

Politechnika Śląska

Wydział Matematyk Stosowanej

Kierunek Informatyka



**Politechnika
Śląska**

Gliwice, 04.02.2021

Aplikacje Webowe

projekt zespołowy

"Kółko i Krzyżyk"

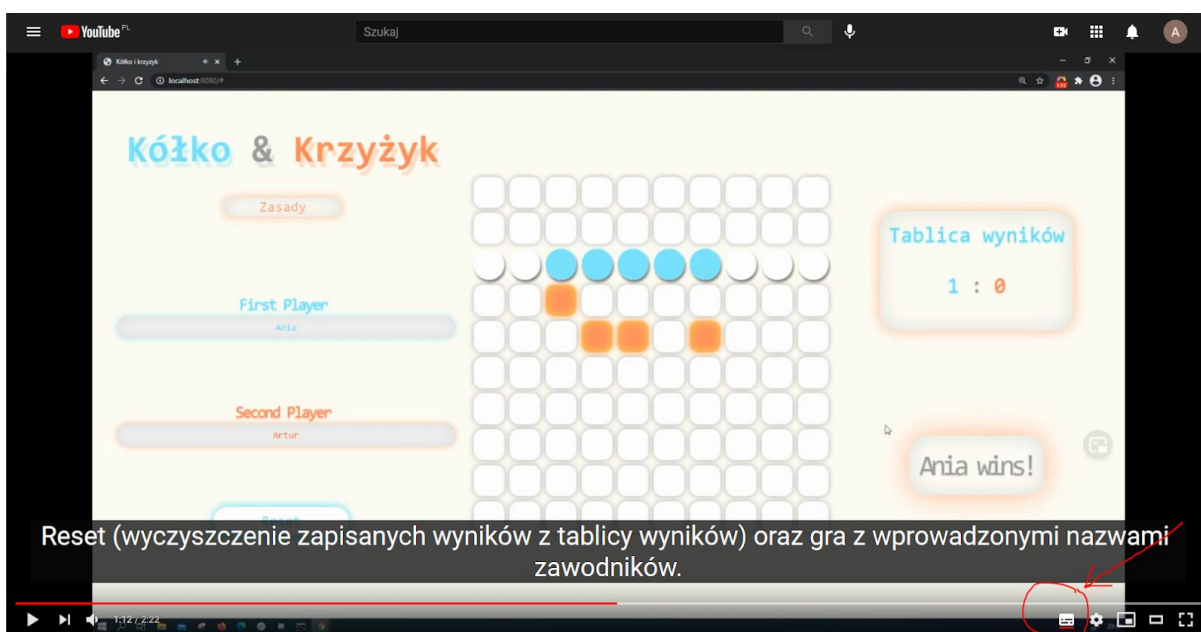
Skład zespołu projektowego:

**Anna Dzierżawa
Artur Giza**

1. Opis projektu i prezentacja rozgrywki.

Nasz projekt to gra przeglądarkowa, która została stworzona z wykorzystaniem technologii webowych takich jak HTML, CSS i Javascript. Pozwala ona na rywalizację maksymalnie 2 zawodników. Gracze stawiają na przemienne figury o odpowiedniej kolorystyce dążąc do zajęcia 5 pól w jednej linii. Wygrywa ten z graczy, któremu jako pierwszemu uda ułożyć się 5 znaków w jednej linii na planszy o wymiarach 10x10.

Prezentację rozgrywki można zobaczyć przechodząc do serwisu Youtube za pośrednictwem linku: <https://youtu.be/YSeCkpxoLzo>. Do filmu zostały dodane napisy. Na poniższym screenie ekranu możemy zobaczyć w którym miejscu można je uruchomić:



Link do pobrania filmu (bez napisów):

<https://www66.zippyshare.com/v/uav1dE4d/file.html>

2. Informacje dla użytkownika.

Aby odpalić grę i cieszyć się m.in. efektami dźwiękowymi, należy postawić lokalny serwer. Najprostszym rozwiązaniem jest uruchomienie serwera za pomocą narzędzia Node.js, aby to zrobić, należy wykonać kroki z poniższego screena.

```
http-server
Microsoft Windows [Version 10.0.18363.1316]
(c) 2019 Microsoft Corporation. Wszelkie prawa zastrzeżone.

D:\Szkola\Informatyka MS\Semestr 5\Aplikacje webowe\kółko i krzyżyk\do_wysłania>npm install --global http-server
C:\Users\Admin\AppData\Roaming\npm\http-server -> C:\Users\Admin\AppData\Roaming\npm\node_modules\http-server\bin\http-s
erver
C:\Users\Admin\AppData\Roaming\npm\hs -> C:\Users\Admin\AppData\Roaming\npm\node_modules\http-server\bin\http-server
+ http-server@0.12.3
updated 4 packages in 1.692s

D:\Szkola\Informatyka MS\Semestr 5\Aplikacje webowe\kółko i krzyżyk\do_wysłania>http-server
Starting up http-server, serving ./
Available on:
  http://192.168.56.1:8080
  http://192.168.99.1:8080
  http://192.168.0.103:8080
  http://127.0.0.1:8080
Hit CTRL-C to stop the server
```

Oczywiście serwer http może zostać postawiony także za pomocą innych narzędzi.

