

Iwona Chomiak-Orsa

Uniwersytet Ekonomiczny we Wrocławiu
e-mail: iwona.chomiak@ue.wroc.pl
ORCID: 0000-0003-3550-8624

Angelika Osińska

Uniwersytet Ekonomiczny we Wrocławiu
e-mail: angelika@esperia.pl

WPLYW ITIL NA PROCESY OBSŁUGI BŁĘDÓW OPROGRAMOWANIA

THE INFLUENCE OF THE ITIL ON THE PROCESS OF HANDLING SOFTWARE ERRORS

DOI: 10.15611/ie.2018.1.02

Streszczenie: Rozwój rynku produktów IT przyczynił się, z jednej strony, do wzrostu rozwiązań powielanych, z drugiej zaś – do poszukiwania przez przedsiębiorców rozwiązań idealnie dopasowanych do ich potrzeb. Zdeterminowało to zmiany w idei dostępności do kodu oraz możliwości realizacji modyfikacji w oprogramowaniu nie tylko przez deweloperów oprogramowania, ale również przez użytkowników końcowych. Każda jednak zmiana w oprogramowaniu przyczynić się może do powstania błędu, dlatego też przebieg procesu obsługi błędów oprogramowania jest ważnym zagadnieniem dla każdego przedsiębiorstwa, którego funkcjonowanie opiera się na systemach informatycznych. Proces ten powinien być jasny, sprawny, kontrolowany i ustawicznie doskonały. Dlatego też coraz częściej poszukuje się skutecznych narzędzi, które umożliwią skuteczne zarządzanie tym procesem. Artykuł niniejszy wskazuje, że skutecznym narzędziem może być model zarządzania usługami IT, jakim jest *Information Technology Infrastructure Library* (ITIL). Celem artykułu jest wykazanie, iż koncepcja ITIL jest właściwym wyborem dla problemu poprawy procesu obsługi błędów oprogramowania.

Słowa kluczowe: błąd oprogramowania, usługa, proces, ITIL, obsługa błędów, eksploatacja usług, doskonalenie procesu.

Summary: The process of handling software applications errors is crucial for all organisation relying on computer systems. The process itself should be clear, efficient, well controlled and constantly improved. In this article, the author is trying to give a sense of some aspects of the nature of software errors and their mitigation and remediation. As a tool used for improvement of the process, the author is using the model of IT services management, described in the Information Technology Infrastructure Library. The purpose of this article is to point out that the ITIL concept can be implemented to improve the process of handling software errors. This article is based on literature, Internet sources and the authors' professional experience. The first part is dedicated to software error issues and how they trigger the process of handling them. The second part describes the ITIL concept, its origins and main premise of Service

Operation. The last part includes a description of the software errors handling process, according to ITIL. The process roles, elements, course and its characteristics are explained.

Keywords: software error, service, process, ITIL, error handling, service operation, process improvement.

1. Wstęp

Specyfika realizacji współczesnych przedsięwzięć informatycznych rzutuje na jakość wytwarzanego produktu. Terminy wdrożeń są nieugięte i często podyktowane wymaganiami biznesowymi. Presja czasu jest czynnikiem wpływającym na występowanie nieprawidłowości w oprogramowaniu, a zbliżający się termin wydania produktu staje się wyznacznikiem jakości kodu. Powyższe czynniki powodują, że błędy są nieodzownym elementem cyklu wytwarzania oprogramowania, natomiast proces obsługi błędów musi być sformalizowany i ciągle doskonalony. Artykuł poświęcony został problematyce błędów oprogramowania i ich obsługi. Wyjaśnione zostały także podstawowe pojęcia związane z tym zagadnieniem. Według autorów właściwym narzędziem doskonalenia procesu obsługi błędów oprogramowania jest *Information Technology Infrastructure Library (ITIL)* w zakresie eksploatacji usług (*service operation*). Eksploatacja usług koordynuje procesy i działania, które są wymagane do dostarczenia usług i zarządzania usługami na poziomie ustalonym z biznesem. Ważnym aspektem w obsłudze produkcyjnych błędów jest perspektywa probiznesowa, którą wspiera ITIL. W końcowej części opracowania przedstawiony został proces obsługi błędów oprogramowania według ITIL w ujęciu kluczowych elementów procesu, przebieg procesu z uwzględnieniem ról w procesie oraz opisana została jego charakterystyka. Celem artykułu jest wskazanie na znaczenie elastycznych koncepcji zarządzania projektami. Podstawę do napisania artykułu stanowiły materiały literaturowe, materiały ze źródeł internetowych, certyfikacja ITIL Foundation, a także doświadczenia praktyczne, które kolejno ukształtowały procedurę badawczą. Pierwsza część artykułu poświęcona została zagadnieniom związanym z błędami oprogramowania. W drugiej opisano koncepcję ITIL, jej pochodzenie i podstawowe założenia w zakresie eksploatacji usług (*Service Operation*). W końcowej części artykułu przedstawiono proces obsługi błędów według ITIL. Opisano role w procesie, elementy procesu, przebieg procesu oraz jego charakterystykę.

