

27.05.2020 r.

System wspomaganie obsługi błędów systemów IT w Dziale IT.

projekt BD2
semestr 20L

Adrian Chlebosz
Tomasz Indeka
Mateusz Kordowski
Filip Przybysz

Spis treści

Wstęp	3
Cechy systemu	4
Moduł analityczno - raportowy	4
Architektura rozwiązania	4
Aktorzy	4
Założenia modelu logicznego	5
Założenia niedotyczące samej bazy danych	5
Bezpieczeństwo	5
Model danych	6
Use case	7
Diagram use case	7
Tabela czynności	8
Analiza częstotliwości wybranych funkcji	9
Analiza tabel	9
Typowe zapytania bazodanowe	9
Zapytania dnia codziennego	9
Zapytania tworzące zestawienia danych	10
Scenariusze przypadków użycia	12
Standardowa procedura rozwiązywania błędów	12
Przeglądanie konta przez klienta	12
Przeglądanie konta przez pracowników	12
Serwisant rozwiązuje zgłoszony problem	13
Konsultant przyjmuje zlecenie	13
Zarządzanie i kontrola pracy działu	13
Nadawanie pracownikom praw	13
Aplikacja webowa	14
Model fizyczny	14
Indeksy	14
Powód założenia indeksów	15
Denormalizacja	15

1. Wstęp

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 obsługiwanym 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.

2. Cechy systemu

2.1. 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

2.2. 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.

2.3. 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 - zgłasza błędy związane z obsługiwany przez firmę oprogramowaniem za pośrednictwem internetowego formularza
 - Główny użytkownik - ma podstawowe uprawnienia zgłaszania i komunikacji
 - Szef klienta - rozszerza uprawnienia głównego użytkownika np. o dodatkowe poziomy priorytetu zgłoszenia

2.4. Założenia modelu logicznego

- Dodawanie i edycja statusów z Active_Status przebiegać będzie poprzez dostarczone procedury gwarantujące odpowiednie zmiany w Archived_Status
- 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 Archived_Ticket jest sztucznym kluczem głównym. Prawdziwą identyfikację zgłoszenia realizuje Ticket_Number. To pozwala na utrzymywanie historii modyfikacji pojedynczej krotki z Tabeli Active_Ticket w tabeli Archived_Ticket.

2.5. Założenia niedotyczące samej bazy danych

- Średnia liczba pracowników - 500
- Średnia liczba jednocześnie obsługiwanych klientów - 200
- Średnia liczba aktywnych zgłoszeń - 1000
- Maksymalna długość wiadomości - 2000 znaków
- Średnia ilość wiadomości odnośnie danego problemu - 5

