



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA PODNIKATELSKÁ**

FACULTY OF BUSINESS AND MANAGEMENT

**ÚSTAV INFORMATIKY**

INSTITUTE OF INFORMATICS

**NÁVRH A IMPLEMENTACE CRM SYSTÉMU  
POMOCÍ LOW-CODE PLATFORMY VE FIREMNÍM  
PROSTŘEDÍ**

DESIGN AND IMPLEMENTATION OF CRM SYSTEM USING LOW-CODE PLATFORM IN  
THE CORPORATE ENVIRONMENT

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. Josef Obrtlík**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. Lukáš Novák, Ph.D.**

**BRNO 2021**

# Zadání diplomové práce

Ústav: Ústav informatiky  
Student: **Bc. Josef Obtlík**  
Studijní program: Systémové inženýrství a informatika  
Studijní obor: Informační management  
Vedoucí práce: **Ing. Lukáš Novák, Ph.D.**  
Akademický rok: 2020/21

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

## **Návrh a implementace CRM systému pomocí low-code platformy ve firemním prostředí**

### **Charakteristika problematiky úkolu:**

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

### **Cíle, kterých má být dosaženo:**

Cílem práce je analyzovat, navrhnout a implementovat CRM systém pomocí softwaru SystemBaker do firemního prostředí a zhodnotit ekonomické přínosy.

### **Základní literární prameny:**

GÁLA, L., J. POUR a Z. ŠEDIVÁ. Podniková informatika. 2. přeprac. a aktualiz. vyd. Praha: Grada Publishing, 2009. ISBN 978-80-247-2615-1.

HARDCASTLE, E. Business Information Systems. Ventus Publishing ApS, 2008. ISBN 978-87-76-1-463-2.

LAUDON, K. a J. P. LAUDON. Management Information Systems: Managing the digital firm. 12th ed. New York: Prentice Hall, 2012. ISBN 978-01-357-9093-9.

MAHEY, H. Robotic Process Automation with Automation Anywhere: Techniques to fuel business productivity and intelligent automation using RPA. Birmingham: Packt Publishing, 2020. ISBN 978--83921-565-0.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2020/21

V Brně dne 28.2.2021

L. S.

---

Mgr. Veronika Novotná, Ph.D.  
ředitel

---

doc. Ing. Vojtěch Bartoš, Ph.D.  
děkan

## **ABSTRAKT**

Diplomová práce se zaměřuje na vytvoření funkčního CRM systému ve webovém rozhraní aplikace SystemBaker ve firemním prostředí – konkrétně pro firmu Mathesio s.r.o. V teoretické části jsou vysvětleny pojmy a technologie, které je třeba znát k úplnému porozumění pozadí vyvíjené aplikace. Poté jsou představeny možné formy vývoje dané aplikace, které slouží pro pozdější zhodnocení ekonomických přínosů vývoje. V praktické části je popsána databázová struktura, případy užití a funkční řešení dané aplikace a následně ekonomické zhodnocení přínosů stylu tohoto vývoje.

## **KLÍČOVÁ SLOVA:**

CRM, Byznys Procesy, BPR, RPA, Workflow, WFM systém, Low-code, SystemBaker

## **ABSTRACT**

This master's thesis is focusing to create an operational CRM system in web interface of SystemBaker application in company environment – specifically for company named Mathesio s.r.o. In theoretical part there are explained concepts and technologies, which are needed to know to fully comprehend background of the developed application. Then there are introduced possible forms of developing this application, which serve for later assessment of economic benefits of this development form. In practical part there is described structure of database, use-cases and operational solution of this application and then economical assessment of benefits of this style of development.

## **KEYWORDS:**

CRM, Business Process, BPR, RPA, Workflow, WFM system, Low-code, SystemBaker

*Per aspera ad astra*

***Bibliografická citace***

OBRTLÍK, Josef. Návrh a implementace CRM systému pomocí low-code platformy ve firmním prostředí [online]. Brno, 2021 [cit. 2021-05-15]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/133706>. Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Lukáš Novák.

# OBSAH

<b>ÚVOD</b> .....	<b>8</b>
<b>1 VYMEZENÍ PROBLÉMU A CÍLE PRÁCE</b> .....	<b>10</b>
1.1 VYMEZENÍ PROBLÉMU .....	10
1.2 CÍLE PRÁCE .....	10
<b>2 TEORETICKÁ VÝCHODISKA</b> .....	<b>11</b>
2.1 INFORMAČNÍ SYSTÉM .....	11
2.1.1 Komponenty informačního systému .....	11
2.1.2 Role informačního systému.....	14
2.2 CRM.....	14
2.2.1 Jak CRM pracuje.....	15
2.2.2 Výhody CRM .....	16
2.2.3 Dopady implementace.....	17
2.3 BYZNYS PROCESY.....	18
2.3.1 Dokumentace procesů .....	18
2.3.2 Reinženýring podnikových procesů .....	19
2.3.3 Techniky BPM .....	20
2.4 ROBOTIC PROCESS AUTOMATION.....	23
2.4.1 Jak RPA funguje .....	23
2.4.2 Výhody.....	24
2.4.3 Obecné využití .....	24
2.5 WORKFLOW .....	26
2.5.1 Workflow Management System.....	26
2.5.2 Definice procesu.....	31
2.5.3 Relevantní a aplikační workflow data.....	33
2.6 LOW-CODE .....	33
2.6.1 Výhody.....	34
2.7 VYTVOŘIT VS. KOUPIŤ .....	35
2.8 SYSTEMBAKER.....	36
2.8.1 Živý systém .....	36
2.8.2 Práce v systému.....	37
2.8.3 Vizualní rozhodovací funkce .....	43
<b>3 ANALÝZA PROBLÉMU A SOUČASNÉ SITAUCE</b> .....	<b>44</b>
3.1 ZÁKLADNÍ INFORMACE O FIRMĚ MATHESIO S.R.O. ....	44
3.2 MCKINSEY 7S MODEL .....	44
3.2.1 Strategie.....	44
3.2.2 Struktura.....	45
3.2.3 Systémy .....	45
3.2.4 Styl .....	45
3.2.5 Spolupracovníci.....	45
3.2.6 Schopnosti .....	45
3.2.7 Sdílené hodnoty.....	46

3.3	ZADÁNÍ A POŽADAVKY NA CRM.....	46
3.4	RIZIKOVÁ POLITIKA.....	47
3.4.1	Identifikace rizik .....	47
3.4.2	Ohodnocení rizik skórovací metodou .....	48
3.4.3	Mapa rizik před opatřeními .....	50
3.4.4	Ohodnocení rizik po zavedení opatření.....	50
3.4.5	Mapa rizik po opatření .....	52
3.5	POROVNÁNÍ SB S OSTATNÍMI PRODUKTY .....	52
3.5.1	Raynet .....	52
3.5.2	Blue CRM .....	53
3.5.3	OutSystems .....	54
3.5.4	Volba softwaru ke zpracování řešení .....	55
<b>4</b>	<b>VLASTNÍ NÁVRH ŘEŠENÍ A PŘÍNOS NÁVRHU .....</b>	<b>56</b>
4.1	PŘÍPADY UŽITÍ KE SPLNĚNÍ POŽADAVKŮ .....	56
4.2	NÁVRH DATABÁZOVÉ STRUKTURY .....	57
4.2.1	Tabulky .....	57
4.2.2	Relační vztahy .....	60
4.3	PŘÍPADY UŽITÍ PODLE ENTIT .....	62
4.4	ROLE .....	63
4.5	WORK BREAKDOWN STRUCTURE.....	63
4.6	ČASOVÁ ANALÝZA .....	65
4.7	IMPLEMENTACE .....	65
4.7.1	Zákazník .....	66
4.7.2	Kontakt.....	73
4.7.3	Obchodní příležitost a Projekt.....	75
4.7.4	Entity napojené na Projekt .....	77
4.7.5	Vytvoření menu.....	79
4.7.6	Osobní zkušenost s vývojem na platformě SystemBaker .....	82
4.7.7	Dosažení požadavků.....	82
4.7.8	Možnosti do budoucna .....	83
4.8	EKONOMICKÉ ZHODNOCENÍ.....	84
	<b>ZÁVĚR .....</b>	<b>86</b>
	<b>SEZNAM POUŽITÉ LITERATURY.....</b>	<b>87</b>
	<b>SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....</b>	<b>92</b>
	<b>SEZNAM OBRÁZKŮ .....</b>	<b>93</b>
	<b>SEZNAM TABULEK.....</b>	<b>95</b>

# ÚVOD

Podnětem diplomové práce je problém, který nastal firmě Mathesio s.r.o. a je třeba jej řešit. Tento problém je popsán v první kapitole vymezení problému a cíle práce.

Ve druhé kapitole jsou zpracovány teoretická východiska, která popisují termíny a technologie potřebné k porozumění o čem práce pojednává. Je zde stručně popsáno, co si můžeme představit pod pojmem Informační systém a CRM, také co představují Byznys procesy. Na ně navazuje jejich automatizace pomocí RPA a Workflow, které zase využívají moderní Low-code platformy. Dále je v této kapitole zmíněn i rozdíl mezi vytvořením a koupí softwaru v dnešní době a kapitolu zakončuje představení softwaru firmy Mathesio – SystemBaker.

Třetí kapitola je kapitolou zabývající se analytickým zkoumáním východisek, tedy zpracovává první informace o firmě Mathesio a její interní faktory pomocí McKinsey 7S analýzy. Následuje zmínění požadavků a zhotovení zadání ohledně nového požadovaného softwaru pro sledování zákazníků. Kapitolu uzavírá porovnání možností zpracování řešení za pomoci různých softwarů či služeb.

Ve čtvrté kapitole je zpracován návrh implementace, ve kterém se identifikují případy užití, podle nich se zpracuje návrh databáze a tento návrh se ještě i spojí s těmito případy užití. Je zde menší zmínka o rolích na platformě SystemBaker a kapitolu uzavírá WBS, ve které je zpracován návrh postupu implementace.

Pátá kapitola nese název implementace a je v ní řešena samotná implementace nového CRM systému. Začíná popisem implementace hlavní entity, na které samotný systém staví – tedy Zákazníka. Na implementaci této entity je popsán základní postup implementace, který lze reprodukovat i pro ostatní potřebné entity. Po naimplementování entity a jejich funkcionalit se ještě vytváří menu pro jednotlivé role Zaměstnanec a Projektového manažera. Dalším podkapitolou je i má osobní zkušenost s implementací na této platformě, ve které zhodnocuji, jak se mi pracovalo se SystemBakerem. Zmiňuji pak také, že všech požadavků bylo dosaženo a uvádím příklady užití implementace pro naplnění daných požadavků. V závěru kapitoly ještě zmiňuji možnosti rozšíření mého implementovaného řešení a jak snadno lze dále rozšířit tuto implementaci díky platformě SystemBaker.



Ve finální šesté kapitole je popsáno ekonomické zhodnocení řešení pomocí low-code platformy a také jiných možných řešení popsaných ve třetí kapitole, čímž se finalizuje jejich zhodnocení.

# 1 VYMEZENÍ PROBLÉMU A CÍLE PRÁCE

## 1.1 Vymezení problému

Firma Mathesio s.r.o. se zabývá převážně vývojem softwaru na zakázku. Díky velkému množství klientů se vyskytl problém s nedostatečným sledováním průběhů komunikací a správou zákazníků, proto majitel vytvořil požadavek na vytvoření CRM systému vhodného a uzpůsobeného pro firmu. Jelikož firma vyvíjí low-code platformu zabývající se digitalizací business procesů za pomoci workflow management systému, zvolila tento způsob vývoje pro daný požadavek s vypracováním ekonomického zhodnocení přínosů vývoje tímto způsobem. V důsledku mého zaměstnaneckého poměru ve společnosti Mathesio s.r.o., se zpracování tohoto projektu stalo předmětem mé diplomové práce. [30]

## 1.2 Cíle práce

Cílem této diplomové práce je analýza stávajících možností, návrh a implementace pomocí vlastního firemního low-code softwaru – SystemBaker aplikace do firemního prostředí a zhodnocení ekonomických přínosů tímto způsobem vývoje.

