Нижегородский государственный технический университет им. Р.Е. Алексеева

Институт радиоэлектроники и информационных технологий

Кафедра «Электроника и сети ЭВМ»

**отчет**

по дисциплине: «Архитектура информационных систем»

Тема: «Разработка программного обеспечения «Крестики – Нолики»»

Руководитель

Ассистент кафедры ЭСВМ,

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Егоров Ю.С.

Нижний Новгород

2014 г.

СОДЕРЖАНИЕ

[Введение 4](#_Toc388986641)

[1. Описание программного обеспечения 5](#_Toc388986642)

[1.1. Назначение ПО 5](#_Toc388986643)

[1.2. Функции ПО 5](#_Toc388986644)

[2. Организация процесса разработки ПО 6](#_Toc388986645)

[2.1. Распределение ответственности в процессе создания ПО 6](#_Toc388986646)

[2.2. График реализации мероприятий жизненного цикла ПО 8](#_Toc388986647)

[3. Проектирование программного обеспечения 9](#_Toc388986648)

[3.1. Описание процесса разработки требований к ПО 9](#_Toc388986649)

[3.2. Описание процесса кодирования 9](#_Toc388986650)

[3.3. Описание процесса тестирования ПО 10](#_Toc388986651)

[4. Описание проекта ПО 11](#_Toc388986652)

[4.1. Описание требований к ПО 11](#_Toc388986653)

[4.2. Описание архитектуры ПО 13](#_Toc388986654)

[4.3. Описание алгоритмов работы ПО 14](#_Toc388986655)

[5. Тестирование ПО 22](#_Toc388986656)

[5.1. Объем тестирования ПО 22](#_Toc388986657)

[5.2. Результаты тестирования ПО 32](#_Toc388986658)

[6. Руководство пользователя 38](#_Toc388986659)

[Об игре «Крестики-нолики» 38](#_Toc388986660)

[Минимальные системные требования 38](#_Toc388986661)

[Начало игры 38](#_Toc388986662)

[Интерфейс 39](#_Toc388986663)

[Выбор режима игры 39](#_Toc388986664)

[Выбор уровня сложности 39](#_Toc388986665)

[Игра «Крестики-нолики»: основы 40](#_Toc388986666)

[Цель игры 40](#_Toc388986667)

[Игровое поле 40](#_Toc388986668)

[Правила игры 40](#_Toc388986669)

[Заключение 41](#_Toc388986670)

[Приложение 1. Исходный код ПО 42](#_Toc388986671)

**Список исполнителей**

1. Авдеев Илья Анатольевич, 11-СБК, 1 подгруппа, руководитель проекта

2. Новожилова Валерия Михайловна, 11-СБК, 1 подгруппа, дизайнер

3. Сазонов Иван Михайлович, 11-СБК, 1 подгруппа, инженер по требованиям

4. Рябчиков Алексей Андреевич, 11-СБК, 1 подгруппа, системный инженер

5.Мурындин Станислав Юрьевич, 11-СБК, 1 подгруппа, инженер – программист

6. Козлов Сергей Владимирович, 11-СБК, 1 подгруппа, инженер по тестированию

7. Касатов Егор Сергеевич, 11-СБК, 1 подгруппа, тестировщик

# Введение

**Цель работы:** разработать программное обеспечение «Крестики - Нолики»

**Задачи:** создать программное обеспечение, имитирующее логическую игры «Крестики – Нолики», имеющее два уровня сложности – «Легкий» и «Сложный» (в режиме «Человек против «Компьютера»), а также режим «Человек против Человека». В качестве дополнительного ограничения установить ограничение времени на ход в 15 секунд.

**Ключевые слова:** крестики – нолики, бот, игра, логика, друг против друга, человек против человека, крестик, нолик.

# 1. Описание программного обеспечения

## 1.1. Назначение ПО

Разрабатываемое программное обеспечение «Крестики – Нолики» представляет собой имитацию логической игры «Крестики – Нолики».

## 1.2. Функции ПО

Программное обеспечение «Крестики – Нолики» выполняет следующие функции:

* Возможность играть с компьютером.
* Возможность играть с человеком.
* Возможность выбора требуемого уровня сложности при игре с компьютером.
* Отображение текущей информации о ходе игры;

# 2. Организация процесса разработки ПО

## 2.1. Распределение ответственности в процессе создания ПО

|  |  |  |  |
| --- | --- | --- | --- |
| **№**  **п/п** | **Наименование роли** | **Ответственный** | **Содержание работы** |
| 1 | *Руководитель проекта* | Авдеев И.А. | *мониторинг хода выполнения проекта, составление отчетной документации, выполнение функции основного связующего звена между заказчиком и командой проекта.* |
| 2 | *Системный инженер* | Рябчиков А.А. | *разрабатывает архитектуру системы программного обеспечения, разрабатывает общий алгоритм функционирования и технологию обработки информации, согласовывает взаимодействие и увязку компонент программного обеспечения.* |
| 3 | *Инженер по требованиям* | Сазонов И.М. | *описывает процесс управления требованиями, разрабатывает спецификации требований, анализирует требования заказчика к разрабатываемому программному продукту.* |
| 4 | *Дизайнер* | Новожилова В.М. | *утверждение базовых планов и концепции дизайна интерфейса, подбор цвета и оформления с учетом психологических критериев,* *исследование и анализ целей и предпочтений заказчика, наряду с разработкой дизайнерских документов, рисунков и диаграмм, отражающих эти нужды.* |
| 5 | *Инженер-программист* | Мурындин С.Ю. | *разрабатывает, отлаживает, анализирует и оптимизирует программный код на основе готовых спецификаций, проводит отладку разработанных программ, корректирует их в процессе доработки, принимает участие в разработке форм документов, подлежащих машинной обработке, в проектировании программ, позволяющих расширить область применения вычислительной техники.* |
| 6 | *Инженер по тестированию* | Козлов С.В. | *разрабатывает простые тестовые наборы и тестовые процедуры, настраивает тестовое окружение согласно установленным требованиям, принимает участие в разработке и внедрении типовых и стандартных программных средств.* |
| 7 | *Тестировщик* | Касатов Е.С. | *тестирует отдельные простые компоненты программного обеспечения, анализирует и разрешает возникающие в системе проблемы и сбои, осуществляет тестирование программного продукта в соответствии с протоколами тестирования, инструкциями и стандартами качества для обеспечения функционирования систем в соответствии со спецификацией.* |

## 2.2. График реализации мероприятий жизненного цикла ПО



*Условные обозначения:*

|  |  |
| --- | --- |
|  | Контрольная точка. Показ результатов заказчику. Присутствуют указанные люди. |
|  | Режим "ожидания". |
|  | Отправка промежуточного результат заказчику для оценки правильности выполнения работ. |
|  | Показ результатов руководителю проекта с целью оценки выполненной работы и формирования промежуточного отчета для отправки заказчику. |

# 3. Проектирование программного обеспечения

## 3.1. Описание процесса разработки требований к ПО

Требования написаны с помощью Microsoft Word 2010. Каждое требование хранится в отдельном файле и обозначается названием требования (например, FR-SWR-1-1-1). Для удобства поиска, просмотра и редактирования требований имеется спецификация, в которой собраны названия, имя автора, дата создания и дата последней редакции всех требований, а также их расположение, и установлены гиперссылки на них с помощью инструментов Microsoft Word 2010.

Написание требования начинается с обозначения. Для примера рассмотрим функциональное требование: FR-SWR-1-16-2 - Цвет крестиков. Так как это функциональное требование, то обозначается FR-SWR. Цифры 1-16-2 означают вторую версию шестнадцатого требования из группы функциональных требований к интерфейсу. Также в каждом требовании указывается автор, дата создания и дата последней редакции, после чего идет сам текст требования и ссылка на тест к нему.

## 3.2. Описание процесса кодирования

*Преимущества C# перед C++*

* наличие сборщика мусора (это очень важно для больших проектов, которые работают продолжительное время);
* Удобства в написании;
* Поддержка и развитие языка фирмой Microsoft (а т.к. VS тоже их творение, С++ скорее будет в угнетенном положении);
* Отсутствие указателей;
* передача (целых) объектов между модулями;
* Управляемый код на выходе (т.е. выполняется чисто на .net машине и понимается ей без лишних преобразований).

*Преимущества C# перед Delphi*

* Расширяемость системы (в С# можно спокойно подгружать любые exe, импортировать классы и объекты из других программ);
* Сложность разработки и сопровождения (подбор кадров, читаемость кода, документированность языка);
* Степень открытости исходных текстов библиотек, исполняемых программ, количество литературы и помощь (MSDN);
* Возможность привлечения сторонних разработчиков при разработке системы для программирования узкоспециализированных задач (сборки, модули, те же .exe);
* Защищенность и контроль версий подключаемых алгоритмов (концепция NET);
* Трудоемкость написания (тот же NET);
* Скорость работы;
* Удобство разработки (удобная среда разработки).

На основании приведенных преимуществ языка программирования C#, он был выбран для написания нашего проекта.

