

GOLD RUSH



Wprowadzenie

W skrócie

Jako że jesteśmy świetnymi, rozchwytywanymi deweloperami, dostaliśmy ciekawe zlecenie:

napisanie gry *Gold Rush*,

która – jak nazwa sugeruje – jest osadzona w świecie amerykańskiej gorączki złota z XIX wieku: twardzi ludzie, pustynia, kilofy, bogactwo, szaleństwo, rewolwery – te klimaty...

Z założenia ma to być sieciowa gra typu multiplayer.

Cel jest prosty: zdobyć jak najwięcej złota i pozbyć się konkurencji.

Gra rozgrywa się na kwadratowej planszy. Obowiązuje system turowy. Na polach rozmieszczane jest złoto oraz inne żetony odpowiadające pożywieniu, narzędziom, bonusom itp.

Na realizację mamy 15 tygodni. To dużo i mało... Trzeba pamiętać, że nie wiemy jeszcze wszystkiego. Tak naprawdę poznaliśmy dopiero ogólny zarys zlecenia.

Cel zajęć

Głównym celem zajęć jest **zdobycie umiejętności planowania i prowadzenia projektu** programistycznego, w szczególności:

- poznanie **technik i narzędzi** prowadzenia procesu,
- zmierzenie się z **projektowaniem** elementów oprogramowania,
- **popętnienie błędów** i nauka ich unikania,
- nauczanie się **testowania** tworzonego oprogramowania,
- przyswojenie **dobrych nawyków** programistycznych,
- **rozwój warsztatu** programistycznego.

Organizacja pracy

Pracujemy **indywidualnie**: każdy nad swoim projektem.

Mimo to staramy się pracować **synchronicznie**: w danym momencie (na danych zajęciach) wszyscy zajmują się tymi samymi zadaniami. Wspólnie omawiamy problemy, szukamy rozwiązań. **Bardzo liczę na Waszą aktywność.**

Praca jest zorganizowana **iteracyjnie**. Długość iteracji ustalamy na **dwa tygodnie**.

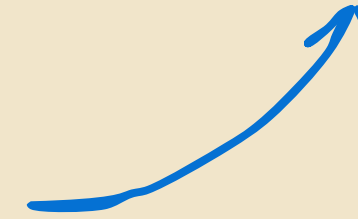
Efekty pracy są oceniane na bieżąco za pomocą **testów**.

Każdy przechowuje kod swojego projektu w **repozytorium** (np. GitHub, GitLab).

Repozytoria

Repozytorium z zasobami: https://gitlab.com/edu.io/gold_rush_resources

Tu znajdziecie materiały z zajęć, fragmenty kodu, potrzebne zasoby itp.



Repozytorium z testami: https://gitlab.com/edu.io/gold_rush_tests


Tu z kolei umieszczam testy, które Wasz kod musi przejść pozytywnie.




Sposób i kryteria oceny

Przede wszystkim...

...z tygodnia na tydzień rozwijamy grę, pisząc kod. Kod musi spełniać przyjęte założenia i działać prawidłowo. Podstawowym sposobem sprawdzenia, czy ten warunek jest spełniony, jest **przeprowadzanie testów**. Zestaw testów z każdego tygodnia jest dostępny w repozytorium.

Jeżeli Wasz kod przechodzi testy, jest dobrze. 
Odnótowuję (+).

Jeżeli do terminu bieżących zajęć testy z poprzednich nie przechodzą, jest źle.
Odnótowuję (-) i oczekuję jak najszybszego poprawienia. 
Jeżeli nie poprawicie kodu do kolejnych zajęć, wystawiam częściowe *ndst*.

Sposób i kryteria oceny

Poza tym...

- Sprawdzam **aktywność w repozytorium**: oczekuję kilku zatwierdzeń zmian (*commit*) w ciągu tygodnia; jest aktywność: (+), nie ma aktywności: (−).
- Na niektórych zajęciach trzeba będzie opracować dodatkowe materiały, np. dokumentację, diagramy itp. To również podlega ocenie.
- Na dalszych etapach projektu przewiduję tzw. **code review** dokonywany przez Was; tu też można zdobyć (+) za wykazanie się i (−) za brak aktywności.
- Finałem zajęć jest **prezentacja** własnej gry; ocenie będzie podlegać zgodność realizacji z ustalonymi wymaganiami.

Pytania

?