Wykorzystaj potencjał platformy iOS!

Tworzenie aplikacji na platformę iOS 5

z wykorzystaniem Xcode, Interface Builder, Instruments, GDB oraz innych kluczowych narzędzi

Brandon Alexander • J. Bradford Dillon • Kevin Y. Kim

Apress®

Helion

Tytuł oryginału: Pro iOS 5 Tools: Xcode Instruments and Build Tools

Tłumaczenie: Robert Górczyński

ISBN: 978-83-246-4887-0

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: helion@helion.pl
WWW: http://helion.pl (księgarnia internetowa, katalog książek)

Pliki z przykładami omawianymi w książce można znaleźć pod adresem:
ftp://ftp.helion.pl/przyklady/twapxc.zip

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
http://helion.pl/user/opinie/twapxc
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

Spis treści

ROZDZIAŁ

3

Trzy ekrany i cóż… to działa

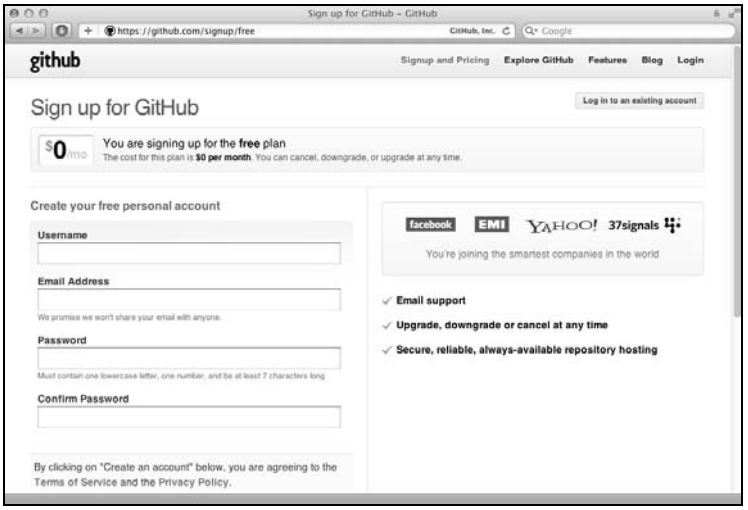
W poprzednim rozdziale pokrótce pokazaliśmy Ci, jak używać nowych narzędzi programistycznych przeznaczonych do tworzenia aplikacji na platformę iOS. W tym rozdziale będziemy kontynuować rozpoczęty wcześniej wątek; poznasz kilka doskonałych technik, których zastosowanie w aplikacji spowoduje, że użytkownicy będą często do niej wracali. W kilku kolejnych rozdziałach będziemy zajmowali się aplikacją, która, delikatnie mówiąc, ma parę wad. Niektóre z nich są oczywiste, natomiast inne pozostają trudne do wykrycia. Omawiana przez nas aplikacja będzie wirtualnym sklepem warzywniczym. Wymagania stawiane przed aplikacją to: wyświetlanie informacji o produkcie łącznie z jego obrazem i opisem, możliwość umieszczenia produktu w koszyku na zakupy, przeglądanie koszyka, wreszcie finalizacja zakupu. Na obecnym etapie aplikacja jest w wersji alfa, to znaczy spełnia podstawowe wymagania, ale wymaga jeszcze wiele pracy, przede wszystkim nad poprawieniem wydajności i stabilności.

Praca z GitHub

Na potrzeby tego produktu kod źródłowy umieścimy w zewnętrznym repozytorium — wykorzystamy społecznościowy serwis o nazwie GitHub. Bezpłatne konto możesz założyć po przejściu na stronę *https://github.com/signup/free*. W chwili pisania tej książki formularz rejestracyjny wyglądał tak jak na rysunku 3.1.

