



## ZADÁNÍ BAKALÁ SKÉ PRÁCE

<b>Název:</b>	Databáze transformací
<b>Student:</b>	Martin Znamená ek
<b>Vedoucí:</b>	Ing. Ji í Pavelka
<b>Studijní program:</b>	Informatika
<b>Studijní obor:</b>	Softwarové inženýrství
<b>Katedra:</b>	Katedra softwarového inženýrství
<b>Platnost zadání:</b>	Do konce letního semestru 2017/18

### Pokyny pro vypracování

Tématem práce je webový portál pro správu a aplikaci textových transformací. Transformace jsou sady skript , které se postupn aplikují na vložený obsah.

- 1) Analyzujete sou asnou situaci a požadavky.
- 2) Navrhnete a implementujete nástroj, který umožní s maximálním využitím externích služeb:
  - vytvá ení a úpravy transformací,
  - vývoj a verzování skript ,
  - pojmenování, popis a ukázkou transformace,
  - odkazování a et zení skript ,
  - aplikaci skript na vlastní text i textové soubory,
  - hodnocení skript a uživatel ,
  - organizaci a vyhledávání skript .
- 3) Prove te vhodné testování.

### Seznam odborné literatury

Dodá vedoucí práce.

Ing. Michal Valenta, Ph.D.  
vedoucí katedry

prof. Ing. Pavel Tvrdí k, CSc.  
d kan

V Praze dne 17. ledna 2017



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Bakalářská práce

## **Databáze transformací**

*Martin Znamenáček*

Vedoucí práce: Ing. Jiří Pavelka

16. května 2017



---

## Poděkování

Rád bych poděkoval vedoucímu práce, panu Ing. Jiřímu Pavelkovi, za jeho podporu a cenné rady, které byly pro tuto práci velkým přínosem. Dále bych chtěl poděkovat své rodině a přátelům za poskytnutí opory. V neposlední řadě bych chtěl poděkovat všem lidem, kteří se zúčastnili průzkumů a testování obsažených v této práci.



---

# Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 16. května 2017

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2017 Martin Znamenáček. Všechna práva vyhrazena.

*Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.*

### **Odkaz na tuto práci**

Znamenáček, Martin. *Databáze transformací*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2017.



---

# Abstrakt

Práce se zabývá analýzou, návrhem a implementací aplikace umožňující aplikování skriptů na textová data přímo z prostředí webového prohlížeče.

Jednou z předností této vzniklé aplikace je možnost řetězení nezávislých skriptů zpracovávajících text do větších celků, nazývaných textové transformace. Vzniklá aplikace umožňuje diskuzi uživatelů nad jejich problémy s textovými daty. Poskytuje možnosti správy, sdílení a vyhledávání textových transformací.

**Klíčová slova**    webový portál, zpracování textu, verzování, sdílení, spouštění skriptů online

---

# Abstract

This thesis deals with analysis, design and implementation of an application allowing to apply scripts on textual data directly from the browser environment.

One of the advantages of this application is the ability to chain independent scripts processing text into greater units, called text transformations. The resulting application allows users to discuss their problems related to text data. The application provides options for managing, sharing, and searching text transformations.

**Keywords** web portal, text processing, versioning, sharing, online script execution

---

# Obsah

<b>Úvod</b>	<b>1</b>
<b>1 Cíl práce</b>	<b>3</b>
<b>2 Analýza</b>	<b>5</b>
2.1 Průzkum existujících řešení . . . . .	5
2.2 Uživatelský průzkum . . . . .	8
2.3 Požadavky na funkce . . . . .	16
2.4 Požadavky na kvality . . . . .	18
2.5 Vyhodnocení požadavků vůči existujícím řešením . . . . .	20
<b>3 Návrh</b>	<b>25</b>
3.1 Procesy . . . . .	25
3.2 Případy užití . . . . .	28
3.3 Architektura aplikace . . . . .	30
3.4 Prototypování . . . . .	36
3.5 Testování prototypu . . . . .	41
<b>4 Implementace</b>	<b>43</b>
4.1 Popis implementace . . . . .	43
4.2 Specifikace API . . . . .	48
4.3 Testování . . . . .	50
4.4 Možnosti rozšíření . . . . .	52
<b>Závěr</b>	<b>53</b>
<b>Literatura</b>	<b>55</b>
<b>A Dotazník ke kvantitativnímu průzkumu</b>	<b>57</b>
A.1 Testování dotazníku . . . . .	57

A.2 Plné znění dotazníku . . . . .	60
A.3 Vyhodnocení dotazníku . . . . .	62
<b>B Pravidla a zásady projektů FIT</b>	<b>67</b>
<b>C Prototyp</b>	<b>77</b>
C.1 Ilustrace testování prototypu . . . . .	77
<b>D Příručka nasazení</b>	<b>85</b>
D.1 Nasazení webového portálu . . . . .	85
D.2 Nasazení komponenty pro spouštění skriptů . . . . .	85
<b>E Seznam použitých zkratk</b>	<b>87</b>
<b>F Obsah příloženého CD</b>	<b>89</b>

---

## Seznam obrázků

3.1	Diagram procesu 1 . . . . .	26
3.2	Diagram procesu 2 . . . . .	26
3.3	Diagram procesu 3 . . . . .	27
3.4	Diagram procesu 4 . . . . .	28
3.5	Schéma architektury . . . . .	36
3.6	Normální rozložení . . . . .	37
3.7	Rozložení pro užší obrazovky . . . . .	37
3.8	Stránka nového skriptu (editace skriptu) . . . . .	38
3.9	Stránka detailu skriptu . . . . .	39
3.10	Stránka spuštění skriptu . . . . .	40
3.11	Stránka výpisu skriptů . . . . .	41
C.1	Pohled na součásti papírového prototypu . . . . .	77
C.2	Krok 1 testovacího scénáře 1 . . . . .	78
C.3	Krok 2 testovacího scénáře 1 . . . . .	78
C.4	Krok 3 testovacího scénáře 1 . . . . .	78
C.5	Krok 4 testovacího scénáře 1 . . . . .	79
C.6	Krok 5 testovacího scénáře 1 . . . . .	79
C.7	Krok 1 testovacího scénáře 2 . . . . .	80
C.8	Krok 2 testovacího scénáře 2 . . . . .	80
C.9	Krok 3 testovacího scénáře 2 . . . . .	81
C.10	Krok 4 testovacího scénáře 2 . . . . .	81
C.11	Krok 1 testovacího scénáře 3 . . . . .	82
C.12	Krok 2 testovacího scénáře 3 . . . . .	82
C.13	Krok 3 testovacího scénáře 3 . . . . .	82
C.14	Krok 4 testovacího scénáře 3 . . . . .	83
C.15	Krok 1 testovacího scénáře 4 . . . . .	83
C.16	Krok 2 testovacího scénáře 4 . . . . .	83
C.17	Krok 3 testovacího scénáře 4 . . . . .	84
C.18	Krok 4 testovacího scénáře 4 . . . . .	84



---

## Seznam tabulek

2.1	Srovnání existujících řešení, část první . . . . .	20
2.2	Srovnání existujících řešení, část druhá . . . . .	22
4.1	Parametry volání požadavku RunScript . . . . .	48
4.2	Datový typ ScriptConfiguration . . . . .	48
4.3	Datový typ RepoConfiguration . . . . .	49
4.4	Parametry odpovědi požadavku RunScript . . . . .	49
4.5	Parametry volání požadavku GetInput . . . . .	49
4.6	Parametry odpovědi požadavku GetInput . . . . .	49
4.7	Parametry volání požadavku PostOutput . . . . .	50
4.8	Parametry odpovědi požadavku PostOutput . . . . .	50





---

# Úvod

V každé firmě se pracuje s velkým počtem různých textů, které jsou různě upravovány. Na základní úpravy nám postačí textový editor. Pokud ale potřebujeme s texty provést složitější operace, např. formátování dat, narážíme na problém. Pro jeho řešení je zapotřebí napsat skript, který tuto operaci provede.

Recepční, sekretářky, obecně lidé v administrativě ale psát skripty neumí a tak jsou odkázáni na pomoc druhých. Mohou se pokusit vyhledat řešení na internetu, ale portál se skripty zabývajícími se transformací textů chybí. I když najdou skript řešící jejich problém, nebo jim IT oddělení skript sepiše, objeví se další překážka: jak skript spustit? Jelikož s příkazovým řádkem zacházet zpravidla neumí, budou odkázáni na online spuštění skriptu. Další službu, kterou si budou muset pracně vyhledat.

Autor práce si téma zvolil z důvodu, že pro něho bylo důležité, aby výsledná práce měla využití. Po seznámení s tímto tématem ho velmi zaujala myšlenka vytvoření komunitního nástroje, který má potenciál zjednodušit život velkému množství lidí.

Tato práce se zabývá průzkumem potřeb potenciálních uživatelů. Následným návrhem reflektujícím tyto potřeby a implementací webové aplikace, která funguje jako komunitní portál umožňující pohodlné vyhledávání, sdílení a spuštění skriptů zabývajících se textovými transformacemi. Aplikace umožňuje zřetězení nezávislých skriptů za sebe.

První část práce se zabývá mapováním existujících řešení a průzkumem potřeb uživatelů. Ze získaných informací stanovuje požadavky na funkcionální aplikaci.

## ÚVOD

---

Náplní druhé části práce je návrh aplikace vycházející ze zjištěných požadavků. Návrh se zaměřoval primárně na probíhající procesy, rozložení webových stránek a s tím související architekturu aplikace.

V poslední části práce je aplikace dle návrhu implementována. Je popsáno jak bylo přistoupeno k řešení vybraných požadavků. V závěru práce je aplikace otestována.

## Cíl práce

Dílčím cílem této práce je zmapovat existující řešení a následně prozkoumat potřeby potenciálních uživatelů. Ze získaných informací poté stanovit požadavky na funkcionalitu aplikace.

Druhým dílčím cílem je navrhnout, implementovat a otestovat aplikaci umožňující pohodlné vyhledávání, sdílení a spouštění textových transformací, které budou v aplikaci verzovány a budou podporovat zřetězení v nich obsažených skriptů.

Cílem práce není detailní návrh uživatelského rozhraní. Návrh bude omezen v kontextu zvoleného existujícího řešení.



---

# Analýza

Tato kapitola popisuje analytickou část práce. Nejprve byl proveden průzkum existujících řešení, ze kterého vyplynuly nejzákladnější požadavky na aplikaci. Tyto požadavky byly v rámci uživatelského průzkumu porovnány se skutečnými potřebami uživatelů. Vyhodnocením jejich potřeb byly následně pro aplikaci vypracovány funkční požadavky.

## 2.1 Průzkum existujících řešení

Úkolem průzkumu existujících řešení bylo zmapovat aktuální situaci a nalézt takové funkce, které nemají chybět ve vytvářené aplikaci.

Existující řešení, nalezená v rámci literární rešerše lze rozřadit do několika skupin. První skupinou jsou řešení pro online běh skriptů. Projekty se zabývají editací, testováním a sdílením skriptů. V této skupině je velmi zajímavý projekt Jdoodle. Další skupinou je kontrola skriptů. Zde je nejznámější projekt ShellCheck, který poskytuje podrobnou kontrolu syntaxe skriptů. Poslední skupinou jsou portály otázek a odpovědí. Zde zcela vládne Stack Overflow. Podrobnější rozpis nalezených významných vlastností je na konci této kapitoly.

### 2.1.1 Online běh skriptů

Tyto projekty umožňují uživatelům spouštět v online prostředí své skripty. Ve většině případů přidávají i další funkce (sdílení a ukládání skriptů atd).

#### 2.1.1.1 Jdoodle

Jdoodle poskytuje možnost spustit kód velkého množství jazyků online. Zajímavé jsou možnosti ukládání a sdílení skriptů. Významné vlastnosti jsou:

- dokončování rozepsaného příkazu,

## 2. ANALÝZA

---

- uložení skriptu ve svém účtu,
- sdílení skriptu podle URL adresy,
- sdílení skriptu vložením do webové stránky pomocí iframe [1],
- možnost spolupráce více uživatelů zároveň na jedné stránce:
  - každý může psát kód skriptu,
  - komunikovat přes chat a pomocí hlasového hovoru. [2]

### 2.1.1.2 Tutorialspoint

Tutorialspoint je propracované online vývojové prostředí na tvorbu bash skriptů. Dobrá funkce je import a export projektu do systémů třetích stran. Nejzajímavější vlastnosti aplikace jsou:

- možnost vytváření projektů,
- exportování a importování z Git projektů a online úložišť,
- sdílení projektů podle URL adresy. [3]

### 2.1.1.3 Learn Shell

Learn Shell je propracovaný výukový portál pro psaní skriptů. Výuka je členěna do lekcí, uživatelé si průběžně své skripty online spouští. Aplikace dále nabízí:

- jednotlivé lekce,
- procvičování. [4]

### 2.1.1.4 Webminal

Webminal poskytuje online terminál operačního systému Linux. Uživatelé mají vyhrazené systémové prostředky a mohou si zkusit bash příkazy. Nejzajímavější vlastnosti aplikace jsou:

- výukové lekce pro práci s OS Linux,
- možnost pomoci od ostatních členů komunity,
- práce se soubory, procesy či databázemi. [5]

## 2.1.2 Kontrola skriptů

Projektů zabývajících se kontrolou syntaxe není mnoho. Jako jejich zástupce byl zvolen ShellCheck.

### 2.1.2.1 ShellCheck

Aplikace ShellCheck poskytuje kromě kontroly syntaxe bash skriptu také varování před možnými chybami v kódu a doporučení jak kód zlepšit. [6]

### 2.1.2.2 Příkaz „bash -n“

Vestavěné řešení příkazové řádky. Provádí pouze základní kontrolu syntaxe.[7]

### 2.1.3 Otázky a odpovědi

Do této kategorie spadají diskuzní portály zabývající se otázkami a odpověďmi. Tomuto typu portálů dominuje platforma Stack Exchange. Další projekty stejného zaměření její funkce kopírují. Společné funkce portálů na otázky a odpovědi:

- prostor pro diskuzi,
- hodnocení uživatelů,
- statistiky aktivity (počty zhlédnutí, poslední odpověď, atd.),
- štítky.

#### 2.1.3.1 Stack Overflow

Aplikace je známá jako největší online komunita pro programátory. Je součástí platformy Stack Exchange. Kromě standardních funkcí nabízí zprostředkování práce, budování dokumentace a mnoho dalšího. [8]

#### 2.1.3.2 Klony Stack Exchange

Nevýhodou Stack Exchange je, že není opensource. Díky tomu vzniklo více projektů s podobnou funkcionalitou, které již opensource jsou. Například:

- OSQA<sup>1</sup>,
- Askbot<sup>2</sup>,
- Question2Answer<sup>3</sup>,
- LampCMS<sup>4</sup>.

### 2.1.4 Významné vlastnosti

Vlastnosti existujících řešení byly roztrženy do několika kategorií, které byly pro vznikající aplikaci klíčové. Nejdůležitějšími vlastnostmi byly:

- Editace skriptů a testování:
  - vytváření projektů,
  - export a import Git projektů,
  - uložení ve svém účtu,
  - dokončování rozepsaného příkazu,

---

<sup>1</sup><http://www.dzonesoftware.com/products/open-source-question-answer-software>

<sup>2</sup><https://askbot.org/en/questions/>

<sup>3</sup><http://www.question2answer.org/>

<sup>4</sup><http://www.lampcms.com/about.htm>

## 2. ANALÝZA

---

- kontrola syntaxe a upozornění na chyby bez spuštění skriptu,
  - online běh skriptů.
- Sdílení skriptů a spolupráce uživatelů:
  - sdílení podle URL adresy,
  - sdílení vložením do stránky pomocí iframe,
  - možnost spolupráce více uživatelů zároveň na jedné stránce.
- Výuka:
  - lekce,
  - procvičování.
- Socializace:
  - prostor pro diskuzi,
  - hodnocení uživatelů,
  - statistiky aktivity (počty zhlédnutí, poslední odpověď atd.).
- Organizace a vyhledávání:
  - popisy,
  - štítky.

### 2.2 Uživatelský průzkum

Úkolem uživatelského průzkumu bylo zjistit jaké funkce by uživatelé v aplikaci uvítali. V kvalitativním průzkumu byly vedeny rozhovory s lidmi z cílových uživatelských skupin, ze kterých následně vznikly hypotézy o použití aplikace. Tyto hypotézy byly v dotazníku potvrzeny. Průzkum ukázal zájem lidí o aplikaci, považují ji za užitečnou a dovedou si představit její používání.

Z průzkumu vyplynulo, že by aplikace měla podporovat:

- Spuštění skriptů:
  - nahrávání vstupních souborů,
  - zobrazení skriptu,
  - zobrazení vzorových dat,
  - dočasné uložení výstupu ze skriptu,
  - použití online vstupů,
  - automatické nasazení.
- Socializaci:
  - hlášení chyb,
  - diskuzi,
  - bodovací systém uživatelů,
  - vyhledávání,
  - sdílení skriptů,
  - sledování aktivity uživatelů,



- doporučování skriptů,
- tématické skupiny skriptů.
- Vytváření a ukládání skriptů:
  - verzování,
  - zobrazení/stažení skriptu,
  - parametrizace skriptu,
  - kombinování skriptů,
  - automatizace přes API,
  - omezení přístupu skriptu,
  - omezení práv na ukládání se skriptem,
  - upozornění na nové verze,
  - integrace s Google Drive.
- Editor:
  - rozlišení syntaxe,
  - kontrola syntaxe,
  - odkazy do manuálových stránek,
  - krokování skriptu.

### 2.2.1 Uživatelské skupiny

Předpokládání uživatelé aplikace byli rozděleni do dvou skupin podle jejich IT znalostí:

- IT znalí (studenti, nadšenci, učitelé, profesionálové)  
Využijí možnosti přehledného ukládání a kontroly syntaxe, naleznou řešení svých problémů a naplno využijí možnosti sdílení. Budou moci pohodlně své skripty spouštět a navzájem porovnávat. Mohou podporovat neodbornou veřejnost - vytvořit řešení jejich problému a jednoduše ho nasdílet.
- Neodborná veřejnost  
Získá možnost spouštět skripty aniž by uměla používat příkazový řádek, či měla nainstalovaný Linux. Dostane se jí pomoci s řešením jejich problémů s textem.

### 2.2.2 Vytvoření hypotéz (kvalitativní průzkum formou rozhovorů)

Cílem rozhovorů bylo získat od potenciálních uživatelů zpětnou vazbu na zamýšlené funkce aplikace.

Rozhovor se držel této osnovy:

1. Představení projektu.
2. Diskutování nad tématy k rozhovorům odvozených z cíle práce (kap. 1) a z významných vlastností existujících řešení (kap. 2.1.4):

## 2. ANALÝZA

---

- a) editace skriptů a testování,
- b) kontrola skriptů (syntaxe),
- c) podpora verzování,
- d) aplikace skriptů na soubor,
- e) sdílení skriptů,
- f) odkazování a řetězení skriptů,
- g) organizace a vyhledávání,
- h) výuka,
- i) socializace.

### 3. Diskuze nad aplikací, sběr zpětné vazby.

Aplikace se svými navrhovanými funkcemi byla velmi dobře přijata. Z navrhovaných funkcí byly respondenty odmítnuty pouze funkce týkající se výuky. IT znalí uživatelé kladli důraz na API a verzování. Pro IT neznalé uživatele bylo velmi důležité intuitivní ovládání bez technických znalostí a podpora produktů Microsoft Office.

#### 2.2.2.1 Bc. Jan Vlnas (zaměstnanec ČVUT FIT, student), 1. prosince 2016

Myšlenku řetězení transformačních skriptů z existujících služeb (jako GitHub) považuje za zajímavý nápad.

- Zdůraznil potřebu větvení a spojování a zmínil službu Yahoo Pipes.
- Jako příklad uvedl potřebu transformace dat z tabulky (Excel nebo Spreadsheet) do syntaxe YAML.

Za důležité považuje:

- možnost více vstupů s podporou online zdrojů,
- podporu API pro automatizované použití,
- podporu externích knihoven (vč. netriviálních, které je náročné instalovat lokálně),
- verzování,
- vlastní organizace skriptů a transformací (na rozdíl od služby Gist),
- oblíbenost skriptu či transformace (socializace obecně),
- jaké budou podporované jazyky,
- možnost stáhnout svoje skripty (pro lokální použití, zálohu).

Přímé aplikování skriptů na soubory považuje za dobrý nápad a upozornil na:

- problém parametrizace,
- specifičnost použití (různý počet různých souborů),
- zadávání vstupu (textové a souborové),
- potřebu aplikace na binární soubory.

