

Krótko o gamifikacji w nauczaniu programowania



Spotkanie projektowe FGPE.
Od lewej: Tomas Blažauskas, Ricardo Queirós, Rytis Maskeliūnas, Jakub Swacha, Sokol Kosta, Raffaele Montella, Justyna Szydłowska-Samsel, Robertas Damaševičius, Varun Gupta.

Fotografia: Michał Gracel, licencja CC

Gamifikacja polega na zorientowanym na użytkownika zastosowaniu elementów gier, mechaniki gier, projektów gier lub myślenia typowego dla gier w kontekstach nie będących grami w celu zaangażowania, motywowania, zmiany zachowania, rozwiązywania problemów, ułatwiania osiągnięcia celów, uczynienia zadań bardziej interesującymi¹.



Jakub Swacha

Autor jest kierownikiem Katedry Informatyki w Zarządzaniu na Uniwersytecie Szczecińskim. Od ponad 20 lat zajmuje się nauczaniem programowania, a od ponad 10 lat stara się korzystać w tym celu z dobrodziejstw gamifikacji. Jest pomysłodawcą i współtwórcą otwartoźródłowej platformy FGPE, łączącej automatyczną ewaluację rozwiązań zadań programistycznych z w pełni konfigurowalną warstwą gamifikacji. Obecnie kieruje realizowanym wraz z partnerami z Danii, Litwy, Portugalii i Włoch projektem FGPE++, którego głównym celem jest umożliwienie korzystania z platformy w dużej skali nauczania.



¹ Źródło: Andrzej Marczewski (2014). *Defining gamification – what do people really think?*.
<http://www.gamified.uk/2014/04/16/defining-gamification-people-really-think>

Przez marketingowców nazywana grywalizacją, w rzeczywistości obejmuje nie tylko aspekty rywalizacyjne, lecz także współpracę i wsparcie w samodzielnym wyznaczaniu i osiąganiu celów. Choć liczba elementów gier nadających się do wykorzystania poza samymi grami jest długa, kluczowe z nich to wyzwanie (zadanie postawione do wykonania, najczęściej z wyznaczonym terminem, często łączone w łańcuchy nazywane misjami) oraz nagroda – najczęściej w postaci punktów, odznak (łączących symbol graficzny z krótkim opisem) i rankingów porządkujących graczy (bo tak zwykło się nazywać użytkowników systemów zgamifikowanych) według klucza punktów lub innych osiągnięć. Inne elementy gier, po które często sięgają projektanci gamifikacji, to: awatar (wizualizujący gracza), narracja (mogąca nadawać nawet rutynowym czynnościom większe znaczenie), poziomy (ukazujące postępowanie dokonane przez gracza) i treści do odkrycia i/lub odblokowania (w miarę dokonywania postępu).

Siła motywacyjna gamifikacji bywa tłumaczona na wiele sposobów. Najczęściej, bazując na teorii samookreślenia Richarda M. Ryana i Edwarda Deciego, wskazuje się, że pozwala ona zaadresować trzy podstawowe rodzaje wewnętrznych potrzeb psychologicznych:

- kompetencji – kończąc kolejne wyzwania, gracz zyskuje poczucie skuteczności i sukcesu w swoim działaniu;
- autonomii – gracz ma poczucie wolności w przyjęciu wyzwania i sposobie jego wykonania;
- więzi społecznych – nie tylko współpraca, lecz także rywalizacja rodzą w graczach poczucie przynależności do społeczności użytkowników danego systemu zgamifikowanego (niekiedy elity tej społeczności), ta pierwsza – także do pewnej węższej grupy osób (bywa, że sformalizowanej w systemie w postaci zespołu lub gildii).

Gamifikacja w nauczaniu programowania

Tak jak wraz z pojawieniem się gier komputerowych szybko pojawiły się próby ich wykorzystania w nauczaniu (niekiedy nader skuteczne), tak też i od początków gamifikacji jednym z podstawowych pól jej wdrażania stała się edukacja. Z pewnością pomaga w tym łatwość zastosowania gamifikacji, zgodnie z teorią uczenia się przez doświadczenie amerykańskiego teoretyka nauczania Davida A. Kolba, w fazach konkretnego doświadczenia (przeżycia związanego z wykonywaniem wyzwania) i aktywnego eksperymentowania (sprawdzania nowo nabytych kompeten-

cji w praktyce), jak też łatwość (bazująca na teorii obciążenia poznawczego Johna Swellera) wyrabiania schematów u uczących się poprzez ich motywowanie do wielokrotnego wykonywania tych samych czynności. Gamifikacja jest także spójna z konstruktywistyczną teorią uczenia się poprzez m.in. aktywny udział uczniów w zdobywaniu wiedzy, uczenie się w zespole, spiralne podejście do nauczania (zamiast progresji liniowej), uczenie się poprzez eksplorację, rosnący poziom trudności zajęć czy też świadomość celów nauki.

Istnieje kilka powodów, dla których zastosowanie gamifikacji jest szczególnie przydatne w przypadku nauczania programowania:

- trudność nauki programowania wymagająca wielu ćwiczeń praktycznych (do wykonywania których gamifikacja może dostarczyć odpowiedniej zachęty);
- postrzeganie nauki programowania w jej tradycyjnym wydaniu jako żmudnej i nużącej (tu gamifikacja może wprowadzić elementy zabawy i zaciekawienia);
- niemal nieodłączne wykorzystanie komputerów w nauczaniu programowania (wystarczy zatem dodać warstwę oprogramowania implementującą gamifikację, co nie byłoby takie proste w przypadku nauki bez użycia komputerów);
- powszechne wykorzystanie w nauczaniu programowania automatycznego oceniania zgłaszanych przez uczniów rozwiązań (co stanowi dogodny punkt do implementacji gamifikacji i wzbogacania przez nią informacji zwrotnej dla uczniów).

Jako prostą formę gamifikacji nauki programowania można wskazać „sędziów online”, takich jak Polski SPOJ². Jednak implementują one zwykle bardzo wąski zestaw elementów (wyzwania, punkty, rankingi), nie wykorzystując pełnego potencjału gamifikacji na tym polu. Znacznie szerzej (pod względem aspektów społecznościowych i/lub narracyjnych) implementują gamifikację platformy do nauki programowania online, takie jak Code Avengers³, CodeEasy⁴ czy CodeWars⁵. Są to platformy zamknięte, które

² Sphere Online Judge. Polski SPOJ. <https://pl.spoj.com/>

³ Code Avengers. <https://www.codeavengers.com/>

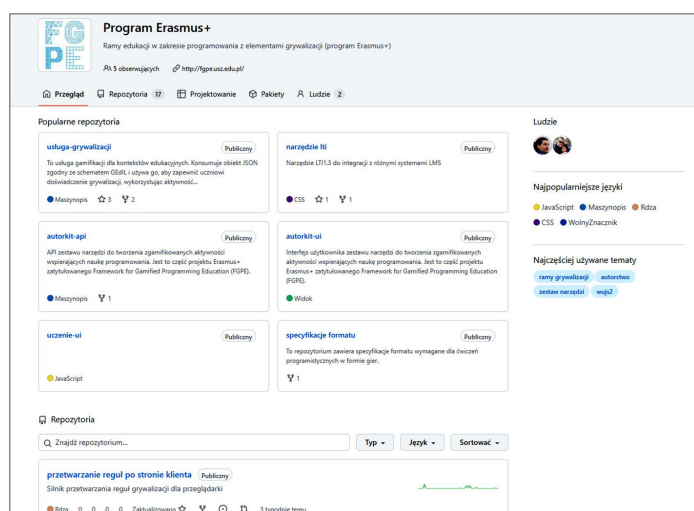
⁴ CodeEasy. <https://codeeasy.io/>

⁵ CodeWars. <https://www.codewars.com/>

nauczyciel programowania może wykorzystać na prowadzonych przez siebie zajęciach lub nie, nie może jednak w łatwy sposób dostosować zakresu treściowego i schematu reguł gamifikacji do swoich specyficznych potrzeb. Wolną od takich ograniczeń jest platforma FGPE, pozwalająca nauczycielowi na samodzielny dobór i uporządkowanie zarówno zadań, jak i reguł gamifikacji.

