

Zestaw 6

Testowanie Gier

Karol Stuła, Wojciech Adamiec

1 czerwca 2021

Spis treści

1	Planowanie i nadzór nad testami	2
1.1	Cele testowania	2
1.2	Ograniczenia testów	2
1.3	Kryteria akceptacji testów	2
2	Analiza testów	2
2.1	Wymagania funkcjonalne	2
2.2	Wymagania нефункционалне	2
2.3	Historyjki użytkownika	2
3	Projektowanie testów	3
4	Implementacja testów	4
5	Ocena kryteriów zakończenia i raportowanie	4
6	Czynności zamykające test	4

1. Planowanie i nadzór nad testami

1.1. Cele testowania

- Wykrywanie błędów
- Zapobieganie błędom technicznym
- Zapewnienie płynnej gry
- Sporządzanie raportów z procesu produkcji
- Zapewnić balans trudności gry dla użytkowników

1.2. Ograniczenia testów

- Ilość zatrudnionych testerów
- Ograniczenia sprzętowe
- Dostępny czas

1.3. Kryteria akceptacji testów

- Każda historyjka użytkownika powinna mieć odpowiednik w postaci testu
- Wszystkie testy powinny zostać zaakceptowane
- Stworzenie wiele różnorodnych testów
- Odpowiednia dokumentacja testów

2. Analiza testów

2.1. Wymagania funkcjonalne

- Możliwość ulepszenia poszczególnych systemów, podsystemów lub reaktora za dostępne punkty złomu.
- Możliwość otwarcia śluzy pomiędzy pomieszczeniami.
- Możliwość zmiany poziomu mocy dla poszczególnych systemów.

2.2. Wymagania niefunkcjonalne

- Czytelne wyświetlanie informacji o aktualnym stanie mocy na statku.
- Odpowiednia animacja i dźwięk towarzyszący otwarciu śluzy.
- Możliwość podglądu stanu statku sprzed potencjalnego zakupu i po potencjalnym zakupie.

2.3. Historyjki użytkownika

- Gracz chce rozdysponować mocą reaktora na systemy
- Gracz chce otworzyć śluzy na statku
- Gracz chce ulepszyć statek

3. Projektowanie testów

Zobaczmy przypadki testowe dla wymagań niefunkcjonalnych

1. Pierwszy przypadek

Warunki początkowe: Gracz znajduje się w oknie dystrybucji mocy reaktora i ma do dyspozycji kilka kreszek reaktora.

Przebieg w krokach:

- Po kliknięciu w system moc reaktora spada o jedną kreskę, a moc systemu zwiększa się o jedną kreskę
- Gdy moc systemu jest naładowana w pełni, zwiększa się poziom mocy systemu i pokazują się od nowa puste kreski systemu.

Rezultat: Gracz może czytelnie odczytać informacje o stanie mocy na statku.

2. Drugi przypadek

Warunki początkowe: Gracz znajduje się w oknie gry i ma podgląd na wszystkie pomieszczenia statku.

Przebieg w krokach:

- Gracz otwiera śluzę między pomieszczeniami
- Poziom tlenu między pomieszczeniami wyrównuje się

Warunki końcowe: Gracz zobaczył i usłyszał odpowiednią animację towarzyszącą otwarcia śluzy.

3. Trzeci przypadek

Warunki początkowe: Gracz znajduje się w oknie ulepszenia statku i ma do dyspozycji daną ilość monet złomu.

Przebieg w krokach:

- Po kliknięciu w system ilość pieniędzy na koncie spada o określoną przy systemie cenę, a poziom systemu zwiększa się o jedną kreskę
- Gdy poziom systemu jest uzupełniony w pełni, zwiększa się poziom systemu i pokazują się od nowa puste kreski systemu.

Warunki końcowe: Gracz zobaczył podgląd aktualnego stanu statku przed i po zakupie.

4. Implementacja testów

Testy powinny być zaimplementowane w języku Python w wersji nie starszej niż 3.6.

Każdy test powinien dokładnie i rzetelnie sprawdzać wszystkie funkcje i metody, które związane są bezpośrednio lub pośrednio z testowanym obiektem lub funkcjonalnością.

Każdy test powinien zawierać treściwy opis tekstowy w postaci komentarzy i pythonowych *docstringów*. W przypadku, gdy test wykryje jakieś nieprawidłowości powinien udostępnić konkretne informacje o błędzie jak i również o właściwościach sprzętowych maszyny, na której został przeprowadzony test.

Testy zostaną napisane ręcznie przez dwóch, niezależnie pracujących testerów.

5. Ocena kryteriów zakończenia i raportowanie

Powinna zostać wykonana ponowna weryfikacja, czy każdy ze zgłoszonych błędów został naprawiony i czy w trakcie jego naprawy nie pojawiły się nowe błędy.

Jeżeli w wyniku działań poza testowych został wykryty błąd nieobjęty testem, to należy natychmiast objąć dany fragment oprogramowania dodatkowymi testami i jeszcze raz upewnić się, że wszystko jest w porządku.

Na koniec testowania powinien zostać sporządzony zbiorczy raport opisujący przebieg testów - w tym statystyki liczby błędów i czasu potrzebnego na ich naprawę. Dla każdej kategorii błędu powinien zostać dodatkowo sporządzony raport szczegółowy opisujący błędy o specyficznej przypadłości.

6. Czynności zamykające test

Wynik każdego testu poddany powinien zostać dokładnej analizie, na podstawie której należy spróbować zaobserwować co jest głównym powodem błędów, i jeśli to możliwe, należy się zastanowić jak ich w przyszłości unikać.

Ostatecznym wynikiem prac jest raport do wiadomości najważniejszych inżynierów i dalej członków zespołu. Powinien on powstać po weryfikacji, że wszystkie kluczowe błędy zostały naprawione i wystawiona została ocena działań zespołu. Rezultaty powinny zostać zarchiwizowane na potrzeby dalszych projektów i rozwoju firmy.