BracketCraft

JavaScript

Technologie Internetowe

Mateusz Bocak

125104, lab1

II rok informatyki stacjonarnie

26.01.2024 – v1.0

Spis treści

Zdefiniowanie Problemu

Problem dynamicznego generowania drabinek, w zależności od:

- Ilości uczestników
- Parzystości uczestników

Aplikacja powinna móc generować i wstawiać dynamiczne mecze – tzw. Moduły drabinki, w odpowiednich ilościach, w zależności od uczestników, zwycięzców wybieranych poprzez zaznaczenie radiobuttona.

Aplikacja również powinna dynamicznie chować i pokazywać formularz do tworzenia takiej drabinki, co uprości czytelność strony.

Propozycja rozwiązania:

Skrypt losuje uczestników z podanej listy przez użytkownika. Następnie przypisuje ich do poszczególnych meczy rundy pierwszej. Po wybraniu zwycięzców generuje rundę drugą zawierającą zwycięzców, itd.

Dodatkowo skrypt odwołując się do struktury HTML dokumentu, generuje dynamicznie moduły drabinki turniejowej.

Program umożliwia pokazanie i schowanie formularza do tworzenia drabinek, za pomocą odwoływania się do klasy elementów, które należy schować/pokazać i zmieniając im wartość display na none.

Sposób Rozwiązania

Obecny stan aplikacji jest w większości zgodny z powyższym konspektem. Aplikacja wykorzystuje podejście funkcyjne do radzenia sobie z generowaniem turnieju i obsługą strony. Użytkownik może zwijać i rozwijać formularz za pomocą czytelnych guzików, uruchamiających wybrane funkcje w aplikacji. Wykorzystywane jest dynamiczne umieszczanie elementów w DOM za pomocą języka skryptowego JavaScript. Zaimplementowano zapis danych z formularza do Localstorage oraz plików w formacie JSON, tak by w razie pomyłki lub omyłkowego odświeżenia strony, dane nie były tracone, a użytkownik nie musiał ponownie uzupełniać formularza. Wystarczy w takim wypadku ponownie wygenerować turniej.

Struktura plików

Ułożenie struktury katalogowej całego projektu w model MVC (Model View Controller), umożliwiające łatwe przeglądanie i dostosowywanie plików, w zależności od ich użycia.

- Doc/ - folder zawierający dokumentację javascript i HTML+CSS
- Src/ - folder zawierający całą strukturę aplikacji
    - Assets/ - folder zawierający potrzebne pliki i foldery
        - Css/ - folder zawierający kaskadowe style dla aplikacji oraz plik css odpowiadający responsywności strony na urządzeniach mobilnych
        - Images/ - folder zawierający poglądowe zdjęcia różnych typów drabinek turniejowych
        - Js/ - folder zawierający logikę napisaną za pomocą jezyka JavaScript, której działanie jest opisywane w tej dokumentacji,
    - Views/ - folder zawierający widoki prezentowane użytkownikowi – stronę główną

Aktualne rozwiązanie problemu

Obecna implementacja BracketCraft prezentuje skuteczne rozwiązanie problemu dynamicznego generowania drabinek turniejowych, biorąc pod uwagę ilość uczestników oraz parzystość tegoż grona. Skrypt napisany w języku JavaScript efektywnie obsługuje formularz, waliduje dane, generuje drabinkę turniejową, oraz interaktywnie przeprowadza użytkownika przez kolejne rundy, umożliwiając wybieranie zwycięzców.

Zalety Aktualnego Rozwiązania:

- Modularność i Czytelność Kodu: Kod został podzielony na funkcje o klarownych zastosowaniach, co ułatwia zrozumienie logiki programu i wprowadzanie ewentualnych modyfikacji.
- Obsługa Formularza: Skrypt umożliwia dynamiczne ukrywanie i pokazywanie formularza, co przyczynia się do czytelności interfejsu użytkownika.
- Walidacja Danych: Funkcje walidujące dane sprawdzają parzystość liczby uczestników i informują użytkownika o błędach, co zwiększa integralność danych.
- Generowanie Drabinki: Mechanizm generowania drabinki jest efektywny, dostosowując się do zmiennej liczby uczestników, a także dynamicznie reagując na wybory zwycięzców.
- Integracja z Local Storage: Zapisywanie danych w lokalnym magazynie przeglądarki pozwala na zachowanie informacji nawet po odświeżeniu strony.
- Interakcja z Użytkownikiem: Aplikacja zapewnia intuicyjną interakcję z użytkownikiem, umożliwiając mu płynne przejście przez kolejne etapy tworzenia i prowadzenia turnieju.

