

**I. IDENTIFIKAČNÍ ÚDAJE**

<b>Název práce:</b>	Kooperativní myšlenkové mapy
<b>Jméno autora:</b>	Michal Kalista
<b>Typ práce:</b>	bakalářská
<b>Fakulta/ústav:</b>	Fakulta elektrotechnická (FEL)
<b>Katedra/ústav:</b>	Katedra počítačů
<b>Oponent práce:</b>	Ing. Martin Ledvinka, Ph.D.
<b>Pracoviště oponenta práce:</b>	Katedra počítačů

**II. HODNOCENÍ JEDNOTLIVÝCH KRITÉRIÍ**

<b>Zadání</b>	<b>průměrně náročné</b>
Zadání má potenciál pro velmi komplexní nástroj, ale lze jen splnit i minimalistickou verzí.	

<b>Splnění zadání</b>	<b>splněno s menšími výhradami</b>
Zadání bylo splněno, až na testování s více uživateli měnícími stejnou mapu. Alespoň text se o tomto testovacím scénáři nijak nezmiňuje.	

<b>Zvolený postup řešení</b>	<b>správný</b>
Postup řešení lze považovat za správný. Jedinou výtku mám k implementaci frontendu v čistém JavaScriptu. To není, podle mého názoru, z dlouhodobého hlediska udržitelná strategie.	

<b>Odborná úroveň</b>	<b>B - velmi dobře</b>
Text je z odborného hlediska velmi dobrý. Zarazilo mě jen tvrzení, že PostgreSQL je rozšířením jazyka SQL (Sekce 3.4.3). Výhrady mám též k návrhu REST rozhraní backendu a tvrzení, že Servlety a JAX-RS jsou na stejné úrovni (Obrázek 4.1).	

<b>Formální a jazyková úroveň, rozsah práce</b>	<b>C - dobře</b>
Text je psán česky, ale obsahuje poměrně hodně překlepů a gramatických chyb. Na můj vkus je také až příliš stručný – zatímco řešerše existujících řešení je velmi podrobná, návrh vlastního řešení se omezuje na seznam funkčních a nefunkčních požadavků, diagram případů užití a krátký popis datového modelu. Student popisuje podrobněji použité technologie než fungování a strukturu svého díla. Popis ovládání pomocí klávesnice je utopen v textu, spíše by mu prospěla např. tabulka. V několika případech též následují nadpisy za sebou bez mezilehlého textu (např. 3, 3.1, 3.1.1).	

<b>Výběr zdrojů, korektnost citací</b>	<b>C - dobře</b>
Zdrojů je dostatečné množství, ale u mnoha z nich chybí autor. Jedná se především o webové stránky jednotlivých nástrojů či dokumentace. Je otázkou, jestli by v takovém případě nebylo vhodnější volit např. jen poznámku pod čarou s odkazem na příslušný web.	

<b>Další komentáře a hodnocení</b>	
Struktura rozhraní aplikace (záměrně nepoužívám termín REST) je poněkud zvláštní. Rozhodně není hierarchická (např. proč se pro vytváření mapy používá endpoint s username uživatele?). Projekt (ani text) neobsahuje návod na spuštění. Při zkoumání Docker konfigurace jsem zjistil, že aplikaci je nejdřív třeba zbuildovat lokálně a pak teprve vytvořit Docker image, do kterého se kopíruje sestavený archiv. To nepovažuji za příliš šťastné řešení.	

### **III. CELKOVÉ HODNOCENÍ, OTÁZKY K OBHAJOBĚ, NÁVRH KLASIFIKACE**

*Práce kolegy Kalisty řeší rozhodně velmi palčivý problém. Sám mám zkušenosti s desktopovými nástroji pro tvorbu myšlenkových map a nemožnost jejich sdílení a editace online byla hlavním problémem, který jsem při jejich použití měl. Výsledné řešení je spíše minimalistickou variantou, ale vzhledem k tomu, že se jedná o bakalářskou práci, nevidím v tom zásadní problém. Pochybnosti mám o snadnosti budoucího rozvoje vzhledem k tomu, že student zvolil pro frontend čistý JavaScript bez jakéhokoliv frameworku. Co se týče textu, myslím, že pan kolega se měl více zaměřit na popis svého díla a méně na technologie, které k jeho tvorbě použil.*

*Otázky k obhajobě:*

*1. V Sekci 4.2 píšete, že si klient na základě notifikace o změnách z WebSocketu načte diagram pomocí REST API a že toto řešení šetří síťovou komunikaci? Můžete tuto myšlenku rozvést? Na první pohled se zdá, že zaslání změn pomocí WebSocketu by naopak ušetřilo komunikaci, protože klient by je mohl rovnou aplikovat a nemusel si načítat mapu (či její části) znovu.*

Předloženou závěrečnou práci hodnotím klasifikačním stupněm C - dobře.

Datum: 31.5.2023

Podpis: