



ZADÁNÍ DIPLOMOVÉ PRÁCE

Název:	Návrh systému na řízení znalostí
Student:	Bc. Marek Elznic
Vedoucí:	Ing. Petra Pavlíčková, Ph.D.
Studijní program:	Informatika
Studijní obor:	Webové a softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	Do konce letního semestru 2019/20

Pokyny pro vypracování

Cílem práce je analyzovat a popsat způsoby a přístupy k uchování firemních znalostí. Dále pak práce poskytne sadu doporučení usnadňujících uchovávání znalostí pro firmy se zaměřením na segment IT.

1. Prostudujte a popište základy řízení znalostní a způsoby, jakým se firemní znalosti uchovávají.
2. Vysvětlete přínosy řízení znalostí pro organizaci a zanalyzujte je vhodnou analytickou metodou.
3. Vyberte typové příklady firem se zaměřením na segment IT a zanalyzujte jejich možnosti uchování znalostí.
4. Navrhněte informační systém, který bude sloužit jako znalostní báze ve střední IT firmě.
5. Zvolený postup zhodnoťte a doporučte další rozvoj.

Seznam odborné literatury

Dodá vedoucí práce.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
děkan

V Praze dne 29. prosince 2018



**FAKULTA
INFORMAČNÍCH
TECHNOLGIÍ
ČVUT V PRAZE**

Diplomová práce

Návrh systému na řízení znalostí

Bc. Marek Elznic

Katedra softwarového inženýrství

Vedoucí práce: Ing. Petra Pavlíčková Ph.D.

7. května 2019

Poděkování

V prvním řadě bych chtěl poděkovat vedoucí práce Ing. Petře Pavlíčkové, Ph.D. za ujmoutí se této práce a za její cenné rady nejen během vzniku práce, ale i během celého studia. Dále bych rád poděkoval své rodině, přítelkyni a přátelům za podporu a trpělivost po dobu psaní této práce.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 7. května 2019

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2019 Marek Elznic. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.

Odkaz na tuto práci

Elznic, Marek. *Návrh systému na řízení znalostí*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2019.

Abstrakt

Tato práce se zabývá problematikou řízení znalostí uvnitř organizace. Konkrétně zkoumá znalostní management a systémy znalostního managementu, které podporují řízení znalostí. Práce představuje principy jejich návrhu, zkoumá aktuální dostupná řešení a doporučuje vhodná řešení pro typové příklady organizací z IT segmentu. Cílem práce je ukázat základy znalostního managementu a navrhnout informační systém, který bude sloužit jako znalostní báze ve středně velké IT organizaci. Hlavním přínosem práce je navržený systém, který může být v představené organizaci snadno implementován.

Klíčová slova znalosti, znalostní management, znalostní systém, znalostní databáze, návrh systému

Abstract

This thesis deals with issues of knowledge management within an organisation. Notably, it examines knowledge management systems that support knowledge management. It introduces the principles of their design, analyses currently available solutions, and recommends appropriate solutions for typical examples of organisations from the IT segment. The thesis aims to show the basics of knowledge management and to design an information system

that will serve as a knowledge base in a medium-sized IT organisation. The main contribution of the thesis is the design of the information system that can be easily implemented in the presented organisation.

Keywords knowledge, knowledge management, knowledge system, knowledge base, system design

Obsah

Úvod	1
1 Cíl práce	3
2 Úvod do managementu znalostí	5
2.1 Data, informace a znalosti	5
2.2 Taxonomie znalostí	9
2.3 Dopady pohledů na znalosti	11
2.4 Znalostní management	12
2.5 Modely znalostních procesů	14
2.6 Strategie knowledge managementu	18
2.7 Přínosy znalostního managementu	20
2.8 Znalostní pracovník	22
2.9 Kritika znalostního managementu	26
2.10 Závěr	27
3 Systémy znalostního managementu	29
3.1 Využití systému znalostního managementu	30
3.2 Fáze vývoje systému znalostního managementu	30
3.3 Rozdíl oproti běžnému informačnímu systému	32
3.4 Závěr	36
4 Doporučení pro návrh systému	37
4.1 Analýza	37
4.2 Definice cílů	37
4.3 Definice znalostí	38
4.4 Integrace	38
4.5 Gamifikace	38
4.6 Závěr	46

5	Existující řešení	47
5.1	Metodika	47
5.2	Vybraná řešení	49
6	Doporučení pro IT firmy	57
6.1	Kritéria	57
6.2	Organizace orientované na služby	58
6.3	Organizace orientované na produkt	60
7	Analýza aktuálního stavu	63
7.1	Definice zákazníka	63
7.2	Analýza aktuálního stavu	67
7.3	Definice cílového stavu	68
7.4	Návrh informačního systému	69
7.5	Použité informační technologie	69
7.6	Zhodnocení návrhu	71
8	Návrh informačního systému	73
8.1	Funkční požadavky	73
8.2	Nefunkční požadavky	75
8.3	Architektura	75
8.4	Aktéři	77
8.5	Případy užití	78
8.6	Gamifikace	90
8.7	Možná vylepšení	91
8.8	Zhodnocení návrhu	92
	Závěr	93
	Literatura	95
	A Seznam použitých zkratk	99
	B Návrh uživatelského rozhraní	101
	C Obsah příloženého CD	109

Seznam obrázků

21	Pohledy znalostního managementu [8]	8
31	Cynefin framework [32]	33
41	Foggův model chování [37]	40
42	Perspektiva hráčů a herních architektů	41
43	Progress bar na sociální síti LinkedIn	43
44	Odznaky v mobilní aplikaci Duolingo	44
45	Vizualizace Social engagement loop	45
81	Monolitická a microservice architektura [48]	76
82	Architektura navrhovaného systému	78
83	Use case diagram	79
B1	Přihlášení do systému	101
B2	Dashboard	102
B3	Dashboard	102
B4	Nový záznam	103
B5	Detail záznamu	103
B6	Profil	104
B7	Přehled označení	104
B8	Feed poslední aktivity	105
B9	Záznamy, pro které je přihlášený uživatel ověřovatel	105
B10	Ověření záznamu	106
B11	Detail expertní otázky	106
B12	Responzivní verze Dashboardu	107

Seznam tabulek

51	Hodnocení Confluence	50
52	Hodnocení Zendesk Guide	51
53	Hodnocení Bloomfire	52
54	Hodnocení Guru	53
55	Hodnocení AllAnswered	53
56	Hodnocení Nuclino	54
57	Hodnocení OpenKM	55
61	Rozdělení podniků dle obratu	58
81	Uživatelé systému	77
82	Bodové zisky	91

Úvod

Každý rok rostou světová data nade všechny meze. Studie firmy Cisco uvádí, že do roku 2025 bude na světě 175 Zettabytů dat. Průměrný roční růst tak činí více než 60 %. Data pohánějí celý segment IT od Umělé inteligence po IoT. S daty je možné volně obchodovat, analyzovat je a získávat z nich cenné informace či znalosti.

Znalosti však mají tendenci se v organizacích ztrácet či spolu s jejich nositeli z organizace odchází úplně. Znalosti, které v organizaci nejsou k nalezení ve správný čas, pro organizaci nemají žádný přínos. Ačkoli odliv znalostí není možné plně eliminovat, za použití správných nástrojů je možné minimalizovat dopady.

Cíl práce

Cílem práce je čtenáře obeznámit s tématem znalostního managementu. Práce popíše základní pojmy a ukáže základní principy, na kterých je znalostní management založen. Dále práce rozebere možnosti uchování znalostí pomocí znalostních databází a řízení znalostí pomocí informačních systémů znalostního managementu. Práce rovněž představí, jak se takové systémy navrhují a jaké systémy již na trhu existují. Pro typové příklady organizací z IT segmentu práce doporučí pravidla pro zavádění znalostních systémů a vybere vhodné řešení. Hlavním výsledkem práce je analýza a návrh systému znalostního managementu pro střední firmu v IT segmentu.

Práce je organizována do 8 kapitol. Každá kapitola má za cíl přestavit či vyřešit nějaký konkrétní problém. Každá kapitola obsahuje vlastní závěr, který shrnuje obsah kapitoly, či představuje její závěr.

Úvod do managementu znalostí

První náznak pohledu na management znalostí (anglicky též *knowledge management*) se začíná objevovat v literatuře o strategickém řízení podniku. Tento pohled říká, že produkty nebo služby jsou formovány jak samotnými vstupů (zdroji), tak i znalostmi, které organizace má (ty jsou často označovány také jako *know-how*). [1]

2.1 Data, informace a znalosti

Aby však bylo možné mluvit o znalostech, je nutné ujasnit, co se pod pojmem znalost rozumí. Většina klasických zdrojů rozděluje data, informace a znalosti. Některé novější zdroje přidávají ještě porozumění (*understanding*) a moudrost (*wisdom*). [2] [3]

2.1.1 Data

Data jsou symboly či znaky, které dokážeme zapsat nebo uložit. Nemají žádný význam a nemusí k ničemu sloužit. [3]

2.1.2 Informace

Informace je pojem velmi obecný, který zkoumá mnoho vědních oborů, každý svými teoretickými nástroji. V teorii komunikace informací definoval Shannon jako statistickou pravděpodobnost určitého symbolu, který je dán na vstup systému. Čím menší je pravděpodobnost daného symbolu, tím větší je jeho informační hodnota. Tedy pakliže je daný symbol na vstupu méně očekáván, jeho výskyt je překvapivější. To vede na definici entropie (míry nejistoty). [4] [5]

Jiné zdroje informací přikládají i další vlastnosti. Pro cíl této práce je důležité, že informace vzniká zpracováním (interpretací) dat. Tedy když je datům dán nějaký význam. Dle Belingera informace nemá žádnou hodnotu (je

neutrální). Hodnotu jí přisuzuje teprve člověk v procesu poznání. Informace obvykle poskytují odpovědi na otázky „Kdo“, „Kde“, „Kdy“ nebo „Co“. [6]

2.1.3 Znalosti

První pokusy o definování znalosti prováděli již klasičtí Řečtí filozofové. Nejznámější je asi *Justified true belief* (česky též důvodné pravdivé přesvědčení), které některé prameny připisují Platónovi.

Dle tradiční definice znalosti jakýkoli subjekt S ví, že výrok P je pravdivý tehdy a jen tehdy, jsou-li splněny tři podmínky:

1. P je skutečně pravdivé.
2. S věří, že P je pravdivé.
3. Víra S o pravdivosti P je důvodná.

V roce 1963 však Edmund Gettier popsal takzvaný Gettierův problém. [7] Ve svém článku uvádí dva příklady, kdy jsou splněny všechny tři podmínky, avšak osoby vystupující v těchto příkladech stále neví, že P je pravdivé. Uvedu zde pouze jeden:

Předpokládám, že Smith je důvodně přesvědčen, že Jones vlastní auto značky Ford (a). Důvodem může být, že Smith zná Jonse dlouho a vždy vlastnil auto značky Ford. Další důvod může být, že nedávno Jones vezl Smitha domů autem značky Ford. Dále nechť si čtenář představí, že Smith má dalšího známého jménem Brown. Smith se příliš nezajímá o to, kde Brown zrovna pobývá a vymyslí si tedy tři místa, na kterých by mohl být a vyřkne následující tři výroky:

1. Buď Jones vlastní Ford nebo Brown žije v Bostonu.
2. Buď Jones vlastní Ford nebo Brown žije v Barceloně.
3. Buď Jones vlastní Ford nebo Brown žije v Brestu.

Smith akceptuje správnost všech tří výroků na základě výroku (a), o kterém je důvodně přesvědčen, že platí, a který je s nimi logicky spojen. Smith však samozřejmě netuší, kde Brown reálně žije.

Nyní položím ještě předpoklad, že Jones auto značky Ford nevlastní, neboť ho má pronajaté. Zároveň, shodou okolností, Brown skutečně v Barceloně žije.

Pakliže tyto podmínky platí, Smith netuší, že výrok 2 je pravdivý, ačkoli výrok 1. pravdivý je, 2. Smith věří, že je pravdivý, a 3. Smith je důvodně přesvědčen, že je pravdivý. [7]

Pro potřeby této práce se však vrátím k modernějším pohledům na znalosti. Alavi a Leidner ve své práci definovaly pět základních pohledů na znalosti:

1. Znalost je možné popsat jako stav vědění, přičemž vědění je chápáno jako porozumění nabitě skrze zkušenost nebo studium (vnímání, objevování a učení).
2. Na znalost lze nahlížet jako na objekt. Tento pohled nejvíce odpovídá znalostnímu managementu, neboť s objektem lze libovolně manipulovat a ukládat ho.
3. Procesní pohled na znalost nahlíží jako na vědění a chování. Subjekt si tedy je vědom znalosti a s přihlédnutím k ní se chová.
4. Čtvrtý pohled zkoumá především přístup ke znalosti. Za klíčové v rámci podnikových znalostí označuje jejich vyhledatelnost a dostupnost.
5. Poslední pohled je definován schopností (způsobilostí) ovlivnit budoucí rozhodnutí. V tomto případě by však znalost neměla sloužit jedné konkrétní akci, ale schopnosti využívat znalosti a učení k interpretaci informací a následnému rozhodování. [2]

Zatímco Kay a Cecez-Kecmanovic staví na předchozích pohledech Alavi a Leidner, vybírají pouze 4 pohledy, které proti sobě staví do dvoudimenzionálního prostoru. Jejich argumentace je založená na výběru dvou hlavních dimenzí znalosti – gnozeologické a ontologické.

Než postoupím dál, připomenu, že gnozeologie se zabývá poznatelností světa. Hlavní otázky jsou, zda je svět poznatelný a případně jak. Naproti tomu ontologie se zabývá samotnou podstatou bytí (jak vznikl svět, co bylo první atp.).

Gnozeologická dimenze obsahuje pohled na znalost jako na objekt a jako na proces. Ontologická dimenze pak obsahuje na jedné straně znalost jedince a na druhé straně znalost skupiny nebo společnosti. Spojením těchto dvou dimenzí dostaneme zmiňovaný dvoudimenzionální prostor znázorněný na obrázku 21. [8]

Zpracování informací vychází z předpokladu, že znalost je individuální a její ideální konceptualizace je objekt. Výsledkem je tradiční pojetí znalostního managementu. Znalosti mohou být uchovány ve znalostních databázích, do kterých jednotlivci mohou přistupovat.

Inovace a spolupráce předpokladem znalosti jako individuální vlastnosti a pohledem na znalost jako na proces. Výsledkem je méně tradiční a velmi složité pojetí znalostního managementu, který se soustředí na maximalizaci užití zaměstnanců a jejich znalostí. Zároveň se snaží poskytnout pracovní prostředí, které k tomu bude napomáhat.

Nehmotná aktiva vznikají viděním znalosti jako vlastnosti společnosti či organizace a předpokladem, že znalost je objektivizovatelná. Jako příklad je možné uvést duševní vlastnictví a patenty, které jsou považovány za součást organizace s měřitelnou hodnotou, kterou je možné vyčíslit (jelikož patenty mohou být například prodány).



Obrázek 21: Pohledy znalostního managementu [8]

Organizační schopnosti jsou spojením znalosti jako vlastnosti společnosti či organizace a procesní konceptualizace. Výsledkem je často konkurenční výhoda, kterou organizace může držet za předpokladu, že celá organizace má schopnosti, které jednotlivci nemají.

2.1.4 Porozumění

Ackoff k tradiční trojici data, informace a znalosti, přidává i porozumění. Popisuje ho jako kognitivní a analytický proces, kterým subjekt ze svých znalostí vytváří nové znalosti. Z nově nabitých znalostí může získat další porozumění. Bez porozumění není možné získávat nové znalosti. Subjekt díky porozumění může volit vhodné budoucí akce či alespoň získávat nové informace nebo znalosti. Rozdíl mezi znalostmi a porozuměním je podobný, jako rozdíl mezi učením a zapamatováním si. [3]

Naproti tomu Bellinger porozumění chápe více jako podpůrný proces pro přechod mezi znalostmi a moudrostí. Nikoli jako nový stav obsahu lidské mysli. [6]

Porozumění je důležité i pro počítačovou vědu. Neboť právě porozuměním se velmi zabývá umělá inteligence. Ta je schopna z dat a informací vybudovat

znalosti, kterým následně porozumí. Jen díky porozumění je umělá inteligence schopna sama vytvářet další znalosti.

2.1.5 Moudrost

Moudrost je nedeterministický a extrapoláčn  proces, kter m je mozn  ovlivnovat budouc  akce. Na z klad  okolnost  a porozum n  je subjekt schopen vybrat dal s  nejvhodn j s  akci. Oproti porozum n  se tak nezab v  jen principy, ale i jejich aplikac . Bellinger moudrosti p isuzuje i rozpozn n  spr vn ho a  patn ho nebo dobra a zla. [6] [9] Naopak jin  auto ri pro tyto vlastnosti vytv r  zcela nov  koncept osv cen    pou en  (*enlightenment*). [10]

V t sina zdroj  se v ak shoduje,  e porozum n  a moudrost je hlavn  rozd l mezi  lov kem a po  ta em. P edev s m argumentuj ,  e um l  inteligence nikdy nebude schopna extrapolovat moudrost ze z skan ho porozum n . Moudrost je tedy v sostn m stavem lidsk ho v d n . [3] [6]

2.2 Taxonomie znalost 

Pochopen  taxonomie znalost  a znalost  obecn  je d lezn , proto e p rav  taxonomie znalost  ovlivnuje v voj znalostn ho managementu. V d sledku tak taxonomie znalost  ovlivnuje i n vrh informa n ch syst m , kter  znalosti spravuj .

Z kladn  taxonomii znalost  polo il Michael Polanyi a pozd ji ji roz s rili Nonaka a Takeuchi. A koli nejzn m j s  v stupem Polanyiho d la je jeho d len  mezi tacitn mi a explicitn mi znalostmi, ve sv  pr ci jde hloub ji. Polanyi se v ce soustred  na to, jak se tacitn  a explicitn  znalosti dopl uj , ne  jak se d l . Podle Polanyiho se lid  pohybuj  mezi explicitn mi a tacitn mi znalostmi neust le. Tacitn  znalosti tak tvo r  z klad, na kter m vznikaj  v schny dal s  formy znalost . [11]

2.2.1 Tacitn  a explicitn 

Tacitn  znalosti jsou p edev s m dovednosti, zku enosti nebo intuice konkr tn ho jedince, kter  jsou zasazeny do kontextu a jsou v dy spojeny s postupy a  innostmi. Jedna z hlavn ch vlastnost  tacitn ch znalost  je vysok  sv zanost s osobnost  jedince. Externalizovat (p en st na jin ho jedince) ji prakticky nelze. Zaj mav  je,  e u tacitn ch znalost  jedinec obvykle nezn  zdroj znalosti nebo zp sob, kter m znalost z skal.

Nonaka a Takeuchi tacitn  znalosti rozd luj  je t  na kognitivn  slo ku a technickou slo ku. Kognitivn  slo ka se skl d  p edev s m z postoj , p esv d en  a ment ln ch model . Jedinci tak dod v  perspektivu, s kterou pohl z  na okoln  sv t. Technick  slo ka se skl d  z dovednost  (*know-how*), kter  jsou u ite n  v ur it m kontextu. Jako p r klad je mozn  uv st schopnost vykonat chirurgickou operaci. [2] [11]

Explicitní znalosti se vyznačují zjevností, přístupností, kodifikovatelností a přenositelností. To znamená, že na rozdíl od tacitních znalostí je možné explicitní znalosti artikulovat, zapsat a přenést na další jedince. V kontrastu k tacitním znalostem, které jsou často označovány jako *know-how*, jsou explicitní znalosti označovány jako *know-that*.

Z popisu tacitních a explicitních znalostí by mělo být jasné, že znalostní management se stará především o znalosti explicitní. Tento fakt vychází z podstaty tacitních znalostí, které nejsou kodifikovatelné a přenositelné na další jedince. Byla by však chyba se striktně omezit na explicitní znalosti, neboť bez tacitních znalostí ty explicitní nemohou existovat. Kvalitně zvládnutý znalostní management by měl podporovat individuální rozvoj tacitních znalostí každého člena organizace. [12]

Mnoho autorů se rovněž rozchází v pohledu na hodnotu tacitních a explicitních znalostí. Ačkoli Polanyi i Nonaka a Takeuchi se shodují, že tacitní znalosti jsou hodnotnější, neboť bez nich explicitní znalosti nemohou existovat, Bohn se domnívá, že hodnotnější jsou znalosti explicitní. [2] [8] [11]

2.2.2 Individuální a společenské

Polanyi mluví o znalosti striktně jako o vlastnosti jedince. Nonaka a Takeuchi přidávají teorii organizační či společenské znalosti. Popisují proces vzniku jako spirálu, ve které expanze explicitní znalosti umožňuje vytváření znalostí na vyšších úrovních organizace. Tento výsledek jejich práce se promítl do novějších pohledů na znalosti diskutovaných v kapitole 2.1.3 a na jejich základě Alavi a Leidner vytvořili vlastní kategorii pro individuální a společenské znalosti. [2] [12]

2.2.3 Procedurální a deklarativní

Deklarativní znalosti se skládají především z faktů a čísel. Naproti tomu procedurální znalosti jsou znalosti o prostředcích použitých k dosažení deklarativních znalostí. Není jasné, zda se procedurální znalosti vyvíjejí z deklarativních nebo zda deklarativní znalosti hrají roli ve vývoji procedurálních znalostí. Avšak je možné říci, že se nevyvíjejí v postupy. Je totiž možné najít příklady postupů, které se vyvinuly bez použití deklarativních znalostí (např. rozpoznávání obličejů).

Z hlediska počítače jsou deklarativní (data) a procedurální (programy) znalosti nerozlišitelné. Oba se nacházejí v paměti počítače ve stejném formátu. Ve Von Neumannově architektuře počítače je pro data i instrukce dokonce využívána stejná paměť. Nicméně je zásadní umět rozhodnout, kterou znalost je vhodné uložit (zapamatovat si) jako procedurální a kterou jako deklarativní. Pamatovat si malou násobilku je často praktické, avšak pro větší čísla je vhodné znát postup násobení dvou čísel. [12]

2.2.4 Selský rozum a expertní znalost

Znalosti vycházející ze selského rozumu jsou takové, které by měl každý člen společnosti znát. Jedná se o společensky akceptovatelné chování, způsob provádění jednoduchých obchodních transakcí nebo jednoduchou fyziku. Tyto znalosti se napříč různými společnostmi mohou velmi lišit.

Expertní znalosti jsou připisovány uznávaným odborníkům, kterých je typicky pouze omezené množství. Zvyšuje-li se počet jedinců, kteří danou znalostí disponují, stává se expertní znalost spíše znalostí selského rozumu. Příkladem takového procesu může být psaní. Původně bylo psaní výsadou vysoce vzdělaných jedinců. S příchodem univerzálního vzdělání se psaní stalo znalostí, kterou disponuje většina společnosti. [12]

2.2.5 Další možnosti dělení

Literatura uvádí i mnoho dalších možností, jak dělit znalosti. Jednou z nich je kontext, který je potřeba, aby mohla být znalost použita k vykonání nějakého úkolu. Kontext může být v rámci organizace interní (např. kam bude ukládána příchozí pošta) a externí (např. tvorba marketingové strategie). Znalost s interním kontextem je snadná na udržení, avšak i její zdroj je nutné v organizaci identifikovat. Znalost s externím kontextem je složitější na udržení, neboť zdroj je nutné hledat mimo organizaci.

Dále je znalosti možné dělit dle pravdivosti. Existují znalosti, o kterých jedinec ví, že jsou pravdivé nebo naopak předpokládá, že jsou nepravdivé. Všechny znalosti se pak nachází někde mezi těmito extrémy.

Znalosti je možné dělit i dle stupně jistoty, který k ní jedinec chová. Jsou znalosti, které mohou vycházet z přesně naměřených dat, a tudíž jim jedinec může plně důvěřovat. Naopak může existovat znalost, která byla vyvozena pouze z několika náhodných pozorování. I taková znalost pak může být užitečná, je-li k ní přistupováno s nižším stupněm jistoty. [12]

2.3 Dopady pohledů na znalosti

V kapitole 2.1.3 jsem se mimo definici znalosti věnoval i různým pohledům na znalosti. Každý pohled na znalost však implikuje jiný pohled na znalostní management (a ve výsledku i na informační systémy znalostního managementu).

Budu-li nahlížet na znalost jako na stav mysli (či vědění), cílem znalostního managementu bude posílení individuálního porozumění jedince. K porozumění je nutné poskytnout informace a umožnit studium, což má dopady i na informační systém. Jeho hlavním úkolem bude poskytování zdrojů znalostí, které umožní studium, avšak ne nutně samotných znalostí.

Pokud je znalost chápána jako objekt, znalostní management se zabývá především manipulací s tímto objektem. Hlavním problémem, který znalostní management řeší, je budování a následná správa znalostí. Jedná se tedy

primárně o vytváření, uchování a zpřístupnění znalostí. Tyto operace poměrně přesně odpovídají pro informační systémy typickým CRUD operacím.

Vezmu-li pohled procesní, tedy že znalost je aplikací nějaké odbornosti, znalostní management se bude soustředit na toky znalostí mezi jedinci. Především pak na proces vzniku, sdílení a distribuce znalostí. Nároky na informační systémy budou spočívat především ve spojování jedinců, aktivaci znalostních toků a jejich prohlubování a rozšiřování mezi další jedince.

Dalším pohledem je přístup ke znalostem. V takovém případě se znalostní management bude snažit o umožnění co nejsnazšího vyhledání a přístupu ke znalostem. Právě efektivní vyhledávání relevantních informací bude největší výzvou pro informační systém.

Posledním pohledem je způsobilost k další akci. Znalostní management se soustředí na budování základních kompetencí a pochopení strategického *know-how* organizace. Informační systém by měl podporovat rozvoj individuálních i organizačních kompetencí a posilovat tak celkový intelektuální kapitál organizace. [2]

2.4 Znalostní management

Již nyní by čtenáři mělo být jasné, co je to znalost a jak se dělí. Právě seznámení s pojmem znalosti a jejího dělení umožní uvést čtenáře do znalostního managementu, jeho definice, historie a modelů.

2.4.1 Definice

Na přelomu tisíciletí se o definici znalostního managementu snažilo mnoho autorů z akademické i firemní sféry. Žádná z nich se však neujala jako obecná definice znalostního managementu jako nové manažerské disciplíny.

V roce 2007 Bureš analyzoval přes 20 definic znalostního managementu, vybral jejich společné znaky a následně položil novou definici, která vychází ze závěru sebraných definic. Mnoho definic obsahovalo popis znalostního managementu jako procesu či jako sady specifických procesů. Dalším společným znakem byla spojitost znalostního managementu se strategií a cíli organizace. Podobně se mnoho definic snažilo orientovat na cíle znalostního managementu, kterými mohou být například zvýšená konkurenceschopnost, produktivita, efektivita nebo míra inovace. Dále některé definice poukazovaly na nutnost vazby s organizačními procesy. Jiné definice myslely více na intelektuální kapitál organizace a práci s ním. [13]

Většina definic však nespojovala znalostní management s informačními a komunikačními technologiemi, což se zdálo být zajímavým zjištěním, neboť mnoho organizací se při zavádění znalostního managementu soustředí právě na vytvoření znalostní databáze a dalších podpůrných systémů. Bureš dále uvádí svoji definici, kterou zde pro úplnost uvádím také:

Znalostní management je způsob manažerského přístupu k vedení organizace, tvorby prostředí v organizaci a dosahování podnikových cílů, spočívající ve sladění organizačních procesů s procesy znalostními, a to prostřednictvím znalostních zdrojů a pomocí vhodných metod, technik a nástrojů. [13]

Ve vysvětlení definice pak uvádí termín *znalostní organizace*, který vlastně vysvětluje nejen celou definici, ale je i hlavním cílem moderního znalostního managementu. Tedy organizace, která extenzivně využívá znalosti a znalostní procesy s nimi související k dosažení stanovených cílů organizace.

2.4.2 Historie

Jak již bylo naznačeno v kapitole 2.1.3, znalostmi se zabývali již řeční filosofové. Na jejich práci navazovali západní filosofové během středověku. Dá se argumentovat, že organizováním vlastních znalostí se zabýval prakticky každý tvor již od počátku věků. Takové organizování však nikdy nebylo systematické.

Cestu pro moderní znalostní management otevřel v průběhu 60. let 20. století právě Michael Polanyi. Samotný termín znalostní management a celkové zaměření na řízení explicitních znalostí uvnitř organizace přišel až v 80. letech 20. století. A velkou roli v této transformaci sehrály informační technologie a jejich rozmach. Konkurenční výhodu tehdy tvořil především kvalitní produkt a služby za co nejnižší cenu. Díky informačním technologiím bylo možné kontrolovat výrobu, logistiku nebo marketing v mnohem větším detailu. Informační technologie rovněž přinesly větší množství komunikace, dat a v důsledku i znalostí, které bylo možné analyzovat a uchovávat. Hlavním cílem logicky bylo zvýšení konkurenceschopnosti organizace a efektivity práce.

