



Diplomová práce

Projektová optimalizace procesu vývoje webové aplikace

Studijní program:

N0688A140016 Systémové inženýrství a informatika

Autor práce:

Bc. Hanka Těšitelová

Vedoucí práce:

Mgr. Tereza Semerádová, Ph.D.
Katedra informatiky

Liberec 2024



Zadání diplomové práce

Projektová optimalizace procesu vývoje webové aplikace

Jméno a příjmení:

Bc. Hanka Těšitelová

Osobní číslo:

E22000450

Studijní program:

N0688A140016 Systémové inženýrství a informatika

Zadávací katedra:

Katedra informatiky

Akademický rok:

2023/2024

Zásady pro vypracování:

1. Definice základních pojmů v oblasti řízení procesů a projektů.
2. Přínosy zavedení projektového řízení.
3. Analýza současného stavu procesu vývoje webové aplikace ve vybrané společnosti a vymezení požadavků na změnu.
4. Výběr metod projektového řízení vhodných pro vývojářské IT projekty a jejich implementace.
5. Vyhodnocení a formulace případných doporučení pro budoucí optimalizaci.

Rozsah grafických prací:
Rozsah pracovní zprávy: 65 normostran
Forma zpracování práce: tištěná/elektronická
Jazyk práce: čeština

Seznam odborné literatury:

- DOLEŽAL, Jan, 2022. *Agilní přístupy vývoje produktu a řízení projektu: komplexně, prakticky a dle světové praxe*. Praha: Grada. ISBN 978-80-271-3705-3.
- KŘIVÁNEK, Mirko a Eduard KUNCE, 2019. *Dynamické vedení a řízení projektů: systémovým myšlením k úspěšným projektům*. Praha: Grada. ISBN 978-80-271-0408-6.
- OZKÄN, Deniz a Alok MISHRA, 2019. Agile Project Management Tools: A Brief Comparative View. *Cybernetics and Information Technologies*, vol. 19, no. 4, s. 17-25. ISSN 1314-4081.
- ŠOCHOVÁ, Zuzana a Eduard KUNCE, 2019. *Agilní metody řízení projektů*. Brno: Computer Press. ISBN 978-80-251-4961-4.
- WYSOCKI, Robert, 2014. *Effective project management: traditional, agile, extreme*. Indiana: Wiley. ISBN 978-1-118-72916-8.

Konzultant: Bc. Filip Svárovský – CEO & Marketing specialist, TGNCA s.r.o.

Vedoucí práce: Mgr. Tereza Semerádová, Ph.D.
Katedra informatiky

Datum zadání práce: 1. listopadu 2023
Předpokládaný termín odevzdání: 31. srpna 2025

L.S.

doc. Ing. Aleš Kocourek, Ph.D.
děkan

doc. Ing. Klára Antlová, Ph.D.
garant studijního programu

V Liberci dne 1. listopadu 2023

Prohlášení

Prohlašuji, že svou diplomovou práci jsem vypracovala samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Jsem si vědoma toho, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědoma povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má diplomová práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědoma následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

Projektová optimalizace procesu vývoje webové aplikace

Anotace

Diplomová práce se věnuje projektové optimalizaci v rámci vývoje webové aplikace. Cílem této práce je provést analýzu současného procesu vývoje webové aplikace ve vybrané společnosti a na základě této analýzy navrhnout a implementovat postup řízení projektů pro jejich optimalizaci. Teoretická část diplomové práce se zabývá obecným pojetím projektového managementu a jeho procesů, včetně definice projektu a projektového týmu. Dále se teoretická část zaměřuje na specifika projektového managementu v oblasti vývoje softwaru, vysvětluje charakteristiky softwarových projektů a životní cyklus vývoje softwaru. V závěru teoretické části jsou uvedeny různé přístupy k vývoji softwaru spolu s přehledem vybraných digitálních nástrojů pro projektové řízení. V úvodu praktické části je představena vybraná společnost, pro kterou je provedena analýza současného vývoje. Na základě identifikovaných nedostatků vývoje a požadavků ze strany společnosti je navržena a zároveň na vybraném projektu implementována projektová optimalizace procesu vývoje webové aplikace. V závěru praktické části je provedeno zhodnocení implementovaných změn, jejich dopadů na vývoj a navržena případná doporučení pro budoucí projekty.

Klíčová slova

Agilní přístup, projekt, projektové řízení, projektový management, projektový tým, SCRUM, software, Software Development Life Cycle, tradiční přístup, vodopádový přístup, vývoj softwaru

Project optimization of the web application development process

Annotation

The Master's thesis focuses on project optimization in the context of web application development. The aim of this thesis is to analyze the current process of web application development in a chosen company and, based on this analysis, to formulate and implement a project management solution for project optimization. The theoretical part of the thesis deals with the general concept of project management and its processes, including the definition of a project and a project team. Furthermore, the theoretical part focuses on the specifics of project management in software development, explaining the characteristics of software projects and the software development life cycle. At the end of the theoretical part, different approaches to software development are presented along with an overview of selected digital tools for project management. In the introduction of the practical part, a chosen company is introduced for which an analysis of current software development is made. Based on the identified development weaknesses and the requirements from the company, a project optimization of the web application development process is proposed and also implemented on the selected project. At the end of the practical part, an evaluation of the implemented changes, their impact on the development and possible recommendations for future projects are made.

Key Words

Agile approach, project, project management, project team, SCRUM, software, software development, Software Development Life Cycle, traditional approach, waterfall approach

Poděkování

Ráda bych touto cestou vyjádřila upřímné poděkování vedoucí diplomové práce Mgr. Tereze Semerádové, Ph.D. za její vstřícný přístup. Její podpora, cenná zpětná vazba a pomoc mi významně pomohly k úspěšnému dokončení této diplomové práce.

Dále bych ráda poděkovala společnosti TGNCA s.r.o. za projevenou důvěru, podporu a volnost, kterou mi poskytla při volbě a zpracování této diplomové práce.

Obsah

| | |
|--|-----------|
| Seznam ilustrací..... | 13 |
| Seznam použitých zkratk, značek a symbolů | 14 |
| Úvod..... | 15 |
| 1 Projektový management..... | 17 |
| 1.1 Projekt..... | 19 |
| 1.1.1 Trojimperativ projektu..... | 21 |
| 1.2 Procesy projektového managementu | 23 |
| 1.2.1 Procesy zahájení..... | 24 |
| 1.2.2 Procesy plánování | 24 |
| 1.2.3 Procesy realizace projektu | 25 |
| 1.2.4 Procesy monitorování a kontroly | 25 |
| 1.2.5 Procesy ukončení..... | 25 |
| 1.3 Projektový tým..... | 25 |
| 1.3.1 Týmové role a odpovědnosti..... | 28 |
| 1.3.2 Projektový tým v organizaci | 29 |
| 1.4 Projektový manažer..... | 33 |
| 2 Specifikace projektů v oblasti vývoje softwaru..... | 36 |
| 2.1 Projektový management vývoje softwaru | 37 |
| 2.2 Softwarový projekt..... | 37 |
| 2.3 Software Development Life Cycle (SDLC)..... | 39 |
| 2.3.1 Formulace nápadů či problémů..... | 40 |
| 2.3.2 Plánování..... | 40 |
| 2.3.3 Analýza požadavků | 41 |
| 2.3.4 Návrh softwaru | 41 |
| 2.3.5 Implementace | 41 |
| 2.3.6 Testování..... | 42 |
| 2.3.7 Nasazení..... | 42 |
| 2.3.8 Údržba | 42 |
| 3 Přístupy k vývoji softwaru | 43 |
| 3.1 Tradiční vodopádový přístup | 43 |
| 3.2 Agilní přístup | 44 |
| 3.2.1 Scrum..... | 46 |

| | |
|---|-----------|
| 3.2.2 Lean Software Development | 49 |
| 3.2.3 Kanban | 50 |
| 3.2.4 Extreme Programming (XP) | 51 |
| 3.3 Vybrané digitální nástroje projektového řízení | 52 |
| 3.3.1 JIRA..... | 52 |
| 3.3.2 Trello..... | 52 |
| 3.3.3 Asana..... | 53 |
| 3.3.4 Freeloo..... | 53 |
| 4 Představení vybrané společnosti | 54 |
| 4.1 Produkt..... | 56 |
| 4.1.1 Uživatelé | 57 |
| 4.1.2 Vývoj produktu | 57 |
| 5 Analýza současného vývoje..... | 59 |
| 5.1 Analýza vývoje v kontextu startupového prostředí..... | 60 |
| 5.1.1 Silné stránky | 61 |
| 5.1.2 Slabé stránky | 62 |
| 5.1.3 Příležitosti | 63 |
| 5.1.4 Hrozby..... | 64 |
| 5.2 Požadavky na zlepšení | 65 |
| 6 Návrh optimalizace procesu vývoje..... | 66 |
| 6.1 Výběr projektu a návrh jeho řízení | 67 |
| 6.2 Implementace návrhu | 71 |
| 7 Vyhodnocení a případná doporučení..... | 77 |
| 7.1 Chybějící metodika..... | 77 |
| 7.2 Nedostatečná zadání..... | 79 |
| 7.3 Přenos informací..... | 80 |
| 7.4 Pomalý vývoj..... | 81 |
| 7.5 Důležitost vytvářených funkcí a rozšíření | 82 |
| 7.6 Ekonomické zhodnocení návrhu..... | 82 |
| Závěr | 84 |
| Seznam použité literatury..... | 86 |

Seznam ilustrací

| | |
|--|----|
| Obrázek 1: Trojimperativ projektu..... | 21 |
| Obrázek 2: Skupiny procesů | 24 |
| Obrázek 3: Hierarchická struktura projektu | 31 |
| Obrázek 4: Projektová organizační struktura..... | 32 |
| Obrázek 5: Maticová organizační struktura..... | 33 |
| Obrázek 6: SDLC | 39 |
| Obrázek 7: Vodopádový přístup | 43 |
| Obrázek 8: Agilní přístup..... | 46 |
| Obrázek 9: Kanban | 50 |
| Obrázek 10: Logotyp společnosti | 54 |
| Obrázek 11: Organizační struktura firmy | 54 |
| Obrázek 12: Vzhled aplikace | 56 |
| Obrázek 13: SWOT analýza | 61 |
| Obrázek 14: Návrh průběhu projektu | 70 |
| Obrázek 15: Příprava Product Backlogu v Asaně..... | 72 |
| Obrázek 16: Náhled na Sprint Backlog..... | 74 |
| Obrázek 17: Průběh Sprintu..... | 75 |

Seznam použitých zkratk, značek a symbolů

| | |
|--------|---|
| b.d. | bez data |
| CEO | Výkonný ředitel (<i>Chief Executive Officer</i>) |
| COO | Provozní ředitel (<i>Chief Operating Officer</i>) |
| CSS | Kaskádové styly (<i>Cascading Style Sheets</i>) |
| CSS | Služby porovnávání zboží (<i>Comparison Shopping Services</i>) |
| CTO | Technologický ředitel (<i>Chief Technology Officer</i>) |
| č. | číslo |
| HTML | Hypertextový značkovací jazyk (<i>Hypertext Markup Language</i>) |
| IT | Informační technologie (<i>Information Technology</i>) |
| PHP | Hypertextový preprocesor (<i>Hypertext Preprocessor</i>) |
| SDLC | Životní cyklus vývoje softwaru (<i>Software Development Life Cycle</i>) |
| s.r.o. | Společnost s ručením omezeným |
| XP | Extrémní programování (<i>Extreme Programming</i>) |

Úvod

Efektivní řízení projektů v oblasti vývoje softwaru je klíčovým prvkem pro úspěch společnosti, zejména pak v rychle se rozvíjícím startupovém prostředí. Stále se měnící požadavky zákazníků, neustálý tlak ze strany konkurence a dynamické technologické prostředí představují výzvy, kterým společnosti zaměřené na vývoj softwaru musí převážně čelit. Ze strany společnosti je vyžadován flexibilní přístup ke změnám a schopnost na veškeré situace rychle reagovat. Pro dosažení rychlého a kvalitního vývoje softwaru je nezbytné minimalizovat chyby a zpoždění v procesu vývoje. Správné projektové řízení umožňuje společností lépe organizovat a řídit projekty vývoje, což vede k optimalizaci procesů ve vývoji softwaru, umožňuje rychlejší reakce na změny v požadavcích zákazníků a flexibilní adaptaci na proměnlivé okolní situace.

Pro diplomovou práci je stanoven hlavní cíl provést analýzu současného procesu vývoje webové aplikace ve vybrané společnosti a na základě identifikovaných slabých míst ve vývoji a požadavků ze strany společnosti navrhnout projektovou optimalizaci procesu vývoje pro minimalizaci či úplné odstranění jeho nedostatků. Součástí diplomové práce je v této vybrané společnosti návrh zároveň implementovat.

Diplomová práce zahrnuje teoretickou a praktickou část. Teoretická část je věnována vysvětlení projektového managementu a jeho procesů, definici projektu a projektovému týmu. Dále je v teoretické části blíže specifikován projektový management v oblasti vývoje softwaru, jaké charakteristiky má softwarový projekt a jakými fázemi prochází vývoj softwaru v rámci jeho životního cyklu. Čtenář je rovněž seznámen s různými přístupy k projektovému řízení ve vývoji softwaru a s digitálními nástroji, které mohou řízení projektů usnadnit.

V praktické části je představena konkrétní společnost, která se zabývá vývojem webové aplikace v oblasti marketingu. Pro tuto společnost je provedena analýza stávajícího procesu vývoje webové aplikace, ve které jsou posouzeny silné a slabé stránky vývoje, včetně příležitostí a hrozeb. Na základě provedené analýzy, identifikovaných nedostatků a stanovených požadavků společnosti je vypracován návrh na projektovou optimalizaci procesu vývoje webové aplikace s cílem

minimalizovat či úplně odstranit tyto slabé stránky. Návrh je poté v dané společnosti implementován a v závěru práce je implementovaný návrh pro projektovou optimalizaci vyhodnocen.

1 Projektový management

První zmínky o projektovém řízení vnímaném jako součást managementu se objevují po skončení druhé světové války. Avšak mnohé historické události poukazují na fakt, že v minulosti, ještě před druhou světovou válkou, byly realizovány akce, jež měly charakter projektů. Příkladem mohou být stavby historických památek nebo dávné vojenské operace, které vedly k vývoji a rozvoji různých postupů, metod a technik pro řešení takto výjimečných, velmi náročných a komplexních akcí (Doležal, 2023).

V porovnání s dnešní dobou se vyskytuje mnoho rozdílů. V minulosti bylo tempo akcí výrazně pomalejší, což může být uvedeno na příkladu komunikace. Dříve takové doručení zprávy bylo závislé na tom, jak, kdy a kým bude na druhou stranu doručena. Formulace odpovědi a následné zaslání zpět zabralo další čas a prodloužilo tak celý proces komunikace. Navíc zde chyběla záruka toho, že bude zpráva v pořádku předána jak jedné, tak druhé straně. V současné době, díky globalizaci a technologickým inovacím zabere celý proces komunikace jen pár vteřin a týmy z celého světa mohou pracovat společně v reálném čase (Doležal, 2023).

Je tedy zřejmé, že postupem času se podoba projektového managementu měnila a měnit se bude i nadále, jelikož je zasazen do dynamického prostředí, kterému se musí přizpůsobovat. Základ projektového managementu však zůstane vždy stejný. To, co ovlivnilo podobu v posledních letech nejvíce, byly změny týkající se pokroků v oblasti moderních technologií, světové ekonomiky a globalizace. V dnešní urychlené době důležitost projektového řízení a projektů roste, jelikož okolní prostředí se neustále vyvíjí, mění a organizace musí umět na tyto změny okamžitě reagovat. Z toho důvodu je pro společnost a její úspěch klíčové mít v rámci organizace začleněn projektový management (Svozilová, 2016).

Projektový management uplatňuje znalosti, dovednosti, různé nástroje a techniky k řízení projektových činností, mezi které se řadí například procesy plánování, přípravy a realizace, následného monitorování a dokončení projektu, s cílem dosáhnout stanovených cílů a požadavků projektů (Wysocki, 2013).

Stále se zvyšuje počet společností, které implementují projektový management mezi své vnitřní procesy. Není totiž nutností využívat pravidla projektového managementu

pouze pro určité projekty, u kterých je to potřebné. U některých ustálených procesů a činností lze principy projektového managementu také uplatit, ovšem je důležité dodržovat jeho pravidla a metody a také umět rozlišit projektové řízení od řízení operativního. Prvním zásadním rozdílem je délka trvání. Projekt končí v době, kdy jsou splněny jeho stanovené cíle. V případě operativního řízení se nastaví cíle nové a činnost pokračuje. Dalším rozdílem je pohlížení obou stran na zdroje. Pro projekt jsou dostupné zdroje plánovány a jsou mu na dobu jeho trvání přiřazeny. Následně jsou po dokončení projektu tyto prostředky buď plně využity a spotřebovány, nebo jsou přesunuty do dalších projektů. Pracovní síly spadající pod operativní řízení jsou nepřetržitě plánovány a v případě potřeby doplňovány. Pokud nastane situace, že tyto zdroje nejsou nadále potřebné, uvolní se pro jiné účely, u kterých však není zaručeno jejich efektivní využití (Svozilová, 2016).

Uplatnění projektového managementu a projektů ve společnosti s sebou nese několik výhod. Projekt je jasně vymezen a s ním i plánovaný časový a finanční rozsah provedení. Pro každou aktivitu v rámci projektu jsou přiděleny určité role a zodpovědnosti, které jsou nezávislé na možné personální změny v projektovém týmu. Zdroje potřebné pro realizaci projektu jsou danému projektu vymezeny po dobu jeho trvání. Následně, po jeho dokončení, jsou buď předány jiným projektům, nebo spotřebovány. To zajistí přizpůsobení využití zdrojů pro konkrétní situaci. Je možné pečlivě kontrolovat, jak se projekt vyvíjí ve srovnání s tím, co bylo plánováno. V průběhu projektu lze sledovat různá odchýlení od původního záměru a provést účinné úpravy, aby bylo dosaženo stanovených cílů (Svozilová, 2016).

Projektové řízení má také své stinné stránky. Je nutné nahlížet na často nečekané situace, jejichž zdárné řešení má v rukou projektový manažer. Může se stát, že nastanou změny v technologiích, jež mohou projekt ovlivnit. Také zákazník, kvůli kterému je projekt realizován, může mít v průběhu trvání projektu nové konkrétní požadavky. V rámci firmy pak mohou nastat změny v organizační složce. Dále velkou roli při realizaci projektu hrají možná rizika projektu a těžce předvídatelné vlivy, jež přichází z vnějšího prostředí. Jak se v těchto situacích projektový manažer zachová závisí především na jeho zkušenostech, schopnostech a připravenosti (Svozilová, 2016).

1.1 Projekt

„A project is a sequence of unique, complex, and connected activities that have one goal or purpose and that must be completed by a specific time, within budget, and according to specification.“ Takto uvádí ve své knize definici projektu Wysocki (2013, s. 4). V překladu: *„Projekt je posloupnost jedinečných, komplexních a vzájemně propojených činností, které mají jeden cíl nebo účel a které musí být dokončeny v určitém čase, v rámci rozpočtu a podle specifikací.“*

Spíše, než rozhodnutí managementu jsou to právě technické požadavky, co rozhoduje o pořadí provedení jednotlivých činností. Je vhodné posuzovat pořadí z hlediska vstupů a výstupů jednotlivých činností, tedy že výstup jedné činnosti nebo činností bude následujícím vstupem další činnosti nebo činností. S posloupností jednotlivých činností souvisí také jejich propojenost, tedy že mezi nimi může být pozorován určitý vztah. Pokud je vstup jedné aktivity výstupem druhé, pak mluvíme o aktivitách spojených, které jsou na sobě závislé. Není vhodné tvořit posloupnost činností na základě tzv. uměle vytvořeného vztahu mezi aktivitami, což například znamená, že subjekt začne vykonávat danou činnost poté, co dokončí činnost předchozí. Zde může nastat problém v případě, pokud daný subjekt nebude k dispozici. O tom, jaké zdroje v projektu použít a kdy je použít se rozhoduje v procesu plánování projektu. (Wysocki, 2013).

Abychom mohli mluvit o projektu, činnosti jej tvořící musí být jedinečné. A to i v případě, že se činnosti v projektu opakují. Změny v činnostech mají obvykle charakter náhodných událostí, jako například opoždění objednané součástky, náhlá nemoc osoby z projektového týmu nebo výpadek, jež zabrání pokračování činnosti. Takové události jsou zejména v rukou projektového manažera, který jim musí čelit a zároveň podporují jedinečný charakter projektu (Wysocki, 2013).

Sekání trávy nebo umytí auta nejsou činnosti, ze kterých by byl složen projekt. Pro projekty jsou typické složité a komplexní činnosti, jako například návrh podoby uživatelského rozhraní v aplikaci. Všechny činnosti dohromady jsou vykonávány, aby bylo dosaženo stanoveného cíle nebo účelu. Pokud je projekt příliš velký a složitý, může být rozdělen do menších podprojektů, které se řídí lépe a snadněji. Ovšem se musí dávat pozor na závislost mezi jednotlivými podprojekty (Wysocki, 2013).

Další charakteristikou projektu je jeho omezená doba trvání. Má svůj začátek i stanovený konec. Termín dokončení může být určen jak společností zastřešující projekt nebo klientem, pro který je projekt vykonáván. Projekt má také své omezené zdroje, které mu jsou přiřazeny. Jedná se jak o finanční, tak i personální a materiální zdroje. Omezen může být také z hlediska legislativy (Wysocki, 2013; Doležal, 2023).

Aby projekt mohl vzniknout, musí být specifikován. Subjekt, pro kterého je projekt realizován, na začátku určuje, co od projektu očekává a jaká změna současného stavu má být uskutečněna. Projektový manažer pak může stanovit zdroje nebo termín dokončení. V ideálním světě by byla specifikace projektu po celou dobu trvání pevně daná a neměnná. Bohužel, velmi často nastávají situace, kdy jsou v průběhu trvání specifikace doplňovány, upřesňovány či měněny (Wysocki, 2013; Křivánek, 2019).

Jedním z dalších klíčových kritérií projektu je sestavení vhodného projektového týmu. Je nutné si uvědomit, že v rámci projektu je potřeba více než schopnosti pouze jednoho člověka. Pro dosažení úspěchu projektu je nezbytné sestavit tým složený z jedinců s různorodými dovednostmi, z různých oborů a specializací, kteří budou efektivně spolupracovat na dosažení společného cíle (Doležal, 2023; Fielding, 2020).

Pro projekt je taktéž charakteristické, že je rizikový. Na základě předchozích kritérií projektu, tedy že se jedná o unikátní soubor komplexních činností, na kterém pracuje projektový tým složený z mnoha různých osob a který je vymezen zdroji, termíny i rozpočtem, je tato vlastnost zcela předvídatelná (Doležal, 2023).

Projekt může vzniknout z určité činnosti i neplánovaně, ať už si vedení organizace nebo jiný zodpovědný subjekt toto vzniknutí uvědomuje nebo ne. Klíčové je zjištění, že daná činnost má charakteristiky projektu, a následné využití metod, technik, zdrojů a týmu pro úspěšné dokončení. Důležité je, aby osoba s rozhodovací pravomocí podnikla správné kroky k tomu, aby se činnost začala řídit jako projekt. Ignorování tohoto faktu může vést k chaosu, zmatku a problémům v průběhu realizace projektu (Fielding, 2020).

Mezi běžné příklady projektů lze uvést například události jako:

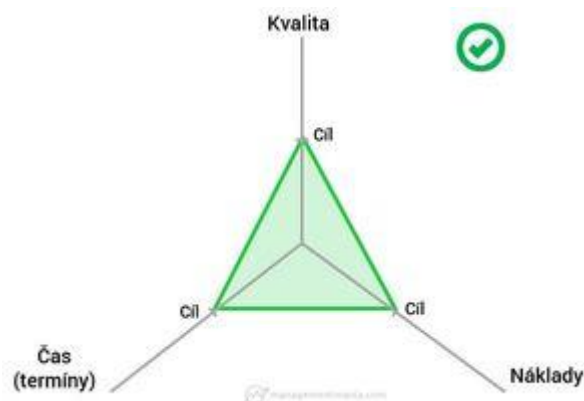
- zavedení nového výrobního procesu,
- vytvoření a implementace informačního systému,

- vytvoření nového produktu,
- vylepšení stálého produktu,
- organizace společenských akcí (festivally, turné kapely atd.),
- výprava za účelem výzkumu,
- stavební práce (Doležal, 2023).

Jak lze vidět, projekty jsou rozmanité, jedinečné a každý z nich má svá určitá specifika. Pokud by na ně byly použity konkrétní návody a instrukce, nefungovalo by to. Proto se v rámci projektového řízení využívají obecně platné principy a standardy k řešení různých problémů v kontextu projektů (Doležal, 2023).

1.1.1 Trojimperativ projektu

Trojimperativ projektu je klíčovým konceptem v projektovém řízení. Definuje tři základní omezení jako vrcholy trojúhelníku, které jsou navzájem provázané a ovlivňované. Pro to, aby byl projekt úspěšný, je důležité na tato omezení striktně nahlížet a dodržovat jejich rovnováhu. Tato kritéria nemusí být vztahována pouze v pohledu na celý projekt, trojimperativ lze využít i na jednotlivé činnosti či fáze projektu. Trojimperativ projektu je znázorněn na Obrázek 1 (Doležal, 2023).



Obrázek 1: Trojimperativ projektu
Zdroj: ManagementMania, 2019

V různých literaturách se lze setkat s rozdílnými vrcholy trojúhelníku. Ve většině případů se vrcholy zaměřené na čas a náklady neliší, ale liší se vrchol třetí, který může být znázorněn například jako výsledky nebo rozsah projektu. V této diplomové práci bude popsán trojúhelník s vrcholy času, nákladů a kvality. Obsahem trojúhelníku je pak výstup projektu.

Provázanost vrcholů trojúhelníku a jejich vzájemné ovlivňování lze vysvětlit na následujícím příkladě. Pokud by bylo z nějakého důvodu rozhodnuto, že se čas dodání projektu zkrátí s nezměněným rozsahem (výstupem), bude muset být provedena změna po ose s náklady nebo kvalitou, aby se prostředí dostalo zpět do rovnováhy. Bude tedy eventuálně změněna oblast nákladů, která se navýší (např. pracovníci), nebo se sníží kvalita výstupu. Takový případ nastává zejména v IT oblasti, ve které se upřednostňuje nahlížet spíše na čas než na kvalitu výsledku. A nejen oblast IT, ale i jiné obory musí s trojúhelníkem neustále pracovat, jelikož v průběhu projektu nastávají nečekané změny, které je nutné řešit (Doležal, 2023).

U projektů se lze setkat se dvěma druhy kvality. Prvním druhem je kvalita produktu, která pojednává o tom, jak kvalitní je výsledek či výstup projektu. Produktem zde může být například hardware či software. Aby bylo dosaženo požadované kvality, jsou využívány běžné nástroje kontroly kvality. Druhým typem je kvalita procesu, jež pojednává o kvalitě řízení projektu. Zabývá se posuzováním a neustálým zlepšováním procesu řízení projektů tak, aby bylo dosaženo stanovených cílů a výsledků (Wysocki, 2013).

Rozpočet ohraničuje projekt z hlediska jeho nákladů, které jsou potřebné pro realizaci projektu. První diskuze nad náklady projektu probíhá v brzkém začátku, kdy se začíná o projektu teprve hovořit. Pokud je projekt realizován pro externího klienta, může klient navrhnout výši rozpočtu projektovému týmu, který s touto nabídkou nadále pracuje a snaží se výši nákladů projektu přizpůsobit podmínkám klienta. V běžných případech připraví projektový manažer podrobný plán nákladů projektu, který s klientem následně diskutuje a v případě žádostí klienta upravuje. Za náklady projektu se považuje veškeré financování zdrojů, které jsou potřebné k úspěšnému dokončení projektu (Wysocki, 2013).

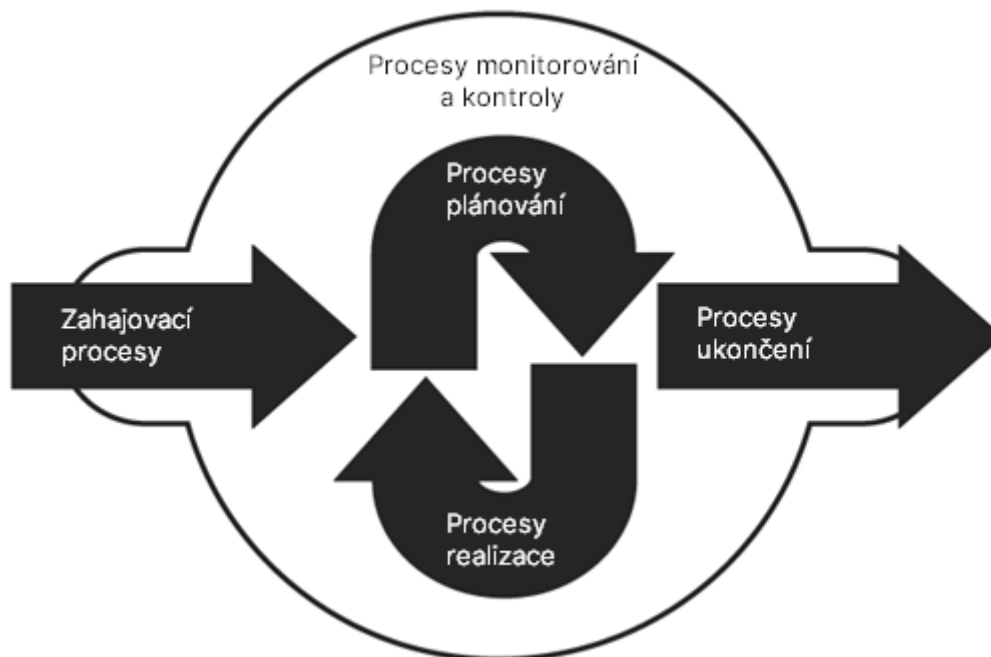
Jak již bylo zmíněno v této diplomové práci výše, projekt je časově vymezen. Stanovení doby trvání projektu závisí na požadavku klienta nebo jiného subjektu, který realizaci projektu požaduje. Projektový manažer by si měl uvědomit, že čas je zdroj, který je spotřebováván bez ohledu na jeho využití a proto by ho měl správně a efektivně řídit pro dosažení největší produktivity (Wysocki, 2013).

Obsah projektového trojúhelníku tvoří rozsah projektu, který formuluje, jaké činnosti budou prováděny a jaké budou výsledky či výstupy projektu. Pokud se rozsah změní, ovlivní to ostatní vrcholy trojúhelníku. Projektový manažer na základě těchto změn provádí takové kroky, aby bylo možné v realizaci projektu nadále pokračovat a úspěšně ho dokončit (Wysocki, 2013).

Projekt je považován za úspěšný, pokud je dokončen ve vymezeném časovém termínu, s definovaným rozpočtem a v požadované kvalitě. Další znaky úspěšného projektu závisí na oblasti a prostředí, ve kterých je projekt realizován. Mohou jimi být například celková spokojenost všech zapojených stran, minimum výskytu nepříznivých situací, jež by žádaly lehké výchyly a projekt tak ovlivnily či bez negativních vlivů všech zúčastněných i okolí. Úspěšný projektový management je takový, který vede k úspěšnému projektu a k dosažení předem stanovených cílů za stanovených podmínek (Křivánek, 2019; Svozilová, 2016).

1.2 Procesy projektového managementu

V rámci řízení projektu probíhá od počátku projektu do jeho ukončení mnoho vzájemně propojených procesů. Tyto procesy jsou organizovány do pěti skupin, jež slouží k uspořádání různých činností vykonávaných v průběhu projektu, aby bylo dosaženo stanovených cílů. Skupiny procesů nejsou na fázích projektu závislé. V průběhu jednotlivých fází projektu mohou být využity různé procesy a zároveň se mnoho z nich může opakovat. Skupiny procesů jsou znázorněny na Obrázek 2 a budou podrobněji rozebrány v následujících podkapitolách (Igberaese, 2022; PMI, 2022).



Obrázek 2: Skupiny procesů
Zdroj: upraveno dle flylib.com, 2017

1.2.1 Procesy zahájení

Procesy týkající se zahájení nového projektu či jeho konkrétní fáze řeší otázky definice a schvalování. V tomto momentě ještě není zahájena žádná projektová práce. Mezi zahajovací procesy může patřit například určení jednotlivců nebo organizace, kteří budou jakýmkoliv způsobem do projektu zasahovat či popis projektu na základě žádostí a potřeb. Výsledkem skupiny procesů zahájení je povolení, nebo také zamítnutí nového projektu či další fáze projektu probíhajícího (PMI, 2022; Wysocki, 2013).

1.2.2 Procesy plánování

V případě schválení nového projektu či úspěšného postoupení do další fáze nastává příprava projektu nebo fáze. Procesy zabývající se plánováním se zaměřují na určení rozsahu, včetně časové a nákladové složky, dále se zaměřují na detailní specifikaci cílů a stanovení posloupnosti činností vedoucích k úspěšnému dokončení projektu nebo fáze. Výsledkem skupiny procesů plánování je konkrétní plán obsahující všechna důležitá kritéria pro efektivní realizaci. (PMI, 2022; Veber, 2021).

1.2.3 Procesy realizace projektu

Po důkladném naplánování přichází zahájení realizace projektu, kdy se věci začínají dávat do pohybu. Jsou realizovány procesy, které vedou k úspěšnému dokončení projektu nebo jeho konkrétní fáze. Průběh projektu je řízen projektovým manažerem, který dohlíží na plnění stanoveného plánu a jeho kritérií, koordinuje projektový tým při provádění činností, kontroluje veškeré procesy a v případě vyskytlého problému efektivně zasahuje. Veškeré procesy a případné změny v průběhu realizace projektu nebo fáze je nutné přesně a srozumitelně komunikovat a zajistit, aby veškeré zapojené strany byly dostatečně informovány o vývoji (Takch, 2020; Veber, 2021).

1.2.4 Procesy monitorování a kontroly

Veškeré aktivity a činnosti provedené od počátku do konce projektu jsou pečlivě sledovány a kontrolovány. Monitorovací procesy zajišťují neustálý dohled nad průběhem a sledují možné odchylky od původního plánu na základě pravidelného měření. Kontrolní procesy jsou klíčové pro identifikaci potencionálních problémů a změn, které mohou mít na projekt negativní dopad (Igberaese, 2022).

1.2.5 Procesy ukončení

Závěrečná skupina obsahuje procesy, které jsou zaměřené na formální uzavření projektu či konkrétní fáze projektu po dokončení všech činností a výstupů a jejich následném předání zadavateli projektu. Zahájením ukončovacích procesů je následně ověřeno, že došlo k úplnému dokončení ostatních procesů v průběhu řízení projektu (Igberaese, 2022).

1.3 Projektový tým

Projekt je jedinečný už tím, kdo na jeho realizaci pracuje. Pro každý jednotlivý projekt je vytvořen projektový tým, který je dočasný. Jeho existence je omezena délkou trvání projektu. Po dokončení projektu se tým rozpadne a pro případné nové projekty je sestaven tým nový. Projektový tým je samostatně řízená složka v organizaci, která

je složena ze skupiny lidí vybraných pro realizaci projektu. Jednotlivci v týmu mají různé dovednosti, znalosti a mohou pocházet z různých organizací i odvětví, což dělá projektový tým rozmanitý. Každý z členů týmu přináší svůj vlastní jedinečný přínos a jako celek vzájemně spolupracují na dosažení společného cíle, kterým je úspěšné dokončení projektu (Chiocchio et al., 2015).

Vytvoření projektového týmu a jeho zasazení do struktury organizace probíhá již v plánovací fázi projektu. Za projektový tým a jeho řízení je zodpovědný projektový manažer. Výběr jednotlivců, kteří budou na projektu pracovat a fáze vzniku projektového týmu je velmi důležitým aspektem pro jeho fungování. Klíčové pro úspěch projektu je sestavit takový tým lidí, ve kterém se budou členové vzájemně doplňovat svými znalostmi a schopnostmi a ve kterém budou spolupracovat pro dosažení efektivního průběhu projektových činností (Máchal et al., 2015; Svozilová, 2016).

Za sestavení projektového týmu by měl být zodpovědný projektový manažer. Do projektového týmu by měli být vybíráni kandidáti na základě stanovených charakteristik, které zajistí maximální efektivitu a souhru při realizaci projektu. Mezi základní charakteristiky se mohou řadit například dovednost jednotlivce začlenit se do týmu a spolupracovat s ostatními členy, požadované kompetence a dovednosti v oboru, samostatnost v rozhodování nebo kreativní a inovativní přístup (Mácha et al., 2015).

Vývoj projektového týmu je definován následujícími fázemi:

- fáze formování (*forming*),
- fáze konfliktů a neshod (*storming*),
- fáze sjednocení a ustálení (*norming*),
- fáze výkonu (*performing*),
- fáze dokončení a rozpadu (*adjourning*) (Doležal, 2023).

V současné době je na projektového manažera ve fázi vývoje týmu nahlíženo jako na vedoucího týmu, který se o tým spíše stará, než aby mu přikazoval, jak se má vyvíjet. Stanovuje určité hranice, vymezuje směr vývoje a usměrňuje jej (Doležal, 2023).

Ve fázi formování se prvně setkávají a seznamují vybraní členové projektového týmu, kteří se navzájem nemusí téměř znát a o projektu toho vědí také minimum. V této fázi je u projektového manažera důležité, aby všechny zúčastněné dostatečně informoval o projektu, jaký je jeho význam, jaké jsou jeho hlavní cíle a jak bude spolupráce vypadat. Také je nutné, aby vytyčil jasné pokyny a očekávání a tým stanovil směr (Doležal, 2023).

Následuje fáze konfliktů a neshod, ve které vyjde najevo, jak podrobně byly probrány důležité věci, co nebylo dořešeno a kde nastaly mezery. V určitých momentech mohou vznikat situace, ve kterých se objevují nedorozumění a mnohdy i hádky. Pokud v této fázi projektový manažer nezasáhne a nezačne vývoj řídit, dalších fází vývoje se projektový tým již nedočká, protože v tomto momentě se stane vývoj poslední fází. Správně by se měl vedoucí projektu zachovat tak, že pomůže stranám, kterých se spor týká, aby společně našly řešení na daný konflikt. Tím docílí toho, že nebude spor řešit pouze on sám a zodpovědnost za vymyšlené řešení je na daných stranách. Zároveň tím pomůže týmu stát se více samostatným a podporujícím v řešení problémů a sporů (Doležal, 2023).

Pokud se vývoj překlene do fáze ustálení, znamená to, že je vývoj na dobré cestě. Většina neshod se vyřešila, mezery se vysvětlily a lidé spolu fungují bez větších problémů. Je však důležité tuto fázi nezanedbávat a stále tým podporovat v samostatném rozhodování. Pokud v této fázi manažer projektu nijak nezasáhne a nevyvine žádné úsilí, může se i z této, jinak stabilní fáze stát fáze konečná, protože lidem v týmu začne chybět podpora a motivace od vedoucího (Doležal, 2023).

Projektový tým, který se dostane do fáze výkonu, je velmi efektivní a plný síly. Lidé uvnitř týmu mají nastavenou příjemnou atmosféru. Pokud vznikne spor, je zpravidla vnímán z pohledu neustálého zlepšování se a tým ho dokáže sám vyřešit. Z pozice projektového manažera je hlavním cílem tým v této fázi udržet až do dokončení projektu. Pokud nastane jakákoliv změna, je nutné urychleně s týmem projít všechny předchozí fáze (Doležal, 2023).

Po dokončení projektu se tým rozpadne. Pokud projekt trvá delší dobu, lidé si na sebe zvyknou a odloučení pro ně bude těžké. Projektový manažer zde funguje hlavně jako

psychická podpora týmu. Tým se rozloučí a jednotliví členové se vydají každý svou cestou, připraveni čelit novým výzvám (Doležal, 2023).

1.3.1 Týmové role a odpovědnosti

Pro projektový tým je klíčová vzájemná spolupráce členů týmu, aby realizace projektu dokázala probíhat efektivně a mohly být dosaženy stanovené cíle. Každý člen projektového týmu má své přednosti a silné stránky. Tým je proto vhodné řídit tak, aby byly využity právě tyto silné stránky jednotlivců, jejich slabé stránky vykompenzovány a využít tak naplno kolektivní potenciál pro dosažení pozitivní synergie (Křivánek, 2019).

Každému jednotlivci v týmu je přiřazena role pro efektivní týmovou práci. Ta určuje, jakým způsobem se jednatel v týmu chová a jak působí na své okolí. Je nutné stanovit role tak, aby odpovídaly dovednostem a schopnostem jedinců. V případě, že přiřazená role jedinci neodpovídá, je mu odebrána, aby se předešlo zbytečným konfliktům v nenaplnění očekávání určitého chování. Všechny role v týmu jsou si sobě rovné. Jednatel může být obsazeno i více rolí, avšak by neměly být protikladem pro zabránění sporu. Pro produktivní týmovou práci by měly být členům přiřazeny takové role, aby byly celkově vyvážené. V tomto ohledu je klíčové využít silné stránky jednotlivých členů a ty vzájemně doplňovat. Na druhou stranu je nutné brát v potaz jejich nedostatky a ty nejlépe kompenzovat (Křivánek, 2019).

Mezi nejznámější a nejvyužívanější teorie o rozdělení rolí v projektovém týmu patří Belbinova teorie. Meredith Belbin provedl rozsáhlou studii chování lidí, jejíž výsledkem bylo zjištění, že v jakékoliv uzavřené skupině se časem vytvoří určité role. V Belbinově teorii je využíváno 9 rolí, které se dělí do tří skupin:

- role zaměřené na myšlení,
- role zaměřené na akci,
- role zaměřené na lidi (Doležal, 2023; Křivánek, 2019).

Jednatelci, kteří jsou zařazeni do skupiny myšlení, mohou nabývat role Myslitele, který přináší nové nápady a řešení problémů, avšak může být zaslepen pouze svojí pravdou. Dále jim může být přiřazena role Vyhodnocovače, jež vše kontroluje

a prověřuje možnosti řešení, jelikož o dost věcech pochybuje. Poslední rolí ve skupině myšlení je Specialista, který je orientovaný pouze na oblast svého působení (Doležal, 2023; Křivánek, 2019).

V rámci skupiny akce se vyskytuje Formovač, který je cílevědomý a jde si silně za svými názory. Snaží se přesvědčit ostatní v realizaci svých nápadů a chce je řídit. Realizátor je jedinec, který je zodpovědný a spolehlivý. Jde rovněž za stanoveným cílem a koná přesně zadanou práci. Následně je zde Dotahovač, který se ve velkém množství věcí shoduje s Realizátorem. Je pečlivý a starostlivý, nechce v práci zanechat chyby, dotahuje činnosti do konce a v termínu (Doležal, 2023; Křivánek, 2019).

V poslední skupině rolí, která je zaměřená na lidi se objevuje role Koordinátora, který je charakterizován sebejistým a vyzrálým chováním v různých situacích. Dohlíží na správné rozdělení činností mezi ostatní. Je pro něho důležité spojení lidí a volný prostor všech pro vyjádření svých názorů. Pro Týmového pracovníka je naopak důležité, aby byl mezi lidmi soulad. Snaží se v týmu předejít konfliktům. Jedinec, který zastává roli Vyhledávače zdrojů má povahu extroverta, který se dokáže nadchnout pro nové věci a příležitosti. Dokáže zajistit vše nutné (Doležal, 2023; Křivánek, 2019).

Správné přiřazení rolí v týmu je velmi důležité. Pokud se setkají role stejných typů, nemusí být práce v týmu efektivní. Je vhodné nahlížet na role z pohledu vzájemné propojenosti, kdo koho doplňuje a usměrňuje. Jednotlivé role lze využít v průběhu celého projektu nebo pouze na jeho části. Zodpovědností projektového manažera je dohlížet na správnou kombinaci a vyváženost rolí v týmu a pokud stanovené role nefungují, včas tuto záležitost řešit, aby nedošlo ke konfliktům (Křivánek, 2019).

1.3.2 Projektový tým v organizaci

Pro efektivní řízení projektového týmu je vytvářena organizační struktura projektu, jež je zasazena do společnosti. Organizační struktura projektu sdružuje lidi, kteří na daném projektu pracují, napomáhá definovat vztahy mezi členy projektového týmu, jejich povinnosti a zodpovědnosti. Projektovému manažerovi usnadňuje řízení a dohled nad průběhem činností. Po dokončení projektu je organizační struktura rozpuštěna (Burke a Barron, 2014).

Vytvoření dočasné organizační struktury projektu ve společnosti může být realizováno mnoha způsoby. S ohledem na neustále se měnící charakter projektů se lze setkat se situací, ve které fungují různé organizační struktury projektů současně (Burke a Barron, 2014).

Hierarchická struktura je jedna z nejzákladnějších organizačních struktur využívaná v různých typech společností. Hlavní charakteristikou této struktury je sdružování lidí dle určitých oblastí (marketing, vývoj a další). V rámci každé oblasti působí jeden manažer, který má zodpovědnost za celý svůj tým specialistů v dané oblasti. Výhodou hierarchické organizační struktury je její přehlednost. Na první pohled lze poznat jednoznačné kompetence a odpovědnosti. Vedení jednotlivců je snazší pod manažerem, který se dané specializaci věnuje a rozumí ji. Využití hierarchické struktury je vhodné zejména pro menší projekty, které jsou realizované pouze uvnitř daného oddělení, nebo projekty, u kterých může každé oddělení vykonávat činnosti nezávisle na ostatních odděleních. Příkladem může být projekt v oblasti staveb, u kterého si jednotlivá oddělení zapojená do realizace projektu předávají práce. Pokud jedno oddělení dokončí svůj rozsah práce, předá práci dál jinému oddělení, které na činnosti naváže (Burke a Barron, 2014; Křivánek, 2019).

Ovšem, pro jiné projekty, zejména v rámci IT, by mohla být hierarchická struktura spíše nevýhodná. Nedokáže se dostatečně přizpůsobit, pravomoci manažera jsou omezeny pouze v rámci daného oddělení a ve většině případů není stanovena hlavní zodpovědnost za celý projekt. Pokud se vyskytne jakýkoliv problém, řeší se složitě, kolektivně, a to celý průběh projektu zpomaluje. Také zde chybí flexibilita při pohledu na rychle měnící se technologie. Znázornění hierarchické struktury projektu je na Obrázek 3 (Brewer a Dittman, 2018; Křivánek, 2019).



Obrázek 3: Hierarchická struktura projektu
Zdroj: upraveno dle Eby, 2023

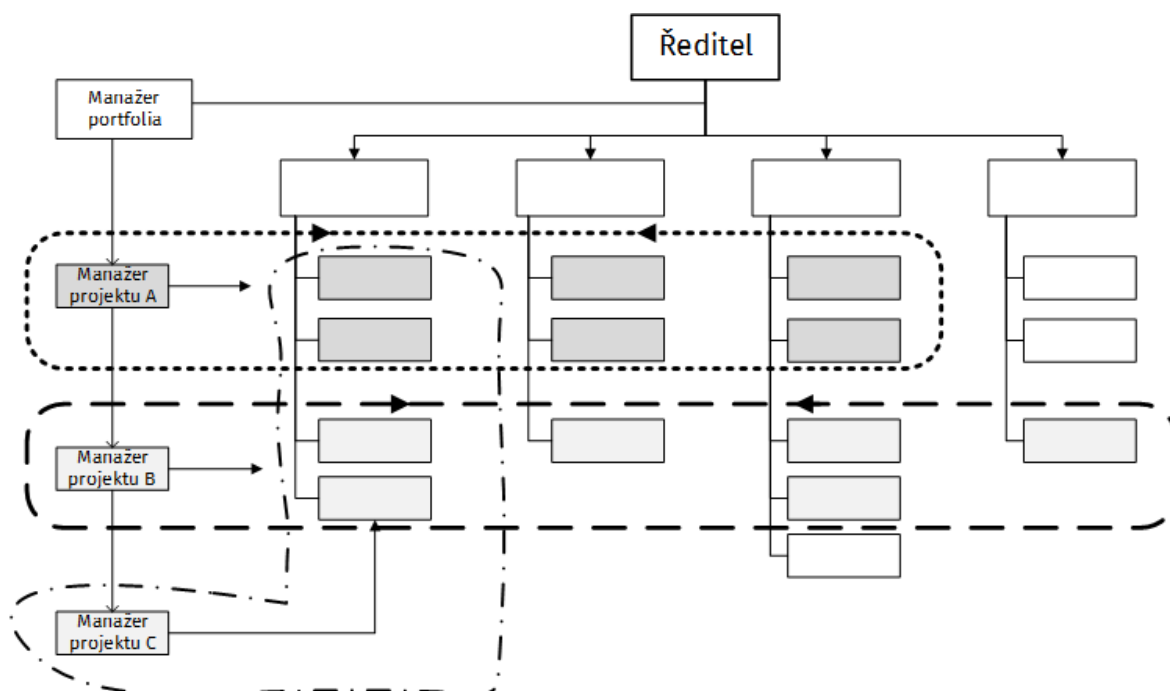
Projektová organizační struktura, jež je znázorněna na Obrázek 4, se vymezuje tím, že jednotlivá oddělení jsou dílčí projektové týmy s vlastním projektovým manažerem, který má nad projektem plnou zodpovědnost a pravomoc, tedy že se nemusí podřizovat žádnému vyššímu managementu ve společnosti. Členové týmu jsou v organizaci uplatňováni pouze pro provedení projektu a plně se zodpovídají svému projektovému manažerovi. Tato struktura rovněž nabízí efektivní komunikaci mezi zapojenými stranami. (Burke a Barron, 2014).



Obrázek 4: Projektová organizační struktura
Zdroj: upraveno dle Eby, 2023

Maticová organizační struktura projektu, kterou lze vidět na Obrázek 5, vznikla spojením prvků hierarchické a projektové organizační struktury. Projektový tým je složen z lidí, kteří za normálních okolností pracují pod svým nadřízeným manažerem v rámci konkrétní oblasti. Pro projekt je pravomoc nad řízením těchto lidí rozdělena mezi projektového manažera a manažera konkrétního oddělení. Pokud je na strukturu nahlíženo svisle, odpovědnost za lidi náleží manažerovi oddělení. V případě vodorovného pohledu má zodpovědnost za členy týmu projektový manažer. V závislosti na tom, jak velké kompetence má projektový manažer, lze mluvit o slabé či silné maticové struktuře, mezi kterými se vyskytuje mnoho dalších variant. Jakmile je projekt dokončen, maticová struktura se rozpustí a členové týmu se opět plně vrátí ke svým stálým pracovním pozicím (Burke a Barron, 2014; Doležal, 2023).

Slabá maticová struktura se vyznačuje malou rozhodovací pravomocí projektového manažera, který s manažery konkrétních oddělení vyjednává a spolupracuje. Zásadní slovo zde má manažer konkrétního oddělení, ze kterého člen týmu pochází. V případě silné maticové struktury veškerá zodpovědnost přechází na projektového manažera a manažerům oddělení náleží omezené pravomoci (Doležal, 2023).



Obrázek 5: Maticová organizační struktura
Zdroj: PM Consulting, b.d.

1.4 Projektový manažer

Projektový manažer zastává v projektu velmi důležitou funkci. Má několik rolí, které je z jeho strany nutné dodržovat a respektovat. Jako vedoucí má dohled nad celým projektem s tím, že naslouchá všem zainteresovaným stranám. Jeho hlavním cílem je efektivně řídit veškeré procesy a činnosti od počátku projektu až po jeho dokončení, včetně vedení lidí ve svém týmu (Fielding, 2020; Křivánek, 2019).

Jedny z klíčových prvků, které ovlivňují fakt, zda bude projekt úspěšný či nikoliv jsou kompetence projektového manažera, zejména pak jeho povaha a měkké dovednosti. Ty hrají velkou roli v tom, jak bude projektový tým veden a zda toto vedení pod daným manažerem s daným charakterem dovede tým k úspěšnému cíli (Křivánek, 2019).

Projekt není jednoduchá záležitost. Je to velmi komplexní a složitá soustava činností, která vyžaduje pro jeho dokončení nejen schopný tým lidí s různými dovednostmi a znalostmi, ale také sadu nástrojů a technik, pomocí kterých je dosahováno stanovených cílů. Projektový manažer by měl mít určité znalosti, zejména pak znalost těchto nástrojů, metod a technik k účinnému projektovému řízení. Dále by měl umět tyto nástroje správně použít, tedy převést teoretickou rovinu do roviny praktické.

Nakonec je důležité, aby byl schopen využít svých znalostí a dovedností k dosažení očekávaného výsledku praktickým využitím projektových nástrojů, metod a technik (Fielding, 2020; Křivánek, 2019)

Jak bylo zmíněno v této diplomové práci výše, významnou roli v projektovém vedení hraje mimo technických kompetencí také osobnost projektového manažera. Pro úspěch projektu je zásadní, jak na lidi a činnosti projektový manažer nahlíží, jak se v určitých situacích zachová a jaký je jeho celkový přístup k řízení. Úspěšný projektový manažer je takový, který uchopí práci s lidmi správně a uvědomuje si, že bez lidí by v projektu nikdy nebylo dosaženo cílů (Fielding, 2020).

Mezi základní dovednosti úspěšného projektového manažera patří správná komunikace s lidmi. Komunikace je nedílnou součástí projektu. Předávání informací může probíhat mnoha způsoby. Ať už je komunikace uskutečněna písemně či ústně, vždy by měla být jasná, stručná a pochopitelná. Dále by měl být projektový manažer schopen naslouchat a zřetelně předávat myšlenky dále. Nedostatečná komunikace nebo špatné předání informací může zapříčinit zmatek, který může vyústit v konflikt. Pokud ke konfliktu z jakéhokoliv důvodu dojde, správný projektový manažer se k němu postaví čelem a bude jej brát jako možnost dalšího rozvoje. Důležité je vyslechnout si všechny strany a na spor nahlížet z různých úhlů pohledu (Fielding, 2020; Veber, 2021).

Další důležitou mezilidskou schopností je práce se stresem a motivace členů týmu. Pokud dokáže projektový manažer ve svém týmu pomáhat ostatním se zvládnutím stresu, může být v týmu nastolena radostná a produktivní atmosféra. Dostatečná motivace zajistí chuť a odhodlání společně dosáhnout cílů projektu (Doležal, 2023; Veber, 2021).

Projektový manažer má velkou zodpovědnost. Za lidi, za průběh i za výsledek projektu, který musí být v souladu s požadavky všech zapojených účastníků projektu. To vše musí zvládnout s ohledem na stanovené náklady, čas i rozsah. Je velmi důležité, aby byl ztotožněn s důvodem vzniku projektu a jeho zadáním. Podstatné je, aby věřil svému projektovému týmu, který vykonává veškeré činnosti spojené s průběhem projektu. Od správného projektového manažera je očekáváno převzetí

plné odpovědnosti za projektový tým, projektový výstup a odhodlání udělat vše pro to, aby bylo úspěšně dosaženo stanovených cílů (Křivánek, 2019).

2 Specifikace projektů v oblasti vývoje softwaru

Software je nedílnou součástí jedince v jeho každodenním životě. Přesahuje do stále více oborů a odvětví. Dynamičnost prostředí kolem nás je z jedné části zapříčiněna právě rychlým vývojem softwarů a systémů. Rychlý technologický pokrok vede k častým změnám a inovacím softwaru, což klade velký důraz na správné řízení vývoje (Ahmed, 2016; Ruhe a Wohlin, 2014).

Vývoj softwaru se řadí mezi jeden z typů projektů, kterými se lze v softwarovém prostředí zabývat. Projekty jsou zaměřeny na tvorbu aplikací nebo systémů a poskytování přidaných služeb či výstupů, které podporují vytvořené aplikace či systémy (Villafiorita, 2016). S ohledem na neustále se měnící prostředí a rozvoj technologií včetně inovací je vývoj softwaru jednou z nejsložitějších činností (Igberaese, 2022).

V souvislosti s projektovým řízením lze rozlišit základní typy softwarů, které jsou vyvíjeny. Prvním typem jsou jednorázové aplikace nebo systémy, které jsou zakázkově vytvořené pro zákazníka dle jeho specifických potřeb. Vytvořený software může být zákazníkovi předán s veškerým vlastnictvím. Dalším typem softwaru, který může být vyvíjen, jsou aplikace nebo systémy určené pro širokou veřejnost a jsou specifické tím, že jsou pro všechny uživatele totožné. Jejich funkcionalita je stanovena společností, která produkt vyvíjí. Potřeby aplikace nebo systému mohou být zjišťovány na základě průzkumů. Posledním příkladem je typ produktu, který kombinuje předešlé dva typy. Aplikace nebo systémy jsou vyvíjeny tak, aby byly široce využitelné, ovšem další vlastnosti softwaru jsou přizpůsobené konkrétnímu zákazníkovi. Příkladem tohoto typu je systém plánování podnikových zdrojů (Villafiorita, 2016).

Uživatelé vnímají softwarové aplikace jako něco jednoduchého, co jim zjednodušuje každodenní život. Očekávají, že aplikace nebo systém budou fungovat spolehlivě a bez problémů. Nevidí však, že se na pozadí odehrává řada komplexních procesů, operací a technických záležitostí, které tuto bezchybnost a spolehlivost zajišťují (Murray, 2016). Softwarové projekty neodrážejí pouze dodržené stanovené požadavky, ale rovněž zahrnují určitou úroveň kreativity a inovace, která musela být do projektu vložena (Ahmed, 2016).

2.1 Projektový management vývoje softwaru

Projektově řízený vývoj softwaru je náročný proces, při kterém se musí, ostatně jako u každého jiného projektu, dodržovat tři stanovené hranice – náklady, kvalita a čas. Softwarové produkty mají svá specifika, která tento proces dělají komplikovanějším. Z důvodu jejich nehmotné podoby mohou být pro jedince obtížně představitelné. Dále mohou být svým řešením komplexní a rovněž vyžadují flexibilitu, například v rámci dostupných technologických možností nebo aplikačních prostředků (Igberaese, 2022).

Pro úspěšné projekty věnující se softwaru je klíčovým východiskem nejen znalost obecného projektového řízení, ale také základní znalosti v oblasti softwarového inženýrství, které se věnuje systematickému přístupu k vývoji, provozu a správě softwaru (Ahmed, 2016; O'Regan, 2022).

Kromě procesů projektového managementu, které byly podrobněji rozebrány v této diplomové práci výše, je nutné začlenit také procesy softwarového inženýrství, mezi které se řadí procesy specifikace požadavků, návrh aplikace nebo systému, tvorba a testování softwaru a jeho následná údržba (Ahmed, 2016).

2.2 Softwarový projekt

Projekty realizované v oblasti vývoje aplikací nebo systémů často představují vyšší náročnost v porovnání s jinými typy projektů. Mezi hlavní charakteristiky, jež dělají vývoj softwaru složitější, lze zařadit:

- software je produkt, který není hmotný,
- u softwarových projektů se vyskytuje vyšší míra nejistoty,
- v průběhu realizace projektu často dochází ke změnám v požadavcích,
- je obtížné zajistit jasnou a zřetelnou komunikaci a koordinaci mezi projektovým týmem a ostatními zainteresovanými stranami,
- vývoj softwaru vyžaduje inovativní přístup pro hledání nových a jedinečných řešení,

- plánování projektu stojí na požadavcích, které jsou ve většině případů neúplné, nepřesné nebo v průběhu projektu měněné,
- objektivní hodnocení kvality softwarového produktu je obtížné z důvodu hodnocení mnoha aspektů, na které je nahlíženo spíše subjektivně,
- oblast softwarových projektů se neustále vyvíjí (Ruhe a Wohlin, 2014).

Výše uvedené základní charakteristiky softwarových projektů přináší mnoho složitých situací, se kterými se může potýkat celý projektový tým. Dobrý projektový manažer, jenž je zodpovědný za řízení softwarového projektu, musí být na tyto nečekané situace připraven, musí být schopen na ně efektivně reagovat a umět s nimi pracovat. Vzhledem k vyšší míře nejistoty softwarových projektů je přínosem, pokud je projektový manažer nápaditý a nakloněný k flexibilitě, jež zaručí zvolení vhodných technik k zajištění požadovaných výsledků. Dále je pro dobrého projektového manažera prospěšné mít smysl pro naléhavost, jež zajistí dokončení projektu v rámci stanoveného časového omezení. Je rovněž nezbytné udržovat projektový tým v efektivním a udržitelném tempu, což zajistí vyšší úspěšnost dokončení projektu (Villafiorita, 2016).

Technické dovednosti a zkušenosti v oblasti řízeného projektu jsou pro projektového manažera softwarového projektu nepostradatelné. Schopnost efektivně zvolit a využít správné nástroje a techniky dokáže zajistit zjednodušení průběhu projektu. Složitost projektu a neustále proměnlivých požadavků nesmí být pro projektového manažera překážkou, nýbrž výzvou, jíž dokáže účinně organizovat (Radaideh, 2023; Villafiorita, 2016).

Oblast IT je specifická tím, že lidé, kteří se IT věnují, pracují často v režimu remote, což znamená, že není nutná jejich fyzická přítomnost v kancelářích či jiných společných prostorech. Z toho důvodu mohou vznikat virtuální týmy, jejichž členové mohou pracovat z různých míst. Pokud na projektu pracují lidé pocházející z různých koutů světa, problémem může být také časový rozdíl v daných zemích (Allan, 2017).

Ve virtuálních týmech musí panovat silná důvěra mezi jednotlivými členy týmu a mezi týmem a projektovým manažerem. Budování dobrých vztahů a důvěry ve virtuálním týmu je mnohdy pomalejší než u týmu, který se setkává osobně. Složitější je také komunikace a dostupnost informací o projektu. Tým, který se setkává osobně, si

předává informace mnohem snadněji než tým, který pracuje na projektu vzdáleně. Pro fungování takového týmu je zapotřebí většího úsilí (Martinelli et al., 2017).

Pro projektového manažera to znamená čelit více výzvám, než pokud by měl tým na jednom fyzickém místě. Zejména musí zajistit vhodný komunikační nástroj, případnou technickou podporu a identifikaci členů v online prostředí pomocí ID a hesel (Allan, 2017).

2.3 Software Development Life Cycle (SDLC)

Software Development Life Cycle (SDLC), neboli životní cyklus vývoje softwaru, označuje proces softwarového vývoje, jenž je rozdělen do několika fází. SDLC napomáhá usměrňovat vývoj a zvyšuje příležitost na jeho úspěšné dokončení. Jednotlivé fáze jsou charakterizovány souborem činností, které jsou v průběhu projektu prováděny a které přispívají k úspěšnému vývoji a nasazení softwarového produktu (Pendyala, 2019). Základní fáze vývoje softwaru, mezi které se řadí plánování, analýza požadavků, návrh, kódování, testování, nasazení a údržba, jsou znázorněny na Obrázek 6.



Obrázek 6: SDLC
Zdroj: upraveno dle Aytimiz, 2023

V této diplomové práci byly zmíněny procesy projektového managementu. Společně s procesy životního cyklu vývoje softwaru zastávají v projektu velmi důležitou roli. Společně tvoří rovnocenné prostředí, uvnitř kterého se procesy vzájemně doplňují

a existují v harmonii. Každý za sebe má svou hlavní roli a společně vytváří komplexní metodiku pro poskytnutí vysoce kvalitního výstupu projektu (Font Jr., 2013).

2.3.1 Formulace nápadů či problémů

Fáze prvotní formulace nápadů na software či problémů, které má software řešit, není v mnoha literaturách uvedena, ale autor této diplomové práce ji považuje za stejně důležitou, jako další následující. Pokud nevznikne nápad či nová myšlenka na novou aplikaci či funkci, další fáze nemají na co navazovat, respektive není co plánovat, není co analyzovat a co implementovat.

V této fázi jsou projednávány potřeby nebo příležitosti, ze kterých vzniká první představa konečného výsledku (Boyde, 2014). Pro rozvoj nápadů nebo řešení problémů je k dispozici mnoho technik, které zároveň podporují kreativitu jedinců. Jedna z nejpoužívanějších technik je metoda Brainstorming, která spočívá v návrhu nových nápadů v malé skupině lidí, u které je vhodné mít maximálně 12 členů. Nápady jsou volně předkládány a následně je vybrán ten nejlepší. V případě, že jsou nápady jednotlivci zapisovány, je mluveno o metodě Brainwriting (Doležal, 2023). Mimo jiné je také velmi důležitým prvkem této fáze zpětná vazba ze strany cílové skupiny. Pokud je nový nápad na aplikaci, novou funkci nebo vylepšení současného systému ze strany cílové skupiny pozitivně ohodnocen, projekt může postoupit do následující fáze plánování (Braude a Bernstein, 2016).

2.3.2 Plánování

Po formulaci nápadu nebo řešení problému následuje vypracování plánu projektu, který zahrnuje informace o tom, v jakém rozsahu bude projekt prováděn, jak bude časově a nákladově omezen a jaká bude jeho konečná kvalita. Dále jsou v plánu zohledněny lidské zdroje a činnosti vedoucí k dokončení projektu. Fáze plánování si zejména klade za cíl nahlédnout na celý naplánovaný projekt a zhodnotit stanovená kritéria. Pokud z plánu na vývoj nové aplikace vzejdou příliš vysoké náklady, může být snížen počet nových funkcí. Nebo naopak, pokud je po nastavení časového harmonogramu zjištěno, že bude aplikace vydána příliš pozdě, může být pro projekt vyhrazeno více zdrojů. Je tedy velmi důležité identifikovat tyto případné problémy,

které mohou v budoucnu projekt výrazně ovlivnit a začít je řešit včas (Boyde, 2014; Braude a Bernstein, 2016).

2.3.3 Analýza požadavků

Základem pro vývoj softwaru je podrobná analýza a definice specifických požadavků na software v závislosti na potřebách a přáních zadávajícího projektu a rovněž na specifických problémech, které má softwarový produkt řešit. V této fázi jsou dopodrobna řešeny specifické funkce a vlastnosti softwaru vzhledem k požadované kvalitě, ve které se může nahlížet na výkon aplikace, její použitelnost nebo spolehlivost (Braude a Bernstein, 2016). Dále je například zjišťováno, jaká má projekt technická omezení (Bartoš, 2021).

2.3.4 Návrh softwaru

Návrh softwaru slouží k vizualizaci, jak budou stanovené specifikace převedeny do reálné podoby v rámci implementace (Boyde, 2014). Návrh modeluje vnitřní strukturu softwaru tak, aby odpovídal stanoveným požadavkům a byly znázorněny vztahy a interakce s ostatními funkcemi aplikace. Je stanovena architektura softwaru, návrh podoby uživatelského prostředí a databáze (Bartoš, 2021; Braude a Bernstein, 2016).

2.3.5 Implementace

Ve fázi implementace probíhá samotné programování návrhu softwaru a vzniká tak první podoba navrženého softwaru. K implementaci kódu se využívají programovací jazyky, jejichž volba závisí na typu softwaru, který je vyvíjen, a jeho požadavcích. Mezi nejznámější programovací jazyky se řadí PHP, C++, Java a další (Bartoš, 2021; Kazim, 2017). Výstupem této fáze je sepsaný kód, který je následně testován (Braude a Bernstein, 2016).

2.3.6 Testování

Po tom, co je kód napsán, je nutné jej otestovat, aby nebyl vydán do produkce s chybami a dohlédnout na to, zda je naprogramován dle specifikací, které byly stanoveny. Testování může probíhat několika způsoby a na několika úrovních. Pokud je ve fázi testování shledán problém, který je v rozporu se stanovenými požadavky, je kód přepsán tak, aby jeho výsledek souhlasil se zadáním. Testování probíhá do té doby, dokud není software připraven k nasazení na produkci, a může se několikrát opakovat (Braude a Bernstein, 2016; Kazim, 2017).

2.3.7 Nasazení

V případě, že naprogramovaný software splňuje veškeré specifikace a požadavky, včetně správné funkčnosti, je nasazen do ostrého provozu. Je potřeba zajistit, aby implementace proběhla bez problémů a nezpůsobila žádné potíže při přechodu z testovací fáze do fáze produkční (Bartoš, 2021).

2.3.8 Údržba

V této fázi je již software plně zpřístupněn koncovým uživatelům a nastává následná údržba a péče o software. V průběhu ostrého provozu je software monitorován a v případě nastání potíží je v co nejkratším čase opravován. Dále je v rámci této fáze shromažďována zpětná vazba na software od uživatelů, která může být základem pro další vylepšení softwaru. Veškeré opravy a úpravy softwaru jsou vydávány jako nové verze softwaru stávajícího (Braude a Bernstein, 2016; Kazim, 2017).

