

# APLIKACJA WSPOMAGAJĄCA ZARZĄDZANIE ZLECENIAMI I WYCENĘ DETALI W FIRMIE

Izabela ROJEK, Grzegorz BŁAWAT

**Streszczenie:** W pracy przedstawiono aplikację wspomagającą zarządzanie zleceniami i wycenę detali w firmie. Utworzono ją zgodnie z zasadami inżynierii oprogramowania. Została ona napisana w języku C#, a bazę danych utworzono przy użyciu MySQL. Aplikacja w prosty sposób wspomaga pracownika, który odpowiada za zlecenia i wycenę detali w firmie. Została ona zaimplementowana na potrzeby rzeczywistej firmy i jest w niej stosowana od kilku miesięcy.

**Słowa kluczowe:** zarządzanie, zlecenie, wycena detalu, aplikacja

## 1. Wprowadzenie

Podstawą działalności firmy, dla której została utworzona aplikacja jest kompleksowa obróbka metalu, produkcja części i podzespołów oraz kompletnych urządzeń i maszyn dla różnych dziedzin przemysłu. Wykonywane są usługi z zakresu obróbki skrawaniem. Wykorzystywane są nowoczesne maszyny CNC oraz konwencjonalne obrabiarki. W procesie projektowania i przygotowania produkcji wykorzystywane jest oprogramowanie AUTOCAD. Firma specjalizuje się w produkcji maszyn rolniczych oraz konstrukcjach stalowych maszyn i urządzeń. Ciągły rozwój firmy oraz duży wzrost zamówień skłonił firmę do zainteresowania się aplikacją wspomagającą zarządzanie zleceniami i wycenę detali.

Na rynku znajduje się wiele systemów zarządzania zleceniami i wyceny detali, między innymi można wyróżnić Ascent SQL [1], czy system obsługi zleceń PL+ [2].

Ascent SQL to wielomodułowy, zintegrowany system informatyczny klasy ERP wspomagający zarządzanie przedsiębiorstwami małej i średniej wielkości. System ERP umożliwia śledzenie transakcji i procesów przedsiębiorstwa na każdym z etapów realizacji, pozwala generować szereg symulacji i zestawień ułatwiających podejmowanie decyzji. Umożliwia on rejestrację dokumentów ofertowych, które po akceptacji ze strony klienta są przekształcane w zlecenia. Po akceptacji klienta zlecenie przechodzi w fazę realizacji, można rejestrować koszty związane ze zleceniem oraz czas pracy pracowników lub podwykonawców poświęcony na realizację. Rejestracja czasu w połączeniu z kalkulacją stawek pozwalają kontrolować rzeczywiste koszty realizacji zleceń oraz wyliczają wypłaty dla pracowników i podwykonawców. Moduł pozwala w szybki sposób uzyskać informacje o zyskowności realizacji zleceń, czasie pracy, osobach zaangażowanych w realizację itp. [1].

System obsługi zleceń PL+ jest to oprogramowanie skierowane do firm usługowo - produkcyjnych wspomagające zarządzanie i obsługę zleceń na wykonanie usługi, produktu, czy też towaru. Ułatwia on kontrolę postępu prac nad sprawami zleconymi przez klientów, ale również usprawnia prowadzenie historii realizacji zleceń jak i kontrolę płatności. Program sprawdza się w wielu dziedzinach, między innymi warsztacie samochodowym, drukarni, firmach remontowo - budowlanych, czy firmach transportowych [2].

Analizując komercyjne systemy stwierdzono, że są one zbyt mocno rozbudowane i drogie jak na potrzeby wybranej firmy. Dlatego firma zdecydowała się na aplikację dedykowaną dla niej i jej potrzeb.

Aplikacja wspomagająca zarządzanie zleceniami i wycenę detali w firmie została zaprojektowana i zaimplementowana zgodnie z zasadami obowiązującymi w inżynierii oprogramowania [3]. Została ona wykonana przy użyciu języka programowania C# [4,5]. Wszystkie dane są zapisywane do bazy danych opartej na bazie danych MySQL [6].

Praca zawiera kilka rozdziałów. Pierwszy jest wprowadzeniem do tematu. Drugi zawiera opis zarządzania zleceniami i wyceny detali. Trzeci dotyczy projektu aplikacji. Czwarty rozdział omawia implementację i testowanie aplikacji. W piątym rozdziale zawarty jest opis działania aplikacji. Ostatni rozdział to podsumowanie i literatura.