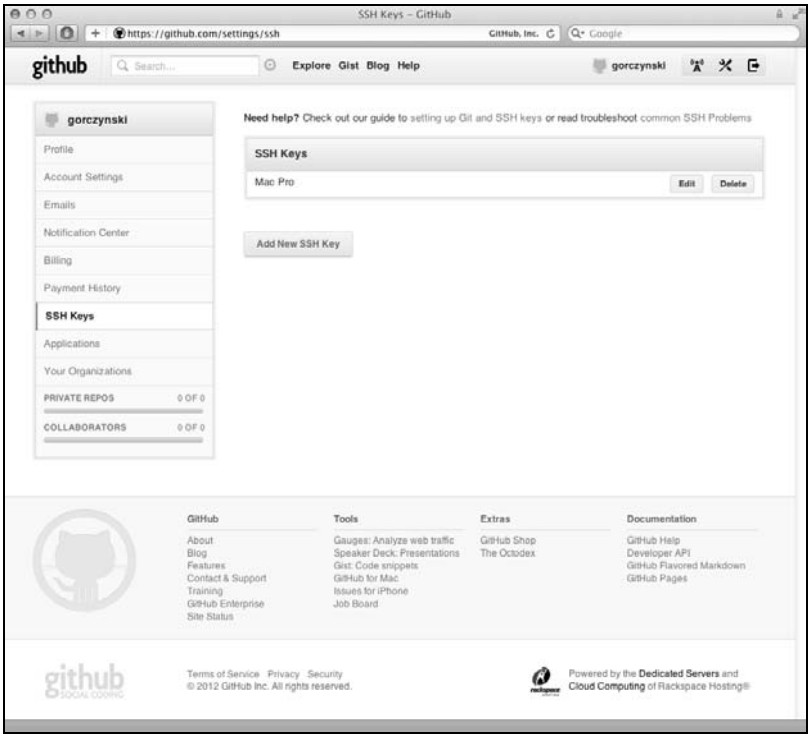
Serwis GitHub oferuje możliwość założenia bezpłatnego konta pozwalającego na utworzenie nieograniczonej (w chwili powstawania książki) liczby publicznych repozytoriów Git. Oferowane są również płatne konta — w takim przypadku można zaprosić innych programistów do pracy z prywatnymi repozytoriami. Aplikacja, nad którą będziemy pracować w tej książce, zostanie umieszczona w serwisie GitHub jako repozytorium publiczne.

Uwaga! Git to rozproszony system kontroli wersji. Został wbudowany w narzędzie Xcode, więc aby z niego korzystać, nie musisz niczego instalować. Wbudowaną w Xcode obsługę repozytorium Git będziemy wykorzystywać w książce do śledzenia wprowadzanych zmian. Git oferuje interfejs wiersza poleceń o wspaniałych możliwościach; poznanie choć części z nich na pewno zwiększy Twoją produktywność jako programisty.



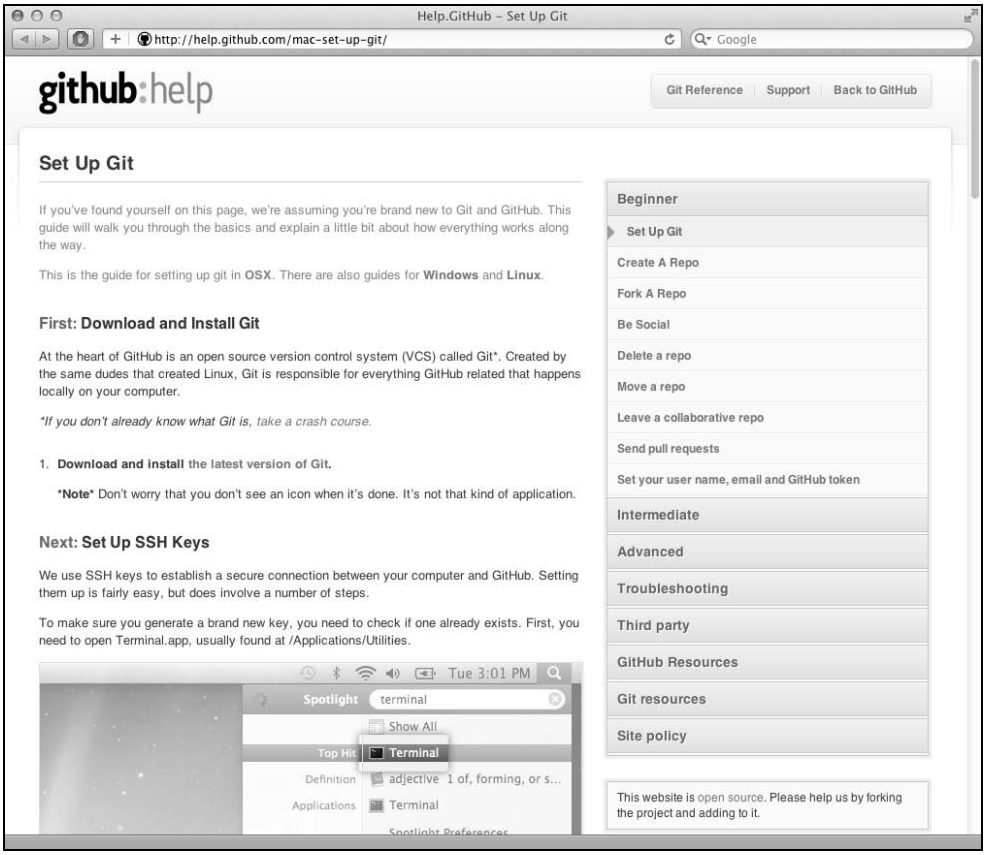
Rysunek 3.1. Formularz rejestracyjny konta w serwisie GitHub

