

Jakość - moda czy konieczność?

Adam Vieweger, Key Account Manager, Borland Polska

Jak większość osób pracujących w branży IT, staram się być na bieżąco z trendami na rynku informatycznym i zmianami na nim zachodzącymi. Jako stały czytelnik gazet, magazynów i wszelkiego rodzaju informatorów branżowych, od pewnego czasu z rosnącą frustracją obserwuję wzrost tematyki poświęconej nieudanym projektom - szczególnie w ramach sektora publicznego. Nowe doniesienia tego typu są niemal na porządku dziennym, a już z pewnością tygodniowym. Z premedytacją nie wymieniam konkretnych przykładów z naszego podwórka, kto ciekaw, zapraszam do próby załatwienia dowolnej sprawy w jednym z licznych urzędów publicznych bez stania w kolejkach i wypełniania skomplikowanych formularzy.

Porażki informatyczne bywają przyczyną poważnych problemów dla dostawców technologii, czasem nawet upadku pomniejszych z nich. W zdecydowanej większości przypadków, wynikają one ze sposobu, w jaki projekt był prowadzony.

Geneza porażki

Niepisana negatywna opinia o projektach toczących się w ramach sektora publicznego, ale i nie tylko, utrzymuje się w mniej lub bardziej nagłościonej atmosferze skandalu. Zwłaszcza tam, gdzie w grę wchodzi projekty o olbrzymich budżetach. Pozostałe jakoś uchodzą naszej uwadze, pewnie głównie ze względu na stosunkowo niewielkie zainteresowanie nimi ze strony mediów. Problem jaki w związku z tym odczuwam, wiąże się z obawą, że wszyscy z wolna wydajemy się oswajać z podobnym stanem rzeczy, przechodząc do porządku dziennego nad kolejnymi "problemami firmy X z kontraktem Y", co samo w sobie jest już przerażające. Czas najwyższy powiedzieć sobie, że wszelkiego typu "porażek" projektów IT możemy uniknąć i że nie ma potrzeby traktowania ich jako zło konieczne.

Organizacje zarówno z sektora publicznego, jak i te z prywatnego, muszą być świadome, że tworzenie oprogramowania nie jest "sztuką", że dziś stało się ono w pełni zarządalnym procesem biznesowym, który może i musi być kontrolowany jak każdy inny proces w przedsiębiorstwie. Jednym z podejść, które mogą nam w tym pomóc jest Cykl Życia Aplikacji (Application Lifecycle Management - ALM). Łączy on wszystkie fazy, zadania i zasoby zaangażowane w dostarczanie oprogramowania w sposób umożliwiający przejście z chaotycznego procesu wytwarzania oprogramowania w kontrolowalną i przede wszystkim przewidywalną procedurę. Niezwykle istotnym aspektem ALM jest ułatwienie zarządzania jakością w całym cyklu życia aplikacji - od sprawdzania poprawności zdefiniowanych

wymagań, aż do weryfikowania wyników testów funkcjonalnych. Jak na ironię, jakość, stanowiąca podstawę oceny dostarczanego produktu, bardzo często jest pomijana w trakcie jego tworzenia.

Jakość to nie Jakość

Działania mające na celu zapewnienie jakości w cyklu wytwarzania oprogramowania są często brane pod uwagę dopiero na ostatnim etapie produkcji aplikacji, tuż przed dostarczeniem jej do klienta. Oznacza to nic innego, jak fakt, że sprawdzanie naszego produktu pod względem jakości zaczyna się bardzo późno, więc jakiegokolwiek problemy wykryte w tej fazie są trudne i kosztowne do usunięcia. W takich przypadkach najczęściej poprawki wprowadzane są w pośpiechu, chaotycznie, rozmijając się nierzadko z wymaganiami i przy zaangażowaniu oraz wiedzy tylko części zespołu projektowego. Niektóre firmy świadomie umieszczają zagadnienia związane z jakością na samym spodzie listy priorytetów w zamian za przesunięcie wyżej zadania dostarczenia produktu na czas i w założonym budżecie - podejście, które w większości przypadków w ostateczności prowadzić może do katastrofy.

