



Sociální síť OneGram

Semestrální práce NSS

Přemek Bělka

Ondřej Bureš

Phuong Dong Cu

Jan Pivoňka

Letní semestr B211

Popis aplikace	3
Motivace	3
Strategický záměr	3
Obchodní přínos	3
Zvolená architektura	3
SWOT analýza	4
Analýza 5F (Porter's Five Forces)	4
Analýza PEST	5
Funkční požadavky	6
Nefunkční požadavky	8
Seznam rolí uživatel	8
Diagramy	9
Class diagram	9
Component diagram	10
Sekvenční diagram	11
Use Case	12
Diagram Nasazení	16
Rozbor a výběr alternativ návrhu řešení	16
Backend:	16
Frontend:	17
Hlavní vývojové prostředky:	17
Komunikační prostředky:	17
WBS - rozdělení projektu na dílčí procesy	17
Zdroje	18
Normy a standardy	18
Harmonogram GANTT	18
Analýza rizik FMEA	19
Znovupoužitelnost	19
Metriky	20
Plán odbavení	20
Plán podpory	20
Vyhodnocení	20

Popis aplikace

Sociální síť OneGram je nová platforma mezi sociálními sítěmi poskytující nový pohled na věc. Zakládáme si na tom, aby naše aplikace byla v první řadě lidská. Proto u nás nečekejte žádné roboty přemotivovaně kontrolující všechny příspěvky. Nebráníme se osočení z podobnosti s Instagramem nebo TikTokem. Naopak, děláme to co oni, ale lépe a bez reklam.

Motivace

Naší hlavní motivací při vývoji aplikace OneGram je široká základna uživatelů nespokojených s neustále se zvyšující komerčností ostatních sociálních sítí. Platící uživatelé mají v algoritmu ukazování příspěvků vyšší přednost, míra reklamy je až neúnosná, kontrolou procházejí příspěvky, které by neměly a naopak. Těmto uživatelům chceme přinést přátelské prostředí neovlivněné penězi a reklamou.

Strategický záměr

Naším cílem je zaměřit se na již zmíněnou skupinu uživatelů, představit jim naši platformu, seznámit je s jejím prostředím a postupně je dostat na svou stranu.

Obchodní přínos

Aplikace si neklade za cíl obchodní přínos, v případě nasazení bychom se snažili prosadit na trh absencí reklamních kampaní. Pokud bychom se toto rozhodli změnit, tak by obchodní přínos vycházel z právě poskytovaných reklam.

Zvolená architektura

Aplikaci vyvinem v microservice architektuře. Konkrétně bude aplikace rozdělena do následujících microservices.

- User - Data a operace s uživateli
- Post - Data a operace o příspěvcích, komentářích atd.

Jednotlivé microservices budou postavené na základě (tiered) architektury. Kde každá microservice bude vnitřně rozdělena do následujících vrstev:

Prezentační vrstva (JS + React)

Webová vrstva (GraphQL)

Bussiness vrstva (servisní vrstva obsahující bussiness logiku)

Datová vrstva

Databáze (Rádi bychom si vyzkoušeli použít MongoDB, anebo jinou noSQL databázi)

SWOT analýza



Analýza 5F (Porter's Five Forces)

Stávající konkurenti:

- Instagram
- Tik Tok
- Facebook
- Reddit
- a další

Všechny aplikace jsou v základní formě zdarma, a tak je zamýšlená i naše aplikace. V případě reklam je nutné počítat s tím, že tyto aplikace mají zajeté jméno a tedy reklamní společnosti budou preferovat je i v případech výrazně vyšších cen. Jejich finanční možnosti jsou ve všech případech prakticky neomezené.

Potenciální konkurence:

V tuto chvíli nevíme o žádné nově se vyskytující konkurenci. Nějaká s nejvyšší pravděpodobností vznikne v následku světové situace v Rusku, ale ta by pro nás nepředstavovala přímou konkurenci. Každý nově vstupující na trh, bude počítat s vysokou již existující konkurencí.

Dodavatelé:

V našem případě dodavatele neřešíme. Jediné, co by se dalo počítat by byli reklamní společnosti a s těmi v počáteční fázi vůbec nekalkulujeme.

Uživatelé:

Obecně se jedná o všechny bez omezení, ale v praxi primárně věková kategorie ve věku od 12-35 s tím, že věkové rozmezí se bude s časem zvětšovat.

Substituty:

Jelikož se v základu v případě sociálních jedná o služby provozované zdarma, tak roli substitutů představují veškeré konkurenční aplikace.

Analýza PEST

Politicko-legislativní vlivy

V této kategorii se na naší aplikaci týká GDPR a ručení za bezpečnost uživatelských dat.

Ekonomické faktory

Jak již bylo zmíněno, počítáme s tím, že aplikace bude poskytována zdarma a neplánujeme v tuto chvíli řešit reklamní prostor. Z toho vychází, že finanční stránka nebude rozebrána.

Sociálně-kulturní faktory

Z pohledu sociálních faktorů musíme řešit omezení obsahu naší aplikace. Z důvodu největší části předpokládané klientely, bychom se měli z výchovných a morálních důvodů vyhnout erotickému, násilnému, hazard podporujícímu nebo jinak nevhodnému obsahu.

Technologické vlivy

Obecně velcí hráči na trhu využívají rychle se vyvíjející technologie pro analyzování dat, predikování zájmů a předhazování obsahu, který byl pro dané uživatele doporučen. V rámci projektu nepředpokládáme, že bychom měli prostor na podobných inovacích pracovat.

Funkční požadavky

Legenda

UserService

PostService

Běžný uživatel

FR1 Vytvořit účet, přihlásit se, odhlásit se

Systém umožní uživateli si vytvořit nový účet, přihlásit se a odhlásit se.

FR2 Přidat/odebrat příspěvek

Systém umožní uživateli přidat do svého profilu příspěvek obsahující maximálně 5 obrázků. Příspěvek má textový popis společný pro všechny obrázky. Systém umožní uživateli odebrat příspěvek, který přidal.