## **2. Zarządzanie zleceniami i wycena detali**

Aplikacja wspomagająca zarządzania zleceniami i wycenę detali w firmie została zaprojektowana z myślą o wspomaganie pracowników zajmujących się organizacją pracy w wybranej firmie. Jednym z założeń aplikacji było wyszukanie informacji na temat zleceń, detali, klientów, cen materiałów i cen roboczo-minut obrabiarek znajdujących się w firmie. Kolejnym założeniem aplikacji była wycena detali. Dzięki przyjaznemu interfejsowi i wcześniejszemu wprowadzeniu danych na temat obrabiarek i półfabrykatów bardzo łatwo można wycenić detal. Informacje o wycenie detalu są zapisywane w bazie danych w celu ponownego użycia.

Wycena detali odbywa się w trzech etapach. Pierwszym etapem wyceny jest nadanie nazwy/numeru detalu. Drugi etap polega na wybraniu półfabrykatu oraz podaniu jego wagi. Trzecim etapem jest dodanie operacji technologicznych. W celu dołączenia operacji należy wybrać rodzaj operacji, później wytypować obrabiarkę i podać czas trwania operacji. Na końcu następuje podsumowanie wyceny dla detalu.

Po akceptacji wyceny detalu przez klienta można przejść do tworzenia zlecenia na wykonanie tego detalu. Tworzenie zlecenia również przebiega etapowo. Po pierwsze należy nadać nazwę zlecenia, następnie wybrać z listy klientów zleceniodawcę, a finalizacją jest dokonanie wyboru detali i podanie ilości zamawianych sztuk. Po utworzeniu zlecenia jest ono monitorowane przez aplikację. Szczególnie kontrolowane jest dotrzymanie terminu ukończenia zlecenia. Aplikacja pozwala również na tworzenie zlecenia dla wielu detali jednocześnie.

## **3. Projekt aplikacji**

Projekt aplikacji został omówiony w zakresie wymagań funkcjonalnych i нефункциональных aplikacji, projektu bazy danych w postaci diagramu związków encji i modelu relacyjnego oraz projektu interfejsu użytkownika w postaci diagramu modułów oraz diagramów przypadków użycia.

### **3.1. Wymagania funkcjonalne i нефункциональные aplikacji**

Wymagania funkcjonalne aplikacji są następujące:

- dodawanie, usuwanie, modyfikowanie danych związanych z ceną pracy obrabiarek i rodzajem wykonywanej operacji technologicznej,
- dodawanie, usuwanie, modyfikowanie danych związanych z materiałami

- potrzebny do produkcji,
- dodawanie, usuwanie, modyfikowanie danych klientów,
- tworzenie, usuwanie, modyfikowanie wycen detali,
- tworzenie, usuwanie, modyfikowanie, zmiana statusu zleceń,
- wyszukiwanie zleceń za pomocą dat wraz z przypominaniem o terminie zakończenia zlecenia, gdy do jego finalizacji pozostał tydzień lub mniej,
- sygnalizacja o upływie terminu zakończenia zlecenia.

Wymagania niefunkcjonalne aplikacji:

- aplikacja do poprawnego działania powinna być instalowana na systemie operacyjnym nie starszym niż Windows XP,
- aby łączność z bazą danych była odpowiednia, jej przepustowość nie powinna być niższa niż 1Mb/s,
- ekran monitora, na którym będzie wyświetlany interfejs aplikacji nie powinien być mniejszy niż 11 cali,
- wersja serwera bazodanowego nie powinna być starsza niż 5.6.17 - MySQL Community Server (GPL),
- wyświetlanie aplikacji powinno być poprawne na ekranach dużych monitorów.

### 3.2. Projekt bazy danych

Projekt bazy danych został wykonany na podstawie analizy zarządzania zleceniami i wyceną detali w wybranej firmie. Baza danych ma przechowywać dane o wszystkich zleceniach, detalach, materiałach, zleceniodawcach oraz obrabiarkach. Dotychczas w firmie jedna osoba zajmowała się organizacją pracy, ale baza danych została rozszerzona o tabele *uzytkownicy* i *uprawnienia* zawierające loginy i hasła. Będą one przydatne, gdy w firmie więcej osób zacznie się zajmować organizacją pracy. Większość tabel ma powiązania do tabeli *uzytkownicy*, dzięki czemu wiadomo, kto dokonuje zmian w danych. Baza danych użyta w aplikacji składa się z jedenastu tabel.

Pierwszym etapem tworzenia bazy danych było utworzenie diagramu związków encji. Diagram ten opisuje rzeczywistość w terminach obiektów (encji), atrybutów i związków. Obiekty są opisywane rzeczownikami w liczbie pojedynczej, atrybuty przymiotnikami i rzeczownikami, a związki czasownikami [7]. Następnie diagram związków encji został przekształcony na model relacyjny danych (rys. 1). Model relacyjny składa się z następujących tabel:

- tabeli *zlecenia* zawierającej klucz podstawowy o nazwie *idZlecenie*, który jest dziedziczony przez tabelę, zwaną dalej tabelą mieszającą *zlecenia\_detale*, pod nazwą *Zlecenie\_idZlecenie* jako klucz obcy. Ta tabela zawiera także klucze obce pochodzące z tabel *klienci*, dzięki temu powiązaniu wiadomo dla jakiego klienta jest utworzone zlecenie. Drugi klucz dziedziczony z tabeli *uzytkownicy*, informuje nas o twórcy zlecenia. Powiązania tabel z tabelą *uzytkownicy* umożliwia uzyskanie informacji kto jest odpowiedzialny za wprowadzenie danych. W przypadku nieprawidłowości można w prosty sposób zidentyfikować osobę odpowiedzialną,
- tabeli *detale* zawierającej klucz podstawowy o nazwie *idDetal* dziedziczony w tabeli mieszającej o nazwie *zlecenia\_detale* pod nazwą *Detal\_idDetal*,
- tabeli *materiały* zawierającej klucz podstawowy o nazwie *idMaterial* dziedziczony w tabeli o nazwie *detal* pod nazwą *Material\_idMaterial*,
- tabeli *detale\_operacje* zawierającej klucz podstawowy o nazwie *Id\_hash*. Klucz



### 3.3. Projekt interfejsu użytkownika

Interfejs w aplikacji jest bardzo ważnym elementem, gdyż służy do komunikacji między użytkownikiem, a komputerem. Aplikacja składa się z kilku modułów, w których użytkownik może wykonywać różnego rodzaju działania związane z zarządzaniem zleceniami i wyceną detali. Wszystkie moduły są bardzo łatwo dostępne przez naciśnięcie przycisku, na którym znajduje się ikona reprezentująca kategorię czynności, co sprawia że interfejs jest przyjazny i łatwy w obsłudze.

Na rys. 2 przedstawiono diagram modułów aplikacji. W celu przejścia do każdego z modułów należy przejść przez moduł logowania, w którym podawany jest login i hasło. Poprawna weryfikacja użytkownika pozwala przejść do modułów aplikacji. W przypadku użycia loginu ADMIN dostępny jest dodatkowy moduł *zarządzający użytkownikami*. Inni użytkownicy aplikacji nie mają do niego dostępu. Administrator aplikacji może dodać, nowego użytkownika, edytować dane użytkowników lub usunąć użytkownika.

Moduł *zarządzający zleceniami* zawiera listę zleceń, w której zawarte są informacje o zleceniach takie jak: nazwa, cena, status (aktywne, zakończone), data planowanego zakończenia, data przyjęcia, nazwa firmy zlecającej. Moduł ten też umożliwi dodanie, edytowanie, usunięcie lub zmianę statusu zlecenia. Dodatkowo można też przejrzeć listę zamówionych detali jaką zawiera określone zlecenie wraz z liczbą sztuk. Wystarczy zaznaczyć wiersz określonego zlecenia i nacisnąć przycisk *Lista detali*, a wtedy z lewej strony okna pojawi się panel z takimi właśnie informacjami.

Moduł *zarządzający detalami* zawiera listę wycenionych detali. Są tam takie informacje jak: nazwa, cena, czas wykonania, waga materiału, rodzaj materiału potrzebnego do produkcji oraz jaki użytkownik utworzył dany rekord danych. W module tym można wycenić nowy detal, edytować istniejącą wycenę lub ją usunąć.

Moduł *park maszyn* zawiera listę obrabiarek, które znajdują się w firmie i na których można wykonywać określone operacje technologiczne. W każdym wierszu można zobaczyć jak nazywa się obrabiarka, ile kosztuje minuta pracy, jaką operację może wykonywać i kto wypełnił te dane.

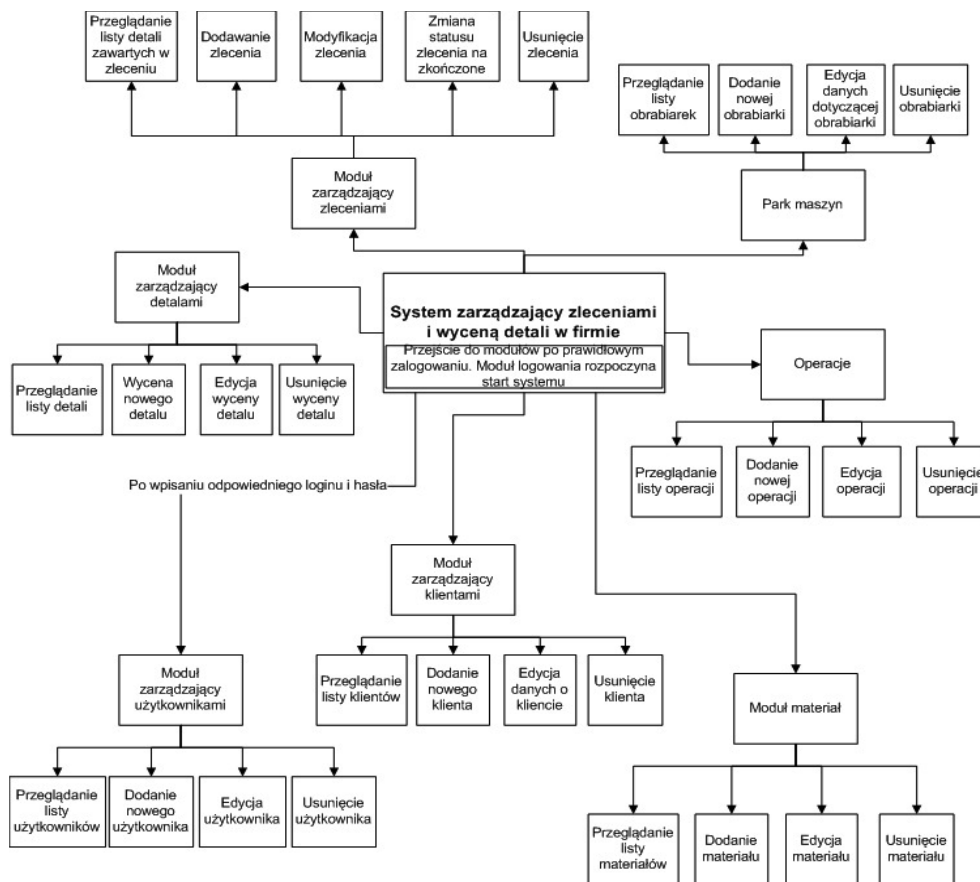
W module *operacje* ukazuje się lista wszystkich operacji technologicznych. Operacje można dodawać, usuwać lub edytować.

Moduł *materiał* zawiera listę materiałów, potrzebnych do produkcji detalu. W każdym wierszu można zobaczyć jak nazywa się materiał, ile kosztuje i kto wypełnił dane.

W module *zarządzającym klientami* znajduje się lista klientów, w której zawarte są informacje takie jak: nazwa firmy, adres, NIP, numer telefonu. W module można dodawać, usuwać lub edytować dane.

W projekcie aplikacji wykonano również szereg diagramów przypadków użycia, które posłużyły do bardziej szczegółowej specyfikacji aplikacji.

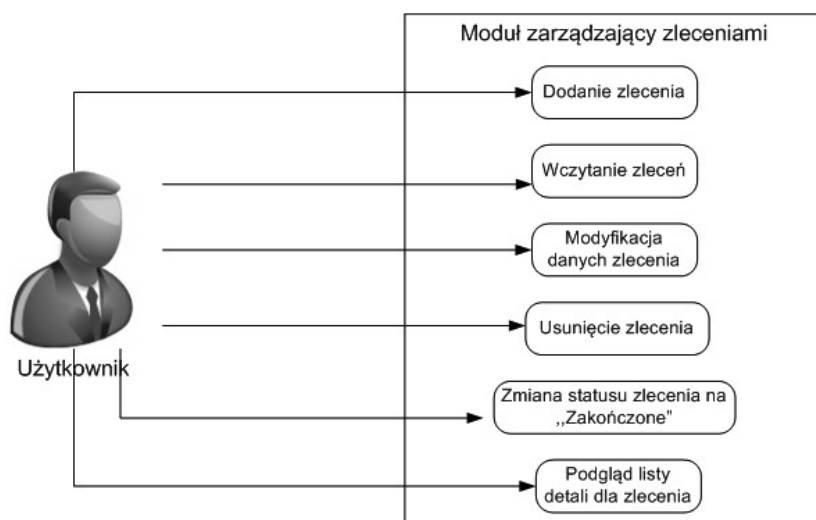
Przypadek użycia jest to specyfikacja ciągu akcji i ich wariantów, które aplikacja może wykonać poprzez interakcję z aktorami (np. użytkownikami) takiej aplikacji. W związku z tym przypadek użycia jest kompleksowym działaniem realizowanym w projektowanej aplikacji w konsekwencji określonej aktywności aktora. Zakres danego działania determinowany jest przez wszystkie wzajemnie powiązane ze sobą przypadki użycia. Pojedynczy przypadek użycia to reprezentant spójnej jednostki funkcjonalności, którą dostarcza aplikacja [8]. Przykład diagramu przypadku użycia dla modułu *zarządzającego zleceniami* został pokazany na rys. 3.