2. Błędy w systemach informatycznych

Błędem oprogramowania nazywamy niepoprawną konstrukcję, która występuje w systemie informatycznym i może prowadzić do niewłaściwego działania programu. Błąd oprogramowania jest wadą programu komputerowego. Nieprawidłowości w systemach informatycznych wynikają z błędów człowieka na różnych etapach tworzenia oprogramowania – zazwyczaj jest to etap tworzenia kodu źródłowego, niejed-

nokrotnie jest to również etap projektowania oprogramowania. W mowie potocznej często używane jest pojęcie „bug”. Po raz pierwszy zostało ono wykorzystane w roku 1947, podczas testów komputera Harvard Mark II, przez admirał Grace Hopper. Gdy komputer przestał działać, okazało się, że przyczyną była ćma, która utknęła w urządzeniu¹. Definicji błędów oprogramowania jest wiele. Błąd jest nieprawidłowością w systemie, która może być anomalią wynikającą z niezgodności ze specyfikacją i wymaganiami biznesowymi. Błędem może być także nieprawidłowość w działaniu systemu wynikająca z wystąpienia awarii².

Błędy oprogramowania mogą mieć różną przyczynę wystąpienia. Ze względu na źródło wyróżnić można cztery podstawowe ich typy, co zostało zaprezentowane w tab. 1.

Tabela 1. Typologia błędów oprogramowania

Typ błędu	Definicja	Etap występowania	Poziom częstotliwości występowania	Koszt wystąpienia
Błąd składniowy	Nieprawidłowość w składni kodu, która nie pozwala na jego kompilację	Etap programowania	Wysoki	5 x
Błąd semantyczny	Błąd znaczeniowy, który nie musi występować podczas kompilacji, powoduje nieprawidłowe funkcjonowanie oprogramowania	Etap programowania, testowania, bardzo rzadko przez użytkowników	Wysoki	10 x
Błąd logiczny	Błędy pozwalające na kompilację kodu, trudne do wykrycia. Nie powodują zatrzymania pracy systemu, jednak sprawiają, że działa on niezgodnie z oczekiwaniami	Etap testowania lub przez użytkowników	Umiarkowany	15 x
Błąd w wyniku awarii	Błędy występujące po niedostępności systemu, np. błędy zasilania baz danych, błędy sekwencji przetwarzania danych	Wykrywane przez administratorów systemów i użytkowników	Umiarkowany do niskiego	30 x

Źródło: opracowanie własne na podstawie [Koomen, Pol 1999].

Gdy błąd zostanie wykryty, konieczne jest jego prawidłowe zgłoszenie, które nie może się odbyć bez właściwej identyfikacji i kategoryzacji. Zgłoszenie błędu powinno zawierać informacje pozwalające osobie, która zajmuje się obsługą zgłoszeń, na jego jednoznaczne zrozumienie. Tabela 2 prezentuje dane niezbędne oraz optymalne dla pełnego zdefiniowania oraz opisanego zgłaszanego błędu.

¹ Więcej na ten temat: <https://www.history.navy.mil/our-collections/photography/numerical-list-of-images/nhhc-series/nh-series/NH-96000/NH-96566-KN.html>, 2017.

² Definicje błędów oprogramowania można znaleźć w źródłach: [Kamińska-Kępa, Kępa 1992; Jaskiewicz 1997; [https://pl.wikipedia.org/wiki/B%C5%82%C4%85d_\(informatyka\)](https://pl.wikipedia.org/wiki/B%C5%82%C4%85d_(informatyka)) (23.07.2018); OGC 2012; Zmitrowicz 2015; Koomen, Pol 1999].

