

Posudek bakalářské práce

Matematicko-fyzikální fakulta Univerzity Karlovy

Autor práce: Xuan Anh Duong

Název práce: Vizualizace algoritmů pro návrh databází

Rok odevzdání: 2024

Studijní program: Informatika

Studijní obor: Informatika se specializací Databáze a web

Autor posudku: Ing. Pavel Koupil, Ph.D.

Role: Oponent

Pracoviště: Katedra softwarového inženýrství, MFF UK

Text posudku:

Cílem bakalářské práce bylo navržení a vytvoření webové aplikace pro vizualizaci algoritmů. Aplikace měla být použitelná pro výuku návrhu databázových modelů a podpořit snadnější pochopení této problematiky studentům. Konkrétně se aplikace měla zaměřit na vhodnou vizualizaci algoritmů týkajících se normálních forem a normalizace relací. Mimo jiné měla být aplikace modulární, aby byla umožněna snadná rozšiřitelnost aplikace.

K celé práci	lepší	obvyklé	horší	nevyhovuje
Obtížnost zadání	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Splnění zadání	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Rozsah práce	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Závěrečná práce se skládá ze 6 kapitol, úvodu a závěru. V úvodu se řešitel skrze vyprávění snaží popsat hlavní motivaci pro vznik softwarového díla, tj. zvládnutí zkoušky z předmětu NDBI025. Zároveň se nejedná o úvod v pravém slova smyslu, neboť řešitel popisuje rovnou výsledky práce, viz tvrzení "tato aplikace je schopna...". Celkově marně

očekávám, že v úvodu zazní kontext řešeného problému a vytyčení základních cílů, které jsou v práci řešeny.

Kapitola 2 nazvaná *Analýza* je nejdelší ze všech kapitol. V této kapitole řešitel vychází z přednášek předmětu NDBI025 a snaží se o formální popis implementovaných algoritmů. Nicméně úroveň textu a korektnost definicí je nedostatečná. Text obsahuje mnoho nepřesností a stylistických chyb, např. $\{X \rightarrow Y, X, A \subseteq A\}$ namísto $\{X \rightarrow Y | X, A \subseteq A\}$. Literatura, ze které student čerpá v této sekci, je **nedostatečně (nebo téměř vůbec) citována**. Student v textu užívá mnoho zkratk (např. ER, UML), ovšem jejich význam není nikde rozepsán. Tento nedostatek se opakuje i ve zbývajících kapitolách.

Především ale **text kapitoly 2 odpovídá nepřesným studentským poznámkám z přednášek předmětu NDBI025**. V lepším případě se jedná o definici problému, která by ale mohla být podstatně stručnější. **V žádném případě se nejedná o analýzu**. Předpokládal bych, že student provede studii za účasti studentů, zjistí, které elementy algoritmů jsou pro studenty obtížně pochopitelné, a na základě této studie budou formulovány konkrétní požadavky.

Subsekcí 2.2.14 (ve které opět chybí uvedení) bych koncipoval jako samostatnou kapitolu. Ovšem i tato sekce požaduje podstatnou revizi. Očekával bych, že přístupy budou porovnány na základě objektivních nikoliv subjektivních kritérií, např. open-source, licence, multi-platformnost, implementované algoritmy, způsob vizualizace, rozšiřitelnost aplikace, ... A zároveň bych očekával, že přístupy budou porovnány z praktického pohledu s ohledem na požadavek lepšího pochopení algoritmů studenty.

Kapitola 3 nazvaná *Specifikace* začíná slovy: "z úvodních požadavků a podrobné analýzy vyplývá...". Ovšem žádné (úvodní) požadavky nezazněly a analýza ve smyslu skutečné analýzy provedena nebyla. Student až v této kapitole uvádí pět hrubých požadavků, avšak jsou málo konkrétní a spíš obecné povahy.

Kapitole 4 nazvaná *Návrh aplikace* opět chybí úvod kapitoly, který by naznačil obsah kapitoly. Kromě názvu totiž kapitola o návrhu aplikace vůbec není. Namísto návrhové dokumentace je součástí kapitoly pouze seznam a popis použitých technologií (viz sekce 4.1) a je popsán algoritmus nazvaný *Zjednodušený uzávěr množiny funkčních závislostí* (viz

sekce 4.2).

Na první pohled se zdá, že sekce 4.1 obsahuje detailní popis použitých technologií. Nicméně v textu chybí reference (např. "novinkou Reactu jsou tzv. hooky" - co jsou to "hooky"?), text obsahuje nevhodnou nebo nepřesnou terminologii a působí dojmem anglického textu přeloženého v překladači nebo pomocí AI vygenerovaného textu (např. "Javascript je nenáročný, interpretovaný nebo právě včas kompilovaný jazyk s prvotřídními funkcemi."). Dále text obsahuje subjektivní argumentaci, která by v praxi neobstála, např. "... každá z nich (knihoven Vue, Angular - doplněno) má svoje výhody a nevýhody. Ty ale nehrají až tak zásadní roli - záleží především na tom, v čem se člověku pracuje nejlépe.". Z toho lze usuzovat, že **hlavním kritériem pro výběr technologií byla obliba knihoven studentem, nikoliv jejich vhodnost použití a další měřitelné parametry** (např. výkon).

V sekci 4.2 pak student zdůvodňuje, proč implementuje pouze zjednodušený algoritmus. Následně uvádí příklad a pseudokód algoritmu, který ovšem není popsán. Nicméně ani jedno do Návrhu aplikace nepatří.

Kapitola 5 nazvaná *Uživatelská příručka* se na první pohled díky obrázkům zdá detailní, ovšem není úplná. Vyberu-li jeden nedostatek za všechny, tak hned úvodní tvrzení: "Pokud nahrajete správně formátovaný soubor..." - neumožňuje v použití programu pokračovat, neboť *správný formát* není v textu definován.

Kapitola 6 nazvaná *Návrh architektury* je popisem struktury projektu, v žádném případě se nejedná o (návrh) architektury programu.

Kapitole 7 nazvané *Programátorská příručka* opět chybí úvod. Očekával bych, že v programátorské dokumentaci naleznou alespoň (1) diagram tříd zachycující objektový návrh aplikace, (2) diagram případů užití a (3) sekvenční diagram vytvořené na základě analýzy a požadavků. Nic z tohoto kapitola 7 neobsahuje.

Naopak kapitola v sekci 7.1 obsahuje minimální požadavky pro spuštění programu a postup instalace. Očekával bych, že toto bude součástí uživatelské dokumentace. Sekce Databáze (7.2) popisuje účel databáze Firebase. Ovšem očekával bych, že zde najdu class diagram zachycující datový model aplikace. Je kolekce "problems" jedinou entitou? A jak

jsou reprezentováni např. uživatelé a jejich atributy, uživatelské session, projekty a jiné entity a vztahy mezi nimi?

Dále autor v sekci 7.3.1 tvrdí, že jádrem implementace jsou soubory Algoritmus.js a NormalFormALG.js a že "tyto soubory jsou implementovány v jazyce JavaScript a jsou navrženy s důrazem na objektově orientovaný přístup, což zajišťuje modularitu a opětovnou použitelnost kódu". S tímto tvrzením zásadně nesouhlasím. Samotný (pokus o) OOP návrh modularitu ani opětovnou použitelnost kódu nezajistí.

Kapitola 8 nazvaná *Závěr* je opět spíše vyprávěním, nikoliv shrnutím přínosů bakalářské práce a implementované aplikace. Není zde obsažen ani nástin future work, případné možnosti rozšiřitelnosti práce.

Text práce je psán průměrnou češtinou s četnými gramatickými chybami. V textu se velmi špatně orientuje. Formální úprava práce je nedostatečná. Práci by prospělo lepší grafické rozlišení pseudokódu algoritmů od (pod)sekcí, rozlišení příkladů od (pod)sekcí, lepší zasazení příkladů, tabulek, obrázků a pseudokódů do kontextu a především jejich popsání. Dále bych očekával úvodní shrnutí na začátku každé kapitoly. Také bych doporučil lepší formátování textu a rozlišení pojmů (např. pomocí kurzívy) a úseků kódu (např. pomocí pevné šířky textu) od zbytku textu. Práci by rozhodně prospělo použití doporučené LaTeX šablony a použití vektorové grafiky. Celkově sazba práce působí amatérsky.

Textová část práce	lepší	obvyklé	horší	nevyhovuje
Formální úprava	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Struktura textu	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Analýza	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Vývojová dokumentace	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Uživatelská dokumentace	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Grafické uživatelské rozhraní (GUI) je neintuitivní. Je použito mnoho barev a tvarů u tlačítek, ovšem není mi jasný význam a jako uživatel se ztrácím v ovládacích prvcích, přestože jich je malý počet. Dále by uživatelskému grafickému rozhraní prospělo, kdyby funkční závislosti byly vizualizovány také jako graf závislostí.

Implementace samotná pak obsahuje kritickou bezpečnostní chybu. Konkrétně řádek `export const password = "123456"`; v souboru `constantvalues.js` nelze myslet vážně. Dále zdrojový kód neobsahuje dokumentaci, ovšem vyzdvihl bych správně vytvořenou lokalizaci.

Řešení neobsahuje jednotkové ani integrační testy. Nelze tak kvalifikovaně ověřit stabilitu implementace.

Implementační část práce	lepší	obvyklé	horší	nevyhovuje
Kvalita návrhu	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Kvalita zpracování	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Stabilita implementace	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Pokrytí testy	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Odevzdaná práce úrovní zpracování v žádném případě neodpovídá závěrečné bakalářské práci. Práce obsahuje **netriviální množství nepřesností**, úroveň zdrojového kódu a dokumentace je nedostatečná, **text neobsahuje podstatné části** (analýza, návrhová dokumentace, programátorská dokumentace v textu zcela chybí nebo z velké části nevyhovují). **Zadání práce považuji za nesplněné**, neboť požadavkem bylo vytvoření aplikace, která pomůže s pochopením látky studentům. Žádná analýza směrem k odhalení problémů s pochopením látky studenty nebyla provedena. A tedy **není vůbec jasné, co bylo implementováno**.

Otázky k obhajobě:

- Jak vypadá struktura a správný formát vstupního souboru (viz kapitola 5)?
- Jaké jsou výhody databáze Firebase oproti jiným dokumentovým databázím, např. MongoDB?
- Proč jste se rozhodl použít dokumentovou databázi namísto např. relačních databází (PostgreSQL, MySQL a jiné)?

- Uvažoval jste o použití sloupcových databází (např. Apache Cassandra)?
- Jaké konkrétní kroky jsou v kódu použity, aby byla zajištěna modularita programu? Jaké jsou jednotlivé moduly? Jak vypadá dekompozice programu na jednotlivé moduly?
- Jaké konkrétní kroky jsou v kódu použity pro zajištění opětovné použitelnosti a především udržovatelnosti a rozšiřitelnosti kódu?
- Lze v aplikaci vykonat krok zpět nebo dopředu? A jak je to implementováno/jak byste to implementoval?

Práci nedoporučuji k obhajobě.

Práci nenavrhuji na zvláštní ocenění.

17. června 2024

Podpis: