

Xcode: Vývojové prostredie

MIP

Dávid Bunca

Fakulta informatiky a informačných technológií

Slovenská technická univerzita

Slovenská republika

6.11.2021

Obsah

1	Úvod	3
2	Apple	4
2.1	História Apple	4
2.2	Súčasnosť Apple	4
3	História Xcode	6
3.1	Začiatky	6
4	Xcode	7
4.1	Ponuka Xcodu	7
4.2	Stabilita Xcodu	7
4.3	Verzovací systém v Xcode	8
5	Hodnotenie Xcodu	9
5.1	Problémy	9
5.2	Možné vylepšenia	10
6	Moje skúsenosti s Xcodom	11
6.1	Kladné	11
6.2	Záporné	11
7	Reakcie na témy - Technológia	12
7.1	Spoločenské súvislosti.	12
7.2	Historické súvislosti.	12
7.3	Technológia a ľudia.	12
8	Záver	13

Xcode: Vývojové prostredie

MIP

Dávid Bunca

Abstrakt

Jeden z najrožírenejších dôvodov všeobecnej akceptácie smartfónov sú mobilné aplikácie. Jednou z ciest, ako takéto aplikácie vyvíjať je aj vývojové prostredie Xcode. Článok vysvetľuje čo vlastne Xcode je, ako sa inštaluje a ako sa s týmto vývojárskym prostredím pracuje rovnako tak rozoberá základné informácie o vydavateľskej spoločnosti Apple a histórie nie len spoločnosti ale aj samotného vývojového prostredia. Cieľom je porovnať, čo vieme vytvoriť v Xcode narozdiel od iných vývojových prostredí, zhodnotiť stabilitu vývojového prostredia a celkovú výpočtovú záťaž na operačný systém, vysvetliť, ako sa pracuje s verzovacím systémom (VS) zabudovaným priamo v Xcode. Článok rovnako tak hodnotí problémy a zisťuje dôvody nízkeho hodnotenia užívateľov v obchode AppStore na operačnom systéme macOS. V závere článok skúma aj vylepšenia, ktoré by mohli byť implementované pre zlepšenie používateľských skúseností. Článok bude obsahovať aj autorove osobné skúsenosti s vývojovým prostredím.

1 Úvod

Každý technologický gigant má svoje špecifické vývojové prostredie. Microsoft má Visual Studio, JetBrains vyvíja Fleet, Apple má svoj Xcode, integrované vývojové prostredie na vývoj pre operačný systém macOS, iOS, iPadOS a ďalšie systémy od tejto spoločnosti. Pokiaľ vývojár plánuje vyvíjať programy pre softvér od spoločnosti Apple, Xcodu sa nevyhne. Problémom je však nestabilita a používateľské hodnotenie Xcodu.

Možnosti, ktoré ponúkajú smartfóny prostredníctvom aplikácií sú obrovské. Účelom aplikácie je priniesť prívetivé interaktívne prostredie a nepodobať sa webovej stránke. V minulosti boli mobilné aplikácie vyvíjané za účelom informovania používateľa a jeho produktivitu. Ako dobrý príklad slúžia e-mailové aplikácie, kalkulačky, aplikácie slúžiace na zobrazenie predpovedí počasia a podobne. Ich rôznorodosť a početnosť za posledných pár rokov však enormne stúpila. [9]

Pri vytváraní aplikácií môžu vývojári využívať rôzne IDE a frameworky. Počas vývoja natívnych mobilných aplikácií má najväčší zmysel používať oficiálne IDE pre určitú platformu. [9] Tieto tvrdenia platia špeciálne pre vývoj aplikácií pre zariadenia Apple, nakoľko neexistuje žiadne iné IDE, ktoré dokáže skompilovať aplikáciu spustiteľnú v zariadení Apple.

