- Marko Golovko, Katsiaryna Yalovik

- Środa 18-20

- Zestaw nr 4

Proces testowy napisany do odpowiedniego typu gry, miejscami uzupełniony ewentualnymi przykładami.

Proces testowy A

Planowanie

- Zakres testów i cel testów

Sprawdzenia działanie gry, wykrywanie bagów, zawieszeń, blokerów, zawieszeń, testowanie funkcjonalności.

- Element testowy

Gra np. „Najlepszy wędkarz w Amazonii” , testowana będzie cala gra w dostępnej wersji.

- Funkcjonalności, które zostaną przetestowane

Możliwość prochodzenia gry. Poprawne działania przedstawionych możliwości.

- Poziomy i typy testów

Testy akceptacyjne, alfa testy, testowanie funkjconalne, testowanie struktruralne, testowanie związane ze zmianami

- Metody testowania

Manualne testowanie.

- Kryteria akceptacji

Gra działa zgodnie z oczekiwaniami, nie wysypuje i nie zawiesza, Oraz poszczególne kryteria akceptacji z historyjek użytkownika.

- Rzeczy które są dostarczane razem z testami

Przypadki testowe, raporty z testów.

- Środowisko testowe

Gra w ostatniej wersji testowej.

- Harmonogram

Każdy sprint planujemy na dwa tygodnia.

Sprint 0, podczas tego sprintu, przeprowadzane są taski inicjujące obejmujące na przykład przygotowanie środowisk testowych, projektowanie testów i analiza procesu testowego . W tej iteracji uzgodniamy założenia biznesowe, zakres projektu, określiamy kluczowe wymagania, zidentyfikujemy ryzyka, oszacujemy wstępne koszty. Takoż robimy planowania procesu testowego.

Sprinty – Iteracje, to na tym etapie odbywają się procesy omówione w rozdziałach implementacji, wykonanie, ocena i raportowanie. Takoż naprawiamy z każdą iteracją planowanie. I dodajemy taski do projektowania.. Ta faza jest zbiorem iteracji w których powstaje gra bez błędów.

Ostatnie dwa tygodnia to faza, która obejmuje testy akceptacyjne. I rozdział zamykania czynności testowych.

Skład teamu –

Producenci gry: Marko Golovko, Katsiaryna Yalovik

Wiodący tester: Stanisław Abacki

Testery: Mateusz Cabacki, Kuba Kabacki, Maria Rabacka

Techniczny tester: Arkadiusz Tabacki

- Ryzyka i zagrożenia

Niemożność dostarczenia produktu w czas (Możliwe przeniesienie czasu zakończenia procesu testowego)

Brak umiejętności lub ludzi. (Zmiana planowanych sprintów zgodnie z analizą umiejętności teamu)

Problem ze zdefiniowaniem właściwych wymagań. (Dodatkowe konsultacje z klientem lub dyskusją z teamem programistów)

Identyfikacja zagrożeń dla realizacji testów podczas etapów.

Monitorowanie zagrożeń dla realizacji prac całego procesu testowego.

Podczas wykonania testów – identyfikowane są głównie ryzyka związane z utrudnieniami bądź d niemożnością realizacji testów poszczególnych obszarów oraz zagrożenia niedotrzymania terminu e e eee eee zakończenia testów

- Założenia

Pomiar i analiza rezultatów. Monitorowanie i dokumentowanie postępu. Inicjowania właściwych reakcji w odpowiedzi na następujące zmiany. Podejmowanie decyzji.

- Osoby zatwierdzające plan

Katsiaryna Yalovik, Stanisław Abacki

Analiza

1. Określamy jako przedmiot testowy naszą grę (Najlepszy wędkarz w Amazonii).
2. Określamy cel testowania
3. Robimy spis funkcjonalności i możliwości, które posiada gra i te, które będą testowane.
4. Z otrzymanej informacji definiujemy warunki testowe.
5. Na podstawie kroku 3 i 4 określamy elementy testowe
6. Dokonujemy analizy ryzyka.

Projektowanie testów

1. Wyznaczamy poziomy testów dla konkretnych testowanych obszarów.
2. Wybieramy techniki projektowania ( w naszym przypadku czarnoszkrzynkowe techniki)
3. Tworzymy historyjki użytkownika
4. Tworzymy przypadki testowe
5. Priorytetyzacja przypadków testowych

Podczas pierwszego sprintu przypadki testowe bardziej ogólne. Z kolejnymi iteracjami testerzy poznają bardziej szczegółowo grę i lepiej orientują się w procesie.

Implementacja

1. Przygotowujemy środowisko testowe (zainstalujemy i przygotowujemy grę).
2. Testujemy większość przypadków manualnie, więc dla każdego przypadku testowego, przygotowujemy warunki początkowe.
3. Dla testów automatycznych Arkadiusz Tabacki, przekłada przypadki testowe na kod testu wykonującego ten przypadek testowy.

Wykonanie testów

1. Wykonujemy testy na naszym elemencie testowym.
2. Po wykonaniu przypadku testowego otrzymujemy wynik.
3. Wyniky zapisujemy w postaci raportu.
4. Po zauważaniu błędu w przypadku testowym ten test przechodzi proces od fazy projektowania.
5. Zgodnie z uwzględnieniem pewnych odstępstw, plan może ulecz zmianie.

Kontrola

Proces, który ściśle powiązany z implementacją i wykonaniem.

Tę fazę dzielimy na dwa procesy: Monitoring i nadzór.

Monitorowanie

Stanisław Abacki — osoba odpowiedzialna. Zadanie polega na monitorowaniu, czyli sprawdzeniu, czy wszystkie taski idą zgodnie z planem, czy nie ma opóźnień i czy wszystkie taski zostaną dostarczone na czas.

Stanisław Abacki używa takich metryk:

1. Stosunek ilości zadań do zadań zaplanowanych w danym spincie
2. Stopień pokrycia funkcjonalności testami
3. Liczba pozytywnie zakończonych przypadków testowych
4. Liczba zgłoczonych błedów
5. Liczba naprawionych błędów

Raportowanie

Do monitorowania projektu zaliczamy też raportowanie testów.

Po, w trakcie, jak i po zakończonych testach tworzymy raport testów.

Wzorzec raportu umieściłem w oddzielnym pliku 'RaportA.docx'.

Nadzór

Zgodnie z wynikami monitoringu w razie sytuacji takich, że proces nie idzie zgodnie z planem lub w ostatniej iteracji zgłoszono znacznie więcej błędów niż zwykle. Wtedy przeprowadzamy analizę źródła problemu, a w następnej kolejności zastanawiamy się nad kolejnymi krokami.

W zależności od raportowania podejmujemy różne decyzji i działania na przykład:

- czy konieczne jest przedłużenie fazy testowej

- czy konieczna będzie redukcja ilości zadań zaplanowanych na następny sprint po to by programiści zdążyli naprawić błędy

Ocena i raportowanie

Podczas tej fazy zadanie polega na określeniu czy uzyskane wyniki są zgodne c celami testowania.

Ocenę wykonujemy dla każdego poziomu testów.

Ocenę wyników tworzymy na podstawie monitorowania postępów w testach oraz spełnieniu określonych kryteriów.

Zgodnie z metrykami opisanymi w fazie monitorowania otrzymujemy takie kryteria:

- Stopień pokrycia funkcjonalności testami musi być większy niż 90%

- Liczba naprawionych błędów do stosunki liczby błędów >0.95

- wszystkie błędy znalezione w danej fazie, oznaczone jako blocker i critical muszą zostać naprawione

Na podstawie otrzymanych ocen tworzymy raport na potrzeby klienta.

Czynności zamykające testowanie

Podsumowujemy to, co się działo w ramach procesu.

Katsiatyna Yalovik ma do wykonania następujące zadania:

- sprawdzić, czy wszystko zostało dostarczone zgodnie z planem i dokumentacją

- zarządzić błędami, czyli zamykanie notek, tworzenie nowych, edycja istniejących, jeśli jest taka potrzeba

- utrzymywanie dokumentacji

- prace niezbędne do utrzymania środowiska testowego i ponownego użycia w przyszłości

- zorganizować spotkanie retrospektywne

- zachować artefakty procesu testowego