

Poznaj nowoczesną platformę mobilną!



Windows Phone 7

Tworzenie efektownych aplikacji

Henry Lee • Eugene Chuvyrov

Jak wykorzystać funkcję geolokalizacji?

Jak przygotować pakiet instalacyjny i udostępnić go użytkownikom?

Jak zapewnić bezpieczeństwo aplikacji mobilnej?

Apress®



» Idź do

- Spis treści
- Przykładowy rozdział
- Skorowidz

» Katalog książek

- Katalog online
- Zamów drukowany katalog

» Twój koszyk

- Dodaj do koszyka

» Cennik i informacje

- Zamów informacje o nowościach
- Zamów cennik

» Czytelnia

- Fragmenty książek online

» Kontakt

Helion SA
ul. Kościuszki 1c
44-100 Gliwice
tel. 32 230 98 63
e-mail: helion@helion.pl
© Helion 1991–2011

Windows Phone 7. Tworzenie efektywnych aplikacji

Autorzy: Henry Lee, Eugene Chuyrov

Tłumaczenie: Marek Pałczyński

ISBN: 978-83-246-3387-6

Tytuł oryginału: [Beginning Windows Phone 7 Development](#)

Format: 158×235, stron: 408



Poznaj nowoczesną platformę mobilną!

- Jak wykorzystać funkcje geolokalizacji?
- Jak przygotować pakiet instalacyjny i udostępnić go użytkownikom?
- Jak zapewnić bezpieczeństwo aplikacji mobilnej?

Na rynku systemów operacyjnych dla telefonów komórkowych najwięcej mówi się obecnie o systemach Android oraz iOS. Nie można jednak zapominać o platformie firmy Microsoft! Jej najnowsza wersja – Windows Phone 7 – dopiero co ujrzała światło dzienne, a już zbiera pozytywne opinie. Wygodny interfejs, ogrom możliwości oraz świetna integracja z otoczeniem to tylko niektóre atuty nowego systemu. Zobacz, jak je wykorzystać!

Dzięki tej książce błyskawicznie rozpoczniesz tworzenie nowych aplikacji dla systemu Windows Phone. Na wstępie zapoznasz się z wymaganymi elementami stanowiska pracy oraz ich konfiguracją. Na kolejnych stronach znajdziesz przydatne informacje o przechowywaniu danych w chmurze, obsłudze błędów, zarządzaniu pakietami instalacyjnymi oraz udostępnianiu aplikacji użytkownikom. W dalszej części książki autorzy omawiają technologie systemu Windows Phone 7. W trakcie lektury nauczysz się używać akcelerometru, paska aplikacji, przeglądarki internetowej, aparatu fotograficznego oraz multimediiów. Ponadto zobaczysz, jak tworzyć aplikacje testowe, obsługiwać wiele języków oraz lokalizować użytkownika telefonu w terenie. Książka stanowi bogaty zbiór niezwykle przydatnych informacji na temat platformy Windows Phone 7. Musisz ją mieć!

- System Windows Phone 7 – wprowadzenie
- Przygotowanie środowiska programistycznego
- Chmura – miejsce składowania danych
- Przechwytywanie i usuwanie błędów
- Pakowanie i udostępnianie aplikacji
- Zarządzanie pakietami instalacyjnymi
- Korzystanie z akcelerometru
- Wykorzystanie paska aplikacji
- Przeglądarka internetowa
- Dostępne kontrolki interfejsu użytkownika
- Integracja aplikacji z systemem operacyjnym Windows Phone
- Aplikacje testowe
- Obsługa wielu języków
- Geolokalizacja
- Obsługa multimediiów
- Przetwarzanie zdjęć
- Obsługa powiadomień
- Bezpieczeństwo aplikacji mobilnych

Stwórz własną niepowtarzalną aplikację mobilną!

Spis treści

O autorach	13	
O redaktorze merytorycznym	15	
Podziękowania	17	
Wprowadzenie	19	
Część I	Podstawy tworzenia aplikacji Windows Phone 7	21
Rozdział 1.	Wprowadzenie do systemu Windows Phone 7 i platformy Windows Phone	23
	Ogólne informacje na temat systemu Windows Phone	23
	Specyfikacja sprzętowa Windows Phone	25
	Platforma programistyczna Windows Phone	25
	Technologia Silverlight w systemie Windows Phone	27
	Technologia XNA w systemie Windows Phone	27
	Narzędzia	27
	Usługi pracy w chmurze	30
	Interfejs Metro	31
	Proces przygotowywania aplikacji	31
	Podsumowanie	32
Rozdział 2.	Tworzenie aplikacji Windows Phone 7	33
	Przygotowanie środowiska programistycznego	33
	Utworzenie pierwszej aplikacji Windows Phone 7	33
	Tworzenie projektu Windows Phone	34
	Wykorzystanie kontrolki Silverlight	35
	Utworzenie kodu pierwszej aplikacji Windows Phone	39
	Uruchomienie pierwszej aplikacji Windows Phone	40
	Dostosowywanie pierwszej aplikacji Windows Phone	41
	Zmiana wyglądu aplikacji	42
	Podsumowanie	45

Rozdział 3. Wykorzystanie usług pracy w chmurze jako przestrzeni składowania danych	47
Wzorzec projektowy MVVM	48
Usługi Microsoft Azure oraz SQL Azure	49
Utworzenie usługi zapewniającej dostęp do bazy danych	49
Utworzenie bazy danych	50
Utworzenie bazy danych SQL Azure	50
Zarządzanie bazą danych SQL Azure	56
Utworzenie projektu Windows Azure	59
Generowanie modelu obiektowego odpowiedzialnego za dostęp do bazy danych	60
Implementacja usługi WCF pośredniczącej w dostępie do bazy danych SQL Azure	63
Aplikacja kliencka korzystająca z usługi działającej w chmurze	68
Utworzenie projektu aplikacji Windows Phone	70
Przygotowanie interfejsu użytkownika	70
Kod strony głównej	75
Kod konwertera BoolToVisibilityConverter	77
Dodanie odwołania do usługi NotepadService	79
Kod klasy NotepadViewModel	79
Testowanie współdziałania aplikacji z usługą uruchomioną lokalnie	86
Uruchomienie usługi w serwisie Windows Azure	86
Testowanie współdziałania aplikacji Notepad z usługą NotepadService	94
Podsumowanie	95
Rozdział 4. Przechwytywanie i usuwanie błędów	97
Debugowanie wyjątków aplikacji	97
Obsługa wyjątków związanych z ładowaniem strony głównej	98
Debugowanie wyjątków usługi sieciowej	102
Testowanie aplikacji	105
Rejestrowanie urządzenia Windows Phone na potrzeby procedury uruchomieniowej	106
Obsługa wyjątków urządzenia	109
Utworzenie projektu CatchingDeviceExceptionsDemo	110
Przygotowanie interfejsu użytkownika	111
Kod aplikacji	113
Testowanie gotowej aplikacji	114
Podsumowanie	114
Rozdział 5. Pakowanie i udostępnianie aplikacji oraz zarządzanie pakietami instalacyjnymi	115
Udostępnianie aplikacji Windows Phone	115
Wymagania certyfikacyjne dla aplikacji Windows Phone	117
Założenia funkcjonalne (Application Policies)	117
Założenia dotyczące przekazywanych treści (Content Policies)	119
Wymagania związane z walidacją aplikacji	119
Wymagania związane z certyfikacją aplikacji	120

Przesyłanie aplikacji Windows Phone do serwisu Windows Phone Marketplace	121
Pakowanie aplikacji	121
Przesyłanie aplikacji	122
Aktualizacja aplikacji	127
Wyszukiwanie własnych aplikacji w serwisie Marketplace	130
Podsumowanie	130
Część II	
Technologie systemu Windows Phone 7	131
Rozdział 6. Akcelerometr	133
Położenie i przemieszczenie	133
Obliczanie odległości	134
Pochylenie i przechylenie	135
Wykorzystanie narzędzi SDK do obsługi akcelerometrów	137
Pobieranie danych z akcelerometru	137
Utworzenie projektu CaptureAccelerometerData	138
Przygotowanie interfejsu użytkownika	139
Kod aplikacji	140
Testowanie ukończonej aplikacji	143
Przesuwanie kulki za pomocą akcelerometru	144
Utworzenie projektu	144
Przygotowanie interfejsu użytkownika	144
Kod aplikacji	146
Testowanie ukończonej aplikacji	148
Podsumowanie	149
Rozdział 7. Pasek aplikacji	151
Podstawowe informacje na temat paska aplikacji	152
Dodawanie paska aplikacji do projektu Windows Phone 7	153
Dodawanie obrazów paska zadań	153
Dodawanie globalnego paska aplikacji w kodzie XAML	155
Dodawanie lokalnego paska aplikacji w kodzie XAML	155
Dodawanie tekstowych pozycji menu	157
Dodawanie paska aplikacji za pomocą zarządzanego kodu	158
Obsługa zdarzeń paska aplikacji	160
Utworzenie kodu łącznikowego i funkcji wykonawczej	160
Reakcja na zdarzenie kliknięcia przycisku „dodaj”	161
Reakcja na zdarzenie kliknięcia przycisku „zapisz”	163
Reakcja na wybranie pozycji menu	164
Dodawanie procedur obsługi zdarzeń w kodzie XAML	164
Podsumowanie	165
Rozdział 8. Kontrolka przeglądarki	167
Podstawowe informacje na temat kontrolki WebBrowser	167
Użycie kontrolki WebBrowser	168
Wykorzystanie kontrolki WebBrowser do wyświetlania stron internetowych	169
Wykorzystanie kontrolki WebBrowser do wyświetlania lokalnych plików HTML	170

Wykorzystanie kontrolki WebBrowser do wyświetlania dynamicznie generowanych treści	173
Zapisywanie stron internetowych w pamięci telefonu	174
Wybór wyświetlacza oraz ustawienia systemu zabezpieczeń	176
Obszar prezentacji	176
CSS	176
Zabezpieczenia	177
Podsumowanie	178
Rozdział 9. Kontrolki i kompozycje graficzne	179
Podstawowe informacje na temat systemu graficznego Metro	179
Obszary Windows Phone Chrome	180
Ułożenie ekranu	181
Kompozycje graficzne urządzeń Windows Phone 7	182
Ustawienie kompozycji graficznej	182
Zmiana kompozycji graficznej	184
Sprawdzenie bieżącej kompozycji graficznej	186
Kontrolki Panorama i Pivot	187
Kontrolka Panorama	187
Kontrolka Pivot	189
Ramki i nawigacja między stronami	190
Utworzenie interfejsu użytkownika w projekcie NavigationTest	191
Dodanie kodu nawigacji	191
Kod przekazywania parametrów między stronami	191
Efekty przejścia	192
Utworzenie interfejsu użytkownika	192
Pobranie pliku TiltEffect.cs i wykorzystanie właściwości zależnej	193
Podsumowanie	194
Rozdział 10. Integracja aplikacji z systemem operacyjnym Windows Phone	195
Podstawowe informacje na temat funkcji uruchamiania i wyboru	195
Funkcje uruchamiania	196
Funkcje wyboru	197
Wykorzystanie funkcji uruchamiania i wyboru	197
Utworzenie interfejsu użytkownika	197
Kod aplikacji	197
Cykl życia aplikacji Windows Phone 7	199
Analizowanie zdarzeń w cyklu życia aplikacji	199
Stan aplikacji	203
Zalecane techniki zarządzania cyklem pracy aplikacji Windows Phone 7	207
Centra Windows Phone 7	208
Podsumowanie	209
Rozdział 11. Budowanie aplikacji testowych	211
Testowa i pełna wersja aplikacji	211
Wykorzystanie metody IsTrial	212
Interfejs API serwisu Marketplace	213
Symulowanie pracy w trybach testowym i pełnym	215