C# (произносится си шарп) — объектно-ориентированный язык программирования. Разработан в 1998—2001 годах группой инженеров под руководством Андерса Хейлсберга в компании Microsoft как язык разработки приложений для платформы Microsoft .NET Framework и впоследствии был стандартизирован как ECMA-334 и ISO/IEC 23270.

C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык имеет статическую типизацию, поддерживает полиморфизм, перегрузку операторов (в том числе операторов явного и неявного приведения типа), делегаты, атрибуты, события, свойства, обобщённые типы и методы, итераторы, анонимные функции с поддержкой замыканий, LINQ, исключения, комментарии в формате XML.

Переняв многое от своих предшественников — языков C++, Pascal, Модула, Smalltalk и в особенности Java — С#, опираясь на практику их использования, исключает некоторые модели, зарекомендовавшие себя как проблематичные при разработке программных систем.

## 3.3. Описание процесса тестирования ПО

Каждый тест представляет собой отдельный файл, написанный с помощью Microsoft Excel 2007. Каждый тест имеет свое уникальное обозначение, которое несет в себе определенную информацию.

Рассмотрим, например, тест " tNFR SWR-2-2" ("Сбои. Режим "Игра с человеком"). Обозначение "t" говорит о том, что данный файл является тестом, "tNFR SWR-2"-свидетельствует о том, что данный тест относится к группе нефункциональных требований подгруппы атрибута качества ПО, а "2"-является уникальным номером теста.

Для удобства поиска, просмотра и редактирования все тесты разбиты на группы, в соответствии с группами требований к которым они относятся, и хранятся в соответствующих папках. В каждом тесте указываются название теста, автор, дата создания, дата последней правки, ссылка на требование, исходные данные, предусловия, тело теста, постусловие, точка управления и наблюдения, ожидаемый результат и комментарий. Предусловия, тело теста и постусловие представляют собой конкретную последовательность действий, которые необходимо выполнить при прохождении теста.

Все тесты разработаны строго по разработанным требованиям, неоднократно редактировались при каждом изменении версии требования. Для удобства и упрощения работы для одного требования составлялось несколько тестов (Например, для каждого режима игры и уровня сложности).

# 4. Описание проекта ПО

## 4.1. Описание требований к ПО

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№ П/п** | **Обозначение** | **Название** | **Ссылка на документ** | **Дата создания** | **Дата последней правки** | **Автор** |
| 1 | FR-SWR-1-1-1 | Части окна | [FR-SWR-1-1-1](FR-SWR-1-1-1.docx) | 21.04.2014 | 21.04.2014 | Сазонов И.М. |
| 2 | FR-SWR-1-2-1 | Общий размер окна | [FR-SWR-1-2-1](FR-SWR-1-2-1.docx) | 21.04.2014 | 21.04.2014 | Сазонов И.М. |
| 3 | FR-SWR-1-3-1 | Размер верхней части | [FR-SWR-1-3-1](FR-SWR-1-3-1.docx) | 21.04.2014 | 21.04.2014 | Сазонов И.М. |
| 4 | FR-SWR-1-4-2 | Размер средней части | [FR-SWR-1-4-2](FR-SWR-1-4-2.docx) | 21.04.2014 | 13.05.2014 | Сазонов И.М. |
| 5 | FR-SWR-1-5-2 | Размер нижней части | [FR-SWR-1-5-2](FR-SWR-1-5-2.docx) | 21.04.2014 | 13.05.2014 | Сазонов И.М. |
| 6 | FR-SWR-1-6-1 | Содержимое верхней части | [FR-SWR-1-6-1](FR-SWR-1-6-1.docx) | 21.04.2014 | 21.04.2014 | Сазонов И.М. |
| 7 | FR-SWR-1-7-1 | Кнопка меню | [FR-SWR-1-7-1](FR-SWR-1-7-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 8 | FR-SWR-1-8-1 | Кнопка настройки | [FR-SWR-1-8-1](FR-SWR-1-8-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 9 | FR-SWR-1-9-1 | Содержимое средней части | [FR-SWR-1-9-1](FR-SWR-1-9-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 10 | FR-SWR-1-10-2 | Фон | [FR-SWR-1-10-](FR-SWR-1-10-2.docx)2 | 26.04.2014 | 12.05.2014 | Сазонов И.М. |
| 11 | FR-SWR-1-11-1 | Цвет текста кнопки «Старт» | [FR-SWR-1-11-](FR-SWR-1-11-1.docx)1 | 12.05.2014 | 12.05.2014 | Сазонов И.М. |
| 12 | FR-SWR-1-12-3 | Содержимое нижней части | [FR-SWR-1-12-3](FR-SWR-1-12-3.docx) | 26.04.2014 | 12.05.2014 | Сазонов И.М. |
| 13 | FR-SWR-1-13-2 | Цвет надписей | [FR-SWR-1-13-2](FR-SWR-1-13-2.docx) | 26.04.2014 | 13.05.2014 | Сазонов И.М. |
| 14 | FR-SWR-1-14-1 | Состав сетки | [FR-SWR-1-14-1](FR-SWR-1-14-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 15 | FR-SWR-1-16-2 | Цвет крестиков | [FR-SWR-1-16-2](FR-SWR-1-16-2.docx) | 26.04.2014 | 28.04.2014 | Сазонов И.М. |
| 16 | FR-SWR-1-17-2 | Цвет ноликов | [FR-SWR-1-17-2](FR-SWR-1-17-2.docx) | 26.04.2014 | 28.04.2014 | Сазонов И.М. |
| 17 | FR-SWR-1-18-3 | Размер ноликов | [FR-SWR-1-18-3](FR-SWR-1-18-3.docx) | 26.04.2014 | 13.05.2014 | Сазонов И.М. |
| 18 | FR-SWR-1-19-2 | Размер крестика | [FR-SWR-1-19-2](FR-SWR-1-19-2.docx) | 26.04.2014 | 13.05.2014 | Сазонов И.М. |
| 19 | FR-SWR-1-20-1 | Кнопка меню: «правила» | [FR-SWR-1-20-1](FR-SWR-1-20-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 20 | FR-SWR-1-21-2 | Всплывающее окно: цвет текста | [FR-SWR-1-21-2](FR-SWR-1-21-2.docx) | 26.04.2014 | 12.05.2014 | Сазонов И.М. |
| 21 | FR-SWR-1-22-2 | Всплывающее окно: цвет фона | [FR-SWR-1-22-2](FR-SWR-1-22-2.docx) | 26.04.2014 | 12.05.2014 | Сазонов И.М. |
| 22 | FR-SWR-1-23-1 | Правила | [FR-SWR-1-23-1](FR-SWR-1-23-1.docx) | 14.05.2014 | 14.05.2014 | Сазонов И.М. |
| 23 | FR-SWR-1-24-1 | О программе | [FR-SWR-1-24-1](FR-SWR-1-24-1.docx) | 14.05.2014 | 14.05.2014 | Сазонов И.М. |
| 24 | FR-SWR-2-1-2 | Очередность хода | [FR-SWR-2-1-2](FR-SWR-2-1-2.docx) | 26.04.2014 | 28.04.2014 | Сазонов И.М. |
| 25 | FR-SWR-2-2-1 | Ограничение хода | [FR-SWR-2-2-1](FR-SWR-2-2-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 26 | FR-SWR-2-3-2 | Переход хода | [FR-SWR-2-3-](FR-SWR-2-3-2.docx)2 | 26.04.2014 | 28.04.2014 | Сазонов И.М. |
| 27 | FR-SWR-2-4-1 | Заполнение ячеек | [FR-SWR-2-4-1](FR-SWR-2-4-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 28 | FR-SWR-2-5-1 | Конец игры – надпись | [FR-SWR-2-5-1](FR-SWR-2-5-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 29 | NFR SWR-1-1-1 | Отчет | [NFR SWR-1-1-1](NFR%20SWR-1-1-1.docx) | 21.04.2014 | 21.04.2014 | Сазонов И.М. |
| 30 | NFR SWR-2-1-1 | Тесты | [NFR SWR-2-1-1](NFR%20SWR-2-1-1.docx) | 21.04.2014 | 21.04.2014 | Сазонов И.М. |
| 31 | NFR SWR-2-2-1 | Сбои | [NFR SWR-2-2-1](NFR%20SWR-2-2-1.docx) | 21.04.2014 | 21.04.2014 | Сазонов И.М. |
| 32 | NFR SWR-4-1-1 | Язык программирования | [NFR SWR-4-1-1](NFR%20SWR-4-1-1.docx) | 21.04.2014 | 21.04.2014 | Сазонов И.М. |
| 33 | UR-1-2 | Режимы работы | [UR-1-2](UR-1-2.docx) | 21.04.2014 | 28.04.2014 | Сазонов И.М. |
| 34 | UR-2-1 | Время хода | [UR-2-1](UR-2-1.docx) | 21.04.2014 | 21.04.2014 | Сазонов И.М. |
| 35 | UR-3-2 | Режим <Игра с человеком> | [UR-3-2](UR-3-2.docx) | 21.04.2014 | 28.04.2014 | Сазонов И.М. |
| 36 | UR-4-2 | Режим <Игра с компьютером> | [UR-4-2](UR-4-2.docx) | 21.04.2014 | 28.04.2014 | Сазонов И.М. |
| 37 | UR-5-1 | Победа крестиков | [UR-5-1](UR-5-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 38 | UR-6-1 | Победа ноликов | [UR-6-1](UR-6-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 39 | UR-7-1 | Ничья | [UR-7-1](UR-7-1.docx) | 26.04.2014 | 26.04.2014 | Сазонов И.М. |
| 40 | UR-8-2 | <Игра с компьютером>: Компьютер | [UR-8-](UR-8-2.docx)2 | 26.04.2014 | 28.04.2014 | Сазонов И.М. |
| 41 | UR-9-2 | <Игра с компьютером>: Игрок | [UR-9-2](UR-9-2.docx) | 26.04.2014 | 28.04.2014 | Сазонов И.М. |
| 42 | UR-10-2 | <Игра с человеком>: Игрок1 | [UR-10-](UR-10-2.docx)2 | 26.04.2014 | 28.04.2014 | Сазонов И.М. |
| 43 | UR-11-2 | <Игра с человеком>: Игрок2 | [UR-11-2](UR-11-2.docx) | 26.04.2014 | 28.04.2014 | Сазонов И.М. |