Rys. 2. Diagram modułów aplikacji

#### 4. Implementacja i testowanie aplikacji

Implementacja aplikacji przebiegała w dwóch etapach. Etap pierwszy dotyczył tworzenia bazy danych na serwerze. Utworzonych zostało 11 tabel. Każda tabela została utworzona poprzez podanie jej nazwy oraz nazw kolumn, typów danych w kolumnach z maksymalną liczbą znaków oraz ograniczeniem, czy wartość pola może być pusta, czy też nie oraz ustawieniami domyślnymi. Tab. 1. pokazuje przykładową strukturę tabeli *zlecenia*.



Rys. 3. Diagram przypadków użycia dla modułu zarządzającego zleceniami

Tab. 1. Struktura tabeli *zlecenie*

Nazwa kolumny	Typ danych z max. długością pola	Ograniczenie na pole, czy może być wartość Null	Ustawienia domyślne ograniczenia pola
idZlecenie	int(11)	Nie	
Nazwa	varchar(45)	Tak	NULL
Cena	double	Tak	NULL
Status	varchar(45)	Tak	NULL
Data_zakonczenia	date	Tak	NULL
Data_przyjecia	date	Tak	NULL
Klient_idKlient	int(11)	Nie	
Uzytkownicy_idUzytkownicy	int(11)	Nie	

Etap drugi dotyczył implementacji interfejsu aplikacji. Przykładowy kod źródłowy, który został pokazany poniżej umożliwia dodawanie lub edycję obrabiarek:

```
private void btnDodaj_Click(object sender, EventArgs e)
{
    zarzadzaj.slKlienci(Polaczenie.select_obrabiarka, -1,0);
    OBRABIARKA = zarzadzaj.dt;
    frmDodajEdytuj dodaj = new
    frmDodajEdytuj(((Button)sender).Tag.ToString(), OBRABIARKA, "obrabiarka");
    DialogResult dr = dodaj.ShowDialog();
    if (dr == DialogResult.OK)
    {
        wczytajListe(0);
    }
}
```

```

    }
}
private void btnEdytuj_Click(object sender, EventArgs e)
{
    w = int.Parse(listViewObrabiarki.Items[wyb].Name);
    if (w > 0)
    {
        zarzadzaj.sIKlienci(Polaczenie.select_obrabiarka, w,0);
        OBRABIARKA = zarzadzaj.dt;
        frmDodajEdytuj dodaj = new
        frmDodajEdytuj(((Button)sender).Tag.ToString(), OBRABIARKA,"obrabiarka");
        DialogResult dr = dodaj.ShowDialog();
        if (dr == DialogResult.OK)
        {
            wczytajListe(0);
        }
    }
}
}

```

Kod źródłowy przedstawiony powyżej obsługuje zdarzenie kliknięcia na przycisk *Dodaj* lub *Edytuj*. Zdarzenie to jest uruchamiane z okna, w którym wyświetlana jest lista obrabiarek. Po zdarzeniu kliknięcia na przycisk *Dodaj* uruchamiana jest opracowana procedura czytająca pusty rekord tabeli *obrabiarki* z bazy danych, w przypadku *Edytuj* modyfikowana jest wybrana *obrabiarka*.

