



STŘEDNÍ PRŮMYSLOVÁ ŠKOLA A OBCHODNÍ AKADEMIE,
BRUNTÁL, PŘÍSPĚVKOVÁ ORGANIZACE

Obor – IT

Školní sociální síť

Maturitní práce

Lukáš Dvorský

Vedoucí maturitní práce: Bc. Václav Paveza

Bruntál 2023

PROHLÁŠENÍ

Prohlašuji, že jsem svou maturitní práci vypracoval samostatně a použil jsem pouze prameny a literaturu uvedené v seznamu bibliografických záznamů.

V Bruntále dne 14.03.2024

PODĚKOVÁNÍ

Chtěl bych poděkovat svému vedoucímu práce Václavu Pavezovi, za pomoc a podporu při tvorbě maturitního projektu.

Dále bych chtěl poděkovat paní učitelce Ing. Barboře Friedlové Ph.D. za trpělivost a pomost při tvorbě podnikatelského záměru.

Speciální poděkování patří také panu učiteli Ing. Petrovi Černému, za pomoc při zprovoznování aplikace.

ABSTRAKT

Cílem práce je vytvořit uzavřenou sociální síť, která bude obsahovat nejzákladnější funkcionality sociálních sítí.

Práce obsahuje veškerý můj postup při tvorbě aplikace. Od návrhu systému až po jeho spuštění.

ABSTRACT

The goal of the thesis is to create a closed social network that will contain the most basic functionalities of social networks.

The thesis contains all my process in creating the application. From system design to its launch.

KLÍČOVÁ SLOVA

Sociální síť, webová aplikace, NextJS, React, podnikatelský záměr

KEYWORDS

Social media, web application, NextJS, React, business plan

OBSAH

Úvod	7
Webová aplikace.....	8
Návrh systému	8
Diagramy případu užití (Use Case Diagram)	8
Diagram aktivit (Activity Diagram)	11
Návrh databáze + Entity-relationship model	14
Výběr technologií	14
Tvorba aplikace	15
Vytvoření aplikace.....	15
Přihlášení uživatele.....	17
Příspěvky	18
Komentáře	19
Třídní skupiny	19
Admin panel	19
Vyhledávání.....	20
Nahrání uživatelů na sociální síť	20
Uvedení systému do provozu	21
Podnikatelský plán.....	22
Oprávnění k podnikání	22
Analýza zákazníků.....	22
Analýza konkurence	22
Analýza dodavatelů	23
Swot analýza.....	23
Reklama a propagace.....	24
Finanční analýza	25
Závěr.....	28
Seznam použitých zdrojů.....	29
Přílohy	29
Seznam zkratk a symbolů	30
Seznam tabulek a obrázků	31

ÚVOD

Zadání práce vychází z předmětu vývoj programových aplikací a databázové a informační systémy. Hlavním cílem práce je vytvořit platformu uzavřené školní sociální sítě. Platforma bude ve formě dynamické webové aplikace s databází a bude určena výhradně pro žáky a zaměstnance školy. Aplikace poskytne základní funkcionalitu, např. správu vlastního profilu, zobrazování a přidávání příspěvků, komentování příspěvků. Sít' bude shromažďovat příspěvky na hlavní sociální zeď. Nabídne také restrikce na zobrazování příspěvků do vybraných uživatelských skupin. Dále pak bude obsahovat administrátorské prostředí, které nabídne kompletní správu systému. Pro zvolené zadání zpracuji podnikatelský záměr. Součástí podnikatelského záměru je analytická, provozní část, výrobní, marketingový, organizační, finanční plán a hodnocení rizik.

WEBOVÁ APLIKACE

Webová aplikace je souhrnné označení pro software zprostředkovaný internetovým prohlížečem. Na rozdíl od desktopového softwaru ho nemusíte instalovat, stačí, když do prohlížeče zadáte požadovanou adresu.

NÁVRH SYSTÉMU

Rozhodl jsem se vytvořit webovou aplikaci pro moji školu. Je tvořená tak, aby ji bylo možno v budoucnu rozšířit. Sít' bude uzavřená, tudíž k ní bude mít přístup pouze omezený počet lidí (studenti a zaměstnanci školy).

Aplikace bude uživatelům poskytovat základní prvky sociálních sítí. Přihlašování, přidávání příspěvků, možnost zpětné vazby komentáři nebo označením palcem nahoru, vyhledání a zobrazení profilu uživatele, sledování uživatelů. Stejně jak na Facebooku, každý uživatel bude mít vlastní zeď s příspěvky, kde bude moci přidávat příspěvky kdokoliv. Následně budou vytvořené skupinové zdi, ke kterým budou mít přístup pouze vybraní uživatelé.

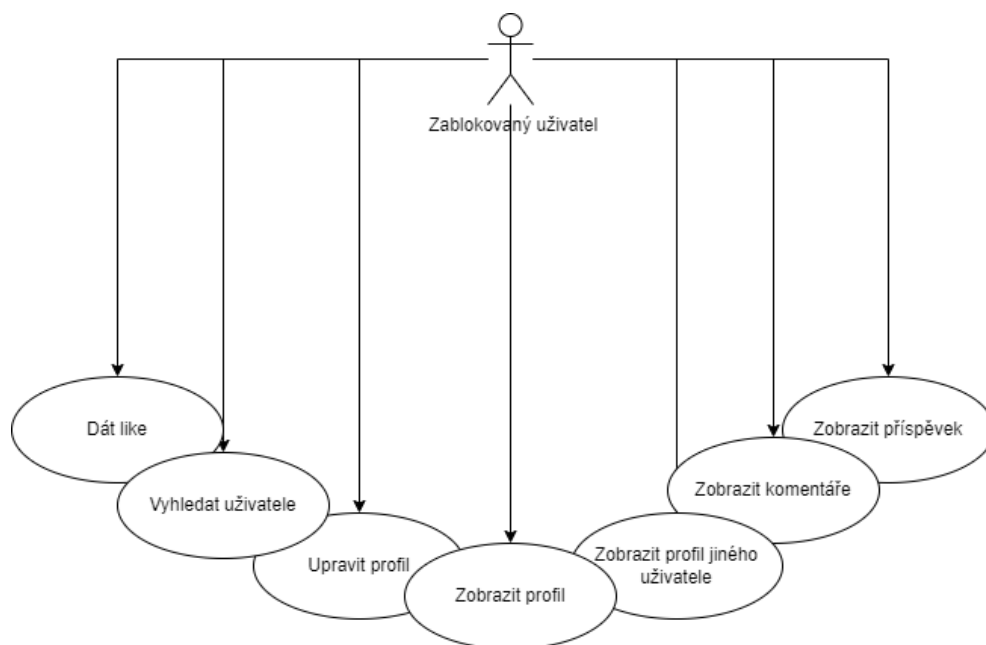
Aplikace bude také obsahovat administrátorský dashboard, kde ji bude moci administrátor spravovat.

Diagramy případu užití (Use Case Diagram)

Případ užití je sada několika akcí, které vedou k dosažení určitého cíle. Use Case může být přidání komentáře k článku, registrování nového uživatele nebo např. vytisknutí dokumentu. Definuje tedy jednu funkcionalitu, kterou by měl navrhovaný systém umět. Ta v sobě obsahuje další akce, např. přidání komentáře bude obsahovat ověření uživatele, validaci zadaných dat, zápis do databáze apod. To v diagramu zachyceno již nebude. [1]

Zablokovaný uživatel

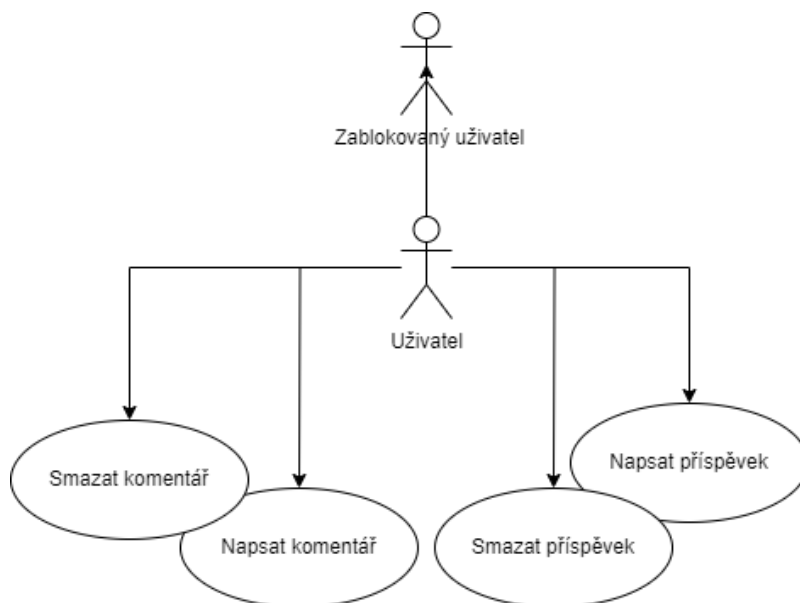
Bude moci prohlížet a ohodnocovat příspěvky, avšak žádný nevytvoří. To samé platí i pro komentáře.



Obrázek 1: Use Case diagram zablokovaného uživatele. Zdroj: autor

Klasický uživatel

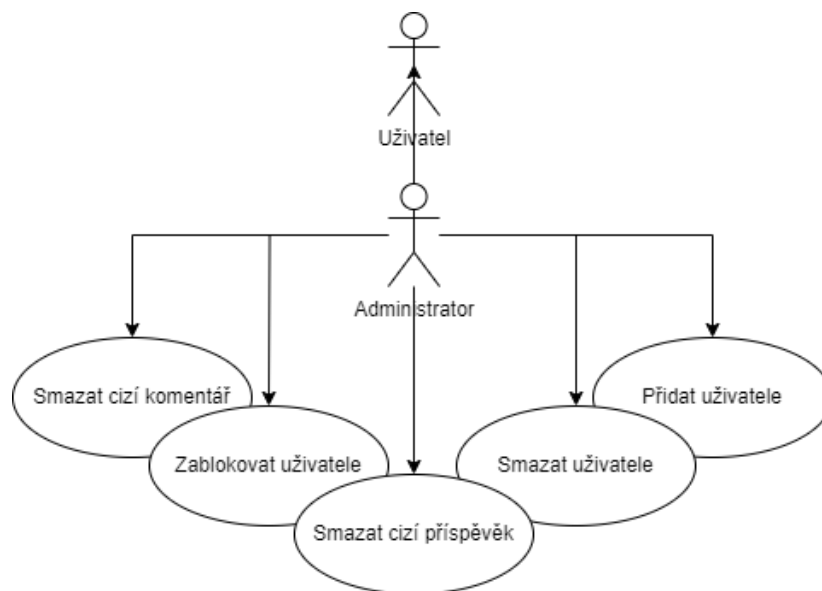
Má stejná práva jako zablokovaný uživatel. Může tvořit a mazat příspěvky a komentáře.



Obrázek 2: Use Case diagram klasického uživatele. Zdroj: autor

Administrátor

Má stejná práva jako klasický uživatel. Může spravovat aplikaci a dohlížet nad ní.



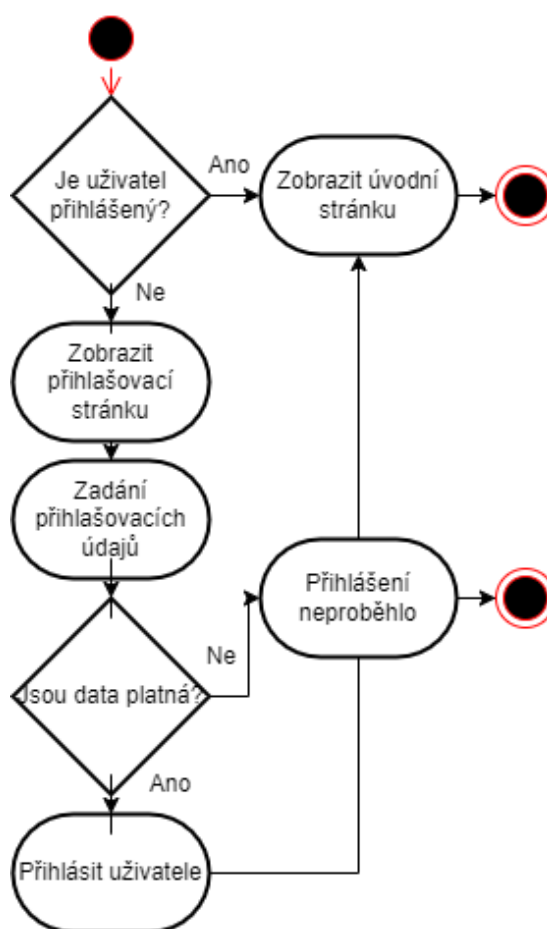
Obrázek 3: Use Case diagram administrátora. Zdroj: autor