2.6. Bezpieczeństwo

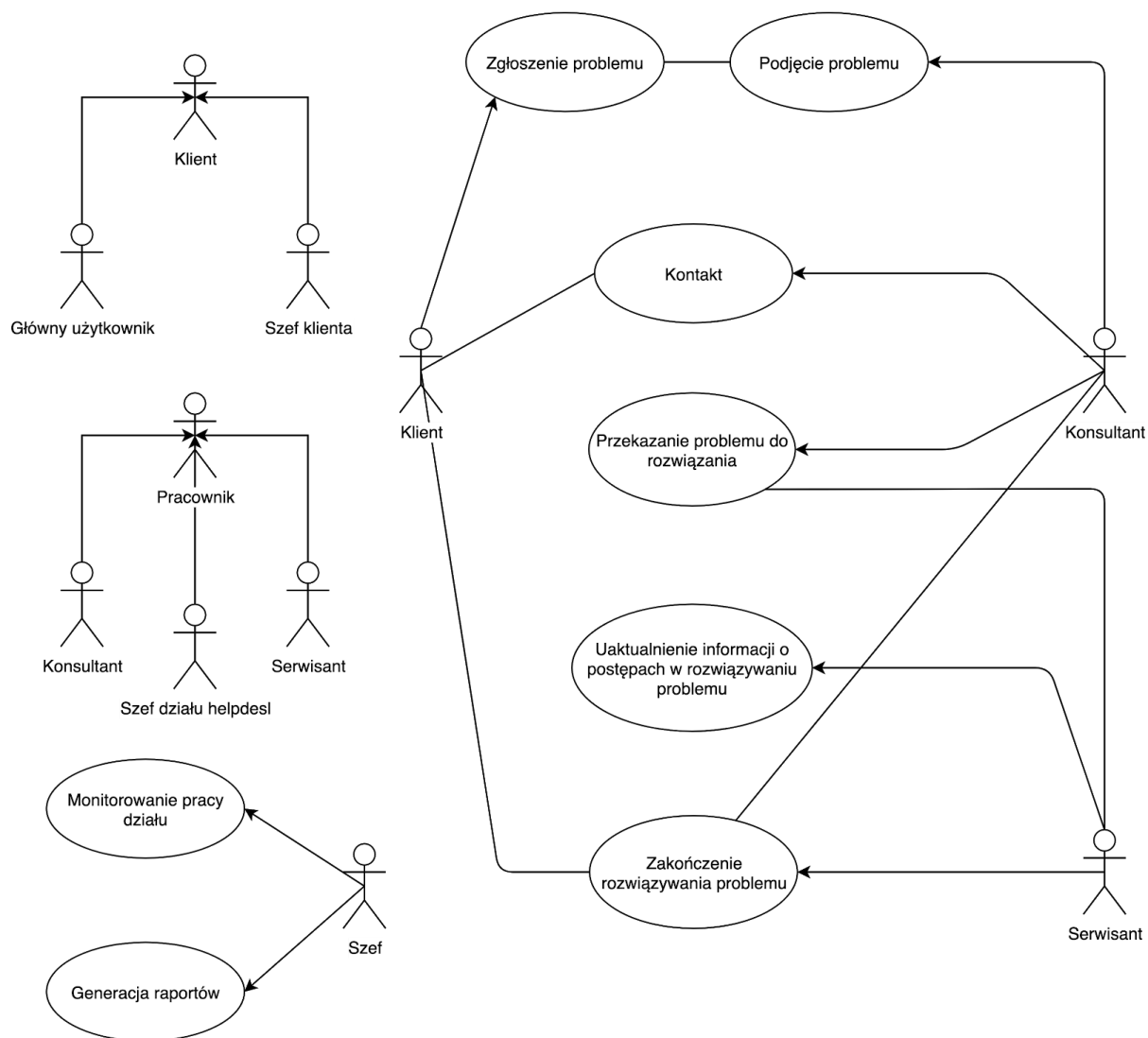
- każdy aktor ma możliwość przeprowadzenia za pośrednictwem aplikacji dostępowej jedynie z góry określonych operacji,
- każdy użytkownik będzie łączył się z bazą danych logując się na konto ze ściśle określonym zbiorem uprawnień (nierealizowalne, serwery Politechniki nie dają uprawnień do tworzenia użytkowników).

2.7. Model danych

- Poza archiwum, tabele w bazie danych są w 3.5 postaci normalnej.
- Tabela "raw" służy do trzymania surowych danych tj. takich, którym jeszcze nie przydzielono poszczególnych danych (niepełne zgłoszenia).
- Tabela "Message" służy do śledzenia wiadomości pomiędzy dwoma osobami odnośnie danej sprawy. Wiadomości mogą układać się w ciąg, dzięki czemu będzie można prześledzić przebieg konwersacji.
 - Dodatkowo każda wiadomość ma możliwość przechowywania plików.
W zamyśle są to screeny problemu, dump files itp.
- Klienci są rozpoznawalni poprzez swoje konta założone na naszej platformie. Z każdym klientem identyfikujemy umowę SLA.

