

U n i w e r s y t e t Z i e l o n o g ó r s k i

WYDZIAŁ ELEKTROTECHNIKI, INFORMATYKI I TELEKOMUNIKACJI

INSTYTUT STEROWANIA I SYSTEMÓW INFORMATYCZNYCH

KIERUNEK: *INFORMATYKA*

SPECJALNOŚĆ: *SIECIOWE SYSTEMY INFORMATYCZNE*

STUDENT (GRUPA): *BARTOSZ BŁASZCZAK (431 IZZ)*

PROMOTOR: *DR INŻ. ANDRZEJ MARCINIAK*

TEMAT PRACY: *SYSTEM MAGAZYNOWO-DYSTRYBUCYJNY DLA MAŁYCH
PRZEDSIĘBIORSTW*

wersja polska

wersja angielska

*WAREHOUSING AND DISTRIBUTING APPLICATION FOR
SMALL ENTERPRISES*

CEL PRACY:

Zaprojektowanie i implementacja systemu informatycznego do zarządzania stanem magazynowym i dystrybucji produktów w przedsiębiorstwie. Powstała aplikacja może być samodzielna, jak i stanowić część systemu ERP.

ZAKRES PRACY:

Analiza funkcjonalności aplikacji do zarządzania stanem magazynowym i dystrybucji produktów

Wykonanie projektu i implementacja aplikacji

Wykonanie testów funkcjonalnych

DATA PODJĘCIA TEMATU: 13.11.2013

PODPISY:

PROMOTOR

STUDENT

DATA PRZYJĘCIA PRACY

PODPIS PROMOTORA

Wyrażam zgodę na wykorzystanie do celów dydaktycznych i badawczych wyników zawartych w pracy dyplomowej i powstałych programów komputerowych.

DATA

PODPIS STUDENTA

UNIwersYTET ZIELONOGÓRSKI

Wydział Elektrotechniki, Informatyki i Telekomunikacji

PRACA DYPLOMOWA

***SYSTEM MAGAZYNOWO-DYSTRYBUCYJNY DLA
MAŁYCH PRZEDSIĘBIORSTW***

Bartosz Błaszczak

Promotor:

dr inż. Andrzej Marciniak

Zielona Góra, 30 VI 2015

Zielona Góra

.....
Imię i nazwisko studenta

Grupa

Wydział Elektrotechniki,

Informatyki i Telekomunikacji UZ

OŚWIADCZENIE

Świadomy odpowiedzialności karnej oświadczam, że przedkładana praca dyplomowa pt.

„SYSTEM MAGAZYNOWO-DYSTRYBUCYJNY DLA MAŁYCH PRZEDSIĘBIORSTW”
została napisana przeze mnie samodzielnie i nie była wcześniej podstawą żadnej innej urzędowej procedury związanej z nadaniem dyplomu wyższej uczelni lub tytułów zawodowych. Jednocześnie oświadczam, że w/w praca nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994r. o prawie autorskim i prawach pokrewnych innych osób (DZ. U. z roku 2000 Nr 80 poz. 904) oraz dóbr osobistych chronionych prawem cywilnym.

Oświadczam również, że egzemplarz pracy dyplomowej w formie wydruku komputerowego jest zgodny z egzemplarzem pracy dyplomowej w formie elektronicznej.

.....
podpis

[Imiona i nazwiska współautorów]

[Miejscowość, data]

Oświadczenie o udziale w realizacji tematu pracy dyplomowej

Temat pracy: SYSTEM MAGAZYNOWO-DYSTRYBUCYJNY DLA MAŁYCH PRZEDSIĘBIORSTW

Promotor: dr inż. Andrzej Marciniak

[Imię i nazwisko współautora]

[wykaz czynności wykonanych przez tę osobę]

podpis składającego oświadczenie

podpis przyjmującego oświadczenie

[imię i nazwisko współautora]

[wykaz czynności wykonanych przez tę osobę]

podpis składającego oświadczenie

podpis przyjmującego oświadczenie

Streszczenie

Niniejsza praca przedstawia i opisuje system informatyczny wspierający procesy magazynowo-dystrybucyjne dla niewielkich przedsiębiorstw.

Omówione zostają założenia techniczne i biznesowe na podstawie których stworzony został ten system a także możliwości jego wykorzystania i późniejszej rozbudowy.

Praca ta zawiera także instrukcje instalacji i użytkowania aplikacji.

Słowa kluczowe: magazynowanie, dystrybucja, handel, ERP, małe i średnie przedsiębiorstwa, Mercury

Spis treści

Spis rysunków.....	
Spis tabel.....	
1. Wstęp.....	
2. Przedstawienie programu.....	
3. Prezentacja programu.....	
4. Wnioski.....	
Literatura.....	
Dodatek A. [tytuł].....	
Dodatek B. [tytuł].....	

Spis rysunków

Rys. 2.1. Walec o promieniu r	3
---------------------------------------	-------------------

Spis tabel

Tabela 2.1. Przykład tabeli.....	3
----------------------------------	-------------------

1. Wstęp

1.1. Wprowadzenie

Oprogramowanie wspierające przechowywanie i dystrybucję towarów jest niezwykle pomocne a często wręcz niezbędne nie tylko dla przedsiębiorstw handlowych ale praktycznie dla każdej firmy od jednoosobowych działalności gospodarczych po wielkie międzynarodowe koncerny. O ile na rynku sporo jest systemów do planowania zasobów przedsiębiorstwa ([ang.](#) enterprise resource planning – *ERP*) i podobnych to jednak większość z nich przeznaczona jest dla średnich i dużych przedsiębiorstw. Produkty te są zazwyczaj poza zasięgiem najmniejszych inicjatyw gospodarczych zarówno pod względem finansowym, zasobów ludzkich do ich obsługi jak i posiadanej infrastruktury IT. Tańsze zaś i mniej wymagające ich odpowiedniki często nie są przetłumaczone na język polski a funkcjonalność którą oferują nie koniecznie odpowiada naszym lokalnym wymaganiom. Często zatem mają miejsce sytuacje kiedy to drobni mikro-przedsiębiorcy używają np. arkuszy kalkulacyjnych co jest mało wygodne i naraża szereg problemów choćby ze spójnością i bezpieczeństwem danych.

Niniejsza praca jest próbą rozwiązania tego problemu przez opracowanie darmowego¹, małego, prostego w obsłudze systemu informatycznego wspierającego przechowywanie i dystrybucję towarów mikro oraz małych przedsiębiorstw.

1.2. Cel i zakres pracy

Celem pracy było zaprojektowanie i implementacja systemu informatycznego do zarządzania stanem magazynowym i dystrybucji produktów w przedsiębiorstwie. Powstała aplikacja może być samodzielna, jak i stanowić część systemu ERP.

Praca swym zakresem obejmuje:

- zgromadzenie i zapoznanie się z literaturą tematu,
- opracowanie założeń projektu,
- zaprojektowanie struktury logicznej i interfejsu programu,

¹ Docelowo.

- realizacja programu, testowanie i usunięcie usterek,
- opracowanie pracy pisemnej.

1.3. *Struktura pracy*

Omówienie treści pracy, w postaci opisu zawartości poszczególnych rozdziałów (1,2,3,...).

Uwaga: treść tego rozdziału należy napisać po napisaniu pozostałej części pracy.

2. Przedstawienie programu

2.1. *Założenia projektowe*

2.1.1 *Użytkownicy docelowi*

Docelowymi użytkownikami prezentowanej aplikacji są:

- osoby dorywczo zajmujące się działalnością gospodarczą, często nieformalnie
- firmy jednoosobowe
- mikroprzedsiębiorstwa (do 10 pracowników)
- małe przedsiębiorstwa (do 50 pracowników)

2.1.2 *Wymagania użytkowe*

Użytkownicy tacy mają określone wymogi którym prezentowana aplikacja powinna sprostać. Są to:

- Zapewnienie funkcjonalności operacyjno-informacyjnej odnośnie przechowywania, zakupu, sprzedaży i przenoszenia towarów
- Prostota użytkowania

mikro i małe przedsiębiorstwa najczęściej nie dysponują wykwalifikowaną kadrą z dziedziny IT do obsługi firmowych aplikacji stąd oprogramowanie które ma im służyć powinno być na tyle łatwe i intuicyjne w użytkowaniu by mógł z niego korzystać niewykwalifikowany personel. Bardzo często będzie to po prostu właściciel jednoosobowej działalności gospodarczej.

- Oczysty dla użytkownika język interfejsu
- Niski koszt użytkowania

mikro i małe przedsiębiorstwa nie dysponują zazwyczaj dużym kapitałem stąd przeznaczony dla nich system nie może pociągać za sobą dużych kosztów użytkowania (np. opłat licencyjnych, kosztów utrzymania serwerów)

- Możliwość pracy wielu użytkowników z wielu stanowisk jednocześnie
- Zapewnienie spójności danych
- Zapewnienie bezpieczeństwa dostępu do danych
- Zapewnienie trwałości danych
- Zapewnienie nieprzerwanej dostępności usług
- Logowanie informacji o kluczowych procesach

2.1.3 Wymagania techniczne

Aplikacja Mercury powinna działać na najpopularniejszych obecnie standardowych komputerach klasy PC. Jej wymagania sprzętowe powinny mieścić się w specyfikacji przeciętnego komputera dostępnego obecnie na rynku tak aby nie było potrzeby wymiany sprzętu specjalnie dla tej aplikacji. Wg badań statystycznych z serwisu [gemiusRanking](http://www.gemius.pl)² 76,12% użytkowników używa systemu operacyjnego z rodziny Windows ale systemu operacyjne z rodziny UNIX (Linux i MacOS) mają także znaczny i co ważne rosnący udział w rynku (23,76%) dlatego też aplikacja Mercury powinna być zaprojektowana tak, by można jej było używać zarówno na systemach Windows jak i Linux i MacOS.

2.1.4 Ogólny schemat aplikacji

Aby pogodzić wymóg zapewnienia spójności danych z wymogiem zapewnienia możliwości pracy wielu użytkownikom na wielu stanowiskach jednocześnie projektowana aplikacja musiałaby:

- A) zapewnić ciągłą synchronizację danych wszystkich instancji programu ze sobą*
albo
- B) być podzieloną na część serwerową (wspólną dla wszystkich użytkowników) odpowiedzialną za przechowywanie danych oraz na część kliencką która mogłaby mieć wiele niezależnych od siebie instancji dla każdego użytkownika.*

² Zob. <http://www.ranking.pl/pl/rankings/operating-systems-families.html>

Przeanalizujemy zalety i wady obu rozwiązań:

A – zalety:

- każda instancja byłaby samodzielna
- dane byłyby powielone w każdej instancji a przez to zmniejszyłoby się ryzyko ich utraty

A – wady:

- każda instancja musiałaby zawierać pełną funkcjonalność (w dużej mierze wspólną dla wszystkich – niepotrzebna redundancja kodu)
- Każda instancja wymagałaby przechowywania kompletu danych co prowadziłoby do niepotrzebnego ich powielania
- Każda instancja musiałaby nieustannie komunikować się ze wszystkimi innymi instancjami w celu synchronizacji danych
- Wymagana synchronizacja danych wymaga także ciągłego dostępu do sieci
- Problem z weryfikacją czy na pewno wszystkie aktualnie działające instancje „widzą” siebie nawzajem
- Problem z rozsynchronizowaniem danych w przypadku korzystania z odrębnych sieci – i ich ponowną synchronizacją
- Problem z bezpieczeństwem danych – w sytuacji kiedy każda instancja przechowuje ich komplet dane te mogą dostać się w niepowołane ręce
- Problem z aktualizowaniem logiki biznesowej – wszystkie instancje musiałby jednocześnie być aktualizowane

B – zalety:

- dane mogłyby być odseparowane i trzymane w jednej bazie
- logika biznesowa mogłaby być osadzona na części serwerowej i przez to wspólna dla każdej instancji klienckiej
- brak potrzeby ciągłej synchronizacji danych gdyż wszystkie one trzymane byłyby w jednej, wspólnej bazie
- lepsza kontrola nad dostępem do danych

- część kliencka mogłaby być „lżejsza” i nie wymagałaby przechowywania kopii wszystkich danych
- aktualizacja schematu bazodanowego i logiki biznesowej mogłaby odbywać się tylko w części serwerowej i w wielu przypadkach być niewidoczną dla części klienckich

B – wady:

- konieczność ciągłego podłączenia do sieci
- jedna wspólna baza danych wymagałaby odrębnego tworzenia kopii zapasowej danych w celu ochrony przed ich utratą

Jak można zauważyć bilans zalet i wad wskazuje zdecydowanie na rozwiązanie drugie. Taki właśnie schemat przyjąłem dla aplikacji Mercury. Warto jednak zauważyć że rozwiązania informatyczne typu A nie są całkowicie porzucone i obecnie znajdują zastosowanie tam gdzie wymagana jest decentralizacja systemu informatycznego choćby ze względu na zagrożenia jej przejęcia lub zablokowania. Najlepszym przykładem jest tutaj sieć Bitcoin.

2.2. Planowana funkcjonalność

Dla zapewnienia funkcjonalności operacyjno-informacyjnej odnośnie przechowywania, zakupu, sprzedaży i przenoszenia towarów Mercury powinno umożliwiać gromadzenie, przetwarzanie oraz prezentację następujących informacji:

- *Magazyny*
- *Rodzaje towarów*
- *Towary*
- *Zakupy*
- *Miejsca zakupów*
- *Sprzedaże*
- *Miejsca sprzedaży*

- *Przeniesienia magazynowe*

2.3. Dobór narzędzi

Aby spełnić założone wymagania potrzeba wybrać odpowiednie narzędzia. Narzędzia te można podzielić na następujące kategorie:

1. Baza danych
2. Serwer aplikacyjny
3. Język programowania
4. Kliencka część systemu

2.3.1 Baza danych

Dane programu Mercury powinny być utrwalane. Najwygodniejszym rozwiązaniem będzie tutaj relacyjna baza danych SQL. Jest to rozwiązanie standardowe do tego typu aplikacji, wydajne i zapewniające transakcyjność operacji. Mercury nie ma niestandardowych wymagań w tym zakresie więc najlepsze będą znane, obecne na rynku rozwiązania. Jako że Mercury jest dedykowany dla niewielkich przedsiębiorstw jego baza danych nie wymaga bardzo dużej wydajności. Przeciętna wydajność będzie wystarczająca. Ważnym jest natomiast by rozwiązanie to było niedrogie a najlepiej darmowe.

Do najpopularniejszych na rynku silników baz danych SQL należą:

2.3.1.1. *MySQL*

2.3.1.2. *Oracle Database*

2.3.1.3. *PostgreSQL*

2.3.1.4. *Firebird*

2.3.1.5. *Microsoft SQL Server*

2.3.2 *Serwer aplikacyjny*

W rozdziale 2.1.4 uzasadniłem podział systemu Mercury na część serwerową i kliencką. Odnośnie części serwerowej pozostaje jeszcze wybór konkretnego rozwiązania. Można

2.4. *Schemat aplikacji*

[opis zakładanych cech projektowanego programu: założenia projektu, planowana funkcjonalność, schematy relacji, przepływ sterowania, itp.]

3. Prezentacja programu

3.1. Wymagania programowo-sprzętowe

[określić parametry komputera, na jakim opracowana prezentacja da się uruchomić: klasa procesora (jeśli to istotne), wymagane oprogramowanie, np. określona wersja przeglądarki, obsługującej określone mechanizmy]

3.2. Instalowanie i uruchomienie programu

[opisać jak program się instaluje i/lub uruchamia]

3.3. Posługiwanie się programem

[ilustrowana instrukcja obsługi (zaczynająca się od ekranu startowego programu), z opisem poszczególnych elementów ekranu, nawigacji itp.]

4. Wnioski

Rozdział ostatni ma zawierać wnioski z realizacji pracy. Wnioski powinny obejmować (w odrębnych akapitach):

- 4.1. *Określenie czy cel pracy został osiągnięty,***
- 4.2. *Wyjaśnienie korzyści wnoszonych przez zrealizowanie pracy,***
- 4.3. *Omówienie problemów napotkanych w trakcie realizacji oraz jak wskazanie sposobów ich rozwiązania,***
- 4.4. *Wskazanie jak w przyszłości pracę można by rozwinąć lub ulepszyć.***

Literatura

- [1] PN-N-01623:1984 *Dokumentacja konstrukcyjna. Zasady wykonywania dokumentów tekstowych*
- [2] PN-N-01180:1974 *Opracowania dokumentacyjne. Postanowienia ogólne*
- [3] PN-N-01180:1979 *Opracowania dokumentacyjne. Informacja ekspresowa i przegląd dokumentacyjny*
- [4] <http://pl.wikipedia.org/wiki/DHCP>
- [5] http://drukarnie.com.pl/statystyki_z_branzy/statystyki_z_branzy.asp
- [6] J. Kowalski, *Jak składać teksty komputerowe*, PWN, Warszawa, 1995
- [7] P. Liszewski, „Skanery”, *PC World Komputer*, Nr 4, Vol. II, IV.2000, ss. 72-73

Dodatek A. [tytuł]

[treść dodatku A]

Dodatek B. [tytuł]

[treść]

co napisać:

- że można rozbudować program o opcje usług