2 Apple

2.1 História Apple

Spoločnosť založili dvaja technologickí nadšenci Steve Jobs a Steve Wozniak, ktorí sa spoznali vo veku 16 a 21 rokov. K nápadu vytvoriť svoj prvý vlastný počítač prišli tak, že Jobs za myšlienku zarobiť si prijal objednávku na 50 počítačov od miestnej počítačovej predajne. Po dlhom presviedčaní svojho kamaráta Wozniaka spolu s niekoľkými nadšencami tak vytvorili svoj prvý počítač Apple 1. Počítače pre objednávku vytvorili na čas, spolu bolo vyrobených až 200ks Apple 1. [10]

Myšlienka výroby osobných počítačov sa dvom nadšencom zapáčila natoľko, že si chceli založiť firmu. Jediný problém boli však peniaze, ktorých zháňanie mal na starosti Steve Jobs. Banky im požičať nechceli, sami tak veľký kapitál nemohli pokryť. Steve hľadal až stretol Mika Markkulu, ktorý s nimi podpísal bankovú pôžičku na 250 000 dolárov a spolu tak založili v roku 1976 spoločnosť Apple Computer, Inc. [10]

V roku 1979 existoval osobný počítač Macintosh len ako domáci nápad Jefa Raskina, veterána tímu Apple II, ktorý navrhol, aby Apple Computer Inc. vyrobil lacný počítač, ktorý by sa dal ľahko používať ako hriankovač. Pán Raskin veril, že počítač, ktorý si predstavoval a ktorý nazval Macintosh, by sa mohol predáť za 1 000 \$, ak by sa vyrábal vo veľkom objeme a používal výkonný mikroprocesor, ktorý by vykonával presne napísaný softvér. [6]

2.2 Súčasnosť Apple

Apple je v súčasnosti americký technologický gigant sídliaci v Silicon Valley v kalifornskom meste Cupertino, USA. Podľa portálu Forbes sa jedná o spoločnosť s najväčšou trhovou hodnotou na svete (2021). Apple je spoločnosť, ktorá pomohla spustiť revolúciu osobných počítačov koncom 70-tych rokov 20. storočia. Je známa vysoko inovatívnym hardvérom ako iPod a iMac, rovnako ako i softvérom ako iTunes, iLife a operačným systémom macOS. Spoločnosť sa zameriava hlavne na výrobu počítačov Mac, smartfónov, tabletov a podobných smart zariadení ale aj na výrobu operačných systémov ako macOS, iOS, programov ale aj vývojových prostredí Xcode. [10]

Celosvetový úspech spoločnosti Apple je zaujímavou prípadovou štúdiou. Okrem dominancie na západných trhoch sa jej podarilo vybudovať ložalnu základňu kupujúcich v krajinách, ktoré sú pre zahraničné spoločnosti

notoricky konkurencieschopné.

Vo svojej podstate sú princípy vzostupu Apple k moci jednoduché. Jej poslanie je jasné a kupujú si ho jeho zamestnanci aj zákazníci. Aj keď je spoločnosť známa inováciami, nebojí sa prijať cudzie nápady a vylepšiť ich. [5]

3 História Xcode

3.1 Začiatky

Prvá verzia Xcode bola publikovaná na jeseň 2003 ako Xcode 1.0, podľa developer.apple.com bola založená na nástroji Project Builder ale narozdiel od neho mala aktualizované používateľské rozhranie, distribuovanú podporu zostavy a indexovanie kódu. Ďalšie vydanie, verzia 1.5 malo vylepšené dokončovanie kódu a ladenie. [11]

Druhá verzia bola predstavená v roku 2005 spolu s macOS X v10.4, ponúkala vylepšené indexovanie kódu pre programovací jazyk Java a zároveň ponúkala podporu pre jazyk Ant. S druhou verziou prišiel aj nástroj na čítanie oficiálnej dokumentácie firmy Apple. [11]

Zaujímavosťou je, že okrem prvej verzie, ktorá bola vytvorená pomocou nástroja Project Builder boli všetky ďalšie vytvorené v Xcode a boli napísané z veľkej časti v jazyku C a C++.