V roce 1989 proběhl průzkum mezi firmami z žebříčku *Fortune 50* (žebříček 50 nejbohatších firem světa). V průzkumu se CEO všech firem shodli, že znalosti jsou nejdůležitějším majetkem jejich firem, avšak nikdo neví, jak je řídit. I proto se některé poradenské firmy začaly interně řízením znalostí zabývat. Ve stejném roce dokonce Price Waterhouse (dnes PricewaterhouseCoopers) zakomponovala znalostní management do své strategie. [14]

Brzo se tak znalostní management stal seriózním vědním oborem. Vědci začali zkoumat odkud se znalosti berou, kudy v organizacích tečou a kdo z nich nejvíce benefituje. V průběhu 90. let tak bylo o znalostním managementu napsáno mnoho, většina se však soustředila na explicitní znalosti a ty tacitní byly často opomíjeny. Ve firemní sféře stále většina organizací o znalostním managementu ještě neslyšela nebo čekala na první výsledky. Jen málo firem se během 90. let do implementace znalostního managementu pustilo. [14]

Později začalo vznikat mnoho nových pohledů na znalostní management. Hlavním hybatelem byly obory, do kterých zasahuje, jako jsou například: 1. antropologie, 2. filosofie, 3. informatika, 4. lingvistika, 5. neurověda nebo 6. psychologie. Každý takový obor přinesl trochu jiný pohled. Často tak docházelo

až k babylonskému syndromu, neboť každý vědní obor začal používat trochu jinou terminologii a názvosloví. [13] [15]

Mimo toto zmatení však znalostní management dělal velké pokroky, o což se zasloužili především již zmínění Nonaka a Takeuchi a také Alavi a Leidner. Mnoho organizací se postupně, za pomoci konzultačních firem, pustilo do implementace znalostního managementu. Avšak dodnes můžeme nalézt mnoho především menších organizací, které stále znalostní management neřeší nebo ho ještě vyřešený nemají.

2.5 Modely znalostních procesů

V sekci 2.4.1 jsem představil definici znalostní organizace jako jednoho z výsledků úspěšného znalostního managementu. Jednou z hlavních myšlenek znalostní organizace bylo využití znalostních procesů. Tato kapitola má za úkol představit modely znalostních procesů. V literatuře je možné najít i označení *frameworky* znalostního managementu.

Modely mohou být normativní, popisné anebo kombinace obou. Normativní modely jmenují aktivity a procesy, kterým je vhodné se v rámci znalostního managementu věnovat. Nediskutují však návody, jak tyto aktivity provádět. Naproti tomu popisné modely popisují znalostní management a jeho důležité atributy, které ovlivňují úspěch či neúspěch implementace znalostních procesů.

Většina modelů znalostních procesů je normativní. Jsou tedy orientované především na aktivity, které by v rámci znalostního managementu měly být vykonávány. Slouží proto jako prvotní vodítko pro implementaci znalostního managementu. Je dobré si všimnout, že žádný z modelů nevyžaduje explicitní zavedení informačního systému, jak si často mnoho organizací mylně myslí. [16]

V roce 1998 DiBella a Nevis ve své knize zkoumali, jak se organizace učí. V rámci zkoumání navrhli asi nejjednodušší model znalostních procesů. Ten se skládá ze 3 částí:

1. získání znalostí,
2. rozšíření znalostí a
3. použití znalostí. [17]

Zde je vhodné podotknout, že všechny procesy se přirozeně periodicky opakují. Bohužel tento model má mnoho chyb a může tak sloužit spíše pro další analýzu než pro jeho aplikaci v organizaci.

Marquardt navrhl rozšířený model znalostních procesů:

1. získání znalostí,
2. tvorba znalostí,

3. přenos a použití znalostí a

4. uchování znalostí.

Tento model opět obsahuje některé chyby. První z nich je, že tvorba znalostí je pouze jedním způsobem, jak je možné znalosti získat. Kvůli prvnímu procesu je druhý proces spíše redundantní. Naproti tomu Marquardt přidává proces přenosu a uchování, které jsou pro znalostní management zásadní. [13]

Wiig rovněž navrhl čtyř-procesový model. Ten obsahuje

1. vytvoření nebo získání znalostí,
2. kompilace a transformace znalostí,
3. rozšíření a aplikace znalostí a
4. ocenění zdroje. [16]

První znalostní proces naznačuje možnost akvizice znalostí více způsoby – vytvořením nebo získáním z interních či externích zdrojů. Oproti předchozím pasivním modelům byla přidána transformace znalostí, což naznačuje i aktivní práci se znalostí.

Rugglese představil další model o 3 procesech, ty jsou však rozděleny na další subprocesy:

1. generování znalostí, které se dále dělí na
 - a) vytvoření,
 - b) získání,
 - c) syntézu,
 - d) směšování a
 - e) adaptaci,

2. kodifikace znalostí, která se dále dělí na

- a) zachycení znalostí a
- b) reprezentaci, a

3. přenos znalostí, který se nijak dále nedělí. [13] [16]

O'Dell se dostává na sedm základních procesů, přičemž přidává především

1. identifikaci znalostí,

kteřá v předchozích modelech zásadně chybí. Je důležité poznamenat, že identifikace znalostí je klíčovým procesem, který by měl logicky stát na začátku všech znalostních procesů. Dalšími procesy jsou

2. ÚVOD DO MANAGEMENTU ZNALOSTÍ

2. sběr znalostí,
3. adaptace znalostí,
4. organizace znalostí,
5. aplikace znalostí,
6. sdílení znalostí a
7. tvorba nových znalostí. [13] [18]

Beckman a Liebowitz rozšiřují model na osm kroků, kterými jsou

1. identifikace znalostí (určení základních kompetencí, tvorba strategie vytěžování zdrojů a identifikace znalostních domén),
2. zachycení znalostí (formalizace existujících znalostí),
3. výběr znalostí (určení relevance znalostí, hodnoty, přesnosti a vyřešení znalostních konfliktů),
4. uchování znalostí (reprezentace firemních znalostí, typicky ve znalostní databázi),
5. sdílení znalostí (automatická distribuce znalostí mezi uživatele, spolupráce na znalostních procesech ve virtuálních týmech),
6. aplikace znalostí (použití znalostí při rozhodování, řešení problémů, automatizaci, podpůrných činnostech, pracovního poradenství či vzdělávání),
7. tvorba znalostí (objev nových znalostí zkoumáním, experimentováním nebo kreativním myšlením) a
8. prodej (vývoj a prodej nových znalostních produktů a služeb).

Tento model přináší především přesné rozepsání činností, které je v rámci procesu třeba řešit. Zcela novým procesem je prodej, jehož primárním cílem není prodávat znalosti, ale využít znalosti ke zdokonalení nabízených produktů a služeb, které jsou prodávány. K termínu znalostní organizace je přidán ještě znalostní produkt nebo služba. [13] [16]

Mimo modely akademické bych zde rád uvedl i modely vyvinuté některými organizacemi. Například konzultační společnost Ernst&Young uvádí jednoduchý model o čtyřech procesech:

1. výroba znalostí,
2. reprezentace znalostí,
3. kodifikace znalostí a

4. aplikace znalostí.

Jedná se o částečně zjednodušený Rugglesův model. Avšak v Rugglesově modelu není explicitně zmíněna aplikace znalostí. [16]

PricewaterhouseCoopers přišla v roce 1997 na konferenci AAAI Spring Symposium: Artificial Intelligence in Knowledge Management s modelem 5F:

1. nalezení znalostí (*find*),
2. vyfiltrování relevantních znalostí (*filter*),
3. uzpůsobení znalostí řešenému problému (*format*),
4. dostání informací ke správným lidem (*forward*) a
5. zpětná vazba (*feedback*).

Model je zajímavý především filtrováním znalostí. Nalezení a vyfiltrování správných znalostí je zásadní, leč často složitý úkol. Mimo model představuje i koncept zpětné vazby. Ta pomáhá vylepšovat samotné znalosti, ale i možnosti jejich nalezení. [16] [19]

Po prostudování všech zde prezentovaných modelů by čtenáři mělo být jasné, že velká část modelů je nominativních. Rovněž je možné si snadno povšimnout, že modely se soustředí na životní cyklus znalosti v organizaci. Krátkou analýzou je možné zjistit, že většina modelů obsahuje

1. získání znalosti,
2. organizace znalosti,
3. sdílení znalosti a
4. aplikaci znalosti.

Dále uvedu i některé popisné modely. Ačkoli jsou výhodné, neboť popisují zásadní atributy, nemusí být vždy univerzálně aplikovatelné a je nutné je vždy přizpůsobit konkrétní situaci. Jako první příklad prezentuji model, který vyvinuli Holsapple a Joshi. Jedná se hybridní model – tedy spojení popisného a nominativního. Skládá se z 5 atributů:

1. manažerské vlivy (vedení, koordinace, kontrola a měření),
2. vlivy zdrojů (lidské, znalostní, finanční, materiální),
3. vlivy prostředí (trh, konkurence, technologie, čas)
4. aktivity (získávání, výběr, internalizace a použití) a
5. učení a projekce jako výsledek znalostního managementu. [16]

Aktuálně je jedním z nejsrozumitelnějších modelů. Také jako jeden z mála modelů definuje i výsledek znalostního managementu, kterým v tomto případě je učení a projekce.

Hybridní a popisné modely často berou v potaz kulturní faktory, které podnik ovlivňují, a strategické záměry celého podniku. Strategie je důležitou součástí každého podniku a každý obor managementu by se jí měl řídit. Dále je vhodné v modelu brát zřetel i na zpětnou vazbu, která může být opomíjena, avšak poskytuje zásadní informace o fungování znalostního managementu.

Druhý, tentokrát čistě popisný model, který zde uvedu, byl vyvinut v roce 1997 firmou Arthur Andersen Consulting. Po přejmenování v roce 2001 je dnes firma známá jako Accenture a je jednou z největších konzultačních firem na světě. Model se skládá z pěti kroků, kterými jsou:

1. vyhodnocení,
2. definování role znalostí v organizaci,
3. vytvoření strategie znalostního managementu, která je v souladu se strategií podniku,
4. identifikace procesů, kultury a technologií nutných pro implementaci strategie znalostního managementu a
5. zpětná vazba a její využití k vylepšení znalostních procesů.

Model je zajímavý hned prvním krokem, který zahrnuje vyhodnocení aktuální situace, aby bylo jasné, na jakých základech implementace může stavět. Dále pak model bere v potaz již zmíněnou strategii podniku, kulturní a technologické aspekty a implementuje i zpětnou vazbu.

2.6 Strategie knowledge managementu

V předchozí sekci 2.5 jsem se věnoval modelům znalostních procesů. Pro některé popisné modely je typická tvorba strategie znalostního managementu. Obecně je tvorba strategie nutná pro každou organizaci, neboť právě strategie zajišťuje, že se věci v organizaci nedějí náhodně, ale mají daný řád a směr. Každý obor managementu by pak měl mít vlastní strategii, která je v souladu se strategií celého podniku.

Možné strategie pro znalostní managementu rozvedl v roce 1997 Karl Martin Wiig, který následoval práci ekonoma Paula Romera. Ten se v roce 2018 stal laureátem Nobelovy ceny za ekonomii a právě jeho publikace na téma *economics of ideas* byla jedním z důvodů tohoto ocenění. *Economics of ideas* představuje myšlenku prakticky neomezeného ekonomického růstu, který vychází z inovací a tvorby znalostních produktů. Na rozdíl od tradičního pojetí, které předpokládá, že možnosti expanze jsou omezené. Tato omezení vychází z nedostatku fyzických zdrojů, kapitálu nebo pracovní síly. [14]

Economics of ideas dále ukazuje, jak kritickou roli v celém procesu hrají znalosti. Pomocí dobře distribuovaných znalostí mezi lidi (v případě organizace zaměstnance) je možné dosáhnout již zmíněných inovací, které tvoří produkty a služby nejvyšších kvalit (znalostní produkty). Myšlenku znalostních produktů do svého modelu zakomponoval i Beckman a Liebowitz, což bylo čtenáři prezentováno v sekci 2.5. [14]

Právě *economics of ideas* ukazuje, jak moc je znalostní management důležitý a proč se s ním většina organizací začala zabývat. A také proč organizace, které ještě nezačaly, brzo narazí na problémy.

Ovlivněn touto myšlenkou Wiig sestavil seznam několika strategií, které organizace typicky nastavují pro znalostní management. Tyto strategie a jejich cíle je možné využít při zavádění znalostních procesů použít. Na základě strategie je pak nutné nastavit cíle, kterých má znalostní management dosáhnout. [14]

2.6.1 Hlavní strategie podniku

Tato strategie říká, že strategie znalostního managementu je i strategií celé organizace. Ta se poté soustředí na vytváření, zachycení, organizaci, obnovu, sdílení a používání znalostí ve všech plánech a operacích. Cíl je získání nejlepších možných znalostí a jejich následné použití ve všech akcích podniku.

2.6.2 Strategie duševního vlastnictví

Strategie se zaměřuje na řízení duševních aktiv na podnikové úrovni. Tedy například patenty, technologie, provozní a řídicí postupy, vztahy se zákazníky a další aktiva. Management by se měl zaměřit na organizaci, úschovu, obnovu a zvýšení dostupnosti znalostí. To vše by mělo vyústit v tvorbu lepších produktů, které mohou být uvedeny na trh.

2.6.3 Strategie osobní zodpovědnosti

Strategie má za úkol zdůraznit osobní odpovědnost za investice, inovace a udržení konkurenceschopnosti. Dále se zaměřuje na efektivní využití a dostupnost znalostí. Zaměstnanci by se tedy měli cítit osobně zodpovědní za uchování a sdílení znalostí, které pro organizaci vytvoří. Výsledkem bude zvýšení konkurenceschopnosti.

2.6.4 Strategie pro tvorbu znalostí

Strategie se zaměřuje na organizační učení, základní a aplikovaný výzkum. Strategie má za cíl motivovat zaměstnance k inovacím. Důležité je také získané poznatky uchovat a získat z nich nové a lepší znalosti, které povedou ke zvýšení konkurenceschopnosti.

2.6.5 Strategie přenosu znalostí

Hlavním úkolem, který strategie určuje je systematický přístup k přenosu znalostí (získávání, organizování, restrukturalizace, skladování a distribuce) na místa, kde s nimi bude vykonána užitečná práce. Strategie obsahuje i sdílení znalostí a osvojování osvědčených postupů.

2.7 Přínosy znalostního managementu

Čtenář by již měl být obeznámen s tím, co znalostní management je, jak se k němu dá přistupovat a jaké strategické cíle může plnit. Podobně by asi mělo být jasné, že zavedení znalostního managementu není samoúčelné. Vždy sleduje vytyčený cíl a má organizaci přinést nějaké benefity. Tato sekce prezentuje, jaké přínosy a benefity pro organizaci může představovat dobře zvládnutý znalostní management.

První výzkum, který se zaměřil na aktuální situaci v oblasti znalostního managementu, provedly již v roce 1999 Alavi a Leidner. V rámci výzkumu se dotazovaly i na benefity, které zavedení znalostního managementu přineslo. Systémy byly navrženy, aby přinesly procesní i organizační vylepšení.

Mezi procesní patří komunikace (zvýšená a rychlejší komunikace, zvýšené zapojení zaměstnanců a větší prezentace jejich názorů) a efektivita (rychlejší řešení problémů, rychlejší výsledky a dodávka produktů na trh). Ty organizační pak zahrnují především finance (zvýšený prodej, snížená cena a vyšší ziskovost), marketing (lepší produkt nebo služba, větší zaměření na zákazníka a cílený marketing) a všeobecné vylepšení (zlepšený projektový management a snížení počtu zaměstnanců). [20]

Bureš rozděluje možné přínosy do dvou kategorií. Jedna z nich jsou výsledky se vztahem k znalostnímu managementu. Tyto výsledky jsou přímo spojené s realizací znalostních procesů. Jako hlavní přínosy v této kategorii uvádí

1. zlepšení sdílení znalostí a kooperace mezi zaměstnanci organizace,
2. rozšíření nejlepších pracovních praktik v organizaci,
3. zlepšení učení se a integrace nových zaměstnanců,
4. eliminace ztráty know-how,
5. zlepšení kvality projektů a inovace,
6. zlepšení vztahů s externím prostředím a příprava na jeho vývoj a
7. zlepšení krizového managementu a reakcí na neočekávané události. [13]

Druhá kategorie jsou výsledky se vztahem k hlavnímu předmětu podnikání organizace (*core business*). Zde jsou přínosy stejné jako u každé inovace a odpovídají každému podnikatelskému záměru:

1. zvýšení výkonnosti organizace,
2. zvýšení zisku,
3. maximalizace tržeb,
4. snížení nákladů,
5. zavedení nových způsobů práce a
6. vytvoření nových příležitostí na trhu. [13]

Je možné najít mnoho případů organizací, které po implementaci znalostního managementu skutečně některý z přínosů pocítily. Společnost British Petrol díky zavedení týmových videokonferencí zvýšila rychlost řešení kritických operačních problémů. Společnost Texas Instruments díky sdílení nejlepších praktik mezi pobočkami ušetřila prostředky ekvivalentní investici do nové pobočky. Společnost Hewlett-Packard zrychlila uvedení nových produktů na trh díky sdílení expertních znalostí, které pro vývojové týmy byly dříve nedostupné. Společnosti Skandia, díky zavedení nových metrik pro intelektuální kapitál a zaměření manažerů na navyšování jeho hodnoty, rostly tržby nejrychleji v celém oboru. Společnost Dow Chemical díky lepší správě intelektuálního kapitálu, zvýšila roční příjmy z licencí o 100 milionů dolarů. Společnost Booz-Allen & Hamilton díky snížení času potřebného k nalezení zodpovědného zaměstnance nebo potřebné informace, ušetřila přes 7 milionů dolarů ročně. [13]

Klaus North a jeho kolegové provedli výzkum založený na analýze 48 společností. Společnosti byly členy Německé asociace pro znalostní management nebo Švýcarského fóra pro znalostní management. K výzkumu využili analytickou metodu Balanced scorecard, která tvoří vazbu mezi strategií a operativními činnostmi organizace. Měření výkonnosti probíhá pomocí čtyř perspektiv:

1. finance (finanční výsledky a efektivita využívání zdrojů),
2. zákazníci (spokojenost, udržení zákazníků, akvizice nových zákazníků),
3. vnitřní procesy (efektivita a kvalita),
4. učení a růst (vzdělávání zaměstnanců).

Pro účely výzkumu však North metodu upravil a perspektivu učení a růst rozdělil na dvě samostatné perspektivy 1. inovace a 2. zaměstnanci. V rámci dotazování respondenti vytvořili přibližně 240 výroků o přínosech znalostního

managementu pro organizaci, ve které pracují. Tato tvrzení byla následně rozdělena do perspektiv Balanced scorecard.

Sledované organizace realizovaly zlepšení především v oblastech vnitřních procesů a výkonu zaměstnanců. Naopak nejmenší zlepšení bylo vidět v oblasti financí a inovací. Autoři dále zkoumali i jednotlivé faktory v každé perspektivě, což přináší ještě větší náhled do zlepšení, kterých organizace dosáhly. V oblasti vnitřních procesů organizace zrychlily interní procesy a nově využívaly organizační znalosti. Při zkoumání perspektivy zaměstnanci autoři zjistili, že největší zlepšení bylo realizováno ve zkrácení doby potřebné pro adaptaci nových zaměstnanců, zvýšení úrovně znalostí jednotlivých zaměstnanců a zvýšení jejich motivace.

Ve vztahu k zákazníkům bylo nejvýznamnější zvýšení kvality výrobků a služeb. Jak již bylo řečeno, nejslabší výsledky se dostavily v oblasti financí. Jedná se pouze o 7 % všech měřitelných přínosů. Konkrétními přínosy jsou zvýšení obrátu, snížení nákladů na administrativu a zlepšení řízení rizik.

Důležité je i podotknout, že výzkum byl lokální, jelikož zapojené organizace byly pouze z Německa a Švýcarska. Znalostní management je vysoce ovlivňován strategií organizace, jejími cíli a stylem řízení. V jiných kulturních podmínkách či v organizacích s jiným stylem řízení a motivací by výzkum mohl dopadnout odlišně. Rovněž je nutné dodat, že přínosy znalostního managementu jsou dlouhodobé a typicky nemají operativní charakter. [13]

2.8 Znalostní pracovník

V sekci 2.4.1 jsem čtenáře seznámil s definicí znalostní organizace a v sekcích 2.5 a 2.7 jsem se věnoval znalostním produktům a službám. Právě znalostní organizace a znalostní produkty jsou závislé na znalostních pracovnících. Jako první termín znalostní pracovník (*knowledge worker*) představil Petr Drucker, který tento termín dával do kontrastu manuálního pracovníka.

Manuální pracovník existoval v každé společnosti od jejího počátku. Ne však každý se zabýval jeho produktivitou. Bylo prakticky axiomatické, že pracovník mohl vyprodukovat více pouze pokud pracoval více hodin. Jestli existovala záležitost, na které se ekonomové 19. století shodli, byl to právě fakt, že existují rozdíly mezi dovednostmi pracovníků, avšak neexistuje jiný rozdíl v jejich produktivitě, než že někteří jsou líní a jiní pracovití. [21] [22]

První, kdo se produktivitou pracovníků skutečně začal zabývat, byl Frederick Winslow Taylor ve 20. století. Od té doby se produktivita manuálních pracovníků zvyšovala průměrně o 3 % každý rok. Drucker argumentuje, že právě zvyšování produktivity manuálních pracovníků byl hnací motor ekonomického růstu a všech sociálních výtoky 20. století. Vždyť během 20. století se produktivita zvýšila padesátkrát, což lze považovat za největší úspěch managementu ve 20. století. Za největší výzvu pro 21. století Drucker předurčil zvýšení produktivity znalostních pracovníků. [22]

Asi nejdůležitější charakteristika znalostního pracovníka je fakt, že alespoň některé z jeho vstupů nebo výstupů jsou nehmotné povahy. Od ostatních forem práce se odlišuje především prováděním úkolů nerutinní povahy nebo řešením problémů, které jsou složité, netriviální a vyžadují specifické znalosti či dovednosti. Znalostní pracovník však neznamená to samé, co znalec nebo expert. Za experty či znalce je možné považovat špičkové vědce, lékaře, myslitele nebo manažery, pojem znalostní pracovník je mnohem širší. V tomto případě je znalostní pracovník nadmnožinou znalce. [13] [22]

Práce znalostních pracovníků (často též nazývaná znalostní práce či *knowledge work*) se od té klasické liší v mnoha aspektech. Prvním je cíl práce, kterým je řešení složitých problémů v komplexních doménách. Náplní práce je kreativní činnost, při které je nutné aplikovat znalosti. Vstupy a výstupy jsou založeny na informacích a datech. Při práci tak znalostní pracovník vytváří nové poznatky, interpretuje již objevené znalosti a transformuje je do jiných oblastí a kontextů. Na rozdíl od manuálního pracovníka znalostní pracovník využívá primárně své intelektuální schopnosti a znalosti, nikoli fyzické schopnosti. Je tedy očekávána vysoká úroveň vzdělání a proškolenosti. [22] [23]

Nonaka v roce 1991 popsal znalosti jako hnací motor inovací a podpořil tak tezi Druckera. Vyjádřil však obavy, že manažeři nechápu využití znalostí, neboť organizace jsou jako živé organizmy podobné jednotlivcům, nikoli výrobní stroje. Tvrdil, že mnoho manažerů vidí znalosti stále jako statický vstup do výrobních strojů. Opravdový prostor pro změnu viděl právě ve znalostních pracovnících a jejich přínosu. [24]

2.8.1 Produktivita

Ve zkoumání znalostních pracovníků a jejich produktivity je naše civilizace ještě na začátku. Ve smyslu produktivity manuálních pracovníků jsme s produktivitou znalostních pracovníků na začátku 20. století. Drucker však již na začátku 21. století identifikoval 6 hlavních faktorů, které určují produktivitu znalostních pracovníků.

1. Produktivita znalostního pracovníka vyžaduje, abychom se ptali na otázku „Co je mým úkolem?“
2. Je žádoucí, aby znalostní pracovníci měli autonomii. S tou však souvisí i zodpovědnost za produktivitu a řízení svého výkonu.
3. Inovace musí být součástí práce, úkolů a zodpovědnosti znalostního pracovníka.
4. Znalostní práce vyžaduje neustálé učení, ale i vyučování na straně znalostního pracovníka.

5. Kvalita výstupů znalostního pracovníka je minimálně stejně důležitá jako kvantita.
6. Znalostní pracovník musí být považován za aktivum, nikoli za náklad. Rozdíl je, že náklady je nutné minimalizovat, kdežto aktiva je nutné řízeně posilovat. Je důležité, aby pracovník chtěl v organizaci pracovat, a to i ve srovnání se všemi ostatními příležitostmi.

S výjimkou posledního jsou všechny faktory skoro přesným opakem požadavků na zvýšení produktivity manuálních pracovníků. Dalo by se argumentovat, že důležitost posledního bodu si uvědomil už Henry Ford v roce 1914, kdy skokově zvýšil mzdu svých kvalifikovaných zaměstnanců z \$0.8 na \$5. Jeho motivace však byla pouze snížit operativní náklady, nikoli budovat intelektuální kapitál jako aktivum firmy. Neboť do té doby, aby udržel ve svých fabrikách 10 000 zaměstnanců, musel jich ročně najmout 60 000. Zvýšení mezd se tedy snažil snížit náklady na výběr a zaučení nových zaměstnanců. [22] [13]

Na rozdíl od manuálních pracovníků je pro znalostní pracovníky kvalita nejdůležitějším kritériem. Drucker argumentuje příkladem učitele: Obvykle společnost nezajímá, kolik studentů učitel vyučuje, ale především jak kvalitní výuka je. Produktivita manuálních pracovníků je závislá na kvantitě s často určeným minimálním standardem pro kvalitu. Naopak produktivita znalostních pracovníků cílí na optimální, či maximální kvalitu.

2.8.2 Metody měření produktivity

Již jsem vysvětlil pojem znalostní pracovník a pojem produktivity znalostního pracovníka. Nyní však ještě zbývá vysvětlit, jak se produktivita znalostních pracovníků měří.

2.8.2.1 Metoda funkčních bodů

Metoda funkčních bodů (též *Function point analysis*) je analytická metoda, která se využívá primárně pro měření produktivity při vývoji software. S metodou přišel v roce 1979 Allan J. Albrecht pro IBM. Metoda spočívá v posuzování funkcionality, kterou software obsahuje nebo by dle požadavků obsahovat měl. Funkční body jsou pak jednotkami, které je možné hodnotit. Ty jsou složeny ze vstupů a výstupů nebo dotazů. Na rozdíl od jiných metod nejsou hodnoceny fyzické aspekty díla (například počet řádek kódu).

Produktivita je pak typicky měřena jako poměr funkčních bodů (FPs) a strávených dnů práce (MDs). Funkční body jsou funkčně orientovaná metrika, která by měla být nezávislá na programovacím jazyku. Tato nezávislost však byla zpochybněna, neboť implementace stejné funkcionality v různých programovacích jazycích je zpravidla různě obtížná. Dále pak je metoda velmi subjektivní a velmi závisí na zkušenostech a dovednostech analytika, který jednotlivé funkcionality oceňuje. [25] [26]

2.8.2.2 Data envelopment analysis

Data envelopment analysis je neparametrická metoda pro měření relativní produktivity jedince oproti populaci. Například tedy měření produktivity jednoho znalostního pracovníka vůči celému týmu. Pro měření využívá rozhodovací jednotky (*Decision making units*). Metoda porovnává objem vstupních a výstupních dat jedné rozhodovací jednotky oproti všem rozhodovacím jednotkám ve stejném prostředí. [26]

2.8.2.3 Provozní funkční analýza

Provozní funkční analýza (též *Operational functional analysis*) je metoda, která je založena na určení poměru mezi poptávkou po produktu a počtem zakázek, které pracovník obslouží. Poptávka přímo určuje kvantitu, kterou je nutné vyprodukovat. Výsledek je tedy možné spočítat jako poměr vyřízených objednávek vůči času, který na jejich vyřízení pracovník potřeboval. [26]

2.8.2.4 Multi-minute measurement

Multi-minute measurement je relativně jednoduchá metoda pro vývoj standardizovaných měření práce znalostních pracovníků. Metoda spočívá v pozorování práce v předem stanovené časy, nikoli náhodně, jak tomu někdy bývá. Metodu je tak možné shrnout v následujících 4 krocích:

1. vytvoření seznamu činností a jejich dohledu,
2. nastavení kódů pro jakoukoli zvláštní klasifikaci, která se má použít,
3. vypracování postupů reportingu zaměstnanců a
4. výběr měřicího intervalu a příprava formulářů pro reporting. [26]

2.8.2.5 Metoda splněných cílů

Metoda splněných cílů jednoduše měří poměr mezi vytyčenými cíli a splněnými cíli. Výsledná produktivita tak může být například počet splněných úkolů vůči všem přiřazeným úkolům. Metoda jako taková nebere v potaz kvalitu, což není typické pro metody měření produktivity znalostních pracovníků. Avšak měření kvality může být zapracováno do evaluace splnění úkolu. [26]

2.8.2.6 Procento činností s přidanou hodnotou

Další zajímavou metodou je procento činností s přidanou hodnotou (*Percentage of time spent in value-added activity*). Jak vyplývá z názvu, je nutné činnosti rozdělit na ty, které pomáhají dosáhnout cílů (tedy těch s přidanou hodnotou) a činností, které jsou podpůrné nebo cílů nedosahují. Produktivita

je pak měřena poměrem mezi časem stráveným činnostmi s přidanou hodnotou a celkovým pracovním časem. První problém metody spočívá ve faktu, že nijak nebere v potaz kvalitu výstupu. Druhý problém je, že metoda neříká, zda by práce mohla být vykonána lépe nebo rychleji. V roce 1993 použili tuto metodu Clark a Wheelwright ke studii dopadu počtu projektů přiřazených jednomu pracovníkovi. [26]

2.8.2.7 Vzájemné hodnocení

Vzájemné hodnocení (též *Peer evaluation*) je metoda, která je pro vyhodnocování produktivity používána relativně často. Je založena na myšlence, že kolegové lépe dokážou odhadnout všechny aspekty práce (včetně nedefinovatelných aspektů) a dokáží zhodnotit produktivitu ostatních vůči té své. V rámci hodnocení mohou brát v potaz i kvalitu výstupu. Metoda nedovoluje produktivitu měřit, pouze subjektivně vyhodnocovat. Je tedy nutné předem nastavit některá kritéria a obory hodnocení. [26]

2.8.3 Zhodnocení měření produktivity

Je možné si všimnout, že největším neduhem všech metod je nejasnost, jakým způsobem měřit výsledek. Většina metod prezentovaných v sekci 2.8.2 je použitelná i pro měření produktivity manuálních pracovníků. Jenže u manuálních pracovníků je často velmi snadné změřit výstup, neboť proces, výstup a kvalita výstupu mohou být velmi dobře standardizovány. Pro znalostní pracovníky žádný podobný standard neexistuje. Hlavním důvodem je, že zatím neznáme spolehlivý postup, jak kvantitativně změřit kvalitu znalostního výstupu. Bude-li nalezena vhodná metrika na měření kvality výstupu, všechny výše zmíněné metody bude možné instantně s touto metrikou použít.