Obszary Potencjalnej Poprawy:

- Rozbudowa Funkcjonalności: W zależności od zaawansowania projektu, można rozważyć rozszerzenie funkcjonalności, na przykład o dodatkowe informacje na temat uczestników czy statystyki turnieju.
- Testy Jednostkowe i Integracyjne: Dodanie kompleksowych testów jednostkowych i integracyjnych pozwoliłoby na systematyczną weryfikację poprawności działania kodu oraz zminimalizowanie potencjalnych błędów.
- Optymalizacja Efektywności Algorytmów: W miarę możliwości, warto byłoby zoptymalizować algorytmy generujące drabinkę pod kątem efektywności, zwłaszcza dla dużych turniejów.

- Dostosowanie do Standardów Bezpieczeństwa: W przyszłości można by rozważyć implementację dodatkowych zabezpieczeń, zwłaszcza jeśli aplikacja miałaby obsługiwać bardziej poufne dane.

Podsumowując, aktualne rozwiązanie dostarcza funkcjonalność spełniającą założenia projektowe, ale istnieje przestrzeń do dalszych ulepszeń pod kątem wyglądu, responsywności oraz rozbudowy funkcji. Regularne testowanie oraz feedback od użytkowników mogą przyczynić się do stałego doskonalenia aplikacji.

Dokumentacja kodu:

Kod JavaScript dla BracketCraft skupia się na obszarze logiki związanej z formularzem, walidacją, zapisywaniem danych w localStorage oraz generowaniem i obsługą drabinki turniejowej. Obszerniejszy opis kodu znajduje się w JSDOC.

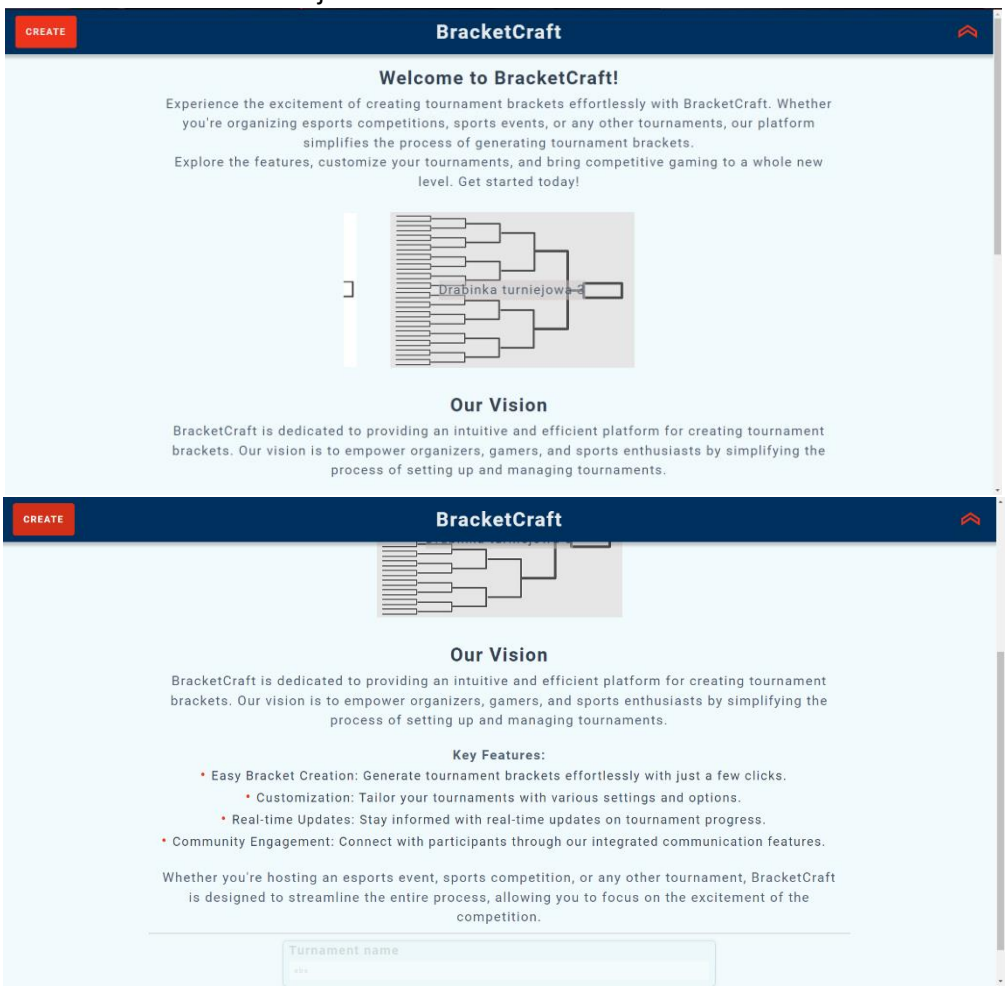
- Zapisywanie i Wczytywanie Danych z Local Storage:
  saveFormDataToLocalStorage: Funkcja zbiera dane z formularza (nazwa turnieju, lista uczestników, data rozpoczęcia) i zapisuje je w localStorage jako obiekt JSON.
  loadFormDataFromLocalStorage: Funkcja wczytuje wcześniej zapisane dane z localStorage i wypełnia nimi odpowiednie pola formularza.
- Obsługa Widoczności Formularza:
  toggleForm: Funkcja odpowiedzialna za przełączanie widoczności formularza. Po wywołaniu sprawdza, czy formularz jest ukryty, a następnie go otwiera, ustawiając jednocześnie odpowiednie animacje i zapisuje tę informację w localStorage.
  hideForm: Funkcja zamykająca formularz. Wywołuje animację ukrycia, a następnie ukrywa formularz i przycisk zamykający.
- Walidacja Formularza:
  validateForm: Funkcja waliduje wprowadzone dane, zwłaszcza liczbę uczestników (musi być potęgą liczby 2), informuje o błędach i w przypadku poprawnej walidacji zapisuje dane w localStorage.
- Generowanie Drabinki Turniejowej:
  generateBracket: Funkcja generuje drabinkę turniejową na podstawie liczby uczestników. Obsługuje zmiany w checkboxach wyboru zwycięzców.
- Generowanie Rund i Zwycięzcy:
  generateRound: Funkcja generuje rundę turnieju, tworząc odpowiednie pary meczowe.
  generateNextRound: Funkcja generuje kolejną rundę lub wyłania zwycięzcę, uwzględniając zaznaczone zwycięzców poprzednich rund.
- Pozostałe Funkcje i Inicjalizacja:
  showWinner: Funkcja prezentuje zwycięzcę turnieju w ostatniej rundzie.
  getRandomOpponent: Funkcja losuje przeciwnika dla danego uczestnika.
  isPowerOfTwo: Funkcja sprawdza, czy liczba jest potęgą liczby 2.
- DOMContentLoaded: Event listener uruchamiający funkcje po załadowaniu strony, w tym wczytywanie danych z localStorage i generowanie drabinki.

Kod ten rozwiązuje problemy związane z obsługą formularza, walidacją danych, generowaniem drabinki turniejowej oraz interakcją użytkownika podczas wyboru zwycięzców. Dzięki modularności i strukturyzacji, kod jest czytelny i łatwy do zrozumienia.

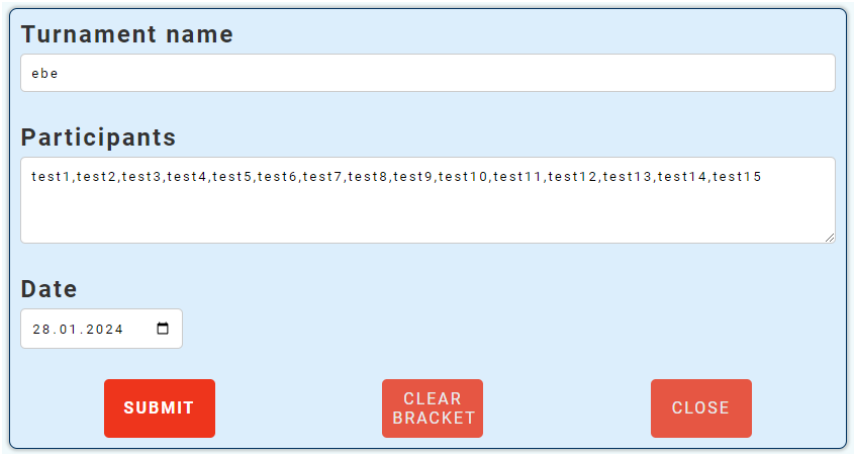
Testowanie

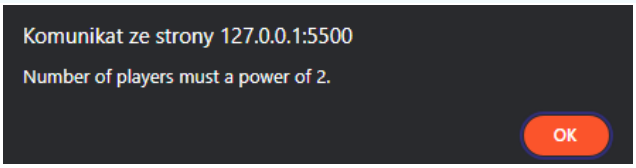
Testowanie aplikacji BracketCraft obejmuje szereg scenariuszy użycia, w tym:

- Sprawdzenie działania i animacji na stronie.

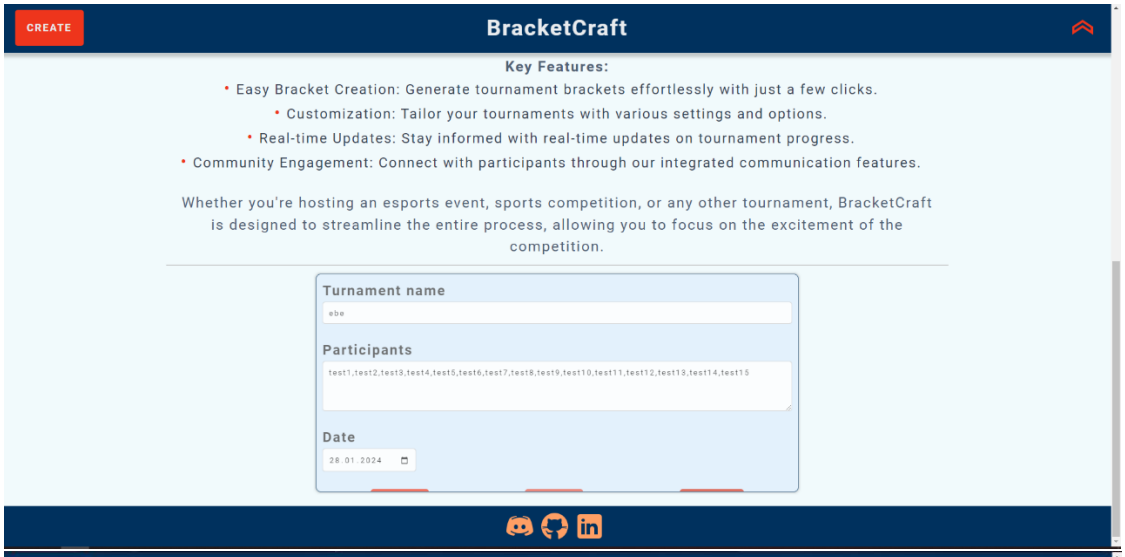


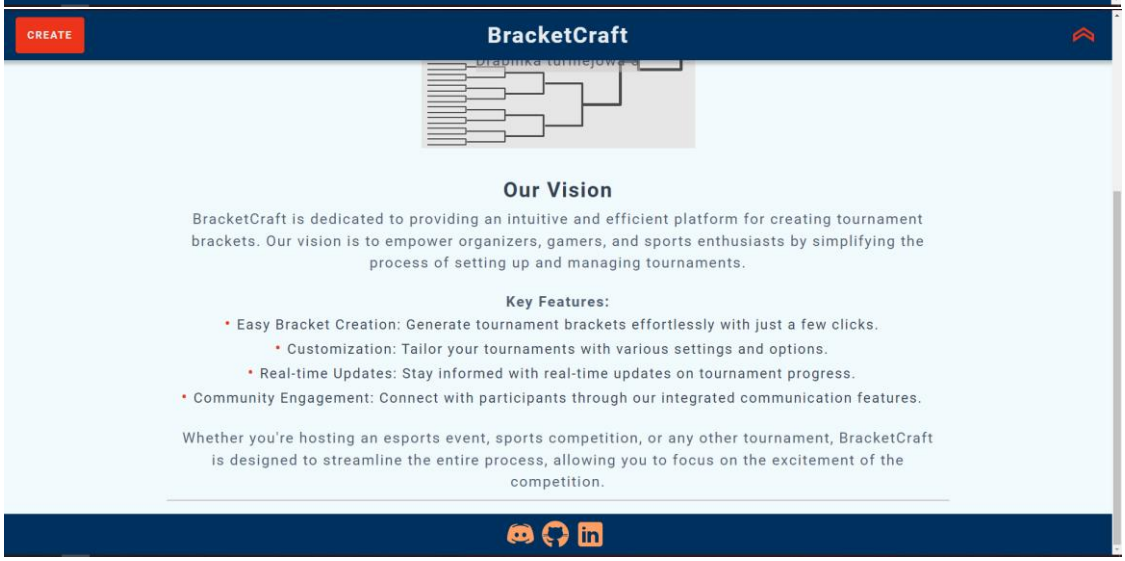
- Sprawdzenie działania walidacji formularza manualnie.