3 Přístupy k vývoji softwaru

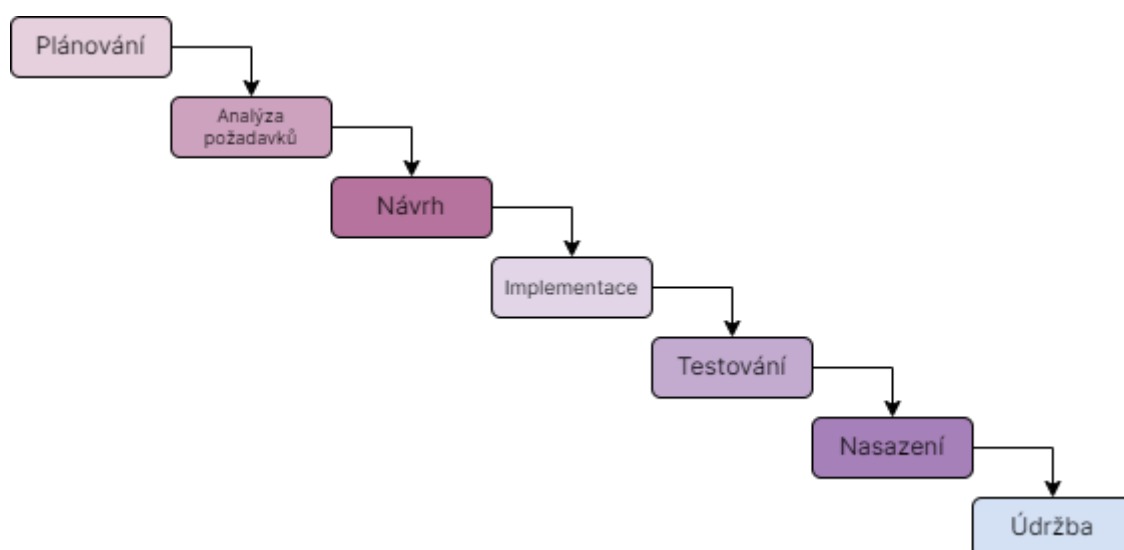
Různé typy softwarů používají pro svůj vývoj různé přístupy. Každý z přístupů má své výhody i nevýhody, a je pouze na rozhodnutí projektového manažera, jaký přístup bude vyhodnocen v závislosti na požadavcích a charakteristice softwarového produktu jako vhodný a který bude následně implementován (Kumar a Rashid, 2018).

Volba přístupu k vývoji softwaru ovlivní, jakým způsobem bude projekt plánován, řízen a kontrolován projektovým manažerem. Také bude ovlivněna volba nástrojů a technik, které budou v průběhu projektu využity (Curry, 2021).

3.1 Tradiční vodopádový přístup

Pro tradiční vodopádový model je charakteristická jeho lineárnost a postupnost. Vývoj softwaru prochází postupně jednou fází po druhé. Následující fáze započne až po dokončení fáze předchozí. Výjimečně může docházet k menšímu překrývání vedlejších fází (Braude a Bernstein, 2016).

Vodopádový přístup, jehož vizuální podoba je znázorněna na Obrázek 7, je vhodný zejména u menších projektů, jež mají jasně definované požadavky, které budou v průběhu projektu neměnné a nejsou očekávány žádné změny (Bartoš, 2021).



Obrázek 7: Vodopádový přístup
Zdroj: vlastní zpracování

Hlavní výhodou vodopádového přístupu je jeho jednoduchost. Jednotlivé fáze jsou přesně definované, mají jasně stanovené výstupy a jsou dokončovány postupně za sebou, což dělá tento model snadno pochopitelným (Kazim, 2017).

Mezi nevýhody se řadí zejména jeho neschopnost být flexibilní. Pravidla vodopádového přístupu nedovolují vracet se ve fázích zpět a dokončené fáze měnit. Z toho důvodu není schopen přizpůsobit se změnám (Brewer a Dittman, 2018). Proces vývoje softwaru dle vodopádového přístupu bývá pomalý a není jisté, co lze očekávat od výsledného produktu a zda bude s výsledkem spokojen zejména koncový uživatel (Sutherland, 2019).

3.2 Agilní přístup

Vodopádový přístup měl v mnoha ohledech jisté nedostatky, co se vývoje softwaru týče. Než došel projekt do svého konce, mohly se požadavky a potřeby koncového zákazníka změnit (Bennett a Bowen, 2018). Časem se požadoval urychlený vývoj s možností efektivně reagovat na změny, které se během vývoje softwaru naskytly. V roce 2001 vznikla skupina *Agile Alliance*, která reagovala na veškeré požadavky, které měly vývoj softwaru zlepšit a vznikl tak *Agile Manifesto*, který shromažďuje hodnoty a principy agilního přístupu (Braude a Bernstein, 2016).

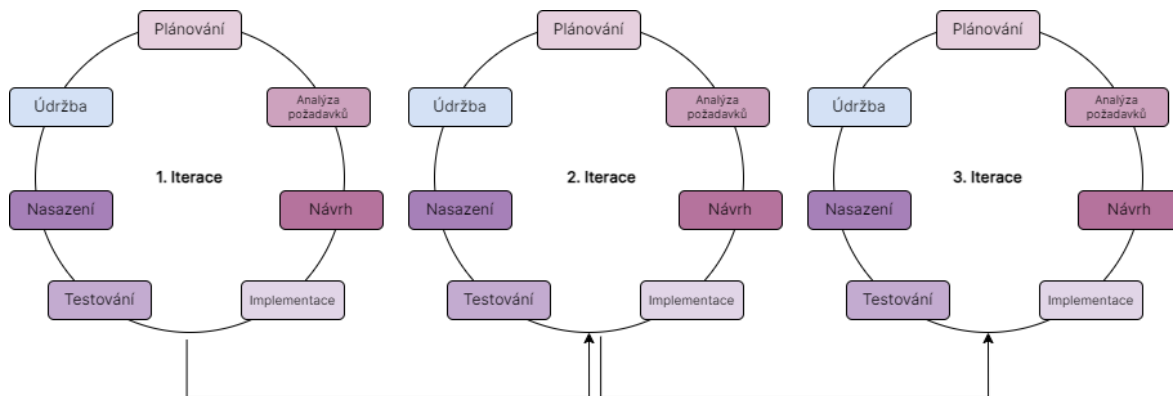
Agilní přístup k vývoji softwaru je oproti vodopádovému přístupu více pružný a přizpůsobivý změnám. Jeho hlavní charakteristikou je rychlé dodání softwaru v jednotlivých opakujících se cyklech, jejichž výstupem je vždy část produktu, který inkrementálně roste. Agile nepředstavuje návod, jak vyvíjet software, ale jedná se o pochopení agilní filozofie, která se zaměřuje na týmovou spolupráci, komunikaci a přijetí změny. Má svá pravidla, která nastavují hranice v týmu, ve kterém není striktně dáno, jakým způsobem mají členové pracovat. Týmy mají větší volnost v rozhodování, jakým způsobem budou software vyvíjet, jak budou postupovat. Cílem je být efektivní, rychlý a vyvíjet kvalitní produkt v krátkém časovém horizontu. Vývoj je zaměřen zejména na přinesení hodnoty koncovému uživateli, aby byl s výsledkem spokojený právě on (Šochová a Kunc, 2019).

Manifest agilního vývoje softwaru definuje několik základních zásad a hodnot, které je vhodné dodržovat, pokud je žádáno k vývoji přistupovat agilně. První zásada pojednává o týmové spolupráci, která je mnohem hodnotnější, než pokud by v týmu pracovali členové s přesně nastavenými procesy a definovanými nástroji bez žádné interakce. Pro rychlejší a kvalitnější vývoj je vhodné vytvořit takové prostředí, ve kterém členové společně spolupracují, vymýšlejí společná řešení nad danou problematikou a vybírají si nástroje, které považují ke své práci za vhodné (Šochová a Kuncce, 2019).

Dalším principem je upřednostnění fungujícího softwaru před dlouhou dokumentací popisující jeho funkčnost. Koncový uživatel dává přednost praktickému vyzkoušení před jeho teoretickou definicí. Dokumentace je vhodná pro představení oblastí softwaru, které nejsou snadno pochopitelné. Ovšem snahou je počet takových částí minimalizovat. Je nutné přemýšlet nad hodnotou, kterou daná dokumentace přinese (Šochová a Kuncce, 2019).

Velmi důležitým principem manifestu je úzká spolupráce se zákazníkem v průběhu celého projektu narozdíl od pouze zaznamenaných pevných požadavků na software ve smlouvě na začátku spolupráce. Průběžná zpětná vazba od koncového uživatele zajistí větší úspěch a spokojenost za dodaný výsledný produkt (Šochová a Kuncce, 2019).

Poslední zásada poukazuje na fakt, že změny a problémy ve vývoji jsou nepředvídatelné. Z toho důvodu nemá smysl tvořit plán, který bude striktně dodržován. Plán by měl sloužit pouze jako vodítko, jakým směrem má vývoj postupovat, a v případě změny požadavků nebo vyskytnutí problému vývoj těmito vzniklým situacím přizpůsobit (Šochová a Kuncce, 2019). Obrázek 8 vizualizuje koncept agilního přístupu k vývoji softwaru.



Obrázek 8: Agilní přístup

Zdroj: vlastní zpracování

Výhody agilního vývoje spočívají zejména ve větší spokojenosti koncového uživatele s výsledkem, tým je více produktivní, vývoj je rychlý a produkt je kvalitnější. Dále pak ve snížení nákladů v závislosti na schopnosti přijmout změnu v průběhu vývoje a vývoj tak změnám přizpůsobit (Özkan a Mishra, 2019). V průběhu vývoje jsou mnohem snadněji objeveny případné problémy a chyby, které lze jednoduše opravit a tak lze dodat produkt zákazníkovi rychle a kvalitně (Bennett a Bowen, 2018).

Agilní přístup může být nevýhodný pro velké aplikace, které jsou složité a komplexní. Další nevýhodou může být nezodpovědný zákazník, který nebude plně spolupracovat. Jelikož není cílem přesně definovat požadavky a těch se striktně držet, ale na celém vývoji spolupracovat se zákazníkem, je v tomto ohledu pro projekt velmi důležitým prvkem. Z důvodu minimální dokumentace vývoje mohou nastat zmatky a nedorozumění v případě zaučení nových členů nebo převedení vývoje na jiný tým (Kazim, 2017).

3.2.1 Scrum

Scrum je označován za jednu z nejvíce užívaných a populárních agilních metodik. Vychází ze hry rugby, při které je základem sladěný tým, který jedná společně a jednotně se snahou společného dosažení stanoveného cíle. V tom tvůrce Scrumu Jeff Sutherland našel smysl i pro projektové týmy (Doležal, 2022; Sutherland, 2019).

Základním smyslem Scrumu je neustálé zlepšování procesu vývoje na základě zpětného dotazování, zda to, co bylo doposud uděláno, je uděláno správně a má to

svoji hodnotu, zda jde vývoj správným směrem nebo zda někde nejsou místa, které je možné ještě vylepšit (Sutherland, 2019).

Základním pilířem Scrumu je tým, který se skládá z lidí, kteří mají dohromady potřebné dovednosti a znalosti k tomu, aby společně dosáhli hodnotného cíle a případně si navzájem doplňovali další vědomosti. V týmu se objevují následující role:

- Product Owner,
- Scrum Master,
- Development tým (Doležal, 2022).

Product Owner je klíčový člen Scrum týmu, který je zodpovědný za to, aby vyvíjený produkt přinesl co největší možnou hodnotu. Na základě komunikace se zákazníkem definuje představu o produktu, je zodpovědný za stanovení priorit v seznamu funkcionalit nazývaného Product Backlog a zajišťuje, aby se všem dostaly veškeré nutné informace. Nejvíce času tráví se zákazníky, aby dokázal určit, jak má vypadat úspěšný projekt, který přinese hodnotu. Zákazníkem může být kdokoliv, kdo vyjadřuje jakýkoli zájem o projekt, ať už se jedná o uživatele aplikace, jakékoliv oddělení uvnitř firmy nebo externího zadavatele. Product Owner je dle potřeby k dispozici Development týmu, kterému udává směr. Jeho hlavní funkcí v rámci Scrum týmu je definování, na čem má tým momentálně pracovat a stanovení priorit jednotlivým položkám Product Backlogu (Šochová a Kunce, 2019).

Product Backlog je seznam, který se skládá z položek funkcionalit, které jsou vytvářeny z pohledu přinesení hodnoty zákazníkovi a je možné na ně získat zpětnou vazbu. Funkcionalita je popisována pomocí tzv. **User Stories**. User Story neposkytuje detailní technické specifikace, ale popisuje přínos, který má daná funkcionalita přinést. Je na ni nahlíženo z pohledu uživatelů. Položky v Product Backlogu jsou seřazovány dle priority, která je určena Product Ownerem. Při vytváření Product Backlogu Product Owner úzce spolupracuje s celým týmem, zákazníkem a dalšími lidmi, kteří mohou vnést další potřebné a užitečné pohledy. V průběhu celého projektu je Product Backlog všem přístupný. (Šochová a Kunce, 2019).

Development tým pracuje společně na dosažení výsledku, který přinese hodnotu zákazníkovi. Členové týmu jsou si navzájem rovni, nemají určené žádné role, mají

pouze jeden společný cíl. Tým je charakteristický tím, že s ohledem na stanovené hranice projektu se uvnitř organizuje sám, je tzv. *self-organized*. Tým si dokáže sám určit, jakým aktivitám se bude věnovat a jakým způsobem je bude vykonávat. To zajišťuje, že jsou členové dostatečně motivováni, provádí efektivní práci a dokážou být flexibilní. Aby takový tým dokázal fungovat, musí mít pevně stanovený společný cíl, který je každým členem následován. Dalším důležitým prvkem týmu je důvěra, jak mezi členy, tak rovněž k firmě a k zákazníkovi, kterému je produkt dodáván. V neposlední řadě je v takovém týmu nutná spolupráce a pochopení, že tým pracuje jako jeden celek a jako celek je zodpovědný za odvedenou práci. Další vlastností Development týmu, která byla již zmíněna výše, je multifunkčnost. Každý člen v týmu přispěje něčím, v čem si věří a má určité dovednosti, které dokáže sdílet mezi ostatní. Zároveň je nápomocen ostatním v závislosti na jeho vytížení (Šochová a Kunce, 2019).

Na tom, aby se tým rozvíjel a dokázal fungovat, se podílí **Scrum Master**. Jeho hlavním cílem je pomáhat týmu jako celku, dále pomáhat jednotlivcům v jejich rozvoji a umírnit případné konflikty. Neměl by ovšem zastávat pozici příkazujícího. Tým vede k samostatnosti a efektivní práci, aby úspěšně dosahoval stanovených cílů. Usiluje o to, aby byl tým motivovaný a spokojený. Povahově je komunikativní, empatický a přístupný celému týmu. Stará se o tým jako takový, aby při vývoji vše fungovalo hladce (Šochová a Kunce, 2019).

Vývoj je rozdělen do několika iterativních **Sprintů**, které mají přesně stanovenou dobu trvání a jejichž výsledkem je dodání hotové funkcionality. Funkcionality pro daný Sprint se vybírají ze **Sprint Backlogu**, jenž je součástí Product Backlogu. Pro každý Sprint je určen **Sprint Goal**, který je definován potřebami zákazníka a stanovuje tak smysl a hodnotu daného Sprintu. Sprint Goal by měl být stanoven pevně a neměnně. Po definování Sprint Goalu nastává fáze **Planning**, ve které je členy týmu naplánováno, jaké User Stories budou dle hodnoty a dostupné kapacity vybrány pro dosažení cíle Sprintu. Vybraná funkcionality je ve Sprintu rozložena do menších úkolů, které jsou postupně zpracovávány. Není vyloženě nutné, aby byly ve Sprintu dodány všechny položky Sprint Backlogu. Důležité je, aby byl dodržen Sprint Goal. Během Sprintu se konají krátká setkání, obvykle na denní bázi, nazývané **Standupy**. Tato setkání se konají z důvodu vzájemného sdílení postupu práce mezi členy týmu. Během Standupu se zhodnotí, co se stihlo za předchozí den, co je v plánu dokončit dnes

a jaké se vyskytly problémy, které zabraňují dosáhnutí Sprint Goalu (Šochová a Kunc, 2019).

Na konci každého Sprintu je důležité předchozí Sprint zhodnotit a získat cennou zpětnou vazbu k vylepšení procesů. K tomu slouží týmový meeting zvaný **Retrospektiva**, během kterého je každým členem týmu zhodnoceno, co považuje za podařené, v čem vidí smysl a rád by v tom pokračoval, co by naopak změnil nebo co zavedl nového. Cílem Retrospektivy je sesbírat veškerou zpětnou vazbu a pokusit se na základě poznatků další Sprint zlepšit. Tým dostává zpětnou vazbu také ze strany zákazníka. Po dokončení Sprintu je dokončená funkcionality prezentována zákazníkovi během **Review Meetingu** s cílem získat kvalitní zpětnou vazbu, zda byla hodnota pro zákazníka pomocí nové funkcionality dosažena (Šochová a Kunc, 2019).

3.2.2 Lean Software Development

Lean Software Development je agilní metodika vymezena Tomem a Mary Poppendieck, vycházející z principu Lean, se kterým původně přišla automobilka Toyota. Lean se obecně zaměřuje na maximalizaci hodnoty pro zákazníka při snaze minimalizovat plýtvání zdroji, které žádnou hodnotu nepřinášejí. Z pohledu na vývoj softwaru řeší pohled na rychlé dodání kvalitního softwarového produktu na základě požadavků a přání zákazníka (Bennett a Bowen, 2018; Kakar, 2023; Yadav et al., 2018).

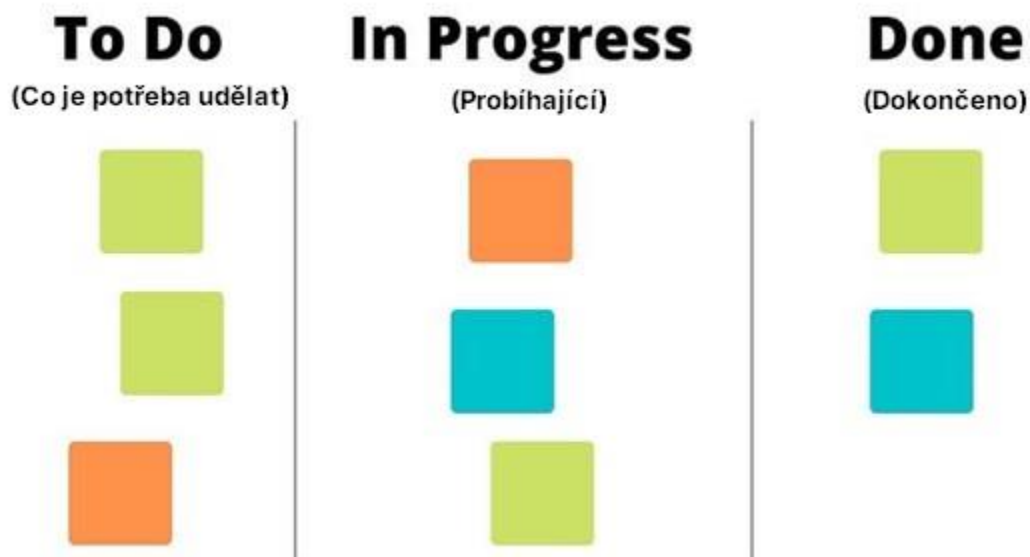
Lean Software Development metodika nahlížející na poskytování hodnoty zákazníkům prostřednictvím veškerých aktivit, které se v procesech vývoje nachází. Zejména klade důraz na rychlé dodání požadovaného výsledku a odstranění zbytečných funkcí, které nejsou pro produkt důležité (Bennett a Bowen, 2018). Snahou je tedy věnovat pozornost především takovým věcem, které zákazník chce a přinesou mu hodnotu. Naopak je vyžadována minimalizace těch věcí, které jsou nechtěné (Doležal, 2022).

Daný pohled platí i pro případy, kdy jsou nějaké aktivity připravovány příliš dopředu. Je nutné si uvědomit, že nemusí být vždy využity, ale již tomu byl věnován čas a další zdroje. Proto je žádoucí vykonávat danou aktivitu, až když je potřeba. Dále je důležité věnovat pozornost průběžné zpětné vazbě, díky které je vymezeno, co je sledováno významným. Lean Software Development zohledňuje rychlost vývoje jako klíčový

prvek, díky kterému lze získat zpětnou vazbu co nejdříve. Pokud je nutné o něčem rozhodovat, je vhodné počkat s finálním verdiktem až na poslední chvíli, kdy bude dostupných co nejvíce informací o celé záležitosti. Ke kvalitnímu a úspěšnému výsledku práce lze dojít v případě, pokud bude tým z případných vzniklých chyb ponaučen a bude se snažit v budoucnu těmto chybám vyvarovat (Šochová a Kunc, 2019).

3.2.3 Kanban

Další často zmiňovanou agilní metodikou je Kanban, kterou lze využít samostatně, nebo ji lze zkombinovat s ostatními metodikami. Kanban metodika nabízí postup pro efektivní správu pracovních toků a řízení procesů pomocí vizuálního znázornění. Metodika je charakteristická podobou nástěnky, prostřednictvím které jsou znázorněny položky projektu s určitou prioritou a jednotlivé pracovní toky zobrazující postup činností projektu. Kanban vychází z Lean metodiky, tedy snaží se omezit zbytečné procesy, předejít potenciálním překážkám a vylepšit slabá místa. Cílem je zajistit pro týmy lepší spolupráci vizualizací pracovního postupu a efektivně řídit veškeré pracovní procesy. (Bennett a Bowen, 2018; Doležal, 2022).



Obrázek 9: Kanban
Zdroj: upraveno dle Barda, 2022

Obrázek 9 znázorňuje jednoduchou podobu Kanbanu, která se od ostatních může mírně lišit. Důležité je zachovat princip fungování metodiky. Sloupce ohraničují jednotlivé činnosti a definují, v jakém jsou aktuálním stavu – zda jsou v přípravě, nebo se na nich již pracuje nebo jsou dokončeny. Sloupců může být mnohem více a mohou znázorňovat jakékoliv stavy postupu práce, záleží pouze na tom, co je pro projekt užitečné. Uvnitř sloupců se nachází položky, jež definují určitou činnost. Počet těchto položek pro daný sloupec může být omezen, aby nedošlo k nahromadění a přehlcení daného stavu. Položky jsou dle své aktuální situace postupně mezi sloupci přesouvány. Kanban zastává pravidla, že by měla být minimalizována rozpracovaná práce, snížena doba přechodu mezi jednotlivými sloupci a postup by měl být vizualizován (Doležal, 2022; Šochová a Kunce, 2019).

3.2.4 Extreme Programming (XP)

Agilní přístup k vývoji softwaru *Extreme Programming (XP)* má snahu o zajištění maximální hodnoty pro zákazníka z každého dne, kdy se na projektu pracuje. Klíčovým prvkem pro XP je zpětná vazba ze strany klienta, kterému je produkt v malých iteracích dodáván. Veškeré požadavky a připomínky klienta udávají týmu ve vývoji směr, aby mohl přinášet co největší hodnotu. Velmi důležitá je nepřetržitá komunikace mezi zákazníkem a týmem, která by měla jasná a stručná, aby byly veškeré nutné informace předány správně a zřetelně. Uvnitř týmu je vedena komunikace na denní bázi, při které je řešen návrh kódu a testování, včetně řešení překážek, které jim stojí v cestě (Šochová a Kunce, 2019; Teles, 2017).

Důležitá hodnota XP, kterou je nutné dodržovat, je jednoduchost. Při vytváření produktu pro zákazníka je zbytečné myslet dopředu, tedy utvářet si domněnky, jak by to zákazník mohl požadovat. Efektivní práci zajistí přístup k současnému stavu produktu a jeho problémech na základě časté zpětné vazby, které je nutno vyřešit. Klient si stanoví hodnotu, která mu bude produktem přinesena a na této hodnotě tým pomalými, ale jistými krůčky pracuje (Teles, 2017; Šochová a Kunce, 2019).

Aby dokázal tým efektivně spolupracovat, vyžaduje se vzájemný respekt mezi členy týmu a stejné vnímání hodnoty, tedy že tým přijímá a dodržuje vyslovená přání a požadavky ze strany klienta. Důležité je přistupovat k vývoji odvážně. Od vývojářů

a testerů se vyžaduje postavit se problémům čelem, neignorovat nepříjemnosti, které mohou během vývoje nastat, s kuráží přijmout veškeré změny a vnímat proces vývoje jako výzvu (Šochová a Kunce, 2019).

