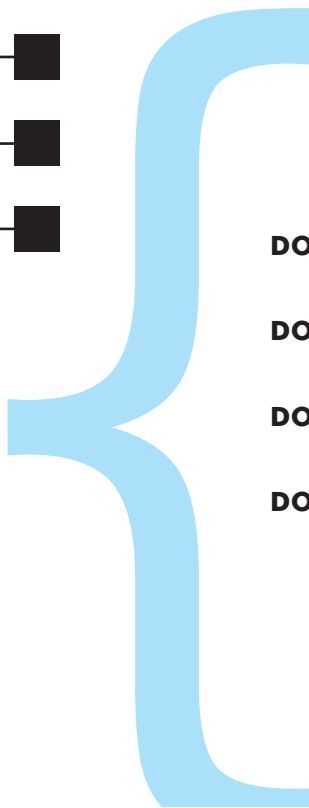


IV. dio

Dodaci

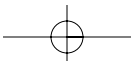
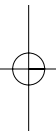


DODATAK A: .NET COMPACT FRAMEWORK

DODATAK B: DEBUGIRANJE

DODATAK C: POMOĆNI ALATI ZA .NET

DODATAK D: DVD



DODATAK A

.NET Compact Framework

U ovom poglavlju:

- Što je .NET Compact Framework
- Kako napraviti aplikaciju za Pocket PC

Cijelo vrijeme pričamo o razvoju aplikacija za osobna računala – što je s ručnim računalima i ostalim pametnim uređajima (engl. *smart devices*)? Implementacija punog .NET Frameworka, zbog njihovih brojnih ograničenja, ne bi bila dobro rješenje. Stoga je napravljen .NET Compact Framework – srezana inačica .NET Frameworka prilagođena mogućnostima tih uređaja.

Slično kao kod osobnih računala, da bi se aplikacije pisane na platformi .NET mogle koristiti, mora postojati podrška, bilo kao naknadna instalacija, bilo kao sastavni dio operativnog sustava. .NET Compact Framework (skraćeno .NET CF) u trenutku pisanja ovih redaka dio je operativnog sustava Windows CE .NET 4.2, a izvjesno je da će tako biti i sa svim sljedećim verzijama. Kako je sustav za ručna računala Windows Mobile 2003 (ponegdje referenciran kao Pocket PC 2003) baziran na spomenutoj verziji Windowsa CE, i on u sebi ima .NET CF.

Na sustave bazirane na Windows CE .NET 4.1 .NET Compact Framework može se instalirati u vidu dodatne instalacije, a isto vrijedi i za neke sustave bazirane na Windowsima CE 3.0, konkretno Pocket PC 2000 i Pocket PC 2002.

IV. DIO: DODACI

Međutim, čak i ako ciljani operativni sustav sadrži .NET CF, preporuka je nadograditi postojeće stanje aktualnim *service packom*. Adresa za preuzimanje nadogradnji i kompletnih inačica je:

<http://msdn.microsoft.com/mobility/downloads/updates/>



Ukoliko imate Visual Studio .NET 2003 Professional (ili bolji) .NET Compact Framework će doći s njim i po potrebi biti automatski instaliran na uređaj pri prvom pokretanju mobilne aplikacije. Više o tome kasnije u poglavlju.

Kao što ste vjerojatno očekivali, i .NET Compact Framework je usko povezan s razvojnim alatom Visual Studio. Međutim, on brojčano nešto sporije napreduje, tako da se uz Visual Studio .NET 2003 koristi .NET Compact Framework inačice 1.0, a aktualna verzija u trenutku pisanja označena je kao SP2. U njoj su popravljene i unaprijeđene određene mogućnosti, no osnovne karakteristike su ostale iste.



.NET Compact Framework podržava programiranje isključivo u jezicima VB.NET i C#.

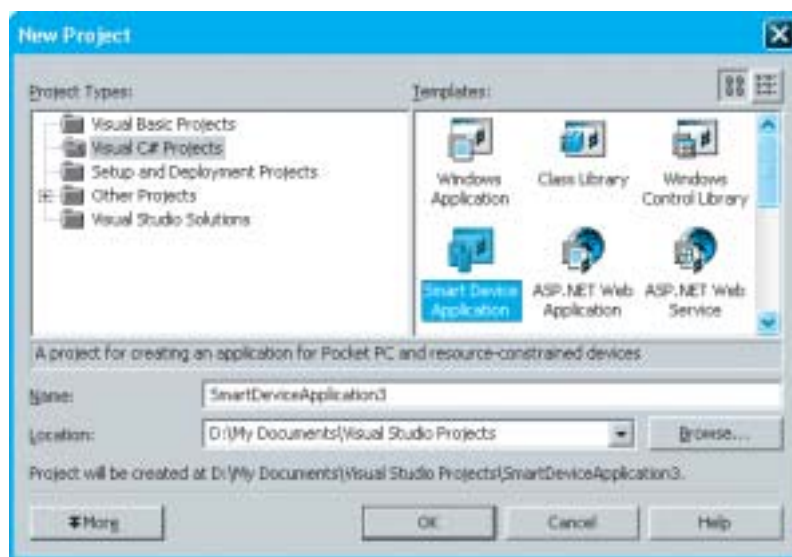
Ukoliko nemate pri ruci ručno računalo, to nije razlog da ne pišete aplikacije za njega. Naime, Visual Studio ima ugrađen emulator ručnog računala, pa vam prilikom razvoja ono uopće nije potrebno. Ipak, planirate li se ozbiljno time baviti, svakako nabavite i uređaj – sasvim je drugačiji osjećaj koristiti aplikaciju u emulatoru ili tapkajući olovčicom po pravoj stvari.

U ovom ćemo dodatku zajedno napraviti jednostavnu aplikaciju koristeći .NET Compact Framework te tako ukratko upoznati ograničenja i specifičnosti takvih aplikacija.

Izrada aplikacije za Pocket PC

Kao i svaku drugu aplikaciju, onu za .NET Compact Framework počinjemo stvaranjem novog projekta. Ovoga puta treba odabrati predložak Smart Device Application, kao što možete vidjeti na slici A-1.

DODATAK A : .NET COMPACT FRAMEWORK



Slika A-1:
Stvaranje novog projekta za pametne uređaje

Ukoliko u popisu nema potrebnog predloška, pokrenite instalaciju Visual Studija, potražite stavku **Smart Device Programmability** (unutar stavke **Visual C# .NET**) i uključite je.



Nakon odabira predloška i unošenja imena, aplikacija pojavit će se dodatni prozor koji će nam pomoći pri postavljanju osnovnih parametara (slika A-2).

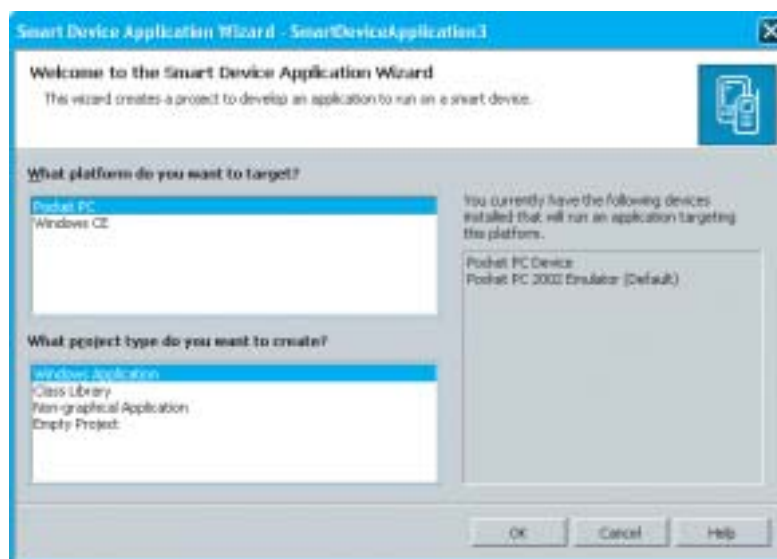
Ako instalirate **Smartphone SDK**, pojavit će se i predložak za pisanje aplikacija za **Smartphone**.



Prva, izuzetno važna odluka je odluka o ciljanoj platformi. Ukoliko izaberete platformu Windows CE, onda će se vaša aplikacija moći izvršavati na svim uređajima koji koriste taj operativni sustav. Kao što smo već spomenuli, sustavi Pocket PC 2000, Pocket PC 2002 i Windows Mobile 2003 koje nalazimo u ručnim računalima Pocket PC baziraju se na toj platformi, što znači da ćete aplikaciju moći koristiti na njima, ali i brojnim drugim uređajima koji koriste tu platformu.

IV. DIO: DODACI

Slika A-2:
Pomoćni prozor za
postavljanje para-
metara aplikacije



Detaljan opis razlika između razvoja aplikacija za Pocket PC i Windows CE općenito potražite na adresi:

<http://msdn.microsoft.com/library/enus/dnnetcomp/html/netcfPPctoCE.asp>

Međutim, mi ćemo za naš primjer izabrati platformu Pocket PC te se tako limitirati isključivo na sustave Pocket PC 2000, Pocket PC 2002 i Windows Mobile 2003. Zauzvrat će Visual Studio prilagoditi sve potrebne parametre kako bi razvoj takve aplikacije bio što jednostavniji. Naime, uređaji koji koriste Windows CE variraju po brojnim karakteristikama, pa je razvoj “općenite aplikacije” toliko kompliciran da je specijalizacija puno elegantnije rješenje. Ionako ćete u većini slučajeva razvijati aplikaciju upravo za uređaje tipa Pocket PC.

U drugom dijelu prozora biramo tip aplikacije koji želimo napraviti – mi smo se odlučili za tip Windows Application, a vi, ovisno o željama i potrebama, slobodno eksperimentirajte i s ostalim tipovima.

Forme i kontrole

Sučelje koje će nas dočekati nakon potvrde ovih izbora gotovo je identično početnom sučelju za izradu prozorskih aplikacija u .NET Frameworku. Zapravo, jedina je razlika broj dostupnih kontrola i njihovih svojstava te izgled određenih kontrola.

(Pri)ručna baza na ručnom računalu

Radite li na pametnom uređaju aplikaciju koja zahtijeva rad s bazom podataka, mogli biste pomisliti da se nalazite pred nepremostivom zaprekom. Srećom, tu je proizvod SQL Server CE, znatno pojednostavljena i optimizirana inačica velike poslužiteljske baze SQL Servera 2000.

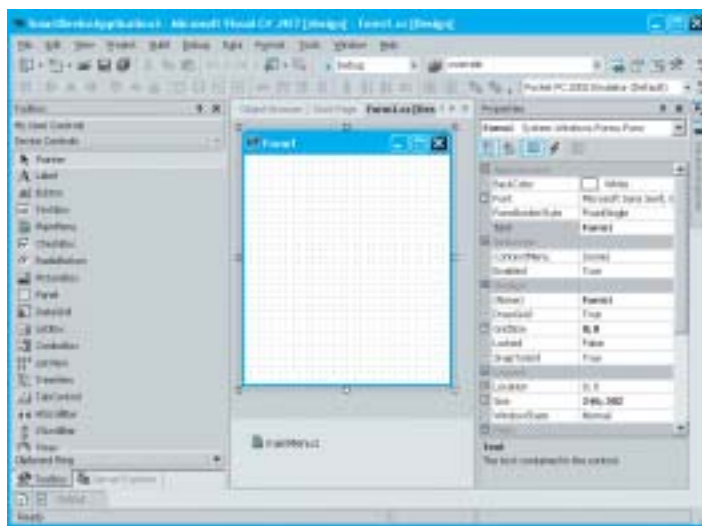
Taj proizvod, koji besplatno možete preuzeti s adrese <http://www.microsoft.com/sql/ce/>, može raditi na dva načina – kao samostalna baza podataka ili kao svojevrsan međuspremnik podataka koji se povremeno sinkronizira s centralnom, velikom bazom.

Iako sama baza podataka nije sastavni dio .NET Compact Frameworka, međusobno vrlo dobro

surađuju. Tako podršku za SQL Server CE možemo pronaći u Visual Studiju .NET 2003, a podacima iz aplikacije pristupamo koristeći klase u biblioteci ADO.NET. Naravno, iz očitih razloga, postoji velik broj stvari koje nisu podržane u odnosu na SQL Server za osobna računala.

Naravno, bazama smještenim na poslužitelju iz aplikacija na ručnom računalu možete pristupati direktno, no u tom slučaju među njima mora postojati stalna veza, što zbog naravi pametnih uređaja često nije slučaj.

Za rad sa SQL Serverom CE koristi se namespace `System.Data.SqlServerCE`, a za direktan rad s poslužiteljskom bazom `System.Data.SqlClient`.

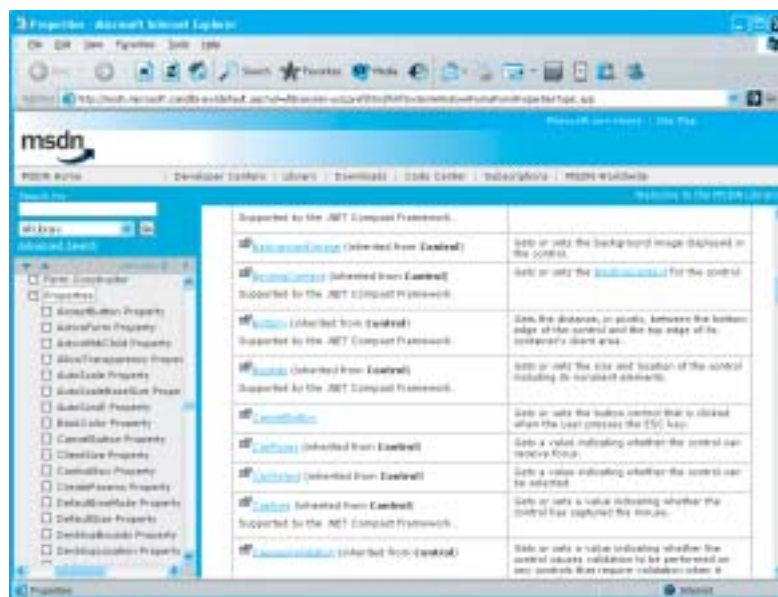


Slika A-3:
Osnovno sučelje za razvoj aplikacija za Pocket PC

IV. DIO: DODACI

Vjerojatno ste prilikom proučavanja dokumentacije primijetili kako određene kontrole, svojstva, metode i događaji imaju napomenu *Supported by the .NET Compact Framework*. One koji imaju tu napomenu podržava .NET Compact Framework. Naravno, za to ne morate svako malo viriti u dokumentaciju – izborom predloška Smart Device Application Visual Studio automatski skriva sve nepodržane karakteristike i mogućnosti, kako u pomoćnim prozorima tako i u tehnologijama poput IntelliSensa.

Slika A-4:
Dokumentacija ima
jasno naznačeno što
podržava u .NET
Compact Frameworku



Prilikom otvaranja nove forme automatski će biti dodana kontrola za izbornik. To je stoga što bi svaka aplikacija za Pocket PC trebala imati izbornik. Vi ga, dakako, možete ukloniti, no time nećete uštedjeti prostor jer se izbornik na ručnim računalima nalazi u sistemskoj traci.