Budowanie aplikacji testowej	218
Tworzenie interfejsu użytkownika	218
Połączenie z usługą sieciową	222
Nawigacja między stronami	224
Sprawdzenie pracy w trybie testowym i pełnym	225
Ostatnie zmiany	226
Podsumowanie	228
Rozdział 12. Internacjonalizacja	229
Podstawowe informacje o internacjonalizacji	229
Ustawienia kulturowe metody ToString w formatowaniu dat, czasu i tekstu	231
Neutralność kulturowa aplikacji — hierarchia kultur	233
Zapisywanie i odtwarzanie bieżących ustawień kulturowych	234
Wykorzystanie plików zasobów do lokalizacji treści	238
Podsumowanie	243
Rozdział 13. Izolowana pamięć masowa	245
Korzystanie z izolowanej pamięci masowej	246
Utworzenie projektu IsolatedStorageStoreImageDemo	247
Kod interfejsu użytkownika	247
Kod aplikacji	249
Izolowana pamięć ustawień	253
Utworzenie nowego projektu	254
Utworzenie interfejsu użytkownika (w kodzie XAML)	254
Kod aplikacji	256
Podsumowanie	258
Rozdział 14. Usługi pozycjonowania	259
Architektura usług pozycjonowania w systemie Windows Phone	259
Interfejs programistyczny usługi pozycjonowania w telefonie Windows Phone	260
Symulowanie pracy usługi pozycjonowania	261
Utworzenie projektu GeoCoordinateWatcherDemo	261
Kod interfejsu użytkownika	261
Kod aplikacji	263
Testowanie ukończonej aplikacji	265
Wykorzystanie klasy GeoCoordinateWatcher i kontrolki serwisu Bing Maps do śledzenia zmian pozycji	266
Rejestracja w serwisie Bing Maps i instalacja pakietu SDK Bing Maps	266
Utworzenie projektu BingMapDemo	268
Utworzenie interfejsu użytkownika	268
Kod aplikacji	270
Testowanie ukończonej aplikacji	274
Zaznaczanie adresu na mapie i korzystanie z usługi Bing Maps	274
Utworzenie aplikacji AddressPlottingDemo	274
Odwołanie do usługi Bing Maps GeoCodeService	275
Utworzenie interfejsu użytkownika	275
Kod aplikacji	278
Testowanie ukończonej aplikacji	280
Podsumowanie	280

Rozdział 15. Multimedia	281
Podstawowe informacje na temat komponentu MediaElement	281
Odtwarzanie plików audiowizualnych	282
Utworzenie projektu MediaPlayerDemo	283
Utworzenie interfejsu użytkownika	283
Kod aplikacji	286
Dźwięki aplikacji	291
Utworzenie projektu RobotSoundDemo	291
Utworzenie interfejsu użytkownika	292
Kod aplikacji	295
Podsumowanie	296
Rozdział 16. Aparat fotograficzny i przetwarzanie zdjęć	297
Funkcje fotograficzne telefonu Windows Phone 7	297
Wykonywanie fotografii za pomocą komponentu CameraCaptureTask	299
Przeglądanie zdjęć za pomocą komponentu PhotoChooserTask	302
Zapisywanie fotografii w pamięci telefonu	303
Integracja aplikacji z systemem Windows Phone 7	304
Wykorzystanie funkcji Extras do uruchomienia aplikacji	305
Wykorzystanie funkcji Share do przesłania zdjęć do serwisu TwitPic	307
Podsumowanie	311
Rozdział 17. Powiadomienia	313
Podstawowe informacje o powiadomieniach	313
Powiadomienia wyskakujące	314
Powiadomienia kafelkowe	314
Powiadomienia niesformatowane	315
Architektura systemu powiadomień	316
Cykl życia powiadomienia	317
Środowisko programistyczne systemu powiadomień	318
Implementacja powiadomień wyskakujących	318
Utworzenie aplikacji klienckiej	318
Utworzenie aplikacji wysyłającej powiadomienia	323
Implementacja powiadomień kafelkowych	327
Utworzenie aplikacji klienckiej	327
Utworzenie aplikacji wysyłającej powiadomienia	328
Implementacja powiadomień niesformatowanych	330
Utworzenie aplikacji klienckiej	330
Utworzenie aplikacji wysyłającej powiadomienia	333
Sprawdzenie mechanizmu dostarczania powiadomień	334
Wykorzystanie usługi sieciowej do rejestrowania klientów powiadomień	335
Utworzenie usługi WCF rejestrującej odbiorców	335
Dodanie wywołania usługi WCF w aplikacji klienckiej	339
Sprawdzenie mechanizmu rejestracji programów odbierających powiadomienia	341
Praktyczne zastosowanie powiadomień	342
Podsumowanie	343

Rozdział 18. Reactive Extensions	345
Wprowadzenie do programowania reaktywnego	346
Implementacja wzorca obserwatora w środowisku Rx.NET	347
Utworzenie projektu Windows Phone	347
Kod utworzenia i odczytania kolekcji Observable	348
Wykorzystanie mechanizmu Rx.NET	
do przeszukiwania zbioru fotografii serwisu Flickr	350
Utworzenie projektu aplikacji Windows Phone	350
Przygotowanie interfejsu użytkownika	351
Kod wyszukiwania zdjęć w serwisie Flickr	351
Udoskonalenie wyszukiwarki	352
Wyświetlanie animacji podczas ładowania zdjęć	354
Zastosowanie środowiska Rx.NET	
do asynchronicznego pobierania danych o pogodzie z usługi sieciowej	358
Utworzenie projektu aplikacji Windows Phone	358
Utworzenie interfejsu użytkownika	359
Kod aplikacji pogodowej	360
Obsługa błędów w środowisku Rx.NET	363
Obsługa błędów połączeń internetowych w środowisku Rx.NET	363
Obsługa połączeń o niskiej jakości w aplikacji WeatherRx	365
Obsługa wywołań równoległych w środowisku Rx.NET	367
Podsumowanie	370
Rozdział 19. Bezpieczeństwo	371
Bezpieczeństwo aplikacji	371
Windows Phone Marketplace	372
Dostarczanie aplikacji do serwisu Windows Phone Marketplace	373
Izolowane środowisko pracy i menedżer uruchamiania	375
Bezpieczeństwo komunikacji sieciowej	375
Zabezpieczanie połączeń za pomocą protokołu SSL	376
Testowanie połączenia SSL	376
Utworzenie samopodpisanego certyfikatu	377
Eksportowanie samopodpisanych certyfikatów	380
Instalacja samopodpisanego certyfikatu w telefonie	382
Bezpieczeństwo danych	383
Wykorzystanie algorytmów HMACSHA1 i HMACSHA256	384
Wykorzystanie algorytmów Rfc2898DeriveBytes i AES	
do szyfrowania informacji	386
Fizyczne zabezpieczenie telefonu	390
Odnajdowanie	390
Dzwonienie	390
Blokowanie i wyświetlanie komunikatów	390
Usuwanie danych	390
Lista zgodności	391
Spełnienie wymagań certyfikacyjnych	391
Aplikacja musi się składać z kodu MSIL	391
Aplikacja nie może zawierać kodu krytycznego pod względem bezpieczeństwa	393
Podsumowanie	394
Skorowidz	395

ROZDZIAŁ 2



Tworzenie aplikacji Windows Phone 7

Celem niniejszego rozdziału jest przedstawienie wszystkich komponentów niezbędnych do rozpoczęcia prac nad aplikacjami Windows Phone 7. Zaprezentowane zostały tutaj narzędzia, takie jak emulator Windows Phone, oraz środowiska Visual Studio 2010 i Microsoft Expression Blend 4. Opisowi programów towarzyszą wskazówki informujące o tym, w jaki sposób można utworzyć pierwszą aplikację Windows Phone.

Jednak przed rozpoczęciem prac projektowych trzeba pobrać i zainstalować niezbędne oprogramowanie. Zadanie to jest tematem kolejnego podrozdziału.

Przygotowanie środowiska programistycznego

Najnowszypakiet instalacyjny narzędzi programistycznych dla systemu Windows Phone 7 można pobrać ze strony http://create.msdn.com/en-us/home/getting_started. Pobrany pakiet oprogramowania (*vm_web.exe*) powinien zainstalować następujące komponenty:

- komercyjną lub darmową wersję programu Visual Studio 2010 — środowiska programistycznego;
- emulator urządzenia Windows Phone — wykorzystywany do uruchamiania i testowania aplikacji Windows Phone;
- Silverlight for Windows Phone — platformę Silverlight przeznaczoną dla systemu Windows Phone, bazującą na technologii Silverlight 3;
- Microsoft Expression Blend for Windows Phone — narzędzie do projektowania interfejsu użytkownika;
- XNA Game Studio 4 — środowisko do projektowania gier.

Po zainstalowaniu pakietu narzędziowego Windows Phone można przystąpić do budowania pierwszej aplikacji dla tego systemu. Proces ten został opisany w kolejnych podrozdziałach.

Utworzenie pierwszej aplikacji Windows Phone 7

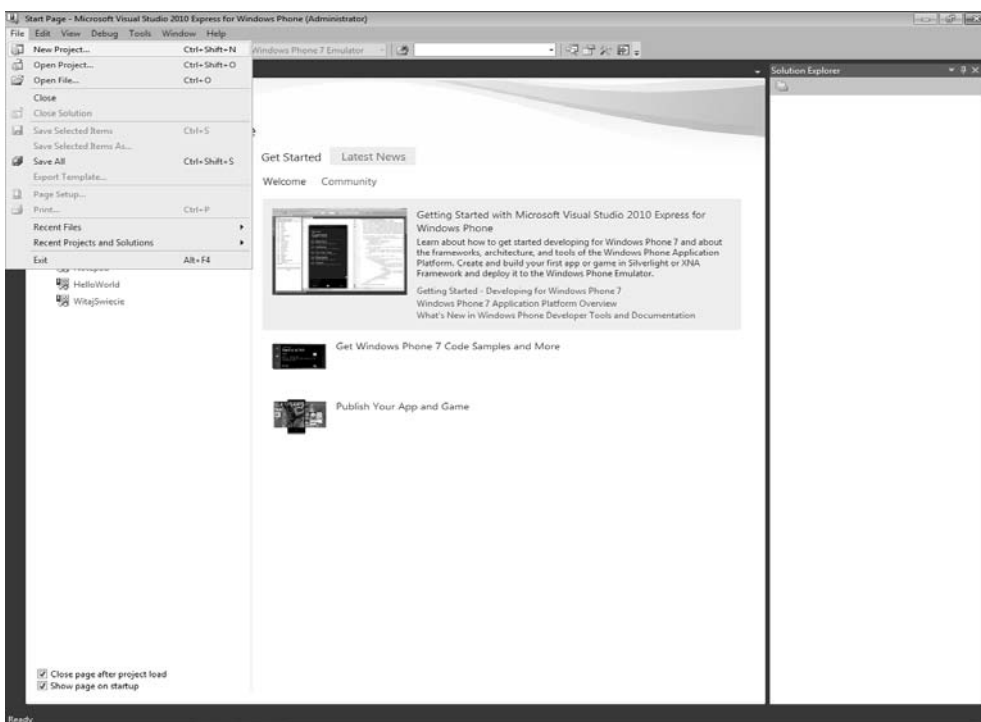
W tej części rozdziału został opisany tok postępowania prowadzącego do powstania prostej aplikacji bazującej na rozwiązaniu Silverlight. Podczas prac projektowych wykorzystywane będą takie narzędzia jak Visual Studio 2010 Express for Windows Phone, Windows Phone 7 Emulator oraz niektóre kontrolki z pakietu Windows Phone Silverlight. W dalszej części bieżącego rozdziału zostało również opisane

użycie aplikacji Blend do opracowania kontrolki Silverlight. Ostateczny wygląd przygotowywanej aplikacji został pokazany na rysunku 2.19, znajdującym się na końcu rozdziału. Kliknięcie przycisku *OK* powoduje wyświetlenie napisu *Witaj świecie!* w polu edycyjnym. Pierwszym etapem procedury tworzenia aplikacji jest utworzenie projektu w środowisku Visual Studio 2010.