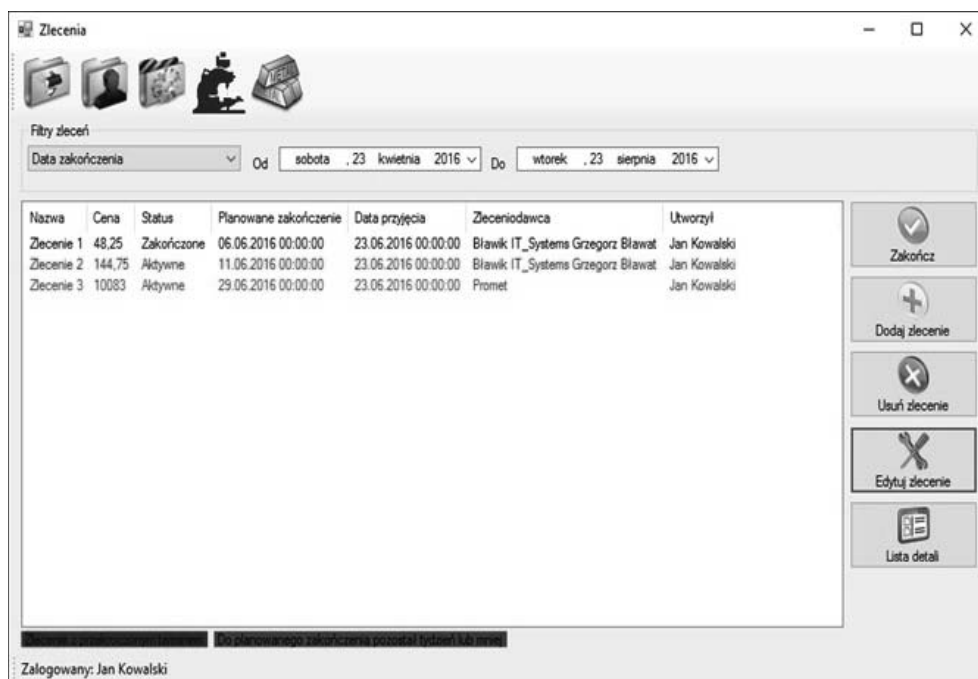
Aplikację poddano wybranym testom jednostkowym, integracyjnym oraz akceptacyjnym. Testom jednostkowym poddane zostały wszystkie funkcje modułów. Testy integracyjne pozwoliły sprawdzić powiązania między modułami. Natomiast testy akceptacyjne posłużyły do sprawdzenia, czy wszystkie funkcjonalności aplikacji działają poprawnie.

## 5. Opis działania aplikacji

Po uruchomieniu aplikacji następuje otwarcie okna logowania. Uprawnienia użytkownika określają dostęp do funkcjonalności aplikacji. Niektóre funkcjonalności aplikacji dostępne są tylko administratorowi, w tym np. zarządzanie użytkownikami (dodawanie, edycja i usuwanie). Tworzenie nowego użytkownika polega na wpisaniu jego nazwy, loginu, hasła oraz uprawnień. Edycja odbywa się podobnie, różni się tym że trzeba wybrać z listy odpowiedni wiersz i nacisnąć *Edytuj użytkownika* a w tym samym oknie będą już dane do edycji. Chcąc usunąć użytkownika należy wybrać go z listy i kliknąć *Usuń użytkownika*.


Po zalogowaniu się użytkownika innego niż administrator pojawi się okno *zarządzania zleceniami*. W oknie znajduje się lista zleceń. Wiersze, których czcionka jest czerwona to zlecenia z przekroczonym terminem wykonania, kolorem fioletowym oznaczone są te, których termin planowanego zakończenia upłynie w ciągu tygodnia (rys. 4).






Rys. 4. Okno modułu *Zarządzanie zleceniami*

W celu dodania zlecenia należy wcześniej utworzyć klienta, materiał, operację, obrabiarkę i wyceniony detal. Chcąc dodać klienta należy przejść do odpowiedniego

modułu klikając na drugi przycisk od lewej strony w górnym pasku .

Po wybraniu przycisku otwiera się okno, w którym można dodawać, edytować i usuwać klientów. Przyciski *Edytuj* i *Usuń* mogą być nieaktywne, jeśli nie zaznaczono żadnego klienta na liście. Gdy klient zostanie już wybrany można kliknąć w przyciski, które spowodują otwarcie okna *edycji* lub *usunięcia klienta*. Może się zdarzyć tak, że żadne przyciski nie będą aktywne, bo użytkownik ma ograniczone uprawnienia. Wtedy można tylko przeglądać dane.

