

Oponentní posudek diplomové práce

„Analýza a návrh datové základny pro vnitřní potřeby překladiště“

Bc. Jan Dušek

2010

Citované a z převážné části použité metody jsou v souladu se zadáním diplomové práce a s výukou navazujícího magisterského studia.

Vlastní popis toho co diplomant vytvořil je až od strany 41 do 59 a přílohy. Strany 1 až 40 obsahují povinný text jako je zadání, titulní list a obsahy, které zabírají prvních 20 stran, a dále strany 20 až 40 obsahují úvod, rešerši základních informací o logistice a teorii o vývoji informačních systémů z citované literatury. Z toho kapitola 3 Teorie vývoje informačních systémů (tj. 10 stran) je zbytečně podrobná a z hlediska zadání diplomové práce de facto zbytečná.

Diplomantovi se podařilo splnit cíle zadání s připomínkami, které jsou uvedeny dále v tomto oponentním posudku.

Text diplomové práce je na dobré úrovni a nejsou k němu žádné závažné připomínky.

Připomínky a nejasnosti:

1. Kapitola 4.1.2 Procesní modely obsahuje několik závažných nepřesností:
 - a. Tvrzení, že Unified Process neobsahuje procesní modelování je nepravdivé, protože například na http://en.wikipedia.org/wiki/Unified_Process je již zařazeno pod pojmem Business Modeling.
 - b. Popis dvou procesů „Kombinovaná přeprava“ a „Proces zpracování kontejneru pomocí IS“ je z hlediska, jak se chápe proces nedostatečný. Protože nejsou splněny základní podmínky pro popis procesu (definice procesu bude vyžadována při obhajobě).
 - c. Obrázek 5 obsahuje chybné použití prvků v diagramu aktivit. Takto sestavený diagram by nikdy nemohl fungovat (Proč tomu tak je bude vyžadováno u obhajoby).
 - d. Obrázek 5 a ostatní diagramy v příloze, které popisují procesy, jsou klasické vývojové diagramy. Přičemž diagramy aktivit umožňují daleko přesněji zachytit procesní modelování (tj. například jako jsou objektové uzly, oddíly a parametry aktivit).
2. Kapitola 4.2.3 Definování analytických tříd a její odpovídající část ve snímku modelu obsahuje též podstatné nepřesnosti:
 - a. Model analytických tříd není členěn do balíčků.
 - b. Všechny analytické třídy jsou vyneseny pouze v jednom diagramu tříd, což velmi znesnadňuje pochopení logiky analytického modelu.
 - c. Většina relací mezi třídami je nevhodně realizována pomocí kompozice, což zcela mění význam vztahů mezi analytickými třídami. Příkladem může být tato slovní interpretace takové relace: „Uživatel je nedílnou součástí úkolu, a proto pro každý úkol bude vytvořena nová instance uživatele“. Což evidentně chybně. V analýze se jako relace vesměs používají pouze asociace.
3. Snímek modelu v příloze G1 – část „Realizace případů užití“:
 - a. V jednom sekvenčním diagramu není splněna podmínka, že každá sekvence musí začínat vždy od aktéra.
 - b. Uvedené sekvenční diagramy nedostatečně pokrývají scénáře případů užití. Na 27 případů užití jsou v modelu k dispozici pouze tři neúplné sekvenční diagramy.
4. Kapitola 4.2.4 Návrh datového modelu a snímek modelu v příloze G1:

- a. Stejně jako u analytických tříd ani zde nejsou jednotlivé entity seskupeny logicky do balíčků a jsou všechny zakresleny v jednom, naprosto nepřehledném diagramu.
 - b. Na obrázku 9 je zobrazena redukovaná množina entit a relací datového modelu, z které se dá usoudit, že návrh datového modelu ještě není úplný. Například chybí evidentně část, která se týká fakturace za poskytnuté služby překladiště. Částečné řešení atributem „cena“ v tabulce „Objednavka“ je nedostatečné. Dalším příkladem může být položka „hrubaMzda“ v tabulce „Uzivatel“, která je též velkou abstrakcí skutečné mzdové evidence. Z hlediska funkce překladiště hrubá mzda uživatele informačního systému je zcela nezajímavá, protože skutečné náklady na provoz překladiště se skládají ještě z mezd dalších zaměstnanců a hlavně z nákladů na vlastní provoz, energie a technické vybavení překladiště.
 - c. Otázkou je samotné řešení datové základny jednou databází pro všechny překladiště, když tyto překladiště budou mít různé konkurenční majitele. Potom jakékoliv informace o mzdách a cenách za poskytované služby by musely být zabezpečeny před zneužitím. Taky otázkou je kdo by takou centralizovanou databázi překladišť spravoval?
5. Kapitola 4.3.2 Vlastní implementace. Vlastní kód nebyl nijak zvlášť zkoumán, ale zde byly nalezeny některé připomínky:
- a. Například ne úplně správná implementace návrhového vzoru „Singleton“, protože konstruktor třídy „Databaze“ je veřejný, tím je umožněno vznik libovolného počtu instancí této třídy.
 - b. Protože k diplomové práci byl přiložen pouze snímek modelu a ne samotný model, provedl jsem tzv. „Reverse Engineering“ ze zdrojových kódů C#, kterým jsem získal model tříd a zjistil jsem, že většinu zdrojových kódů tvoří formuláře a že další třídy, které umožňují přístup k databázi, jsou automaticky vygenerovány. Tedy nezískal jsem tím větší přehled o implementaci.

Otázky:

1. Definujte obecně co je to proces.
2. Vysvětlete, proč diagramy aktivit ve vašem modelu jsou chybné a jak by měly být správně.
3. Zdůvodněte, proč jste použil řešení tzv. „těžkým klientem“. Jaké jsou nevýhody takového řešení? Jaké jsou jiné možnosti řešení?

Závěr

I přes evidentní chyby se jedná o práci velmi rozsáhlou, v které se autor snažil dodržet a dokumentovat metodicky správné postupy. Podle mého názoru měl být autor na chyby včas upozorněn vedením diplomové práce. Diplomovou práci **doporučuji** k obhajobě a hodnotím známkou **velmi dobře**.

V Pardubicích dne 9.6.2010

Ing. Karel Šimerda



Katedra softwarových technologií