Tworzenie projektu Windows Phone

Aby rozpocząć prace programistyczne, trzeba uruchomić środowisko Visual Studio Express 2010 i utworzyć nowy projekt.

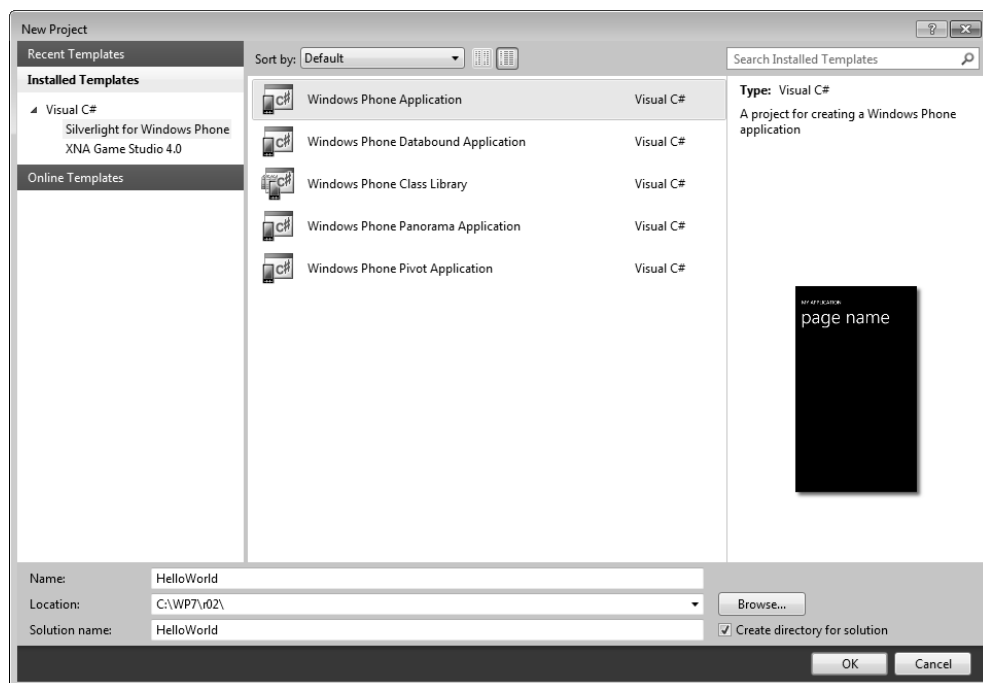
1. Aby uruchomić Visual Studio 2010 Express, wybierz z menu *Start* opcje *Wszystkie programy/Microsoft Visual Studio 2010 Express/Microsoft Visual Studio 2010 Express for Windows Phone*.
2. Utwórz nowy projekt, wybierając z menu środowiska Visual Studio opcje *File/New Project* (jak pokazano na rysunku 2.1).



Rysunek 2.1. Tworzenie nowego projektu

3. Na liście szablonów języka C#, widocznej w oknie dialogowym *New Project*, wskaż pozycję *Windows Phone Application* (zgodnie z rysunkiem 2.2).

■ **Uwaga:** Visual Studio udostępnia trzy różne szablony projektów odnoszące się do aplikacji Windows Phone. Szablon *Windows Phone Application* odpowiada jednostronicowej aplikacji. Szablon *Windows Phone List Application* wykorzystuje kontrolkę *ListBox* oraz mechanizm nawigacji między stronami do generowania programów złożonych z wielu stron. Szablon *Windows Phone Class Library* znajduje zastosowanie podczas tworzenia własnych bibliotek klas, które następnie są wykorzystywane w innych projektach Windows Phone.



Rysunek 2.2. Tworzenie nowej aplikacji Silverlight dla systemu Windows Phone

4. Zmień nazwę przygotowywanej aplikacji na *HelloWorld* (właściwość *Name*), jak pokazano na rysunku 2.2. Na tym etapie można również wskazać (inną niż domyślna) lokalizację projektu. Wystarczy zmienić ścieżkę dostępu w polu *Location*.
5. Kliknij przycisk *OK* w oknie dialogowym *New Project* — Visual Studio 2010 utworzy nowy projekt z komponentami przedstawionymi na rysunku 2.3.

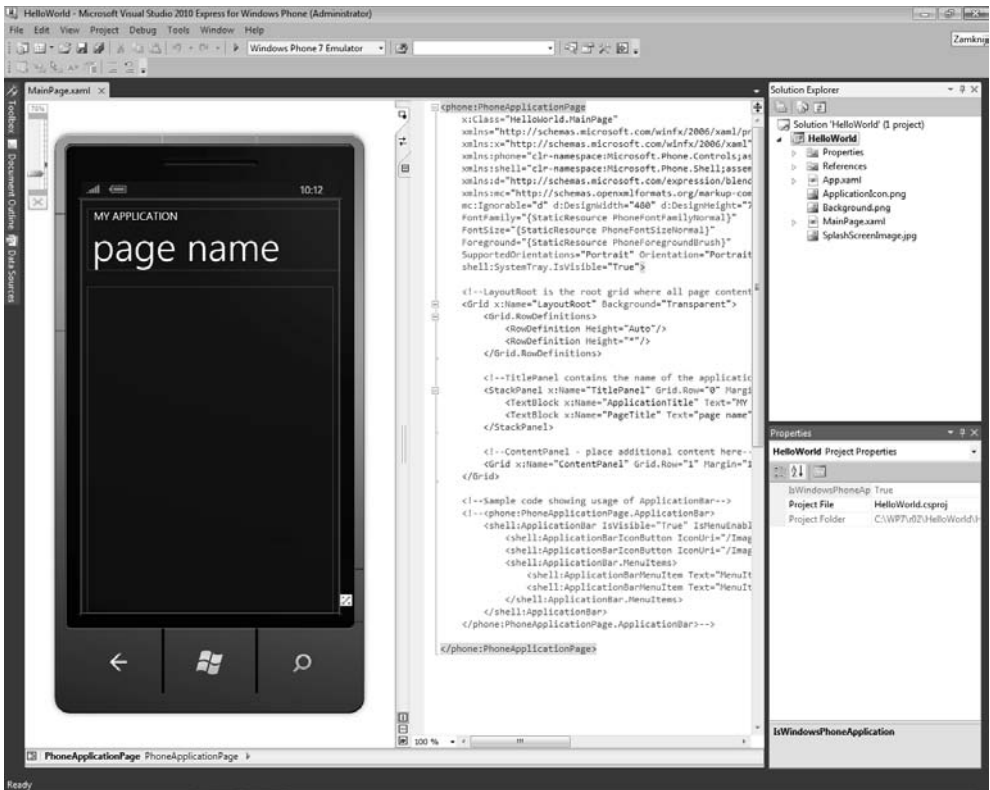
W obszarze projektowym domyślnie umieszczane są dwie kontrolki *TextBox* — widoczne z lewej strony rysunku 2.3.

Projekt jest gotowy do dalszego przetwarzania. Teraz można go uzupełnić o odpowiednie funkcje oraz elementy interfejsu użytkownika. Dalsze działania rozpoczniemy od przygotowania interfejsu użytkownika. W tym celu umieścimy w czarnym polu kontrolki, które użytkownik będzie mógł kliknąć i które wyświetlą odpowiedni tekst.

Wykorzystanie kontrolki Silverlight

Kolejny etap działań polega na dodaniu kontrolki Silverlight do aplikacji Windows Phone, która została utworzona w poprzednim kroku. Opisywane w tym punkcie rozwiązanie stanowi przykład wykorzystania właściwości kontrolki do automatycznego skalowania i pozycjonowania elementów programu niezależnie od tego, czy telefon pracuje w trybie portretowym (ang. *portrait mode*), czy krajobrazowym (ang. *landscape mode*).

1. Pracując w trybie *Design*, kliknij komponent *TextBlock* o treści *MY APPLICATION* i zmień właściwość *Text* z *MY APPLICATION* na *Aplikacja HelloWorld* (pole właściwości jest dostępne w oknie *Properties* widocznym w prawej dolnej części okna Visual Studio). Wprowadzony tekst jest automatycznie wyświetlany w obszarze projektowym, jak pokazano na rysunku 2.4.

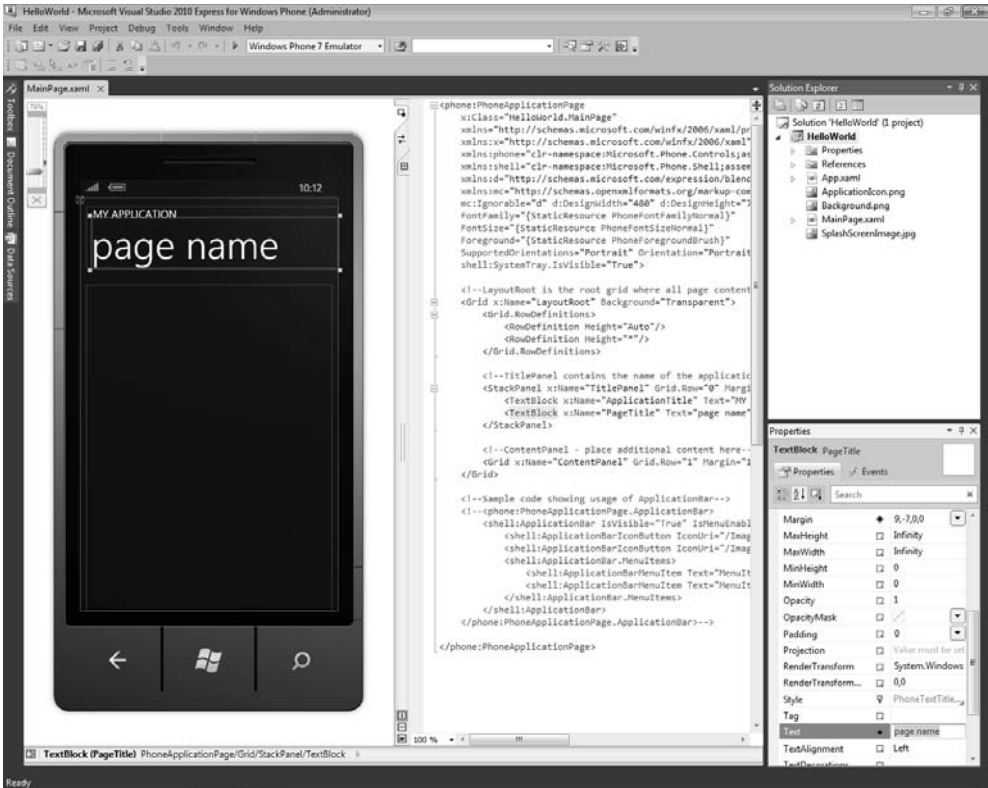


Rysunek 2.3. Projekt Silverlight HelloWorld przeznaczony dla systemu Windows Phone

2. Otwórz panel *Toolbox* środowiska Visual Studio. Z niego zostaną pobrane kontrolki składające się na interfejs aplikacji *HelloWorld*. Jeśli panel *Toolbox* nie jest widoczny, można go uaktywnić przez wybranie z menu opcji *View*, a następnie *Toolbox*. W wyniku tej operacji w lewej części ekranu powinien zostać wyświetlony pionowy panel z kontrolkami (zgodnie z rysunkiem 2.5).

