



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

NÁVRH A IMPLEMENTACE MODULU INFORMAČNÍHO SYSTÉMU

IMPLEMENTATION OF THE INFORMATION SYSTEM

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Veronika Marečková

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Lukáš Novák, Ph.D.

BRNO 2020

Zadání bakalářské práce

Ústav:	Ústav informatiky
Studentka:	Veronika Marečková
Studijní program:	Systémové inženýrství a informatika
Studijní obor:	Manažerská informatika
Vedoucí práce:	Ing. Lukáš Novák, Ph.D.
Akademický rok:	2019/20

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává bakalářskou práci s názvem:

Návrh a implementace modulu informačního systému

Charakteristika problematiky úkolu:

Úvod
Vymezení problému a cíle práce
Teoretická východiska práce
Analýza problému a současné situace
Vlastní návrhy řešení, přínos návrhů řešení
Závěr
Seznam použité literatury
Přílohy

Cíle, kterých má být dosaženo:

Cílem je analyzovat stávající stav modulu informačního systému vybrané organizace a jeho efektivnosti, posoudit tento stav a připravit alternativní možnosti nového informačního systému včetně posouzení variant, optimálního návrhu a implementace.

Základní literární prameny:

BĚBR, Richard a Petr DOUCEK. Informační systémy pro podporu manažerské práce. Praha: Professional Publishing, 2005. ISBN 80-86419-79-7.

GÁLA, Libor, Jan POUR a Zuzana ŠEDIVÁ. Podniková informatika: počítačové aplikace v podnikové a mezipodnikové praxi. 3. aktualiz. vyd. Praha: Grada Publishing, 2015. ISBN 978-80-247-5457-4.

KOCH, Miloš. Informační systémy a technologie. 2. vyd. Brno: Zdeněk Novotný, 2002. Učební texty vysokých škol. ISBN 80-214-2193-2.

SODOMKA, Petr a Hana KLČOVÁ. Informační systémy v podnikové praxi. 2. aktualiz. a rozš. vyd.
Brno: Computer Press, 2010. ISBN 978-80-251-2878-7.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2019/20

V Brně dne 29.2.2020

L. S.

doc. RNDr. Bedřich Půža, CSc.
ředitel

doc. Ing. et Ing. Stanislav Škapa, Ph.D.
děkan

Abstrakt

Tato bakalářská práce se zaměřuje na výběr vhodného systému pro vnitřní firemní proces žádank. Výstupem mé práce bude aplikace pro žádanky obecné. První část obsahuje teoretické pojmy. V druhé části jsou základní informace o firmě a možném řešení. Třetí část pak popisuje vytvoření aplikace.

Abstract

This bachelor thesis focuses on the selection of a suitable system for internal business process requisitions. The output of my work will be an application for general requests. The first part contains theoretical concepts. The second part contains basic information about the company and possible solutions. The third part is about creating application.

Klíčová slova

informační systém, informace, data, workflow, JIRA

Key words

information system, information, data, workflow, JIRA

Bibliografická citace

MAREČKOVÁ, Veronika. Návrh a implementace modulu informačního systému. Brno, 2020. Dostupné také z: <https://www.vutbr.cz/studenti/zav-prace/detail/127481>.
Bakalářská práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Lukáš Novák.

Čestné prohlášení

Prohlašuji, že předložená bakalářská práce je původní a zpracovala jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušila autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně 17. 05. 2020

podpis studenta

Poděkování

Mé poděkování patří panu Ing. Lukáši Novákovi, Ph.D. za ochotu a cenné rady při vedení mé práce. Také bych chtěla poděkovat firmě K-net Technical International Group, s.r.o. za spolupráci při získávání údajů a odborný dohled.

Obsah

Úvod.....	11
Cíle práce, metody a postupy zpracování	12
1 Teoretická východiska práce	13
1.1 Základní pojmy	13
1.1.1 Informace	13
1.1.2 Data	14
1.1.3 Podnik	15
1.1.4 Proces	15
1.1.5 Workflow – pracovní postup	16
1.2 Informační systém	17
1.2.1 ERP systémy	19
1.2.2 CRM systémy	19
1.2.3 SCM systémy	20
1.2.4 MIS systémy	20
1.3 Informační technologie	21
1.4 Infrastruktura podnikového informačního systému	21
1.5 Analýzy	22
1.5.1 7S analýza	22
1.5.2 SWOT analýza	12
1.5.3 Riziková analýza.....	13
1.5.4 Časová analýza	15
1.6 Programovací jazyk.....	17
1.6.1 Java	17
2 Analýza Současného stavu.....	19

2.1	Základní informace o společnosti	19
2.2	Popis společnosti	20
2.3	Služby společnosti K-net.....	20
2.4	Popis hlavních firemních procesů	22
2.5	Konkurence	23
2.6	Používané systémy IT	23
2.7	7S analýza	24
2.8	SWOT analýza	27
2.9	Navrhované možnosti řešení	28
3	Vlastní návrh řešení	29
3.1	Předprojektová fáze.....	29
3.1.1	Zhodnocení současného stavu	29
3.1.2	Časová analýza	31
3.1.3	Riziková analýza.....	44
3.2	Možnosti výběru IS	45
3.2.1	Základní požadavky – Popis požadavků zákazníka.....	45
3.2.3	Možnosti výběru systému	49
3.3	Implementace	52
3.3.1	Instalace	52
3.3.2	Změna nastavení	53
3.3.3	Nastavení zákaznického portálu	54
3.3.4	Uživatelé	56
3.3.6	Nastavení notifikací	58
3.3.7	Tvorba a nastavení workflow	58
3.3.8	Zákaznický portál	58
3.4	Ekonomické zhodnocení projektů.....	65

3.4.1	Výdaje.....	65
3.4.2	Přínosy	66
	Závěr	68
	Seznam použitých zdrojů.....	69
	Seznam obrázků.....	73
	Seznam příloh	76
	Přílohy.....	77

ÚVOD

V dnešní době se jak firmy, tak i jednotlivci snaží automatizovat práci a využívat co nejvíce výpočetní techniku a různé aplikace pro efektivnější život.

Cílem této bakalářské práce je najít vhodný systém pro firmu K-net. Obchodní společnost K-net, s. r. o. se zabývá IT technologiemi. Obchodní společnost vznikla v roce 1993. Současný systém již nevyhovuje jejich požadavkům. Je potřeba jej modernizovat a upravit některé funkce.

Prvním krokem mé práce bude analyzovat požadavky pro informační systém, tzn.: které informace budeme získávat, shromažďovat a zjistit, co náš systém bude muset umět. Poté, co budou tyto požadavky specifikovány, bude vybráno několik informačních systémů, které porovnáme a vybereme z nich nejvhodnější pro nás. Vhodný informační systém bude vybrán podle určitých požadavků. V této práci bude také postup této implementace blíže specifikován. Celý projekt bude na konci ekonomicky zhodnocen, také uvedu přínosy práce.

CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ

Cílem této bakalářské práce je analyzovat konkrétní proces ve firmě K-net, na základě této analýzy vybrat informační systém, který usnadní řízení tohoto procesu a zpříjemní tak zaměstnancům jejich požadavky. Práce se zaměřuje na proces žádanky obecné. Tento postup však bude možné do budoucna aplikovat i na žádanky o dovolenou a o školení.

Dílčím cílem je, aby byl informační systém jednoduchý na uživatelskou obsluhu a práce s ním tak nečinila potíže zaměstnancům firmy.

Jelikož se jedná o zásah do softwaru společnosti, bude provedena analýza rizik tohoto řešení společně s návrhem na snížení rizikovosti projektu s ohledem na časové hledisko, které bude zohledněno pomocí časového harmonogramu vytvořeného pomocí metody PERT a síťového grafu. Ve stěžejní návrhové části bude věnována pozornost výběru a následně implementaci zvoleného nástroje pro projektové řízení, přičemž bude věnována pozornost nastavení základních oblastí důležitých pro řízení projektů. Závěrem bude provedeno ekonomické zhodnocení navrhovaného řešení a očekávaných přínosů.

1 TEORETICKÁ VÝCHODISKA PRÁCE

V této části jsou definovány a vysvětleny použité pojmy související s touto bakalářskou prací.

1.1 Základní pojmy

Mezi základní pojmy patří: informace, data, proces. Tyto pojmy jsou v následujících podkapitolách více přiblíženy.

1.1.1 Informace

Informace je důležitou částí informatiky. Informace jsou také to, jak působíme na vnější svět a on na nás vzájemným přizpůsobováním.

Tento pojem na nás působí již od středověku, a to v těchto činnostech – obchod, soudnictví a církevní život. Pokud si s někým vyměňujeme informace, vzniká komunikace. Díky naší schopnosti komunikovat a psát jsme schopni informace interpretovat do strojů (počítačů), který může spoustu činností převzít za nás. (1; str. 13)

Důležité je, aby zpráva měla pro příjemce význam. Informaci můžeme brát z několika hledisek. Všechny ale musí splňovat tři požadavky a to syntaktická, sémantická a pragmatická relevance.

Je zde mnoho členění např.: informace operativní, strategické a taktické. Další dělení je třeba podle stupně řízení a existuje jich mnohem více. (2; str. 12)

Zde je pár příkladů, jak může vypadat informace: zpráva, fakt o určité skutečnosti, předmětu či jevu, slohový postup, poznatek o objektivní realitě. (3; str. 27)

Kvalita informace

Informace, která je kvalitní, musí být spolehlivá, důvěryhodná a solidní.

- spolehlivost – podle souladu informace s předlohou,
- důvěryhodnost – podle zabezpečení,
- solidnost – patří sem poctivost, spravedlivost, korektnost, mravnost apod.

Jsou zde i špatné vlivy, které mohou kvalitu informací zhoršit.

- chyba – výsledek špatně provedené operace, ať už omylem, strojem, nebo neúmyslně člověkem,
- šum – může změnit informaci od bezvýznamné, až po zcela užitečnou,
- manipulace s informacemi – jedná se o vědomou chybu lidského původu, bývá obtížně rozpoznatelná. (3; str. 32)

1.1.2 Data

Data můžeme považovat za záznam lidského poznání. Jsou zaznamenávána pomocí symbolů a ty jsou schopné přenosu, uchování, interpretace a zpracování.

Pokud data dobře zpracujeme a dodáme za jasným účelem, vzniká informace. (1; str. 14)

Významným způsobem ovlivnila data i nástup technologie, a to díky automatizaci.

Informační technologie působí do všech sfér osobního života, zaměstnání, chod, způsoby a zvyky. Základním principem těchto technologií je důvěra a víra, že systémy budou pracovat, tak jak mají. Informace, které jsou zpracovávány, mohou být velice citlivé a zranitelné. (3; str. 25)

Data jsou informačním bohatstvím každé firmy. Data do firmy vstupují, jsou tam uloženy a dají se různě zpracovávat. Současné technologie umožňují schraňovat v systému velké množství dat.

Data můžeme rozdělit do mnoha skupin dle pohledu, ze kterého na ně nahlížíme. Prvním z nich je pohled z hlediska doby, která uplynula od jejich uložení.

Dělení dat:

- aktuální – popisuje současný stav reality, je využíván pro získání aktuálních informací. Po ztrátě aktuálnosti se tato data stávají archivními,
- archivní – skladujeme je pro pozdější použití. Slouží k analýzám historie, vývoje nebo změn,
- prognostická – využíváme je jako obraz prognóz, výhledů, záměrů, plánů nebo projektů. (3; str. 126)

1.1.3 Podnik

Podnik můžeme vnímat jako subjekt, kterým jsou informace vyjadřovány, zpracovávány a přenášeny. Dochází zde tedy k výměně vstupů na výstupy. Rozsáhleji je pojem vymezen jako soubor prostředků, zdrojů, práv a dalších majetkových hodnot.

Na podnik můžeme také nahlížet jako na živý, otevřený a komplexní systém. Možnost vnímání podniku zde je, protože v podniku můžeme mít zavedený podnikový informační systém. Podnik je většinou založen podnikatelem či společností.

Za klíčové prvky podniku považujeme zákazníka, konkurenci, dodavatele, dalšími faktory jsou politika, ekonomika, sociální kultura, technologie a legislativa. Důležitými prvky podniku jsou pak lidé, technologie, informace a řízení. (1; str. 16)

1.1.4 Proces

K důležitým pojmům patří i proces.

„Proces je definován jako „soubor vzájemně souvisejících nebo vzájemně působících činností, který přeměňuje vstupy na výstupy“ (ČSN EN ISO 9000:2006). „ (1; str. 19)

Procesy můžeme členit na tři kategorie:

- **řídící procesy** – strategické plánování (vytvářejí podmínky pro fungování ostatních procesů),
- **hlavní procesy** – obstarávají hlavní podnikové funkce, které jsou přímo spojené s uspokojováním potřeb zákazníků,
- **podpůrné procesy** – probíhají uvnitř podniku, mají podpůrný charakter pro základní procesy. (4; str. 43)

Další rozlišení:

- **interní procesy** – firma je má plně pod kontrolou a může jim přidělit odpovědného člověka za jejich chod,
- **externí procesy** – není zde přesně definovaný vlastník, Jsou to procesy, které spadají do oblasti řízení vztahů se zákazníky. (4; str. 44)

1.1.5 Workflow – pracovní postup

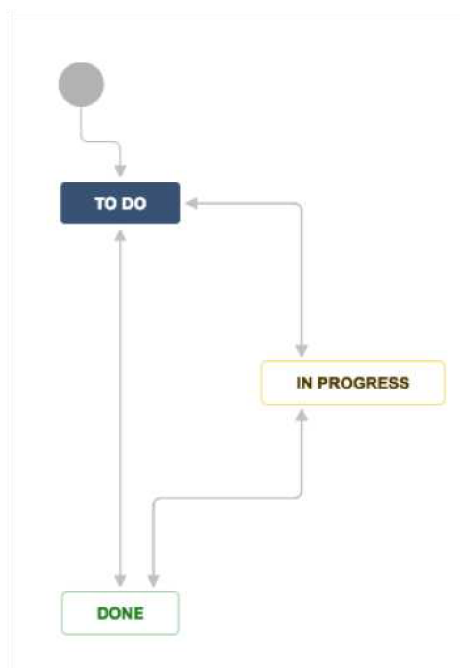
Díky pracovním postupům, které se postupem času staly automatizovanými, byla veškerá práce pro firmy i jednotlivce snazší. Vše bylo ukládáno do počítačů. Pokud byla potřeba kopie, nebyl problém ji vytisknout. Dokumenty byly obvykle ukládány v pořadí, v jakém byly zaregistrovány. (5; str. 96)

Technologie otevřely nové příležitosti pro podnikání. Je nutné identifikovat systém, aby byl správně zaveden. Vyžaduje vytvoření jasné linie pro každého zákazníka. (5 str. 98)

Pracovní postup je posloupnost úkolů nebo nezbytných kroků, které obsahují obchodní proces. Abychom pochopili pracovní postup organizace, je nutné se dívat na věci z pohledu procesu, a ne z organizačního hlediska. (5; str. 106)

Jedná se o prostředek pro modelování, sledování a řízení procesů ve firmě. Pomocí grafického návrhu je možné popsat procesy ve firmě. Tyto šablony jsou v reálném prostředí řízeny systémem automaticky. V každém okamžiku má uživatel přehled o současném stavu procesu – jaké úkoly jsou momentálně zpracovávány, kdo je řeší, jaké jsou termíny. (6; str. 75)

Základní workflow může vypadat takto:



Obrázek 1 Ukázka workflow (Zdroj: 18)

Šedá tečka značí původní stav, tedy vytvořeno. Modrý čtvereček „To do“ je již vytvořený požadavek. Žlutý znamená v procesu, zde se tedy zpracovávají údaje. Zelený je pak hotovo. Šipky představují potenciaální přechody z jednoho stavu do druhého. Workflow může být jednoduché, ale i mnohem složitější, než je toto. (6)

1.2 Informační systém

Informace jsou vstupy a výstupy podnikového informačního systému.

V podnikovém informačním systému lze nalézt části:

1. Které jsou vyměňovány a zpracovávány lidmi, využíváme k tomu i další prvky neverbální komunikace (mimika, gesta). Tahle část je nazývána neformální informační systém.
2. Které jsou založeny na formalizovaných pracovních a informačních tocích realizovaných na základě popsaných politik, cílů, strategií, pravidel a předpisů. Tuto část pojmenováváme jako formální informační systém.
3. Které jsou založeny na strojích (počítačích), kde počítač provádí klíčové operace s informacemi, tj. jejich vyjádření a podobu, zpracování a přenášení.

Informační systém představuje pevnou množinu dílů, které spolupracují za účelem tvorby, shromažďování, zpracování, přenášení a rozšiřování informací. Části systému jsou lidé – uživatelé, zdroje informací, data, řízení a procesy. (1; str. 21)

Aniž by nás to napadlo, pod pojmem informačních systémů si můžeme představit nejen síť počítačů, ale také televize, noviny, rozhlas, kino, knihovna, škola apod. Jeden z dostupných informačních systémů, který máme každý k dispozici a je levný a dostupný je lidský mozek. Když je informační systém správně vymyšlen a používán stává se z něj skvělý pomocník hlavně při náročných činnostech. (3; str. 44)

Při tvorbě IS je důležité určení rolí v rámci jejich odpovědností a pravomocí na projektu, vymezení předpokladů a vlastností, které jsou očekávány. (3; str. 74)

Informační systém je tvořený několika prvky a to hardwarem (technické vybavení), softwarem (programové vybavení) a orgwarem (normy, vnitřní předpisy). (2; str. 10)

Informační systém obsahuje údaje o podstatných skutečnostech, které úzce souvisí s podnikem. Lze je rozdělit do tří skupin:

- **Data o společenských podmínkách** podnikání, tato část obsahuje vnější okolí podniku z makroekonomického hlediska. Patří sem demografické, sociální a politické vlivy, ekonomické trendy. Také sem můžeme zahrnout náklady, dostupnost a produktivitu těchto zdrojů.
- **Data o trhu**, sem patří také vnější okolí podniku. Převážně informace, které se týkají zboží a služeb, poměry konkurentů. Hodnotíme zde aktiva, produktivitu, plány naší firmy.
- **Interní data podniku** jedná se o data, která jsou tvořena uvnitř podniku. Bývají používána k tomu, aby firma mohla komunikovat s vnějším prostředím. Patří sem finance, kapitál, zařízení a stroje. Také sem patří informace ohledně firemních procesů a informačních toků.

Informační technologie představují práci s daty jako je zpracování, ukládání, uchování atd. Když vložíme metody do stroje (počítače) můžeme je nazývat programy. K tomu, abychom mohli softwary používat, potřebujeme i technologie – hardware. Informační technologie jsou metody a procesy určené k předávání informací. (1; str. 22)

Typy uživatelů

Lidé představují důležitý prvek podnikového systému. Lze uvažovat minimálně o dvou základních kategoriích – uživatel a IT personál:

- **Uživatelé** jsou součástí systému, využívají jeho výsledků a zároveň se také podílí na formulaci jeho účelu.
- **IT personál** (informatik, někdy též IT odborník) potřebuje ke své práci určité znalosti a zkušenosti o tvorbě, interpretaci a provozu informačních technologií v dané oblasti. (1; str. 21)

Rozdělení informačních systémů:

1. **ERP** – zaměření na interní podnikové procesy
2. **CRM** – systémy, které souvisí se zákazníky
3. **SCM** – správa dodavatelského řetězce
4. **MIS** – zpracovává údaje z databází dle požadavků manažera (4; str. 77)

1.2.1 ERP systémy

Jedná se o aplikaci, která má schopnost automatizovat a spojovat důležité firemní procesy a funkce v celé firmě. (1; str. 97)

Pracuje s klíčovými daty firmy. Mezi základní funkce ERP systémů, kterými se zabývá, je ekonomické řízení, prodej a marketing, řízení skladů a nákupu, správa lidských zdrojů a výroba. (1; str. 100)

Kategorie ERP

Vybrat správný systém pro naši firmu je velmi těžké. Systémů je totiž mnoho jen na českém trhu. Jedním z třídění je dle velikosti zákazníka:

- velké celopodnikové systémy,
- střední celopodnikové systémy,
- menší celopodnikové systémy,
- menší obchodní systémy,
- malé a domácí systémy.

Jednodušeji se dá říci velké, střední a malé společnosti.

ERP v podniku zahrnuje především činnosti, které souvisí se správou stěžejních dat. Pracuje s dlouhodobým, střednědobým i krátkodobým plánováním zdrojů, s plánováním realizace a výroby, se zpracováním veškerých aktivit finančního účetnictví.

ERP se tedy zabývá dvěma funkčními oblastmi:

- logistikou – veškerá podniková logistika, nákup, skladování, výroba, distribuce,
- financemi – nákladové a investiční účetnictví. (6; str. 67)

1.2.2 CRM systémy

Informační systémy se dnes především zaměřují na podporu podniku prodat svoje výrobky či služby. Firmy se snaží díky informačním systémům být v trvalejším vztahu se zákazníky. Firmy využívají mnoho způsobů a to: posílání různých publikací, zasílání

elektronické pošty, diskuze na webu, call centra. Důležitým úkolem systému se pak tedy stává vytváření a zlepšování vztahu se zákazníkem. Aplikace, která tohle zajišťuje, je pak nazývána CRM.

CRM je komplex software, hardware, podnikových procesů a personálních dat, které průběžně zajišťují vztahy se zákazníky. Jedná se hlavně o oblast marketingu, podporu prodeje a podporu zákaznických služeb. (6; str. 90)

1.2.3 SCM systémy

Řízení celého dodavatelského řetězce z jednoho místa se stává velkou konkurenční výhodou. Díky systémům SCM dochází ke zkracování času a zvyšování spolehlivosti dodávky zákazníkovi. Řízení dodavatelských řetězců se snaží optimalizovat procesy a nástroje, které vedou k maximální efektivitě provozu všech prvků celého řetězce s ohledem na zákazníka. Systémy SCM tedy umožňují propojit dodavatele a odběratele. Mohou sdílet informace, spolupracovat, koordinovat celkový postup, plánovat a zdokonalit tak celý řetězec. (14; str. 31)

1.2.4 MIS systémy

Cílem těchto informačních systémů je hlavně zpřístupnit různé přehledy či sestavy jako například: počty objednávek, zisk v jednotlivých měsících, celkovou hodnotu prodaného zboží, atd. S těmito systémy pracují řídicí pracovníci hlavně v oblastech kontroly výkonnosti, kteří zkoumají výkonnost nejen jednotlivých pracovníků, ale také celého oddělení. Mohou provádět různé analýzy dat. Rozhodně nenahrazují vedoucího pracovníka, jen se mu snaží pomáhat v rozhodovací činnosti. (15; str. 12)

Systémy MIS pracují efektivně nad datovými sklady. V datových skladech jsou dlouhodobě uložena strukturovaná data.