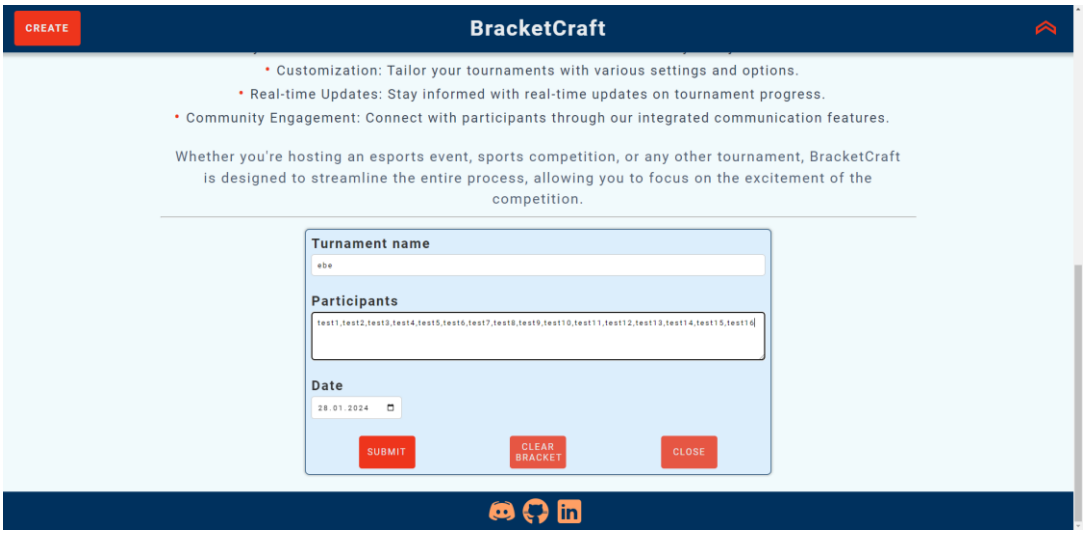


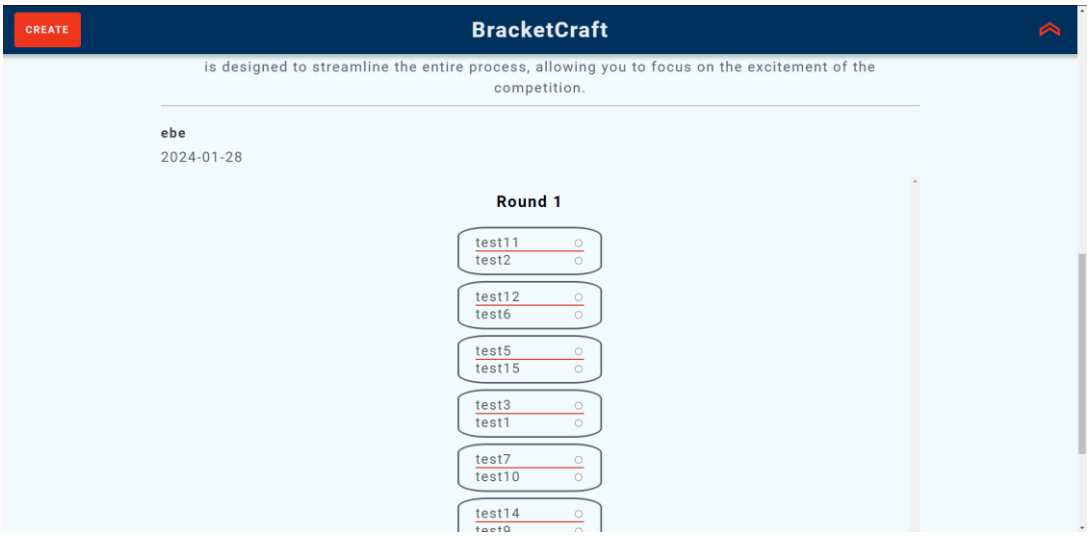
- Sprawdzenie, czy formularz jest prawidłowo ukrywany i pokazywany w odpowiedzi na interakcje użytkownika.