Panel *Toolbox* zawiera komponenty Windows Phone, które są rozpowszechniane wraz z zainstalowanymi wcześniej narzędziami programistycznymi. W dalszej części książki poszczególne z nich będą wykorzystywane do budowania coraz bardziej wyrafinowanych interfejsów użytkownika. Aby skorzystać z któregośkolwiek z elementów, wystarczy przeciągnąć go do obszaru projektowego aplikacji Windows Phone.

3. Tworzenie interfejsu aplikacji *HelloWorld* rozpoczniemy od dodania komponentu *TextBox*, odpowiedzialnego za wyświetlanie tekstu. W tym celu należy przeciągnąć kontrolkę *TextBox* z panelu *Toolbox* do obszaru projektowego i umieścić ją bezpośrednio pod elementem *TextBlock* wyznaczającym nagłówek strony. Kontrolka upuszczona w obszarze projektowym jest automatycznie zaznaczana. Można więc od razu zmieniać wartości poszczególnych jej właściwości w oknie *Properties* (jeśli okno *Properties* nie jest widoczne, należy wybrać opcję *View*, a następnie *Properties Window*).
 - a) Ustaw właściwości *Width* i *Height* na *Auto*.
 - b) Ustaw właściwość *HorizontalAlignment* na *Stretch*.
 - c) Ustaw właściwość *VerticalAlignment* na *Top*.



Rysunek 2.4. Zmiana nazwy widocznej w nagłówku aplikacji

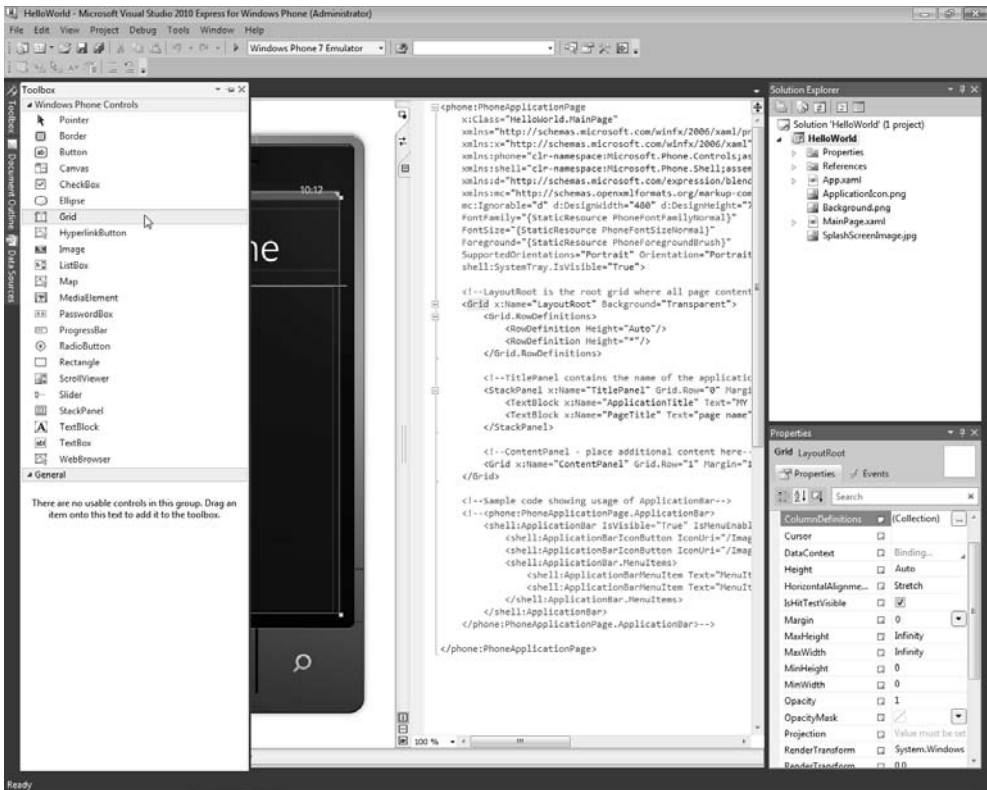
- d) Zmień rozmiar kontrolki TextBox tak, aby z jej prawej strony pozostało wolne miejsce na przycisk z napisem OK.
- e) Zmień nazwę komponentu (właściwość Name) na txtMessage.

Gdy zostaną wykonane zadania wymienione w podpunktach a, b, c i d, w edytorze XAML powinien być widoczny następujący kod:

```
<TextBox Height="Auto" Margin="0,55,166,0" Name="txtMessage" Text="TextBlock"
VerticalAlignment="Top" HorizontalAlignment="Right" Width="290" />
```

W podpunkcie b właściwość HorizontalAlignment (rozmiesszczenie w poziomie) została przypisana wartość Stretch (rozciągaj), tak aby po obróceniu telefonu do widoku poziomego (krajobrazowego) kontrolka TextBox została automatycznie rozciągnięta i wypełniła całą powstałą przestrzeń. Właściwości Width (szerokość) i Height (wysokość) otrzymały wartość Auto po to, by komponent TextBox automatycznie zmieniał wymiary wraz ze zmianą wielkości czcionki. Z kolei ustawienie właściwości VerticalAlignment (rozmiesszczenia w pionie) na Top (góra) sprawia, że pole tekstowe zawsze będzie wyrównywane do górnej krawędzi okna. Aby odwołać się do kontrolki z poziomu kodu, wystarczy użyć jej nazwy txtMessage.

4. Uzupełnij interfejs użytkownika o przycisk OK. W tym celu przeciągnij kontrolkę Button z panelu Toolbox i upuść ją po prawej stronie pola edycyjnego. Następnie zmień właściwości przycisku w oknie Properties.
 - a) Ustaw właściwość Content na OK.
 - b) Ustaw właściwość HorizontalAlignment na Right.



Rysunek 2.5. Pasek narzędziowy środowiska Visual Studio zawierający kontrolki Windows Phone

- c) Ustaw właściwość VerticalAlignment na Top.
- d) Zmień nazwę (właściwość Name) na btnOk.

Gdy zostaną wykonane zadania z podpunktów a, b i c, w edytorze XAML powinien zostać wyświetlony kod zgodny z poniższym listingiem. Przypisanie właściwości HorizontalAlignment wartości Right spowoduje, że przycisk zawsze będzie zachowywał swoje położenie względem prawej krawędzi okna.

```
<Button Content="OK" Height="72" HorizontalAlignment="Right" Margin="308,31,0,0"
Name="btnOk" VerticalAlignment="Top" Width="160" />
```

5. Projekt interfejsu graficznego wydaje się ukończony. W edytorze XAML powinien być widoczny kod siatki (ang. *grid*) z dodanymi kontrolkami:

```
<Grid x:Name="ContentGrid" Grid.Row="1">
<TextBox Height="Auto" Margin="0,55,166,0" Name="txtMessage" Text="TextBlock"
↳ VerticalAlignment="Top" HorizontalAlignment="Right" Width="290" />
<Button Content="OK" Height="72" HorizontalAlignment="Right" Margin="308,31,0,0"
↳ Name="btnOk" VerticalAlignment="Top" Width="160" />
</Grid>
```

6. Ostateczny wygląd okna aplikacji (po dodaniu kontrolki TextBox i Button) został pokazany na rysunku 2.6.



Rysunek 2.6. Końcowy wygląd interfejsu aplikacji HelloWorld

Utworzenie kodu pierwszej aplikacji Windows Phone

Celem operacji opisywanej w tym punkcie jest utworzenie kodu C#, który obsłuży zdarzenie kliknięcia przycisku i wpisze do kontrolki TextBox o nazwie txtMessage tekst Witaj świecie!.

1. Aby dodać instrukcje przeznaczone do realizacji po kliknięciu przycisku OK, należy dwukrotnie kliknąć przycisk OK w obszarze projektowym. Środowisko Visual Studio automatycznie otworzy plik *MainPage.xaml.cs*, dodając do niego metodę `btnOk_Click`. W obszarze kodu metody można wówczas zapisać instrukcje odpowiadające za obsługę zdarzenia.

```
using System.Windows;
using Microsoft.Phone.Controls;
```

```
namespace HelloWorld
{
```

```
    public partial class MainPage : PhoneApplicationPage
    {
        public MainPage()
```

```
        {
            InitializeComponent();
            //Ustawienie właściwości SupportedOrientations pozwala na określenie
            //zachowania telefonu podczas obracania go. Dzięki niej poziome ustawienie
            //telefonu spowoduje wyświetlenie aplikacji w trybie krajobrazowym.
            SupportedOrientations = SupportedPageOrientation.PortraitOrLandscape;
```



```

    }

    private void btnOk_Click(object sender, RoutedEventArgs e)
    {

    }
}

```

2. Kod pliku *MainPage.xaml* jest również automatycznie uzupełniany o procedurę obsługi zdarzenia *Click* przycisku *OK*.

```

<Button Content="OK" Height="70"
    HorizontalAlignment="Right" Margin="0,155,-4,0"
    Name="btnOk" VerticalAlignment="Top" Width="160" Click="btnOk_Click" />

```

3. Treść samej metody *btnOk_Click* należy zmienić zgodnie z poniższym przykładem.

```

private void btnOk_Click(object sender, RoutedEventArgs e)
{
    txtMessage.Text = "Witaj świecie!";
}

```

Uruchomienie pierwszej aplikacji Windows Phone

Po zakończeniu projektowania aplikacji *Hello World* nadchodzi czas na jej skompilowanie i uruchomienie w środowisku emulatora systemu Windows Phone 7.

1. Aby skompilować kod, wybierz z menu Visual Studio pozycję *Build/Build Solution*.
2. Uruchomienie aplikacji następuje po wybraniu opcji *Debug/Start Debugging*.
3. Po wyświetleniu okna emulatora i kliknięciu przycisku *OK* na ekranie powinien się pojawić napis *Witaj świecie!* (zgodnie z rysunkiem 2.7).



