Dawid Holewa 317898 Kolokwium IO nr 1

1. Aby podjąć decyzję, czy warto napisać aplikację muszę przeprowadzić fazę wstępną: zorientować się co i dlaczego potrzebuje klient poprzez wspólną rozmowę, określić cele z punktu widzenia klienta, ogólnie określić wymagania, wykonać zgrubną analizę i koncepcję systemu. Należy również oszacować koszty rozwiązań oraz wstępny harmonogram prac i strukturę zespołu. Po przeanalizowaniu tych kroków należy zaprezentować wyniki swojej pracy klientowi na przykład w postaci tablicy koncepcyjnej, aby uniknąć ewentualnych nieporozumień. Dopiero wtedy możemy podejmować decyzję o wytworzeniu oprogramowania.

Aby móc zatrudnić współpracowników, musimy najpierw oszacować takie czynniki jak koszt robocizny, rozmiar projektu (w celu określenia ilu pracowników będzie nam potrzebnych), ograniczenia czasowe, warunki wstępne i umiejętności niezbędne do realizacji projektu.

1a. Należy się zapoznać z obowiązującym prawem finansowym, sposobem importu i eksportu usług, czy kraj należy do UE (wtedy rozliczenia są łatwiejsze). Należy zbadać koszt wytwarzania oprogramowania w naszym kraju i kraju xxx, koszty operacji bankowych i rozliczeń, koszty zatrudnienia specjalistów w kraju XXX i naszym.

1b. Wytwarzanie części oprogramowania w kraju XXX wiąże się z pewnymi dodatkowymi kosztami: należy przeanalizować ilu pracowników wytwarzałoby część oprogramowania poza granicami kraju, ponieważ trzeba zapewnić im również kosztowną przestrzeń biurową oraz sprzęt. Należy również zatrudnić zaufaną osobę zarządzającą projektem w kraju XXX, która kontroluje swoich podopiecznych. Wykonywanie części oprogramowania w innym kraju wiąże się również z rozpoznaniem prawnym w tym kraju, co tworzy potrzebę zatrudnienia prawnika, który rozpoznałby nowe problemy powstałe w wyniku pracy na innych prawach.

1. Wymagania klienta zawrę w dokumencie Specyfikacji wymagań produktu, który będzie zawierał historyjki użytkowania, co pomoże zobrazować pracownikom funkcjonalności, na jakich się skupiamy. Wypunktuję dokładnie wymagania funkcjonalne w postaci hierarchicznej oraz wymagania niefunkcjonalne, które zostaną ówcześnie przedyskutowane i zatwierdzone przez klienta. Można również zatrudnić osobę, która specjalizuje się w tłumaczeniu dokumentacji technicznych, dzięki czemu unikniemy problemów związanych z barierą językową.
2. Model powinien pokazać, co system ma robić, ma być zorganizowany hierarchicznie i być na wysokim poziomie abstrakcji. W tym celu wykonamy następujące kroki: przeanalizujemy wymagania klienta i na podstawie tego utworzymy koncepcję fizycznej budowy systemu, przeanalizujemy wymagania funkcjonalne i niefunkcjonalne, napiszemy historyjki użytkowników, i analizę przypadków użycia w celu utworzenia wspólnego wyobrażenia. Po tak skonstruowanej koncepcji postaramy się zidentyfikować logiczne składniki systemów, utworzyć pierwszy model baz danych i klas. Postaramy się również rozpoznać potrzebne nam algorytmy. Będziemy starali się utworzyć model stanów i przejść między nimi, procesów i przepływu danych oraz przepływ sterowania.
3. Utworzę dokument, który będzie dokładnie opisywał następujące modele: diagram baz danych, klas, interakcji między obiektami, planowanymi object design pattern, diagram stanów, modułów, konfiguracji. Przydatne będą również informacje dotyczące definicji klas, ich atrybutów, danych złożonych i elementarnych, sygnatury i opis niezbędnych metod. Zaprezentuję również prototyp interfejsu użytkownika. W dokumencie powinien również znajdować się projekt fizycznej struktury systemu, plan testów oraz harmonogram fazy implementacji. Tak dobrana dokumentacja powinna rozwiać wątpliwości pracowników. W razie jednak zaistnienia problemów, posiadamy osobę nadzorującą pracę i kontrolującą niezrozumienia, która będzie sygnalizowała niedomówienia osobom przełożonym i dostarczała odpowiednich rozwiązań.
4. Aby zweryfikować pracę i zamierzone cele do zrealizowania w kraju XXX wykorzystamy project managera, który powinien przez cały czas trwania projektu kontrolować i monitorować starania swoich pracowników. Wymagania niefunkcjonalne będziemy kontrolowali również poprzez regularne code review, regularne testy obciążeniowe, weryfikację testów oprogramowania i dokładne sprawdzanie dokumentacji technicznej. W celu kontroli zadowolenia klientów możemy wykorzystywać testerów, którzy będą sprawdzali poszczególne moduły systemów i wysyłali swój subiektywny feedback dotyczący przyjemności z użytkowania systemu.