Kolejnym krokiem jest przeprowadzenie konfiguracji Twoich publicznych kluczy SSH. Przejdź na stronę *https://github.com/account/ssh*, a następnie kliknij łącze *Add another public key* (zob. rysunek 3.2).



Rysunek 3.2. Dzięki dodaniu klucza publicznego będziesz mógł bezpiecznie pracować z kodem źródłowym poprzez połączenie SSH

Konfiguracja klucza publicznego SSH jest bardzo prostym zadaniem. Cała procedura została przedstawiona na stronie *http://help.github.com/mac-set-up-git/* (zob. rysunek 3.3).



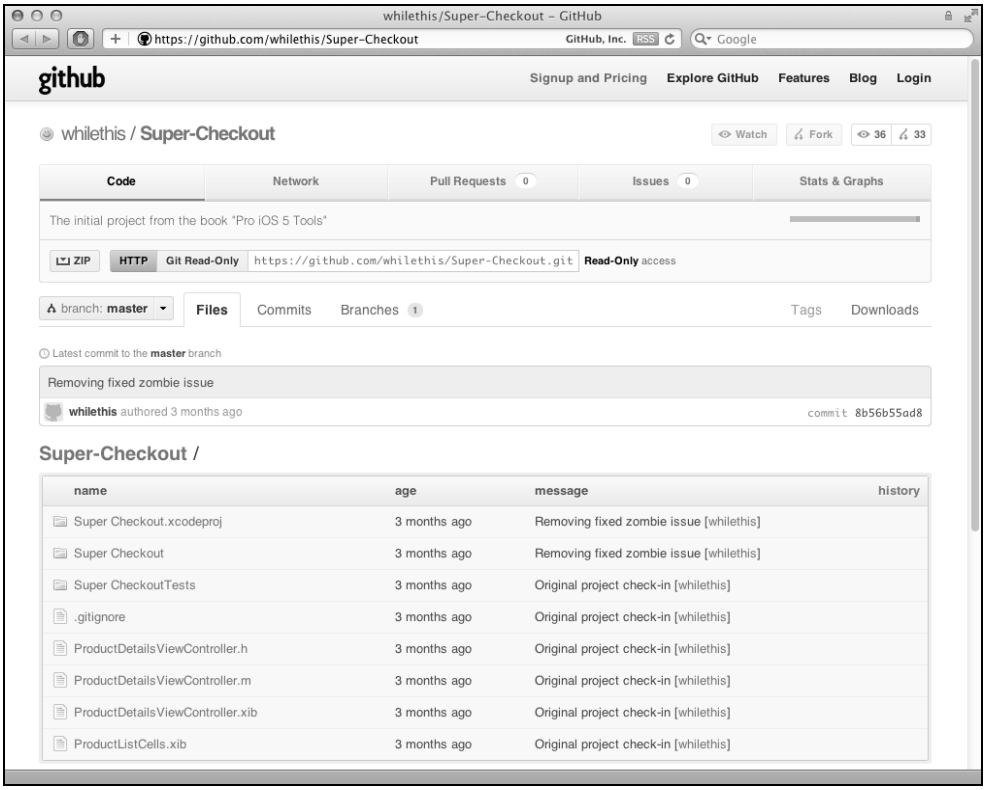
Rysunek 3.3. Konfiguracja kluczy SSH w systemie Mac OS X

Po zakończeniu konfiguracji repozytorium Git oraz kluczy SSH w komputerze można przystąpić do umieszczenia projektu w repozytorium oraz pracy nad projektem z poziomu Xcode.