2.9 Kritika znalostního managementu

Znalostnímu managementu se během jeho vývoje dostalo i mnoho kritiky. Truneček již v roce 2004 kritiku rozdělil do dvou skupin. Do první řadí kritiky bez znalosti podstaty, kteří mají o znalostním managementu jen malé povědomí. Jejich názor obvykle vyjadřuje, že znalostní management je módní záležitost, kterou prosazují především velké konzultační společnosti a nemá cenu se jím zabývat. Druhá skupina jsou autoři, kteří znalostnímu managementu rozumí a cíleně adresují jeho vlastní problémy. Dále se v textu budu zabývat pouze druhou skupinou autorů. [27]

Asi nejzásadnější je otázka, zda se dají znalosti řídit a zda není vhodnější se soustředit na udržování a zlepšování prostředí, ve kterém znalosti vznikají, předávají se a využívají. Zavedení znalostního managementu toto prostředí výrazně naruší. Pokud jsou při zavádění přehlíženy předchozí kulturní hod-

noty, ve výsledku může zavedení znalostního managementu zhoršit sdílení tacitních znalostí uvnitř organizace. [23]

Wilson má na předchozí otázku velmi jasnou a odmítavou odpověď. V zásadě tvrdí, že vše uvnitř mozku, s čím je možné manipulovat, jsou data a informace. Ty je možné řídit, avšak znalosti nikoli. Dále uvádí, že lidé mají velmi malou kontrolu nad tím, co doopravdy vědí. Jako příklad uvádí informace, které se jedinec naučí a zapomene, aniž by si tento fakt uvědomoval. [28]

Dalším problémem je pohled na znalosti. Aktuálně se většina oblastí znalostního managementu (a především pak informační systémy znalostního managementu) na znalosti dívá jako na objekt. Avšak mnoho autorů se domnívá, že znalost není plně objektivizovatelná. V důsledku tohoto pohledu je znalostní management často orientován na technologie a opomíjí fakt, že znalosti vytváří a užívají lidé. [23]

Problémové je i tvrzení ze sekce 2.8, že znalostní pracovníci jsou aktivem, avšak manuální pracovníci jsou nákladem, neboť toto tvrzení zcela zřejmě odporuje ekonomické realitě. [22]

2.10 Závěr

V této kapitole jsem vysvětlil rozdíl mezi daty, informacemi a znalostmi. Dále jsem čtenáře seznámil s teorií, taxonomií znalostí a dopady různých pohledů na znalosti. Na základě této teorie jsem představil problém nedefinovanosti znalostního managementu. Uvedl jsem několik definic, které jsou pro znalostní management aktuální a z nich se pokusil vyzdvihnout zásadní myšlenky.

Dále jsem prezentoval stručnou historii znalostního managementu. Na historii jsme navázal modely, které pokrývají základní fáze znalostního managementu a které ukazují možnosti jeho implementace. Rovněž jsem ukázal hlavní přínosy, které znalostní management pro organizaci může mít. Vše jsem spojil s termíny znalostní organizace, znalostní produkt a znalostní pracovník. Vysvětlil jsem, jaký k sobě mají vztah a proč jsou pro moderní znalostní management důležité. Nakonec jsem uvedl i kritické pohledy na znalostní management a stručně ukázal jeho aktuální problémy.

Systemy znalostního managementu

V rámci kapitoly 2 jsem popsal znalostní management. V kapitole bylo řečeno, že zavedení znalostního managementu v organizaci často spočívá v zavedení informačního systému. Tyto informační systémy se obvykle nazývají *knowledge base* systémy či znalostní databáze a slouží především k třídění, ukládání a vyhledání znalostí.

Existují i systémy znalostního managementu, které jsou nadmnožinou znalostních databází. Takové systémy často slouží k podpoře všech procesů znalostního managementu. Ačkoli ne každý aspekt znalostního managementu vyžaduje implementaci informačního systému, mnoho z nich právě na informační systémy spoléhá. [2]

Aby hned na začátek nedošlo k omylu, ještě jednou je třeba zdůraznit, že zavedení znalostního managementu neznamená zavést pouze informační systém. Je nutné provést i další podpůrné kroky. A ačkoli to již v dnešní době není obvyklé, je možné zavést znalostní management bez zavedení informačního systému. [29]

V literatuře je možné se setkat i se systémy organizační paměti, anglicky *Organizational Memory Systems* (OMS). Tyto systémy slouží k přenášení dříve získaných znalostí na současné činnosti. Někdy jsou OMS považovány za synonymum pro systémy znalostního managementu, jindy jako jejich možná součást. [23]

Je vhodné uvést i existenci *e-learning* platforem, které slouží především k soustavnému vzdělávání. Shromážďují dohromady studijní materiály a podporují interakci mezi studenty a vyučujícími. Podobně jako OMS i *e-learning* platformy jsou často součástí jednoho většího systému znalostního managementu. [23]

3.1 Využití systému znalostního managementu

Alavi a Leidner uvádí 3 hlavní využití systémů znalostního managementu. Prvním je kódování a sdílení nejlepších praktik a postupů (*best practice*). Jako příklad uvádějí pojišťovací společnost, která čelila poklesu zisků. Společnost zavedla proces pro hledání nejlepších rozhodnutí, který byl podpořen informačním systémem. Společnost tak dokázala zaujmout ziskovější trhy a v důsledku i zvýšit zisk. [2]

Druhým uvedeným využitím je mapování interní odbornosti. Mnoho znalostí v organizaci stále zůstává nekodifikováno (především ve vlastnictví konkrétních zaměstnanců). V takovém případě je mapování nekodifikovaných znalostí velmi užitečné. V důsledku některé znalosti není možné najít přímo, je však možné najít jiného pracovníka, který dané znalosti má. Jako příklad uvádí průzkum z roku 1998, který uvádí, že 74 % respondentů věří, že nejlepší znalosti, které organizace vlastní, jsou nedostupné. Dále pak 68 % respondentů uvedlo, že se v jejich organizaci opakují chyby. [2]

Jako třetí využití systémů znalostního managementu uvádí tvorbu znalostních sítí. Základem pro sdílení znalostí je komunikace. Znalostní sítě však jdou ještě dál a nesoustředí se pouze na sdílení znalostí, ale i na podporu vzniku nových znalostí. Propojením expertů, kteří mezi sebou sdílejí znalosti, mohou vznikat nové kolektivní znalosti. Jako příklad uvádí společnost Chrysler, která změnila organizační členění firmy. Uvědomila si, že je podstatné, aby se experti ve stejné oblasti ale různých týmů mohli nadále potkávat a komunikovat mezi sebou. [2]

Maier a Hadrich představují více prakticky zaměřené využití systémů znalostního managementu. Jejich pohled vychází z konceptu znalostí práce, rozebrané v sekci 2.8, a využití existujících informačních systémů. Jejich využití vidí primárně v komunikaci a zpracování nestrukturovaných nebo semi-strukturovaných dat. K tomu doporučují využít systémy pro správu obsahu (CMS), systémy pro správu dokumentů (DMS) nebo *e-learning* platformy. Jako protiklad uvádějí relační databázové systémy nebo datové sklady. Se změnou systémů, které znalostní pracovníci využívají přichází i nové problémy a výzvy. Těmi největšími jsou sdílení organizačních znalostí mimo hranice organizace a vyhledání nebo získání dokumentů a zpráv v nestrukturovaných formátech. [23]

3.2 Fáze vývoje systému znalostního managementu

Woodman a Zade ve své práci představili framework pro vývoj systémů znalostního managementu. V práci argumentují, že vývoj systému znalostního managementu se liší od vývoje běžného informačního systému. Zde bych rád zdůraznil, že záleží na úhlu pohledu. Informační systémy tradičně vznikají

dvěma způsoby. První je interní, kde na informačním systému pracují firemní zaměstnanci a systém vzniká jako firemní produkt, který je možné využívat v rámci firmy nebo ho poskytnout třetím stranám. Druhý způsob je externí, při kterém je do vývoje zapojena třetí strana. Na samotné organizaci zůstává definice požadavků na systém, avšak návrh a vývoj systému je přenesen na třetí stranu. Z pohledu třetí strany je tedy vývoj informačního systému znalostního managementu zcela totožný s vývojem běžného informačního systému, neboť práce spojená s definicí požadavků zůstává na straně organizace jako zadavatele. [29]

První fází je analýza aktuálního stavu. Během ní je nutné podívat se na aktuální stav fungování organizace a identifikovat problémy. Dále je nutné definovat jasné cíle, které má implementace přinést a které všechny zúčastněné strany chápou stejným způsobem. K cílům je potřeba přidat i kvalifikační kritéria pro jejich splnění.

V druhé fázi je definován kýžený stav, do kterého by se organizace ráda dostala. Ten by měl reflektovat problémy, které byly identifikovány v první fázi. Organizace by měla primárně řešit business problémy, jejichž vyřešení přinese nové produkty a služby nebo dostatečně vylepší ty stávající. Výstupem by měla být jasně definovaná vize zlepšeného stavu. Ta musí být v souladu se strategií a vizí celé organizace.

Třetí fází je návrh systému znalostního managementu. Úkolem fáze je přenést vizi kýženého stavu do informačního systému a splnit tak stanovené cíle. K tomu je využito znalostních agentů, znalostních toků a znalostních rozhraní. Znalostní agenti jsou entity, které jsou schopné držení znalostí a interakce (zaměstnanci, dodavatelé, dokumenty atp.). Znalostní toky reprezentují přenos znalostí mezi agenty. Znalostní rozhraní jsou pak místa, kde jsou znalosti předávány mezi agenty. Rozhraní dále určují pravidla pro přenos informací mezi agenty. [29] Návrh informačního systému je však možné provést plně dle principů softwarového inženýrství.

Ve čtvrté fázi jsou zkoumány informační technologie, které je možné použít. Informační technologie by měly splňovat požadavky (organizační, sociální, technologické atp.), které jsou stanoveny návrhem ve fázi 3. Informační technologie mohou být posuzovány i na základě dalších kritérií (cena, dostupnost, složitost atp.), které si organizace stanoví. Výsledkem této fáze by mělo být rozhodnutí, zda komponenty pro systém budou nakoupeny, vytvořeny svépomocí nebo vzniknou integrací stávajících řešení uvnitř organizace.

Pátá fáze slouží pro řízení evolučního potenciálu systému. Organizace by se tedy měla ujistit, že systém je navržený správně a že se zvládne přizpůsobovat změnám v organizaci a jejím okolí. Tyto změny nemusí být čistě technické, avšak jedná se i o nové trendy. V případě identifikace problému je nutné se vrátit na začátek a postupovat opět od jeho definice. [29]

3.3 Rozdíl oproti běžnému informačnímu systému

Z přechozí sekce 3.2 by již čtenáři mělo být jasné, že vývoj systémů znalostního managementu se od vývoje jiných informačních systémů částečně liší. Tato sekce má za cíl představit všechny zásadní rozdíly.

Prvním z rozdílů je iniciativa, která v tomto případě přichází od znalostního managementu. Výsledný informační systém má sloužit cílům znalostního managementu (například zvýšení produktivity), nikoli přímo business cílům organizace. Informační systém je tak opravdu pouze podpora pro znalostní management a vzniká pouze z jeho iniciativy. [23]

Dalším rozdílem je kontext. Alavi a Leidner znalost v rámci systémů znalostního managementu charakterizují jako informaci, která je smysluplně organizovaná a zasazená do kontextu. [2] Znalost závisí na kontextu a s různými akcemi a interakcemi se znalost mění. Takové znalosti jsou obvykle tacitní a tudíž mají část, která je z principu nepřenositelná. [23]

Kontextu se více věnoval Dourish, jehož práce se dá shrnout ve čtyřech bodech: 1. kontext není pouze informace, kontext je relační vlastnost, 2. kontext je definován dynamicky, 3. kontext se mění s měnící se situací a 4. kontext je neoddělitelný od obsahu. Tyto body jsou obzvláště důležité pro princip včasnosti vysvětlený v sekci 3.3.1.3 a sociotechnický princip v sekci 3.3.1.4. [29] [30]

Na kontext také přímo navazují uživatelé nebo skupiny uživatelů (například jednotlivé týmy) systémů znalostního managementu. Ti jsou držitelé znalostí, které jsou v rámci systému udržovány a poskytovány. Systematické řízení kontextu dovoluje udržovat spojení mezi kodifikovanými znalostmi a jejich držiteli. Právě kontext poté poskytuje odpovědi, kdy znalosti vznikly, proč vznikly nebo kdo je jejich autorem.

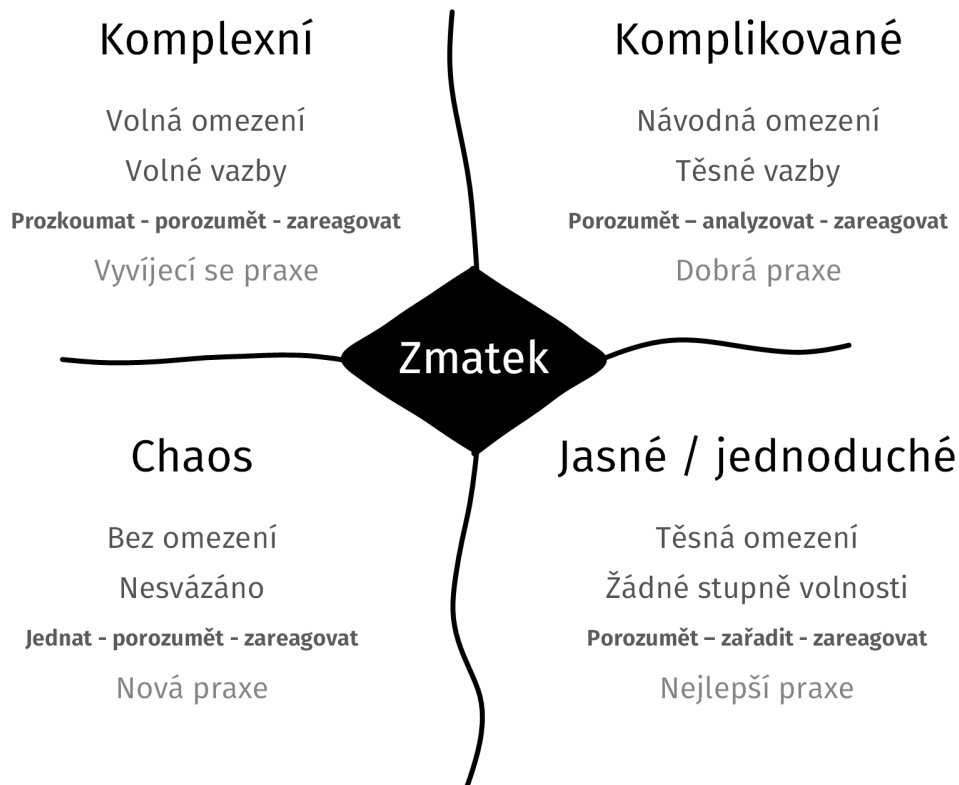
Jak bylo řešeno v úvodu této kapitoly, systémy znalostního managementu jsou vyvíjeny především proto, aby podporovaly znalostně intenzivní procesy uvnitř organizace. Jako příklad je možné uvést vznik, identifikaci, zachycení, formalizaci, přenos nebo distribuci znalostí. Maier a Hadrich právě podporu procesů uvádí jako odlišnost od běžných informačních systémů. [23] S tímto tvrzením by se však dalo polemizovat a namítnout, že role většiny informačních systémů je podpora zavedených procesů.

3.3.1 Pět principů vývoje systému znalostního managementu

Woodman a Zade dále představili i pět principů, které by měl každý systém znalostního managementu následovat.

3.3.1.1 Princip komplexnosti

Systémy znalostního managementu mají mnoho společných vlastností s komplexními adaptivními systémy. Systémy znalostního managementu se nepře-



Obrázek 31: Cynefin framework [32]

tržitě mění, jsou nepředvídatelné a jejich výstupy nejisté. Jejich stav autoři popisují i jako „na hraně chaosu“. Nepředvídatelnost je možné chápat jako neschopnost předpovědět výstup na základě vstupu, ale i jako neschopnost identifikovat a správně pochopit všechny potřebné vstupy již během vývoje systému. [29]

K ilustraci autoři používají framework *Cynefin*, s kterým přišel v roce 1999 Dave Snowden. Framework je zaměřen na podporu rozhodování a obsahuje pět domén:

1. jednoduché situace,
2. komplikované situace,
3. komplexní situace,
4. chaotické situace a
5. zmatek. [31]

Jednoduché situace jsou dobře známé problémy s jasným a očekávaným výsledkem. Pro takové problémy jsou známá řešení a nejlepší pracovní prak-

tiky. Takové problémy se řídí pravidlem „porozumět, zařadit, zareagovat“. První je tedy nutné situaci pochopit, správně jí zařadit a poté zareagovat dle jasných pravidel. Tyto situace jsou často podchyceny firemními procesy.

Komplikované situace jsou takové, kde jsou známá možná řešení, ale není znám celý problém. V takovém případě je potřeba provést analýzu problému a vybrat některé z možných řešení. Situace se dá přirovnat například ke stavbě mostu – řešení pro takové problémy jsou známá, avšak je nutné prozkoumat konkrétní situaci a následně vybrat vhodné řešení.

Hlavní rozdíl mezi komplikovanými a komplexními situacemi je ten, že pro komplexní situace není známá žádná množina možných řešení. Řídí se tak spíše schématem „prozkoumat, porozumět, zareagovat“. To často znamená provést analýzu situace, vymyslet řešení, a to vyzkoušet. Zpětně je možné vyhodnotit, zda řešení bylo vhodné či nikoli. Jako komplexní jsou popisovány například situace na trzích, v organizacích či ekosystémech.

Chaotické situace se vyznačují především velkou počáteční nejasností. Takovou situaci není možné pořádně uchopit a porozumět jí. Doporučené schéma tedy je „jednat, porozumět, zareagovat“. Tedy začít konat, snažit se najít něco jasného a stabilního a časem se situace pravděpodobně ustálí a bude dostána pod kontrolu. Jako příklad je možné uvést různé katastrofické situace, kdy není jasné, co se stalo, jak je situace rozsáhlá a co vše bude potřeba. Je lepší začít rychle jednat a postupně situaci dostat pod kontrolu.

Poslední doménou je zmatek, který by se dal zaměřovat s chaotickými situacemi, avšak zmatek se v jedné zásadní věci liší – není jasné, kam v rámci předchozích 4 domén situaci zařadit. Aktéři, kteří by tedy měli konat, se nedokážou shodnout. Řešením je rozdělit situaci na dílčí problémy, ty zařadit do jedné ze čtyř domén a konat. [31]

Ve vztahu k systémům znalostního managementu se ukázalo, že komplexní situace často není možné v rámci takových systémů postihnout. Pro komplexní situace totiž jsme schopni odhadnout vztah mezi příčinnou a následkem pouze zpětně. U chaotických situací je stav často ještě horší, neboť mezi příčinnou a následkem nelze odvodit žádný vztah. [29]

3.3.1.2 Princip celistvosti

Woodman a Zade v rámci svého výzkumu narazili na velkou provázanost mnoha problémů typicky adresovaných v první fázi jejich frameworku (například požadavek na zlepšení spolupráce je velmi provázan s požadavkem využití předchozí práce kolegů). V rámci principu celistvosti říkají, že systémy znalostního managementu by měly být analyzovány jako celky. Nikoli rozpadnuty na menší části, které jsou řešeny separátně a následně sestaveny do komplexního řešení, jako je častou praxí v softwarovém inženýrství. Tomuto principu částečně slouží druhá fáze jejich frameworku, neboť právě v té jsou požadavky zpracovány do ucelené vize budoucí zlepšené situace.

Autoři argumentují, že fungování celku je v tomto případně odlišné od fungování jednotlivých částí a přirovnávají systém s holismem z oblasti teorie komplexního myšlení. Analýza jednotlivých částí v tomto případě není dostatečná a výsledkem může být systém, který řeší jiné problémy, než ty formulované ve fázi 1. [29]

3.3.1.3 Princip včasnosti

Výzkum ukázal, že systémy znalostního managementu mají za úkol především dostat relevantní znalosti k relevantním uživatelům, a to v ten správný čas. Předem určit kdy, kdo a kde bude danou znalost potřebovat je prakticky nemožné. Systémy je tak nutné navrhovat s velkou mírou flexibility, aby bylo možné reagovat na nové požadavky. Toto zjištění je v rozporu s literaturou, která naznačuje, že systém by měl být správně analyzován a navržen ještě před začátkem implementace. Literatura totiž počítá s faktem, že již ve fázi návrhu jsou analytici schopni určit: 1. co je správná znalost, 2. kdo je držitelem znalosti, 3. kdo znalost bude potřebovat, 4. kdy ji bude potřebovat, 5. kde ji bude potřebovat a 6. v jakém formátu ji bude potřebovat. [29]

Na tomto místě je také vhodné připomenout práci Dourishe a jeho čtyři závěry: 1. kontext není pouze informace, kontext je relační vlastnost, 2. kontext je definován dynamicky, 3. kontext se mění s měnící se situací a 4. kontext je neoddělitelný od obsahu. [29] [30] Druhý a třetí výrok jsou pro princip včasnosti nejpodstatnější. Identifikace správné znalosti a její doručení na správné místo ve správný čas vysoce záleží na kontextu, který se s měnícími se situacemi mění také.

3.3.1.4 Sociotechnický princip

Sociotechnický princip říká, že systémy znalostního managementu nejsou pouze o informačních technologiích. Aby byl systém pro organizaci přínosný, již při jeho návrhu je třeba brát v potaz sociální složky organizace a její fungování. Technické prvky by z těch sociálních měly přímo vycházet. [29]

Alavi a Leidner o systémech znalostního managementu mluví jako o třídě informačních systémů a k jejich návrhu nepřístupují žádným zvláštním způsobem. [2] Tento fakt podporuje i výzkum Woodmana a Zade, který ukázal, že mnoho organizací se při implementaci systémů znalostního managementu obrací právě k informačním technologiím. Ukazují však, že takové implementace často nedopadly úspěšně. Argumentují, že návrh systémů musí být podložen organizačními procesy, chováním a kulturou, nikoli pouze informačními technologiemi. [29]

Zde se opět vrátím právě ke kontextu. Zack tvrdí, že metodiky používané pro návrh informačních systémů pro systémy znalostního managementu nejsou dostatečné. Jako hlavní důvody uvádí komplikovanost těchto systémů a také jejich vysokou závislost na kontextu. A právě proto je vhodné při návrhu

systémů znalostního managementu brát zřetel i na sociální složky organizace. [33]

3.3.1.5 Princip více pohledů

Princip více pohledů adresuje především problém různorodosti uživatelů, kteří systém využívají. Většina podnikových informačních systémů je navržena pro jednu skupinu uživatelů (např. uživatelé v rámci jednoho oddělení), která vykonává podobnou práci podobným způsobem. Z toho plyne, že mají podobné požadavky na systém.

Systémy znalostního managementu musí sloužit uživatelům napříč organizací. Uživatelům z různých týmů, s rozdílnými požadavky i znalostmi informačních technologií. Navrhovat znalostní informační systém pouze jedním pohledem nebo předem stanovenou množinou pohledů může být nejen nemožné, ale i kontraproduktivní.

Další výzvou pro systémy znalostního managementu jsou uživatelé, kteří zastávají více rolí. Neboť nemusí být jasné, z jaké role systém popisují. Nakonec je důležité i připomenout, že požadavky jednotlivých uživatelů se budou měnit v čase nejen s měnící se situací v organizaci, ale i s měnící se rolí uživatelů v rámci organizace. [29]

3.4 Závěr

V této kapitole jsem se věnoval především typům systémů, které podporují procesy znalostního managementu. Uvedl jsem jejich využití a také jak se navrhuje a vyvíjí. Framework popsany v sekci 3.2 bude zásadní pro analýzu navrhovaného informačního systému v kapitole 7. Dále jsem rozebral rozdíly systémů znalostního managementu a jiných informačních systémů. Jednotlivá doporučení pro návrh systémů znalostního managementu rozeberu v kapitole 4 následovanou kapitolou 5, která analyzuje trh existujících řešení.

Doporučení pro návrh systému

V kapitole 3 jsem se věnoval systémům znalostního managementu, odlišnostem od jiných informačních systémů a také principům návrhu. Než však začnu s návrhem nového systému, představím některá doporučení a teorii související s těmito doporučeními. Ta bude sloužit jako základ pro pochopení prezentovaného návrhu v kapitole 8.

4.1 Analýza

První lekcí softwarového inženýrství je, že není možné začít implementaci informačního systému bez předchozí analytické práce, která by měla obsahovat analýzu aktuální situace a návrh řešení. Při zavádění znalostního managementu a případné implementaci a návrhu nového informačního systému je vhodné dodržovat podobná pravidla.

Zavedení znalostního managementu není možné zahájit implementací informačního systému znalostního managementu. Každá implementace by měla začít analýzou situace v organizaci, pokračovat návrhem řešení a až následně implementací řešení. Při zavádění je pak nutné se na znalostní management dívat v celém jeho rozsahu. [13]

4.2 Definice cílů

Znalostní management vyžaduje jasnou definici cílů, které má jeho zavedení přinést. Tyto cíle by pak měly být v souladu s vizí a strategií celé organizace. Ve vztahu k informačnímu systému je vhodné určit cíle, které mají být naplněny jeho implementací. Vybrané cíle by se měly promítnout do návrhu systému. Dále je možné identifikovat přesné funkční požadavky, které cíle plní.

Cíle by měly být dobře definované, měřitelné a krátkodobě dosažitelné. Mimo tyto 3 vlastnosti by cíle zavedení znalostního managementu a cíle systému znalostního managementu, podobně jako všechny ostatní cíle, měly

následovat metodiku SMART (popřípadě SMARTER). Ta určuje, že cíle musí být konkrétní, měřitelné, dosažitelné, realistické a časově ohraničené.

Zavedení znalostního managementu s sebou zpravidla nese změnu firemních procesů. Tato změna může ve výsledku trvat i několik let. Je tedy nutné všechny strany seznámit s dlouhodobými přínosy zavedení znalostního managementu. Dlouhodobost je také nutné promítnout do časové ohraničenosti jednotlivých cílů. [13]

4.3 Definice znalostí

Pro úspěšné zavedení znalostního managementu je zásadní identifikace znalostí, které jsou pro organizaci kritické. Tyto znalosti jsou následně středem návrhu informačního systému. Po jejich identifikaci je také zapotřebí popsat jejich dostupnost v rámci organizace a oprávněné uživatele.

Spolu s přístupy ke kritickým informacím se definují i vlastníci procesů a odpovědnost za jejich výstupy. V rámci procesů jsou zřetelné způsoby, kterými jsou znalosti zachytávány a jak se s nimi dále pracuje. Přidaná hodnota znalostí netkví v jejich držení, ale především v jejich použití. Pro vyspělý znalostní management není vlastnictví znalostí faktorem úspěchu. [13]

4.4 Integrace

Systém znalostního managementu by se neměl objevit jako zbrusu nový nástroj v rámci organizace. Lepší přístup je snaha integrovat systém s ostatními systémy a nástroji, které se již v organizaci používají. Pakliže je systém dostupný skrze již dříve známé rozhraní, uživatelé mají tendenci se systémem více interagovat a angažovat se v rozvoji jeho obsahu.

