


SPRAWOZDANIE NR			
Nazwa ćwiczenia	Gra kółko krzyżyk		 POLITECHNIKA BYDGOSKA Wydział Telekomunikacji, Informatyki i Elektrotechniki
Przedmiot	Programowanie obiektowe		
Student grupa	Paweł Jońca gr 7		
Data ćwiczeń	26.11.2024r	26.11.2024r	Data oddania sprawozdania

Implementacja popularnej gry kółko/krzyżyk z wykorzystaniem biblioteki **JavaFX**.



W ramach wprowadzenia do ćwiczenia prowadzący objaśnia i pokazuje jak korzystać z narzędzia **Scene Builder** do szybkiej budowy interfejsów **JavaFX**. Wspólnie ze studentami buduje prosty interfejs graficzny z komponentem **javafx.scene.control.Button** (zdefiniowany w pliku **fxml**), pokazuje jak obsługiwać zdarzenia i uzyskać dostęp do komponentów w kontrolerze poprzez wstrzykiwanie zależności.

```
package com.example.grakolkokrzyzyk;
import javafx.fxml.FXML;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
public class HelloController {
    @FXML
    private Label welcomeText; // Etykieta z informacją o stanie gry
    @FXML 29 usages
    private Button[][] boardButtons = new Button[3][3]; // Przyciski planszy 3x3

    private boolean isXTurn = true; // Określa, czy ruch należy do gracza X 6usages
```

```
@FXML
private void initialize() {
    boardButtons[0][0] = button00; boardButtons[0][1] = button01; boardButtons[0][2] = button02; // Inicjalizacja wiersza 1
    boardButtons[1][0] = button10; boardButtons[1][1] = button11; boardButtons[1][2] = button12; // Inicjalizacja wiersza 2
    boardButtons[2][0] = button20; boardButtons[2][1] = button21; boardButtons[2][2] = button22; // Inicjalizacja wiersza 3

    for (int i = 0; i < 3; i++) { // Ustawienie akcji dla przycisków
        for (int j = 0; j < 3; j++) {
            int r = i, c = j;
            boardButtons[i][j].setOnAction(event -> handleMove(r, c)); // Obsługa kliknięcia
        }
    }
    updateTurnLabel(); // Wyświetl gracza, który zaczyna
}
```

```

27     private void handleMove(int row, int col) { 1 usage
28         if (!boardButtons[row][col].getText().isEmpty()) return; // Ignoruj, jeśli pole zajęte
29
30         boardButtons[row][col].setText(isXTurn ? "X" : "O"); // Ustaw X lub O
31         boardButtons[row][col].setDisable(true); // Zablokuj pole po użyciu
32
33         if (checkWin()) { // Sprawdzenie wygranej
34             welcomeText.setText("Gracz " + (isXTurn ? "X" : "O") + " wygrał!");
35             disableBoard(); // Zablokowanie planszy po wygranej
36         } else if (isBoardFull()) { // Sprawdzenie remisu
37             welcomeText.setText("Remis!");
38         } else {
39             isXTurn = !isXTurn; // Zmiana gracza
40             updateTurnLabel(); // Aktualizacja informacji o graczu
41         }
42     }

```

```

43     private void updateTurnLabel() { 3 usages
44         welcomeText.setText("Ruch gracza " + (isXTurn ? "X" : "O")); // Wyświetl aktualnego gracza
45     }
46     private boolean checkWin() { 1 usage
47         for (int i = 0; i < 3; i++) { // Sprawdzenie wygranej w wierszach i kolumnach
48             if (checkCombination(boardButtons[i][0], boardButtons[i][1], boardButtons[i][2]) ||
49                 checkCombination(boardButtons[0][i], boardButtons[1][i], boardButtons[2][i])) {
50                 return true;
51             }
52         }
53         return checkCombination(boardButtons[0][0], boardButtons[1][1], boardButtons[2][2]) || // Sprawdzenie przekątnych
54             checkCombination(boardButtons[0][2], boardButtons[1][1], boardButtons[2][0]);
55     }

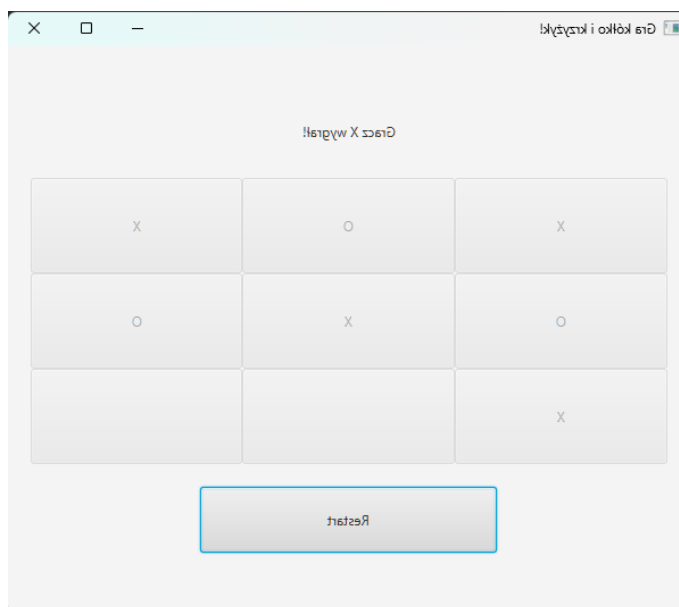
```