FR3 Zobrazit seznam příspěvků uživatele

Systém umožní uživateli zobrazit seznam příspěvků jiného uživatele.

FR4 Zobrazit seznam komentářů u příspěvku

Systém umožní uživateli zobrazit seznam komentářů u příspěvků.

FR5 Přidat/odebrat komentář k příspěvku

Systém umožní uživateli smazat vlastní komentář/okomentovat příspěvek.

FR6 Přidat/odebrat komentář ke komentáři

Systém umožní uživateli přidat/odebrat komentář ke komentáři. Komentář lze přidat pouze ke komentáři, který není komentářem ke komentáři. Ale je komentářem k příspěvku.

FR7 Odebrat komentář jiného uživatele na vlastním příspěvku

Systém umožní uživateli odebrat komentář jiného uživatele, v případě, že je to komentář pod příspěvkem, který uživatel sám přidal.

FR8 Liknout/odliknout příspěvek nebo komentář

Systém umožní uživateli odebrat like/dát like na příspěvek/komentář.

FR9 Poslat žádost o sledování/přestat sledovat jiného uživatele

Systém umožní uživateli poslat jinému uživateli žádost o sledování. Systém umožní uživateli přestat sledovat uživatele. Sledování znamená, že se příspěvky sledovaného uživatele zobrazují sledovateli na k tomu určené stránce.

FR10 Zobrazit seznam nejnovějších příspěvků od sledovaných uživatel

Systém umožní uživateli zobrazit nejnovější příspěvky, které přidali uživatelé, které sleduje. Příspěvky jsou seřazeny od nejnovějších po nejstarší.

FR11 Nastavit viditelnost profilu

Systém umožní uživateli nastavit, jestli jeho příspěvky vidí všichni, anebo pouze lidé, kteří jej sledují.

FR12 Nastavit potvrzování žádostí o sledování

Systém umožní uživateli nastavit, jestli jsou žádosti o sledování jeho profilu automaticky ihned potvrzeny, anebo je nutné je potvrdit manuálně.

FR13 Zobrazit seznam žádostí o sledování

Systém umožní uživateli zobrazit seznam žádostí o sledování jeho profilu.

FR14 Zobrazit seznam sledujících a sledovaných profilů uživatele

Systém umožní uživateli zobrazit seznam sledujících a sledovaných uživatel, jakéhokoliv uživatele.

FR15 Změnit profilovou fotku

Systém umožní uživateli změnit svou profilovou fotku.

FR16 Odebrat vlastního sledujícího

Systém umožní uživateli zastavit jiného uživatele, který jej sleduje.

FR17 Zobrazení profilového bio

Systém umožní zobrazovat bio jiných uživatelů

FR18 Editace bio

Systém umožní editovat na svém účtu své bio

Administrátor (*administrátora jsme nakonec neimplementovali*)

FR19 Zablokovat/odblokovat uživatele

Systém umožní administrátorovi zablokovat/odblokovat uživatele. Blokace uživatele znamená, že se uživatel nemůže přihlásit do svého účtu. A pro ostatní uživatele jsou jeho příspěvky a profil nepřístupné.

FR20 Odebrat příspěvek

Systém umožní administrátorovi odebrat jakýkoliv příspěvek.

FR21 Odebrat komentář

Systém umožní administrátorovi odebrat jakýkoliv komentář.

Nefunkční požadavky

NFR 01 - Větší množství stylů aplikace

NFR 02 - Dostupnost aplikace

NFR 03 - Zabezpečení uživatelských dat

NFR 04 - Splňování GDPR pravidel

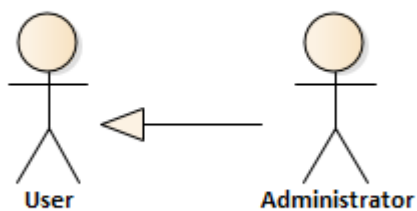
NFR 05 - Neomezenost přidaného uživatelského obsahu (početní)