Poznato je da ručna računala slijede logiku nešto drugačiju od one na osobnim računalima. Jedna od glavnih razlika je da programi na ručnom računalu klikom na ikonicu ne bivaju zatvoreni, nego se minimiziraju. Takvo ponašanje posebno je nezgodno za vrijeme razvoja aplikacije jer će *debugger* ostati aktivan sve dok aplikaciju ne zatvorimo. Stoga prilikom kreiranja nove forme najprije treba postaviti svojstva `MinimizeBox` na `false`. Na taj način onemogućit ćemo minimiziranje aplikacije, a umjesto ikonića pojavit će se gumbić s natpisom "ok". Klik na taj gumbić zatvorit će formu, a ukoliko je osnovna, i aplikaciju.

Mi ćemo za primjer napraviti jednostavnu aplikaciju za prevođenje s engleskog jezika na talijanski, i obrnuto. Koje smo kontrole koristili možete vidjeti na slici A-4 – uz kozmetičke, nužno je postaviti sljedeće: `textBox1` (tip `TextBox`), `button1` i `button2` (tip `Button`) te `label1` (tip `Label`).

DODATAK A : .NET COMPACT FRAMEWORK



Slika A-3:
Osnovno sučelje za razvoj
aplikacija za Pocket PC

Za sam čin prevođenja koristimo besplatan web-servis, koji se nalazi na adresi

<http://www.webservicex.com/TranslateService.asmx>

Postupak dodavanja Web Reference u projekt znate iz prošlog poglavlja – desni klik na ime projekta u Solution Exploreru, zatim Add Web Reference gdje unosimo sljedeću adresu:

<http://www.webservicex.com/TranslateService.asmx?WSDL>

Kôd koji će odraditi funkcionalnost programa izgleda ovako – sljedeća funkcija vezana je uz oba gumba na našoj formi:

```
private void button_Click(object sender, System.EventArgs e)
{
    com.webservicex.www.Language jezik;
```

IV. DIO: DODACI

```

    if ((Button)sender == buttonEnIt)
        jezik = com.webservicex.www.Language.EnglishTOItalian;
    else
        jezik = com.webservicex.www.Language.ItalianTOEnglish;

    com.webservicex.www.TranslationService ws = new
        com.webservicex.www.TranslationService();
    ws.Timeout = 20000; // na odgovor čekamo do 20 sekundi

    try
    {
        labelPrijevod.Text = ws.Translate(jezik, textUnos.Text);
    }
    catch
    {
        labelPrijevod.Text = "Web-servis nije dostupan.";
    }
}

```

Virtualna tipkovnica

Sve dosad rečeno nije ništa novo. Međutim, ako pokrenete aplikaciju (na ručnom računalu, ne u emulatoru), primijetit ćete da ručno morate dozvati tipkovnicu kada želite upisati riječ. Nije da je to nešto strašno, no korisnici bi cijenili kada bismo napravili da se virtualna tipkovnica sama uključuje prilikom ulaska u polje za upis i isključuje izlaskom iz njega.

Stoga kraj forme treba dovući kontrolu InputPanel, a uz događaje GetFocus i LostFocus kontrole textUnos vezati sljedeće funkcije:

```

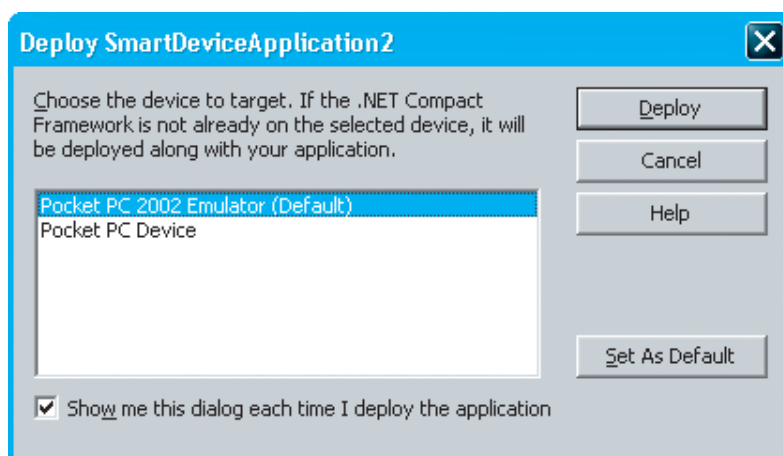
private void textUnos_GotFocus(object sender, System.EventArgs e)
{
    inputPanel1.Enabled = true;
}

private void textUnos_LostFocus(object sender, System.EventArgs e)
{
    inputPanel1.Enabled = false;
}

```

Testiranje aplikacije

Preostaje nam još testirati aplikaciju koju smo upravo napravili. I tu je priča vrlo slična izradi ostalih tipova aplikacija – jednostavno pritisnemo F5 i kreće postupak kompajliranja i pokretanja aplikacije.



Slika A-6:
Izbor uređaja na
kojem ćemo testirati
aplikaciju

Kako aplikacija koju smo upravo napisali nije namijenjena izvršavanju na osobnom računalu, morat ćemo koristiti emulator računala Pocket PC ili koristiti pravo ručno računalo. Zato će nam Visual Studio prilikom pokretanja programa postaviti pitanje sa slike A-6.

U pravilu je praktičnije testiranje raditi na emulatoru, pa stoga odabiremo njegovu stavku među ponuđenim opcijama. Kako sada prvi put pokrećemo emulator, na njega će prije pokretanja aplikacija automatski biti instaliran .NET Compact Framework.

Ista stvar bi se dogodila i na (pravom) ručnom računalu da smo njega izabrali za *deployment* aplikacije. Naravno, u trenutku pokretanja ručno računalo mora biti spojeno na osobno računalo, a ActiveSync, softver za njihovu sinkronizaciju i povezivanje, pravilno podešen i pokrenut.

Kao i kod ostalih aplikacija pisanih za platformu .NET, i ovdje je dovoljno kopirati izvršnu datoteku na ciljni uređaj i stvar će raditi. Želite li pak instalacijsku proceduru, morat ćete se pomučiti nešto više, kako je opisano na adresi <http://msdn.microsoft.com/library/en-us/dnnetcomp/html/-netcfdeployment.asp>.

IV. DIO: DODACI

Slika A-7:
Prilikom prvog pokretanja aplikacije iz Visual Studija automatski će se na odabrani uređaj instalirati nova inačica .NET Compact Frameworka.



Kako dalje?

Na ovih nekoliko stranica spomenuli smo samo osnove izrade aplikacija za pametne uređaje. Kako se cijela priča uvelike oslanja na koncepciju koju smo upoznali kod razvoja ostalih tipova aplikacija, vjerujemo da nećete imati problema sa samostalnim proučavanjem. Uvijek morate imati na umu da je .NET Compact Framework prilično osakaćena inačica .NET Frameworka te da nisu podržane mnoge stvari koje bismo mogli smatrati osnovnim. U takvim se okolnostima morate snalaziti koristeći ono što je dostupno i nadograđujući

podržanu funkcionalnost. Takav način rada zahtijeva određeno programersko iskustvo i domišljatost, a strpljivijima ostaje i nada da će sljedeća inačica .NET Compact Frameworka donijeti novosti koje će riješiti dio problema.

Više informacija, zanimljivosti i specifičnosti o .NET Compact Frameworku možete pronaći na adresama <http://msdn.microsoft.com/mobility>, <http://samples.gotdotnet.com/quickstart/CompactFramework/> i brojnim drugim web-stranicama te grupama na Usenetu.