Nawiązanie połączenia z aplikacją Super Checkout

Kolejnym krokiem jest utworzenie gałęzi projektu, aby móc go pobrać i zacząć wprowadzać własne zmiany. Główny projekt jest dostępny pod adresem *https://github.com/whilethis/Super-Checkout*. Dzięki gałęzi projektu otrzymujesz własną kopię repozytorium, w którym możesz wprowadzać zmiany. Utworzenie gałęzi następuje po kliknięciu przycisku *Fork* znajdującego się w prawym górnym rogu ekranu na stronie głównej projektu (zob. rysunek 3.4).

Na stronie utworzonej gałęzi projektu skopiuj adres URL, który będzie podobny do pokazanego na rysunku 3.5.



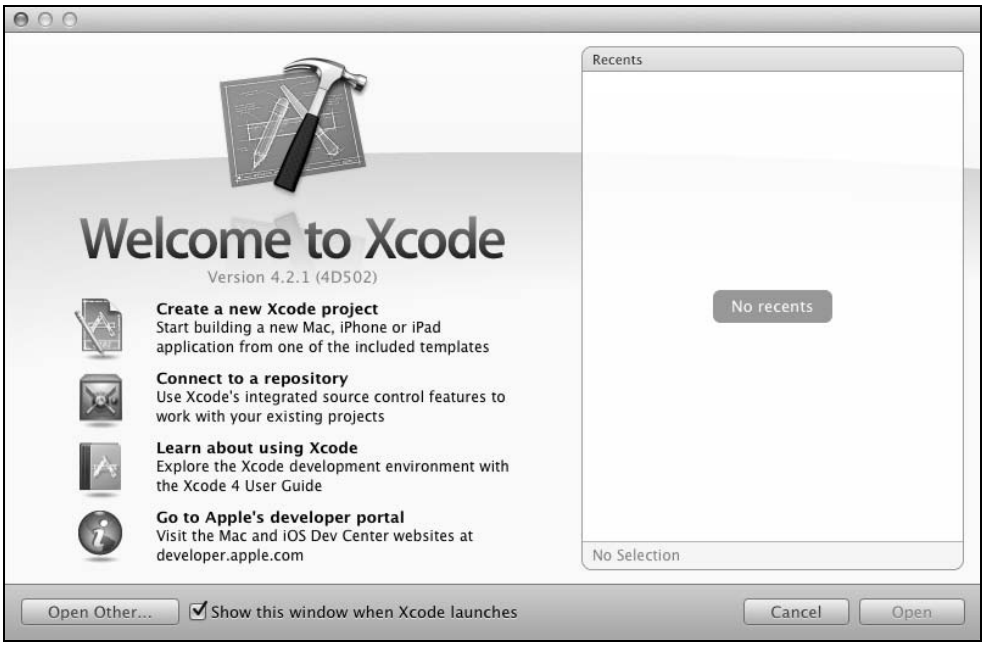
Rysunek 3.4. Kliknięcie przycisku Fork na stronie głównej projektu w GitHub powoduje utworzenie własnej kopii repozytorium

Rysunek 3.5. W adresie URL znajdzie się Twoja nazwa użytkownika serwisu GitHub zamiast pokazanej na rysunku nazwy gorczynski