Diagram aktivit (Activity Diagram)

Activity diagram slouží k zaznamenání procesu, nikoli stavu. V anglické literatuře se v souvislosti s Activity diagramem často setkáte s pojmem *workflow*, který zahrnuje právě jednotlivé kroky pracovního procesu, rozhodování mezi nimi a případně další pokročilejší prvky. [2]

Přihlášení uživatele

Activity diagram mluví sám za sebe. Na každé stránce, probíhá kontrola, zda je uživatel přihlášený. Jestliže není, je uživatel přesměrován na stránku s přihlášením, kde vyplní data.



Obrázek 4: Activity diagram přihlášení uživatele. Zdroj: autor

Příspěvky

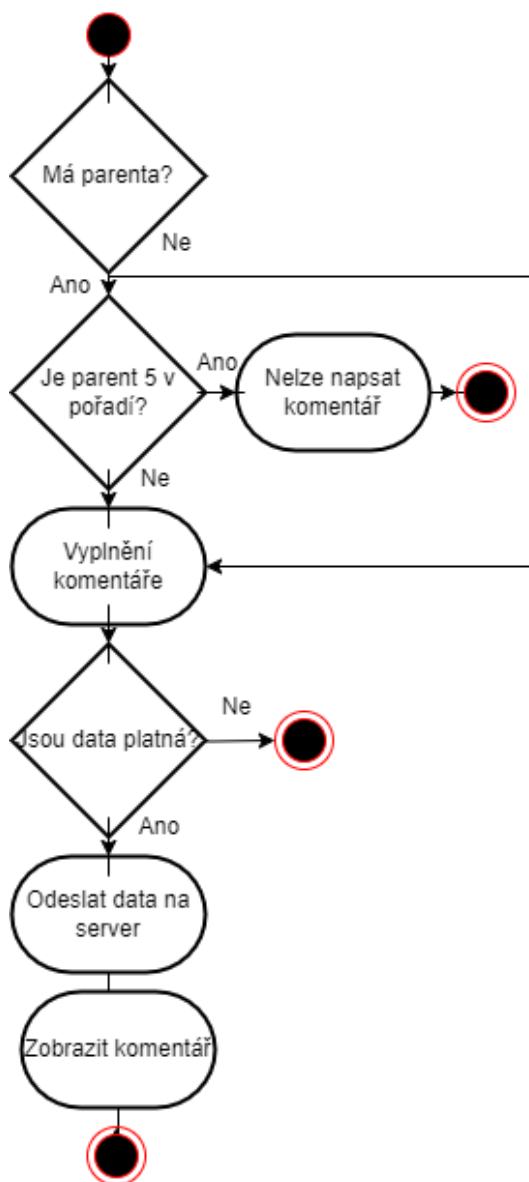
Probíhá kontrola, zda uživatel vyplnil potřebné údaje k vytvoření příspěvku. Pokud zveřejňuje i obrázky, nejprve se uloží na server a až potom se odešlou data a zobrazí příspěvek.



Obrázek 5: Activity diagram přidání příspěvku. Zdroj: autor

Komentáře

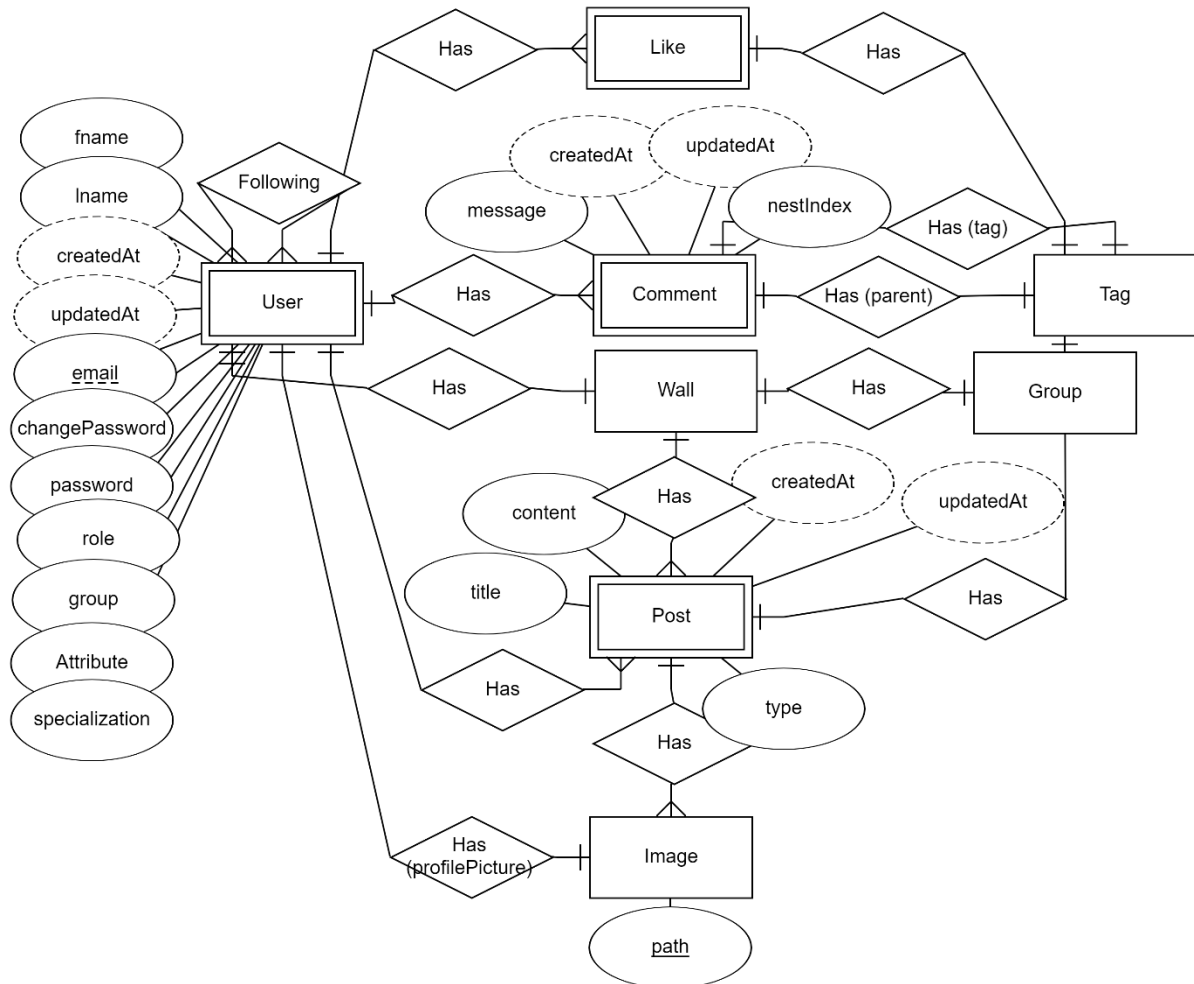
Nejprve program zjišťuje, zda uživatel odpovídá na příspěvek nebo na komentář, popřípadě kolikátý komentář v pořadí. Maximální počet komentářů jdoucích po sobě je pět, tudíž nelze odpovídat na komentář, který by byl šestý v pořadí. Dále probíhá kontrola, zda uživatel vyplnil vše potřebné pro vytvoření komentáře. Pokud ano, odešlou se data a komentář se zobrazí.



Obrázek 6: Activity diagram přidání komentáře, Zdroj: autor

Návrh databáze + Entity-relationship model

Tento digram vypovídá podrobněji o tom, jak databáze funguje. Popisuje všechny tabulky, jejich atributy a relace mezi nimi.



Obrázek 7: Entity-relationship diagram, Zdroj: autor

Výběr technologií

Rozhodl jsem se využít TypeScript a knihovnu React pro vývoj. Pro backend jsem zvolil NextJS a pro propojení frontendu s backendem a také pro jednoduché ladění jsem se rozhodl použít tRPC. Všechny tyto nástroje podporují TypeScript, což mi usnadňuje práci. Pro práci s databází jsem si vybral PrismaORM, která běží v Dockeru. Na stylování jsem se rozhodl použít TailwindCSS. Abych si udržel přehled o vývoji, ukládám projekt do Gitu a používám GitLab pro správu.

TVORBA APLIKACE

Vytvoření aplikace

Nainstaloval jsem vše potřebné ke tvorbě aplikace. Pomocí příkazu `npm create t3-app@latest` jsem nainstaloval základ aplikace, která obsahovala i soubor k automatickému vytvoření a spuštění databáze v Dockeru. Na začátku tvorby tento soubor, avšak nebyl přístupný, přišel až později s novou verzí T3stacku. Ještě před tím, než jsem začal používat Docker běžela databáze na stránce Planetscale. Tato stránka bohužel přestala fungovat na území české republiky, a proto jsem musel vymyslet jiný způsob, což je právě ten Docker. Za tuto nepříjemnost jsem rád, protože mi to přijde jako lepší řešení. Postup nahrávání dat do databáze je stejný v obou případech.

Ještě, než jsem začal něco tvořit, tak jsem si vytvořil nový repozitář na GitLabu. V konzoli jsem spustil příkaz `git init`, který inicializuje lokální Git repozitář. Dále jsem spustil příkaz `git add .`, tímto příkazem se nahrají všechny soubory a změny do lokálního repozitáře. Potom stačí jen spustit příkaz `git commit -m "komentář ke změnám"`, který vytvoří commit se změnami a napíše libovolnou zprávu ke commitu, která popisuje provedené změny. Abych mohl nahrát soubory na GitLab je nutné přidat URL repozitáře pomocí příkazu `git remote add origin URL`. Když už jsou repozitáře propojeny stačí spustit poslední příkaz `git push -u origin master`, který nahraje změny do hlavního repozitáře umístěného na GitLabu.

Takhle si vytvářím různé branchy, mezi kterými mohu přepínat. Pokud udělám něco špatně, nebo přepíšu nějaký kód, který ve výsledku nemá smysl a potřebuji vrátit změny zpět stačí nahrát repozitář a vrátí se mi projekt do stavu, ve kterém jsem ho naposledy ukládal. Užitečný je GitLab i tom, že můžu pracovat na projektu odkudkoliv a kdykoliv, aniž bych s sebou všude musel brát nějaké externí uložště jako USB.

Jakmile jsem přepsal schéma databáze do Prismy stačilo jen spustit příkaz `npx prisma db:push` a schéma se nahrálo do databáze.

Zde můžete vidět, jak vypadá tabulka v Prismě. Tabulka se nazývá model. Model se skládá ze tří sloupců. První obsahuje název atributu, druhý datový typ atributu a třetí obsahuje dodatečné nastavení jako relace mezi tabulkami, unikátnost atributu nebo automatické vyplnění atributu například získáním času.

```

model Comment {
  id          String          @id @default(uuid())
  message     String
  createdAt   DateTime        @default(now())
  updatedAt   DateTime        @updatedAt
  user        User            @relation(fields: [userId], references:
[id], onDelete: Cascade)
  tag         Tag             @relation(fields: [tagId], references: [id],
onDelete: Cascade)
  parent      Tag             @relation("parent", fields: [parentTag],
references: [id], onDelete: Cascade)
  userId      String
  tagId       String          @unique
  parentTag   String
  nestIndex   Int

  @@index([userId])
  @@index([id])
  @@index([parentTag])
}

```


Přihlášení uživatele

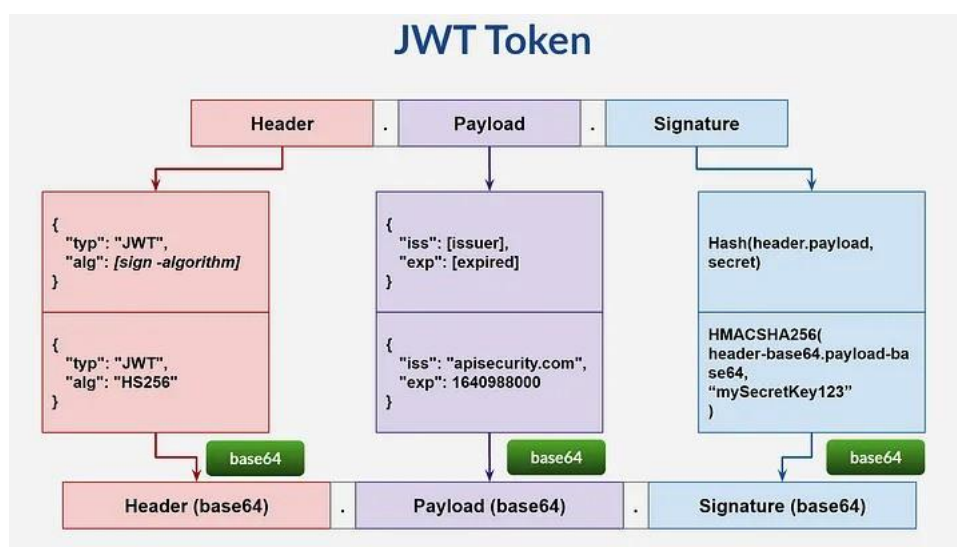
Vzhledem k tomu, že aplikaci budou využívat pouze studenti, zaměstnanci a vybraní lidé, není nutné přidávat registraci. Studentům bude založen účet automaticky po jejich souhlasu. Lidé, kteří nejsou studenty nebo zaměstnanci, budou přidáni na platformu pomocí administrátora. Jakmile uživatel odsouhlasí tvorbu účtu, vygeneruje se náhodné heslo, které bude odesláno na emailovou adresu. Po prvním přihlášení bude uživateli doporučeno změnit heslo na vlastní. Uživatel se tedy přihlašuje pomocí emailové adresy a hesla.

Pro přihlášení uživatele jsem využil knihovny NextAuthJS, která obsahuje vše potřebné pro jednoduché a bezpečné přihlašování a ověřování. NextAuthJS nabízí spoustu přihlašovacích metod, avšak jediná vhodná metoda pro tuto aplikaci je pomocí Credentials (umožňuje provádět přihlašování pomocí libovolných přihlašovacích údajů).

Jakmile se uživatel přihlásí jeho data jsou zabezpečena pomocí JWT.

JWT (JSON Web Token)

Tento token zabezpečuje API. Token je přidán do HTTP požadavku a následně API ověří token proti databázi. Token se ukládá v cookies.



Obrázek 8: Jak funguje JWT token. Zdroj: [3]

JWT se skládá ze tří částí. První z nich je header, který obsahuje informace o tom, jak se má token zpracovat (typ tokenu, algoritmus a další). Jako další je to payload, který obsahuje tzv. nároky, které obsahují užitečné informace, jako například, kdo token vytvořil, pro koho je určen a další. Jako poslední obsahuje signature, který obsahuje zakryptovaná data.

Příspěvky

Vytvořil jsem prototyp příspěvku, jak bude vypadat a jaké informace bude obsahovat. Titulek, podrobnější informace, kdo a kdy příspěvek vytvořil, nějaké obrázky, komentáře, hodnocení. Dále příspěvek obsahuje možnosti, kde najdeme například tlačítko ke smazání příspěvku. K tomuto tlačítku má přístup pouze uživatel, který daný příspěvek vytvořil a administrátoři.



Obrázek 9: Ukázka příspěvku. Zdroj: autor

Vytváření příspěvků

Na vytváření příspěvků byl použit klasický formulář s inputy. Jediné, co potřebuje uživatel vyplnit je titulek. Popis a obrázky jsou volitelné. Po vyplnění se nahrají obrázky na server a odešlou data do databáze. Společně s daty se také odesílá id zdi. Zeď obsahuje příspěvky. Vzhledem k tomu, že některé příspěvky jsou přístupné jen na profilu konkrétního uživatele nebo v konkrétní skupině, musí existovat zeď, která příspěvky bude oddělovat.

Po kliknutí na tlačítko *Odeslat příspěvek* program zkontroluje, zda jsou uživatel splnil všechny podmínky pro odeslání komentáře. Pokud ano, odešlou se data do databáze. Příspěvek je vytvořen a viditelný všem uživatelům, kterým je určen.

Přidávání obrázků

Příspěvky jsou bez obrázků suché, nudné. Proto lze přidat na příspěvek až tři obrázky, které zaručeně ožíví celou sociální síť.

Po kliknutí na přidat obrázek se otevře okno, kde může uživatel vybrat z určitých typů souborů (.jpg, .png, .jpeg). Každý obrázek je omezený, velikost souboru nesmí být větší jak 4 Mb. Po výběru souborů se obrázky objeví v okně, kde uživatel fotky vidí a může je při případné chybě změnit, nebo přidat další fotku. Pokud uživatel vybere více než tři fotky, program vybere první tři fotky z výběru.

Když už, kvůli bezpečnosti, byla uživatelům odebrána práva změnit si jméno a další údaje. Přidal jsem možnost změnit si profilový obrázek přímo na profilu uživatele. Nahrání profilového obrázku funguje na stejný princip jako u příspěvků. Celý komponent na tvorbu a odesílání obrázku jsem naprogramoval tak, aby byl nastavitelný podle potřeb.

Když uživatel smaže příspěvek nebo změní profilový obrázek, je zbytečné nechávat v souborech uložené staré obrázky, protože už nebudou nikdy nikomu dostupné. Tento jednoduchý kód zařizuje mazání přebytných obrázků.

Komentáře

Komentáře jsou obyčejné, jako na každé jiné sociální síti. S možností označení jako *to se mi líbí* a odpovědi. Lze komentovat příspěvky a také lze komentovat komentáře. Takhle se dá napsat až pět odpovědí po sobě, potom je možnost odpovědi odebrána.

Třídní skupiny

Cílem třídních skupin je seskupit studenty dané třídy, aby mohli sdílet příspěvky mezi sebou. Studenti budou mít přístup do téhle skupiny i po tom, co opustí školu a budou tak moct mezi sebou sdílet zážitky i nadále. Fungují na stejný princip jako každá jiná zeď, aplikace si na základě informací, které má zjistí, pod kterou skupinu uživatel spadá. Když studenti výjdou školu, nebudou moct sdílet příspěvky s ostatními uživateli. Budou jim odebrána všechna práva kromě možnosti likování. Budou moct sdílet příspěvky pouze ve své třídní skupině.

Admin panel

Cílem admin panelu je snadné ovládání aplikace, aniž by musel administrátor zasahovat do struktury aplikace jako takové. Tento panel obsahuje funkcionality jako banování uživatelů, přidávání uživatelů, kteří nejsou návštěvníci školy a podobné.

Vyhledávání

Je potřeba se na sociální síti nějakým způsobem pohybovat. Když bude chtít uživatel vyhledat nějakého svého kamaráda, najde ho pomocí vyhledávače. Vyhledávač jednoduše vezme výraz zadaný uživatelem a provede se vyhledávání v databázi. Jména, příjmení a emailové adresy, které obsahují zadaný výraz databáze vrátí zpět na frontend, kde se data zpracují a uživateli se zobrazí výsledky.

Nahrání uživatelů na sociální síť

Dostal jsem přístup do školní databáze. Vytvořil jsem program, který získá potřebná data o studentech a zaměstnancích školy. Tato data se zpracují a dále probíhá vytvoření uživatelů, kteří nejsou součástí sociální sítě. Tuto funkcionalitu může admin kdykoliv spustit a přidat tak nově příchozí studenty a zaměstnance.

Novým uživatelům se odešle email s náhodně vygenerovaným heslem. Heslo jim bude doporučováno změnit, dokud si ho nezmění. Heslo musí obsahovat minimálně šest znaků.

UVEDENÍ SYSTÉMU DO PROVOZU

Aplikace jako taková by postrádala smysl, kdyby nebyla veřejná a nemohl se na ni připojit žádný návštěvník školy. Proto je potřeba spustit aplikaci s databází na virtuálním počítači, který poběží ve škole, kde může zároveň aplikace číst data ze školní databáze.

Nainstaloval jsem package git, screen, nodejs a docker.

Byl mi poskytnut virtuální počítač se systémem Ubuntu s již nastavenou statickou ip adresou. Propojil jsem GitLab s adresářem a naimportoval projekt. Celé spuštění databáze mám v souboru `start-database.sh`, který vytvoří nový container v Dockeru. Pomocí `npm run db:push` jsem naimportoval strukturu databáze ze souboru `schema.prisma`. Pomocí screen jsem spustil příkaz `npm install`, který nainstaluje všechny potřebné package. Následně příkaz `npm run dev`, který spustí aplikaci na portu 3000.

PODNIKATELSKÝ PLÁN

Oprávnění k podnikání

Cílem je prodávat uzavřenou sociální síť. Na internetu lze najít, že se jedná o volnou živnost s označením 56, která zní následovně *“Poskytování software, poradenství v oblasti informačních technologií, zpracování dat, hostingové a související činnosti a webové portály”*

Analýza zákazníků

Produkt je navržen s ohledem na univerzální využití, oslovující širokou paletu uživatelů a skupin.

Jeho atraktivnost pro každého zákazníka spočívá v přívětivém UI/UX, což zahrnuje jednoduchý a přehledný design webové stránky a pohodlnou interakci s aplikací. Zákazníkovi je k dispozici pravidelná správa aplikace a možnost rozšíření funkcí aplikace za případný příplatek.

Vzhledem k tomu, že se jedná o můj první rozsáhlejší projekt, mohou se vyskytnout určité nedokonalosti. Přesto se odlišuji od konkurence v cenové dostupnosti mé aplikace, což může být rozhodujícím faktorem pro zákazníky při porovnání s alternativami na trhu.

Analýza konkurence

Konkurence je na trhu obrovská. Moje malá aplikace se nemá šanci rovnat masivním sociálním sítím jako je například Instagram, což je zároveň největší konkurence. Avšak oproti nim nabízím uzavřenou sociální síť pro určitou skupinu lidí, kteří si mohou aplikaci upravit dle vlastních představ. Výhodou uzavřené sítě je zvýšená úroveň bezpečnosti. Uživatelé, kteří mají přístup k síti, jsou pečlivě vybráni, což snižuje riziko nežádoucích incidentů. Tato uzavřenost také umožňuje rychlé a přesné zjištění příčiny případných problémů či incidentů na sociální síti. Společnosti mohou efektivně spravovat aplikaci a zajistit bezpečné prostředí pro své uživatele.

Analýza dodavatelů

Vzhledem k tomu, že můj produkt je software, nejsou vyžadovány žádné materiály. K vytvoření tohoto programu je potřeba jen počítač a vhodný software na vývoj.

Jako dodavatele elektřiny a plynu jsem si vybral firmu ČEZ. Je to největší a nejspolehlivější firma v republice, co se týče prodeje elektřiny a plynu. Vodu budu čerpat z jediné vodárny v Bruntále od firmy VaK Bruntál. Internet budu brát od firmy Baron Computer, mám s nimi dlouholeté zkušenosti a nabízí 100 Mbit/s za slušnou cenu.

Režie	Společnost	Cena
Elektřina	ČEZ	5,80 Kč/kWh
Voda	VaK Bruntál a.s.	56,78 Kč/m ³
Plyn	ČEZ	15,79 Kč/m ³
Internet	Baron Computer	280 Kč/měsíc
Pronájem kanceláře	Město Bruntál	1680 Kč/měsíc

Tabulka 1: Výběr dodavatelů a cena režie

Swot analýza

SWOT analýza je způsob, kterým firmy zhodnocují faktory ovlivňující firmu samotnou.

Silná stránka mé aplikace je unikátnost. Nikdo na trhu podobnou aplikaci neprodává, avšak existují firmy, které vám podobnou aplikaci naprogramují, pokud si o ni zažádáte. Má aplikace funguje jako šablona, která se pouze upraví dle představ zákazníka a může být uvedena do provozu. Aplikace je naprogramovaná tak, aby byla otevřena dalším aktualizacím. Tudiž lze do aplikace přidat další vymoženosti.

Jako další silnou stránku můžeme zahrnout možnost balíčků a monetizaci. Aplikace by se prodávala v různých balíčcích, kde by každý balíček obsahoval jiné funkce. Záleželo by na výběru zákazníka, co mu přijde nejvhodnější a které funkce by vyžadoval. Podle toho by se odvodila cena balíčku. Některé levnější balíčky, nebo nějaké datově omezené zkušební verze by teoreticky mohly obsahovat reklamy, které by vynášely nějaké drobné peníze.