## 2 TEORETICKÁ VÝCHODISKA

### 2.1 Informační systém

“Informační systémy jsou provázané komponenty, které spolupracují se sběrem, zpracováním, ukládáním a šířením informací k podpoře rozhodování, koordinace, kontroly, analýzy a vizualizace v organizaci.“ [1]

Z této definice, se zaměřuje na popis informačního systému jako na komponenty, které utváří informační systém a role, které tyto komponenty zastávají v organizaci. [2]

#### 2.1.1 Komponenty informačního systému

Informační systémy jsou složeny z různých komponent, které spolupracují k vytvoření hodnoty pro organizaci. [2]

Těchto komponent je pět: hardware, software, data, lidé a procesy. První tři zmíněné spadají pod technologickou kategorii a jsou nejčastěji označovány jako hlavní části, které definují informační systém. Ale poslední dvě: lidé a procesy jsou opravdu tím, co odděluje ideu informačních systémů od více technických oborů jako třeba počítačové vědy. [2]

Abychom mohli plně porozumět informačním systémům, je nutné pochopit, jak všechny tyto komponenty spolupracují k vytvoření hodnoty pro organizaci. [2]

Jak již bylo zmíněno první tři komponenty informačních systémů – hardware, software a data, spadají všechny do kategorie technologií. Proto si je představíme, abychom mohli plně porozumět pojmu informační systém. [2]

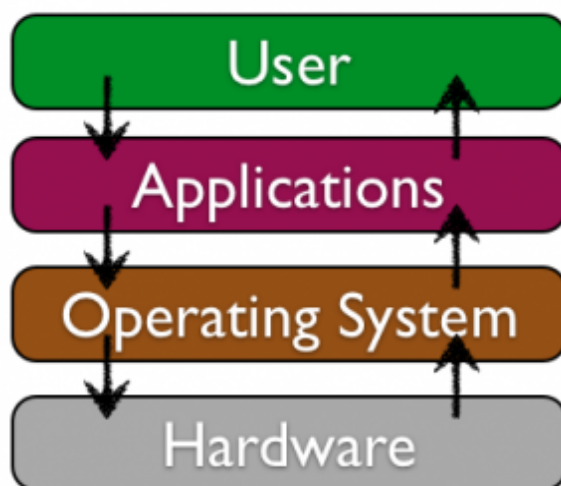
##### 2.1.1.1 *Hardware*

Hardware v informačních systémech je částí, které se můžete dotknout – jedná se o fyzické komponenty technologie. Počítače, klávesnice, pevné disky, iPady a flash disky – to jsou všechno příklady hardware, se kterým mohou informačních systémy

pracovat. Dělí se na vstupní a výstupní zařízení. Vstupní zařízení jsou používána k zachycení, či vložení dat do systému a výstupní zařízení zpracovávají data do výsledné podoby, která je čitelná člověkem. [2, 3]

### 2.1.1.2 Software

Software je posloupnost instrukcí, která říká hardwaru, co má dělat. Software není hmatatelnou komponentou – nelze se jí dotknout. Když programátoři vytváří softwarové programy, tak ve skutečnosti píšou list instrukcí, kterými říkají hardwaru, co dělat. Je několik kategorií softwaru, hlavními dvěma jsou operační systémy, které umožňují použití hardwaru a aplikační software, se kterým pracuje uživatel. Příkladem operačního systému může být Microsoft Windows na osobním počítači nebo Google Android na chytrém mobilním telefonu. Příkladem aplikačního softwaru je třeba Microsoft Excel či Adobe Acrobat Reader. [2]



Obrázek 1: Postup interakce uživatele s SW a HW [4]

### 2.1.1.3 Data

Třetí komponentu – data si můžeme představit jako soubor faktů. Například adresa vašeho domu, město, v kterém žijete a číslo vašeho mobilního telefonu – to všechno jsou data. Jako software, tak i data nejsou hmatatelná. Sami o sobě nejsou kousky dat příliš užitečné, ale agregovaná, indexovaná a organizovaná data v jedné databázi mohou být velmi schopným nástrojem k podnikání. Ve skutečnosti definice uvedená na začátku této první kapitoly se soustředí na to, jak informační systémy spravují data.

Organizace sbírají všechny druhy dat a používají je při rozhodování. Tato rozhodnutí mohou být poté analyzována z hlediska jejich účinnosti, a tak se může organizace rozvíjet a zlepšovat. [2]

#### ***2.1.1.4 Lidé***

Pokud se zamyslíme nad informačními systémy láká nás se zaměřit na technologické komponenty a opomenout fakt, že se musíme dívat nad rámec těchto nástrojů k dosažení plného pochopení, jak se integrují do organizace. [2]

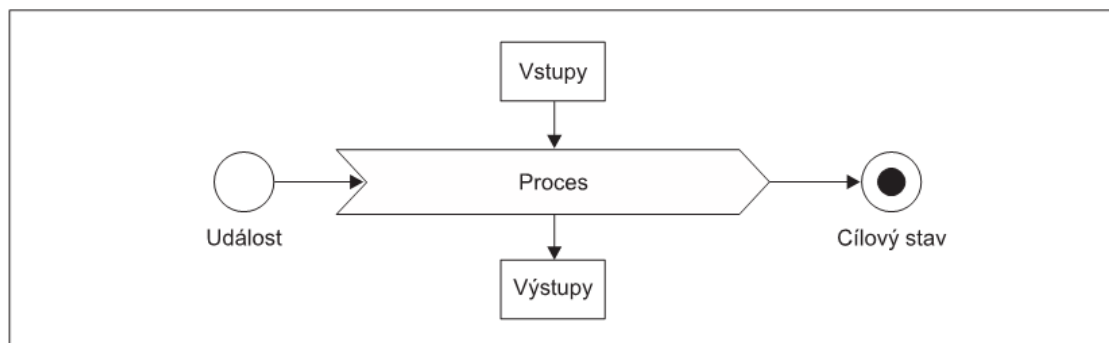
Dalším krokem je třeba se zaměřit na zapojení osob do informačních systémů. Od předních helpdesk pracovníků, přes systémové analytiku k programátorům až po hlavního informačního pracovníka (CIO), jsou lidé zabývající se informačními systémy základním prvkem, který nesmí být přehlížen. [2]

#### ***2.1.1.5 Procesy***

Poslední komponentou informačního systému je proces. Procesem rozumíme činnosti, které spolu se sebou navzájem souvisí, či na sebe působí. Tyto činnosti používají zdroje (např. lidi, materiál, nástroje apod.). Procesy mohou mít více vstupů a rovněž i více výstupů. Dalo by se také říci, že proces je série kroků podstoupených k dosažení požadovaného cíle, nebo výsledku. Informační systémy jsou stále více integrovány v organizačních procesech, což přináší více produktivity a lepší kontroly těchto procesů. Ale prostá automatizace aktivit používáním technologií není dostačující – podniky, které chtějí efektivně využívat informační systémy dělají více... [2, 5]

Používají totiž technologií ke správě a zlepšení procesů – obojí v rámci firmy a externě (s dodavateli a zákazníky) a to je finálním cílem. Technická hesla typu „reen-

gineering obchodních procesů“ nebo „management obchodních procesů“ anebo „plánování podnikových zdrojů“ mají, co dočinění s neustálým zlepšováním těchto obchodních postupů a integrací technologií s nimi. [2]



*Obrázek 2: Model procesu [5]*

Podniky, které doufají v získání výhody oproti jejich protivníkům jsou vysoce zaměřené na tuto komponentu informačních systémů. [2]

## 2.1.2 Role informačního systému

Nyní když chápeme rozdíly mezi jednotlivými komponentami informačních systémů, měli bychom upnout naši pozornost k roli, kterou informační systém hraje v podniku. [2]

Doposud jsme zkoumali jednotlivé komponenty informačního systému, ale co opravdu tyto komponenty dělají pro podnik? Z výše uvedené definice lze zjistit, že tyto komponenty sbírají, ukládají, organizují a distribuují data v podniku. Ve skutečnosti bychom mohli říct, že jednou z rolí informačního systému je vzít data, přeměnit je v informaci a poté ji zařadit do podnikové znalostní báze. S tím, jak se vyvíjely technologie, tato role se vyvinula v základní páteř organizace. [2]

## 2.2 CRM

Customer Relationship Management (zkráceně CRM) shromažďuje interakce zákazníků na jednom místě. Správa centralizovaných dat napomáhá podnikům ke zlepšení zákaznického zážitku, uspokojení, udržení a celkové služby zákazníkům. CRM

umožňuje podnikům všech velikostí řídit růst a s ním spojené navyšování výnosů, případně zisků. [6]

CRM napomáhá firmám k propojení s jejich zákazníky, podpoře pozitivního vztahu s nimi, což ve výsledku zvyšuje tržby. [7]

Zaměřit se na vztahy se zákazníky může mimo jiné zlepšit konverzi potenciálních zákazníků, udržení zákazníků a spokojenost zákazníků. Současně také snížit míru úbytku zákazníků a čas potřebný k implementaci nových strategií. Pozitivní zkušenosti zákazníků mohou odůvodnit zvýšení ceny v povědomí zákazníků. [7]

CRM systémy samozřejmě neprospívají pouze prodejci a podnikům. Zákazníci se dočkají rovněž lepší obsluhy. Lépe cílené profily zákazníků dokáží zajistit spokojenost zákazníka, zatímco data z CRM mohou zlepšit ocenění produktů k lepšímu porozumění rozpočtu zákazníka. [7]

Personalizované produkty a nabídky služeb činí nákupy více efektivní a individualizované, dále marketingové zprávy mohou pomoci zákazníkům najít to co ve skutečnosti hledají. [7]

### **2.2.1 Jak CRM pracuje**

CRM získává data z emailů, a dalších kanálů, aby Vám pomohlo získat více zákazníků a udržet si ty, které již máte. Rovněž vám dodá jednotný prostor k uspořádání vašich pracovních toků a obchodních procesů, takže můžete lépe pracovat, uzavírat více obchodů a stihnout více věcí. [6]

Automatizace marketingu, prodeje, komunikace a projektového managementu to jsou esenciální funkce CRM. [6]

V praxi by mělo CRM pracovat tak, jak daný podnik pracuje. Existuje mnoho typů CRM a dobrých, ale žádný z nich nepasuje vždy správně na každý podnik. Nicméně existuje určitě CRM, které je uzpůsobeno na řešení vaší business strategie. [6]

## **2.2.2 Výhody CRM**

### ***2.2.2.1 Zlepšuje zákaznickou obsluhu***

CRM spravuje všechny vaše kontakty a agreguje informace o potenciálních i stálých zákaznících za účelem profilování každého s kým jednáte. Toto umožňuje snadný přístup k důležitým informacím o chování zákazníka. Zákazníci se nemusí opakovat stále dokola, když vám mají vysvětlit jejich příběh, a proto budete schopni řešit problémy pomocí osvědčených postupů a vynaložíte tak menší snahu ke zlepšení loajality zákazníků. [6]

### ***2.2.2.2 Zvyšuje prodeje***

Zefektivnění a zdokonalení prodejního procesu, vybudování prodejních cest, automatizace úkolů a analýza vašich prodejních dat nevyhnutelně povedou k jednomu výsledku – zvýšení prodejů a prodejní aktivity. Zajistíte tak více dohod vytvořením opakovatelného a osvědčeného prodejního procesu a budete moci doručit správné sdělení na správný kanál ve správný čas. [6]