NFR 06 - Blokování nevhodného obsahu

NFR 07 - Neomezená kvalita příspěvků v aplikaci

NFR 08 - Zdarma provoz bez využití reklam

Seznam rolí uživatel

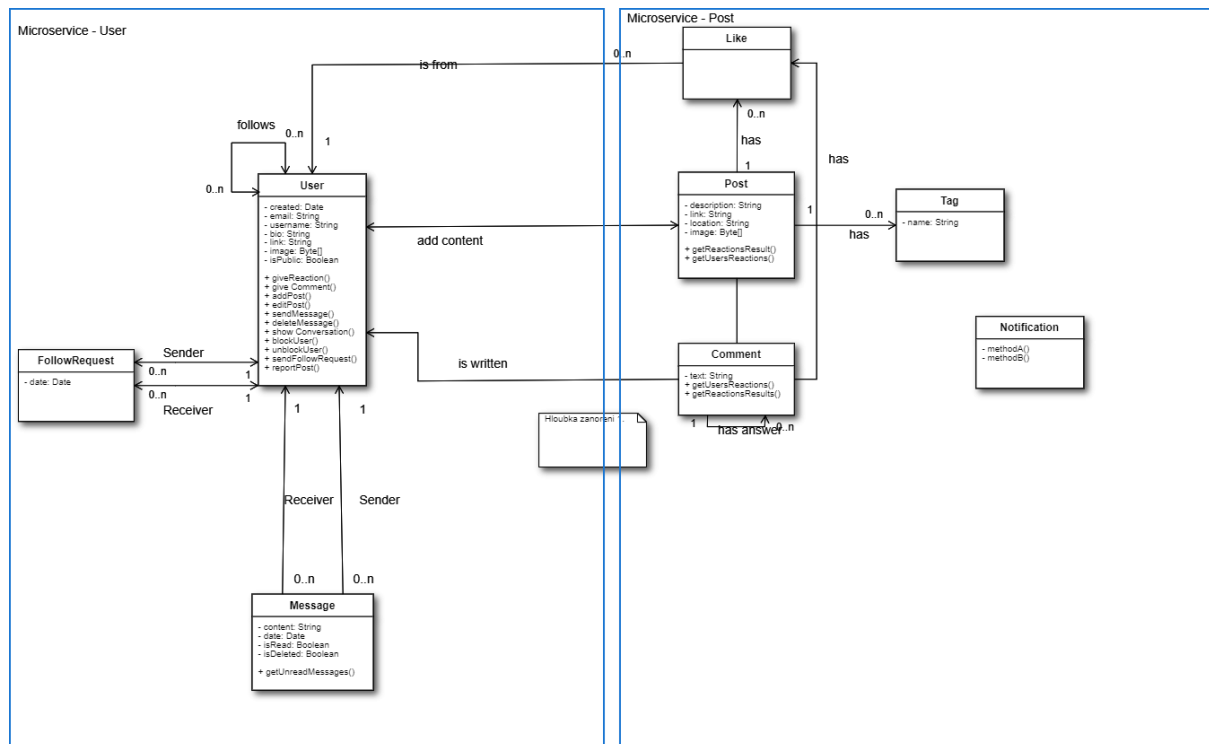


Standardní uživatel (dále pouze jako uživatel)

Administrátor

Diagramy

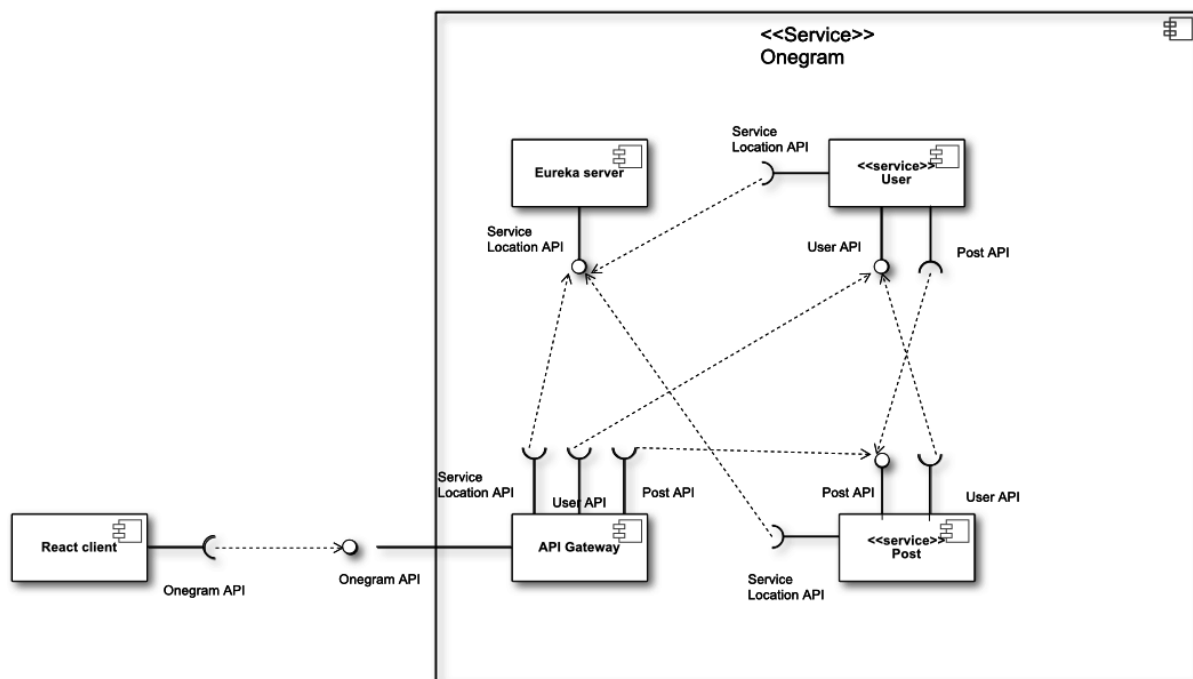
Class diagram



Návrhové vzory:

- **Simple factory** - Vytváření objektů pro testování. (User/test/environment/Generator, Post/test/environment/Generator)
- **Builder** - Pomocí `@Builder` anotace z Lombok. (Post/service/SysteminitializerImpl)
- **Composite** - Model Post mikroservisy je strom (Příspěvek -> komentý -> subkomentý). Pro každý uzel tohoto stromu je potřeba jednotný interface pro vytvoření/smazání líku. (Post/model/interfaces/Likeable).
- **Ghost** - Komentý, příspěvky a líky mají svého autora. Autor patří do bounded contextu User mikroservisy. Příspěvek, koment a like do bounded contextu Post mikroservisy. Autor se nenačítá hned přímo, ale pouze se na něj odkazuje idčkem.
- **Dependency Injection** - Spring.

Component diagram



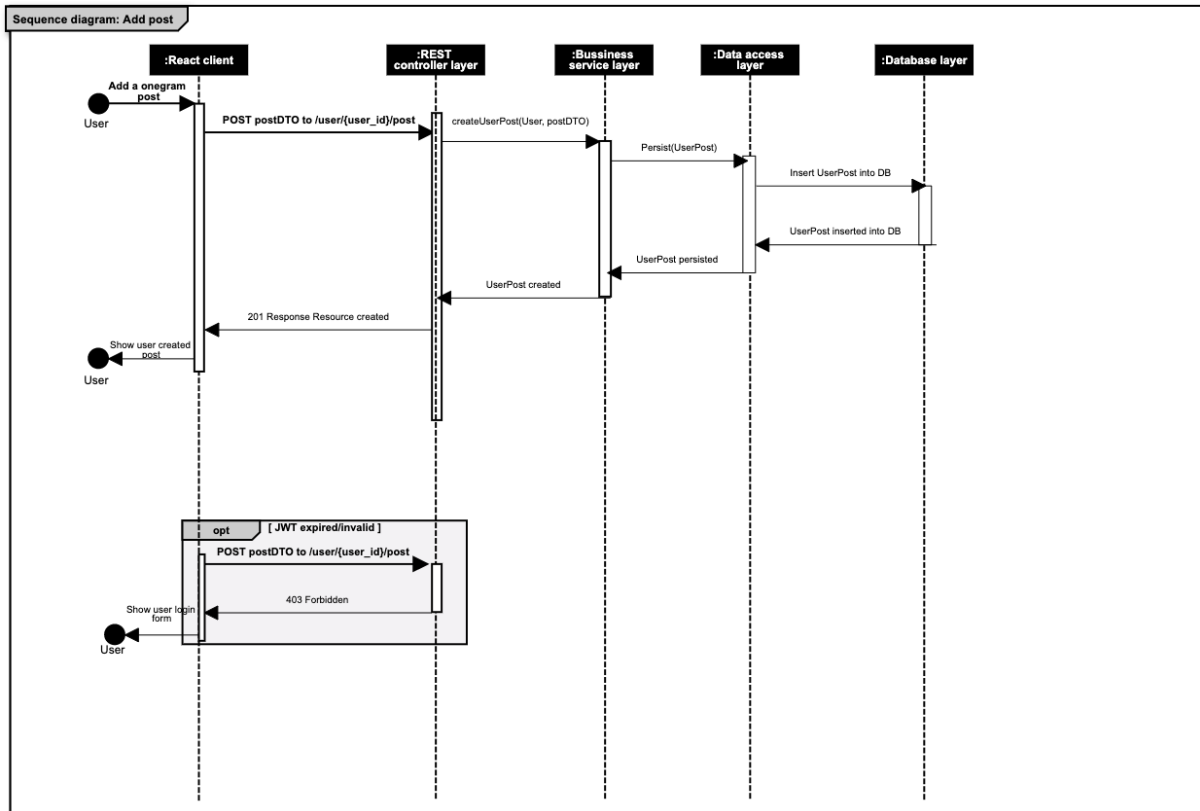
Návrhové vzory

- **API Gateway** - pro komunikaci s jednotlivými services z venčí (*nepodařilo se nám implementovat*)
- **Facade** - API jednotlivých services

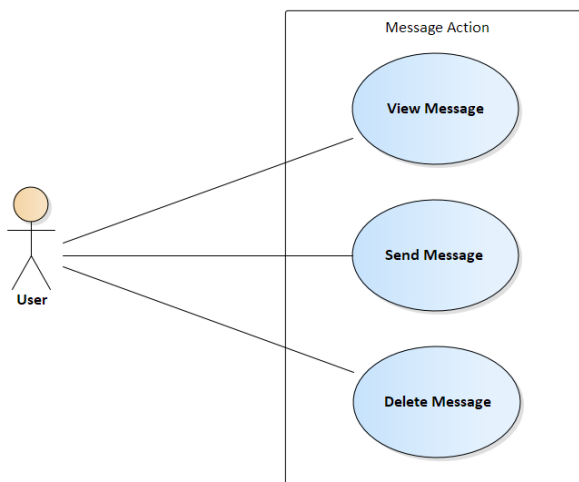
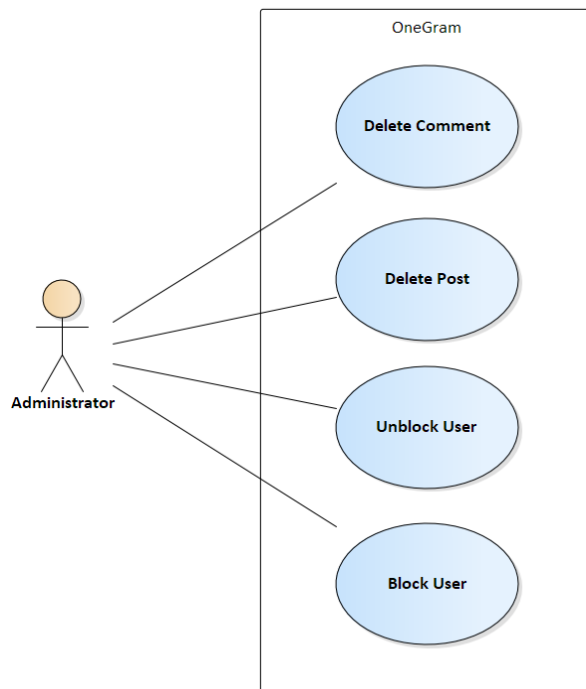
Eureka Server a API Gateway, se nám nakonec nepodařilo implementovat. Viz readme v repu.

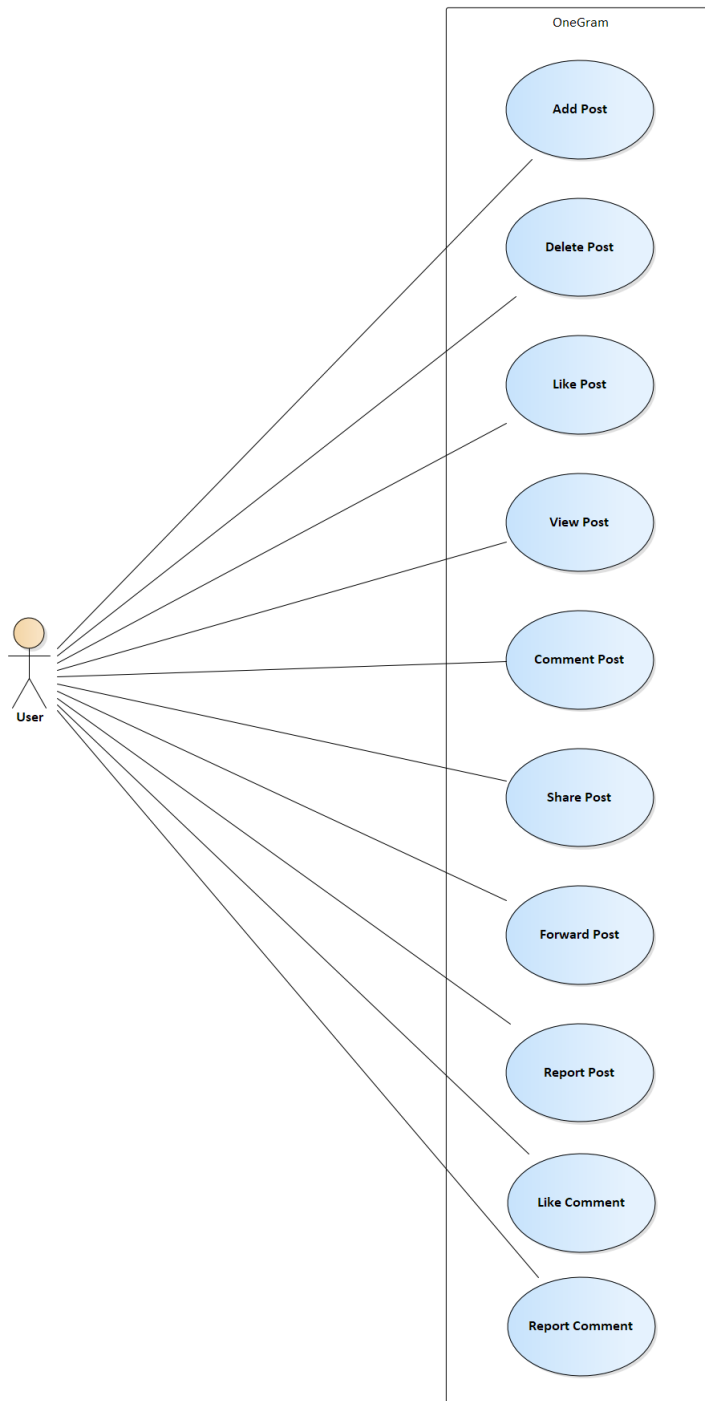
Sekvenční diagram

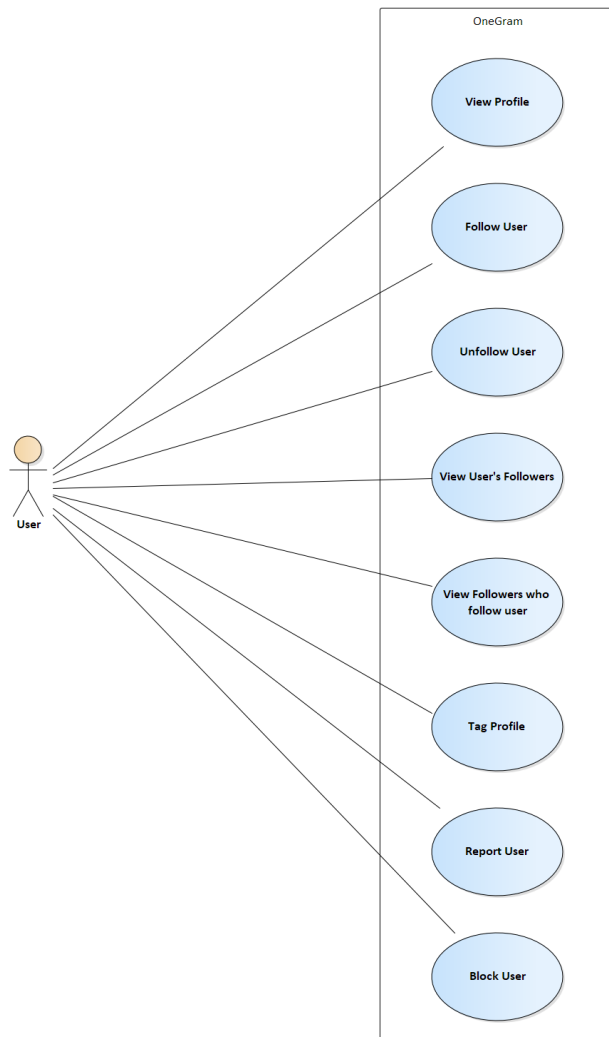
Pro demonstraci toho, jak jednotlivé vrstvy v naší architektuře budou zpracovávat požadavek od klienta jsme vytvořili vzorový diagram případu, kdy chce uživatel přidat nový příspěvek.



