

bLoG

Obsah

Obsah	2
Autoři	3
Popis projektu, motivace	3
Strategický záměr „TO BE“	3
Obchodní přínos	3
Stav „AS IS“	3
Analýza SWOT	3
Analýza 5F	4
Analýza PESTLE	4
Funkční požadavky	5
Nefunkční požadavky	5
Seznam uživatelů	5
Případy užití	5
Class diagram	8
Výběr vhodné architektury	8
Frontend	8
Backend	8
Testování Backend	9
Testování Frontend	9
Hosting	9
Diagram nasazení	9
Diagram komponent	10
Rozbor a výběr alternativ návrhu řešení	10
WBS – rozdělení projektu na dílčí procesy	11
Zdroje	11
Normy a standardy	11
Matice zodpovědnosti	11
Analýza rizik FMEA	13
Znovu použitelnost	13
Metriky	13
Plán podpory	13
Vyhodnocení	13

Autoři

Dominik Prokš
Michael Slavev
Matěj Bárta

Popis projektu, motivace

Po několika minutové diskuzi jsme se rozhodli vytvořit službu pro správu webového blogu. Jelikož si myslíme, že tvorba takovéto služby je úměrná naším znalostem a v tomto projektu nám jde hlavně o to otestovat naše znalosti a schopnosti.

Strategický záměr „TO BE“

Ukázka naší práce / znalostí / dovedností.

Obchodní přínos

Tento projekt nebude sloužit jako zdroj finančního ohodnocení, ale jako testování našich zkušeností, znalostí vývoje.

Stav „AS IS“

Projekt slouží jako platforma pro další rozvoj týmu.

Analýza SWOT

SWOT – tabulka - 1

Silné stránky <ul style="list-style-type: none">• Zkušený tým vývojářů• Kvalitní znalosti problematiky (Studenti ČVUT FEL)• Nízké náklady	Slabé stránky <ul style="list-style-type: none">• Časová vytíženost vývojářů• Stres z domácího vězení (nařízené karantény)
Příležitosti <ul style="list-style-type: none">• Možnost prodeje podobného systému	Hrozby <ul style="list-style-type: none">• Hrozba zhroucení pracovníků z důvodu nařízené karantény• Výpadek elektřiny

Analýza 5F

Stávající konkurenti:

- Nemá hrozbu - aplikace nebude finančně výnosná

Potenciální konkurenti:

- Nemá hrozbu - aplikace nebude finančně výnosná

Dodavatelé:

- Nemá hrozbu - aplikaci vytváří interní tým a nemusí čekat na externí dodavatele

Kupující:

- Nemá hrozbu - aplikace nebude finančně výnosná

Substituty:

- Nemá hrozbu - aplikace nebude finančně výnosná

Analýza PESTLE

Politické:

- Je zde hrozba, že KSČM bude zakládat skvělé blogy

Ekonomické:

- Vzhledem k tomu, že aplikace není vytvářena za účelem zpeněžení, není toto hrozba
- Aktuální situace je velice nestabilní

Sociální:

- Situace stabilní

Technologické:

- Vysoký výběr technologií k použití

Legislativní:

- Situace stabilní

Ekologická :

- Kromě spotřeby el. energie, IT obor není náročný na přírodní zdroje

Funkční požadavky

- přidávat, editovat, kategorizovat, mazat a třídit články.
- přidávat, editovat a mazat své komentáře
- mazat komentáře ostatních
- ohodnocovat články
- nahlížet a vyhledávat články
- možnost mazání uživatelů
- přihlášení a registrace uživatelů

Nefunkční požadavky

- Aplikace musí mít webové rozhraní optimalizované v prohlížečích Firefox, Chrome, Opera
- Aplikace bude obrněná vůči SQLInjection a XSS
- “User-friendly” GUI
- Možnost rozšíření aplikace v budoucnu
- Uložení všech dat v centralizované databázi (PostgreSQL)

Seznam uživatelů

Systém disponuje dvěma uživateli: Blog Owner, Blog User

Blog Owner

Slouží ke správě celého blogu. Má k dispozici všechny funkcionality.

Blogu User

Slouží k nahlížení a komentování jednotlivých článků. Má k dispozici nahlížení, vyhledávání a komentování článků

Případy užití

Use case diagram napojený na jednotlivé role níže.

Registrace

Neregistrovaný uživatel má možnost vytvořit uživatelský účet.

Přihlášení

Každý uživatel má možnost se přihlásit do již vytvořeného účtu.

Prohlížení článku

Všichni uživatelé mohou prohlížet články.

Vyhledávat články

Všichni uživatelé mohou vyhledávat články.

Přidat komentář

Přihlášení uživatelé mohou přidávat vlastní komentáře.

Smazat vlastní komentář

Přihlášení uživatelé mohou smazat vlastní komentáře.

Editovat vlastní komentář

Přihlášení uživatelé mohou editovat vlastní komentáře.

Ohodnotit článek

Přihlášení uživatelé mohou hodnotit komentáře.

Smazat komentář jiného uživatele

Admin může smazat komentáře cizích uživatelů. (Např. vulgární či rasistické komentáře)

Přidat článek

Admin může přidávat obsah ve formě článku.

Smazat článek

Admin může smazat obsah ve formě článku.

Editovat článek

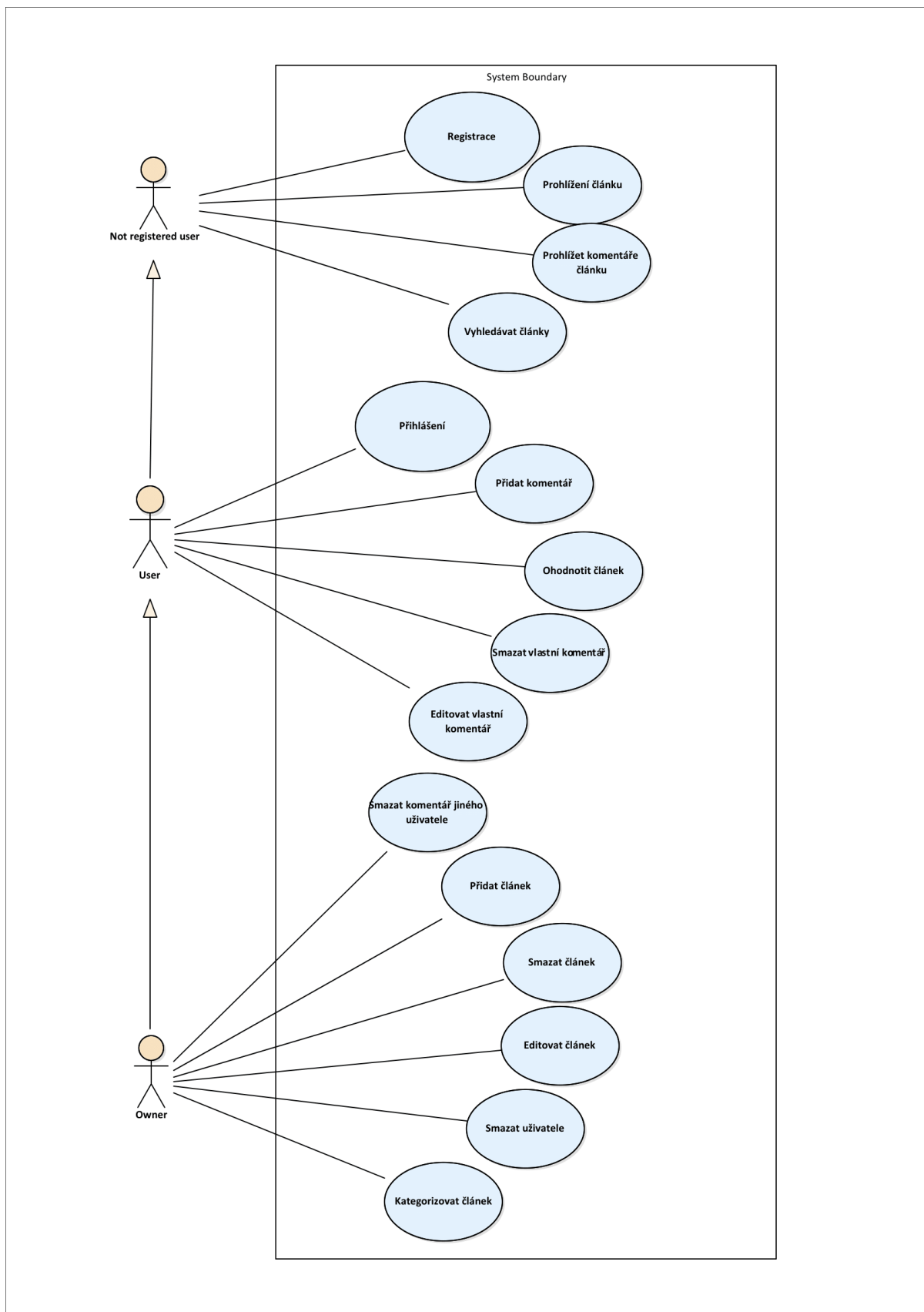
Admin může editovat obsah ve formě článku.

