

Ustawienia zadań z wartościami domyślnymi określanymi wspólnie dla konkursu

Mieszko Grodzicki (mieszko.grodzicki@gmail.com)

Wstęp

Podczas dyskusji nad ticketem <https://jira.sio2project.mimuw.edu.pl/browse/SIO-2483> pojawił się pomysł, by ustawienie dotyczące częściowego ujawniania raportów oceniania było własnością instancji zadania, ale taką, której domyślne wartości mogą być dziedziczone po ustawieniach konkursu/rundy. Jedno takie ustawienie - limit zgłoszeń - funkcjonuje już teraz, choć w sposób nie całkiem użyteczny. W ustawieniach konkursu można bowiem sprecyzować domyślny limit zgłoszeń, który zostanie ustawiony w instancji zadania podczas dodawania jej do konkursu, ale gdy potem zachodzi potrzeba zmiany limitu dla wszystkich zadań w konkursie, trzeba dla każdego oddzielnie zrobić to ręcznie. **Uważam, że warto ustalić jednolity, bardziej intuicyjny sposób wprowadzania ustawień instancji zadań, które mogą być dziedziczone po konkursie/rundzie**, z myślą o limicie zgłoszeń, ujawnianiu wyników i ewentualnych nowych ustawieniach dodanych w przeszłości.

Propozycja

1. Utworzenie nowego modelu (robocza nazwa - ProblemInstanceConfig) i przeniesienie do niego ustawień, które dotyczą instancji zadania i które często mają te same wartości dla wszystkich (bądź prawie wszystkich) zadań w konkursie (na chwilę obecną - limit zgłoszeń i ustawienia odsłaniania).
2. Utworzenie w modelu Contest **obowiązkowego** klucza obcego na obiekt ProblemInstanceConfig (relacja 1-1, robocza nazwa problem_default_config), w którym przechowywane będą domyślne wartości ustawień dla zadań w danym konkursie. Z perspektywy użytkownika byłaby to ramka "Domyślna konfiguracja zadania" w ustawieniach konkursu.
3. Utworzenie w modelu ProblemInstance **nieobowiązkowego** klucza obcego na obiekt ProblemInstanceConfig (relacja 1-1, robocza nazwa config), którego brak oznaczać będzie, że zadanie dostaje ustawienia contest.problem_default_config, a obecność - że dla tego zadania ważne są ustawienia w config. Z perspektywy użytkownika byłoby to "Dodaj obiekt Niestandardowa konfiguracja zadania" i możliwość jej edytowania i usunięcia po dodaniu.
4. Do modelu ProblemInstance można byłoby dynamicznie dodać gettery symulujące wcześniejsze pola - i tak na przykład odwołanie do ProblemInstance.submissions_limit zwracałoby config.submissions_limit jeśli config istnieje i contest.default_problem_config.submissions_limit w przeciwnym wypadku.

Zalety

- Nowa funkcjonalność: łatwa zmiana limitu zgłoszeń dla wszystkich zadań w konkursie, łatwe ustawienie wspólnych zasad odsłaniania wyników dla wszystkich zadań w konkursie
- W celu zrealizowania ticketu SIO-2483 (Mitigate possibility of input data tasks leaks) będzie wystarczyło zmienić moduł odsłaniania wyników tak, by umożliwił też odsłanianie raportów i komentarzy do nich - wówczas za darmo dostaniemy też te same ustawienia globalnie dla całego konkursu, z możliwością nadpisania dla konkretnych zadań.
- Mechanizm umożliwiający łatwe dodanie dziedziczonych po konkursie ustawień zadań w przyszłości.

Uwagi

- Przedstawione rozwiązanie umożliwia zmianę ustawień na poziomie całego konkursu lub pojedynczych zadań, nie umożliwia zaś zmiany na poziomie poszczególnych rund. Dodanie trzeciego poziomu wspólnych ustawień - dla każdej rundy, jest możliwe, ale moim zdaniem niewiele wniesie i niepotrzebnie skomplikuje ustawienia konkursu.
- Można zwrócić uwagę, że w powyższym pomysłe nie można nadpisać pojedynczego spośród wspólnych ustawień, tylko albo żadne, albo wszystkie. Jest to z pewnością ograniczenie, choć moim zdaniem nie bardzo znaczące, bo wspólnych ustawień nie będzie wiele. Ograniczenie to jest za to trudne do obejścia. Intuicyjne wydawałoby się podejście - niech w ustawieniach zadania będą pola dla wszystkich wspólnych ustawień, pozostawienie pola pustego oznacza, że wartość jest dziedziczona po konkursie. Niestety to nie sprawdza się np. przy podawaniu limitów - czy pusty limit zgłoszeń oznacza brak limitu, czy przyjęcie wartości domyślnej? Można oczywiście rozróżniać np. 0 jako brak limitu, a puste pole jako dziedziczenie wartości domyślnej, ale to jest już dosyć nieintuicyjne. Rozwiązaniem mogłoby być dodanie checkboxa - dziedziczyć domyślny limit czy nadpisać? Ale dodanie takiego checkboxa do każdego pola z osobna też bardzo skomplikuje formularz. Wydaje mi się więc, że nadpisywanie albo całości, albo niczego jest najlepszym wyjściem.