### 2.2.2.2 Samuel Castillo (student ČVUT FIT), 6. prosince 2016

Líbí se mu nápad s online editací skriptů s kontrolou syntaxe. Zmínil portál JSfiddle<sup>5</sup>, který poskytuje online editování a testování HTML, CSS, JS. Obohatit systém o možnost výuky mu přijde zbytečné. Uvítal by:

- barevné rozlišení syntaxe,
- kontrolu syntaxe (včetně různých doporučení pro využívání různých funkcí),
- odkaz do manuálových stránek, ideálně přímo ze editoru klávesovou zkratkou (nejlépe zobrazovatelné ve vyskakovacím okně),
- drobné online úložiště pro dočasné uložení souborů (samotné skripty, výstupy, vstupy),
- při řetězení skriptů možnost zachování proměnných prostředí napříč skripty,
- nastavení přístupových práv ke skriptům pro skupiny uživatelů,
- počítání uživatelského skóre podle počtu využití uživatelského skriptu v řetězených skriptech ostatních uživatelů,
- možnost výběru z více linuxových distribucí v jejichž prostředí lze skripty spouštět.

Upozorňuje na nebezpečí spouštění skriptů na serveru:

- skripty by měly být spouštěny virtualizovaně,
- navrhuje využít Vagrant (virtuální prostředí, široká nabídka linuxových distribucí).

### 2.2.2.3 Dana Spořepová (studentka ČVUT FA), 8. prosince 2016

Funkčnost portálu ji velmi zaujala. Považuje ho za užitečný a v budoucnu by ho i použila. Klade důraz na:

- podporu Microsoft Office souborů,
- sdílení řešení podle URL adresy,
- možnost vložení vstupu formulářem i souborem,
- zpracování více souborů najednou,
- hledání dostupných řešení klíčovými slovy (štítky).

Hodnocení uživatelů (příspěvků) jí připadá jako dobrý nápad. Nemá zkušenosti s portály typu Stack Overflow.

### 2.2.2.4 Jan Maryško (student VŠE, podnikatel), 13. prosince 2016

Chválí nápad, vidí v něm obchodní potenciál. Líbí se mu možnost hodnotit uživatele. Požaduje podporu souborů ve formátu Microsoft Office. Přináší několik nápadů:

---

<sup>5</sup><https://jsfiddle.net/>

## 2. ANALÝZA

---

- možnost při spouštění skriptů:
  - si zobrazit skript,
  - přejít do diskuze o skriptu,
  - použít tlačítka kopírovat a vložit místo klávesových zkratk,
  - sdílet skript na sociálních sítích,
- rozdělení skriptů do kategorií a podkategorií,
- možnost automatizace přes emailovou adresu skriptu,
- živý chat (služba by byla zpoplatněna),
- možnost přidání skriptu do oblíbených,
- možnost výstupu více souborů do jediného,
- při zpracování skriptu zobrazovat reklamu.

Dále zmínil potřebu responzivního webu a více jazykových mutací.

### 2.2.2.5 Ing. Viktor Černý (vyučující ČVUT FIT), 15. prosince 2016

Práce ho zaujala, zdůraznil potřebu robustního návrhu jádra aplikace a použití API. Vidí využití i pro získávání informací z textu, např. výpis neplatných emailových adres.

- Za důležité považuje:
  - přehledné vyhledávání skriptů (vyhledávač, štítky),
  - schvalovací proces:
    - \* každá verze skriptu musí mít testovací data a očekávaný výsledek,
    - \* nová verze musí splnit testy všech předchozích verzí,
    - \* přítomnost moderátora tvorby skriptů,
  - přehledné a jednoduché uživatelské rozhraní,
  - podporu kancelářských formátů (Word, Excel, atd.).
- Upozornil na:
  - fakt, že ne vždy je k dispozici textový vstup - interní převod např. Excel -> CSV,
  - velký rozsah práce, doporučuje pro začátek vybrat jen některé funkce (diskuze, spouštění skriptů, verzování).

### 2.2.3 Ověření hypotéz (kvantitativní průzkum formou dotazníku)

Dotazník byl sestavován podle návodu dostupného na Quanda.cz<sup>6</sup>. Funkčnosti systému byly respondenty přijaty dobře, žádná funkce nebyla odmítnuta. Za nejdůležitější funkce respondenti označili:

<sup>6</sup>Viz <https://www.quanda.cz/blog/online-dotazniky/jak-vytvorit-dotaznik>

- zobrazení vzorových dat,
- nahrávání vstupních souborů,
- hlášení chyb skriptu,
- stažení skriptu,
- testovací data,
- barevné rozlišení syntaxe.

### 2.2.3.1 Krok 1: Cíl dotazníku

Cílem dotazníku bylo ověřit požadavky a připomínky získané v osobních rozhovorech. Získané informace byly využity při tvoření funkčních požadavků.

### 2.2.3.2 Krok 2: Návrh otázek

Otázky byly rozděleny podle tematických okruhů. Pro lepší pochopení otázek bylo přiloženo představení projektu.

**Představení projektu** Systém na organizaci, sdílení a spouštění skriptů zpracovávajících text. Uživatel potřebující automaticky zpracovat text (např. doplnění pevných mezer) může najít hotové řešení, kde si text nechá zpracovat. Když řešení nenalezne, může si skript sám vytvořit a následně ho nabídnout ostatním, nebo pokud neumí vytvářet skripty, popat jeho vytvoření.

### Otázky

1. Umíte psát a používat skripty (programovat)? (Jedna možnost)
  - a) Ne
  - b) Ano, učím se
  - c) Ano, profesionálně
  - d) Jiné
2. Využili byste následující funkce týkající se online spouštění skriptů? (\*)
  - a) Vkládání vstupních dat do textového pole (CTRL+C, CTRL+V)
  - b) Nahrávání vstupních souborů
  - c) Podpora souborů formátu Microsoft Word
  - d) Možnost zpětné vazby autorovi skriptu (udělení bodů, slovní hodnocení)
  - e) Možnost zobrazit si skript
  - f) Možnost použití online vstupů (odkaz na internet, systém si data stáhne sám)
  - g) Dočasné úložiště výstupu ze skriptu v uživatelském účtu
3. Využili byste následující funkce týkající se diskuzí (poptávek skriptů)? (\*)
  - a) Diskuze u každého zveřejněného skriptu, či zadání na skript

## 2. ANALÝZA

---

- b) Bodovací systém uživatelů
  - c) Sdílení skriptu jako odkaz na internet
  - d) Seznam oblíbených skriptů
  - e) Vyhledávač podle zadání skriptu
  - f) Profil uživatele se všemi zveřejněnými skripty
4. Využili byste následující funkce týkající se vytváření a ukládání skriptů? (\*)
- a) Verzování skriptů
  - b) Parametrizace skriptů
  - c) Řetězení skriptů za sebou
  - d) Možnost automatizace (ukládání, spouštění) přes API
  - e) Schvalovací proces úpravy již existujícího skriptu
  - f) Možnost stažení skriptu do PC
  - g) Přístupová práva ke skriptům
5. Využili byste následující funkce online editoru skriptů? (\*)
- a) Barevné rozlišení syntaxe
  - b) Kontrola syntaxe
  - c) Odkazy do manuálových stránek
  - d) Příložením testovacích dat
6. S jakými problémy u formátování textu se setkáváte? (více možností)
- a) Práce s mezerami (např. převod mezer na tabulátor)
  - b) Převod dat do jednotného formátu (např. jednotný formát telefonních čísel)
  - c) Konverze mezi formáty (např. XML, JSON)
  - d) Výskyty (např. hledání řádků obsahujících výraz)
  - e) Změna znaku konce řádku (CRLF)
  - f) Jiné
7. Co Vás napadá za funkčnost, která výše v dotazníku chybí?

Otázky označené hvězdičkou (\*) mají formu tabulky s možnostmi: Určitě ano, spíše ano, nevím, spíše ne, určitě ne.

### 2.2.3.3 Krok 3: Otestování otázek

Otázky byly předvedeny náhodným studentům. Techničtí studenti neměli problém s pochopením všech otázek. Netechničtí studenti neznali odpovědi na otázky číslo 4 a 5, klesalo jim u nich soustředění. Řešení: učinění těchto otázek dobrovolných, přidání varování, že otázka vyžaduje IT znalosti.

#### 2.2.3.4 Krok 4: Sestavení dotazníku

Dotazník byl sestaven pomocí Formulářů Google <sup>7</sup>. Obsahuje představení projektu a otázky. Otázky 1, 2, 3 jsou povinné, ostatní nejsou. Otázky 4 a 5 opatřeny popiskem „Tato otázka vyžaduje IT znalost a můžete jí přeskocit“.

#### 2.2.3.5 Krok 5: Otestování dotazníku

Dotazník byl před zveřejněním detailně otestován. Testování probíhalo formou rozhovorů. Po každém rozhovoru byl dotazník přepracován podle získaných připomínek. Detailní zápis z tohoto testování je v příloze A.1.

#### 2.2.3.6 Krok 6: Otestování otázek dotazníku

Testování na větším vzorku uživatelů (6 lidí) objevilo problém. Techničtí uživatelé zvládli dotazník vyplnit, avšak netechničtí nikoliv. Nedovedli si představit funkci systému, některé výrazy neznali (např. skript). Řešením by bylo opatřit každou otázku vysvětlivkou, což by ale způsobilo naprosté zneřehlednění dotazníku. Nakonec byla zvolena úprava popisu. Byl rozšířen o přeskočitelný popis funkčnosti systému, kde je na vzorové situaci vyprávěním vysvětlena základní funkčnost. Ve větách jsou použity výrazy, které se vyskytují v dotazníku, čímž jsou pro dotazovaného následně již známé.

#### 2.2.3.7 Hrubé vyhodnocení dotazníku

Dotazník byl vyplněn 71 respondenty v období mezi 4. a 12. březnem 2017. Z výsledků průzkumu vyplývá, že uživatelé by využili celou navrhovanou funkčnost aplikace, některé funkce více než jiné. Tyto rozdíly ovlivní úroveň nezbytnosti funkčních požadavků vytvářených podle výsledků průzkumu. Plné znění dotazníku je v příloze A.2, podrobné vyhodnocení dotazníku je v příloze A.3.

V části dotazníku, kde odpovídali techničtí i netechničtí respondenti, týkající se spouštění skriptů a socializace, byly nejoblíbenější následující funkce:

- zobrazení vzorových dat,
- nahrávání vstupních souborů,
- zobrazení skriptu,
- hlášení chyb (špatná funkce skriptu),
- vyhledávač podle popisu (štítku) skriptu,
- bodovací systém uživatelů podle hodnocení jejich skriptů,
- diskuze u každého zveřejněného skriptu, či u zadání na skript.

V čistě technické části dotazníku týkající se vytváření a ukládání skriptů či online editoru se jako hlavní funkce projeví tyto:

---

<sup>7</sup> [https://www.google.com/intl/cs\\_CZ/forms/about/](https://www.google.com/intl/cs_CZ/forms/about/)

- zobrazení/stažení skriptu,
- přiložení testovacích dat,
- parametrizace skriptů,
- verzování skriptů,
- nastavení přístupnosti skriptu (veřejný, kdokoli s odkazem, konkrétní uživatelé),
- barevné rozlišení syntaxe ,
- kontrola syntaxe.

Dle očekávání většina respondentů používá Stack Overflow, naopak užitečný nástroj ShellCheck zná jen velmi málo respondentů.

### 2.3 Požadavky na funkce

Požadavky na funkce vycházely z cíle práce (kap. 1) a z informací zjištěných v uživatelském průzkumu (kap. 2.2). Pro určení úrovně nezbytnosti bylo použito rozdělení na MUST/SHOULD/MAY<sup>8</sup>.

#### 2.3.1 Vytváření a ukládání skriptů

Systém poskytuje uživatelům možnost vytvoření skriptu. Skript je v kontextu aplikace chápán jako záznam o zřetězení externě uložených spustitelných skriptů, který byl uživatelem přidán do aplikace. Takto chápaný skript lze nazvat textovou transformací. Požadavky na vytváření a ukládání skriptů jsou:

- popis skriptu [MUST],
- verzování skriptu [MUST],
  - např. při každé provedené změně, je vytvořena nová verze, mezi verzemi lze procházet,
- kombinování skriptů - řetězení [MUST],
  - např. zřetězení dvou externích skriptů, kde první skript data přefiltruje a druhý zpracuje,
- nastavení přístupnosti skriptu [MUST],
  - např. omezení použití skriptu pouze pro uživatele vlastníci odkaz ke spuštění,
- automatizace spuštění skriptů přes API [SHOULD],
- zobrazení a stažení externího skriptu [SHOULD],
  - např. odkaz a externí službu, ve které je kód skriptu uložen,
- práva skript použít/zobrazit/upravit [SHOULD],

---

<sup>8</sup>Viz. <https://www.ietf.org/rfc/rfc2119.txt>



- např. skript mohou zobrazit a spustit všichni uživatelé, ale upravit pouze registrovaní,
- příložením testovacích dat [SHOULD],
  - např. otestování správné funkčnosti skriptu při vytváření změn,
- schvalovací proces [SHOULD],
  - kontrola proti poškození funkce skriptu vytvořením nové verze,
  - např. nová verze skriptu musí splnit testovací data všech předchozích verzí,
- parametrizace skriptů [SHOULD],
- kombinování skriptů - větvení [SHOULD],
  - např. skript pošle svůj výstup do dvou navazujících skriptů, kteří data zpracují a odešlou do dalšího skriptu, který oba výstupy sloučí a vyhodnotí,
- upozornění na nové verze skriptu [SHOULD],
- přivlastnění anonymního skriptu registrovaným uživatelem [SHOULD],
  - např. neregistrovaný uživatel v aplikaci vytvoří skript, posléze se v aplikaci registruje a přivlastní si skript který předtím vytvořil,
- zálohování kódu skriptu pro případ nedostupnosti [MAY],
  - např. repositář je z GitHub odstraněn,
- integrace s Google Drive [MAY].

### 2.3.2 Spouštění skriptů

Přihlášení i nepřihlášení uživatelé mohou v aplikaci spouštět skripty online bez jakékoliv znalosti programování či obsluhy příkazové řádky. Požadavky na spouštění skriptů jsou:

- vložení vstupních dat [MUST],
  - ruční zadání vstupu,
  - nahrávání vstupních souborů,
- zobrazení vzorových dat [SHOULD],
  - např. data, která na jednoduchém příkladu předvedou funkčnost skriptu,
- zpracování více vstupních dat zároveň [SHOULD],
  - např. sadu výrazů (regex) a sadu souborů, na které je má aplikovat,
- dočasné uložení výstupu ze skriptu [MAY],
- použití online vstupů [MAY],
  - na uvedeném odkazu si systém vstupní data pro skript stáhne sám.

### 2.3.3 Socializace

Nepostradatelnou funkcí aplikace jsou diskuze mezi uživateli. Uživatelé mohou požádat o vytvoření skriptů a následně diskutovat s řešiteli. Požadavky na socializaci jsou:

- diskuze u zadání na skript [MUST],
- bodovací systém uživatelů (gamifikace) [MUST],
- sdílení skriptů [MUST],
- hlášení chybné funkce skriptu [SHOULD],
- statistiky aktivity [SHOULD],
- seznam oblíbených skriptů [MAY].

### 2.3.4 Online editor

Online editor umožní uživatelům vytvářet a upravovat (své) skripty online. Poskytne jim také podpůrné nástroje pro usnadnění jejich práce. Přítomnost editoru v aplikaci má stupeň nezbytnosti [SHOULD]. Požadavky na funkce editoru jsou:

- barevné rozlišení syntaxe [SHOULD],
- kontrola syntaxe [SHOULD],
- dokončování rozepsaného příkazu [MAY],
- odkazy do manuálových stránek [MAY],
- krokování skriptu [MAY],
- automatické formátování kódu [MAY].

### 2.3.5 Organizace a vyhledávání

Skripty jsou v aplikaci označeny štítkem, podle kterého mohou být následně i vyhledány. Další možnost hledání skriptu je podle jeho názvu a popisu. Požadavky na organizaci a vyhledávání jsou:

- štítkování skriptů [MUST],
- vyhledávání podle popisu skriptu [MUST],
- vyhledávání podle štítku skriptu [MUST],
- automatické oštítkování skriptu [MAY].

## 2.4 Požadavky na kvality

Požadavky na kvalitu se řídily dokumentem „Pravidla a zásady projektů FIT“ uvedeným v příloze B. Zde jsou uvedeny požadavky, které se od zmíněného dokumentu liší.

### 2.4.1 Obecné

Obecné požadavky na projekty FIT.

- Architektura a integrace do infrastruktury:
  - aplikace není součástí informačního fakultního systému,
  - zvolené hotové řešení striktně neodděluje frontend a backend.
- Webová přístupnost:
  - zvolené hotové řešení plně nerespektuje požadavky WCAG 2.0 AA.
- Kód aplikace:
  - verzování bude vedeno na verzovacím systému GIT,
  - z důvodu nedostatku času nebude kontrola kvality kódu prováděna,
  - vývoj nebude řízený testy.
- Provoz, údržba a rozvoj aplikace a podpora uživatelů:
  - aplikace bude vyvíjena a používána výhradně v testovacím prostředí.
- Bezpečnost a ochrana osobních údajů:
  - aplikace nevyužívá autorizaci FIT,
  - aplikace bude používána výhradně v testovacím prostředí, komunikace nebude vedena přes HTTPS,
  - nařízení děkana o ochraně osobních údajů je vzhledem k povaze aplikace irelevantní.

### 2.4.2 Koncepty webových aplikací

Zde jsou uvedeny návrhy konceptů webových aplikací:

- RSS:
  - poskytování notifikací [SHOULD],
    - \* nový skript ze sledované kategorie skriptů,
    - \* příspěvek v diskuzi u zvoleného skriptu.
- Personalizace:
  - cílená reklama [MAY],
  - přizpůsobení obsahu dle toho, zda uživatel obvykle skripty vytváří či pouze spouští [MAY].
- Customizace:
  - nastavení rozložení a motivu stránek aplikace [MAY].
- Folksonomie (tagování):
  - aplikace umožňuje tagování skriptů a diskuzí [MUST].
- Social networking:

## 2. ANALÝZA

---

- diskuze a sdílení skriptů mezi uživateli [MUST].
- Real-time web:
  - okamžité zobrazení [SHOULD]:
    - \* přidaného příspěvku v diskuzi,
    - \* výsledku skriptu,
    - \* bodů za ohodnocení.
- Crowdsourcing:
  - aplikace umožňuje uživatelům uveřejnit výzvu (zadání) na skript, kterou následně řeší ostatní uživatelé [MUST].
- Kolaborace:
  - práce více uživatelů na jednom skriptu zároveň [MAY].
- Průvodce:
  - úvodní průvodce s představením funkce aplikace [MAY].
- Konfigurátory (rozšířených dotazů vyhledávání, parametrů služby):
  - skripty v aplikaci mohou být uzpůsobeny pro parametrizaci [MUST].
- Gamifikace:
  - uživatelé dostávají body za svojí aktivitu v aplikaci [MUST].
- Mikrodata:
  - v popisech skriptu [MAY],
  - v diskuzích [MAY].?

### 2.5 Vyhodnocení požadavků vůči existujícím řešením

Vyhodnocení požadavků vůči existujícím řešením srovnalo vlastnosti existujících řešení (kap. 2.1) s funkčními požadavky na aplikaci (kap. 2.3). Toto srovnání je vyobrazeno na následujících tabulkách 2.1 a 2.2.

Tabulka 2.1: Srovnání existujících řešení (Jdoodle, Tutorialspoint, Learn Shell) a funkčních požadavků na aplikaci

Funkce	Jdoodle	Tutorialspoint	Learn Shell
Popis skriptu	ne	ne	ne
Verzování skriptu	ne	ano	ne
Kombinování skriptů	ne	ne	ne
Nastavení přístupnosti skriptu	částečně <sup>9</sup>	částečně <sup>10</sup>	ne

*Pokračování na další straně*

<sup>9</sup>Přímé nastavení zde není. Přístupnost lze ovlivnit (ne)nasdílením odkazu na skript.

<sup>10</sup>Přímé nastavení zde není. Přístupnost lze ovlivnit (ne)nasdílením odkazu na skript.

## 2.5. Vyhodnocení požadavků vůči existujícím řešením

Tabulka 2.1 – Pokračování na předchozí straně

Funkce	Jdoodle	Tutorialspoint	Learn Shell
Automatizace přes API	ne <sup>11</sup>	ne	ne
Zobrazení a stažení skriptu	ano	ano	částečně <sup>12</sup>
Práva skript použít/zobrazit/upravit	ne	ne	ne
Přiložení testovacích dat	ne	ne	ne
Schvalovací proces	ne	ne	ne
Parametrizace skriptů	ne	ano	ne
Upozornění na nové verze skriptu	ne	ne	ne
Přivlastnění anonymního skriptu	ne	ne	ne
Zálohování kódu skriptu	ano	ano	ne
Integrace s Google Drive	ne	ano	ne
Vložení vstupních dat	ano	ano	ne
Zobrazení vzorových dat	ne	ne	ne
Zpracování více vstupních dat zároveň	ne	ne	ne
Dočasné uložení výstupu ze skriptu	ne	ne	ne
Použití online vstupů	ne	ano	ne
Diskuze u zadání na skript	ne	ne	ne
Bodovací systém uživatelů	ne	ne	ne
Sdílení skriptů	ano	ano	ne
Hlášení chybné funkce skriptu	ne	ne	ne
Statistiky aktivity	ne	ne	ne
Seznam oblíbených skriptů	ne	ne	ne
Barevné rozlišení syntaxe	ano	ano	ano
Kontrola syntaxe	částečně <sup>13</sup>	částečně <sup>14</sup>	ano
Odkazy do manuálových stránek	ne	ne	ne
Krokování skriptu	ne	ne	ne
Automatické formátování	ne	ne	ne
Dokončování rozepsaného příkazu	ano	ne	ne
Štítkování skriptů	ne	ne	ne
Vyhledávání podle popisu skriptu	ne	ne	ne
Vyhledávání podle štítku skriptu	ne	ne	ne
Automatické oštitkování skriptu	ne	ne	ne

<sup>11</sup>V přípravě.

<sup>12</sup>Skript lze zobrazit a zkopírovat.

<sup>13</sup>Skript je spuštěn, chyby jsou objeveny až za běhu.

<sup>14</sup>Skript je spuštěn, chyby jsou objeveny až za běhu.

## 2. ANALÝZA

Tabulka 2.2: Srovnání existujících řešení (Webminal, Shell-Check, Stack Overflow) a funkčních požadavků na aplikaci

Funkce	Webminal	ShellCheck	Stack Overflow
Popis skriptu	ne	ne	částečně <sup>15</sup>
Verzování skriptu	ne	ne	ne
Kombinování skriptů	částečně <sup>16</sup>	ne	ne
Nastavení přístupnosti skriptu	ne	ne	ne
Automatizace přes API	ne	ne	ne
Zobrazení a stažení skriptu	částečně <sup>17</sup>	částečně <sup>18</sup>	částečně <sup>19</sup>
Práva skript použít/zobrazit/upravit	ne	ne	ne
Přiložení testovacích dat	ano	ne	ne
Schvalovací proces	ne	ne	ne
Parametrizace skriptů	ano	ne	ne
Upozornění na nové verze skriptu	ne	ne	ne
Přivlastnění anonymního skriptu	ne	ne	ano
Zálohování kódu skriptu	ano	ne	ano
Integrace s Google Drive	ne	ne	ne
Vložení vstupních dat	ano	ne	ne
Zobrazení vzorových dat	ne	ne	částečně <sup>20</sup>
Zpracování více vstupních dat zároveň	ano	ne	ne
Dočasné uložení výstupu ze skriptu	ano	ne	ne
Použití online vstupů	ano	ne	ne
Diskuze u zadání na skript	ne	ne	ano
Bodovací systém uživatelů	ne	ne	ano
Sdílení skriptů	ne	ne	částečně <sup>21</sup>
Hlášení chybné funkce skriptu	ne	ne	ne
Statistiky aktivity	ne	ne	ano
Seznam oblíbených skriptů	ne	ne	částečně <sup>22</sup>
Barevné rozlišení syntaxe	ne	ano	ano
Kontrola syntaxe	ano	ano	ne
Odkazy do manuálových stránek	ne	ne	ne
Krokování skriptu	ne	ne	ne
Automatické formátování	ne	ne	ne
Dokončování rozepsaného příkazu	ne	ne	ne
Štítkování skriptů	ne	ne	ano

*Pokračování na další straně*

<sup>15</sup>Do příspěvku lze vložit kód skriptu společně s dalším textem.

<sup>16</sup>Zřetězení skriptů lze docílit spuštěním více skriptů v terminálu za sebou.

<sup>17</sup>Skript lze zobrazit a zkopírovat

<sup>18</sup>Skript lze zobrazit a zkopírovat

<sup>19</sup>Skript lze zobrazit a zkopírovat

<sup>20</sup>Vzorová data mohou být uvedena v příspěvku ve kterém je uvedený (nasdílený) skript.

<sup>21</sup>V příspěvku lze uvést odkaz na skript.

<sup>22</sup>Příspěvek (obsahující skript) lze uložit do oblíbených.

## 2.5. Vyhodnocení požadavků vůči existujícím řešením

---

Tabulka 2.2 – Pokračování na předchozí straně

<b>Funkce</b>	<b>Webminal</b>	<b>ShellCheck</b>	<b>Stack Overflow</b>
Vyhledávání podle popisu skriptu	ne	ne	ano
Vyhledávání podle štítku skriptu	ne	ne	ano
Automatické oštitkování skriptu	ne	ne	ne





---

# Návrh

V návrhové části práce byla aplikace navržena s ohledem na informace získané v analýze. Návrh aplikace zpracovával zejména nejzákladnější požadavky na funkcionalitu (stupně nezbytnosti MUST).

Nejprve byly v aplikaci určeny základní vykonávané procesy. Případy užití následně tyto procesy navzájem provázaly.

Další fází bylo navržení architektury aplikace, včetně výběru použitých technologií a hotových řešení. S ohledem na vybrané hotové řešení webového portálu byly vytvořeny prototypy stěžejních obrazovek aplikace.

Na závěr návrhové části byl vytvořen papírový prototyp aplikace, pomocí kterého byl ověřen návrh procesů a rozložení prvků na obrazovkách aplikace.

## 3.1 Procesy

Byly vytvořeny popisy stěžejních procesů aplikace. Proces v aplikaci popisuje jeden standardní úkol uživatele při použití aplikace. Při navrhování procesů byl kladen důraz na jejich co největší jednoduchost. Každý proces je popsán a má svůj diagram.

### 3.1.1 Vyhledání a zobrazení diskuze o skriptu {P1}

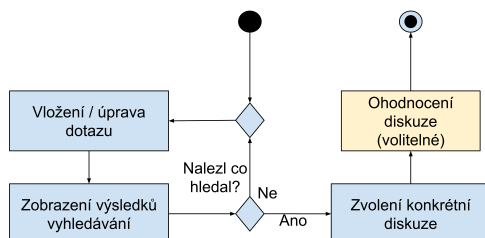
Uživatel může vyhledat zadání skriptu zadáním klíčových slov do vyhledávacího pole. Systém v reakci na zadaná data aktualizuje výsledky hledání. Kliknutím na konkrétní diskusi přejde uživatel na její detail.

1. Vyhledávání [iterativní upřesňování]
  - 1.1. Vložení/upřesnění dotazu do vyhledávacího pole na úvodní obrazovce.
  - 1.2. Aplikace zobrazí výsledky hledání na stránce vyhledávání.
2. Uživatel zvolí konkrétní diskusi.

### 3. NÁVRH

---

3. Aplikace zobrazí detail diskuze.
4. Uživatel volitelně diskusi ohodnotí. Pokud kladně, vyjádří tím svoji podporu dané diskusi.

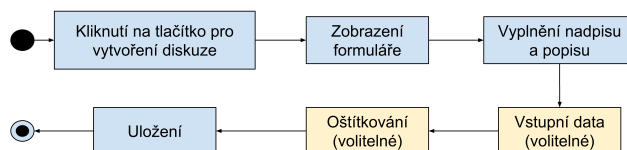


Obrázek 3.1: Diagram procesu 1

#### 3.1.2 Vytvoření požadavku na skript {P2}

Uživatel může založit novou diskusi, do které napíše zadání na skript. Vytvořit diskusi je možné přímo z úvodní stránky aplikace. Pokud je uživatel přihlášen získá body a bude moci získávat notifikace, pokud někdo v diskusi přidá příspěvek.

1. Uživatel klikne na tlačítko pro přidání nového zadání (diskuze) na úvodní obrazovce.
2. Aplikace zobrazí stránku pro vytváření nové diskuze.
3. Uživatel vyplní nadpis příspěvku a jeho popis.
4. Uživatel volitelně do popisu přidá příklady vstupních dat.
5. Uživatel volitelně vyplní štítky.
6. Uživatel potvrdí vytvoření.



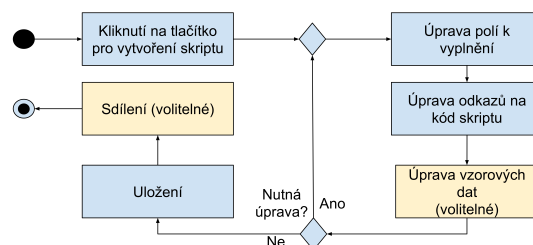
Obrázek 3.2: Diagram procesu 2

#### 3.1.3 Vytvoření skriptu {P3}

Uživatel může vytvořit nový skript. Vytvořit nový skript je možné přímo z úvodní stránky aplikace. Pro vytvoření skriptu musí být uživatel přihlášen.

1. Uživatel klikne na tlačítko pro vytvoření nového skriptu.

2. Aplikace zobrazí stránku pro vytváření skriptu.
3. Vyplňování [iterativní vyplňování]
  - 3.1. Uživatel edituje informace o skriptu (název, popis, štítky).
  - 3.2. Uživatel edituje zřetězení skriptů přidáním či odebráním záznamu o externím skriptu (odkaz do GitHub repozitáře, soubor ke spuštění, commit). Minimální počet přidanych externích skriptů je jeden.
  - 3.3. Uživatel volitelně upraví vzorová data.
4. Uživatel uloží skript.
5. Uživatel volitelně zkontroluje odkaz na skript pro jeho sdílení.

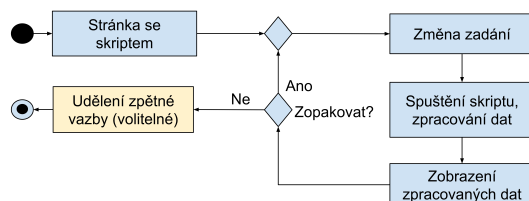


Obrázek 3.3: Diagram procesu 3

### 3.1.4 Spuštění skriptu {P4}

Uživatel může spustit nasdílený skript přes odkaz. Odkaz se může nacházet kdekoli v aplikaci nebo i mimo ni. Přihlášený uživatel může skript ohodnotit.

1. Uživatel zobrazí stránku se skriptem, aplikace vloží do pole vstupních dat data vzorová (pokud jsou přítomna).
2. Uživatel zadá vstupní data:
  - a) do textového pole.
  - b) nahráním souboru.
3. Uživatel stiskne tlačítko spustit skript.
4. Aplikace zpracuje vložená data.
5. Aplikace zpracovaná data:
  - a) zobrazí v textovém poli.
  - b) stáhne do uživatelova zařízení.
6. Uživatel se může vrátit do bodu 2.
7. Uživatel volitelně udělí skriptu zpětnou vazbu udělením bodů (plus,mínus).



Obrázek 3.4: Diagram procesu 4

## 3.2 Případy užití

Případy užití vycházejí z procesů, mezi kterými vytvářejí souvislosti. Každý případ užití obsahuje zadání, předpoklady a seznam mapovaných procesů.

### 3.2.1 Vyhledání diskuze a použití skriptu {PU1}

Uživatel dostal od svého vedoucího za úkol převést česká telefonní čísla zákazníků do jednotného formátu, aby je mohl importovat do nového firemního informačního systému. Uživatel chce najít diskuzi, ve které se tento problém řeší a v ní skript, na který aplikuje svá data.

#### Vstupní podmínky

- Existuje diskuze o formátování telefonních čísel.
- V diskuzi je nasdílený skript na formátování čísel.

#### Průběh

1. Uživatel vloží do vyhledávacího pole „formátování telefonních čísel“.
2. Aplikace zobrazí výsledky vyhledávání.
3. Uživatel otevře detail diskuze „Formátování českých telefonních čísel“.
4. Uživatel mezi příspěvky nalezne odkaz na skript a ten otevře.
5. Aplikace zobrazí stránku pro spuštění skriptu.
6. Uživatel nahraje svůj textový soubor s telefonními čísly a spustí zpracování
7. Aplikace vrátí zpracovaný soubor.
8. Uživatel je se zpracovanými daty spokojený, ohodnotí skript kladně.

#### Mapované procesy

- {P1}
- {P4}

### 3.2.2 Vyhledání diskuze a založení nové diskuze {PU2}

Uživatel se zabývá formátováním telefonních čísel, chce tedy najít diskuzi, ve které se řeší formátování telefonních čísel, protože ho zajímá, jak tento problém řešili jiní uživatelé. Pokud žádnou nenalezne, vytvoří diskuzi novou, aby v ní získal názory ostatních.

#### Vstupní podmínky

- Neexistuje diskuze o formátování telefonních čísel.

#### Průběh

1. Uživatel vloží do vyhledávacího pole „formátování telefonních čísel“.
2. Aplikace zobrazí prázdný výsledek vyhledávání.
3. Uživatel stiskne tlačítko vytvořit diskuzi a aplikace zobrazí stránku pro vytvoření.
4. Uživatel vyplní titulek diskuze - „Formátování českých telefonních čísel“.
5. Uživatel vyplní popis svého problému a štítky (formátování, telefonní čísla).
6. Uživatel uloží diskuzi.

#### Mapované procesy

- {P1}
- {P2}

### 3.2.3 Vytvoření skriptu a jeho nasdílení do diskuze {PU3}

Uživatel vytvořil ve svém GitHub repositáři kód skriptu, který by rád přidal do aplikace. Svůj skript chce následně přidat do diskuze, pamatuje si ale pouze její název.

#### Vstupní podmínky

- Uživatel má kód skriptu uložen v GitHub repositáři.
- Uživatel je na stránce vytvoření skriptu.
- Existuje diskuze „formátování telefonních čísel“

#### Průběh

1. Uživatel na stránce pro vytvoření skriptu vyplní:
  - a) název skriptu „formátovač českých telefonních čísel“,
  - b) popis, ve kterém zmíní funkci skriptu, podporované formáty,
  - c) štítky „formátování, telefonní čísla“,

### 3. NÁVRH

---

- d) odkaz na repositář v GitHub - „<https://github.com/uživatel/repositář>“,
  - e) soubor ke spuštění - „script.sh“,
  - f) commit - zvolenou verzi ke spuštění.
2. Uživatel vyplní vzorová data, na kterých uživatelé skriptu snadno pochopí jeho funkčnost.
  3. Uživatel uloží skript, aplikace ho přesměruje na detail nově založeného skriptu, kde si uživatel zkopíruje odkaz.
  4. Uživatel vloží do vyhledávače název diskuze „formátování telefonních čísel“.
  5. Aplikace zobrazí výsledky vyhledávání.
  6. Uživatel přejde do diskuze a přidá příspěvek, ve kterém uvede odkaz na svůj skript.

#### Mapované procesy

- {P1}
- {P3}

### 3.3 Architektura aplikace

Architektura aplikace byla rozdělena do dvou komponent. První komponentou je webový portál, jenž zajišťuje správu uživatelů, otázek a skriptů. Pro její implementaci byla vybrána platforma Question2Answer. Druhá komponenta se stará o komunikaci s odděleným systémem pro běh skriptů (Docker). Vztahy mezi komponentami zachycuje schéma architektury uvedené na konci kapitoly.

#### 3.3.1 Webový portál

Tato podkapitola popisuje výběr softwaru, který pokryje dvě subkomponenty. Komponentu frontendu, zajišťující zobrazování informací uživateli, a komponentu backendu, která tyto informace zpracovává a ukládá.

Pro vývoj byla použita platforma Question2Answer. Volba této platformy vzešla ze srovnání s jinými platformami poskytujícími funkcionalitu webového fóra (OSQA, LampCMS, Askbot) a ze srovnání s řešením implementace za použití frameworku Nette.

U porovnávaných řešení byly sledovány tyto vlastnosti:

- Je publikováno pod volnou licencí, která umožní legální úpravy systému.
- Pokrývá minimum poskytovaných funkcí potřebných pro portál otázek a odpovědí (otázky a odpovědi, štítky, vyhledávání, uživatelské účty, gamifikace).

- Umožňuje úpravy systému pomocí témat a pluginů (funkcionalita, vzhled atd.), aby nebylo nutné upravovat kód systému, což umožní kompatibilitu s budoucími verzemi.
- Nabízí samostatnou dokumentaci, nebo alespoň okomentovaný kód.
- Existuje aktivní komunita uživatelů, kteří používají a vylepšují systém.
- Existuje API, které umožní komunikaci s cizími systémy.
- Backend je oddělený od frontendu, což umožní snadnější výměnu jednotlivých komponent.

#### 3.3.1.1 OSQA

OSQA je opensource verze placeného systému AnswerHub. Protože se jedná o neplacenou verzi, je systém velmi omezen. Jeho vývoj je již zastaven, tudíž není vhodný pro další rozšiřování.

- Výhody:
  - Je dostupný pod licencí GPLv3.
  - Pokrývá požadované minimum funkcí.
  - V minulosti byl komunitou velmi oblíben.
- Nevýhody:
  - Založen na již zastaveném projektu CNProg <sup>23</sup>.
  - Má omezenou funkcionalitu, jelikož je verze zdarma jinak placené aplikace.
  - Nepodporuje žádné přizpůsobení nebo pluginy.
  - Jeho kód je neokomentovaný, dokumentace nebyla nalezena.
  - Vývoj je zastaven, aplikovány jsou pouze opravy chyb. Poslední commit je 2 roky starý (viz. repositář <sup>24</sup>).
  - Ostatním službám nevystavuje API.
  - Jeho backend není od frontendu oddělen přes API, ale přes interface (programová konstrukce). [9]

#### 3.3.1.2 LampCMS

LampCMS je webový portál se zajímavými funkcemi integrace se službami třetích stran. Nyní je ale projekt již opuštěný a neudržovaný.

- Výhody:
  - Je dostupný pod licencí LGPLv3.
  - Podporuje integraci se službami Facebook, Twitter, Google a LinkedIn.
  - Pokrývá požadované minimum funkcí a nabízí velké množství funkcí navíc (např. zabezpečení, moderování administrátorem atd).

---

<sup>23</sup><https://github.com/chagel/CNPROG>

<sup>24</sup><https://github.com/dzone/OSQA/tree/master>

### 3. NÁVRH

---

- Jeho kód je zdokumentovaný.
- Nabízí možnost pluginů. Jsou omezené, naslouchají událostem v systému.
- Nevýhody:
  - Nepodporuje možnost rozšíření systému pokročilejšími pluginy.
  - Vývoj zastaven - poslední commit je 2 roky starý (viz. repositář<sup>25</sup>).
  - Ostatním službám nevystavuje API.
  - Jeho backend není od frontendu oddělen přes API, ale přes interface (programová konstrukce). [10]

#### 3.3.1.3 Askbot

Askbot je komunitou aktivně udržovaný nástupce projektu OSQA. Neposkytuje široké možnosti úprav formou témat a pluginů.

- Výhody:
  - Je dostupný pod licencí GPLv3.
  - Pokrývá požadované minimum funkcí a nabízí funkce navíc.
  - Poskytuje jednoduché API jen pro čtení několika základních údajů.
  - Je možné upravit CSS stránek formou skinu. Aplikace je komunitou aktivně vyvíjena (viz. repositář<sup>26</sup>).
- Nevýhody:
  - Stručná dokumentace, kód není okomentovaný.
  - Několik stránek<sup>27</sup> využívajících Askbot. Stránky jsou si velmi podobné, důkaz špatné upravovatelnosti.
  - Jeho backend není od frontendu oddělen přes API, ale přes interface (programová konstrukce). [11]

#### 3.3.1.4 Question2Answer

Question2Answer je populární platforma pro otázky a odpovědi. Poskytuje velmi široké možnosti úprav vzhledu i funkčnosti formou pluginů a témat. Tvorba nových je dobře zdokumentována.

- Výhody:
  - Je dostupný pod licencí GPLv2.
  - Pokrývá požadované minimum funkcí a nabízí velké množství funkcí navíc (např. zabezpečení, moderování administrátorem atd).

---

<sup>25</sup><https://github.com/snytkine/LampCMS>

<sup>26</sup><https://github.com/ASKBOT/askbot-devel>

<sup>27</sup><https://askbot.org/en/question/2488/list-of-askbot-sites/>



- Propracovaný systém pluginů a témat, které umožňují rozsáhlé úpravy [12].
- Aplikace je komunitou aktivně vyvíjena (viz. repositář <sup>28</sup>).
- Zdokumentovaný kód, tutoriály na vytváření pluginů.
- Aplikace je využívána na více než 20 000 stránek <sup>29</sup>. Stránky se od sebe velmi liší.
- Na platformě jsou založeny obchody s pluginy a tématy Q2Amarket<sup>30</sup>, QA-themes<sup>31</sup> a Q2Apro<sup>32</sup>.
- Nevýhody:
  - Nemá nativní API, existuje plugin pro základní funkcionalitu (čtení otázek a odpovědí, informace o uživatelích atd). Viz. repositář <sup>33</sup>).
  - Jeho backend není od frontendu oddělen přes API, ale přes interface (programová konstrukce). [13]

#### 3.3.1.5 Framework Nette

Implementace portálu pomocí frameworku Nette<sup>34</sup> nabízí velmi široké možnosti úprav za cenu mnohem vyšší časové náročnosti ve srovnání s existujícími platformami na otázky a odpovědi.

- Výhody:
  - Je dostupný pod licencí GPLv3.
  - Funkce a zjev aplikace mohou být prakticky neomezené.
  - Framework má velkou komunitu, která může pomoci s problémy.
  - Framework je zdokumentovaný.
  - Vytvořený systém by měl oddělený frontend a backend přes API a toto API by vystavoval dalším systémům.
- Nevýhody:
  - Hrozí vytvoření zbytečných chyb, které jsou již v hotových aplikacích ošetřeny.
  - Je nutné důkladné testování.
  - Vysoká časová náročnost. Je nutné implementovat funkce, které nejsou přímým předmětem této práce.