3. Use case

3.1. Diagram use case



3.2. Tabela czynności

	Raw Ticket	Active Ticket	Archived Ticket	Messages	Attachment	Clients	Contracts	Employees
Zgłoszenie problemu	I						R	
Podjęcie problemu	D	I						
Kontakt		M	I	I	I			
Przekazanie problemu do rozwiązania		M	I					
Uaktualnienie informacji o postępach w rozwiązywaniu problemów		M	I					
Zakończenie rozwiązywania problemu		D	I					
Generacja raportów		R	R			R	R	R
Wgląd w stan systemu		R	R				R	R

4. Analiza częstotliwości wybranych funkcji

4.1. Analiza tabel

- RawTicket - prognozowane wykorzystanie: średnio 21/h, w piku 60/h, jedynie operacje insert i delete
- ActiveTicket - prognozowane wykorzystanie: średnio 200/h, w piku 400/h, wszystkie dostępne typy operacji, najwięcej modify
- ArchivedTicket - prognozowane wykorzystanie: średnio 200/h, w piku 400/h, podczas każdej modyfikacji ActiveTicket kopia jest wrzucana do archiwum, operacje insert i read
- Messages - prognozowane wykorzystanie: średnio 52/h, w piku 150/h, odbieranie wiadomości w cyklu życia zgłoszenia, tylko insert
- Attachment - prognozowane wykorzystanie: średnio 40/h, w piku 120/h, dodawanie załączników w cyklu życia zgłoszenia, tylko insert
- Clients - prognozowane wykorzystanie: średnio 1/h, w piku 30/h, tylko read
- Contracts - prognozowane wykorzystanie: średnio 12/h, w piku 30/h, tylko read
- Employees - prognozowane wykorzystanie: średnio 1/h, w piku 30/h, tylko read

4.2. Typowe zapytania bazodanowe

4.2.1. Zapytania dnia codziennego

- wybranie naglących aktywnych zgłoszeń
 - Active_Tickets
 - Due_date
 - Modification_date
 - Submission_date
 - Priority
 - Employee_ID
- przydział sprawy
 - Raw_Tickets - operacja delete
 - Active_ticket - operacja insert
- kontakt z klientem
 - Employee
 - ID
 - Active_Tickets
 - Ticket_number
 - Client_ID
 - Message - operacja insert
 - Attachment - możliwa operacja insert
 - Clients
 - Email
 - Contracts
 - Client_ID

- SLA
 - End_Date
- SLA
 - Name
 - Response_Time
- przesłanie starej wersji do archiwum
 - Active_Tickets
 - Modification_Date
 - modyfikacja innych pól ticketu
 - Archived_Tickets - operacja insert
- wyświetlanie wszystkich aktywnych spraw pracownika
 - Employee
 - ID
 - Position
 - Team
 - Active_Tickets
 - Submission_date
 - Due_date
 - Issue_Type
 - Software
 - Environment
 - Priority

4.2.2. Zapytania tworzące zestawienia danych

- sprawdzenie jakie problemy były najczęściej uaktualniane
 - Archived_Tickets
 - Modification_Date
- sprawdzenie średnich czasów obsługi zgłoszeń w zależności od priorytetu
 - Archived_Tickets
 - Ticket_Status
 - Submission_date
 - Due_date
 - Priority
 - Contracts
 - SLA
 - SLA
 - Response_Time
- sprawdzenie wydajności pracowników
 - Employee
 - First_Name
 - Last_Name
 - Position
 - Team
 - Archived_Tickets
 - Ticket_Status

- Due_date
- Priority

5. Scenariusze przypadków użycia

5.1. Standardowa procedura rozwiązywania błędów

Aktorzy: klient, konsultant

- Klient wchodzi na stronę do zgłaszania problemów
- Klient loguje się na stronie na swoje konto
- Klient zgłasza problem za pośrednictwem formularza
- Formularz jest zapisywany w systemie obsługi zgłoszeń i zostaje przydzielony do konsultanta
- Konsultant w razie potrzeby kontaktuje się z klientem w celu uzupełnienia potrzebnych informacji
- Zgłoszenie jest przetwarzane i rozwiązywane przez serwis
- Klient jest informowany o znalezionym rozwiązaniu za pośrednictwem maila

5.2. Przeglądanie konta przez klienta

Aktor: klient

- Klient wchodzi na stronę do zgłaszania problemów
- Klient loguje się na stronie na swoje konto
- Klient ma możliwość przejrzeć zgłoszone problemy, klient na stanowisku szefa ma możliwość przejrzania problemów zgłoszonych przez jego pracowników, każdy zgłoszony problem może być edytowany.