Aby móc zagrać w grę wymagana jest mysz komputerowa lub touchpad, za pomocą których prowadzona jest rozgrywka. Klawiatura jest z kolei wymagana, jeśli gracze chcą wprowadzić swoje pseudonimy::



Pod tytułem "Kółko i Krzyżyk" możemy zauważyć przycisk Zasady, który pozwala graczowi po naciśnięciu poznać zasady jakie panują w grze:

Kółko & Krzyżyk

Zasady

The screenshot shows a game interface with a central modal window titled "Zasady gry w Kółko & Krzyżyk". The modal contains two bullet points: "Zawodnicy zaznaczają pola na planszy" and "Wygrywa zawodnik, który jako pierwszy zaznaczy 5 sąsiadujących pól". Below the text is a "Zamknij" button. In the background, there are two 5x5 grids representing the game board. On the right side, there is a "Tablica" (Scoreboard) showing a score of 0. On the left side, there are labels for "Player" and "Player" with names "Artur" and "Ania" respectively.

Player

Artur

Zasady gry w Kółko & Krzyżyk

- Zawodnicy zaznaczają pola na planszy
- Wygrywa zawodnik, który jako pierwszy zaznaczy 5 sąsiadujących pól

Zamknij

Tablica

0

Player

Ania

Po wprowadzeniu nazw przez zawodników można rozpocząć rozgrywkę. O kolejności ruchu informuje nas pole widoczne pod tablicą wyników:

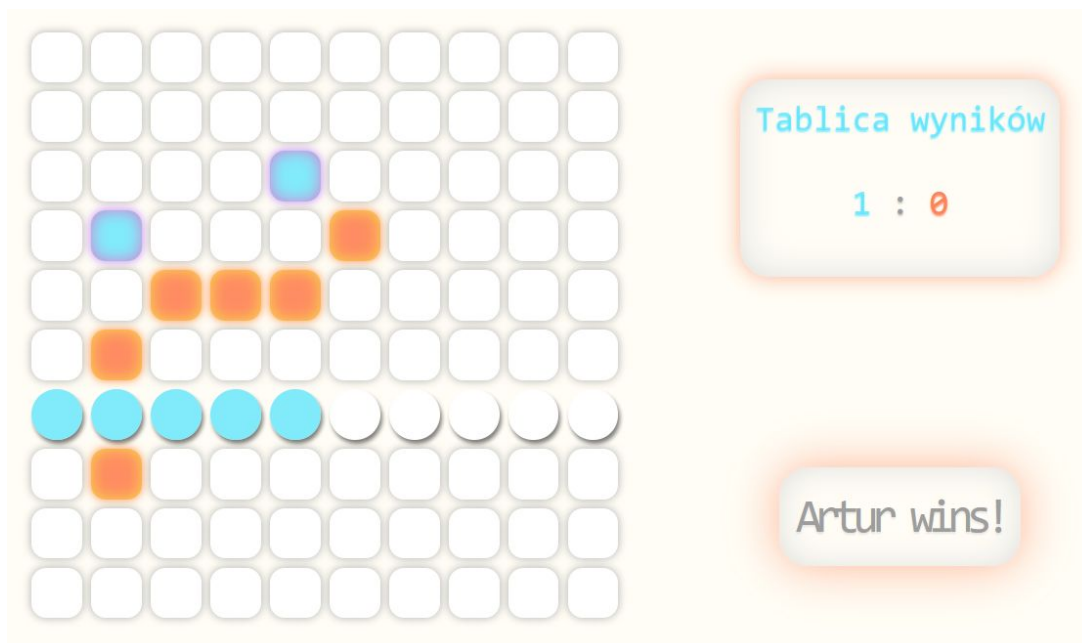
The screenshot shows a scoreboard titled "Tablica wyników" with a score of 0 : 0. Below the scoreboard is a button labeled "Artur".

Tablica wyników

0 : 0

Artur

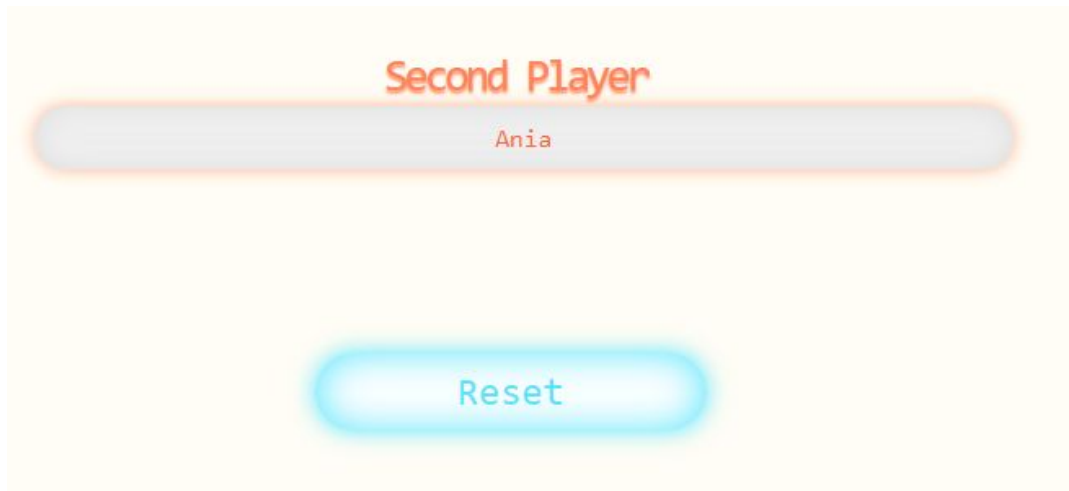
W momencie kiedy jeden z zawodników wygra rundę, to samo pole wskazuje zwycięzcę a w tablicy wyników możemy zauważyć zmieniający się wynik:



Aby przejść do następnej rundy należy nacisnąć na plansze i wybrać opcję *Nowa Gra*:



Aby zresetować tablicę wyników i rozpocząć rozgrywkę od nowa należy nacisnąć przycisk Reset:

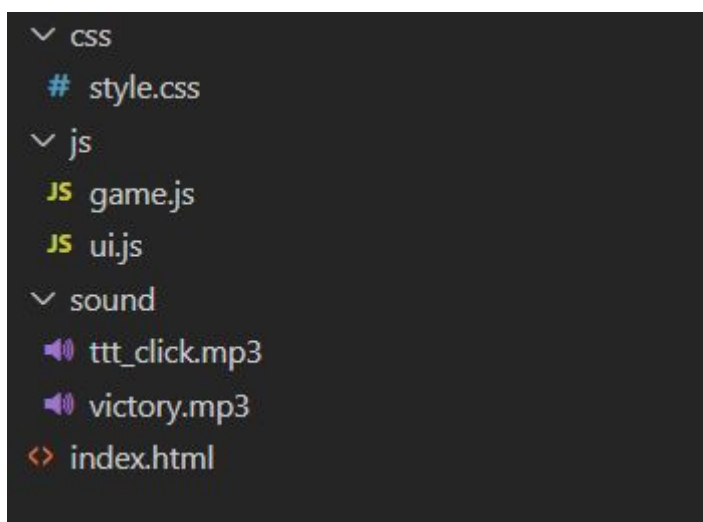


3. Informacje o wykorzystanych technologiach.

Logika aplikacji została zaprogramowana w języku JavaScript. Kod został podzielony na dwa foldery. W jednym z nich znajduje się warstwa odpowiedzialna za komunikację z użytkownikiem, która mając referencję do kodu hermetyzującego logikę gry pozwala wyświetlić stan rozgrywki oraz prowadzić interakcje z graczem.

Style gry zostały zdefiniowane za pomocą języka CSS. W celu informowania użytkownika, które pola zostały zaznaczone, do odpowiednich pól dynamicznie są przypisywane różne style.

Struktura projektu:



Fragment kodu z pliku game.js odpowiedzialny za sprawdzenie, czy gra dobiegła końca, ciekawy przykład transponowania macierzy w celu ujednolicenia procesu sprawdzania:

```
checkIfGameOver: function () {
  return this.checkGameAxes(this.getGameAxesAsRows(this.board));
},

getGameAxesAsRows: function (matrix) {
  return [...matrix, ...this.transpose(matrix), ...this.getDiagonalAxes(matrix)];
},

checkGameAxes: function (matrix) {
  for (let i = 0; i < matrix.length; i++) {
    let winner = this.checkAxis(matrix[i]);
    if (winner !== null) {
      return [winner, i];
    }
  }
  return [null, null];
},

checkAxis: function (row) {
  for (let i = 0; i < 5; i++) {
    if (row[i] === row[i + 1] && row[i + 1] === row[i + 2] && row[i + 2] === row[i + 3] && row[i + 3] === row[i + 4]) {
      return row[i];
    }
  }
  return null;
},

transpose: (matrix) => matrix[0].map((_, i) => matrix.map(x => x[i])),

getDiagonalAxes: function (matrix) {
  let diagonals = [[], []];
  for (let i = 0; i < matrix.length; i++) {
    diagonals[0].push(matrix[i][i]);
    diagonals[1].push(matrix[i][matrix[0].length - 1 - i]);
  }
  return diagonals;
},
```

W pliku game.js znajduje się obiekt zawierający wszelkie dane i logikę dotyczącą gry. Kod z pliku ui.js wykorzystuje ten obiekt w celu interakcji z użytkownikiem:

```
const StatusCodes = {  
  INVALID: 0,  
  VALID: 1,  
  DRAW: null,  
  PLAYER1WIN: true,  
  PLAYER2WIN: false,  
  COMPLETE: -1  
}  
  
const GameModel = {  
  board: [],  
  boardID: [],  
  player: true,  
  turnCount: 0,  
  moves: [],  
  status: StatusCodes.VALID,  
  score: [0, 0],  
  winAxis: null,  
}
```

niżej znajdują się już metody zawierające logikę gry.