W celu przeglądania listy, usunięcia lub dodania operacji technologicznej, w oknie gdzie znajduje się moduł zarządzania zleceniami (rys.4) należy nacisnąć trzeci przycisk od

lewej strony w górnym pasku . Po naciśnięciu na przycisk ukazuje się okno z listą operacji, w którym można dodać nową operację, edytować lub usunąć.

Kliknięcie na czwartą ikonę od lewej strony w górnym pasku (rys. 4) powoduje otwarcie okna *Parę maszyn*. Dodawanie obrabiarki przebiega dokładnie tak samo jak we wcześniej wymienionym module *zarządzania klientami* i module *operacje*. W oknie edycji i tworzenia nowej obrabiarki istnieje jedna różnica, polegająca na wybraniu z rozwijanej listy przynależnej do obrabiarki operacji technologicznych. Moduł o nazwie *Materiał* został tak samo zaprojektowany jak wcześniejsze trzy moduły.

Moduł *zarządzania wyceną detali* również rozpoczyna się listą, ale edycja i tworzenie

wyceny przebiega trochę inaczej niż w poprzednich przypadkach. Utworzenie wyceny i edycja wyceny przebiegają w dwóch krokach. W kroku pierwszym po wybraniu z listy i naciśnięciu *Edytuj* lub tylko naciśnięciu przycisku *Dodaj* otworzy się okno w którym trzeba uzupełnić lub edytować dane przedstawione na rys. 5. Do kroku drugiego przechodzimy naciskając *Zatwierdź*.

Rys. 5. Okno służące do edycji lub tworzenia wyceny detalu – krok 1

W kroku drugim z górnego paska wybieramy operację, po czym w oknie pojawi się nowa lista rozwijana z obrabiarkami przynależącymi do wybranej operacji, a obok niej trzy pola, gdzie trzeba podać czas operacji (rys. 6).

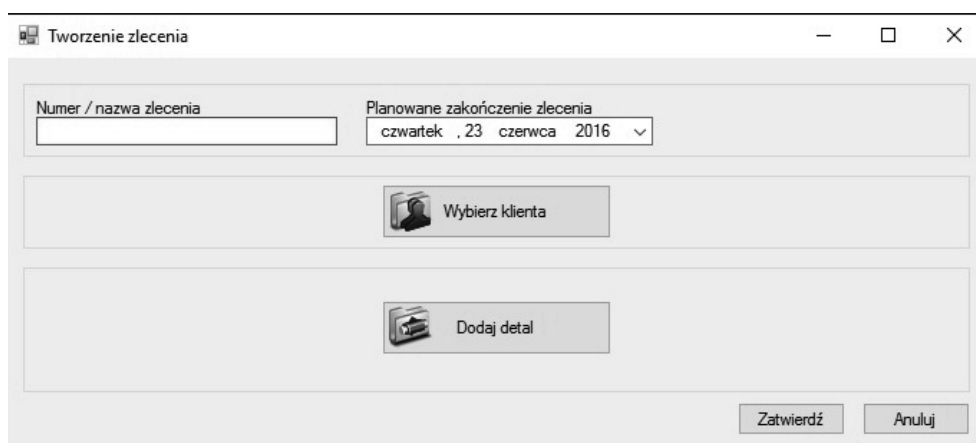
OPERACJE	GODZINY	MINUTY	SEKUNDY
Okuma LB10II	0	2	45
Fadal	0	0	18

Rys. 6. Okno służące do edycji lub tworzenia wyceny detalu – krok 2

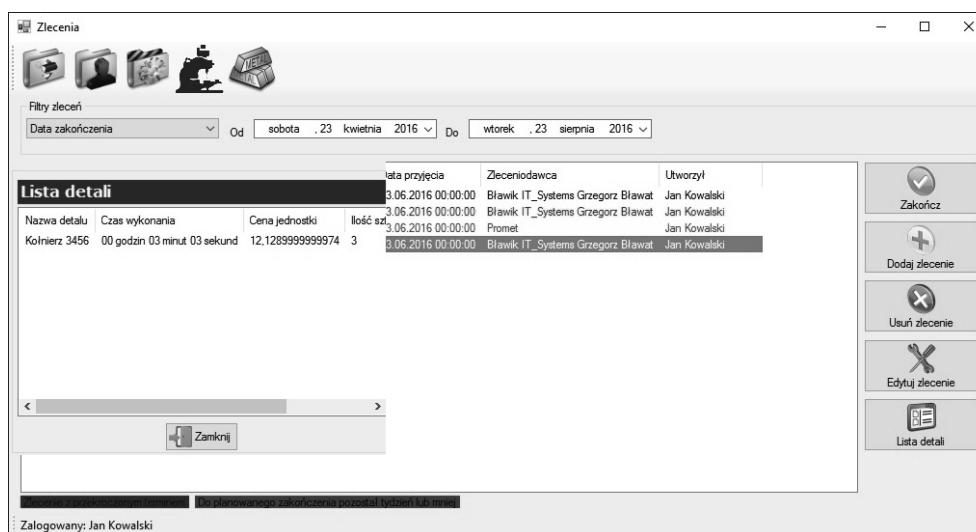
Po wykonaniu wyceny detalu można przejść do tworzenia zlecenia. W oknie pokazanym już wcześniej *Zlecenia* (rys. 4) należy kliknąć przycisk *Dodaj zlecenie*, wtedy otworzy się okno z rys. 7.