Use Case







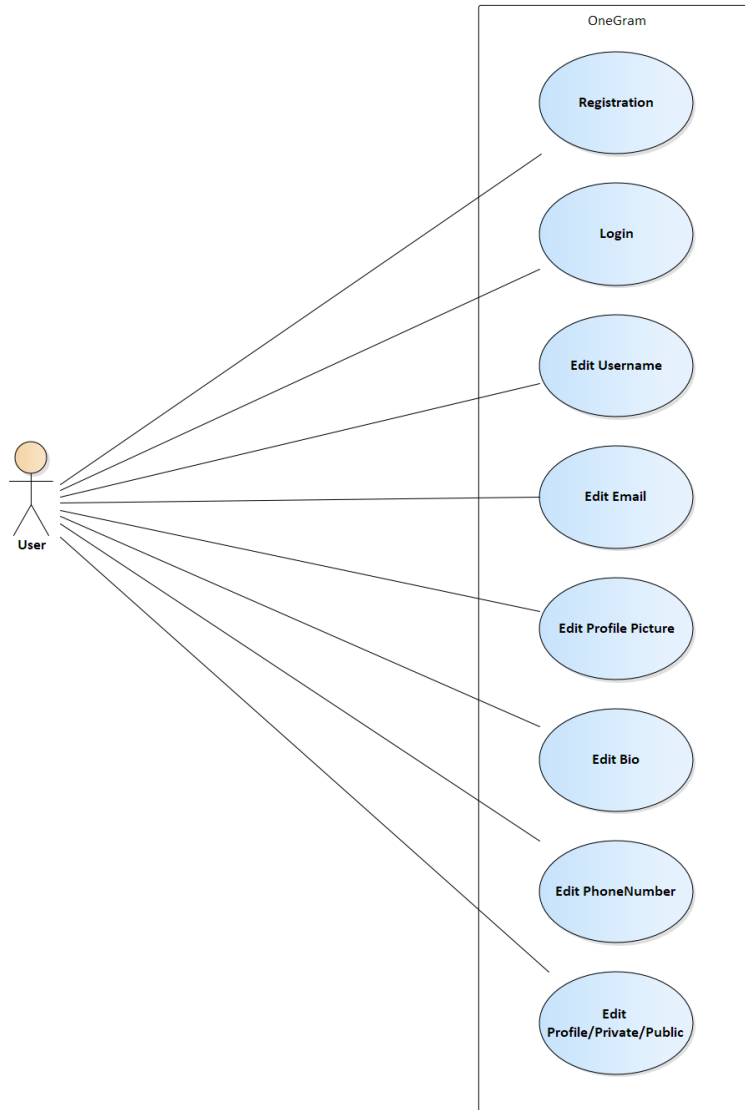
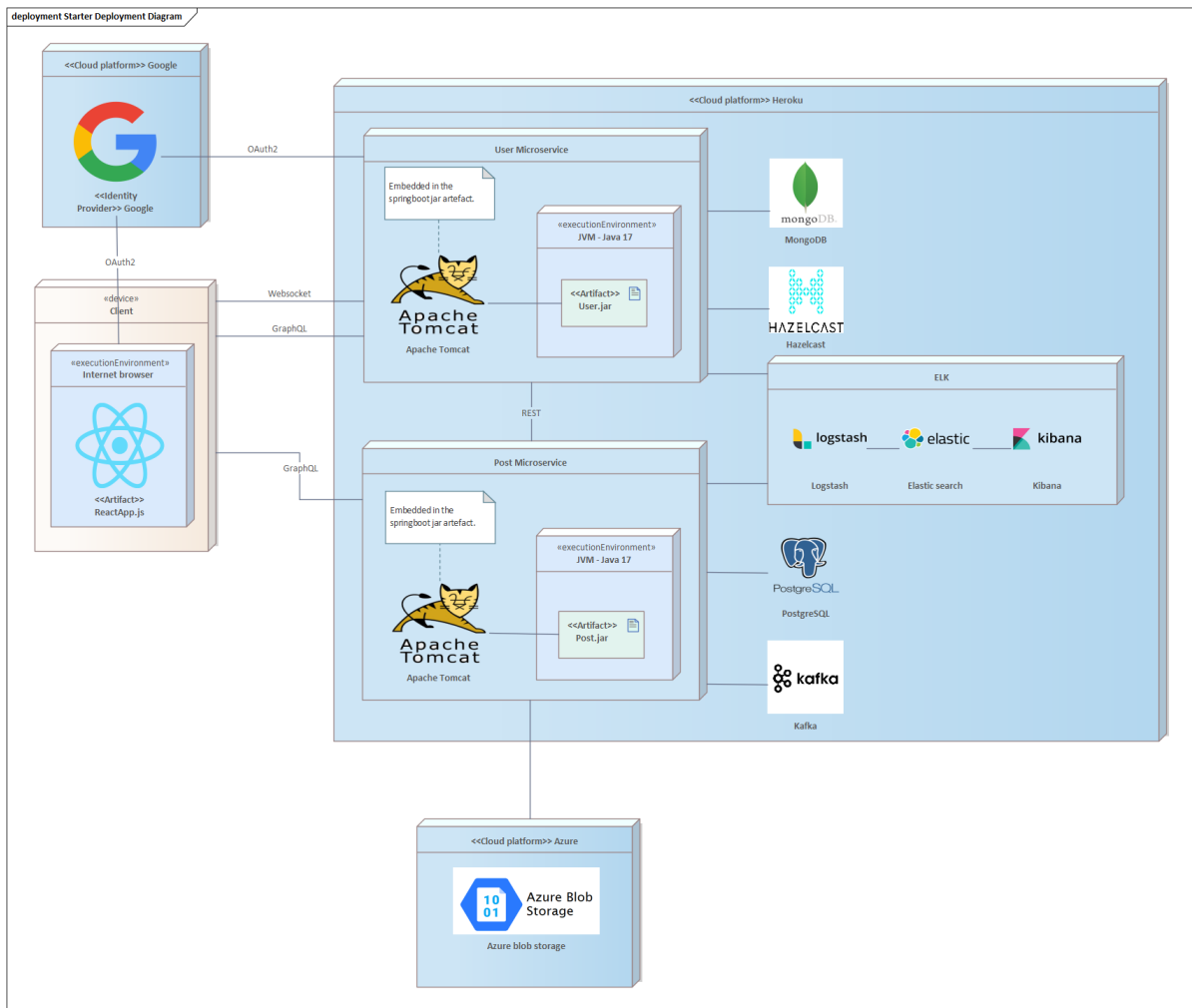


Diagram Nasazení



Rozbor a výběr alternativ návrhu řešení

Backend:

- **Docker** - Pro lokální vývoj jsem používali docker-compose
- **GraphQL** - Obě dvě microservices vystavují pro FE GraphQL API
- **REST** - Pro komunikaci mezi mikroservisama
- **Azure Blob Storage** - Použili jsme pro ukládání obrázků, FE dostává pak link na místo v Azure Storage
- **MongoDB**
- **Scheduling** - Pravidelné mazání cache
- **Hazelcast** - Autentikaci v systému provádí User mikroservisa
- **Interceptors** - Logujem příchozí dotazy
- **OAuth2** - Aplikace využívá google login
- **Websocket** - K messaging nám poslouží i websocket
- **Kafka** - Použili jsme pro messaging
- **ELK** - Logování

Frontend:

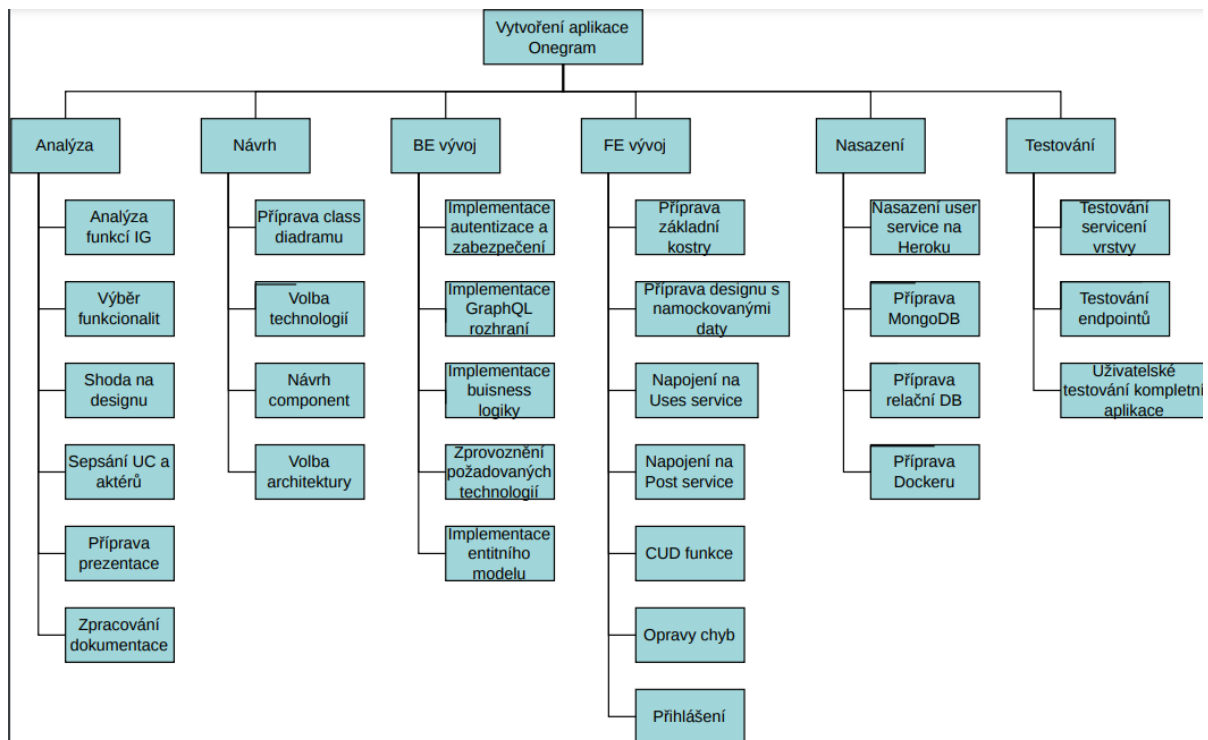
Hlavní vývojový prostředky:

- Pouze standardní webové technologie: nenabízelo by výhody reaktivních aplikací, nepřinášelo by větší prostor pro učení.
- **React**: zvolená technologie, o této technologii, jsme měli alespoň základní hrubou představu (jediná byla ve škole alespoň zmíněná), jeden z členů (paradoxně z BE týmu) měl zkušenost z jiného školního projektu. Dalším faktorem pak byla jeho současné zřejmě nejvyšší propagovanost.
- Vue: hlavní přímý konkurent.
- Angular: v poslední době často kritizovaná technologie na ustupujícím trendu.

Komunikační prostředky:

- Standardní XHR: obecně nedostatečné a nepraktické využívat pouze XHR. Šlo by proti směru využívaným v praxi
- Apollo: nejčastěji používaná technologie pro graphql v kombinaci s Reactem, původně vybráno jako stavební kámen, nicméně pro vleklé zásadní problémy s kompletním rozchozením od Apolla bylo opuštěno
- **Axios**: pro GraphQL méně používaná varianta, nicméně je s ním snadnější manipulace a jednodušší rozchození. V časové tísní, do které jsme se díky problémům s Apollem dostali ideální technologie.

WBS - rozdělení projektu na dílčí procesy



Zdroje

- Přednášky a cvičení NSS
- Zkušenosti získané z předmětů EAR, OMO a RSP
- Dokumentace jednotlivých technologií
- Youtube tutoriály
- Stack overflow (aby člověk mohl nejrychleji zjistit, která varianta řešení nefunguje)

