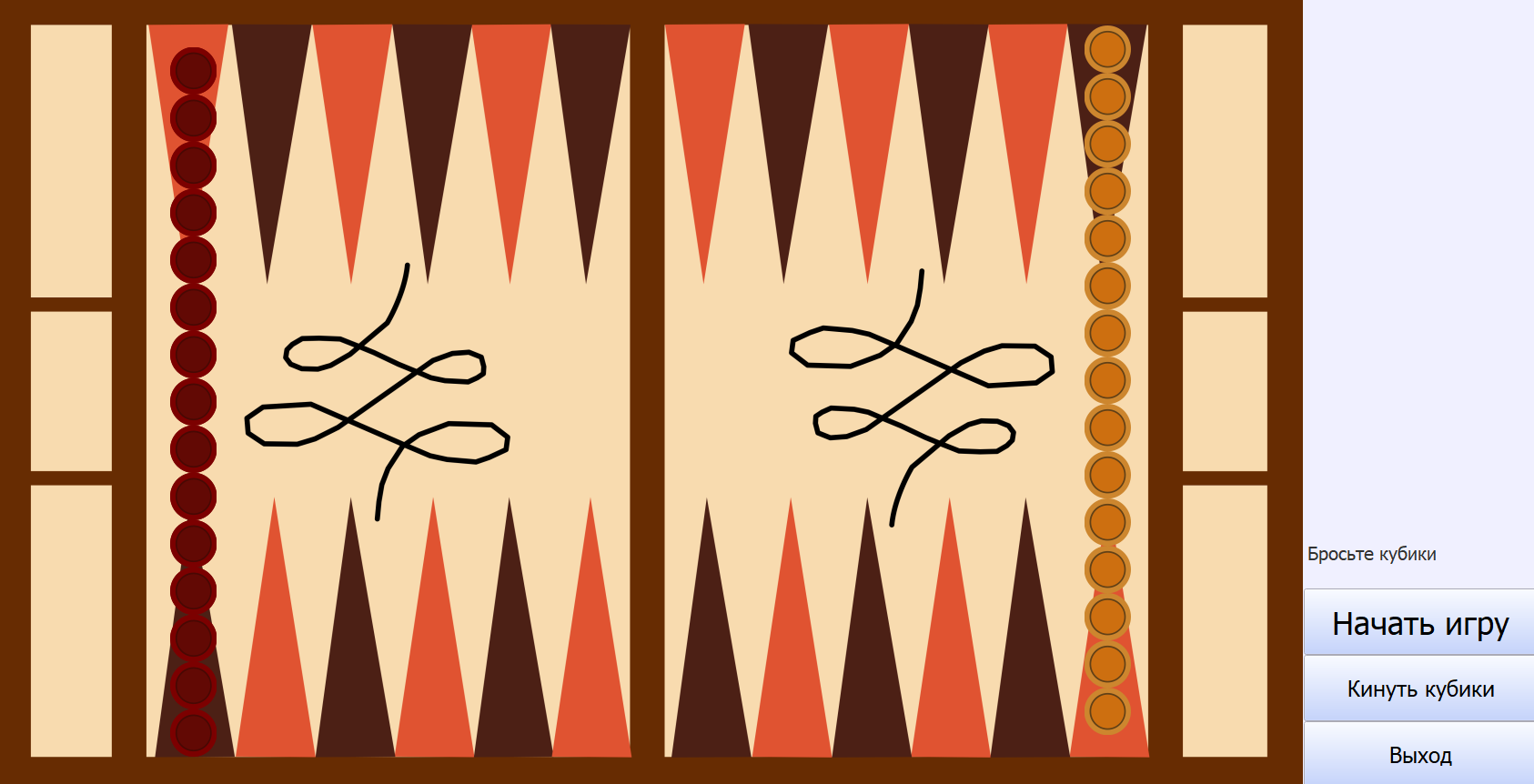
Рис. 8

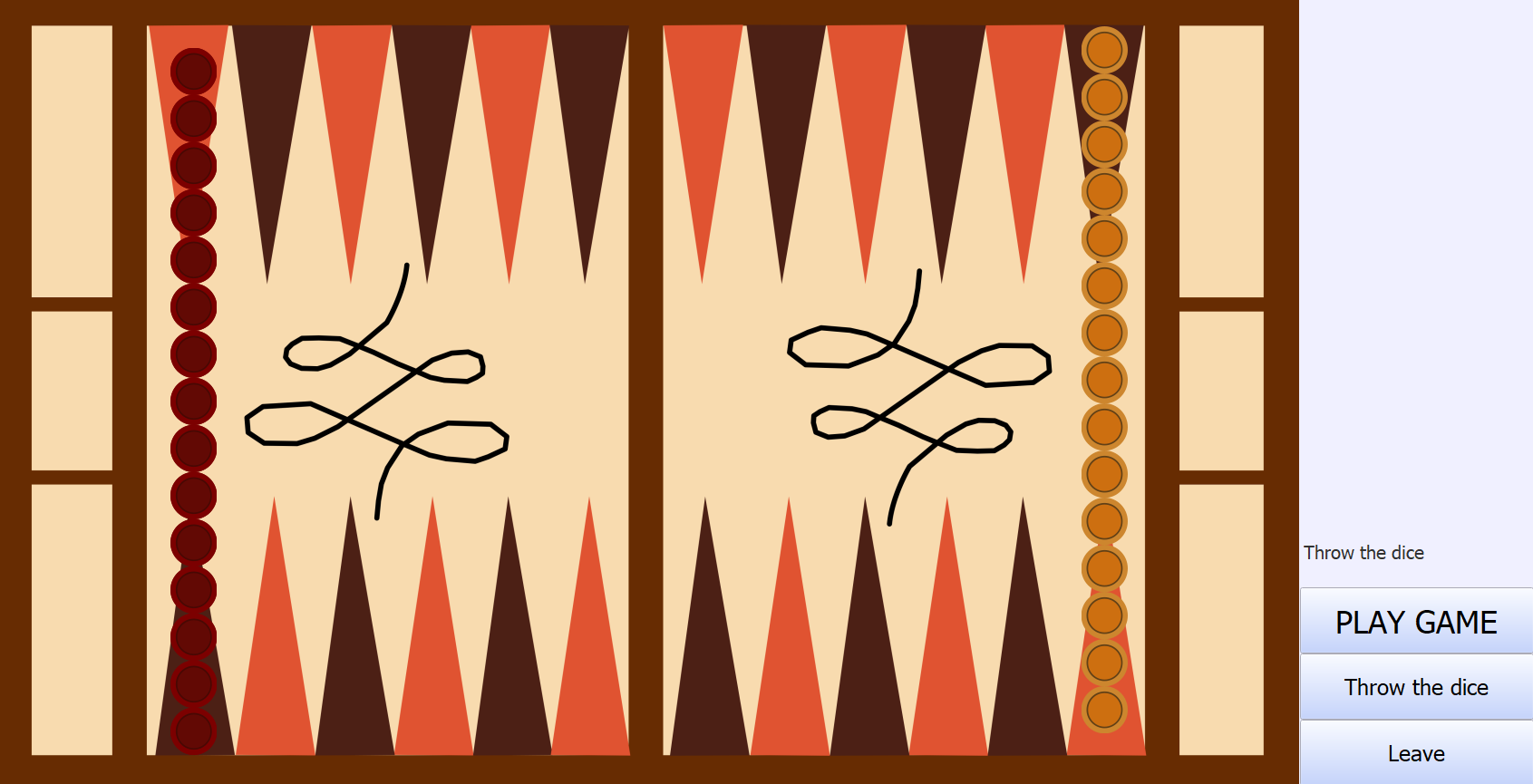
* 1. Окно выбора цвета кубиков: белые / черные / 1 белый и 1 черный (рис. 9 и 10)

Рис. 10

Рис. 9

1. Сама игра «Длинные нарды» (рис. 11 и 12)

Рис. 12

Рис. 11

2.1) Кнопка Play Game (Начать игру) – начало игры. Надпись, о которой говорится в пункте 2.4 оповестит о том, кто ходит первым. Выбор осуществляется через random.

2.2) Кнопка Throw the dice (Бросить кубики) – бросок кубиков. Эта кнопка блокируется, после нажатия, чтобы избежать выбивания игроком нужной ему комбинации выпавших на кубиках значений до того момента, пока текущий игрок не сделает ход.