## 4.2. Описание архитектуры ПО

**Автор:** Рябчиков Алексей Андреевич

Была использована следующая архитектура (рис.1):

|  |
| --- |
| **Архитектура.jpg** |
| Рис.1. Архитектура |

## 4.3. Описание алгоритмов работы ПО

**Описание алгоритмов**

**Автор:** Рябчиков Алексей Андреевич

**Алгоритм старта «Новой игры».**

*Назначение*: Запускает новую игру.

*Содержание:* Обнуляется массив клеток поля, запускается таймер.

**Блок схема основного алгоритма программы (рис.2)**

|  |
| --- |
| Блок схема основного алгоритма программы.jpg |
| Рис.2. Основной алгоритм программы |

**Блок схема алгоритма выбора сложности игры с компьютером (рис.3)**

|  |
| --- |
| **Блок схема выбора сложности игры с компьютером.jpg** |
| Рис.3. Выбор сложности игры с компьютером |

**Блок схема алгоритма Player vs Player (рис.4)**

|  |
| --- |
| **Блок схема алгоритма Player vs Player.jpg** |
| Рис.4. Player vs Player |

**Блок схема алгоритма Player vs Computer (рис.5)**

|  |
| --- |
| **Блок схема алгоритма Player vs Computer.jpg** |
| Рис.5. Player vs Computer |

**Алгоритм Хода компьютера в «Простом режиме»**

*Назначение:* Описывает алгоритм хода компьютера в «Простом режиме» (рис.6).

*Содержание:* Нолик ставится компьютером в случайно выбранную не занятую клетку.

|  |
| --- |
| Блок схема алгорима Хода компьютера в Простом режиме.jpg |
| Рис.6. Простой режим |

**Алгоритм Хода компьютера в «Сложном режиме»**

*Назначение:* Описывает алгоритм хода компьютера в «Сложном режиме» (рис.7).

*Содержание:* Если есть одна свободная клетка (определение критической ситуации) в линии с двумя занятыми ноликами, то компьютер ставит в нее нолик и побеждает. Если есть свободная клетка в линии, занятой игроком,(определение критической ситуации) то компьютер ставит в нее нолик, это не дает победить игроку. Если свободна средняя клетка, то компьютер ставит в нее нолик, как в самую выгодную позицию. Следующая позиция – угловая клетка, позволяет контролировать три линии. И, если все перечисленное выше уже занято, то компьютер занимает среднюю боковую клетку.

|  |
| --- |
| Блок схема алгорима Хода компьютера в Сложном режиме.jpg |
| Рис.7. алгоритм хода компьютера в «Сложном режиме» |

**Алгоритм «Определения критической ситуации»**

*Назначение:* Определяет наличие критической ситуации. Под ней понимается такое расположение крестиков или ноликов, при котором для выигрыша, достаточно поставить крестик или нолик в свободную клетку на линии. (т. е. на линии из 3-х элементов, свободен только один) (рис.8).

*Содержание:* Алгоритм заключается в проверке на наличие линии с двумя занятыми клетками и одной свободной. Для этого последовательно проверяются все строки, столбцы и диагонали. В случае если успешно найдена критическая ситуация – возвращается номер пустой клетки. В функцию передается переменная, которая определяет для крестиков или ноликов проводить оценку.

|  |
| --- |
| **Блок схема алгоритма Определения критической ситуации.jpg** |
| Рис.8. Критическая ситуауия |

**Алгоритм «Проверки на победу»**

*Назначение:* Позволяет определить: кто одержал победу, крестики или нолики.

*Содержание:* Проверяются строки, столбцы и диагонали поля. Если все три клетки заняты одним значком, то возвращается чья-либо победа, если все клетки заняты, но все линии не содержат трех одинаковых значков, то ничья. В противном случае – игра продолжается.

|  |
| --- |
| **Блок схема алгоритма Проверки на победу.jpg** |
| Рис.9. Проверка на победу. |

**Алгоритм работы «Основного таймера статуса игры»**

*Назначение:* Таймер предназначен для отслеживания текущего статуса игры.

*Содержание:* Если статус равен 2, то начинается новая игра. Если статус равен 0, то выводится сообщение о ходе игрока и таймер останавливается в ожидании хода игрока. Если статус равен 1 (то есть ход компьютера), то производится его ход, после чего ход передается игроку (статус меняется), производится проверка, если она возвращает 0, то игра продолжается, если нет, то вызывается функция для обработки победы или ничьей.

**Алгоритм работы «Таймера времени хода»**

*Назначение:* Ограничивает время хода 15-ю секундами.

*Содержание:* Если в течении 15 секунд игрок не сходил, ход передается следующему игроку.

