

Magdalena DOBRZAŃSKA, Sławomir GÓRKA
Politechnika Rzeszowska, Zakład Informatyki

PERSONAL WEB SERVER JAKO PRZYKŁAD SERWERA WWW DLA SIECI LOKALNYCH

Streszczenie. W niniejszym artykule przedstawiono działanie i możliwości programu Personal Web Server (PWS). Opisano zastosowanie technologii Active Server Pages (ASP), która jest obsługiwana przez PWS. Całość poparta została praktycznym przykładem wykorzystania w ww. technologii do przeprowadzenia ankiet studenckich.

PERSONAL WEB SERVER IN LOCAL AREA NETWORKS

Summary. The article presents the abilities of Microsoft Personal Web Server (PWS). Using of Active Server Pages Technology in PWS was described. The practical example concerning the process of managing students' questionnaires was prepared with the use of ASP Technology.

1. Wstęp

Wraz z rozwojem sieci Internet zmieniły się fundamentalnie sposoby projektowania i dostępu do sieci lokalnych korzystających z protokołu IP. Obecnie noszą one nazwę sieci Intranet i stały się miniaturowymi replikami Internetu, które obsługują firmowe dokumenty i aplikacje. Sieci lokalne przeobraziły się w sieci Intranet pod wpływem takich zjawisk, jak komercjalizacja Internetu oraz rozwój hipertekstu. Bezpośrednią przyczyną przeobrażenia była chęć korzystania z bogatych zasobów informacyjnych Internetu.

Utworzenie Intranetu w przypadku istniejącej fizycznie sieci wymaga podłączenia serwera i zainstalowania w nim odpowiedniego oprogramowania, np. programu Personal Web Server (PWS) dostarczanego wraz z systemem Windows 98 i Windows NT Workstation.

Personal Web Server jest stosowany nie tylko jako mały serwer intranetowy, ale także jako narzędzie służące do tworzenia i testowania profesjonalnych stron WWW.

Wraz z PWS otrzymuje się wiele dodatkowych produktów znacznie rozszerzających możliwości serwisu WWW. Należą do nich:

- *Microsoft FrontPage Server Extension*, będący zestawem rozszerzeń umożliwiających ścisłą współpracę serwera WWW z programem Microsoft FrontPage, służącym do projektowania witryn WWW i zarządzania nimi.
- *Microsoft Transaction Server 2.0*, jako składnik umożliwiający tworzenie aplikacji serwera obsługi transakcji. Transakcjami są operacje zawsze traktowane jako całość niezależnie od tego, czy składają się z jednej czy wielu czynności. Jeśli wszystkie one zakończą się pomyślnie, to cała transakcja jest pomyślna, jeśli choć jedna zakończy się niepowodzeniem – transakcja jest ignorowana.
- *Microsoft Data Access Components 1.5*, będący zestawem składników umożliwiających korzystanie z baz danych za pośrednictwem serwera WWW. Obsługa odbywa się za pomocą obiektów ActiveX oraz sterownika Microsoft Access.
- *Message Queue Server 1.0*, jako składnik ułatwiający komunikację między aplikacjami. Umożliwia im przekazywanie informacji o transakcjach i dalsze kontynuowanie działania bez oczekiwania na potwierdzenie zakończenia zgłoszonej transakcji.

Wymienione powyżej składniki są przeznaczone dla zaawansowanych administratorów i projektantów serwisów WWW.

2. Zarządzanie serwerem PWS

Do zarządzania serwerem intranetowym PWS służy program Personal Web Manager (PWM). Opcje programu zgrupowane są w pięciu kategoriach. Pierwsza z nich *Strona główna*, pokazuje aktualny stan serwera WWW, lokalizację katalogu głównego witryny oraz umożliwia zastopowanie i uruchomienie serwera. Dwie następne opcje: *Publikuj* i *Witryna sieci Web*, zawierają kreatory do automatycznego projektowania prostych witryn. Najważniejsza jest ostatnia opcja *Zaawansowane*, umożliwiająca ustalenie podstawowych właściwości witryny. W polu *Katalogi wirtualne* prezentowana jest lista aktualnie zdefiniowanych katalogów zawierających udostępniane w sieci pliki, do których można dodawać kolejne pozycje. W wyświetlonym oknie w polu *Katalog* podaje się fizyczną lokalizację mapowanego katalogu, a w polu *Alias* nazwę, pod którą będzie on widziany przez internautów. Właściwości uprzednio zdefiniowanych katalogów wirtualnych można zmieniać, wybierając opcję *Edytuj właściwości*.

Ważną czynnością jest określenie nazwy dokumentu domyślnego witryny, a więc tego, który zostanie wysłany do przeglądarki wówczas, gdy wpisany w niej adres URL będzie zawierał tylko nazwę katalogu. Zwyczajowo uznana nazwą dokumentu głównego witryny jest *index.htm* (bądź *index.html*), jednak z bliżej nie znanych przyczyn w standardowej konfiguracji lista dokumentów w polu *Domyślne dokumenty* zawiera jedynie nazwy *default.htm* i *default.asp*. Jeśli więc w katalogu głównym witryny umieszczonych będzie kilka plików, do przeglądarki zostanie wysłany ten, którego nazwa jest pierwsza na liście.

Następna opcja *Zezwalaj na przeglądanie katalogów* umożliwia przeglądanie zawartości katalogów w sytuacji, gdy nie zawierają one dokumentu domyślnego. Jeśli jest ona włączona, serwer wyświetli w takiej sytuacji listę zawartości katalogów, w przeciwnym razie ukaże się jedynie komunikat o błędzie.

Ostatnia opcja, *Zapisuj dziennik aktywności witryny sieci Web*, uruchamia notowanie działalności serwera w pliku tekstowym (logu) w formacie NCSA.

3. Katalogi macierzyste, wirtualne i publikacyjne

Chcąc umieścić witrynę sieci Web na serwerze należy określić katalogi zawierające dokumenty przewidziane do publikacji. Każda witryna sieci Web musi mieć jeden katalog macierzysty, który jest punktem początkowym dla gości witryny. W katalogu tym znajduje się plik indeksu, który z reguły zawiera powitanie gości i łączy do innych stron w sieci Web. Katalog macierzysty jest mapowany do nazwy domeny witryny. Jeśli na przykład nazwą domeny witryny Internet jest *www.zi.prz.rzeszow.pl*, a katalogiem macierzystym jest *C:\inetpub\wwwroot*, wówczas przeglądarki klientów w celu uzyskania dostępu do plików w tym katalogu używają adresu URL *http://www.zi.prz.rzeszow.pl/*.

Dodając nowe pliki HTML dla witryny sieci Web można wykorzystać domyślny katalog macierzysty *C:\inetpub\wwwroot*, utworzony podczas instalacji serwera. Pliki należy umieścić w katalogu macierzystym lub zorganizować je w podkatalogach katalogu macierzystego. Wszystkie pliki w katalogu macierzystym i jego podkatalogach są automatycznie dostępne dla gości odwiedzających witrynę. Znając ścieżkę i nazwę pliku, gość może obejrzeć plik, nawet jeśli na stronie głównej nie ma do niego łączy. Dlatego w katalogu macierzystym i jego podkatalogach należy przechowywać tylko te pliki, które mają być dostępne dla wszystkich gości.

Aby opublikować pliki z innego katalogu, który nie jest zawarty w katalogu macierzystym, tworzy się go jako katalog wirtualny. Katalog taki nie jest fizycznie zawarty w katalogu macierzystym, lecz przeglądarkom klientów wydaje się, że jest w nim zawarty.

Katalog wirtualny ma alias, czyli nazwę, którą przeglądarki klientów wykorzystują w celu uzyskania dostępu do tego katalogu. Alias jest zazwyczaj krótszą nazwą niż nazwa ze ścieżką dostępu do katalogu, i dzięki temu wygodniejszy do wpisywania dla użytkowników. Stosowanie aliasów jest poza tym bezpieczniejsze, ponieważ alias nie wskazuje, gdzie pliki są fizycznie umieszczone na serwerze. Wprowadza on też ułatwienie przy przenoszeniu katalogów w witrynie. Zamiast zmieniać adres URL dla strony, wystarczy zmienić przyporządkowanie między aliasem i fizyczną lokalizacją strony.