Tabela 2. Informacje niezbędne w zgłoszeniu błędu

Informacja	Opis	Konieczność uzupełnienia
Tytuł zgłoszenia	Pozwalający na identyfikację systemu oraz zawierający wstępne nakierowanie na problem	Niezbędne
Środowisko, w którym wykryto błąd	Na przykład środowisko testowe, preprodukcyjne, produkcyjne	Niezbędne
Wersja systemu	Wskazanie wersji systemu we właściwej nomenklaturze	Przydatne
Data wystąpienia błędu	W preferowanym formacie	Niezbędne
Godzina wystąpienia błędu	W preferowanym formacie	Przydatne
Dane użytkownika	Login lub imię i nazwisko	Niezbędne
Warunki wstępne	Uwarunkowania mogące mieć wpływ na działanie systemu	Przydatne
Kroki, jakie wykonano w systemie	Ścieżka dojścia do błędu	Przydatne
Opis błędu	Zwięzły, jednoznaczny opis nieprawidłowości w systemie	Niezbędne
Zrzut ekranu	W preferowanym formacie, najlepiej z datą i godziną wystąpienia	Przydatne
Opis poprawnego działania	Opis oczekiwanej reakcji systemu	Niezbędne

Źródło: opracowanie własne na podstawie [<http://testyoprogramowania.blogspot.com/2012/12/bad-defekt-niezgodnosc-incydent.html#!/2012/12/bad-defekt-niezgodnosc-incydent.html>; <http://nf.pl/po-pracy/10-bledow-popelnianych-przez-testerow,,15126,295>; <https://blog.goyello.com/pl/2017/07/13/dekalog-testera-czyli-co-musisz-umiec-aby-zostac-testerem-czesc-3/>].

Tabela 3. Kategorie błędów

Priorytet błędu	Opis	Optymalny termin naprawy
Błąd krytyczny	System lub jego część nie działa, nie można wykonać podstawowych funkcjonalności, np. nie można się zalogować do systemu	Natychmiast
Błąd wysoki	Działanie systemu nie jest zablokowane, lecz jedna z jego podstawowych funkcjonalności nie działa prawidłowo, np. błędnie księgowane są odsetki w systemie bankowym	Szybko
Błąd średni	System działa, jednak jedna z funkcjonalności niższej rangi nie działa prawidłowo, np. złe dane na raporcie	Jeśli znacznie nie utrudnia pracy użytkownika, to w kolejnej wersji systemu lub możliwie szybko w przypadku dużego wpływu na pracę użytkownika
Błąd niski	System działa prawidłowo, błąd ma niewielki wpływ na pracę użytkownika, np. „literówka”	W kolejnej wersji systemu

Źródło: opracowanie własne na podstawie [<http://testyoprogramowania.blogspot.com/2012/12/bad-defekt-niezgodnosc-incydent.html#!/2012/12/bad-defekt-niezgodnosc-incydent.html>; <http://nf.pl/po-pracy/10-bledow-popelnianych-przez-testerow,,15126,295>; <https://blog.goyello.com/pl/2017/07/13/dekalog-testera-czyli-co-musisz-umiec-aby-zostac-testerem-czesc-3/>].

Kategorie błędów, inaczej zwane priorytetami, określają wagę błędu ze względu na zasięg (wpływ) oraz pilność. W tabeli 3 zaprezentowano najczęściej występującą priorytetyzację błędów. Priorytet błędu wpływa bezpośrednio na wymagany czas reakcji zgłoszenia błędu oraz jego poprawy (szczególnie ważne dla systemów produkcyjnych, udostępnionych użytkownikowi), co zostało określone w ostatniej kolumnie tab. 3.

Nie ma wątpliwości, iż błąd to niezgodność, sytuacja szczególna i nieakceptowalna przez użytkownika, którą należy w odpowiedni sposób zgłosić w celu uzyskania naprawy. Podczas zgłaszania błędu powinno się stosować do wytycznych obowiązujących w danej organizacji, które należy opierać na najlepszych praktykach w IT, w celu usprawnienia cyklu obsługi błędów oprogramowania.

3. ITIL jako metoda obsługi błędów

Information Technology Infrastructure Library jest metodą opisującą najlepsze praktyki zarządzania usługami w IT, która koncentruje się na ustawicznym mierzeniu i poprawie jakości świadczonych usług. Właścicielem ITIL jest Cabinet Office (rząd Wielkiej Brytanii). W rozwój, wykorzystanie, a także propagowanie ITIL od początku istnienia zaangażowany jest IT Service Management Forum.

Geneza powstania ITIL

i. 1987 rok – początek projektu Government IT Infrastructure Management Methodology, ITIL v 1.

iii. 2001 rok – ITIL v 2.

iii. 2007 rok – ITIL v 3.

iv. 2011 rok – ITIL 2011 edition (aktualizacja ITIL).

ITIL może być wykorzystywany w organizacjach komercyjnych oraz publicznych bez względu na ich wielkość czy typ. Każdy proces w tym modelu ma zdefiniowane role i przypisany im zakres odpowiedzialności. Zawarte w ITIL kompleksowe rekomendacje dla branży informatycznej dały podwaliny międzynarodowej normie zarządzania usługami informatycznymi ISO/IEC 20000 Service Management³.

ITIL jest mocno skoncentrowany na usługach, a spojrzenie na nie daje obraz zamkniętego cyklu, gwarantującego ciągle doskonalenie. Biblioteka ITIL składa się obecnie z pięciu pozycji opisujących kolejne etapy cyklu życia usługi:

- strategia usług (*service strategy*),
- projektowanie usług (*service design*),
- przekazanie usług (*service transition*),
- eksploatacja usług (*service operation*),
- ustawiczne doskonalenie usług (*continual service improvement*).

Szesty podręcznik jest omówieniem pięciu powyższych pozycji oraz zawiera wstęp do zarządzania usługami IT jako całością.

³ Więcej na ten temat w [OGC 2012].

ITIL jest zestawem reguł stosowanych do zarządzania usługami IT (*IT service management*). Koncepcja ITIL powstała przy współpracy dostawców rozwiązań IT, ekspertów, konsultantów oraz użytkowników końcowych. ITIL jest popularny na całym świecie. Został opracowany przez OGC (Office of Government Commerce), czyli brytyjską organizację zajmującą się standardami. Obecnie nad rozwojem standardu OGC współpracuje z itSMF (IT Service Management Forum) oraz BSI (British Standards Institute). Biblioteka ITIL zawiera wspólny słownik pojęć oraz systematykę procesów IT i relacji między nimi. ITIL skupia się na jakościowym podejściu do świadczenia usług, co ma na celu osiągnięcie korzyści biznesowych. Koncepcja może być stosowana w organizacjach komercyjnych, które świadczą na zewnątrz usługi IT, a także w wewnętrznych działach IT. Szacuje się, że z ITIL korzysta obecnie ponad 15 000 przedsiębiorstwach z całego świata⁴.

Reasumując, należy stwierdzić, że ITIL jest ogólnie dostępnym modelem opisującym najlepsze praktyki zarządzania usługami informatycznym oraz nadzoru IT. Dostarcza schemat cyklu życia usług IT, koncentruje się na ciągłym mierzeniu oraz poprawie jakości świadczonych usług, mając na uwadze perspektywę zarówno klienta, jak i biznesu. Te wszystkie elementy przyczyniły się do sukcesu, jaki ITIL odniósł na całym świecie. ITIL jest powszechnie wykorzystywany i przynosi wiele korzyści organizacjom, które stosują procesy i techniki zebrane w bibliotece ITIL. Ponadto ITIL stał się podstawą dla międzynarodowej normy zarządzania usługami informatycznymi ISO/IEC 20000 Service Management.

4. Proces obsługi błędów według ITIL

Obsługa błędów według ITIL zarządzana i doskonalona jest procesowo. Według tej idei, proces to uporządkowany zbiór czynności, zaprojektowany, by spełniać określone cele. Każdy proces może mieć jedno lub wiele wejść do procesu, które zostają przekształcone w zdefiniowane wyjścia z procesu, czyli rezultaty. Proces może zawierać dowolne narzędzia, role, zakresy, a także elementy nadzoru, wymagane w celu osiągnięcia wiarygodnego rezultatu. Jeśli są one niezbędne, proces może definiować również normy, wytyczne, politykę, czynności czy instrukcje robocze. Kontrola procesu jest czynnością opierającą się na planowaniu i regulowaniu przebiegu procesu w taki sposób, aby proces przebiegał w sposób sprawny, skuteczny i konsekwentny.

Role w procesie to zakresy obowiązków i uprawnień, jakie ma osoba lub zespół. Rola może być definiowana w procesie lub funkcji, a jednak osoba (lub jeden zespół) może odgrywać wiele ról.

⁴ Więcej na ten temat: [<http://www.zaradzajonline.pl/publikacje/art5,zaradzanie-uslugami-it-itil.html>].

Tabela 4. Role w procesie według ITIL

Rola	Zakres odpowiedzialności
Właściciel procesu (<i>process owner</i>)	Osoba odpowiedzialna za zapewnienie zgodności procesu z jego przeznaczeniem. Zakres odpowiedzialności właściciela procesu: sponsorowanie, projektowanie, zarządzanie zmianami, ustawiczne doskonalenie procesu i jego miar
Menedżer procesu (<i>process manager</i>)	Rola odpowiedzialna za zarządzanie utrzymaniem procesu. Do jego obowiązków należy: planowanie i koordynacja czynności niezbędnych do przeprowadzenia, monitorowania i raportowania procesu
Praktyk procesu (<i>process practitioner</i>)	Rola odpowiedzialna za realizację jednego lub kilku czynności procesu
Funkcja (<i>function</i>)	Termin określający zespół wraz zasobami i narzędziami realizujący jeden lub kilka procesów. Innym znaczeniem dla terminu „funkcja” jest zamierzony cel osoby, zespołu, procesu. Funkcja może określać również prawidłową realizację zamierzonego celu

Źródło: opracowanie własne na podstawie [ITIL Service Operation 2016⁵].

Proces składa się z wielu elementów występujących po sobie. Tabela 5 przedstawia kolejne elementy procesu według ITIL w odniesieniu do obsługi błędów.

Handbook Handbook – Pocketbook from the Official Publisher of ITIL”, TSO, London].

Elementy procesu według ITIL, również procesu obsługi błędów oprogramowania, wzajemnie na siebie wpływają. Rysunek 1 obrazuje, jak na proces wpływają czynniki kontroli procesu oraz czynniki umożliwiające proces.

Na szczególną uwagę wśród elementów procesu zasługuje kontrola procesu (*process control*). To dzięki niej proces przebiega we właściwy sposób, czyli skutecznie, sprawnie i konsekwentnie.

Na charakterystykę procesu wpływają cztery wartości:

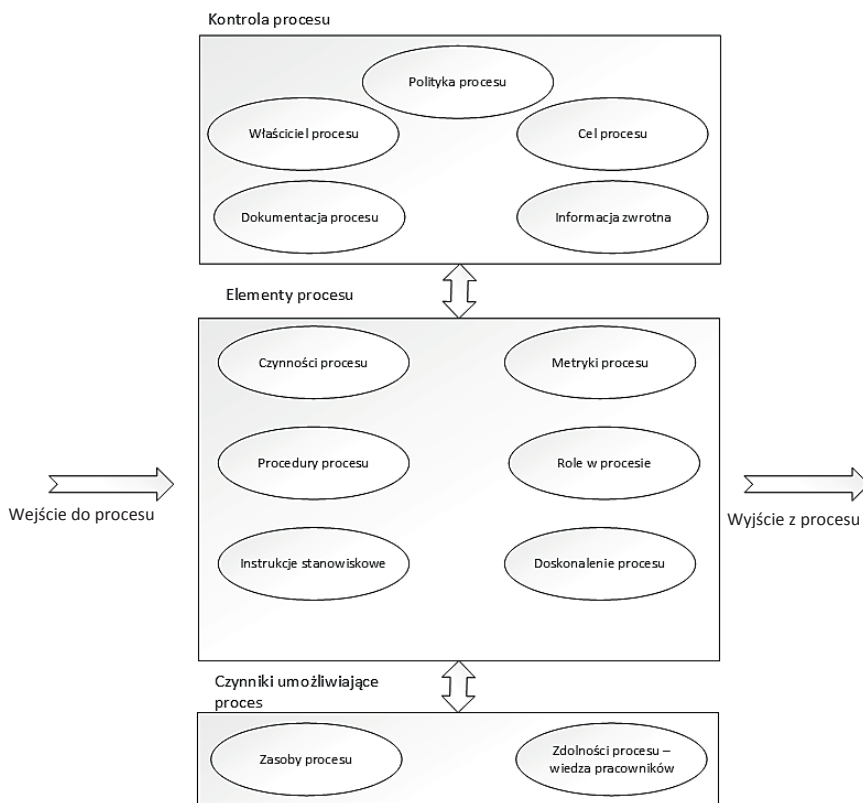
- Mierzalność (*measurability*) – kontrolowanie wydajności procesu pozwala na jego mierzenie. Mierzony jest koszt, jakość, czas trwania, a także produktywność.
- Zdefiniowane rezultaty (*specific results*) – rezultaty są powodem istnienia procesu. Rezultaty muszą być możliwe do zidentyfikowania i zmierzenia.
- Klienci (*customers*) – proces dostarcza rezultaty dla klienta. Klienci mogą być wewnątrzni lub zewnątrzni (pochodzący z wnętrza lub spoza organizacji), a proces musi spełniać ich oczekiwania. Procesy tworzą wartość dla interesariuszy.
- Odpowiadanie za zdefiniowane zdarzenia (*responsiveness to a specific triggers*) – proces musi reagować na określone wyzwalacze, niezależnie od tego, czy jest ciągły czy iteracyjny