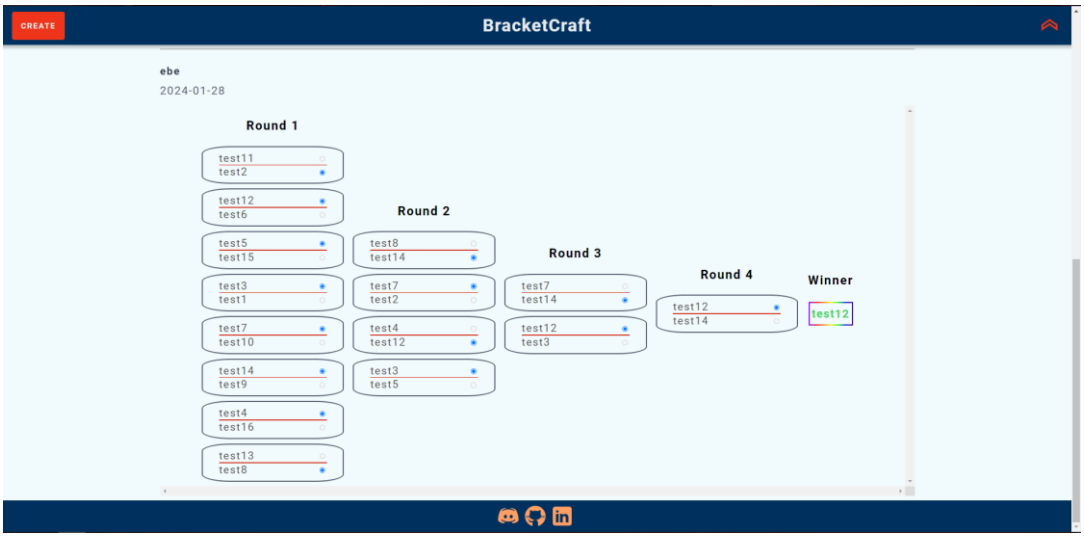


- Sprawdzenie, czy formularz zamyka się poprawnie po zatwierdzeniu danych.

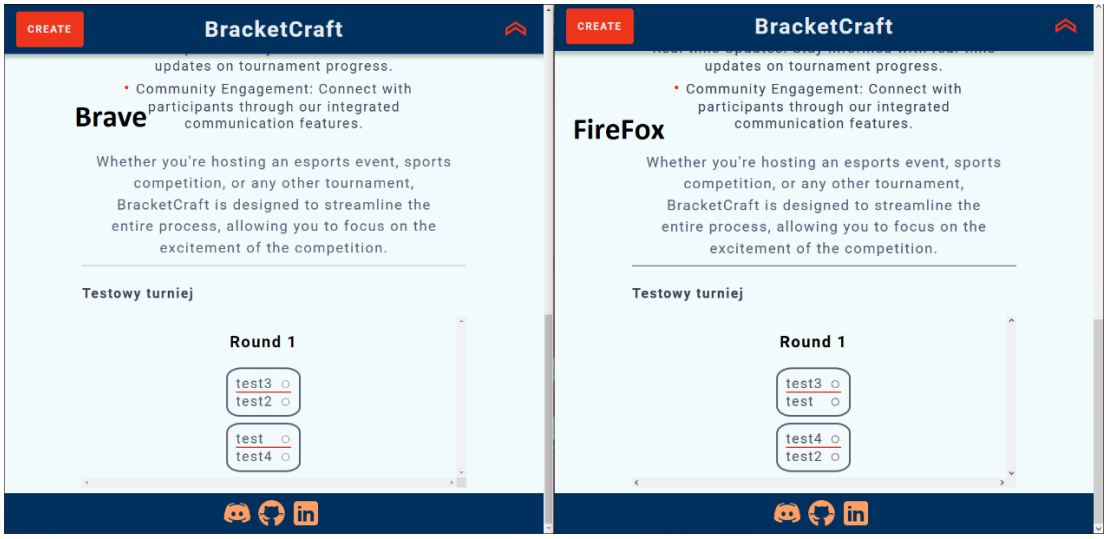




- Sprawdzenie poprawności generowanych drabinek.



- Upewnienie się, że logika JavaScript jest prawidłowo zintegrowana z warstwą prezentacji HTML.
- Sprawdzenie, czy aplikacja działa poprawnie na różnych przeglądarkach internetowych.



- Testy Czasu Ładowania: Pomiar czasu ładowania strony oraz generowania drabinki dla różnych rozmiarów turniejów.