Smazat uživatele

Admin může smazat uživatelské účty.

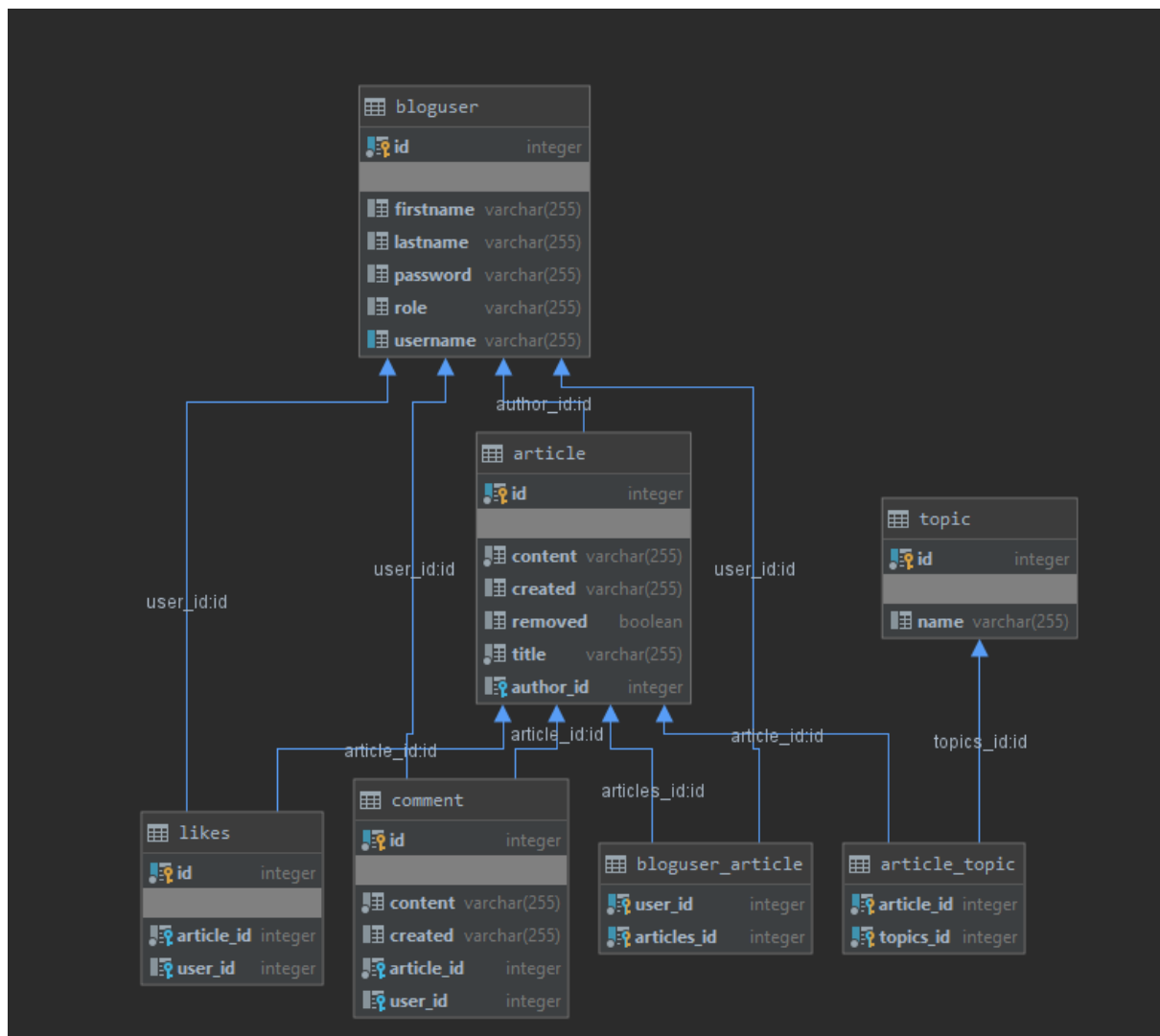
Kategorizovat článek

Admin může přidávat různé kategorie článkům.



UseCase diagram - Obrázek 1

Class diagram



Class diagram - Obrázek 2 (Role - varchar, protože je vytvořená přes enum ROLE_ADMIN, ROLE_USER)

Výběr vhodné architektury

Frontend

- React
- HTML5, CSS3, ECMAScript 6 (Javascript 6)