4 Xcode

4.1 Ponuka Xcodu

Xcode je integrované vývojové prostredie aplikácie Mac OS X, ktoré je určené pre vývojárov. Registrovaní vývojári si môžu stiahnuť ukážkové beta vydania a predchádzajúce verzie balíka prostredníctvom webovej stránky Apple Developer. Predchodca Xcode zdedený od NeXT, Project Builder. Balík Xcode obsahuje bezplatný softvér GNU Compiler Collection (GCC, apple-darwin9-gcc-4.0.1 a apple-darwin9-gcc-4.2.1) a podporuje jazyk C, C++, Fortran, Objective-C, Objective-C ++, Java, AppleScript a Python, poskytuje tiež programovací model Ruby, Cocoa, Carbon a Java. Partneri tiež poskytujú GNU Pascal, Free Pascal, Ada, C Sharp, Perl, Haskell a jazyk D. Sada Xcode používa tiež nástroje na ladenie GDB. [3] Xcode tiež ako jediné IDE vie skompilovať a nainštalovať aplikáciu vo vývoji určenú pre operačné systémy od spoločnosti Apple použitím iOS SDK, tvOS SDK, watchOS SDK alebo macOS SDK. [11]

Xcode obsahuje viacero nástrojov na vývoj v rôznych operačných systémoch vrátane: macOS, iOS, iPadOS, watchOS a tvOS. Kódovanie v Xcode je vykonávané zväčša pomocou programovacieho jazyka Swift. Apple urobil Swift ako jazyk s otvoreným zdrojovým kódom, ktorý sa od jeho vzniku široko používa na programovanie aplikácií. Medzi výhody tohto jazyka patrí hlavne jednoduchosť, rýchlosť algoritmov a kompilátorov a bezpečné bezpečnostné prvky. [7]

4.2 Stabilita Xcodu

Xcode je na rozdiel od iných vývojových prostredí náročný na výpočtový výkon a úložisko. Tento problém sa preukazuje hlavne pri vývoji náročnejších a komplexných projektov.

Pri vývoji aplikácii v Xcode je podľa testovania github používateľa ashfurrow [2] dôležitejší výpočtový výkon procesora ako kapacita RAM. V jeho rebríčku môžeme vidieť, že ani pomerne veľká kapacita operačnej pamäte (32GB) neurýchlila čerstvú kompiláciu aplikácie tak, ako ju urýchlil vyšší výpočtový výkon procesora (Apple M1).

Rovnako tak je Xcode náročný aj na úložný priestor. Inštalačný súbor vývojového prostredia Xcode má sám o sebe približne 10GB, no po nainštalovaní viacerých súčastí sa táto hodnota môže zväčšiť až na 50GB.

4.3 Verzovací systém v Xcode

Xcode podporuje dva najznámejšie verzovacie systémy, a to Git a Subversion. Jedna veľká výhoda Gitu oproti Subversion je tá, že pomocou Gitu vieme vytvoriť lokálne repozitár na našom zariadení, zatiaľ čo Subversion je čisto serverovo orientovaný. Pokiaľ vývojár pracuje na projekte sám, najlepšie pre neho bude implementovanie Gitu, lebo nemusí používať server. Xcode pri vytváraní nového projektu automaticky vytvorí aj Git repozitár za nás. [1]

Veľkou výhodou používania Gitu je, že pomocou neho vieme vrátiť pôvodné verzie projektu. Ak teda do programu nechceme zavedieme chyby, môžeme použiť editor verzie Xcode na porovnanie novej verzie súboru s predchádzajúcou verziou, ktorá fungovala správne, aby sme našli zdroj problému.

Keď na projekte pracujú viacerí ľudia, kontrola zdroja pomáha predchádzať konfliktom a pomáha riešiť konflikty, ak nastanú. Udržiavaním centrálného úložiska, ktoré obsahuje hlavnú kópiu softvéru, systém kontroly zdroja umožňuje každému programátorovi pracovať na lokálnej kópii bez rizika poškodenia hlavnej kópie. Pomocou systému kontroly súborov môžete zabezpečiť, aby dvaja ľudia nepracovali na rovnakom kóde súčasne. Ak dvaja ľudia zmenia rovnaký kód, systém vám pomôže tieto dve verzie zlúčiť.

5 Hodnotenie Xcodu

Podľa viacerých recenzií na stránke g2.com [4] je Xcode „veľmi dobré IDE s veľa chybami“. Xcode je veľmi jednoduchý na používanie pre začiatočníkov a je ľahké naučiť sa ho používať za pár dní.

5.1 Problémy

Viacerým užívateľom vadí, že Xcode je jediné užívateľské prostredie, ktoré dokáže kompilovať aplikácie pre iOS, vývojár si teda nevie vybrať iné IDE, kde by testoval aplikácie pre iOS. Tento problém pramení skôr z politiky spoločnosti Apple ako zo samotného vývojového prostredia.

