

BUDOWA APLIKACJI NA PLATFORMĘ IPHONE/IPAD

dr inż. Marcin Tomana

iPod Touch

- 📌 iPod nano - najpopularniejsze urządzenie do odtwarzania muzyki - ograniczone jednak funkcje!
- 📌 iPod Touch - to możliwości prawie iPhone bez telefonu, gps
- 📌 Prawie pełna zgodność aplikacji
- 📌 Normalna platforma iOS



Telefon iPhone



- 📌 Telefon to jedno z wielu zastosowań urządzenia
- 📌 Fenomen urządzenia to prostota obsługi oraz duża liczba aplikacji
- 📌 Bardzo popularny w USA
- 📌 Apple sprzedając tylko ten telefon jest 4-tym producentem telefonów (po Nokia, Samsung, LG)

Tablet iPad

- 📌 Pierwszy tablet sprzedający się na masową skalę
- 📌 Ta sama platforma co iPhone, iPod Touch
Te same zalety, te same ograniczenia
- 📌 Niestety duży koszt urządzenia



System operacyjny

📌 Wszystkie te urządzenia to jeden wspólny system operacyjny iOS

📌 Duża konkurencja w platformach mobilnych!

📌 Android, BlackBerry, Nokia Symbian, Samsung Bada, Windows Phone 7, HP WebOs

📌 Automatyczny darmowy update (inaczej niż Android)

📌 Wersja 5 (beta) - iCloud

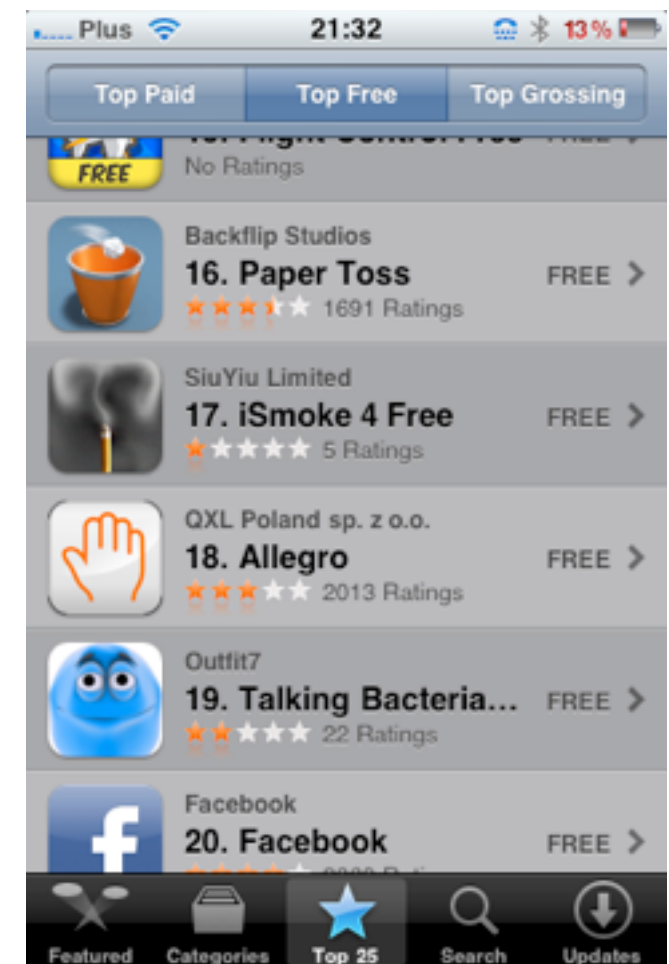
📌 Wersja 4.2 - AirPrint, AirPlay

📌 Wersja 4.1 - iPhone 4, multitasking

Jeden wspólny
system
operacyjny na
wszystkie te
urządzenia

Aplikacje tylko poprzez AppStore

- 📌 Duża liczba dopracowanych, użytecznych aplikacji
- 📌 Niskie koszty programów
- 📌 Nie da się instalować aplikacji z poza AppStore



AppStore

📌 Największy sklep aplikacji mobilnych

📌 99% sprzedaży aplikacji mobilnych na wszystkich platformach w 2009

📌 Ponad 300 tys. aplikacji (11/2010)

📌 Ponad 7 mld pobrań aplikacji (11/2010)

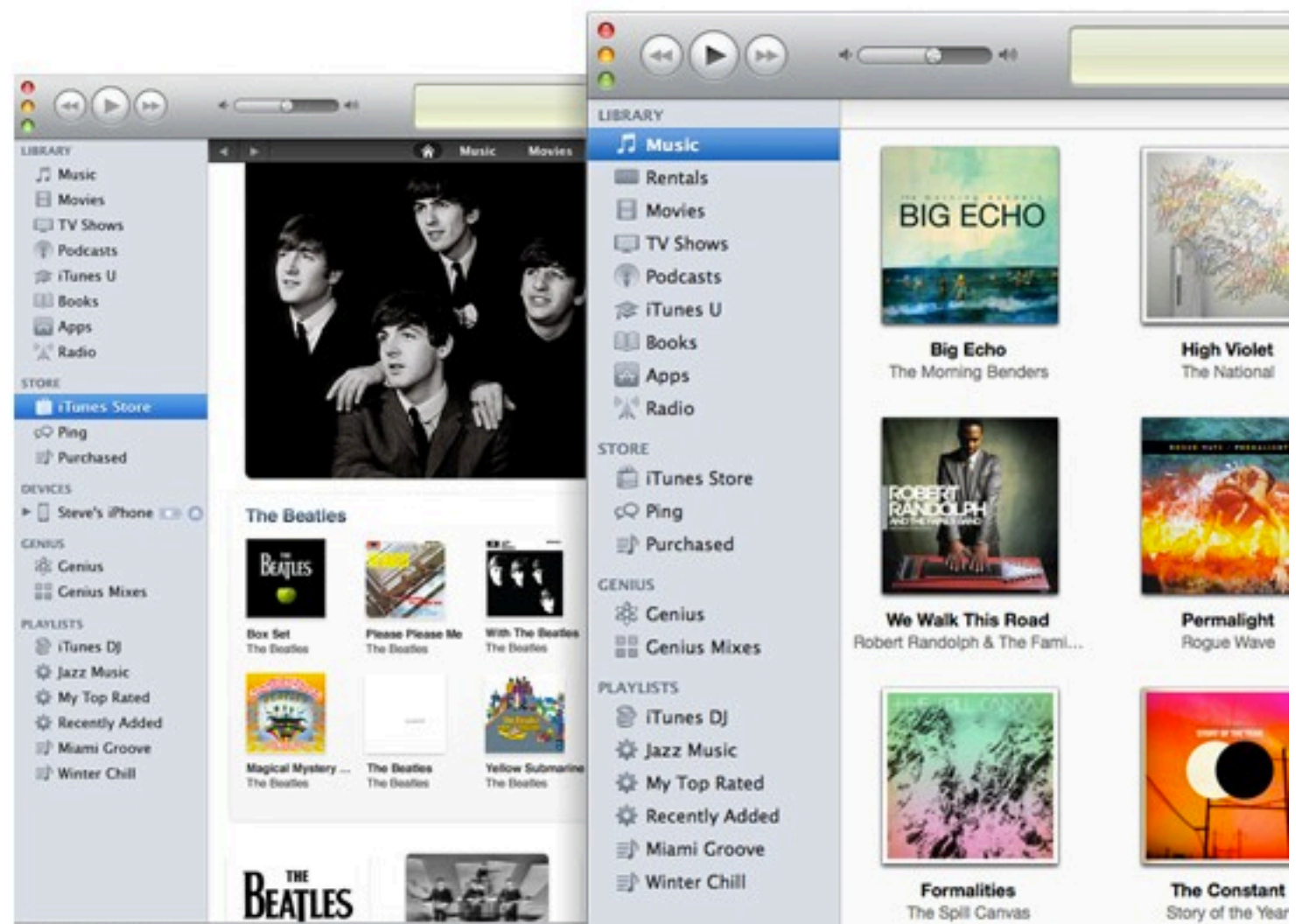


Biznes w AppStore

- 📌 Dla developera: globalny zasięg, hosting, obsługa sprzedaży
- 📌 Przez ostatnie 2 lata sprzedaż ponad 1,4 mld dolarów (2010)
- 📌 Wiele przykładów samodzielnych programistów, którzy zarobili duże pieniądze
 - 📌 Sylwester Łoś w 1 miesiąc 30 tys. sprzedaży aplikacji graficznej po \$1
- 📌 Powstają wielkie firmy osiągające milionowe obroty na platformie iPhone (np. Zynga)

iOS - Platforma zamknięta

- 📌 Nie ma slotu na wymienną kartę pamięci
- 📌 Pliki można wgrywać tylko poprzez program iTunes
 - 📌 Potrzebne połączenie kablowe
 - 📌 Tylko pliki określonych formatów - multimedia (muzyka, video, zdjęcia)
 - 📌 Wgrywać można jeszcze dokumenty zainstalowanych aplikacji



Wielozadaniowość

- 📌 Rozwiązanie całkowicie inne niż na innych platformach
- 📌 Do szybkiego przełączania zapamiętywany jedynie stan aplikacji
- 📌 Aplikacje chcące aktywnie działać w tle muszą być specjalnie zbudowane
- 📌 Brak przypadkowej pracy w tle
- 📌 Mniejsze zużycie baterii



Jak tworzyć aplikacje na platformę?

POTRZEBNY KOMPUTER APPLE

- 📌 Potrzebny MAC OS X
- 📌 Niekoniecznie potrzebne urządzenie
- 📌 Znajomość języka Objective C
 - 📌 Alternatywa to aplikacje Web z Safari Extensions
- 📌 Potrzebne konto developerskie
 - 📌 koszt 100\$ rocznie



Zestaw narzędzi XCode

- 📌 Nowoczesne darmowe środowisko IDE na platformie Apple MAC OS X
- 📌 Rozbudowane narzędzia diagnozujące
- 📌 Symulatory urządzeń iPhone/iPad
- 📌 Rozbudowana pomoc



Historia języka Objective C

- 📌 Język obiektowy wywodzący się z języka C (całkiem inny niż C++)
- 📌 Wiele zapożyczeń z języka SmallTalk
- 📌 Apple przejął go od firmy NEXT (razem ze S.Jobsem)
- 📌 Dziedziczenie jednokrotne oraz interfejsy
- 📌 Cocoa - API w Mac OS X



Implementacja obiektów w ObjC

Deklaracje klas poprzez definicje @

 @interface i @implementation, @property i @synthesize

Wykonywanie metod obiektów poprzez []

 [kowalski zaznaczObecnosc]

 W C++/Javie to samo: *kowalski.zaznaczObecnosc()*

Nazwy metod wieloskładnikowe

 [kowalski zapiszKwote:wplata dlaDaty:data]

 W C++/Javie to samo: *kowalski.zapiszKwote(wplata,data)*

Zarządzanie pamięcią w ObjC

- 📌 Ręczna kontrola pamięci (alokacja, dealokacja)
- 📌 Implementacja licznika referencji