Rysunek 2.7. Aplikacja *HelloWorld* w oknie emulatora *Windows Phone 7*

- Emulator telefonu udostępnia opcję obrócenia urządzenia. Przycisk opcji znajduje się na pasku zadań pokazanym na rysunku 2.8.



Rysunek 2.8. Opcja obrócenia telefonu w emulatorze Windows Phone 7

Po wybraniu widoku poziomego kontrolka TextBox jest automatycznie rozciągnięta, tak aby w pełni wykorzystać obszar wyświetlacza w trybie krajobrazowym. Efekt ten został zaprezentowany na rysunku 2.9.



Rysunek 2.9. Widok aplikacji HelloWorld w poziomym ułożeniu telefonu

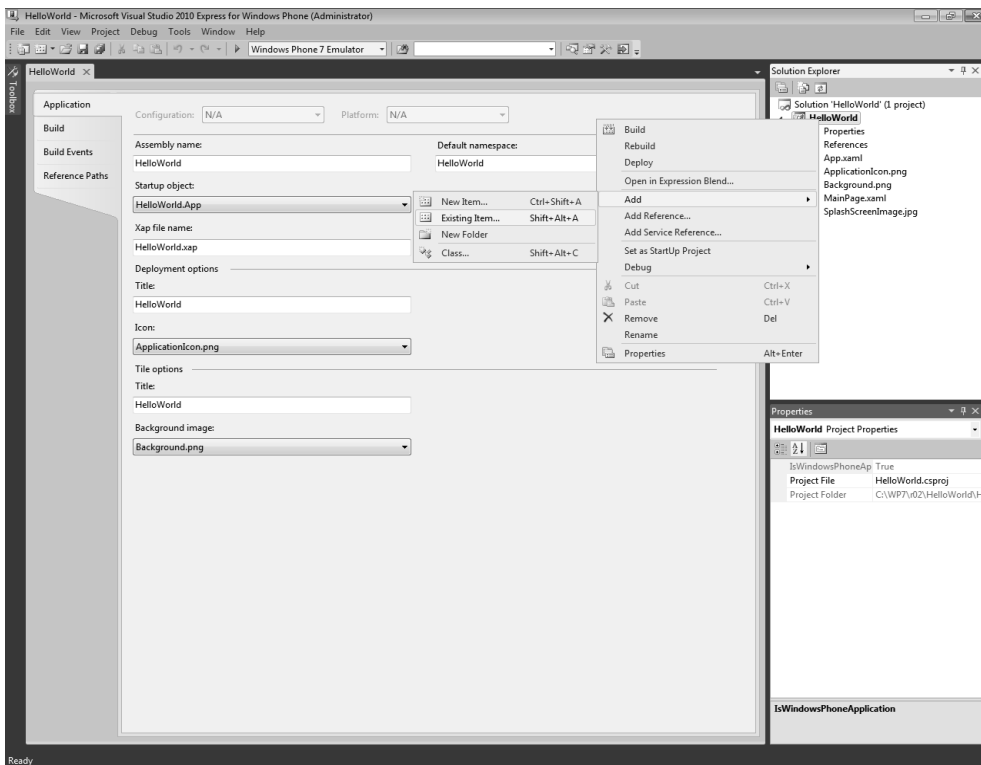
- Aby zakończyć pracę aplikacji, można wybrać z menu opcje *Debug/Stop Debugging*.

-
- **Wskazówka:** Uruchomienie emulatora Windows Phone 7 zajmuje sporo czasu. Dlatego nie powinno się go wyłączać, jeśli nie jest to konieczne. Aby zatrzymać testowaną aplikację, można wybrać z menu Visual Studio opcje *Debug/Stop Debugging* bez przerywania pracy emulatora. Taki sposób postępowania pozwala na ponowne załadowanie aplikacji (po jej uruchomieniu) do działającego już emulatora.
-

Dostosowywanie pierwszej aplikacji Windows Phone

W tym punkcie rozdziału zostały opisane czynności prowadzące do ustawienia ikony aplikacji Windows Phone 7 (widocznej w interfejsie użytkownika) oraz zmiany nazwy aplikacji.

- W panelu *Solution Explorer* kliknij prawym przyciskiem myszy pozycję projektu *HelloWorld*, a następnie wybierz opcję *Add Existing Item...* (zgodnie z rysunkiem 2.10).
- Wskaż folder z rozpakowanymi przykładami kodu i znajdź plik `\r01\Assets\HelloWorldIcon.png`. Ikonami aplikacji Windows Phone 7 mogą być dowolne pliki formatu PNG o wymiarach 62×62 piksele. Domyślnie po utworzeniu projektu wykorzystywany jest do tego celu rysunek *ApplicationIcon.png*.
- Kliknij prawym przyciskiem myszy projekt *HelloWorld* i wybierz opcję *Properties*.
- Uaktywnij zakładkę *Application*.



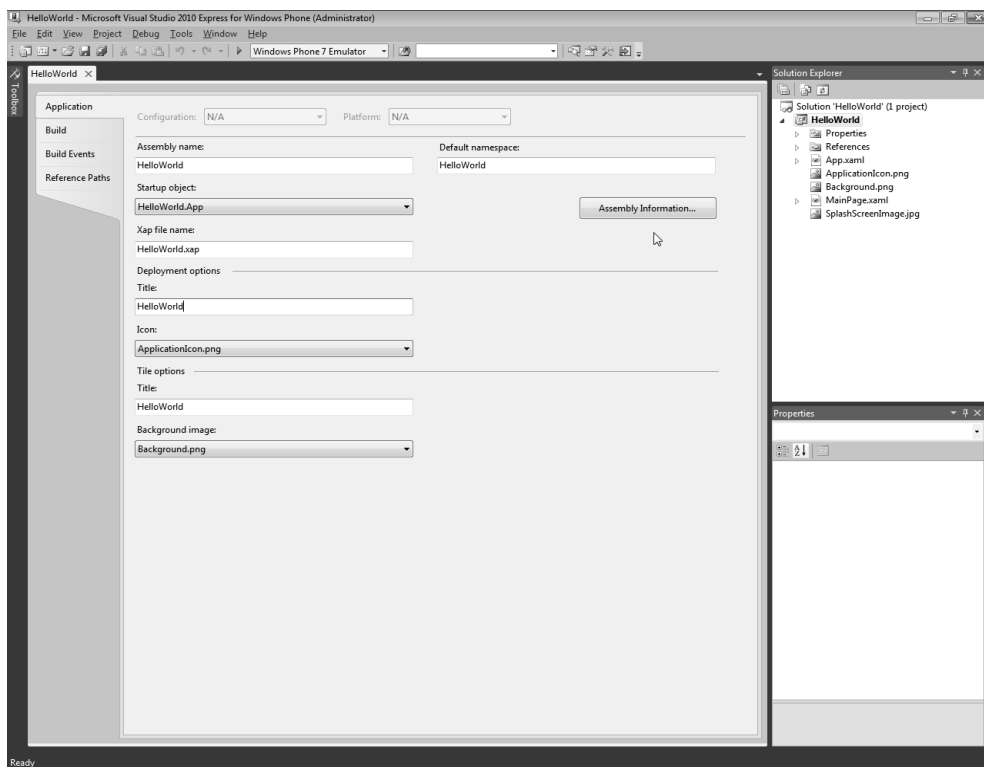
Rysunek 2.10. Dodawanie pliku do projektu Visual Studio

5. W sekcji *Deployment* zmień ikonę na *HelloWorldIcon.png*.
6. Zmień tytuł aplikacji na *HelloWorld*, jak pokazano na rysunku 2.11.
7. Naciśnij klawisz *F5*, aby uruchomić aplikację.
8. W czasie działania aplikacji kliknij klawisz powrotu znajdujący się na obudowie symulowanego telefonu. Wygląd klawisza został przedstawiony na rysunku 2.12.
9. Na liście zainstalowanych aplikacji powinna być widoczna pozycja *HelloWorld* (pokazana na rysunku 2.13).

Zmiana wyglądu aplikacji

Do zmiany sposobu prezentacji interfejsu użytkownika bazującego na kodzie XAML (technologii Silverlight) można wykorzystać zarówno środowisko Visual Studio, jak i program Microsoft Expression Blend 4. Druga z wymienionych aplikacji udostępnia jednak znacznie bardziej zaawansowane narzędzia do tworzenia grafiki i animacji niż Visual Studio. Dlatego warto poznać podstawowe zasady działania aplikacji Blend oraz techniki pracy ułatwiające przygotowanie wizualnej strony kontrolki.

1. Uruchom środowisko Microsoft Expression Blend 4, wybierając z menu *Start* opcje *Wszystkie Programy/Microsoft Expression/Microsoft Expression Blend*. Okno aplikacji zostało pokazane na rysunku 2.14.
2. W oknie powitalnym kliknij przycisk *Close*.
3. Wybierz z menu opcje *File/Open Project* i wskaż plik utworzonego wcześniej projektu *HelloWorld* (zgodnie z rysunkiem 2.15).



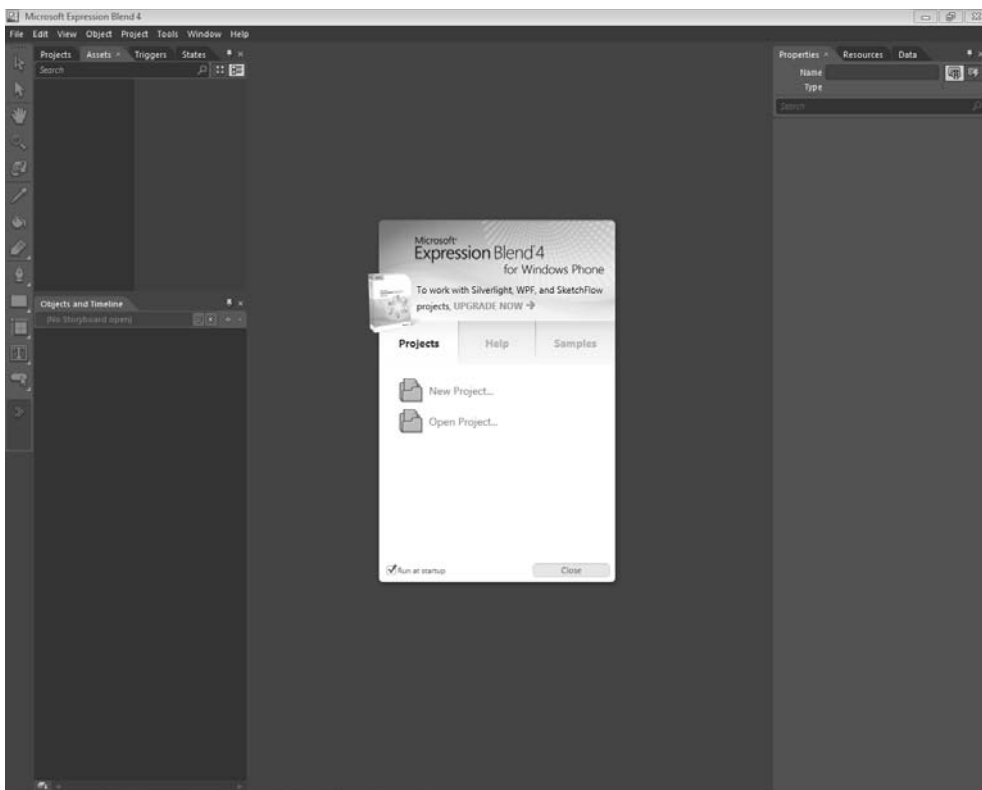
Rysunek 2.11. Zmiana tytułu i ikony aplikacji