Co se týče slabých stránek aplikace neobsahuje žádný kontrolér, který by kontroloval nevhodný obsah na síti. Aplikace se spoléhá na adminy. Na druhou stranu, kdyby nějaký uživatel na platformu přidal nějaký nevhodný obsah, je lehce dohledatelný.

Jakožto nadšenec těchto technologií jsem neustále v obraze, co se týče aktualizací a novinek. Aplikace bude dostávat patřičné aktualizace v co nejmenším časovém úseku.

Mezi hrozby aplikace patří zabezpečení samotné. Proto, aby byla aplikace zabezpečená jsem udělal vše, co bylo v mých silách, avšak nejsem expert na kybernetiku a stále se učím. Všechny bezpečnostní díry, kterých jsem si všiml jsem opravil. I přesto si myslím, že aplikace bude obsahovat drobné chyby, co se týče bezpečnosti. Kdyby chtěl nějaký zdatný hacker napadnout tuto platformu, myslím, že by se mu to povedlo.

Reklama a propagace

Cílem reklam bude dostat, co nejvíce společností do podvědomí, že vůbec taková aplikace existuje a že mají možnost si něco takového pořídit. Tento styl sociální sítě není obvyklý, a proto málokoho napadne něco takového vyhledávat.

Webové stránky

Pro zakoupení takového produktu jsou potřeba webové stránky, kde si zákazník bude moci zjistit podrobnější informace a v případě zájmu i zakoupit aplikaci.

Mít dobře nastavený SEO pro zviditelnění webové stránky ve vyhledávačích.

Sociální sítě

Reklamy na sociálních sítích bych nezaváděl, protože sociální sítě navštěvují různé věkové kategorie. Reklama by mohla zaujmout některé lidi, ale málokdo si takovou aplikaci pořídí. Lidé by sice podvědomě věděli, že nějaká taková aplikace existuje, ale nemyslím si, že by se to finančně vyplatilo.

Proto bych na sociálních sítích zvolil taktiku propagace. Založil bych si nový účet na různých sociálních sítích a zveřejňoval bych různý obsah. Populárnější sociální sítě mají skvěle vypracované algoritmy pro zobrazování příspěvků a videí. Lidem, kteří se zajímají o podobný obsah se můj příspěvek nebo video může zobrazit. Taková propagace vyžaduje pouze čas, nikoliv peníze.

Nabízení aplikace

Sledovat trh a zaměřit se na společnosti, které by mohli takovou aplikaci reálně využít. Potom tyto společnosti oslovit s nabídkou.

Finanční analýza

Je potřeba vypočítat, kolik bude produkt stát a následně vypočítat výdělek. Základní kapitál nebude nijak vysoký. K výrobě takového produktu je potřeba převážně počítač a přístup k internetu.

Vyvíjím software, ke kterému nejsou potřeba žádné licence a všechny programy, které využívám jsou zadarmo. Tudíž není potřeba využívat dlouhodobého majetku.

Při zahájení činnosti je nutné pořídit počítač. Pořizovací ceny jsou následující.

položka	Kč
Živnostenské oprávnění	1 000
Počítač	25 000
Celkem	26 000

Tabulka 2: Ostatní vstupní náklady

Průměrná spotřeba počítače je 122 kWh za rok. Do toho se počítá internet i elektřina. 40 hodin vychází na zhruba 5 pracovních dnů, což je týdenní pracovní doba běžného člověka. Ohledně vody jsem počítal s tím, že průměrný člověk by měl vypít 2 litry vody denně. Zaokrouhlil jsem průměrnou cenu 1l vody nahoru, vyšla mi cena 1 Kč za litr vody.

Přímé mzdy	Položka	Sazba Kč/hod	Počet hod	Kč na MJ
	Hrubý příjem	300	40	12 000
	Zákonné odvody	102	40	4 080
Celkem				16 080
Režie	Položka	Sazba Kč/hod	Počet hodin	
	Elektrická energie a internet	0.4	40	16
	Plyn	3	40	120
	Nájemné	11	40	440
	Voda	0,25	40	10
Celkem				586
Celkem				16 666

Tabulka 3: Náklady na produkt (služby), kalkulace prodejní ceny

Při plánovaném objemu produkce 4 za měsíc budou předpokládané měsíční náklady 66 664 Kč.

Cena se může zdát vysoká, avšak je nízká v porovnání s ostatními firmami. 300 Kč na hodinu je na takovou práci nízká mzda. Když se vytváří podobná aplikace, sice zabere méně času, ale pracuje na ní mnohem více lidí a ti si berou i 800 Kč na hodinu. Ve finále bude můj produkt levnější.

Položka	Kč
Náklady na produkt (zakázku)	16 666
Zisk	8 333
Prodejní cena	24 999

Tabulka 4: Kalkulace prodejní ceny

Pro začátek není potřeba vysokého obnosu peněz. Ze začátku bych byl schopný financovat projekt sám.

Položka	Kč
Investiční kapitál	0
Provozní kapitál	799 968
Potřeba kapitálu celkem	799 968

Tabulka 5: Potřeba kapitálu a zdroje financování

Výnosy a zisk za měsíc a za rok. Můj zisk se tvoří čistě z rozdílu tržeb a nákladu. Nemám žádné odpisy, ani nic podobného.

Položka	Kč/měsíc	Kč/rok
Tržby z prodeje	99 996	1 199 952
Náklady na produkt	66 664	799 968
Zisk	33 332	399 984

Tabulka 6: Výnosy z prodeje a kalkulační zisk

Výpočet daně z příjmu a čistého zisku.