Jeśli przewiduje się publikowanie plików, które mają być udostępniane w swoich pierwotnych formatach (np. dokument tekstowy Word), lub plików rozproszonych w wielu katalogach na jednym lub kilku dyskach twardych, można wykorzystać kreator publikacji PWS do umieszczenia kopii tych plików w katalogu *C:\inetpub\Webpub*. Katalog ten jest katalogiem tylko do odczytu, toteż umieszczone w nim pliki nie mogą być modyfikowane. Kreator zapisuje także mapowanie opisujące miejsce położenia oryginalnych plików. Jeśli w oryginalnym pliku dokonywane są zmiany, istnieje możliwość odświeżenia kopii w katalogu Webpub. Jeśli plik ma być wycofany z publikacji, plik w katalogu Webpub jest usuwany, lecz plik oryginalny pozostaje nietknięty.

Po skopiowaniu pliku przez kreatora do katalogu Webpub, dodawane jest także łącze na stronie głównej. Istnieje możliwość dodania opisu ułatwiającego użytkownikom nawigację po witrynie sieci Web.

W przypadku gdy witryna sieci Web jest zbyt złożona lub jeśli przewiduje się użycie na jej stronach skryptów i aplikacji, może wystąpić potrzeba utworzenia dodatkowych katalogów dla zawartości przeznaczonej do publikacji. Katalogom tym można nadać różne uprawnienia dostępu, a mianowicie: *Odczyt*, *Wykonywanie* lub *Skrypt*.

Uprawnienie "*Odczyt*" umożliwia klientom sieci Web czytanie i pobieranie plików przechowywanych w katalogu macierzystym lub wirtualnym. Jeśli klient wskazuje plik, który znajduje się w katalogu bez uprawnienia "*Odczyt*", serwer sieci Web wysyła komunikat o błędzie. Na ogół katalogom zawierającym informacje do publikacji należy nadawać uprawnienie "*Odczyt*". Natomiast dla katalogów zawierających strony dynamiczne uprawnienie „*Odczyt*” należy wyłączyć, aby uniemożliwić klientom pobieranie plików aplikacji.

Uprawnienie „*Wykonywanie*” umożliwia uruchamianie wszelkich aplikacji z tego katalogu, włącznie z interpreterem skryptów i plików binarnych systemu Windows NT (pliki .dll i .exe).

Uprawnienie „*Skrypt*” nadawane katalogom umożliwia uruchamianie w tych katalogach aparatu skryptu bez uprawnienia "*Wykonywanie*". Uprawnienie "*Skrypt*" należy stosować do katalogów zawierających skrypty np. typu ASP (Active Server Pages), IDC (Internet Database Connector). Jest ono bezpieczniejsze niż uprawnienie "*Wykonywanie*", ponieważ

można ograniczyć aplikacje, które mogą być uruchamiane w katalogu. Jeśli klient wysyła żądanie uruchomienia skryptu w folderze, który nie ma uprawnień "Skrypt", serwer sieci Web wysyła komunikat o błędzie. Wszystkie katalogi tworzone podczas instalacji programu mają domyślnie ustawione uprawnienie "Skrypt".

PWS tworzy dziennik komputerów, które kontaktują się z witryną. W dzienniku rejestrowani są goście witryny, daty wizyt i listy przeglądanych plików. Sprawdzając dzienniki można się dowiedzieć, jak użytkownicy korzystają z witryny i upewnić się, czy nikt jej nie nadużył.

Dziennik jest zapisywany w formacie plików NCSA i może być przeglądany za pomocą dowolnego edytora tekstu. Nowe pliki są tworzone co miesiąc i umieszczane w katalogu Windows/System/LogFiles/W3SVC1. Nazwy plików zaczynają się od liter „Nc”, po których następuje symbol roku i miesiąca w postaci cyfrowej. Na przykład, plik dziennika sporządzonego za luty 2001 ma nazwę Nc0102.log.

4. Wydajność witryny sieci Web

Większość przeglądarek nie tworzy pojedynczego połączenia z witryną. Wiele przeglądarek może utrzymywać do czterech równoczesnych połączeń. Program PWS ma limit 10 połączeń. Oznacza to, że witryna może wygodnie obsługiwać dwóch lub najwyżej trzech użytkowników jednocześnie. Mały limit połączeń powoduje, że PWS nie może być stosowany do bezpośredniego publikowania w sieci Internet. Mimo to za pomocą programu PWS można utworzyć witrynę i dokładnie ją przetestować przed wysłaniem do profesjonalnego usługodawcy internetowego.

Aby ułatwić monitorowanie wydajności i wyszukiwanie potencjalnych przyczyn zmniejszenia wydajności lub innych problemów, program PWS tworzy wykres monitorowania witryny. Monitorowanie obejmuje różne sposoby prezentacji statystyki o aktywności witryny, np.:

- *Aktywne połączenia* określają liczbę aktywnych połączeń do przesyłania danych,
- *Start* o zawiera datę i godzinę ostatniego uruchomienia serwera,
- *Goście* jest liczbą unikatowych adresów gości, którzy byli połączeni z serwerem od chwili jego ostatniego uruchomienia. Jeśli ta sama osoba odwiedza witrynę kilka razy, wszystkie wizyty oprócz pierwszej nie są zliczane. Do celów porównawczych przechowywanych jest tylko 50 adresów. Powtórne wizyty są zliczane dopiero wówczas, gdy od ostatniej wizyty danego gościa do listy zostało dodanych więcej niż 50 innych adresów,

- *Żądania* jest to całkowita liczba żądań, które zostały odebrane przez serwer od chwili jego ostatniego uruchomienia,
- *Pobrano bajtów* jest to całkowity rozmiar danych, które zostały wysłane przez serwer od chwili jego ostatniego uruchomienia,
- *Najbardziej zbieżne połączenia* jest to liczba równocześnie otwartych połączeń z serwerem.

5. Porównanie serwerów PWS dla Windows 98 i Windows NT Workstation

Windows NT Workstation również wyposażony jest w serwer WWW. Mimo nazwy identycznej z serwerem przeznaczonym dla Windows 98 jest to produkt dużo bardziej zaawansowany, wywodzący się w prostej linii z programu Internet Information Server (IIS), przeznaczonego dla systemu Windows NT Server.

PWS dla NT Workstation to w istocie IIS z ograniczoną do 10 liczbą jednocześnie obsługiwanych połączeń, natomiast wszystkie zasadnicze funkcje programu IIS zostały w nim zachowane. Jest on idealnym środowiskiem do testowania funkcjonowania witryn skonstruowanych z wykorzystaniem ASP i innych mechanizmów charakterystycznych dla serwera IIS.

Program PWS zarówno w jednym, jak i drugim systemie zawiera graficzny interfejs administratora, opracowany z myślą o tych użytkownikach, którzy nie zajmowali się wcześniej tworzeniem witryn sieci Web i administracją serwera WWW. Ponadto Windows NT Workstation umożliwia zarządzanie witryną WWW przy użyciu Menadżera Usług Internetowych, dzięki któremu uzyskuje się dostęp do o wiele większej liczby funkcji. Zasadniczą różnicą w stosunku do PWS dla Windows 98 jest możliwość określania parametrów dostępu nie tylko do poszczególnych katalogów wchodzących w skład witryny, ale i pojedynczych plików oraz definiowania praw dostępu do nich. Można również uruchomić kilka niezależnych witryn przypisanych do jednego lub kilku osobnych adresów IP. PWS dla NT Workstation zawiera również serwer FTP, ale nie oferuje funkcji zarządzania poszczególnymi serwisami za pośrednictwem przeglądarki WWW tak jak IIS. Inną cenną cechą odziedziczoną po programie IIS jest możliwość rozbudowywania PWS o dodatkowe funkcje za pomocą tzw. filtrów ISAPI.

W tabeli 1 zostały przedstawione najważniejsze różnice pomiędzy przedstawionymi produktami.

Tabela 1

Porównanie produktów: PWS dla Windows 98 i Windows NT Workstation

Funkcja	PWS dla Windows 98	PWS dla Windows NT Workstation
Typowe zastosowania	Publikowanie informacji w mniejszych sieciach korporacyjnych intranet	Uniwersalne projektowanie witryn lub publikowanie informacji w sieciach korporacyjnych intranet
Usługa WWW	Tak	Tak
Usługa FTP	Nie	Tak
Limit połączeń	10	10
Active Server Pages	Tak	Tak
Site Use Logging (rejestrowanie użycia witryn)	Format plików dziennika NCSA	Format plików dziennika NCSA w standardzie MSCSV (domyślnie) oraz rozszerzony (opcjonalnie)
Źródła publikacji	Tylko dyski lokalne	Dyski lokalne i sieciowe
Opcjonalny interfejs ISM	Nie	Tak
Uwierzytelnienia	Brak	Podstawowe lub Windows NT Wezwanie/Odpowiedź
FrontPageExtensions	Tak	Tak
Filtry ISAPI	Nie	Tak
Zarządzanie przez WWW	Nie	Tak
Zarządzanie przez MMC	Nie	Tak
Kontrola dostępu	Nie	Tak
Ustalenie parametrów	Na poziomie katalogów	Na poziomie katalogów i plików

