Politechnika Krakowska

Wydział Inżynierii Elektrycznej i Komputerowej

Dokumentacja

**Systemy Baz Danych**

**Projekt: Serwis Komputerowy**

Paweł Wądolny

Marcin Zieliński

Spis treści

[1. Cel i zakres projektu: 3](#_Toc503013579)

[2. Charakterystyka użytkowników 3](#_Toc503013580)

[3. Główne funkcje produktu 3](#_Toc503013581)

[4. Wymagania Funkcjonalne 3](#_Toc503013582)

[5. Wymagania Niefunkcjonalne 3](#_Toc503013583)

[6. Architektura i technologie 3](#_Toc503013584)

[7. Instalacja Oracle Database 3](#_Toc503013585)

[8. Konfiguracja wysyłania emaili z localhost 4](#_Toc503013586)

[9. Konfiguracja reCaptcha 5](#_Toc503013587)

[10. Główne funkcje strony internetowej 5](#_Toc503013588)

[10.1 Strona startowa 5](#_Toc503013589)

[10.1.1 Przypomnienie hasła 5](#_Toc503013590)

[10.1.2 Rejestracja 5](#_Toc503013591)

[10.2 Strona serwis 5](#_Toc503013592)

[10.2.1 Serwis – dodawanie zgłoszenia 5](#_Toc503013593)

[10.2.2 Serwis – podgląd zgłoszeń 5](#_Toc503013594)

[10.2.3 Serwis – szczegóły zgłoszenia 5](#_Toc503013595)

[10.2.4 Edycja danych osobowych – zmiana hasła 5](#_Toc503013596)

[11. Główne funkcje aplikacji serwisu 6](#_Toc503013597)

[11.1 Tytuły główne styl Nagłówek 2 6](#_Toc503013598)

[11.1.1 Tytuły podrzędne styl nagłówek 3 itd. 6](#_Toc503013599)

[12. Proces powstawania bazy danych 6](#_Toc503013600)

[13. Kod SQL tworzący struktury bazodanowe 6](#_Toc503013601)

[13.1 Tabela Klienci 6](#_Toc503013602)

[13.2 Tabela Pracownicy 6](#_Toc503013603)

[13.10 Indeksy 6](#_Toc503013604)

[14. Kod PL/SQL 6](#_Toc503013605)

[14.1 DODAJ\_UZYTKOWNIKA 6](#_Toc503013606)

[14.2 DODAJ\_PRACOWNIKA 6](#_Toc503013607)

[15. Przykładowe zapytania do bazy 6](#_Toc503013608)

[16. Plany wykonania bardziej złożonych, ciekawszych zapytań 6](#_Toc503013609)

[17. Wnioski 7](#_Toc503013610)

1. Cel i zakres projektu:

Celem projektu jest stworzenie strony internetowej dla serwisu komputerowego, oraz aplikacji na komputer stacjonarny dla administratora i pracowników serwisu. Aplikacja będzie działała w następujący sposób: Na serwerze umieszczamy bazę danych Oracle DataBase 12c. Strona internetowa serwisu jest umieszczona na serwerze z bazą danych, a aplikacja na komputer stacjonarny znajduje się na komputerach serwisu.

Zakres projektu obejmuje stworzenie strony i aplikacji z intuicyjnym GUI.

1. Charakterystyka użytkowników

* **Administrator** – jego zadaniem jest przydzielenie zadań serwisowych do danych pracowników.
* **Pracownik serwisu** – ma możliwość zmiany statusu zlecenia oraz potwierdzenie wykonania naprawy. Podczas diagnozy komputera wpisuje potrzebne czynności do bazy danych.
* **Użytkownik** – ma możliwość rejestracji i dodawania nowych zgłoszeń naprawy oraz możliwość podglądu szczegółów już istniejących zgłoszeń

1. Główne funkcje produktu
   1. Serwer bazy danych (Oracle DataBase 12c)
   2. Aplikacja serwisu
   3. Strona kliencka
2. Wymagania Funkcjonalne
3. Łączność strony internetowej z bazą danych
4. Łączność aplikacji serwisu z bazą danych
5. Dodawanie zleceń
6. Zmiana statusu zleceń
7. Informacja o płatności
8. Wymagania Niefunkcjonalne
9. Stabilność – aplikacja musi działać niezawodnie
10. Intuicyjność – aplikacja powinna być przyjazna dla użytkownika; niedoświadczony klient nie może mieć problemów z obsługą
11. Łatwość instalacji – system powinien być możliwy do wdrożenia w jeden dzień roboczy
12. Architektura i technologie
13. Do stworzenia strony internetowej zostanie użyty język php oraz html.
14. Bazą danych będzie Oracle DataBase 12c
15. Aplikacja serwisu zostanie wykonana w języku Java wersja 8.
16. Instalacja Oracle Database
17. Zainstalować Oracle Database 12c – przy instalacji SID bazy ustawić na: orcl

http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html

1. Pobrać serwer Apache np. XAMPP

<https://www.apachefriends.org/ro/download.html>

1. Pobrać InstantClient12\_2 (dla Oracle database 12c)

<http://www.oracle.com/technetwork/topics/winsoft-085727.html>

1. Po instalacji bazy i serwera apache wypakować instantclient np. na dysku C:/
2. W zmiennych środowkiskowych systemu dodać do parametru PATH (w zmiennych systemowych) ścieżkę do folderu z instantclient (np. C:/instantclient\_12\_2 )
3. W pliku php.ini (domyślnie C:\xampp\php\php.ini ) odkomentować (usunąć ; ) z linii extension=oci8\_12c
4. W bazie danych utworzyć użytkownika login: SERWIS\_KOMP pass: sbdserwis
5. Zaimportować bazę danych
6. Skopiować pliki ze stroną do folderu C:\xampp\htdocs\nazwa-strony (gdzie nazwa-strony to nazwa np. serwis lub serwis-komputerowy)
7. Uruchomić serwer Apache (w XAMPP)
8. W przypadku chęci zmiany użytkownika, hasła, należy zmienić dane logowania w pliku   
   setup-connect.php . Po zmianie danych, można uruchomić skrypt testujący z pliku basetest.php
9. W przeglądarce wpisać localhost/nazwa-strony (gdzie nazwa strony to nazwa folderu z plikami w folderze htdocs)
10. Konfiguracja wysyłania emaili z localhost