---

<sup>28</sup><https://github.com/q2a/question2answer/tree/1.8>

<sup>29</sup><http://www.question2answer.org/sites.php>

<sup>30</sup><http://store.q2amarket.com/>

<sup>31</sup><http://qa-themes.com/>

<sup>32</sup><https://www.q2apro.com/>

<sup>33</sup><https://github.com/miky91/q2a-rest-api>

<sup>34</sup><https://nette.org/>

### 3.3.1.6 Vyhodnocení porovnání

Obecné řešení implementované pomocí frameworku Nette nevyváží možnostmi úprav svoji vysokou časovou náročnost implementace a testování. Z existujících aplikací je pro rozšiřování nejvíce vhodná platforma Question2Answer z důvodu vysoké přizpůsobitelnosti pomocí pluginů a témat, která nejméně omezuje případné úpravy a rozšíření. Platforma je osvědčena v praxi a podporována komunitou. Z těchto důvodů jsou její nedostatky akceptovatelné.

### 3.3.2 Komponenta pro spouštění skriptů

Tato kapitola popisuje strukturu součásti systému, která zajišťuje spouštění skriptů v odděleném prostředí Docker. Tato komponenta je napsána v programovacím jazyce Java, ve kterém existují knihovny pro práci se zmíněnými systémy [14]. Komponenta komunikuje s bash skriptem (tzv. Masterscript) umístěným v odděleném prostředí, který stahuje kód uživatelských skriptů a stará se o jejich spuštění. Návrh této komponenty byl konzultován v [15].

#### 3.3.2.1 Verzovací systém

Pro ukládání a verzování kódu externích skriptů je použita služba GitHub<sup>35</sup>. V budoucnu se počítá s rozšířením i o další Git služby (např. GitLab<sup>36</sup>, Bitbucket<sup>37</sup>).

#### 3.3.2.2 Spouštění skriptů

Spuštěné skripty je z hlediska bezpečnosti nutné oddělit od okolního systému. Pro toto oddělení byl zvolen systém Docker. Ten byl srovnáván s dalšími možnostmi zabezpečení (Chroot, VirtualBox).

Ze srovnání vyšel nejlépe systém Docker, který nabízí dobrou bezpečnost, je rychlý a nespotřebovává systémové prostředky navíc.

#### Chroot

Vestavěný nástroj do OS Linux umožňující omezení práv aplikací a uživatelů. Není zcela bezpečný, existují metody na jeho prolomení.

- Výhody:
  - Izoluje spuštěné procesy a jejich podprocesy.
  - Izoluje souborový systém.
- Nevýhody:
  - Existují způsoby, jak ho prolomit [16].

---

<sup>35</sup><https://github.com/about>

<sup>36</sup><https://about.gitlab.com/>

<sup>37</sup><https://bitbucket.org/product>

- Běžící programy vyžadují kopii knihoven.
- Složitější na nastavení. [17]

#### Docker

Program izolující aplikace do kontejneru, ve kterém běží operační systém sdílející jádro s hostujícím systémem. Toto řešení je bezpečné a rychlé.

- Výhody:
  - Bezpečné oddělení spouštěné aplikace od okolního systému [18].
  - V kontejneru je obsažen celý operační systém s nainstalovanými knihovnami, nástroji atd.
  - Kontejneru lze omezit systémové prostředky.
  - Jeho režie na spuštění je velmi malá.
- Nevýhody:
  - Pravděpodobně existuje možnost prolomení (tzv. útok nultého dne<sup>38</sup>). [19]

#### VirtualBox

Poskytuje plnou virtualizaci operačního systému. Poskytuje vyšší míru ochrany za cenu vyšších požadavků na systémové prostředky a čas.

- Výhody:
  - Bezpečné oddělení spouštěné aplikace od hostujícího systému.
- Nevýhody:
  - Pravděpodobně existuje možnost prolomení (tzv. útok nultého dne).
  - Velká režie na spuštění virtualizovaného OS.
  - Režie prostředků na běh virtualizovaného OS. [20]

#### 3.3.3 Schéma architektury

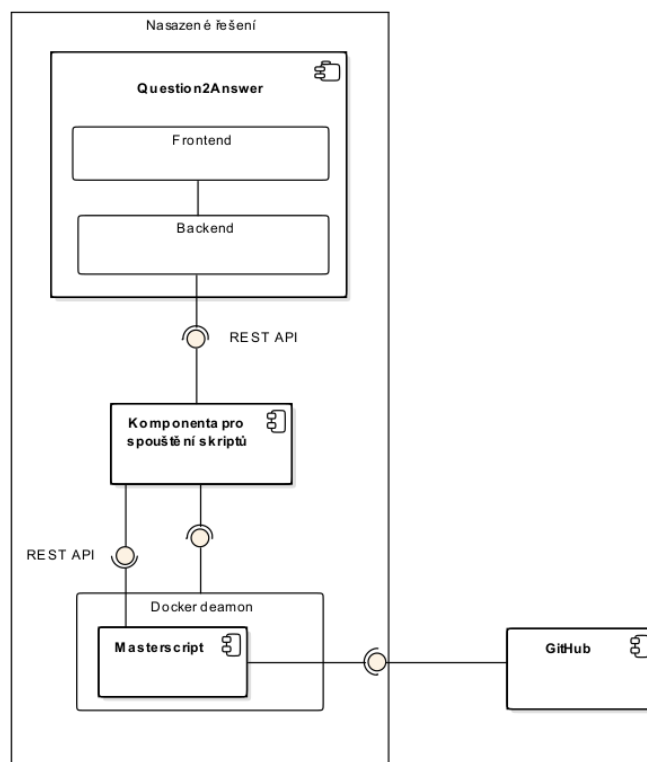
Obrázek popisující vztahy mezi jednotlivými komponentami systému.

- Frontend a backend jsou spojeny v platformě Question2Answer pomocí programové konstrukce (interface)<sup>39</sup>.
- Question2Answer skrz REST API volá službu na spuštění skriptů.
- Služba na spuštění skriptů ovládá oddělené prostředí, ve kterém spouští Masterscript.
- Masterscript stahuje kód skriptu z Github, získává vstupní data od komponenty spuštění, spouští uživatelův skript a odesílá zpracovaná data zpět.

---

<sup>38</sup><http://www.pctools.com/security-news/zero-day-vulnerability/>

<sup>39</sup><http://docs.oracle.com/javase/tutorial/java/IandI/interfaceDef.html>



Obrázek 3.5: Schéma architektury

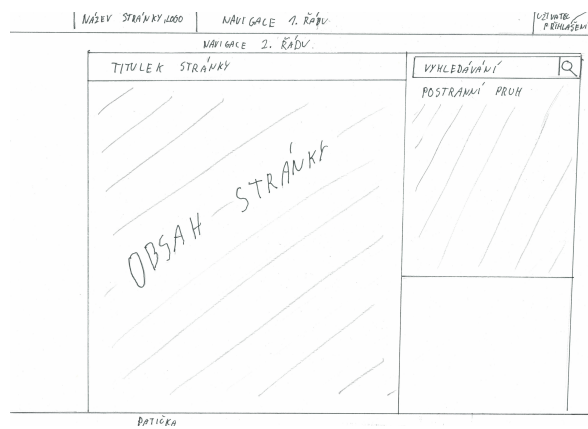
## 3.4 Prototypování

Prototypování stránek bylo inspirováno metodikou UCD [21]. Nebylo navrhováno kompletní uživatelské rozhraní, cílem prototypování bylo primárně určit rozložení prvků na jednotlivých stránkách. Prototypy byly navrženy s ohledem na zvolené řešení webového portálu. V kontextu Pravidel a zásad projektů FIT tato kapitola zastupuje LoFi a HiFi prototypy.

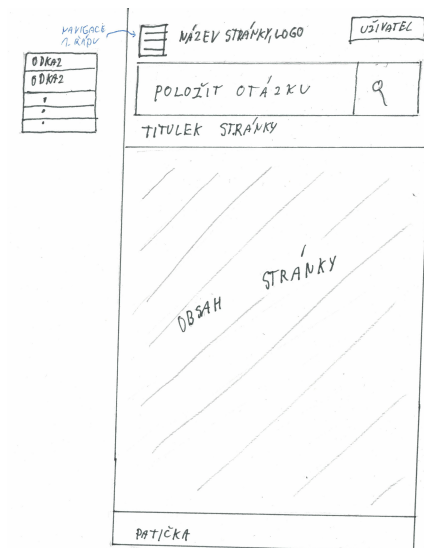
### 3.4.1 Náčrty celku

Náčrty celku ukazují základní rozložení prostředí aplikace. Toto rozložení je na užších obrazovkách (obr. 3.7) odlišné než na ploše velké obrazovky počítače (obr. 3.7). Rozložení je převzato od zvoleného řešení.

### 3.4. Prototypování



Obrázek 3.6: Normální rozložení



Obrázek 3.7: Rozložení pro užší obrazovky

#### 3.4.2 Náčrty stránek

V rámci prototypování byly navrhovány čtyři stránky, které jsou nutné pro poskytování funkcionality vytváření a spouštění skriptů:

- stránka nového skriptu a jeho editace,
- stránka detailu skriptu,
- stránka spuštění skriptu,
- stránka s výpisem skriptů.

### 3.4.2.1 Stránka nového skriptu (editace skriptu)

Tato stránka slouží uživateli pro zadávání a editaci veškerých informací o skriptu. Uživatelem zadané informace jsou kontrolovány, při zadání neplatných dat jsou zobrazeny chybové hlášky. U trojice polí (odkaz do GitHub, soubor ke spuštění, commit) je možné tlačítka „přidat skript“ a „odebrat skript“ měnit počet jejich výskytů na stránce, čímž jsou přidávány (resp. odebrány) skripty k zřetězení. V režimu editace musí uživatel povinně v příslušném poli popsat provedené změny.

Obrázek 3.8: Stránka nového skriptu (editace skriptu)

### 3.4.2.2 Stránka detailu skriptu

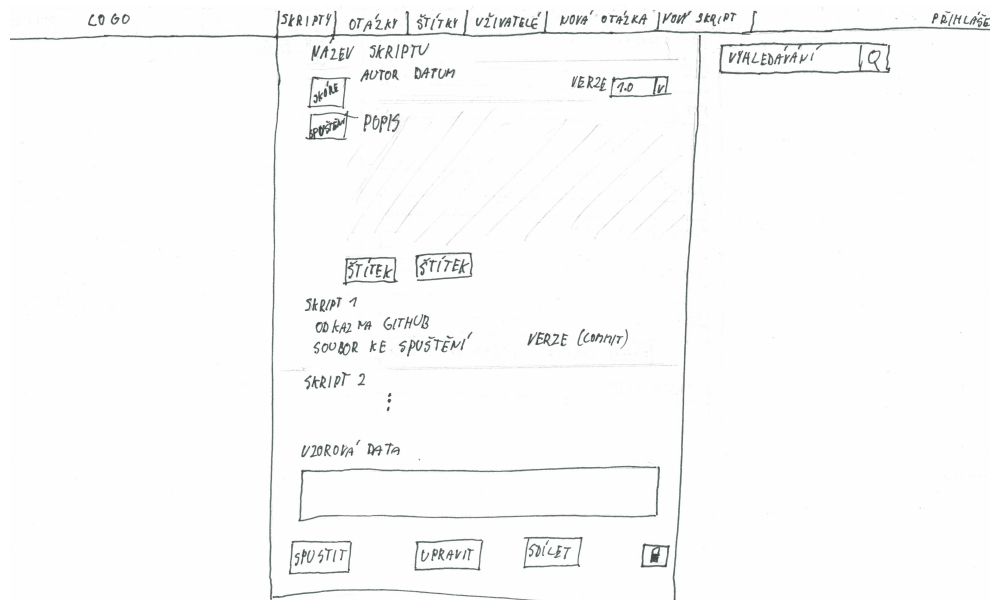
Stránka s detailem skriptu zobrazuje veškeré informace o skriptu. Kromě informací, které upravují sami uživatelé, zobrazuje také doplňkové informace:

- údaje o majiteli skriptu,
- údaje o uživateli, který naposledy editoval skript,
- skóre skriptu,
- počet spuštění skriptu,
- číslo aktuálně prohlížené verze.

Tato stránka dále umožňuje tlačítka přechod na stránku spuštění skriptu (kap. 3.4.2.3) a na stránku editace (kap. 3.4.2.1). Majitel skriptu zde může stiskem

### 3.4. Prototypování

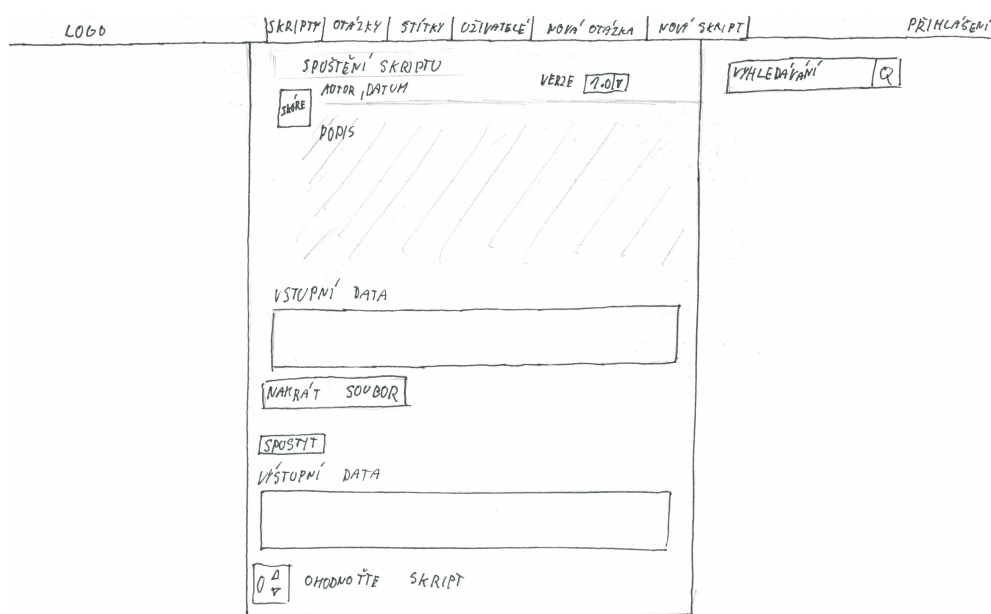
tlačítka se symbolem zámku uzamknout svůj skript proti úpravám. Tlačítkem sdílet získá uživatel odkaz na tuto stránku, který může dále sdílet.



Obrázek 3.9: Stránka detailu skriptu

#### 3.4.2.3 Stránka spuštění skriptu

Stránka spuštění skriptu umožňuje uživatelům skripty použít. Zobrazuje přehled o skriptu (veškeré informace mimo seznam zřetězených skriptů). Do pole vstupní data mohou uživatelé vložit svá textová data, nebo mohou využít možnosti nahrát soubor. Stiskem tlačítka spustit zahájí textovou transformaci. Během zpracování se zobrazuje načítací animace. Po skončení transformace zobrazí zpracovaná data (resp. stáhne soubor), nebo zobrazí chybu, pokud k nějaké došlo.

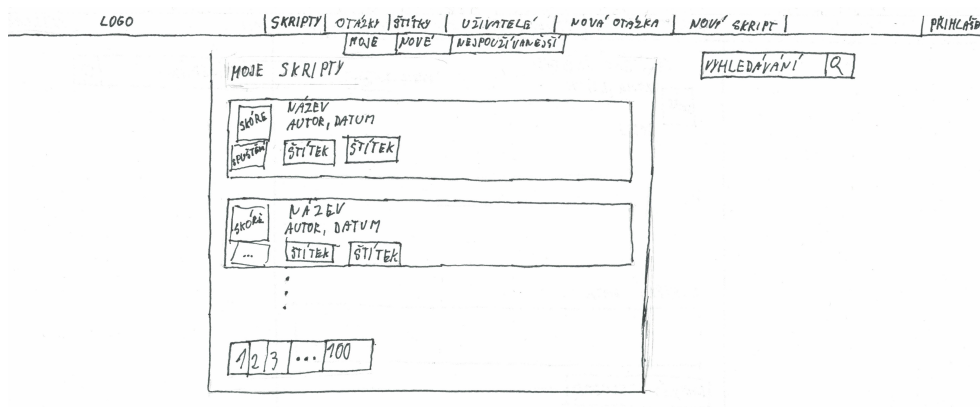


Obrázek 3.10: Stránka spuštění skriptu

#### 3.4.2.4 Stránka výpisu skriptů

Stránka výpisu skriptů se v aplikaci vyskytuje na více místech v různých obměnách. Konkrétně je využita při vypisování skriptů (nové, nejpoužívanější, nejlépe hodnocené, moje), při vyhledávání skriptů dotazem a při výpisu skriptů podle štítku. V nadpisu specifikuje podle jakého klíče zobrazuje skripty. Jednotlivé skripty jsou od sebe odděleny, jsou u nich zobrazeny nejdůležitější informace (název, štítky) a doplňkové informace (skóre, počet spuštění, majitel, editor). Výpis je stránkovaný.





Obrázek 3.11: Stránka výpisu skriptů

## 3.5 Testování prototypu

Pro otestování procesů a prototypů obrazovek byl vytvořen papírový prototyp aplikace. Tento papírový prototyp vychází z prototypů obrazovek, které dále rozvíjí. Testování proběhlo formou několika testovacích scénářů. U každého je uvedeno zadání a ideální průchod. Na konci kapitoly je uvedeno vyhodnocení. Obrázky ilustrující ideální průchod jsou v příloze C.

### 3.5.1 Vyhledání skriptu v diskuzi a jeho spuštění {T1}

Respondent má za úkol vyhledat diskuzi o formátování telefonních čísel. Po jejím úspěšném nalezení vybere v diskuzi odkaz na skript a přejde na spuštění skriptu. Zde zadá vstup a skript spustí. Po spuštění ohodnotí funkci skriptu.

#### 3.5.1.1 Ideální průchod

1. Respondent zadá vyhledávanou frázi do pole vyhledávání (obr. C.2).
2. Na obrazovce vyhledávání vybere odpovídající diskuzi (obr. C.3).
3. V detailu diskuze klikne na odkaz ke spuštění skriptu (obr. C.4).
4. Na stránce spuštění skript spustí (obr. C.5).
5. Na stránce spuštění skript ohodnotí (obr. C.6).

### 3.5.2 Přidání skriptu do aplikace a jeho spuštění {T2}

Respondent má za úkol přidat do aplikace zřetězený skript. Při jeho přidávání vyplní název, popis a štítky skriptu, nastavení odkazů do repozitářů a vzorová data. Po uložení přejde na stránku se spuštěním skriptu, kde skript spustí.

#### 3.5.2.1 Ideální průchod

1. Respondent klikne na tlačítko přidat skript (obr. C.7).
2. Na stránce přidání skriptu vyplní data o skriptu (obr. C.8).
3. Po jeho uložení je přesměrován na obrazovku detailu skriptu kde klikne na spuštění skriptu (obr. C.9).
4. Na stránce spuštění zadá vstupní data a skript spustí (obr. C.10).

#### 3.5.3 Editace skriptu {T3}

Respondent má za úkol zobrazit si výpis skriptů. Dále přejde na výpis svých skriptů, zde jeden z nich vybere a upraví přidáním dalšího odkazu na GitHub do zřetězení.

##### 3.5.3.1 Ideální průchod

1. Respondent přes navigaci přejde na obrazovku výpisu skriptů (obr. C.11).
2. Zde vybere možnost vypsát své skripty (obr. C.12).
3. Ve výpisu náhodně vybere skript a přejde na jeho detail (obr. C.13).
4. Na obrazovce editace skriptu přidá nový odkaz do zřetězení, vyplní popis změny a uloží (obr. C.14).

#### 3.5.4 Spuštění neaktuální verze skriptu {T4}

Respondent má za úkol vyhledat skript o formátování textu. Po jeho úspěšném nalezení spustí jeho verzi číslo 2.

##### 3.5.4.1 Ideální průchod

1. Do vyhledávacího pole vyplní hledanou frázi (obr. C.15).
2. Na obrazovce vyhledávání zvolí vyhledávání ve skriptech (obr. C.16).
3. Na stránce vyhledávání vybere skript a přejde na jeho detail (obr. C.17).
4. Pomocí rozevírací nabídky zvolí verzi skriptu číslo 2. Stránka zobrazí verzi 2. Stiskem tlačítka spustit přejde na stránku spuštění, kde tuto verzi skriptu spustí (obr. C.18).

#### 3.5.5 Vyhodnocení testování

Cílem testování, kterého se zúčastnilo pět respondentů, bylo ověřit, že navržené procesy a stránky jsou pro uživatele srozumitelné.

Testování ukázalo, že tlačítko sdílet na obrazovce detailu skriptu je zbytečné, uživatelům stačí kopie adresy stránky. Uživatelé také narazili na problém s vyhledáváním. Nelíbilo se jim, že vyhledávání probíhá nejprve v otázkách a až po překliknutí ve skriptech. Tento problém byl vyřešen upřednostněním vyhledávání ve skriptech podle kontextu. Pokud se uživatel nachází na stránce, která zobrazuje skripty, vyhledávání je provedeno nejprve ve skriptech.

---

# Implementace

V implementační části práce byl podle návrhu architektury (kap. 3.3) vytvořen plugin do systému Question2Answer starající se o správu skriptů a backendová aplikace, která tyto skripty spouští v odděleném prostředí Docker. Implementace se zabývala převážně požadavky stupně nezbytnosti MUST.

Tato kapitola se nejprve věnuje popisu vytvořeného řešení a jeho vlastností. Poté popisuje provedené testování aplikace. V závěru kapitoly se věnuje možnostem dalšího rozvoje aplikace. V příloze D je uveden postup k nasazení aplikace.

## 4.1 Popis implementace

Tato kapitola popisuje řešení vybraných požadavků na funkce a kvality. Hotová aplikace se skládá ze dvou hlavních komponent:

- Komponenta webového portálu starající se o správu diskuzí, skriptů a uživatelů. Pro funkcionalitu správy skriptů byl vytvořen plugin do existujícího řešení Question2Answer [12].
- Komponenta na spouštění skriptů spouštějící skripty v odděleném prostředí Docker. Tato JAVA komponenta se stará o frontu požadavků na spuštění skriptů. S prostředím Docker komunikuje pomocí knihovny Docker-client [14].  
V odděleném prostředí je umístěn skript, který komunikuje s komponentou na spouštění skriptů a ovládá spouštění uživatelských skriptů.

### 4.1.1 Otázky a odpovědi

Funkčnost otázek a odpovědí, v kontextu aplikace - zadání na skript a diskuze o něm, poskytuje zvolené hotové řešení Question2Answer.

Uživatelé, ať už přihlášení, či nikoliv, mohou založit diskuzi, ve které popíše svůj problém s formátováním textu. V popisu mohou uvést vzorová data, která ulehčí práci uživatelům, kteří se rozhodnou problém vyřešit. Pro lepší organizaci těchto diskuzí mohou k otázce vyplnit štítky. V diskuzi mohou dále diskutovat a sdílet mezi sebou své řešení.

Aplikace dále umožňuje v diskuzích vyhledávat podle fráze, zobrazovat otázky podle štítku, podle času, hodnocení atd.

### 4.1.2 Správa skriptů

Aplikace umožňuje uživatelům spravovat své skripty. Externí skript je v aplikaci chápán jako sada parametrů:

- odkaz na repositář s kódem skriptu umístěným v systému GitHub,
- cesta k spustitelnému souboru,
- konkrétní commit (verze repositáře).

Důvodem, proč se v aplikaci nepracuje se zdrojovými kódy skriptů ale rovnou s celým repositářem, je velké množství funkcí, které systém GitHub nabízí. Funkce typu práce s větvemi v repositáři, požadavky na sloučení od jiných uživatelů nemělo smysl duplikovat, vznikl by systém, který nabízí jen menší část z nich a který nutí uživatele nastavovat některé volby, např. přístupová oprávnění, dvakrát.

Záznam o skriptu v aplikaci obsahuje údaje, které pomáhají uživatelům s jeho použitím:

- název skriptu,
- popis skriptu,
- štítky,
- vzorová data.

Záznam o skriptu v aplikaci dále obsahuje jeden nebo více odkazů na repositáře. Pokud je jich více, jedná se o zřetězení skriptů. Tyto skripty jsou spuštěné postupně za sebou, s tím, že výstup prvního skriptu je vstupem druhého.

V aplikaci je důležitým prvkem komunita, proto jsou všechny skripty v aplikaci otevřené úpravám ostatních registrovaných uživatelů. Registrace pro upravovatele je vyžadována, aby byla omezena možnost záznamy o skriptech záměrně poškozovat. U každého skriptu je možné provést uzamčení proti úpravám komunitou. Toto uzamčení může provést pouze uživatel, který vytvořil záznam o skriptu v aplikaci (majitel skriptu).

### 4.1.3 Verzování skriptů

Jak již bylo zmíněno v sekci správa skriptů (kap. 4.1.2), skripty mohou být registrovanými uživateli aplikace upravovány. Tyto úpravy přináší riziko (nechtěné) úpravy, která např. smaže důležitou informaci. Tento problém řeší verzování. Jakákoliv úprava provedená na záznamu ve skriptu vytvoří novou verzi. Mezi verzemi je možné volně procházet a spouštět je.

### 4.1.4 Spouštění skriptů

Aby mohl být externí skript v aplikaci spuštěn, musí:

- jeho repositář být nastaven jako veřejný,
- být schopný číst ze standardního vstupu a zapisovat na standardní výstup.

#### 4.1.4.1 Průběh spouštění skriptu

Jedna z nejdůležitějších funkcí aplikace je možnost jednoduchého spuštění uložených skriptů bez jakýchkoliv znalostí. Aplikace umožňuje spouštět všechny uložené skripty i nepřihlášeným uživatelům:

1. Pro uživatele skriptu není důležité jestli se jedná o samostatný skript, nebo jestli je skript zřetězen. Uživatel zadá svá data ke zpracování, případně nahraje soubor. Pokud nahrál soubor, je jeho obsah načten a požadavek odeslán s jeho obsahem.
2. Tento požadavek přijme webový portál, který sestaví konfiguraci skriptů ke spuštění a společně s daty ke zpracování ji odešle na API komponenty na spouštění skriptů.
3. Tato komponenta má na starosti nejen samotné spouštění skriptů, ale také řídí frontu požadavků, aby nedošlo k vyčerpání systémových prostředků spuštěním nadměrného množství skriptů zároveň.
4. Komponenta postupně pro každý externí skript inicializuje kontejner Docker, který odstiňuje běžící skripty od okolního systému, předá mu konfiguraci skriptu ke spuštění a unikátní identifikátor spouštěného skriptu.
5. V kontejneru je přítomný Masterscript, bash skript, který má na starosti spuštění samotného skriptu. Masterscript:
  - 5.1. si od komponenty na spouštění vyžádá vstupní data skriptu podle unikátního id,
  - 5.2. stáhne repositář skriptu a převede ho do požadované verze (commit),
  - 5.3. spustí kód skriptu a předá mu vstupní data,
  - 5.4. zachytí zpracovaná data, která následně odešle zpět komponentě na spouštění.

## 4. IMPLEMENTACE

---

6. Kroky 4 a 5 se opakují přes všechny externí skripty, které jsou součástí spouštěného skriptu.
7. Po posledním zpracování jsou finální výstupní data odeslána zpět webovému portálu, který je předá uživateli.

Během každé z těchto operací může nastat chyba. Tyto chyby jsou v aplikaci zachytávány a odesílány přímo uživateli, který skript spustil.

### 4.1.4.2 Omezení běhu skriptů

Pro skripty a jejich vstup a výstup byla vytvořena omezení, která mají za cíl chránit systém proti zahlcení a útokům.

#### Omezení času běhu skriptu

Běh jednoho skriptu bylo nutné omezit. Při vykonávání skriptu může dojít k zacyklení, které by natrvalo vyřadilo jeden z kontejnerů z provozu. Proto bylo přistoupeno k omezení běhu na 20 vteřin. Toto omezení by mělo stačit na zpracování rozumně velkých vstupů a zároveň zbytečně nezabere jeden kontejner na dlouhou dobu při zacyklení skriptu.

#### Velikost a typ souboru

V rámci této práce jsou podporovány pouze textové soubory kódování UTF-8. Maximální velikost souboru byla omezena na jeden megabyte. Tato velikost je pro textové soubory zcela dostačující. Toto omezení souvisí i s omezením na čas.

#### Omezení prostředků systému

Uživatelský skript je během svého běhu dále omezen:

- nemá přístup na internet aby nemohl odesílat data vložená ostatními uživateli (např. rodná čísla),
- nemá v odděleném prostředí práva root,
- má omezené prostředky CPU, paměti RAM a kapacitu přiděleného souborového systému, aby nemohl omezovat chod aplikace.

### 4.1.5 Organizace a vyhledávání skriptů

V aplikaci je možné zobrazovat výpisy skriptů podle

- data změny,
- udělených bodů,
- počtu spuštění,
- štítku,
- přihlášeného uživatele.

Dále je v aplikaci dostupné textové vyhledávání v názvech a popisech skriptů, které bylo inspirováno [22].

### 4.1.6 Bodovací systém

Webový portál Question2Answer poskytuje robustní bodovací systém. Tento systém byl v rámci aplikace dále rozšířen o podporu skriptů, které lze hodnotit a jejich autor z jejich hodnocení dostává body. Registrovaní uživatelé získávají body za:

- vytvoření:
  - otázky,
  - odpovědi,
  - skriptu,
- za úpravu skriptu,
- za každý bod na jím vytvořené:
  - otázce,
  - odpovědi,
  - skriptu.

### 4.1.7 Statistiky

Aplikace u každého uživatele zaznamenává jeho aktivitu při udílení hodnocení. Tyto statistiky jsou následně dostupné na profilu uživatele, což motivuje uživatele aplikace k vyšší aktivitě.

U skriptů je kromě sledování jejich hodnocení také sledován počet spuštění. Bodování i počet spuštění je společný pro všechny verze skriptu.

### 4.1.8 Bezpečnost

Bezpečnost aplikace zajišťuje Question2Answer. Hesla uživatelů nejsou ukládána v podobě prostého textu. Přístup do databáze je odolný proti SQL útokům. [23]

Bezpečnost při spouštění skriptů je zajištěna jejich spouštěním v kontejneru Docker, který je odděluje od okolního systému a umožňuje omezení jejich systémových prostředků. Toto zabezpečení zvyšuje šance, že se nepodaří útočníkovi proniknout do okolního systému. Nicméně hrozba tzv. útoku nultého dne stále platí.

### 4.1.9 Vzhled

Při vývoji webové části aplikace bylo použito hotového tématu Snow Flat, který byl formou pluginu rozšířen zobrazování stránek se skripty. Zjev ani kód toho tématu nebyl nijak upravován.

## 4.2 Specifikace API

Komponenta na spouštění skriptů vystavuje dvě API. První je určeno pro webový portál, přes který jsou přijímány požadavky na spuštění skriptu. Druhé je interní, slouží pro komunikaci se skriptem spuštěným v odděleném prostředí. Tato kapitola popisuje vystavené metody zmíněných API.

### 4.2.1 API pro webový portál

Toto API slouží pro přijímání požadavků na spuštění skriptu. Komunikace probíhá ve formátu JSON.

#### 4.2.1.1 RunScript

Jedná se o metodu přijímající konfiguraci skriptu ke spuštění a vstupní data ve formátu JSON skrz POST požadavek. Metoda zkontroluje konfiguraci a spustí podle ní skripty, na které aplikuje vstupní data. Metoda vrací výstupní data, nebo, pokud došlo k chybě, chybovou hlášku.

Cesta k požadavku: „POST /runner/“. Parametry volání zachycuje tabulka 4.1. Ve volání jsou použity datové typy ScriptConfiguration (tab. 4.2) a RepoConfiguration (tab. 4.3). Odpověď je odeslána ve formátu popsaného v tabulce 4.4.

Tabulka 4.1: Parametry volání požadavku RunScript

Jméno atributu	Datový typ	Popis atributu
scriptConfiguration	ScriptConfiguration	Konfigurace skriptu ke spuštění

Tabulka 4.2: Datový typ ScriptConfiguration

Jméno atributu	Datový typ	Popis atributu
input	String	Vstupní data ke zpracování
repos	RepoConfiguration[]	Pole obsahující informace o jednotlivých zřetězených skriptech



Tabulka 4.3: Datový typ RepoConfiguration

Jméno atributu	Datový typ	Popis atributu
git	String	URL adresa GitHub repositáře
file	String	Cesta ke skriptu v repositáři
comm	String	Hash verze (commit) ke spuštění
order	Integer	Pořadové číslo skriptu

Tabulka 4.4: Parametry odpovědi požadavku RunScript

Jméno atributu	Datový typ	Popis atributu
status	String	Status provedení požadavku (OK, ERROR)
content	String	Zpracovaná data, odesláno pokud je stav OK
errorMessage	String	Chybová hláška, odesláno pokud je stav ERROR

## 4.2.2 API pro Docker

Toto API slouží pro interní komunikaci se spuštěným ovládacím skriptem (masterscript) v Dockeru. Poskytuje vstupní data a přijímá data zpracovaná. Komunikace probíhá ve formátu „text/plain“.

### 4.2.2.1 GetInput

Metoda volaná skrz GET požadavek vrací vstupní data ke zpracování odpovídající přijatému ID. Pokud neexistují data s vloženým ID je požadavek odmítnut.

Cesta k požadavku: „GET /docker/getinput/{id}“. Parametry volání zachycuje tabulka 4.5, parametry odpovědi tabulka 4.6.

Tabulka 4.5: Parametry volání požadavku GetInput

Jméno atributu	Datový typ	Popis atributu
id	Integer	ID vstupních dat

Tabulka 4.6: Parametry odpovědi požadavku GetInput

Jméno atributu	Datový typ	Popis atributu
	String	Vstupní data jako text/plain

### 4.2.2.2 PostOutput

Metoda volaná skrz POST požadavek přijímá zpracovaná data ze skriptu. Pokud neexistují data s vloženým ID je požadavek odmítnut.

Cesta k požadavku: „POST /docker/postoutput/{id}“. Parametry volání zachycuje tabulka 4.7, parametry odpovědi tabulka 4.8.

Tabulka 4.7: Parametry volání požadavku PostOutput

Jméno atributu	Datový typ	Popis atributu
id	Integer	ID zpracovaných dat
data	String	Zpracovaná data

Tabulka 4.8: Parametry odpovědi požadavku PostOutput

Jméno atributu	Datový typ	Popis atributu
	String	OK

## 4.3 Testování

Aplikace byla průběžně testována již během svého vývoje. Na závěr bylo provedeno testování s potenciálními uživateli. Někteří z nich stáli u zrodu této aplikace, účastnili se kvalitativního průzkumu. Dále byla testována odolnost komponenty na spouštění skriptů proti několika základním škodlivým útokům.

Uživatelé, se kterými bylo testování provedeno, byly s funkčností aplikace spokojeni. Ocenili zejména verzování skriptů a jejich snadné spouštění. Testování odolnosti prokázalo, že je komponenta spouštění připravena odolat základním útokům.

### 4.3.1 Uživatelské testování

Uživatelské testování spočívalo v představení projektu, po kterém následovalo zopakování testování prototypu (kap. 3.5), nyní ale na reálné aplikaci. Po skončení této části testování byl s respondenty veden nestrukturovaný rozhovor o aplikaci, ve kterém se zjišťoval jejich uživatelský dojem z aplikace.

Testování se zúčastnilo pět respondentů z řad studentů ČVUT FIT a jiných vysokých škol. Tři z nich se již dříve účastnili kvalitativního průzkumu. Testování ukázalo, že respondenti byli spokojeni s aktuálně nabízenými funkcemi aplikace a považují ji za využitelnou. Respondenti ocenili jednoduchost spouštění skriptů, jejich řetězení a systém hodnocení uživatelů.

**4.3.1.1 Dana Spořepová (studentka ČVUT FA), 11. května 2017**

Uvítala, že většina jejích požadavků z původního rozhovoru byla naplněna. Ocenila celkový stav aplikace, který jí přišel funkční. Neměla problém s ovládáním aplikace. Její jediná výtka směřovala na neexistující upozornění, že je spouštěna neaktuální verze skriptu.

**4.3.1.2 Jan Maryško (student VŠE, podnikatel), 11. května 2017**

Byl překvapen, že z jeho původních požadavků jich bylo naplněno jen pár. Nicméně se mu aplikace líbila. Ocenil přehledné statistiky aktivity uživatelů a verzování skriptů. Jako velmi praktické označil použití aplikace bez nutnosti přihlášení. Nebyl spokojen s uživatelským prostředím aplikace, zjev webu mu přijde zastaralý. Nicméně neměl problém s použitím aplikace. Doporučil přidat podporu sociálních sítí pro rozšíření aplikace do povědomí uživatelů.

**4.3.1.3 Daniel Němec (student ČVUT FIT), 11. května 2017**

Ocenil vysokou variabilitu technologií, které je schopna aplikace, s jen malými úpravami, spouštět. Dále pochválil statistiky uživatelů, možnosti vypisování skriptů podle kritérií a verzování skriptů. Funkce vyhledávání mu v aplikaci přišla nepřehledná.

**4.3.1.4 Jana Srogončíková (studentka ČVUT FIT), 12. května 2017**

Zaujalo ji řešení zabezpečení spouštění skriptů a jednoduchost použití. Dále jako klad uvedla možnost použít aplikaci i na mobilním telefonu. U hodnocení skriptů se jí nelíbilo, že body za hodnocení skriptu získává pouze majitel skriptu a uživatelé, kteří skript editovali nedostávají nic.

**4.3.1.5 Samuel Castillo (student ČVUT FIT), 12. května 2017**

Přivítal, že vzniklá aplikace využívá oddělené spouštění skriptů dle jeho upozornění. Za zajímavou funkci systému považuje řetězení skriptů. Oceňuje rychlost zpracování dat s ohledem na použití odděleného prostředí pro spouštění. Zvláště se mu líbily široké možnosti konfigurace webové části aplikace pro administrátora aplikace. Chyběla mu možnost aplikovat skript na více souborů zároveň.

### 4.3.2 Testování odolnosti

Jelikož skripty spouštěné v aplikaci nejsou kontrolovány, je nutné aby systém spouštění byl odolný. Toto testování ověřovalo odolnost proti těm nejzákladnějším útokům:

- nekonečný cyklus,
- smazání kořenového adresáře,
- fork bomba,
- odesílání dat na internet.

Testování prokázalo odolnost systému proti těmto hrozbám.

## 4.4 Možnosti rozšíření

Aplikaci lze dále rozšířit o další funkce popsané v požadavcích (kap. 2.3). Z navrhovaných funkcí je nejdůležitější požadavek na schvalovací proces. Je třeba zajistit, aby nově provedené změny na skriptu, nemohly funkci skriptu rozbít. Aplikace je na schvalování již připravena tím, že zavedla vlastnická práva na skripty. Majitel skriptu může dohlížet na provedené změny. Přidáním testovacích dat vzniknou testy, které případné nové verze skriptu musí splnit. Pokud by nově vytvořená verze skriptu potřebovala změnit sadu testů, byl by nutný souhlas majitele skriptu.

Funkce aplikace lze dále rozšířit přidáním dalších technologií:

- Textové transformace nemusí provádět pouze bash skript, kontejner Dockeru umožňuje jednoduché doinstalování nových technologií. Tudíž je možné v rámci stávající aplikace, pouhou změnou konfigurace kontejneru, spouštět například programy napsané v Javě, C, Pythonu atd. Pro ověření tohoto konceptu byl do kontejneru přidán program g++, což umožňuje kompilaci a spouštění programů napsaných v jazyce C++.
- Kód skriptů nemusí být uložen pouze na portálu GitHub. Jediným požadavkem na online repositáře je jejich podpora příkazu „git clone“. Do aplikace byla na závěr zkušebně přidána podpora portálu GitLab. Spouštění skriptů fungovalo bez problémů.

Tato práce se návrhu uživatelského rozhraní aplikace zabývala pouze okrajově. Navazující práce se může zabývat jeho přepracováním.

---

# Závěr

Cílem práce bylo analyzovat, navrhnout, implementovat a otestovat aplikaci umožňující pohodlné vyhledávání, sdílení a spouštění textových transformací. Toto zadání bylo splněno.

Výsledkem je komunitní portál na správu textových transformací s možností jejich online spuštění. Aplikace uživatelům umožňuje diskutovat nad problémy s textem, navzájem si sdílet, hodnotit a vylepšovat vytvořené transformace. Transformace lze v systému vyhledávat a pro lepší organizaci jsou označeny štítky.

Samotný kód skriptů, které jsou v textových transformacích zřetězeny, je vyvíjený a verzovaný s pomocí portálu GitHub. Tyto skripty jsou spouštěny v odděleném prostředí Docker, což zajišťuje jistou úroveň zabezpečení.

Textové transformace mohou zpracovávat data uživatelů ve formě textu nebo textových souborů. Uživatelé těchto transformací k jejich použití nepotřebují žádné technické znalosti.

Tato práce byla vypracována v souladu s dokumentem Pravidla a zásady projektů FIT. V průběhu jejího vypracování bylo provedeno velké množství průzkumů a testování, což mělo za cíl vytvoření aplikace s jednoduchým použitím, která bude přístupná široké veřejnosti.

Na tuto práci lze navázat více způsoby. Je možné pokračovat v rozšiřování funkcionality o požadavky získané v analytické fázi. Také je možné provést přepracování uživatelského rozhraní aplikace, které nebylo primárním cílem této práce.



---

# Literatura

- [1] Nutpan: Run programs directly from your Blog or Website. [online], 2016, [cit. 2016-12-1]. Dostupné z: <http://blog.nutpan.com/2016/07/run-programs-directly-from-your-blog-or.html>
- [2] Nutpan.com: Jdoodle.com. [online], 2016, [cit. 2016-12-11]. Dostupné z: <https://www.jdoodle.com/test-bash-shell-script-online>
- [3] Tutorialspoint. [online], 2016, [cit. 2016-12-1]. Dostupné z: [https://www.tutorialspoint.com/execute\\_bash\\_online.php](https://www.tutorialspoint.com/execute_bash_online.php)
- [4] LearnShell.org: Learn Shell Programming - Free Interactive Shell Programming Tutorial. [online], 2016, [cit. 2016-12-1]. Dostupné z: <http://www.learnshell.org/>
- [5] LakshmiPATHI: Webminal - Learn and Practise Linux online. [online], 2016, [cit. 2016-12-1]. Dostupné z: <http://www.webminal.org/>
- [6] Holen Vidar: ShellCheck – shell script analysis tool. [online], 2016, [cit. 2016-12-1]. Dostupné z: <http://www.shellcheck.net/>
- [7] Hurricane Electric: Man page for bash. [online], 2009, [cit. 2016-12-1]. Dostupné z: <http://man.he.net/?topic=bash&section=all>
- [8] Stack Exchange Inc: About - Stack Overflow. [online], 2016, [cit. 2016-12-1]. Dostupné z: <http://stackoverflow.com/company/about>
- [9] DZone Software: OSQA – Open Source Question and Answer Software for Basic Q&A Sites. [online], 2017, [cit. 2017-4-14]. Dostupné z: <http://www.dzonesoftware.com/products/open-source-question-answer-software>
- [10] LampCMS: About LampCMS. [online], 2017, [cit. 2017-4-14]. Dostupné z: <http://www.lampcms.com/about.htm>

- [11] Askbot Project: Askbot Project Documentation. [online], 2011, [cit. 2017-4-14]. Dostupné z: <https://askbot.org/doc/index.html>
- [12] Greenspan Gideon: Creating plugins for Question2Answer. [online], [cit. 2017-4-14]. Dostupné z: <http://docs.question2answer.org/plugins/>
- [13] Greenspan Gideon: Question2Answer - Free Open Source Q&A Software for PHP. [online], [cit. 2017-4-14]. Dostupné z: <http://docs.question2answer.org/>
- [14] Spotify: Docker Client. [online], 2017, [cit. 2017-5-1]. Dostupné z: <https://github.com/spotify/docker-client>
- [15] Bc. Zvolský Grant: Spouštění skriptů v prostředí Docker. [konzultace], 8. dubna 2017, Strahov.
- [16] Breaking Out of a Chroot Jail Using PERL. [online], [cit. 2017-4-15]. Dostupné z: <https://linux.die.net/man/2/chroot>
- [17] chroot(2) - Linux man page. [online], [cit. 2017-4-15]. Dostupné z: <https://linux.die.net/man/2/chroot>
- [18] Docker Inc.: Docker security. [online], 2017, [cit. 2017-4-15]. Dostupné z: <https://docs.docker.com/engine/security/security/>
- [19] Docker Inc.: What is Docker? [online], 2017, [cit. 2017-4-15]. Dostupné z: <https://www.docker.com/what-docker>
- [20] Oracle: Oracle VM VirtualBox. [online], 2017, [cit. 2017-4-15]. Dostupné z: <https://www.virtualbox.org/>
- [21] Greenberg Saul: User Centered Design and Prototyping. [online], [cit. 2017-5-15]. Dostupné z: [https://pages.cpsc.ucalgary.ca/~saul/hci\\_topics/pdf\\_files/prototyping.pdf](https://pages.cpsc.ucalgary.ca/~saul/hci_topics/pdf_files/prototyping.pdf)
- [22] How to build a MySQL-powered search engine. [online], [cit. 2017-4-30]. Dostupné z: <https://www.mullie.eu/mysql-as-a-search-engine/>
- [23] Greenspan Gideon: Making Question2Answer extra secure. [online], [cit. 2017-4-14]. Dostupné z: <http://docs.question2answer.org/install/security/>
- [24] Vlčková Kateřina: Škálování. [online], [cit. 2017-5-15]. Dostupné z: [https://is.muni.cz/el/1441/podzim2015/SZ7BK\\_MET1/um/59559878/59801521/skalovani.ppt](https://is.muni.cz/el/1441/podzim2015/SZ7BK_MET1/um/59559878/59801521/skalovani.ppt)
- [25] Greenspan Gideon: Install Question2Answer. [online], [cit. 2017-5-14]. Dostupné z: <http://docs.question2answer.org/install/>



---

# Dotazník ke kvantitativnímu průzkumu

Pro účely kvantitativního průzkumu (kap. 2.2.3) byl vytvořen online dotazník.

## A.1 Testování dotazníku

Před zveřejněním dotazníku bylo v kapitole 2.2.3.5 jeho znění otestováno na několika respondentech formou rozhovorů. Připomínky z rozhovorů byly průběžně zapracovávány. Tato příloha slouží pro ilustraci tohoto testování.

### A.1.1 Ing. Filip Navara, programátor a podnikatel, 27. února 2017

- Vnímá dotazník jako „systém na sdílení a aplikaci skriptů“. Navrhuje tento nadpis zvážit jako výstižnější.
- K dotazu, zda „byste využili“ dodává, že něco z toho dokonce využívá. To je zajímavá otázka, kde a jak pro doplnění nebo aspoň rozšíření otázky na „využíváte nebo využili byste“.
- Klávesové zkratky pro kopírování a vložení se píše s plusem (CTRL+C nebo dokonce Ctrl+C). To by chtělo ověřit a opravit.
- Otázka s dočasným úložištěm je podle něho zaváděcí, ačkoli on si pod tím něco představit dokáže. Specifikovat?
- Bodovací systém by rozšířil o známý a funkční koncept „budování karm“. Zní to dobře a asociuje to gamifikaci.
- Slovo „zadání“ vnímá v kontextu vyhledávání jako zavádějící. Výstižnější mu přijde termín popis, štítky, případně další metadata o skriptu.
- Postrádá dotaz, jaké existující systémy uživatelé již využívají a jaké tam oceňují nebo naopak postrádají funkce. Sám zmiňuje např. Gist.
- Zřetězování skriptů vnímá jen jako malou podmnožinu možných kombinací existujících skriptů. Chápe myšlenku, že v Gitu může vzniknout

skript využívající jiné skripty... Ocenil by specifikování, že zřetězení nemusí být jen lineární (např. více vstupů atp). Líbilo se mu přirovnání k Yahoo Pipes.

- Schvalovací proces považuje pouze za podmnožinu problému, který by nás mohl zajímat. Vedle „schvalování změn“ navrhuje také rozlišit vylepšení (upgrade) skriptu a alternativu (fork).
- Předpokládá, že by mohla/měla být zajímavá také otázka licencování. Navrhuje se doptat na zjednodušený model (oss only), ale připouští privátní (placené) použití.
- Zmiňované problémy na konci dotazníku jsou podle něho také spíše pro IT specialisty (laik jim nerozumí) a uvítal by upozornění, že je otázka volitelná. Případně by přidal skupinu laických problémů.
- Do pole Jiné u problémů uvedl zajímavý příklad generování struktury (XML) ze seznamu souborů (výpis adresáře).
- Zmínil, že by mohl být zajímavý názor uživatelů na publicitu skriptů (public/protected/private) vs. jejich modifikace (edit/view/hidden).

### A.1.2 Bc. Jan Vlnas (zaměstnanec ČVUT FIT, student), 2. března 2017

- Přeformulovat slova „poptat“, „ukládání“ („zadání“)
- Není si jistý realizovatelností podpory MS Word, jak by probíhalo vnitřní zpracování. Spíše by uvítal „integraci“ s Google Diskem (Drivem), konkrétně např. jako možnost pluginu pro zavolání skriptu jako makra.
- U budování karmy by uvítal přirovnání ke StackOverflow.
- Věta „Diskuze u každého zveřejněného skriptu, či zadání na skript“ mu přijde špatně pochopitelná. Přeformulovat? (přidat „u“)
- K otázce sdílení skriptu jako odkaz na internet ho napadá termín „adresovatelný skript“, navrhuje připsat.
- Štítkování skriptu mu připomnělo funkci automatického oštitkování. Nová funkce?
- Možnost spouštění offline přímo z terminálu. Kromě použití offline je pro něho důležitá i možnost napojení na jiné utility, případně spouštění jako součást automatického workflow.
- Schvalovací proces není jednoznačný. Smazat?
- Přístupová práva - tyto otázky navrhuje přeformulovat.
- Testování ukázalo, že otázka na používání jiných systémů je příliš obsáhlá a obecná. Řešením je otázku rozdělit a přidat výčty (systémů a jejich vlastnosti).
- Vzorová data jako ukázka u skriptu (pro uživatele skriptu).
- Zmínil používání online formátovačů a převodníků, u kterých mu vadí neveřejná implementace, nemožnost úpravy a dávkování, zátěž (velké soubory). Dále zmínil programy z repozitáře Debianu, u kterých mu vadí netransparentní dokumentace – je nutné nejprve nainstalovat balíček aby

zjistil, co program umí. Dále zmínil správce balíčků pro různé jazyky – npm, Ruby Gems a PIP/PyPI. Zde je většinou dokumentace lépe dostupná, ale balíčky občas vyžadují instalaci a kompilaci netriviálních závislostí.

- U zmiňovaných problémů navrhuje sloučit práci s mezerami a konci řádků do jednotného problému práce s bílými znaky.
- Zde by uvedl problém filtrování a modifikace existujících struktur.
- Průzkum ukázal, že umístění otázky na chybějící funkčnost až na konci dotazníku není vhodné. Řešením je její umístění ke každé otázce.
- Postrádá možnost zanechat kontakt pro další zapojení na projektu.

### A.1.3 Samuel Castillo (student ČVUT FIT), 2. března 2017

- Zadání vstupních dat vnímá jako přebytečnou otázku, bere to jako implicitní funkci.
- Otázky na zpětnou vazbu a hodnocení uživatelů vnímá jako duplicitní.
- Specifikace uživatelů skriptu (každý / pouze s odkazem) navrhuje změnit na Nastavení přístupnosti skriptu (veřejný, kdokoli s odkazem, konkrétní uživatelé).
- Navrhuje přesunout Přílohu testovacích a vzorových dat do otázky Vytváření skriptu a její rozdělení na dvě otázky s jasným vysvětlením o jaká data jde.
- V praxi by uvítal by podporu RTF souborů.

### A.1.4 Ing. Lukáš Bařinka (vyučující ČVUT FIT), 3. března 2017

- V popisu projektu navrhuje zaměnit slovo „kde“ za „kterým“.
- Všiml si nekonstence v názvosloví (aplikace vs. spouštění skriptu).
- Poslední větu popisu navrhuje rozdělit na dvě, nyní je příliš dlouhá.
- Přeformuloval by otázky aby neobsahovali slovo „možnost“.
- Slovo „Diskuze“ by přeformuloval na „Socializace“.
- „Stážení do PC“ by zaměnil za Zobrazení / stažení skriptu.
- Upozornil, že hodnocení uživatele přímo závisí na hodnocení jeho skriptů.
- Napadla ho možnost poslání dat, na kterých skript nefunguje (report chyb).
- Testovací data (mezní hodnoty) mu nepříjdou dostatečně vysvětlena.
- Zaměnil by slovo „formátování“ za „manipulace“.
- V editoru skriptů by uvítal krokování.
- Chybí mu příklady již existujících řešení.

### A.1.5 Michil Jabali (student ČVUT FEL), 3. března 2017

Otázka na Bodovací systém mu nepříjde dobře pochopitelná, navrhuje zaměnit slovo „skriptu“ na „skriptů“ pro lepší pochopení. Jiné nedostatky nenalezl.

## A.2 Plné znění dotazníku

V této kapitole je uvedené plné znění dotazníku ve znění, ve kterém byl uveřejněn.

### A.2.1 Předmluva

Výsledky toho dotazníku ovlivní návrh nového systému na organizaci, sdílení a spouštění skriptů zpracovávajících text. Časová náročnost vyplnění dotazníku je 4 - 8 minut.

Uživatel potřebující automaticky zpracovat text (např. doplnění pevných mezer) může najít hotové řešení, kterým si text nechá zpracovat. Když řešení nenalezne, může si skript sám vytvořit a následně ho nabídnout ostatním. Pokud neumí vytvářet skripty, požádá o jeho vytvoření.

Vzorové použití (můžete přeskočit): Vlastním dlouhý seznam telefonních kontaktů. Čísla jsou však v různých formátech (00420 123 456 789, +420 123 45 67 89, 123456789). Potřebuji jejich formát sjednotit, abych je mohl nahrát do svého nového telefonu. Práci si neumím zautomatizovat (napsat skript), tak se podívám, jestli řešení už neexistuje. Pokud jej nenalezu, založím nový požadavek, kde svůj problém popíšu. Po čase mi přijde upozornění, že někdo můj požadavek vyřešil. Na uvedeném odkazu vložím svůj seznam čísel (mohu ho i nahrát v souboru), a pomocí jediného kliknutí obdržím čísla v jednotném formátu. Jako poděkování odměním autora skriptu pozitivním hodnocením.

### A.2.2 Otázky

1. Umíte psát a používat skripty (programovat)? (jedna možnost)
  - Ne
  - Ano, učím se
  - Ano, profesionálně
  - Jiné:
2. Využili byste (využíváte) následující funkce týkající se online spouštění skriptů? (\*)
  - Nahrávání vstupních souborů
  - Zobrazení skriptu
  - Použití online vstupů (odkaz na internet, systém si data stáhne sám)
  - Dočasné uložení výstupu ze skriptu pro pozdější stažení
  - Zobrazení vzorových dat (jednoduchý příklad funkce skriptu)
3. Napadá Vás nějaká další funkce online spouštění skriptů? Která? Pokud ne, přeskočte.

4. Využili byste (využíváte) následující funkce týkající se socializace? (\*)
  - Diskuze u každého zveřejněného skriptu, či u zadání na skript
  - Bodovací systém uživatelů podle hodnocení jejich skriptů (budování karmy jako u Stack Overflow)
  - Hlášení chyb (špatná funkce skriptu)
  - Sdílení skriptu jako odkaz na internet (adresovatelný skript)
  - Seznam oblíbených skriptů
  - Vyhledávač podle popisu (štítku) skriptu
  - Profil uživatele se všemi zveřejněnými skripty
5. Napadá Vás nějaká další funkce socializace? Která? Pokud ne, přeskočte.
6. Využili byste (využíváte) následující funkce týkající se vytváření a ukládání skriptů? Tato otázka vyžaduje IT znalost a můžete ji přeskočit. (\*)
  - Verzování skriptů
  - Parametrizace skriptů
  - Kombinace skriptů (řetězení, větvení)
  - Automatizace (ukládání, spouštění) přes API
  - Zobrazení / stažení skriptu
  - Práva skript použít / zobrazit / upravit
  - Nastavení přístupnosti skriptu (veřejný, kdokoli s odkazem, konkrétní uživatelé)
  - Příložením testovacích dat (Continuous Integration)
7. Napadá Vás nějaká další funkce vytváření a sdílení skriptů? Která? Pokud ne, přeskočte.
8. Využili byste (využíváte) následující funkce online editoru skriptů? Tato otázka vyžaduje IT znalost a můžete ji přeskočit. (\*)
  - Barevné rozlišení syntaxe
  - Kontrola syntaxe
  - Odkazy do manuálových stránek
  - Krokování skriptu (ladění)
9. Napadá Vás nějaká další funkce online editoru? Která? Pokud ne, přeskočte.
10. S jakými problémy u manipulace s textem se setkáváte? Příklady problémů, některé mohou být technického charakteru. Pokud máte, uveďte i své problémy. (více možností)
  - Práce s bílými znaky (např. převod mezer na tabulátor, pevná mezera)
  - Výskyty (např. hledání řádků obsahujících výraz)
  - Převod dat do jednotného formátu (např. jednotný formát telefonních čísel)
  - Konverze mezi formáty (XML, JSON, MS Word, RTF)

- Jiné:
- 11. Používáte nějaké již existující řešení? Jaké? Pokud ne, přeskočte. (více možností)
  - Stack Overflow
  - Gist
  - Jdoodle
  - ShellCheck
  - Jiné:
- 12. Pokud nějaké používáte, které funkce oceňujete a které postrádáte?
- 13. V případě zájmu se podílet na dalším testování uveďte svojí e-mailovou adresu.

Otázky označené hvězdičkou (\*) mají formu tabulky s možnostmi: Určitě ano, spíše ano, nevím, spíše ne, určitě ne.

### A.3 Vyhodnocení dotazníku

Dotazník byl vyplněn 71 respondenty, z nichž 53 uvedlo, že umí psát skripty (profesionálně, učí se) a 17 lidí naopak uvedlo, že skripty psát neumí.

Pro každou funkcionalitu bylo vypočteno skóre zohledňující zájem respondentů. Pro výpočet byla stupnice využitelnosti převedena na čísla (určitě ano (1,5); spíše ano (1); nevím (0); spíše ne (-1); určitě ne (-1,5)). Tato čísla jsou následně sečtena, čímž vyjde [skóre]. Jednotlivé funkce jsou rozděleny podle skóre do dvou skupin: hodně využívané a méně využívané. Metoda výpočtu skóre byla inspirována [24].

Odpovědi v otevřených otázkách byly převzaty přímo z odpovědí respondentů a byly pouze mírně upraveny (odstranění překlepů, či nesmyslů).

- Využili byste (využíváte) následující funkce týkající se online spouštění skriptů?
  - Hodně využívané
    - \* Zobrazení vzorových dat [63,5]
    - \* Nahrávání vstupních souborů [58,5]
    - \* Zobrazení skriptu [51,5]
  - Méně využívané
    - \* Dočasné uložení výstupu ze skriptu pro pozdější stažení [31]
    - \* Použití online vstupů [28,5]
- Napadá Vás nějaká další funkce online spouštění skriptů? Která?
  - historie nahraných skriptů, volba zveřejnění skriptu s popisem
  - automaticky deploy na github, gitlab a pod

- Vytěžování určitých dat z textu. Například hodnot ze záznamu v logovacích souborech.
- gitlab CI
- online úprava skriptu (pro moje potřeby ho online upravím)
- hw náročnost
- Debuging
- Využili byste (využíváte) následující funkce týkající se socializace?
  - Hodně využívané
    - \* Hlášení chyb (špatná funkce skriptu) [81,5]
    - \* Vyhledávač podle popisu (štítku) skriptu [64,5]
    - \* Bodovací systém uživatelů podle hodnocení jejich skriptů [63,5]
    - \* Diskuze u každého zveřejněného skriptu, či u zadání na skript [57,5]
  - Méně využívané
    - \* Sdílení skriptu jako odkaz na internet (adresovatelný skript) [45]
    - \* Seznam oblíbených skriptů [29,5]
    - \* Profil uživatele se všemi zveřejněnými skripty [9,5]
- Napadá Vás nějaká další funkce socializace? Která?
  - Sledování aktivity uživatelů
  - Sledování vývoje/změn skriptu (t.j. např. notifikace, že nějaký skript byl napsán - např. na něčí žádost - či změněn)
  - Doporučování skriptů
  - Tématické skupiny skriptů podle funkcionality
  - Hodnocení někdy nestačí, protože skript může fungovat správně, mít vysoké hodnocení, ale pak se objeví nějaká malá chyba, kterou 1 člověk nalezne, ale hodnocení se už nezbavíme. Tak ještě k tomu bod - něco jako „považované za správné řešení“.
- Využili byste (využíváte) následující funkce týkající se vytváření a ukládání skriptů?
  - Hodně využívané
    - \* Zobrazení / stažení skriptu [60]
    - \* Přiložení testovacích dat [34,5]
    - \* Parametrizace skriptů [33,5]
    - \* Verzování skriptů [31,5]
    - \* Nastavení přístupnosti skriptu (veřejný, kdokoli s odkazem, konkrétní uživatelé) [31,5]
  - Méně využívané
    - \* Automatizace (ukládání, spouštění) přes API [30]
    - \* Práva skript použít / zobrazit / upravit [28,5]
    - \* Kombinace skriptů (řetězení, větvení)[26,5]

- Napadá Vás nějaká další funkce vytváření a sdílení skriptů? Která?
  - (Navazuje na navrhnutou další možnost socializace ohledně sledování) Možnost notifikovat sledující uživatele o „finálním stavu“ skriptu (resp. jiném milníku)
  - Upozornění na nové verze
- Využili byste (využíváte) následující funkce online editoru skriptů?
  - Hodně využívané
    - \* Barevné rozlišení syntaxe [74,5]
    - \* Kontrola syntaxe [62,5]
  - Méně využívané
    - \* Krokování skriptu (ladění) [28]
    - \* Odkazy do manuálových stránek [26,5]
- Napadá Vás nějaká další funkce online editoru? Která?
  - Možnost prezentovat skupině s real-time změnami (podobně jako Google Docs)
  - Skupinová práce na tvorbě skriptu u složitějších
  - Nápořádání
  - Automatické formátování
  - Porovnání dvou verzí skriptu
  - Napovídání kódu
  - Autodoplňování
  - Přehledné členění ve stylu one note
- S jakými problémy u manipulace s textem se setkáváte?
  - Práce s bílými znaky (61% respondentů)
  - Výskyty (52% respondentů)
  - Převod dat do jednotného formátu (65% respondentů)
  - Konverze mezi formáty (51% respondentů)
- Používáte nějaké již existující řešení? Jaké?
  - Stack Overflow (54% respondentů)
  - Gist - (7% respondentů)
  - Jdoodle - (3% respondentů)
  - ShellCheck - (7% respondentů)
- Pokud nějaké používáte, které funkce oceňujete a které postrádáte?
  - Pochvala
    - \* ShellCheck - oceňuji návrhy / doporučení na ošetření příkazů (např. při chybovém výstupu)
    - \* Stack Overflow - množství informací
    - \* Stack Overflow - bodovací systém
    - \* Stack Overflow - hodnocení odpovědi (kvalita odpovědi)



- \* Stack Overflow - diskuze s připomínkami týkajícími se dané problematiky
- \* Stack Overflow - diskuze, Jdoodle - interaktivní
- Stížnost
  - \* ShellCheck - Postrádám možnost spustit si skript rychle v sandboxu (izolovaném prostředí)



## **Pravidla a zásady projektů FIT**

V této příloze je uvedeno plné znění dokumentu „Pravidla a zásady projektů FIT“ platné ke dni 10. 3. 2017. Tato práce byla vypracována podle tohoto dokumentu. V kapitole 2.4 jsou uvedeny požadavky na kvality odvozené z tohoto dokumentu.