# 5. Тестирование ПО

## 5.1. Объем тестирования ПО

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№ п/п** | **Обозначение Требования** | **Обозначение теста** | **Ссылка на документ** | **Планируемая дата проведения тестирования** | **Тестировщик** |
| 1 | NFR SWR-2-1-1 | tNFR SWR-2-1 | \Тесты\Нефункциональные\Атрибуты качества ПО\tNFR SWR 2-1 | 12.05.2014 | Касатов Г.С |
| 2 | NFR SWR-2-2-1 | tNFR SWR-2-2 | \Тесты\Нефункциональные\Атрибуты качества ПО\tNFR SWR 2-2 | 12.05.2014 | Касатов Г.С |
| 3 | NFR SWR-2-2-1 | tNFR SWR-2-3 | \Тесты\Нефункциональные\Атрибуты качества ПО\tNFR SWR 2-3 | 12.05.2014 | Касатов Г.С |
| 4 | NFR SWR-2-2-1 | tNFR SWR-2-4 | \Тесты\Нефункциональные\Атрибуты качества ПО\tNFR SWR 2-4 | 12.05.2014 | Касатов Г.С |
| 5 | NFR SWR-1-1-1 | tNFR-SWR-1-1 | \Тесты\Нефункциональные\Бизнес правила\tNFR SWR-1-1 | 12.05.2014 | Касатов Г.С |
| 6 | NFR SWR-4-1-1 | tNFR-SWR-4-1 | \Тесты\Нефункциональные\Ограничения\tNFR SWR-4-1 | 12.05.2014 | Касатов Г.С |
| 7 | UR-1-2 | tUR-1 | \Тесты\Пользовательские\tUR-1 | 12.05.2014 | Касатов Г.С |
| 8 | UR-1-2 | tUR-2 | \Тесты\Пользовательские\tUR-2 | 12.05.2014 | Касатов Г.С |
| 9 | UR-1-2 | tUR-3 | \Тесты\Пользовательские\tUR-3 | 12.05.2014 | Касатов Г.С |
| 10 | UR-2-1 | tUR-4 | \Тесты\Пользовательские\tUR-4 | 12.05.2014 | Касатов Г.С |
| 11 | UR-2-1 | tUR-5 | \Тесты\Пользовательские\tUR-5 | 12.05.2014 | Касатов Г.С |
| 12 | UR-2-1 | tUR-6 | \Тесты\Пользовательские\tUR-6 | 12.05.2014 | Касатов Г.С |
| 13 | UR-3-2 | tUR-7 | \Тесты\Пользовательские\tUR-7 | 12.05.2014 | Касатов Г.С |
| 14 | UR-4-2 | tUR-8 | \Тесты\Пользовательские\tUR-8 | 12.05.2014 | Касатов Г.С |
| 15 | UR-4-2 | tUR-9 | \Тесты\Пользовательские\tUR-9 | 12.05.2014 | Касатов Г.С |
| 16 | UR-5-1 | tUR-10 | \Тесты\Пользовательские\tUR-10 | 12.05.2014 | Касатов Г.С |
| 17 | UR-5-1 | tUR-11 | \Тесты\Пользовательские\tUR-11 | 12.05.2014 | Касатов Г.С |
| 18 | UR-5-1 | tUR-12 | \Тесты\Пользовательские\tUR-12 | 12.05.2014 | Касатов Г.С |
| 19 | UR-6-1 | tUR-13 | \Тесты\Пользовательские\tUR-13 | 12.05.2014 | Касатов Г.С |
| 20 | UR-6-1 | tUR-14 | \Тесты\Пользовательские\tUR-14 | 12.05.2014 | Касатов Г.С |
| 21 | UR-6-1 | tUR-15 | \Тесты\Пользовательские\tUR-15 | 12.05.2014 | Касатов Г.С |
| 22 | UR-7-1 | tUR-16 | \Тесты\Пользовательские\tUR-16 | 12.05.2014 | Касатов Г.С |
| 23 | UR-7-1 | tUR-17 | \Тесты\Пользовательские\tUR-17 | 12.05.2014 | Касатов Г.С |
| 24 | UR-7-1 | tUR-18 | \Тесты\Пользовательские\tUR-18 | 12.05.2014 | Касатов Г.С |
| 25 | UR-8-2 | tUR-19 | \Тесты\Пользовательские\tUR-19 | 12.05.2014 | Касатов Г.С |
| 26 | UR-8-2 | tUR-20 | \Тесты\Пользовательские\tUR-20 | 12.05.2014 | Касатов Г.С |
| 27 | UR-9-2 | tUR-21 | \Тесты\Пользовательские\tUR-21 | 12.05.2014 | Касатов Г.С |
| 28 | UR-9-2 | tUR-22 | \Тесты\Пользовательские\tUR-22 | 12.05.2014 | Касатов Г.С |
| 29 | UR-10-2 | tUR-23 | \Тесты\Пользовательские\tUR-23 | 12.05.2014 | Касатов Г.С |
| 30 | UR-11-2 | tUR-24 | \Тесты\Пользовательские\tUR-24 | 12.05.2014 | Касатов Г.С |
| 31 | FR-SWR-2-1-2 | tFR-SWR-2-1 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-1 | 12.05.2014 | Касатов Г.С |
| 32 | FR-SWR-2-1-2 | tFR-SWR-2-2 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-2 | 12.05.2014 | Касатов Г.С |
| 33 | FR-SWR-2-1-2 | tFR-SWR-2-3 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-3 | 12.05.2014 | Касатов Г.С |
| 34 | FR-SWR-2-2-1 | tFR-SWR-2-4 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-4 | 12.05.2014 | Касатов Г.С |
| 35 | FR-SWR-2-2-1 | tFR-SWR-2-5 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-5 | 12.05.2014 | Касатов Г.С |
| 36 | FR-SWR-2-2-1 | tFR-SWR-2-6 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-6 | 12.05.2014 | Касатов Г.С |
| 37 | FR-SWR-2-3-2 | tFR-SWR-2-7 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-7 | 12.05.2014 | Касатов Г.С |
| 38 | FR-SWR-2-3-2 | tFR-SWR-2-8 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-8 | 12.05.2014 | Касатов Г.С |
| 39 | FR-SWR-2-3-2 | tFR-SWR-2-9 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-9 | 12.05.2014 | Касатов Г.С |
| 40 | FR-SWR-2-4-1 | tFR-SWR-2-10 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-10 | 12.05.2014 | Касатов Г.С |
| 41 | FR-SWR-2-4-1 | tFR-SWR-2-11 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-11 | 12.05.2014 | Касатов Г.С |
| 42 | FR-SWR-2-4-1 | tFR-SWR-2-12 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-12 | 12.05.2014 | Касатов Г.С |
| 43 | FR-SWR-2-5-1 | tFR-SWR-2-13 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-13 | 12.05.2014 | Касатов Г.С |
| 44 | FR-SWR-2-5-1 | tFR-SWR-2-14 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-14 | 12.05.2014 | Касатов Г.С |
| 45 | FR-SWR-2-5-1 | tFR-SWR-2-15 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-15 | 12.05.2014 | Касатов Г.С |
| 46 | FR-SWR-1-1-1 | tFR-SWR-1-1 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-1 | 12.05.2014 | Касатов Г.С |
| 47 | FR-SWR-1-2-2 | tFR-SWR-1-2 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-2 | 12.05.2014 | Касатов Г.С |
| 48 | FR-SWR-1-3-2 | tFR-SWR-1-3 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-3 | 12.05.2014 | Касатов Г.С |
| 49 | FR-SWR-1-4-2 | tFR-SWR-1-4 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-4 | 12.05.2014 | Касатов Г.С |
| 50 | FR-SWR-1-5-2 | tFR-SWR-1-5 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-5 | 12.05.2014 | Касатов Г.С |
| 51 | FR-SWR-1-6-1 | tFR-SWR-1-6 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-6 | 12.05.2014 | Касатов Г.С |
| 52 | FR-SWR-1-7-1 | tFR-SWR-1-7 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-7 | 12.05.2014 | Касатов Г.С |
| 53 | FR-SWR-1-8-1 | tFR-SWR-1-8 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-8 | 12.05.2014 | Касатов Г.С |
| 54 | FR-SWR-1-9-1 | tFR-SWR-1-9 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-9 | 12.05.2014 | Касатов Г.С |
| 55 | FR-SWR-1-10-2 | tFR-SWR-1-10 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-10 | 12.05.2014 | Касатов Г.С |
| 56 | FR-SWR-1-11-1 | tFR-SWR-1-11 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-11 | 12.05.2014 | Касатов Г.С |
| 57 | FR-SWR-1-12-3 | tFR-SWR-1-12 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-12 | 12.05.2014 | Касатов Г.С |
| 58 | FR-SWR-1-13-2 | tFR-SWR-1-13 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-13 | 12.05.2014 | Касатов Г.С |
| 59 | FR-SWR-1-14-1 | tFR-SWR-1-14 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-14 | 12.05.2014 | Касатов Г.С |
| 60 | FR-SWR-1-16-2 | tFR-SWR-1-16 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-16 | 12.05.2014 | Касатов Г.С |
| 61 | FR-SWR-1-17-2 | tFR-SWR-1-17 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-17 | 12.05.2014 | Касатов Г.С |
| 62 | FR-SWR-1-18-3 | tFR-SWR-1-18 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-18 | 12.05.2014 | Касатов Г.С |
| 63 | FR-SWR-1-19-2 | tFR-SWR-1-19 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-19 | 12.05.2014 | Касатов Г.С |
| 64 | FR-SWR-1-20-1 | tFR-SWR-1-20 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-20 | 12.05.2014 | Касатов Г.С |
| 65 | FR-SWR-1-21-2 | tFR-SWR-1-21 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-21 | 12.05.2014 | Касатов Г.С |
| 66 | FR-SWR-1-22-2 | tFR-SWR-1-22 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-22 | 12.05.2014 | Касатов Г.С |
| 67 | FR-SWR-1-23-1 | tFR-SWR-1-23 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-23 | 12.05.2014 | Касатов Г.С |
| 68 | FR-SWR-1-24-1 | tFR-SWR-1-24 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-24 | 12.05.2014 | Касатов Г.С |
|  |  |  |  |  |  |
|  |  |  | |  |  |
|  |  | Итерация 2 | |  |  |
| **№ п/п** | **Обозначение Требования** | **Обозначение теста** | **Ссылка на документ** | **Планируемая дата проведения тестирования** | **Тестировщик** |
| 69 | NFR SWR-2-1-1 | tNFR SWR-2-1 | \Тесты\Нефункциональные\Атрибуты качества ПО\tNFR SWR 2-1 | 13.05.2014 | Касатов Г.С |
| 70 | NFR SWR-1-1-1 | tNFR SWR-1-1 | \Тесты\Нефункциональные\Атрибуты качества ПО\tNFR SWR 1-1 | 13.05.2014 | Касатов Г.С |
| 71 | UR-1-2 | tUR-2 | \Тесты\Пользовательские\tUR-2 | 13.05.2014 | Касатов Г.С |
| 72 | UR-2-1 | tUR-5 | \Тесты\Пользовательские\tUR-5 | 13.05.2014 | Касатов Г.С |
| 73 | UR-4-2 | tUR-8 | \Тесты\Пользовательские\tUR-8 | 13.05.2014 | Касатов Г.С |
| 74 | UR-8-2 | tUR-19 | \Тесты\Пользовательские\tUR-19 | 13.05.2014 | Касатов Г.С |
| 75 | UR-9-2 | tUR-21 | \Тесты\Пользовательские\tUR-21 | 13.05.2014 | Касатов Г.С |
| 76 | FR-SWR-2-2-1 | tFR-SWR-2-5 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-5 | 13.05.2014 | Касатов Г.С |
| 77 | FR-SWR-2-3-2 | tFR-SWR-2-8 | \Тесты\Функциональные\Общие функциональные требования\tFR-SWR-2-8 | 13.05.2014 | Касатов Г.С |
| 78 | FR-SWR-1-2-2 | tFR-SWR-1-2 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-2 | 13.05.2014 | Касатов Г.С |
| 79 | FR-SWR-1-3-2 | tFR-SWR-1-3 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-3 | 13.05.2014 | Касатов Г.С |
| 80 | FR-SWR-1-4-2 | tFR-SWR-1-4 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-4 | 13.05.2014 | Касатов Г.С |
| 81 | FR-SWR-1-5-2 | tFR-SWR-1-5 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-5 | 13.05.2014 | Касатов Г.С |
| 82 | FR-SWR-1-18-3 | tFR-SWR-1-18 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-18 | 13.05.2014 | Касатов Г.С |
| 83 | FR-SWR-1-19-2 | tFR-SWR-1-19 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-19 | 13.05.2014 | Касатов Г.С |
|  |  |  |  |  |  |
|  |  | Итерация 3 | |  |  |
| **№ п/п** | **Обозначение Требования** | **Обозначение теста** | **Ссылка на документ** | **Планируемая дата проведения тестирования** | **Тестировщик** |
| 84 | NFR SWR-1-1-1 | tNFR SWR-1-1 | \Тесты\Нефункциональные\Бизнес правила\tNFR SWR-1-1 | 26.05.2014 | Касатов Г.С |
| 85 | FR-SWR-1-23-1 | tFR SWR-1-23 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-23 | 26.05.2014 | Касатов Г.С |
| 86 | FR-SWR-1-24-1 | tFR SWR-1-24 | \Тесты\Функциональные\Требования к интерфейсу\tFR-SWR-1-24 | 26.05.2014 | Касатов Г.С |

