

```

}

//Diese Funktion überprüft ob es ein Gewinner gibt, gibt N0 für nein, U1 für
1-Verweis
public String TestForWin()
{
    bool fieldIsFull = true;
    for(int i = 0; i < 10; i++)
    {
        for(int B = 0; B < 5; B++)
        {
            Color PossibleWinner = Feld[i, B].BackColor;
            if (PossibleWinner == Color.White) { fieldIsFull = false; }
            else if (TestRows(i, B, PossibleWinner))
            {
                if (pEins.Color == PossibleWinner)
                {
                    return pEins.Name;
                }
                else
                {
                    return pZwei.Name;
                }
            }
        }
    }

    if (fieldIsFull) { return "U1"; } // Unentschieden
    return "N0";
}

// Nimmt kordinaten eines Feldes und eine Spielerfarbe und überprüft ob die
1-Verweis
public bool TestRows(int A, int B, Color toTestFor)
{
    //Test for stones underneath
    try
    {
        if(Feld[A,B+1].BackColor == toTestFor && Feld[A, B + 2].BackColor =
        {
            return true;
        }
    }
}

```

Dokumentation

4 Gewinnt

M404

24.April.2020

Nicola Fioretti

Dokumentation

Form1:

Btn_StartGame_Click()

Diese Funktion wird durch einen Klick auf den Button im Form1 aufgerufen. Sie validiert die eingegebenen Textwerte durch ein Unterprogramm, wenn diese Valide sind wird Form2 gestartet und die beiden Namen werden als Startwerte mitgegeben.

TestInputs(String One, String Two) Void

Diese Funktion nimmt 2 Strings und prüft ob diese nicht identisch sind und keine Zahlen beinhalten, diese Funktion returnt nichts, ändert jedoch eine Boolesche variable.

Form2:

Dieses Form ist das Spiel. Es nimmt 2 Strings als Inputs, diese sind die Namen der beiden Spieler in dieser Funktion werden die Felder geleert und Variablen gefüllt.

Form2_FormClosing()

Diese Funktion beendet die gesamte Applikation. Beim Start wird Form1 aufgerufen, Form1 kann daher nicht geschlossen werden. Beim wechseln auf Form2 wird das Form1 nur minimiert, nicht geschlossen. Diese Funktion beendet jedoch beim Schliessen von Form2 die Gesamte Applikation, auch das Form1 im Hintergrund.

GetColorOfCurrentPlayer()

Diese Funktion gibt die Farbe des Spielers welcher aktuell am Zug ist zurück.

switchTurn()

Diese Funktion wechselt den Spieler welcher aktuell am Zug ist.

SetStoneInRow(Button[] row, Color toSet) bool

Diese Funktion nimmt eine Reihe entgegen in form eines 1D Button Array und setzt die Farbe des Untersten unbesetzten Feldes dieser reihe auf die Farbe, welche entgegengenommen wird. Falls die Komplette Reihe voll ist gibt die Funktion eine Fehlermeldung aus und returnt false.

goDown(Button[], Color toSet, int Pos)

Diese Funktion wird rekursiv aufgerufen. Wenn das Feld der aktuellen Position leer ist wird es entsprechend eingefärbt und das Feld oberhalb wird geleert. Wenn das Feld zuvor leer war wird die Funktion erneut aufgerufen bis unterhalb ein eingefärbtes Feld ist oder der Boden erreicht ist.

Row1-10Click()

Diese Funktion ruft die SetStoneInRow() Funktion auf. Wenn ein stein gesetzt werden kann wird die Funktion switchTurn() aufgerufen.

TestForWin()

Diese Funktion überprüft ob es ein Gewinner gibt oder das Feld voll ist. Die Rückgabe von „U1“ steht für Unentschieden, „N0“ steht für kein Gewinner. Wenn es ein Gewinner gibt wird dessen Name reutrn. Die Überprüfung auf einen Gewinner erfolgt durch aufrufe der Funktion TestRows.

TestRows(int A, int B, Color toTestFor)

Diese Funktion überprüft ob der durch Koordinaten angegebene stein teil einer 4-er Reihe ist. Es wird nur nach unten, nach rechts, diagonal nach unten rechts und Diagonal nach unten Links überprüft da diese Funktion für jeden Stein aufgerufen wird.

ClearField()

Diese Funktion setzt alle Felder auf Leer.

Btn_Send_click()

Diese Funktion nimmt die Eingabe aus der Textbox entgegen und wenn diese eine zahl von 1-10 ist wird die entsprechende Reihe aufgerufen.

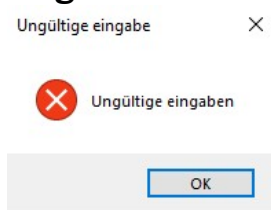
Testfälle

Teil 1 Login Form Testfälle:

The screenshot shows a window titled 'Form1' with a light gray background. It contains two text input fields. The first field is labeled 'Spieler 1' and the second is labeled 'Spieler 2'. Below these fields is a large gray button labeled 'Start Game'.

Fall 1: Beide Felder Leer:

Folgende Meldung erscheint



Durch einen Klick auf „Ok“ oder auf „X“ Schliesst sich das Fenster und man kann erneut eine Eingabeversuch unternehmen.

Fall 2: Ein Feld Leer:

Folgende eingaben wurden eingereicht:

Spieler 1: A

Spieler 2:

Das Spiel startet wie gewohnt, es Treten keine Fehler auf auch wenn Spieler 2 gewinnt.

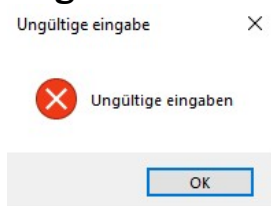
Fall 3: Gleiche Eingabe in beide Felder:

Folgende eingaben wurden eingereicht:

Spieler 1: A

Spieler 2: A

Folgende Meldung erscheint



Durch einen Klick auf „Ok“ oder auf „X“ Schliesst sich das Fenster und man kann erneut eine Eingabeversuch unternehmen.

Teil 2: Spielfeld Testfälle

Form2

1	2	3	4	5	6	7	8	9	10

Spieler A: Rot Spieler B: Blau

Spieler: Spieler A ist am Zug

send

Fall 1: Klick auf einen Button:

Form2

1	2	3	4	5	6	7	8	9	10

Spieler A: Rot Spieler B: Blau

Spieler: Spieler B ist am Zug

send

Die Farbe des Untersten feldes dieser reihe ändert sich und das Label welches anzeigt welcher spieler am zug ist Aktualisiert sich.

Fall 2: Klick auf eine Volle Reihe:

Form2

1	2	3	4	5	6	7	8	9	10

Spieler A: Rot Spieler B: Blau

Spieler: Spieler B ist am Zug

send

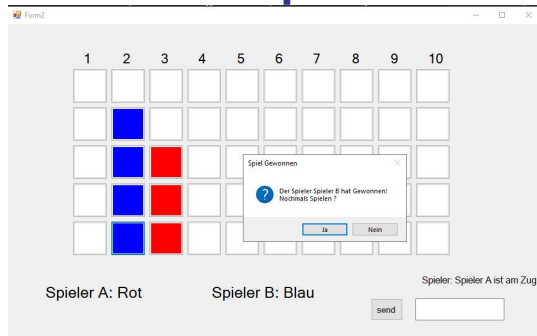
Fehler

✖ Diese Reihe ist bereits Voll

OK

Es Erscheint eine Fehlermeldung welche durch einen Klick auf „X“ oder „Ok“ geschlossen werden kann. Es ist immer noch derselbe Spieler am Zug.

Fall 3: Ein Spieler hat 4 Steine nebeneinander:



Es erscheint eine Nachricht, dass der Spieler gewonnen hat. Es wird gefragt ob man erneut spielen möchte, falls man auf ja Klick leert sich das Spielfeld, wenn man auf Nein klickt wird die Applikation geschlossen. Die MessageBox kann nicht durch einen Klick auf „X“ Geschlossen werden.

Fall 4: Eine Zahl von 1-10 wird in das Eingabefeld geschrieben und mit „Send“ bestätigt:

In der Ausgewählten Reihe passiert dasselbe wie bei Fall 1.

Fall 5: Eingabe von Negativzahlen, 0, Zahlen über 10, Buchstaben:

Es erscheint folgende Meldung:

