

Studencka Pracownia Inżynierii Oprogramowania
Instytut Informatyki Uniwersytetu Wrocławskiego

Krystian Jasionek, Jakub Skalski

Faza wprowadzenia na rynek

Wrocław, 24 stycznia 2021

Wersja 0.3

Historia zmian dokonanych w dokumencie

Data	Numer wersji	Opis	Autor
2021-01-23	0.1	Utworzenie dokumentu	Krystian JasioneK
2021-01-24	0.2	Korekta dokumentu	Krystian JasioneK
2021-01-24	0.3	Korekta dokumentu	Jakub Skalski

Spis treści

1. Plan wdrożenia	3
1.1. Przeprowadzenie analizy przedwdrożeniowej	3
1.2. Wypełnienie baz danych	3
1.3. Szkolenie użytkowników docelowych i administratorów systemu .	3
1.4. Kontrola systemu i usuwanie błędów	3
1.5. Udostępnienie gotowego systemu	3
2. Oszacowanie pojemności bazy danych	3
3. Koncepcja organizacji szkoleń użytkowników	5
4. Koncepcja wsparcia technicznego oraz usuwania błędów	5
5. Główne punkty umów	5
5.1. Gwarancja	5
5.2. Odpowiedzialność za wady oprogramowania	5
5.3. Zaplanowanie zmian	5
5.4. Prawa autorskie	6
5.5. Odebranie oprogramowania	6
6. Koncepcja pomiaru satysfakcji klienta	6
7. Przypisy	6

1. Plan wdrożenia

1.1. Przeprowadzenie analizy przedwdrożeniowej

Możemy zwiększyć szanse sukcesu wprowadzenia aplikacji na rynek poprzez przeprowadzenie odpowiedniej analizy, np. szacując koszty, przewidując możliwe błędy i problemy lub opóźnienia.

1.2. Wypełnienie baz danych

Do baz danych należy wprowadzić wbudowane w aplikację podstawowe algorytmy oraz wzory. Należy również przygotować ich kopie zapasowe, potrzebne w razie awarii systemu.

1.3. Szkolenie użytkowników docelowych i administratorów systemu

Aplikacja jest z założenia prosta w użyciu, nawet przez użytkowników słabo zaznajomionych z algorytmiką i grafiką komputerową, zatem program ich szkolenia ograniczymy do zamieszczenia samouczka, wprowadzającego w najważniejsze funkcje systemu oraz zapewnienia pomocy w postaci, między innymi infolinii oraz FAQ.

1.4. Kontrola systemu i usuwanie błędów

Wiemy, że nie możemy przetestować wszystkich danych wejściowych i stanów systemu, ponieważ liczba ich kombinacji jest zbyt duża. Możemy natomiast podzielić dane wejściowe na klasy (tj. spodziewamy się, że system zachowa się w podobny sposób dla elementów danej klasy) i przetestować go dla każdej z tych klas. Następnie przeprowadzimy testy przejść między stanami systemu, które można przedstawić np. w formie diagramu. Bardzo skutecznym sposobem mogą się również okazać testy strukturalne, gdzie postaramy się sprawdzić wszystkie rozgałęzienia instrukcji warunkowych. Zależy nam również, by każdy fragment kodu został wykonany przynajmniej raz. Ponadto zorganizujemy odpowiedni zespół wsparcia technicznego i konserwacji systemu.

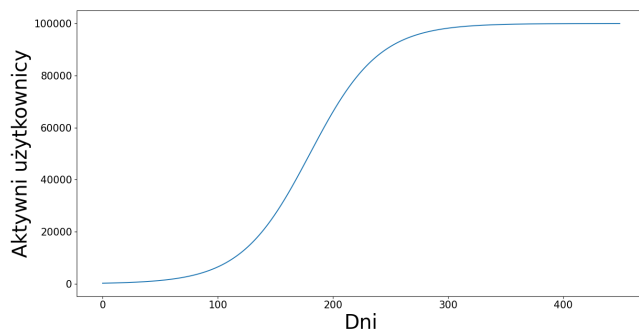
1.5. Udostępnienie gotowego systemu

Aplikacja zostaje zamieszczona w Sieci. Użytkownicy mogą ją pobrać na swoje komputery lub korzystać z niej z poziomu przeglądarki. Aplikacja jest dostępna dla każdego bezpłatnie, natomiast jej kod źródłowy nie jest upubliczniany.

2. Oszacowanie pojemności bazy danych

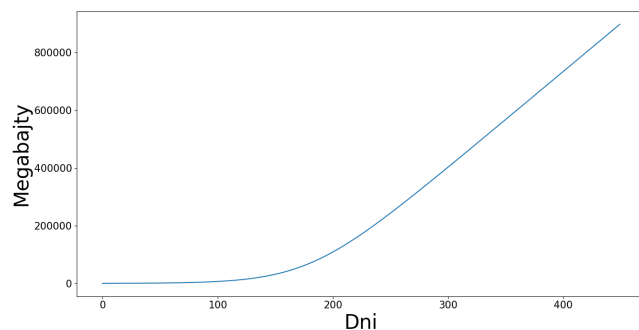
Przeprowadziliśmy poniższe szacowania na okres roku kalendarzowego. Przewidujemy, że liczba aktywnych użytkowników naszej aplikacji będzie rosła powoli

przez pierwsze kilka tygodni od wprowadzenia jej na rynek. Dopiero po tym czasie nabierze ona popularności i zaobserwujemy znaczny wzrost ich liczby, utrzymujący się przez dwa miesiące. Po tej fazie nastąpi spadek zainteresowania i liczba użytkowników będzie utrzymywała się na w przybliżeniu stałym poziomie.



Rysunek 1. Aktywni użytkownicy

Przyjeliśmy w szacowaniach, że pojedynczy użytkownik jest w stanie wygenerować co najwyżej 1 MB danych miesięcznie, gdzie danymi jest kod algorytmów oraz informacje o koncie użytkownika – jego dane personalne, zawartość biblioteki wzorów itp., co po pierwszym roku daje nam przyrost ok. 1 TB danych i w kolejnych latach 100 GB miesięcznie. Ustaliliśmy, że na potrzeby naszego systemu wystarczająca będzie baza danych o pojemności 50 TB. Dla oszczędności miejsca co roku będziemy usuwali konta nieaktywnych użytkowników oraz najrzadziej wykorzystywane algorytmy z biblioteki, informując uprzednio ich twórców o możliwości utraty tych danych.



Rysunek 2. Rozmiar danych

3. Koncepcja organizacji szkoleń użytkowników

System będzie oferował samouczek dla nowych użytkowników oraz dokumentację systemu i wszystkich zawartych w nim funkcji, np. obsługi kreatora wzorów i języka skryptowego algorytmów.

4. Koncepcja wsparcia technicznego oraz usuwania błędów

Zatrudniony zostanie zespół wsparcia technicznego, którego zadaniem będzie odbieranie zgłoszeń o błędach oraz udzielanie pomocy użytkownikom. System zostanie wyposażony w odpowiednie środki komunikacji z pomocą techniczną – infolinia, mechanizm zgłaszania błędów, formularze z pytaniami do zespołu, stale uaktualniane FAQ. Zespół wsparcia technicznego będzie prowadził rejestr zgłaszanych błędów, który następnie przekaże zespołowi programistycznemu odpowiedzialnemu za przeprowadzenie odpowiednich poprawek.

5. Główne punkty umów

5.1. Gwarancja

Zobowiązujemy się do bezpłatnej naprawy wszelkich błędów występujących w systemie, wykrytych w ciągu roku od wdrożenia aplikacji, przy czym należy zaznaczyć, że błędy podlegające gwarancji odnoszą się wyłącznie do wymagań funkcjonalnych systemu.

5.2. Odpowiedzialność za wady oprogramowania

Jeżeli końcowy produkt nie spełnia wszystkich ustalonych wymagań narzuconych przez zleceniodawcę, godzimy się wypłacić odszkodowanie nieprzekraczające 20 procent kosztu wyprodukowania oprogramowania oraz zobowiązujemy się wprowadzić odpowiednie zmiany w celu naprawy systemu. Jeżeli niespełnione wymagania dotyczą zatwierdzonego już wcześniej etapu produkcji oprogramowania, wtedy zleceniodawcy nie przysługuje odszkodowanie.

5.3. Zaplanowanie zmian

Jesteśmy gotowi wprowadzić odpowiednie zmiany na życzenie klienta tak długo, jak nie zmieniają one drastycznie ogólnej wizji projektu. Wszelkie zmiany tego typu wykraczające poza przyjętą specyfikację wymagają dokładnego ustalenia szczegółów ich realizacji oraz wiążą się z naliczeniem dodatkowych kosztów.

5.4. Prawa autorskie

Umowa powinna zawierać ustalenia dotyczące praw autorów aplikacji do gotowego produktu, kwestii związanych z udzielaniem licencji na korzystanie lub rozpowszechnianie oprogramowania oraz przenoszeniem praw autorskich na inny podmiot. Konieczne jest również ustalenie praw użytkowników do utworzonych przez nich algorytmów i prac graficznych oraz warunków, na jakich mogą być one rozpowszechniane w sposób odpłatny lub darmowy.

5.5. Odebranie oprogramowania

Zawieramy również opis procedury odebrania gotowego produktu, co pozwoli na jasne określenie, kiedy ono nastąpiło. Jako załącznik do umowy można również zawrzeć wzór protokołu odbioru produktu.

6. Koncepcja pomiaru satysfakcji klienta

Użytkownicy będą raz w miesiącu proszeni o wzięcie udziału w nieobowiązkowej ankiecie, zawierającej pytania odnośnie do oceny wprowadzonych do aplikacji w ostatnim czasie modyfikacji i funkcji, propozycji zmian i usprawnień produktu oraz ogólnej oceny usługi. W pierwszych kilku godzinach interakcji z systemem użytkownicy będą zapytani o ich stopień zadowolenia (w skali liczbowej od 1 do 5) z korzystania z aplikacji. Tym sposobem lepiej zrozumiemy jakie wrażenie aplikacja wywiera na nowych, jak i regularnych użytkownikach. Ponadto będziemy mierzyć współczynnik retencji aktywnych użytkowników (jak długo średnio udaje się utrzymać zainteresowanie użytkownika aplikacją).

7. Przypisy

- Główne punkty umów na podstawie:

<https://legalgeek.pl/umowa-wdrozeniowa-it-co-powinna-zawierac/>