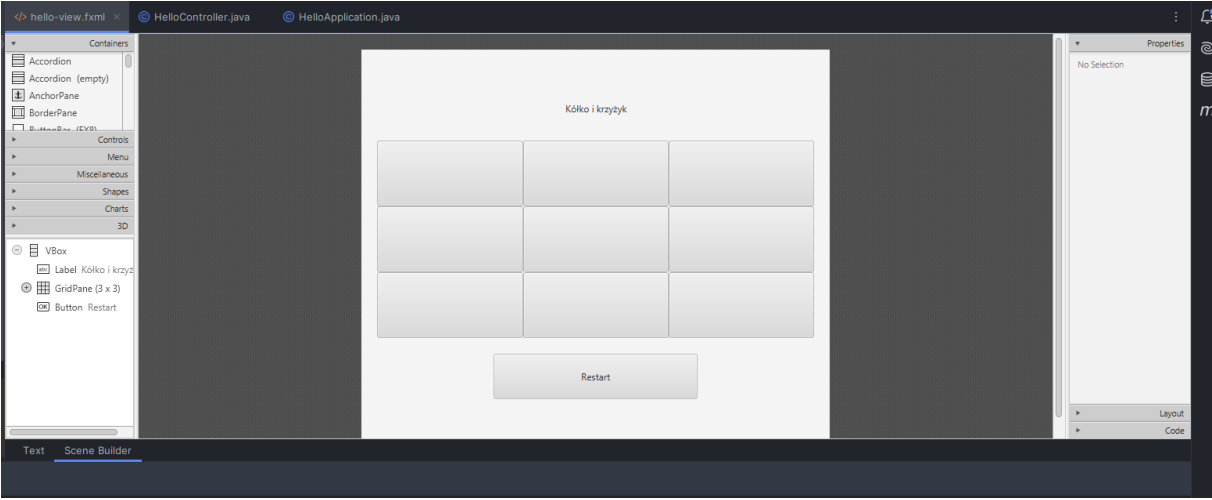
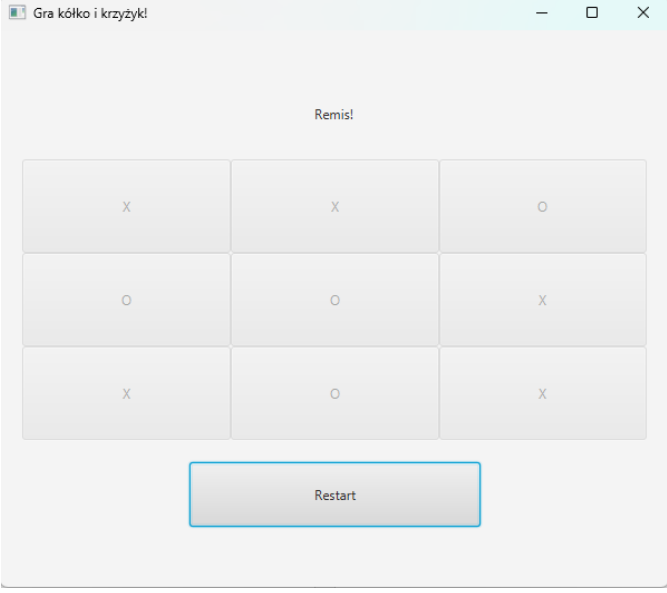
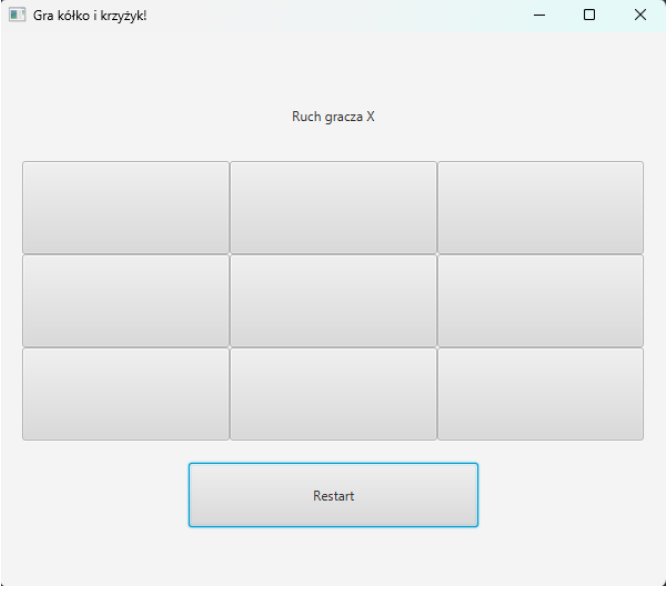
```

56     private boolean checkCombination(Button b1, Button b2, Button b3) { 4 usages
57         return !b1.getText().isEmpty() && b1.getText().equals(b2.getText()) && b2.getText().equals(b3.getText()); // Sprawdź, czy pola są takie same
58     }
59     private boolean isBoardFull() { 1 usage
60         for (Button[] row : boardButtons) { // Sprawdzenie, czy wszystkie pola są zajęte
61             for (Button button : row) {
62                 if (button.getText().isEmpty()) return false;
63             }
64         }
65         return true;
66     }
67     private void disableBoard() { 1 usage
68         for (Button[] row : boardButtons) { // Zablokuj wszystkie przyciski
69             for (Button button : row) {
70                 button.setDisable(true);
71             }
72         }
73     }
74     @FXML
75     private void resetGame() {
76         for (int i = 0; i < 3; i++) { // Reset planszy
77             for (int j = 0; j < 3; j++) {
78                 boardButtons[i][j].setText(""); // Usuń tekst
79                 boardButtons[i][j].setDisable(false); // Odblokuj przycisk
80             }
81         }
82         isXTurn = true; // Resetuj kolejność graczy
83         updateTurnLabel(); // Wyświetl komunikat startowy
84     }
85     // Przyciski planszy
86     @FXML private Button button00, button01, button02, button10, button11, button12, button20, button21, button22;
87 }
88

