

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačního inženýrství



Teze diplomové práce

Architekturou řízený vývoj softwaru v Javě

Bc. Petr Prošek

© 2015 ČZU v Praze

1 Souhrn a úvod

Diplomová práce obsahuje představení architekturou řízeného vývoje aplikací v jazyce Java. Popisuje základní architektury softwaru, tj. jednovrstvá, dvouvrstvá a více vrstvá (typicky třívrstvá) architektura (JEE). Zobrazuje možnosti využití rozhraní v programovacím jazyce Java, jeho implementace a použití. Popisuje návrhový vzor Model View Controller a prakticky jej implementuje ve vzorové aplikaci. Zhodnocuje možnosti vývoje softwaru pomocí architektury Black Box.

V praktické části je popsána jednovrstvá aplikace, kterou autor vytvořil jako součást diplomové práce. V této části jsou popsány použité postupy a architektury při tvorbě aplikace a závěrečná diskuse, kde autor poukazuje na nedostatky a možná vylepšení programovacích postupů programovacího nástroje Java a jejích komponent.

S programovacím jazykem Java se autor seznámil na kurzech, které absolvoval v rámci studijního programu Erasmus na Institutu technologií v Irském Waterfordu. Java se autorovi jevila jako zajímavý oblast k rozvoji svých odborných znalostí s možností budoucího karierního využití a proto se rozhodl „Architekturou řízený vývoj softwaru v Javě“ použít jako téma své diplomové práce.

Java je objektově orientovaný jazyk vycházející z programovacího jazyka C a je jedním z nejpoužívanějších programovacích jazyků v současnosti. Své popularity Java dosáhla díky své přenositelnosti, a také proto, že je distribuována jako opensource.

V první části práce jsou popsány teoretická východiska z oblasti přístupu řešení problémů, jsou zde popsány základní architektury programů a jejich druhy. Dále je zde popsána základní charakteristika návrhových vzorů a programových architektur odvozených o nich. Dále je zde vysvětlena současně velice oblíbená architektura konstrukce aplikací pomocí vzoru Model-View-Controller. Je zde objasněn pojem Black Box programming a další programovací postupy.

Další část práce popisuje vývoj softwaru prostřednictvím nejrozšířenější podnikové technologie Java Enterprise Edition. V této části jsou vysvětleny pojmy úzce související s problematikou tvorby aplikací Java EE. Na tuto část navazuje představení možných nadstavb a rozšíření, které vývoj podnikových aplikací v Java EE velice usnadňují. Ať už se jedná o nadstavby komerční nebo opensourcové.

V praktické části je popsána jednovrstvá aplikace, kterou autor vytvořil jako součást diplomové práce. V této části jsou popsány použité postupy a architektury při tvorbě aplikace a závěrečná diskuse, kde autor poukazuje na nedostatky a možná vylepšení programovacích postupů programovacího nástroje Java a jejích komponent.

2 Klíčová slova

Java, Objektově orientované programování, Architektury softwaru, Black Box programming.

3 Cíl práce a metodika

Cílem diplomové práce je představení architekturou řízeného vývoje aplikací v jazyce Java. Práce popíše základní architektury softwaru, tj. jednovrstvá, dvouvrstvá a více vrstvá (typicky třívrstvá) architektura (JEE). Dále uvede možnosti využití rozhraní v programovacím jazyce Java, jeho implementace a použití. Charakterizuje návrhový vzor Model View Controller a prakticky jej implementuje ve vzorové aplikaci. Práce také zhodnotí možnosti vývoje softwaru pomocí architektury Black Box.

V praktické části autor vytvoří jednovrstvou aplikaci v programovacím prostředí Eclipse. Aplikace bude využívat MVC návrhový vzor a architekturu Black Box. Na závěr autor zhodnotí výhody použité Black Box architektury a navrhne případně varianty jejího vylepšení.

Rešeršní část DP bude založena na analýze odborných a vědeckých dokumentů (zejména monografií) a následně budou získané poznatky prakticky využity k navrhnutí a konstrukci jednovrstvé aplikace v modelovací prostředí Eclipse. Aplikace bude popsána v praktické části spolu s použitými metodami a architekturami. Na závěr autor testuje a zhodnotí použité architektury a navrhne případně varianty jejího vylepšení.

4 Závěr

Java sama o sobě se jeví jako složitý programovací nástroj, ale díky různým nadstavbám a opensource knihovnám se jeho využití značně zjednodušuje. Vytváření složitějších aplikací v Javě by bylo značně složitou a dlouhodobou záležitostí pro jednoho člověka, ale díky opensource nadstavbám (ODB, Spring Framework, Enterprise Java Beans) a zdrojového kódu lidí, kteří je veřejně vystavili pro širokou veřejnost, se celý proces tvorby aplikace značně zjednodušuje.

Nástroj Eclipse se pro programování v jazyce Java velmi osvědčil. Je vyvíjen pod licencí Common Public Licence (CPL) a umožňuje přidávat a distribuovat moduly v jazyce Java. Eclipse je příjemné vývojové prostředí a nabízí spoustu základních funkcí, které jsou k základnímu programování zapotřebí.

Cílem diplomové práce bylo představení architekturou řízeného vývoje aplikací v jazyce Java. Popsání základních architektur softwaru. Popsání pojmů jako Interface (rozhraní), návrhový vzor Model View Controller, architektura Black Box.