Používateľom tiež vadí debugger a slabý popis problémov v logovacích súboroch v porovnaní s Android Studio (AS). Debugovanie cez bezdrôtovú sieť je v porovnaní s AS veľmi pomalé.

Problémom sú aj vysoké nároky na ukladací priestor. Xcode potrebuje minimálne 10GB úložného priestoru, čo je pre niektoré zariadenia veľmi veľké.

Aj napriek týmto problémom je Xcode niečo, čo vývojári potrebujú ak chcú programovať aplikácie pre Apple zariadenia, ktoré majú v súčasnosti pomerne veľké zastúpenie na trhu.

Veľa problémov však pramení nie len z IDE Xcode ale aj zo samotného programovacieho jazyka, ktorý používatelia používajú najviac v IDE Xcode. 11 z 12 opýtaných sa sťažovalo na kompilátor Swift a chybové hlásenia, uviedli, že pri používaní jazyka boli tieto hlásenia priveľmi obťažujúce. Ako uviedol jeden z opýtaných: „Kompilátor bol niekedy dosť nestabilný, čo viedlo k chybám, ktoré sme neočakávali. Niekedy som ani nevedel príčinu, ako to napraviť.“ Swift changelog ale ukazuje, že tieto problémy boli návrhárom jazyka známe. Ďalší opýtaný bol kritickejší a povedal, že „zďaleka najväčším problémom je nestabilita nástrojov vo Swift, kompilátor Swift je ako najhorší kompilátor, aký som si kedy dokázal predstaviť, a myslím, že to potvrdia stovky používateľov. Neznáme chybové hlásenia sú tiež bežné problémy hlásené v StackOverflow. Jeden z ďalších opýtaných uviedol, že „zmeny a aktualizácie spôsobili, že niektoré zo zastaraných kódov prestali fungovať, čo bol problém pri používaní rozhraní API“. [8]

5.2 Možné vylepšenia

Na vyriešenie problému s úložiskom, by bolo účinné dať používateľovi/vývojárovi možnosť nainštalovať si súčasti ktoré potrebuje, alebo odstrániť tie ktoré nepotrebuje.

Keďže niektorí vývojári tvrdili, že mechanizmus spracovania chýb, ktorý poskytuje Swift, nie je efektívny (napr. nespracúva asynchrónny kód), výskumníci môžu vykonať empirické štúdie na ďalšie skúmanie tohto tvrdenia a zaviesť techniky na zlepšenie tohto mechanizmu. [8]

Výrobcovia nástrojov a IDE môžu využiť veľký počet otázok súvisiacich s používateľským rozhraním a vyvinúť nástroje na uľahčenie používania, prispôsobenia a animácií prvkov používateľského rozhrania. Napriek tomu väčšina problémov súvisiacich s voliteľným použitím súvisela s nevhodným používaním voliteľných premenných. [8]

6 Moje skúsenosti s Xcodom

S Xcodom pracujem už približne tri a pol roka. Mám za sebou zopár projektov s rôznou komplexnosťou.

6.1 Kladné

Určite pozitívne hodnotím užívateľské rozhranie Xcodu. Farby sú pekne zladené, texty viditeľné a čitateľné. Ikonky súborov sú minimalistické a odkazujú na príponu.

6.2 Záporné

Problém som spozoroval vo VS pri konfliktoch po zjednotení dvoch vetvách. Konflikt bol veľmi komplexný a Xcode ho nevedel správne vyhodnotiť.

Pri tvorbe komplexnejších aplikácií v ktorých potrebujeme použiť služby, napr. push notifikácie, musíme si zakúpiť ročný Apple Developer program, kdežto Android to má zadarmo. Toto opatrenie je hlavne kvôli bezpečnosti.

7 Reakcie na témy - Technológia

7.1 Spoločenské súvislosti.

Spoločnosť si vie ľahko zvyknúť na nové a jednoduché vynálezy. Keď daná problematika je jednoduchá a zrozumiteľná ľahko sa dostane do povedomia spoločnosti a spoločnosť začne daný vynález používať. Pri komplexných a doposiaľ nepoznaných je to zložitejšie. Vynárajú sa rôzne konšpirácie, nepravdivé príbehy a podobne. To vyvolá v spoločnosti strach a tak sa tento vynález stáva nechceným, aj keď jeho účel je reálne prospešný.