## 5.2. Результаты тестирования ПО

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Тестировщик:** | **Дата проведения тестирования:** | **Версия ПО:** | **Тестовая процедура:** | **Результат тестирования:** | **Комментарий:** |
| Касатов Г.С. | 12.05.2014 | 0,8 | tNFR SWR-2-1 | Не пройден | Несоответствие требований и тестов по обозначению. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tNFR SWR-2-2 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tNFR SWR-2-3 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tNFR SWR-2-4 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tNFR SWR-1-1 | Не пройден | Отсутствует отчет. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tNFR SWR-4-1 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-1 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-2 | Не пройден | Надпись, демонстрирующая, что ход передан следующему игроку, не всегда корректно отображает чей сейчас ход. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-3 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-4 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-5 | Не пройден | Надпись, демонстрирующая, что ход передан следующему игроку, не всегда корректно отображает чей сейчас ход. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-6 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-7 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-8 | Не пройден | Надпись, демонстрирующая, что ход передан следующему игроку, не всегда корректно отображает чей сейчас ход. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-9 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-10 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-11 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-12 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-13 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-14 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-15 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-16 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-17 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-18 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-19 | Не пройден | Надпись, демонстрирующая, что ход передан следующему игроку, не всегда корректно отображает чей сейчас ход. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-20 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-21 | Не пройден | Надпись, демонстрирующая, что ход передан следующему игроку, не всегда корректно отображает чей сейчас ход. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-22 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-23 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tUR-24 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-1 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-2 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-3 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-4 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-5 | Не пройден | Надпись, демонстрирующая, что ход передан следующему игроку, не всегда корректно отображает чей сейчас ход. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-6 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-7 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-8 | Не пройден | Надпись, демонстрирующая, что ход передан следующему игроку, не всегда корректно отображает чей сейчас ход. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-9 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-10 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-11 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-12 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-13 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-14 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-2-15 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-1 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-2 | Не пройден | Ширина равна 341-у пикселю, вместо заявленных 354-х. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-3 | Не пройден | Ширина равна 29-и пикселям, вместо заявленных 35-и. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-4 | Не пройден | Ширина равна 225-и пикселям, вместо заявленных 178-и. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-5 | Не пройден | Ширина равна 85-и пикселям, вместо заявленных 141-ого. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-6 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-7 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-8 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-9 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-10 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-11 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-12 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-13 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-14 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-16 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-17 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-18 | Не пройден | Масштаб изображения составляет 50х50 пикселей, вместо заявленных 60х70. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-19 | Не пройден | Масштаб изображения составляет 50х50 пикселей, вместо заявленных 60х70. |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-20 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-21 | Пройден |  |
| Касатов Г.С. | 12.05.2014 | 0,8 | tFR-SWR-1-22 | Пройден |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  | Итерация 2 | |  |  |
| **Тестировщик:** | **Дата проведения тестирования:** | **Версия ПО:** | **Тестовая процедура:** | **Результат тестирования:** | **Комментарий:** |
| Касатов Г.С. | 13.05.2014 | 0,9 | tNFR SWR-2-1 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tNFR SWR-1-1 | Не пройден | Отсутствует отчет. |
| Касатов Г.С. | 13.05.2014 | 0,9 | tUR-2 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tUR-5 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tUR-8 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tUR-19 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tUR-21 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tFR-SWR-2-5 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tFR-SWR-2-8 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tFR-SWR-1-2 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tFR-SWR-1-3 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tFR-SWR-1-4 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tFR-SWR-1-5 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tFR-SWR-1-18 | Пройден |  |
| Касатов Г.С. | 13.05.2014 | 0,9 | tFR-SWR-1-19 | Пройден |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  | Итерация 3 | |  |  |
| **Тестировщик:** | **Дата проведения тестирования:** | **Версия ПО:** | **Тестовая процедура:** | **Результат тестирования:** | **Комментарий:** |
| Касатов Г.С. | 26.05.2014 | 1,0 | tNFR SWR-1-1 | Пройден |  |
| Касатов Г.С. | 26.05.2014 | 1,0 | tFR SWR-1-23 | Пройден |  |
| Касатов Г.С. | 26.05.2014 | 1,0 | tFR SWR-1-24 | Пройден |  |

# 6. Руководство пользователя

## Об игре «Крестики-нолики»

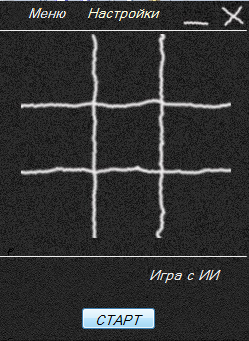
Крестики-нолики — логическая игра между двумя противниками на квадратном поле 3 на 3 клетки. Один из игроков играет «крестиками», второй — «ноликами».

## Минимальные системные требования

* Операционная система: Windows 7 x64
* Процессор: 1,6 ГГц
* Оперативная память: 128 Мб
* Свободное место на диске: 3 МБ
* Видеокарта: объем памяти 128 Мб
* Монитор: разрешение 800x600 пикселей в режиме True Color (32 бит)
* Мышь

## Начало игры

1. Дважды щелкните Крестики-нолики.exe
2. Выбрать режим сложности в «Настройках». По-умолчанию начнется игра с компьютером.
3. Нажмите кнопку «Старт».



## Интерфейс

В приложении доступна встроенная справка с правилами игры. Для ее просмотра, щелкните «Меню» -> «Правила».

Для просмотра информации о приложении щелкните «Меню» -> «Об игре».

Для того, чтобы свернуть приложение или выйти из него, нажмите на черту или крестик в верхнем левом углу соответственно:

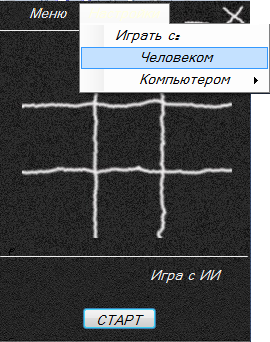


## Выбор режима игры

В игре можно выбрать один из двух режимы игры:

* Игра с человеком
* Игра с компьютером

Для выбора нужного режима щелкните «Настройки»

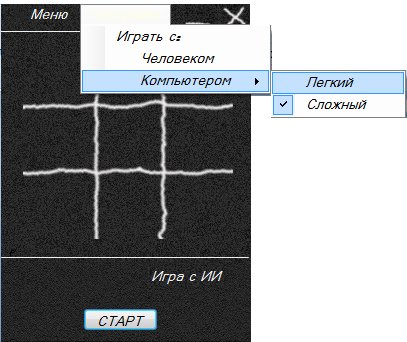


## Выбор уровня сложности

При игре с компьютером имеется два уровня сложности:

* Легкий
* Сложный

Для выбора нужного уровня щелкните «Настройки» -> «Играть с компьютером» -> требуемый уровень.



## Игра «Крестики-нолики»: основы

### Цель игры

Поставить в одну линию три крестика или нолика и не дать своему сопернику сделать то же самое.

### Игровое поле

Игра ведется на поле размером 3х3 квадрата. Ход можно совершить в любой пустой квадрат.

### Правила игры

Игроки ходят по очереди, ставя крестик или нолик. Первый ход делает игрок, ставящий крестики. На ход дается 15 секунд.

Чтобы совершить ход, нужно на рабочем поле поставить кликом мышью крестик или нолик.

Если игрок не уложился в отведенное время, право хода передается сопернику.

Первый, выстроивший в ряд 3 свои фигуры по вертикали, горизонтали или диагонали, выигрывает.

# Заключение

Поставленные перед коллективом задачи выполнены в полном объеме. Цель работы достигнута после 4 недель с началом разработки программного обеспечения.

Результатом работы является полностью рабочий и исправный исполняемый файл (.exe файл) «Крестики – Нолики.exe», корректно написанный программный код, а так же сопутствующая документация: список требований, список тестов, концепт дизайна, описание архитектуры и алгоритмов и итоговый отчет.