W celu działania funkcji „przypomnij hasło” korzystając z localhost niezbędna jest konfiguracja pakietu XAMPP:

1. W pliku C:/xampp/sendamail/sendmail.ini wpisujemy

(Jeżeli nie mamy zainstalowanego pakietu sendmail razem z xampp należy go ręcznie doinstalować do folderu C:/xampp/sendmail)

smtp\_server=smtp.gmail.com

smtp\_port=25

error\_logfile=error.log

debug\_logfile=debug.log

auth\_username=<username>

auth\_password=<password>

force\_sender=<e-mail username>@gmail.com

gdzie <username> to nazwa użytkownika poczty gmail, a <password> to hasło.

Jeżeli mamy podwójną autoryzację w poczcie (token) to musimy utworzyć „hasło do aplikacji”

<https://security.google.com/settings/security/apppasswords>

1. W pliku C:/xampp/php/php.ini należy odkomentować i edytować linie:

[mail function]

; For Win32 only.

SMTP = smtp.gmail.com

smtp\_port = 25

; For Win32 only.

sendmail\_from = <e-mail username>@gmail.com

sendmail\_path = "\"C:\xampp\sendmail\sendmail.exe\" -t"

1. Edytować plik setup-mail.php w folderze strony

Należy w nim wpisać mail, z którego będą wysyłane wiadomości.

1. Konfiguracja reCaptcha

W celu zapewnienia funkcjonalności ochrony przed botami, w formularzu rejestracyjnym został dodany mechanizm reCaptcha. Jest on domyślnie ustawiony na serwer localhost. Jeżeli chcemy wstawić stronę na serwer publiczny, należy wygenerować nowe klucze reCaptcha na stronie:

<https://www.google.com/recaptcha>

W zakładce „get reCaptcha”.

Następnie należy zmienić klucz publiczny i prywatny w pliku setup-recaptcha.php .

1. Główne funkcje strony internetowej

## 10.1 Strona startowa

Wygląd strony startowej – bla bla bla

### 10.1.1 Przypomnienie hasła

### 10.1.2 Rejestracja

## 10.2 Strona serwis

### 10.2.1 Serwis – dodawanie zgłoszenia

### 10.2.2 Serwis – podgląd zgłoszeń

### 10.2.3 Serwis – szczegóły zgłoszenia

### 10.2.4 Edycja danych osobowych – zmiana hasła

1. Główne funkcje aplikacji serwisu

## 11.1 Tytuły główne styl Nagłówek 2

Jakiś tekst - normalny

### 11.1.1 Tytuły podrzędne styl nagłówek 3 itd.

Jakiś tekst – normalny

Po napisaniu najechać na spis treści i kliknąć aktualizuj spis -> aktualizuj wszystko

1. Proces powstawania bazy danych

Blbablabal

tutaj jakiś schemat bazy, proces powstawania

1. Kod SQL tworzący struktury bazodanowe

## 13.1 Tabela Klienci

## 13.2 Tabela Pracownicy

Itd.

## 13.10 Indeksy

Tutaj coś o indeksach

1. Kod PL/SQL

## 14.1 DODAJ\_UZYTKOWNIKA

Tutaj opis procedury

## 14.2 DODAJ\_PRACOWNIKA

Tutaj opis procedury

Itd.

1. Przykładowe zapytania do bazy

Tutaj coś

1. Plany wykonania bardziej złożonych, ciekawszych zapytań

Jakiś screen z sqldevelopera do bazy

1. Wnioski

No i winioski jakieś na razie skopiowane:

Na etapie projektowania i testowania zostały wprowadzone dosyć znaczące zmiany mające wpływ na połączenia między encjami oraz ilość kolumn w różnych tabelach. Początkowa koncepcja okazała się nie być wystarczająca lub też jak w przypadku tabeli WORKER\_SERVICE zupełnie zbędna. Końcowa wersja bazy danych jest bardziej optymalna jeśli chodzi o szybkość wyszukiwania oraz relacje między encjami. Jest także w pełni sprawna w działaniu całego systemu ASO.

Na etapie implementacji pojawiło się kilka problemów związanych z nie spójnością typów danych, lecz łatwo rozwiązaliśmy je modyfikując typ danych dla danej kolumny.

Stwierdzić można że baza, wszelkie tabele jak również więzły integralności, oraz kolumny w encjach, w czasie projektowania, tworzenia bazy mogą a raczej przeważnie ulegają zmianom ponieważ podczas projektowania aplikacji która ma używać tej bazy, orientujemy się jakie funkcjonalności zapisu, odczytu modyfikacji danych w bazie są tak naprawdę istotne dla poprawnego, użytecznego oraz użytecznego dla użytkownika działania aplikacji, oraz dla programisty aby mógł zaprojektować oraz stworzyć w miarę możliwości najlepiej i możliwie bezawaryjnie działającą aplikacje