# Pravidla a zásady projektů FIT

Tento dokument popisuje preferovaný způsob řešení projektů FIT. Pokud některá část tohoto dokumentu není vůči konkrétnímu projektu efektivní nebo na ní není dostatek prostředků, je jí snížena úroveň nezbytnosti<sup>1</sup> nebo je od ní zcela odstoupeno. O takovém rozhodnutí musí existovat záznam.

Tento dokument je nedílnou součástí projektové dokumentace jako příloha ke kapitole Požadavky na kvalitu. Veškeré oblasti tohoto dokumentu, které uvedená kapitola nezmíní, se předpokládají a požadují. Výchozí úroveň nezbytnosti požadavků (není-li uvedeno jinak) je [MUST].

## *Obsah dokumentu*

### [Dokumenty](#)

#### [Harmonogram](#)

#### [Ostatní dokumenty projektu](#)

### [Architektura a integrace do infrastruktury](#)

### [Webová přístupnost](#)

### [Kód aplikace](#)

#### [Verzování](#)

#### [Kontrola kvality \(QA\)](#)

### [Provoz, údržba a rozvoj aplikace a podpora uživatelů](#)

### [Bezpečnost a ochrana osobních údajů](#)

### [Další koncepty webových aplikací \(W2.0\)](#)

## Dokumenty

Ke každému projektu vzniká sada samostatných dokumentů v čele s harmonogramem. Dokumenty jsou (až na výjimky) psané česky. Výchozím dokumentem je Harmonogram, který podléhá schvalování. Není-li uvedeno jinak, o projektu rozhoduje vedení fakulty (grémium děkana).

---

<sup>1</sup> [Key words for use in RFCs to Indicate Requirement Levels](#)

# Harmonogram

Stěžejním rozcestníkem projektu je iterativní harmonogram. V každém běhu se předpokládá realizace jen takových funkcí a vlastností aplikace, které náleží do daného běhu. Každý běh (iterace) obsahuje všechny uvedené fáze harmonogramu. Pro každou fázi je stanoven termín předpokládaného dokončení. Každá fáze podléhá schvalování vedením písemnou formou. Obsah harmonogramu se s postupem času upřesňuje.

## 1. Analytická fáze

- Průzkum existujících řešení (dále jen SOTA<sup>2</sup>).
- Vytvoření hypotéz jako kvalitativním průzkum, typicky formou rozhovorů.
  - i. Uživatelské skupiny
  - ii. Potřeby uživatelů
- Ověření hypotéz jako kvantitativní průzkum, typicky dotazníkem v souladu s explicitně zmíněnou metodikou<sup>3</sup>.
- Sestavení požadavků na kvality (vycházející z tohoto dokumentu).
- Sestavení požadavků na funkce (prioritní seznam rozdělený dle úrovně nezbytnosti).
- Posouzení SOTA vůči požadavkům na funkce.

## 2. Návrhová fáze

- Procesy (back-end, front-end)
- Prototypování (paralelní).
- Diagramy (procesní, aktivit).
- Scénáře průchodu, user-stories, případové studie
  - i. pro všechny vznikající funkce
  - ii. testování (např. formou storyboarding, inspekce).
- Hi-fi prototypy (grafický návrh, ...).

## 3. Implementační fáze

- Koncepty řešení dílčích požadavků (funkční / nefunkční).
- Technická specifikace (API, parametry, manuál) jako příloha k projektové dokumentaci nebo přímo u projektu v repozitáři.
- Testování (inspekce, heuristika).

## 4. Vyhodnocení

- Testování, logování.
- Vyhodnocení (feedback, statistiky, logy).

# Ostatní dokumenty projektu

V rámci vývoje vznikají další typy dokumentů s níže uvedenými náležitostmi pro různé skupiny čtenářů. Obsah každého dokumentu je cílený na příslušnou skupinu uživatelů a zohledňuje jejich schopnosti a možnosti.

## ● Projektová dokumentace

- je množina samostatných dokumentů vznikajících pro jednotlivé iterace agilního vývoje aplikace<sup>4</sup>,
- má strukturu podle fází harmonogramu,
- odkazuje na související legislativní úpravu problémové domény aplikace,
- je určená pro zadavatele, návrháře a vývojáře projektu a případně pro uživatele.

---

<sup>2</sup> [State of the Art](#)

<sup>3</sup> [6 kroků, jak vytvořit dotazník](#)

<sup>4</sup> [Agile software development](#)

- **Instalační a provozní příručka**
  - obsahuje kompletní postup pro sestavení (build) a nasazení (deployment) aplikace a nových verzí,
  - popisuje dostupná prostředí (staging/produkční verze) v návaznosti na [kap. Údržba a rozvoj](#),
  - popisuje provozuschopnost v případě nedostupnosti souvisejících služeb,
  - popisuje proces obnovení provozu v případě výpadku.
- **Uživatelská dokumentace**
  - je průběžně udržovaný samostatný dokument,
  - obsahuje informaci, k čemu a komu aplikace souží,
  - je určená koncovým uživatelům frontendové aplikace<sup>5</sup>, resp. aplikačního rozhraní (RESTful API)<sup>6</sup>,
  - je dostupná z webu FIT (stačí odkazem)<sup>7</sup>,
  - zahrnuje CHANGELOG (viz [kap. Verzování](#)).
- **Dokumentace vnitřního API**
  - je sada dokumentů generovaná z kódu průběžně udržovaná společně s kódem aplikace,
  - je psaná anglicky v příslušné standardizované syntaxi<sup>8</sup>, přičemž dokumentace veřejných entit zahrnuje minimálně:
    - souhrnný popis dokumentované entity (funkce, třídy, metody, proměnné, ...),
    - souhrnný popis parametrů (funkce/metody) nebo typových proměnných (generické typy),
    - popis vyřazených výjimek (které výjimky a kdy vznikají),
    - popis návratové hodnoty (a její význam).

## Architektura a integrace do infrastruktury

Projekt je webovou aplikací, která efektivně využívá existující technologie a služby FIT. Aplikace je členěná na nezávislé části, které je možné vyměnit a je provozuschopná i v případě výpadků souvisejících služeb.

- Projekt je webovou aplikací s
  - uživatelským rozhraním (UI) pro webový prohlížeč, nebo
  - RESTful API s upřesněním standardu vč. formátu<sup>9</sup>.
- Architektura aplikace
  - striktně odděluje frontend a backend,
  - správa uživatelů (user-management) je zajištěna fakultním IDM,
  - využívá maximum dostupných služeb (např. notifikace),
  - [SHOULD] podporuje použití pro více fakult na jediné instanci.
- Aplikace
  - je součástí katalogu služeb FIT<sup>10</sup> od počátku práce na projektu (stav „připravuje se“),
  - využívá mezipaměť pro urychlení obsluhy požadavků,

<sup>5</sup> [10 Examples of Great End User Documentation](#)

<sup>6</sup> Doporučené nástroje pro dokumentaci RESTful API: [RAML](#), [Swagger](#) / [OpenAPI](#)

<sup>7</sup> [Návod ke psaní dokumentace ICT FIT](#)

<sup>8</sup> Např. JavaDoc nebo DoxyGen

<sup>9</sup> Doporučujeme vycházet ze standardu [JSON API](#)

<sup>10</sup> [Katalog služeb ICT FIT](#)

- je provozuschopná i v případě nedostupnosti (zpomalení) souvisejících služeb<sup>11</sup>.
- [MAY] Aplikace je implementovaná na platformě/jazyku:
  - Ruby,
  - JavaScript, resp. izomorfní JavaScript<sup>12</sup>,
  - Groovy/Java na Spring Frameworku,
  - Python,
  - PHP na frameworku Symfony.

## Webová přístupnost

Aplikace je přístupná pro uživatele bez ohledu na jejich omezení a zařízení, kterým k aplikaci přistupují.

- Aplikace respektuje požadavky WCAG 2.0 AA<sup>13</sup>, zejména
  - sémantické značkování výstupu HTML,
  - jednoznačné perzistentní URL jednotlivých stránek<sup>14</sup>,
  - podpora tisku,
  - *progressive enhancement*<sup>15</sup>.
- Výstup aplikace (HTML) je v souladu s principem *mobile-first*<sup>16</sup>, *media-first*<sup>17</sup>, zejména
  - přizpůsobivé uživatelské rozhraní<sup>18</sup>,
  - použitelnost ovládacích prvků pro manipulaci prsty,
  - nenáročnost s ohledem na výkon CPU a spotřebu baterie,
  - minimalizace přenesených dat.
- [MAY] Aplikace je odolná vůči výpadkům připojení a funkčnost bez připojení k Internetu.<sup>19</sup>
- [MAY] Aplikace podporuje *Web App Manifest*<sup>20</sup> a integraci do operačního systému<sup>21</sup> zahrnující
  - podporu push notifikací<sup>22</sup>,
  - synchronizaci na pozadí přes *Service Workers*<sup>23</sup>.

---

<sup>11</sup> Např. bez datového, resp. autentifikačního, zdroje, informace z mezipaměti, resp. zobrazí jen veřejné informace (s příslušným upozorněním).

<sup>12</sup> Viz [Isomorphic JavaScript](#).

<sup>13</sup> [Web Content Accessibility Guidelines \(WCAG\) 2.0](#)

<sup>14</sup> Viz [Cool URIs](#) a [Why JavaScript web applications should embrace traditional URLs](#).

<sup>15</sup> Poskytnout klientovi úplnou funkcionalitu i v případě, že nepodporuje dynamické technologie; viz [článek na Gov.UK](#).

<sup>16</sup> Viz [kniha Mobile First \(Luke Wroblewski\)](#).

<sup>17</sup> Společná definice zobrazení od sémantického obsahu pro čtečky a textové interprety, přes tisk a malé obrazovky až po velké obrazovky.

<sup>18</sup> Viz [Responsive Web Design](#).

<sup>19</sup> Tzv. [Offline-First](#).

<sup>20</sup> Viz [Web App Manifest](#).

<sup>21</sup> Viz články [Progressive Web Apps: Escaping Tabs Without Losing Our Soul](#) a [Getting started with Progressive Web Apps](#).

<sup>22</sup> Viz [Push Notifications on the Open Web](#).

<sup>23</sup> Viz [Introduction to Service Worker](#).

# Kód aplikace

Veškerý kód je psaný kompletně v angličtině s prioritou udržitelnosti a čitelnosti. Vývoj kódu přehledně odděluje provozní větve od vývojových. Před nasazením prochází každý nově vzniklý kód kontrolou kvality na několika úrovních.

- Aplikace respektuje *best practices* pro psaní udržitelného a čitelného kódu<sup>24</sup>; zejména
  - logické členění kódu do modulů podle funkcionality,
  - specifikace konvencí používaných technologií (např. CSS<sup>25</sup>, JavaScript<sup>26</sup>, Java<sup>27</sup>),
  - dodržování stylu autora při editaci cizího kódu,
  - minimalizace importů<sup>28</sup> (import, include, using, atd.),
  - používání existujících knihoven<sup>29</sup>, kdykoli je to efektivní a smysluplné,
  - používání návrhových vzorů<sup>30</sup>,
  - komentování potenciálně nejasných částí.
- Veškerý kód je psaný
  - v UTF-8 s unixovým koncem řádek (řídící znak LF / 0x0A),
  - anglicky (názvy funkcí a proměnných, komentáře, systémová a jiná hlášení).
- Veškeré výstupy (texty pro uživatele) podporují lokalizaci a internacionalizaci.
- Pro aplikaci existuje kompletní česká lokalizace.

## Verzování

Vývoj kódu je organizovaný s přehledným oddělením provozní a vývojové větve. Umožňuje operativní opravy kritických chyb (hotfix) a nezávislý vývoj nových funkcí. Podporuje bezpečný model nasazování nových verzí<sup>31</sup> pro účely testování a ladění (akceptační testy).

- Kód aplikace je vyvíjen na revizním systému Git<sup>32</sup> využívající
  - repositář na službě GitLab provozované fakultou<sup>33</sup> nebo oddělením ICT<sup>34</sup>,
  - standardní branching model Git Flow<sup>35</sup> (nástroje OMGF<sup>36</sup> nebo Git-Flow Cheatsheet<sup>37</sup>) a
  - sémantické verzování<sup>38</sup>.
- Používání revizního systému se řídí pravidly *commitování*, zejména
  - *commit* každé dílčí změny funkcionality,
  - zachování funkcionality celku přes jednotlivé *commity*,

---

<sup>24</sup> Viz [Best Practices](#) a kniha [The Pragmatic Programmer](#).

<sup>25</sup> Konkrétně [konvenci SUIT CSS](#).

<sup>26</sup> [JavaScript Quality Guidelines and Recommendations](#)

<sup>27</sup> [Code Conventions for the Java Programming Language](#)

<sup>28</sup> Např. neimportovat celý balíček, když z něj bude použita jen malá část.

<sup>29</sup> Pod svobodnými nebo open-source softwarovými licencemi a respektovat podmínky těchto licencí.

<sup>30</sup> [Gang of Four Design Patterns](#) či [Design Patterns na Wiki](#)

<sup>31</sup> [Deployment environment](#)

<sup>32</sup> [Jak na Git](#)

<sup>33</sup> [GitLab FIT ČVUT](#)

<sup>34</sup> [GitLab ICT](#)

<sup>35</sup> [Git Flow](#)

<sup>36</sup> [OMGF](#)

<sup>37</sup> [Git-Flow Cheatsheet](#)

<sup>38</sup> [Semantic Versioning 2.0.0](#)



- používání rozkazovacího tvaru v přítomném čase<sup>39</sup>.
- Součástí vývoje je udržování aktuálního souboru CHANGELOG<sup>40</sup> dostupného z webu na úrovni
  - nových funkcí (či inovací) vždy při jejich začlenění do vývojové větve,
  - nových MINOR verzí vždy při začlenění vývojové větve do provozní.
- Na společných (sdílených) větvích není povoleno přepisování historie.
- Veškeré texty verzování jsou anglicky.

## Kontrola kvality (QA)

Veškerý kód se před nasazením patřičně kontroluje na úrovni automatizovaných nástrojů a dílčích (jednotkových a dalších) testů. Součástí kontroly kódu je (jednoduchý) schvalovací proces.

Alternativně se kód vyhodnocuje, zda splňuje stanovené kvalitativní metriky.

- Vývoj kódu se opírá o kontrolní nástroje jako zejména
  - příslušný *linter*<sup>41</sup>.
- Veškeré nasazování změn kódu (merge) procházejí kontrolním procesem<sup>42</sup> s následujícími pravidly.
  - Veškeré merge jsou prováděny formou požadavků na začlenění<sup>43</sup> (dále PR).
  - Veškeré PR (bez ohledu na svou podstatu a závažnost) budou potvrzované minimálně druhým členem týmu – programátorem, alternativně nadřízeným.
  - PR kritického požadavku si může jeho řešitel sám akceptovat. O takovém úkonu neprodleně vyrozumí členy týmu. Povinnost potvrzení podle předchozího bodu zůstává, však může být učiněno dodatečně (bez zbytečného prodlení).
  - Součástí kontrolního procesu nasazování je continuous integration<sup>44</sup> na GitLabu<sup>45</sup>.
- Kód obsahuje automatické testy, mezi které patří zejména
  - jednotkové testy,
  - integrační testy (API, resp. automatizované průchody).
- Klíčová funkcionalita je ošetřena testy, které jsou specifikované v dokumentaci s maximální mírou automatizace.
- Vývoj kódu je řízený testy<sup>46</sup> [SHOULD].
- Minimální požadované hodnoty metrik<sup>47</sup> pomocí fakultní služby Sonar<sup>48</sup> jsou stanoveny následujícím způsobem: [SHOULD]
  - Method Total Length (< 30 lines)
  - Class Total Length (< 300 lines)
  - Unit Tests Line coverage (> 70 %)
  - Unit Tests Branch coverage (> 70 %)
  - Density of duplicated lines (< 5 %)
  - Lack of cohesion of methods (< 3)
  - Average complexity by method (< 5)

<sup>39</sup> [How to Write a Git Commit Message](#)

<sup>40</sup> [Keep a Changelog](#)

<sup>41</sup> Platí zejm. pro dynamické a značkovací jazyky; viz např. [doporučení pro JavaScript](#) a [CSSLint](#).

<sup>42</sup> [Best Practices for Code Review](#) a [Code reviews v praxi](#)

<sup>43</sup> [Code Review Via GitLab Merge Requests](#)

<sup>44</sup> [Continuous integration](#)

<sup>45</sup> [GitLab Continuous Integration](#)

<sup>46</sup> [Test-Driven Development](#)

<sup>47</sup> [Sonar Metric Definitions](#)

<sup>48</sup> [Sonar FIT ČVUT](#)

- Rules compliance index (žádné závady úrovně „blocker“ ani „critical“)

## Provoz, údržba a rozvoj aplikace a podpora uživatelů

Veškerá (nově vznikající) funkcionalita je uživatelům dostupná přehledně a jednoduše. Aplikace (nová verze) se nasazuje do provozu po splnění akceptačních testů. Součástí údržby a dlouhodobého rozvoje aplikace je sběr informací o používání a jejich pravidelné vyhodnocování následované patřičným zapracováním do aplikace.

- Vzhled uživatelského rozhraní (UI) aplikace je moderní, přehledný, vzdušný a tvořený obsahem.<sup>49</sup>
- Sada akceptačních testů je specifikovaná pro účely testování všech dostupných a nově vznikajících funkcí.
- Sběr dat se provádí na základě
  - zpětné vazby uživatelů prostřednictvím
    - funkce issue tracking v rámci GitLabu a
    - e-mailu na helpdesk,
  - logování<sup>50</sup> a integrace s monitorovacími službami na úrovni
    - chyb (fatal, warning),
    - informačních zpráv o používání<sup>51</sup> (používanost funkcí, doby trvání, přístupy) a
    - systémových zpráv a dalších výstupů.
- Proces vyhodnocování dat zahrnuje
  - podporu uživatelů a
  - opravy chyb včetně klasifikace jejich závažnosti.
- [SHOULD] Proces rozvoje od návrhu po realizaci za účelem
  - vylepšování stávajících funkcí (optimalizace chodu a procesů),
  - přidávání nových funkcí.
- [SHOULD] Proces testování nových verzí aplikace zahrnuje
  - provoz nezávislé (beta) verze,
  - podporu AB testování,
  - provádění inspekcí a heuristik,
  - pozorování.

## Bezpečnost a ochrana osobních údajů

Aplikace je standardně zabezpečená; zejména nepracuje s hesly uživatelů a veškerá komunikace probíhá přes šifrovaný protokol. Aplikace také respektuje nařízení rektora o ochraně osobních údajů.

- Aplikace respektuje principy bezpečných webových aplikací<sup>52</sup>, jmenovitě
  - veškerá komunikace (S2S, S2C) probíhá přes HTTPS,
  - jako API poskytuje různé úrovně oprávnění pomocí scopes<sup>53</sup>,

---

<sup>49</sup> [UXMyths: You don't need the content to design a website.](#)

<sup>50</sup> [Logging Best Practices](#)

<sup>51</sup> [Google Analytics s využitím událostí \(events\)](#)

<sup>52</sup> Viz [principy OWASP.](#)

<sup>53</sup> [Securing Access with OAuth2: How to deal with OAuth Scopes](#)

- nepracuje s hesly uživatelů; autentizace, resp. autorizace, probíhá přes Shibboleth (není-li potřeba autorizace), resp. autorizační server FIT (protokol OAuth 2.0)<sup>54</sup>.
- Aplikace respektuje nařízení rektora o ochraně osobních údajů<sup>55</sup>, zejména dokumentace definuje,
  - jaké informace jsou citlivé/osobní,
  - jaká data jsou dostupná kterým uživatelům v závislosti na autorizaci uživatele (např. anonymní uživatel, přihlášený uživatel, student, vyučující, administrátor),
  - které citlivé/osobní informace jsou přístupné v rozporu s nařízením.