*Список сокращений:*

1. ПО – программное обеспечение.

*Использованная литература:*



# Приложение 1. Исходный код ПО

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.Net.Sockets;

using System.Threading;

namespace Лабораторная\_работа.КрестикиНолии

{

**public** partial **class** Form2 : Form

{

**int** MouseX, MouseY;

**int** x, y, Ai = 1;

**int** qq;

**int** timer = 15;

Int16[,] arr = **new** Int16[3, 3];// определяет занятые и свободные клетки

**int** element = 1; // флаг определяет кто сейчас ходит, 1- крестики , 2- нолики

**public** Form2()

{

InitializeComponent();

**this**.TransparencyKey = Color.White;

**this**.BackgroundImage = **new** Bitmap(GetType(), "Fons.jpg");

**this**.pictureBox1.Image = **new** Bitmap(GetType(), "Close.jpg");

**this**.pictureBox2.Image = **new** Bitmap(GetType(), "Svernut.jpg");

**this**.menuStrip1.BackgroundImage = **new** Bitmap(GetType(), "Fons.jpg");

**this**.label1.Image = **new** Bitmap(GetType(), "Fon.jpg");

label2.Text = ("Игра с ИИ");

label3.Visible = **false**;

label4.Visible = **false**;

}

**private** **void** Cross(**int** n, **int** m)

{

**if** (arr[n, m] == 0)

{

Graphics g = CreateGraphics();

Bitmap i = **new** Bitmap(GetType(), "Krestcl.png");

g.DrawImage(i, x, y, 54, 54);

arr[n, m] = 1;

element = 2;

x = 0;

y = 0;

timer = 15;

}

}

**private** **void** Nought(**int** n, **int** m)

{

**if** (arr[n, m] == 0)

{

Graphics g = CreateGraphics();

Bitmap i = **new** Bitmap(GetType(), "Nol.png");

g.DrawImage(i, x, y, 54, 54);

arr[n, m] = 2;

element = 1;

x = 0; y = 0;

timer = 15;

}

}

**private** **void** Line\_45(**int** x, **int** y)

{

Graphics g = CreateGraphics();

Image i = **new** Bitmap(GetType(), "Line-45.png");

g.DrawImage(i, x, y);

g.Dispose();

} // рисует линию по диагонали

**private** **void** Line45(**int** x, **int** y)

{

Graphics g = CreateGraphics();

Image i = **new** Bitmap(GetType(), "Line45.png");

g.DrawImage(i, x, y);

g.Dispose();

} // рисует линию по диагонали

**private** **void** LineVer(**int** x, **int** y)

{

Bitmap i = **new** Bitmap(**this**.GetType(), "LineVer.png");

Graphics g = CreateGraphics();

g.DrawImage(i, x, y);

} // Рисует линию по вертикали

**private** **void** LineGor(**int** x, **int** y)

{

Bitmap i = **new** Bitmap(**this**.GetType(), "LineGor.png");

Graphics g = CreateGraphics();

g.DrawImage(i, x, y);

} // рисует линию по горизонтали

**private** **void** Mouse(object sender, MouseEventArgs e)

{

**if** (e.X >= 25 && e.X <= 75) MouseX = 25;

**if** (e.X >= 100 && e.X <= 150) MouseX = 100;

**if** (e.X >= 167 && e.X <= 217) MouseX = 167;

**if** (e.Y >= 41 && e.Y <= 91) MouseY = 41;

**if** (e.Y >= 109 && e.Y <= 159) MouseY = 109;

**if** (e.Y >= 178 && e.Y <= 228) MouseY = 178;

} // Определяет обасть щелчка мыши

**private** **int** check()//Проверка победителя , если крестики то возвращает 1, если нолики то 2 , при ничьей 3

{

//Проверка крестиков по горизонтали

**if** (arr[0, 2] == 1 && arr[1, 2] == 1 && arr[2, 2] == 1)

{

LineGor(25, 41);

**return** 1;

}

**if** (arr[0, 1] == 1 && arr[1, 1] == 1 && arr[2, 1] == 1)

{

LineGor(25, 109);

**return** 1;

}

**if** (arr[0, 0] == 1 && arr[1, 0] == 1 && arr[2, 0] == 1)

{

LineGor(25, 178);

**return** 1;

}

//проверка крестиков по вертикали

**if** (arr[0, 2] == 1 && arr[0, 1] == 1 && arr[0, 0] == 1)

{

LineVer(25, 41);

**return** 1;

}

**if** (arr[1, 2] == 1 && arr[1, 1] == 1 && arr[1, 0] == 1)

{

LineVer(100, 41);

**return** 1;

}

**if** (arr[2, 2] == 1 && arr[2, 1] == 1 && arr[2, 0] == 1)

{

LineVer(178, 41);

**return** 1;

}

// Проверка крестиков по дианонали

**if** (arr[0, 2] == 1 && arr[1, 1] == 1 && arr[2, 0] == 1)

{

Line\_45(25, 41);

**return** 1;

}

**if** (arr[2, 2] == 1 && arr[1, 1] == 1 && arr[0, 0] == 1)

{

Line45(25, 41);

**return** 1;

}

//Проверка ноликов по горизонтали

**if** (arr[0, 2] == 2 && arr[1, 2] == 2 && arr[2, 2] == 2)

{

LineGor(25, 41);

**return** 2;

}

**if** (arr[0, 1] == 2 && arr[1, 1] == 2 && arr[2, 1] == 2)

{

LineGor(25, 109);

**return** 2;

}

**if** (arr[0, 0] == 2 && arr[1, 0] == 2 && arr[2, 0] == 2)

{

LineGor(25, 178);

**return** 2;

}

//Проверка ноликов по вертикали

**if** (arr[0, 2] == 2 && arr[0, 1] == 2 && arr[0, 0] == 2)

{

LineVer(25, 41);

**return** 2;

}

**if** (arr[1, 2] == 2 && arr[1, 1] == 2 && arr[1, 0] == 2)

{

LineVer(100, 41);

**return** 2;

}

**if** (arr[2, 2] == 2 && arr[2, 1] == 2 && arr[2, 0] == 2)

{

LineVer(178, 41);

**return** 2;

}

//Проверка ноликов по диагонали

**if** (arr[0, 2] == 2 && arr[1, 1] == 2 && arr[2, 0] == 2)

{

Line\_45(25, 41);

**return** 2;

}

**if** (arr[2, 2] == 2 && arr[1, 1] == 2 && arr[0, 0] == 2)

{

Line45(25, 41);

**return** 2;

}

**if** (arr[0, 0] != 0 && arr[0, 1] != 0 && arr[0, 2] != 0 && arr[1, 0] != 0 && arr[1, 1] != 0 && arr[1, 2] != 0 && arr[2, 0] != 0 && arr[2, 1] != 0 && arr[2, 2] != 0)

{

label1.Text = ("Ничья");

timer2.Stop();

**return** 3;

}

**else**

**return** 0;

}

**private** **void** pictureBox1\_Click(object sender, EventArgs e)

{

timer1.Enabled = **true**;

timer1.Start();

timer1\_Tick(sender, e);

}

**private** **void** timer1\_Tick(object sender, EventArgs e)

{

**for** (; Opacity > 0; Opacity -= .01, System.Threading.Thread.Sleep(10)) ;

**if** (**this**.Opacity <= 0)

{

//Таймер останавливается

timer1.Stop();

//Форма закрывается

**this**.Close();

} // выключение формы

}

**private** **void** pictureBox2\_Click(object sender, EventArgs e)

{

WindowState = FormWindowState.Minimized;

}

**private** **void** Form2\_Load(object sender, EventArgs e)

{

}

**private** **void** button1\_Click(object sender, EventArgs e)

{

button1.Visible = **false**;

label3.Visible = **true**;

label4.Visible = **true**;

NewGames();

}

**private** **void** NewGames()

{

timer2.Stop();

MouseX = MouseY = x = y = 0;

**this**.BackgroundImage = **new** Bitmap(**this**.GetType(), "Fons.jpg");

**int** stroka = arr.GetLength(0), stolbec = arr.GetLength(1);

**for** (**int** i = 0; i < stroka; i++)

{

**for** (**int** j = 0; j < stolbec; j++)

{

arr[i, j] = 0;

}

} qq = 0;

element = 1;

timer = 15;

timer2.Start();

}

**private** **void** timer2\_Tick(object sender, EventArgs e)

{

label3.Text = (timer--).ToString();

**if** (timer < 0)

{

**if** (element == 1) { element++; timer = 15; }

**else** { element--; timer = 15; }

timer = 15;

}

// sp2.Load();

x = MouseX;

y = MouseY;

**if** (element == 1) label1.Text = ("Ходят крестики");

**if** (element == 2) label1.Text = ("Ходят нолики");

AI();

Ai\_low();

//if (element == 1) label1.Text = ("Ходят крестики");

//if (element == 2) label1.Text = ("Ходят нолики");

TooPlayer();

// Проверяет в какой области был произведен клик мышкой и рисует там крестик или нолик( в зависимости от хода)

**if** (x == 25 && y == 41)

{

**if** (element == 1)

{

Cross(0, 2); **return**;