Backend

- Java 8+
 - Spring Boot
 - Rest API
- PostgreSQL

Testování Backend

- JUnit
- Postman

Testování Frontend

- Selenium web driver

Hosting

- Heroku

Diagram nasazení

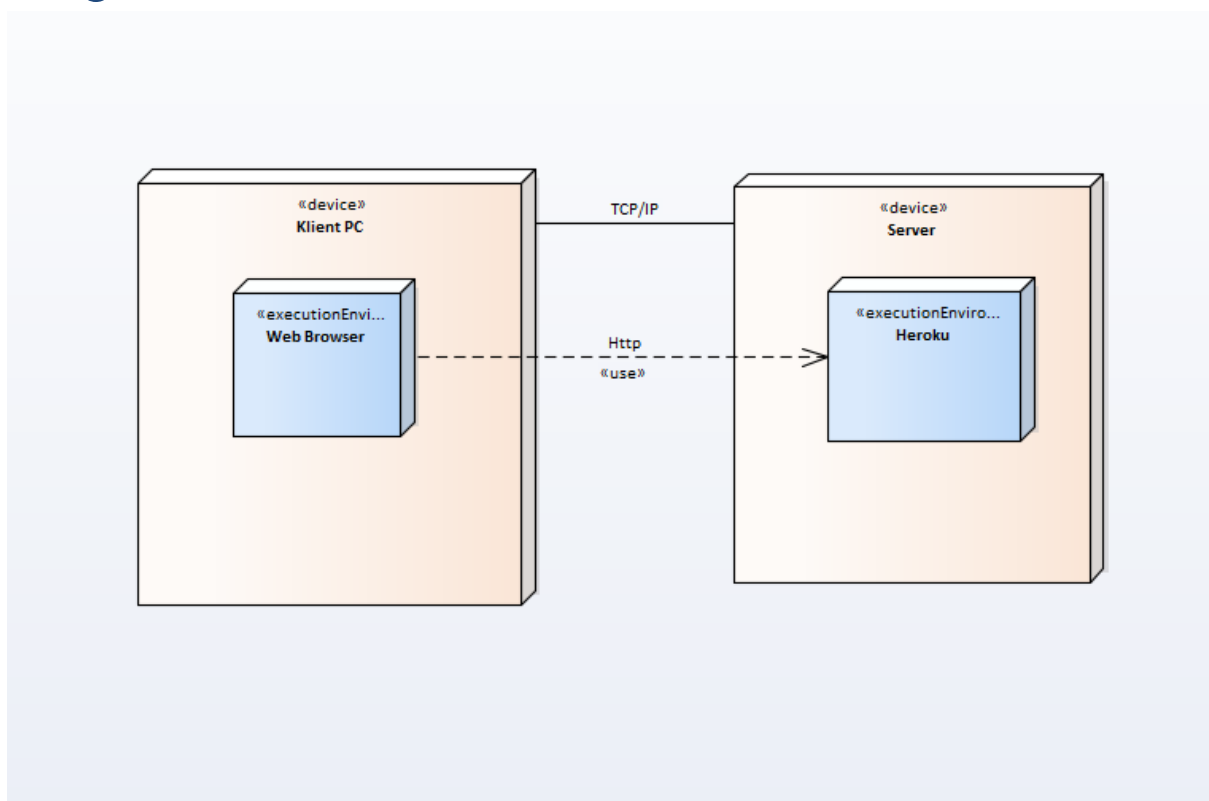


Diagram nasazení - Obrázek 3

Diagram komponent

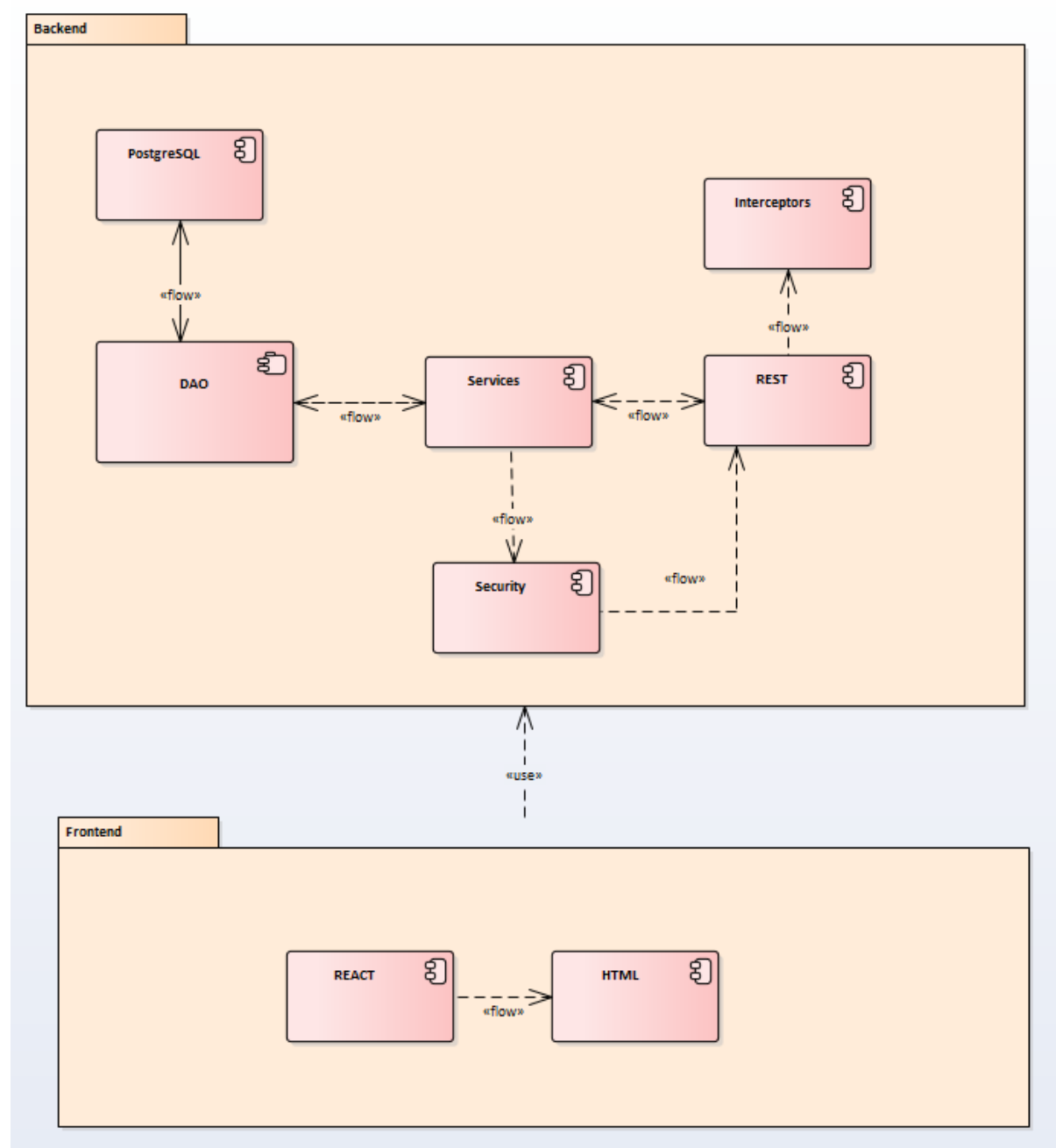


Diagram komponent - Obrázek 4

Rozbor a výběr alternativ návrhu řešení

Vzhledem k motivu projektu nemá cenu přemýšlet nad variantou vývoje aplikace třetí stranou, či využití open-source řešení. Jelikož chceme využít své znalosti a naučit se nové technologie jediná možnost pro nás tedy je vyvinout aplikaci sami. To ovšem zabere hodně času a proto jsme se rozhodli v některých částech využít námi již vytvořeného kódu z předchozí projektů, abychom si vývoj časově usnadnili.

WBS – rozdělení projektu na dílčí procesy

Projekt jsme rozdělili na technologie:

1. Frontend
 - 1.1. React
 - 1.2. Bootstrap
2. Backend
 - 2.1. PostgreSQL
 - 2.2. DAO
 - 2.3. Hazelcast
 - 2.4. Spring Security Basic Authentication
 - 2.5. Interceptors
 - 2.6. REST

Zdroje

Pro vývoj a nasazení není potřeba velké množství zdrojů. Pouze:

- PC s funkčním IDE, Tomcat