⁵ Więcej na ten temat w TSO, „ITIL Service Operation,” TSO, London 2011.

Tabela 5. Elementy procesu obsługi błędów

Element procesu	Opis	Obsługa błędów
Wejście do procesu (<i>process inputs</i>)	Czynniki inicjujące proces przez wyzwalacze (<i>triggers</i>)	Trigerami dla procesu obsługi błędów są zgłoszenia błędów przez użytkowników systemu, administratorów lub systemy monitorujące pracę aplikacji
Czynności procesu (<i>process activities</i>)	Elementy procesu wykonywane w celu realizacji procesu	Podczas obsługi błędów nadawana i weryfikowana jest np. priorytetyzacja zgłoszeń, sprawdzana jest zasadność zgłoszenia, ich kompletność
Procedury (<i>process procedures</i>)	Zestawy czynności wykonywane przez określone role w procesie (<i>process roles</i>) i w zakresie metryk procesu (<i>proces metrics</i>)	Określonymi rolami w procesie obsługi błędów są zespoły obsługujące zgłoszenia błędów, np. zespół przyjmujący zgłoszenia (tzn. pierwsza linia wsparcia), zespół realizujący obsługę zgłoszeń (tzn. druga linia wsparcia), specjaliści wykonujący naprawę błędu – zespoły deweloperskie (tzn. trzecia linia wsparcia). Metrykami w procesie obsługi błędów są np. czas realizacji zgłoszenia w korelacji z priorytetami lub zasobami (<i>process resources</i>) przeznaczonymi na realizację procesu
Instrukcje stanowiskowe (<i>process work instructions</i>)	Zasady postępowania dla poszczególnych stanowisk biorących udział w procesie	Na przykład instrukcje dla zespołu przyjmującego zgłoszenia dotyczące sposobu formułowania tytułów zgłoszeń, uzupełniania informacji niezbędnych do przyjęcia zgłoszenia, nadawania właściwych priorytetów czy udzielania porad dla użytkowników systemu
Czynniki umożliwiające realizację procesu (<i>process enablers</i>)	Zasoby (<i>process resources</i>) i zdolności (<i>process capabilities</i>)	Zasoby w podziale na zasoby ludzkie oraz zasoby systemowe i sprzętowe. Zdolności w rozumieniu wiedzy pracowników biorących udział w procesie
Kontrola procesu (<i>process control</i>)	Kontrola procesu według ITIL odbywa się przez politykę procesu (<i>process policy</i>), na podstawie dokumentacji procesu (<i>process documentation</i>), celów (<i>process objectives</i>) i informacji zwrotnej (<i>process feedback</i>). Rolą biorącą udział w kontroli procesu jest właściciel procesu (<i>process owner</i>)	Procesy obsługi błędów według ITIL opierają na instrukcjach obsługi błędów, celem jest np. usprawnienie i skrócenie procesu, informację zwrotną zaś mogą stanowić raporty z systemów do rejestracji i obsługi błędów czy badania zadowolenia klientów zewnętrznych i wewnętrznych procesu
Wyjście z procesu (<i>process outputs</i>)	Następuje po wykonaniu procesu	Dla obsługi błędów oprogramowania wyjściem z procesu jest usunięcie problemu – błędu

Źródło: opracowanie własne na podstawie [OGC 2012; ITIL® Foundation]



Rys. 1. Wzajemny wpływ elementów procesu według ITIL

Źródło: opracowanie własne na podstawie [OGC 2012].