### ***2.2.2.3 Udržuje zákazníky***

Míra udržení zákazníka je velmi důležitým ukazatelem pro úspěšnost společnosti, odliv zákazníka bývá velkou překážkou růstu podniku. CRM nástroje jako analýza názoru zákazníka, automatizované řešení support ticketů a automatizovaná zákaznická obsluha může vysoce zlepšit udržení vašich zákazníků. Analytické nástroje, které zkoumají životní cyklus zákazníka mohou identifikovat kdy nastává odliv zákazníků a proč tomu tak je, díky tomu můžete identifikovat a minimalizovat tyto kritické body. [6]

### ***2.2.2.4 Vyšší efektivita***

Díky CRM máme všechny své hlavní každodenní obchodní funkce na jednom místě a tím získáváme lepší kontrolu pracovních toků, snazší spolupráci mezi členy týmu a lepší správu projektů. Automatizace úkolů eliminuje podřadnou, opakující se



práci a poskytuje více času na kognitivní úkoly, v nichž jsou lidé nejlepší. Řídící panely a analytika vám pomohou získat přehled o vaší práci a optimalizovat všechny druhy obchodních procesů. [6]

#### ***2.2.2.5 Lepší sdílení znalostí***

Nedorozumění a nedostatek informačního toku jsou dva hlavní plýtvací času. Když si lidé udělají čas na sebevzdělávání, aby mohli dělat věci, které již ostatní členové týmu vědí, nebo pracovat na nadbytečných úkolech, ztrácíte spoustu hodin týdně. Kolaborativní nástroje CRM mohou zefektivnit vaši týmovou práci tím, že vám umožní vybudovat základnu znalostí, vytvořit osvědčené pracovní postupy a umožnit komunikaci mezi členy týmu bez zbytečných prodlev. [6]

#### ***2.2.2.6 Více transparentnosti***

CRM vám umožňuje podporovat větší transparentnost ve vašem podniku přidělováním úkolů, sledováním práce a přesným vymezením toho, kdo je kdo a kdo dělá co. Pokud je vaším hlavním záměrem prodej, můžete využít sledování výkonu pro jednotlivé prodejce. CRM umožňuje všem ve vašem podniku získat přehled o vašich obchodních procesech a podpořit je, čímž se upevňuje vzájemné porozumění a spolupráce. [6]

### **2.2.3 Dopady implementace**

„Vezmeme-li v potaz všechny problémy, kterým dnes organizace čelí, lze konstatovat, že CRM jako obchodní koncept pokrývá komplexní přístup ke klientovi a zejména určuje schopnost podniku implementovat produkt, službu a jejich kombinaci, tak aby byla atraktivní pro zákazníky. To napomáhá stanovit schopnost podniku získávat nové a udržet stávající zákazníky, zvyšovat spokojenost zákazníků a zvyšovat také jejich loajalitu.“ Implementace CRM strategie umožňuje znatelně optimalizovat náklady a zvyšovat marži. Rovněž se zvyšuje i procento tržeb a tím i zisk společnosti a úroveň důvěry zákazníků. Politika zaměřená na potřeby zákazníků zlepšuje image společnosti na trhu a činí ji tak více atraktivní pro potenciální zákazníky. Nicméně je třeba si také uvědomit, že zavedení CRM ve firmě může změnit způsob myšlení ve firmě. Rovněž zavedení CRM, hlavně počítačového systému CRM do firmy je dlouhý

a nákladný proces. Jestliže si firma jasně nedefinuje své cíle může být implementace neúspěšná. Přestože je implementace CRM filozofie v podniku finančně i časově nákladný proces, je ziskový a dlouhodobě prospěšný. S ohledem na fakt, že CRM je nástroj, který pomáhá budovat dlouhotrvající vztahy se zákazníky a tím zvyšovat tržby, lze předpokládat, že se investice do implementace CRM vrátí, pokud budou důkladně promyšleny operace společnosti. [8]

## **2.3 Byznys procesy**

Proces je řada úkolů, které jsou zaměřeny na dosažení cíle. Obchodní proces je tedy proces, který je zaměřen na dosažení obchodního cíle. Jestliže jste pracovali ve firmním prostředí, účastnili jste se obchodního procesu. Vše od jednoduchého procesu jako např. udělení sendviče v Subwayi až po postavení raketoplánu využívá jeden či více obchodních procesů. [9]

Procesy jsou něčím, čím podniky prochází každý den k dosažení jejich cílů. Čím lepší procesy, tím efektivnější je daný podnik. Pro některé podniky jsou jejich procesy stěžejní k dosažení konkurenční výhody. Proces, díky kterému podnik dosáhne cíle jedinečnou cestou, může odlišit daný podnik od ostatních. Proces, který eliminuje náklady, může firmě dovolit snížit ceny, či zajistit větší zisk. [9]

### **2.3.1 Dokumentace procesů**

V čase se procesy stále mění a stávají se více komplexními, proto je třeba je zdokumentovat. Pro podniky je tato činnost podstatná, protože dovoluje podnikům zajistit kontrolu nad tím, jak jsou aktivity v podniky podchyceny. Dokumentace procesů rovněž dovoluje standardizaci: třeba firma McDonald's má stejný proces udělení Big Macu ve všech svých restauracích. [9]

Jakmile organizace začnou dokumentovat své procesy, stane se administrativní úlohou je sledovat. V procesu zlepšování stávajících procesů, je důležité vědět, které procesy jsou nejnovější. Je také důležité řídit proces tak, aby jej bylo možné snadno aktualizovat. Požadavek na správu dokumentace procesů byl hnací silou k vytvoření systému pro správu dokumentací. Systém pro správu dokumentací ukládá a sleduje dokumentace a podporuje následující funkce:

- **Verzování a časové značky.** Systém pro správu dokumentací udržuje vícero verzí dokumentací. Nejnovější dokumentace je tak snadno identifikovatelná a systém tuto verzi zobrazuje ve výchozím stavu.
- **Schválení a pracovní postupy.** Když je třeba změnit proces, systém bude spravovat jak přístup k dokumentům pro úpravy, tak i to, kdo může tyto změny schvalovat.
- **Komunikace.** Když se proces změní je třeba dát vědět těm, kdo ho implementují. Systém upozorňuje příslušné osoby po schválení dané změny. [9]

Jistěže systém pro správu dokumentů není používán pouze k dokumentaci obchodních procesů, mnoho jiných dokumentací či jen dokumentů je spravováno v tomto systému jako jsou třeba i právní dokumenty, či návrhové anebo konstrukční dokumenty. [9]

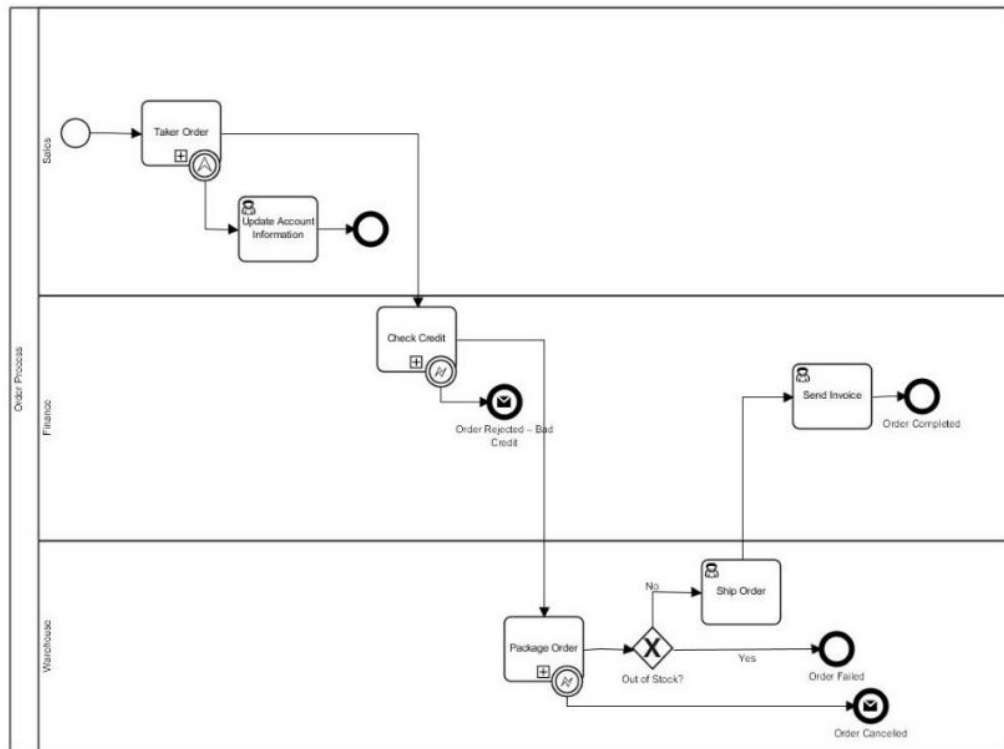
### 2.3.2 Reinženýring podnikových procesů

Jak se podniky snaží řídit své procesy k tomu, aby získaly konkurenční výhodu, je rovněž nutné, aby pochopili, že pracovní postup nemusí být nejefektivnější. Jelikož proces vytvořený v roce 1950 nebude lepším jen proto, že má nyní podporu nových technologií. [9]

Reinženýring obchodních procesů (Business process reengineering – BPR) není jen o tom vzít existující proces a zautomatizovat jej. BPR je o plném pochopení cílů procesů a následném přepracování jej od píky až k dosažení dramatického zlepšení produktivity a kvality. Ale to se snadněji řekne, než udělá. Většina z nás přemýšlí o tom, jak udělat malá lokální vylepšení procesu, ale kompletní přepracování vyžaduje přemýšlení ve větším kontextu. [9]

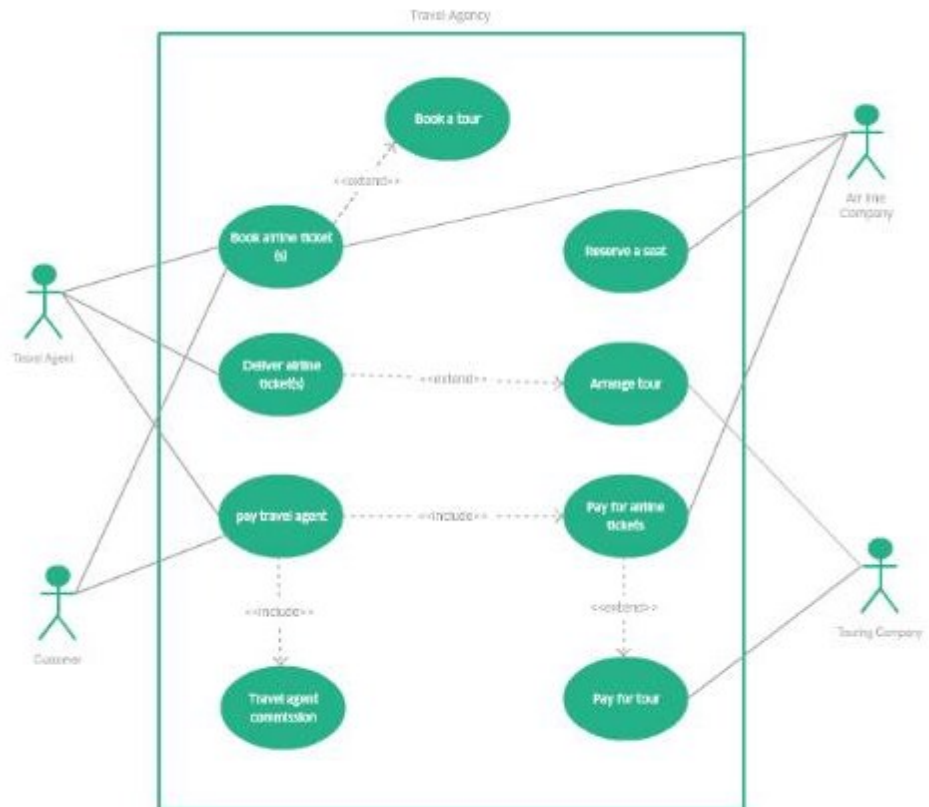
### 2.3.3 Techniky BPM

- **BPM Notace:** modeluje kroky obchodního modelu od začátku do konce. Klíč ke správě obchodních procesů je vizualizace podrobných kroků obchodního procesu a toků informací potřebných k dokončení procesu. [10]