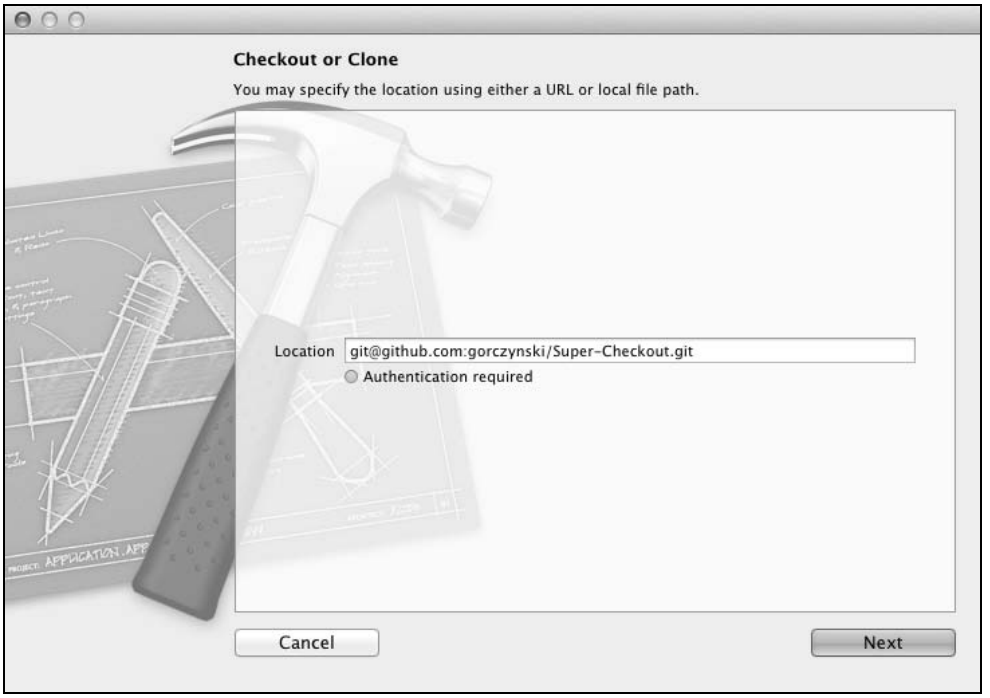
Teraz uruchom narzędzie Xcode 4. Na ekranie wyświetli się pokazany na rysunku 3.6 ekran powitalny. W poprzednim rozdziale utworzyliśmy nowy projekt od podstaw. Tym razem nawiążemy połączenie z repozytorium.

Kliknij przycisk *Connect to a repository*. Zostaniesz poproszony o podanie położenia repozytorium (zob. rysunek 3.7). W tym miejscu wklej adres URL skopiowany z serwisu GitHub; Xcode automatycznie sprawdzi dostępność projektu i pozwoli kontynuować pracę. Kliknięcie przycisku *Next* spowoduje zapis kodu na dysku.

Jeżeli z jakiegokolwiek powodu spróbujesz nawiązać połączenie z serwisem GitHub przed przeprowadzeniem konfiguracji kluczy SSH, zostaniesz poproszony o potwierdzenie tożsamości w tym serwisie poprzez podanie nazwy użytkownika i hasła. Celem przeprowadzenia wspomnianej wcześniej konfiguracji kluczy SSH jest uniknięcie konieczności podawania danych uwierzytelniających. Jeśli jednak skonfigurowałeś klucze SSH i nie wydałeś polecenia `ssh -T`, to w pokazanym na rysunku 3.7 oknie dialogowym pojawi się informacja, że musisz potwierdzić tożsamość.



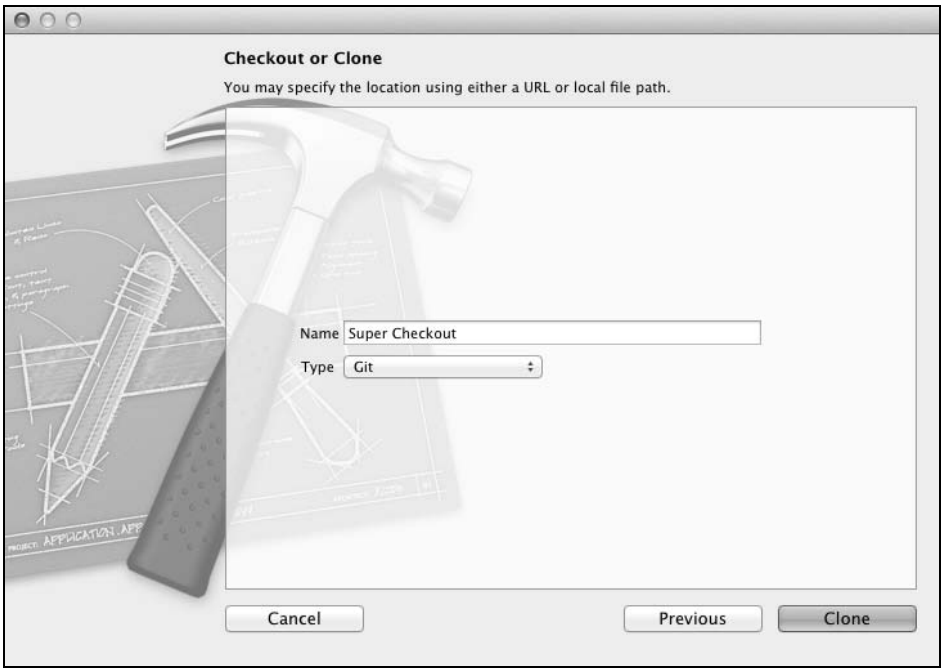
Rysunek 3.6. Ekran powitalny Xcode pozwala na utworzenie nowego projektu bądź nawiązanie połączenia z repozytorium



Rysunek 3.7. Xcode automatycznie wykrywa typ repozytorium podczas pobierania kodu

Uwaga! Podczas bezpośredniego łączenia się ze zdalnym repozytorium upewnij się, że wybrałeś odpowiedni typ repozytorium. Xcode spróbuje klonować pliki lub przekazać je do repozytorium, używając wskazanego protokołu (zob. rysunek 3.8).

Kolejnym krokiem jest nadanie projektowi nazwy w Xcode oraz wskazanie typu repozytorium (zob. rysunek 3.8). Wpisz nazwę ***Super Checkout*** i kliknij przycisk *Clone*.

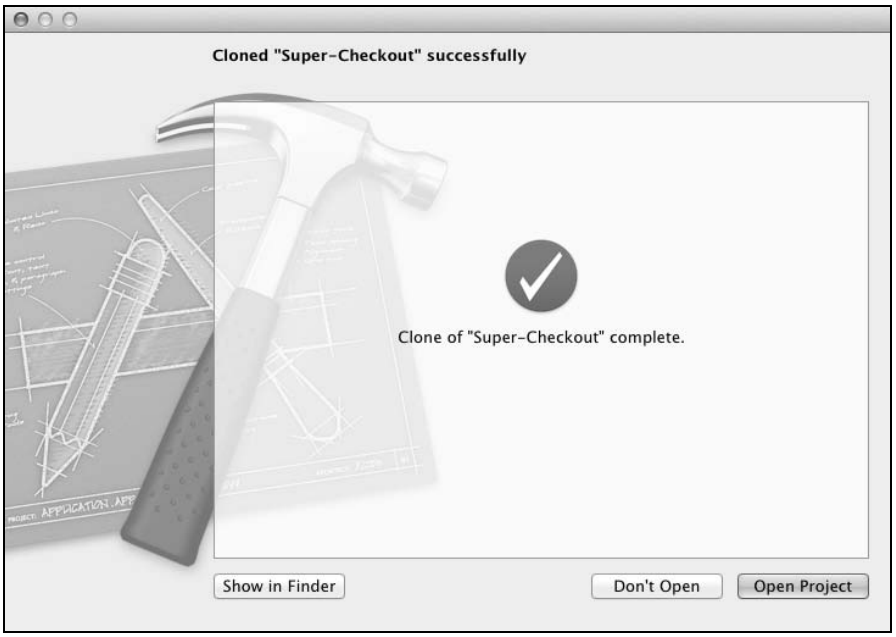


Rysunek 3.8. Nadanie projektowi nazwy i wybór typu repozytorium

W następnym oknie dialogowym musisz podać położenie, w którym zostanie zapisana Twoja kopia repozytorium. Wybierz katalog na dysku i kliknij przycisk *Clone*. Po zakończonej powodzeniem operacji klonowania zobaczysz okno dialogowe z przyciskami *Open Project* i *Don't Open* (zob. rysunek 3.9). Kliknij przycisk *Open Project*; teraz jesteś gotowy do rozpoczęcia pracy nad projektem. Jeśli zrezygnowałeś z omówionego wcześniej kroku utworzenia kluczy publicznych, zobaczysz pewne pola dialogowe związane z podaniem i potwierdzeniem tożsamości. Gdy napotkasz jakiekolwiek problemy, wróć do kroku tworzenia kluczy publicznych i spróbuj jeszcze raz.

Rozejrzyj się

Aplikacja jest bardzo prosta. Składa się z kontrolera nawigacyjnego, kontrolera widoku tabeli odpowiedzialnego za wyświetlanie dostępnych produktów, kontrolera widoku szczegółowego wyświetlającego informacje o wybranym produkcie oraz kontrolera koszyka na zakupy. Ostatni z wymienionych jest kontrolerem widoku modalnego. Moduł odpowiedzialny za kontakt z usługą jest prostym silnikiem i używa tego samego podejścia, które Matt Gemmell (*http://instinctivecode.com/*) zastosował w opracowanym przez siebie silniku aplikacji obsługującej serwis Twitter.



Rysunek 3.9. Klonowanie projektu zakończyło się powodzeniem

API silnika zapewnia komunikację z serwerem i wywołuje analizator w celu przetworzenia danych pochodzących z serwera. Serwer przekazuje dane w formacie JSON (ang. *JavaScript Object Notation*), natomiast projekt wykorzystuje bibliotekę SBJSON (*https://github.com/stig/json-framework/*) do przetwarzania otrzymanych danych w rodzimych klasach Cocoa Touch. Każdy kontroler widoku otrzymuje własny egzemplarz silnika oraz stosuje mechanizm delegacji w celu zapewnienia asynchronicznej komunikacji z serwerem.

Interfejs użytkownika to przede wszystkim widok tabeli wraz z kilkoma własnymi komórkami zaprojektowanymi w module Interface Builder.

Uruchomienie aplikacji Super Checkout

Uruchom aplikację w symulatorze lub urządzeniu iOS — powinna wyglądać tak jak na rysunku 3.10. Pierwszym spostrzeżeniem jest opóźnienie we wczytywaniu obrazów (ten problem rozwiążemy w jednym z kolejnych rozdziałów). Naciśnięcie obrazka produktu powoduje wyświetlenie szczegółowych informacji o produkcie, natomiast naciśnięcie przycisku *Cart* umieszcza dany produkt w koszyku. Warto w tym miejscu dodać, że naciśnięcie przycisku *Cart* powoduje „obrót" widoku i wyświetlenie zawartości koszyka na zakupy.

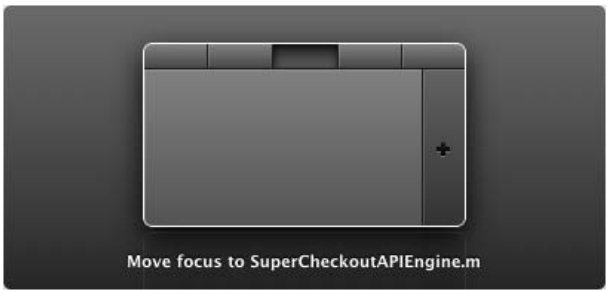
Kiedy zaczniesz poruszać się po aplikacji i poznasz ją lepiej, zobaczysz, że szybki powrót z ekranu zawierającego informacje szczegółowe o produkcie do listy produktów najczęściej powoduje awarię aplikacji. Szczerze mówiąc, aplikacja ulega awarii także w innych przypadkach — to naprawdę wersja alfa, która wymaga jeszcze sporo pracy.

Rysunek 3.10. Aplikacja Super Checkout w wersji alfa

Poruszanie się po projekcie (i po Xcode)

Zarządzanie pracą w Xcode jest kluczem do osiągnięcia biegłości w tworzeniu aplikacji i usprawnieniu każdego projektu. Umiejętność przemieszczania się między poszczególnymi edytorami bez konieczności odrywania rąk od klawiatury może znacznie zwiększyć szybkość prac nad projektem, a poza tym jest to po prostu wygodne. W Xcode 4 wprowadzono wiele użytecznych funkcji pozwalających na pełne wykorzystanie nowego interfejsu. W stosunku do Xcode 3 zmianie uległa także spora liczba najczęściej używanych skrótów klawiszowych.

Jednym z najużyteczniejszych skrótów klawiszowych w Xcode 4 jest prawdopodobnie *Command+J*. Po jego naciśnięciu wyświetla się okno *Move focus to…* (zob. rysunek 3.11). Podświetlony obszar wskazuje miejsce, w którym się znajdziesz po naciśnięciu klawisza *Enter*. Poruszanie się między kartami jest bardzo proste i odbywa się za pomocą klawiszy kursora. Przejście do prawej krawędzi bieżącego edytora (obszar ze znakiem plusa) powoduje utworzenie nowego edytora asystenta.



Rysunek 3.11. Funkcja Move focus to… pozwala na poruszanie się między kartami i edytorami asystentami

Przejście między plikiem implementacji i plikiem nagłówkowym następowało po naciśnięciu klawiszy *Command+Option*[1]*+kursor w górę*. W Xcode 4 ten skrót został zamieniony na *Command+Control+kursor w górę*. Można również użyć gestu machnięcia trzema palcami po gładziku w górę bądź w dół, jeśli ten gest nie został już przypisany innej funkcji w systemie operacyjnym (system Mac OS X 10.7 Lion intensywnie wykorzystuje gest machnięcia trzema palcami).