XP klade důraz na sepsání testovacích případů pro potenciálně rizikové oblasti před napsáním samotného kódu. Cílem testování je vyhnout se možným problémům, odhalit případná omezení a zajistit tak kvalitní a rychlý vývoj softwaru dle požadavků zákazníka (Afshari a Javdani Gandomani, 2022; Šochová a Kunce, 2019).

3.3 Vybrané digitální nástroje projektového řízení

V této podkapitole bude představen výběr nástrojů pro řízení projektů, které umožňují rychlejší komunikaci, efektivnější plánování, snadnější správu a přehled nad jednotlivými činnostmi. Výhodou těchto nástrojů je možnost spravovat projekty dálkově a sdíleně, bez nutnosti být s týmem v přímém osobním kontaktu (Özkan a Mishra, 2019).

3.3.1 JIRA

Softwarový nástroj JIRA je velmi populární nástroj pro efektivní správu a řízení projektů, správu a sledování úkolů a řízení procesů. Je využíván zejména z důvodu množství důležitých funkcí, které mají manažeři projektu a vývojáři k dispozici, navíc je možné v něm sledovat celý životní cyklus vývoje softwaru. V prostředí nástroje je možné nahlašovat a sledovat problémy a plánovat vývoj. JIRA nabízí svůj nástroj jak pro oblast vývoje softwaru, tak i například pro oblast marketingu nebo designu (Atlassian, 2024; Özkan a Mishra, 2019).

3.3.2 Trello

Nástroj pro správu projektů Trello je založen na metodice Kanban pro lepší přehlednost nad správou úkolů a činností. Uživatelské rozhraní je rozděleno na nástěnky, které reprezentují jednotlivé projekty. V rámci každé nástěnky jsou dostupné sloupce, které představují fáze projektu nebo činností. Mezi sloupci jsou

přenášeny karty představující jednotlivé aktivity. V každé kartě je možné přidávat kontrolní seznamy, přikládat přílohy nebo psát komentáře (Atlassian, 2023; Özkan a Mishra, 2019).

3.3.3 Asana

Asana umožňuje jednoduchou správu úkolů a komunikaci napříč všemi týmy. Nástroj nabízí přidávání jednotlivých činností, využití časové osy, kontrolu stanovených termínů a automatizaci pracovních procesů. Rovněž nabízí souhrnné reporty a přehled nad všemi vytvořenými projekty. Součástí nástroje je také Inbox, který shromažďuje veškerá oznámení ohledně aktualizací v rámci jednotlivých projektů a také lze skrze něj komunikovat napřímo se členy týmu (Asana, 2024; Özkan a Mishra, 2019).

3.3.4 Freelo

Český nástroj Freelo nabízí jednoduché rozhraní pro malé a střední podniky z různých oblastí. V uživatelském rozhraní lze přehledně spravovat projekty a checklisty pro celý tým, včetně možnosti komunikace. Umožňuje ukládat a efektivně spravovat soubory každého projektu a využít Kanban nástěnku. Zobrazuje přehled nad časovou osou jednotlivých úkolů a činností. Také je možné zobrazit souhrnné statistiky a přehledy ohledně projektů a členů týmu (Freelo, 2024).

4 Představení vybrané společnosti

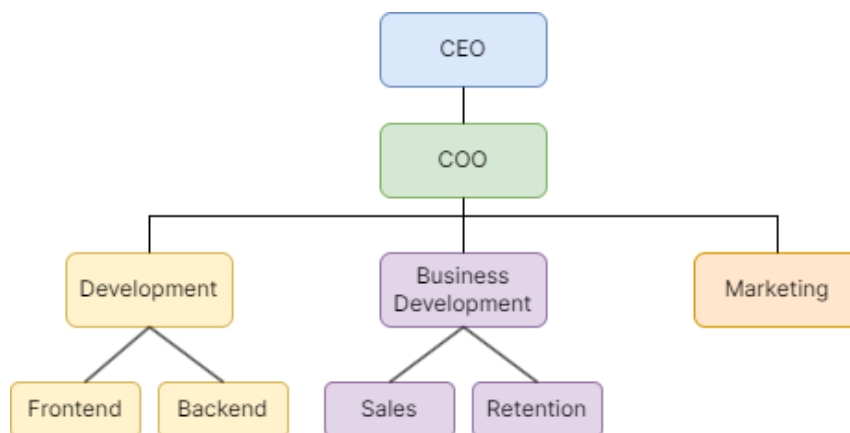
Praktická část této diplomové práce bude věnována návrhu a implementaci projektové optimalizace procesu vývoje webové aplikace nesoucí název Tanganica, jež reprezentuje klíčový produkt společnosti TGNCA s.r.o. (dále jen TGNCA). Společnost TGNCA je mladým a dynamickým startupem, fungujícím na trhu od roku 2020. Firma sídlí momentálně v Liberci. Za vznik nápadu a jeho realizaci stojí tři spoluzakladatelé – Bc. Filip Svárovský, Samuel Machat a Robert Tichý. Na Obrázek 10 je představen logotyp společnosti.



Obrázek 10: Logotyp společnosti
Zdroj: interní materiály společnosti

K březnu 2024 bylo v rámci organizace interně zaměstnáno 17 pracovníků, kteří se aktivně podílejí na denním provozu firmy. TGNCA rovněž využívá služeb externích partnerů, jejichž postavení sehrává roli zejména v plnění specifických úkolů, jako je například účetnictví nebo překladatelství. Organizační struktura, jež je znázorněna na Obrázek 11, je rozdělena do tří hlavních oblastí činností:

- Development (vývoj),
- Business Development,
- Marketing.



Obrázek 11: Organizační struktura firmy
Zdroj: vlastní zpracování

Největší podíl zaměstnanců je soustředěn v Development týmu, který se primárně zabývá vývojem a správou webové aplikace Tanganica. Tým vývojářů je rozdělen na dvě části – frontendový a backendový vývoj. Frontend vývojáři vyvíjejí funkční uživatelské rozhraní, jež musí být přehledné a intuitivní. Backend vývojáři se starají o data, jejich zpracování a bezpečnost. Taktéž pracují s dalšími provozními hledisky zajišťující funkčnost a stabilitu celé aplikace.

Druhým největším týmem je Business Development. Ten se dělí na Sales tým, jehož hlavní prioritou je akvizice nových uživatelů a rozvíjení nových partnerství, a Retention tým, který si klade za cíl udržení stávajících zákazníků a maximalizaci jejich spokojenosti.

Hlavní náplní marketingového týmu je správa a dohled nad kampaněmi uživatelů aplikace. Marketingové oddělení zajišťuje, aby byly kampaně správně optimalizované a přinášely uživatelům tržby. Dále realizuje marketingové aktivity, jejichž záměrem je propagace aplikace formou reklam nebo prezentační webové stránky k získání nových potenciálních uživatelů.

Autorka diplomové práce zastává pozici Chief Operating Officer (COO). Její hlavní povinností je zajištění plynulého chodu organizace, optimalizace stávajících procesů a zajištění efektivního rozvoje společnosti s ohledem na stanovenou vizi, misi a hodnoty.

Při rozvoji společnosti je zejména velmi důležité brát v úvahu startupové prostředí, ve kterém se TGNCA nachází. To se vyznačuje rychlým a flexibilním vývojem jak firmy, tak samotného produktu, a to na základě měnících se požadavků a nových potřeb okolí. Rovněž je startupové prostředí charakterizováno kreativním a inovativním přístupem k vývoji společnosti i produktu. Rychlost v tomto prostředí hraje velmi důležitou roli vzhledem k dynamickému a konkurenčnímu trhu, a z toho důvodu je právě tato vlastnost zohledňována při přístupu k vývoji produktu.

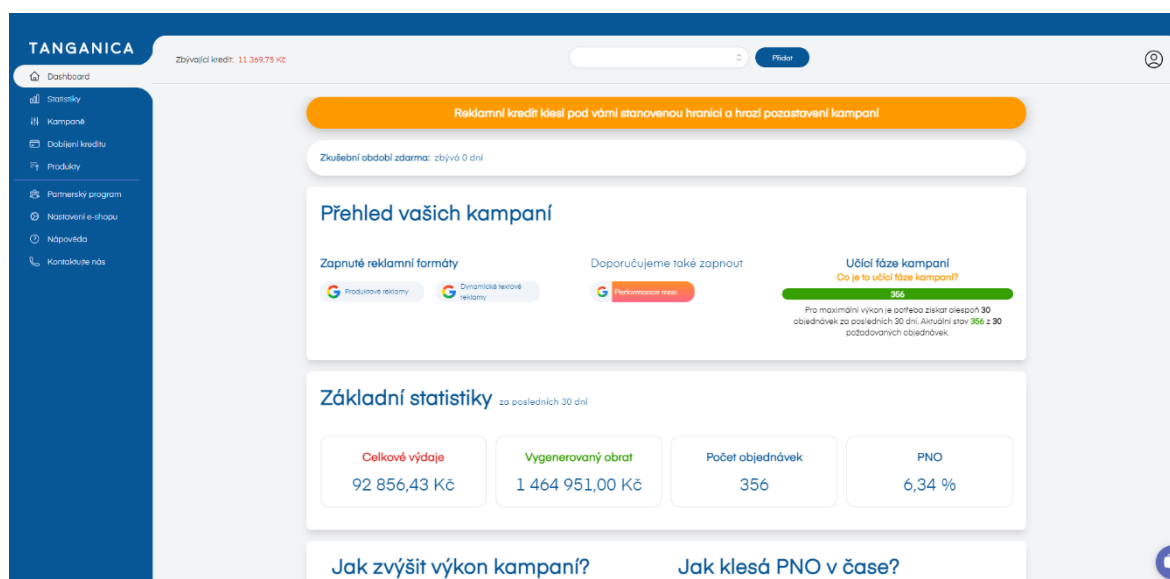
Momentální snahou společnosti TGNCA je přinášet na trh takový produkt, který co nejvíce usnadní e-shopům nastavení a správu reklamních kampaní v prostředí Google Ads a Meta Ads, a zároveň zpřístupní svět propagace i těm nejmenším e-shopům, kteří v této oblasti nemají žádné zkušenosti. Společnost zastává hodnoty

jednoduchosti, kvality a férovosti. Společným cílem firmy, ke kterému směřuje, je vytvoření takového prostředí pro své uživatele, ve kterém budou moci snadno analyzovat výkon svého e-shopu a další relevantní aspekty. V tomto prostředí jim bude rovněž poskytnuta příležitost vzdělávání a zlepšování se, stejně jako možnost propagace e-shopu s minimálním manuálním zásahem ze strany uživatele.

4.1 Produkt

Hlavním a jediným produktem společnosti TGNCA je webová aplikace Tanganica. Vzhled webové aplikace je zobrazen na Obrázek 12. Aplikace je přístupná a stejná pro všechny uživatele. Hlavní účel aplikace spočívá ve zjednodušení nastavení a spuštění reklamních kampaní na Googlu, Facebooku či Instagramu. Tanganica také zajišťuje automatickou správu spuštěných kampaní a jejich rozpočtů nebo poskytuje přehledný report statistik o výkonu reklam, a to v reálném čase. Ve svém prostředí momentálně nabízí ke spuštění tyto reklamní formáty:

- Produktové reklamy na Googlu (PLA – *Product Listing Ads*),
- Maximálně výkonné kampaně na Googlu (PRX – *Performance Max*),
- Dynamické reklamy ve vyhledávání na Googlu (DSA – *Dynamic Search Ads*),
- Dynamické produktové reklamy na Facebooku a Instagramu (DPA – *Dynamic Product Ads*).



Obrázek 12: Vzhled aplikace

Zdroj: vlastní zpracování

Aplikace poskytuje pro své uživatele spoustu funkcí, které byly vyvinuty za účelem zjednodušení jejího používání, zpříjemnění uživatelského zážitku a na základě vyslechnutých potřeb získaných ze zpětné vazby od uživatelů.

4.1.1 Uživatelé

Převážnou většinou uživatelů aplikace jsou majitelé nebo správci e-shopů. Pro ty je aplikace vhodná zejména, pokud nemají v oblasti nastavování a správy reklamních kampaní žádné zkušenosti a agentury nebo specialisté, kteří by se za ně kampaním věnovali, pro ně znamenají veliký výdaj, který si nemohou dovolit. Druhým opakem této skupiny jsou uživatelé, kteří zkušenosti s reklamními kampaněmi mají, ale nemají čas se jim věnovat, nebo chtějí tento čas věnovaný zdlouhavému nastavování a optimalizaci ušetřit.

Další skupinou uživatelů aplikace jsou agentury nebo jednotliví specialisté, kteří spravují několik e-shopů. Pro ně je připraven partnerský program s možností vyplácení provize. Pro tuto skupinu slouží aplikace jako usnadnění spravování všech e-shopů, jejich zpřehlednění a ušetření času, který mohou věnovat dalším klientům.

Uživatelé aplikace mají možnost spustit si vybrané reklamní kampaně pouhými pár kliknutími, bez nutného vytváření vlastních účtů v reklamních systémech a potřebných větších znalostí v oblasti online marketingu. Firma TGNCA si za využívání produktu účtuje provizi maximálně 3 % z tržeb e-shopu, které byly vygenerovány za pomoci Tanganicy. Uživatel má možnost využití měsíčního zkušebního období, ve kterém není tato provize účtována. K březnu 2024 byl počet registrovaných e-shopů v aplikaci přibližně 3000, z toho bylo aktivních, tedy aktuálně provozujících kampaně skrze Tanganicu, kolem 400 e-shopů.

4.1.2 Vývoj produktu

Vývoj aplikace probíhal téměř od samého počátku fungování společnosti. Nejdříve bylo za pomoci zjednodušeného reportu zjišťováno, zda je o produkt na trhu zájem. Jakmile se zájem potvrdil, započala tvorba jednoduché funkční podoby aplikace, která trvala necelé 2 měsíce. V tu dobu na aplikaci pracoval pouze jeden vývojář.

Aplikace byla z pohledu backendu kompletně naprogramována v programovacím jazyce PHP, a z pohledu frontendu v HTML a CSS. Později přišel do týmu nový vývojář věnující se čistě frontend vývoji. Ten frontendovou část aplikace přepracoval do JavaScriptového frameworku Vue.js a CSS frameworku Tailwind CSS. Tím webové aplikaci dodal novou vizuální podobu a zlepšené uživatelské rozhraní. S postupem času přirůstal počet uživatelů aplikace a s tím rostly nároky na hladký a bezproblémový provoz aplikace, s čímž byla spojena i její bezpečnostní stránka. Z těchto a mnoho dalších důvodů bylo ve firmě rozhodnuto, že bude aplikace ze strany backendu přepsána do frameworku .NET.

Na vývoji aplikace se podílí téměř všichni interní zaměstnanci ve firmě. Vývojáři vyvíjí produkt z hlediska programování a ostatní zaměstnanci přispívají do vývoje nápady na nové funkce či nová rozšíření, které přinesou uživatelům lepší zážitek z aplikace, nebo větší přidanou hodnotu. Uživatelé mají rovněž vliv na vývoj, a to především v podobě jejich požadavků a zpětné vazby k funkcím, nabízeným službám a uživatelskému rozhraní.

5 Analýza současného vývoje

Od počátku vývoje webové aplikace Tanganica se změnilo mnoho přístupů, jakými byl vývoj řízen. Vývoj byl mnohdy přizpůsobován situacím, které se v daném období naskytly. Zároveň nebylo přesně stanoveno, jaká pravidla nebo metody projektového řízení vývoje budou implementovány. Vždy se vycházelo z nejlepšího úsudku zaměstnanců. To bylo společností shledáno jako špatný přístup, který způsoboval spíše převahu problémů oproti rychlému růstu, který je pro úspěšnou firmu ve startupovém prostředí potřebný.

V době, kdy začal přibývat počet vývojářů, byla vytvořena role *Chief Technology Officer* (CTO) pro lepší a přehlednější management. Hlavní rolí CTO je dohlížet a řídit práci vývojářů dle požadavků, které vycházejí z potřeb okolí nebo samotného softwaru. Většinou jsou požadavky od uživatelů nebo zaměstnanců komunikovány právě skrze CTO. Za celý tým vývojářů se účastní Management Meetingů, které se konají za účelem pravidelných schůzek manažerů, COO a CEO, a během nichž se probírají témata ohledně aktuální situace společnosti, důležitých rozhodnutí a současných problémů. Samotný CTO má pak pravidelná setkání s týmem vývojářů, která jsou zaměřena na současnou práci vývojářů, na které momentálně pracují, jsou řešeny vzniklé problémy při vývoji a další aspekty, které jsou potřeba společně vyřešit.

Nápady nebo požadavky na nové funkcionality či rozšíření webové aplikace vycházely ze společných setkání celého týmu, ať už pravidelných na týdenní bázi, či při uzavírání každého kvartálu. Ve většině případů si vývojáři poznamenali poznámky a poté na dané funkcionality nebo rozšíření začali pracovat s užším počtem lidí z jiných oddělení, převážně z důvodu detailnějšího zadání a popisu úkolu. Vyskytly se rovněž případy, kdy zadání nového funkčního rozšíření přišlo od vedení společnosti, bez většího brainstormingu s ostatními zaměstnanci firmy.

Ke každodenní komunikaci s týmem vývojářů je využíván komunikační kanál Slack. Ten zajišťuje výměnu informací mezi zaměstnanci online. Většina vývojářů pracuje z kanceláře v Liberci, zde tedy probíhá komunikace i osobně. Na žádost CTO byla vytvořena nástěnka požadavků a ohlašování chyb. Nejdříve byla vytvořena v nástroji Trello a posléze byla přesunuta do Asany, ve které je nástěnka využívána

i momentálně. Nástěnka byla vytvořena z důvodů lepší přehlednosti, uchování veškerých záznamů a změn, efektivnější komunikace a snazšího řízení karet dané nástěnky. Nástěnka je přístupná pro všechny zaměstnance, kteří zde mohou oznamovat nedostatky nebo žádosti o jakékoliv vylepšení. Vývojáři mezi sebou komunikují jak osobně, tak skrze komunikační kanály Slack či Discord. Rovněž využívají platformu GitHub, do které jsou importovány kartičky z nástěnky v Asaně. Na této platformě mají možnost mezi sebou sdílet kód, spolupracovat na úkolech nebo sledovat postup ve vývoji.

Po tom, co bylo ve společnosti rozhodnuto, že bude webová aplikace Tanganica přepsána do .NETu, se pozastavil veškerý vývoj nových funkcí a rozšíření, pokud to nebylo shledáno za nutné. Odhadovaná doba přepsání celé aplikace byla 1 rok. Reálná doba přepsání překročila rok a půl a přepsání bylo finálně dokončeno na konci března 2024. Během fáze přepisování aplikace bylo odhaleno mnoho nedostatků, jak v řízení celého procesu, tak v komunikaci nebo v neúplnosti zadání. Proces přepsání celé webové aplikace byl pro firmu velkou výzvou, ze které si vzala mnoho ponaučení a návrhů na zlepšení pro budoucí projekty.

5.1 Analýza vývoje v kontextu startupového prostředí

Na základě sledování procesu vývoje po dobu dvou a půl roku byla autorkou diplomové práce provedena SWOT analýza, jež zkoumá silné stránky (*Strengths*), slabé stránky (*Weaknesses*), příležitosti (*Opportunities*) a hrozby (*Threats*) vývoje webové aplikace Tanganica s ohledem na startupové prostředí, ve kterém se společnost nachází. Pro zpřehlednění byla SWOT analýza vizualizována pomocí Obrázek 13.

| | |
|--|---|
| <p style="text-align: center;">S</p> <p style="text-align: center;">Unikátní produkt</p> <p style="text-align: center;">Tým Inovativnost</p> <p style="text-align: center;">Finanční zdroje Flexibilita</p> | <p style="text-align: center;">W</p> <p style="text-align: center;">Nedostatečná zadání</p> <p style="text-align: center;">Chybějící metodika Pomalý vývoj</p> <p style="text-align: center;">Přenos informací Nedůležité věci</p> |
| <p style="text-align: center;">O</p> <p style="text-align: center;">Nástroje pro usnadnění</p> <p style="text-align: center;">Umělá inteligence</p> <p style="text-align: center;">Otevřený trh</p> | <p style="text-align: center;">T</p> <p style="text-align: center;">Ztráta uživatelů</p> <p style="text-align: center;">Konkurence</p> <p style="text-align: center;">Časté změny</p> |

Obrázek 13: SWOT analýza
Zdroj: vlastní zpracování

5.1.1 Silné stránky

Mezi silné stránky společnosti spadá produkt, který je svým způsobem unikátní. Jedná se o aplikaci, prostřednictvím které firma nabízí své služby a jedinečné funkce pro zjednodušení výkonnostního marketingu e-shopu. Dle posledního průzkumu provedeného společností mezi uživateli vyšlo najevo, že je aplikace přínosná, ovšem je potřeba stále přinášet nové funkce a rozšíření, které danou problematiku ještě více zjednoduší. Tedy základ jako takový má mezi uživateli hodnotu, kterou je potřeba nadále rozvíjet. Produkt se výrazně odlišuje od ostatních služeb a nepřímé konkurence díky svým inovativním funkcím a přístupu k dané problematice. Jeho hlavním cílem je zjednodušit výkonnostní marketing pro e-shopy, což ho činí jedinečným a efektivním nástrojem.

Tým představuje základní kámen úspěchu v rámci společnosti. Firma jej považuje za velmi silnou stránku, zejména díky ambicióznímu kolektivu. V každém jednotlivci se nachází jedinečné schopnosti a dovednosti, za pomoci kterých se nebojí přijmout jakoukoliv výzvu. V týmu je každý schopen pracovat nezávisle bez nutnosti přímého dohledu nadřízeného. Přínosem v týmu je rovněž efektivní zvládnutí spolupráce mezi jednotlivými členy, díky které dokáže tým pracovat jako celek samostatně.

Vzhledem ke startupovému prostředí, ve kterém se společnost nachází, je její velmi silnou stránku inovativní produkt a inovativní myšlení zaměstnanců firmy. Pro úspěch startupu je zásadní přinášet inovativní a originální nápady na vylepšení a rozšíření produktu. Ve firmě se příležitostně konají brainstormingová setkání, při kterých jsou rozebírány nové myšlenky a nápady, které mají potenciál posunout produkt i firmu na vyšší úroveň. Každý nápad je vítán a každý z firmy má možnost svými originálními nápady produkt vylepšit či rozšířit.

Jednou z dalších vnímaných silných stránek je flexibilita firmy a produktu. Společnost přijmula fakt, že startupy se často nachází ve velmi proměnlivých situacích, kterým se musí přizpůsobit. Mezi takové situace může patřit potřeba adaptabilního vývoje, potřeba změn na základě zpětných vazeb od uživatelů k nové či stávající funkci nebo omezené zdroje. Tento přístup je aplikován jak při vývoji produktu, tak při organizaci práce zaměstnanců.

Startupy se mnohdy dostanou do bodu, kdy jim dojdou veškeré finanční zdroje a musí tak vývoj produktu pozastavit či omezit jiné zdroje, aby ve vývoji mohli pokračovat. Jednou z možností, jak zvýšit kapitál pro rozvoj startupu je získání investice od investora. Společnosti TGNCA se investici podařilo v polovině roku 2023 úspěšně obdržet a tím získat dostatek potřebných finančních zdrojů, které je možné využít na veškeré aktivity, kterými je mladý startup rozvíjen.