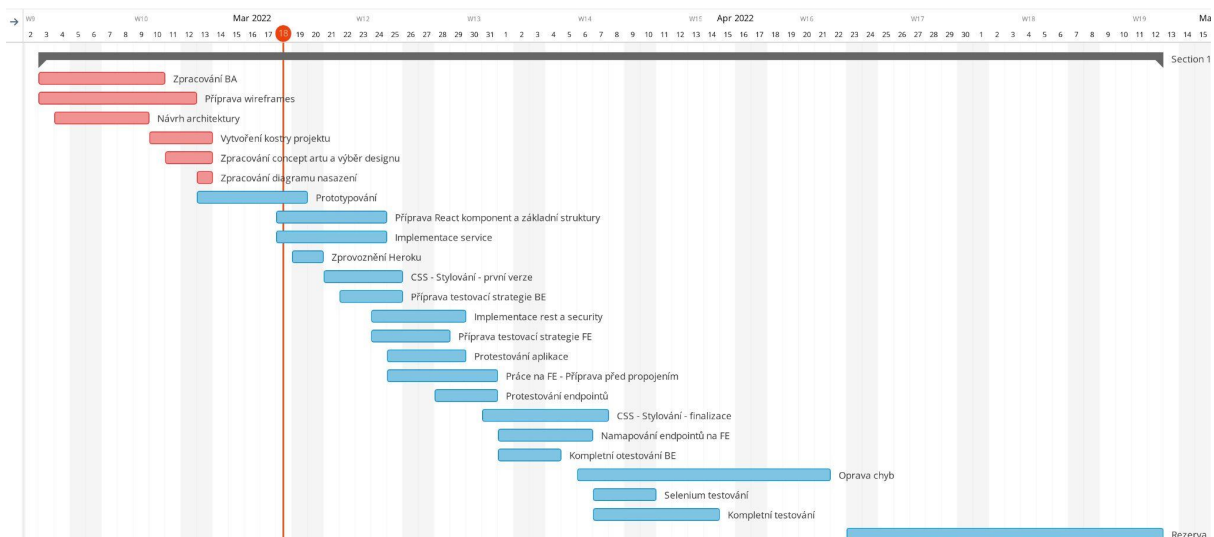
Normy a standardy

Vychází z informací dostupných pro IG

- Legislativní standardizované podmínky Instagramu (obdobné jiným sociálním sítím):
<https://1url.cz/ZrJPU>

Harmonogram GANTT

Přesné zmapování práce nemáme. Byla chaoticky rozložená průběžně po dobu celého semestru. Mapování naší práce by stejně neodpovídalo, proto zde použijeme již optimalizovaný návrh blížíící se ideálu z předmětu RSP.



Analýza rizik FMEA

Rizika projektu							
ID	Kategorie rizika	Popis	PST	Dopad	kategorie rizika	Strategie mitigace	Konkrétní mitigace
R01	Znalostní riziko	Nedostatečná znalost problematiky	Medium	Medium	B	Reduce	Důkladné vysvětlování podrobností celému týmu, názorná ukázka, týmové debaty
R02	Znalostní riziko	Nedostatečná znalost použitých technologií	Medium	High	B	Reduce	Použití některých již známých technologií, využití zkušeností lidí, kteří s danými technologiemi již
R03	Týmové riziko	Nedostatek motivace	Medium	Medium	B	Reduce	Popohánění lidí projektovými managery
R04	Týmové riziko	Nedostupnost lidí	Low	Medium	C	Accept	Kontrolování naplňování plánů, změna předávání informací, bodová penalizace dotyčné osoby
R05	Týmové riziko	Špatné řízení lidí	Medium	Medium	B	Reduce	Kontrolování naplňování plánů, změna stylu řízení, přenechání role
R06	Týmové riziko	Špatný odhad času	High	Medium	B	Reduce	Kontrolování naplňování plánů, dohánění vzniklého manka
R07	Technologické riziko	Potíže s fungováním serveru	High	High	A	Accept	Konzultace s učiteli
R08	Technologické riziko	Selhání záloh	Low	Low	C	Accept	Vytvoření nových dat
R09	Znalostní riziko	Problémy s kompatibilitou - serverová strana	Medium	High	B	Share	Průběžné testování a kontrolování, včasný začátek, více odpovědných lidí
R10	Znalostní riziko	Problémy s kompatibilitou - klientská strana	Medium	Medium	B	Share	Průběžné testování a kontrolování, včasný začátek, více odpovědných lidí
R11	Týmové riziko	Potíže s komunikací	Low	Medium	C	Fallback	Dodělování úkonů jinými lidmi, domluva se cvičícím, časové kompromisy
R12	Obecné riziko	Nucený přechod do home office módu	Low	Low	C	Accept	-
R13	Týmové riziko	Ukončení/přerušení studia některých členů	Low	Medium	C	Fallback	Domluva se cvičícím, přerozdělení práce
R14	Obecné riziko	Jiná zásadní světová komplikace	?	High	A	Accept	-
R15	Týmové riziko	Nestíhání termínů	Medium	High	B	Share	Domluva s cvičícím, průběžná kontrola pracovního harmonogramu, přerozdělování práce, vyšší časová investice do dokončení projektu
R16	Technologické riziko	Problémy s nasazením aplikace	Medium	High	B	Accept	Konzultace s učiteli
R16	Znalostní riziko	Neimplementovaná funkcionality	High	High	A	Fallback	Domluva se cvičícím
R17	Technologické riziko	Špatné zobrazování dat	High	High	A	Reduce	Kontrola a průběžné testování
R18	Technologické riziko	Chybná funkcionality	High	High	A	Reduce	Průběžné testování
R19	Znalostní riziko	Problémy s workflow Gitu	Medium	Medium	B	Reduce	Vysvětlení všem členům týmu, vytvoření návodů
R20	Znalostní riziko	Špatná kvalita kódu	High	Low	C	Reduce	Pověření osoby zodpovědné za kontrolu kódu ostatních

Znovupoužitelnost

- Jedná se o školní semestrální projekt. Projekt pokrývá většinovou funkcionalitu sociální sítě Instagram, nicméně další využití bude maximálně v podobě vývojového portfolia.

Metriky

- Při vývoji byl dbán ohled na následující metriky:
 - Přehlednost kódu
 - Funkcionalita

Výkonnost	7/10
Zabezpečení	8/10
Dostupnost	7/10
Spolehlivost	7/10
Funkčnost	9/10
Použitelnost	8/10
Znovupoužitelnost	6/10
Přehlednost	8/10
Testovatelnost	6/10

Plán odbavení

- Postservice je nasazena na <https://nss-onegram.herokuapp.com/>
- Projekt si uživatel stáhne a lokálně si spustí

Plán podpory

- Projekt dále podporován nebude

Vyhodnocení

- Projekt byl pro nás určitě přínosem, protože nám dal příležitost/ otevřel dveře k novým technologiím jak už na backendu, tak i na frontendu. Co se týče týmové práce, tak jsme si rozdělili na tým backend a tým frontend. Tým backend fungovalo celkem bez problému a tým frontend mělo občas lehké nedorozumění/"problémy", jak už kvůli nové technologii, nedostatečné komunikace, podceňování pracnosti/složitosti nebo nedostatečné samostatnosti. Co se týče výstupu projektu jako takového si myslíme, že jsme se s tím vypořádali dobře.