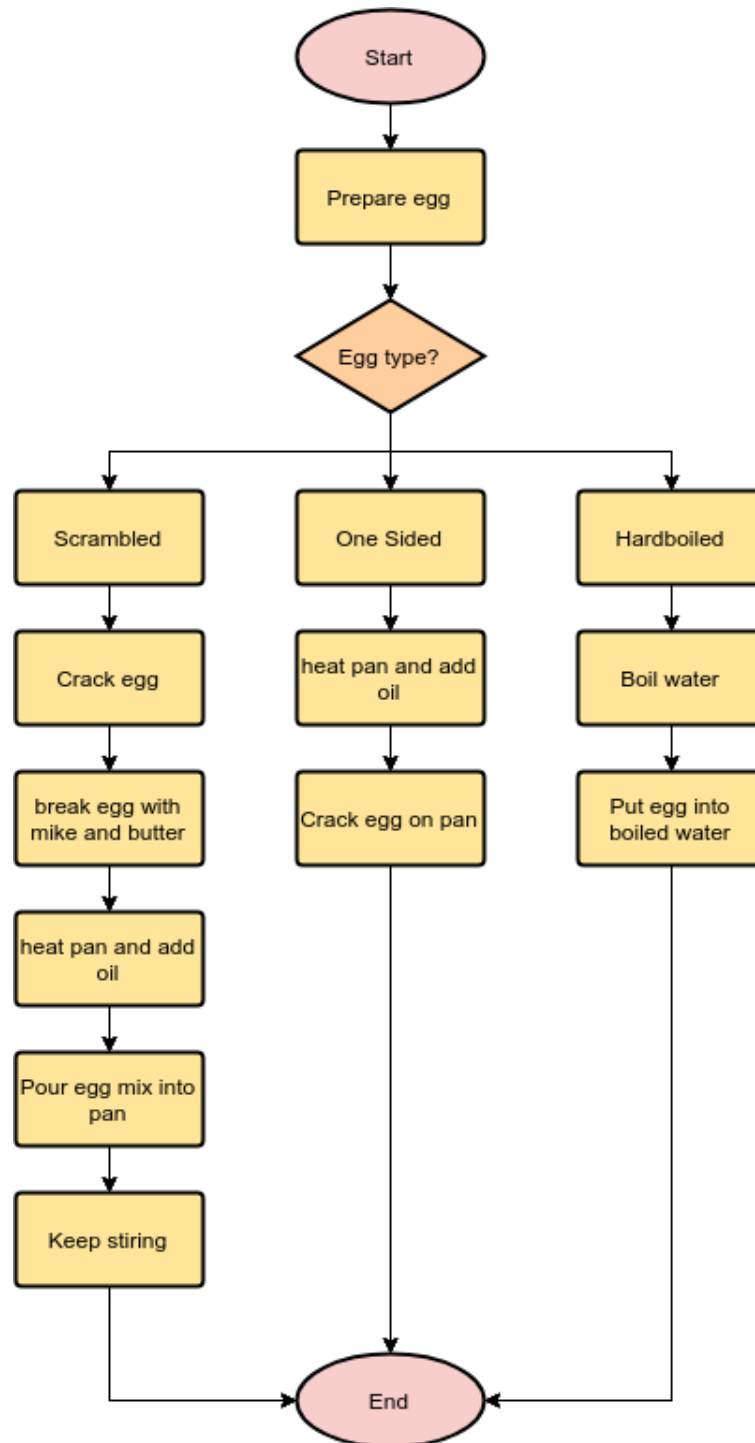
Obrázek 3: Příklad BPM notace [11]

- **UML Diagramy:** jsou standardním vizuálním modelovým jazykem, který se užívá k modelování podnikových a jiných podobných procesů, analýze, návrhům a implementaci softwarově založených systémů. [12]



**Obrázek 4: Příklad Use Case UML Diagramu [13]**

- **Vývojové diagramy:** jsou diagramy, které zobrazují algoritmy procesů, systémů či počítačů. Používají se v mnoha oborech k dokumentaci, studiu, plánování, zlepšování a komunikaci často složitých procesů v jasných a snadno srozumitelných diagramech. [14]



*Obrázek 5: Příklad vývojového diagramu [15]*

- **Diagramy datových toků (DFD):** mapují tok informací jakéhokoli procesu či systému. [16]
- **Ganttovy grafy:** převážně se používají v projektovém managementu, je to nejpopulárnější a nejpoužívanější způsob zakreslení aktivit (úkolů nebo událostí) v čase. [17]
- **PERT Diagramy:** jsou vizuálním nástrojem projektového managementu, který se používá k zakreslení a sledování úkolů a časových os. Jméno PERT je akronymem pro Project (nebo Program) Evaluation and Review Technique – Projektové (nebo Programové) vyhodnocovací a posuzovací techniky. [18]
- **Funkční bloková schémata:** jsou způsobem vykreslení a popisu procesů ve vývojových a produkčních systémových prostředích. Jsou víceúrovňové, časově řazené podrobné vývojové diagramy funkčního toku systému. [19]
- **Integrovaná definice pro modelování funkcí (IDEF):** Pokrývají široký rozsah použití od funkcionálního modelování po data, simulace, objektově-orientované analýzy/návrhy a získávání znalostí. [20]

## 2.4 Robotic process automation

Robotic Process Automation (RPA) je specificky navrženo k automatizaci úkolů, které jsou prováděny lidmi na počítačích. Je to softwarová technologie, kterou může používat každý k automatizaci úkolů v digitálním prostředí. S RPA softwarem vytváří uživatelé „roboty“, které naučí napodobovat a provádět obchodní procesy na základě určitých pravidel. RPA automatizace umožňuje uživatelům vytvářet roboty sledováním digitálních akcí prováděných člověkem. Stačí ukázat softwaru co má udělat, a ten práci provede za vás. „Roboti“ v RPA systému mohou interagovat s jakoukoli aplikací nebo systémem stejně jako lidé, ale RPA může operovat také na základě času, nebo nonstop a mnohem rychleji a se 100% spolehlivostí a přesností. [21, 22]

### 2.4.1 Jak RPA funguje

RPA má stejnou základnu digitálních dovedností jako lidé, a ještě něco navíc. O RPA můžete smýšlet jako o digitální pracovní síle, která interaguje se systémem nebo

aplikací. Například je schopná provádět copy-paste, získávat data z webu, provádět kalkulace, otevírat a přesouvat soubory, číst emaily, předávat informace programům, napojovat se na API a extrahovat nestrukturovaná data. A protože se může RPA přizpůsobit jakémukoli rozhraní nebo workflow není potřeba měnit podnikové systémy, aplikace nebo stávající procesy v zájmu automatizace. [22]

## 2.4.2 Výhody

RPA má pozitivní efekt sněhové koule na podnikové operace a výstupy. RPA dodává měřitelné podnikové výhody, např. snížení nákladů, větší přesnost, rychlost dodání, větší přidaná hodnota, díky zvyšování dynamiky a rozšíření se napříč organizací. RPA zlepšuje výstupy podniku jako spokojenost zákazníků a umožňuje konkurenční výhodu tím, že osvobozuje lidi, aby se mohli soustředit pouze na řešení problémů, zlepšení procesů, provádění analýz a jiných prací, co přináší přidanou hodnotu práce, což ve výsledku zvyšuje zaměstnanecké zapojení a potenciální nový růst zisku. [22]

## 2.4.3 Obecné využití

- **Napodobuje lidské akce:** Provádí opakovatelné úkony v procesu místo člověka za pomoci různých aplikací a systémů.
- **Provádí velkoobjemové opakované úkoly:** RPA může snadno simulovat přepisování dat z jednoho systému do druhého. Provádí úkoly jako zadávání, kopírování a vkládání dat.
- **Provádí více úkolů:** Provozuje několik složitých úkolů napříč více systémy. Což pomáhá zpracovávat transakce, manipulaci s daty a zaslání reportů.
- **„Virtuální“ systémová integrace:** Tento automatizační systém může přenášet data mezi různorodými a staršími systémy jejich propojením na úrovni uživatelského rozhraní namísto vývoje nové datové infrastruktury.
- **Automatizované generování reportů:** Automatizuje extrakci dat k vytvoření přesných, efektivních a časových reportů.
- **Validace a audit informací:** Řeší a křížově ověřuje data mezi různými systémy k ověření a kontrole informací pro poskytnutí výstupů pro dodržování předpisů a audit.



- **Správa technického dluhu:** Napomáhá snižovat code debt (výběr snadnějšího řešení, před správným v rámci vývoje) zmenšením propasti mezi systémy a brání zavádění vlastních implementací.
- **Product management:** Pomáhá překlenout propast mezi IT systémy a souvisejícími platformami pro správu produktů automatizovanou aktualizací obou systémů.
- **Zajištění kvality (Quality Assurance):** Může být výhodným nástrojem pro QA procesy, které pokrývají regresní testování a automatizaci use-case scénářů zákazníka.
- **Migrace dat:** Umožňuje automatickou migraci dat prostřednictvím systémů, což není možné pomocí tradičních médií, jako jsou dokumenty, tabulky nebo jiné soubory zdrojových dat.
- **Řešení mezer:** Robotická automatika vyplňuje mezery a nedostatky procesů. Což zahrnuje například jednoduché úkoly, jako je resetování hesla, resetování systému atd.
- **Predikce výnosů:** Dostupnost automatické aktualizace finančních výkazů pro předpovědi a prognózy příjmů. [23]



**Obrázek 6: Úkoly, které lze automatizovat pomocí RPA [21]**

## 2.5 Workflow

Workflow (WF = pracovní tok) se zabývá automatizací procedur, kde dokumenty, informace nebo úkoly jsou si mezi sebou předávány podle předem definovaných pravidel (v sekvenčním toku) k dosažení, či aspoň přispění k hlavnímu podnikovému cíli. Zatímco workflow může být organizován ručně, v praxi je většina workflow obvykle organizována v kontextu IT systému, aby poskytovala počítačovou podporu pro procesní automatizaci. [24]

Workflow je často spojován s BPR, který se zabývá zhodnocením, analýzou, modelováním, definicí a následnou operační implementací podstatných podnikových procesů v organizaci. Ačkoli ne všechny BPR aktivity vedou k implementaci workflow, workflow technologie je často příhodným řešením, jelikož poskytuje možnost oddělení logiky podnikových procedur a jejich operační podpory v IT, což umožňuje, aby následné změny byly začleněny do pravidel definujících obchodní proces. Naopak ne všechny implementace workflow nutně tvoří součást BPR, například implementace k automatizaci existující obchodní procedury. [24]

### 2.5.1 Workflow Management Systém

Workflow Management Systém (WFM systém) je systém, který poskytuje procedurální automatizaci obchodních procesů řízením toku pracovních činností a informováním vhodného člověka a/nebo IT zdroje souvisejícího s krokem dané činnosti. [24]

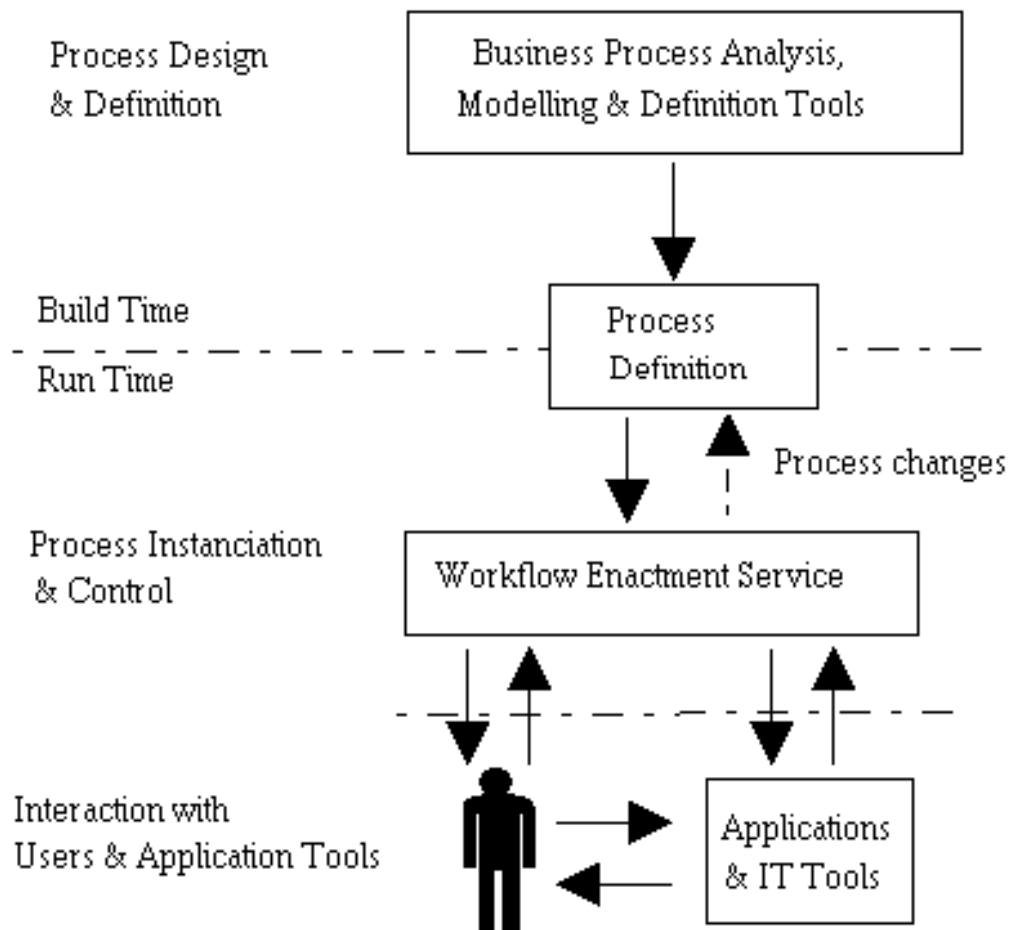
Workflow Management Systém je systém, který zcela definuje, spravuje a vykonává „workflow“ prostřednictvím softwaru, který je řízen počítačovou reprezentací logiky pracovního toku. [24]

Jednotlivé podnikové procesy mohou mít životní cyklus od minut ke dnům (nebo dokonce měsícům), což záleží na jejich složitosti a době vykonávání jednotlivých kroků. Takové systémy mohou být implementovány různými způsoby, mohou využívat širokou škálu IT a komunikační infrastruktury a fungovat v prostředí od malé lokální pracovní skupiny až po mezipodnikové. Referenční model Workflow Management Coalition (WFMC) tedy zaujímá široký pohled na správu pracovního toku, který je určen k přizpůsobení se různým implementačním technikám a provozováním prostředí, která tuto technologii charakterizují. [24]

Navzdory této rozdílnosti, všechny WFM systémy vykazují jisté společné vlastnosti, které poskytují základ pro rozvoj integrace a interakce mezi různými produkty. Referenční model popisuje společný model pro konstrukci workflow systémů a identifikuje, jak mohou souviset s různými alternativami implementačních přístupů. [24]

Na nejvyšší úrovni lze všechny WFM systémy charakterizovat jako systémy, které poskytují podporu ve třech funkčních oblastech:

- **Funkce v čase sestavování** – zabývající se definováním a také modelováním workflow procesu s jeho podstatnými kroky.
- **Funkce řízení běhu systému** – zabývající se správou workflow procesů v operačním prostředí a řízením návaznosti různých činností, které mají být zpracovány jako součást každého procesu.
- **Interakce za běhu systému** s lidskými uživateli a aplikačními nástroji IT pro zpracování různých kroků činností. [24]



*Obrázek 7: Základní charakteristiky WFM systémů a souvislost mezi hlavními funkcemi [24]*

### **2.5.1.1 Funkce v čase sestavování**

Funkce v čase sestavení jsou ty, které končí definicí obchodního procesu v digitalizované formě. V průběhu této fáze se obchodní proces transformuje ze skutečného světa do formální, digitalizované podoby za použití jedné či více analýz, modelování a technik definovaných systémem. Výsledné digitalizované podobě se někdy také říká: model procesu, procesová šablona, metadata procesu nebo definice procesu. [24]

Členové WFMC neuvažují nad standardizací oblasti počátečního vytvoření procesních definicí. Spíše je tato oblast považována jako hlavní odlišovací oblast mezi produkty tohoto typu na trhu. Nicméně výsledek operací v čase sestavení, tedy definice procesu, je identifikována jako jedna z potenciálních oblastí ke standardizaci, která umožňuje výměnu dat definice procesu mezi různými nástroji v čase sestavení a produkty v době běhu systému. [24]

### ***2.5.1.2 Funkce řízení procesu za běhu systému***

Za běhu systému je definice procesu interpretována softwarem, který je zodpovědný za vytváření a kontrolu operačních instancí procesu, plánování různých činností v rámci procesu a výzvu vhodnému člověku a aplikačnímu IT prostředí atd. Tyto řídicí funkce se chovají jako vazby mezi procesem, jak je modelován v rámci definice procesu a procesem jaký lze vidět ve skutečném světě, a proto je u těchto vazeb v průběhu systému nutná interakce uživatelů a aplikačních nástrojů IT. [24]

Základní komponentou je ovládací software pro řízení workflow (nebo také „engine“), zodpovědný za vytvoření a smazání procesu, kontrolu plánování činností v rámci operačního procesu a interakce s aplikačními nástroji IT nebo lidmi. Tento software je často distribuován na mnoho počítačových platform, aby zvládl procesy, které fungují na široké geografické bázi. [24]

### ***2.5.1.3 Interakce za běhu systému***

Jednotlivé úkoly v rámci workflow procesu se obvykle zabývají lidskými operacemi, často realizovanými ve spojení s použitím konkrétního IT nástroje (například vyplňování formulářů) nebo operacemi zpracování informací vyžadujícími určitý aplikační program, který operuje na určitých definovaných informacích (například úprava objednávek v databázi vytvoří nový záznam). Interakce se softwarem pro řízení procesů je nezbytná k přenosu kontroly mezi úkoly, ke zjištění provozního stavu procesů, k vyvolání aplikačních nástrojů a předání příslušných údajů atd. Standardizovaný rámec pro podporu tohoto typu interakce má několik výhod, včetně použití konzistentního rozhraní pro více workflow systémů a schopnosti vyvíjet obecné aplikační nástroje pro práci s různými workflow produkty. [24]

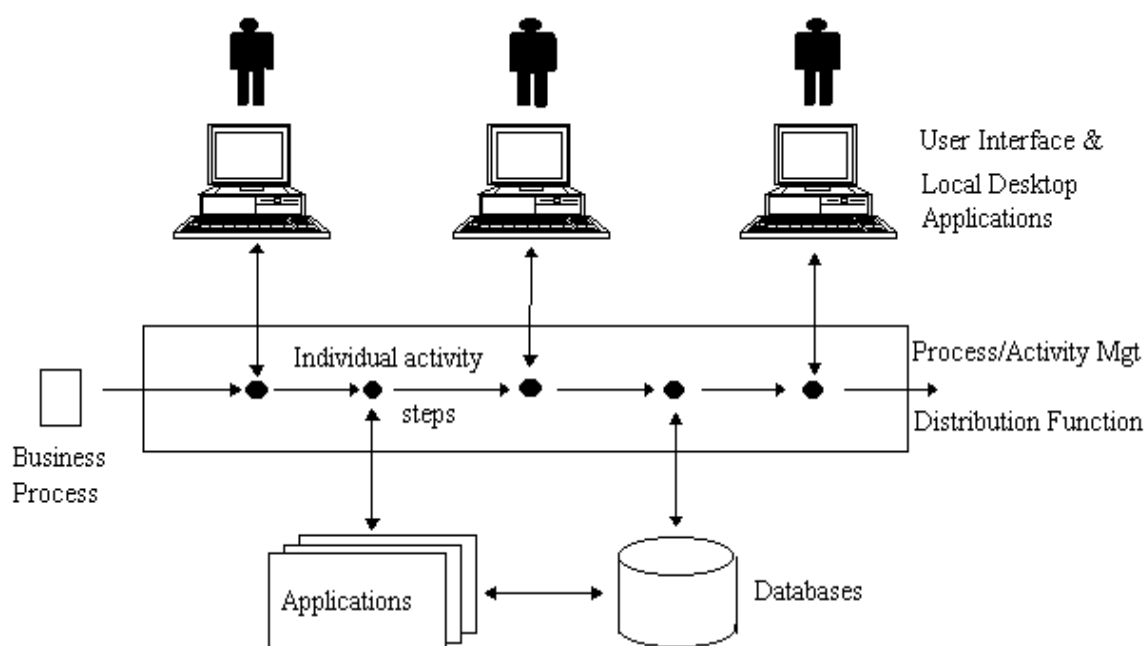
### ***2.5.1.4 Systémové rozhraní a distribuce***

Schopnost distribuovat úkoly a informace mezi účastníky je hlavní odlišovací funkcí infrastruktury workflow při řízení procesu za běhu systému. Distribuční funkce může pracovat na různých úrovních (od pracovní skupiny po práci mezi organizacemi), což záleží na rozsahu daného workflow systému. Může používat množství zá-

kladných komunikačných mechanizmů (elektronická pošta, předávání zpráv, technologie distribuovaných objektů atd.). Alternativní vysokoúrovňový pohled workflow architektury, který zdůrazňuje tento distribuční aspekt je zobrazen na **obrázku 8**. [24]

Služba ustanovení workflow je funkce, která interpretuje a řídí sekvenci úkolů a také řídí specifikaci služeb. [25]

Služba ustanovení workflow je zobrazena jako základní funkce infrastruktury s rozhraními pro uživatele a aplikace distribuované napříč workflow doménou. Každé z těchto rozhraní je potenciálním bodem integrace mezi službou ustanovení workflow a jinou infrastrukturou nebo aplikační komponentou. [24]



**Obrázek 8: Distribuce v rámci služby ustanovení workflow [24]**

Postup práce může zahrnovat předání úkolů mezi jednotlivými aktéry workflow, aby bylo možné ustanovit různé části obchodních procesů na různých platformách nebo podsítích pomocí konkrétních produktů vhodných pro danou fázi procesu. [24]

Kompletní rozsah rozhraní, které WFMC definuje pokrývá:

- Specifikace pro data definice procesu a jejich výměnu
- Rozhraní pro podporu spolupráce mezi různými workflow systémy.
- Rozhraní pro podporu interakce s rozmanitými typů IT aplikací.

- Rozhraní pro podporu interakce s funkcemi pro uživatelské rozhraní na stolních PC.
- Rozhraní zajišťující monitorování systému a metrické funkce usnadňující správu prostředí aplikací různých workflow. [24]

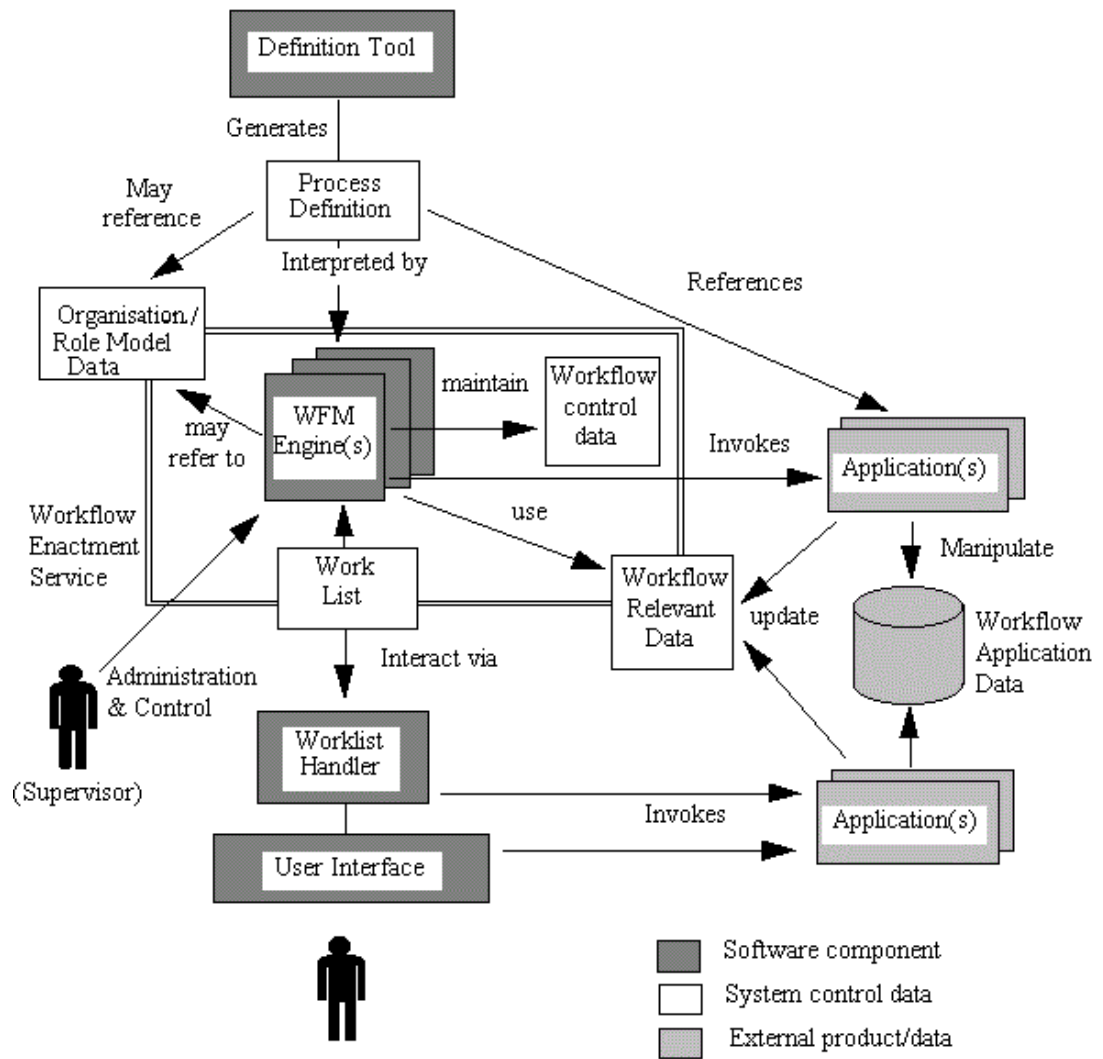
### ***2.5.1.5 Nástroj definování procesu***

Nástroj definování procesu se používá k vytvoření popisu procesu v digitální formě. To může být založeno na formálním definičním jazyku procesu, objektově relačním modelu, nebo na jednodušším systému, skriptu nebo souboru směrovacích příkazů pro přenos informací mezi zúčastněnými uživateli. Definiční nástroj může být dodán jako součást specifického workflow produktu nebo může být součástí produktu analýzy obchodního procesu, který má další komponenty pro zpracování analýz nebo modelování obchodních operací. V tomto druhém případě musí existovat kompatibilní výměnný formát pro přenos definice procesu do a z workflow softwaru za běhu. [24]

## **2.5.2 Definice procesu**

Definice procesu obsahuje všechny potřebné informace o procesu k tomu, aby byl vykonán pomocí služby ustanovení workflow. To zahrnuje informace o:

- podmínkách spuštění a dokončení,
  - podstatných aktivitách a podmínkách pro navigaci mezi nimi,
  - uživatelských akcí, které mají být provedeny,
  - referencích na aplikace, které mají být vyvolány,
  - definice jakýchkoli relevantních dat workflow, na které se má odkazovat atd.
- [24]



**Obrázek 9: Obecná struktura Workflow produktu [24]**

Proces definice může odkazovat na model organizace / role, který obsahuje informace ohledně organizační struktury a rolí v rámci organizace (např. Adresář organizace). To umožňuje definici procesu specifickou v ohledu na entity organizace a funkční role spojené s určitými aktivitami nebo informačními objekty, spíše než specifickými účastníky. Služba ustanovení workflow má poté za úkol spojení entit organizace nebo rolí se specifickými účastníky v rámci prostředí celého běhu workflow. [24]



### 2.5.3 Relevantní a aplikační workflow data

Rozhodování v navigaci procesů nebo jiné kontrolní/rozhodovací operace v rámci workflow engine úzce souvisí s generovanými nebo upravovanými daty od aplikačních programů workflow. Tyto data jsou přístupná workflow engine a označovány jako relevantní workflow data (také známé jako „případová data“ [case data]). Taková data jsou jediným typem aplikačních dat přístupných workflow engine. [24]

S aplikačními workflow daty je manipulováno přímo (a pouze) vyvolanými aplikacemi, ačkoli workflow engine může být zodpovědný za přenos těchto dat mezi aplikacemi (pokud je potřeba), jako jsou různé aplikace vyvolávány různými činnostmi v úkolech workflow procesu. [24]

## 2.6 Low-code

Low-code je označení přístupu pro vývoj softwaru, který umožňuje rychlejší vývoj aplikací s minimalizací ručně psaného kódu. Za použití vizuálního modelování v grafickém rozhraní k sestavení a konfiguraci aplikací. Vývojáři tak mohou přeskočit všechny infrastrukturní a stále opakované implementační vzory, které zabírají mnoho času, a tak mohou rovnou přejít k vývoji unikátní části dané aplikace. [26]

Například, když si představíme podnik vyrábějící auta. Stroje zahrnuté v automatizaci nerozhodují, jak dané auto bude vypadat, ale urychlují jejich sestavení a proces dokončení. Víceméně to je, co low-code umí. [26]

Low-code je software, jako výrobní linky pro automobilní průmysl: obojí automatizují ručně prováděné úkoly, které jsou obtížné a časově náročné, aby osvobodily lidi k vykonávání důležitějších věcí. [26]

Low-code je sada nástrojů, která se užívá při vytváření kompletních aplikací ve vizuálním prostředí za pomoci drag-and-drop rozhraní. Radši než psát tisíce řádků složitěho kódu a syntaxe, low-code platformy umožňují uživatelům rychle a vizuálně sestavit kompletní aplikace s moderními uživatelskými rozhraními, integracemi, daty a logikou. [26]

## **2.6.1 Výhody**

### ***2.6.1.1 Zlepšení agility***

V dlouhodobém měřítku aplikace vytvořené za pomoci low-code platforem pomáhají organizacím stát se více agilními. Vizuální design, který umožňuje „kreslit“ místo psaní kódu může exponenciálně zrychlit vývoj. Spojením méně kódování s automatizovaným můžete vytvářet aplikace rychleji než kdykoli předtím. [27]

### ***2.6.1.2 Snížení nákladů***

S možností vytvářet více aplikací za kratší čas se snižují náklady. Ale to není jediný důvod. Low-code vývoj snižuje potřebu mít více vývojářů a redukuje se tím také náklady na zaměstnance. A správná low-code platforma může všechny v organizaci, nejen IT lidi, udělat více produktivními. [27]

### ***2.6.1.3 Vyšší produktivita***

Low-code vývoj umožňuje vytvářet více aplikací za kratší čas. Co dříve trvalo měsíce může být sníženo na dny, dokonce minuty. S vývojem pomocí low-code platformy čas již není bariérou pro opravdové inovace. [27]

### ***2.6.1.4 Lepší zážitek zákazníků***

Low-code vývoj má větší dopad než IT organizace. Následný efekt zvýšení rychlosti zahrnuje lepší zážitek zákazníkům. Díky low-code vývoji se organizace rychleji přizpůsobí změnám na trhu nebo potřebám zákazníků. [27]

### ***2.6.1.5 Efektivita správy a řízení rizik***

Jak může váš podnik stále držet krok s neustále se měnícími regulacemi, nemluvě o jejich celosvětovém rozsahu? Low-code vývoj umožňuje rychlé změny, tak abyste splňovali dané požadavky, a ještě před termínem. [27]

### **2.6.1.6 Snadná změna**

Low-code vývoj umožňuje snadnou změnu aplikací (adaptaci na nové požadavky). Bez potřeby složitého programování, tento vývoj usnadňuje okamžitou změnu, v případě potřeby. [27]

### **2.6.1.7 Rychlejší transformace**

V dnešním digitálním světě je potřeba transformace. Low-code vývoj odstraňuje složitost z vývoje velkých moderních podnikových aplikací. A snížená složitost znamená hladší průběh. [27]

S těmito výhodami jsou podniky lépe vybaveny pro rychlou adaptaci a odpověď na rychle se měnící obchodní podmínky. [27]

## **2.7 Vytvořit vs. koupit**

Když se podnik rozhoduje, že je potřeba nového softwaru, musí určit, jestli má víc smysl vytvořit jej od nuly, nebo si jej koupit od jiné firmy. Tedy zjednodušeně rozhodnutí: Vytvořit nebo koupit. [28]

Existuje mnoho výhod, pokud bychom si software koupili od jiné firmy. První výhodou je, že tím vynaložíme méně nákladů, než pokud bychom jej vytvořili. Druhou, že je tato metoda rychlejší a softwaru se dočkáme mnohem dříve. Vytvořit softwarové aplikace z časového hlediska může zabrat měsíce, klidně i roky, zatímco koupený software můžeme zprovoznit i do měsíce od koupení. [28]

Zakoupený software byl již navíc otestován a mnoho chyb již bylo vyřešeno. Při této metodě je potřeba hlavně systémového integrátora, jehož úkolem je zajistit, aby zakoupený systém spolupracoval se stávajícími systémy v organizaci. [28]

Koupě softwaru má ale rovněž několik nevýhod. První nevýhodou je, že stejný software si může koupit a používat i konkurence. Jestliže se firma snaží odlišit na základě obchodního procesu, který je obsažen v zakoupeném softwaru, bude to mít těžké, pokud její konkurenti používají stejný software. Další nevýhodou nákupu softwaru je proces přizpůsobení. Pokud zakoupíte softwarový balíček od dodavatele a poté si jej

přizpůsobíte, budete muset tato přizpůsobení spravovat pokaždé, když dodavatel poskytne upgrade verze. Což může nastat i několikrát do roka a pokaždé to stojí určité náklady. [28]

Dokonce i když se podnik rozhodne software koupit, stále má smysl udělat analýzy, které by provedl i pokud by si jej chtěl postavit sám. Jelikož se jedná o důležité rozhodnutí, které má dlouhodobý strategický dopad na podnik. [28]

## 2.8 SystemBaker

Jedná se o webovou platformu, která kombinuje technologii RPA, WFM systému a low-code vývoje. [31]

Díky této kombinaci může normální uživatel v systému modifikovat cokoli dle svých představ. To je velkou přidanou hodnotou od klasických řešení, kdy se musí editovat dané požadavky přímo v kódu aplikace. A samozřejmě to může pouze programátor, tedy člověk znalý syntaxe programovacího jazyku, ve kterém byla aplikace naprogramována. Z toho důvodu se systém může rychleji a jednodušeji upravit a přizpůsobit novým požadavkům, což je atraktivnější volbou pro podnikatele. [31]

Výstupem SystemBakeru je webová aplikace běžící na samostatné webové doméně, která je plně konfigurovatelná. [31]

SystemBaker ještě nemá pevně stanovenou cenu, zatím se uvažuje o možnostech. Aktuálně se jeví jako nejperspektivnější finanční model zavedení měsíčního subscription poplatku dle jednotlivých úrovní uživatelů, definovaných podle stráveného času v systému a frekvenci přihlášení. [31]

Tento software je představitelem tzv. živého systému, díky své velké modulárnosti. Každý jednotlivý proces, lze totiž namodelovat do detailu na jednotlivé proměnné a funkce jako u klasického programování.[31]

### 2.8.1 Živý systém

V současné době jsou lidé koordinováni ve své činnosti velkým množstvím informačních systémů, které usměřňují jejich činnost tak, aby byla v rámci firmy efektivní. Nicméně problém občas bývá v tom, že systém byl navržen např. před deseti či dvaceti

lety, často je postaven i na procesech, které byly v té době uznávané jako správné. Lidé ve firmě si občas i uvědomují, že některé firemní procesy firmě úplně nesedí. Ať už z pohledu toho, že systém je zastaralý nebo, že se systém vyvíjel bez detailní uživatelské znalosti fungování firmy. [31]

V některých organizacích je služebně starší systém než většina zaměstnanců. Tudiž zaměstnanci, kteří vědí, že by něco šlo dělat lépe to často dělají postaru. Z důvodu toho, že jim to systém nedovolí anebo ten systém prostě obchází. To pak znesnadňuje celkovou organizaci v podniku. Ideální by bylo, aby se systém dokázal přizpůsobit novým požadavkům prakticky ihned. [31]

### ***2.8.1.1 Umožňuje rychlou adaptabilitu na evoluci organizace***

Živý systém by měl být tedy adaptabilní – tj měl by se měnit rychle, tak rychle jak se mění firma. Firma se musí měnit, firma, která se nerozvíjí dostatečně rychle časem zanikne a předběhne ji konkurence. [31]

### ***2.8.1.2 Systém se umí změnit i za jeho vlastního běhu.***

Součástí systému jsou lidé. Lidé by se měli podílet na jeho formování, neboť systém by měl být jejich výkonný kolega, který jim pomáhá v jejich práci. Měla by být dobrá komunikace mezi uživatelem a jeho kolegou systémem. Stejně jako člověk o něco požádá jiného člověka, tak uživatelé by řekli systému, jak se má chovat a on by to tak dělal. [31]

### ***2.8.1.3 Dokáže komunikovat o každém detailu, který se v něm děje.***

Živý systém by měl řešit komunikaci se všemi účastníky systému, měl by jim být tedy schopen rozdělovat úkoly, sbírat a předávat informace vhodným lidem, kontrolovat, zda vše probíhá v pořádku a upozorňovat, zda tomu tak není. [31]

## **2.8.2 Práce v systému**

### ***2.8.2.1 Datové objekty a formuláře***

V softwaru se pracuje s tzv. Datovými Objekty (DO), které si můžeme představit jako tabulky v databázi. [31]

**Systembaker**

MENU

- Dashboard
- Data Objects
- Forms
- Screens
- Processes
- Users
- Role
- Menu Creator
- Scripts
- File Manager
- Customize

**Sloupce**

Přidat nový sloupec

**Jméno:** email\_osobni

**Popisek:** Osobní e-mail

**Datový zdroj:** Internal

**Tabulka:**  Zobrazit vše

**Sloupec:** Id [INTEGER]

**form\_kontakt**

6

18. 4. 2021 17:58:47

Je enum:

Je aktualizovatelné:

**Popisek:**

**Existující sloupce**

**Jméno:** id

**Popisek:**

**Jméno:** email

**Popisek:** E-mail

**Jméno:** jmeno

**Popisek:** Jméno

**Jméno:** poznamky

**Popisek:** Poznámky

**Jméno:** telefon

**Popisek:** Telefon

**Jméno:** zakaznik

**Popisek:** Zákazník

Jméno	Typ	Zdroj
id	SERIAL	internal.form.kontaktL6.id
email	VARCHAR	internal.form.kontaktL6.em
jmeno	VARCHAR	internal.form.kontaktL6.jm
poznamky	VARCHAR	internal.form.kontaktL6.po
telefon	VARCHAR	internal.form.kontaktL6.tel
zakaznik	INTEGER	internal.form.kontaktL6.zal

**Info**

**Jméno:** form\_kontakt

**Revize:** 6

**Upraveno:** 18. 4. 2021 17:58:47

Je enum:

Je aktualizovatelné:

**Popisek:**

Zobrazit formulář

Zobrazit data

Uložit

**Zdroje**

- internal.form.kontaktL6

**sloupce**

- id
- email
- jmeno
- poznamky
- telefon
- zakaznik

Obrázek 10: Detail Datového Objektu [Vlastní zpracování]

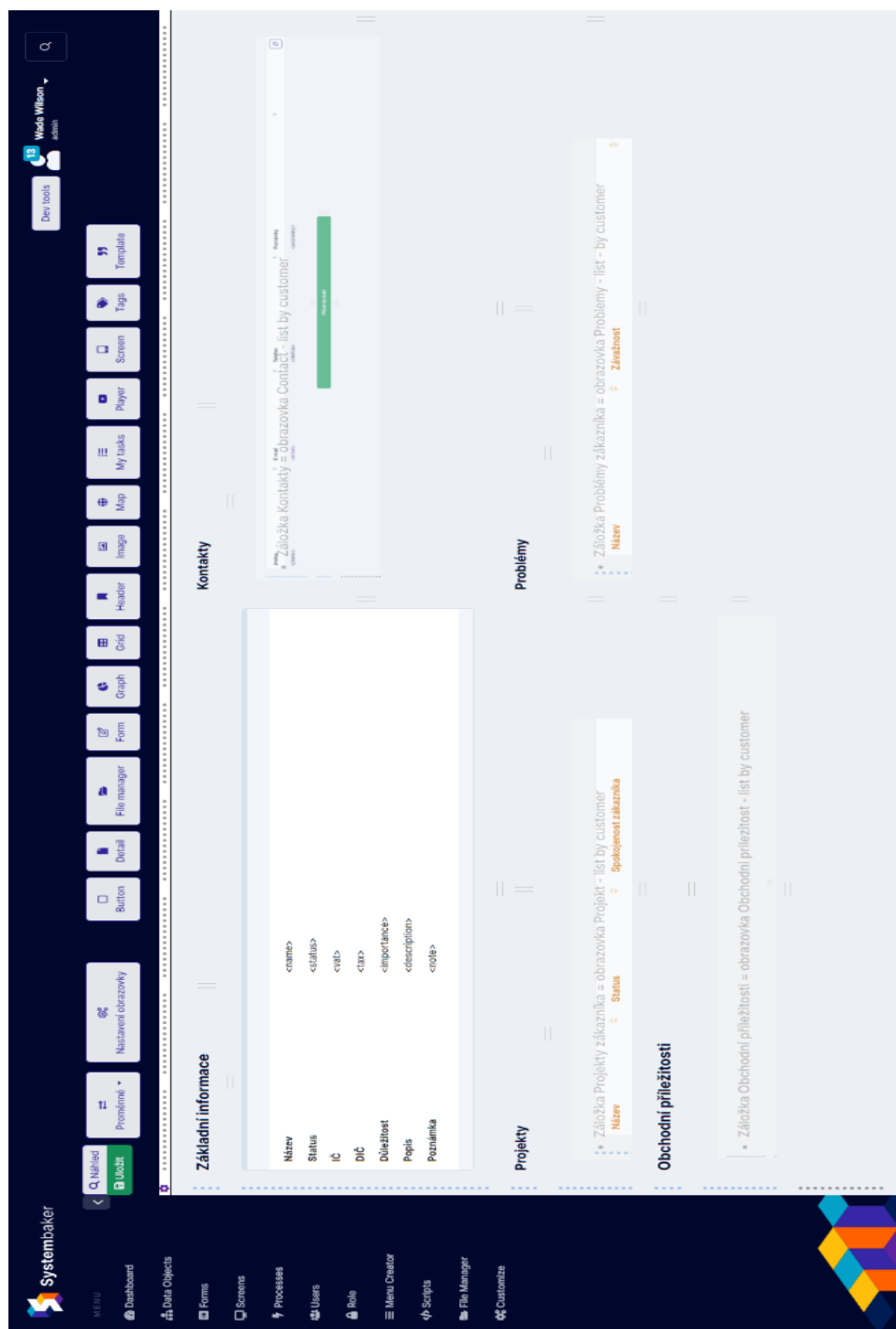
Další komponentou, se kterou se v softwaru pracuje, jsou formuláře (Forms), které zaštiťují získávání dat od uživatele, tedy vstupy a jsou většinou navázány na datové objekty, tedy jejich data se ukládají do databáze. [31]



Obrázek 11: Editace Formuláře [Vlastní zpracování]

## 2.8.2.2 Obrazovky

Dále se pracuje s komponentou obrazovky (Screens), v této komponentě se definuje vzhled a funkcionální obrazovky, které uživatelé vidí. Lze v nich třeba použít formulářů, které jsme si předtím nadefinovali, nebo lze přidávat jakékoli texty, či tlačítka a jim poté dodávat funkcionální. [31]



Obrázek 12: Editace Obrazovky [Vlastní zpracování]



### **2.8.2.3 Procesy**

Workflow tok určuje komponenta nazvaná Proces (Processes), ve které se modeluje posloupnost instrukcí, funkcí a výpočtů. Tyto procesy mohou být navázány na různé tlačítka na obrazovkách a spouštět tak požadovanou funkcionalitu. [31]

Toto jsou nejzákladnější komponenty softwaru, které v podstatě představují MVC (Model, View, Controller) architekturu vývoje softwaru. [31,34]

Model – Datové objekty; View – Formuláře a obrazovky; Controller – Procesy. [31,34]

Práce v systému je pak následující: Rovnou se vytvoří proces, tedy logika dané funkcionality, kterou tvoříme. V procesu se definují různé typy úkolů, např. úkoly pro automatizované posílání emailů, tvorby dokumentů, předávání dat do externích systémů přes API, či obrazovky, které uvidí uživatel, a případně se vytváří i formuláře pro vstupy dat od uživatel. Většinou jsou navázány tyto formuláře na datové objekty, tedy se pro ně vytvoří i tabulka v databázi. Definování procesu je hodně robustní a dovoluje modelovat i na bázi samostatných proměnných. [31]

The screenshot displays the Systembaker process editor. At the top left, the user is logged in as 'Wade Wilson' (admin). The main workspace shows a workflow diagram with the following steps:

- projekt ids** (Search icon)
- projektjid** (Code icon)
- problem select** (Search icon)
- problem/ids** (Code icon)
- problem** (Code icon)

Below the workflow, there are several toolbars and panels:

- Problem - projekt - get ids by customer**: Includes navigation and 'Nastavení procesu' (Process Settings).
- Vyhledat úkol...**: Search for tasks.
- Základní**: Basic actions like back, forward, refresh, and search.
- Datový objekt**: Data object actions like add, search, and delete.
- Ostatní**: Other actions like help and settings.