7.2 Historické súvislosti.

Technológia je neustále sa rozširujúce odvetvie. Každým rokom vznikajú nové niekoľko-krát výkonnejšie zariadenia, ktoré nám uľahčujú prácu a majú nám pomáhať. No nebolo to tak vždy. Najväčší technologický skok bol bohužiaľ v čase vojen. Ľudia vymýšľali nové zbrane, vozidla ale aj spôsoby tajnej komunikácie. Užitočným výsledkom takýchto pokusov je aj internet. Tento jav pretrváva až dodnes. Väčšina nových technológií je práve vyvíjaná v zbrojarenskom priemysle a následne poskytnutá verejnosti.

7.3 Technológia a ľudia.

Človek sa vždy snaží si svoju prácu zľahčiť. Technológie mu pomáhajú každý deň napr. pri preprave do každodennej práce alebo pri komunikácii s milovanými. Pri pohľade na technológiu môžu vzniknúť aj negatíva vo forme závislosti. Niektoré vynálezy sa stali súčasťou nášho života do takej miery, že život bez nich si ani nevieme predstaviť. Jednou z nebezpečných závislostí je závislosť na sociálnych sieťach. Sociálne siete nás vystavujú množstvu nebezpečenstiev typu dezinformácií, šikane ale aj pocitu nedokonalosti alebo neúspechu. No pri zdravom používaní sociálnych sietí sa vieme dobre zabaviť, porozprávať a zoznámiť sa s ľuďmi ale aj veľa nového naučiť.

8 Záver

Článok opisuje začiatky spoločnosti Apple, ale aj súčasnosť. Venuje sa prvým verziám vývojového prostredia Xcode. Opisuje čo Xcode ponúka, ako sa správa a čo obnáša jeho používanie. Článok sumarizuje hlavné problémy a negatívne reakcie používateľov. Venuje sa aj možným vylepšeniam, ktoré by odstránili niektoré negatívne hodnotenia.

Referencie

- [1] Apple. Using source code control. https://developer.apple.com/library/archive/documentation/ToolsLanguages/Conceptual/Xcode_Overview/UsingSourceCodeControl.html.
- [2] ashfurrow. Xcode hardware performance. <https://github.com/ashfurrow/xcode-hardware-performance>, 2021.
- [3] Chen-Chia Chuang, Jin-Tsong Jeng, Wen-June Wang, Shun-Feng Su, and Dong-Han Lin. ios based speed estimation for the video car recorder system. In *2013 IEEE 9th International Conference on Computational Cybernetics (ICCC)*, pages 61–65, 2013.
- [4] Contributors. Xcode reviews. <https://www.g2.com/products/xcode/reviews#details>, 2021.
- [5] Maiorca D. What makes apple successful? <https://www.makeuseof.com/what-makes-apple-successful/>, 2021.
- [6] Fred Guterl. Design case history: Apple’s macintosh: A small team of little-known designers, challenged to produce a low-cost, exceptionally easy-to-use personal computer, turns out a technical milestone. *IEEE Spectrum*, 21(12):34–43, 1984.
- [7] Muhammad Javed and Mike Estep. Teaching undergraduate software engineering: Xcode mobile app development during dedicated lab periods. In *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 843–848, 2019.
- [8] Marcel Rebouças, Gustavo Pinto, Felipe Ebert, Wesley Torres, Alexander Serebrenik, and Fernando Castor. An empirical study on the usage of the swift programming language. In *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, volume 1, pages 634–638, 2016.
- [9] Jakopce T. Vilček T. Comparative analysis of tools for development of native and hybrid mobile applications. <https://ieeexplore.ieee.org/document/7973662/>, 2017.

- [10] Wikipedia contributors. Apple — Wikipedia, the free encyclopedia. <https://en.wikipedia.org/w/index.php?title=Apple&oldid=1053793088>, 2021. [Online; accessed 6-November-2021].
- [11] Wikipedia contributors. Xcode — Wikipedia, the free encyclopedia. <https://en.wikipedia.org/w/index.php?title=Xcode&oldid=1052291725>, 2021. [Online; accessed 6-November-2021].