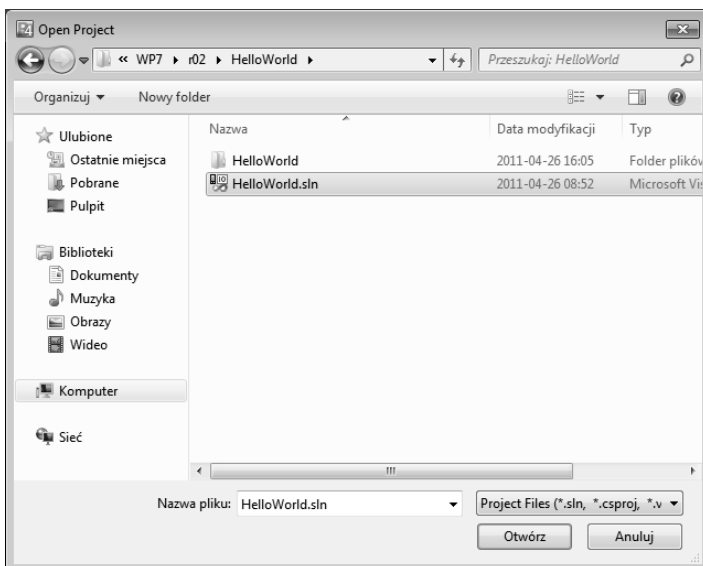
Rysunek 2.12. Przycisk powrotu w telefonie Windows Phone 7



Rysunek 2.13. Lista aplikacji systemu Windows Phone 7

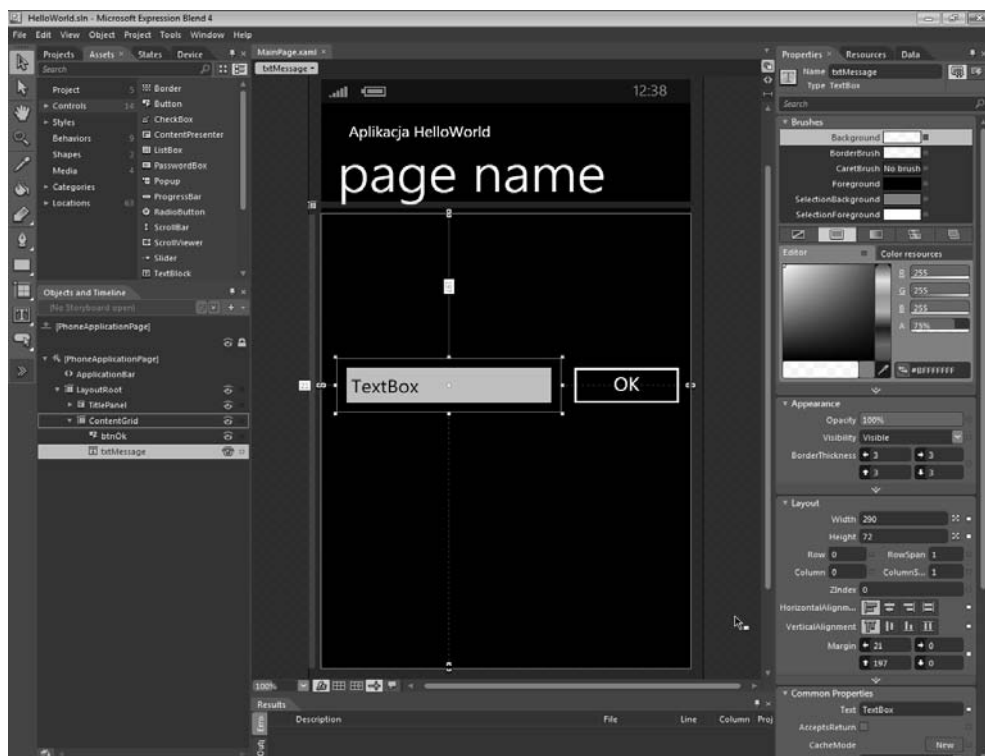


Rysunek 2.14. Aplikacja Microsoft Expression Blend 4



Rysunek 2.15. Otwieranie przygotowanego wcześniej projektu w programie Blend 4

- Po otwarciu projektu kliknij kontrolkę `TextBox`. W oknie *Properties* zostaną wyświetlone właściwości, które można zmieniać za pomocą aplikacji Blend. Jeżeli okno *Properties* nie jest wyświetlane automatycznie, należy wybrać z menu opcję *Window*, a następnie *Properties*. Wygląd okna został przedstawiony na rysunku 2.16.



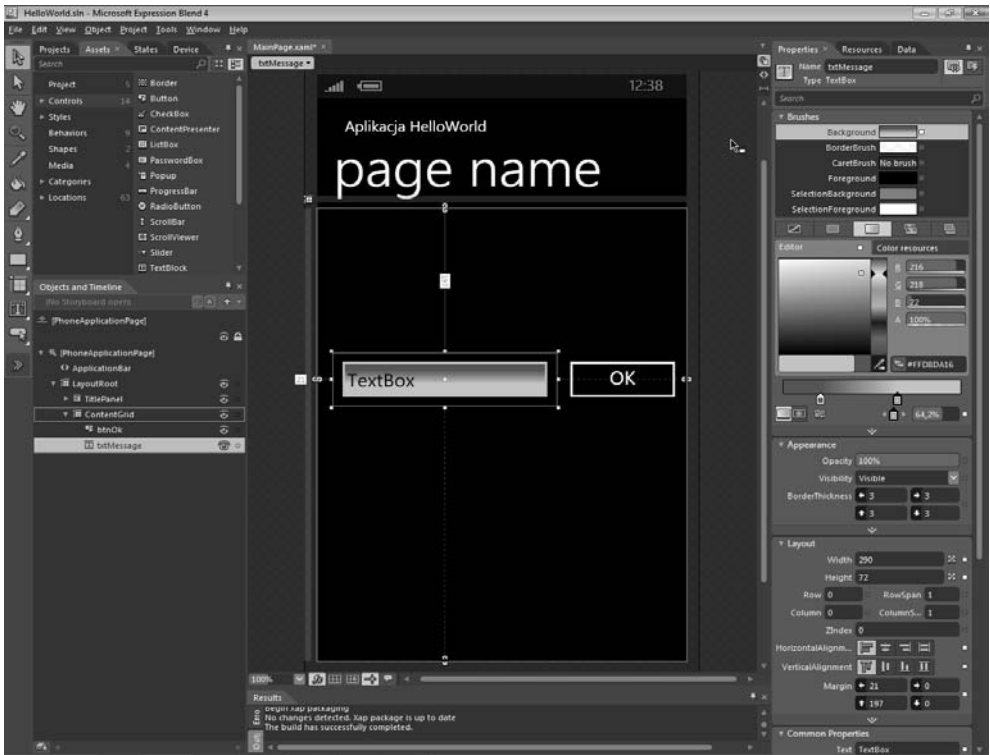
Rysunek 2.16. Okno właściwości zaznaczonej kontrolki

- W oknie *Properties* wskaź kategorię *Brushes* (pędzle). Wybierz *Background* (tło), a następnie *Gradient Brush* (pędzel gradientowy). Po tej operacji edytor kolorów zostaje przełączony w taki tryb pracy, który umożliwia ustalenie gradientowego wypełnienia tła kontrolki `TextBox`.
- Jako pierwszy kolor gradientu wybierz niebieski i ustaw próg przejścia na 21%. Jako drugi kolor wskaź żółty i wyznacz próg na poziomie 64%. Ustawienie to zostało pokazane na rysunku 2.17.
- Wybierz z menu opcje *Project/Run Project*.
- Jeśli na ekranie zostanie wyświetlone okno wyboru urządzenia docelowego (pokazane na rysunku 2.18), zaznacz pozycję *Windows Phone 7 Emulator*.
- Kliknij *OK*. Aplikacja *HelloWorld* powinna zostać uruchomiona z interfejsem użytkownika pokazanym na rysunku 2.19.

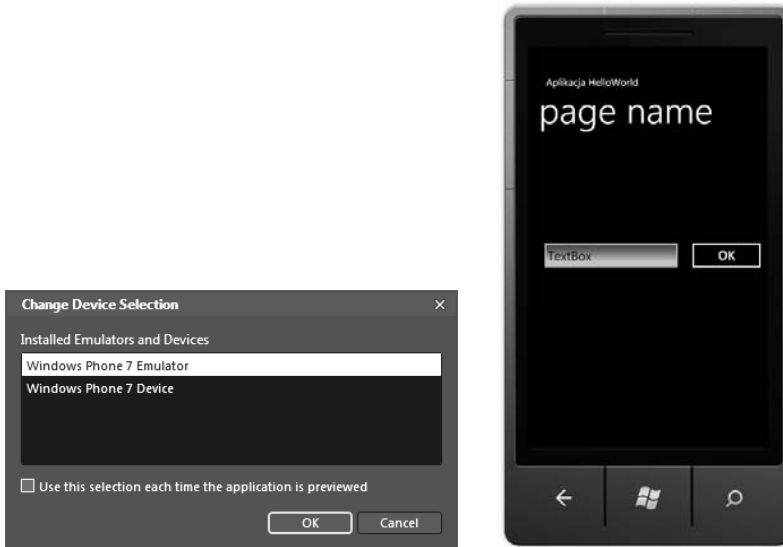
Podsumowanie

W tym rozdziale zostały opisane czynności pozwalające na przygotowanie środowiska projektowego Windows Phone 7. Ponadto przedstawiono proces tworzenia aplikacji przy użyciu środowiska Visual Studio 2010, emulatora telefonu oraz programu graficznego Microsoft Expression Blend.

Tematem kolejnego rozdziału jest budowa aplikacji współdziałającej z systemem Microsoft SQL Azure, który umożliwia przechowywanie danych.



Rysunek 2.17. Nałożenie gradientowego wypełnienia na tło kontrolki TextBox



Rysunek 2.18. Okno zmiany urządzenia docelowego

Rysunek 2.19. Wygląd aplikacji HelloWorld po zmianach dokonanych za pomocą programu Blend 4

Skorowidz

.NET, platforma, 25, 27
wersja kompaktowa, 239
.NETCF, platforma, 239

A

Accelerometer, klasa, 137
 ReadingChanged, zdarzenie, 137
AccelerometerFailedException, 109, 113, 114
Advanced Encryption Standard, *Patrz* algorytmy
 szyfrujące, AES
AesManaged, klasa, 387
akcelerometr, 133
 obliczanie odległości, 134
 pobieranie danych, 137
 pochylenie i przechylenie, 135
 położenie i przemieszczanie, 133
 użycie, 137
 zasada pomiaru, 134
algorytmy szyfrujące, 383
 AES, 383, 386, 389
 HMACSHA1, 383, 384
 HMACSHA256, 383, 384
 Rfc2898DeriveBytes, 383, 386
animacja
 tworzenie, 354, 355, 356, 357
 z podziałem czasu, 355
aplikacja
 bezpieczeństwo, 371, 373
 cykl pracy, 207
 cykl życia, 32, 199
 pakowanie, 119, 121
 stan, 203
 zdarzenia, 200, 203
aplikacje testowe, 211
 budowanie, 218
 połączenie z usługą sieciową, 222
 symulowanie, 215

App Hub, 31
application bar, *Patrz* pasek aplikacji
Application Policies, 117, 118
Application_Activated, 100, 200, 203
Application_Closing, 100, 200, 203
Application_Deactivated, 100, 200, 203
Application_Launching, 100, 200, 203, 206
Application_UnhandledException, 100
architektura
 klasy enterprise, 48
 trójwarstwowa, 47
ArgumentOutOfRangeException, 98

B

baza danych
 tabele, 57
 testowanie połączenia, 54, 55
 utworzenie, 50, 52
 zarządzanie, 56
bezpieczeństwo
 aplikacji, 371
 danych, 383
 komunikacji sieciowej, 375
 niezaprzeczalność, 372
biblioteka obrazów, *Patrz* Pictures Library, folder
Bing Maps, 30, 259
 instalacja pakietu SDK Bing Maps, 266
 kontrolka, 268, 269, 273
 rejestracja w serwisie, 266, 267
 wyświetlanie pozycji, 273
BingMap, kontrolka, 273
 SetView, metoda, 273

C

Camera Roll, folder, 302
CameraCaptureTask, 197, 298, 299, 300
 Completed, zdarzenie, 298

centrum, 208

- App, *Patrz* centrum, Marketplace Games, 208
- koncepcja, 25
- Marketplace, 25, 208
- Music and Video, 25, 208, 209
- Office, 208
- People, 25, 208
- Pictures, 208

certyfikacja, wymagania, 391

chmura obliczeniowa, 47

- wykorzystanie, 30

choosers, *Patrz* funkcje wyboru

cloud computing, *Patrz* chmura obliczeniowa

CompositionTarget.Rendering, zdarzenie, 288

Content Policies, 119

crossdomain.xml, 178

CultureInfo, klasa, 231, 233, 234, 238

- CurrentCulture, właściwość, 233
- DisplayName, właściwość, 233

cykl pracy aplikacji

- zalecane techniki, 207

cykl życia aplikacji, 32, 199

- analiza zdarzeń, 199, 200

D

dane, bezpieczeństwo, 383

daty, różnice kulturowe, 230, 233

Dispatcher, klasa, 353

- BeginInvoke, metoda, 265

DRM, technika, 283

dźwięki, 291

E

efekty przejścia, 192

- tilt, 193

ekran, ułożenie, 181

- tryb krajobrazowy, 181
- tryb portretowy, 181

elipsa, wyświetlenie, 183

EmailAddressChooserTask, 197, 298

EmailComposeTask, 196, 298

emulator telefonu, 29

- obrócenie urządzenia, 41
- uruchomienie, 40, 41

Entity Framework, 49, 50, 60, 65

Execution Manager, *Patrz* menedżer wykonywania programu

Expression Blend, 28, 29

- tworzenie animacji, 354, 355, 356, 357
- zapoznanie z programem, 42

Extensible Application Markup Language, *Patrz* XAML

F

funkcje uruchamiania, 195, 196, 298

- wykorzystanie, 197
- zestawienie, 196, 298

funkcje wyboru, 195, 196, 298

- wykorzystanie, 197
- zestawienie, 197, 298

G

GeoCodeService, 274, 275

GeoCoordinateWatcher, klasa, 266, 273

- DesiredAccuracy, właściwość, 260
- Movement Threshold, właściwość, 260, 271
- PositionChanged, zdarzenie, 260, 271, 272
- StatusChanged, zdarzenie, 260, 271, 272
- TryStart, metoda, 260, 273

GeoPositionChangedEventArgs, 264

globalizacja, 230

GPS, 259

- symulowanie odczytów, 264

gradient, tworzenie, 45

H

HttpChannelNotification, klasa, 318

- BindToShellTile, metoda, 318
- BindToShellToast, metoda, 318
- Close, metoda, 318
- Find, metoda, 318
- Open, metoda, 318
- ShellToastNotificationReceived, zdarzenie, 322
- UnbindToShellTile, metoda, 318
- UnbindToShellToast, metoda, 318

HttpRequest, klasa, 325

hub, *Patrz* centrum

I

IIS, 283, 377

INotifyPropertyChanged, interfejs, 80

- implementacja, 86

internacjonalizacja, 229, 230, 233, 243

Internet Explorer Mobile, 176, 177

Internet Information Services, *Patrz* IIS

Invalid cross thread access, 137, 142, 147

ISO 8601, 230

isolated storage, *Patrz* izolowana pamięć masowa

- IsolatedStorage.DeleteDirectory, metoda, 251
- IsolatedStorage.DeleteFile, metoda, 251
- IsolatedStorageFile, przestrzeń nazw, 245
- IsolatedStorageFileStream, strumień, 251
- IsolatedStorageSettings, przestrzeń nazw, 245, 253

IsScriptEnabled, właściwość, 177
 IsTrial, metoda, 211, 212, 215
 IValueConverter, interfejs, 77, 78
 implementacja, 78
 izolowana pamięć masowa, 172, 245, 246, 257
 korzystanie, 246
 sprawdzenie dostępności, 251
 izolowana pamięć ustawień, 253, 257
 izolowana przestrzeń plików, 245
 izolowana przestrzeń ustawień, 245
 izolowane środowisko pracy, 375

J

język aplikacji, 229

K

kafelek aplikacji, 314
 kął
 pochylenia, 135, 137
 przechylenia, 135, 137
 klawiatura, podłączenie do emulatora, 376
 klisza fotograficzna, *Patrz* Camera Roll, folder
 kod
 krytyczno-bezpieczny, 393
 krytyczny, 393
 przezroczysty, 393
 wymagania, 119
 kolekcje obserwacji, *Patrz* Observable
 kompozycje, 182
 bieżąca kompozycja, 186
 zmiana, 184
 komunikacja sieciowa, bezpieczeństwo, 375
 konwertery, 77
 krajobrazowy, tryb, 181

L

lambda, wyrażenia, 288, 349
 landscape mode, *Patrz* krajobrazowy, tryb
 launchers, *Patrz* funkcje uruchamiania
 LicenseInformation, klasa, 211, 212, 215, 216
 LINQ, 346, 368
 lokalizacja, 230
 pliki zasobów, 238, 239

M

MAC, 383, 384
 Marketplace, 23, 25, 115, 372
 aktualizacja aplikacji, 127, 129, 375
 API, 213
 certyfikacja aplikacji, 31, 120, 391

 bezpieczeństwo, 121, 373
 niezawodność, 120
 wydajność, 120
 wymagania, 117, 119
 założenia dotyczące przekazywanych treści, 119
 założenia funkcjonalne, 117, 118
 cykl publikacji rozwiązań, 116
 dystrybucja aplikacji, 115
 ochrona praw intelektualnych, 372
 opłata, 31, 32
 przesyłanie aplikacji, 121, 122, 123, 124, 125, 126, 127, 128, 373, 374
 raporty, 32
 tożsamość autora, 372
 wyszukiwanie aplikacji, 130
 MarketplaceDetailTask, 196, 213, 298
 MarketplaceHubTask, 196, 298
 MarketplaceReviewTask, 196, 213, 298
 MarketplaceSearchTask, 196, 298
 MediaElement, kontrolka, 281, 282, 291
 CanPause, właściwość, 288
 IsMuted, właściwość, 289
 Pause, metoda, 288
 Position, właściwość, 290
 Stop, metoda, 289
 MediaHistory, klasa, 209
 WriteAcquiredItem, metoda, 209
 WriteRecentPlay, metoda, 209
 MediaHistoryItem, klasa, 209
 MediaPlayerLauncher, klasa, 196, 281, 282, 291, 298
 Location, właściwość, 291
 menedżer wykonywania programu, 375
 Message Authentication Code, *Patrz* MAC
 Metro, interfejs, 31, 179, 180
 czcionki, 180
 wytyczne, 31
 Microsoft Advertising, usługa, 115
 Microsoft Azure, 30, 48, 49, 316
 Microsoft Intermediate Language, *Patrz* MSIL
 Microsoft Push Notification Service, 316, 317, 323
 punkt końcowy, 323
 Microsoft.Devices.Sensors, biblioteka, 138
 Microsoft.Devices.Sensors, przestrzeń nazw, 113, 137, 140, 146
 Microsoft.Phone.Controls.Maps, biblioteka, 268
 Microsoft.Phone.Controls.Maps, przestrzeń nazw, 270
 Microsoft.Phone.Marketplace, przestrzeń nazw, 215
 Microsoft.Phone.Notification, przestrzeń nazw, 318
 Microsoft.Phone.Reactive, przestrzeń nazw, 263
 Microsoft.Phone.Tasks, przestrzeń nazw, 196, 213, 286
 Microsoft.Xna.Framework.GamerServices, przestrzeń nazw, 215
 Model-View-ViewModel, *Patrz* MVVM
 Model-Widok-ModelWidoku, *Patrz* MVVM

MPNS, *Patrz* Microsoft Push Notification Service
 MSIL, 373, 391
 ms-text-size-adjust, właściwość, 176, 177
 MVVM, 48, 49

N

NavigationService, klasa, 190, 191
 nawigacja, 190
 NBC Olympics, 281
 nonrepudiation, *Patrz* bezpieczeństwo,
 niezaprzeczalność

O

Object-Relational Mapping, *Patrz* ORM
 Observable, 261, 263, 347
 odczytanie kolekcji, 348
 Subscribe, metoda, 348
 utworzenie kolekcji, 348
 ObserveOn, metoda, 353, 354
 obszar prezentacji, 176
 ochrona praw intelektualnych, 372
 odległość euklidesowa, 134
 odwzorowanie obiektowo-relacyjne, *Patrz* ORM
 OnError, zdarzenie, 363
 Opacity, właściwość, 157
 Orientation, właściwość, 182

P

pakiet instalacyjny, przygotowanie, 119, 121
 pakowanie, 119, 121
 Panorama, kontrolka, 187, 189
 wykorzystanie, 187
 PanoramaItem, kontrolka, 187, 189
 pasek aplikacji, 151, 152, 153, 180, 181
 globalny, 151, 153, 155
 ikony, 153
 liczba elementów, 152
 lokalny, 151, 153, 155
 obsługa zdarzeń, 160, 164
 tworzenie, 153
 wykorzystanie, 152
 pasek systemowy, 180, 181
 PayloadFormatError, 317
 persistence layer, *Patrz* warstwa zachowania danych
 PhoneApplicationFrame, kontrolka, 190
 PhoneApplicationPage, klasa, 203
 OnBackKeyPress, metoda, 204
 OnNavigatedFrom, metoda, 203
 OnNavigatedTo, metoda, 203
 PhoneApplicationPage, kontrolka, 190