Badanie powyższych wartości wpływających na charakterystykę procesu obsługi błędów oprogramowania pozwala na efektywne doskonalenie tego procesu.

5. Zakończenie

Celem artykułu było zaproponowanie rozwiązania poprawiającego procedurę obsługi błędów oprogramowania. Szczególna uwaga poświęcona została jednej z metodyk, jaką jest ITIL. Wynika to przede wszystkim z bogatych doświadczeń w stosowaniu tej koncepcji przez jednego z autorów. Artykuł napisany został na podstawie materiałów literaturowych, źródeł internetowych, materiałów szkoleniowych, rozmów, wywiadów, certyfikacji ITIL Foundation oraz udziału w projekcie, dzięki czemu zdobyto wiedzę, która pozwoliła na przeprowadzenie procedury badawczej.

W opracowaniu została opisana tematyka błędów oprogramowania, mechanizmów doskonalenia procesów IT oraz metodologia ITIL w tym zakresie.

Tezą postawioną w artykule było założenie, że wykorzystanie koncepcji ITIL da podwaliny do poprawy procesu i przyniesie wymierne korzyści dla aktorów procesu⁶. Środowisko IT jest specyficzne, ulega ciągłym zmianom i dlatego niezbędne jest ustawiczne doskonalenie tego typu jednostek. Pojmowanie obsługi błędów produkcyjnych jako usługi dostarczanej dla klienta jest podstawą do jej ciągłego doskonalenie przez IT ku zadowoleniu klienta zewnętrznego i wewnętrznego.

Literatura

- ITIL Service Operation, 2017, TSO, London.
- Jaszkiwicz A., 1997, *Inżynieria oprogramowania*, Helion, Gliwice.
- Jong A., Kolthof A., Pieper M., Tjassing R., Veen A., Verheijen T., 2009, *ITIL® V3 Foundation Exam*, Van Haren, Zaltbommel.
- Kamińska-Kępa D., Kępa R., 1992, *Informatyzacja Banku Komercyjnego*, Biblioteka Menedżera i Bankowca, Warszawa,
- Kan S.H., 2006, *Metryki i modele w inżynierii jakości oprogramowania*, MIKOM, Warszawa.
- Koomen T., Pol M., 1999, *Test Process Improvement*, ACM, Oxford.
- Materiały szkoleniowe CT PARTNERS „ITIL FOUNDATION” v 3.1, 2015.
- Miller P., 2011, *Systemowe zarządzanie jakością. Koncepcja systemu, ocena systemu, wspomaganie decyzji*, DIFIN Spółka Akcyjna, Warszawa.
- OGC, 2012, *ITIL® Foundation Handbook – Pocketbook from the Official Publisher of ITIL*, TSO, London.
- Patton R., 2002, *Testowanie oprogramowania*, Mikom
- Roman A., Zmitrowicz K., 2018, *Testowanie oprogramowania w praktyce*, PWN, Warszawa.
- Sommerville I., 2003, *Inżynieria oprogramowania*, WNT, Warszawa,
- TSO, 2011a, *ITIL Service Design*, TSO, London,
- TSO, 2011b, *ITIL Service Operation*, TSO, London,
- TSO, 2011c, *ITIL Service Strategy*, TSO, London,
- TSO, 2011d, *ITIL Service Transition*, TSO, London,
- Zmitrowicz K., 2015, *Jakość projektów informatycznych. Rozwój i testowanie oprogramowania*, Helion, Gliwice.

Źródła internetowe

- <http://nf.pl/po-pracy/10-bledow-popelnianych-przez-testerow,,15126,295> (23.07.2018).
- <http://testyoprogramowania.blogspot.com/2012/12/bad-defekt-niezgodnosc-incydent.html#/2012/12/bad-defekt-niezgodnosc-incydent.html> (23.07.2018).
- <http://www.zaradzajonline.pl/publikacje/art5,zaradzanie-uslugami-it-til.html> (23.07.2018).
- <https://blog.goyello.com/pl/2017/07/13/dekalog-testera-czyli-co-musisz-umiec-aby-zostac-testerem-czesc-3/> (23.07.2018).
- <https://www.history.navy.mil/our-collections/photography/numerical-list-of-images/nhhc-series/nh-series/NH-96000/NH-96566-KN.html> (23.07.2018).

⁶ Więcej na ten temat w [TSO 2011b].