Systémy MIS umožňují:

- vytváření a zpracování sestav,
- uplatnění některých funkcí v rámci multidimenzionality,
- náhled do současných i historických dat. (15; str. 18)

1.3 Informační technologie

V této části si představíme principy a charakteristiky, které jsou spojeny s technickými prostředky podnikového informačního systému. Třeba základními informačními technologiemi, včetně těch, jimiž je zajištěna komunikace v podniku. (1; str. 25)

Technické prostředky (hardware)

Hardware je široký pojem, do kterého patří počítače, přídavná zařízení, elektronika a další součástky ICT. (1; str. 26)

Informační technologie (software)

Informační technologie znázorňují procesy spojené s informacemi. Software můžeme dělit:

- aplikační software – aplikace informačních technologií ve firmě,
- základní software – např. operační systémy, technologie spojené s workflow,
- software podporující rozvoj informačního systému. (1; str. 27)

1.4 Infrastruktura podnikového informačního systému

Infrastruktura podnikového informačního systému je základem pro současné i budoucí aplikace informačních technologií. Je tvořena technickými nástroji, informačními technologiemi, informatiky, řízením podnikové informatiky a IT governancí.

Infrastruktura podnikového informačního systému používá:

- infrastrukturu podniku,
- infrastrukturu segmentu trhu,
- veřejnou infrastrukturu,
- infrastruktura podnikového informačního systému.

Služby podnikového systému jsou propojeny technickými prostředky a informačními technologiemi a také jsou směřovány na IT management a IT governancí. (1; str. 38)

Jsou zde dvě skupiny, na které můžeme služby dělit:

- Do první patří **služby spojené s technickými prostředky a informačními technologiemi**. To obsahuje např.: služby internetu – e-mail, web, telekomunikaci (fax), řízení rizik a bezpečnost podniku, správu dat, správu IT a správu aplikací, které jsou využívány podnikem.
- Druhou skupinou jsou pak **služby orientované na IT management a IT governance**, což tvoří správně fungující informační systém a jeho rozvoj, IT vzdělávání, IT výzkum a vývoj, který umožňuje nové možnosti a vývoj. (1; str. 39)

Od návrhu informačního systému se očekává, že bude flexibilní, agilní a efektivní. Tyto vlastnosti dosáhneme, pokud je systém: kompatibilní, modulární/transparentní a pracovníci mají vhodnou úroveň znalostí, zkušeností a dovedností. (1; str. 40)

1.5 Analýzy

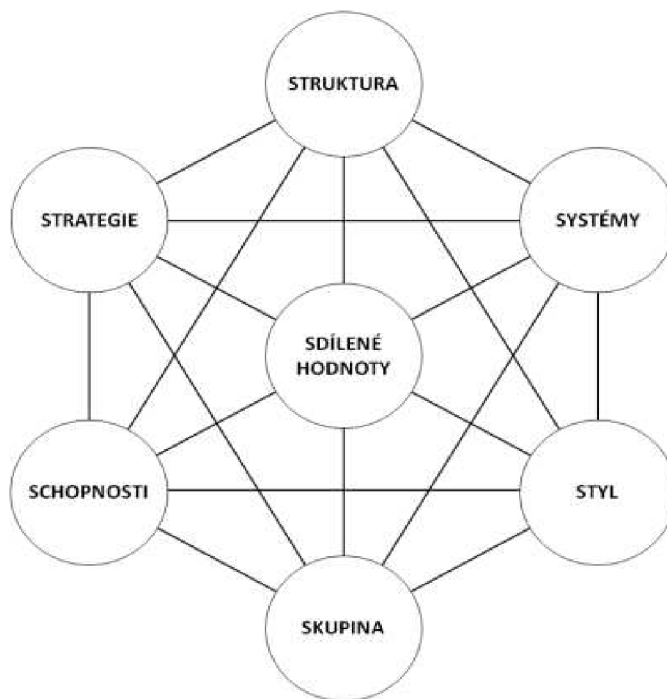
Předtím, než začneme systém navrhovat, je nutné provést některé analýzy. Musíme zjistit, co může projekt ohrozit; určit čas na projektu strávený a udělat ekonomický rozpočet.

1.5.1 7S analýza

U analýz by se také měla posoudit strategie, organizace, firemní kultura a dalších faktorů firmy. Při pohledu na firmu pomocí analýzy 7S je to množina sedmi základních aspektů. Ty se vzájemně ovlivňují a dohromady rozhodují o tom, jak bude stanovená strategie naplněna. Mezi těmito faktory je potřeba hledat klíčové úspěchy všech firem. Používá se pro hodnocení kritických faktorů organizace, je používána zejména v podnicích. Navrhli ji američtí konzultanti ze společnosti McKinsey&Company. V praxi se používá například ve strategickém auditu, strategickém řízení a v řízení změn.

Model je nazýván „7S“, protože obsahuje sedm faktorů a všechny začínají v anglickém překladu na S:

- strategie,
- struktura,
- systémy,
- styl,
- spolupracovníci,
- schopnosti,
- sdílené hodnoty. (28)



Obrázek 2 7S analýza (Zdroj:29)

Strategie – je to způsob, jakým společnost udržuje konkurenční výhodu. Vedení podniku hledá odpovědi na otázky, jakým stylem realizovat podnikové strategie, jak dosáhnout stanovených cílů, jak vést konkurenční boj a jak ovlivnit zákazníka.

Struktura – rozdělení činností podniku a koordinační mechanismy. Tato část řeší členění společnosti, hierarchii, koordinaci jednotlivých útvarů společnosti a komunikace v rámci podniku.

Systémy – formální mechanismy pro měření, odměňování a alokaci zdrojů. V rámci tohoto bodu jsou definovány hlavní systémy řízení společnosti, nástroje pro zpětnou vazbu, aktuálnost a odbornost interních dokumentů.

Styl – komunikace a jednání manažerů s podřízenými, zákazníky či spolupracovníky. Řeší efektivnost vedení společnosti, správnost a rychlost rozhodnutí a pracovní prostředí společnosti.

Spolupracovníci – specializace zaměstnanců, řízení lidských zdrojů a jejich motivace. Odpovídá na otázky ohledně kvalifikace a specializace zaměstnanců, způsobu výběru nových pracovníků, možností osobního rozvoje či motivace pracovníků.

Schopnosti – návyky, schopnosti a znalosti zaměstnanců podniku. Určuje nejlepší a nejsilnější vlastnosti podniku.

Sdílené hodnoty – podniková kultura a etiketa společnosti. Definiuje základní hodnoty podniku, úroveň a růst podnikové kultury. (30)

1.5.2 SWOT analýza

Při SWOT analýze se identifikují faktory a skutečnosti, které pro objekt představují silné a slabé stránky, příležitosti a hrozby z okolí. Tyto faktory jsou poté ohodnoceny či verbálně popsány ve čtyřech kvadrantech tabulky. Tato analýza se používá při taktickém, operativním i strategickém řízení. (16; str. 120)

SWOT analýza se většinou řadí na závěr analýzy, jelikož jsou v ní shrnuté všechny důležité informace. (16; str. 123)

	Nápomocné v dosažení cílů	Škodlivé v dosažení cílů
Interní faktory	S Silné stránky Strengths	W Slabé stránky Weaknesses
Externí faktory	O Příležitosti Opportunities	T Hrozby Threats

Obrázek 3 SWOT analýza (Zdroj: 17)

1.5.3 Riziková analýza

Jedná se o nikdy nekončící proces, který se děje v průběhu všech fází projektu. Tým, který projekt vytváří, by měl mít ponětí o tom, jaká nebezpečí mohou nastat a ohrozit náš projekt. Je potřeba na tyto hrozby připravit opatření a nejlépe jim předejít. Řízení rizik zahrnuje analýzu rizik a jejich sledování. Analýza se provádí na začátku každého projektu v rámci fáze plánování.

Základní pojmy

Hrozba

Většinou se jedná o událost, která má nepříznivý vliv na bezpečnost našeho aktiva nebo jej může zlehka či vážně poškodit. Škoda, kterou napáchá hrozba při působení na dané aktivum je dopad hrozby. Mezi hlavní vlastnosti hrozby patří její úroveň. Úroveň se hodnotí dle následujících faktorů:

- nebezpečnost – schopnost hrozby negativně působit na aktiva,
- přístup – pravděpodobnost, že hrozba získá přístup k aktivu,
- motivace – zájem vyvolat hrozbu vůči aktivu.

Zranitelnost

Jedná se o slabinu zkoumaného aktiva, která může být využita k poškození aktiva. Říká nám, jak citlivé je aktivum vůči dané hrozbě. Zranitelnost vzniká při vzájemném kontaktu hrozby a aktiva. Základní vlastností je úroveň. Úroveň se hodnotí podle citlivosti a kritičnosti.

Protiopatření

Postup je speciálně navržen pro snížení působení hrozby. Vznikají z důvodu úplné eliminace dané hrozby, nebo se snaží zmírnit její následky. Hodnotí se dva hlavní parametry – efektivita a náklady. Efektivita spočívá ve znázornění, o kolik se zmenší účinek hrozby. Náklady se pak počítají, jde o číselné vyjádření. Započítávají se náklady na pořízení, zavedení a provozování. V důsledku se hledají nejúčinnější protiopatření s co nejmenšími náklady. (9; str. 52)

Metoda RIPRAN

Zkratka je vytvořená ze slov RIsk PROjekt ANalysis. Postup je popsán již v předcházejícím textu. Metoda RIPRAN však umožňuje i tzv. verbální ohodnocení, jedná se tedy o slovní popis. (10, str. 80)

Tato metoda se skládá ze čtyř podstatných kroků:

- identifikace nebezpečí projektu,
- kvantifikace rizik projektu,
- reakce na rizika projektu,
- celkové posouzení rizik projektu.

Pro ohodnocení projektových rizik umožňuje metoda RIPRAN jak numerickou, tak verbální verzi. (10, str. 80)

Tabulka 1 Tabulka verbálních hodnot pravděpodobnosti (Zdroj: 10, str. 80)

Vysoká pravděpodobnost	VP	Nad 66 %
Střední pravděpodobnost	SP	33 % - 66 %
Nízká pravděpodobnost	NP	Méně než 33 %

Tabulka 2 Tabulka verbálních hodnot nepříznivých dopadů na projekt (Zdroj: 10, str. 80)

Velký nepříznivý dopad na projekt	VD	Ohrožení cíle projektu
		Ohrožení koncového termínu projektu
		Možnost překročení celkového rozpočtu projektu
		Škoda více než 20 % z hodnoty projektu
Střední nepříznivý dopad na projekt	SD	Škoda 0,51 – 19,5 % z hodnoty projektu
		Ohrožení termínu, nákladů, resp. zdrojů některé dílčí činnosti, což bude vyžadovat mimořádné akční zásahy do plánu projektu
Malý nepříznivý dopad na projekt	MD	Škody do 0,5 % z celkové hodnoty projektu
		Dopady vyžadující určité zásahy do plánu projektu

Tabulka 3 Vazební přiřazení pro verbální hodnoty rizika (Zdroj: 10, str. 80)

	VD	SD	MD
VP	Vysoká hodnota rizika – VHR	Vysoká hodnota rizika – VHR	Střední hodnota rizika – SHR
SP	Vysoká hodnota rizika – VHR	Vysoká hodnota rizika – VHR	Nízká hodnota rizika – NHR
NP	Nízká hodnota rizika – NHR	Nízká hodnota rizika – NHR	Střední hodnota rizika – SHR

1.5.4 Časová analýza

Rozdělení času pro jednotlivé úkoly je další klíčová část projektu. Při sestavování je nutné uvažovat všechny vlivy – cíl, čas a zdroje.

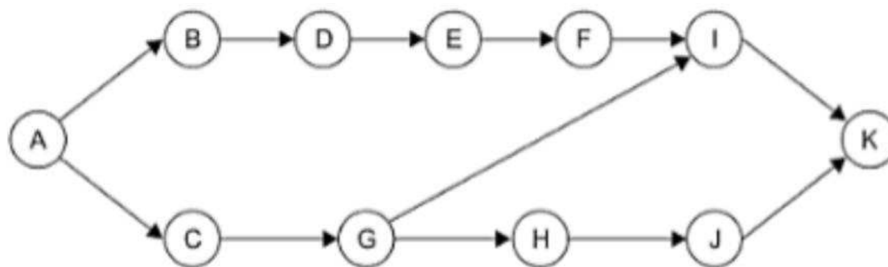
V první části je nutné určit činnosti, které jsou ke splnění cíle potřeba. V první fázi neřešíme souslednost, ale jejich výčet. V druhé fázi je potřeba určit návaznost všech činností a určit vazby mezi nimi. Některé činnosti na sebe třeba musí navázat a nelze je realizovat, pokud již není hotová předcházející.

Pro znázornění výsledného seřazení činností se využívají síťové grafy. Ty mají následující pravidla:

- graf má jeden začátek,
- graf má jeden konec,
- graf je acyklický (10, str. 50).

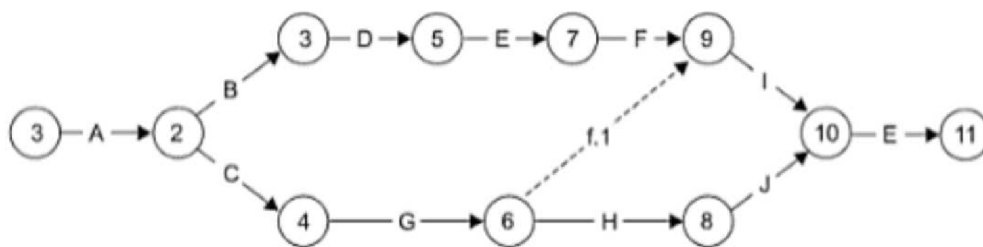
Síťové grafy můžeme podle způsobu zobrazení činností dělit takto:

- Uzlově definovaný – činnost je představována uzlem, který je ohodnocený délkou trvání dané činnosti, hrany představují závislosti a návaznost jednotlivých činností. (10, str. 52)



Obrázek 4 Uzlově definovaný síťový grad (Zdroj: 10, str. 52)

- Hranově definovaný – činnost je znázorněna pomocí orientované ohodnocené hrany, uzly znamenají začátek a konec činnosti. (10, str. 53)



Obrázek 5 Hranově definovaný síťový graf (Zdroj: 10, str. 53)

- Úsečkový (liniový, Ganttův) diagram – s pomocí časové osy jsou činnosti zobrazené jako ohraničené úsečky, přičemž délka této úsečky značí trvání činnosti. (zdroj: 10, str. 52)

PERT

Tato metoda používá pro stanovení doby trvání jednotlivých činností vážený průměr založený na 3 pravděpodobnostních odhadech a to optimistickém, pesimistickém a realistickém. Síťový graf postavený na tomto principu proto většinou využíváme, pokud nejsme přesně a s jistotou schopni stanovit dobu trvání. To se typicky stává u projektů, u kterých je podstatou změna, jejíž průběh nelze předem předvídat, protože se nejedná o opakované činnosti. (10; str. 60)

Pro výpočet váženého průměru a zároveň i doby trvání jednotlivých činností využijeme následujícího vzorce, který nám umožní převod na deterministický model:

$$t = \frac{a + b + 4m}{6}$$

Kde

- a – optimistický odhad,
- b – pesimistický odhad,
- m – realistický odhad. (10, str. 60)

Jelikož se jedná o odchylky, je nutné ze statického hlediska spočítat také rozptyl a směrodatnou odchylku. Rozptyl zde udává, jak moc je optimistický a pesimistický odhad rozptýlen kolem realistického odhadu. (9; str. 21)

$$\sigma^2 = \frac{(b - a)^2}{36}$$

S rozptylem související směrodatná odchylka. Je matematicky druhou odmocninou z rozptylu a ukazuje nám, jak se od sebe hodnoty optimistického a pesimistického odhadu liší. (9; str. 22).

$$\sigma = \frac{(b - a)}{6}$$

1.6 Programovací jazyk

K tomu, aby aplikace fungovala i bez drahých pluginů je nutné využít tento programovací jazyk. Díky němu aplikaci upravíme dle našich představ. V jazyku Java si plugin můžeme navrhnout i sami.

1.6.1 Java

Java je univerzální, moderní programovací jazyk původně vyvinutý společností Sun Microsystems a v současné době vlastněný společností Oracle Corporation. Klíčovou vlastností jazyka je nezávislost na platformě: to znamená, že stejný program Java lze spustit na více platformách bez jakýchkoli změn. Tento princip je také známý jako „psát jednou, spouštět kdekoli“ (nebo WORA).

Java je po mnoho let jedním z nejpobulárnějších programovacích jazyků. Tento jazyk používá velká komunita vývojářů po celém světě.

Java se používá v našich chytrých telefonech, v odvětví finančních služeb, telekomunikacích, vestavěných systémech a v mnoha dalších oblastech. Lékařské aplikace jej používají k ukládání dat o pacientech, počítačové hry jako Minecraft jsou vytvářeny pomocí Java; některé vývojové nástroje by bez něj neexistovaly. (11; str. 31)

Tento jazyk je jedním z interpretovaných a překládaných jazyků. Zdrojový text se však nepřekládá do strojového kódu ale do univerzálního jazyka, který se jmenuje univerzální kód. (12; str. 14)

Funkce Java

Jak jsme již zmínili, nejdůležitější vlastností Java je nezávislost na platformě. Další důležitou funkcí je jednoduchá a jasná syntaxe. Mnoho prvků jazyka je snadno čitelných a jsou široce používány v jiných programovacích jazycích, jako jsou C / C ++, C #, JavaScript a Go.

Pokud jste někdy psali programy v C / C ++, pak víte, že ruční čištění paměti může vést k chybám v kódu. Naštěstí má Java sběrač odpadků, který průběžně automaticky čistí paměť od nepoužívaných objektů. (13)

Rozdělení Java/ Verze

K tomu, abychom mohli pracovat v Javě, potřebujeme dva programy, které je možné stáhnout v sadě JDK – Java Development Kit. Po nainstalování tedy budeme moci kompilovat i programy spouštět. Prvním programem je javac, kompilátor Javy. Druhý je java, což je standardní interpret, který se označuje jako spouštěč aplikací. (11; str. 31)

2 ANALÝZA SOUČASNÉHO STAVU

V této části bude charakterizovaná společnost K-net, s. r. o. a uvedu bližší informace o firmě.



Obrázek 6 Logo společnosti (Zdroj: 20)

2.1 Základní informace o společnosti

Název společnosti:	K-net Technical International Group, s.r.o.
Provozovna:	Olomoucká 703/170, 627 00 Brno
Zapsáno:	7. 6. 1993
IČ:	47916745
DIČ:	CZ47916745
Právní forma:	Společnost s ručením omezeným
Počet zaměstnanců:	cca 50
Datum vzniku a zápisu:	7. 6. 1993
Právní forma:	Společnost s ručením omezeným
Předmět podnikání:	<ul style="list-style-type: none">- výroba, obchod a služby neuvedené v přílohách 1 až 3 živnostenského zákona- výroba, instalace, opravy elektrických strojů a přístrojů, elektronických telekomunikačních zařízení
Statutární orgán:	Jednatel; Ing. Tomáš Knettig
Způsob jednání:	Jednatel jedná za společnost samostatně
Základní kapitál:	10 000 000,-Kč (19)

2.2 Popis společnosti

K-net Technical International Group s.r.o. je česká firma, která úspěšně působí na trhu informačních technologií již 27 let. Zákazníkům pomáhá uspořádat výpočetní systémy a sítě, aby vyhovovaly současným i budoucím potřebám z pohledu dostupnosti, bezpečnosti a spolehlivosti v dané oblasti.

Hlavním předmětem činnosti jsou sofistikovaná řešení v oblasti serverových technologií a počítačových sítí, bezpečný přístup uživatelů k aplikacím kdykoliv a odkudkoliv, terminálová řešení s využitím vlastních unikátních technologií, poskytování služeb cloudového centra a ucelená nabídka služeb zahrnující technickou podporu, organizační podporu a zajištění provozu informačních technologií.

Plní nejvyšší normy kvality, jsou držiteli certifikátů ISO 9001, ISO 14001, ISO/IEC 27001 a ISO/IEC 20000 a na základě těchto certifikací vyhláší politiku integrovaného systému managementu (Politika IMS). (20)

..

2.3 Služby společnosti K-net

Firma se zabývá různými službami v oblasti IT. Nabízí různá řešení jak pro domácnosti, firmy, veřejnou správu, ale i pro školy. Nejprve vždy provedou analýzu potřeb a současného stavu. Nabízí různá řešení přímo zákazníkům na míru. **A zde je detailnější popis služeb:**

QNET

Qnet je profesionální připojení k internetu pro firmy s garantovanou rychlostí a dostupností. K-net nabízí připojení k internetu prostřednictvím mikrovlnných, metalických a optických spojů. Společnost je připojena k primárním poskytovatelům prostřednictvím optického kabelu a záložním bezdrátovým spojením. Úmyslně byly zvoleny 2 odlišné technologie připojení k primárním poskytovatelům, aby se minimalizovala možnost výpadku.

Qvoice

Qvoice je provozovaná hlasová služba VoIP, jedná se o telefonování přes internet. Telefonování přes internet umožňuje výrazně snížit náklady na telefonování, a navíc

získat služby vyspělé digitální ústředny (záznamník, přeposílání hlasových zpráv do mailu, forwarding v rámci sítě Qvoice). To všechno jako standard nabízených služeb.

Vzdělávání v IT

Firma nabízí vzdělávací kurzy pro firmy a organizace, pro management firem, pro IT specialisty a pro školy.

Profesionální IT služby

Naším cílem je pomoci Vám uspořádat výpočetní systémy a sítě, aby vyhovovaly Vaším současným i budoucím potřebám z pohledu dostupnosti, bezpečnosti a spolehlivosti v dané oblasti. Účelem služby je zajistit uživatelům definovanou podporu při zpracování a řešení Vašich potřeb a požadavků, zejména v organizační rovině a v místech Vámi určených.

Mají mnoho zkušeností a znalostí, které díky našemu dynamickému oboru neustále rozšiřujeme a prověřujeme v praxi. Rádi se o své poznatky dělíme a seznamujeme s nimi své posluchače tak, aby pro ně byly užitečné a jejich práce díky nim efektivnější.

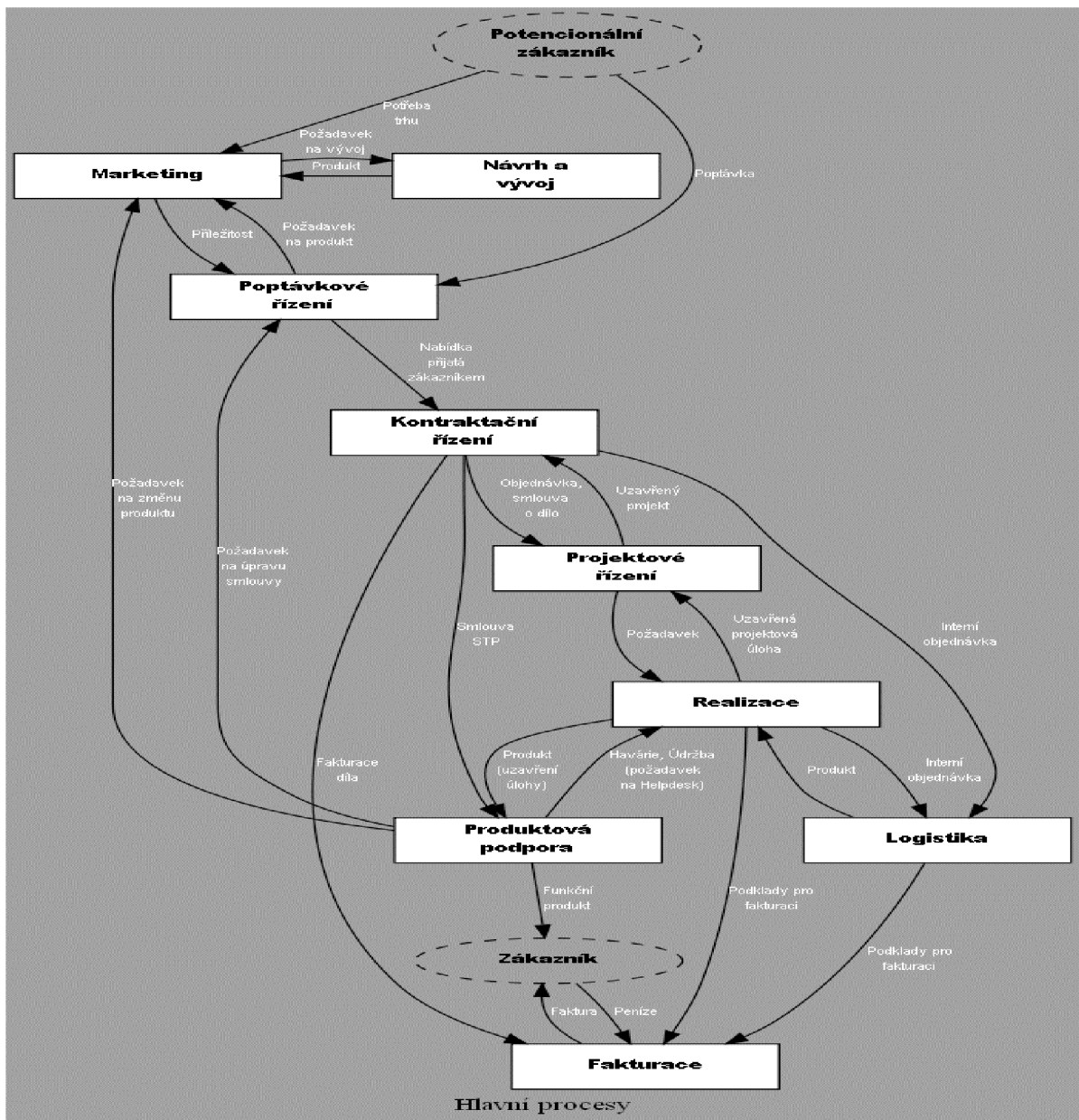
Poskytování IT technologií

K-net také nabízí spoustu dalších služeb, jako jsou:

- správa a ochrana sdílených dokumentů,
- aplikace v cloudu,
- pronájem licencí software,
- zálohování dat a systémů,
- archivace dat a systémů,
- hosting serverů,
- housing serverů,
- monitoring a dohled,
- monitoring fyzického prostředí.

2.4 Popis hlavních firemních procesů

Následující obrázek popisuje tok informací ve firmě.



Obrázek 7 Popis hlavních firemních toků (Zdroj: vlastní)

2.5 Konkurence

V dnešní době na trhu existuje mnoho firem, které nabízejí tyto služby. Tato firma, ale nabízí individuální přístup, což je její výhodou. Nenabízí několik variant, ale s každým klientem se domluví na přesných požadavcích. Díky tomu je přístupná jak pro domácnosti, tak menší firmy i podniky ze státního sektoru. Protože firma se pohybuje již 27 let ve svém oboru, má dobré dodavatele, kontakty a používá nejmodernější vybavení dostupné na trhu.

2.6 Používané systémy IT

Společnost K-net využívá několik jednotných systémů pro evidenci personalistiky, komunikaci, sdílení dokumentů apod. Mezi ty nejvýznamnější patří:

- **NetInfobase** – Znalostní báze K-net. Je tam dokumentace k projektům, zápisy z porad, žádanky, popis pracovních pozic a jejich náplně. Také zde jsou firemní směrnice, vzory smluv a další důležité dokumenty pro firmu.
- **Dashboard** – ticketovací a vykazovací systém. Zaměstnanci a především helpdesk, tam se zakládají úlohy, které jsou jasně definované. Mají zakladatele, řešitele, patří k určité smlouvě, a především se na ně vykazuje čas.
- **Gmail** – poštovní klient pro e-mailovou komunikaci.
- **Google Kalendář** – nástroj pro plánování schůzek a interních školení.
- **Git** – webová služba, která podporuje vývoj softwaru. GitHub nabízí bezplatný Webhosting. Git postupně ukládá historii projektu, díky tomu je možné vrátit se do různých verzí projektu. Další předností Gitu je větvení projektu.
- **IntelliJ IDEA** – jedná se o editor vývojového prostředí pro programování. Je navrženo tak, aby zlepšilo produktivitu vývojářů. Jeho inteligentní editor, analyzátor kódu a výkonná sada refaktorování podporuje širokou škálu programovacích jazyků, rámců. Umožňuje psát, ladit, testovat a učit se kód bez problémů. Zaznamenává a opravuje chyby na všech úrovních.
- **Money S3** – Jedná se o účetní podnikový systém. Nabízí přehledný seznam fakturací, spolehlivé účetnictví a mzdy, vyhodnocování obchodu a detailní analýzy.

- **OS Windows** – podporovaný operační systém v aktuální verzi. Operační systém je základní program počítače, který se spouští zapnutím počítače a končí jeho vypnutím. Verzí Windows je mnoho, například: klasická verze, mobilní verze a serverová verze.
- **MS Office** – balík kancelářských nástrojů. Jsou používány pro práci a tvorbu dokumentů. Nabízí programy jako je Word, Excel, PowerPoint, Access, Outlook, OneNote a spoustu dalších.
- **Citrix** – technologie, která nabízí uživatelům bezpečný přístup k firemním datům a aplikacím. Nabízí přístup ze všech zařízení. Tato technologie zvyšuje produktivitu uživatelů, zajišťuje maximální bezpečnost a optimalizuje fungování vaší sítě. Sama společnost K-net tyto služby nabízí a využívá.

Pravidla pro používání těchto výše zmíněných informačních systémů jsou popsána v příslušných směrnících a zaměstnanci jsou s nimi seznámeni buď formou školení při nástupu do práce, nebo v průběhu vykonávání práce.

2.7 7S analýza

Strategie

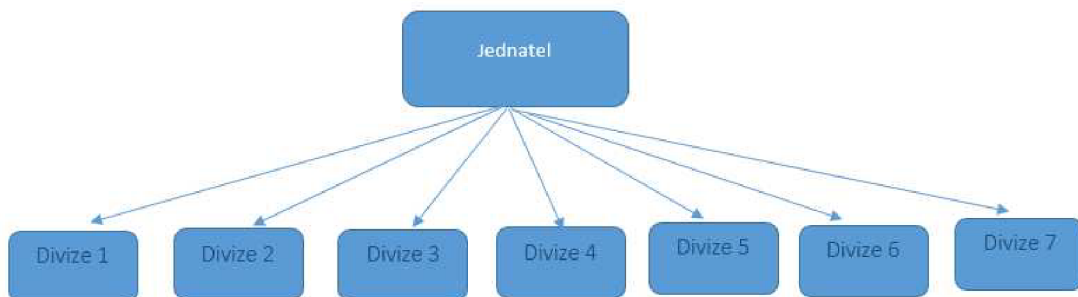
Základním posláním společnosti je vývoj a prodej svých služeb. Firma chce být na trhu velmi úspěšná a žádaná. Do budoucna by chtěla zvýšit povědomí o firmě mezi zákazníky. Těmito kroky očekává větší obrát firmy a zvýšení tržeb. Důležitou součástí strategie podniku je neustálý vývoj nových produktů a inovace stávajících dle požadavků zákazníků.

Struktura

Společnost K-net Technical International Group s.r.o. je společnost s ručením omezeným působící v České republice už 27 let. Má celkem čtyři pobočky, a to v Praze, Brně, Olešnici a Ostravě.

Zde bych chtěla představit organizační strukturu firmy K-net Technical International Group, s.r.o. Firma je rozdělena do několika divizí.

- Divize 1 – komunikace a inspekce. První divize je rozdělena do tří částí: komunikace a personál, informační a telekomunikační systém a poslední inspekce a zprávy.
- Divize 2 – obchod a marketing. Druhá divize je rozdělena na marketing, vztahy se zákazníky a obchod.
- Divize 3 – ekonomika a majetek. Třetí je rozdělena na příjmy, výdaje, majetek a evidenci.
- Divize 4 – výroba. Část výroby se zabývá přípravou, logistikou a realizací.
- Divize 5 – kvalita a kvalifikace. Pracovní úsek kvality a kvalifikace má na starosti výstupný kontrolu, korekci a vzdělávání.
- Divize 6 – propagace. Oddělení propagace je rozděleno na veřejnost, úvodní služby a směřování veřejnosti.
- Divize 7. V této poslední části firmy jsou správci, pracovníci úklidu atd. Jedná se především o externí pracovníky.



Obrázek 8 Struktura firmy K-net (Zdroj: vlastní)

Systemy

Komunikace ve firmě probíhá osobně, nebo pomocí telefonů a e-mailů. Mezi firemní aplikace patří Dashboard a NetInfobase, ty byly zpracovány zaměstnanci firmy. Se zákazníkem komunikují převážně osobně. Používané systémy firmy, jsou již konkrétněji popsány v bodě 2.6.

Styl

Styl a jednání vůči zákazníkům je pro firmu velmi důležité. Firma nabízí individuální řešení, není to pro ni problém a každému zákazníkovi vždy najde optimální řešení.

Ve firmě panuje demokratický styl vedení. Vedení pravidelně seznamuje pracovníky s novými změnami a pravidly. Pro názory a připomínky dostávají zaměstnanci dostatečný prostor v týdenních reportech. Konečné rozhodnutí má vždy vedoucí zodpovídající za daný úkol.

Spolupracovníci

Zakladatel a majitel firmy je znalý v oblasti podnikání i informačních technologií. Firma má asi 50 zaměstnanců, většina je ve firmě dlouhodobě. Zaměstnanci jsou převážně muži. Průměrný věk pracovníků je 35 let.

Schopnosti

Firma si zakládá na vysoké oborové způsobilosti u svých pracovníků. Všichni zaměstnanci jsou dlouhodobě zaměstnáni a mají v daném oboru velmi kvalitní zkušenosti. Pracovníci se zúčastňují častých školení na nové technologie a nové produkty.

Sdílené hodnoty

Nejdůležitější věc pro vlastníky podniku je mít stabilní příjem a nabízet lidem produkty, ke kterým se budou zákazníci rádi vracet. Je důležité, aby měli všichni zaměstnanci zájem na prosperitě firmy. Pro sdílení hodnot společnosti zaměstnanci je důležité, aby měli ponětí o strategiích organizace a o směru, kam se společnost bude ubírat. Podnik se tyto informace snaží předávat prostřednictvím schůzí jednotlivých útvarů.

2.8 SWOT analýza

Pomocí této analýzy je provedeno shrnutí informací o firmě. Díky této analýze je jednoduché popsat ve formě jednoznačné tabulky současný stav, odhalit silné a slabé stránky a možné příležitosti či hrozby. SWOT analýza vychází již ze zmíněných informací.

Tabulka 4 SWOT analýza firmy (Zdroj: vlastní zpracování)

Silné stránky (strengths)	Slabé stránky (weaknesses)
<ul style="list-style-type: none">• stálá klientela• dlouholeté zkušenosti• mají vizi• nezádluženost podniku• dlouhodobí zákazníci• individuální řešení• dobré jméno společnosti• vysoká spokojenost zákazníků• přehledné webové stránky• zkušené vedení a zaměstnanci• široký rozsah obsluhovaných technologií a poskytovaných služeb	<ul style="list-style-type: none">• nedostatečný přehled o informacích a potřebách zákazníků• prodlužování řešení interních problémů• reporting interních záležitostí firmy• masivní expanze – firma je založená na individualizovaných službách.
Příležitosti (opportunities)	Hrozby (threats)
<ul style="list-style-type: none">• zavedení nového informačního systému• zavedení dílčí části informačního systému• rozšíření sortimentu• rozšíření nabídky služeb pro další oblasti• využití dobrých referencí• podchycení nových technologií	<ul style="list-style-type: none">• příchod nové konkurence• ztráta zákazníků• ztráta dat• špatná reklama firmy• zvýšení provozních nákladů• odliv klientely v případě nepřizpůsobování se novým trendům• rychle se měnící technologie

Mezi silné stránky této firmy tedy patří to, že jsou již několik let na trhu a mají kvalifikované zaměstnance, kteří svou práci odvádí velice dobře.

Ze SWOT analýzy vyplývá, že firma potřebuje nový informační systém. Věcí, které by se daly najít a vylepšit je určitě mnoho, ale moje práce se bude zabývat konkrétně dílčí částí tohoto informačního systému. Pokud půjde navrhnout tato malá část, je možné pak

implementovat celý systém. Současný systém žádanek má pár věcí, které je potřeba odstranit a změnit.

Firma používá hodně aplikací přímo od společnosti Atlassian, proto i zde by nejradši zvolila opět jejich aplikaci.

2.9 Navrhované možnosti řešení

V této fázi má společnost několik možností, jakou volbu inovace informačního systému zvolí. Na výběr máme ze dvou možností – koupě hotového řešení nebo kompletní vývoj IS na míru.

Koupě nového řešení

Hotové řešení IS je variantou s nejrychlejší realizací. Firma se k němu může rozhodnout v případě, že chce rychlé, ozkoušené a stabilní řešení, které je již odzkoušenou řadou zákazníků. Získá tak velmi rychle IS, který je možno nasadit téměř okamžitě, na základě výběrových kritérií z široké škály produktů na trhu. Hlavními výhodami je kromě rychlosti také ověřená funkčnost, rychlá dostupnost podpory řešení problémů, existence mnoha uživatelů, a tudíž jejich možné sdílení zkušeností, a také nižší náklady z dlouhodobého hlediska.

Kompletní vývoj IS na míru

Druhou variantou je pak vývoj nového řešení na míru, který je každopádně nejpřesnější variantou s cílem splnit veškeré zákaznické požadavky a mít právě takové funkcionality, které firma potřebuje, neinvestovat do žádných částí navíc. Tato varianta je časově nejnáročnější, nese s sebou několik rizik, zejména chybovosti na základě faktu, že se jedná o nové neozkoušené řešení. Druhým negativem je finanční nákladnost, která bývá obvykle značně vyšší, ve srovnání s jinými variantami.

3 VLASTNÍ NÁVRH ŘEŠENÍ

V této části se budu zabývat návrhem informačního systému pro popisovanou firmu. Podnik momentálně má informační systém, který se zabývá žádankami, ale již není pro firmu perfektní. Je zde potřeba udělat pár změn, aby se celý proces žádanek urychlil a zaměstnanci i vedení byli spokojenější. Návrh bude vytvořený tak, aby nahradil současnou aplikaci žádanek s dalšími požadavky firmy. Následně bude popsána implementace a ekonomické zhodnocení. Na závěr budou shrnuty přínosy této práce.

Firma měla již na začátku jasno v tom, že bude potřebovat informační systém šitý na míru. Není možné tedy použít již některý z vyvinutých softwarů, jelikož se bude jednat pouze o „plugin“ do obrovského řídicího informačního systému.

3.1 Předprojektová fáze

V předcházejících kapitolách vyplynulo, že je potřebné vyvinout nový systém pro žádanky. K tomu, abychom mohli navrhnout perfektní systém pro firmu K-net je nutné prvně analyzovat současný systém. Musíme ho popsat, navrhnout změny a začít tvořit nový systém.

3.1.1 Zhodnocení současného stavu

Momentálně systém vypadá takto:

Stav	Uživatel	Datum počátku	Rozhodnutí	Datum konce	Komentář
Žádanka	Marečková Veronika	17.10.2019 19:28		null	

Žádanka obecná

Jméno žadatele: **Marečková Veronika** zastupuje uživatele: Boháček Jiří

Role: [vypádník]

Vedoucí oddělení: [vypádník] Vedoucí divize: [vypádník]

Oddělení: [vypádník] Divize: [vypádník]

Předmět žádosti: [vypádník] Termín (do kdy to potřebuji): 17 ▼ 10 ▼ 2019 ▼

Důvod žádosti: [vypádník] Předpokládaná cena: [vypádník] Kč bez DPH ▼

Číslo zakázky: [vypádník] Číslo smlouvy z money: [vypádník] Způsob platby: Hotově ▼ Předpokládaná splatnost faktury: 17 ▼ 10 ▼ 2019 ▼

K žádance můžete přiložit bližší popis nebo produktový list.

+ Zobrazit tabulku V. Legenda - Schvalovací řízení

Přílohy

There are currently no attachments on this page.

Vybrat soubor | Soubor nevybrán | Enter a comment for this file here | Attach

Obrázek 9 Současný stav žádanek (Zdroj: vlastní)

Jsou zde pole: jméno žadatele, role, vedoucí oddělení, oddělení, vedoucí divize a divize. Tyhle políčka se týkají přímo uživatele. V nižší tabulce jsou již konkrétní informace k žádosti. Sem tedy patří pole: předmět žádosti, důvod žádosti, číslo zakázky, číslo z Money, způsob platby a informace týkající se platby, termín (do kdy to potřebuji), předpokládaná cena a předpokládaná splatnost faktury.

Ke každé žádosti je pak možné připojit přílohu. Například obrázek židle, co žádám, katalog, atp...

Věc, která zaměstnancům není příjemná, je potvrzování žádanky. Tu chceme tedy odstranit. Je zde tlačítko odeslat, po kliknutí se otevře nové dialogové okno a musíte žádost znovu potvrdit.

Další špatně udělanou věcí jsou povinná pole. V současném stavu nejsou žádná, a tak jde tedy žádost odeslat i prázdná. Pak se zde množí omylem vytvořené žádosti a administrátor má spoustu práce navíc, jelikož mazat je může pouze on, takže se mu hromadí práce.

Všechny tyto pole je potřeba přenést i do nového systému, jsou plně využívána.

Systém tedy funguje takto: Uživatel si vybere typ žádosti. V žádance vyplní pole, která jsou pro něj důležitá. Některá pole jsou již předvyplněná. Jedná se konkrétně o pole, která souvisí se zařazením uživatele do organizační struktury. Což je tedy divize, vedoucí divize, vedoucí oddělení, oddělení. Až má uživatel vše vyplněné dá „Odeslat žádost“. Otevře se mu nové dialogové okno, ve kterém je nutné žádost znovu schválit a poté je odeslána nadřízenému.

Nadřízenému se odešle e-mail o žádosti. K tomu, aby žádost mohl schválit, se musí přihlásit do systému a schválit či zamítnout žádost. Tenhle proces chceme v rámci nového systému také vylepšit. Budeme chtít systém navrhnout tak, aby bylo možné o žádosti rozhodnout již e-mailem.

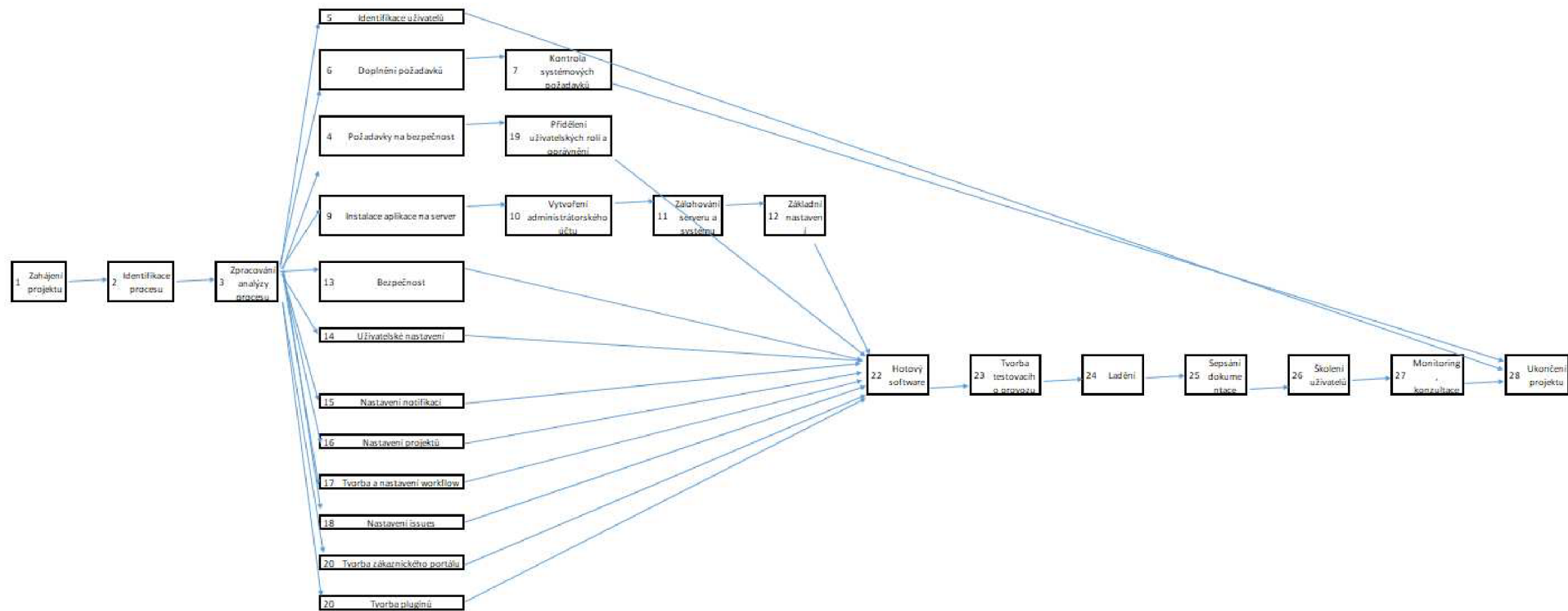
Jakmile nadřízený provede operaci, je podřízený informován o stavu své žádosti emailem. Pokud nastanou nějaké komplikace, je možné se podívat na stav žádosti v systému, popřípadě se domluvit na základě komentářů, které je možné k žádostem psát.

3.1.2 Časová analýza

Následující tabulka a síťový graf jsou zpracovány pomocí MS Excel.

Tabulka 5 Seznam činností (Zdroj: vlastní zpracování)

ID	Název úkolu	aij	bi j	mi j	Doba trvání	Rozpt yl	Směr. odchylka	Předchůdce
	Implementace IS				126,55			
1	Zahájení projektu				0,00			
	Příprava na implementaci				23,08			
2	Identifikace procesu	2	4	3	3,00	0,028	0,17	1
3	Zpracování analýzy procesu	3	7	5	5,00	0,111	0,33	2
4	Požadavky na bezpečnost	1	3	2	2,00	0,028	0,17	3
5	Identifikace uživatelů	0,5	3	2	1,92	0,028	0,17	3
6	Doplnění požadavků	1	3	2	2,00	0,028	0,17	2;3
7	Kontrola systémových požadavků	1	3	2	2,00	0,028	0,17	6
8	Testování softwaru	5	10	7	7,17	0,250	0,50	3
	Implementace				3,63	0,000	0,00	
9	Instalace aplikace na server	1	3	2	2,00	0,028	0,17	8
10	Vytvoření administrátorského účtu	0,3	1	0,5	0,55	0,007	0,08	9
11	Zálohování serveru a systému	0,5	2	1	1,08	0,028	0,17	10
	Nastavení				56,17	0,000	0,00	
12	Základní nastavení	5	10	5	5,83	0,694	0,83	11
13	Bezpečnost	1,5	4	3	2,92	0,028	0,17	3
14	Uživatelské nastavení	4	7	5	5,17	0,111	0,33	3
15	Nastavení notifikací	2	5	3	3,17	0,111	0,33	s
16	Nastavení projektů	2	6	4	4,00	0,111	0,33	3
17	Tvorba a nastavení workflow	5	10	7	7,17	0,250	0,50	3
18	Nastavení issues	3	5	4	4,00	0,028	0,17	3
19	Přidělení uživatelských rolí a oprávnění	2	5	3	3,17	0,111	0,33	4
20	Tvorba zákaznického portálu	5	14	10	9,83	0,444	0,67	3
21	Tvorba pluginů	5	15	10	10,00	0,694	0,83	3
22	Hotový software	0,5	1	1	0,92	0	0	12;13;14;15;16;17;18;19;20;21
	Testování				22,67	0,000	0,00	
23	Tvorba testovacího provozu	5	10	8	7,83	0,111	0,33	22
24	Ladění	5	10	7	7,17	0,250	0,50	23
25	Sepsání dokumentace	4	8	5	5,33	0,250	0,50	24
26	Školení uživatelů	1	5	2	2,33	0,250	0,50	25
	Ostrý provoz				21,00	0,000	0,00	
27	Monitoring, konzultace	10	30	20	20,00	2,778	1,67	26
28	Ukončení projektu				1,00			27



Obrázek 10 Grafické zpracování časové analýzy (Zdroj: vlastní zpracování)

3.1.3 Riziková analýza

Následující část se věnuje rizikům, která souvisí s implementací nového systému projektového řízení JIRA Service Desk. Rizika by mohla ohrozit hladký průběh projektu. Rizika byla identifikována metodou brainstormingu a doplněna nastudováním dostupné literatury.

Cílem změny je instalace a zavedení vybraného projektového nástroje JIRA Service Desk včetně jeho nastavení. Pro analýzu rizik je zvolena metoda RIPRAN.

Tabulka 6 Identifikace rizik

Č.	Hrozba	Scénář	Pravděp. výskytu	Dopad	Výsledná hodnota
1.	Nepřesná definice požadavků	Špatně navržený software	SP	VD	VHR
2.	Výběr nesprávného systému	Špatná funkcionality systému; velké pořizovací náklady	NP	VD	NHR
3.	Chybějící komponenty systému	Chybějící komponenty budou potřeba doplnit modulem v JAVĚ	SP	SD	VHR
4.	Ukončení podpory systému	Výběr nového systému	NP	VD	NHR
5.	Nedostatečné testování funkčnosti	Dodatečně bychom museli zvolit jiný systém	NP	SD	NHR
6.	Zásadní nedostatky systému po spuštění	Doplnění modulem či výběr nového systému	NP	SD	NHR
7.	Únik citlivých dat společnosti	Ztráta důvěry ve společnost	NP	VD	NHR
8.	Neochota zaměstnanců k využití systému	Úprava systému	NP	SD	NHR
9.	Špatné proškolení zaměstnanců	Nedostatečné znalosti o systému → špatné využití	SP	SD	VHR
10.	Organizační změny ve společnosti	Drobná úprava v systému	NP	MD	SHR

Jak lze vidět z tabulky vysokou hodnotu rizika – VHR byly ohodnoceny 1 - Nepřesná definice požadavků, 3 - Chybějící komponenty systému, 9 - Špatné proškolení zaměstnanců. Ostatní rizika jsou ohodnocena nízkou hodnotou rizika, krom bodu 10 – Organizační změny ve společnosti, který byl ohodnocen střední hodnotou rizika. Pro nás je tento bod, ale spíše považován za NHR. Ostatní body jsou ohodnoceny nízkou hodnotou rizika, jsou pro nás tedy téměř zanedbatelná, ale i tak je nesmíme opomíjet.

I když byla hodnota všech rizik snížena, je nutné rizika po celou dobu projektu dále sledovat a monitorovat. Může se stát, že se některé riziko změní, vyskytne se nová hrozba nebo také může zaniknout. V té nejhorší verzi dojde k naplnění hrozby a je nutné aplikovat připravené opatření.

3.2 Možnosti výběru IS

Na trhu existují v dnešní době desítky firem, které nabízejí tyto softwary.

Informační systém, který firma momentálně používá je od Confluence, Z tohoto důvodu bylo nutné volit program, se kterým nebude problém s implementací. Firma zvažovala variantu Microsoft Flow, ale pokud by chtěli realizovat tuhle variantu, bylo by nutné přejít na platformu MS celkově. Tzn., převést celý systém, což by bylo velmi náročné. Jak pro jejich programátory, tak určitě i finančně. Potom je zde možná implementace přes Confluence, nebo jednu z variant JIRA. Jelikož současný systém je od Confluence, chtěla firma zkusit něco nového a tím je právě jedna z verzí JIRA.

3.2.1 Základní požadavky – Popis požadavků zákazníka

Před samotným návrhem nového informačního systému budou v následující kapitole popsány požadavky, které zákazník definoval. Při návrhu systému budou tyto body brány v potaz, tak aby daný systém vše splňoval. V následujících odstavcích budou specifikovány požadavky spojené s uživatelskými rolemi a právy a dalšími nutnostmi.

Účel aplikace

Účelem tohoto systému je přehledně spravovat všechny žádanky v jedné aplikaci. Bude zde možné tedy žádanky vytvářet, schvalovat či je zamítat. Chceme zjednodušit proces

ukládání žádank pouze přes jedno tlačítko bez potvrzování. Zjednodušení by se mělo týkat i schvalování/zamítání žádank. Mým cílem by bylo umožnit schvalovací proces pomocí e-mailu, aby se nadřazený nemusel pokaždé přihlašovat do systému.

Jedná se o interní aplikaci a bude využívána k shromažďování, vytváření, editaci žádank ve firmě. Důraz bude kladen zejména na efektivnost, snadné ovládání a existenci nejnütnějších funkcí.

Vstup do informačního systému bude vždy umožněn až po přihlášení do systému. V systému tedy může pracovat pouze přihlášený uživatel. Požadavkem zákazníka je, aby běžný zaměstnanec mohl pouze vytvářet žádanky a měl přístup k jejich stavu. Správným přiřazením práv jednotlivým uživatelům zajistíme zvýšení bezpečnosti dat před neoprávněným zásahem uživatelů a také zajistíme, aby daný zaměstnanec viděl pouze své data a neměl přístup k cizím údajům. Zaměstnanci samozřejmě mohou být součástí více skupin, proto mohou mít přiřazených více uživatelských rolí.

Uživatelé systému budou nejen samotní programátoři, administrátoři systému a běžní uživatelé systému. Jelikož jsou všichni pracovníci české národnosti, program bude v češtině.

Uživatelské role:

- administrátor – má všechny práva,
- agent – může se přihlásit do systému a může provádět různé úpravy na projekt,
- běžný uživatel („zákazník“) – může se přihlásit a pouze vytvářet či editovat žádanky.

Projektový nástroj by měl kromě jednoduchého a intuitivního ovládání poskytovat i přehledy pro reporting žádank. Dále je žádoucí správa uživatelů, správa zabezpečení a přístupů.

Jak již bylo zmíněno v předchozí kapitole, vyžaduje se kompatibilita se stávajícím systémem firmy. Nástroj by měl být modulární a flexibilní s možností přidání nových funkcionalit.

Požadavky na aplikaci – obecná žádanka

Zachovat stávající "kolonky"/vytváření žádanky obecné

Pro vytvoření obecné žádanky je potřeba zadat základní parametry – jméno žadatele, role. Automaticky se doplní pole vedoucí oddělení, oddělení, vedoucí divize, divize. Další důležité informace pro nás jsou předmět žádosti, důvod žádosti, termín do kdy to potřebuji a předpokládaná cena. U některých zakázek může být důležité číslo zakázky, způsob platby, předpokládaná splatnost faktury.

Informace se stavem žádanky – ponechat

Jedná se o informace, které se uživateli zobrazí při kliku na jednotlivou žádanku. Ve sloupcích schvalovatel, datum počátku, rozhodnutí, datum konce a komentář. Schvalovatel je daná osoba, která ji schválila/neschválila. Datum počátku je datum vytvoření požadavku. Rozhodnutí může nabývat hodnot ano/ne (schválená/zamítnutá). Datum konce – do kdy je potřeba žádanku vyřídit. Komentář pro případ nějaké poznámky k případu.

Kolonka pro přílohy – ponechat

Je dobré zde mít část pro přílohy. Pokud by byl k dispozici katalog, či dokumenty, které by mohly být rozhodující, je dobré je uvést. Takže zde bude možné vybrat soubory z počítače a nahrát je k žadance.

Komentář – ponechat

Komentář pro případ, že je potřeba něco prokonzultovat. Komentovat mohou vždy všichni účastníci jednotlivé žádanky, což je tedy žadatel a ti, kteří ji musí schválit. Tenhle způsob komunikace je jednodušší a přehlednější než třeba domluva e-mailem.

Upravit ukládání/odeslání – zjednodušit

Na konci stránky by mělo být tlačítko pro odeslání žádanky. Vynechal by se tedy momentální krok potvrzení, který by práci trochu zjednodušil.

Vylepšit informace co chodí emailem, nejlépe vše podstatné o žadance + tlačítko na potvrzení

Po zmáčknutí tlačítka "uložit" dojde k odeslání e-mailu žadateli i jeho nadřízenému, který má žádanku schválit. Žadateli dojde e-mail s informacemi, že byla odeslána jeho žádost

ohledně daného předmětu. Nadřazenému dojde e-mail, který bude obsahovat všechny základní informace žádanky – jméno žadatele, role, předmět žádosti, důvod žádosti, termín, cena. Pokud je příloha, tak i ta. V e-mailu vedoucího jsou také dvě tlačítka, jedno ke schválení žádosti, druhé pak k jejímu zamítnutí.

Všeobecné požadavky

Musí být srozumitelné pro uživatele, kteří nemají technické znalosti. Designově tedy navrhnuté tak, aby to všechny navedlo a bylo jim jasné, jak postupovat.

Automatické doplňování některých políček

V každé žádosti jsou pole, která jsou neměnná, jedná se o divizi, oddělení divize, oddělení a vedoucího oddělení. Tyto pole budeme chtít doplňovat automaticky, aby se uživatel nemusel zdržovat vypisováním a byla menší šance, že nějaké pole bude vyplněné špatně.

„Soukromí“

Žádanky budou viditelné pouze pro žadatele a schvalovatele. Nikdo jiný kromě nich a administrátora je neuvidí.

3.2.2 Specifikace požadavků

Na základě předcházejících analýz a za pomoci spolupráce se zaměstnanci firmy K-net, byly definovány následující požadavky na projektový nástroj:

- Instalace aplikace na server ve vlastní správě: tento požadavek vychází z bezpečnostních politik společnosti, která zajišťuje bezpečnost dat.
- Webová aplikace: aby bylo možné se systémem jednoduše pracovat, musí být pro zaměstnance přístupná jako webová aplikace, která bude kompatibilní s internetovými prohlížeči Google Chrome, Internet Explorer, Mozilla Firefox a Safari.
- Správa uživatelů: do rozhraní projektového nástroje bude po přihlášení uživatel vyzván k zadání jména a hesla. Uživatelé budou mít předělena oprávnění příslušící jeho uživatelské roli. Aktuálně se počítá s přístupem pro max. 2-5 administrátorů a asi 50 klientů.
- Jednoduchá správa: nástroj nebude zatěžovat administrátory přílišnými požadavky a bude uživatelsky přívětivý.

3.2.3 Možnosti výběru systému

Microsoft Flow

Jedná se o cloudový software, který umožňuje zaměstnancům vytvářet a automatizovat pracovní postupy a úkoly. Automatizované pracovní toky se nazývají – workflow. Pro vytvoření uživatel musí specifikovat, jaká akce by se měla provést, když dojde k určité události. Jakmile se tok vytvoří, lze jej spravovat na ploše nebo prostřednictvím aplikace. Flow je integrovatelný s různými službami a aplikacemi společnosti Microsoft.

Microsoft Flow má mnoho způsobů využití, například: spouštění oznámení, kopírování souborů, sběr dat. (21)

Pokud by firma zvolila toto řešení, bylo by nutné přejít na platformu Microsoft Flow celkově, což by pro firmu bylo velice nákladné.

Jira

Jira od společnosti Atlassian je jedním z nejpoužívanějších nástrojů pro projektové řízení. Cílovou skupinou jsou spíše vývojáři a programátoři. Jira je založená na tzv. úkolech. Jejichž zpracování je velice detailní s cílem pomoci agilně orientovaným týmům vyvíjet a vydávat software spolu s vyřizováním klientských požadavků. Tento systém byl tedy už od počátku vyvíjen a přizpůsobován potřebám agilního způsobu vývoje v IT prostředí. Výhodou jsou měsíční zkušební lhůty. Webová aplikace je velmi intuitivní. Pomocí průvodce v několika krocích provede uživatele základním nastavením projektu. V technické oblasti pak podporuje zařízení fungující na operačním systému Windows, Linux, Android, iPhone a samozřejmostí je webová aplikace. Významné firmy, které software používají, jsou Twitter, Spotify, BlackRock, Splunk a mnoho dalších. Platí se zde pouze za aktivní uživatele. K dispozici je také mobilní aplikace. (22)

Jsou tři pro nás zajímavé verze softwaru Jira, a to: Core, Service Desk a Software. Každá z nich má jiné možnosti.

Jira Core

Slouží k řízení jednoduchých projektů. Je to základní verze Jiry, která je určena pro netechnické uživatele. Má tedy zjednodušené pohledy. Díky workflow umožňuje sledování projektu a uživatelům chodí oznámení ohledně stavu jejich žádosti. Umožňuje delegovat odpovědnost. (23)

Jira Software

Primárně je určena pro vývojářské týmy. Obsahuje nástroje a funkce pro podporu každé fáze vývojového cyklu. Umožňuje vytvářet workflow. (23)

Jira Service Desk

Primárně určen pro IT a servisní týmy. Má podporu standartních procesů. Lze snadno přizpůsobit firemním procesům. Vhodný pro komunikaci se zákazníky – zákaznický portál. Je zde spousta šablon, má podporu e-mailu. Jelikož je tato verze určena pro komunikaci se zákazníky, můžeme jich zde mít neomezený počet. (23)

Cenové porovnání systémů

Všechny ceny jsou uvedeny za měsíc a za jednotlivé uživatele.

Tabulka 5 Cenové porovnání produktů JIRA (Zdroj: 24)

Počet uživatelů/Software	Jira Core	Jira Software	Jira Service Desk	Microsoft Flow
1-10	10 USD	10 USD	10 USD	15 USD
11-100	7 USD	7 USD	7 USD	15 USD
101-250	6 USD	6 USD	6 USD	15 USD

Cenově vidíme, že Microsoft Flow je z nabídky nejdražší a nabídky produktů Jira, jsou všechny ohodnoceny stejně. Takže nejvíce nás budou zajímat funkce jednotlivých produktů Jira.

Tabulka 6 Porovnání produktů JIRA

Název/Informace	Jira Software	Jira Service Desk	Jira Core
Základní popis...	<ul style="list-style-type: none"> plánování sledování vydávání software 	<ul style="list-style-type: none"> jednoduchá cesta komunikace pro zákazníky rychlejší způsob řízení pro agenty 	<ul style="list-style-type: none"> spravování libovolného projektu (marketing, schvalování a kontrola dokumentů)
Uživatelé jsou...	<ul style="list-style-type: none"> vývojáři projektoví manažeři 	<ul style="list-style-type: none"> zákazníci agenti IT manažeři manažeři uživatelé 	<ul style="list-style-type: none"> obchodní uživatelé
Možnosti použití...	<ul style="list-style-type: none"> sledování chyb zákaznická podpora správa procesů správa úkolů vývoj softwaru 	<ul style="list-style-type: none"> IT podpora zákaznická podpora řízení služeb správa žádostí správa incidentů správa změn 	<ul style="list-style-type: none"> projekty netechnických týmů schválení workflow správa úkolů

Po prozkoumání funkcí je pro nás velmi zajímavá Jira Service Desk, která nabízí tyto typy projektů. Pro nás je dobře využitelný Customer service. Je zde totiž možnost využít tento typ programu tak, že na naše zaměstnance bude pohlíženo jako na zákazníky. Ti se budou moct přihlásit a podávat žádosti. Firma tedy bude platit jen za agenty, kteří budou projekty spravovat a tím ušetří velké peníze oproti jiným systémům.

Typy projektů:

Základní SD – používá se pro interní obchodní týmy, jako jsou HR, finance nebo malé IT týmy. Šablona je dodávána s několika doporučenými formuláři žádostí a je snadno přizpůsobitelná a rozšiřitelná podle našich potřeb.

IT Service Desk – pro IT týmy, které udržují složitější infrastrukturu. Šablona je dodávána s pracovními postupy pro řízení změn, incidentů a problémů.

Customer service – pro komunikaci s externími zákazníky je vhodný tento typ projektu. Můžete sbírat hlášení o chybách, nebo přijímat návrhy na novou funkci. Šablona je připravena pro Vaše zákazníky, aby mohli požádat o technickou nebo fakturační pomoc a ohlásit problémy.

Princip SD Customer service je jednoduchý a to:

- 1) Zákazník odešle žádost agentům nebo prostřednictvím portálu nebo e-mailu.
- 2) Agent oddělení vidí požadavek ve své frontě a zkoumá problém.
- 3) Váš zákazník a další účastníci používají portál nebo e-mail k projednání žádosti s agentem.
- 4) Váš agent vyplní požadavek a zákazník je spokojen. (25)

3.3 Implementace

Tato část se věnuje již samotné implementaci nástroje pro projektové řízení, který byl zvolen.

3.3.1 Instalace

Vzhledem k požadavkům na bezpečnost je zvolena instalace na vlastní server společnosti, a ne cloudové řešení. Instalace softwaru není nijak složitá. Na webových stránkách najdete tutoriál pro instalaci. Zde je aspoň stručně popsán proces.

1. Stažení instalačního souboru pro operační systém Windows 64-bit.
2. Spuštění instalačního souboru jako administrátor.
 - a. Volba složky pro instalaci JIRA Software.
 - b. Volba domovské složky pro ukládání záznamů a souborů ze systému.
 - c. Volba TCP portů.

3. Nastavení webové aplikace

- a. Napojení aplikace na předem připravenou databázi.
- b. Nastavení názvu: „PwC“.
- c. Nastavení URL adresy k JIRA instanci.
- d. Nastavení módu na privátní.
- e. Vložení zakoupené licencí.
- f. Nastavení administrátorského účtu.

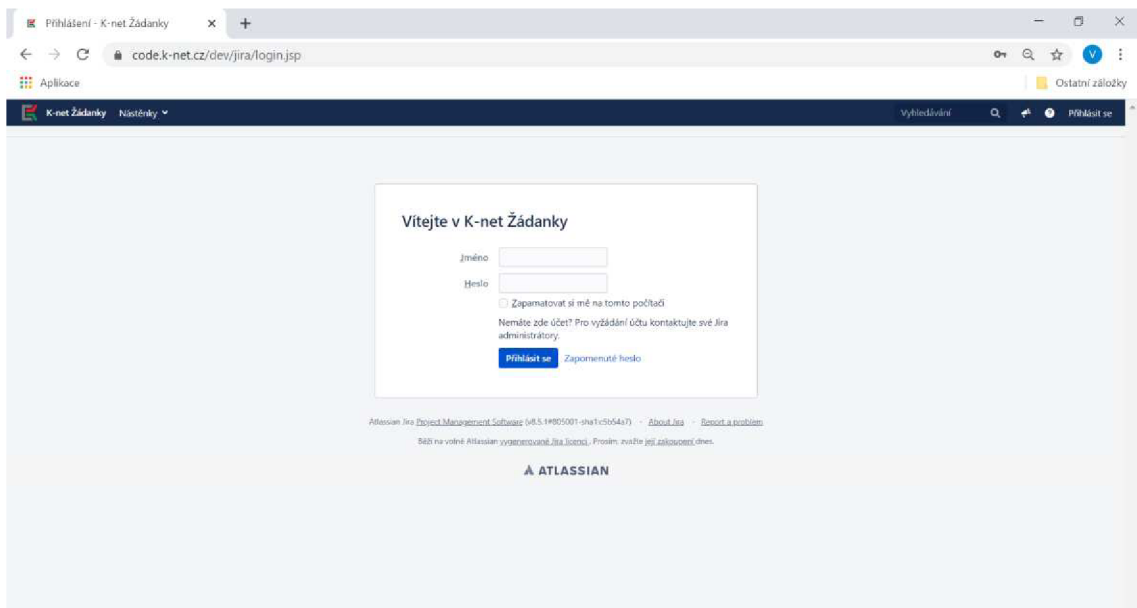
4. Záloha serveru a aplikace. (26)

Tímto je instalace prostředí hotová.

