

I. IDENTIFIKAČNÍ ÚDAJE

| | |
|----------------------------|--|
| Název práce: | Aplikace pro organizování výletů |
| Jméno autora: | Timotej Barbuš |
| Typ práce: | bakalářská |
| Fakulta/ústav: | Fakulta elektrotechnická (FEL) |
| Katedra/ústav: | Katedra počítačů |
| Oponent práce: | Ing. Zdeněk Rybola, Ph.D. |
| Pracoviště oponenta práce: | Katedra softwarového inženýrství, Fakulta informačních technologií (FIT), ČVUT v Praze |

II. HODNOCENÍ JEDNOTLIVÝCH KRITÉRIÍ

| | |
|---|-------------------------|
| Zadání | průměrně náročné |
| <i>Hodnocení náročnosti zadání závěrečné práce.</i> | |
| Cílem práce je vytvoření vcelku jednoduché webové aplikace pro vytváření výletů a jejich sdílení s dalšími uživateli. | |

| | |
|--|------------------------------------|
| Splnění zadání | splněno s menšími výhradami |
| <i>Posuďte, zda předložená závěrečná práce splňuje zadání. V komentáři případně uveďte body zadání, které nebyly zcela splněny, nebo zda je práce oproti zadání rozšířena. Nebylo-li zadání zcela splněno, pokuste se posoudit závažnost, dopady a případně i příčiny jednotlivých nedostatků.</i> | |
| Výsledkem práce je prototyp aplikace pro plánování a sdílení výletů. Aplikace je vcelku jednoduchá a nabízí jen minimální možnosti pro tvorbu skutečné komunity. Využití prostředků softwarového inženýrství je omezené na řadu diagramů a jednotkové testování. | |

| | |
|---|----------------|
| Zvolený postup řešení | správný |
| <i>Posuďte, zda student zvolil správný postup nebo metody řešení.</i> | |
| Postup řešení práce je zvolený vhodně. Student postupuje logicky od analýzy prostředí, přes stanovení požadavků, analýzu možností realizace, návrh řešení, jeho implementace a na závěr otestování. | |

| | |
|---|-----------------------|
| Odborná úroveň | D - uspokojivě |
| <i>Posuďte úroveň odbornosti závěrečné práce, využití znalostí získaných studiem a z odborné literatury, využití podkladů a dat získaných z praxe.</i> | |
| V rámci analytické části student analyzuje podobné aplikace a srovnává jejich možnosti. Ve výsledném srovnání ale nezohledňuje možnosti manuálního plánování a plánování s časy, ve kterých se dle popisu jednotlivé aplikace značně liší. Autor identifikuje velké množství požadavků, jejichž popis je však velmi stručný. Některé jsou zbytečně rozdrobené a daly by se sloučit, u jiných naopak chybí dostatečné podrobnosti (např. způsoby filtrování výletů při hledání, důsledky zrušení výletu na stávající účastníky apod.). Stejně tak některé nefunkční požadavky jsou velmi vágní a těžko ověřitelné. V rámci modelu případů užití jsou v diagramech použita označení funkčních požadavků, ale v textu jsou odlišné názvy skutečných případů užití. U alternativních scénářů pak chybí informace, kdy se použijí místo standardního scénáře. Analýza technologií předchází jakékoli úvaze o systémové architektuře řešení. V analýze technologií pro Front-end jsou srovnávány frameworky pro zcela odlišné části řešení (React+Angular vs. Bootstrap). Analýza map je omezena pouze na Google Maps bez jakékoli diskuze alternativ. Není jasný účel diagramu tříd na obrázku 5.1. Svým obsahem je mezi doménovým modelem a implementačním datovým modelem, nevhodně kombinující prvky obou modelů. Návrh je obecně složený z několika diagramů bez popisu a diskuze. Postrádám podrobnější vysvětlení zamýšlené architektury, členění na vrstvy/komponenty, jejich závislosti, apod. Stavový model je pak zcela vytržený z kontextu. Sekvenční modely jsou také formálně chybné, na rozmezí zachycení komunikace tříd a modelu případu užití napříč komponentami. Kapitola o zabezpečení je velmi povrchní a nekonkrétní, zaměřující se spíše na fakt, že uživatel se bude do aplikace | |

přihlašovat, čímž se omezí množina dostupných dat.

Výsledná aplikace může sloužit jako prototyp určený k dalšímu rozvoji. Z textu ale není jasné, jaký je její účel. Zdá se, že vytvoření výletu a pozvání účastníků není nijak navázáno na samotnou organizaci výletu, zajištění ubytování nebo jízdenek/letenek, takže přidání se k výletu nic neznamená. Organizátor vlastně jen publikuje své osobní plány. Kvůli možnosti pozvat konkrétní přátele je třeba výlet zveřejnit, což umožní komukoli se také přidat (požádat). Výlety nejsou ani omezeny kapacitně. Kterýkoli účastník výletu pak může kompletně měnit plán bez jakékoli kontroly organizátorem. Pro pozvání uživatele k výletu probíhá hledání podle jména s našeptáváním existujících uživatelů, což umožňuje odhalit všechny uživatele systému. To považuji za hrubé porušení bezpečnosti a soukromí uživatelů. V aplikaci jsou zobrazeny doporučené výlety, ale v práci chybí jakákoli informace o tom, jak se tyto doporučené výlety vyberou/určí.

Aplikace integruje Google Maps API pro zobrazení mapy na úvodní stránce. Mapa však neobsahuje žádná data, nemá žádnou vazbu na výlety ani jejich aktivity či přesuny. Integrace je tak v této fázi naprosto zbytečná.

Aplikace byla otestována manuálně několika uživateli a zpětná vazba byla částečně implementována. Automatizace testování se omezuje na několik málo testů prezentační vrstvy – kontrola zobrazení/nezobrazení stránky v návaznosti na práva. Ač jsou testy prezentovány jako jednotkové, nevyužívají žádnou izolaci od dalších částí aplikace a testují tak de facto celou aplikaci, avšak bez reálné kontroly skutečně správného obsahu.

Přílohou práce je čistá kopie projektové adresáře, včetně všech stažených závislostí a skrytých souborů IDE. Zdrojové kódy jsou vhodně komentované, avšak v podstatě pouze na úrovni controllerů. Tam je také veškerá logika aplikace. Řešení je celkově velmi primitivní. Je založené na nativním využití frameworku Laravel a postrádá jakoukoli abstrakci či zobecnění. Logika aplikace je implementována přímo v controllerech přímo s pomocí modelů pro data založených opět přímo na Laravel frameworku.

Celkově není řešení špatné, jen působí nedomyšleně a nedotaženě a velmi jednoduše.

Formální a jazyková úroveň, rozsah práce

B - velmi dobře

Posuďte správnost používání formálních zápisů obsažených v práci. Posuďte typografickou a jazykovou stránku.

Po formální a jazykové stránce je práce na dobré úrovni. V práci je pouze malé množství prohřešků, gramatických chyb či tvarů slov.

V diagramech použitých v práci jsou na pár místech drobnější nedostatky.

Výběr zdrojů, korektnost citací

E - dostatečně

Vyjádřete se k aktivitě studenta při získávání a využívání studijních materiálů k řešení závěrečné práce. Charakterizujte výběr pramenů. Posuďte, zda student využil všechny relevantní zdroje. Ověřte, zda jsou všechny převzaté prvky řádně odlišeny od vlastních výsledků a úvah, zda nedošlo k porušení citační etiky a zda jsou bibliografické citace úplné a v souladu s citačními zvyklostmi a normami.

Práce odkazuje pouze na 16 zdrojů, přičemž všechny představují odkazy na domácí stránky zkoumaných aplikací či použitých/zvažovaných technologií. Práce neobsahuje žádné reference na použité principy softwarového inženýrství apod.

Další komentáře a hodnocení

Vyjádřete se k úrovni dosažených hlavních výsledků závěrečné práce, např. k úrovni teoretických výsledků, nebo k úrovni a funkčnosti technického nebo programového vytvořeného řešení, publikačním výstupům, experimentální zručnosti apod.

Viz část hodnocení **Odborná úroveň**.

III. CELKOVÉ HODNOCENÍ, OTÁZKY K OBHAJOBĚ, NÁVRH KLASIFIKACE

Shrňte aspekty závěrečné práce, které nejvíce ovlivnily Vaše celkové hodnocení. Uveďte případné otázky, které by měl student zodpovědět při obhajobě závěrečné práce před komisí.

Z hlediska softwarového inženýrství je práce velmi primitivní. Analýza je na rozumné úrovni, ač trpí určitými nedostatky (např. podrobnost požadavků a absence analýzy doménových entit). Návrh řešení je pak velmi vágní a samotné řešení pak z pohledu architektury velmi primitivní. Samotná aplikace pak působí nedomyšleně z pohledu jejího účelu a používání.

Předloženou závěrečnou práci hodnotím klasifikačním stupněm **D - uspokojivě**.

Otázky k obhajobě:

- Během prezentace práce objasněte výhody uživatele z přihlášení se k výletu.
- Popište architekturu výsledného řešení a popište, jak by se dala architektura rozčlenit na třívrstvou (prezentační, doménovou/servisní/obchodní a datovou vrstvu). Jakou část logiky byste přesunul do doménové vrstvy a jak by se pak realizovaly závislosti?
- Vysvětlete lépe princip jednotkových testů. Jak byste vyřešit izolaci testovaných částí od jejich závislosti při testování?

Datum: 9.6.2023

Podpis:

