Sprawozdanie z projektu

na Informatyczne Narzędzia Pracy Grupowej

**SAPER**



Joanna Stanisz

Patrycja Tokarczyk

Kamil Barczyński

Piotr Gołyźniak

***Cel projektu:***

* Nauka metodyki Scrum w pracy grupowej.
* Nauka informatycznych narzędzi pracy grupowej takich jak: Github,Tortoise Git.
* Wdrożenie się w z zasady i organizację pracy grupowej.
* Implementacja konsolowej wersji gry *Saper* w języku *C++*.

***Saper, czyli?***

* gra komputerowa
* oryginalna nazwa to *Minesweeper*, została napisana w 1981 roku
* polega na odkrywaniu pól planszy w taki sposób, aby nie trafić na minę
* każde odkryte pole zawiera informację o liczbie min, z którymi sąsiaduje (0-8)
* nieodkryte pola można oznaczać flagą (*domniemana bomba*), przez co blokowane jest jego odsłonięcie i przypadkowe odkrycie miny

***Opis poszczególnych sprintów:***

1.PLANSZA

Pierwszy sprint został przez nas zaplanowany tak, aby w ramach pięciu dni roboczych napisać kod zawierający prototyp gry. Wyznaczyliśmy pierwszy milestone: „Plansza”, który przypominał nam, że celem ówczesnego sprintu jest stworzenie podstaw, do dalszego rozwoju gry. Wybraliśmy także pierwszego Scrum Mastera: Piotra. Ustaliliśmy, że będziemy korzystać z serwisu github.com w celu stworzenia repozytorium, które umożliwiło nam ogólny dostęp do kodu oraz możliwość jego modyfikacji.

        Pierwszy dzień poświęciliśmy na sprawy organizacyjne. Pobraliśmy na nasze komputery system kontroli wersji: Git, oraz oprogramowanie klienckie TortoiseGit umożliwiające sprawne korzystanie z tego systemu. Tego dnia przeprowadziliśmy także pierwsze commity, które pozwoliły rozstrzygnąć, czy nasze programy działają prawidłowo. Wyznaczyliśmy także, zadania na najbliższe dni: dokładne zapoznanie się z zasadami działania gry saper, oraz napisanie kodu, pozwalającego stworzenie tablicy dwuwymiarowej.

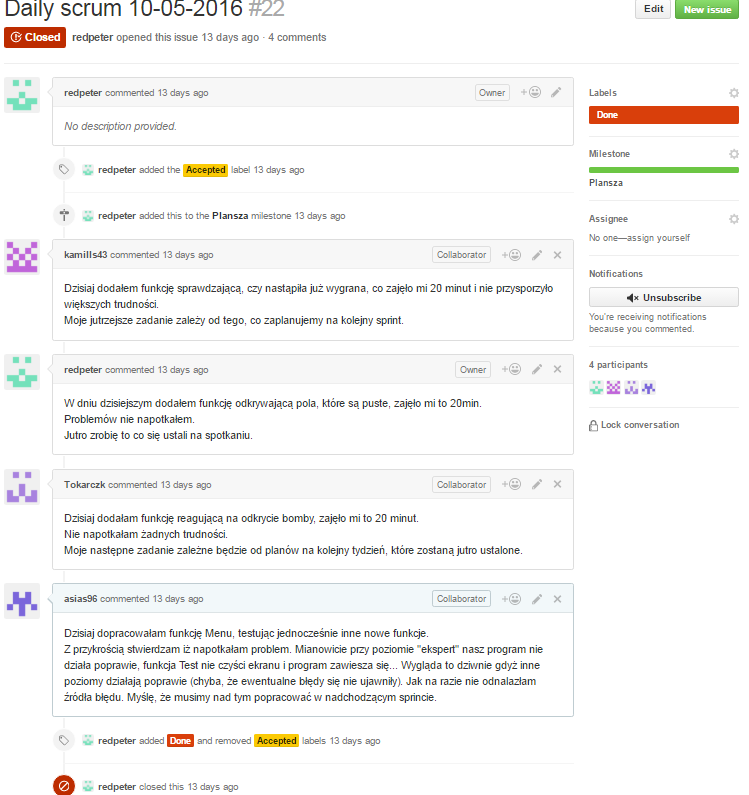
Następnego dnia zabraliśmy się do pisania kodu. Każdy pracował nad przydzieloną mu do napisania funkcją, dlatego już po pierwszym dniu pracy mieliśmy 4 najważniejsze funkcje naszego programu: tworzącą, usuwającą i wypisującą tablice dwuwymiarową oraz strukturę do planszy. Każdy bez problemów wypełnił swoje zadanie a przeprowadzone testy pokazały, że napisane funkcje działają prawidłowo.

Ustaliśmy, że po napisaniu i wprowadzeniu do repozytorium zmian w kodzie będziemy zmienić labels w odpowiednim issue na Test, dzięki czemu będzie wiadomo, że funkcja jest skończona i można zająć się jej testowaniem. Testowanie będzie polegać na sprawdzaniu siebie nawzajem.

Trzeciego dnia rozwijaliśmy stworzone wcześniej podstawy. Dodana została funkcja zliczająca bomby na planszy jednocześnie wyświetlająca ową ilość bomb, funkcja rozmieszczająca losowo bomby, funkcja testowa wyświetlająca tablicę z bombami oraz z ilością bomb pozostałych na planszy. Dokonane zostały też inne niezbędne ulepszenia jak np. dodanie dyrektywy preprocesora dla bomby, zmiany wyświetlania tablicy w funkcji wypisującej czy ujednolicenie kodu.

Czwartego dnia dodaliśmy do naszego programu funkcję odkrywającą pola wybrane przez użytkownika oraz szkielet funkcji obsługującej menu. Zostały dokonane również ulepszenia istniejących już funkcji: Write, Test i Random np. zwiększanie wartości pól wokół bomby. Każdy poradził sobie ze swoim zadaniem nie napotykając po drodze problemów.

Ostatniego dnia pierwszego sprintu pracowaliśmy nad kolejnymi ulepszeniami zamykającymi milestone “Plansza”. Były to następujące funkcje: sprawdzająca czy nastąpiła już wygrana, reagująca na odkrycie bomby, (czyli sygnalizująca przegraną), dokończona funkcja obsługująca menu oraz wykonane rekurencyjnie odsłanianie pól zerowych. Testowanie świeżo dodanych funkcji nie wykazywało błędów, jednak…



Wykryty został problem, którego źródła nie udało nam się ustalić przed zakończeniem ówczesnego sprintu. Mimo to zamykając milestone “Plansza” mogliśmy pochwalić się sporym nabytkiem napisanego kodu, działającym, lecz z malutkim wyjątkiem. Znalezienie powstałego błędu zostało przełożone na kolejny sprint nazwany: “User”.

2.USER

3.INTERFEJS

4. Plany na kolejne sprinty

Aktualna wersja gry, chociaż jest już funkcjonalna, może wciąż być rozwijana. Gdyby nasz projekt trwał dłużej, wprowadzalibyśmy do niej kolejne ulepszenia, pamiętając, by na koniec każdego z kolejnych sprintów dostarczyć gotowy do potencjalnego wydania Produkt.

W kolejnych sprintach skupilibyśmy się przede wszystkim na dołączeniu do gry biblioteki graficznej (przejście z konsoli do programu okienkowego), co uczyniłoby środowisko gry bardziej przyjazne użytkownikowi, a także na dodaniu możliwości odkrywania lub zaznaczania pól przy pomocy myszki. Ponadto chcielibyśmy dołączyć pomiar czasu gry oraz możliwość eksportowania swoich wyników (oraz pseudonimu gracza) do pliku, zawierającego najlepsze czasy gry na poszczególnych poziomach.

Dodatkowo chcielibyśmy, aby pierwsze odkryte pole nie mogło być bombą (aby w pierwszym kroku nie dało się przegrać). Ciekawymi pomysłami wydają nam się także: wykorzystanie efektów dźwiękowych, np. w razie przegranej i możliwość *resetowania* planszy w dowolnym momencie gry.

Mamy jednak na uwadze, że przed przystąpieniem do kolejnego sprintu musielibyśmy go zaplanować - wybrać jego główny cel i oszacować, ile jesteśmy w stanie zrobić w przyjętym czasie (biorąc pod uwagę dotychczasowe tempo prac), dlatego obecnie nie jesteśmy w stanie wskazać, które z powyższych udogodnień uwzględnilibyśmy w najbliższym sprincie.

Zakończenie:

Jakieś podsumowanie pracy, gita itd?