

**Česká zemědělská univerzita v Praze**

**Provozně ekonomická fakulta**

**Katedra informačního inženýrství**



**Diplomová práce**

**Proces vývoje mobilní aplikace pro platformu iOS**

**Tomáš Hamerník**

© 2018 ČZU v Praze

# **1. Abstrakt a klíčová slova**

## **1.1. Abstrakt**

Tato práce pojednává o procesu mobilního vývoje, specificky pro operační systém iOS. Hlavním cílem práce je aplikovat získané znalosti v oblasti mobilního vývoje do realizace vývoje mobilní aplikace v praktické části práce.

Teoretická část práce pojednává o základních krocích v procesu mobilního vývoje. Jedná se například o specifikace zadání a analýzu požadavků, metodiky řízení softwarového projektu a jeho nástroje. Následně je věnována pozornost nástrojům pro návrhy a prezentaci grafického rozhraní. V druhé polovině teoretické práce je věnována pozornost tématům více technickým, jako jsou například architektury mobilních aplikací, základním knihovnám třetích stran nebo testování.

Praktická část poté popisuje tvorbu reálné aplikace na podkladech vědomostí z teoretické části práce a praxe.

## **1.2. Klíčová slova**

iOS, Platforma Apple, Swift, mobilní vývoj, programování mobilních zařízení, CocoaPods

# **2. Vlastní text**

Cílem práce je popsat ideální postup vývoje mobilní aplikace pro platformu iOS s aplikací těchto postupů při tvorbě reálného projektu. Vytvoří tedy obecný podklad, který řeší problémy od počátečního návrhu aplikace, přes vytvoření základní struktury, volby důležitých knihoven třetích stran, testování až po dokumentaci zdrojového kódu. Tyto poznatky budou aplikovány v praktické části práce, při vývoji mobilní aplikace.

Metodika diplomové práce je založena na odborné literatuře a reálných vědomostech v oblasti mobilního vývoje. Na základě těchto znalostí bude popsán proces mobilního vývoje tak, jak by měl v ideálním případě probíhat a jaké nástroje a technologie k němu použít. Takto popsáný proces je následně použit při tvorbě demonstrativní mobilní aplikace.

První polovina teoretické části práce se zabývá problematikou analýzy zadání a metodologií řízení softwarových projektů a jejich nástrojů. Jsou zde vysvětleny pojmy Agile a Waterfall development spolu s patřičnými výhodami a nevýhodami. Dále byla pozornost

věnována nástrojům, které jsou pro řízení softwarových projektů ideální. Nejvýraznějším z nich je nástroj JIRA, který je velice funkčně obsáhlý a velmi rozšířený. V dalším kroku bylo přistoupeno k nástrojům pro tvorbu grafického rozhraní aplikací a jejich distribuci vývojářům. V dnešní době je jedním z nejběžnějších softwarových nástrojů pro tvorbu grafického rozhraní aplikací nástroj Sketch, kterému byla věnována celá kapitola. Hotové výstupy ze Sketche je třeba distribuovat programátorům. K tomu je možno využít několika nástrojů, přičemž v této práci jsou zmíněny dva, a to Zeplin a Invision. Druhá polovina teoretické části se věnuje tématům více technickým. V první řadě se jedná o architektury aplikací. Zde je věnována pozornost čtyřem nejznámějším druhům architektur, jejich výhodám a nevýhodám. V další kapitole byl také zmíněn verzovací nástroj, v tomto případě Git. Běžnou součástí mobilního vývoje jsou také aplikace třetích stran. Zde byla pozornost zaměřena na jeden specifický nástroj, kterým je CocoaPods. Tento nástroj vkládá knihovny třetích stran jako závislosti do projektu. Z knihoven třetích stran jsou vybrány tři známé knihovny, které jsou hojně využívány v praxi. Jedná se o Alamofire, SnapKit a SwiftyJSON. K těmto knihovnám jsou přiloženy i praktické ukázky jejich použití. Dalším obsáhlým tématem je testování. Zde jsou popsány základní druhy testování, využívané v běžné praxi. Větší pozornost je věnována Unit a UI testům, které je možné implementovat při vývoji v praktické části práce. Předposlední kapitola pojednává o dokumentaci zdrojového kódu. V poslední kapitole jsou rozebrány základy distribuce vytvořené aplikace.