- Účet na Github, Heroku

Normy a standardy

Při vývoji jsme se drželi následujícími standardy:

- Java
- Bootstrap
- React
- Javascript
- PostgreSQL
- Spring
- Javax Interceptors
- REST
- Java Design Patterns
- Hazelcast

Matice zodpovědnosti

Matice zodpovědnosti - Tabulka 2

	Matěj Bárta	Michael Slavev	Dominik Prokš
React			X
Bootstrap			X
PostgreSQL	X	X	X
DAO	X	X	X

Hazelcast		X	
Spring Security Basic Authentication		X	
Interceptors	X		
REST	X	X	X
Dokumentace	X		

Analýza rizik FMEA

Prvek Funkce	Možná vada	Možné následky vady	V Ý Z N A M	Možná příčiny	V Ý S K Y T	Stávající opatření pro prevenci	OD HA LI TE LN OST	R P N	Dopo rčená opatření	Odpovědnost Termín	Proveden á opatření	V Ý Z N A M	V Ý S K Y T	OD HA LI TE LN OST	R P N
I M P L E M E N T A C E	Chybná Implementace	Špatná funkcio nalita	8	Neznalost Technologií Příliš rychlá implementace Pře používání kódů	7	Opatrnost	3	168	Revize na konci dne	Celý tým 5.6.2020 17:00	Revize před odevzdání m	8	6	6	2 8 8
	Nestih nutí deadline	Pozdní odevzdání	5	Špatný time management týmu	6	Opatrnost	9	270	Vytvořit harmonogra m	Celý tým 5.6.2020 17:00	žádné	5	6	9	2 7 0

FMAE - Tabulka 3

Znovu použitelnost

Kvůli tomu, jak je projekt specifický znovu použitelnost není stoprocentní, ale rozhodně se některé části dají znovu použít (např. Autentizace, Hazelcast)

Metriky

V projektu jsou sledovány následující metriky:

- Technologie a jejich správné využití
- Čas strávený na projektu
- Funkcionalita (use case)

Plán podpory

- Testování aplikace
- Rozšíření funkcionality

Vzhledem k tomu, že jsme si chtěli vyzkoušet nové technologie a naše dovednosti tak se testování příliš neřešilo a proto je velká šance, že některé chyby nebyli odhaleny. Co se týče druhého bodu rádi bychom přišli s více funkcionalitami, jako je např. propracovaný upvote systém (Reddit style) či cenzurování.

Vyhodnocení

Tento projekt, odhalil že zkušenosti v programování nejsou všechno. Mnohým chybám by se dalo předejít prvotním návrhem softwaru a harmonogramem prací. Jelikož jsme tuto část podcenili,

implementace tak probíhala lehce chaoticky a některé technologie nevyužívají svůj plný potenciál. Nicméně projekt přinesl velké množství nových zkušeností a přinutil nás udělat první krůčky v učení nových technologií jako je např. REACT.