5.1.2 Slabé stránky

Mezi nedostatky společnosti lze zařadit nedostatečně specifikovaná zadání, která se objevují ať už v požadavcích na vylepšení stávající aplikace, nebo v požadavcích na nová rozšíření. Z toho důvodu velmi často dochází k mylné představě zadávajícího a vývojáře, který následně změnu provádí. To má za následek zpomalení vývoje vzhledem k nutnosti výsledek předělat.

Dalším významným problémem, se kterým se firma potýká, je zpomalený vývoj produktu a s tím související nedodržování stanovených termínů pro dokončení zadání. Pro startupy je obecně zásadní mít rychlý vývoj, a pokud k urychlení nedojde, lze očekávat nepříjemné výzvy ze strany konkurence.

Provázanou slabou stránkou je chybějící metodika, podle které by měly být vývoj a projekty správně řízeny. To znamená, že při vývoji produktu nejsou stanovena žádná konkrétní pravidla a postupy. Následkem absence metodiky dochází k prodlužování termínů dodání, nesplnění očekávání ze strany zadavatele a zbytečně dlouhému procesu vývoje.

Významně vnímanou slabou stránkou při vývoji je přenos informací mezi vývojáři a dalšími zaměstnanci, a jejich zachování. Častým problémem je situace, kdy se probírá ať už zadání či konkrétní kroky ve vývoji specifického úkolu mezi skupinou vývojářů a zaměstnanců, ale již není nikde zaznamenáno, jaké padlo rozhodnutí či vybráno řešení. Poté nastávají zmatky, kdy není známo, kdo o dané věci rozhodl a z jakého důvodu tomu tak je. S tím souvisí i zodpovědnost za dané komplikace. Jelikož více lidí může rozhodovat o důležitých krocích a rozhodnutí není nikde zaznamenáno, není jisté, kdo za dané rozhodnutí nese odpovědnost a za kým se lze s případnými otázkami obrátit.

Další slabinou, která zpomaluje vývoj produktu a růst startupu, je nedostatečné přemýšlení nad kontextem, tedy nejsou brány v úvahu opravdové požadavky a potřeby uživatelů, aktuální situace a trendy na trhu či není zohledněna situace s konkurencí. Převážně jsou vymyšleny nové funkce a nová rozšíření na základě domněnek, které nejsou ničím podloženy. Dochází tak ke zbytečně vynaloženému času k vývoji něčeho, co není většinou uživatelů využito.

5.1.3 Příležitosti

Pro společnost TGNCA a její vývoj aplikace je velkou příležitostí stále narůstající zájem o digitální nástroje, které ostatním lidem usnadní práci nebo čas. Z toho důvodu je klíčové, aby společnost neustále usilovala o zjednodušování své aplikace, vylepšování svých služeb a snažila se uživatelům poskytnout takové uživatelské prostředí, které je přehledné a intuitivní. To může zajistit nárůst spokojených uživatelů, kterým aplikace usnadní práci s výkonnostním marketingem.

V poslední době dochází k velkému nárůstu využívání umělé inteligence v mnoha odvětvích, včetně vývoje. Využití umělé inteligence může přinést mnoho výhod a zjednodušit mnoho procesů. Z toho důvodu může být umělá inteligence pro tým

vývojářů skvělou příležitostí, jak vývoj urychlit, usnadnit a zefektivnit. Také lze umělou inteligenci zakomponovat do funkcí pro uživatele, což může podpořit jednoduchost aplikace bez zbytečné námahy cokoliv manuálně nastavovat.

Výhodou webových aplikací je jejich dostupnost. Jelikož se jedná o produkt dostupný online, lze k němu přistupovat odkudkoliv a kdykoliv. Pro společnost to je příležitost expandovat s produktem na další trhy, rozšiřovat tak základnu svých uživatelů a zvyšovat povědomí o aplikaci.

5.1.4 Hrozby

Jednou z klíčových hrozeb, která může ohrozit vývoj produktu, je ztráta uživatelů. Tato nežádoucí situace může nastat například z důvodu vytváření aplikace, která nesplňuje požadavky a potřeby uživatelů, nebo aplikace, která má nedostatečnou úroveň kvality. Také může nastat přechod uživatelů k produktům konkurence nebo se mohou změnit jejich preference, což může zapříčinit ztrátu zájmu o stávající produkt. Z tohoto pohledu je nutné věnovat pozornost zejména uživatelům, jejich potřebám, přáním a zájmům. Pravidelným získáváním jejich zpětné vazby lze vyzorovat, jaké preference momentálně mají a jakým způsobem produkt vyvíjet tak, aby byla splněna všechna jejich očekávání.

Ohrožení může přijít také ze strany konkurence. Například by mohlo dojít k situaci, ve které by měla konkurenční společnost větší finanční zdroje a tím si mohla dovolit zaměstnat větší počet vývojářů. Tito vývojáři by mohli odvádět kvalitnější a rychlejší práci ve vývoji produktu, který by mohl poté konkurovat produktu společnosti TGNCA a dosáhnout většího úspěchu. Proti hrozbě konkurence se lze chránit vývojem produktu, který reflektuje potřeby a přání uživatelů a který jim nabízí jedinečnou hodnotu. Dále je důležité pravidelné vymýšlení a rozvíjení inovativních nápadů implementovaných do aplikace, díky kterým je udržován zájem uživatelů aplikaci využívat.

Další hrozbu pro vývoj představují časté změny v oblasti technologií a online marketingu, stejně jako měnící se preference uživatelů aplikace. Je nezbytné věnovat pozornost průběžným aktualizacím softwaru a systémů, sledovat aktuální trendy a flexibilně reagovat na veškeré změny v oblastech, které se společnosti dotýkají.

5.2 Požadavky na zlepšení

Na základě SWOT analýzy vývoje webové aplikace byly analyzovány a představeny slabé stránky, u kterých je požadováno jejich omezení či případně úplné odstranění. Společností byly definovány nové hlavní zásady, a budoucí snahou je tyto zásady při vývoji aplikace implementovat a dodržovat. Jedná se o jednoduchost, rychlost a zaměření převážně na důležité věci.

Ze strany společnosti je požadováno:

- zlepšení procesu formulace a specifikace zadání,
- urychlení vývoje webové aplikace,
- nastavení postupů a metodik pro efektivnější vývoj,
- zlepšení komunikace a přenosu informací,
- větší zaměření na důležitost a kontext vyvíjené funkce či rozšíření.

V následující kapitole bude představen návrh autorky diplomové práce pro optimalizaci procesu vývoje webové aplikace, který má za cíl realizaci požadavků na zlepšení ze strany společnosti. Návrh bude zaměřen na minimalizaci či úplné zamezení identifikovaných slabých stránek, které ohrožují produkt i celou společnost.

6 Návrh optimalizace procesu vývoje

V této kapitole bude autorkou diplomové práce představen návrh optimalizace procesu vývoje webové aplikace Tanganica, jehož cílem je odstranění nebo alespoň minimalizace identifikovaných slabých stránek vývoje.

Jako první krok je autorkou diplomové práce pro společnost TGNCA navrhováno začlenění všech fází životního cyklu vývoje softwaru. Dodržováním těchto fází je možné zajistit strukturovaný a systematický přístup k obecnému vývoji webové aplikace. Pomocí etap životního cyklu vývoje softwaru je jasně definován směr vývoje, minimalizován nebo úplně eliminován chaos a nejistota v procesu vývoje a snižováno riziko chyb, které mohou v průběhu vývoje nastat. Pro tým je jasně stanoveno, jakými kroky vývoj doposud prošel, jaké činnosti jsou následovány a v jaké fázi se momentálně vývoj nachází.

Zejména implementací první fáze životního cyklu vývoje, která se věnuje formulaci nápadů a problémů, které má vývoj řešit, je možné předejít neustálému vymýšlení nových konceptů, které nejsou zasazeny do kontextu a tím není jasné, zda jsou nové funkce či rozšíření vůbec potřebné. Fáze formulace nápadů by byla věnována důkladnému průzkumu mezi uživateli aplikace, zda je nově navrhovaná funkce žádaná. Dále je nezbytné pravidelné získávání zpětné vazby uživatelů, ať už formou dotazníků či přímého telefonického dotazování na již vydané funkce a rozšíření, u kterých se může vyskytnout jejich nepochopení či jiný zásadní problém, který by mohl ovlivnit spokojenost uživatelů s používáním aplikace.

Na základě požadavků formulovaných společností navrhuje autorka diplomové práce implementaci agilního přístupu k řízení projektů vývoje, konkrétně zavedení metodiky SCRUM. Agilní přístup byl zvolen z důvodů pružnosti, rychlosti a schopnosti rychle se přizpůsobit náhlým změnám. Tyto vlastnosti jsou pro vývoj webové aplikace Tanganica nezbytné. Řízení projektů vývoje dle SCRUM by vyřešil požadavek společnosti pro nasazení postupu a metodik pro efektivnější vývoj. Výhodou implementace metodiky SCRUM jsou jasně definované týmové role, pomocí kterých je zajištěno jednoznačné rozdělení odpovědností a činností. Díky tomuto rozdělení je zřejmé, kdo nese zodpovědnost za jednotlivá rozhodnutí. Tím je eliminován zmatek v případech vzniklých situacích žádajících si zodpovězení dotazů.

Pro daný projekt bude definován Product Backlog, který obsahuje seznam všech požadavků na funkci nebo rozšíření. Všechny položky Backlogu jsou jasně definované a seřazené dle priority stanovené Product Ownerem. Pomocí přesně vymezeného zadání je minimalizována nejistota vývojářů ohledně toho, co mají přesně implementovat.

Jednotlivými Sprints bude zajištěno pravidelné inkrementální dodávání funkčních částí produktu, což může urychlit vývoj daných funkcí nebo rozšíření. Tyto krátké a časově vymezené úseky jsou určeny k plnění položek ze Sprint Backlogu, jež jsou specifické pro daný Sprint.

V rámci Sprintů se budou konat denní Standupy, které přináší pravidelnou komunikaci mezi vývojáři a Scrum Masterem. Pravidelným kontaktem může být zajištěno včasné odhalení problémů, které by později mohly přinést ve vývoji komplikace. Bude vytvořen prostor pro otevřenou komunikaci, která umožní identifikovat a vyřešit jakékoliv nejasnosti ohledně zadání nebo požadavků, které by se mohly v průběhu Sprintu objevit.

Podpůrnou pomoc nabídne softwarový nástroj Asana poskytující vhodnou platformu pro řízení projektu vývoje. Asana umožňuje efektivní správu a sledování plnění činností s přesně definovanými a zaznamenanými požadavky, jež jsou dostupné celému týmu. Díky práci v nástroji bude každý člen týmu dostatečně informován o průběhu projektu a jeho aktuálním stavu.

6.1 Výběr projektu a návrh jeho řízení

Společnost na základě předchozích zkušeností a špatných rozhodnutí přehodnotila svůj přístup k vývoji nových funkcí a rozšíření, a začala aktivně naslouchat potřebám a požadavkům uživatelů. Namísto vytváření funkcionalit, které nemusí mít skutečné využití, a tedy by byly zbytečně vyčerpány zdroje na vývoj těchto funkcionalit, začala společnost sbírat zpětnou vazbu od uživatelů prostřednictvím telefonické komunikace nebo rozesílaných elektronických dotazníků. Tato zpětná vazba je pro společnost velmi cenná, jelikož slouží jako zdroj inspirace pro nové nápady na funkce, rozšíření nebo nová vylepšení funkcí stávajících. S ohledem na připomínky od

uživatelů bylo vymyšleno mnoho projektů, které má společnost v plánu do aplikace implementovat za účelem jejího zlepšení.

Autorce diplomové práce byl společností umožněn volný výběr projektu pro otestování. Bohužel, z důvodu probíhajícího přepisu celé aplikace z PHP do .NETu byla autorka diplomové práce nucena počkat, až bude přepis dokončen, jelikož v průběhu přepisování nemohly být zahájeny a implementovány žádné nové projekty. Termín dokončení přepisování byl oproti plánovanému nasazení prodloužen o několik měsíců, a nakonec proběhlo spuštění nové aplikace na konci března 2024. Z toho důvodu byl pro otestování návrhu optimalizace procesu vývoje vybrán menší projekt zaměřující se na expanzi.

Projekt Expanze, jehož vytvoření vzešlo z požadavků uživatelů aplikace, se zaměřuje na umožnění uživatelům aplikace inzerovat do dalších zemí, které se nachází v seznamu zemí pro Služby porovnávání zboží (CSS) na Googlu, a které aplikace prozatím ve své nabídce zemí nenabízí. Nově by uživatelé měli mít možnost kromě Česka, Slovenska, Rumunska, Itálie, Maďarska a Polska inzerovat své produkty skrze webovou aplikaci Tanganica také v Rakousku, Belgii, Dánsku, Finsku, Francii, Německu, Řecku, Irsku, Nizozemsku, Norsku, Portugalsku, Španělsku, Švédsku, Švýcarsku a ve Velké Británii.

Projekt bude kompletně řízen v softwarovém nástroji Asana. Tento nástroj byl vybrán z toho důvodu, že v něm společnost již nějaké projekty v minulosti vedla a je využíván na denní bázi zejména v závislosti zaznamenávání vzniklých problémů aplikace a urgentních požadavků směrem na oddělení vývojářů. V Asaně bude vytvořena nástěnka nového projektu, ve které budou vizualizovány a zaznamenány prostřednictvím karet jednotlivé položky Product Backlogu. Nástěnka daného projektu bude sdílena se všemi členy týmu, tudíž do ní bude mít každý přístup. V jednotlivých kartách bude detailně popsáno zadání úkolu a případně přiložené potřebné dokumenty, obrázky a další podklady. V kartách lze přidávat komentáře, tedy se bude jednat o další zdroj komunikace, právě skrze tyto karty. Každá položka Product Backlogu, reprezentovaná kartou v Asaně, bude mít svého Ownera, který na jejím vypracování bude pracovat. Ten je za kartu, respektive dokončení úkolu zodpovědný a veškeré případné dotazy budou řešeny převážně s ním.

Projekt započne svou realizaci tím, že bude naplánován. To zahrnuje vytvoření týmu, který se bude projektu věnovat, a vytvoření Product Backlogu, který bude obsahovat všechny funkcionality a požadavky pro daný projekt, které budou seřazené dle priorit ze strany zákazníka. Zákazníkem je v této situaci vedení společnosti, které na základě sesbírané zpětné vazby projekt nadefinovalo a rozhodlo o jeho vytvoření. Tým se bude skládat z vybraných vývojářů a zaměstnanců z marketingového oddělení, dále bude definována role Product Ownera a Scrum Mastera. Položky v Product Backlogu budou týmem detailněji rozebrány, průběžně upřesňovány a aktualizovány dle potřeby. Jelikož se jedná o menší projekt, bude rozdělen pouze do dvou týdenních Sprintů.

Na začátku každého Sprintu proběhne jeho naplánování s týmem, konkrétně bude vytvořen plán náplně daného Sprintu a jeho Sprint Goal, kterého se bude tým snažit během Sprintu dosáhnout. Vybrané položky z Product Backlogu budou pro daný Sprint přesunuty do Sprint Backlogu. Ve Sprint Backlogu se ocitnou úkoly, které budou týmem plněny a dokončeny během daného Sprintu.

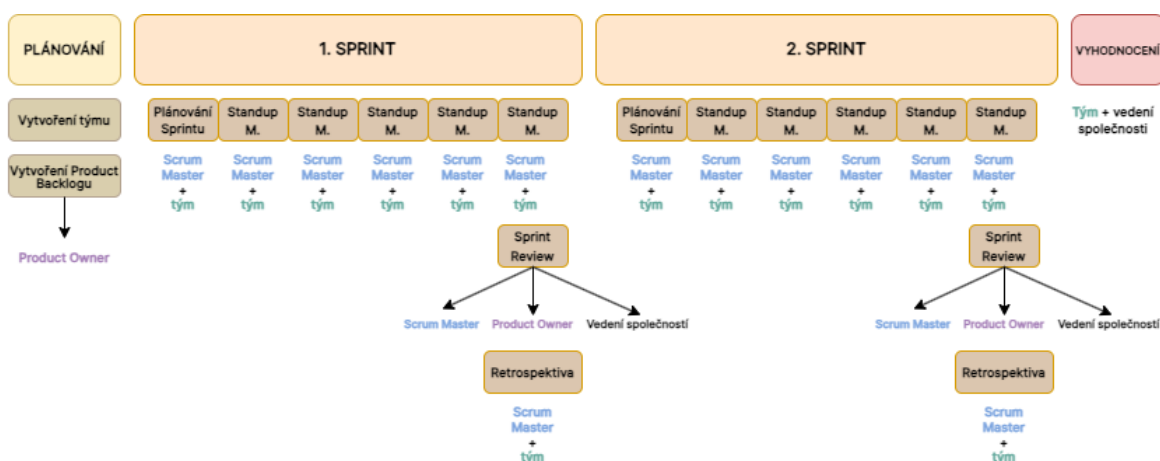
V průběhu daného Sprintu se budou konat denní Standup Meetingy, za pomoci kterých bude zajištěna pravidelná komunikace členů týmu. Na těchto setkáních se bude řešit konající se průběh Sprintu a případně zde vznikne prostor pro řešení problémů, pokud nějaké vzniknou. Na Standup Meetingcích bude sdílána Scrum Masterem nástěnka v Asaně. Autorka diplomové práce by ráda pro efektivnější práci s nástěnkou a sledování řízení úkolů zakomponovala prvky přístupu Kanban, tedy je navrhováno vytvoření sloupců v nástěnce, kde by jeden sloupec představoval seznam úkolů čekající na rozpracování ve Sprint Backlogu, následující sloupec by představoval seznam činností, které by byly momentálně vypracovávány a v posledním sloupci by se řadily úkoly, které by byly již dokončeny. Na každém Standup Meetingu by byl tento průběh diskutován, co za úkoly bylo dokončeno, co dokončit zbývá a zda se neobjevily nějaké překážky v dokončení.

Jakákoliv týmová komunikace, ať už v rámci prvotního plánování nebo Standup Meetingů, bude probíhat skrze platformu Google Meet, a to z toho důvodu, že někteří členové týmu nepracují fyzicky z kanceláře, ale nejčastěji ze svých domovů. Dále bude pro potřebu komunikace možností využití firemního komunikačního kanálu Slack.

Na konci každého Sprintu bude funkční část dokončené práce prezentována Product Ownerovi a vedení společnosti prostřednictvím Review Meetingu. Zde bude vedením zhodnoceno, zda je za něho zadání splněné či se vyskytly nějaké nedostatky, které je nutné opravit. Dále bude poskytnuta zpětná vazba ze strany vedení, na jejímž základě mohou vzniknout nové požadavky vyplývající z dodané funkční části. Na základě získaných poznatků budou naplánovány další kroky pro následující Sprint. Plánem je na konci každého Sprintu nasadit funkční část dokončené práce na produkci.

Kromě Review Meetingu bude rovněž uskutečněna Retrospektiva, která umožní Scrum Masterovi, vývojářům a dalším zapojeným členům týmu zhodnotit práci na projektu. Bude vyhodnocováno, co šlo dobře a zda se během Sprintu vyskytly nějaké problémy. Taktéž může být navrženo, co by mohlo být do dalších Sprintů vylepšeno, případně budou naplánovány akční kroky pro implementaci změn nebo zlepšení.

Po dokončení všech naplánovaných Sprintů se uskuteční setkání se všemi zapojenými členy do projektu a s vedením společnosti. Během tohoto setkání bude zhodnocen celkový průběh projektu a způsob jeho řízení. Autorkou této diplomové práce bude sesbírána zpětná vazba, kterou použije k vyhodnocení svého návrhu optimalizace procesu vývoje webové aplikace a případně k doporučení budoucích vylepšení. Na následujícím Obrázek 14 je vizualizován návrh průběhu projektu Expanze, včetně plánovaných setkání a zapojených členů.

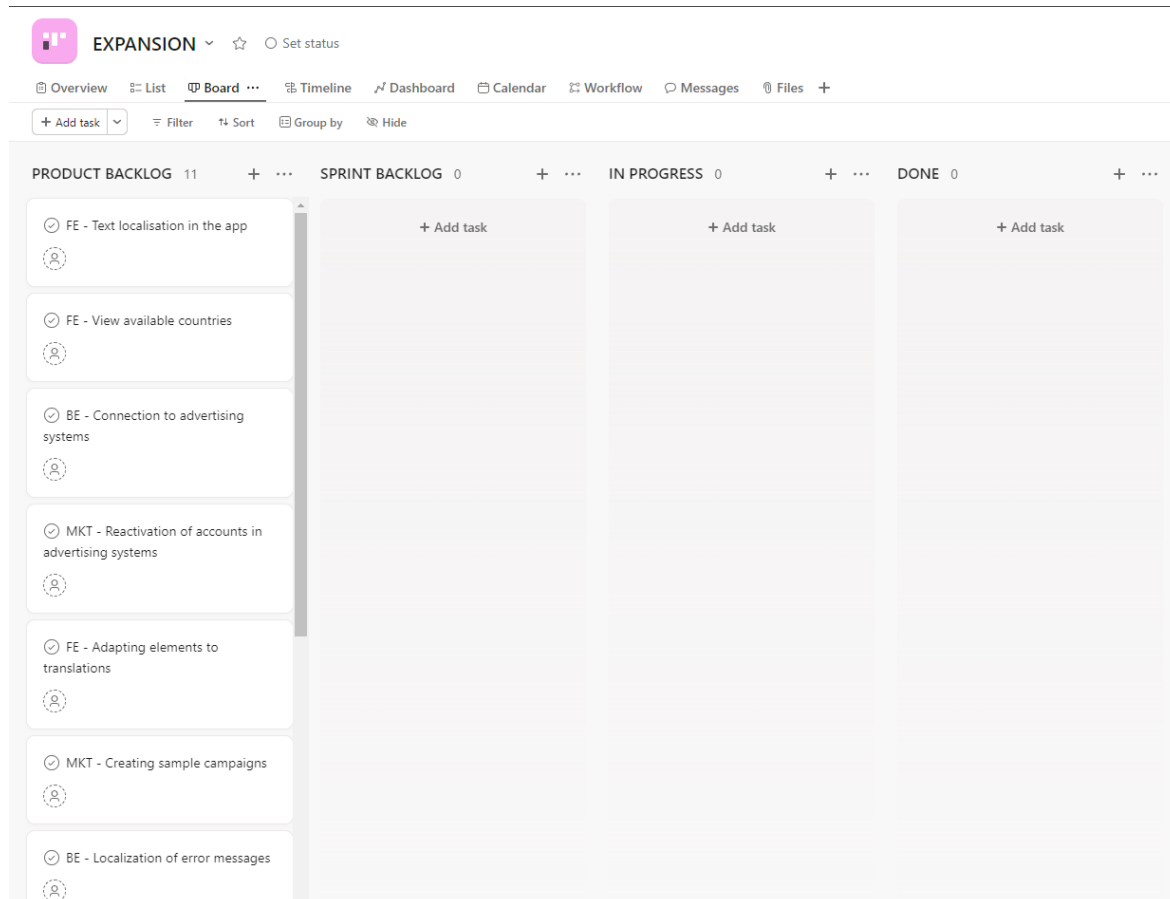


Obrázek 14: Návrh průběhu projektu
Zdroj: vlastní zpracování

6.2 Implementace návrhu

Tato kapitola bude zaměřena na realizaci návrhu optimalizace procesu vývoje webové aplikace, jenž byl společnosti předložen autorkou této diplomové práce. V kapitole bude podrobně popsáno, jak byl projekt od jeho počátku skutečně řízen, jak probíhal proces plánování Sprintů, vytváření Product Backlogu a přidělování úkolů členům týmu, kteří na projektu pracovali. Bude rozebrána komunikace během projektu a rovněž realizace činností, které byly pro projekt stanoveny.