3.3.2 Změna nastavení

K tomu, abychom Jiru mohli využívat, je potřeba upravit její současné nastavení na námi požadované. Měli bychom tedy nastavit pozadí, odesílání e-mailů, notifikace, oprávnění, workflow a také zákaznický portál.

Nastavení v Jira je celkově přehledné. Vše je krásně seřazeno. Prvním krokem k tomu, aby náš systém působil méně genericky, je upravit logo, název webu, přidat favicon a upravit barvy. Všechny tyto změny lze najít na jedné straně v nastavení. Vše si pak pouze naklikáme a máme systém, který vypadá pro naši firmu přívětivěji. Na této stránce lze upravit i informace jako je zobrazení formátu času, data. V příloze najdete podrobnější popis nastavení – viz příloha 1.



Obrázek 11 Přihlašovací stránka (Zdroj: vlastní)

Emailové notifikace

K tomu, aby nám chodili e-mailové notifikace, je nutné nastavit předem určité informace: jméno, server, port a další fakta. Po zadání všech údajů si můžeme i vyzkoušet, zda je to správně a poslat si testovací e-mail.

Dá se také nastavit formát emailu. Buď mohou chodit formou textu, nebo HTML, my volíme HTML kvůli přehlednosti. Podrobné nastavení viz příloha 1.

Nastavení issue

Jira nabízí velké množství druhů úkolů. Pro náš projekt použijeme „task“. Je to základní typ. „Task“ je vlastně úkol. V našem projektu je to tedy jedna žádost.

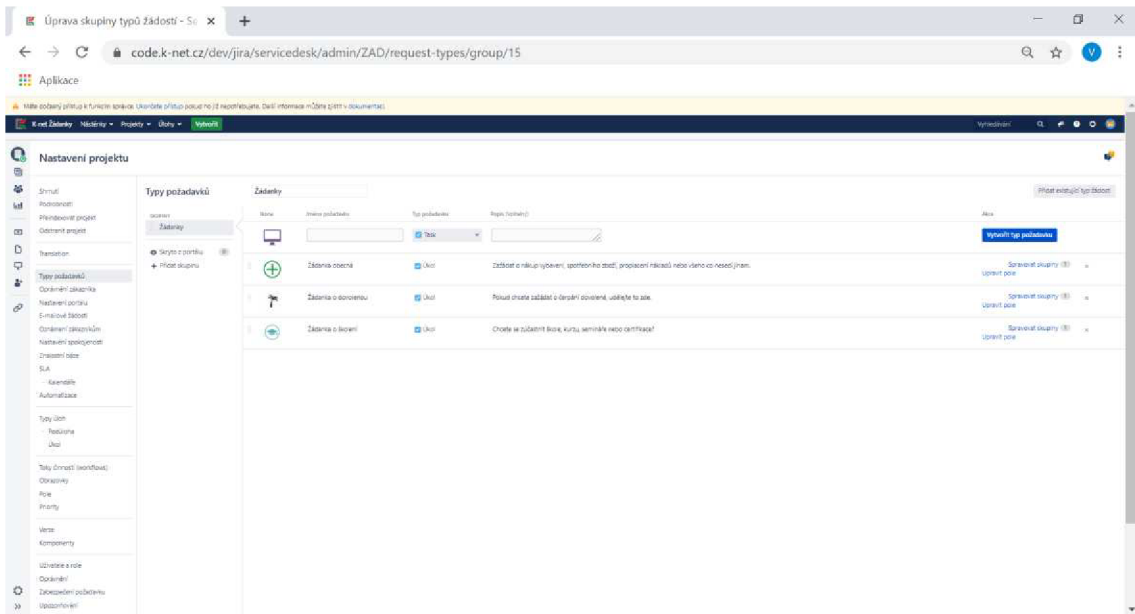
3.3.3 Nastavení zákaznického portálu

Pokud máme nainstalovanou Jiru, jedním z prvních kroků je zvolit typ projektu, který budeme vytvářet. Jira Service Desk nabízí tyto druhy: základní service desk a IT service desk. Pro náš projekt volíme customer service, ten totiž umožňuje zákaznický portál.

V základním nastavení můžeme měnit jméno projektu, jeho klíč, typ projektu a jeho popis. Důležitý je název a typ projektu. Od toho se odvíjí spousta dalších věcí.

V podrobnějším nastavení pak můžeme měnit ostatní informace jako je typ úkolu, workflow, pole, role uživatelů a další.

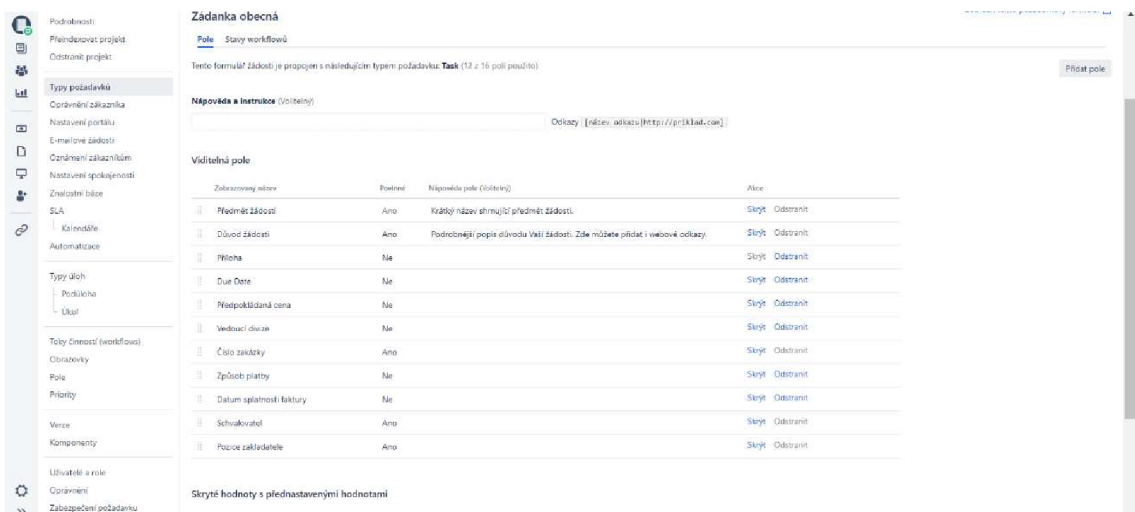
Jelikož používáme produkt Jira Service Desk, který nám umožňuje použít zákaznický portál, musíme i ten nastavit. Na obrázku vidíme typy požadavků, které máme vytvořené. Je možné jich mít i mnohem více. Lze zvolit typ úkolu a můžeme je zde i spravovat.



Obrázek 12 Nastavení projektu – zákaznický portál (Zdroj: vlastní)

Úprava polí

Momentálně se nacházíme ve správě požadavku žádanky obecné. Zde je možné přidat / odebrat pole a zvolit jejich pořadí.



Obrázek 13 Správa žádanky obecné (Zdroj: vlastní)

3.3.4 Uživatelé

Jira je zaměřená na dvě licencované role, a to administrátory a správce. Administrátoři nastavují a konfiguruji projekty. Agenti pracují na daných projektech. U Jiry Service Desku je navíc zákazník, který má přístup pouze do zákaznického portálu a vidí „zjednodušenou verzi“.

Administrátor:

- má přístup ke všem vlastnostem projektu,
- spravuje uživatele a uživatelské role,
- nastavuje zákaznické portály, typy požadavků, fronty a reporty,
- může provést stejné úkoly jako agent.

Agent (Správce):

- může si prohlédnout zákaznický portál, fronty, reporty a metriky pro projekt,
- přidávat, upravovat a odstraňovat interní komentáře k úkolům,
- přidávat zákazníky do projektu,
- přečíst si obsah znalostní báze.

Zákazník:

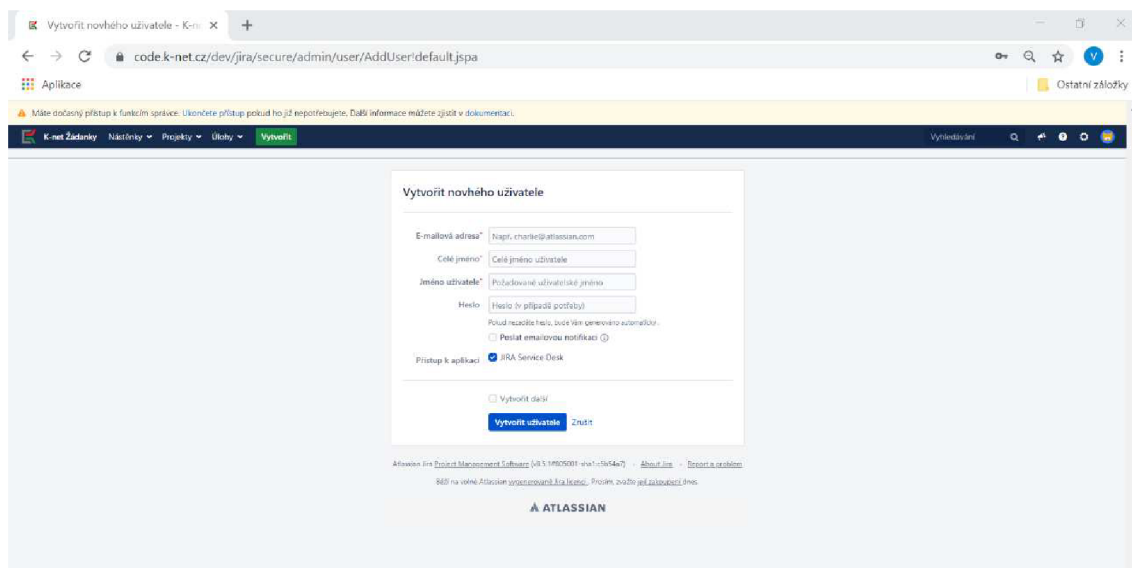
- může vznést požadavek pomocí zákaznického portálu nebo e-mailového kanálu,
- sledovat jeho požadavky na zákaznickém portálu,
- komentovat jeho požadavky,
- číst znalostní báze,
- schvalovat žádosti ostatních zákazníků,
- sdílet žádosti s ostatními (pokud to umožňují práva),
- nepotřebují licenci, jsou zadarmo. (27)

Náš projekt chceme realizovat právě formou zákaznického portálu. Tato varianta je pro firmu finančně velice výhodná a nabízí spoustu možností konfigurace. Bude zde tedy jen pár zaměstnanců, kteří budou mít zakoupenou licenci. Ostatní zaměstnanci budou

fungovat na principu zákazníků. Budou mít omezený přístup pouze k vytváření žádanek. Vše zde pro ně bude jednoduché a práce pro ně tak bude příjemnější.

Vytvoření nového uživatele

Nového uživatele vytváříme v nastavení přímo v sekci uživatelé a skupiny. Pro vytvoření nového uživatele potřebujeme e-mailovou adresu, jméno a heslo. Všechny uživatele je tedy nutné nahrát do systému. Pokud chceme, aby měl uživatel přístup do zákaznického portálu, musíme zaškrtnout právě přístup k aplikaci pomocí JIRA Service Desku.



Obrázek 14 Vytvoření nového uživatele (Zdroj: vlastní)

3.3.5 Uživatelská práva

Jira má velké možnosti co se týče nastavení práv pro jednotlivé úkony a uživatele. Je možné si zde vybrat z široké škály možností podle skupin, přístupu a počtu přihlášení.

Pro náš projekt bylo nutné nastavit pro každého zaměstnance zobrazení pouze svých žádostí, a ne ty ostatních uživatelů. Aby je mohl jak žadatel, tak schvalovatel komentovat a ponechat možnost vkládání příloh pro uživatele. Navíc skupiny budeme potřebovat pro rozdělení zaměstnanců do jednotlivých divizí, aby se žádanky samy přidělily jejich vedoucímu. K tomu, aby se vedoucí sám přiřadil, však budeme potřebovat plugin, protože toto Jira neumí.

Se skupinou se dá pak pracovat v rámci oprávnění. Oprávnění můžeme přidat k jakékoliv činnosti, například: přiřaditelný uživatel, uzavírat úlohy, odstranit úlohy, upravit úlohy,

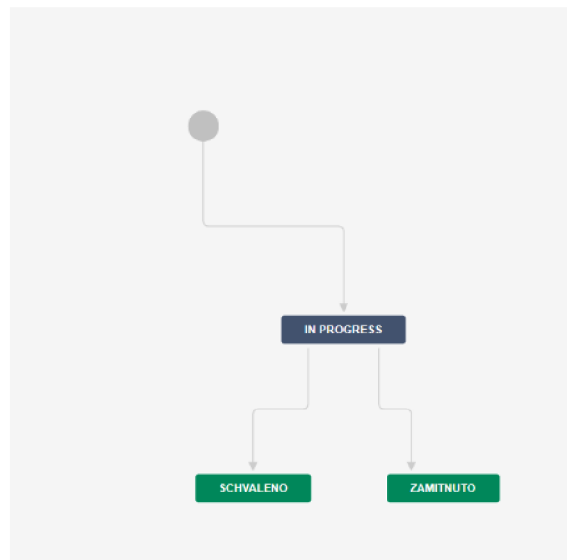
měnit zadavatele, vyřešit úlohy, přidávat komentář, vytvářet přílohy a mnoho dalších funkcí. Podrobnější nastavení uživatelů najdete v příloze 1.

3.3.6 Nastavení notifikací

Jedna z věcí, kterou jsme právě chtěli zlepšit oproti stávajícímu řešení, byly e-mailové notifikace. Do e-mailu Vám přijde přehledná zpráva, ve které může schvalovatel žádost hned schválit. Můžeme si vybrat, zda chceme „html“ či „txt“ zobrazení. Notifikace se upravují úplně stejně jako oprávnění uživatelů. Jsou zde vypsány různé úkoly a k těm můžeme přiřadit, komu má upozornění přijít. Podrobné nastavení viz příloha 1.

3.3.7 Tvorba a nastavení workflow

Workflow můžeme použít již přednastavené nebo si vytvořit vlastní. Jira umožňuje jak grafické, tak textové úpravy. Je možné doplnit další kroky či akce, co se mají stát po procesu. V našem projektu je potřeba nastavit workflow tak, aby měla dva konečné stavy – schváleno a zamítnuto.



Obrázek 15 Ukázka workflow (Zdroj: vlastní)

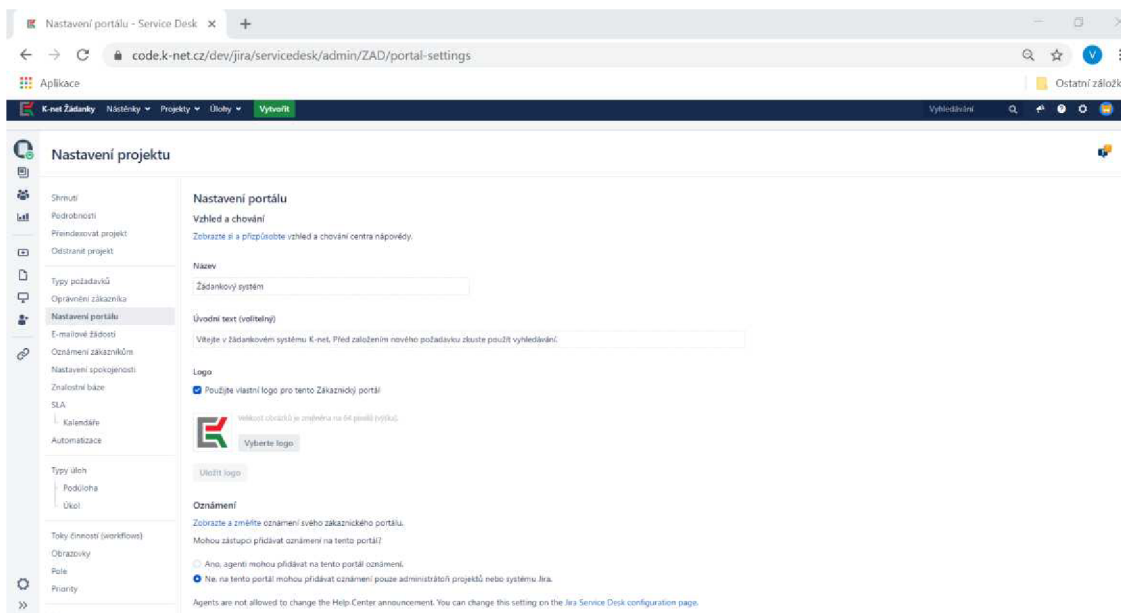
3.3.8 Zákaznický portál

Možnost zákaznického portálu pro nás při výběru byla rozhodující. Díky této možnosti firma ušetří velké finanční prostředky a navrhne jednoduchý uživatelský systém pro netechnické uživatele.

Jak již bylo řečeno, jen pár zaměstnanců – administrátoři budou mít přístup do nastavení a ostatní zaměstnanci, kteří potřebují pouze vytvořit žádanky, budou mít přístup jen do této jednodušší verze. Ta je velice intuitivní, přehledná a jednoduše se s ní pracuje. Navíc těmto zaměstnancům nemusíme kupovat licence a tím se ušetří.

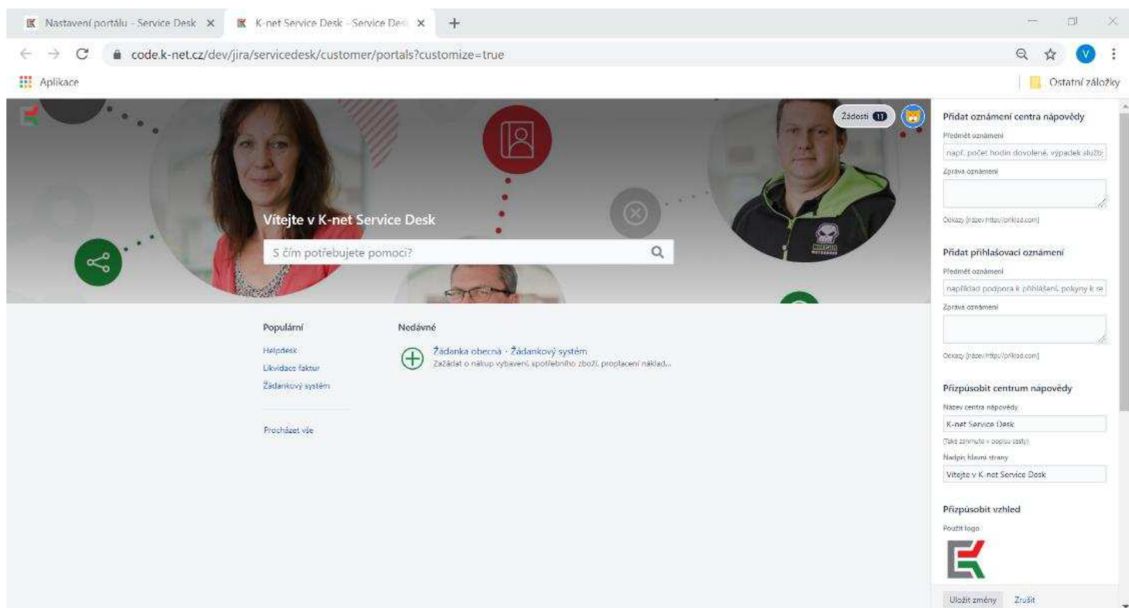
Zákaznický portál můžeme celkem jednoduše upravit díky možnostem v nastavení. Je možné editovat grafickou stránku, takže stranu můžeme přizpůsobit našim požadavkům a udělat ji více uživatelskou pro naši organizaci.

Možnosti editace: logo, ikonka, úvodní text a podobně.



Obrázek 16 Nastavení portálu (Zdroj: vlastní)

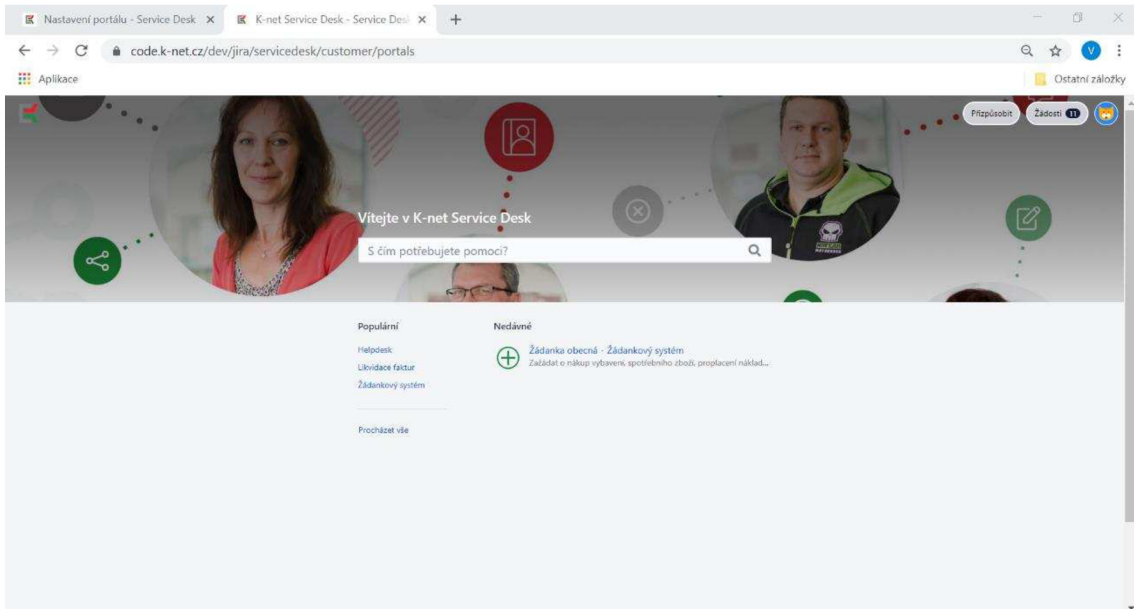
Portál můžeme buď upravit pomocí tohoto nastavení, nebo již konkrétně v systému. A tak tedy co nejvíce přiblížit finální podobě, aby byl web odlišný a konkrétní pro jednu firmu.



Obrázek 17 Ukázka zákaznického portálu – nastavení (Zdroj: vlastní)

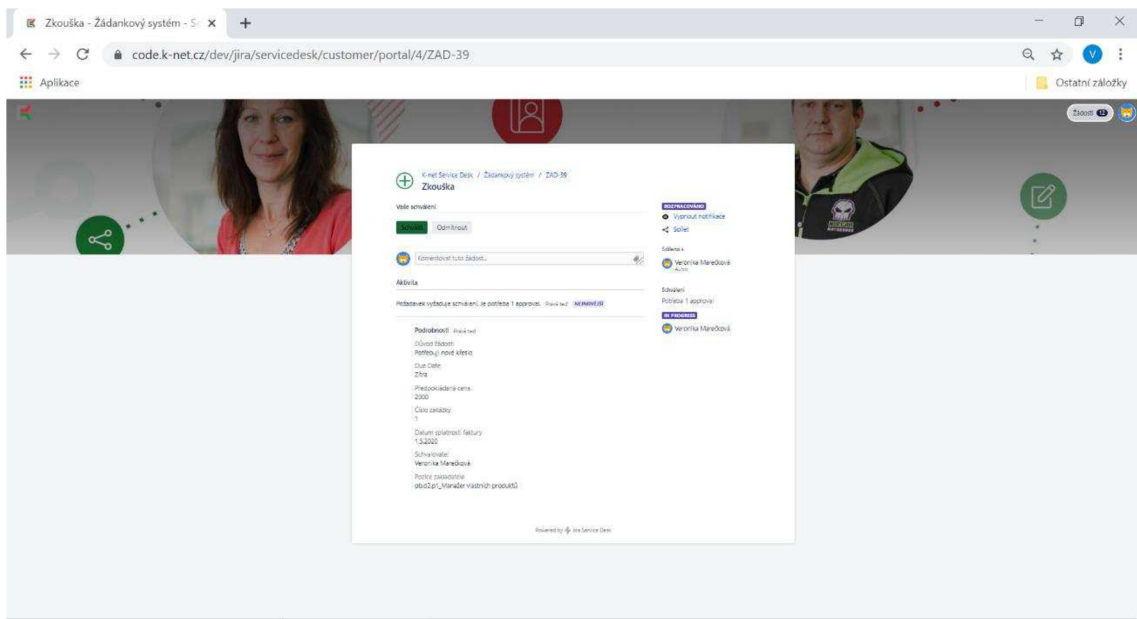
Zákaznický portál z pohledu uživatele

Na uživatele již takto upravený systém působí přívětivě a známě. Vše je zde pro něj přehledné. Jednoduše může přidat žádost nebo si prohlédnout již vytvořené. V pravém rohu nahoře se také nachází odkaz na profil uživatele.



Obrázek 18 Zákaznická portál – uživatelské nastavení (Zdroj: vlastní)

Poslední, co si můžeme rozkliknout je profil uživatele. Zde vidíme nejdůležitější informace o uživateli.

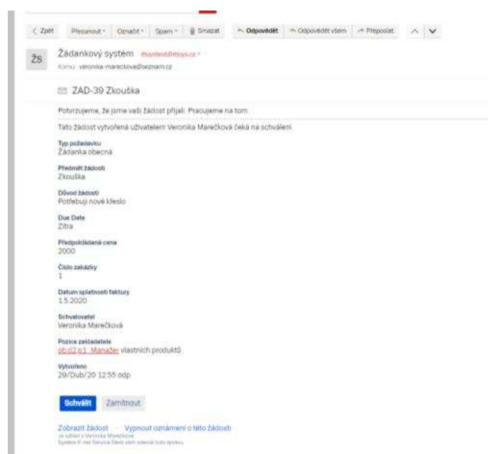


Obrázek 21 Zobrazení vytvořené žádosti (Zdroj: vlastní)

Po vytvoření žádosti

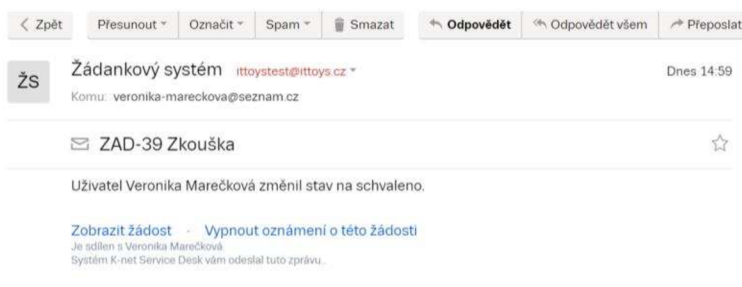
Velkou výhodou oproti nynějšímu systému je rozhodování ohledně žádosti. Schvalovatel ji může potvrdit či zamítnout již v emailu a nemusí se tedy přihlašovat do systému. Všechny potřebné informace mu přijdou v e-mailu. Pokud jsou zde nějaké nejasnosti, tak se musí přihlásit a v komentářích k problému můžou tyto detaily vyřešit. E-mail poskytne vedoucímu všechny potřebné informace k rozhodnutí a je přehledný.

E-mail je automaticky odeslán po vytvoření žádosti. Tuto funkci Jira nabízí automaticky. Je potřeba ji však nastavit a dodat informace ohledně e-mailových notifikací, jak již bylo uvedeno. V každém prohlížeči je však zobrazení trochu jiné. Tento e-mail přijde schvalovateli.



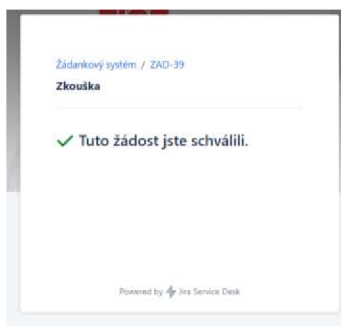
Obrázek 22 Zobrazení e-mailu – vytvoření (Zdroj: vlastní)

Po tom, co o žádosti rozhodneme, buď v e-mailu popřípadě přímo v portálu, na který se dostaneme odkazem. Odkaz je i v e-mailu, schovává se pod slovy – zobrazení žádosti. Přejde nám další e-mail, ve kterém nám dojde oznámení o rozhodnutí. E-mail o rozhodnutí přijde žadateli.



Obrázek 23 Zobrazení e-mailu (Zdroj: vlastní)

S žádostmi můžeme také pracovat přímo v portálu. Jejich vytvoření a zobrazení jsme si již ukázali. Pokud žádost schválíme nebo ji zamítneme, zobrazí se nám nové dialogové okno, které informuje o provedené akci.



Obrázek 24 Nové okno – rozhodnutí o žádosti (Zdroj: vlastní)

3.3.9 Plugin

Možnost využít placených pluginů jsme hned na začátku zamítli. Za poslední roky ceny pluginů rapidně vzrostly a není možné je kupovat za tuto cenu pro celou firmu – jedná se o finanční politiku.

Příjemnější varianta, však o trochu složitější pro nás je napsání kódu v programovacím jazyku JAVA. Nejprve bylo potřeba zjistit, zda je možné tuto variantu vůbec realizovat. Díky vkládání vlastních pluginů jsme zjistili, že to možné je.

Za další jsme museli zjistit, zda je vůbec možné zasáhnout do jednotlivých částí workflow. Touto částí se zabíral IT programátor firmy a po několika dnech zkoumání zjistil, že i tady máme štěstí a lze využít jednotlivé procesy a stavy v náš prospěch.

V důsledku těchto informací jsme museli zjistit co Jira neumí a co tedy budeme muset doprogramovat. Nám konkrétně se jedná o následující dva moduly Knet OrgBoard a Knet Zadanky. Jednotlivé moduly jsou lépe popsány v následující části.

Kód je napsaný anglicky, systémové hlášky jsou také anglicky. Okomentovaný je již česky. Komentáře jsou u každé třídy a až na pár výjimek i u všech metod.

Modul Knet OrgBoard

- Udržuje firemní organizační strukturu, kterou se synchronizuje s Active Directory (read-only)
- Hlavním smyslem modulu je vytvoření hierarchických vazeb mezi uživateli a skupinami, které jsou jinak v Confluence pouze v ploché rovině – tedy bez vazeb hierarchických vazeb
- Tyto vazby jsou důležité pro všechny žádankové systémy
- Pracuje s Confluence uživateli (users)
- Pracuje s Confluence skupinami (groups)
- Dokáže poskytnout nadřazené uživatele a skupiny pro určitého uživatele a jeho pracovní pozici

Například vstup **uživatel**: Petr Kalas; **pracovní pozice (role)**: Programátor, výstupem pro nás budou následující data **oddělení**: vývoj; **vedoucí oddělení**: Petr Nepustil; **divize**: D4; **vedoucí divize**: Tomáš Knettig

Kód pro tento plugin viz Příloha 2.

Modul Knet Zadanky

- Doplnuje funkcionalitu žádankového systému vytvořeného v JIRA Service Desk
- Automatické přiřazování nadřízených uživatelů a skupin pomocí modulu Knet OrgBoard
- Automatické přiřazování schvalovatelů (approvers)
- Případné další automatické přiřazování hodnot custom polí (custom-fields) v JIRA, které se nám podaří nastavit pouze konfigurací systému
- "Všechno co nepůjde nijak naklikat, tak řešit pomocí pluginu"

Kód pro tento plugin viz Příloha 3.

3.4 Ekonomické zhodnocení projektů

V této části je provedeno vyčíslení nákladů projektu popsaného v předchozích kapitolách. Nejvýznamnější položku tvoří licence na zvolený nástroj pro podporu projektového řízení Jira Service Desk.

3.4.1 Výdaje

Náklady, které jsou vyčísleny v tabulce níže, vycházejí z ocenění náročnosti jednotlivých úkolů, které bylo provedeno při odhadu časové náročnosti projektu. Tyto hodinové údaje jsou ohodnoceny průměrnými hodinovými sazbami pro dané zodpovědné role ve společnosti v Kč. Implementace a analýzy tedy budou prováděny z vlastních zdrojů, přičemž vedoucím celé realizace by měl být administrátor.

Náklady je možné rozdělit do dvou částí – do té první patří jednorázové náklady, které se skládají z pořizovací ceny licence na Jira Service Desk, implementaci nástroje a s tím spojené náklady na zavedení.

Tabulka 7 Pořizovací náklady

Pořizovací výdaje	Cena Kč	Jednotka	Počet jednotek	Celkem Kč
Příprava na implementaci	600 Kč	hod.	23,08	5 770,00 Kč
Moje práce	200 Kč	hod.	17,31	3 462,00 Kč
IT pracovník firmy	400 Kč	hod.	5,77	2 308,00 Kč
Implementace a nastavení	600 Kč	hod.	59,7975	35 878,50 Kč
Moje práce	200 Kč	hod.	44,1275	8 825,50 Kč

IT pracovník firmy	400 Kč	hod.	15,67	6 268,00 Kč
Testování	600 Kč	hod.	22,6725	13 603,50 Kč
Moje práce	200 Kč	hod.	17,0025	3 400,50 Kč
IT pracovník firmy	400 Kč	hod.	5,67	2 268,00 Kč
JIRA Service Desk Server licence	2 961,6 Kč	licence	5 ks	14 808,00 Kč
Cena celkem				70 060,00 Kč

Na většině výdajů se podílel pracovník firmy, který pracuje na pozici programátor, se kterým jsem celou dobu spolupracovala. Výdaji jsou tedy jeho čas strávený na práci vynásobený mzdou 400 Kč/hod. V cenách výše uvedených je započítaná i moje práce na projektu, která je ohodnocena částkou 200 Kč/hod.

V celkových výdajích jsou započítány přibližné ceny licencí. Jedná se o investice pro licence agentů, ostatní mají přístup jako „customer“. Ceny licencí jsou na rok.

Používáním této aplikace budou také vznikat každý rok provozní náklady na údržbu a správu aplikace a update, což bude zajišťovat administrátor Jira Service Desk.

Tabulka 8 Provozní náklady

Opakující se výdaje	Cena Kč	Počet ks	Celkem Kč
Licence pro 5 uživatelů	2 961,6 Kč	5 ks	14 808 Kč
Cena celkem			14 808 Kč

3.4.2 Přínosy

Pro organizaci jsou nejdůležitější hlavně faktory, které souvisejí se správným nastavením procesu, motivace a efektivita práce zaměstnanců a v neposlední řadě také finanční hledisko. Tyto jednotlivé faktory je potřeba brát v úvahu, když navrhujeme software, aby zaměstnancům v práci pomáhal. Nový systém povede k:

- automatizaci podnikového procesu,
- vylepšení současného systému,
- zjednodušení práce zaměstnanců a tím zvýšení produktivity (z 5 min na 3 min),
- administrátor má více času na jiné záležitosti,
- snížení času odpovědi u žádanky (8 min na 5).

Podle informací poskytnutých firmou se za letošní rok vytvořilo:

- leden – 64 žádank obecných,
- únor – 55 žádank obecných,
- březen – 54 žádank obecných.

Průměrně je to 57 žádank. Pokud by byl již implementovaný náš systém, ušetřilo by to firmě celkově 285 minut. A to jsou pouze žádanky obecné. Pokud by se tedy řešení aplikovalo i na další žádanky, firma tím ušetří víc času.

ZÁVĚR

Cílem této bakalářské práce bylo provést analýzu současného procesu žádanek ve firmě K-net Technical International Group, s.r.o. a vybrat vhodný nástroj pro jeho nové navržení.

V teoretické části práce byly shrnuty dosavadní poznatky z oblastí informačních systémů a metodiky analýzy řízení rizik a časové analýzy. Samotný návrh tedy obsahuje výběr projektového nástroje. Na základě definovaných požadavků bylo vybráno řešení Jira Service Desk. Pro jistotu byla provedena analýza rizik a časová analýza, která byla vytvořena s pomocí metody PERT a síťového grafu. Díky těmto metodám jsme mohli určit kritickou cestu a zároveň bylo stanoveno, že doba trvání projektu vychází na téměř 127 dní.

Samotný návrh tedy obsahuje nejdříve návrh projektového nástroje. Na základě definovaných požadavků bylo vybráno řešení JIRA Service Desk. Pro tohle řešení se také firma rozhodla z finančních důvodů.

Závěrem musím říci, že celá práce byla přínosem jak pro mě, tak pro firmu. Navržené řešení je již aplikované a funguje tak jak má.

SEZNAM POUŽITÝCH ZDROJŮ

- (1) GÁLA, Libor, Jan POUR a Zuzana ŠEDIVÁ. *Podniková informatika: počítačové aplikace v podnikové a mezipodnikové praxi*. 3., aktualizované vydání. Praha: Grada Publishing, 2015. Management v informační společnosti. ISBN 978-80-247-5457-4
- (2) KOCH, Miloš. *Informační systémy a technologie*. Vyd. 2. Brno: Zdeněk Novotný, 2002. Učební texty vysokých škol. ISBN 80-214-2193-2.
- (3) BÉBR, Richard a Petr DOUCEK. *Informační systémy pro podporu manažerské práce*. Praha: Professional Publishing, 2005. ISBN 80-86419-79-7.
- (4) SODOMKA, Petr a Hana KLČOVÁ. *Informační systémy v podnikové praxi*. 2., aktualiz. a rozš. vyd. Brno: Computer Press, 2010. ISBN 978-80-251-2878-7.
- (5) Workflow (online). 2005, , (95)-103 (cit. 2020-02-04). ISBN 978-3-598-44011-3.
- (6) BASL, Josef a Roman BLAŽÍČEK. *Podnikové informační systémy: podnik v informační společnosti*. 3., aktualiz. a dopl. vyd. Praha: Grada, 2012, 323 s. : il., portréty ; 25 cm. ISBN 978-80-247-4307-3.
- (7) Atlassian. Atlassian | Software Development and Collaboration Tools (online). Copyright © 2020 Atlassian (cit. 04. 02. 2020). Dostupné z: <https://www.atlassian.com/software/jira/guides/getting-started/overview#key-terms-to-know>
- (8) SMOLÍKOVÁ, Lenka. *Projektové řízení: studijní text pro prezenční a kombinovanou formu studia*. Brno: Akademické nakladatelství CERM, s.r.o, 2018, 88 stran : ilustrace. ISBN 978-80-214-5695-2.
- (9) RAIS, Karel a Radek DOSKOČIL. *Risk management: studijní text pro kombinovanou formu studia*. Brno: Akademické nakladatelství CERM, 2007, 152 s. : il. ISBN 978-80-214-3510-0.
- (10) DOLEŽAL, Jan, MÁCHAL, Pavel a LACKO, Branislav. *Projektový management podle IPMA*. 2. Praha : Grada, 2012. ISBN 978-80-247-4275-5.
- (11) SCHILDT, Herbert. *Java 7: výukový kurz*. Brno: Computer Press, 2012, 664 s. ISBN 978-80-251-3748-2.

- (12) VIRIUS, Miroslav. *Java pro zelenáče*. Praha: Neocortex, c2001. ISBN isbn80-902230-9-5.
- (13) Introduction to Java – JetBrains Academy. (online). Dostupné z: <https://hyperskill.org/learn/step/3500>
- (14) RUKOVANSKÝ, Imrich a Petr MATUŠÍK. *Základy podnikových informačních systémů*. Kunovice: Evropský polytechnický institut, 2006. ISBN isbn80-7314-104-3.
- (15) LENERT, Jiří, Vlastimil MATULA a Lucie MATUŠKOVÁ. *Informační management*. Ostrava: Vysoká škola podnikání, 2005. ISBN isbn80-86764-35-4.
- (16) KEŘKOVSKÝ, Miloslav a Oldřich VYKYPĚL. *Strategické řízení: teorie pro praxi*. 2. vyd. Praha: C.H. Beck, 2006. C.H. Beck pro praxi. ISBN isbn80-7179-453-8.
- (17) Vytvoření SWOT analýzy. corphalos | Petr Háze – byznys konzultant | Poradenství v oblasti rozvoje a řízení firem, procesů, interim management, implementace web CMS a CRM řešení. [online]. Copyright © 2020 Petr Háze, corphalos [cit. 19. 04. 2020]. Dostupné z: <https://www.corphalos.eu/vytvoreni-swot-analyzy/>
- (18) 5 recipes to make your work flow in Jira - codecentric AG Blog. codecentric AG Blog - Expertenwissen rund um agile Softwareentwicklung, Java und Performance Solutions. [online]. Dostupné z: <https://blog.codecentric.de/en/2017/05/5-recipes-make-work-flow-jira/>
- (19) K-net Technical International Group, s.r.o. , Brno IČO 47916745 - Obchodní rejstřík firem | Kurzy.cz. Obchodní rejstřík firem - vazby a vztahy z justice.cz | Kurzy.cz [online]. Copyright © 2000 [cit. 02.04.2020]. Dostupné z: <https://rejstrik-firem.kurzy.cz/47916745/k-net-technical-international-group-sro/>
- (20) Váš IT partner | K-net. Váš IT partner | K-net [online]. Copyright © K [cit. 02. 04. 2020]. Dostupné z: <https://www.k-net.cz/>
- (21) What is Microsoft Flow? - Definition from WhatIs.com. Enterprise content management strategy, news and information –SearchContentManagement.com [online]. Dostupné z: <https://searchcontentmanagement.techtarget.com/definition/Microsoft-Flow>

- (22) JIRA Service Desk vs ServiceNow: Ticketing System Comparison. UpGuard: Security Ratings and Vendor Risk Management Software [online]. Copyright ©2020 UpGuard, Inc. [cit. 08. 04. 2020]. Dostupné z: <https://www.upguard.com/articles/servicenow-vs-jira>
- (23) Atlassian: Jira Overview [online]. [cit. 2020-04-17]. Dostupné z: <https://www.atlassian.com/software/jira/guides/getting-started/overview#about-the-jira-platform>
- (24) Jira Core – ceny | Atlassian. Atlassian | Software Development and Collaboration Tools [online]. Copyright © 2020 Atlassian [cit. 08. 04. 2020]. Dostupné z: <https://www.atlassian.com/cs/software/jira/core/pricing>
- (25) | Atlassian. Atlassian | Software Development and Collaboration Tools [online]. Copyright © 2020 Atlassian [cit. 08. 04. 2020]. Dostupné z: <https://www.atlassian.com/cs/software/jira/guides/getting-started/overview#about-the-jira-platform>
- (26) Installing Jira applications on Windows - Atlassian Documentation. [online]. Copyright © [cit. 08. 04. 2020]. Dostupné z: <https://confluence.atlassian.com/adminjiraserver/installing-jira-applications-on-windows-938846835.html>
- (27) Getting started with Jira Service Desk - Atlassian Documentation. [online]. Copyright © [cit. 08. 04. 2020]. Dostupné z: <https://confluence.atlassian.com/servicedeskserver/getting-started-with-jira-service-desk-939926015.html>
- (28) KEŘKOVSKÝ, Miloslav a Miloš DRDLA. *Strategické řízení firemních informací: teorie pro praxi*. Praha: C.H. Beck, 2003. C.H. Beck pro praxi. ISBN isbn80-7179-730-8.
- (29) McKinsey 7S - ManagementMania.com. [online]. Copyright © 2011 [cit. 10.05.2020]. Dostupné z: <https://managementmania.com/cs/mckinsey-7s>
- (30) The McKinsey 7S Framework - Strategy Skills From MindTools.com. Management Training and Leadership Training – Online [online]. Copyright © 2016.

All rights reserved. [cit. 10.05.2020]. Dostupné z:
https://www.mindtools.com/pages/article/newSTR_91.htm

SEZNAM OBRÁZKŮ

Obrázek 1 Ukázka workflow (Zdroj: 18)	16
Obrázek 2 7S analýza (Zdroj:29).....	11
Obrázek 3 SWOT analýza (Zdroj: 17).....	12
Obrázek 4 Uzlově definovaný síťový graf (Zdroj: 10, str. 52)	15
Obrázek 5 Hranově definovaný síťový graf (Zdroj: 10, str. 53).....	16
Obrázek 6 Logo společnosti (Zdroj: 20).....	19
Obrázek 7 Popis hlavních firemních toků (Zdroj: vlastní)	22
Obrázek 8 Struktura firmy K-net (Zdroj: vlastní).....	25
Obrázek 9 Současný stav žádanek (Zdroj: vlastní).....	29
Obrázek 10 Grafické zpracování časové analýzy (Zdroj: vlastní zpracování)	39
Obrázek 11 Přihlašovací stránka (Zdroj: vlastní)	54
Obrázek 12 Nastavení projektu – zákaznický portál (Zdroj: vlastní).....	55
Obrázek 13 Správa žádanky obecné (Zdroj: vlastní).....	55
Obrázek 14 Vytvoření nového uživatele (Zdroj: vlastní)	57
Obrázek 15 Ukázka workflow (Zdroj: vlastní).....	58
Obrázek 16 Nastavení portálu (Zdroj: vlastní)	59
Obrázek 17 Ukázka zákaznického portálu – nastavení (Zdroj: vlastní)	60
Obrázek 18 Zákaznická portál – uživatelské nastavení (Zdroj: vlastní).....	60
Obrázek 19 Zákaznický portál – zobrazení všech žádostí (Zdroj: vlastní).....	61
Obrázek 20 Vytvoření žádosti (Zdroj: vlastní).....	61
Obrázek 21 Zobrazení vytvořené žádosti (Zdroj: vlastní).....	62
Obrázek 22 Zobrazení e-mailu – vytvoření (Zdroj: vlastní).....	63
Obrázek 23 Zobrazení e-mailu (Zdroj: vlastní)	63
Obrázek 24 Změna nastavení (Zdroj: vlastní)	77
Obrázek 25 Změna nastavení – barvy (Zdroj: vlastní)	78
Obrázek 26 Změna nastavení – data (Zdroj: vlastní).....	78
Obrázek 27 Změna nastavení – e-mailové notifikace (Zdroj: vlastní)	79
Obrázek 28 Změna nastavení – e-mailové notifikace (Zdroj: vlastní)	79
Obrázek 29 Nastavení projektu (Zdroj: vlastní)	80
Obrázek 30 Tvorba hesla (Zdroj: vlastní).....	81

Obrázek 31 Změna nastavení – skupiny (Zdroj vlastní).....	82
Obrázek 32 Oprávnění (Zdroj: vlastní).....	82
Obrázek 33 Workflow – grafická podoba (Zdroj: vlastní)	83
Obrázek 34 Workflow – textová podoba (Zdroj: vlastní).....	83

SEZNAM TABULEK

Tabulka 1 Tabulka verbálních hodnot pravděpodobnosti (Zdroj:10, str. 80).....	14
Tabulka 2 Tabulka verbálních hodnot nepříznivých dopadů na projekt (Zdroj: 10, str. 80)	14
Tabulka 3 Vazební přiřazení pro verbální hodnoty rizika (Zdroj: 10, str. 80)	15
Tabulka 4 SWOT analýza firmy (Zdroj: vlastní zpracování)	27
Tabulka 5 Seznam činností (Zdroj: vlastní zpracování)	31
Tabulka 6 Identifikace rizik	44
Tabulka 7 Pořizovací náklady	65
Tabulka 8 Provozní náklady	66

SEZNAM PŘÍLOH

Příloha 1 Popis nastavení softwaru podle požadavků firmy	77
Příloha 2 Modul Knet OrgBoard	84
Příloha 3 Modul Knet Zadanky	90

PŘÍLOHY

Příloha 1 Popis nastavení softwaru podle požadavků firmy

Základní nastavení

Nastavení v Jira je celkově přehledné. Vše je krásně seřazeno.

Prvním krokem k tomu, aby náš systém působil méně genericky je upravit logo, název webu, přidat favicon a upravit barvy. Všechny tyto změny lze najít na jedné straně v nastavení. Vše si pak pouze naklikáme a máme systém, který vypadá pro naši firmu přívětivěji.

The screenshot shows the 'Logo' section of the Jira administration interface. It includes a 'Logo' heading, a 'Obnovení výchozího nastavení' button, and a note: 'Výška obrázku loga bude omezena na maximálně 30 px. Je rovněž doporučeno používat obrázek se šířkou 57 px.' Below this are two options for uploading a logo: 'Nahrát ze souboru' with a 'Vybrat soubor' button and 'Soubor nevybrán', and 'Nahrát z adresy URL' with a text input field and a 'Nahrát Logo' button. The 'Název webu' section has a checked checkbox 'Zobrazit název webu' and a 'Změnit' button. The 'Favicona' section has a 'Obnovení výchozího nastavení' button and a 'Nahrát ze souboru' section with a 'Vybrat soubor' button and 'Soubor nevybrán'.

Obrázek 24 Změna nastavení (Zdroj: vlastní)

Barvy

Pro použití barevného schématu, které odpovídá vašemu logu, klikněte sem.

Barva pozadí záhlaví	#172b4d	<input type="checkbox"/>	Vrátit
Barva zvýrazněného pozadí záhlaví	#136632	<input type="checkbox"/>	Vrátit
Barva oddělovače záhlaví	#09140d	<input type="checkbox"/>	Vrátit
Barva textu záhlaví	#ffffff	<input type="checkbox"/>	Vrátit
Barva zvýrazněného textu záhlaví	#ffffff	<input type="checkbox"/>	Vrátit
Barva pozadí zvýrazněné položky menu	#136632	<input type="checkbox"/>	Vrátit
Barva textu zvýrazněné položky menu	#ffffff	<input type="checkbox"/>	Vrátit
Barva pozadí tlačítka	#1a8743	<input type="checkbox"/>	Vrátit
Barva textu tlačítka	#ffffff	<input type="checkbox"/>	Vrátit
Barva aktivního odkazu	#0055ff	<input type="checkbox"/>	Vrátit
Barva záhlaví	#172b4d	<input type="checkbox"/>	Vrátit
Barva odkazu	#0052cc	<input type="checkbox"/>	Vrátit

Obrázek 25 Změna nastavení – barvy (Zdroj: vlastní)

Na této stránce lze upravit i informace jako je zobrazení formátu času, data.

Formáty data a času

Documentaci ohledně formátů data a času najdete na internetu.

Formát času	h:mm a Např. 3:55 AM
Formát dne	EEEE h:mm a Např. Wednesday 3:55 AM
Kompletní formát data a času	dd/MMM/yy h:mm a Např. 23/May/07 3:55 AM
Formát den/měsíc/rok	dd/MMM/yy Např. 23/May/07
Použít ISO8601 standard v Date Pickeru	Ne

Obrázek 26 Změna nastavení – data (Zdroj: vlastní)

Emailové notifikace

K tomu, aby nám chodili e-mailové notifikace, je nutné nastavit předem určité informace: jméno, server, port, a další fakta. Po zadání všech údajů si můžeme i vyzkoušet, zda je to správně a poslat si testovací e-mail.

Aktualizovat SMTP mail server

Tato stránka slouží k aktualizaci poštovního serveru SMTP. Prostřednictvím tohoto serveru bude odesílána veškerá odchozí pošta ze systému Jira.

Jméno	<input type="text" value="ittoastest@ittoys.cz"/>
Název serveru v rámci Jira.	
Popis	<input type="text" value="Testovací mail schránka"/>
Z adresy	<input type="text" value="ittoastest@ittoys.cz"/>
Výchozí adresa, kterou bude tento server používat k odeslání e-mailů.	
Předpona e-mailu	<input type="text" value="Service-Desk"/>
Tato předpona bude přidána před předmět všech odchozích e-mailů.	

Detaily serveru
Zadejte *buď* název hostitele serveru SMTP, *nebo* umístění JNDI objektu javax.mail.Session pro použití.

SMTP host

Protokol	<input type="text" value="SMTP"/>
Název hostitele	<input type="text" value="10.1.101.11"/>
Název SMTP hosta vašeho mailserveru.	
SMTP port	<input type="text" value="25"/>
Nepovinné - Číslo SMTP portu k použití. Ponechejte prázdné pro výchozí hodnoty (výchozí: SMTP - 25, SMTPS - 465).	
Časový limit	<input type="text" value="10000"/>
Časový limit v milisekundách - 0 nebo záporné hodnoty znamenají neomezený časový limit. Pokud nevyplníte, použije se výchozí hodnota (10000 ms).	
TLS	<input type="checkbox"/>
Nepovinné - mailový server vyžaduje použití TLS zabezpečení.	
Jméno uživatele	<input type="text" value="ittoastest@ittoys.cz"/>
Volitelné: Pokud k odeslání e-mailů používáte ověřený server SMTP, zadejte své uživatelské jméno.	
Změna hesla	<input type="checkbox"/>

nebo

Umístění JNDI

Umístění JNDI	<input type="text"/>
Umístění JNDI objektu javax.mail.Session, který již byla nastaven na aplikačním serveru systému Jira.	

Obrázek 27 Změna nastavení – e-mailové notifikace (Zdroj: vlastní)

Dá se také nastavit formát emailu. Buď mohou chodit formou textu, nebo HTML, my volíme HTML kvůli přehlednosti.

Implicitní nastavení uživatele

Nastavit výchozí hodnoty pro předvolby uživatelů. Můžete nastavení změnit pro každého z uživatelů byla jeho výchozí hodnotou.

Jméno

Výchozí formát odchozích e-mailů

Počet úloh zobrazených na jedné stránce

Přehled úloh

Výchozí sdílení pro filtry a řídicí panely

Upozornit uživatele na jejich vlastní změny

Automaticky sledovat vlastní úlohy

Upravit výchozí hodnoty

Akce

Použit

Formát emailu

Úloha na stránku

Počet úloh zobrazených na jedné stránce Přehledu úloh

Default access

Pro filtry a řídicí panely

Upozornit uživatele na jejich vlastní změny

Automaticky sledovat vlastní úlohy

Změnit Zrušit

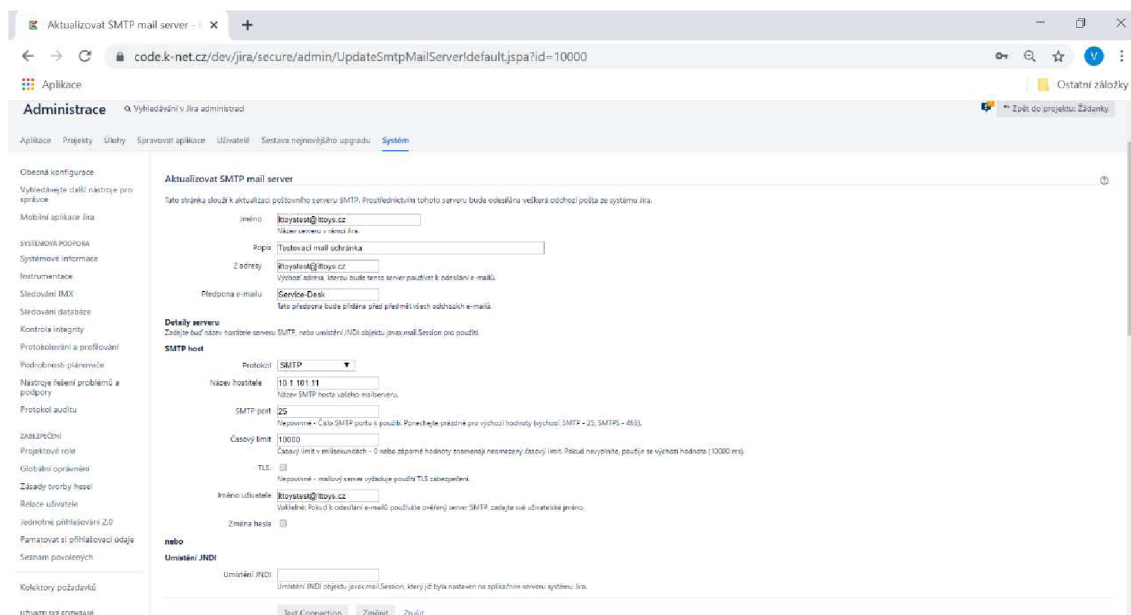
Obrázek 28 Změna nastavení – e-mailové notifikace (Zdroj: vlastní)

Nastavení projektů

Pokud máme nainstalovanou Jiru, jedním z prvních kroků je zvolit typ projektu, který budeme vytvářet. Jira Service Desk nabízí tyto druhy: základní service desk a IT Service desk. Pro náš projekt volíme Customer Service, ten totiž umožňuje zákaznický portál.

V základním nastavení můžeme měnit jméno projektu, jeho klíč, typ projektu a jeho popis. Důležitý je název a typ projektu. Od toho se odvíjí spousta dalších věcí.

V podrobnějším nastavení pak můžeme měnit ostatní informace jako je typ úkolu, workflow, pole, role uživatelů a další.



Obrázek 29 Nastavení projektu (Zdroj: vlastní)

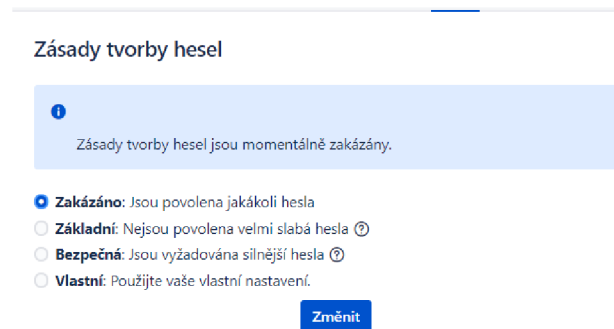
Jelikož používáme produkt Jira Service Desk, který nám umožňuje použít zákaznický portál, musíme i ten nastavit. Na obrázku vidíme typy požadavků, které máme vytvořené. Je možné jich mít i mnohem více. Lze zvolit typ úkolu a můžeme je zde i spravovat.

Heslo

V nastavení je možné i určit jaká hesla budeme po uživatelích v rámci bezpečnosti chtít. Základní – systém vyžaduje hesla, která musí mít délku alespoň 8 znaků a musí používat minimálně dva typy znaků. Odmítne hesla, která mají značnou podobnost s předchozím heslem, nebo s údaji uživatele.

Bezpečná – Systém vyžaduje hesla, která musí mít délku minimálně 10 znaků a musí používat minimálně 3 typy znaků včetně alespoň 1 speciálního znaku. Odmítne hesla, která mají značnou podobnost s předchozím heslem, nebo s údaji uživatele.

Pokud zvolíme možnost „vlastního“ je nutné zadat, co v hesle požadujeme.



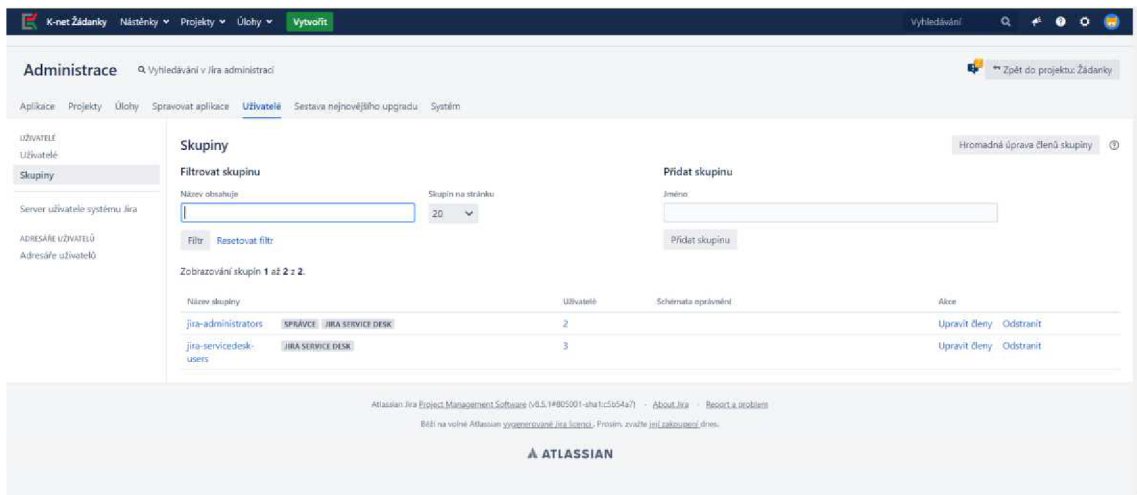
Obrázek 30 Tvorba hesla (Zdroj: vlastní)

Uživatelská práva

Jira má velké možnosti co se týče nastavení práv pro jednotlivé úkony a uživatele. Je možné si zde vybrat z široké škály možností podle skupin, přístupu a třeba přihlášení.

Pro náš projekt bylo nutné nastavit, aby každý zaměstnanec viděl pouze své žádosti, a ne ty ostatní. Aby je mohl jak žadatel, tak schvalovatel komentovat a aby mohl žadatel vkládat přílohy. Navíc skupiny budeme potřebovat pro rozdělení zaměstnanců do jednotlivých divizí, aby se žádanky samy přidělily jejich vedoucímu.

Vytvoření skupiny je velmi jednoduché. Vytváří se ve stejné sekci jako uživatelé, což je tedy – Uživatelé a skupiny. Vytvoří se skupina a pak se vyberou její členové.



Obrázek 31 Změna nastavení – skupiny (Zdroj vlastní)

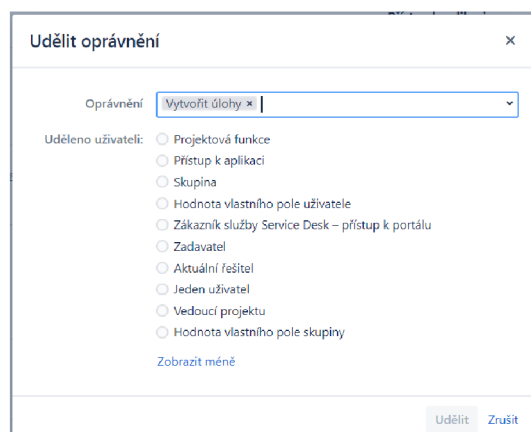
Se skupinou se dá pak pracovat v rámci oprávnění. Oprávnění můžeme přidat k jakékoliv činnosti, například: přiřaditelný uživatel, uzavírat úlohy, odstranit úlohy, upravit úlohy, měnit zadavatele, vyřešit úlohy, přidávat komentář, vytvářet přílohy a mnoho dalších funkcí.

Nastavení notifikací

Jedna z věcí, kterou jsme právě chtěli zlepšit, oproti stávajícímu řešení byly e-mailové notifikace. Do e-mailu Vám přijde přehledná zpráva, ve které může schvalovatel žádost hned schválit.

Můžeme si vybrat, zda chceme „html“ či „txt“ zobrazení.

Notifikace se upravují úplně stejně jako oprávnění uživatelů. Jsou zde vypsány různé úkoly a k těm můžeme přiřadit komu má upozornění přijít.

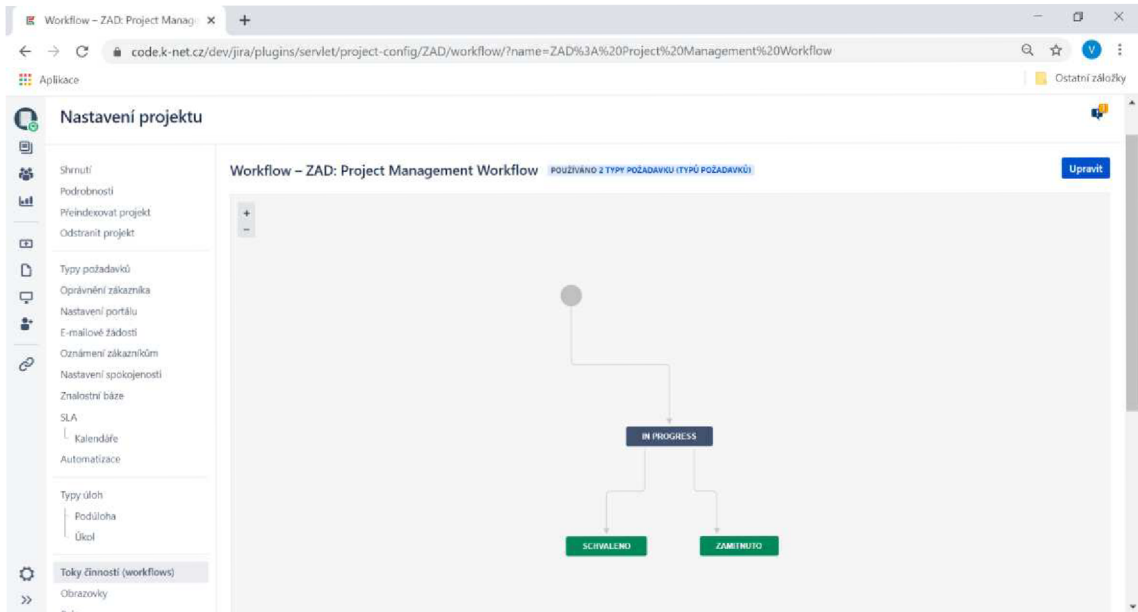


Obrázek 32 Oprávnění (Zdroj: vlastní)

Tvorba a nastavení workflow

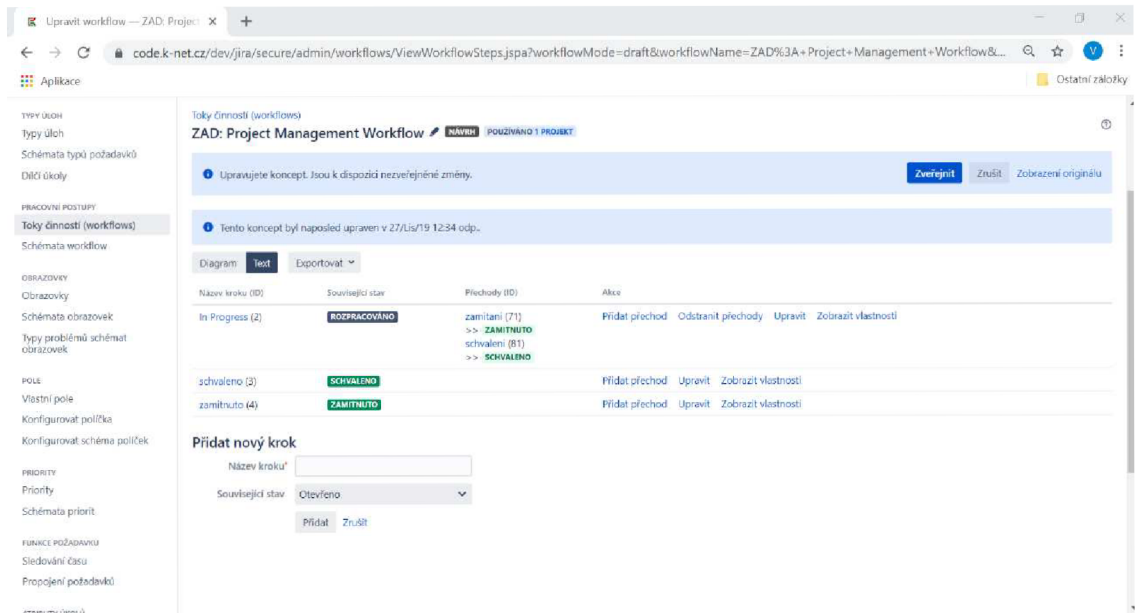
Workflow můžeme použít již přednastavené, nebo si vytvořit vlastní. Jira umožňuje jak grafické, tak textové úpravy. Je možné doplnit další kroky, či akce co se mají stát po procesu.

Takhle vypadá workflow pro náš projekt v grafické podobě.



Obrázek 33 Workflow – grafická podoba (Zdroj: vlastní)

Pokud se na něj podíváme v textové podobě, vypadá takto.



Obrázek 34 Workflow – textová podoba (Zdroj: vlastní)

Příloha 2 Modul Knet OrgBoard

V následující části se nachází kód k pluginům. Jednotlivé části jsou rozděleny na třídy, která jsou nadepsány třída a název třídy.