Položka	Kč
Zisk	399 984 (399 900)
Daň z příjmů	59 985
Sleva na poplatníka	30 840
Výsledná daň z příjmů	29 145
Čistý zisk	370 839

Tabulka 7: Výpočet daně z příjmu

Shrnutí

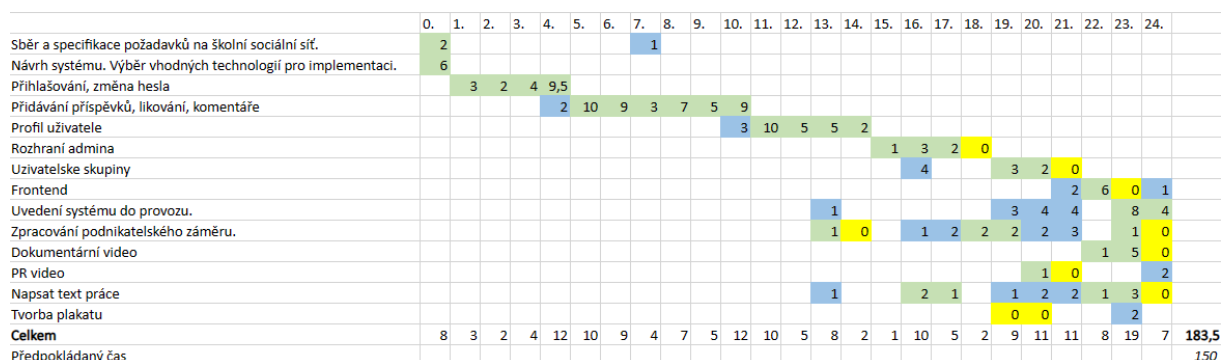
Z výsledku vyplývá celkem vysoký zisk, který může být reálný i nemusí. Záleží na počtu zakázek, které se naskytnou. Někdy může být zákazník náročnější a tvorba zabere více času. Když zákazník bude chtít přidat nějaké vlastní funkce, také to zabere více času, ovšem tyto úpravy budou pro zákazníka za příplatek.

Z toho vyplývá, že nelze momentálně říct, kolik zakázek za měsíc dokážu zpracovat. V tomhle výpočtu jsem počítal s tím, že zákazník vyžaduje minimální úpravy pro aplikaci.

Z celkového výsledku můžeme říct, že bych se mohl touto prací živit a věnovat se jí dlouhodobě. Lze aplikaci stále vylepšovat a zdokonalovat. To může vést k dalšímu rozvoji i navýšení cen.

ZÁVĚR

S tím, že tvorba aplikace bude náročná jsem počítal už od začátku. Toto téma jsem si vzal, protože mám rád programování a chtěl jsem se zdokonalit ve vývoji aplikací pomocí Reactu. Protože to je maturitní projekt, věděl jsem, že se budu muset programování vzít vážně, a tak ve finále získám mnoho zkušeností.



Obrázek 10: Gantův diagram, Zdroj: autor

Z diagramu lze vidět, že vývoj probíhal ze začátku podle plánu, avšak na konci nastalo hned několik problémů, se kterými jsem nepočítal a musel jsem je v posledních týdnech řešit. Jeden z problému bylo, když skončila podpora externí databáze, kterou jsem používal, v české republice. Bez databáze jsem nemohl pokračovat, protože bez ní je aplikace nefunkční. Nakonec jsem tento problém vyřešil a vytvořil lokální databázi přes Docker, což je ve finále lepší řešení.

Zprovoznění aplikace zabralo mnohem více času, než jsem počítal. Aplikace měla původně běžet na Raspberry Pi. Zabralo to několik hodin a musel jsem si projít několika problémy, než jsem došel k závěru, že Raspberry Pi není vhodné pro tuto aplikaci.

Zpracování podnikatelského záměru jsem si rozvrhnout nemohl, protože jsme na něm pracovali v hodinách podnikání.

Dokud nebyl zprovozněný virtuální počítač na síti školy, tak jsem nemohl číst data ze školní databáze. Uživatelské skupiny fungovaly správně, akorát nebylo zautomatizované jejich vytváření.

Nakonec tvorba aplikace splnila svůj cíl. Prošel jsem si celou tvorbou aplikace, od návrhu systému až po samotnou implementaci a spuštění. Vyzkoušel jsem si nové technologie a získal mnoho zkušeností. Už vím, na co si při tvorbě aplikace dávat pozor a dokážu si lépe rozvrhnout čas. Tento projekt mě posunul a udělal ze mě zkušenějšího programátora.

SEZNAM POUŽITÝCH ZDROJŮ

- [1] Lekce 2 - UML - Use Case Diagram. Online. Itnetwork. Dostupné z: <https://www.itnetwork.cz/navrh/uml/uml-use-case-diagram>. [cit. 2024-03-07].
- [2] Lekce 7 - UML - Activity diagram. Online. Itnetwork. Dostupné z: <https://www.itnetwork.cz/navrh/uml/uml-activity-diagram>. [cit. 2024-03-07].
- [3] JWT for Dummies (OK, not 100% dummies). Online. Medium. Dostupné z: <https://medium.com/batc/jwt-for-dummies-ok-not-100-dummies-1f08d3279a0b>. [cit. 2024-03-10].

PŘÍLOHY

Všechny přílohy jsou dostupné v elektronické podobě ve školním archivu maturitních prací.

SEZNAM ZKRATEK A SYMBOLŮ

USB	Universal Serial Bus
JSON	JavaScript Object Notation
JWT	JSON Web Token
API	Application Programming Interface
HTTP	Hypertext Transfer Protocol
UI	User Interface
UX	User Experience
SEO	Search Engine Optimization
LDAP	Lightweight Directory Access Protocol

SEZNAM TABULEK A OBRÁZKŮ

Obrázek 1: Use Case diagram zablokovaného uživatele. Zdroj: autor.....	9
Obrázek 2: Use Case digram klasického uživatele. Zdroj: autor	9
Obrázek 3: Use Case diagram administrátora. Zdroj: autor	10
Obrázek 4: Activity diagram přihlášení uživatele. Zdroj: autor	11
Obrázek 5: Activity diagram přidání příspěvku. Zdroj: autor	12
Obrázek 6: Activity diagram přidání komentáře, Zdroj: autor	13
Obrázek 7: Entity-relationship diagram, Zdroj: autor	14
Obrázek 8: Jak funguje JWT token. Zdroj: [3].....	17
Obrázek 9: Ukázka příspěvku. Zdroj: autor	18
Obrázek 10: Gantův diagram, Zdroj: autor	28

Tabulka 1: Výběr dodavatelů a cena režie.....	23
Tabulka 2: Ostatní vstupní náklady	25
Tabulka 3: Náklady na produkt (služby), kalkulace prodejní ceny	25
Tabulka 4: Kalkulace prodejní ceny	26
Tabulka 5: Potřeba kapitálu a zdroje financování	26
Tabulka 6: Výnosy z prodeje a kalkulace zisku	26
Tabulka 7: Výpočet daně z příjmu	26