}

}

**if** (x == 25 && y == 109)

{

**if** (element == 1)

{

Cross(0, 1); **return**;

}

}

**if** (x == 25 && y == 178)

{

**if** (element == 1)

{

Cross(0, 0); **return**;

}

}

**if** (x == 100 && y == 41)

{

**if** (element == 1)

{

Cross(1, 2); **return**;

}

}

**if** (x == 100 && y == 109)

{

**if** (element == 1)

{

Cross(1, 1); **return**;

}

}

**if** (x == 100 && y == 178)

{

**if** (element == 1)

{

Cross(1, 0); **return**;

}

}

**if** (x == 167 && y == 41)

{

**if** (element == 1)

{

Cross(2, 2); **return**;

}

}

**if** (x == 167 && y == 109)

{

**if** (element == 1)

{

Cross(2, 1); **return**;

}

}

**if** (x == 167 && y == 178)

{

**if** (element == 1)

{

Cross(2, 0); **return**;

}

}

**if** (check() == 1)

{

label1.Text = ("Победа крестиков");

timer2.Stop();

}

**if** (check() == 2)

{

label1.Text = ("Победа ноликов");

timer2.Stop();

}

} // Таймер отвечает за проверку условий победы, условий постановки крестиков , вызыв ИИ

**private** **void** TooPlayer()

{

**if** (Ai == 0)

{

**if** (x == 25 && y == 41)

{

**if** (element == 2)

{

Nought(0, 2); **return**;

}

}

**if** (x == 25 && y == 109)

{

**if** (element == 2)

{

Nought(0, 1); **return**;

}

}

**if** (x == 25 && y == 178)

{

**if** (element == 2)

{

Nought(0, 0); **return**;

}

}

**if** (x == 100 && y == 41)

{

**if** (element == 2)

{

Nought(1, 2); **return**;

}

}

**if** (x == 100 && y == 109)

{

**if** (element == 2)

{

Nought(1, 1); **return**;

}

}

**if** (x == 100 && y == 178)

{

**if** (element == 2)

{

Nought(1, 0); **return**;

}

}

**if** (x == 167 && y == 41)

{

**if** (element == 2)

{

Nought(2, 2); **return**;

}

}

**if** (x == 167 && y == 109)

{

**if** (element == 2)

{

Nought(2, 1); **return**;

}

}

**if** (x == 167 && y == 178)

{

**if** (element == 2)

{

Nought(2, 0); **return**;

}

}

}

}

**private** **void** человекомToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

**if** (Ai == 1 || Ai == 2)

{

Ai = 0;

label2.Text = ("Игра с другом");

NewGames();

}

} // Расстановка ноликов для второго игрока

**private** **void** Ai\_low()

{

**if** (element == 2 && Ai == 2)

{

**int** n, m;

Random rnd = **new** Random();

n = rnd.Next(3);

m = rnd.Next(3);

**if** (n == 0) x = 25;

**if** (n == 1) x = 100;

**if** (n == 2) x = 167;

**if** (m == 0) y = 178;

**if** (m == 1) y = 109;

**if** (m == 2) y = 41;

**if** (arr[n, m] == 0)

{

Nought(n, m);

element = 1;

x = 0;

y = 0;

MouseX = 0;

MouseY = 0;

n = 0;

m = 0;

**return**;

}

}

}

**private** **int** CritSituation()

{

**if** (arr[0, 0] == 2 && arr[0, 1] == 2 && arr[0,2]==0)

{ **return** 8; }

**if** (arr[0, 0] == 2 && arr[1, 1] == 2 && arr[2,2]==0)

{ **return** 2; }

**if** (arr[0, 0] == 2 && arr[1, 0] == 2 && arr[2,0]==0)

{ **return** 0; }

**if** (arr[0, 0] == 2 && arr[0, 2] == 2 && arr[0,1]==0)

{ **return** 7; }

**if** (arr[0, 0] == 2 && arr[2, 2] == 2 && arr[1,1]==0)

{ **return** 4; }

**if** (arr[0, 0] == 2 && arr[2, 0] == 2 && arr[1,0]==0)

{ **return** 3; } // для клетки 0 0

**if** (arr[0, 1] == 2 && arr[0, 2] == 2 && arr[0,0]==0)

{ **return** 6; }

**if** (arr[0, 1] == 2 && arr[1, 1] == 2 && arr[2,1]==0)

{ **return** 1; }

**if** (arr[0, 1] == 2 && arr[2, 1] == 2 && arr[1,1]==0)

{ **return** 4; } // для клетки 0 1

**if** (arr[0, 2] == 2 && arr[1, 1] == 2 && arr[2,0]==0)

{ **return** 0; }

**if** (arr[0, 2] == 2 && arr[2, 0] == 2 && arr[1,1]==0)

{ **return** 4; }

**if** (arr[0, 2] == 2 && arr[1, 2] == 2 && arr[2,2]==0)

{ **return** 2; }

**if** (arr[0, 2] == 2 && arr[2, 2] == 2 && arr[1,2]==0)

{ **return** 5; } // для клетки 0 2

**if** (arr[1, 0] == 2 && arr[2, 0] == 2 && arr[0,0]==0)

{ **return** 6; }

**if** (arr[1, 0] == 2 && arr[1, 1] == 2 && arr[1,2]==0)

{ **return** 5; }

**if** (arr[1, 0] == 2 && arr[1, 2] == 2 && arr[1,1]==0)

{ **return** 4; } //для клетки 1 0

**if** (arr[1, 1] == 2 && arr[1, 0] == 2 && arr[1,2]==0)

{ **return** 5; }

**if** (arr[1, 1] == 2 && arr[1, 2] == 2 && arr[1,0]==0)

{ **return** 3; }

**if** (arr[1, 1] == 2 && arr[2, 1] == 2 && arr[0,1]==0)

{ **return** 7; }

**if** (arr[1, 1] == 2 && arr[2, 0] == 2 && arr[0,2]==0)

{ **return** 8; }

**if** (arr[1, 1] == 2 && arr[2, 2] == 2 && arr[0,0]==0)

{ **return** 6; } //для клетки 1 1

**if** (arr[1, 2] == 2 && arr[2, 2] == 2 && arr[0,2]==0)

{ **return** 8; } //для клетки 1 2

**if** (arr[2, 0] == 2 && arr[2, 1] == 2 && arr[2,2]==0)

{ **return** 2; }

**if** (arr[2, 0] == 2 && arr[2, 2] == 2 && arr[2,1]==0)

{ **return** 1; } //для клетки 2 0

**if** (arr[2, 1] == 2 && arr[2, 2] == 2 && arr[2,0]==0)

{ **return** 0; } //для клетки 2 1

**if** (arr[0, 0] == 1 && arr[0, 1] == 1 && arr[0,2]==0)

{ **return** 8; }

**if** (arr[0, 0] == 1 && arr[1, 1] == 1 && arr[2,2]==0)

{ **return** 2; }

**if** (arr[0, 0] == 1 && arr[1, 0] == 1 && arr[2,0]==0)

{ **return** 0; }

**if** (arr[0, 0] == 1 && arr[0, 2] == 1 && arr[0,1]==0)

{ **return** 7; }

**if** (arr[0, 0] == 1 && arr[2, 2] == 1 && arr[1,1]==0)

{ **return** 4; }

**if** (arr[0, 0] == 1 && arr[2, 0] == 1 && arr[1,0]==0)

{ **return** 3; } // для клетки 0 0

**if** (arr[0, 1] == 1 && arr[0, 2] == 1 && arr[0,0]==0)

{ **return** 6; }

**if** (arr[0, 1] == 1 && arr[1, 1] == 1 && arr[2,1]==0)

{ **return** 1; }

**if** (arr[0, 1] == 1 && arr[2, 1] == 1 && arr[1,1]==0)

{ **return** 4; } // для клетки 0 1

**if** (arr[0, 2] == 1 && arr[1, 1] == 1 && arr[2,0]==0)

{ **return** 0; }

**if** (arr[0, 2] == 1 && arr[2, 0] == 1 && arr[1,1]==0)

{ **return** 4; }

**if** (arr[0, 2] == 1 && arr[1, 2] == 1 && arr[2,2]==0)

{ **return** 2; }

**if** (arr[0, 2] == 1 && arr[2, 2] == 1 && arr[1,2]==0)

{ **return** 5; } // для клетки 0 2

**if** (arr[1, 0] == 1 && arr[2, 0] == 1 && arr[0,0]==0)

{ **return** 6; }

**if** (arr[1, 0] == 1 && arr[1, 1] == 1 && arr[1,2]==0)

{ **return** 5; }

**if** (arr[1, 0] == 1 && arr[1, 2] == 1 && arr[1,1]==0)

{ **return** 4; } //для клетки 1 0

**if** (arr[1, 1] == 1 && arr[1, 0] == 1 && arr[1,2]==0)

{ **return** 5; }

**if** (arr[1, 1] == 1 && arr[1, 2] == 1 && arr[1,0]==0)

{ **return** 3; }

**if** (arr[1, 1] == 1 && arr[2, 1] == 1 && arr[0,1]==0)

{ **return** 7; }

**if** (arr[1, 1] == 1 && arr[2, 0] == 1 && arr[0,2]==0)

{ **return** 8; }

**if** (arr[1, 1] == 1 && arr[2, 2] == 1 && arr[0,0]==0)

{ **return** 6; } //для клетки 1 1

**if** (arr[1, 2] == 1 && arr[2, 2] == 1 && arr[0,2]==0)

{ **return** 8; } //для клетки 1 2

**if** (arr[2, 0] == 1 && arr[2, 1] == 1 && arr[2,2]==0)

{ **return** 2; }

**if** (arr[2, 0] == 1 && arr[2, 2] == 1 && arr[2,1]==0)

{ **return** 1; } //для клетки 2 0

**if** (arr[2, 1] == 1 && arr[2, 2] == 1 && arr[2,0]==0)

{ **return** 0; } //для клетки 2 1

**return** 9;

}

**private** **void** тупойToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

**if** (Ai == 1 || Ai == 0)

{

Ai = 2;

label2.Text = ("Игра с ИИ");

