

Przybysz Filip
Kordowski Mateusz
Adrian Chlebosz
Tomasz Indeka

System wspomagania obsługi błędów systemów IT w Dziale IT.

Celem projektu jest stworzenie systemu wspomagającego pracowników działu IT w obsłudze problemów zgłaszanych za pośrednictwem helpdesku. Do helpdeska wprowadzane są zgłoszenia od klientów związane z problemami z obsługiwany przez nas oprogramowaniem. Klienci będą zgłaszać swoje problemy pracownikom działu obsługi klienta, którzy kolejno będą umieszczać zgłoszenia klientów w systemie helpdesk.

System pozwalał będzie pracownikom obsługi klienta wprowadzać do systemu przyjęte zgłoszenia i w razie konieczności oddelegowywać je do odpowiednich zespołów, specjalizujących się w rozwiązywaniu poszczególnych problemów.

Każde zgłoszenie zawierać będzie: unikatowy numer, datę zgłoszenia, aktualny status zgłoszenia, kategorię, priorytet, informacje o zgłaszającym kliencie, informacje o przyjmującym zgłoszenie pracowniku, szczegółowy opis problemu, identyfikator programu, którego dotyczy zgłoszenie, specyfikację środowiska klienta, podmiot odpowiedzialny za naprawę oraz listę wypróbowanych rozwiązań.

Możliwy będzie podgląd i edycja części informacji o każdym zgłoszeniu, wraz z zachowaniem historii wprowadzonych zmian. Pozwoli to prześledzić postęp prac nad każdym zgłoszeniem i podejmowane każdorazowo kroki. Za utrzymywanie dodatkowej tabeli zgłoszeń archiwalnych odpowiedzialny będzie mechanizm wyzwalaczy, który zachowa w niej kolejne stany każdego zgłoszenia po każdej jego modyfikacji.

Powstanie moduł analityczno-raportowy pozwalający na czytelną prezentację i filtrację danych statystycznych dotyczących przetwarzanych zgłoszeń.

Moduł analityczno - raportowy:

Zadaniem modułu analityczno-raportowego jest generowanie dokumentów w formacie PDF, które obejmować mogą wybrane przez tworzącą je osobę informacje z systemu. Możliwymi do wygenerowania raportami będą raporty zawierające:

- informacje o programach, co do których zostało zgłoszone najwięcej błędów, razem ze zgłaszającym i środowiskiem uruchomieniowym
- informacje o klientach najczęściej zgłaszających problemy, razem z reklamowanym programem oraz informację o środowisku uruchomieniowym
- informacje o aktualnie rozwiązywanych problemach: ich ilość, krótki opis, kto zgłasza, którego programu dotyczy problem
- informacje o wydajności pracowników - ile zgłoszonych do nich problemów zostało rozwiązanych w danych przedziałach czasowych oraz ile nadal jest rozwiązywane
- informacje o średnim czasie, który zajęło rozwiązanie problemów o danych priorytetach

Architektura rozwiązania:

- Do utworzenia koncepcyjnego modelu bazy danych wykorzystany zostanie program Oracle SQL Developer Data Modeler. Korzystać będziemy z zaprezentowanej na wykładzie notacji Barkera.
- Model fizyczny także zostanie opracowany za pomocą wcześniej wspomnianego programu.
- Łączyć się będziemy ze specjalnie utworzoną dla naszego projektu bazą danych Oracle.
- Aplikacja dostępowa dla bazy danych zostanie napisana w języku Java. W celu przyspieszenia procesu programowania użyjemy Java Persistence API w połączeniu z frameworkiem Hibernate, co pozwoli nam szybko i w prosty sposób połączyć się bazą danych.

Aktorzy:

- Konsultant - obsługuje system na poziomie przyjmowania zgłoszeń od klientów oraz odpowiedniego rozdysponowywania ich do podmiotów rozwiązujących zaistniałe problemy
- Serwisant - obsługuje system na poziomie rozwiązywania zgłoszeń skierowanych przez konsultantów

- Szef działu helpdesk - ma dostęp do generowania raportów z działania systemu
- Klient - nie ma bezpośredniej styczności z systemem, jego zgłoszenia kierowane są do pracownika, który to obsługuje system informatyczny

Założenia modelu logicznego:

- Nowe typy statusu, jakie mogą przyjmować krotki w tabelach Active Ticket Status oraz Archived Ticket Status będą mogły zostać dodane jedynie przez określoną procedurę, która zapewni, że nowy status znajdzie się zarówno w jednej jak i w drugiej tabeli.
- Tabele Active Ticket Status oraz Archived Ticket Status będą różniły się jedynie tym, iż w Archived Ticket Status będą dodatkowe typy statusu, takie jak solved, abandoned itp.
- ID w encji Ticket jest sztucznym kluczem głównym. Prawdziwą identyfikację zgłoszenia przejmie Ticket_Number. To pozwoli na trzymanie historii modyfikacji pojedynczej krotki z Tabeli Ticket.Active w tabeli Ticket.Archived.

Bezpieczeństwo:

- każdy aktor ma możliwość przeprowadzenia za pośrednictwem aplikacji dostępowej jedynie z góry określonych operacji,
- aby zapobiec zmianom wprowadzanym poprzez bezpośrednie połączenie z bazą danych albo przy złośliwym wykorzystaniu aplikacji do wykonywania niezamierzonych przez projektanta operacji, każdy użytkownik będzie łączył się z bazą danych poprzez user-a ze ściśle określoną rolą i minimalnym zbiorem pozwoleń. Dane tego user-a nie będą upubliczniane. (Nierealizowalne ze względu na wykorzystanie serwerów Politechniki, na których nie mamy uprawnień do tworzenia i zarządzania user-ami.)