

Dokumentacja projektowa

RZUT MONETA

DOKUMENT	DOKUMENTACJA PROJEKTOWA	
TEMAT	RZUT MONETA	
WERSJA DOKUMENTU	DATA POWSTANIA WERSJI	PRZYCZYNA POWSTANIA WERSJI
2	26.07.2021 21:21:00	DODANIE MODYFIKACJI
AUTOR DOKUMENTU	TOMASZ CIEŚLIK	
STATUS	UKOŃCZONY	
PLIK DOKUMENTU	PROJEKT - ORZEŁ CZY RESZKA.DOCX	

1	Wymagania.....	3
2	Etap 1 – Utworzenie wersji konsolowej programu	4
2.1	Założenia	4
3	RZUT MONETA.....	5
3.1	Testy Jednostkowe	5
4	Etap 3 – Utworzenie aplikacji z interfejsem graficznym	6
5	Etap 4 – Dodatkowe prace	6
6	7	
7	Instalacja – Wdrożenie	8
7.1	Udostępnienie programu na Github.com	8

1 Wymagania

1. Utworzenie programu mającego symulować rzut monetą.
2. Program ma być stworzony w wersji konsolowej oraz w wersji okienkowej.
3. Użytkownik decyduje ile monet jednocześnie bierze udział w losowaniu – **maksymalnie 5.**
4. Użytkownik może zmienić nazwę awersu i rewersu monety.

2 Etap 1 – Utworzenie wersji konsolowej programu

2.1 Założenia

1. Program ma być wykonany przy wykorzystaniu **C#** technologii **.Net Framework**
2. Po uruchomieniu programu pojawia się pytanie „*Ile monet? – wpisz liczbę w przedziale od 1 do 5*”.
 - a) W przypadku braku wpisania liczby program automatycznie ma wylosować jedną monetę,
 - b) W przypadku wpisania liczby/ znaku spoza zdefiniowanego zakresu pojawia się komunikat:
 - a. „Podano błędną wartość wpisz liczbę w przedziale od 1 do 5”.
3. Wyniki losowania mają zostać przedstawione w następujący sposób: (poniżej opisano strukturę poszczególnych wierszy):
 - i. Wylosowano: (pojawia się szczegółowa informacja o wyniku losowania np.:
ORZEŁ – RESZKA – ORZEŁ --ORZEŁ
 - ii. *ORZEŁ – Z*,
 - iii. *RESZKA – Y*.
4. Program ma być zapętlony:
 - a) Po jego wykonaniu użytkownikowi wyświetla się następujące pytanie:
 - a. Czy chcesz jeszcze raz zagrać? (wpisz **"Nie"** jeśli chcesz zakończyć), następnie zatwierdź).

3 RZUT MONETA

3.1 Testy Jednostkowe

Automatyczne testy jednostkowe mają uwzględniać następujące przypadki testowe:

- 1) Losowanie:
 - a. 5 monet orzeł i reszka prawidłowo wprowadzonych
 - b. 1 moneta ze zmienionym opisem na Awers i Rewers – przypadek tworzony w trakcie realizacji etapu 4
 - c. Brak wprowadzonej ilości monet
 - d. Wprowadzona wartość cyfrowa spoza zdefiniowanego zakresu
 - e. Wprowadzony inny znak niż liczba
- 2) Kończenie działania programu:
 - a. Wpisanie Tak – oczekiwane zamknięcie programu
 - b. Wpisanie Nie – oczekiwanie dalsze działanie programu
 - c. Wpisanie innej wartości – oczekiwanie ponownego pytania w sprawie zamknięcia programu.

4 Etap 3 – Utworzenie aplikacji z interfejsem graficznym

1. Nazwa programu **Orzeł czy Reszka**
2. Program ma być wykonany przy wykorzystaniu **C#** technologii **.Net Framework**
3. Domyślna moneta o nominale 5 zł
4. Po uruchomieniu programu pojawia się pytanie „*Ile monet? (od 1 do 5)*”.
 - a) Domyślnie jest ustawiona wartość 1
 - b) W przypadku skasowania liczby lub w przypadku wpisania liczby/ znaku spoza zdefiniowanego zakresu pojawia się komunikat:
„Należy wpisać liczbę z przedziału od 1 do 5”.
5. Wyniki losowania mają zostać przedstawione w następujący sposób: (poniżej opisano strukturę poszczególnych wierszy):
 - a) *Wyniki – i poniżej graficznie przedstawione orły i reszki na monecie 5 zł*
 - b) *Liczba orłów*
 - c) *Liczba reszek*
6. *Poniżej na stałe przypięte pytanie „Ile monet? (od 1 do 5)”.*
7. *Rzuć monetą*

5 Etap 4 – Dodatkowe prace

- 1) Dodanie opcji zmiany Awersu i Rewersu w wersji okienkowej
- 2) Pokolorowanie, pogrubienie wyników w wersji konsolowej
- 3) Prace nad wyglądem graficznym – ustawienie rozmiaru okna wielkości oraz położenia danych.
- 4) Animacja losowania wersja okienkowa

6 Instalacja – Wdrożenie

Udostępnienie programu na [Github.com](https://github.com)