5.3. Przeglądanie konta przez pracowników

Aktor: pracownik (zarówno konsultant jak i serwisant)

- Pracownik loguje się do systemu obsługi zgłoszeń na swoje konto
- System wyświetla stan konta, listę wszystkich otwartych zgłoszeń i ich czas terminowego zakończenia przypisane do pracownika
- Pracownik może przeglądać zakończone zgłoszenia jak również przystąpić do pracy z otwartymi zgłoszeniami

5.4. Serwisant rozwiązuje zgłoszony problem

Aktor: serwisant

- Serwisant loguje się do systemu obsługi zgłoszeń na swoje konto
- Serwisant odbiera oczekujące zgłoszenie przekazane mu przez konsultanta
- Serwisant rozwiązuje problem każdorazowo uzupełniając informację o podjętych działaniach w zgłoszeniu
- Serwisant po znalezieniu rozwiązania oznacza zgłoszenie jako rozwiązane

5.5. Konsultant przyjmuje zlecenie

Aktor: konsultant

- Konsultant loguje się do systemu obsługi zgłoszeń na swoje konto
- Konsultant odbiera zgłoszenie, czekające w systemie na obsługę
- Konsultant sprawdza poprawność zgłoszenia i w razie potrzeby kontaktuje się z klientem, proponuje proste metody rozwiązania
- Konsultant uzupełnia w zgłoszeniu informacje otrzymane w wywiadzie z klientem i przekazuje zgłoszenie odpowiedniemu zespołowi serwisowemu

5.6. Zarządzanie i kontrola pracy działu

Aktor: szef działu Obsługi Klienta

- Szef działu chce skontrolować wydajność działu i wszystkich pracowników
- Szef działu loguje się na swoje konto w systemie obsługi zgłoszeń
- Szef ma możliwość wygenerowania raportów o wykonanych zgłoszeniach i stanie działu

5.7. Nadawanie pracownikom praw

Aktor: szef działu Obsługi Klienta

- Szef działu loguje się na swoje konto w systemie obsługi zgłoszeń
- Szef ma możliwość nadawać i odbierać pracownikom prawa dostępu, dodawać i usuwać pracowników, a także modyfikować ich dane

6. Aplikacja webowa

1. Technologia:
 - a. Warstwa serwerowa: Java (z wykorzystaniem frameworku Spring)
 - b. Warstwa interfejsu użytkownika: Angular z wykorzystaniem Angular Material UI
2. Funkcjonalności:
 - a. Logowanie
 - b. Przeglądanie zgłoszeń w formie tabelarycznej z dodaną paginacją dla komfortu przeglądania:
 - i. tylko swoich dla zwykłych pracowników
 - ii. zarówno swoich jak i podwładnych dla szefów
 - c. Możliwość wyboru ilości zgłoszeń na pojedynczej stronie
 - d. Możliwość rozwinięcia danego zgłoszenia w celu zobaczenia opisu
 - e. Dodanie nowego zgłoszenia
 - f. Edycja istniejącego zgłoszenia
 - g. Walidacja polegająca na sprawdzeniu, czy wszystkie pola zostały wypełnione przez zgłaszającego lub edytującego
 - h. Wylogowanie
3. Bezpieczeństwo:
 - a. W aplikacji serwerowej została włączona obsługa przeciwdziałania atakom typu CSRF