2.3) Кнопка Leave (Выход) – выход в главное меню

2.4) Надпись над кнопками оповещает о том, какое действие нужно совершить.

2.5) Выпавшие кубики отображаются на панели справа (рис. 13-15)

# **Геймплей**

Для начала игры одному из игроков надо нажать кнопку Play game (Начать игру), после чего игра с использованием random выберет случайно игрока, который будет ходить первым. Ходы осуществляются нажатием на свою фишку (имеются ввиду фишки текущего игрока), если будет нажата вражеская фишка, игра проигнорирует это нажатие. После нажатия на свою фишку на игровое поле выведутся подсказки в виде бледно-зелёных фишек – те места, в которые может пойти игрок выбранной фишкой. Дальше два варианта исхода событий:

1. Игрок решит пойти выбранной фишкой. Тогда игроку следует нажать на фишку-подсказку, после чего фишка переместится, а ход засчитается (кубик, на значение которого переместили фишку, пропадёт с панели)
2. Игрок решит пойти другой фишкой. Тогда следует нажать на другую свою фишку. После чего «два варианта исхода событий»

После чего, если у игрока остался кубик, он может сделать ещё один ход своей фишкой, и опять будут те же «два варианта исхода событий». Если же у игрока не осталось кубиков, то ход переходит к следующему игроку. Следующий игрок должен будет бросить кубики, о чем уведомит игра на панели.

И так до момента, пока не начнется «финал» игры – выбрасывание фишек с доски. Далее геймплей схож с тем, что был до «финала», то есть игроки будут поочерёдно ходить, но теперь появляется новая механика – выбрасывание. Опять же через фишки-подсказки может осуществляться выбрасывание.

Параллельно описывая геймплей, я кратко объяснил правила «на пальцах». Подробные правила написаны в самой игре: Menu(Меню) 🡪 Rules (Правила).

# **Проблемы, возникшие при разработке версии v1.0**

Как упоминалось ранее, у меня возникли проблемы при разработке этой версии, а именно функционал, который я реализовывал, всё время ловил ошибки и был багованным, над исправлением багов по ощущению было проведено 40% времени разработки этой версии. Итак, возникавшие проблемы можно разделить на 2 блока:

1. UI. Достаточно немало времени уделялось проработке UI, ведь просто рабочая игра, которая плохо выглядит вызывает отвращение. Проблемы в основном возникали в расположении элементов интерфейса и их отображением: кнопок и текста
2. Реализация внутренней составляющей игры. Были как небольшие проблемы, возникавшие в основном из-за невнимательности или передача неправильных параметров, которые решались быстро, так и большие, например, несовместимость функций и методов, просто методов, зацикливание игры, проработка механик: броски кубиков, реализация ходов.

# **Версия v2.0(man vs bot)**

Эта версия включает в себя уже два режима игры: «человек против человека» и «человек против бота». Режим игры «человек против человека» был доработан, основными изменениями стали исправление багов с отображением и улучшенная логика игры. Режим «человек против бота» был реализован с использованием дерева принятия решений, которое принимает на вход «глубину дерева», то есть его сложность: easy(легкая) – 3, medium(средняя) – 5 и hard(сложная) – 7. В этом решении есть недостаток: чем дальше зашла игра, и чем выше сложность, тем дольше бот думает над ходом. Также, как показали тесты бота, иногда он не воспроизводит ход, а просто отдает его игроку, причину данного поведения я выявить не смог, следовательно, и исправить это я не смог, однако, бот думает и ходит без ошибок и крашей. В этой версии игры также был улучшен UI и доработаны некоторые ошибки и недоделанные элементы интерфейса и функционала. Эта версия игры запускается через уже как .exe приложение. Текущий UI сохранил весь функционал из версии v1.0 и выглядит следующий образом:

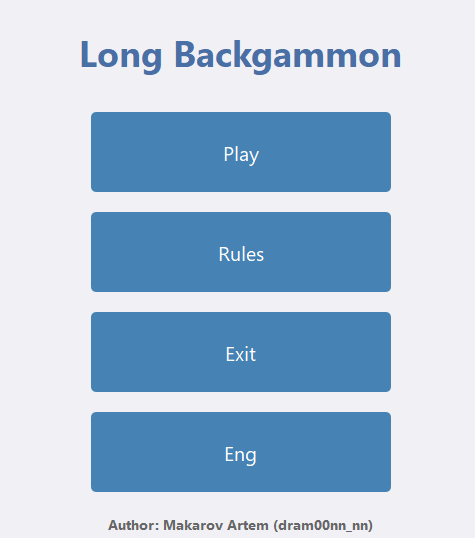
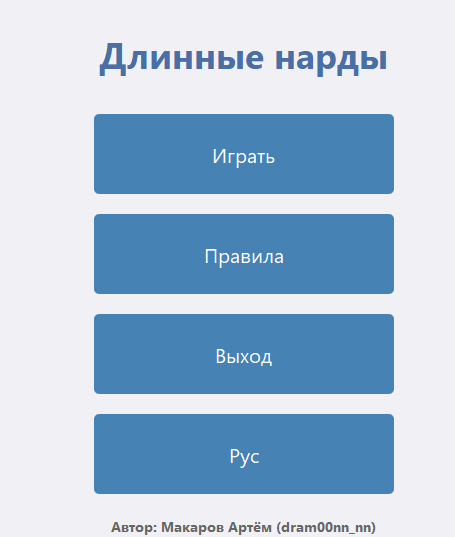
1. Главное меню игры (рис. 13 и 14)

Рис. 4

Рис. 13

1. Окно выхода из игры(рис 15 и 16)

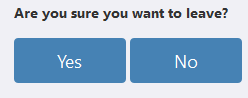
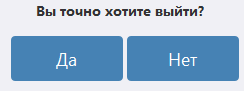


Рис. 16

Рис. 15

1. Окно с правилами(рис 17 и 18)

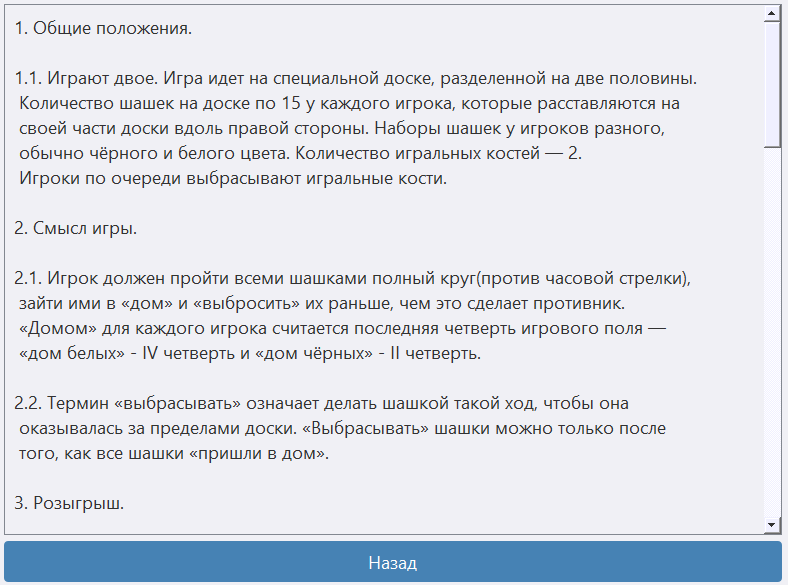
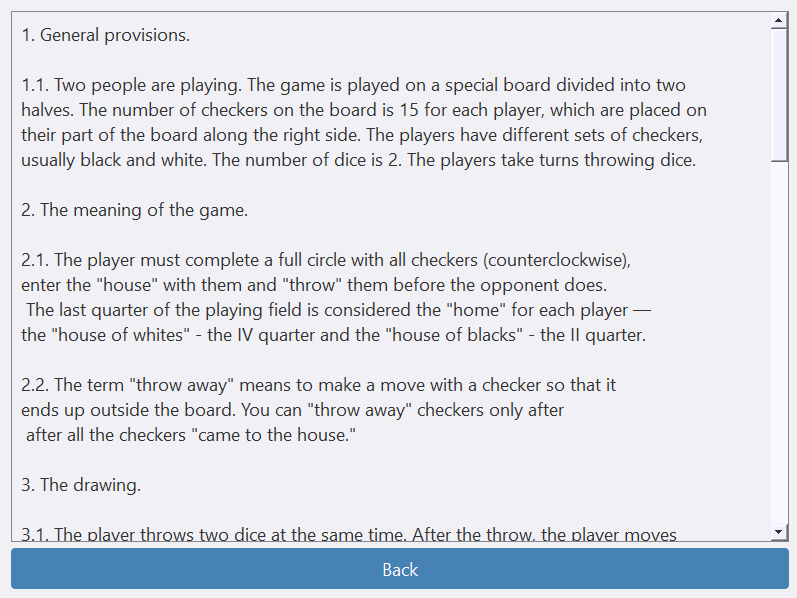