Należy podać nazwę / numer zlecenia, później wybrać planowaną datę zakończenia, a następnie klikając przycisk *Wybierz klienta* należy wybrać klienta z listy klientów, utworzonych wcześniej w module *zarządzania klientami*. Po zaznaczeniu na liście i kliknięciu *Dodaj* następuje dodanie klienta do zlecenia. Następnie naciskając przycisk *Dodaj detal*, należy wybrać z listy detal, w polu tekstowym *Liczba sztuk* wpisać zamówioną przez klienta liczbę detali i kliknąć *Dodaj*. Po zaakceptowaniu wszystkich

działań należy kliknąć *Zatwierdź*, wtedy pokazana zostanie lista zleceń. W celu podglądu listy detali w zleceniu, trzeba wybrać z listy odpowiedni wiersz i nacisnąć przycisk *Lista detali*, wtedy z lewej strony okna ukaze się panel, w którym można zobaczyć te dane o detalach (rys. 8).



Rys. 7. Okno tworzenia nowego zlecenia



Rys. 8. Okno modułu zarządzania zleceniami z otwartym panelem *Lista detali*

Edycja zlecenia przebiega podobnie jak dodawanie. W module *zarządzanie zleceniami* jest też funkcja usuwania zlecenia, lub zmieniania jego statusu na „Zakończone”, do czego służą przyciski z prawej strony okna.

## 6. Podsumowanie

W utworzonej aplikacji zostały zawarte wszystkie funkcje potrzebne do zarządzania

zleceniami i wyceną detali w firmie. Wszystkie założenia postawione w fazie projektu takie jak wymagania funkcjonalne i нефункционалне aplikacji zostały zrealizowane i w rezultacie aplikacja zarządza zleceniami i wycenia detale. Aplikacja ta została wdrożona w wybranej firmie i działa do dzisiaj, gdyż została dokładnie zaprojektowana pod potrzeby firmy.

W miarę rozwoju firmy zakłada się też rozwój aplikacji, co dzięki modułowej budowie nie powinno być trudne do zrealizowania.

## **Literatura**

1. <http://absolutt.pl/ascentssql.html>, dostęp 15.06.2016.
2. <http://www.softwareprojekt.com.pl/oprogramowanie/system-obslugi-zleceń>, dostęp 15.06.2016.
3. Sacha K.: Inżynieria oprogramowania, Wydawnictwo Naukowe PWN, Warszawa, 2010.
4. Griffiths I., Adams M., Liberty J.: Programowanie C# , Helion, Gliwice, 2012.
5. Matulewski J.: VisualStudio 2013 Podręcznik programowania w C# z zadaniami, Helion, Gliwice, 2014.
6. Nixon R.: PHP, MySQL i JavaScript. Wprowadzenie, Helion, Gliwice, 2015.
7. Elmasri R., Navathe S.B.: Database Systems, Models, Languages, Design and Application Programming, Pearson, 2011.
8. Wrycza S., Marcinkowski B., Wyrzykowski K.: Język UML 2.0 w modelowaniu systemów informatycznych, Helion, Gliwice, 2005.

Dr inż. Izabela ROJEK

Grzegorz Bławat, student IV roku informatyki  
Instytut Mechaniki i Informatyki Stosowanej  
Uniwersytet Kazimierza Wielkiego  
85-064 Bydgoszcz, ul. Chodkiewicza 30  
tel./fax: (0-52) 325 76 30/ (0-52) 325 76 44  
e-mail: izarojek@ukw.edu.pl