4.5 Gamifikace

Znalostní databáze jsou systémy, které jsou založené na komunitě a komunitně budovaném obsahu. Firemní znalostní databáze mají jasně definovanou komunitu, odpadá tedy problém s budováním komunity. Stále však je nutné udržet komunitu angažovanou a motivovanou systémem používat a tvořit obsah.

Mnoho lidí se asi shodne, že hry (nebo alespoň některé hry) jsou zábava. Pohlcení, které některé hry přináší, mohou vést až ke skutečné závislosti. A proto se software architekti začali ptát, jak přenést osvědčené herní koncepty do systémů, které z principu hry nejsou. Na těchto základech postupně vznikl celý obor zvaný gamifikace (*gamification*). Hlavním cílem gamifikace je využít osvědčených herních (a videoherních principů) pro udržení angažovanosti uživatelů.

4.5.1 Složitost hry

Jak uvádí Raph Koster, lidský mozek stále potřebuje motivaci, aby něco dělal. Lidský mozek stále hledá nějaký úspěch, nějaký moment, kdy si osvojí novou znalost nebo dovednost.

A právě v tento moment přichází na řadu hry a důvod, proč jsou zábavné. Během hraní se mozek stále snaží najít vhodnou strategii nebo trik, kterým hru přemůže. V momentě, kdy se mu to povede, přichází úspěch a potěšení z vyřešení daného problému. Avšak ihned po vyřešení problému přichází ztráta motivace. Mozek potřebuje novou výzvu, bez které se hra stává nudnou a opakující se. Mozek chce řešit nové problémy, nikoli stále aplikovat jednu nalezené řešení. [34]

V tuto chvíli si možná některý čtenář vybaví například hru World of Warcraft a její uživatele, kteří jí hrají klidně i několik hodin denně po dobu několika let. Jak taková hra udrží uživatele motivované, neopakuje se a stále nabízí nové výzvy pro uživatele? To logicky po několika tisících hodinách hraní není lehké. Hra však svým uživatelům konstantně přináší radost a pocit úspěchu. K tomu využívá techniku zvanou *Reward Schedule*, která má za úkol přinášet nové výzvy a odměny v ten pravý čas.

Hra však může přinášet i frustraci. Pokud hráč dlouho není schopen řešit problémy nebo se problémy zdají příliš složité a neřešitelné, může velmi snadno ztratit motivaci dál hrát. V takovém případě nejspíše získá názor, že hra prostě není postavená fér a on ji nemůže vyhrát.

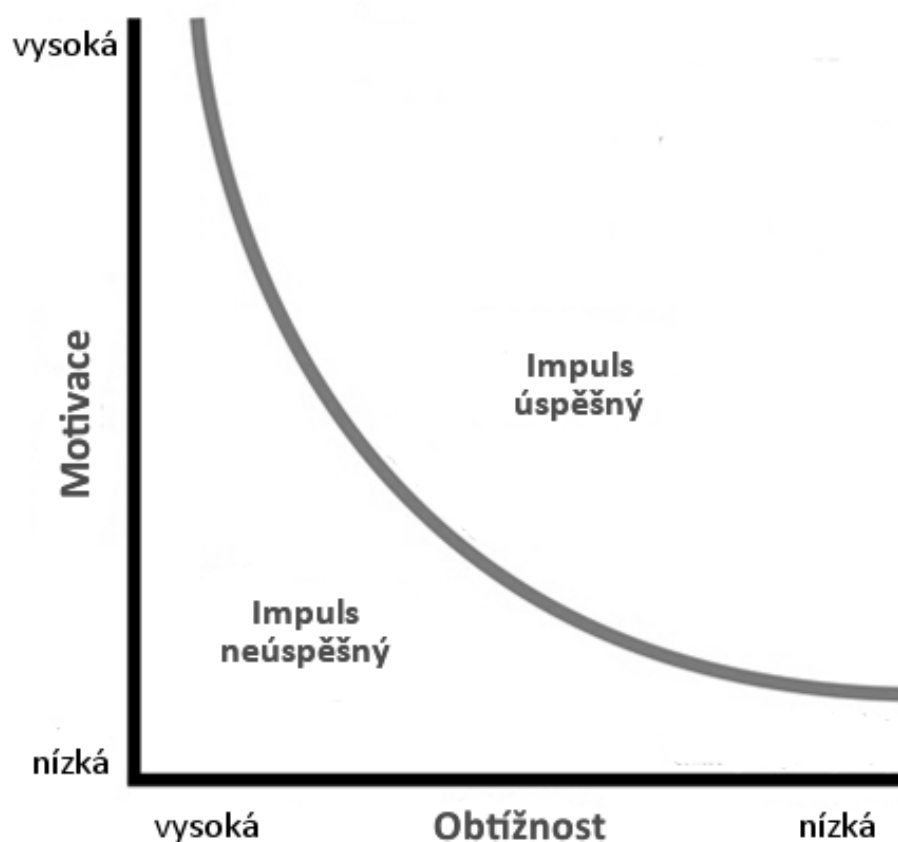
Pokud jsou herní prvky v informačních systémech využity dobře, jsou správně nastaveny, poskytují uživatelům férovou hru a přináší výzvy, používání takových systémů se stává zábavou. Ve výsledku uživatelé zůstávají déle a používají systém častěji. [35]

4.5.2 Foggův model

Aby bylo možné pokročit dále a ukázat principy gamifikace potřebné pro návrh informačního systému (především pak MDA frameworku), je nutné čtenáři představit Foggův model chování.

B. J. Fogg představil model chování, který ukazuje, co je potřeba pro docílení změny v určitém chování lidí. Ten tvrdí, že jsou potřeba tři faktory, aby mohlo k žádané akci dojít: 1. motivace (*Motivation*), 2. schopnost (*Ability*) a 3. impuls (*Trigger*).

Tyto tři faktory říkají, že v tu správnou chvíli musí přijít impuls. Aby byl impuls úspěšný, musí být jedinec schopen akci provést a zároveň musí být k akci dostatečně motivován. Úroveň motivace se poté může lišit dle obtížnosti akce. Pro těžké akce bude zapotřebí vyšší motivace než pro akce snadné. Tento fakt znázorňuje graf 41. Obtížnost akce je závislá nejen na schopnostech jedince, ale i na možnostech, které poskytuje okolí (například dostupné zdroje). [35] [36]



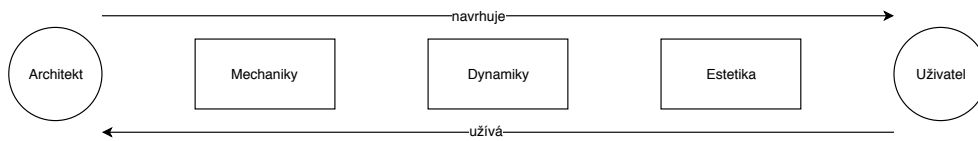
Obrázek 41: Foggův model chování [37]

Jako příklad je možné uvést telefonní hovor. Vyzvánění telefonu je impuls, který přichází. Aby akce byla provedena, musí mít dotyčný telefon u sebe a musí ho být schopen zvednout. Je-li schopen ho zvednout, musí mít motivaci to udělat (chtít s volajícím mluvit).

4.5.3 MDA framework

MDA framework byl představen v roce 2004 a jeho hlavním úkolem je formalizovat pojmy a postupy na poli teorie herního designu. Framework pomáhá sbližovat výsledek, kterým je zapojený uživatel a softwarové prostředky, které jsou pro vývoj hry k dispozici.

Hlavním rozdílem mezi hrami a jinými zábavnými produkty (například knihami, hudbou nebo filmy) je předvídatelnost výsledku. Výsledek hry a posloupnost událostí, které k němu vedou, jsou v době designu neznámé a nepředvídatelné. MDA framework proto rozkládá hru do tří komponent:



Obrázek 42: Perspektiva hráčů a herních architektů

1. pravidla,
2. systém a
3. zábava.

Každá komponenta má poté svůj designový protějšek:

1. mechaniky (*mechanics*),
2. dynamiky (*dynamics*) a
3. estetiku (*aesthetics*).

Mechaniky popisují jednotlivé komponenty hry na úrovni datové reprezentace a algoritmů. Dá se tak říci, že mechaniky jsou funkčními komponentami hry. Dynamiky popisují chování programu za běhu a reagují na uživatelské vstupy. Estetika je nejvyšší vrstva, se kterou uživatel typicky interaguje. Estetika rovněž popisuje emocionální reakce, které má hra v hráčích vyvolat. [35] [38]

Mechaniky a dynamiky jsou často zaměňovány. Čtenář by měl chápat, že mechaniky pouze podporují dynamiky. Jako příklad se dá uvést karetní hra poker. Mezi herní mechaniky patří míchání karet nebo sázení. Tyto mechaniky pak podporují herní dynamiky jako je blafování.

Může se zdát, že MDA framework mimo formalizace nepřináší nic nového. Je ale důležité se na chvíli vcítit do hráče a do herního architekta. Každý z nich se totiž na hru dívá trochu jinou perspektivou. Herní architekt ve hře vidí především mechaniky, zatímco hráč ve hře vidí především estetiku. Tuto myšlenku ilustruje obrázek 42. A právě formalizace pomáhá dostatečně rozmyslet obě perspektivy.

4.5.4 Používané mechaniky

Architekt herního systému se nejvíce věnuje herním mechanikám. Tato sekce má za cíl čtenáři představit vybrané herní mechaniky, které je možné pro návrh systému využít a které budou využity v navrhovaném systému v kapitole 8

4.5.4.1 Bodové systémy

Jednou z nejpoužívanějších mechanik a vlastně také nejúspěšnějších je bodový systém. S bodovým systémem je možné se setkat nejen ve značné části videoher, ale i sportů. Hlavním důvodem je fakt, že bodový systém dokáže být silný motivační prvek. Body ovšem mohou znít nudně, proto je bodový systém často nazýván odlišně.

Každý úkol nebo akce v rámci systému by měla být ohodnocená body. Samotná hodnota bodů ve výsledku není důležitá, avšak je důležitý bodový rozdíl mezi úkoly, který tvoří bodovou hierarchii. Bodovým ohodnocením uživatel dostává najevo důležitost a složitost daného úkolu oproti ostatním.

Obecně je složité bodový systém navrhnout správně hned napoprvé. Proto je vhodné sledovat chování uživatelů a jejich bodové zisky. Pokud se zdá, že něco není v pořádku, je možné upravit bodové zisky, aby hra byla zábavnější a plnila architektův záměr. Ačkoli návrh bodového systému není snadný, je vhodné ho vždy mít. Často ani nemusí být uživateli přístupný, i interní bodový systém umožňuje architektům hru vylepšovat. [35]

Bodové systémy se dají rozdělit do tří základních kategorií:

1. zkušenostní bodový systém,
2. karma systém a
3. systém směnitelných bodů.

Bodový systém založený na zkušenostech (*experience point system*) má za úkol měřit zapojení uživatele v rámci hry. Zkušenostní body jsou obvykle neomezené a nedají se koupit, prodat ani vyměnit za žádné benefity. Jediným smyslem zkušenostních bodů je ukázat a měřit úspěch a angažovanost v rámci hry.

Karma systém je založený na odměňování uživatelů za dokončených stanovených úkolů. Za body však není možné zakoupit žádné benefity, avšak je možné je využít na odměňování dalších uživatelů. Odměňování ostatních uživatelů pomáhá budovat komunitu, a proto jsou karma systémy často používány pro gamifikaci informačních systémů.

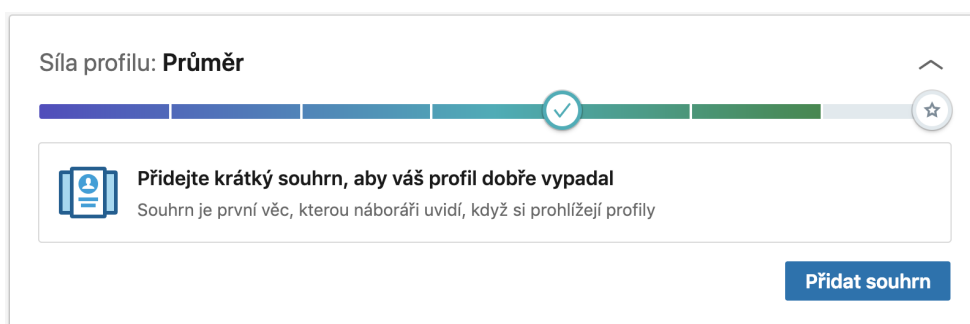
Systém směnitelných bodů je založený na odměňování uživatelů body, za které je následně možné zakoupit různé benefity nebo herní rozšíření. Tento systém je populární nejen ve videohrách, ale i v reálném životě, kde typicky vytváří malou virtuální ekonomiku (například body za nakupování v supermarketu, které jsou směnitelné za nějaký typ zboží). [35]

4.5.4.2 Levely

Levely jsou systémem, který na základě zvolené metriky odemyká uživatelům další části hry. Metrikou může být nějaký bodový systém (typicky založený na zkušenostních bodech) nebo třeba splnění určitých úkolů. Systém levelů

je pro gamifikaci informačních systémů obtížně použitelný, neboť uživatel by měl mít zpřístupněné všechny části systému již od začátku.

Ze systému levelů je však možné využít vizuální reprezentaci pokroku (*progress bar*). Tedy ukazatel, jak daleko se v rámci dané akce či úkolu uživatel nachází. Právě vědomí nedokončeného úkolu uživatele motivuje k jeho dokončení a případnému získání odměny. Zajímavým příkladem je síla profilu na sociální síti LinkedIn (na obrázku 43). [35]



Obrázek 43: Progress bar na sociální síti LinkedIn

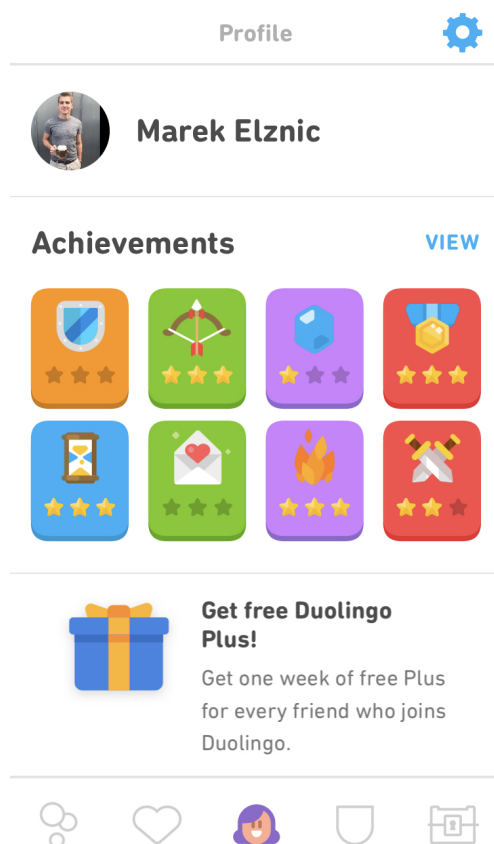
4.5.4.3 Odznaky a výzvy

Odznaky nebo také *badges* jsou vizuální reprezentace úkolů nebo sady úkolů, které má hráč splnit. Pro čtenáře méně znalé v oblasti videoher je možné odznaky přirovnat ke skautským bobříkům nebo armádním vyznamenáním.

Jedná se o jednoduchou vizuální motivaci k plnění stanovených úkolů. Od doby, kdy Foursquare ve své aplikaci prosadil používání odznaků, už odznaky používají tisíce aplikací a uživatelé na ně postupně přestávají reagovat. Hlavním důvodem je nepromyšlené používání odznaků, které nic nepřináší, a tedy ani nemotivuje uživatele. [35] Reálné použití odznaků v aplikaci Duolingo je vidět na obrázku 44

Odznaky mohou být spojeny s výzvami (*challenge* nebo *quest*). Výzvy jsou také silnými motivátory. Velkou výhodou je, že vedou hráče po předem připravených scénářích a navádějí ho k další akci. Využití výzev je v informačních systémech komplikovanější než ve videohrách. Avšak výzvy je možné využít v rámci *onboardingu*, tedy procesu představení informačního systému uživateli.

S nástupem éry sociálních sítí se začaly využívat i kolaborativní výzvy. Jedná se o výzvy, které je možné splnit pouze dohromady s ostatními uživateli. Taková výzva nutí uživatele se zorganizovat do větší skupiny, kde následně vzniká sociální tlak na dokončení výzvy. [35]



Obrázek 44: Odznaky v mobilní aplikaci Duolingo

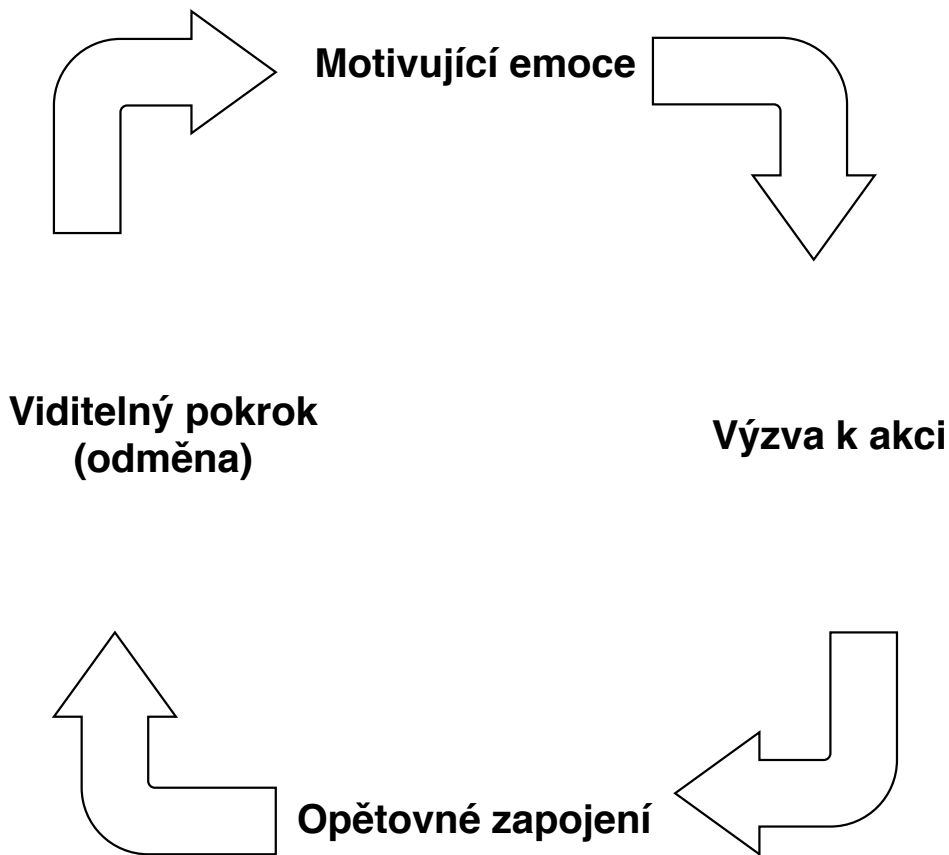
4.5.4.4 Leaderboardy

Leaderboardy jsou speciální žebříčky navržené tak, aby ukazovaly skóre uživatele v kontextu ostatních uživatelů. Vždy je zásadní prezentovat skóre uživatele v kontextu nejbližších soupeřů. Je-li uživatel na 100. pozici, měl by vidět i uživatele na pozici 99 a 101. Uživatel takto získává informaci, jak si ve hře stojí. Lehce vidí, kolik bodů mají nejbližší uživatelé před ním i za ním.

S érou sociálních sítí je možné profil propojit se sociální sítí a prezentovat žebříček v kontextu spojení na sociální síti. Tento use case je pro firemní znalostní databázi obzvláště užitečný, jelikož se uživatelé obvykle znají a propojení uživatelů je definováno společným pracovištěm. [35]

4.5.4.5 Social engagement loop

Social engagement loop se dá přeložit jako smyčka sociální angažovanosti. Ta má za úkol udržet uživatele zapojeného do dění ve hře. K tomu využívá pobídky nebo jiné pozitivní mechanismy. Ve výsledku smyčka přitahuje



Obrázek 45: Vizualizace Social engagement loop

uživatele, stimuluje ho k akci a odměňuje ho za dokončení akce tím, že nabízí další pobídku k přijetí další akce. Skládá se ze 4 základních kroků:

1. motivující emoce,
2. výzva k akci,
3. opětovné zapojení hráče a
4. viditelný pokrok (odměna).

Tyto kroky jsou vizuálně znázorněny na obrázku 45.

Zásadní však je, že tyto 4 kroky je nutné sledovat již při designu produktu. Pozdější pokus o použití *social engagement loop* je problematický a typicky končí nezdarem. [35]

4.6 Závěr

V této kapitole jsem shrnul nejdůležitější poznatky, na které je dobré brát zřetel při návrhu a implementaci systémů znalostního managementu. Věnoval jsem se analýze informačního systému, definici cílů a znalostí, integraci s dalšími systémy a gamifikaci informačních systémů.

Existující řešení

Aktuálně na trhu existují stovky softwarových řešení pro podporu znalostního managementu. V nabídce jsou znalostní databáze, *knowledge sharing* platformy i systémy znalostního managementu. Řešení se liší dle odvětví, na které se specializují, dle škálovatelnosti, ceny a dalších parametrů.

Vybral jsem některá nejznámější řešení. Tato sekce nemá za cíl uvést kompletní přehled všech znalostních systémů, avšak prezentovat jejich podmnožinu, s kterou bude možné pracovat v kapitole 6. Proto jsem vybral i různé typy systémů od jednoduchých systémů na správu dokumentů po placené korporátní řešení.

U systémů popisuji silné a slabé stránky a snažím se poukázat na možné komplikace při zavádění nebo údržbě systému.

5.1 Metodika

Pro popis jednotlivých řešení jsme stanovil funkcionalitu, která bude posuzována. Na základě posuzované funkcionality bude možné rozhodnout, pro jaký typ organizace je řešení vhodné.

Posuzovány mohou být jak systémy znalostního managementu, které podporují většinu znalostních procesů, tak i znalostní databáze, které primárně slouží na ukládání znalostí. Ačkoli se tyto systémy v mnohém liší, různé typy podnikání potřebují různé systémy.

5.1.1 Základní funkcionalita

Po zdefinování procesů znalostního managementu a zavedení informačního systému na jeho podporu přichází krok, kdy je nutné do systému vložit základní data. O tento krok by se měl postarat znalostní manažer, případně v menších firmách má znalostní management na starosti HR manažer.

Vkládání dat do systému, jejich úprava, vyhledávání a mazání je naprosto základní funkcionalita. Ta vychází z modelů znalostních procesů prezento-

5. EXISTUJÍCÍ ŘEŠENÍ

vaných v sekci 2.5. Tuto funkcionalitu musí mít bezpodmínečně každý systém, aby ho vůbec bylo možné považovat za systém pro znalostní management. Zkoumat tedy budu:

- přidání znalosti (záznamu),
- smazání znalosti a
- vyhledávání.

5.1.2 Správa uživatelů

Po naplnění systému znalostmi přichází na řadu přidání uživatelů do systému. Správa uživatelů se v tomto případě dá rozdělit na dva menší požadavky: 1. samotná správa uživatelů a 2. definice uživatelských rolí a přístupu.

Ačkoli pro menší firmy není problém pro každého zaměstnance vytvořit účet a nastavit vhodná práva, pro větší firmy je toto řešení příliš omezující. V takovém případě je vhodné mít možnost nadefinovat pracovní pozice, dle kterých jsou rozdělena práva pro čtení nebo zápis. Místo pozic je možné definovat i skupiny uživatelů, kterým jsou přidělena různá přístupová práva.

Přidávání uživatelů po jednom také není vhodným řešením. Proto je lepší mít možnost uživatele hromadně naimportovat, případně propojit například s firemním SSO či přímo LDAP. Díky použití SSO odpadá nutnost uživatele spravovat v systému a je možné ho řídit centrálně.

5.1.3 Integrace

Zavedení nového informačního systému se často nepotká s počátečním nadšením, neboť nové systémy mají tendenci měnit nastavené (či pouze zaběhlé) firemní procesy, na které jsou zaměstnanci zvyklí. Díky integraci s existujícími nástroji, je možné informační systém znalostního managementu začlenit mezi stávající systémy. Uživatelé tak mohou přistupovat ze známých nástrojů k novému obsahu a nový obsah také vytvářet.

V rámci integrace budu hodnotit nejen předpřipravené integrace, kterými systémy disponují, ale i zda a jaké poskytují API pro tvorbu vlastních integrací (například s interními nástroji v rámci organizace).

5.1.4 Zapojení uživatelů

Systém znalostního managementu je především komunitní systém a je do velké míry závislý na uživatelsky generovaném obsahu. Aby uživatelé interagovali se systémem a vytvářeli obsah, je důležité udržet je angažované. Tomuto tématu jsem se věnoval v sekci 4.5 o gamifikaci. V rámci zapojení uživatelů budu hodnotit především mechanismy pro angažování uživatelů, kterými systém disponuje.

5.1.5 Cena

Cena je důležitým faktorem, který ovlivňuje výběr informačního systému. Hostovaná řešení mají nižší pořizovací náklady, avšak vyžadují pravidelné měsíční platby. Naopak *open-source* řešení vyžadují vyšší počáteční investici, která se vrátí až v průběhu let. Podobná situace nastává i v případě řešení na míru, u kterého však pořizovací náklady ještě převyšují cenu *open-source* řešení. Abych tuto nerovnost vyřešil, náklady na informační systém budu počítat na první tři roky provozu systému.

Na tomto místě je vhodné zmínit, že cena řešení by nikdy neměla být hlavním kritériem. Výběr nevhodného informačního systému může být v průběhu let velmi nákladný. Pokud zaměstnanec denně ztratí 5 minut času kvůli nevhodně navrženému systému, za rok to může být až 20 hodin času. Tyto náklady, podobně jako další náklady spojené se zaměstnanci, však samozřejmě nemohou být zahrnuty do ceny řešení.

5.2 Vybraná řešení

5.2.1 Confluence

Firma Atlassian přišla se systémem na správu dokumentů již v roce 2004. Tehdy Confluence představila jako podnikový systém pro podporu znalostního managementu. Její princip vychází z konceptu wiki, tedy volně editovatelných hypertextových dokumentů. Za 15 let existence se systém vyvinul v plnohodnotnou znalostní databázi, která je hojně nasazována ve velkých IT podnicích.

Confluence vyniká ve vyhledání dokumentů. Systém umí vyhledávat především fulltextově, a to nejen ve všech dokumentech, ale i jejich verzích.

Jelikož se jedná o korporátní řešení je jeho součástí i flexibilní systém pro autorizaci. Díky mnoha rozšířením, které jsou však většinou placené, je možné Confluence propojit s takřka libovolným poskytovatelem SSO. Mezi ty nejpoužívanější určitě patří Google Suite, Salesforce nebo Azure AD od Microsoftu.

Jednou z hlavních charakteristik Confluence je její provázání s dalšími produkty firmy Atlassian, především pak softwarem na evidenci chyb (*issue tracking software*) Jira a verzovacím systémem BitBucket. Čisté a promyšlené UI je pak doménou většiny produktů firmy Atlassian.

Co se integrací týče, disponuje Confluence jednoduchým, ale o to silnějším API, přes které je možné se systémem komunikovat. API je navrženo pomocí principu REST, kterému není co vytknout. [39]

Zapojení uživatelů probíhá primárně propojením s dalšími produkty firmy Atlassian. V celém ekosystému se Confluence prakticky nelze vyhnout. Žádnými speciálními strategiemi pro angažování uživatelů však systém nedisponuje.

Tabulka 51: Hodnocení Confluence

Kritérium	Hodnocení
Správa uživatelů	90 %
Integrace	80 %
Zapojení uživatelů	30 %
Cena	80 %
Celkem	70 %

Confluence je hodně variabilní a je možné jí používat i jako DMS. A právě variabilita může být i nevýhodou. Při špatném nastavení se může struktura znalostní databáze časem rozvolnit a začít způsobovat problémy. Postupem času může být těžší a těžší najít informace, které jsou v daný moment potřeba. Je důležité si totiž uvědomit, že dokumenty nejsou znalosti. Ke znalostem se musí přistupovat odlišně a mít na paměti jejich tacitnost.

Confluence je možné pořídit jako *Software as a Service*. Cena se pohybuje dle počtu uživatelů od \$1 do \$5 měsíčně za uživatele. Alternativně je možné jednorázově pořídit licenci dle počtu uživatelů. Poté se cena pohybuje kolem \$10 na uživatele a zahrnuje údržbu na jeden rok, avšak nezahrnuje náklady na hardware a nasazení. [40]

5.2.2 Zendesk Guide

Guide je znalostní databáze od firmy Zendesk, která se prosadila především svým řešením pro zákaznickou podporu a ticketování. Guide organizuje dokumenty a znalosti do kategorií, sekcí a článků.

Pro správu uživatelů používá Zendesk stejný systém pro všechny své produkty. Jelikož se jedná o korporátní řešení, podporuje Zendesk SSO postavené nad SAML 2.0 a JWT. Mimo to nabízí předpřipravené řešení pro Azure AD a Google Suite. [41]

Nejlépe je Guide integrován s ostatními produkty firmy Zendesk (například Support nebo Chat). Nabízí však i další možnosti, a to především skrze Zendesk Marketplace, kde je možné dokupovat jednotlivá rozšíření. Ačkoli Zendesk nabízí API k mnoha svým produktům, Guide mezi nimi není. Mezi systémy, které API mají, jsou některé jiné systémy, které Guide využívá (například Answer Bot). K některým funkcionalitám je tak možné stále přistupovat. [42]

Zapojení uživatelů Guide řeší primárně integrací s ostatními systémy Zendesk. V rámci celého ekosystému jejich produktů je přístup do Guide možný odkudkoli.

Běžný tarif se pohybuje od \$15 do \$29 měsíčně za uživatele, avšak je možné zvolit i tarif zdarma, který nabízí jen minimum funkcí. Zajímavá je možnost za \$50 měsíčně přikoupit i Answer Bota.

Tabulka 52: Hodnocení Zendesk Guide

Kritérium	Hodnocení
Správa uživatelů	90 %
Integrace	40 %
Zapojení uživatelů	30 %
Cena	50 %
Celkem	52,5 %

5.2.3 Bloomfire Knowledge Sharing Platform

Bloomfire svůj produkt prezentuje jako platformu na sdílení informací, za jejímž úspěchem stojí umělá inteligence. Hlavní vlastností celé platformy je automatické tagování všech dokumentů, které jsou do systému vloženy. Tagování probíhá automaticky za pomoci umělé inteligence, což má za úkol šetřit čas zaměstnanců. Zajímavostí je, že Bloomfire dokáže rozpoznat slova ve videu a každé z nich zaindexovat. Přirozeně tato funkce funguje primárně v anglickém jazyce.

Nástroj má poměrně jednoduché UI, ve kterém je snadné se zorientovat. Základními prvky jsou vyhledávání a filtrování. Nové záznamy jsou zobrazeny na velkých kartách, které připomínají sociální síť Pinterest.

Správa uživatelů je řešena skrze interní administraci. Dále existují rozšíření, které podobně jako u Confluence přidávají některé možnosti SSO. Další rozšíření pak přidávají podporu pro SAML 2.0, OAuth nebo OpenID.

Co se týče integrací s existujícími nástroji, nabízí Bloomfire Slack, Salesforce, Zendesk a doplněk pro Chrome. Pro integraci s dalšími nástroji nabízí Bloomfire přehledné REST API. [43]

Jednou z hlavních výhod (například oproti Confluence) je Q&A. Jedná se o sekci systému, ve které je možné pokládat otázky a ostatní uživatelé mohou odpovídat. Jak se ukázalo na úspěchu *Stack Overflow* nebo *Quora*, model otázek a odpovědí je velmi jednoduchý a oblíbený. Navíc v tomto případě tento model může pomoci s vytvářením znalostí. Ty mohou vznikat přímo z odpovědí na otázky. Žádné jiné mechanismy pro udržení uživatelů však systém nemá.

Bloomfire je poskytován jako *Software as a Service* a cena se odvíjí od počtu uživatelů. Do 35 uživatelů je měsíční předplatné \$639, do 100 uživatelů pak \$2299. Při plném využití se cena za uživatele blíží \$20 měsíčně. Dražší verze *Growth* obsahuje některé funkce navíc (například SSO nebo API přístup). [44]

5.2.4 Guru

Znalosti jsou v Guru ukládány a zobrazovány jako karty. Každá karta patří do nějaké kolekce, která se následně skládá z tabulí (*boards*). Asi nejzásadnějším

Tabulka 53: Hodnocení Bloomfire

Kritérium	Hodnocení
Správa uživatelů	85 %
Integrace	95 %
Zapojení uživatelů	50 %
Cena	50 %
Celkem	70 %

rozdílem Guru oproti ostatním systémům znalostního managementu je důvěra v znalosti (*trust*). Znalost je označena jako důvěryhodná, pokud je vybraným uživatelem pravidelně zkontrolována.

Správa uživatelů probíhá v interní administraci. Uživatele je možné pozvat pomocí seznamu e-mailových adres. Následně jim lze přiřadit různé role a stupně oprávnění. Od balíčku Team je možné používat i SSO, přičemž Guru automaticky podporuje přihlášení skrze nejpoužívanější poskytovatele identity.

Příjemnou funkcí je možnost migrace znalostí z vybraných znalostních databází jako je třeba Confluence nebo Zendesk Guide. Dále je pak možné Guru integrovat s nespočtem nástrojů mezi které patří například Slack, Salesforce, cloudová úložiště Box a Google Drive, nástroj na projektové řízení Asana a mnoho dalších. Pro tvorbu dalších integrací služba nabízí REST API. [45]

Jako jeden z prvků pro zapojení uživatelů používá Guru takzvané *trust score*. To označuje procento karet každého uživatele, kterým je možné důvěřovat. Tímto mechanismem se systém snaží přesvědčit uživatele, aby pravidelně udržovali a verifikovali jimi vložené znalosti. Zajímavá je i míra propojení s ostatními aplikacemi. Guru například dokáže napovídat různé informace při psaní e-mailu.

Guru má nejpropracovanější integraci se Slackem ze všech nástrojů. Mimo to, že umí vytvářet nové karty přímo ze zpráv, umí rovněž přímo ze Slacku prohledat všechny karty. Z vrácených výsledků uživatel může rovnou vytvořit odpověď.

Guru je podobně jako ostatní nástroje poskytován v režimu SaaS. Cena se odvíjí nejen podle počtu uživatelů, ale i dle požadovaných funkcí. Konečná cena za uživatele se pohybuje od \$19 do \$29 měsíčně a jedná se o jednu z nejvyšších.

5.2.5 AllAnswered

AllAnswered je webově založený systém, který je organizován kolem komunit. Informace a znalosti jsou primárně sdíleny v rámci komunity. Komunitou může být pracovní tým, projektový tým nebo jen skupina lidí se společným zájmem. Informace a znalosti se ukládají do stránek (*pages*), které jsou tříděny do

Tabulka 54: Hodnocení Guru

Kritérium	Hodnocení
Správa uživatelů	90 %
Integrace	98 %
Zapojení uživatelů	90 %
Cena	30 %
Celkem	77 %

Tabulka 55: Hodnocení AllAnswered

Kritérium	Hodnocení
Správa uživatelů	20 %
Integrace	40 %
Zapojení uživatelů	90 %
Cena	80 %
Celkem	57,5 %

složek. Je možné nastavit periodu, během které musí být znalosti verifikovány, a také typy uživatelů, kteří mohou znalost verifikovat. Nakonec je stránky možné hodnotit jako to nabízí například *Stack Overflow*. Mimo stránky systém nabízí i ankety nebo průzkumy.

Pro správu uživatelů jsou využívány pozvánky. Každý uživatel se musí individuálně zaregistrovat a následně ho na základě e-mailu může administrátor komunity přidat do své komunity. Asi není nutné zdůrazňovat, že tento způsob není pro použití ve větších firmách vhodný.

Nepodařilo se mi nikdy nelézt dostupné API ani seznam integrovaných aplikací. Existuje však Slack bot, který umožňuje propojit kanály na Slacku s komunitami v AllAnswered. Všechny nové záznamy se pak automaticky zobrazí i v příslušném Slackovém kanálu.

Pro udržení angažovanosti uživatelů systém používá klasický leaderboard. Ten typicky ukazuje nejaktivnější uživatele uvnitř komunity a dále pak nejaktivnější komunity. Každý uživatel má dle hodnocení vypočítanou reputaci. Opět je zde možné pozorovat prvky, které fungují na *Stack Overflow*.

AllAnswered je také poskytován jako SaaS a cena se rovněž odvíjí od počtu uživatelů. Tarif se liší dle organizace – pro výukové nebo neziskové účely je cena \$1 měsíčně za každého uživatele. Pro ostatní je cena \$5 za uživatele a měsíc.

5.2.6 Nuclino

Nuclino třídí znalosti, informace a dokumenty do pracovních ploch (*workspaces*), které se dále dělí na clustery a položky. Zajímavá je vizualizace položek

Tabulka 56: Hodnocení Nuclino

Kritérium	Hodnocení
Správa uživatelů	95 %
Integrace	80 %
Zapojení uživatelů	30 %
Cena	70 %
Celkem	68,75 %

pomocí grafu, ve kterém je názorně vidět množství ploch, clusterů a položek a jejich vzájemné propojení.

Uživatele je možné pozvat skrze URL adresu, což pro menší týmy může být velmi rychlý způsob přidání uživatelů. Mimo tento způsob je k dispozici pozvánka skrze list e-mailů. V placené verzi je možné nalézt i SSO. Možnosti jsou podobné jako u ostatních produktů – Google Suite, Azure AD, Auth0 nebo OneLogin.

Integrací Nuclino rovněž nabízí nespočet. Mimo standardní Slack nabízí například online software pro tvorbu prezentací Prezi, či online nástroje zaměřené na webový vývoj CodePen, JSFiddle a Plunker. Dále systém nabízí i API, avšak nebylo možné k němu dohledat dokumentaci.

Žádné mechanismy pro udržení uživatelů jsem v systému neobjevil. V zásadě se systém dá spíše přirovnat k lépe uspořádaným Google Docs.

Jako jeden z mála systémů nabízí Nuclino i verzi zdarma. V základní verzi se však jedná spíše o jednoduchý organizér na dokumenty, který spíše soupeří s Google Docs. Placená verze je za \$5 nebo za \$10 za uživatele měsíčně. Přičemž dražší verze obsahuje pouze prémiovou podporu a více místa pro položky.

5.2.7 OpenKM

OpenKM je svobodný software vydávaný pod licencí GNU GPL. Hlavní rozdílem oproti dříve prezentovaným řešením je právě licence a systém poskytování. Software je poskytován zdarma, bez podpory a včetně zdrojového kódu. Uživatel musí řešení nasadit na vlastní server, což jsou spolu s podporou jediné tři náklady na provoz software.

Dále už se však od dříve prezentovaných řešení tolik neliší. Systém poskytuje webové rozhraní a slouží primárně k ukládání dokumentů nebo jiných nestrukturovaných souborů. Rozhraní, jak je častou praxí u Open source softwaru, není z nejpřehlednější. Pro práci je však dostačující.

Pro správu uživatelů systém nabízí interní administraci včetně pokročilého řízení rolí a nastavení úrovně bezpečnosti jednotlivých dokumentů. Pro správu uživatelů je možné nakonfigurovat i LDAP a SSO. [46]

Tabulka 57: Hodnocení OpenKM

Kritérium	Hodnocení
Správa uživatelů	80 %
Integrace	80 %
Zapojení uživatelů	30 %
Cena	70 %
Celkem	65 %

OpenKM nabízí mimo klasického REST API i SDK pro programovací jazyky Java, .NET a PHP. Z předpřipravených integrací se dá zmínit WordPress, Joomla nebo doplňky pro MS Office. Dále je možné OpenKM propojit s tiskárnami a skenery Canon nebo čtečkami čárových kódů.

Systém pro zapojení uživatelů žádné zvláštní mechanismy neposkytuje. Tento fakt vychází primárně ze zaměření systému na větší korporace a na poskytování robustních služeb.

Licence na software je zdarma a není nutné platit za počet uživatelů. Jediné náklady se tak týkají nasazení, serverů a údržby a případné customizace systému. Pořizovací náklady jsou tedy vyšší, avšak postupem času mohou klesat.

Doporučení pro IT firmy

V této kapitole se budu věnovat organizacím působícím v segmentu informačních technologií. Na základě zkušeností a analýzy trhu vyberu různé typy organizací působící v tomto odvětví a pro jejich potřeby doporučím vhodný informační systém na podporu procesů znalostního managementu. Vyhnu se organizacím, které používají informační technologie na podporu svého chodu, avšak jejich hlavní zaměření není v oboru informačních technologií. Důvodem tohoto rozhodnutí je fakt, že v dnešní době prakticky každá organizace využívá nějaké informační technologie pro svůj chod. To z ní však nutně nedělá organizace působící v IT segmentu.

6.1 Kritéria

6.1.1 Zákazník

Asi největším rozdílem mezi organizacemi je definice zákazníka a trhu, na kterém organizace působí. Nejznámější termíny, které označují tyto vztahy jsou B2B a B2C.

B2B (*Business-to-business*) označuje situaci, kdy organizace nabízí produkty nebo služby dalším organizacím. Proti tomu stojí B2C (*Business-to-consumer*), které označuje situaci, kdy organizace nabízí produkty nebo služby koncovým zákazníkům (spotřebitelům). Modelových vztahů existuje mnohem více (C2C, B2G, atp.), avšak tyto dva jsou nejvhodnější pro základní rozdělení IT organizací.

6.1.2 Produkt

Dalším kritériem je typ produktu, který organizace nabízí. V sekci 6.1.1 jsem produkt organizace záměrně označoval jako produkt nebo službu. Toto rozdělení je pro typologii organizace také důležité.

Tabulka 61: Rozdělení podniků dle obrátu

Typ podniku	Min. počet zaměstnanců	Max. počet zaměstnanců
Mikropodnik	-	10
Malý podnik	10	50
Střední podnik	50	250
Velký podnik	250	-

Organizace poskytující služby mohou být agentury poskytující programátorské nebo analytické práce. V poslední době se rozmohl i dříve zmíněný model SaaS, tedy poskytování softwaru jako služby.

Naproti tomu organizace zaměřené na produkt mohou být firmy prodávající hardware či krabicový software. Za zvýšením zisku k těmto produktům typicky poskytují i doplňkové služby jako je podpora, údržba nebo servis.

6.1.3 Velikost

Dalším kritériem je velikost podniku, která může při rozhodování hrát zásadní roli. Velikost podniku se obvykle měří dle obrátu nebo dle počtu zaměstnanců. Jelikož se náklady na informační systém odvíjí nejčastěji od počtu uživatelů, jak jsem představil v kapitole 5, vyberu jako kritérium počet zaměstnanců podniku.

Pro rozdělení podniků na malé, střední a velké jsem vybral aktuálně platné nařízení Evropské komise z roku 2008. Z důvodů uvedených dříve vynechám podmínku na bilanční sumu roční rozvahy. Nařízení definuje malý podnik jako podnik, který má mezi 10 a 50 zaměstnanci. Střední podnik definuje jako podnik s minimálně 50 a maximálně 250 zaměstnanci. Velký podnik je pak definován počtem zaměstnanců vyšším než 250. Dále nařízení definuje ještě mikropodnik, který však v tomto dělení vynechám. Kompletní rozdělení je zachyceno v tabulce 61. [47]

6.2 Organizace orientované na služby

6.2.1 Agentury

Organizace orientovaná na služby (často také označovaná jako digitální agentura) typicky působí na B2B trhu a svým klientům dodává programátorské a další podpůrné služby, případně software na zakázku. V prvním případě jde o dodávku služeb. Služba je typicky předem objednána a následně je placena dle odpracovaných hodin.

Ve druhém případě by se mohlo zdát, že jde o dodávku produktu, avšak jedná se spíše o zhotovení díla. Klíčový rozdíl je v majetkových právech. V případě díla je agentura pouze zhotovitel. Všechna majetková práva přechází na objednatele. Kdyby se jednalo o produkt, agentura by dále disponovala i

majetkovými právy, která by ji opravňovala produkt dále distribuovat (což je zásadní charakteristika produktu).

Agentura je efektivní díky vysoké optimalizaci využití zdrojů a také díky *know-how*, které během realizace zakázek získává. Toto *know-how* optimálně musí zůstat uvnitř organizace i po odchodu jeho nositelů.

Velikost agentury není možné jednoznačně definovat. Agentura může být mikropodnik s jednotkami zaměstnanců, ale i nadnárodní organizace se stovkami či tisíci zaměstnanci. Navíc se počet zaměstnanců může velmi dynamicky měnit v čase například dle počtu a objemu zakázek. Primárně dle počtu zaměstnanců se pak bude lišit doporučení pro výběr systému znalostního managementu.

Menší agentury (mikropodniky a malé podniky) řízení znalostí a znalostní management často neřeší. Potřebují se plně věnovat akvizici zakázek a jejich plnění. Tyto organizace hledají jednoduchý systém, který nebude výrazně zatěžovat rozpočet firmy, a především nebude zdržovat zaměstnance při práci.

V takových případech je pro ukládání znalostí možné využít ty nejprimitivnější způsoby, jakými jsou například sdílené dokumenty nebo složky. Z informačních systémů je pak možné vybrat AllAnswered (5.2.5). Systém je jednoduchý na používání a v porovnání s ostatními systémy vyniká i cenou.

Větší agentury (střední a velké podniky) by si již měly plně uvědomovat, jak kritickou roli pro jejich podnikání hrají znalosti. Proto by se o ně měly také řádně starat. Větší agentura často zvažuje zřízení pozice znalostního manažera. Ten k podpoře svých procesů potřebuje profesionálnější systém, za který je však nutné si připlatit.

Na tomto místě je dobré rozmyslet, kolik implementace vhodného systému zaměstnancům ušetří času. Ušetřený čas je často cennější než ušetřené peníze. K tomu je možné přidat ještě spokojenost zaměstnanců, která je pro jejich setrvání také důležitá.

Ze představených existujících řešení je možné vybrat například Guru (5.2.4). Hlavní silou tohoto systému je jednoduché vyhledávání a integrace s nástroji, které IT agentury obvykle používají (například Slack nebo Zendesk).

6.2.2 Služby k produktu

Speciálním typem organizace orientované na služby, je organizace, která interně vyvíjí produkt, který nabízí zákazníkům za velmi malý poplatek nebo dokonce zdarma (například jako open-source). Primárním zdrojem příjmů není prodej produktu, ale nabídka doplňkových služeb k tomuto produktu. Doplňkovou službou tak může být například servis, podpora, školení nebo pomoc s instalací.

V takovém případě organizace interně opravdu vyvíjí produkt, který však není přímo monetizován. Monetizace přichází až skrze služby, které by však

bez produktu nemohly existovat. Příkladem takové společnosti je poskytovatel open-source řešení Red Hat nebo tvůrce PHP frameworku Symfony.

Tyto organizace primárně působí na B2B trhu a nabízí své řešení ostatním organizacím, nikoli koncovým spotřebitelům. Z pohledu velikosti se pak organizace řadí mezi střední a velké podniky se stovkami či tisícovkami zaměstnanců.

Je-li hlavní nabízenou službou podpora produktu, je výhodné zvolit řešení, které bude informace a znalosti poskytovat velmi rychle. Jen tak je možné řešit akutní problémy zákazníků co nejefektivněji. Vhodné je proto hledat řešení, které kvalitně a rychle vyhledává a zároveň co nejlépe integruje s používaným systémem zákaznické podpory (*Service Desk*). Například tedy Confluence a Jira Service Desk nebo Guide a Zendesk.

Dobré je i zvážit, zda systém podporuje zákaznickou znalostní bázi. Jejím vytvořením je možné pomoci uživatelům řešit problémy svépomocí a snížit tak zátěž zákaznické podpory.

6.3 Organizace orientované na produkt

6.3.1 Vývoj a prodej produktu

Organizace orientovaná na produkt může působit v podstatě na kterémkoli trhu (B2B, B2C, B2G). Její záměr je poskytnout co nejlepší produkt co největšímu množství zákazníků.

Produkt zde může být software nebo hardware. V případě softwaru se typicky jedná o počítačové programy, které organizace interně vyvíjí a zákazníkům nabízí licenci na užívání těchto programů. Licence může být časově omezená (například pouze na rok) nebo neomezená (doživotní licence).

Na rozdíl od organizace orientované na služby není její zisk omezený počtem zaměstnanců, odpracovanými hodinami nebo velikostmi zakázek. V případě hardwaru je zisk omezený počtem kusů, které je organizace schopná vyrobit. V případě softwaru je zisk prakticky neomezený (omezený počtem uživatelů, kterým je produkt možné prodat).

Produktové organizace z praktických důvodů nebývají mikropodniky. Vývoj a distribuce produktu vyžaduje mnoho rozdílných činností. Proto je možné narazit na malé a střední podniky, které poskytují software s lokálními ambicemi (například účetní software, který je vázaný na legislativu dané země).

Na druhé straně jsou zde nadnárodní organizace, které spadají do kategorie velkých podniků. Tyto organizace poskytují desítky či stovky softwarových produktů, které mají globální působnost. Mezi takové je možné zařadit například Microsoft nebo Oracle.

Ještě trochu stranou stojí společnosti, které vyrábějí hardware. Vývoj hardwaru je nákladný, a proto se jedná opět o velké podniky s nadnárodní působností (například Apple nebo Samsung).

Menší společnosti se mohou spolehnout na některá levnější řešení jako je například AllAnswered nebo dokonce open-source. Obzvláště pokud mají vlastní zdroje, které mohou na provoz a správu dedikovat.

Velké podniky by měly především hledat řešení, které podpoří jejich procesy a bude integrovatelné s používanými nástroji. Pokud tedy organizace používá systém Jira pro ticketování, bude vhodnou volbou Confluence (obdobně například Zendesk a Guide).

6.3.2 Software as a Service

Organizace, které poskytují *Software as a Service* (software formou služby), mohou působit na B2B nebo B2C trhu, popřípadě na obou zároveň. Ačkoli organizace poskytuje uživatelům službu, jedná se pouze o business model. Uvnitř organizace funguje více jako organizace orientovaná na produkt, neboť skutečně vyvíjí produkt pro uživatele. Tento typ organizace sdílí mnoho podobného s organizacemi zaměřenými na prodej doplňkových služeb k produktu (6.2.2), avšak liší se především monetizací produktu.

Velikost organizace v tomto případě neklade žádné speciální požadavky na systém. Je však nutné zvážit, jak je které řešení škálovatelné s velikostí organizace a počtem nabízených produktů. Dále bude velikost organizace ovlivňovat náklady spojené s pořízením a provozem systému.

Z popsaných existujících řešení je možné vybrat například Bloomfire, který je možné použít jako interní i zákaznickou znalostní bázi.

Analýza aktuálního stavu

7.1 Definice zákazníka

Zadáním bylo určeno, že navržený informační systém má sloužit jako znalostní báze ve střední IT firmě. Zaprvé je vhodné rozvést, že informační systém bude primárně sloužit jako znalostní databáze, avšak některé jeho funkce budou rámec znalostní databáze přesahovat.

Zadruhé je nutné vybrat definici pro středně velkou firmu. K tomu opět využijí definici Evropské unie. Tato definice je užívána především při žádostech o podporu v rámci programů Evropské unie. Základním kritériem pro posouzení velikosti firmy je počet zaměstnanců, roční obrat a bilanční suma roční rozvahy. Definice pak uvádí, že střední firma má mezi 50 a 250 zaměstnanci, roční obrat nepřesahuje 50 milionů EUR a výše aktiv nepřesahuje 43 milionů EUR. [47]

Zatřetí je potřeba rozvést, co znamená IT firma. Aby nebylo o splnění tohoto kritéria pochyb, zvolil jsem prostředí zakázkového (agenturního) vývoje software. Jedná se tedy o firmu, která má vlastní zaměstnance a svým klientům dodává zakázkový software a další podpůrné služby. Těmi mohou být například UX a UI design, projektové řízení, *quality assurance* a podobně. Tento typ organizace byl podrobněji popsán v sekci 6.2.1.

Výběr agenturního prostředí je zajímavý především nutností být efektivní a svoji efektivitu kontinuálně zvyšovat. Ačkoli být efektivní si přeje každá organizace, pro agenturu je to naprostá nutnost. Klienti si vybírají agentury právě proto, že dokážou danou službu poskytnout efektivněji a kvalitněji. Tento fakt se netýká pouze agentur na vývoj software, ale i reklamních agentur, personálních agentur anebo třeba advokátních kanceláří.

7.1.1 Strategie organizace

7.1.1.1 Předmět podnikání a stav organizace

Organizace byla založena v roce 2011 a již osmým rokem pomáhá ostatním firmám se rozvíjet. Hlavním předmětem podnikání je vývoj software a poradenství v oblasti start-upů. Organizace se specializuje na kompletní start projektu od nápadu, přes realizaci až po jeho údržbu. Mimo jiné však pomáhá firmám s digitalizací, revizí procesů nebo IT infrastrukturou.

7.1.1.2 SWOT analýza

Nejdůležitější silné stránky firmy jsou

- know-how v oblasti exekuce projektů,
- silný a zkušený tým analytiků a projektových manažerů a
- v poslední řadě mimořádné týmy vývojářů

Slabými stránkami jsou

- postupná ztráta produktivity,
- stárnutí týmu a
- pomalá expanze.

Potencionální příležitosti jsou

- zisk nových, velkých projektů od stávající klientské základny,
- expanze na nové trhy a
- specializace v nových oblastech.

Největšími hrozbami pro firmu jsou

- ztráta klíčových zaměstnanců a odliv know-how,
- ztráta konkurenceschopnosti a
- hospodářská krize, která zastaví růst nových firem.

7.1.1.3 Vize

Vizí organizace je kompletně eliminovat plýtvání zdrojů. Skrze optimalizace a inovace se organizace snaží vytvářet nové firmy, které pracují s maximální možnou efektivitou. Podobně se snaží transformovat existující firmy na vysoce efektivní podniky, které jsou schopny vytěžovat poskytované zdroje na hranici možného. To vše organizace dělá s ohledem na životní prostředí, a především pak s ohledem na celkový dopad na společnost. Dalším posláním je pomáhat firmám vytvářet inovativní produkty založené na špičkových technologiích.

7.1.1.4 Strategie pro rok 2020

Organizace aktuálně působí v Česku, Německu a Polsku. V roce 2020 se firma zaměří na zahraniční expanzi. Hlavními cíli pro expanzi jsou země Beneluxu a Skandinávie. V tomto roce by společnost chtěla otevřít minimálně 5 nových poboček a nabrat alespoň 10 nových klientů z těchto zemí.

Spolu s expanzí souvisí i růst počtu zaměstnanců. Společnost plánuje růst počtu zaměstnanců z aktuálních 70 na 100. Více než polovinu nových zaměstnanců by měli tvořit lidé pracující v nově otevřených pobočkách.

7.1.1.5 Strategie pro rok 2021

Během roku 2021 by společnost ráda stabilizovala firemní procesy v nově vzniklých pobočkách a rovněž vytvořila pevnou zaměstnaneckou základnu tvořenou vysoce kvalifikovanými zaměstnanci se specializací na trh, na kterém daná pobočka působí.

Dále společnost plánuje spustit první klientské projekty nabrané během expanze a tím získat první velké reference z nově vzniklých poboček. Rovněž se firma zaměří na budování strukturálních fondů, ze kterých bude možné financovat další investice.

7.1.1.6 Strategie pro rok 2022

V roce 2022 plánuje společnost zdokonalit své vlastní procesy a technické zázemí. Spíše než na nové projekty, by se firma ráda soustředila na revizi stávajících projektů, jejich zdokonalení a katapultování na špičku v jejich oboru.

Mimo to firma plánuje zaměřit se i na vlastní zaměstnance a jejich dovednosti. Chce tak investovat do jejich vzdělání, které věří, že se odrazí v nabízených službách. Spolu s tím však chce společnost nabitě znalosti šířit dál mezi odbornou veřejnost.

7.1.1.7 Strategie pro rok 2023

Rok 2023 by měl být rokem společenské zodpovědnosti. Firma plánuje zavést CSR (*Corporate Social Responsibility*) program a investovat do společensky prospěšných a významných projektů. V rámci programu firma plánuje pracovat i s bývalými či stávajícími klienty, pomoci jim založit vlastní CSR program a pracovat na zlepšení společnosti, ve které žijí nejen zaměstnanci, ale i klienti a koncoví spotřebitelé. To vše firma plánuje provést v souladu se svojí vizí.

7.1.1.8 Strategie pro rok 2024

Hlavním cílem pro rok 2024 je diverzifikace portfolia nabízených služeb. Společnost by ráda našla nové příležitosti a technologie a ty využila k rozšíření svých služeb. Mezi některé může patřit například Industry 5.0, kvantové

počítače, pokročilá aplikace umělé inteligence a počítačového vidění nebo i rozšíření k financování start-upů či risk managementu v rámci bankovního sektoru.

Společnost by tak opět měla nabídnout nové a inovativní služby, které jí pomohou udržet náskok před konkurencí.

7.1.2 Týmy

Modelová firma je tedy agenturou, která poskytuje služby z oblasti UX a UI designu, vývoje webových aplikací a vývoje mobilních aplikací. Uvnitř firmy je celkem 9 týmů:

1. Backend tým, který vyvíjí backend systémy a aplikační rozhraní,
2. Frontend tým, který vyvíjí interaktivní webové aplikace,
3. Android tým, který vyvíjí mobilní aplikace pro platformu Android,
4. iOS tým, který vyvíjí mobilní aplikace pro platformu iOS,
5. Design tým, který navrhuje UX a UI,
6. DevOps tým, který se stará o provoz serverů a infrastrukturu,
7. Quality assurance tým, který se stará o testování a
8. Projektový tým, který se stará o projektové řízení.
9. Analytický tým, který analyzuje business záměry klientů a pomáhá jim hledat nové a inovativní produkty.

Každý vývojový tým (1.–4.) má jednoho tým leadera a 9 členů. Ostatní týmy (5.–9.) mají jednoho tým leadera a 5 členů. Rozdělení na týmy je důležité, jelikož mnoho znalostí vzniká a je vlastněno kolektivně v rámci týmů. Znalosti vznikají primárně během projektů, při kterých členové týmu pracují na zakázkách pro klienty. Znalosti však mohou vznikat i během pravidelných týmových meetingů, při kterých se členové týmu scházejí, sdílejí *know-how*, diskutují nové postupy a technologie nebo řeší problémy na projektech, na kterých pracují.