```

```
© HelloController.java  © HelloApplication.java x
1 package com.example.grakolkokrzyzyk;
2 import javafx.application.Application;
3 import javafx.fxml.FXMLLoader;
4 import javafx.scene.Scene;
5 import javafx.stage.Stage;
6 import java.io.IOException;
7
8 public class HelloApplication extends Application {
9     @Override
10    public void start(Stage stage) throws IOException {
11        FXMLLoader fxmlLoader = new FXMLLoader(HelloApplication.class.getResource("hello-view.fxml"));
12        Scene scene = new Scene(fxmlLoader.load(), 600, 500);
13        stage.setTitle("Gra kółko i krzyżyk!");
14        stage.setScene(scene);
15        stage.show();
16    }
17
18    public static void main(String[] args) {
19        launch();
20    }
21 }
```





Wnioski:

Zadanie pozwoliło mi nauczyć się, jak używać JavaFX. Dzięki stworzeniu gry kółko/krzyżyk zrozumiałem, jak działa podział na interfejs (FXML) i logikę (kontroler), a także jak obsługiwać zdarzenia przy klikaniu przycisków. Udało mi się też przećwiczyć podstawy pracy z tablicami i pętlami, co na pewno przyda mi się w przyszłości. Fajne zadanie bo szybko można przez długi czas cieszyć się rezultatem jak już aplikacja gra i można grać z kolegami w swoje własne kółko krzyżyk. Można byłoby dodać funkcjonalność do zapamiętywania wyników lub statystyk.