## Další koncepty webových aplikací (W2.0)

Aplikace explicitně zohledňuje možnosti využití níže uvedených webových konceptů a případně dalších. Vzhledem k omezení jednotlivých projektů mohou být využití konceptů pouze součástí projektové dokumentace – byť jen jako potenciální rozšíření funkcionality s uvedenými přínosy a konkrétními příklady.

- Dokumentace (např. v příloze) popisuje možnosti a úroveň nezbytnosti využití všech následujících konceptů:
  - RSS,
  - personalizace,
  - customizace,
  - folksonomie (tagování),
  - social networking,
  - real-time web<sup>56</sup>,
  - crowdsourcing<sup>57</sup>,
  - kolaborace,
  - průvodce (wizardy),
  - konfigurátory (rozšířených dotazů vyhledávání, parametrů služby),
  - gamifikace<sup>58</sup>,
  - mikrodata<sup>59</sup>.

---

<sup>54</sup> Viz [Autorizační server FIT \(OAuth 2.0\)](#).

<sup>55</sup> [Příkaz rektora č. 5/2015 Ochrana osobních údajů na ČVUT v Praze](#)

<sup>56</sup> [Real-time web \(Wiki\)](#)

<sup>57</sup> [Crowdsourcing \(Wiki\)](#)

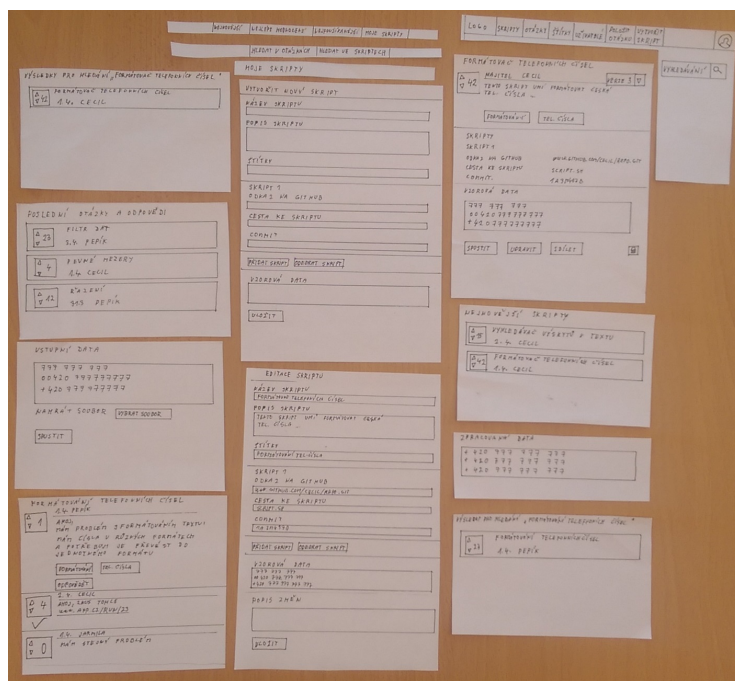
<sup>58</sup> [Gamification \(Wiki\)](#)

<sup>59</sup> [Microdata \(Wiki\)](#)



# Prototyp

Pro otestování procesů a prototypů obrazovek (kap. 3.5) byl vytvořen papírový prototyp pomocí kterého lze předvést nejdůležitější obrazovky aplikace.

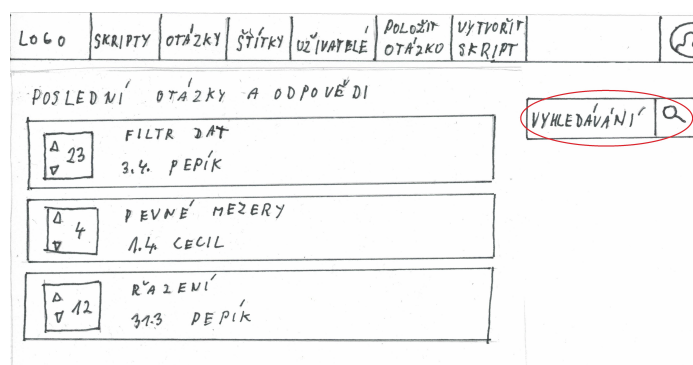


Obrázek C.1: Pohled na součásti papírového prototypu

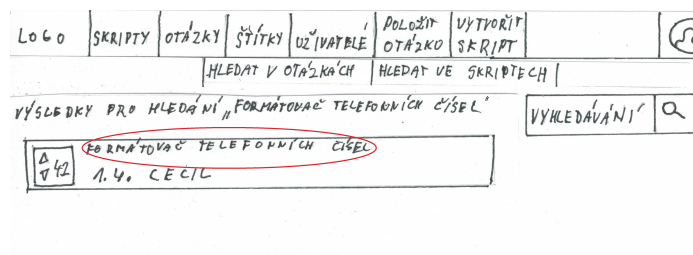
## C.1 Ilustrace testování prototypu

Tato kapitola obsahuje obrázky ilustrující ideální průchod aplikací při procházení testovacích scénářů.

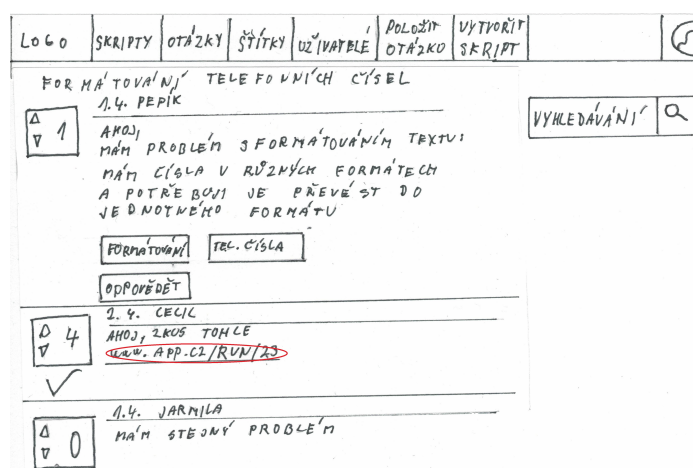
## C. PROTOTYP



Obrázek C.2: Krok 1 testovacího scénáře 1



Obrázek C.3: Krok 2 testovacího scénáře 1



Obrázek C.4: Krok 3 testovacího scénáře 1

## C.1. Ilustrace testování prototypu

LOGO	SKRIPTY	OTÁZKY	ŠTÍTKY	UŽIVATELE	POLOŽIT OTÁZKU	VYTVOŘIT SKRIPT	
FORMÁTOVAČ TELEFONNÍCH ČÍSEL							
42	MAJITEL CECIL	VERZE 3	VYHLEDÁVÁNÍ				
TENTO SKRIPT UMI FORMÁTOVAT ČESKÁ TEL. ČÍSLA ...							
FORMÁTOVÁNÍ		TEL. ČÍSLA					
VSTUPNÍ DATA							
<div style="border: 1px solid black; padding: 5px;">           777 777 777            00420 777777777            +420 777 777777         </div>							
NAHRÁT SOUBOR		VYBRAT SOUBOR					
<b>SPUSTIT</b>							

Obrázek C.5: Krok 4 testovacího scénáře 1

LOGO	SKRIPTY	OTÁZKY	ŠTÍTKY	UŽIVATELE	POLOŽIT OTÁZKU	VYTVOŘIT SKRIPT	
FORMÁTOVAČ TELEFONNÍCH ČÍSEL							
42	MAJITEL CECIL	VERZE 3	VYHLEDÁVÁNÍ				
TENTO SKRIPT UMI FORMÁTOVAT ČESKÁ TEL. ČÍSLA ...							
FORMÁTOVÁNÍ		TEL. ČÍSLA					
VSTUPNÍ DATA							
<div style="border: 1px solid black; padding: 5px;">           777 777 777            00420 777777777            +420 777 777777         </div>							
NAHRÁT SOUBOR		VYBRAT SOUBOR					
<b>SPUSTIT</b>							
ZPRACOVANÁ DATA							
<div style="border: 1px solid black; padding: 5px;">           + 420 777 777 777            + 420 777 777 777            + 420 777 777 777         </div>							

Obrázek C.6: Krok 5 testovacího scénáře 1

## C. PROTOTYP

LOGO	SKRIPTY	OTÁZKY	ŠTÍTKY	UŽIVATELE	POLOŽIT OTÁZKU	VYTVOŘIT SKRIPT	
------	---------	--------	--------	-----------	----------------	-----------------	--

POSLEDNÍ OTÁZKY A ODPOVĚDI

▲ 23 ▼	FILTR DAT 3.4. PEPIK
▲ 4 ▼	PEVNÉ MEZERY 1.4. CECIL
▲ 12 ▼	ŘAZENÍ 3.3. PEPIK

VYHLEDÁVÁNÍ

Obrázek C.7: Krok 1 testovacího scénáře 2

LOGO	SKRIPTY	OTÁZKY	ŠTÍTKY	UŽIVATELE	POLOŽIT OTÁZKU	VYTVOŘIT SKRIPT	
------	---------	--------	--------	-----------	----------------	-----------------	--

VYTVOŘIT NOVÝ SKRIPT

NÁZEV SKRIPTU

POPIS SKRIPTU

ŠTÍTKY

SKRIPT ?  
ODKAZ NA GITHUB

CESTA KE SKRIPTU

COMMIT

VZOROVÁ DATA

VYHLEDÁVÁNÍ

Obrázek C.8: Krok 2 testovacího scénáře 2



## C.1. Ilustrace testování prototypu

LOGO	SKRIPTY	OTÁZKY	ŠÍTKY	UŽIVATELE	POLOŽIT OTÁZKU	VYTVOŘIT SKRIPT	
------	---------	--------	-------	-----------	----------------	-----------------	--

FORMÁTOVAČ TELEFONNÍCH ČÍSEL

MAJITEL CECIL VERZE 3 VYHLEDÁVÁNÍ

42 TENTO SKRIPT UMIÍ FORMÁTOVAT ČESKÁ TEL. ČÍSLA ...

---

SKRIPTY

SKRIPT 1

ODKAZ NA GITHUB WWW.GITHUB.COM/CECIL/REPO.GIT

CESTA KE SKRIPTU SCRIPT.SH

COMMIT 1A3M462B

---

VSTUPNÍ DATA

777 777 777

00420 777 777 777

+420 777 777 777

Obrázek C.9: Krok 3 testovacího scénáře 2

LOGO	SKRIPTY	OTÁZKY	ŠÍTKY	UŽIVATELE	POLOŽIT OTÁZKU	VYTVOŘIT SKRIPT	
------	---------	--------	-------	-----------	----------------	-----------------	--

FORMÁTOVAČ TELEFONNÍCH ČÍSEL

MAJITEL CECIL VERZE 3 VYHLEDÁVÁNÍ

42 TENTO SKRIPT UMIÍ FORMÁTOVAT ČESKÁ TEL. ČÍSLA ...

---

VSTUPNÍ DATA

777 777 777

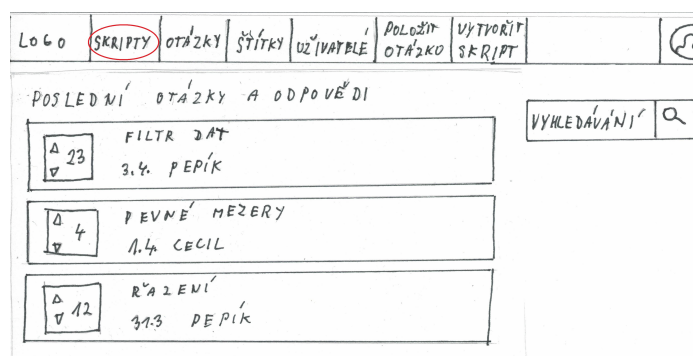
00420 777 777 777

+420 777 777 777

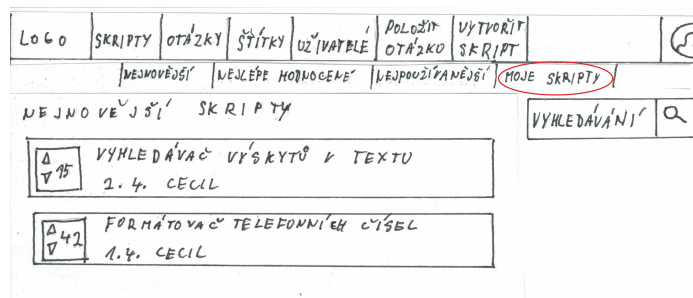
NAKRAJ+ SOUBOR

Obrázek C.10: Krok 4 testovacího scénáře 2

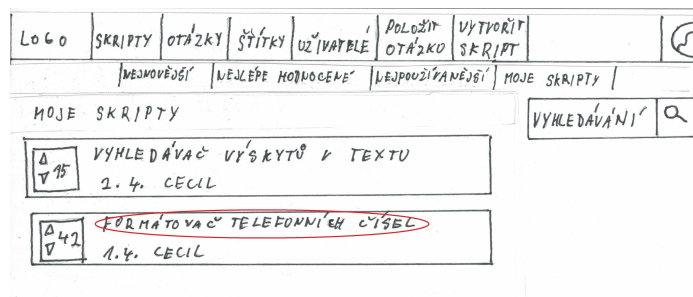
## C. PROTOTYP



Obrázek C.11: Krok 1 testovacího scénáře 3



Obrázek C.12: Krok 2 testovacího scénáře 3



Obrázek C.13: Krok 3 testovacího scénáře 3

## C.1. Ilustrace testování prototypu

LOGO	SKRIPTY	OTÁZKY	ŠTÍTKY	UŽIVATELE	POLOŽIT OTÁZKU	VYTVORIT SKRIPT	
FORMÁTOVAČ TELEFONNÍCH ČÍSEL							
	MAJITEL CECIL	VERZE 3	VYHLEDÁVÁNÍ				
42	TENTO SKRIPT UMIÍ FORMÁTOVAT ČESKÁ TEL. ČÍSLA ..						
FORMÁTOVÁNÍ		TEL. ČÍSLA					
SKRIPTY							
SKRIPT 1							
ODKAZ NA GITHUB			WWW.GITHUB.COM/CECIL/REPO.GIT				
CESTA KE SKRIPTU			SCRIPT.SH				
COMMIT.			1A3M462B				
VZOROVÁ DATA							
<pre> 777 777 777 00420 777 777 777 +420 777 777 777                 </pre>							
SPOSTIT		UPRAVIT	SDÍLET				

Obrázek C.14: Krok 4 testovacího scénáře 3

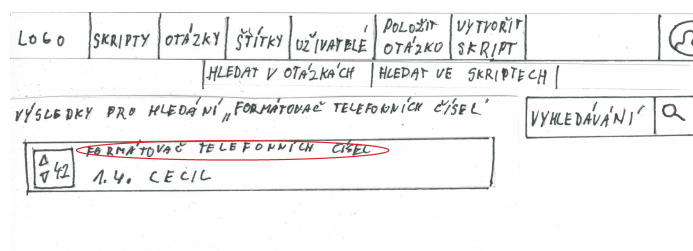
LOGO	SKRIPTY	OTÁZKY	ŠTÍTKY	UŽIVATELE	POLOŽIT OTÁZKU	VYTVORIT SKRIPT	
POSLEDNÍ OTÁZKY A ODPOVĚDI							
	FILTR DAT	VYHLEDÁVÁNÍ					
23	3.4. PEPIK						
	PEVNÉ MEZERY						
4	1.4. CECIL						
	RÁZENÍ						
12	3.3. PEPIK						

Obrázek C.15: Krok 1 testovacího scénáře 4

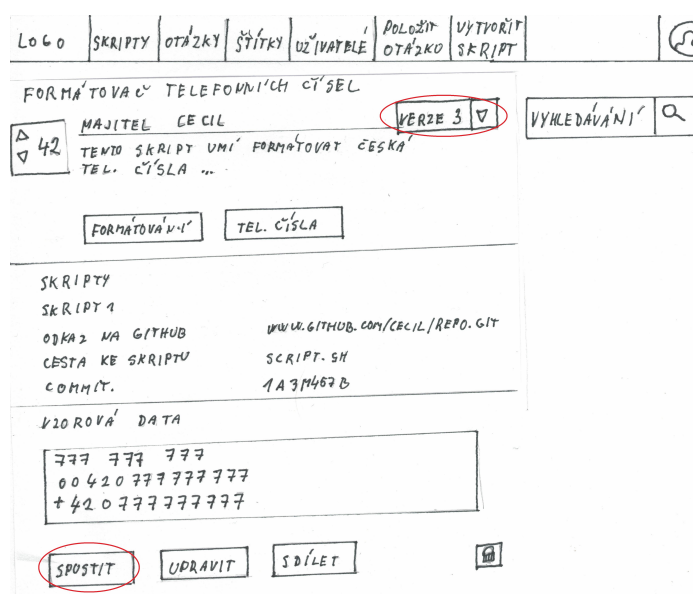
LOGO	SKRIPTY	OTÁZKY	ŠTÍTKY	UŽIVATELE	POLOŽIT OTÁZKU	VYTVORIT SKRIPT	
				HLEDAT V OTÁZKÁCH	HLEDAT VE SKRIPTECH		
VÝSLEDKY PRO HLEDÁNÍ 'FORMÁTOVÁNÍ TELEFONNÍCH ČÍSEL'							VYHLEDÁVÁNÍ
	FORMÁTOVÁNÍ TELEFONNÍCH ČÍSEL						
23	1.4. PEPIK						

Obrázek C.16: Krok 2 testovacího scénáře 4

## C. PROTOTYP



Obrázek C.17: Krok 3 testovacího scénáře 4



Obrázek C.18: Krok 4 testovacího scénáře 4

---

## Příručka nasazení

Tato kapitola popisuje postup pro sestavení a nasazení aplikace.

### D.1 Nasazení webového portálu

Pro nasazení webového portálu je nejprve nutné nasadit základní webovou aplikaci Question2Answer. Instrukce k nasazení a požadavky na technologie jsou popsány v [25].

Po úspěšném nasazení Question2Answer vytvořte novou složku v adresáři „plugins“ s názvem „bash-overflow“. Do té nakopírujte zdrojový kód pluginu. V administračním systému Questions2Answer přejděte do záložky plugins a stiskem tlačítka proveďte inicializaci databáze. Poté v seznamu pluginů najděte BashOverflow a rozklikněte jeho možnosti. Zde nastavte vlastnosti pluginu, primárně URL adresu lokace komponenty pro spouštění skriptů.

### D.2 Nasazení komponenty pro spouštění skriptů

Služba spouštějící skripty vyžaduje pro svůj běh následující technologie:

- Java<sup>40</sup> verze 1.8 nebo výše,
- Apache Maven<sup>41</sup> verze 3.3.9 nebo výše,
- Docker Server<sup>42</sup> verze 17.05 nebo výše,
- Apache Tomcat<sup>43</sup> verze 8.5.14 nebo výše.

#### Postup nasazení:

---

<sup>40</sup><http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>

<sup>41</sup><https://maven.apache.org/>

<sup>42</sup><https://www.docker.com/community-edition>

<sup>43</sup><http://tomcat.apache.org/>

## D. PŘÍRUČKA NASAZENÍ

---

1. V souboru „config.properties“ změňte nastavení konfigurace aplikace podle použitého prostředí. Následně soubor přesuňte do adresáře „/Script-Executor-Service/“.
2. Nasazení aplikace probíhá pomocí WAR souboru, jenž lze sestavit pomocí Maven příkazu „mvn clean install“.
3. V systému je nutné vytvořit obraz kontejneru spouštěného v Docker. V podložce „Docker“ příkazem „docker build -t name ./“ dojde k vytvoření obrazu kontejneru se jménem „name“.

## Seznam použitých zkratk

- API** Application Programming Interface
- BASH** Bourne again shell
- CPU** Central processing unit
- CRLF** Carriage return and line feed
- CSS** Cascading Style Sheets
- CSV** Comma-separated values
- ČVUT** České vysoké učení technické v Praze
- FA** Fakulta architektury
- FEL** Fakulta elektrotechnická
- FIT** Fakulta informačních technologií
- HiFi** High-Fidelity
- HTML** HyperText Markup Language
- JS** JavaScript
- JSON** JavaScript Object Notation
- LoFi** Low-Fidelity
- OS** Operační systém
- PC** Personal computer
- RAM** Random-access memory
- REST** Representational State Transfer

## E. SEZNAM POUŽITÝCH ZKRATEK

---

**RSS** Rich Site Summary

**SQL** Structured Query Language

**UCD** User Centered Design

**URL** Uniform Resource Locator

**UTF** UCS/Unicode Transformation Format

**VŠE** Vysoká škola ekonomická

**WAR** Web archive

**XML** eXtensible Markup Language

**YAML** Ain't Markup Language



---

## Obsah přiloženého CD

readme.txt.....	stručný popis obsahu CD
src	
_ impl.....	zdrojové kódy implementace
_ web.....	zdrojové kódy pluginu do aplikace Question2Answer
_ run.....	zdrojové kódy komponenty spouštějící skripty
_ thesis.....	zdrojová kódy práce ve formátu $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
text.....	text práce
_ BP_Znamenáček_Martin_2017.pdf.....	text práce ve formátu PDF