Kliknięcie symbolu z jednocześnie naciśniętym klawiszem *Option* powoduje wyświetlenie małego okna dialogowego zawierającego informacje na temat danego symbolu. Z poziomu tego okna można wyświetlić plik nagłówkowy klasy bądź przejść do dokumentacji, pod warunkiem że jest to klasa należąca do struktury dostarczanej przez Apple.

Kliknięcie symbolu z jednocześnie naciśniętym klawiszem *Command* powoduje przejście do deklaracji tego symbolu. Jeżeli symbol jest zmienną, podświetlona zostanie deklaracja tej zmiennej. Jeśli symbol jest klasą, w edytorze wyświetli się plik nagłówkowy, w którym został zdefiniowany dany symbol.

Kliknięcie symbolu z jednocześnie naciśniętymi klawiszami *Option* i *Command* powoduje wyświetlenie edytora asystenta i przeprowadzenie takiej samej operacji, jak w przypadku kliknięcia z naciśniętym jedynie klawiszem *Command*.

Naciśnięcie klawiszy *Command+Shift+O* powoduje wyświetlenie okna dialogowego *Open Quickly*. W nim możesz zacząć wprowadzać początek nazwy pliku, dopasowane nazwy plików zostaną wyświetlone w dolnej części okna. To bardzo wygodny sposób, pozwalający na szybkie otwieranie plików.

Opisane powyżej skróty klawiszowe (oraz wiele innych) przedstawiono w rozdziale 12.

Podsumowanie

Aplikacja przedstawiona w tym rozdziale znajduje się na etapie alfa i wymaga usprawnienia na wielu różnych obszarach. Naszym celem jest przygotowanie produktu końcowego, czyli aplikacji, która nie będzie ulegała awarii i będzie miała poprawną architekturę. W kolejnych rozdziałach przeprowadzimy więc operacje m.in.: profilowania, refaktoringu oraz przepisania na nowo pewnych fragmentów aplikacji. Odbywamy dość długą podróż, więc przewróć kartkę i zacznij pracę!

[1] W nowszych klawiaturach Apple klawisz *Option* jest opisany jako *Alt — przyp. tłum.*

Skorowidz

K

L

N

O

P

Q

R

S

T

U

W

X

Z

System operacyjny iOS, wykorzystywany w urządzeniach Apple, cały czas zyskuje na popularności. Jego udział w rynku jest ogromny, a z każdym dniem staje się coraz większy. Co sprawia, że użytkownicy go uwielbiają? Atrakcyjna szata graficzna, błyskawiczne reagowanie na polecenia użytkownika oraz ogromny wybór dopracowanych aplikacji to główne atuty decydujące o jego sukcesie.

Aby zagwarantować ciągły dopływ aplikacji wysokiej jakości, twórcy tego systemu udostępnili programistom świetne środowisko do ich wytwarzania. W trakcie lektury tej książki zaznajomisz się z możliwościami systemu. Dowiesz się, jak używać wszystkich narzędzi dostępnych w arsenale programisty iOS: Xcode, Interface Builder, Instruments, a także narzędzi służących do diagnostyki sieci, obsługi systemu kontroli wersji i innych. Poznasz techniki usuwania błędów z aplikacji, znajdziesz informacje o strukturach Cocoa, zarządzaniu pamięcią, technologii ARC oraz automatyzacji. Nauczysz się tworzyć atrakcyjny i wydajny interfejs użytkownika, rozwiązywać typowe problemy oraz szanować baterię użytkownika. Na szczególną uwagę zasługuje rozdział poświęcony testowaniu aplikacji oraz najlepszym technikom rozprowadzania jej wśród beta-testerów. Książka ta jest idealną pozycją dla każdego programisty chcącego stworzyć jedyną w swoim rodzaju aplikację dla platformy iOS 5.

Odkryj tajemnice iOS 5:

- **Zaprojektuj atrakcyjny i wydajny interfejs użytkownika**
- **Zapewnij niezawodność dzięki korzystaniu z testów**
- **Rozwiąż typowe problemy związane z wyciekami pamięci**
- **Dostarcz aplikację beta-testerom**

Zaskocz użytkowników nowatorską aplikacją!

ISBN 978-83-246-4887-0

Cena: 69,00 zł

9 788324 648870

Helion SA
ul. Kościuszki 1c, 44-100 Gliwice
tel.: 32 230 98 63
e-mail: helion@helion.pl
http://helion.pl