6. Technologia ASP

Strony ASP są kompozycją elementów języka HTML oraz tekstu i komend języka skryptowego. Technologię ASP można wykorzystać do tworzenia zarówno prostych, dynamicznych stron WWW, jak i zaawansowanych aplikacji opartych na bazach danych, dla których interfejsem użytkownika jest przeglądarka internetowa. Skrypty ASP interpretowane są w całości przez serwer WWW (rys. 2), który przetwarza polecenia skryptu (uwzględniając ewentualne przekazane parametry). Kod skryptu ASP zawarty jest pomiędzy znacznikami „<%....%>”. Dzięki tym znacznikom interpreter wie, gdzie się zaczyna i kończy kod ASP. Następnie do przeglądarki wysyłana jest gotowa strona zawierająca już tylko statyczny format HTML. Dzięki temu cały ciężar przetwarzania pozostaje na serwerze, gdzie

dodatkowo, przy wykorzystaniu mechanizmów ActiveX, skrypty ASP mogą komunikować się z bazami danych oraz wykonywać inne zadania wymagające dużej mocy obliczeniowej lub centralnego przetwarzania.

W omawianym w rozdziale 7 przykładzie, w kodzie źródłowym HTML formularza umieszczony jest znacznik *FORM* z atrybutem *ACTION* (rys. 1), który wskazuje, do jakiej strony zostaną przekazane dane z formularza.

formularz.html

```
<HTML><BODY>
...
<FORM ACTION="zapisz.asp" METHOD="POST">
...
</FORM>
</BODY></HTML>
```

zapisz.asp

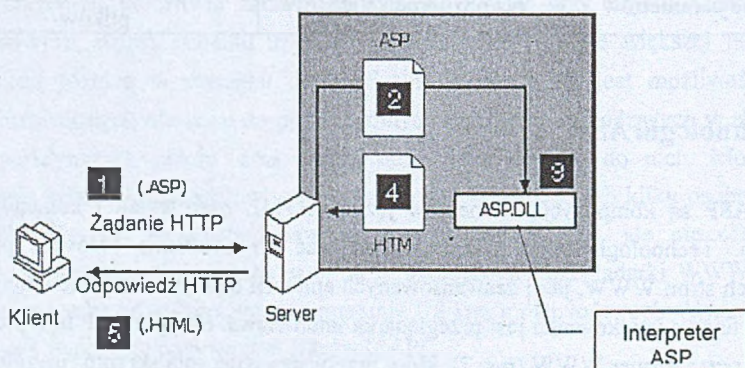
```
<%
Kod strony ASP
%>
```

Rys. 1. Działanie atrybutu ACTION w znaczniku FORM

Fig. 1. ACTION attribute working model

Technologia ASP jest ściśle związana z produktami firmy Microsoft. Skrypty pisane w tym języku mogą być wykonywane bez dodatkowych zabiegów przez Personal Web Server dla Windows 98 i dla Windows NT Workstation.

Do projektowania witryn internetowych opartych na technologii ASP można wykorzystać dowolny skryptowy język programowania, oczywiście pod warunkiem, że system będzie dysponował odpowiednim interpreterem. Serwer PWS zawiera standardowo interpreter ASP przeznaczony do wykonywania skryptów napisanych w języku VBScript i JScript. Niezależni producenci dostarczają także interpretry dla takich popularnych języków skryptowych, jak PERL i REXX.



Rys. 2. Model funkcjonowania ASP [6]

Fig. 2. ASP working model [6]

Alternatywną wobec ASP techniką dynamicznego tworzenia stron jest Personal Home Pages (PHP). Swoją konstrukcją i mechanizmami funkcjonowania przypomina technologie ASP, z tą różnicą że PHP samo w sobie jest pełnoprawnym językiem programowania, a ASP jedynie środowiskiem programistycznym. Dystrybucje tego systemu dostępne są w wersjach odpowiednich dla większości płaszczyzn sprzętowych i systemowych, w tym najpopularniejszych – Linuxa i Windows.

7. Przykład zastosowania technologii ASP

Zastosowanie Personal Web Server interpretującego skrypty ASP daje możliwość stworzenia witryny obsługującej dynamiczną wymianę danych. Mechanizmy te mogą zostać wykorzystane do realizacji np. ankiet.

Wykonany został przykład ankiety studenckiej, będącej elementem wewnętrznej oceny jakości nauczania. Przykład prezentuje możliwość współpracy ASP z bazami danych. Baza, do której trafiają dane z formularza, została utworzona w MS Access. Składa się ona z tabeli „studenci”, w której jeden rekord zawiera dane pochodzące z jednej wypełnionej ankiety, dotyczące:

- przedmiotu i wykładowcy podlegającego ocenie,
- punktów przyporządkowanych do odpowiedzi na pytania ankiety,
- danych statystycznych o osobie wypełniającej ankietę.

Analiza i dalsze przetwarzanie zapisanych danych odbywają się za pomocą narzędzi dostępnych w środowisku MS Access po przesłaniu do innych jednostek organizacyjnych, takich jak dziekanaty czy dział nauczania.

Aby dane wprowadzane do formularza (rys. 3) mogły zostać zapisane w bazie danych MS Access, niezbędne jest zastosowanie technologii łączenia baz danych z WWW. Należy również skonfigurować interfejs ODBC umożliwiający komunikację z bazami danych zapisanych w różnych formatach. W tym celu w oknie dialogowym administratora ODBC należy wybrać odpowiedni sterownik: „Microsoft Access Diver (*.mdb)” oraz określić nazwę źródła danych, ścieżkę i nazwę pliku używanej bazy danych.

Państwowa Wyższa Szkoła
Wydział Edukacji Matematyki i Logiki
Ocena pracy dydaktycznej nauczycieli akademickich

Proszę wypełnić ten formularz i przesłać go do: **Wydział Edukacji Matematyki i Logiki, Katedra Matematyki, ul. 15 Stycznia 1, 00-908 Warszawa**

Imię i nazwisko prowadzącego zajęcia:

Kierunek studiów:

Przedmiot:

Grupa:

Pytania dotyczące:	ocena 1	ocena 2	ocena 3	ocena 4	ocena 5
1. Przekazuje wiedzę w sposób zrozumiały i ciekawy	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2. Wskazuje powiązania i zależności między zagadnieniami	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3. Inspirował studentów do samodzielnego myślenia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4. Przekazuje wiadomości z zakresu historii matematyki	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5. Prowadzi zajęcia regularnie i punktualnie	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6. Nosiłoby wyniki oceny nauczyciela	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7. Jest dostępny dla studentów także poza wykładami	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8. Prowadzi zajęcia interesujące	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9. Wiele studentów jest zadowolonych z wykładu	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10. Chętnie wyraża opinię o wykładzie	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
11. Wyraża opinie o wykładzie także poza wykładami	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Łiczba odpowiedzi: Liczba ocen:

Rys. 3. Formularz przykładowej ankiety studenckiej
 Fig. 3. Form of students questionnaires

W opisywanym przykładzie dane z formularza trafiają do bazy „studenci.mdb”, natomiast procedura ich zapisu jest realizowana za pomocą skryptu „zapisz.asp”. Aby zapoznać się z metodą zapisu danych do bazy, należy przeanalizować kod źródłowy pliku „zapisz.asp”.

Obsługa interfejsu, pozwalającego na korzystanie z biblioteki ADODB zawierającej wszystkie obiekty ADO (ActiveX Data Objects) – umożliwiające tworzenie połączeń do baz danych dowolnego typu (np. lokalnych i zdalnych) - ma postać definicji:

```
Set Conn=Server.CreateObject("ADODB.Connection")
```

Instrukcja

```
Conn.Open"DBQ=C:\inetpub\Baza\studenci.mdb;  
Driver={Microsoft Access Driver (*.mdb)};  
DriverId=25;FIL=MS Access;"
```

tworzy połączenie ze źródłem danych, oraz określa parametry wykorzystywanej bazy.

Tworzony jest obiekt typu *RecordSet* o nazwie „dane”:

```
Set dane=Server.CreateObject("ADODB.RecordSet")
```

który pośredniczy w przesyłaniu informacji z formularza do bazy danych.

Przy użyciu instrukcji IF sprawdzany jest warunek, czy wypełnione zostały wszystkie pola nagłówka formularza:

```
If
```

```
Trim(Request.Form("Wykladowca"))<>"" And  
Trim(Request.Form("Kierunek"))<>"" And
```



```
Trim(Request.Form("Przedmiot")) <> "" And  
Trim(Request.Form("Semestr")) <> ""
```

Then...

Jeżeli nie wypełniono wszystkich pól, to użytkownik informowany jest o tym stosownym komunikatem:

```
<br>UWAGA!!<br>
```

Aby dane zostały zapisane do bazy danych należy wypełnić wszystkie pola w formularzu.

```
<br><br>
```

W przypadku gdy użytkownik poprawnie wypełnił formularz, w bazie danych tworzony jest nowy rekord:

```
dane.AddNew
```

Pobranie danych z formularza do zmiennej rekordowej wykonywane jest za pomocą funkcji Request. Jako argument wywołania funkcji podawana jest nazwa pola formularza:

```
dane("Wykładowca")=Request.Form("Wykładowca")
```

```
dane("Kierunek")=Request.Form("Kierunek")
```

Zapis danych z rekordu do bazy MS Access wykonuje instrukcja:

```
dane.UpdateBatch
```

O tym, że dane zostały dopisane do bazy, użytkownik informowany jest odpowiednim komunikatem:

```
<font face="Courier">
```

Informacje zostały dopisane do bazy

```
</font>
```

Przedstawione omówienie nie zawiera całości programu źródłowego „zapisz.asp”. Przedstawione zostały tylko wybrane fragmenty programu, które stanowią istotę zapisu danych do bazy „studenci.mdb”.

Niniejsze opracowanie omawia realizację ww. funkcji z wykorzystaniem interfejsu ADO za pośrednictwem technologii ODBC. Istnieją alternatywne rozwiązania pozwalające na implementację omawianego systemu z wykorzystaniem technologii OLE DB (Object Linking and Embedding DataBase), dającej możliwość wstawiania jako źródło danych plików dowolnego typu (np. dokumentów MS Word, MS Excel, wiadomości e-mail itp.). Technologia OLE DB kreowana jest przez Microsoft Corp. na następcę ODBC.

8. Podsumowanie

Oprogramowanie Personal Web Server jest małym serwerem WWW obsługującym technologię ASP, która umożliwia dynamiczną wymianę danych między witryną a dowolną bazą danych. PWS jest programem powszechnie dostępnym, gdyż dołączony jest do systemu operacyjnego Windows 95/98 oraz Windows NT Workstation.

LITERATURA

1. Bielecki J.: Java od podstaw. Intersoftland, Warszawa 1997.
2. Comer D.: Sieci komputerowe i intersieci. WNT, Warszawa 2000.
3. Hettihewa S.: Active Server Pages. Helion, Gliwice 1999.
4. Siyan K.: Windows NT: TCP/IP. Wydawnictwo Robomatic, Wrocław 1998.
5. Thayer R.: Visual Basic 6 – księga eksperta. Helion, Gliwice 1999.
6. Wrzalik P.: Technologia Active Server Pages.
http://pckurier.pl/praxis/artykuly/wrzalik_piotr/technasp/

Recenzent: Dr inż. Dariusz R. Augustyn

Wpłynęło do Redakcji 2 kwietnia 2001 r.

Abstract

The paper presents the implementation process of WWW pages' dynamical part. The Personal Web Server which is integral part of Microsoft Windows 95/98 and Windows NT Workstation system was described. PWS is Web server which is often used during the process of Web sites testing and publishing. The ASP Technology allows to implementation of data interchange between Web site and database located on server (fig. 1). In the practical part, the dynamic Web site was created (fig.2). The program gets data from the user and writes it to database in MS-Access format. The Web site was written in the form of interactive students' survey which has to be performed by every lecturer. The students in our department have been filled in the survey during the classes. The entered data are written via LAN to database running on PWS. Wide accessibility and simplicity of PWS software make the processes of performing a survey and data analysis straightforward.