NewGames();

}

}

**private** **void** умныйToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

**if** (Ai == 0 || Ai == 2)

{

Ai = 1;

label2.Text = ("Игра с ИИ");

NewGames();

}

}

**private** **void** AI()

{

**if** (element == 2 && Ai == 1)

{

**int** \_sitatuation = CritSituation();

**if** (\_sitatuation == 0)

{

**if** (arr[2, 0] == 0)

{ x = 167; y = 178; Nought(2, 0); element = 1; **return**; }

**else** { funcCornerForAI(); **return**; }

}

**if** (\_sitatuation == 1)

{

**if** (arr[2, 1] == 0)

{ x = 167; y = 109; Nought(2, 1); element = 1; **return**; }

**else**{ funcCornerForAI(); **return**;}

}

**if** (\_sitatuation == 2)

{

**if** (arr[2, 2] == 0)

{ x = 167; y = 41; Nought(2, 2); element = 1; **return**; }

**else** {funcCornerForAI(); **return**;}

}

**if** (\_sitatuation == 3)

{

**if** (arr[1, 0] == 0)

{ x = 100; y = 178; Nought(1, 0); element = 1; **return**; }

**else** {funcCornerForAI(); **return**;}

}

**if** (\_sitatuation == 4)

{

**if** (arr[1, 1] == 0)

{ x = 100; y = 109; Nought(1, 1); element = 1; **return**; }

**else** {funcCornerForAI(); **return**;}

}

**if** (\_sitatuation == 5)

{

**if** (arr[1, 2] == 0)

{ x = 100; y = 41; Nought(1, 2); element = 1; **return**; }

**else**{ funcCornerForAI(); **return**;}

}

**if** (\_sitatuation == 6)

{

**if** (arr[0, 0] == 0)

{ x = 25; y = 178; Nought(0, 0); element = 1; **return**; }

**else** {funcCornerForAI(); **return**;}

}

**if** (\_sitatuation == 7)

{

**if** (arr[0, 1] == 0)

{ x = 25; y = 108; Nought(0, 1); element = 1; **return**; }

**else** {funcCornerForAI(); **return**;}

}

**if** (\_sitatuation == 8)

{

**if** (arr[0, 2] == 0)

{ x = 25; y = 41; Nought(0, 2); element = 1; **return**; }

**else** { funcCornerForAI(); **return**; }

}

**if** (\_sitatuation == 9)

{

**if** (arr[1, 1] == 0)

{ x = 100; y = 109; Nought(1, 1); element = 1; **return**; }

**if** (arr[0, 0] == 0)

{ x = 25; y = 178; Nought(0, 0); element = 1; **return**; }

**if** (arr[2, 2] == 0)

{ x = 167; y = 41; Nought(2, 2); element = 1; **return**; }

**if** (arr[2, 0] == 0)

{ x = 167; y = 178; Nought(2, 0); element = 1; **return**; }

**if** (arr[0, 2] == 0)

{ x = 25; y = 41; Nought(0, 2); element = 1; **return**; }

}

funcRandForAI();

}

}

**void** funcCornerForAI()

{

**if** (arr[1, 1] == 0)

{ x = 100; y = 109; Nought(1, 1); element = 1; **return**; }

**if** (arr[0, 0] == 0)

{ x = 25; y = 178; Nought(0, 0); element = 1; **return**; }

**if** (arr[2, 2] == 0)

{ x = 167; y = 41; Nought(2, 2); element = 1; **return**; }

**if** (arr[2, 0] == 0)

{ x = 167; y = 178; Nought(2, 0); element = 1; **return**; }

**if** (arr[0, 2] == 0)

{ x = 25; y = 41; Nought(0, 2); element = 1; **return**; }

funcRandForAI();

}

**void** funcRandForAI()

{

**int** n, m;

Random rnd = **new** Random();

n = rnd.Next(3);

m = rnd.Next(3);

**if** (n == 0) x = 25;

**if** (n == 1) x = 100;

**if** (n == 2) x = 167;

**if** (m == 0) y = 178;

**if** (m == 1) y = 109;

**if** (m == 2) y = 41;

**if** (arr[n, m] == 0)

{

Nought(n, m);

element = 1;

x = 0;

y = 0;

MouseX = 0;

MouseY = 0;

n = 0;

m = 0;

**return**;

}

**else**

{

**for** (**int** i = 0; i < 3; i++)

{

**for** (**int** j = 0; j < 3; j++)

{

**if** (arr[i, j] == 0)

{

n = i; m = j;

**if** (n == 0) x = 25;

**if** (n == 1) x = 100;

**if** (n == 2) x = 167;

**if** (m == 0) y = 178;

**if** (m == 1) y = 109;

**if** (m == 2) y = 41;

Nought(n, m);

element = 1;

}

}

}

}

}

**private** **void** новаяИграToolStripMenuItem\_Click(object sender, EventArgs e)

{

button1.Visible = **false**;

label3.Visible = **true**;

label4.Visible = **true**;

NewGames();

}

**private** **void** правилаToolStripMenuItem\_Click(object sender, EventArgs e)

{

MessageBox.Show("Для начала игры необходимо в настройках выбрать режим игры: с компьютером или человеком. Нажать кнопку 'Старт'.\n\nИгроки ходят по очереди, ставя крестик или нолик. Первый ход делает игрок, ставящий крестики. На ход дается 15 секунд.\n\nЧтобы совершить ход, нужно на рабочем поле поставить кликом мышью крестик или нолик.\nЕсли игрок не уложился в отведенное время, право хода передается сопернику.\n\nПервый, выстроивший в ряд 3 своих фигур по вертикали, горизонтали или диагонали, выигрывает.\n\n Для повтора игры, выбрать в 'Меню' пункт 'Новая игра'. ");

}

**private** **void** оИгреToolStripMenuItem\_Click(object sender, EventArgs e)

{

MessageBox.Show(" Версия игры 1.0\n Команда мечты\n 1. Авдеев Илья Анатольевич, 11-СБК, руководитель проекта\n 2. Новожилова Валерия Михайловна, 11-СБК, дизайнер \n 3. Сазонов Иван Михайлович, 11-СБК, инженер по требованиям \n 4. Рябчиков Алексей Андреевич, 11-СБК, системный инженер\n 5. Мурындин Станислав Юрьевич, 11-СБК, инженер – программист\n 6. Козлов Сергей Владимирович, 11-СБК, инженер по тестированию\n 7. Касатов Егор Сергеевич, 11-СБК, тестировщик");

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace Лабораторная\_работа.КрестикиНолии

{

**public** partial **class** Form1 : Form

{

**public** Form form2;

**public** Form1()

{

InitializeComponent();

**this**.TransparencyKey = Color.White;

**this**.BackgroundImage = **new** Bitmap(GetType(), "Fons.jpg");

**this**.pictureBox1.Image = **new** Bitmap(GetType(), "Close.jpg");

**this**.pictureBox2.Image = **new** Bitmap(GetType(), "Svernut.jpg");

**this**.menuStrip1.BackgroundImage = **new** Bitmap(GetType(), "Fons.jpg");

}

**private** **void** Form1\_paint(object sender, PaintEventArgs e)

{

}

**private** **void** Form1\_Load(object sender, EventArgs e)

{

}

**private** **void** pictureBox1\_Click\_1(object sender, EventArgs e)

{

timer1.Enabled = **true**;

timer1.Start();

timer1\_Tick(sender, e);

} // Запускает таймер выключения WindowsForm

**private** **void** timer1\_Tick(object sender, EventArgs e)

{

**for** (; Opacity > 0; Opacity -= .01, System.Threading.Thread.Sleep(10)) ;

**if** (**this**.Opacity <= 0)

{

//Таймер останавливается

timer1.Stop();

//Форма закрывается

**this**.Close();

} // выключение формы

}

**private** **void** pictureBox2\_Click(object sender, EventArgs e)

{

WindowState = FormWindowState.Minimized;

}

**private** **void** button1\_Click\_1(object sender, EventArgs e)

{

form2 = **new** Form2();

form2.ShowDialog();

}

}

}