Platforma FGPE

Platforma FGPE (*Framework for Gamified Programming Education*) rozwijana jest od 2019 r. przez pięć europejskich instytucji edukacyjnych (Uniwersytet Szczeciński, INESC TEC w Porto, Uniwersytet Parthenope w Neapolu, Uniwersytet Aalborg w Kopenhadze oraz Politechnikę w Kownie) z dofinansowaniem ze środków Unii Europejskiej w ramach programu Erasmus+. Cała platforma (oprogramowanie wraz z zestawem ponad 1000 ćwiczeń programistycznych) dostępna jest bezpłatnie na licencji otwartej⁶.

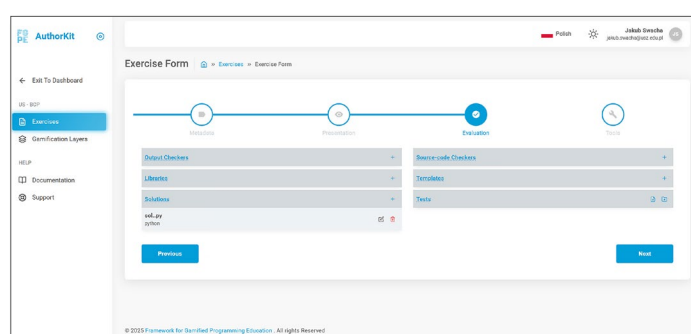


Źródło: <https://github.com/FGPE-Erasmus/>

Platforma FGPE obejmuje: ustandaryzowany zestaw siedmiu typów zadań programistycznych i kilkunastu elementów gamifikacji do wykorzystania, formaty danych (ćwiczeń i schematów gamifikacji), interaktywną platformę do nauki programowania online dla uczniów i studentów wraz z panelem nauczyciela (FGPE PLE), a także edytor zgamifikowanych zadań programistycznych (AuthorKit).

FGPE PLE jako aplikacja PWA może być instalowana na urządzeniach mobilnych i poprzez interfejs LTI – wymieniać się danymi z oprogramowaniem klasy LMS, pozwalając np. na osadzanie zgamifikowanych zadań programistycznych w kursach udostępnianych w popularnym systemie Moodle.

Dotychczasowa wersja platformy umożliwia zarówno testowe wykonywanie kodu nadsyłanych rozwiązań (poprzez zapewniający *sandbox* i mechanizmy ewaluacji program Mooshak), jak i przetwarzanie reguł gamifikacji (poprzez opracowany w tym celu web service oparty na GraphQL) po stronie serwera.



Zrzut ekranu z edytora zadań FGPE

Obecnie, w ramach projektu FGPE++⁷, trwają prace nad ukończeniem i udostępnieniem lekkiej wersji platformy, w której testowanie i ewaluacja kodu uczniów (obecnie możliwa już dla języków JavaScript, Python i C++) oraz przetwarzanie reguł gamifikacji wykonywane będzie po stronie klienta, w środowisku przeglądarki internetowej. Choć jest to rozwiązanie mniej odporne na manipulację (co w przypadku platformy ukierunkowanej na samodzielną naukę, a nie prowadzenie konkursów programistycznych nie stanowi wielkiego problemu), znacząco zmniejsza wymagania dotyczące serwera w przypadku udostępnienia platformy dużej liczbie użytkowników, a także pozwala uczniom na kontynuowanie nauki nawet przy dłuższym braku połączenia z serwerem. W ramach projektu FGPE++ edytor AuthorKit wzbogacono również o interfejs OpenAI API, co ułatwia nauczycielom szybkie generowanie dużej liczby zadań programistycznych na wybrany temat. Rozważanym na przyszłość kierunkiem rozwoju platformy jest dodanie funkcji wykrywania rozwiązań wygenerowanych przez AI.

⁶ Adres głównego repozytorium: <https://github.com/FGPE-Erasmus/>

⁷ Projekt FGPE++ Gamified Programming Learning at Scale (2023-1-PL01-KA220-HED-000164696) jest dofinansowany ze środków Unii Europejskiej. Więcej informacji pod adresem: <https://fgpeplus2.usz.edu.pl/>