7.1.3 Projekt

Na každém klientském projektu pracuje projektový tým. Ten si sestavuje projektový manažer pomocí takzvaných *pitchů*. Projektový manažer vydefiniuje, které pozice bude pro projekt potřebovat a kdy je bude potřebovat. Následně při pravidelném meetingu (*pitch*) se snaží vysvětlit o čem projekt je a jak bude probíhat. Pokud projekt některé zaměstnance zaujme a je pro jejich pozici vypsáno místo v týmu, mohou se přihlásit. Z přihlášených si nakonec

projektový manažer vybere tým, který na projektu bude pracovat a za který je zodpovědný. Do úvahy přitom bere zkušenosti zaměstnanců, schopnost spolupráce mezi jednotlivými zaměstnanci, ale i jejich aktuální vytížení na jiných projektech. Členy je možné v rámci projektových týmů měnit, avšak vždy musí souhlasit všichni zúčastnění.

Během projektu jsou pořádány pravidelné stand-upy, na kterých se sejdou všichni členové týmů a diskutují, co od poslední schůzky dělali, jak pokročili a na jaké problémy narazili. Meeting má za účel primárně sdílet zkušenosti, synchronizovat členy týmu mezi sebou a eliminovat řešení již vyřešených problémů. Četnost a podoba meetingů je v kompetenci projektového manažera.

Po každém dokončeném projektu následuje retrospektiva. Tedy schůzka, v rámci které se hodnotí průběh projektu, co se povedlo, co se nepovedlo a co by se dalo zlepšit. Retrospektiva je čas, kdy členové projektového týmu mezi sebou sdílí znalosti a zkušenosti. Retrospektiva může probíhat pravidelně již v průběhu projektu (například každý měsíc, po každém dokončeném sprintu atp.), avšak to je plně v kompetenci projektového manažera.

Zde i dále v práci se budu záměrně vyhýbat termínům spojeným s vývojem software (např. vývojáři, vyvíjí, programují atp.), neboť ne každý tým má v popisu práce vývoj software.

7.2 Analýza aktuálního stavu

Pro analýzu a návrh informačního systému využiji framework představený v sekci 3.2. První krok frameworku je komplexní analýza aktuálního stavu znalostí v organizaci.

V organizaci již dříve byly pokusy o zavedení znalostního managementu, které spočívaly primárně v implementaci znalostní databáze založené na systému wiki. Znalostní databáze neměla na začátku definovanou strukturu, a tak byly jednotlivé sekce přidávány náhodně bez většího rozmyslu. Aktuálně znalostní databáze obsahuje mnoho informací a znalostí. Většina z nich je již neudržovaných a mnoho z nich dokonce neaktuálních.

S postupující rozvolněnou strukturou začali mít zaměstnanci problém s vyhledáním relevantních informací a znalostí. Wiki systém obsahuje pouze jednoduché fulltextové vyhledávání, které pro potřeby uživatelů již nestačí. Kvůli tomu uživatelé přestávají znalostní databázi používat k vyhledávání znalostí. S postupným úpadkem používání souvisí i zastarávání obsahu. Uživatelé již nejsou stimulováni do systému nové poznatky vkládat či upravovat.

Tento problém vedl k tvorbě alternativních řešení. Každý tým má obvykle složku s textovými a tabulkovými dokumenty, do kterých své znalosti vkládá. Tyto složky jsou však těžko přístupné pro uživatele mimo daný tým. Složkám i dokumentům chybí jednotná struktura i pokročilé možnosti vyhledávání. Mimo jiné dokumenty nejsou uzpůsobeny pro kodifikaci tacitních znalostí.

Rovněž tento systém není možné využít pro znalosti a informace, které z principu mají být sdílené v rámci celé organizace.

Dalším problémem dokumentů je verzování. Sdílené dokumenty na cloudových úložištích typu Google Docs nebo Microsoft Office 365 mají implicitní verzování dokumentů, avšak není možné fulltextově vyhledávat v předchozích verzích.

K problému verzování se váže i problém soukromí. Někteří uživatelé nemají dostatečný pocit soukromí a jistoty při sdílení znalostí. Ať už se tato obava týká nepodloženosti informace či znalosti nebo formy, ve které je znalost předávána. Ve verzovacích systémech je typicky vidět každá změna, což některým uživatelům nevyhovuje.

Problémem je rovněž jazyk, ve kterém probíhá komunikace a ve kterém znalosti vznikají. Z aktuálního stavu organizace a její strategie vychází, že organizace působí na více trzích a do budoucna by ráda vstoupila na nové národní trhy. Spolu se vstupem na trh chce nabírat zaměstnance, kteří mluví místními jazyky a znají místní kulturu. Aktuálně však všechny znalosti vznikají a ukládají se v češtině.

Hlavním cílem zavedení znalostního managementu má být zpřístupnění znalostí pro všechny členy organizace. Znalosti by měly být dostupné a snadno vyhledatelné. Cílem není zvýšení efektivity práce. Organizace se chce více zaměřit na zvýšení kvality poskytovaných služeb. Výstupem by mohly být i nové knihovny či frameworky, které zvýší kvalitu a rychlost vývoje. Ty by následně mohly být publikovány například jako *open-source*.

Kvalifikační kritéria vychází především z aktivního používání systému. Za úspěch by tak bylo považováno

1. pokud se podaří převést znalosti z textových a tabulkových dokumentů do systému,
2. pokud průměrně každý uživatel týdně provede 10 hledání a
3. pokud průměrně každý uživatel měsíčně vloží jeden nový záznam.

7.3 Definice cílového stavu

Vizí organizace je svět, ve kterém se neplýtvá zdroji. A jelikož opakování stejných chyb a nevyužívání znalosti na maximum vede ke značnému plýtvání, organizace si uvědomuje důležitost správného zavedení znalostního managementu. Tento cíl tak má plnou podporu vedení.

Cílovým stavem je znalostní organizace, ve které znalosti volně plynou mezi jednotlivými uživateli bez jakýchkoli omezení. Kanály jsou jak centralizované, tak *peer-to-peer*. Důležité znalosti jsou zaznamenávány a jsou přístupné všem.

Organizace by ráda, aby vznikl jednotný systém, který bude sloužit všem zaměstnancům neohledně na jazyk nebo stát ve kterém pracují. Jednotný systém

by měl vyřešit problémy s ukládáním znalostí na několika místech a jejich následné nedostupnosti. Z tohoto cíle vychází první dva jasné požadavky na systém: 1. systém musí podporovat pokročilé možnosti vyhledávání a 2. systém musí být schopen ukládat všechny typy znalostí. Vyhledávání by mělo co nejintuitivnější a nejrychlejší. Uživatelé by měli mít možnost vyhledávat full-textově, podle týmů nebo podle stáří záznamu, a to napříč celým systémem včetně verzí záznamů.

Pro úspěch je zásadní angažovanost uživatelů. A pro tu je důležité, aby nevznikl čistě nový systém, na jehož používání bude nutné myslet. Implementace by měla spočívat především v integraci s ostatními nástroji, které týmy používají na denní bázi. Pro vývojové týmy by měla být připravena integrace s GitLabem, pro projektový tým integrace s Google Docs a pro všechny pak integrace se Slackem (případně jinými firemními komunikátory) a internetovými prohlížeči.

Angažovanost je plánováno podpořit i gamifikací. Ta by mohla obsahovat například *karma points* za plnění určitých úkolů, četnost používání nebo přidávání znalostí. Mimo uživatelů by mezi sebou mohly soupeřit i jednotlivé týmy. Co se gamifikace týče, všeobecně panuje konsenzus, že pravidla hry musí vycházet od uživatelů. Právě ti by si měli stanovit pravidla hry.

7.4 Návrh informačního systému

Třetí fází je návrh informačního systému. Ten bude navrhnout za použití metod softwarového inženýrství v kapitole 8.

7.5 Použité informační technologie

Jelikož se organizace z velké části zabývá vývojem zakázkového software, dává smysl si systém znalostního managementu vyvinout svépomocí. Toto rozhodnutí vychází především z přehledu dostupných řešení na trhu, které nevyhovují požadavkům organizace. Dalším důvodem tohoto rozhodnutí bylo vytvoření vlastního produktu, který bude nabízen klientům v rámci poradenské činnosti.

Celý systém bude navržený jako webově orientovaná aplikace skládající se ze webového frontendu a aplikačního backendu. Tyto dvě hlavní části budou doplňovat různé doplňky (například doplněk do prohlížeče nebo plugin pro Slack). V rámci dalšího rozvoje bude možné vyvinout aplikace pro mobilní telefony nebo tablety s operačními systémy Android a iOS.

7.5.1 Node.js

Organizace většinu moderních backend systémů implementuje v Node.js. Jedná se o běhové prostředí pro backend systémy psané v jazyce JavaScript. Node.js velmi často využívá asynchronní I/O operace, čímž minimalizuje režii

procesoru a snižuje náklady na aplikační servery. Tato technologie je *open-source* a lze jí použít zdarma. Rovněž všechny používané moduly jsou zdarma, nehrozí tedy nebezpečí v podobě velkého zpoplatnění některé ze služeb.

Systém bude nasazen do cloudového prostředí. Cloud computing je model využívání informačních technologií založený na poskytování služeb prostřednictvím internetu. Není tedy nutné vlastnit žádné servery, avšak pouze si je pronajímat takzvaně *on-demand*. Díky tomu, že aplikace není fyzicky přímo nasazena na konkrétní počítač, nedochází tak často k výpadkům. Platba poté probíhá dle spotřebovaných prostředků (například za vteřiny procesorového času nebo za GB operační paměti). A právě minimalizace režie procesorového času se promítne do výsledné ceny za spotřebované výpočetní prostředky.

7.5.2 Docker a Kubernetes

Většina moderních cloudových poskytovatelů funguje s technologií Docker, která zajišťuje kontejnerovou virtualizaci. Každá jednotlivá aplikace je zabalena do tohoto kontejneru a spuštěna. Uvnitř kontejneru je předem známé a definované prostředí, s vnějším komunikuje kontejner pouze pomocí sítě a zvolených portů.

Tento model je vhodný pro použití v cloudu z mnoha důvodů. Jedním z nich je oddělení procesů. To v praxi znamená, že skončí-li jeden program chybou nebo pádem aplikace, tento pád nijak negativně neovlivní aplikace běžící na stejném počítači. A právě tento model umožňuje sdílení výpočetních prostředků, které je základem cloud computingu. Podobně jako na klasickém počítači běží mnoho programů, které se střídají o prostředky, v cloudu běží několik aplikací na jednom počítači či clusteru počítačů a střídají se ve využívání prostředků.

Docker (a kontejnerizace všeobecně) umožňuje horizontální škálování rychlým spuštěním více instancí téže aplikace. K orchestraci kontejnerů se obvykle využívá Kubernetes. Tato technologie odstiňuje kontejnery od jednotlivých počítačů (výpočetních uzlů) a chová se jakoby se jednalo o jeden stroj. Dále pak umožňuje automatický *load-balancing* spuštěním nových instancí nebo *zero-downtime deployment*, tedy nasazení nové verze bez jakéhokoli výpadku. Pokud dojde k selhání instance, Kubernetes automaticky přestane posílat požadavky na danou instanci. Následně kontejner smaže a automaticky nastartuje nový, který opět zapojí do clusteru. Kubernetes typicky podporuje většina cloudových poskytovatelů (Google Cloud Platform, Amazon Web Services nebo Microsoft Azure).

Je dobré zmínit, že provozovat aplikace s perzistentním úložištěm uvnitř Docker kontejnerů není nejvhodnější volba.

7.5.3 React a Redux

Frontendovou aplikaci budou pohánět knihovny React a Redux. Redux je knihovna, která se stará o stav JavaScriptové aplikace. V klasickém architektuře *Model-View-Controller* představuje model. Redux bude poté doplněn Reactem (který se stará o *view* část).

Aby bylo možné vůbec tyto technologie použít (například kvůli *code splittingu*), vše bude pomocí Webpacku zabaleno do jednoho JavaScriptového balíčku, který následně bude distribuován. Webpack se pak postará i o další nutné úkony, kterými jsou například mimifikace, obfuskace případně transpilace.

Většina technologií byla vybrána na základě dostupnosti a ceny. Všechny technologie již organizace někdy použila a má vhodné zaměstnance pro vývoj v těchto technologiích. S určitými změnami v návrhu by pro vývoj backend systému posloužil například jazyk PHP (s vybraným frameworkem) nebo jazyk Ruby (s frameworkem Ruby on Rails).

7.6 Zhodnocení návrhu

V páté fázi jde především o řízení evolučního potenciálu a zhodnocení, zda je systém připravený na případně změny v organizaci. Toto zhodnocení proběhne v sekci 7.6.

Návrh informačního systému

Tato kapitola přináší návrh systému znalostního managementu, který bude implementován a použit v organizaci představené v sekci 7.1. Návrh systému se řídí zásadami softwarového inženýrství. První jsou rozebrány funkční a nefunkční požadavky na systém. Následně jsou představeni uživatelé systému a jeho architektura. Nakonec jsou prezentovány detailní případy užití.

8.1 Funkční požadavky

Tato sekce představuje funkční požadavky, které uživatelé na systém mají. Požadavky jsou konkrétní, kompletní a závazné pro implementaci.

F1 Přihlášení do aplikace pomocí firemního Google účtu

8.1.1 Profil

F2 Výpis profilu se základními informacemi o uživateli (firemní identita, sledované tagy, týmy atp.)

F3 Úprava základních informací (sledovaných tagů, týmů atp.)

F4 Výpis uživatelových statistik (karma points, získané badges atp.)

F5 Výpis informací o týmu (zařazení do týmu, karma points týmu atp.)

F6 Změna jazyka systému

8.1.2 Přehled

F7 Zobrazení aktivity (nových a upravených znalostí) za zvolené časové období

F8 Zobrazení aktivity spojené se sledováním

8.1.3 Vložení a detail znalosti

- F9 Vložení nové znalosti
- F10 Výpis detailu vložené znalosti
- F11 Prohlížení předchozích verzí znalosti
- F12 Obnovení znalosti do předchozí verze
- F13 Okomentování znalosti
- F14 Návrh nové verze znalosti (kterou musí schválit majitel znalosti či komunita)

8.1.4 Q&A

- F15 Vložení nové otázky
- F16 Vložení odpovědi na otázku
- F17 Vložení komentáře k otázce
- F18 Hodnocení otázky (například palec nahoru nebo palec dolů)
- F19 Označení otázky jako zodpovězené
- F20 Vytvoření nové znalosti z otázky a odpovědi

8.1.5 Integrace

- F21 Vložení záznamu přímo z komunikačního nástroje Slack
- F22 Vložení záznamu z webových prohlížečů Google Chrome a Mozilla Firefox
- F23 Automatické fulltextové vyhledávání při vyhledávání na Google.com ve vybraných prohlížečích
- F24 Vyhledávání přímo v rozhraní nástroje Slack
- F25 Odpovědět na zprávu ve Slacku znalostí ze systému

8.2 Nefunkční požadavky

Tato sekce představuje nefunkční požadavky na navrhovaný software. Část těchto požadavků odpovídá čtvrté fázi vývoje informačního systému znalostního managementu ze sekce 3.2. Především pak nefunkční požadavky týkající se vybraných technologií.

- N1 Doplnky budou podporovat moderní prohlížeče Google Chrome verze 68 a vyšší a Mozilla Firefox verze 60 a vyšší.
- N2 Využití MariaDB a MongoDB databáze
- N3 Implementace backend systému v technologii Node.js
- N4 Implementace frontend v technologiích React a Redux
- N5 Nasazení pomocí technologie Docker v cloudovém prostředí (například Google Cloud Platform)
- N6 Distribuce doplňku pro komunikátor Slack skrze oficiální Slack Marketplace
- N7 Distribuce doplňků do prohlížečů skrze oficiální obchody pro dané prohlížeče

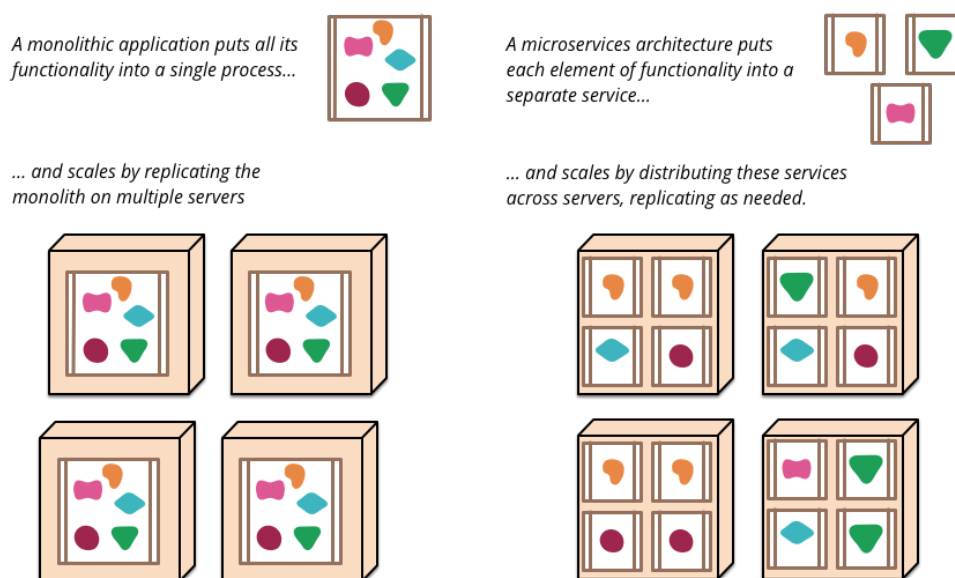
8.3 Architektura

Již dříve jsem představil nástin architektury informačního systému. Ten by měl být webově orientován a všechny informace poskytovat skrze globální síť internet.

8.3.1 Microservice architektura

Hlavní částí informačního systému bude backend systém, který se stará o zpracování, ukládání a servírování dat. Backend systém jsem navrhl pomocí vzoru mikroslužeb (*microservice architecture pattern*). Hlavní myšlenkou je rozdělit celý informační systém na komponenty, které mají jeden jediný účel. Tento účel však plní dobře a poskytují ho jako službu jiným komponentám informačního systému. Ten pak vznikne propojením jednotlivých komponent skrze aplikační rozhraní.

Snaha rozdělit kód do více komponent byla v softwarovém inženýrství již dlouho. Začaly vznikat knihovny, které poskytovaly funkcionalitu. Program vznikl linkováním knihoven dohromady. Pokud program potřebuje funkci knihovny, zavolá funkci v paměti. Služby naproti tomu dále budou používat knihovny, avšak v dělení půjdou ještě dál. Služba bude plně oddělena od programu a bude jí možné využít skrze webový request nebo vzdálené volání procedury (RPC). [48]



Obrázek 81: Monolitická a microservice architektura [48]

Hlavní výhodou mikroslužeb je znovupoužitelnost jednotlivých služeb. Ty jsou vždy samostatné a nejsou nijak vázané na ostatní služby v informačním systému. Služby je možné používat i pro jiné projekty nebo přímo prodávat zákazníkům.

Výhodou je právě nízká závislost mezi službami. Nedostupnost jedné služby by neměla ohrozit provoz ostatních služeb. Tedy například nedostupnost autentizační služby by neměla zabránit již autentizovaným uživatelům v používání informačního systému.

Architektura je vhodná i pro škálování. Je totiž možné škálovat jednotlivé služby dle potřeby. Nikdy není nutné škálovat celý systém, ačkoli přestává stačit pouze některá jeho část. Nejlépe tento fakt popisuje dnes již legendární schéma tvůrce architektury mikroslužeb Martina Fowlera na obrázku 81.

Tato architektura přináší výhody i v oblastech programování. Každá z komponent má vlastní zdrojový kód (*codebase*). Je tedy snazší se ve zdrojovém kódu zorientovat a díky menšímu počtu závislostí je jednodušší do zdrojového kódu zasahovat. Vývojáři mohou pracovat na různých komponentách systému současně a nehrozí, že budou vznikat konflikty při slučování jejich kódu.

Výhodou je i technologická nezávislost jednotlivých služeb. Pro každou službu může být vybrána jiná technologie, která je pro implementaci nejvhodnější. Například bankovní instituce dlouhodobě vyvíjejí systémy a knihovny v Javě. Každá mikroslužba tedy může mít vlastní JVM plně nezávislou od ostatních komponent. Ty navíc nemusí být ani vyvíjeny v Javě. Například

Tabulka 81: Uživatelé systému

Uživatel	Popis aktéra
Běžný uživatel	Reálný uživatel systému (zaměstnanec), který má právo prohlížet všechny veřejné záznamy, přidávat záznamy a zobrazovat statistiky.
Admin	Admin systému, který má všechny práva jako běžný uživatel. Navíc má právo prohlížet všechny uživatele, měnit jejich zařazení do týmu, mazat kterékoli záznamy a zobrazovat neviditelné záznamy.

gateway, která funguje jako vstupní bod do aplikace a rozděljuje požadavky na ostatní služby, může být postavena na Node.js, který se dlouhodobě ukazuje jako nejrychlejší a nejlevnější variantou pro rozesílání požadavků a pro jednoduché zpracování dat.

Mezi nevýhody patří složitý proces nasazení a následná správa běžících instancí. Tento problém je však minimalizován použitím Docker kontejnerů běžících v cloudovém prostředí. Na jejich nasazení je možné připravit Continuous Integration infrastrukturu, která složitý proces nasazení dokáže automatizovat.

Další nevýhodou je nutnost psát více kódu znovu, aby například všechny služby byly schopné obsluhovat requests, načítat konfigurace atp. Tento problém je však možné minimalizovat tvorbou šablony, ze které následně všechny služby vychází.

8.3.2 Architektura systému

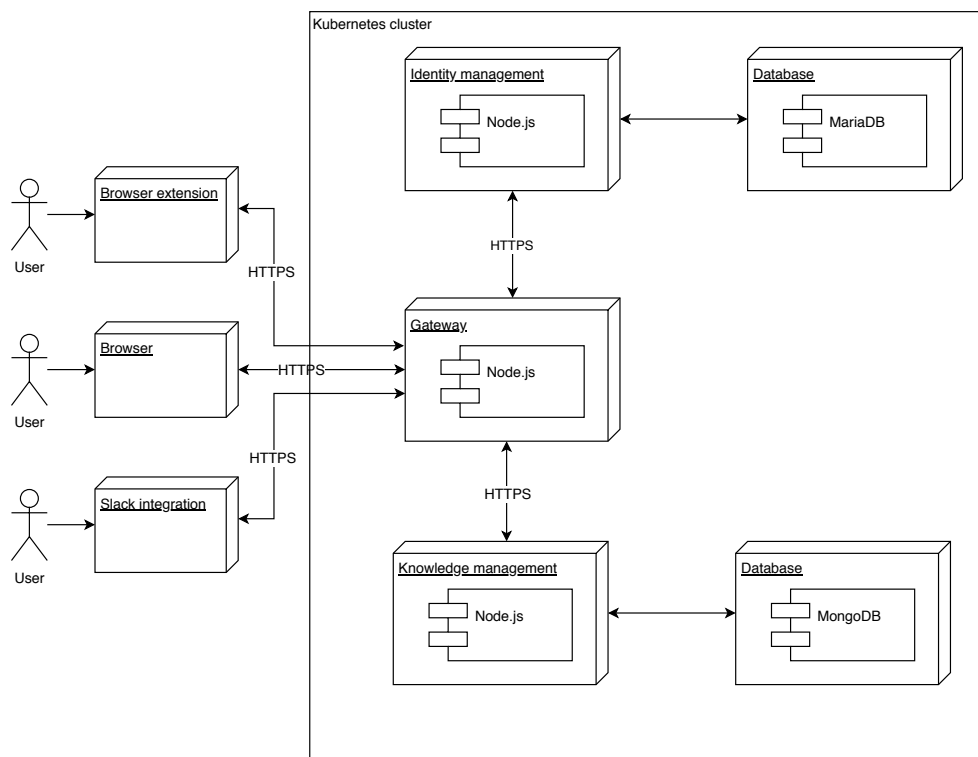
Konkrétní architekturu navrhovaného systému ukazuje obrázek 82. Systém jsem rozdělil do 3 mikroslužeb. První je *Identity management service*, která se stará o autentizaci a autorizaci uživatelů. Má vlastní databázi MariaDB, do které ukládá nezbytné informace o uživatelích.

Dále pak Gateway, která funguje jako vstupní bod a rozesílá požadavky do jednotlivých mikroslužeb. Gateway je bezstavová a nemá vlastní databázi. Každé ověření provádí v koordinaci s Identity management service.

Třetí mikroslužbou je Knowledge management service, která se stará o ukládání a distribuci znalostních záznamů. Pro ukládání těchto záznamů využívá NoSQL databázi MongoDB.

8.4 Aktéři

Aby bylo možné přistoupit k popisu případů užití (*use-cases*, viz sekce 8.5), je nutné definovat aktéry. Aktéři jsou uživatelé, kteří se systémem budou interagovat. Aktéry a jejich práva shrnuje tabulka 81.



Obrázek 82: Architektura navrhovaného systému

8.5 Případy užití

Tato sekce postupně představuje detailní případy užití systému. Všechny případy užití shrnuje i diagram případů užití na obrázku 83.

8.5.1 UC1 - Přihlášení

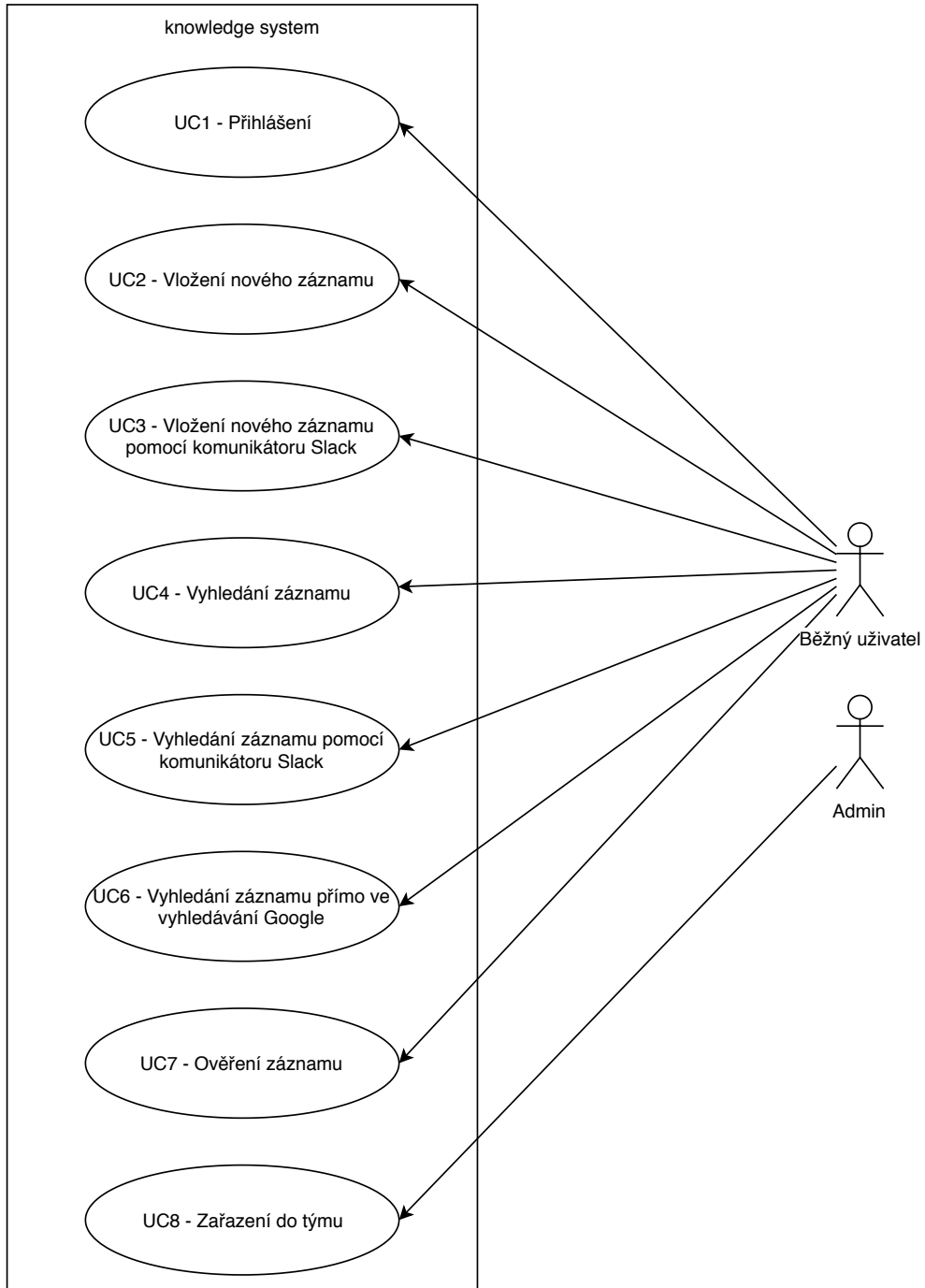
Hlavní účastník: Běžný uživatel

Rozsah: Znalostní systém

Úroveň: Uživatelský cíl

Zúčastněné strany a zájmy:

- Uživatel – Přihlásit se.
- Gateway – Přiřadit požadavek správné mikroslužbě.
- Identity management service - Autentizace uživatele.



Obrázek 83: Use case diagram

- Google OAuth server - Autentizace uživatele.
- Databáze

Předpoklady:

- Funkční spojení mezi frontendovou aplikací a gateway.
- Funkční spojení mezi mikroslužbami.

Minimální plnění: Uživatel získá zpět informaci o výsledku.

Úspěšné plnění: Uživatel získá zpět informaci o výsledku a je přihlášen.

Spouštěč: Kliknutí na tlačítko Login with Google na stránce přihlášení (viz obrazovka B1).

Scénář úspěšného plnění:

1. Uživatel klikne na tlačítko Login with Google.
2. Frontend aplikace zašle požadavek na OAuth servery Google.
3. Uživatel vyplní nebo vybere požadovaný e-mail a heslo.
4. Frontend aplikace od OAuth serveru Google obdrží jednorázový token.
5. Identity management service vymění jednorázový token za přístupový token.
6. Identity management service autentizuje uživatele.
7. Frontend aplikace přihlásí uživatele.

Alternativní scénář:

- E-mail nebo heslo na straně OAuth serverů nebylo validní. Uživatel je vyzván k vyplnění správného e-mailu nebo hesla.
- Daný uživatel není oprávněn přistupovat k systému. Uživatel je vyzván k vyplnění správného e-mailu nebo hesla.

Výjimky:

- Nepodařilo se navázat spojení s některou z mikroslužeb nebo OAuth servery. Uživateli je zobrazena chybová hláška s výzvou k přidání záznamu později.

8.5.2 UC2 - Vložení nového záznamu

Hlavní účastník: Běžný uživatel

Rozsah: Znalostní systém

Úroveň: Uživatelský cíl

Zúčastněné strany a zájmy:

- Uživatel – Vložit záznam.
- Frontendová aplikace – Umožnit uživatelům požadavek.
- Gateway – Přiřadit požadavek správné mikroslužbě.
- Knowledge service – Uložit záznam do databáze.
- Databáze

Předpoklady:

- Funkční spojení mezi frontendovou aplikací a gateway.
- Funkční spojení mezi mikroslužbami.
- Přihlášený uživatel.

Minimální plnění: Uživatel získá zpět informaci o výsledku.

Úspěšné plnění: Uživatel získá zpět informaci o výsledku a záznam je vložen do databáze.

Spouštěč: Kliknutí na tlačítko Přidat na stránce nového záznamu (viz obrazovka B4).

Scénář úspěšného plnění:

1. Uživatel vyplní povinné údaje ve formuláři.
2. Frontend aplikace odešle request na Gateway.
3. Gateway v koordinaci Identity management service zkontroluje, je-li uživatel přihlášen.
4. Gateway přesměruje požadavek na Knowledge service.
5. Knowledge service zapíše nový záznam do databáze.

6. Knowledge service vrátí výsledek.
7. Frontendová aplikace uživateli ukáže nový záznam.

Alternativní scénář:

- Některý z údajů nebyl validní, systém uživateli zobrazí chybovou hlášku a vyzve ho k opravě.

Výjimky:

- Uživatel není přihlášen: Uživatel je přesměrován na přihlašovací stránku a je mu zobrazena výzva k přihlášení.
- Nepodařilo se navázat spojení s některou z mikroslužeb. Uživateli je zobrazena chybová hláška s výzvou k přidání záznamu později.

8.5.3 UC3 - Vložení nového záznamu pomocí komunikátoru Slack

Hlavní účastník: Běžný uživatel

Rozsah: Znalostní systém

Úroveň: Uživatelský cíl

Zúčastněné strany a zájmy:

- Uživatel – Vložit záznam.
- Slack extension – Zařídít vložení záznamu.
- Slack app – Autorizovat požadavek.
- Gateway – Přiřadit požadavek správné mikroslužbě.
- Knowledge service – Uložit záznam do databáze.
- Databáze

Předpoklady:

- Funkční spojení mezi Slack extension a gateway.
- Funkční spojení mezi mikroslužbami.
- Autentizovaný uživatel v Slack extension.

Minimální plnění: Uživatel získá zpět informaci o výsledku.

Úspěšné plnění: Uživatel získá zpět informaci o výsledku a záznam je vložen do databáze.

Spouštěč: Kliknutí na tlačítko Přidat ve Slacku.

Scénář úspěšného plnění:

1. Uživatel vybere zprávu, ze které chce vytvořit záznam.
2. Slack app v koordinaci se Slack extension zobrazí formulář pro vyplnění dodatečných informací.
3. Uživatel vyplní údaje a odešle formulář.
4. Slack extension v koordinaci se Slack app ověří zprávu a pošle payload na Gateway.
5. Gateway v koordinaci Identity management service zkontroluje, je-li uživatel přihlášen.
6. Gateway přeměruje požadavek na Knowledge service.
7. Knowledge service zapíše nový záznam do databáze.
8. Knowledge service vrátí výsledek.
9. Slack extension informuje uživatele o výsledku.

Alternativní scénář:

- Některý z údajů nebyl validní, Slack extension uživateli zobrazí chybovou hlášku a vyzve ho k opravě.

Výjimky:

- Uživatel není přihlášen: Uživatel je vyzván k propojení Slack extension se znalostním systémem.
- Nepodařilo se navázat spojení s některou ze služeb. Uživateli je zobrazena chybová hláška s výzvou k přidání záznamu později.

8.5.4 UC4 - Vyhledání záznamu

Hlavní účastník: Běžný uživatel

Rozsah: Znalostní systém

Úroveň: Uživatelský cíl

Zúčastněné strany a zájmy:

- Uživatel – Vyhledat záznam.
- Frontendová aplikace – Umožnit uživateli požadavek a zobrazit výsledek.
- Gateway – Přiřadit požadavek správné mikroslužbě.
- Knowledge service – Vybrat vhodné záznamy z databáze.
- Databáze

Předpoklady:

- Funkční spojení mezi frontendovou aplikací a gateway.
- Funkční spojení mezi mikroslužbami.
- Přihlášený uživatel.

Minimální plnění: Uživatel získá zpět informaci o výsledku.

Úspěšné plnění: Uživatel získá zpět informaci o výsledku a hledané záznamy.

Spouštěč: Kliknutí na tlačítko Hledat na kterékoli stránce (například obrazovka B2), případně upravení parametrů hledání na stránce s výsledky (viz obrazovka B3).

Scénář úspěšného plnění:

1. Uživatel vyplní hledanou frázi a případně další kritéria hledání.
2. Gateway v koordinaci Identity management service zkontroluje, je-li uživatel přihlášen.
3. Gateway přesměruje požadavek na Knowledge service.
4. Knowledge service vrátí vhodné záznamy z databáze.
5. Frontendová aplikace uživateli prezentuje vhodné záznamy.

Výjimky:

- Uživatel není přihlášen: Uživatel je přesměrován na přihlašovací stránku a je mu zobrazena výzva k přihlášení.
- Nepodařilo se navázat spojení s některou z mikroslužeb. Uživateli je zobrazena chybová hláška s výzvou k opakování akce později.

8.5.5 UC5 - Vyhledání záznamu pomocí komunikátoru Slack

Hlavní účastník: Běžný uživatel

Rozsah: Znalostní systém

Úroveň: Uživatelský cíl

Zúčastněné strany a zájmy:

- Uživatel – Vyhledat záznamy.
- Slack extension – Zařídít vyhledání záznamů.
- Slack app – Autorizovat požadavek a zobrazit záznamy.
- Gateway – Přiřadit požadavek správné mikroslužbě.
- Knowledge service – Vybrat vhodné záznamy z databáze.
- Databáze

Předpoklady:

- Funkční spojení mezi Slack extension a gateway.
- Funkční spojení mezi mikroslužbami.
- Autentizovaný uživatel v Slack extension.

Minimální plnění: Uživatel získá zpět informaci o výsledku.

Úspěšné plnění: Uživatel získá zpět informaci o výsledku a hledané záznamy.

Spouštěč: Použití příkazu */knowledge [hledaná klíčová fráze]* ve kterékoli konverzaci.

Scénář úspěšného plnění:

1. Uživatel ve kterékoli konverzaci napíše */knowledge* s hledanou frází a odešle.
2. Slack extension v koordinaci se Slack app ověří zprávu a pošle payload na Gateway.
3. Gateway v koordinaci Identity management service zkontroluje, je-li uživatel přihlášen.
4. Gateway přesměruje požadavek na Knowledge service.
5. Knowledge service vrátí vhodné záznamy z databáze.
6. Slack extension prezentuje widget se záznamy ze znalostního systému.

Alternativní scénář:

- Nebylo možné nalézt žádné odpovídající záznamy. Systém uživatele spraví o nemožnosti nalézt záznamy.

Výjimky:

- Uživatel není přihlášen: Uživatel je vyzván k propojení Slack extension se znalostním systémem.
- Nepodařilo se navázat spojení s některou ze služeb. Uživateli je zobrazena chybová hláška s výzvou k přidání záznamu později.

8.5.6 UC6 - Vyhledání záznamu přímo ve vyhledávání Google

Hlavní účastník: Běžný uživatel

Rozsah: Znalostní systém

Úroveň: Funkce

Zúčastněné strany a zájmy:

- Uživatel – Vyhledat záznam.
- Browser extension – Zařídít vyhledání záznamů a zobrazit je.
- Gateway – Přiřadit požadavek správné mikroslužbě.
- Knowledge service – Vybrat vhodné záznamy z databáze.
- Databáze

Předpoklady:

- Funkční spojení mezi Browser extension a Gateway.
- Funkční spojení mezi mikroslužbami.
- Autentizovaný uživatel v Browser extension.

Minimální plnění: Uživatel získá zpět informaci o výsledku.

Úspěšné plnění: Uživatel získá zpět informaci o výsledku a hledané záznamy.

Spouštěč: Vyhledání záznamů na stránce Google.com

Scénář úspěšného plnění:

1. Uživatel vyplní hledanou frázi na Google.com
2. Browser extension zašle požadavek na Gateway.
3. Gateway v koordinaci Identity management service zkontroluje, je-li uživatel přihlášen.
4. Gateway přesměruje požadavek na Knowledge service.
5. Knowledge service vrátí vhodné záznamy z databáze.
6. Browser extension zobrazí výsledky v SERP Google.com.

Výjimky:

- Uživatel není přihlášen: Uživatel je vyzván k propojení Browser extension se znalostním systémem.
- Nepodařilo se navázat spojení s některou ze služeb. Uživateli je zobrazena nemodifikovaná SERP.

8.5.7 UC7 - Ověření záznamu

Hlavní účastník: Běžný uživatel

Rozsah: Znalostní systém

Úroveň: Uživatelský cíl

Zúčastněné strany a zájmy:

- Uživatel – Ověřit záznam.
- Frontendová aplikace – Umožnit uživatelův požadavek.
- Gateway – Přiřadit požadavek správné mikroslužbě.
- Identity management service - Autentizace uživatele.
- Knowledge service – Zapsat výsledek do databáze.
- Databáze

Předpoklady:

- Funkční spojení mezi frontendovou aplikací a gateway.
- Funkční spojení mezi mikroslužbami.

Minimální plnění: Uživatel získá zpět informaci o výsledku.

Úspěšné plnění: Uživatel získá zpět informaci o výsledku a záznam je ověřen.

Spouštěč: Kliknutí na tlačítko Ověřit v detailu záznamu (viz obrazovka B10) nebo v panelu ověření (viz obrazovka B9).

Scénář úspěšného plnění:

1. Uživatel zkontroluje záznam a klikne na tlačítko Ověřit.
2. Frontend aplikace odešle request na Gateway.
3. Gateway v koordinaci Identity management service zkontroluje, je-li uživatel přihlášen.
4. Gateway přesměruje požadavek na Knowledge service.
5. Knowledge service zkontroluje, že uživatel je skutečně ověřitelem.
6. Knowledge service aktualizuje záznam.
7. Knowledge service vrátí výsledek.
8. Frontendová aplikace uživateli ukáže zkontrolovaný záznam.

Alternativní scénář:

- Některý z údajů nebyl validní. Systém uživateli zobrazí chybovou hlášku a vyzve ho k opravě.
- Uživatel nemá právo verifikovat záznam. Systém uživatele spraví o nemožnosti provést akci.

Výjimky:

- Uživatel není přihlášen: Uživatel je přesměrován na přihlašovací stránku a je mu zobrazena výzva k přihlášení.
- Nepodařilo se navázat spojení s některou z mikroslužeb. Uživateli je zobrazena chybová hláška s výzvou k ověření záznamu později.

8.5.8 UC8 - Zařazení do týmu

Hlavní účastník: Admin

Rozsah: Znalostní systém

Úroveň: Funkce

Zúčastněné strany a zájmy:

- Uživatel – Zařadit uživatele do týmu.
- Frontendová aplikace – Umožnit uživatelův požadavek.
- Gateway – Přiřadit požadavek správné mikroslužbě.
- Identity management service - Autentizace uživatele.
- Databáze

Předpoklady:

- Funkční spojení mezi frontendovou aplikací a gateway.
- Funkční spojení mezi mikroslužbami.

Minimální plnění: Uživatel získá zpět informaci o výsledku.

Úspěšné plnění: Uživatel získá zpět informaci o výsledku a uživatel je zařazen do týmu.

Spouštěč: Výběr týmu ze select boxu v detailu uživatele.

Scénář úspěšného plnění:

1. Uživatel v select boxu v detailu uživatele vybere tým.
2. Frontend aplikace odešle request na Gateway.
3. Gateway v koordinaci Identity management service zkontroluje, je-li uživatel přihlášen a zda má právo akci provést.
4. Identity management service změní zařazení uživatele do týmu a vrátí výsledek.
5. Frontendová aplikace ukáže nový tým uživatele.

Alternativní scénář:

- Uživatel nemá právo měnit zařazení uživatelů do týmů.

Výjimky:

- Uživatel není přihlášen: Uživatel je přesměrován na přihlašovací stránku a je mu zobrazena výzva k přihlášení.
- Nepodařilo se navázat spojení s některou z mikroslužeb. Uživateli je zobrazena chybová hláška s výzvou k ověření záznamu později.

8.6 Gamifikace

V sekci 4.5 jsem rozebíral základní teorii pro gamifikaci informačních systémů a aplikací. Tyto principy jsem využil i pro návrh informačního systému znalostního managementu.

Základní mechanikou celého systému je bodový systém. Konkrétní bodové zisky nejsou pro návrh systému zajímavé, zajímavé jsou spíše bodové rozdíly. Bodové zisky za jednotlivé splněné úkoly přehledně prezentuje tabulka 82.

Jako podpůrný prvek pro bodový systém jsem vybral leaderboardy. Základní leaderboard, který ukazuje bodový rozdíl mezi jednotlivými týmy, je umístěn na dashboardu (B2) a v detailu profilu (B6). Doplněn je leaderboardem jednotlivých uživatelů.

V detailu profilu je rovněž měření aktivity za sledované období. Aktivita je sledována pro celý tým i pro jednotlivé členy týmu.

Ačkoli popularita odznaků již dosáhla vrcholu a jejich použití se může zdát sporné, jejich využití pro onboarding stále vnímám jako pozitivní. Odznaky jsou v systému tedy skutečně použity primárně pro onboarding. Mají za úkol uživateli ukázat základní funkce systému (přidání znalostního záznamu,

Tabulka 82: Bodové zisky

Úkol	Bodový zisk
První přihlášení	100
První přihlášení daný den	10
Položení otázky	50
Odpověď na otázku	100
Vybraná odpověď na otázku	200
Nový znalostní záznam	200
Ověřený znalostní záznam	200
Ověření znalostního záznamu	100
Up-vote u záznamu/otázky	100
Mimořádná týmová aktivita	100

přidání otázky, odpověď na otázku, verifikace atp.), nikoli ho motivovat k používání systému.

8.7 Možná vylepšení

Systém byl navrhován pro konkrétní typ organizace. A ačkoli byl navržen vysoce škálovatelný, obsahuje některé prvky, které jsou přímo vázané na fungování typové organizace. Pokud by tedy systém měl sloužit i jiným organizacím (které se například nedělí na jednoznačně definované týmy), vyžadoval by návrh drobné úpravy.

Dále návrh konkrétně nediskutuje zacházení s textem a jeho formátování. Pro tuto potřebu je však možné využít již některý z existujících systémů (například *wiki syntax*).

Zajímavým vylepšením by mohlo být automatická indexace multimediálních souborů. Znalosti se totiž mohou ukrývat například ve zvukových záznamech nebo videozáznamech z prezentací, schůzek atp. Tyto informace a znalosti v systému aktuálně nejsou indexovány dle svého obsahu, ale pouze dle metadat, která k souboru poskytne autor. Tyto metadata však mohou být nedostatečná pro plnou indexaci záznamu.

Rovněž systém aktuálně nemůže sloužit jako *e-learning* platforma. Ta by mohla být využita pro vzdělávání stávajících zaměstnanců, ale i pro snazší *onboarding* nových zaměstnanců. Přidat tuto funkcionalitu je díky modulárnímu pojetí systému možné. Platforma by mohla být vyvinuta jako nová mikroslužba, jejíž funkcionalita bude zasazena do existujícího UI frontendové aplikace.

V budoucnu může organizace narazit na potřebu použití informačního systému na mobilních zařízeních (mobilních telefonech a tabletech). Aktuálně je UI systému připraveno na responzivní design (viz B12), který umožňuje pohodlné použití na mobilních zařízeních. Časem však možná přijde potřeba

využívat informační systém na mobilních zařízeních intenzivněji. Proto bude do budoucna možné vytvořit nativní mobilní aplikace pro systémy Android i iOS. K tomu bude možné využít REST API, kterým aktuálně komunikuje frontendová aplikace s backendovou aplikací.

8.8 Zhodnocení návrhu

Pátou a poslední fází frameworku je zhodnocení navrženého systému. Systém byl navržen za pomoci architektonického vzoru mikroslužeb. Je tedy modulární a snadno rozšiřitelný. Pro implementaci systému byly vybrány vhodné a moderní technologie. Nebude tedy vznikat technologický dluh již ve stádiu vývoje systému. Díky populárním technologiím by měl být systém snadno udržovatelný. Do budoucna je možné systém snadno rozšiřovat o novou funkcionalitu. Díky jednoduchému REST API je systém propojitelný s jinými systémy. Rovněž je nad ním možné stavět další aplikace, které budou konzumovat obsah uchovávaný ve znalostním systému.

Systém plně respektuje všechnu teorii probranou v kapitolách 2 a 3. Měl by tedy odpovídat nejnovějším trendům v oblasti řízení znalostí. Navíc byl systém plně navržen pro potřeby modelové organizace. Systém má velký evoluční potenciál a po drobných úpravách a zapracování první zpětné vazby z používání systému, je možné systém nabídnout dalším organizacím.

Nebyly identifikovány žádné závažné problémy, které by znamenaly návrat do předchozí fáze a úpravu navrhovaného systému.

Závěr

Cílem práce bylo představit znalostní management, základní pojmy a nástroje. Dále pak rozebrat roli informačních systémů jako nástrojů pro uchování znalostí. Ukázat, jaké informační systémy existují, jak se navrhují a jak se vyvíjí. Práce měla za cíl vybrat typové příklady firem z IT segmentu a doporučit informační systém pro řízení znalostí. Posledním krokem byla analýza a návrh informačního systému znalostního management pro typovou IT organizaci. Všechny tyto cíle byly naplněny.

V teoretické části jsem čtenáře seznámil se znalostním managementem, s rozdílem mezi daty, informacemi a znalostmi. Představil jsme taxonomii znalostí a modely znalostních procesů. Vše jsem dal do kontrastu strategie organizace. Dále jsem rozebral systémy znalostního managementu a představil framework pro jejich návrh. Přidal jsem i praktická doporučení pro jejich návrh, které sloužily k objasnění principů použitých v praktické části.

V praktické části jsem se věnoval řešerši existujících řešení a následnému výběru typových IT organizací. Těmto organizacím jsem doporučil vhodné řešení z úvodní řešerše. Nakonec jsme představil modelovou organizaci z IT segmentu, zanalyzoval její požadavky na systém a pomocí frameworku z teoretické části navrhl informační systém znalostního managementu. Návrh jsem doplnil o UX design.

Osobním přínosem práce bylo prohloubení mých znalostí v oblasti znalostního managementu. Během práce jsem kompletně přehodnotil pohled, kterým jsem se na znalosti dříve díval. Rovněž jsem zjistil, jak velkou roli hrají informace a znalosti v dnešní společnosti a jaký je jejich význam v 21. století. Vyzkoušel jsem si návrh komplexního informačního systému, nad kterým jsem dostal volnou ruku. Na jednom projektu jsem si také mohl vyzkoušet mnoho pozic, které typicky zastávají různí lidé, což mě profesně obohatilo.

Literatura

- [1] Kay, R.; Cecez-Kecmanovic, D.: Towards an Autopoietic Perspective on Knowledge Management. 01 2002.
- [2] Doležal, J.: CYNEFIN pomůže určit, kdy aplikovat agile. [online], [cit. 2019-02-06]. Dostupné z: <https://www.pmconsulting.cz/2018/01/cynefin-pomuze-urcit-aplikovat-agile/>
- [3] Brdička, B.: Foggův model ovládání lidí technologiemi. [online], [cit. 2019-03-14]. Dostupné z: <https://clanky.rvp.cz/clanek/c/Z/21910/fogguv-model-ovladani-lidi-technologie.html>
- [4] Fowler, M.: Microservices. [online], [cit. 2019-03-12]. Dostupné z: <https://martinfowler.com/articles/microservices.html>
- [5] Grant, R. M.: Prospering in Dynamically-Competitive Environments: Organizational Capability as Knowledge Integration. *Organization Science*, ročník 7, č. 4, 1996: s. 375–387, ISSN 10477039, 15265455. Dostupné z: <http://www.jstor.org/stable/2635098>
- [6] Alavi, M.; Leidner, D. E.: Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. *MIS Quarterly*, ročník 25, č. 1, 2001: s. 107–136, ISSN 02767783. Dostupné z: <http://www.jstor.org/stable/3250961>
- [7] Ackoff, R. L.: From data to wisdom. *Journal of applied systems analysis*, ročník 16, č. 1, 1989: s. 3–9.
- [8] Lombardi, O.; Holik, F.; Vanni, L.: What is Shannon information? *Synthese*, ročník 193, č. 7, 2016: s. 1983–2012.
- [9] Shannon, C. E.: A Mathematical Theory of Communication. *Bell System Technical Journal*, ročník 27, č. 3: s. 379–423, doi:10.1002/j.1538-7305.1948.tb01338.x. Dostupné z: <https://onlinelibrary.wiley.com/doi/abs/10.1002/j.1538-7305.1948.tb01338.x>

- [10] Bellinger, G.; Castro, D.; Mills, A.: Data, information, knowledge, and wisdom. 2004. Dostupné z: <http://www.systems-thinking.org/dikw/dikw.htm>
- [11] Gettier, E.: Is Justified True Belief Knowledge? *Analysis*, ročník 23, č. 6, 1963: s. 121–123.
- [12] Vance, D.: Information, Knowledge and Wisdom: The Epistemic Hierarchy and Computer-Based Information Systems. 1997. Dostupné z: <https://aisel.aisnet.org/amcis1997/165/>
- [13] Sharma, N.: The Origin of Data Information Knowledge Wisdom (DIKW) Hierarchy. 04 2008.
- [14] Polanyi, M.: *Personal Knowledge: Towards a Post-Critical Philosophy*. The University of Chicago Press, 1958, ISBN 9780226232768.
- [15] Ein-Dor, P.: *Taxonomies of Knowledge*. 01 2006, 848-854 s., doi:10.4018/978-1-59904-931-1.ch142.
- [16] Bureš, V.: *Znalostní management a proces jeho zavádění-průvodce pro praxi*. Grada Publishing as, 2007.
- [17] Wiig, K. M.: Knowledge management: where did it come from and where will it go? *Expert systems with applications*, ročník 13, č. 1, 1997.
- [18] Wiig, K. M.: Knowledge management: an emerging discipline rooted in a long history. *Knowledge horizons: the present and the promise of knowledge management*, 2000.
- [19] Rubenstein-Montano, B.; Liebowitz, J.; Buchwalter, J. J.; aj.: A systems thinking framework for knowledge management. *Decision Support Systems*, ročník 31, č. 1, 2001: s. 5 – 16, ISSN 0167-9236, doi: [https://doi.org/10.1016/S0167-9236\(00\)00116-0](https://doi.org/10.1016/S0167-9236(00)00116-0).
- [20] Dibella, A. J.; Nevis, E. C.: *How organizations learn: an integrated strategy for building learning capability*. Jossey-Bass San Francisco, California, 1998, ISBN 0787911070.
- [21] O'Dell, C.: A current review of knowledge management best practice. *Conference on knowledge management and the transfer of best practices*, 1996.
- [22] David M. Steier, D. I. K., Scott B. Huffman: Beyond Full-text Search: AI-Based Technology to Support the Knowledge Cycle. In *AAAI Technical Report SS-97-01*, mar 1997.

-
- [23] Alavi, M.; Leidner, D. E.: Knowledge Management Systems: Issues, Challenges, and Benefits. *Commun. AIS*, ročník 1, č. 2es, 1999. Dostupné z: <http://dl.acm.org/citation.cfm?id=374116.374117>
- [24] Drucker, P. F.: Knowledge-worker productivity: The biggest challenge. *California management review*, ročník 41, č. 2, 1999: s. 79–94.
- [25] Drucker, P. F.: *Management Challenges for the 21st Century*. Routledge, 2007, ISBN 978-0-7506-8509-2.
- [26] Maier, R.; Hadrich, T.: *Knowledge management systems*. IGI Global, 2011.
- [27] Nonaka, I.: The knowledge-creating company. *California management review*, ročník 69, č. 6, 1991: s. 96–104.
- [28] Albrecht, A. J.: Measuring application development productivity. *Proc. of IBM Application Development Symposium*, October 1979: s. 83–92.
- [29] Ramírez, Y. W.; Nembhard, D. A.: Measuring knowledge worker productivity: A taxonomy. *Journal of Intellectual Capital*, ročník 5, č. 4, 2004: s. 602–628, doi:10.1108/14691930410567040.
- [30] Truneček, J.: *Management znalostí*. Nakladatelství CH Beck, 2004, ISBN 80-7179-884-3.
- [31] Wilson, T. D.: The nonsense of knowledge management. *Information research*, ročník 8, č. 1, 2002.
- [32] Woodman, M.; Zade, A.: Five grounded principles for developing knowledge management systems. *Electronic Journal of Knowledge Management*, ročník 10, č. 2, 2012: s. 183–194.
- [33] Dourish, P.: What we talk about when we talk about context. *Personal and ubiquitous computing*, ročník 8, č. 1, 2004: s. 19–30.
- [34] Snowden, D. J.; Boone, M. E.: A leader’s framework for decision making. *Harvard business review*, ročník 85, č. 11, 2007: str. 68.
- [35] Zack, M. H.: Developing a Knowledge Strategy. *California Management Review*, ročník 41, č. 3, 1999: s. 125–145, doi:10.2307/41166000.
- [36] Koster, R.: *Theory of fun for game design*. O’Reilly Media, Inc., 2013.
- [37] Marovt, J.: *Gamification of software applications*. Dizertační práce, Univerza v Ljubljani, 2012.
- [38] Fogg, B. J.: A behavior model for persuasive design. In *Proceedings of the 4th international Conference on Persuasive Technology*, ACM, 2009, str. 40.

- [39] Hunicke, R.; LeBlanc, M.; Zubek, R.: MDA: A formal approach to game design and game research. In *Proceedings of the AAAI Workshop on Challenges in Game AI*, ročník 4, 2004, str. 1722.
- [40] Atlassian: Confluence Server REST API. [online], [cit. 2019-03-26]. Dostupné z: <https://developer.atlassian.com/server/confluence/confluence-server-rest-api/>
- [41] Atlassian: Confluence Licensing — Atlassian. [online], [cit. 2019-03-26]. Dostupné z: <https://www.atlassian.com/licensing/confluence>
- [42] Zendesk: Enabling SAML single sign-on (Professional and Enterprise). [online], [cit. 2019-03-30]. Dostupné z: <https://support.zendesk.com/hc/en-us/articles/203663676-Enabling-SAML-single-sign-on-Professional-and-Enterprise>
- [43] Zendesk: API Docs - Zendesk APIs - Zendesk Developer Portal. [online], [cit. 2019-03-30]. Dostupné z: https://developer.zendesk.com/rest_api/docs/zendesk-apis/resources
- [44] Bloomfire: Bloomfire API V2. [online], [cit. 2019-03-28]. Dostupné z: <http://api-docs-v2.bloomfire.info/>
- [45] Bloomfire: Bloomfire Pricing - Bloomfire. [online], [cit. 2019-03-28]. Dostupné z: <https://bloomfire.com/pricing/>
- [46] Guru: Guru – API Documentation. [online], [cit. 2019-03-29]. Dostupné z: <https://help.getguru.com/categories/api-documentation>
- [47] OpenKM: OpenKM 6.4 SSO Configuration. [online], [cit. 2019-04-02]. Dostupné z: <https://docs.openkm.com/kcenter/view/okm-6.4/sso-configuration.html>
- [48] Evropská komise: Nařízení komise (ES) č. 800/2008. 2008. Dostupné z: <https://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2008:214:0003:0047:cs:PDF>

Seznam použitých zkratek

API Application Programming Interface

CEO Chief executive officer

CRUD Create, Read, Update, Delete

IoT Internet of Things

JWT JSON Web Token

LDAP Lightweight Directory Access Protocol

PHP Hypertext Preprocessor

Q&A Questions and Answers

REST Representational State Transfer

RPC Remote procedure call

SaaS Software as a Service

SAML Security Assertion Markup Language

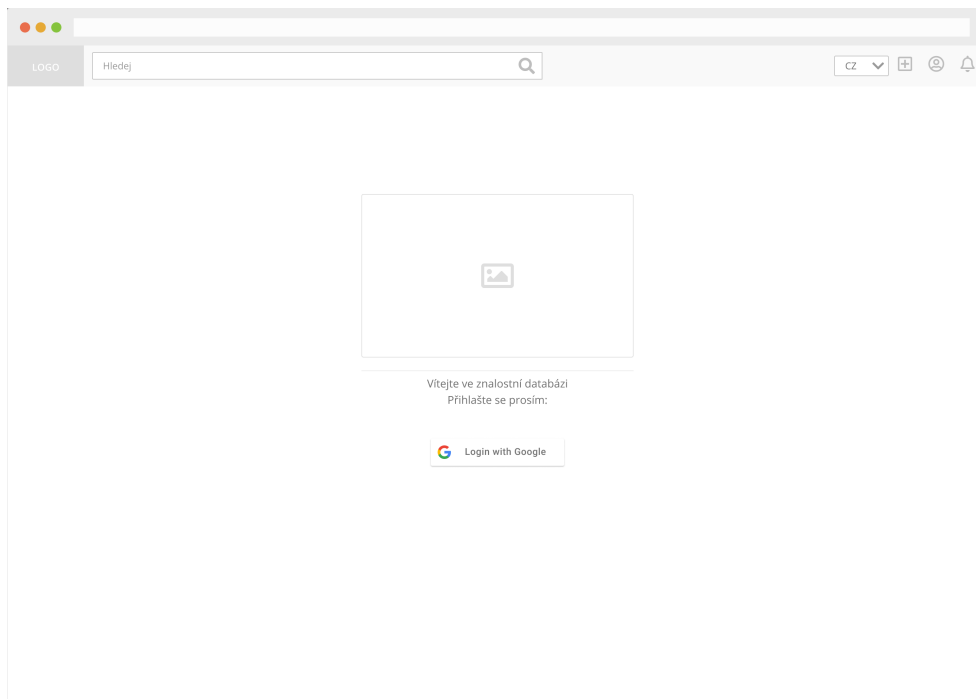
SERP Search engine result page

SSO Single sign on

UI User Interface

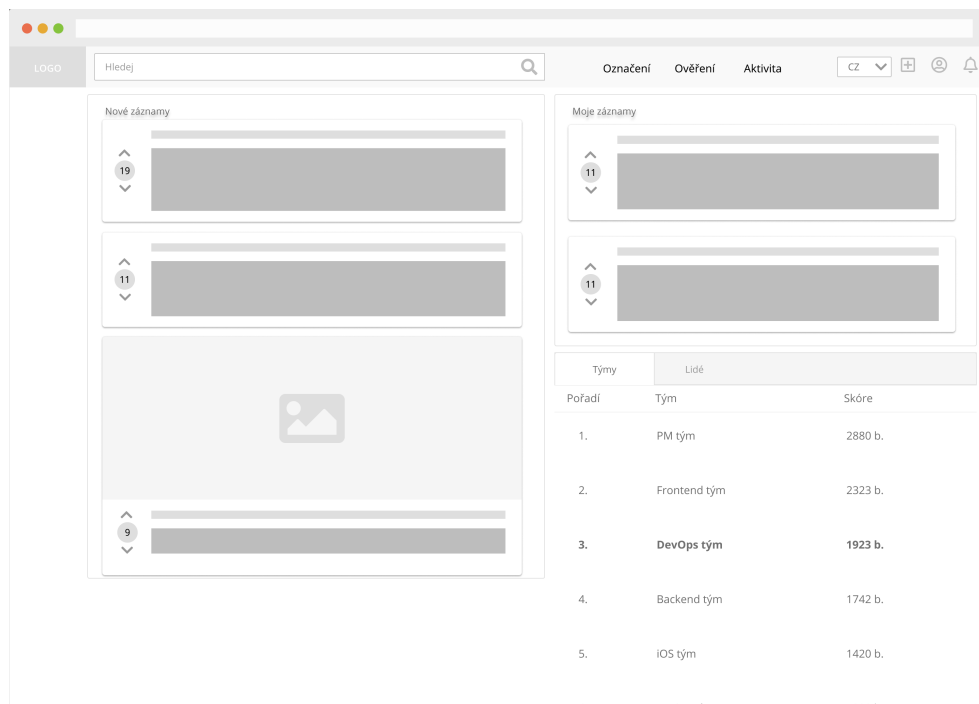
Návrh uživatelského rozhraní

Tato příloha obsahuje návrh všech obrazovek systému. Kompletní návrh včetně možných stavů na jednotlivých obrazovkách je možné nalézt na přiloženém médiu.

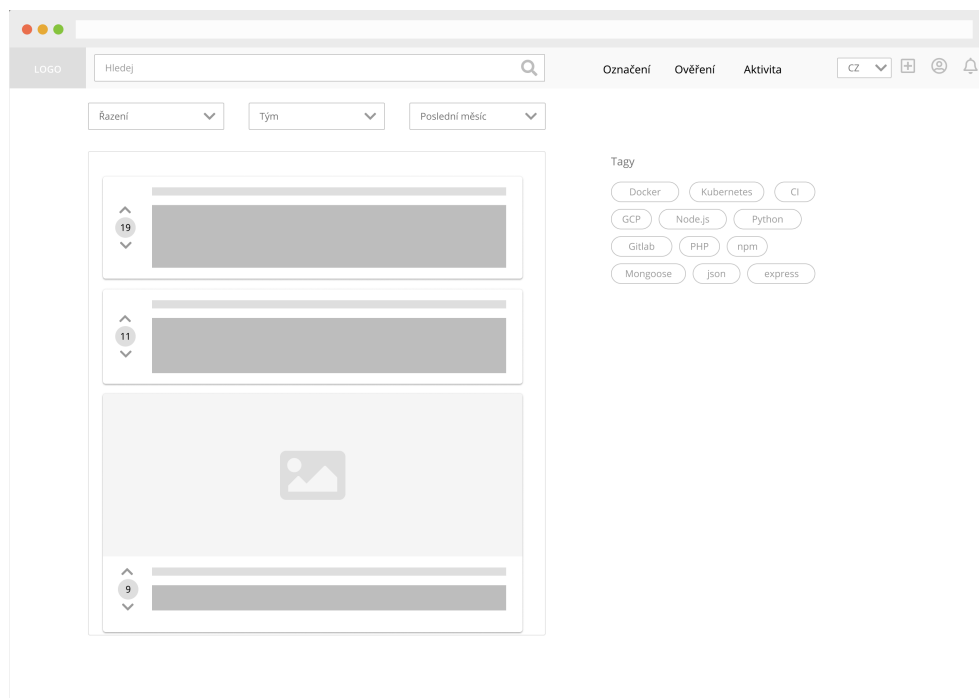


Obrázek B1: Přihlášení do systému

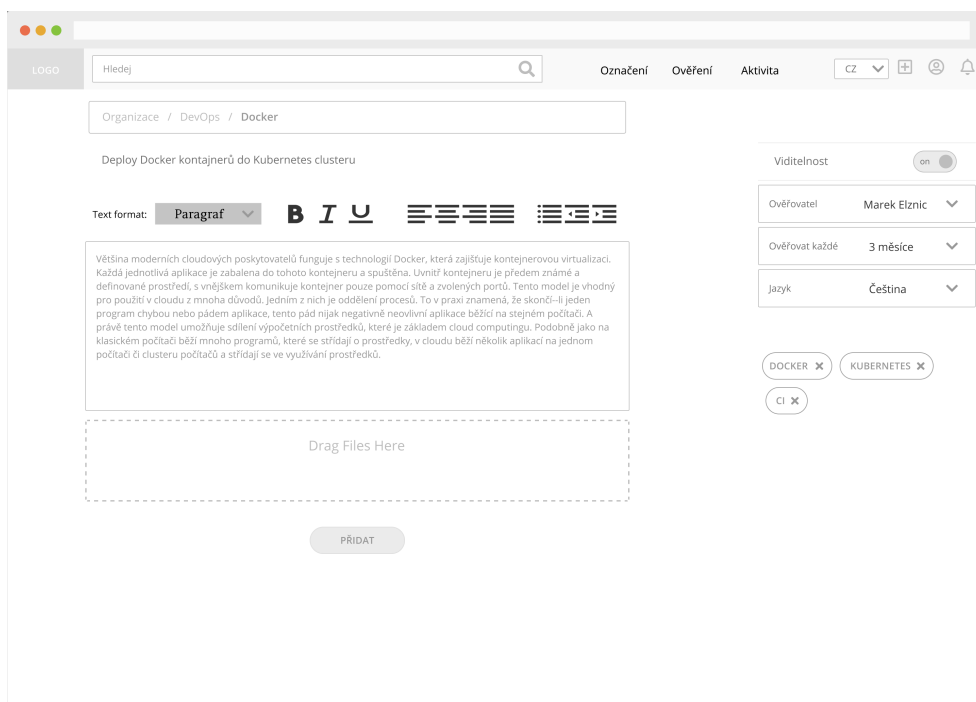
B. NÁVRH UŽIVATELSKÉHO ROZHRAVNÍ



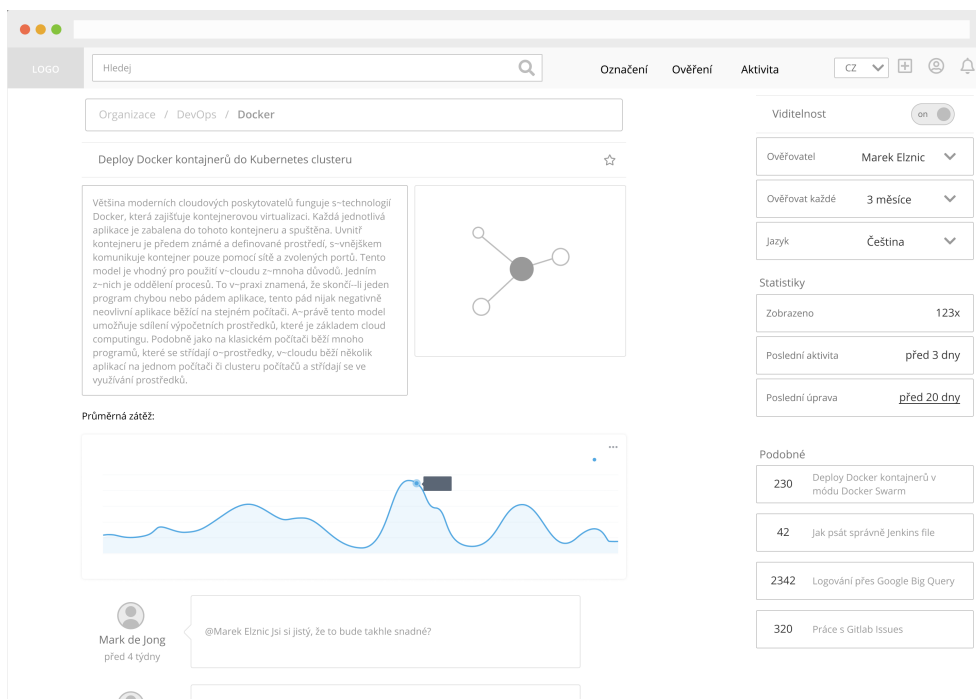
Obrázek B2: Dashboard



Obrázek B3: Dashboard

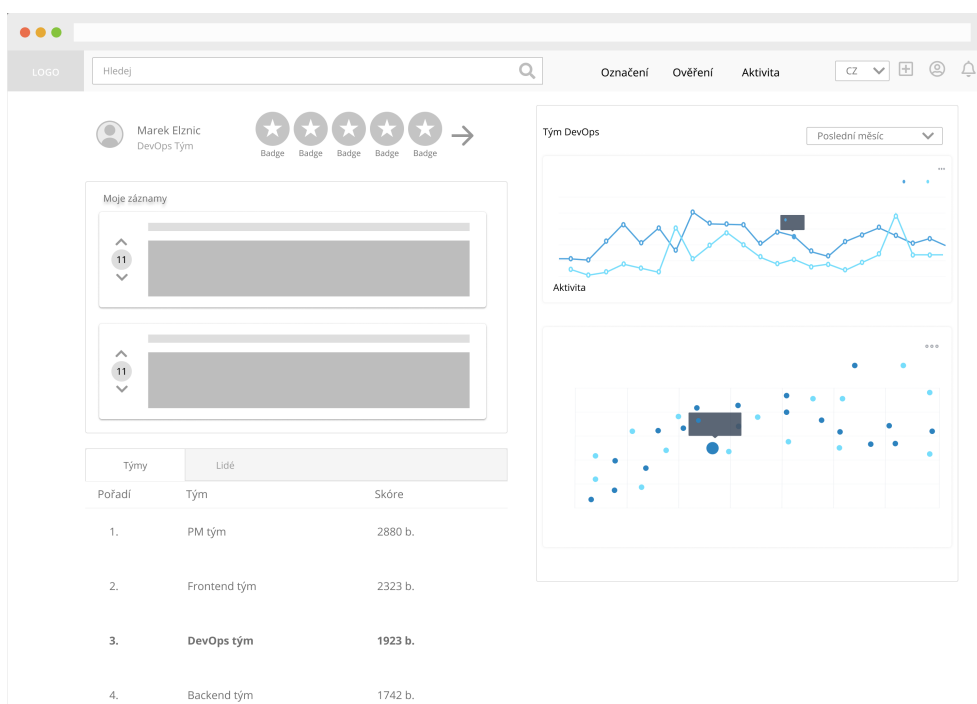


Obrázek B4: Nový záznam

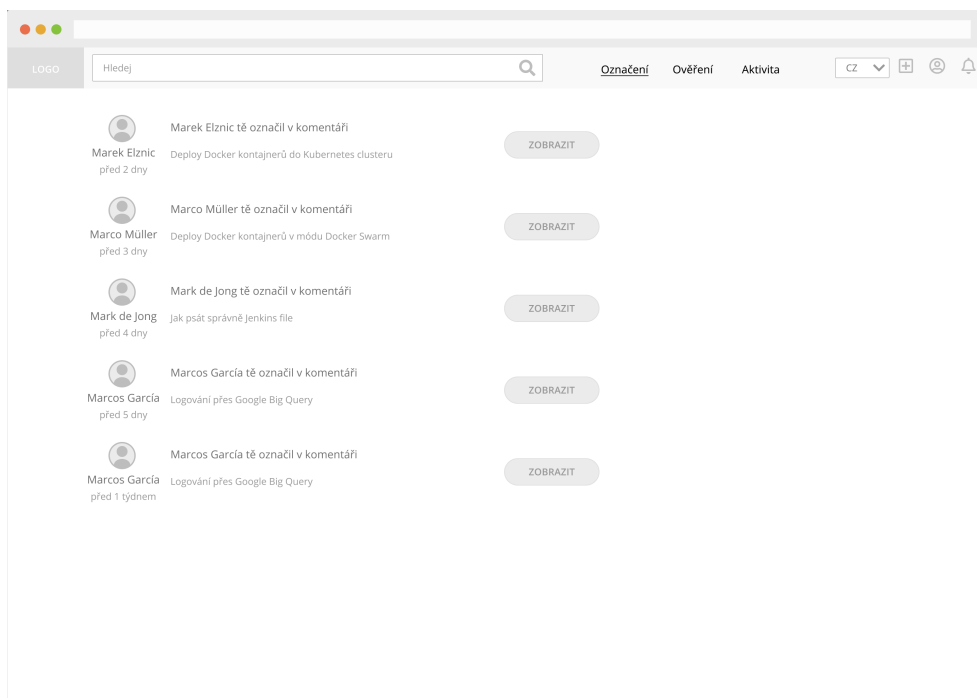


Obrázek B5: Detail záznamu

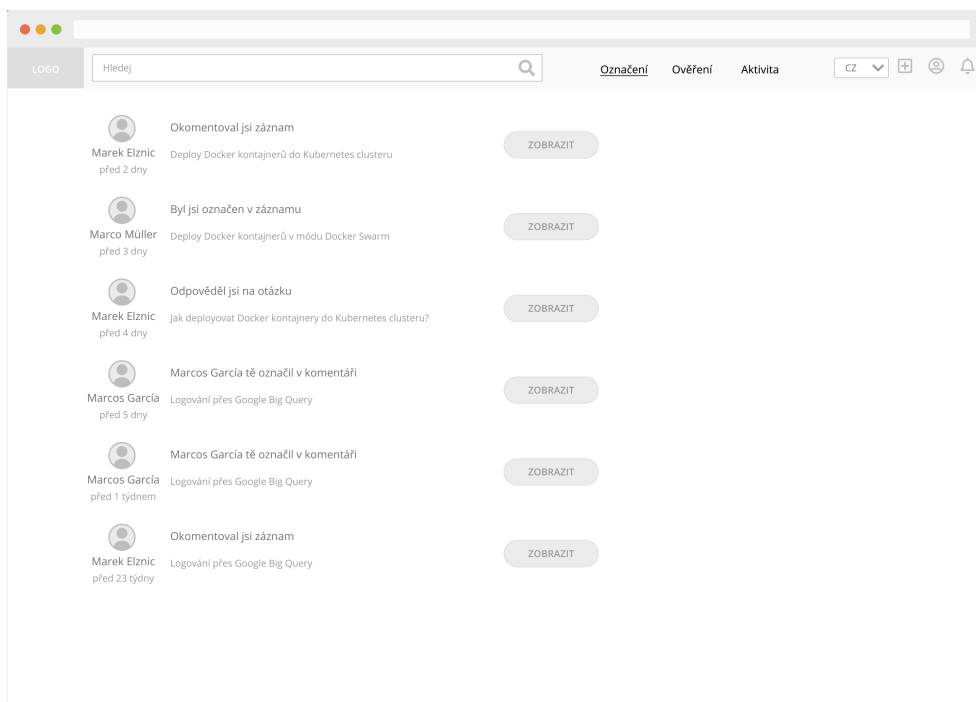
B. NÁVRH UŽIVATELSKÉHO ROZHRAVNÍ



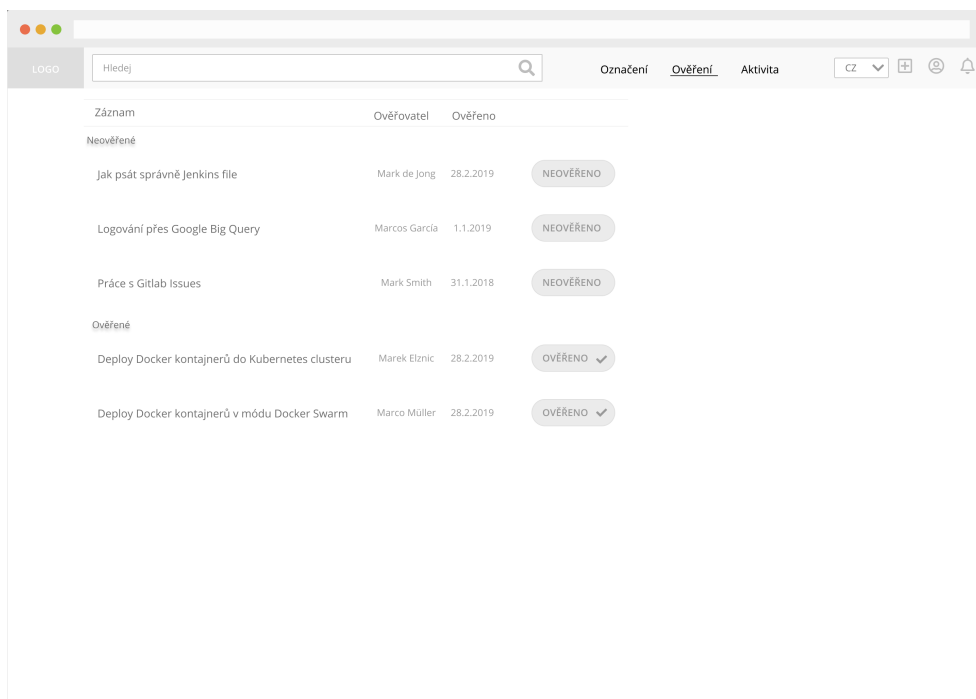
Obrázek B6: Profil



Obrázek B7: Přehled označení

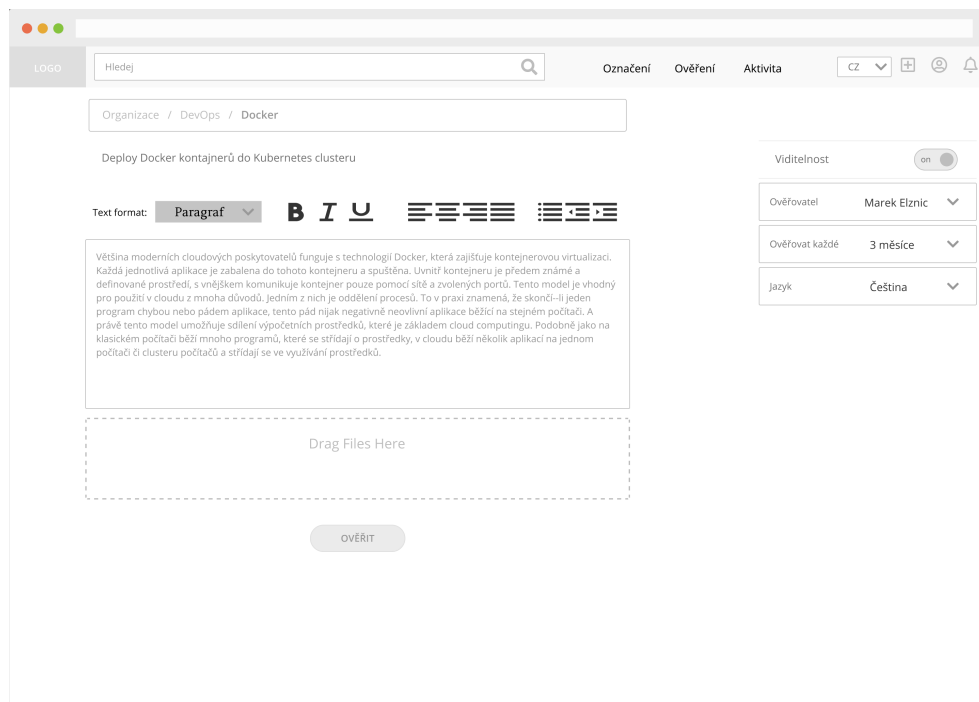


Obrázek B8: Feed poslední aktivity

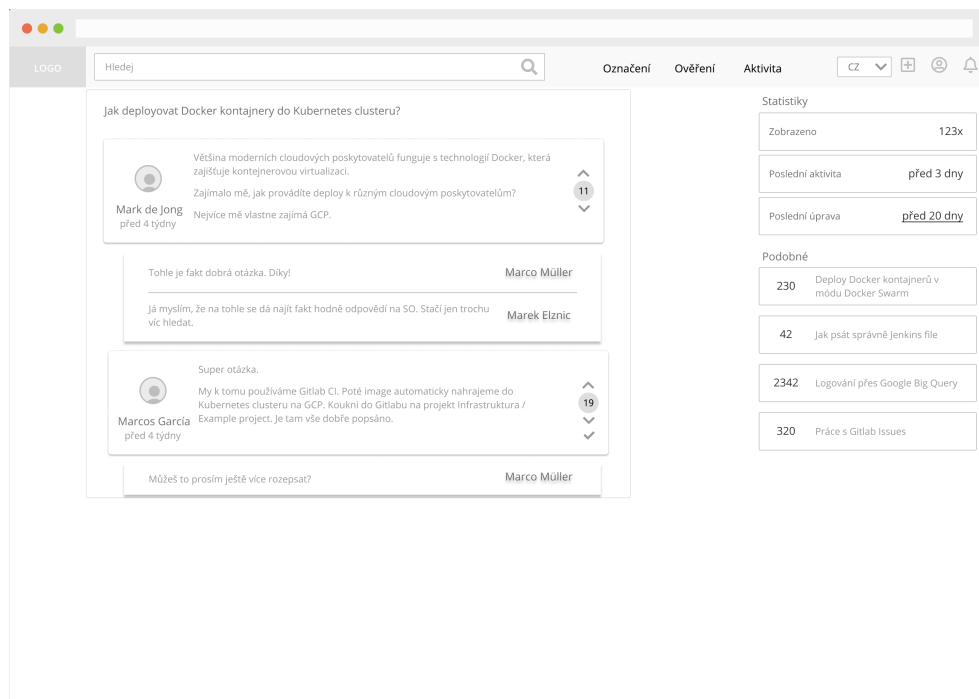


Obrázek B9: Záznamy, pro které je přihlášený uživatel ověřovatel

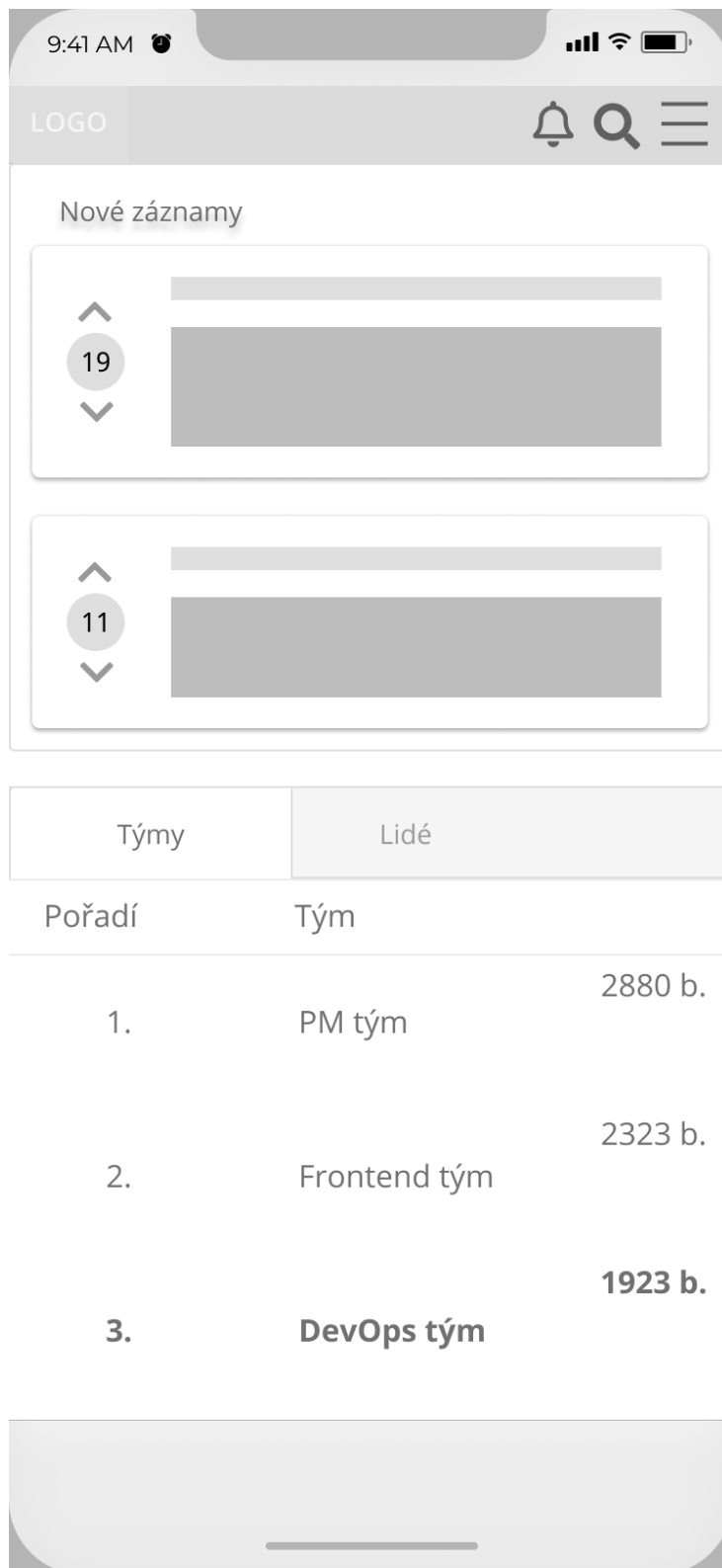
B. NÁVRH UŽIVATELSKÉHO ROZHRAVNÍ



Obrázek B10: Ověření záznamu



Obrázek B11: Detail expertní otázky



Obrázek B12: Responzivní verze Dashboardu

Obsah přiloženého CD

	readme.txt	stručný popis obsahu CD
	UX	návrh obrazovek znalostního systému
	source	
	thesis	zdrojová forma práce ve formátu $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
	text	text práce
	thesis.pdf	text práce ve formátu PDF