V praktické části měl autor vytvořit jednovrstvou aplikaci v programovacím prostředí Eclipse. Za použití MVC návrhového vzoru a architektury Black Box.

V práci bylo dosaženo všech stanovených cílů. V teoretické části, byla popsána architektura rozdělující aplikaci na funkční vrstvy (jednovrstvá, dvouvrstvá a vícevrstvá). Dále byl vysvětlen pojem návrhových vzorů a jejich využití s příkladem konkrétních implementací. V další části práce byla popsána architektura MVC a její druhy. Definice a použití přístupových metod a rozhraní. Byl definován pojem Back Box programming. Velká část práce se věnuje programování v Java Enterprise Edition a nadstavbám ve třívrstvé architektuře, což nebylo konkrétně stanoveným cílem ale hluboce souvisí s tematikou architektonické tvorby aplikací v Javě.

V praktické části byla popsána studentem vytvořená jednovrstvá aplikace. A ukázky implementace architektur při konstrukci aplikace. Aplikace byla při vývoji testována metodami Black Box a White Box testing.

Rozložení aplikace podle architektury MVC se jeví jako efektivní pro funkčnost i zpřehlednění celé aplikace. Použití návrhových vzorů při konstrukci aplikace je také vysoce efektivním postupem.

Rozšíření funkcí aplikace by se dalo uskutečnit prostřednictvím customizace služby *upozornit emailem*, aby si klient mohl připojit svůj emailový účet a nebyl omezen pouze na Google klienta. Další funkcí, která by aplikaci prospěla je možnost vyhledávání, nebo třízení tabulek. Funkce třízení tabulek je deaktivována, protože by byla narušena funkcionality programu.

Možné rozšíření aplikace na více vrstev za použití architektury MVC. Kdy by uživatel měl na svém zařízení klienta aplikace, který by si po přihlášení aktualizoval knihovnu ze serveru, a před ukončením by se změny přenesli na server. V případě komercializace aplikace byt bylo třeba přidat doplňující pole s atributy k lidem a předmětům. Nebo vnořit další JTabbedPane do panelu s předměty.

5 Vybrané zdroje

PECINOVSKÝ, Rudolf. *Java 8: úvod do objektové architektury pro mírně pokročilé*. 1. vyd. Praha: Grada, 2014, 655 s. Knihovna programátora (Grada). ISBN 978-80-247-4638-8.

ČERMÁK, Miroslav. CLEVERANDSMART. *Vícevrstvá architektura* [online]. 2012 [cit. 2015-03-29]. Dostupné z: www.cleverandsmart.cz.

BRUCKNER, Tomáš. *Tvorba informačních systémů: principy, metodiky, architektury*. 1. vyd. Praha: Grada, 2012, 357 s. Management v informační společnosti. ISBN 978-80-247-4153-6.

GAMMA, Erich. *Návrh programů pomocí vzorů: stavební kameny objektově orientovaných programů*. 1. vyd. Praha: Grada, 2003, 386 s. Moderní programování. ISBN 80-247-0302-5.

PECINOVSKÝ, Rudolf. *Návrhové vzory: [33 vzorových postupů pro objektové programování]*. Vyd. 1. Brno: Computer Press, 2007, 527 s. ISBN 978-80-251-1582-4.

ALEXANDER, Christopher, Sara ISHIKAWA a Murray SILVERSTEIN. *A pattern language: towns, buildings, construction*. New York: Oxford University Press, 1977, xliv, 1171 p. ISBN 0195019199.

ČÁPKA, David. ITNETWORK.CZ. *Objektově orientované programování v Javě* [online]. 2013. vyd. 2013 [cit. 2015-03-29]. Dostupné z: <http://www.itnetwork.cz/java-objektove-orientovane-programovani-navody-a-tutorialy>.

BOUDA, Radek. PROGRAMUJTE.COM. *Seriál návrhových vzorů* [online]. 2012. vyd. 2012 [cit. 2015-03-29]. Dostupné z: <http://programujte.com/>.

PECINOVSKÝ, Rudolf. *OOP: naučte se myslet a programovat objektově*. Vyd. 1. Brno: Computer Press, 2010, 576 s. ISBN 978-80-251-2126-9.

PECINOVSKÝ, Rudolf. *Myslíme objektivě v jazyku Java: kompletní učebnice pro začátečníky*. 2., aktualiz. a rozš. vyd. Praha: Grada, 2009, 570 s. Myslíme v--. ISBN 978-80-247-2653-3.

ALLINTERVIEW.COM. *Differences Between Whitebox testing and Blackbox testing* [online]. 2015. vyd. 2015 [cit. 2015-03-29]. Dostupné z: <http://www.allinterview.com/showanswers/33254/differences-between-whitebox-testing-and-blackbox-testing.html>.

ČÁPKA, David. ITNETWORK.CZ. *Java Enterprise Edition (JEE)* [online]. 2014. vyd. 2014 [cit. 2015-03-29]. Dostupné z: <http://www.itnetwork.cz/>.

ORACLE A/NEBO JEJÍ DCEŘINÉ SPOLEČNOSTI. *The Java EE 6 Tutorial* [online]. 2013. vyd. 2013 [cit. 2015-03-29]. Dostupné z: <http://docs.oracle.com/javaee/6/tutorial/doc/gipjf.html>.