Projekt započal tím, že byly rozděleny příslušné role Product Ownera a Scrum Mastera. Autorka diplomové práce zaujala roli Scrum Mastera, aby mohla být s týmem v denním kontaktu, pozorovat průběh projektu a zaznamenávat veškeré poznatky. Vedení společnosti, které zaujalo roli zákazníka, definovalo Product Ownerovi požadavky a výstupy, které od projektu očekává. Na základě těchto požadavků Product Owner připravil první položky Product Backlogu. V Asaně byla pro zaznamenání veškerých informací založena nová nástěnka, která lze vidět na Obrázek 15. V této nástěnce byly vytvořeny čtyři sloupce. První sloupec sloužil pro uchování položek Product Backlogu. Další sloupec obsahoval položky z Product Backlogu pro daný Sprint. Poslední dva sloupce sloužily pro sledování průběhu projektu. Jeden z nich zaznamenával položky, na kterých se momentálně pracovalo, a do posledního sloupce byly přesouvány položky, které byly za daného člena týmu dokončeny. Do karet, které představovaly dané položky Product Backlogu, bylo Product Ownerem sepsáno zadání úkolu, případně dodané další podklady potřebné pro jeho vyhotovení. Snahou bylo zadání co nejvíce konkretizovat, aby nenastalo jeho špatné pochopení.



Obrázek 15: Příprava Product Backlogu v Asaně

Zdroj: vlastní zpracování

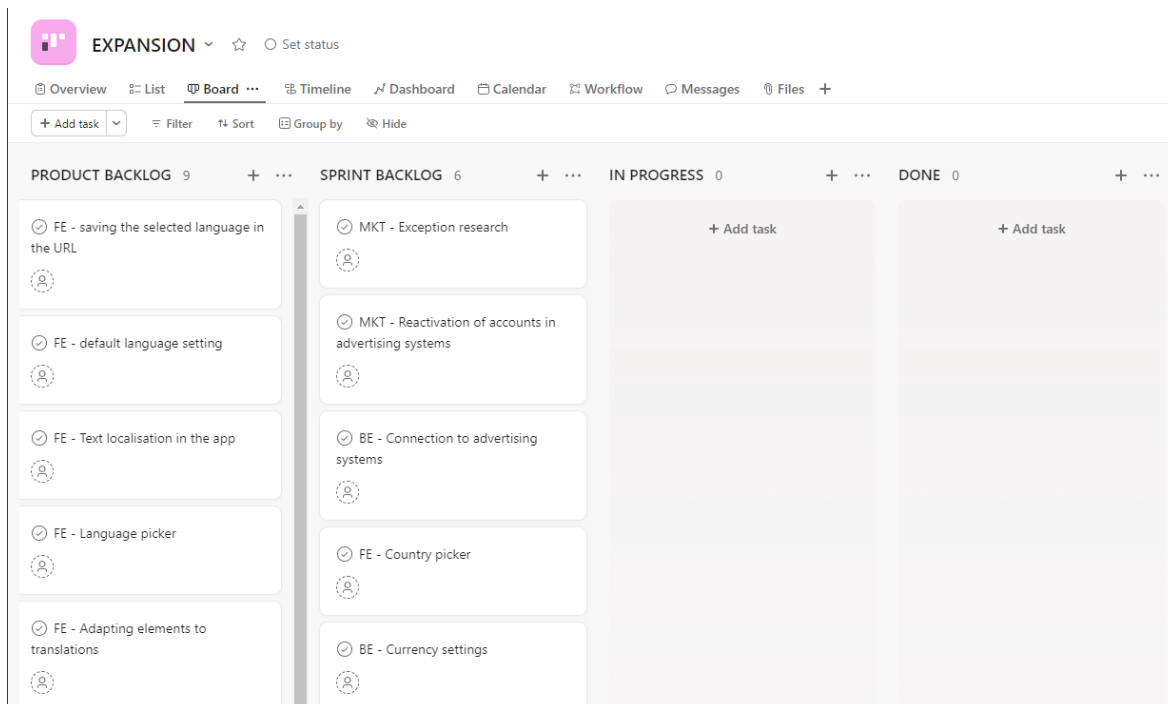
Následně proběhlo setkání Scrum Mastera s Product Ownerem, jehož součástí bylo naplánování počtu Sprintů, jejich obsah a definování jejich Sprint Goals (cíle Sprintu). Pro projekt Expanze byly naplánovány konkrétně dva Sprints. První Sprint byl zaměřen na rozšíření nabídky zemí pro inzerci a jeho cílem bylo: „Umožnit uživatelům aplikace vybrat si ze všech dostupných zemí v rámci Služby porovnávání zboží (CSS), ve kterých chtějí inzerovat své produkty“. Druhý Sprint se soustředil na jazykovou lokalizaci celé aplikace, s cílem zajistit, aby uživatelé z nových cizích zemí, kteří by si aplikaci chtěli vyzkoušet, dokázali aplikaci snadno porozumět.

Ve fázi přípravy před začátkem prvního Sprintu se uskutečnilo setkání Scrum Mastera s celým oddělením vývojářů. Byly představeny jednotlivé položky Product Backlogu a diskutovány konkrétní úkoly. Během této diskuze bylo vzneseno plno připomínek, na základě kterých byl Product Backlog upraven a aktualizován. Bylo důležité rozhodnout, kteří konkrétní vývojáři se budou na daném projektu podílet. S ohledem na rozsah projektu a interní potřebu mít k dispozici většinu vývojářů z důvodu stále probíhajících úprav kódu po nasazení nové aplikace bylo do projektu začleněno pouze

omezené množství vývojářů. Scrum Master se následně s vytvořeným týmem dohodnul na dalším setkání, již v rámci prvního Sprintu a umožnil jim přístup do nástěnky v Asaně.

V den začátku prvního Sprintu se setkal Scrum Master s týmem vývojářů a dalšími členy z jiných oddělení, kterým byly rovněž přiřazeny vybrané položky z Product Backlogu. Scrum Masterem byly v Asaně úkoly přesunuty pro daný Sprint do sloupce Sprint Backlog a odprezentovány členům týmu. Podoba Sprint Backlogu je představena na Obrázek 16. Tým byl rovněž seznámen s cílem Sprintu a s tím, jak celý průběh Sprintu bude probíhat. Byly projednány následující body:

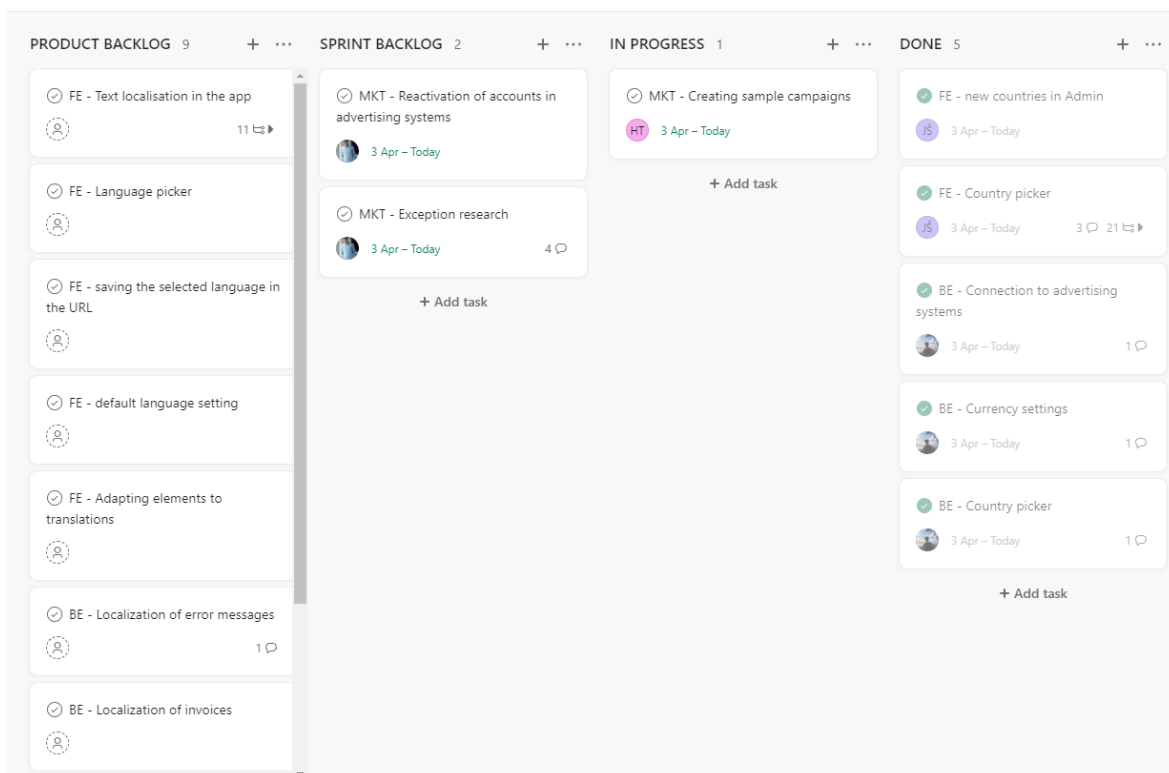
- každý den se budou konat Standup Meetingy, při kterých bude v týmu diskutováno o dosaženém pokroku, co zbývá dokončit a zda se vyskytly potíže, které je potřeba vyřešit.
- položky ze Sprint Backlogu, na kterých se má během Sprintu pracovat, byly seřazeny dle priority,
- určité úkoly jsou závislé na splnění těch dalších, je tedy vhodné postupovat při plnění úkolů postupně,
- každá položka je reprezentována svou vlastní kartou v Asaně s detailním popisem úkolu. Pokud se vyskytnou jakékoliv nejasnosti, tým má možnost položit dotaz formou komentáře přímo v dané kartě pro dovysvětlení zadání,
- každý člen týmu má své přidělené karty s konkrétními úkoly, na kterých bude pracovat a které bude na základě aktuálního stavu práce přemísťovat mezi sloupci, kde každý sloupec reprezentuje jiný stav průběhu,
- na konci Sprintu bude dokončená funkční část nasazena do produkce.



Obrázek 16: Náhled na Sprint Backlog

Zdroj: vlastní zpracování

Během probíhajícího Sprintu všichni členové týmu aktivně pracovali na svých úkolech, kartičky s úkoly přesouvali dle aktuálního stavu a celý jeho průběh byl bezproblémový. Průběh Sprintu v Asaně je zaznamenán na Obrázek 17. Na konci Sprintu bylo zjištěno, že nebyl splněn jeden úkol. Po konzultaci s Product Ownerem bylo rozhodnuto daný úkol přesunout do následujícího Sprintu, ve kterém se bude pokračovat v jeho plnění. Ostatní dokončené části funkcionality byly úspěšně nasazeny do produkce a následně prezentovány vedení společnosti během Sprint Review. Při testování funkcionality ze strany vedení společnosti byla identifikována chyba, jejíž řešení bylo přesunuto do následujícího Sprintu s vysokou prioritou. Vedení společnosti bylo jinak s dodanou funkcionalitou spokojeno a mimo to předložilo další návrh na budoucí vylepšení vzhledu, kterému se bude tým v budoucnu věnovat. Vzhledem k identifikované chybě došlo k závěru, že v následujícím Sprintu bude provedeno důkladné komplexní testování dané části funkcionality.



Obrázek 17: Průběh Sprintu

Zdroj: vlastní zpracování

Při retrospektivním setkání vývojářů a ostatních členů týmu se Scrum Masterem byl diskutován a zhodnocen celkový průběh Sprintu. Tým sdělil Scrum Masterovi zpětnou vazbu, že průběh Sprintu byl plynulý, zadání bylo dostačující, srozumitelné a nebyla shledána žádná závažná problémová situace.

Druhý Sprint, který plynule navazoval na první, byl věnován lokalizaci celé aplikace do cizích jazyků zemí, které byly nově přidány do aplikace v předchozím Sprintu. Na jeho začátku proběhlo setkání Scrum Mastera s týmem, aby mohl být průběh druhého Sprintu společně naplánován. Došlo k lehké obměně vývojářů v týmu, jelikož pro daný Sprint byly některé z položek Product Backlogu přiřazeny vývojářům, kteří v prvním Sprintu žádné položky k vypracování neměli a naopak. Z toho důvodu bylo opět Scrum Masterem zopakováno a vysvětleno, jakým způsobem bude celý Sprint probíhat, stejně jako tomu bylo na začátku prvního Sprintu. Rovněž byl stanoven cíl druhého Sprintu: „Nový uživatel pocházející ze země, pro kterou aplikace nově nabízí inzerci, bude celé aplikaci rozumět“. Tým byl seznámen se zbývajícím položkami pro daný Sprint, jež byly přesunuty do Sprint Backlogu.

Členové týmu si postupně prostudovali své úkoly a zahájili jejich vypracování. Prioritně byla řešena nalezená chyba z prvního Sprintu, kterou se podařilo ihned zpočátku úspěšně vyřešit. Dále byla kromě ostatních stále řešena nedokončená položka z prvního Sprintu. Po celou dobu druhého Sprintu se pravidelně konaly denní Standup Meetingy, při kterých se sledoval průběh práce, zjišťovalo se, zda vše probíhá v pořádku a případně byly konzultovány již dokončené položky. Během plnění položky věnující se nastavení jazyka pro inzerci ve službě Google Merchant Center nastala situace, ve které bylo zjištěno, že některé z nově přidanych zemí disponují více oficiálními jazyky. Tento fakt komplikoval jednoznačné určení výchozího jazyka pro inzerci. Týmem byla tato komplikace diskutována a bylo přijato společné rozhodnutí, že pro tyto země bude zvolen jazyk, kterým v dané zemi mluví nejvíce obyvatel, a v budoucnu bude toto řešení rozšířeno o možnost přímé volby jazyka pro inzerci uživatelem. V průběhu Sprintu byly některé záležitosti komunikovány a podrobnosti vysvětlovány skrze karty v Asaně, prostřednictvím kterých byly jednotlivé položky reprezentovány. Pokud to bylo možné, všechny dokončené položky byly individuálně otestovány, aby se ověřilo, že bylo jejich zadání splněno.

V rámci druhého Sprintu se všechny položky ze Sprint Backlogu stihly řádně dokončit ve stanoveném termínu a nebyly shledány žádné větší potíže při jejich plnění. Po dokončení poslední položky byla aplikace celkově zkontrolována a otestována, včetně provázané funkčnosti nově přidanych položek, aby bylo minimalizováno riziko případných problémů nebo chyb při prezentaci funkcionality vedení společnosti. Při této kontrole bylo zjištěno, že vše funguje v souladu s požadavky, a tudíž mohla být naplánována prezentace dokončené funkcionality pro vedení společnosti.

Sprint Review s vedením společnosti proběhlo úspěšně, dodaná funkcionality byla považována za splněnou dle zadání a dle požadavků. Ze strany vedení společnosti nebyly vzneseny žádné připomínky. Následovalo také retrospektivní setkání Scrum Mastera s vývojáři a ostatními členy týmu, při kterém byla z jejich strany sdělena pozitivní zpětná vazba. S průběhem druhého Sprintu byl tým spokojen, v případě potřeby bylo některé ze zadání upřesněno či podrobněji dovysvětleno a nebyly zaznamenány žádné problémy s plněním úkolů.

7 Vyhodnocení a případná doporučení

V rámci této kapitoly bude vyhodnocen návrh projektové optimalizace procesu vývoje webové aplikace, který byl ve vybrané společnosti úspěšně implementován. V kapitole 5.1 byla autorkou diplomové práce provedena analýza vývoje webové aplikace ve vybrané společnosti, při které byly identifikovány určité nedostatečné procesy ve vývoji, které vyžadovaly zlepšení. Mezi tyto procesy patřil například neefektivní průběh vývoje webové aplikace z důvodu chybějící metodiky, nedostatečných specifikací jednotlivých zadání či nesprávného způsobu přenosu informací. Na základě těchto identifikovaných slabých stránek ve vývoji byl stanoven cíl návrhu tyto slabé stránky minimalizovat či úplně odstranit a zároveň splnit konkrétní požadavky a potřeby společnosti, které byly zmíněny v kapitole 5.2.

7.1 Chybějící metodika

Zásadním problémem pro společnost a celý proces vývoje webové aplikace byla absence metodiky pro řízení projektů, což vedlo k neefektivnímu řízení dosavadního vývoje nových funkcí a rozšíření. Chybějící konkrétní pravidla a postupy způsobily prodlužování stanovených termínů nebo příliš dlouhý proces vývoje daných funkcí nebo rozšíření. Ze strany společnosti byl vysloven požadavek na nastavení postupů a metodik, dle kterých by byly projekty řádně řízeny. Autorka diplomové práce navrhla řízení projektů dle agilní metodiky SCRUM, což bylo následně aplikováno a testováno na vybraném projektu.

Autorka diplomové práce hodnotí celý průběh řízení projektu dle agilní metodiky SCRUM jako přínosný a efektivní. Tato metodika poskytla vývoji jasný směr, pravidelnou komunikaci a zpětnou vazbu, čímž bylo umožněno rychle reagovat na změny, a zřetelně definované role, které měly v rámci projektu určitou odpovědnost. Ze strany týmu byla k průběhu a řízení projektu dle návrhu autorky diplomové práce dodána následující zpětná vazba:

- implementace agilní metodiky SCRUM umožnila lepší přehled o tom, kdo pracuje na jakých úkolech a v jaké fázi se projekt momentálně nachází. To přispělo ke komplexnímu přehledu a snadnějšímu sledování pokroku projektu.

- Rozdělení projektu do jednotlivých Sprintů zajistilo možnost lépe předpovídat situaci ohledně skutečného termínu dokončení projektu a identifikovat případná zpoždění.
- Přiřazení jasně definovaných položek z Product Backlogu jednotlivým členům týmu zajistilo jejich plynulé a efektivní vypracování.
- Důležité informace a zadání úkolů byly spravovány na jednom místě, čímž byl zaručen snadný a rychlý přístup k těmto informacím. Nebylo nutné vyhledávat potřebné informace ve více komunikačních kanálech a nedocházelo tak ke zbytečným ztrátám důležitých informací.
- Pomocí jasně definovaných rolí byla zajištěna přehlednost v odpovědnosti za dané položky a jejich nadefinování. Byla tím usnadněna komunikace a minimalizoval se vznik zbytečných nejasností, což přispělo k efektivnějšímu průběhu projektu.
- Využití metodiky SCRUM zajistí hladší průběh zejména u větších, dlouhodobějších projektů, při kterých je do procesu zapojeno větší množství lidí.

Pro další budoucí projekty je týmem navrhováno v rámci řízení projektu v nástroji Asana přidání dalšího sloupce, ve kterém budou shromažďovány dokončené položky připravené ke kontrole. K tomuto testování by měl být přiřazen jeden člověk, který v případě plné funkčnosti přesune položku do sloupce vyhotovených úkolů a v opačném případě vrátí úkol danému člověku, který na něm pracoval, aby jej opravil. Tím bude zajištěna důkladná kontrola položek v každém Sprintu a předejde se tak zbytečným chybám v budoucnu.

Autorkou diplomové práce je doplněno doporučení testovat dokončené položky v jednotlivých Srintech komplexněji, aby bylo možné lépe porozumět interakcím mezi různými částmi funkcionality a bylo tak možné odhalit potencionální problémy, které by pouhým testováním jednotlivých položek mohly být přehlédnuty. Vzhledem k tomu, že v době, kdy probíhal projekt, nebylo dostupné testovací prostředí, nebylo možné provádět testování funkcionality mimo produkční prostředí. Pro budoucí projekty již bude zajištěno důkladné a kompletní testování funkcionality před nasazením do produkce, aby se předešlo přidání nové funkcionality do aplikace s chybami.

7.2 Nedostatečná zadání

Dalším požadavkem ze strany společnosti bylo zlepšení procesu formulace a specifikace zadání, aby nedocházelo k nedorozumění mezi zadávajícím a vývojáři, což mělo často za následek nutnost přepracování výsledné funkcionality a zpomalení vývoje webové aplikace. V rámci návrhu byl tento požadavek řešen pomocí Product Backlogu, ve kterém byly stručně a jasně definovány jednotlivé položky. Tyto položky byly v nástroji Asana reprezentovány prostřednictvím karet, do kterých měli přístup všichni členové týmu. Karty byly shromážděny v určeném sloupci a prioritně seřazeny. Obsahovaly stručný popis požadavku, případně byly jejich součástí grafické či ostatní podklady potřebné pro vyhotovení úkolu. Proces vytváření a předávání zadání úkolů byl týmem zhodnocen následovně:

- jasná a přesně stanovená zadání umožnila lépe porozumět požadavkům a úkolům, což usnadnilo jejich realizaci a minimalizoval se výskyt případného nedorozumění a neshody.
- U každé karty byl uveden termín, ve kterých bylo žádané danou položku dokončit. Tato vizualizace poskytovala časový přehled o všech zbývajících položkách, které bylo nutné dokončit, což výrazně usnadňovalo organizaci práce.

Autorkou diplomové práce je tvorba zadání hodnocena za uspokojivou. U většiny položek bylo zadání pro členy týmu jasné a pochopitelné. S vypracováním úkolu dle zadání neměli výrazné problémy. Nicméně, u některých položek Product Backlogu bylo nutné poskytnout drobné dovysvětlení či doplňující informace. Pro příští projekty bude zohledněn fakt, že je nezbytné důkladně a detailně popsat danou problematiku u jednotlivých položek pro jejich bezproblémové vypracování.

Tým by pro budoucí projekty v rámci specifikace zadání uvítal informaci o tom, jak lze danou funkcionalitu v budoucnu dále rozvíjet, aby byl na tyto budoucí možnosti rozvoje brán ohled již při prvním vypracování. Rovněž při tvorbě jednotlivých položek a zadání shledává tým za užitečné, pokud by jednotliví členové týmu mohli navrhnout řešení pro danou funkcionalitu, která by se následně s ostatními zvažovala. Dále je dle týmu a vedení společnosti klíčové klást důraz na důkladné shromáždění všech potřebných informací při tvorbě Product Backlogu a zahrnout do procesu rovněž přehled věcí, u kterých může s novou funkcionalitou nastat vzájemné ovlivnění. Pro

zajištění přehlednosti o vlivu jednotlivých položek na ostatní je navrhováno vytvoření souhrnného schématu celé webové aplikace, v němž budou zachyceny všechny důležité aspekty.

7.3 Přenos informací

Špatně nastavený přenos informací mezi členy týmu a ostatními zaměstnanci vedl v procesu vývoje k častým zmatkům v komunikaci. Opakovaně se stávalo, že byla přijímána rozhodnutí ohledně dalších kroků ve vývoji, ale nikde nebyla tato rozhodnutí zaznamenána, případně kým to rozhodnutí bylo stanoveno. Návrhem autorky diplomové práce bylo jasné rozdělení rolí, aby byla zodpovědnost za jednotlivá rozhodnutí jasně stanovena. Byly definovány role Product Ownera, který nesl zodpovědnost za Product Backlog, a Scrum Mastera, jehož zodpovědností byl dohled nad průběhem Sprintů a celým týmem. Dále byl vytvořen prostor pro sdílenou komunikaci a sledování průběhu projektu prostřednictvím nástěnky v nástroji Asana. V jednotlivých kartách byly uchované veškeré informace, podklady, dotazy a případná rozhodnutí o dalších krocích. Pro pravidelnou komunikaci v týmu se konaly Standup Meetingy, při kterých měl tým možnost diskutovat o konkrétních úkolech či vzniklých problémech. Autorka diplomové práce zpozorovala, že nastavení procesu komunikace bylo prováděno efektivně. Komunikace probíhala bez zásadních komplikací či nejasností, což přispělo k plynulému průběhu projektu. Pro tým, který na projektu pracoval, byly přínosné zejména následující aspekty:

- Standup Meetingy byly užitečné, zejména díky jejich rychlosti. Umožnily rychlejší vysvětlení určitých detailů a usnadnily porozumění. Bylo to mnohem efektivnější, než kdyby byla komunikace vedena skrze komunikační kanál Slack.
- Všechny potřebné informace byly přehledně sepisovány a shromažďovány v konkrétní kartě, která představovala určitý úkol.

Pro další projekty vyžaduje vedení společnosti zejména udržení rychlosti průběhu Standup Meetingů, a to z důvodu zachování co nejvyšší možné efektivity a soustředěnosti na důležité body. Dále se tým shodnul na názoru, že je potřebné komunikovat vše jasně, stručně a srozumitelně, aby nedošlo k případným neshodám v budoucím procesu vývoje.

Ze strany autorky je doporučeno, že by bylo vhodné se na schůzkách setkávat osobně, namísto online. Osobní setkání se všemi členy týmu umožňuje lepší interakci a komunikaci, usnadňuje vyjádření dotazů a připomínek a zvyšuje zapojení celého týmu do diskuze, což přispívá k celkové efektivitě. Dále při řešení problémů a nových rozhodnutí umožňuje osobní setkání rychlejší reakce a aktivní zapojení všech členů týmu.

7.4 Pomalý vývoj

Výše zmíněné nedostatky měly za následek nedodání hotových funkcionalit včas, prodlužování stanovených termínů a celkové zpomalení vývoje. Pro společnost je rychlost vývoje zásadní, neboť se nachází ve startupovém prostředí, kde je rychlý a inovativní vývoj očekáván. Ze strany společnosti byl vznesen požadavek, aby byl vývoj urychlen.

Autorkou je vnímáno, že na základě řízení projektu dle SCRUM a nastavení pravidel v týmu došlo k významnému zlepšení v procesu vývoje. Projekt byl v daném termínu dokončen, byť byla jedna nedokončená položka z prvního Sprintu přesunuta do Sprintu druhého. Tým zhodnotil celkový průběh projektu a snahu urychlit vývoj následovně:

- byl poskytnut větší přehled o stavu jednotlivých položek Product Backlogu a o tom, kdo aktuálně pracuje na jakém úkolu. Díky tomu měl tým přesnou vizualizaci průběhu projektu, respektive v jaké fázi se projekt momentálně nachází.
- Pomocí systematického sledování pokroku vývoje byla lépe organizována koordinace mezi frontendovými a backendovými vývojáři, což umožnilo efektivnější spolupráci. Úkoly na sebe plynule navazovaly a zabránilo se tak zbytečnému čekání na dokončení práce jiných členů týmu.

Během plnění jednotlivých položek při Sprintech byla zjištěna chyba, která mohla vývoj zpomalit. Z toho důvodu je nezbytné, jak již bylo zmíněno výše, položky důkladně otestovat, a to nejlépe v testovacím prostředí. V tomto případě ke zpomalení vývoje a nedodržení termínu dokončení nedošlo, zejména z důvodu

správného zorganizování a řízení této situace. Bylo rozhodnuto, že bude chyba prioritně opravena ihned na začátku druhého Sprintu, zatímco vývoj pokračoval beze změn.

7.5 Důležitost vytvářených funkcí a rozšíření

Vybraný projekt Expanze byl uživateli shledán za důležitý z ohledu toho, že byl často ve zpětných vazbách žádán. Bohužel, nyní není možné posoudit využitelnost nové funkcionality na základě konkrétních dat, protože nebyl dostatek času na získání relevantních dat pro vyhodnocení. Nyní se nedá bohužel ani celkově zhodnotit, zda byla záležitost s vytvářením nových funkcí a rozšíření bez kontextu plně odstraněna, nicméně společnost podnikla určité kroky k tomu, aby vývoj kladl důraz na potřeby a požadavky ze strany uživatelů. Jsou připraveny další nové projekty, které reflektují potřeby a požadavky uživatelů na základě jejich zpětné vazby. Tyto projekty budou v budoucnu postupně vypracovány a bude sbírána další zpětná vazba, díky které bude udáván směr vývoje.

7.6 Ekonomické zhodnocení návrhu

Je důležité posoudit, jaký přínos měl návrh projektové optimalizace procesu vývoje webové aplikace rovněž z ekonomického hlediska. Pro dosažení co nejpřesnějšího ekonomického zhodnocení návrhu je nezbytné dlouhodobě sledovat chování uživatelů v porovnání s minulými měsíci, ve kterých byl vývoj nových funkcí pozastaven z důvodu přepisu aplikace, což prozatím není možné vyhodnotit, jelikož nová funkcionality byla uvedena do produkce teprve nedávno.

Pro udržení růstu a rozvoje společnosti je důležité nejen sledovat míru odchodu uživatelů, ale také pravidelně získávat z jejich strany zpětnou vazbu. Rovněž je klíčové nadále dodržovat navržený proces řízení vývoje k tomu, aby byl vývoj webové aplikace Tanganica rychlý a efektivní. Cílem a snahou společnosti by mělo být postupné odstranění všech slabých stránek v procesu vývoje, aby nedošlo k ohrožení aplikace a celé společnosti jak ze strany konkurence, tak ze strany nespokojených uživatelů.

Pokud by byly započítány náklady na vývoj, největším nákladem jsou mzdy vývojářů, které činí celkem 237 000 Kč měsíčně. Pro uživatele, kteří aktivně využívají aplikaci, je nastaven ceník s provizí společnosti ve výši 3 % z vygenerovaného obrátu z reklam skrze aplikaci Tanganica. Pokud by byla brána v potaz pro pokrytí nákladů výše provize aktivních e-shopů, kterých je v aplikaci průměrně 400, musela by hodnota vygenerovaného obrátu skrze reklamy u každého jednotlivého e-shopu dosáhnout 19 750 Kč měsíčně.

Rychlý vývoj a implementace nových funkcí a rozšíření do aplikace přispějí k rychlejšímu nárůstu počtu aktivních e-shopů, včetně nárůstu jejich spokojenosti s aplikací. S narůstajícím počtem aktivních e-shopů poroste i zisk společnosti.

Závěr

Cílem diplomové práce byla formulace návrhu pro projektovou optimalizaci procesu vývoje webové aplikace a jeho následná implementace, a to na základě provedené analýzy současného procesu vývoje webové aplikace Tanganica ve společnosti TGNCA, která odhalila slabé stránky, které by mohly do budoucna ohrozit nejen vývoj webové aplikace, ale i celkové fungování společnosti. Vzhledem k tomu, že se společnost TGNCA nachází ve startupovém prostředí, bylo nezbytné navrhnout potřebné kroky k tomu, aby byl současný proces vývoje optimalizován a umožněn rychlejší a efektivnější vývoj webové aplikace.

První polovina této diplomové práce byla věnována teoretické části, ve které byl vysvětlen projektový management, jeho důležitost a procesy. Dále byla v práci rozebrána definice projektu a projektového týmu, rozdělení rolí a funkce projektového manažera. Čtenář byl seznámen s problematikou projektového řízení v oblasti vývoje softwaru, charakteristikami softwarového projektu a se všemi fázemi životního cyklu vývoje softwaru. V závěru teoretické části byly představeny různé přístupy k vývoji softwaru a byly zmíněny digitální nástroje pro usnadnění projektového řízení.

V úvodu praktické části byl čtenář seznámen s vybranou společností TGNCA s.r.o., dále s webovou aplikací Tanganica, která je hlavním předmětem podnikání společnosti a rovněž byl popsán dosavadní vývoj této webové aplikace. Následně byla provedena analýza současného procesu vývoje webové aplikace s ohledem na startupové prostředí, ve kterém se společnost nachází. Na základě identifikovaných slabých stránek vývoje a požadavků společnosti byl formulován návrh pro projektovou optimalizaci procesu tohoto vývoje, jehož cílem bylo identifikovaná slabá místa procesu vývoje minimalizovat či úplně odstranit, aby nedošlo k ohrožení samotné webové aplikace nebo celé společnosti. Následně byl vybrán projekt, na kterém byl návrh implementován. V závěru práce byl celý průběh projektu na základě návrhu vyhodnocen jak autorkou diplomové práce, tak zaměstnanci a vedením vybrané společnosti.

Na základě pozorování průběhu projektu je autorkou diplomové práce doporučeno pro budoucí projekty dodržovat implementovanou metodiku s drobnými úpravami týkající se komplexního testování nových funkcionalit pro zajištění porozumění jejich

interakcím s různými částmi aplikace. Dále je doporučena osobní přítomnost na pravidelných schůzkách pro jejich lepší efektivitu a neustále a pravidelně naslouchat potřebám a požadavkům uživatelů.

Na základě získané zpětné vazby od zaměstnanců a vedení společnosti na implementovaný návrh je doporučeno dodržovat detailní popisy jednotlivých položek, aby byla jejich zadání jasná, stručná a srozumitelná. Dále se navrhuje zahrnout do těchto popisů položek informace o dalším budoucím rozvoji projektu nebo dané položky a provést drobné úpravy v nástroji Asana pro vizuální řízení položek.

Seznam použité literatury

AFSHARI, Mahnaz a Taghi JAVDANI GANDOMANI, 2022. A novel risk management model in the Scrum and extreme programming hybrid methodology. online. *International Journal of Electrical and Computer Engineering (IJECE)*, roč. 12, č. 3, s. 2911. Dostupné z: <https://doi.org/10.11591/ijece.v12i3.pp2911-2921>.

AHMED, Ashfaque, 2016. *Software Project Management: A Process-Driven Approach*. online. CRC Press. ISBN 978-1-4398-4656-8. Dostupné z: https://www.google.cz/books/edition/Software_Project_Management/VCHOBQAAQBAJ?hl=cs&gbpv=1.

ALLAN, Barbara, 2017. *The No-Nonsense Guide to Project Management*. online. Facet Publishing. ISBN 978-1-78330-203-1. Dostupné z: https://www.google.cz/books/edition/The_No_Nonsense_Guide_to_Project_Management/NbstDwAAQBAJ?hl=cs&gbpv=1.

ASANA, 2024. Asana's Work Tracking & Project Management Features • Asana. online. In: *Asana*. Dostupné z: <https://asana.com/features>. [citováno 2024-03-17].

ATLASSIAN, 2023. Spravujte projekty svého týmu odkudkoli | Trello. online. In: *Atlassian Trello*. Dostupné z: <https://trello.com/cs>. [citováno 2024-03-17].

ATLASSIAN, 2024. Jira | Issue & Project Tracking Software | Atlassian. online. In: *Jira Software*. Dostupné z: <https://www.atlassian.com/software/jira>. [citováno 2024-03-17].

AYTEMIZ, Togay, 2023. How does the SDLC play a role in the success of product development and launch? In: *Agile Insider*. 2023-01-14. Dostupné z: <https://medium.com/agileinsider/how-does-the-sdlc-play-a-role-in-the-success-of-product-development-and-launch-a17baaac1054>. [citováno 2024-03-16].

BARDA, Lukáš, 2022. Video návod na Kanban - Lukáš Barda. In: *Lukáš Barda*. 2022-03-10. Dostupné z: <https://lukasbarda.cz/video-navod-na-kanban/>. [citováno 2024-03-17].

BARTOŠ, Pavel, 2021. *Úvod do programování: Základní orientace v problematice vývoje*. online. Evropská akademie vzdělávání SE. ISBN 978-80-11-04386-5. Dostupné z: https://www.google.cz/books/edition/%C3%9Avod_do_programov%C3%A1n%C3%AD/jbnvEAAQBAJ?hl=cs&gbpv=0.

BENNETT, Jason a Jennifer BOWEN, 2018. *Lean: 8-in-1 book ultimate collection*. Jason Bennett & Jennifer Bowen. ISBN 978-1-72021-629-2.

BOYDE, Joshua, 2014. *A Down-To-Earth Guide To SDLC Project Management (2nd Edition): Getting your system / software development life cycle project successfully across the line using PMBOK adaptively*. online. Joshua Boyde. Dostupné z: https://www.google.cz/books/edition/A_Down_To_Earth_Guide_To_SDLC_Project_Ma/qchnBgAAQBAJ?hl=cs&gbpv=0.

BRAUDE, Eric J. a Michael E. BERNSTEIN, 2016. *Software Engineering: Modern Approaches, Second Edition*. online. Waveland Press. ISBN 978-1-4786-3303-7. Dostupné z: https://www.google.cz/books/edition/Software_Engineering/kLICwAAQBAJ?hl=cs&gbpv=0.

BREWER, Jeffrey L. a Kevin C. DITTMAN, 2018. *Methods of IT Project Management: Third Edition*. online. Purdue University Press. ISBN 978-1-61249-552-1. Dostupné z: https://www.google.cz/books/edition/Methods_of_IT_Project_Management/4Q2yDwAAQBAJ?hl=cs&gbpv=1.

BURKE, Rory a Steve BARRON, 2014. *Project Management Leadership: Building Creative Teams*. online. John Wiley & Sons. ISBN 978-1-118-82540-2. Dostupné z: https://www.google.cz/books/edition/Project_Management_Leadership/R0PeAgAAQBAJ?hl=cs&gbpv=1&dq=project+team&printsec=frontcover.

CURRY, Cate, 2021. Project Life Cycle Vs. Development Life Cycle. In: *RMC Learning Solutions*. 2021-05-18. Dostupné z: RMC Learning Solutions, <https://rmcls.com/project-versus-development-life-cycle/>. [citováno 2024-03-16].

DOLEŽAL, Jan, 2022. *Agilní přístupy vývoje produktu a řízení projektu: komplexně, prakticky a dle světové praxe*. První vydání. Praha: Grada. ISBN 978-80-271-3705-3.

DOLEŽAL, Jan, 2023. *Projektový management*. 2. vydání. Praha: Grada Publishing. ISBN 978-80-271-3619-3.

EBY, Kate, 2023. Project Organizational Structure | Smartsheet. online. In: *Smartsheet*. Dostupné z: <https://www.smartsheet.com/content/project-management-organization>. [citováno 2024-02-17].

FIELDING, Paul J., 2020. *Jak správně řídit projekty: osvojte si nezbytné dovednosti pro časově a finančně efektivní řízení projektů*. 1. vydání. V Brně: Lingea s.r.o. ISBN 978-80-7508-622-8.

FLYLIB.COM, 2017. Project Management Process Groups | PMP Exam Cram 2 (2nd Edition). online. In: *Flylib.com*. Dostupné z: <https://flylib.com/books/en/3.170.1.32/1/>. [citováno 2024-02-22].

FONT JR., Victor M., 2013. The Project Management Method and the SDLC - The Ultimate Guide to the SDLC. online. In: *The Ultimate Guide to the SDLC*. Dostupné z: <https://ultimatesdlc.com/project-management-method-sdlc/>. [citováno 2024-03-16].

FRELO, 2024. Aplikace na projektové řízení, Správu úkolů a Spolupráci. online. In: *FreeloCZ*. Dostupné z: <https://www.freelo.io/cs>. [citováno 2024-03-17].

CHIOCCHIO, François; E. Kevin KELLOWAY a Brian HOBBS, 2015. *The Psychology and Management of Project Teams*. online. Oxford University Press. ISBN 978-0-19-986137-8. Dostupné z: https://www.google.cz/books/edition/The_Psychology_and_Management_of_Project_Teams/LSYABwAAQBAJ?hl=cs&gbpv=1&kptab=overview.

IGBERAESE, Davies A., 2022. *Introduction to Project Management: A Source Book for Traditional PM Basics*. online. Taylor & Francis. ISBN 978-1-00-064598-9. Dostupné z: https://www.google.cz/books/edition/Introduction_to_Project_Management/uzh7EAAAQBAJ?hl=cs&gbpv=0.

KAKAR, Ashish, 2023. What more can Software Development Learn from Agile Manufacturing? Some Pointers on the 20th Anniversary of the Agile Manifesto. online. *Journal of the Southern Association for Information Systems*, roč. 10, č. 1, s. 30–40. Dostupné z: <https://doi.org/10.17705/3JSIS.00031>.

KAZIM, Ali, 2017. A Study of Software Development Life Cycle Process Models - ProQuest. online. In: *ProQuest*. Dostupné z: <https://www.proquest.com/docview/1901446145/C9CF35F8E1C54716PQ/7?accountid=17116&sourcetype=Scholarly%20Journals>. [citováno 2024-03-14].

KŘIVÁNEK, Mirko, 2019. *Dynamické vedení a řízení projektů: systémovým myšlením k úspěšným projektům*. První vydání. Praha: Grada. ISBN 978-80-271-0408-6.

KUMAR, Madhup a Ekbal RASHID, 2018. An Efficient Software Development Life cycle Model for Developing Software Project. online. *International Journal of Education and Management Engineering*, roč. 8, č. 6, s. 59–68. Dostupné z: <https://doi.org/10.5815/ijeme.2018.06.06>.

MÁCHAL, Pavel; Martina KOPEČKOVÁ a Radmila PRESOVÁ, 2015. *Světové standardy projektového řízení: pro malé a střední firmy*. online. Grada Publishing, a.s. ISBN 978-80-247-5321-8. Dostupné z: https://www.google.cz/books/edition/Sv%C4%9Btov%C3%A9_standardy_projektov%C3%A9ho_%C5%99%C3%ADze/4rGjBqAAQBAJ?hl=cs&gbpv=1&dq=vznik+projektov%C3%A9ho+t%C3%BDmu&pg=PA26&printsec=frontcover.

MANAGEMENTMANIA, 2019. Magický trojúhelník projektového řízení. online. In: *ManagementMania.com*. Dostupné z: <https://managementmania.com/cs/magicky-trojuhelnik-projektoveho-rizeni>. [citováno 2024-02-08].

MARTINELLI, Russ J.; James M. WADDELL a Tim J. RAHSCHULTE, 2017. *Projects Without Boundaries: Successfully Leading Teams and Managing Projects in a Virtual World*. online. John Wiley & Sons. ISBN 978-1-119-14254-6. Dostupné z: https://www.google.cz/books/edition/Projects_Without_Boundaries/RzpGDgAAQBAJ?hl=cs&gbpv=1.

MURRAY, Anna P., 2016. *The Complete Software Project Manager: Mastering Technology from Planning to Launch and Beyond*. online. John Wiley & Sons. ISBN 978-1-119-16183-7. Dostupné z: https://www.google.cz/books/edition/The_Complete_Software_Project_Manager/OhJOCgAAQBAJ?hl=cs&gbpv=1.

O'REGAN, Gerard, 2022. *Concise Guide to Software Engineering: From Fundamentals to Application Methods*. online. Springer Nature. ISBN 978-3-031-07816-3. Dostupné z: https://www.google.cz/books/edition/Concise_Guide_to_Software_Engineering/rPuLEAAAQBAJ?hl=cs&gbpv=1.

ÖZKAN, Deniz a Alok MISHRA, 2019. Agile Project Management Tools: A Brief Comparative View. online. *Cybernetics and Information Technologies*, roč. 19, č. 4, s. 17–25. Dostupné z: <https://doi.org/10.2478/cait-2019-0033>.

PENDYALA, Vishnu, 2019. *Tools and Techniques for Software Development in Large Organizations: Emerging Research and Opportunities: Emerging Research and Opportunities*. online. IGI Global. ISBN 978-1-79981-865-6. Dostupné z: https://www.google.cz/books/edition/Tools_and_Techniques_for_Software_Development/-LjjDwAAQBAJ?hl=cs&gbpv=0.

PM CONSULTING, [b. d.]. Maticová organizační struktura. online. In: *PM Consulting*. Dostupné z: <https://www.pmconsulting.cz/pm-wiki/maticova-organizacni-struktura/>. [citováno 2024-02-18].

PMI, Project Management Institute, 2022. *Process Groups: A Practice Guide*. online. Project Management Institute. ISBN 978-1-62825-784-7. Dostupné z: https://www.google.cz/books/edition/Process_Groups_A_Practice_Guide/mB-bEAAAQBAJ?hl=cs&gbpv=0.

RADAIDEH, Moh'd A., 2023. *Software Project Management: With PMI, IEEE-CS, and Agile-SCRUM*. online. Walter de Gruyter GmbH & Co KG. ISBN 978-3-11-120761-2. Dostupné z: https://www.google.cz/books/edition/Software_Project_Management/ZrTiEAAAQBAJ?hl=cs&gbpv=0.

RUHE, Günther a Claes WOHLIN, 2014. *Software Project Management in a Changing World*. online. Springer. ISBN 978-3-642-55035-5. Dostupné z: https://www.google.cz/books/edition/Software_Project_Management_in_a_Changing_World/aM1uBAAAQBAJ?hl=cs&gbpv=0.

SUTHERLAND, Jeff, 2019. *Scrum: the art of doing twice the work in half the time*. London: Random House Business. ISBN 978-1-84794-110-7.

SVOZILOVÁ, Alena, 2016. *Projektový management: systémový přístup k řízení projektů*. 3., aktualizované a rozšířené vydání. Praha: Grada Publishing. ISBN 978-80-271-0075-0.

ŠOCHOVÁ, Zuzana a Eduard KUNCE, 2019. *Agilní metody řízení projektů*. 2. vydání. Brno: Computer Press. ISBN 978-80-251-4961-4.

TAKCH, Hussein, 2020. *The Project Management-Je*. online. Hussein Takch. Dostupné z: https://www.google.cz/books/edition/The_Project_Management_Je/FNvWDwAAQBAJ?hl=cs&gbpv=1.

TELES, Vinícius Manhães, 2017. *Extreme Programming: Aprenda como encantar seus usuários desenvolvendo software com agilidade e alta qualidade*. online. Novatec Editora. ISBN 978-85-7522-574-5. Dostupné z: https://www.google.cz/books/edition/Extreme_Programming/zt3QDqAAQBAJ?hl=cs&gbpv=0.

VEBER, Jaromír, 2021. *Management: základy, přístupy, soudobé trendy*. I. vydání. Praha: Ekopress. ISBN 978-80-87865-69-9.

VILLAFIORITA, Adolfo, 2016. *Introduction to Software Project Management*. online. CRC Press. ISBN 978-1-4665-5954-7. Dostupné z: https://www.google.cz/books/edition/Introduction_to_Software_Project_Managem/UFHSBQAAQBAJ?hl=cs&qbpv=1.

WYSOCKI, Robert K., 2013. *Effective project management: traditional, agile, extreme, seventh edition*. 7th edition. Indianapolis, IN: John Wiley and Sons. ISBN 978-1-118-72916-8.

YADAV, RamKaran; M.L. MITTAL a Rakesh JAIN, 2018. Adoption of lean principles in software development projects. online. *International Journal of Lean Six Sigma*, roč. 11, č. 2, s. 285–308. Dostupné z: <https://doi.org/10.1108/IJLSS-03-2018-0031>.