Třída - OrganizationBoardComponent

```
import com.atlassian.crowd.embedded.api.Group;
import cz.knet.internal.impl.exception.GroupNotInHierarchyException;
import cz.knet.internal.impl.exception.HierarchyNotDefinedException;

import java.util.List;

/**
 * Rozhraní modulu organizační struktury K-net
 * Tato třída je využívána dalšími moduly a mělo by to také být jediná třída,
 * která je viditelná z venčí
 * Třída je myšlena jako fasáda pro vnitřní implementaci modulu dle
 * návrhového vzoru Fasáda
 * https://en.wikipedia.org/wiki/Facade\_pattern
 */
public interface OrganizationBoardComponent {

    Group getDirectSupervisor(Group group) throws
    GroupNotInHierarchyException, HierarchyNotDefinedException;

    List<Group> getSupervisors(Group group) throws
    GroupNotInHierarchyException, HierarchyNotDefinedException;

}
```

Třída - Group hierarchi component

```
import com.atlassian.crowd.embedded.api.Group;
import com.atlassian.jira.security.groups.GroupManager;
import com.atlassian.plugin.spring.scanner.annotation.imports.ComponentImport;
import cz.knet.internal.api.OrganizationBoardComponent;
import cz.knet.internal.impl.KnetNamingConvention;
import cz.knet.internal.impl.dao.CacheHierarchyRepository;
import cz.knet.internal.impl.dao.HierarchyRepository;
import cz.knet.internal.impl.exception.GroupNotInHierarchyException;
import cz.knet.internal.impl.exception.HierarchyNotDefinedException;
import cz.knet.internal.impl.model.TreeNode;
import cz.knet.internal.impl.searcher.TreeSearcher;
import org.apache.log4j.LogManager;
import org.apache.log4j.Logger;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;

import java.util.List;

/**
 * Třída pracující s hierarchickou strukturou skupin
```

```

    * Jejím úkolem je odpovídat na požadavky jako je "Kdo je přímý nadřízený
    uživatelů v dané skupině?"
    * Bude dále rozšiřována o další funkce
    */
@Component
public class GroupHierarchyComponent {

    static final Logger log =
LogManager.getLogger(OrganizationBoardComponent.class);
    /**
     * Instance JIRA třídy GroupManager pro práci se skupinami
     */
    private final GroupManager groupManager;
    /**
     * Instance třídy TreeSearcher pro vyhledávání ve stromové struktuře
     */
    private final TreeSearcher treeSearcher = new TreeSearcher();
    /**
     * Instance třídy HierarchyRepository, která je zodpovědná za uloženou
    hierarchii
     */
    private final HierarchyRepository hierarchyRepository = new
CacheHierarchyRepository();

    @Autowired
    public GroupHierarchyComponent(@ComponentImport GroupManager groupManager)
    {
        this.groupManager = groupManager;
    }

    /**
     * Vrátí skupinu nadřazené skupině v parametru
     * Pokud skupina v parametru nemá nadřazenou skupinu (je na nejvyšším
    Levelu hierarchie), vrátí zadanou skupinu
     * <p>
     * Metoda načte existující hierarchickou strukturu, vyhledá v ní pomocí
    algoritmu prohledávání do hloubky uzlu
     * reprezentují danou skupinu a vrátí skupinu z uzlu jemu nadřazenému
     *
     * @throws GroupNotInHierarchyException Chyba značící, že zadaná skupina v
    hierarchii vůbec neexistuje
     * @throws HierarchyNotDefinedException Chyba značící, že hierarchie není
    definována
     */
    public Group getDirectSupervisor(Group group) throws
GroupNotInHierarchyException, HierarchyNotDefinedException {

        String groupName = group.getName();
        checkGroupNameMatchesConvention(groupName);

        TreeNode hierarchyRoot = hierarchyRepository.load()
            .orElseThrow(HierarchyNotDefinedException::new);

        TreeNode groupNode = treeSearcher.bfsSearchNode(hierarchyRoot,
            groupName)
            .orElseThrow(() -> new GroupNotInHierarchyException("Direct
            supervisor wasn't found"));
    }
}

```

```

        TreeNode parentNode = groupNode.getParent();

        if (parentNode.isRoot) {
            Log.warn("Returning the same group as input, because group is on
top of hierarchy");
            return group;
        }

        Group directSuperVisorGroup =
groupManager.getGroup(parentNode.getEntity().getGroupName());
//      Group directSuperVisorGroup = null;

        if (directSuperVisorGroup == null) {
            throw new RuntimeException(
                "Group with name [" +
parentNode.getEntity().getGroupName() + "]" +
                "wasnt found using groupManager");
        }

        return directSuperVisorGroup;
    }

    /**
     * Vrátí všechny nadřazené skupiny skupině v parametru
     * <p>
     * Pokud skupina v parametru nemá nadřazenou skupinu (je na nejvyšším
     Levelu hierarchie), vrátí zadanou skupinu
     *
     * @throws GroupNotInHierarchyException Chyba značící, že zadaná skupina v
     hierarchii vůbec neexistuje
     * @throws HierarchyNotDefinedException Chyba značící, že hierarchie není
     difinována
     */
    public List<Group> getSupervisors(Group group) throws
GroupNotInHierarchyException, HierarchyNotDefinedException {
        throw new IllegalStateException("Not implemented yet");
    }

    /**
     * Zkontroluje zda daný název skupiny odpovídá konvenci jinak vyhodí
     vyjímku GroupNotInHierarchyException
     *
     * @param groupName
     * @throws GroupNotInHierarchyException
     */
    private void checkGroupNameMatchesConvention(String groupName) throws
GroupNotInHierarchyException {
        if (!KnetNamingConvention.groupNameMatchesConvention(groupName)) {
            throw new GroupNotInHierarchyException("Group with name [" +
groupName + "]" + "doesn't matches naming convention, thus it cant be in
hierarchy");
        }
    }
}

```

Třída - Knet Naming Convection

```
import cz.knet.internal.impl.model.OrganizationUnitType;

/**
 * Třída pro práci s konvencemi pojmenování skupin dle organizační struktury
 * K-net
 */
public class KnetNamingConvention {

    public static Boolean groupNameMatchesConvention(String groupName) {
        return groupName.startsWith("ob.")
            && groupName.split("\\.").length > 2
            && groupName.split("_").length > 1
            && groupName.charAt(6) != 'o';
    }

    public static OrganizationUnitType decodeOrganizationUnitType(String
prefix) {

        String significantPrefixPart = prefix.split("\\.")[2];

        if (significantPrefixPart.equals("md")) {
            return OrganizationUnitType.DIVISION;
        }

        if (significantPrefixPart.matches("m\\d")) {
            return OrganizationUnitType.DEPARTMENT;
        }

        if (significantPrefixPart.matches("p\\d")) {
            return OrganizationUnitType.JOB;
        }

        throw new RuntimeException("Cant decode organization unit type");
    }
}
```

Třída - OrganizationBoardComponent

```
import com.atlassian.crowd.embedded.api.Group;
import com.atlassian.plugin.spring.scanner.annotation.export.ExportAsService;
import cz.knet.internal.api.OrganizationBoardComponent;
import cz.knet.internal.impl.exception.GroupNotInHierarchyException;
import cz.knet.internal.impl.exception.HierarchyNotDefinedException;
import org.apache.log4j.LogManager;
import org.apache.log4j.Logger;
import org.springframework.beans.factory.annotation.Autowired;

import javax.inject.Named;
import java.util.List;

/**
 * Defaultní implementace třídy OrganizationBoardComponent představující
 * fasádu nad vnitřní logikou modulu
 */
```



```

*/
@ExportAsService({OrganizationBoardComponent.class})
@Named("organizationBoardComponentImpl")
public class OrganizationBoardComponentImpl implements
OrganizationBoardComponent {

    private static final Logger Log =
LogManager.getLogger(OrganizationBoardComponent.class);
    /**
     * Instance třídy GroupHierarchyComponent
     */

    private final GroupHierarchyComponent groupHierarchyComponent;

    @Autowired
    public OrganizationBoardComponentImpl(GroupHierarchyComponent
groupHierarchyComponent) {
        this.groupHierarchyComponent = groupHierarchyComponent;
    }

    @Override
    public Group getDirectSupervisor(Group group) throws
HierarchyNotDefinedException, GroupNotInHierarchyException {
//         return groupHierarchyComponent.getDirectSupervisor(group);
        return null;
    }

    @Override
    public List<Group> getSupervisors(Group group) throws
HierarchyNotDefinedException, GroupNotInHierarchyException {
//         return groupHierarchyComponent.getSupervisors(group);
        return null;
    }
}

```

Třída - Plugin enabled listener

```

import com.atlassian.event.api.EventListener;
import com.atlassian.event.api.EventPublisher;
import com.atlassian.plugin.Plugin;
import com.atlassian.plugin.event.events.PluginEnabledEvent;
import com.atlassian.plugin.spring.scanner.annotation.imports.JiraImport;
import cz.knet.internal.Utils;
import cz.knet.internal.impl.dao.CacheHierarchyRepository;
import cz.knet.internal.impl.dao.HierarchyRepository;
import cz.knet.internal.impl.model.OrganizationUnit;
import cz.knet.internal.impl.model.TreeNode;
import cz.knet.internal.impl.parser.HierarchyParser;
import cz.knet.internal.impl.parser.HierarchyParserImpl;
import cz.knet.internal.impl.reader.FileGroupReader;
import cz.knet.internal.impl.reader.GroupReader;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.beans.factory.DisposableBean;

```

```

import org.springframework.beans.factory.InitializingBean;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;

import java.io.FileNotFoundException;
import java.util.List;

/**
 * Tato třída poslouchá události probíhající univř JIRA a umožňuje na ně
 reagovat
 * Třída se po spuštění pluginu zaregistruje jako posluchač, po deaktivaci
 nebo smazání pluginu je posluchač odregistrován.
 */
@Component
public class PluginEnabledListener implements InitializingBean,
DisposableBean {

    private static final Logger log =
LoggerFactory.getLogger(PluginEnabledListener.class);

    @JiraImport
    private final EventPublisher eventPublisher;

    private HierarchyRepository hierarchyRepository = new
CacheHierarchyRepository();

    private GroupReader groupReader = new FileGroupReader();

    private HierarchyParser hierarchyParser = new
HierarchyParserImpl(groupReader);

    @Autowired
    public PluginEnabledListener(EventPublisher eventPublisher) {
        this.eventPublisher = eventPublisher;
    }

    /**
     * Registrace posluchače po aktivaci pluginu
     */
    @Override
    public void afterPropertiesSet() throws Exception {
        log.info("Enabling plugin");
        eventPublisher.register(this);
    }

    /**
     * Odregistrace posluchače po deaktivaci nebo smazání pluginu
     */
    @Override
    public void destroy() throws Exception {
        log.info("Disabling plugin");
        eventPublisher.unregister(this);
    }

    /**
     * Událost reagující na zapnutí pluginu, pokud neexistuje uložená
 organizační hierarchie, tak spustí proces vytvoření nové

```

```

    */
    @EventListener
    public void onPluginEnabled(PluginEnabledEvent event) throws Exception {
        log.info("organizationBoardComponent onPluginEnabled _-");
        Plugin plugin = event.getPlugin();
        if ("cz.knet.internal.orgboard".equals(plugin.getKey())) {
            log.info("Plugin Enabled Event");
            if (!hierarchyRepository.load().isPresent()) {
                log.info("No existing hierarchy, let initialize one.");
                persistHierarchy(initHiearchy());
            }
        }
    }

    /**
     * Načte skupiny a vytvoří orgnaizační hierchii
     */
    private TreeNode initHiearchy() {
        Log.info("Initializing hiearchy...");
        List<OrganizationUnit> organizationUnits =
        hierarchyParser.parseOrganizationUnits();
        if (organizationUnits.size() == 0) {
            throw new RuntimeException("No organization units parsed,
            terminating initialization");
        }

        return hierarchyParser.parseHiearchy(organizationUnits);
    }

    /**
     * Uloží organizační hierarchii
     */
    private void persistHierarchy(TreeNode hierarchyRoot) {
        Log.info("Persisting hiearchy...");
        hierarchyRepository.save(hierarchyRoot);
        Log.info("Hiearchy persisted");
    }
}

```

Příloha 3 Modul Knet Zadanky

Třída - Issue Event Listener

```

import com.atlassian.event.api.EventListener;
import com.atlassian.event.api.EventPublisher;
import com.atlassian.jira.bc.issue.IssueService;
import com.atlassian.jira.bc.issue.search.SearchService;
import com.atlassian.jira.bc.project.ProjectService;
import com.atlassian.jira.config.ConstantsManager;
import com.atlassian.jira.event.issue.IssueEvent;
import com.atlassian.jira.issue.Issue;
import com.atlassian.jira.security.JiraAuthenticationContext;
import com.atlassian.plugin.spring.scanner.annotation.imports.JiraImport;

```

```

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.beans.factory.DisposableBean;
import org.springframework.beans.factory.InitializingBean;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;

/**
 * Tato třída poslouchá události probíhající univř JIRA a umožňuje na ně
 reagovat
 * Třída se po spuštění pluginu zaregistruje jako posluchač, po deaktivaci
 nebo smazání pluginu je posluchač odregistrován.
 */
@Component
public class IssueEventListener implements InitializingBean, DisposableBean {

    private static final Logger log =
LoggerFactory.getLogger(IssueEventListener.class);

    @Autowired
    private final IssueModifier issueModifier;

    @JiraImport
    private final EventPublisher eventPublisher;

    @JiraImport
    private IssueService issueService;
    @JiraImport
    private ProjectService projectService;
    @JiraImport
    private SearchService searchService;
    @JiraImport
    private JiraAuthenticationContext authenticationContext;
    @JiraImport
    private ConstantsManager constantsManager;

    @Autowired
    public IssueEventListener(IssueModifier issueModifier, EventPublisher
eventPublisher,
                               IssueService issueService,
                               ProjectService projectService,
                               SearchService searchService,
                               JiraAuthenticationContext
authenticationContext,
                               ConstantsManager constantsManager) {
        this.issueModifier = issueModifier;
        this.eventPublisher = eventPublisher;
        this.issueService = issueService;
        this.projectService = projectService;
        this.searchService = searchService;
        this.authenticationContext = authenticationContext;
        this.constantsManager = constantsManager;
    }

    /**
     * Registrace posluchače po aktivaci pluginu
     */

```

```

@Override
public void afterPropertiesSet() throws Exception {
    log.info("Enabling plugin");
    eventPublisher.register(this);
}

/**
 * Odregistrace posluchače po deaktivaci nebo smazání pluginu
 */
@Override
public void destroy() throws Exception {
    System.out.println("Disabling plugin");
    eventPublisher.unregister(this);
}

/**
 * Poslouchá události typu IssueEvent a reaguje na ně
 * Funguje také jako filtr. Filtrují se issues, které
 * - jsou v projektu žádánek
 * - mají daný issue type
 *
 * @param issueEvent
 */
@EventListener
public void onIssueEvent(IssueEvent issueEvent) {
    log.info("onIssueEvent cz.bp");
    Long eventTypeId = issueEvent.getEventTypeId();
    Issue issue = issueEvent.getIssue();

    if (IssueFilter.isListenedIssueEventType(eventTypeId)) {
        log.info("Listed issue event has been captured at {} issue key
        {}.", issue.getCreated(), issue.getKey());

        if (IssueFilter.isIssueInRequisitionProject(issue)) {
            log.info("Issue key {} is in target requisition project.",
            issue.getKey());

            if (IssueFilter.isIssueTypeRequisitionIssueType(issue)) {
                log.info("Issue key {} is in also in desired issue type.
                Let process edit", issue.getKey());
                issueModifier.handleIssueEdit(issue, eventTypeId);
            }
        }
    }
}
}
}

```

Třída - Issue Filter

```

import com.atlassian.jira.event.type.EventType;
import com.atlassian.jira.issue.Issue;
import cz.knet.internal.bp.impl.constants.IssueTypeConstants;
import cz.knet.internal.bp.impl.constants.ProjectConstants;

```

```

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

import java.util.Arrays;
import java.util.HashSet;
import java.util.Objects;
import java.util.Set;

/**
 * Pomocné metody pro filtrování takovým issue, které chceme modulem
 odchyťávat
 */
public class IssueFilter {

    private static final Logger log =
LoggerFactory.getLogger(IssueFilter.class);
    /**
     * Vrací zda se daný issue nachází uvnitř námi sledovaného projektu
     */
    public static Boolean isIssueInRequisitionProject(Issue immutableIssue) {
        return Objects.equals(immutableIssue.getProjectId(),
ProjectConstants.REQUISITION_PROJECT_ID);
    }
    /**
     * Vrací zda daný issue je typu, který chceme sledovat
     */
    public static Boolean isIssueTypeRequisitionIssueType(Issue
immutableIssue) {
        log.info("cz issue type" + immutableIssue.getIssueTypeId());
        return Objects.equals(
Objects.requireNonNull(immutableIssue.getIssueType()).getId().toLowerCase().trim(),
IssueTypeConstants.REQUISITION_ISSUE_TYPE_ID.toLowerCase().trim());
    }
    /**
     * Vrací zda zadané ID typu eventu je jedno ze sledovaných typu eventu
     */
    public static Boolean isListenedIssueEventType(Long eventTypeId) {
        return listenedIssueEventType().contains(eventTypeId);
    }
    /**
     * Vrací množinu sledovaných typů eventu
     */
    public static Set<Long> listenedIssueEventType() {
        return new HashSet<>(Arrays.asList(
EventType.ISSUE_CREATED_ID,
EventType.ISSUE_UPDATED_ID,
EventType.ISSUE_MOVED_ID));
    }
}
}

```

Třída - Issue Modifier

```

import com.atlassian.crowd.embedded.api.Group;
import com.atlassian.jira.bc.issue.IssueService;
import com.atlassian.jira.event.type.EventType;
import com.atlassian.jira.issue.CustomFieldManager;
import com.atlassian.jira.issue.Issue;
import com.atlassian.jira.issue.IssueInputParameters;
import com.atlassian.jira.issue.MutableIssue;
import com.atlassian.jira.issue.fields.CustomField;
import com.atlassian.jira.security.JiraAuthenticationContext;
import com.atlassian.jira.security.groups.GroupManager;
import com.atlassian.jira.user.ApplicationUser;
import com.atlassian.plugin.spring.scanner.annotation.imports.JiraImport;
import cz.knet.internal.bp.impl.constants.CustomFieldsConstants;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;

import java.util.ArrayList;

/**
 * Třída která provádí všechny operace související se samotnou editací issue
 */
@Component
public class IssueModifier {
    /**
     * Instance JIRA třídy IssueService pro práci s issues
     */
    @JiraImport
    private IssueService issueService;
    /**
     * Autentikační kontext JIRA pro získání aktuálního uživatele
     */
    @JiraImport
    private JiraAuthenticationContext authenticationContext;
    /**
     * Instance JIRA třídy GroupManager pro práci se skupinami
     */
    @JiraImport
    private GroupManager groupManager;
    /**
     * Instance JIRA třídy CustomFieldManager pro práci s uživatelskými
     definovanými poli
     */
    @JiraImport
    private CustomFieldManager customFieldManager;
    /**
     * Instance K-net třídy OrganizationBoardComponent pro získání
     hierarchické organizační struktury
     */
    @Autowired
    private final OrganizationBoardComponent organizationBoardComponent;

    @Autowired
    public IssueModifier(
        OrganizationBoardComponent organizationBoardComponent,
        CustomFieldManager customFieldManager,
        GroupManager groupManager,

```

```

        IssueService issueService,
        JiraAuthenticationContext authenticationContext) {
    this.groupManager = groupManager;
    this.customFieldManager = customFieldManager;
    this.issueService = issueService;
    this.authenticationContext = authenticationContext;
}

/**
 * Obecné provedení upravení polí issue
 *
 * @param immutableIssue Issue k upravení
 */
public void handleIssueEdit(Issue immutableIssue, Long eventTypeId) {
    ApplicationUser user = authenticationContext.getLoggedInUser();
    MutableIssue issue = issueService.getIssue(user,
immutableIssue.getKey()).getIssue();

    if (EventType.ISSUE_CREATED_ID.equals(eventTypeId)) {
        handleRequisitionCreatedEvent(issue, user);
    }
}

/**
 * Specifcké provedení úpravy pro vytvoření issue
 * @param issue Zakládání issue
 * @param user Uživatel zakládající issue
 */
private void handleRequisitionCreatedEvent(MutableIssue issue,
ApplicationUser user) {

    IssueInputParameters issueInputParameters =
issueService.newIssueInputParameters();

    // Získat skupinu zakladatele
    CustomField creatorGroupCustomField =

customFieldManager.getCustomFieldObject(CustomFieldsConstants.CUSTOM_FIELD_ID
_CREATOR_POSITION);
    String creatorGroupName = (String)
creatorGroupCustomField.getValue(issue);

    // Získat skupinu nadřízeného
    Group directSuperVisitorGroup =
getSupervisorGroupForGroup(groupManager.getGroup(creatorGroupName));
    // Získat skupinu vedoucího divize
    Group divisionLeadGroup =
getSupervisorGroupForGroup(directSuperVisitorGroup);

    // Nastavit pole s vedoucím (schvalovatelem)
    CustomField customFieldApprovers =
customFieldManager.getCustomFieldObject(CustomFieldsConstants.CUSTOM_FIELD_ID
_DIRECT_SUPERVISOR);
    issue.setCustomFieldValue(customFieldApprovers,
getFirstUserFromGroup(directSuperVisitorGroup));
}

```



```

        // Nastavit pole s vedoucím divize (schvalovatelem)
        CustomField customFieldDivision =
customFieldManager.getCustomFieldObject(CustomFieldsConstants.CUSTOM_FIELD_ID
_DIVISION_SUPERVISOR);
        issue.setCustomFieldValue(customFieldDivision,
getFirstUserFromGroup(divisionLeadGroup));

        IssueService.UpdateValidationResult result =
            issueService.validateUpdate(user, issue.getId(),
issueInputParameters);
        issueService.update(user, result);
    }

    /**
     * Získání nadřazené skupiny pro danou skupinu
     * @param group Skupina pro kterou potřebujeme nadřazenou skupinu
     * @return Nadřazená skupina dle firemní organizace
     */
    private Group getSupervisorGroupForGroup(Group group) {
        return organizationBoardComponent.getDirectSupervisor(group);
    }

    private ApplicationUser getFirstUserFromGroup(Group group) {
        return new
ArrayList<>(groupManager.getUsersInGroup(group.getName())).get(0);
    }
}

```

Třída - Requisition Component

```

import com.atlassian.plugin.spring.scanner.annotation.export.ExportAsService;
import com.atlassian.plugin.spring.scanner.annotation.imports.ComponentImport;
import com.atlassian.sal.api.ApplicationProperties;
import cz.knet.internal.bp.api.RequisitionComponent;

import javax.inject.Inject;
import javax.inject.Named;

@ExportAsService({RequisitionComponent.class})
@Named("requisitionComponentImpl")
public class RequisitionComponentImpl implements RequisitionComponent {

    @Inject
    public RequisitionComponentImpl() {
    }
}

```