МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Измерительно-вычислительные комплексы»

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | |  | |  |  |
| |  |  | | --- | --- | | ***Подп. и дата*** |  | | ***Инв. № дубл.*** |  | | ***Взам. инв. №*** |  | | ***Подп. и дата*** |  | | ***Инв. № подл*** |  | |  | | Тестовая документация  на лабораторную работу №9  по дисциплине «Алгоритмы и структуры данных»  Тема «Разработка компьютерной игры Крестики Нолики» | | | | | | |
|  | | |  | |  | | | |
|  | | Исполнитель  студент гр. ИСТбд-21  Салова А.С.  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2024 г. | | | |
|  | | 2024 | | | | | | |

1. **Описание тестируемых функций**
2. **draw\_grid():**

* **Описание:** Отрисовка сетки игрового поля.
* **Тестируемые моменты:**
  + Корректная отрисовка линий сетки 3x3.
  + Правильное размещение строк и столбцов.

1. **draw\_board():**

* **Описание:** Отображение символов (Крестики/Нолики) на игровом поле.
* **Тестируемые моменты:**
  + Корректная отрисовка символов X и O.
  + Обновление поля после каждого хода.

1. **on\_click():**

* **Описание:** Обработка действий пользователя по нажатию на ячейки игрового поля.
* **Тестируемые моменты:**
  + Установка символа в выбранную ячейку.
  + Корректная смена ходов между игроком и AI.
  + Обработка конца игры (выигрыш/ничья).

1. **bot\_move():**

* **Описание:** Ход компьютера в режиме AI.
* **Тестируемые моменты:**
  + Корректный выбор ячейки для хода с учётом победных комбинаций и блокировки ходов игрока.
  + Корректное размещение символа AI на поле.

1. **check\_winner():**

* **Описание:** Проверка наличия победителя после каждого хода.
* **Тестируемые моменты:**
  + Выявление победителя по горизонталям, вертикалям и диагоналям.
  + Правильное завершение игры при наличии победителя.
  + Отображение корректного сообщения о результате игры.

1. **update\_status():**

* **Описание:** Обновление статуса игры (чей ход, результат игры).
* **Тестируемые моменты:**
  + Корректное отображение текста для текущего игрока.
  + Отображение сообщений о победе, ничьей или продолжении игры.

1. **reset\_game():**

* **Описание:** Сброс игрового поля и параметров игры.
* **Тестируемые моменты:**
  + Очистка поля и возврат к исходному состоянию.
  + Сброс состояния игры (игрока, хода, окончания игры).
  + Корректное начало новой игры.

**2. Mind Map для тестирования**

**Основные категории тестирования:**

1. **Инициализация игры:**
   * Корректное создание UI (сеток, кнопок и текстовых меток).
   * Убедиться, что все элементы интерфейса видимы и работают.
2. **Игровой процесс:**
   * Правильная обработка кликов по игровым ячейкам.
   * Чередование ходов между игроком и компьютером.
   * Сохранение и обновление состояния игрового поля.
3. **AI:**
   * Адекватность выбора ячейки для хода.
   * Проверка, чтобы AI блокировал победные ходы игрока.
   * Обеспечение победных ходов для AI при возможности.
4. **Определение результатов:**
   * Проверка правильности завершения игры (выигрыш/ничья).
   * Отображение соответствующих сообщений.
5. **Рестарт:**
   * Полное обновление интерфейса и состояния после рестарта.

**3. Чек-лист**

| **№** | **Тестируемая функция** | **Шаги выполнения** | **Ожидаемый результат** | **Статус** |
| --- | --- | --- | --- | --- |
| 1 | draw\_grid() | Запуск игры | Корректная отрисовка сетки | Пройдено/Не пройдено |
| 2 | draw\_board() | Ходы игроков и AI | Корректная отрисовка символов | Пройдено/Не пройдено |
| 3 | on\_click() | Клик по пустой ячейке | Установка символа игрока | Пройдено/Не пройдено |
| 4 | bot\_move() | ИИ делает ход | Логический выбор AI и обновление поля | Пройдено/Не пройдено |
| 5 | check\_winner() | Проверка результатов игры | Верное определение победителя | Пройдено/Не пройдено |
| 6 | reset\_game() | Нажатие кнопки "Сбросить игру" | Полная очистка поля | Пройдено/Не пройдено |

**4. Тест-кейсы**

**Тест-кейс 1: Проверка работы режима игрок против игрока (PvP)**

**Цель:** Убедиться в правильной работе механики игры двух игроков.  
**Предусловия:** Приложение запущено, выбрана игра PvP.  
**Шаги:**

1. Нажать на пустую ячейку.
2. Сделать ходы обоими игроками.
3. Завершить игру (победа одного из игроков или ничья).  
   **Ожидаемый результат:** Корректное чередование ходов, правильное завершение игры.

**Тест-кейс 2: Проверка игры с компьютером (PvE)**

**Цель:** Проверить работу AI.  
**Предусловия:** Приложение запущено, выбран режим PvE.  
**Шаги:**

1. Сделать первый ход игроком.
2. Проверить ответный ход AI.
3. Продолжать игру до завершения.  
   **Ожидаемый результат:** AI делает оптимальные ходы, корректно реагирует на действия игрока.

**Тест-кейс 3: Проверка алгоритма определения победителя**

**Цель:** Убедиться в правильной обработке всех возможных вариантов победы.  
**Предусловия:** Игра начата.  
**Шаги:**

1. Построить горизонтальную линию.
2. Построить вертикальную линию.
3. Построить диагональную линию.  
   **Ожидаемый результат:** Программа корректно определяет победителя в каждом случае.

**Тест-кейс 4: Проверка рестарта игры**

**Цель:** Убедиться в корректности сброса игры.  
**Предусловия:** Игра завершена или в процессе.  
**Шаги:**

1. Сделать несколько ходов.
2. Нажать кнопку "Сбросить игру".  
   **Ожидаемый результат:** Игровое поле очищается, параметры сбрасываются.

**Тест-кейс 5: Проверка определения ничьей**

**Цель:** Убедиться, что игра корректно определяет ничью при полном заполнении игрового поля.  
**Предусловия:** Игра начата.  
**Шаги:**

1. Заполнить игровое поле без построения победной комбинации.
2. Сделать последний ход, заполняющий поле.  
   **Ожидаемый результат:** Программа правильно определяет ничью и выводит соответствующее сообщение.

**Тест-кейс 6: Проверка заблокированных ячеек**

**Цель:** Убедиться, что игроки и AI не могут делать ходы в уже занятые ячейки.  
**Предусловия:** Игра начата.  
**Шаги:**

1. Сделать ход в любую ячейку.
2. Повторно кликнуть на ту же ячейку.  
   **Ожидаемый результат:** Программа игнорирует клик по занятой ячейке.

**Тест-кейс 7: Проверка выбора первого игрока**

**Цель:** Убедиться, что игрок может выбрать, кто начинает игру.  
**Предусловия:** Игра запущена, пользователь находится в меню выбора режима.  
**Шаги:**

1. Выбрать первого игрока (человек или AI).
2. Наблюдать за первым ходом.  
   **Ожидаемый результат:** Первый ход выполняет выбранный игрок.

**Тест-кейс 8: Проверка сообщений об ошибках**

**Цель:** Убедиться, что игра корректно обрабатывает ошибки (например, некорректные ходы).  
**Предусловия:** Игра запущена.  
**Шаги:**

1. Попробовать выполнить некорректное действие (например, выбрать занятую ячейку).
2. Проверить реакцию программы.  
   **Ожидаемый результат:** Игра отображает соответствующее сообщение об ошибке или игнорирует действие.

**5. Методы тестирования**

1. **Функциональное тестирование:**  
   Проверка всех функций программы на соответствие требованиям.
2. **Тестирование интерфейса:**
   * Проверка корректности отображения элементов интерфейса.
   * Проверка доступности кнопок, текстовых меток и их правильного состояния.
3. **Тестирование производительности:**
   * Оценка времени реакции программы на действия пользователя.
   * Проверка скорости работы AI при сложных ситуациях на поле.
4. **Проверка устойчивости:**
   * Тестирование игры на некорректные действия (например, многократные клики).
   * Проверка поведения программы при резком завершении работы и последующем перезапуске.
5. **Кроссплатформенное тестирование:**
   * Проверка работоспособности программы на различных операционных системах (Windows, macOS, Linux).

**6. Заключение**

Тестовая документация охватывает все основные аспекты функциональности программы "Крестики-нолики", включая тестирование пользовательского интерфейса, игровых сценариев, алгоритма AI, а также обработки ошибок.

Тестирование проводится в соответствии с чек-листом и тест-кейсами, чтобы убедиться в полной работоспособности программы, её надежности и соответствию заявленным требованиям.

Для успешной сдачи проекта рекомендуется провести тщательную проверку всех предусмотренных сценариев и обеспечить исправление выявленных ошибок.

**Готово к выполнению тестирования.**