Рис. 17

Рис. 18

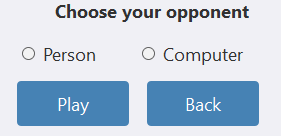
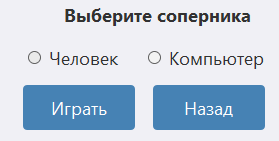
1. Окно выбора режима игры(рис 19 и 20)

Рис. 20

Рис. 19

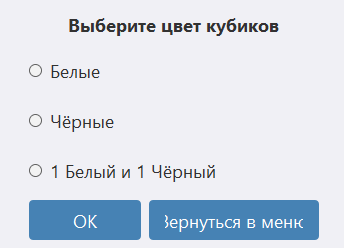
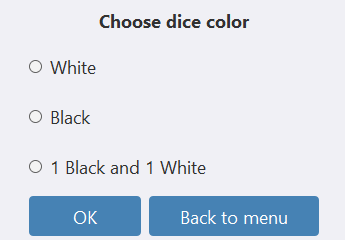
1. Окно выбора цвета кубиков(рис. 21 и 22)

Рис. 2

Рис. 21

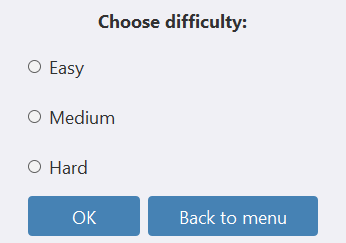
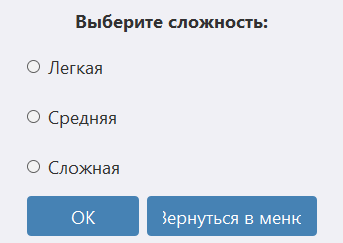
1. Окно выбора режима сложности для игры против бота, появляется только при игре с ботом(рис. 23 и 24)

