

AskAnything - PLAN TESTÓW

by the Hackin9 crew

Jan Urbański (ju219721) Sebastian Tomaszewski (st219712)
Jakub Rauch (jr219554) Maciej Stefański (ms219668)

5 czerwca 2006

<http://students.mimuw.edu.pl/~ju219721/AskAnything/>
(strona projektu)

Spis treści

1	Wprowadzenie	2
2	Cel i zakres działania	2
2.1	Korzyści płynące z dokumentu	2
2.2	Zakres testów	2
2.2.1	Obejmuje	2
2.2.2	Wykluczenie	3
2.2.3	Szczególne wykluczenie	3
2.3	Proces testowania	4
2.4	Kryteria	4
2.4.1	Warunki początkowe	4
2.4.2	Warunki końcowe	5
3	Fazy testów i cykle	5
3.1	Fazy	5
3.2	Wydania systemu AskAnything	6
3.3	Podsumowania testów	6
4	Błędy	6
4.1	Wykrywanie błędów	6
4.2	Klasyfikacja błędów	6
4.3	Naprawa	7
4.4	Status błędów	7
5	Harmonogramy	8
6	Zasoby	8
6.1	Zasoby ludzkie	8
6.1.1	Inne	8
6.1.2	Zasoby sprzętowe	8
6.1.3	Oprogramowanie środowiska testowego	8
7	Ryzyko i zależności	9
7.1	Ryzyko	9
7.2	Zależności	9
8	Historia zmian	10
9	Podpisy	10

1 Wprowadzenie

Celem tego projektu jest implementacja nowatorskiego systemu przeprowadzania ankiet jakim jest AskAnything, który :

- Bazuje na nowoczesnych środkach przekazu informacji: internet.
- Motywuje uczestników do jak najczęstszego brania udziału w programie poprzez przydziałanie punktów wymienianych na nagrody.
- Dostarczanie ankiet drogą internetową, (czas od otrzymania zlecenia do dostarczenia do ankietowanych-maks.24h)
- dostępność dla klienta, jak i pomoc dla uczestników programu - 24/7.

2 Cel i zakres działania

2.1 Korzyści płynące z dokumentu

Celem tego dokumentu jest opisanie podejścia do testowania systemu AskAnything, którego szczegóły zostaną podane w kolejnej podsekcji.

Skrót zawartości dokumentu:

1. Aplikacje, które zostaną i nie zostaną przetestowane.
2. Rodzaje testów, które zostaną wykonane.
3. Ogólna strategia oraz czynności do wykonania.
4. Wymagane zasoby.
5. Harmonogram testów.
6. Osoby odpowiedzialne.

2.2 Zakres testów

2.2.1 Obejmuje

W tym wydaniu produktu chcemy zapewnić:-

Pierwsza faza:

- Gotową do użycia bazę danych ankiet.
- Zbiór gotowych ankiet przeprowadzonych wcześniej w tradycyjny sposób.

- Gotowy przejrzysty serwis WWW.
- Możliwość rejestracji ankietowanych on-line.
- Możliwość wypełnienia ankiety on-line.
- Możliwość zdefiniowania ankiety.
- W pełni funkcjonalny system punktowy.
- Gotowy sklepik internetowy.

Druga faza:

- Możliwość skorzystania z konsultacji.
- Powiadamianie o nowych ankietach.
- Powiadamianie o wynikach ankiet.
- Konta klientów on-line.

2.2.2 Wykluczenie

Kiedy zakres testów zostanie zatwierdzony i podpisany, żadne przyszłe wymagania nie będą rozważane jako dotyczące tego wydania, z wyjątkiem:

- Kiedy będzie wyraźne przyzwolenie i zgoda Kierownika Testów.
- Kiedy zmienione/dodatkowe wymagania nie będą wymagały znaczących wysiłków ze strony ekipy testującej (np. wymagających dodatkowych przygotowań) i nie wpłyną znacząco na harmonogram testów.
- Testowanie będzie typu „czarna-skrzynka” (patrz niżej).

2.2.3 Szczególne wykluczenie

- Nie będzie innych testów niż te zawarte w specyfikacji.
- Testowanie na sprzęcie innym niż podany w tym dokumencie jest poza zakresem tego planu.
- Testowanie przez osoby inne niż podany w tym dokumencie jest poza zakresem tego planu.

2.3 Proces testowania

Testowanie systemu - testowanie typu „czarna sprzynka” oparte na wymaganiach funkcjonalnych zawarych w specyfikacji. Muszą zapewniać adekwatne pokrycie wymagań oraz powtarzalność.