Dojrzałość przedsiębiorstwa informatycznego ocenia się według jakości tworzonych i dostarczanych systemów oraz, a może nawet przede wszystkim, według umiejętności dotrzymania terminów i ustalonych z klientami budżetów na realizowanie projektów. W praktyce oznacza to, że wszystkie podejmowane w ramach danego projektu zadania muszą być zgodne z planem, a kontrola jakości produktu częścią każdego etapu jego dostarczania. O ile w sensie potocznym pojęcie jakości może być rozumiane subiektywnie, to w obszarze IT musi być to proces w pełni mierzalny w zakresie wymagań, poprawności kodu oraz liczby usterek, tak, by całość projektu spełniała stawiane przed nim wymagania i cele biznesowe.

Łatwiej zapobiegać niż leczyć

Dostarczanie oprogramowania powinno charakteryzować się podejściem proaktywnym, w przeciwieństwie do reaktywnego, lawinowego, które wciąż trzyma się w czołówce najczęściej stosowanych praktyk na rynku. Jak twierdzą lekarze, zapobiegać jest łatwiej niż leczyć. Firmy powinny skupić się usprawnianiu swojego podejścia do jakości, gdyż dzięki temu będą w stanie świadomie i w powtarzalny sposób dostarczać oprogramowanie najwyższej jakości, co w efekcie da wymierne rezultaty w postaci kończenia projektów w założonym czasie i budżecie, a często ich pomyślnego zakończenia w ogóle. Błędy spowodowane przez brak kontroli jakościowej na wszystkich etapach realizowania projektu mogą mieć przyczyny między innymi w:

- sprzeczności z warunkami licencji
- niepoprawnym modelu
- małej czytelności kodu
- wieloznacznych wymaganiach
- nie spełnianiu zasad bezpieczeństwa

Wszystkie powyższe aspekty nie tylko mogą, ale powinny być uwzględnione, chociażby poprzez zapewnienie wysokiego priorytetu zadaniom jakościowym w całym cyklu tworzenia i dostarczania oprogramowania.

Zbyt często audyty jakości rozpoczynają się dopiero w fazie testowania, kiedy to projekt pochłonął już wiele czasu i pracy. W podejściu zapobiegawczym względy jakościowe wchodzi na scenę dużo wcześniej, często zanim powstanie pierwsza linijka kodu - kontrola jakości powinna pojawić się już na etapie definiowania założeń projektu.

Podsumowując - musimy wprowadzić kontrolę jakości do projektu już w momencie jego rozpoczęcia, a nie zwlekać z tym do fazy testów. Wczesne testowanie pozwala na uniknięcie wielu problemów z pozytywnym zamykaniem projektów IT.

Najważniejsze pierwsze dwa kroki

Firmy zaangażowane w realizowanie projektów informatycznych mogą zainauguować bardziej dojrzały model podejścia do jakości za pomocą dwóch prostych kroków. Pierwszym z nich jest skupienie się na jakości w fazach przedwdrożeniowych i zapobieganiu problemom już na wczesnych etapach projektu. Stosując takie podejście, możemy szybko i łatwo wyizolować błędy zanim ich wpływ na projekt i tworzony system będzie znaczny - lub w ekstremalnych przypadkach spowoduje wstrzymanie projektu.

Drugim krokiem jest zagwarantowanie, że pozycjonujemy jakość nie jako dodatkowy etap w procesie dostarczania oprogramowania, lecz jako nieodłączna i integralna część całego projektu z odpowiednio wysokim priorytetem. Dzięki temu zapewnimy sobie jasność, poprawność i jednolitość wymagań, klarowność kodu, zminimalizujemy liczbę usterek oraz, co nie mniej istotne, sprawimy, że nasze produkty będą bardziej zbliżone do realnych potrzeb klientów.

Historia nieudanych projektów informatycznych w sektorze publicznym jednoznacznie dowodzi, że niska jakość powstającego w ich ramach oprogramowanie pociąga za sobą

olbrzymie konsekwencje. Mimo to element jakości wciąż nie jest postrzegany jako czynnik biorący istotny udział w tych porażkach.

Dzisiaj, bardziej niż kiedykolwiek przedtem, istnieje potrzeba przełamania barier pomiędzy tradycyjnym podejściem do tworzenia oprogramowania, a rolą jakości. Musimy nauczyć się i zrozumieć, że jakość jak tama w strumieniu projektu reguluje jego przepływ skutecznie powstrzymując napływ błędów. Żywię gorącą nadzieję, że dzięki takiemu spojrzeniu na dostarczanie oprogramowania z czasem przyjdzie nam czytać coraz mniej krytycznych artykułów w prasie branżowej.