Praktická část implementuje poznatky z teoretické části práce a praxe tak, aby alespoň částečně simulovala proces mobilního vývoje pro operační systém iOS. V první části je tak vytvořena specifikace požadavků, které by měla aplikace splňovat. Tyto požadavky následně projdou analýzou, kde je vytvořen teoretický popis řešení daného požadavku. Po připraveném řešení bylo třeba vytvořit grafické podklady, které tvoří základ pro vzhled aplikace. K tvorbě podkladů byl využit softwarový nástroj Sketch. Výstupy byly následně exportovány do nástroje Zeplin, který byl také zmíněn v teoretické části práce. S takto vytvořenými podklady bylo možné přejít k tvorbě projektu samotného, což zahrnuje verzování pomocí nástroje Git (v tomto případě specificky nástroj Bitbucket). Následovala volba architektury aplikace. Tou byla zvolena upravená verze MVVM, která odstiňuje logiku z view controllerů. V architektuře se pak nacházejí další vrstvy služeb, které se starají o specifické úkoly jako je například autentifikace uživatele, nebo komunikace s databází. Po volbě architektury bylo nutné projekt strukturovat do logických částí jako jsou například

modely, služby, view controllery a další. Následně bylo možné přistoupit k tvorbě aplikace jako takové a realizaci jednotlivých požadavků. Tato realizace je detailně popsána vždy pro každý požadavek. Jedná se například o realizaci databáze pomocí služby Firebase Database, nebo tvorbu grafického rozhraní pomocí knihovny SnapKit. Současně s vytvářením aplikace bylo vytvořeno několik Unit testů, které jsou zde detailně popsány. Dále byl vytvořen také UI test, který simuluje průchod aplikací od přihlášení, přes přidání záznamu až po odhlášení uživatele. Poslední kapitola se věnuje dokumentaci zdrojového kódu.

Výsledkem práce je na základě podkladů z teoretické části vytvořená reálná aplikace fungující jako mobilní očkovací kalendář dle specifikovaných požadavků v praktické části práce. Při její tvorbě byla implementována většina postupů a praktik popisovaných v teoretické části. Pokud by se mělo jednat o produkční aplikaci uplatnitelnou pro širokou veřejnost, bylo by nutné vytvořit vlastní serverovou část aplikace. Dále by bylo vhodné aplikaci rozšířit například o možnosti záznamů svého dítěte či dalších členů rodiny, detailní informace o vakcínách spolu s jejich účinky a další. Takto aplikovaný postup vývoje je tedy využitelný v běžné praxi pro tvorbu mobilních aplikací na platformu iOS.

### 3. Seznam vybraných použitých zdrojů

1. SCHWARZ, D. *Jump Start Sketch: Master the Tool Made for UI Designers*. SitePoint, 2016. 150 s. ISBN 978-0994346964.
2. KEUR, C. HILLEGASS, A. CONWAY, J. *iOS Programming: The Big Nerd Ranch Guide*. Atlanta: Big Nerd Ranch Guides, 2014. 560 s. ISBN 978-0321942050.
3. Gauchat, J. D. *iOS Apps for Masterminds 3rd Edition*. Mink Books, 2017. 828 s. ISBN 1979692181.
4. SEGUETECH. *8 Benefits of Agile Software Development*. [online]. [cit. 2017-09-07]. Dostupné z: <https://www.seguetech.com/8-benefits-of-agile-software-development/>.
5. ORLOV, B. *iOS Architecture Patterns*. [online]. 2015. [cit. 2017-10-06]. Dostupné z: <https://medium.com/ios-os-x-development/ios-architecture-patterns-ecba4c38de52>.
6. TAM, A. *iOS Unit Testing and UI Testing Tutorial*. [online]. 2017. [cit. 2017-12-12]. Dostupné z: <https://www.raywenderlich.com/150073/ios-unit-testing-and-ui-testing-tutorial>.