Testowanie wstecz - powtórne testowanie po naprawie wszystkich błędów lub modyfikacji produktu, bądź środowiska. Może być trudne określenie jak dużo testów wstecz jest potrzebnych zwłaszcza pod koniec rozwijania systemu AskAnything. Zautomatyzowane narzędzia testujące mogą być bardzo przydatne dla tego typu testów.

Testowanie wydajnościowe - testowanie w celu sprawdzenia czy aplikacja spełnia wymagania wydajnościowe zawarte w specyfikacji: wysokie obciążenia serwera WWW, duże złożone zapytania do bazy danych, wprowadzanie ogromnych wartości liczbowych, itp.

Testowanie bezpieczeństwa - sprawdzanie jak dobrze system AskAnything jest zabezpieczony przed nieautoryzowanym dostępem, umyślnych uszkodzeń („sql injections”, „cross-site scripting”), itp; może wymagać wyszukanych sposobów testowania.

Testowanie funkcjonalności - patrz testowanie systemu.

Testowanie integralności - testowanie połączonych części systemu AskAnything w celu sprawdzenia czy poprawnie funkcjonują razem. Częściami mogą być moduły, oddzielne aplikacje, aplikacje klienta i serwera w sieci, itp.

Testowanie typu „czarna-skrzynka” - testowanie aplikacji bez szczegółowej wiedzy na temat projektu bądź kodu. Testy oparte o wymagania i funkcjonalność.

Testowanie typu „biała-skrzynka” - oparte o wiedzę na temat projektu oraz kodu aplikacji, gdy programista wie co oddzielny fragment kodu ma za zadanie wykonywać i jest w stanie prześledzić to krok po kroku.

2.4 Kryteria

2.4.1 Warunki początkowe

Warunki początkowe określone przez Kontrolera Testów, powinny być spełnione zanim Testy Systemu będą mogły się zacząć. W przypadku kiedy jakiś warunek nie został spełniony, testy mogą się rozpocząć pod warunkiem, że Kierownik Projektu i Kontroler Testów będą zgodni, że ryzyko jest pod kontrolą.

- Cały kod musi być przetestowany. Testowanie jednostkowe musi być zakończone i podpisane przez ekipę programującą.
- Test Systemu musi być podpisany przez Analityka i Kontrolera Testów.

- Wszystkie zasoby ludzkie muszą być przydzielone oraz stawić się na miejsce.
- Wszystkie testy sprzętu oraz środowiska muszą być dostarczone do wglądu użytku przy testowaniu systemu.
- Testy akceptacji muszą być zakończone ze stanem nie mniejszym niż 80%.

2.4.2 Warunki końcowe

Określone poniżej cele muszą być osiągnięte, aby uznać pierwszą, a potem drugą fazę testów za wykonaną.

- Wszystkie błędy o wysokim priorytecie(krytyczne), które wystąpiły podczas procesu testowania muszą być naprawione i powtórnie przetestowane.
- Jeśli jakiegokolwiek błędy o średnim bądź niskim priorytecie nie są naprawione - błędy implementacyjne muszą zostać podpisane jako „do przyjęcia” przez Analityka i Kierownika Projektu.
- Testy Integracji Systemu musi być podpisany przez Kontrolera Testów oraz Analityka.

3 Fazy testów i cykle

3.1 Fazy

Testy zostały podzielone na 4 fazy wyszczególnione na Harmonogramie. Planujemy przetestowanie każdej w większych części systemu AskAnything, w celu wydania pierwszej wersji jak najwcześniej, nawet bez dodatkowych funkcjonalności. W każdym cyklu testujemy jak nowe funkcjonalności współdziałają same oraz z już gotowymi, w każdej fazie wykonujemy testowanie wstecz, aby upewnić się, że następny etap implementacji systemu na solidne podstawy. Każda z faz testowania powinna zakończyć się rozwiązaniem wszystkich krytycznych problemów.

Narzędzia do automatycznego testowania będą wykorzystywane przy testach funkcjonalnych oraz wstecznych, aczkolwiek głównym celem jest zautomatyzowanie testów wstecz. Będą one wykorzystywane w fazach testowania bazy danych oraz testów pracy z ankietami. W fazie trzeciej i czwartej testy te będą również wykonywane lecz, aby zapewnić maksymalny zysk z tych testów, będą one wykonywane kilkakrotnie. Na koniec przeprowadzimy testowanie systemu tylko pod kątem wykrytych wcześniej błędów, aby upewnić się, że problemy zostały rozwiązane.

Szczegółowe schematy testów będą ustalone w każdej fazie przygotowywania testów dla odpowiedniej części systemu.

3.2 Wydania systemu AskAnything

Planowane są dwa wydania systemu: po trzeciej i czwartej fazie testów. Szczegóły w Harmonogramie. Po zakończeniu trzeciej fazy testów żadne nowe wymagania nie mogą zostać dostarczone.

3.3 Podsumowania testów

Po pomyślnym zakończeniu każdej z czterech faz zostanie dostarczony raport z przebiegu każdej z nich. Dodatkowo po fazie drugiej i czwartej zostaną dostarczone wyniki testów wydajnościowych systemu.

4 Błędy

4.1 Wykrywanie błędów

Wszystkie błędy i podejrzewane błędy znalezione podczas testowania będą umieszczone przez każdego testera w Systemie Zarządzania Błędami z jasno określonym statusem i ścisłym opisem. System będzie zarządzany przez Kierownika Ekipy Testującej. Każdy krytyczny błąd mogący przedłużyć fazę testów, będzie bezpośrednio zgłaszany do Kierownika Ekipy.

Codziennie Kierownik Ekipy Testującej, Kierownik Przedsięwzięcia oraz Kierownik Programistów (Ekipa Nadzorująca Błędy) będą się spotykać w celu prześledzenia wykrytych błędów oraz dla każdego: przypisania doń osoby w celu naprawy, odłożenia, bądź zamknięcia. Będą sporządzane dzienne raporty zawierające listę wykrytych błędów, status, listę naprawionych błędów.

4.2 Klasyfikacja błędów

Zaakceptowane błędy są klasyfikowane do:

- Kategoria A - Poważne błędy, które wstrzymują przeprowadzanie dalszych testów danej funkcji, bądź poważne błędy danych (np. brak komunikacji między serwisem a bazą danych).
- Kategoria B - Poważne błędy danych bądź brak danych, które nie zatrzymują procesu (np. dane nie przyjmują domyślnych wartości (przyciski typu radio lub checkbox w ankietach) i trzeba samemu podawać odpowiednią wartość).
- Kategoria C - Drobne błędy nie wstrzymujące procesu ani nie szkodzą funkcjonalności (np. literówki).

5 Harmonogramy

Szczegółowe harmonogramy faz przygotowań testów oraz terminy ich przeprowadzenia zostały zawarte w harmonogramach załączonych na końcu dokumentu.

6 Zasoby

6.1 Zasoby ludzkie

Typ	Nazwa	Liczba	Kiedy potrzebny	Kto	Status
Kierownik	Kierownik Projektu	1	18.02.2006	Jan Urbański	Przypisany
Przedsięwzięcie	Kierownik Przedsięwzięcia	1	18.02.2006	Maciej Stefański	Przypisany
Pomoc techniczna	Inżynier Baz Danych	1	18.02.2006	Max Maksymilian	Przypisany
Programowanie	Kierownik Programistów	1	18.02.2006	Jakub Rauch	Przypisany
Testowanie	Kontroler Testów	1	18.02.2006	Sebastian Tomaszewski	Przypisany
Testowanie	Tester	3	18.02.2006		Nieprzypisany
Automatyzacja	Tester - automatyzacja	2	18.02.2006		Nieprzypisany

6.1.1 Inne

Dla celów testowych będziemy potrzebowali oddzielnego systemu złożonego ze sprzętu i oprogramowania. Aby zapewnić integralność środowiska, sieć nie może być dostępna z zewnątrz.

6.1.2 Zasoby sprzętowe

- 6 komputerów (P IV 2.0GHz, 100Gb HD, 512Mb RAM [Minimalna konfiguracja])
- Baza danych Oracle 10g
- Server Apache 2.2.0
- HP LaserJet 1160

6.1.3 Oprogramowanie środowiska testowego

- Windows XP/2000

- Linux Gentoo
- Mac OS 9
- Internet Explorer 6.0
- Opera 8.5
- NetscapeBrowser 8.1
- MS Office XP/2000
- MS Access 2003

7 Ryzyko i zależności

Testowanie może być poważnie zaburzone, owocując w niekompletne lub nieadekwatne przetestowanie systemu AskAnything, bądź w wyraźne przesunięcie terminów, poprzez następujące ryzyka oraz zależności.

7.1 Ryzyko

1. Braki w testowaniu jednostkowym.
2. Ograniczenia budżetowe.
3. Niedokładne wymagania w specyfikacji.
4. Przeciągnięcie się okresu implementacji.
5. Problemy przy pracy z kodem.

7.2 Zależności

1. Oprogramowanie zostanie dostarczone na czas.
2. Programowanie jest wymaganej jakości pozwalającej na testowanie.
3. Błędy powstrzymujące pracą będą natychmiast dostarczane do Kierownika Ekipy Testującej.
4. Znalezione błędy systemu będą naprawione i przetestowane jednostkowo przez Ekipę Programującą.
5. Problemy przy pracy z kodem.
6. Wymagane zasoby będą dostępne.

7. Wszystkie zobowiązania zostaną spełnione.
8. Automatyczne narzędzie testujące będzie działało prawidłowo z systemem AskAnything.
9. Dostarczone dokumenty będą aktualne.
10. Łacze internetowe będzie w sprawne na tyle, na ile zostało określone w specyfikacji.

8 Historia zmian

01.06.2006 - pierwsza wersja\\

02.06.2006 - dodany spis treści oraz poprawione wykresy

9 Podpisy

Ten dokument musi być formalnie zatwierdzony zanim testy się rozpoczną. Następujące osoby muszą wyrazić zgodę:

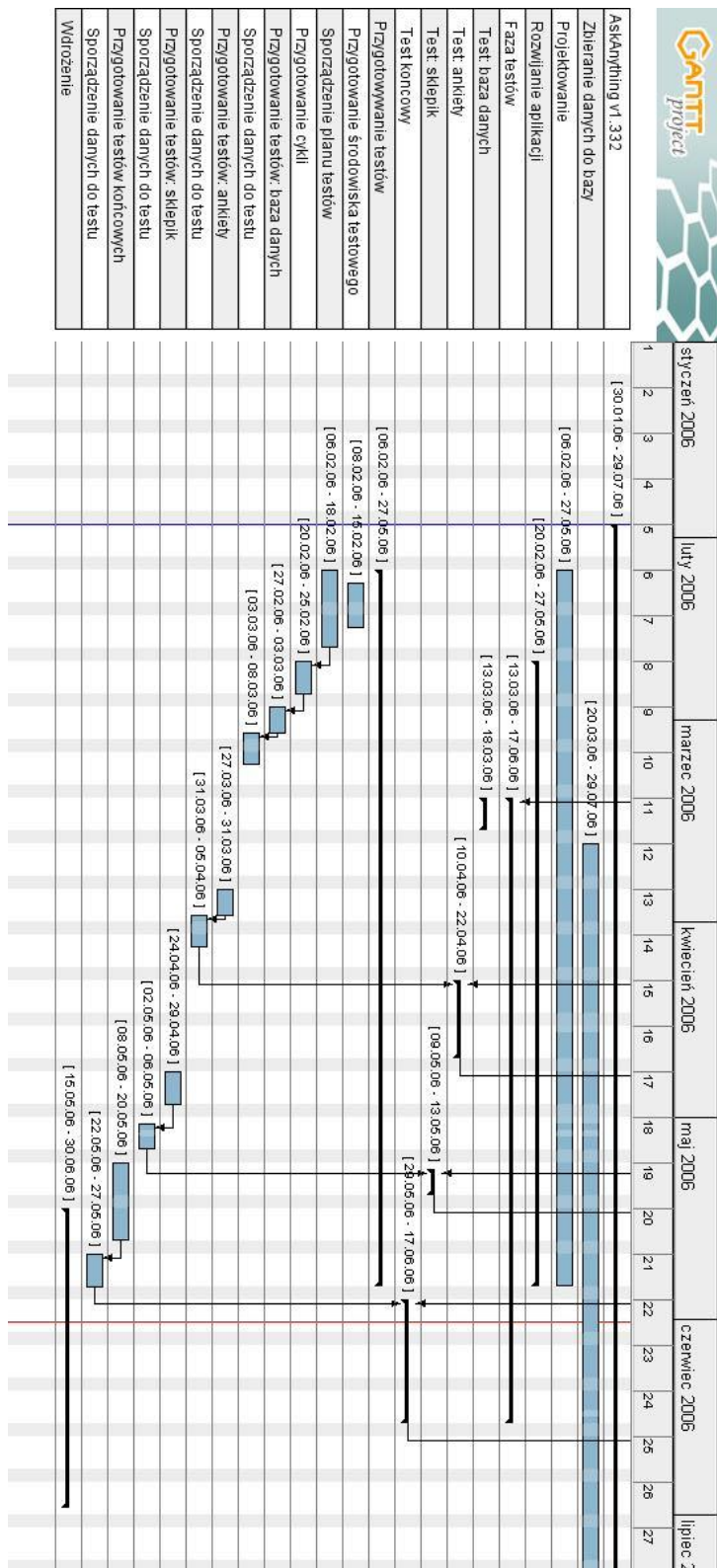
Podpis zbiorowy

Kierownik Projektu - Jan Urbański

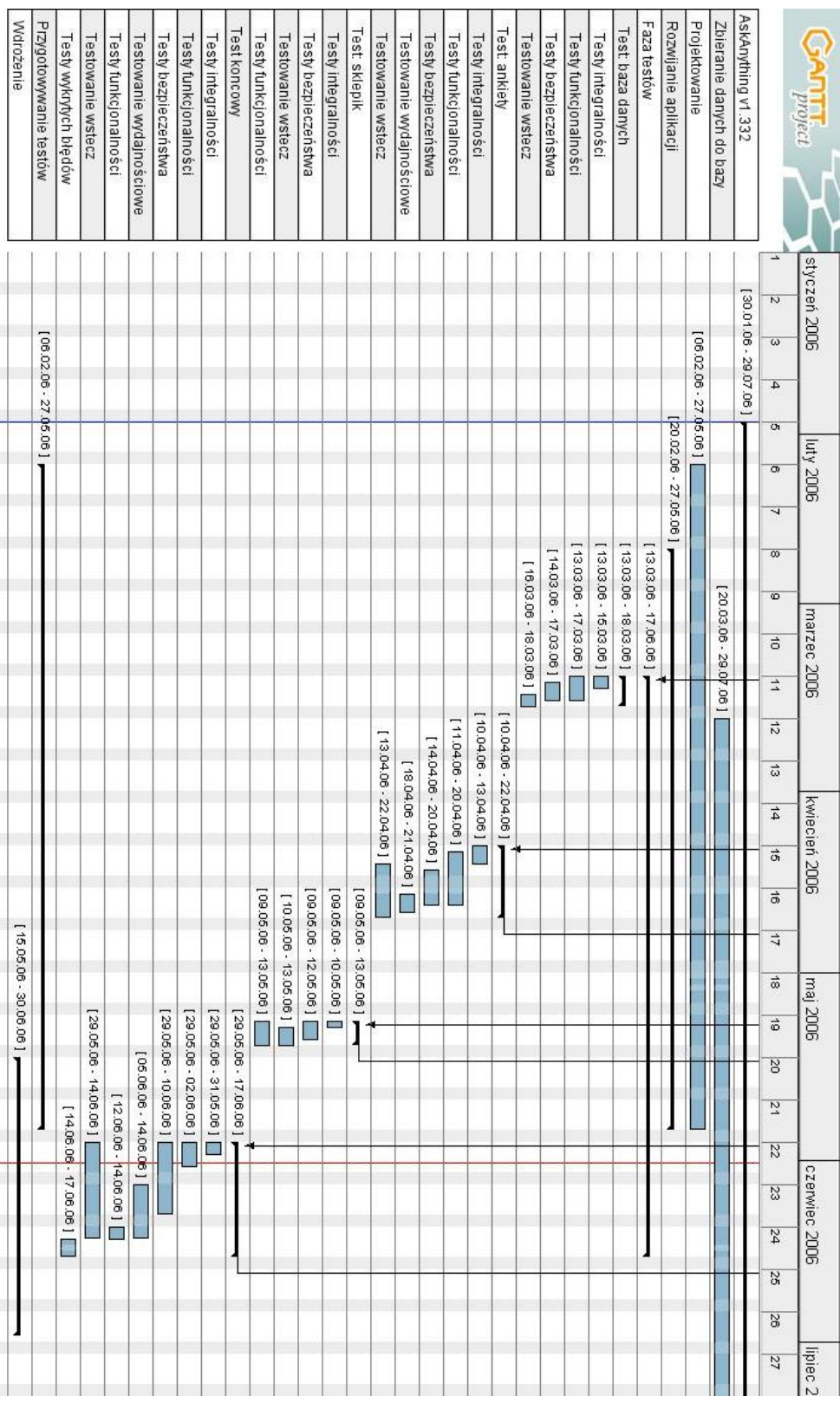
Kierownik Przedsięwzięcia - Maciej Stefański

Ekipa Programistyczna - Jakub Rauch

Ekipa Testująca - Sebastian Tomaszewski



Rysunek 2: Przygotowanie testów.



Rysunek 3: Przeprowadzanie testów.