7. Model fizyczny

7.1. Indeksy

- Tabela "Active_Tickets" oraz "Archived_Tickets"
 - b-drzewo na Due date - często dodawane i odczytywane
 - b-drzewo na Submission date - często dodawane i odczytywane
 - b-drzewo na Modification date - często dodawane i odczytywane
 - b-drzewo na Ticket Number - często odczytywane
 - bitmapa na Status
 - bitmapa na Priority
 - bitmapa na Software
 - bitmapa na Issue Type
 - bitmapa na Environment
 - b-drzewo na Employee ID
 - b-drzewo na Client ID
- Tabela "Raw_Ticket"
 - b-drzewo na Ticket Number - często odczytywane
 - b-drzewo na Submission date - często dodawane i odczytywane
 - b-drzewo na Due date - często dodawane i odczytywane

- bitmapa na Priority
- bitmapa na Software
- bitmapa na Issue Type
- bitmapa na Environment
- b-drzewo na Client ID
- Tabela "Employee"
 - b-drzewo na email - często odczytywane, rzadko dodawane lub zmieniane
 - bitmapa na Position
 - bitmapa na Team
- Tabela "Message"
 - b-drzewo na date - przyspieszy szukanie w dużym zbiorze danych.
 - b-drzewo na Previous_Message
 - b-drzewo na Ticket_Number
- Tabela "Clients"
 - b-drzewo na email - często odczytywane i dodawane, rzadko zmieniane
 - b-drzewo na last name - często odczytywane.
- Tabela "Contracts"
 - b-drzewo na end date - często odczytywane i dodawane, rzadko zmieniane.
 - bitmapa na SLA
 - b-drzewo na Client_ID
 - b-drzewo na Software
- Tabela "Attachments"
 - b-drzewo na Message_ID

7.2. Powód założenia indeksów

W dostępnej bazie danych istnieją tylko dwa użyteczne rodzaje indeksowania.

Bitmapa jest dobra dla pól, które mają powtarzające się wartości oraz gdy nie dodajemy często nowych krotek. Jest również polecana na klucze obce.

Indeksy typu b-drzewo zostały założone na polach, które mogą się często zmieniać lub nie było sensu używać na nich bitmapy.

7.3. Denormalizacja

Ze względu na dużą dynamikę bazy danych oraz przenoszenia danych między tabelami denormalizacja jest problematyczna. Większość aktywności wymaga aktualnych informacji o stanie tabel ale nie wykraczających poza już znajdujące się w nich pola.

Niewiele jest też operacji wymagających łączenia ze sobą tabel. Najważniejsze informacje z innych tabel, szczególnie słownikowych, zawierają się bezpośrednio w kluczach obcych więc nie wymagają łączenia by się do nich odwołać.

Długie łańcuchy połączeń nie są wykorzystywane w żadnych operacjach.

Niemniej jednak istnieje niewielkie pole do poprawy:

- Powtórzenie wartości pola Response_Time tabeli SLA wewnątrz Contracts pozwoli uniknąć każdorazowego łączenia tabel przy wyliczaniu wartości pola Due_Date nowych zgłoszeń.
- Stworzenie materialized view dla prostych wykresów obrazujących stan systemu powinno zmniejszyć obciążenie odpowiednich tabel. Wgląd taki dokonywany będzie często i nie wszystkie jego aspekty muszą być bieżące. Wystarczy codzienna aktualizacja.
 - Ilość zgłoszeń otwartych, podjętych i zakończonych w kolejnych godzinach.
 - Ilość wiadomości wymienianych z klientem w związku z różnymi problemami dotyczącymi różnych programów w kolejnych godzinach.
- Również w generowaniu raportów przydadzą się perspektywy zmaterializowane. Pozwolą one na wykonanie bardziej skomplikowanych zapytań do bazy poza godzinami pracy i wielokrotne odwoływanie się do ich wyników bez dalszego obciążania istotnych tabel. Raporty generowane są stosunkowo rzadko i z założenia nie zawierają najbardziej aktualnych danych.
 - Ilość nowych zgłoszeń dla każdego z obsługiwanych programów, które nadeszły w kolejnych okresach raportowych.
 - Ilość zgłoszeń zamkniętych (rozwiązanych lub porzuconych) w kolejnych okresach raportowych.
 - Średni czas rozwiązania problemów o danym priorytecie w kolejnych okresach raportowych.