PhoneCallTask, 196, 298
 PhoneNumberChooserTask, 197, 298
 PhotoChooserTask, 197, 298, 302
 Pictures Library, folder, 302
 pinezka, 270
 Pivot, kontrolka, 187, 189, 190
 PivotItem, kontrolka, 190
 plik, utworzenie, 251
 PNS, 118
 podkatalog, tworzenie, 251
 portrait mode, *Patrz* portretowy, tryb
 portretowy, tryb, 181
 POST, 317, 323
 nagłówki, 325
 powiadomienia Windows Phone, 313, 316
 architektura, 316
 cykl życia, 317
 kafelkowe, 313, 314, 315, 316, 327
 niesformatowane, 313, 315, 316, 330
 ograniczenie ilości, 342
 środowisko programistyczne, 318
 wykorzystanie usługi sieciowej, 335
 wyskakujące, 313, 314, 315, 316, 318
 zastosowania, 342
 pozycjonowania, usługi, 118, 259, 266, 274
 interfejs programistyczny, 260
 symulowanie, 261, 264
 uruchamianie, 273
 programowanie interaktywne, 346
 programowanie reaktywne, 346
 przecinek dziesiętny, różnice kulturowe, 230
 push notification, *Patrz* powiadomienia Windows Phone
 Push Notification Service, *Patrz* PNS
 Pushpin, 270

R

ramki, 190
 raw notification, *Patrz* powiadomienia Windows
 Phone, niesformatowane
 Reactive Extensions, 29, 137, 222, 264, 345, 346, 347, 353
 asynchroniczne pobieranie danych, 358, 362, 363
 obsługa błędów, 363
 obsługa wywołań równoległych, 367
 przeszukiwanie serwisu Flickr, 350
 Retry, metoda, 365
 Switch, metoda, 367
 TakeUntil, metoda, 367
 Timeout, metoda, 364
 wzorzec obserwatora, implementacja, 347
 Reflector .NET, 216
 przykład użycia, 216
 Resource.MergedDictionaries, właściwość, 182
 resx, pliki, 238, 239
 Rfc2898DeriveBytes, klasa, 387

RootFrame_NavigationFailed, 100, 101
 RouteService, 274
 Rx.NET, *Patrz* Reactive Extensions

S

salt, 383
 samodzielna usługa, 335
 samopodpisane certyfikaty, 377
 eksportowanie, 380
 instalacja w telefonie, 382
 tworzenie, 377
 Saved Pictures, folder, 302
 SaveEmailAddressTask, 196, 298
 SavePhoneNumberTask, 196, 298
 scenorys, 355
 SearchService, 274
 SearchTask, 196, 298
 self-hosted service, *Patrz* samodzielna usługa
 self-signed certificate, *Patrz* samopodpisane certyfikaty
 Silverlight, 27
 kontrolki, 35
 SimulateTrialMode, właściwość, 215
 Single Photo Viewer, aplikacja, 305
 SmsComposeTask, 196, 197, 298
 SPV, *Patrz* Single Photo Viewer, aplikacja
 SQL Azure, 30, 49, 50
 rejestracja w serwisie, 50
 testowanie połączenia, 54, 55
 tworzenie tabel, 57
 utworzenie bazy danych, 50, 52
 zapora sieciowa, 52, 54, 55
 zarządzanie bazą danych, 56
 SQL Server Management Studio, 56
 SSL, protokół, 376, 377
 testowanie połączenia, 376
 stan aplikacji, 203
 zarządzanie, 203, 206
 storyboard, *Patrz* scenorys
 StreamReader, klasa, 257
 StreamWriter, klasa, 251, 257
 strony internetowe
 wyświetlanie, 167, 169, 176
 zapisywanie w pamięci telefonu, 174
 system tray, *Patrz* pasek systemowy
 System.ComponentModel, przestrzeń nazw, 80
 System.Device.Location, przestrzeń nazw, 260, 263, 270
 System.IO, przestrzeń nazw, 249
 System.IO.IsolatedStorage, przestrzeń nazw, 80, 249, 256
 System.IO.Path.Combine, metoda, 251
 System.Linq, przestrzeń nazw, 80
 System.Threading, przestrzeń nazw, 263
 System.Windows.Data, przestrzeń nazw, 78

Ś

środowisko programistyczne, przygotowanie, 33

T

telefon
 zabezpieczenie, 390
 TextBox, kontrolka, 36
 Height, właściwość, 37
 HorizontalAlignment, właściwość, 37
 TextWrapping, właściwość, 319
 VerticalAlignment, właściwość, 37
 Width, właściwość, 37
 theme, *Patrz* kompozycje
 tile notification, *Patrz* powiadomienia Windows
 Phone, kafelkowe
 tilt, efekt, 193
 timeline animation, *Patrz* animacja z podziałem czasu
 toast notification, *Patrz* powiadomienia Windows
 Phone, wyskakujące
 tombstoning, *Patrz* wstrzymywanie
 Transparent Security, model, 393
 TwitPic
 API, 309
 przesłanie obrazu, 308

U

unsafe, 391
 usługa wysyłania powiadomień, *Patrz* PNS
 ustawienia kulturowe, 231, 233
 neutralne, 233
 niezmienne, 233
 specyficzne, 233
 zapisywanie, 234
 ustawienia regionalne, 233

V

viewport, *Patrz* obszar prezentacji
 Visual Studio, 28
 edytor XAML, 70
 Entity Framework, dodanie komponentu, 60
 Immediate, okno, 99
 kompilacja, 40
 nowy projekt, 34, 35
 Output, okno, 200, 202
 szablony projektów, 34

W

warstwa zachowania danych, 49
 WCF, 335, 339, 341
 implementacja usługi, 63
 kontrakt usługi, 64, 65
 testowanie usługi w lokalnym systemie, 68
 WebBrowser, kontrolka, 167, 168, 177
 Loaded, zdarzenie, 169
 NavigateToString, metoda, 173, 178
 ograniczenia, 178
 SaveToString, metoda, 174, 176, 178
 użycie, 168
 wyświetlanie dynamicznie generowanych treści, 173
 wyświetlanie lokalnych plików HTML, 170
 wyświetlanie stron internetowych, 169
 WebBrowserTask, 196, 200, 298
 webkit-text-size-adjust, właściwość, 177
 wielozadaniowość, 298
 Windows Azure, 30, 49
 instalowanie usługi, 90, 91
 konfiguracja usługi, 89
 projekt, tworzenie, 59
 rejestracja, 87
 uruchomienie usługi w serwisie, 86
 utworzenie usługi, 87
 Windows Azure AppFabric, 49
 Windows Communication Foundation, *Patrz* WCF
 Windows Phone
 centra, 208
 dokumentacja, 29, 30
 ekran, 181
 formaty multimedialne, 282
 fotografia, 297
 informacje, 23, 25
 internet, 167
 narzędzia, 27
 obsługiwany język, 25
 platforma programistyczna, 25
 rejestrowanie urządzenia, 106
 testowanie aplikacji, 110
 ustawienia kulturowe, 234
 wsparcie techniczne, 29
 wymagania sprzętowe, 25, 26
 Windows Phone Application, szablon, 34
 Windows Phone Chrome, 180, 181
 Windows Phone Class Library, szablon, 34
 Windows Phone Emulator, 29
 Windows Phone List Application, szablon, 34

Windows Phone Location Service, *Patrz*
 pozycjonowania, usługi
 Windows Phone Notification, *Patrz* powiadomienia
 Windows Phone
 WPN, *Patrz* powiadomienia Windows Phone
 wstrzymywanie, 199, 299
 wyjątki
 błędu nawigacji, 97
 debugowanie, 97
 obsługa, 98, 101, 109
 przechwytywanie, 98, 100
 urządzenia, 109
 usługi sieciowej, 102
 wymagania sprzętowe, 25, 26
 wzorzec obserwatora, 347
 implementacja, 347

X

XAML, 27
 XAP, plik, 117
 przesłanie do Marketplace, 374
 przygotowanie, 373
 Xbox, 27
 XNA, 27, 303
 zapis zdjęć, 303
 XNA.Framework.Media, biblioteka, 303

Z

zabezpieczenie telefonu, 390
 blokowanie, 390
 dzwonienie, 390
 odnajdywanie, 390
 usuwanie danych, 390
 wyświetlanie komunikatów, 390
 zapisane obrazy, *Patrz* Saved Pictures, folder
 zdjęcia
 Extras, polecenie, 305
 obróbka, 297
 przeglądanie, 302
 przesyłanie, 308
 Share, polecenie, 305, 307
 wykonywanie, 299
 zapisywanie, 303
 Zune, 23, 25, 106
 wyszukiwanie aplikacji, 130

Na rynku systemów operacyjnych dla telefonów komórkowych najczęściej mówi się obecnie o systemach Android oraz iOS. Nie można jednak zapominać o platformie firmy Microsoft! Jej najnowsza wersja — Windows Phone 7 — dopiero co ujrzała światło dzienne, a już zbiera pozytywne opinie. Wygodny interfejs, ogrom możliwości oraz świetna integracja z otoczeniem to tylko niektóre atuty nowego systemu. Zobacz, jak je wykorzystać!

Dzięki tej książce błyskawicznie rozpoczniesz tworzenie nowych aplikacji dla systemu Windows Phone 7. Na wstępie zapoznasz się z wymaganymi elementami stanowiska pracy oraz ich konfiguracją. Na kolejnych stronach znajdziesz przydatne informacje o przechowywaniu danych w chmurze, obsłudze błędów, zarządzaniu pakietami instalacyjnymi oraz udostępnianiu aplikacji użytkownikom. W dalszej części książki autorzy omawiają technologie systemu Windows Phone 7. W trakcie lektury nauczysz się używać akcelerometru, paska aplikacji, przeglądarki internetowej, aparatu fotograficznego oraz multimediów. Ponadto zobaczysz, jak tworzyć aplikacje testowe, obsługiwać wiele języków oraz lokalizować użytkownika telefonu w terenie. Książka stanowi bogaty zbiór niezwykle przydatnych informacji na temat Windows Phone 7. *Musisz ją mieć!*

- System Windows Phone 7 — wprowadzenie
- Przygotowanie środowiska programistycznego
- Chmura — miejsce składowania danych
- Przechwytywanie i usuwanie błędów
- Pakowanie i udostępnianie aplikacji
- Zarządzanie pakietami instalacyjnymi
- Wykorzystanie paska aplikacji
- Korzystanie z akcelerometru
- Przeglądarka internetowa
- Dostępne kontrolki interfejsu użytkownika
- Integracja aplikacji z systemem operacyjnym Windows Phone 7

Stwórz własną niepowtarzalną aplikację mobilną!

helion.pl
Książki
ePub
PDF

W katalogu 7237

Katalog internetowy
<http://helion.pl>

Zamówienia telefoniczne:
0 801 339900
0 601 339900

Helion

Sprawdź najnowsze promocje!
• <http://helion.pl/promocje>
Książki na półce swojej szafki!
• <http://helion.pl/klubczestych>
Zamów informacje o nowościach!
• <http://helion.pl/newsy>

Helion SA
ul. Rakowiecka 17, 44-100 Gliwice
tel.: 02 230 78 62
e-mail: kontakt@helion.pl
<http://helion.pl>

ściągnij po **WIECEJ!**



KOD KORKZYŚCIO

ISBN 978-83-246-3387-6



Cena: 69,00 zł

Informatyka w najlepszym wydaniu