Рис. 24

Рис. 23

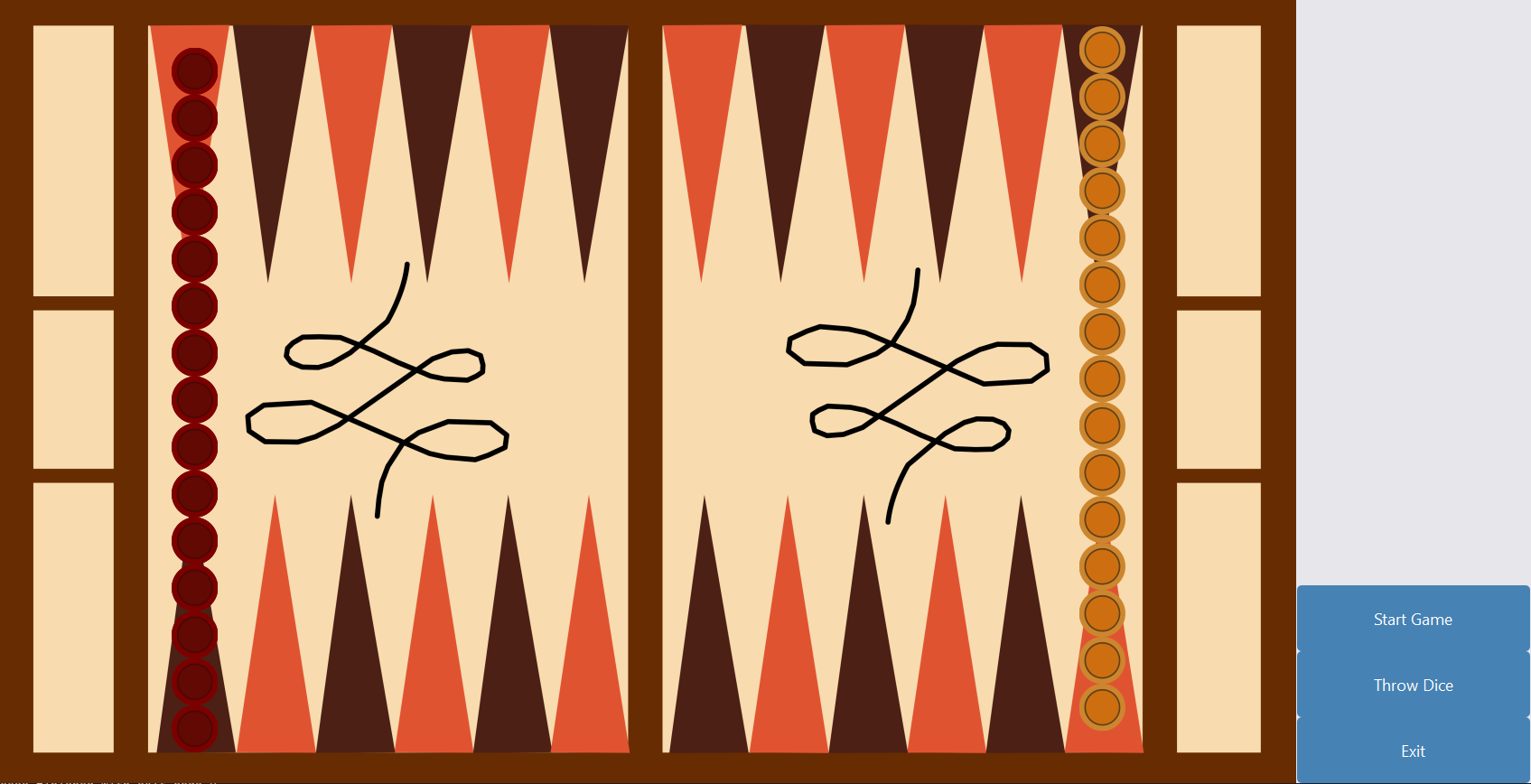
1. Сама игра «Длинные нарды» (рис. 25 и 26)

Рис. 25

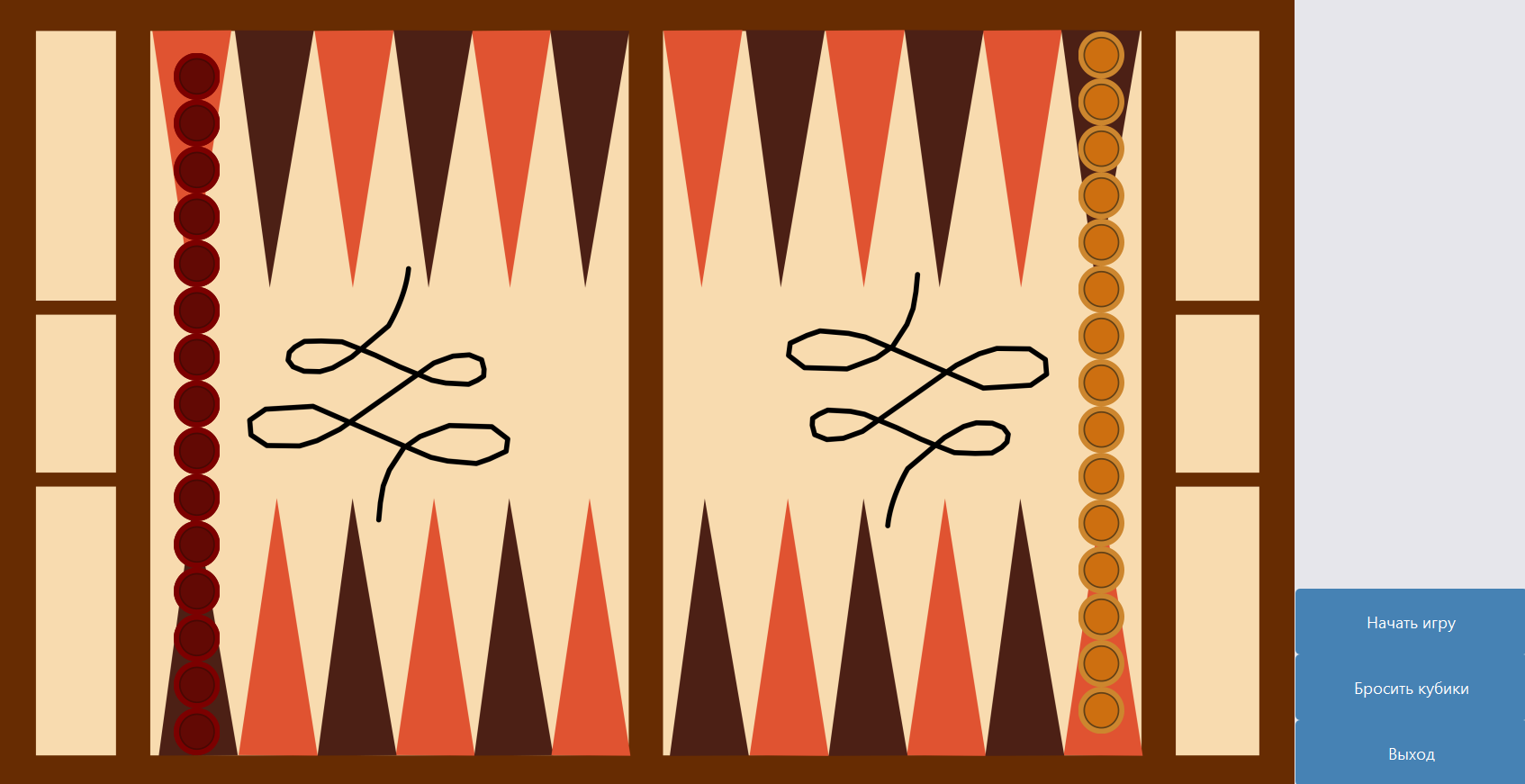


Рис. 26

# **Проблемы, возникшие при разработке версии v2.0**

Во время разработки этой версии игры возникло достаточно немало проблем по той причине, что по началу я хотел написать нейронную сеть, которая будет анализировать ситуацию на игровом поле и делать ходы, основываясь на предыдущих ходах, но, к сожалению, ничего толкового из этого не получилось, из-за чего я принял решение остановиться на дереве принятия решений, которое изначально было задумано. Я потратил много времени как на изучение материалов про нейронные сети для разработки своей, что не увенчалось успехом, так и на изучение материалов про дерево принятия решений, которое работает по принципу minmax - максимизация своих ходов и минимизация ходов соперника, из чего получилось то, что умеет бот сейчас. Изначально я написал дерево принятия решений, как отдельный class, но из-за проблем с доступом, я решил, что не составит проблемы реализовать дерево принятия решений не как отдельный class, а встроить его в class Game, то есть в саму игру, что оказалось удачным решений, ведь проблем с доступом теперь не было. Также проблемой оказалось различие поведения методов в режиме против бота и против игрока, а именно методы, которые работали в режиме против игрока, работали некорректно в режиме против бота, из-за чего пришлось значительно переписать их, чтобы всё работало для обоих режимов.

# **Результаты**

1. Был изучен модуль PyQt
2. Первый опыт разработки приложения
3. Изучена и использована на практике теория по дереву принятия решений

# **Заключение**

Из поставленных задач выполнены все, кроме шестой поставленной задачи, которая заключается в доработке приложения и исправлении ошибок. В приложении есть недоработки, которые мне исправить, к сожалению, пока не удалось. Также, в связи с тем, что у меня абсолютно нет опыта в создании UI, интерфейс выглядит не идеально, что тоже надо будет доработать. В ходе выполнения первых пяти задач я получил реализацию игры «Длинные нарды» с двумя режимами игры: «человек против человека» и «человек против бота», где реализовано дерево принятия решений через minmax в режиме игры против бота.

# **Список использованных источников**

1. <https://habr.com/ru/companies/productstar/articles/523044/>
2. <https://ru.wikipedia.org/wiki/%D0%9C%D0%B8%D0%BD%D0%B8%D0%BC%D0%B0%D0%BA%D1%81>
3. https://habr.com/ru/companies/otus/articles/785512/
4. <https://doc.qt.io/qtforpython-5/>
5. Множество статей на <https://habr.com/>
6. Множество обсуждений на <https://ru.stackoverflow.com/>