At the bottom, the 'Proměnné' (Variables) panel is visible, showing input and output variables:

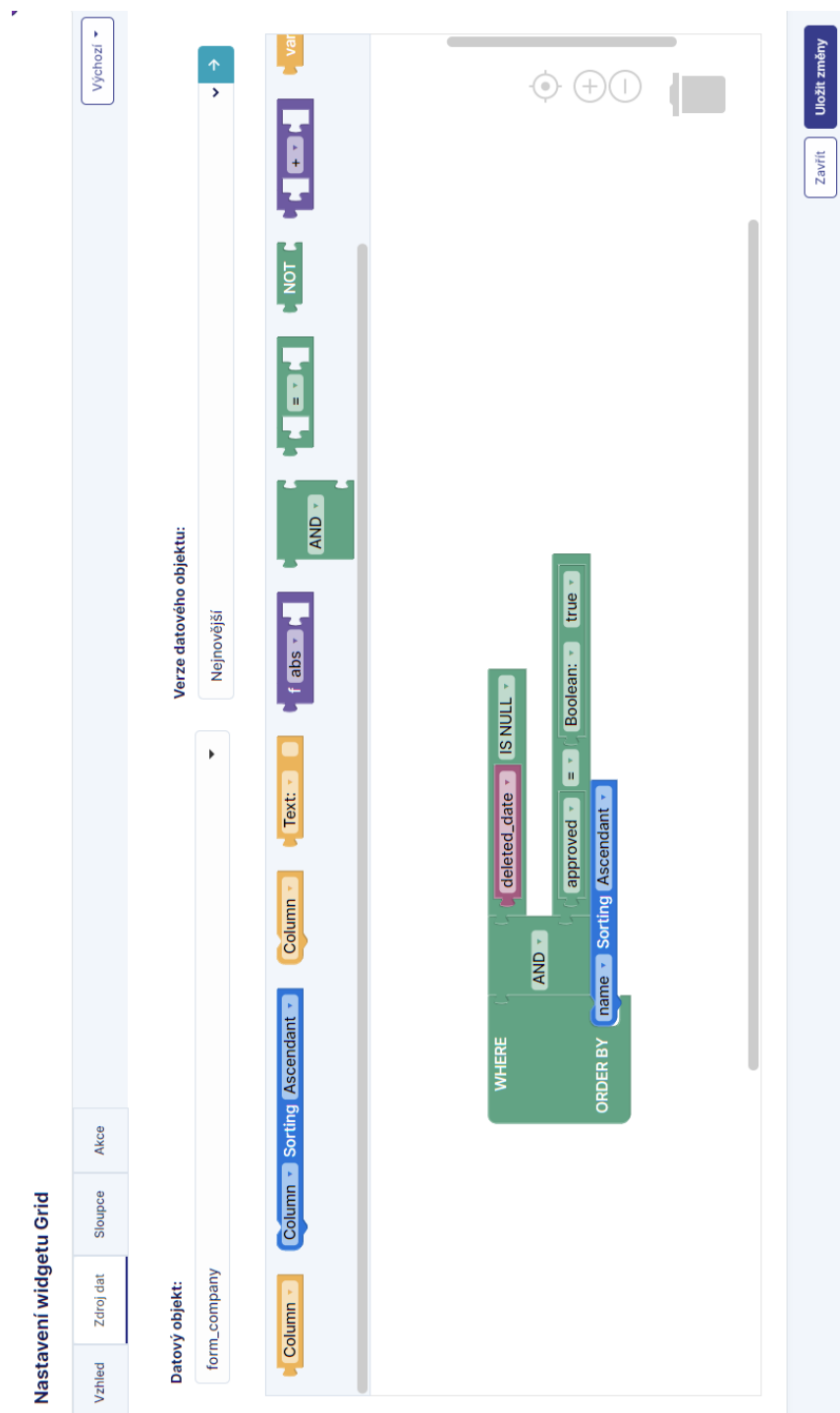
- Vstupy (Inputs)**: 'input' (Variable), 'problem/ids\_result\_array - Array result'.
- Výstupy (Outputs)**: 'output' (Variable), 'problem\_output - problem\_ids'.

The bottom navigation menu includes: Dashboard, Data Objects, Forms, Screens, Processes, Users, Role, Menu Creator, Scripts, File Manager, and Customize.

Obrázek 13: Editace Procesu [Vlastní zpracování]

## 2.8.3 Vizuální rozhodovací funkce

Na platformě se používá javascriptová knihovna (aktuálně Blockly, ale může se i měnit), která umožňuje vizuálně tvořit podmínky. Například pro přístup k datovým objektům a vyfiltrování určitých dat. [31]



Obrázek 14: Ukázka Blockly při vyhledávání dat z DO [Vlastní zpracování]

## 3 ANALÝZA PROBLÉMU A SOUČASNÉ SITAUCE

### 3.1 Základní informace o firmě Mathesio s.r.o.

Firma Mathesio s.r.o. vznikla v roce 2013 a nyní sídlí v Brně – Za divadlem 2, dříve firma sídlila v ulici Soukenická 558/3. Založil ji Ing. et Ing. Vojtěch Mates, Ph.D. Firma se zabývá vývojem software na zakázku. Firmě poskytují služby kontraktori, kteří pro firmu vyvíjí software. Tito pracovníci se ve firmě relativně mění, je zde ovšem pár stálejších, nicméně firma si udržuje stabilní počet okolo 20-ti pracovníků. Podle obchodního rejstříku firma ke konci roku 2018 dosahovala obrátu 6,4 mil. Kč. Většina zákazníků této firmy je tuzemská. [29, 30]



*Obrázek 15: Logo Mathesio s.r.o. [30]*

Společnost zpracovala za svou dobu působení již řadu projektů, například pro Ministerstvo dopravy (IS pro centrální registr dovozců), Eurolines (e-shop napojený na externí systémy), Ministerstvo práce a sociální věcí (robustní CMS – Koopolis), Legito (editor právnických smluv), ČD Cargo (3D simulátor pro strojevůdce), Ministerstvo životního prostředí (IS pro Autovraky) a mnoho dalších... [30]

### 3.2 McKinsey 7S model

#### 3.2.1 Strategie

Mathesio poskytuje vývoj softwaru na zakázku primárně jiným firmám, ale také i osobám. Neustále se snaží držet krok v oblasti nových technologií v programování, aby dodávala spolehlivý, moderní a bezpečný software. [31,32,33]

### 3.2.2 Struktura

Struktura je procesně, či projektově orientovaná, tj. jednotliví lidé mají své role a jsou v rámci procesů a projektu zapojeni dle svých rolí. Role, které zastávají jsou např. Backend, Frontend, Projektový management atd. Každá z jednotlivých rolí je zastoupena v každém týmu při řešení jednotlivých projektů a zakázek. [31,32,33]

### 3.2.3 Systémy

V oblasti systémů jsou informace o zákaznících roztroušeny v několika systémech jako Redmine, Gitlab, účetnictví Pohoda, či v obyčejných dokumentech. Toto roztroušení přímo nemusí jednotně zachycovat aktuální stav projektu – například v rámci problémů vyskytujících se na projektu (mimo těch, které přímo souvisí s produktem), nebo spokojeností zákazníka. Dále tyto systémy vhodně nepodporují onboardingové procesy, což firma aktuálně postrádá. [31,32,33]

### 3.2.4 Styl

Typ stylu, který ve firmě převládá je **Laissez-faire**, kdy panuje ve firmě velká volnost. Pracovníci pracují stylem, jaký uznají za vhodný a je zde velký prostor pro inovativní řešení. Řešení jsou diskutována, každý může vyjádřit svůj názor, který je vždy brán v potaz. [31,32,33]

### 3.2.5 Spolupracovníci

Pracovníci se specializují pomocí různých workshopů, školení, prezentací v rámci programátorských technologií, se kterými operují. Pracovníci jsou tak specializováni pro vývoj za pomoci různých technik a nástrojů jako .NET Core, AngularJS aj. Pracovníci jsou motivováni různými benefity: příspěvky a teambuilding akcemi. Mají také možnost osobního rozvoje v podobě tréninku soft skills atd. [31,32,33]

### 3.2.6 Schopnosti

Vývoj softwaru pro stálé pracovníky je již zaběhlou rutinou, kdy se nejprve zpracuje návrh řešení, zkonzultuje se se zainteresovanými stranami. Poté nastává fáze inkrementálního vývoje, kdy se postupně dodávají funkcionality daného softwaru s tím,

že jsou průběžně testovány a je kontrolováno, jestli splňují požadavky zákazníků. Velmi osvědčenou praktikou ve firmě je také tzv. „Code review“, kdy se po naprogramování nějaké funkcionality ještě kontroluje jiným (převážně zkušenějším) pracovníkem daný kód. Nebo se tím kontroluje, či lze zlepšit kvalita a výkonnost daného kódu a zda neobsahuje chyby, čímž se dosahuje dvojí kontroly kvality. [31,32,33]

### **3.2.7 Sdílené hodnoty**

Ve firmě panuje přátelská atmosféra, lidé si pomáhají, rádi se spolu baví i mimo pracoviště a obecně se zastává nepsané pravidlo: „Já ulehčím práci tobě, ty ji ulehčíš mě.“ Dále také panuje v komunikaci upřímnost, díky které si pracovníci mohou mezi sebou sdělit i kritiku, která je samozřejmě konstruktivní. Každý může svobodně vyjadřovat své názory, a i noví pracovníci brzy pochopí tento styl komunikace a dokážou se díky tomu rychle začlenit do týmu. [31,32,33]

## **3.3 Zadání a požadavky na CRM**

Pro zlepšení interní situace v rámci systémů majitel dospěl k rozhodnutí, že je ve firmě potřeba systém, který bude sledovat procesy kolem zákazníků v souvislosti s aktuálními projekty a spokojeností zákazníků.

System má umožňovat:

- Vidět a chápat spokojenost zákazníků u projektů, co narušuje tuto spokojenost a co děláme / co bychom měli dělat pro to, aby se zlepšila.
- Vidět a chápat v rozumné struktuře základní informace ke konkrétnímu zákazníkovi / projektu a link na další zdroje.
- Možnost vytvářet a sledovat obchodní příležitosti, tedy nové potenciální projekty.
- Rychlé zaučení nového pracovníka do business kontextu projektu, případně rychlá výměna projektového manažera.
- Pomoc při plánování týdenních priorit.
- Zjistit potenciální problémy okolo deadlinů u konkrétních projektů. [31]

## 3.4 Riziková politika

Pro zpracování rizikové politiky byla využita skórovací metoda, která identifikuje rizika a za použití „Pravděpodobnosti“ (která je udávána v rozmezí 0–100 %) a „Míry dopadu“ (v rozmezí 1-10) je ohodnotí. Součinem „pravděpodobnosti“ a „míry dopadu“ rizika získáme výslednou hodnotu rizika v rozmezí 0-10. Čím vyšší je tato hodnota, tím závažnější je riziko. Poté se navrhnou opatření ke snížení pravděpodobnosti rizik, a tím vznikne nová výsledná hodnota rizik. Pro porovnání rizik před opatřením a po opatření se využije pavučinového grafu. [39]

### 3.4.1 Identifikace rizik

V rámci navrhované změny byly identifikovány tyto rizika:

- Špatně zvolená technologie vývoje
- Špatně navržené uživatelské rozhraní
- Podhodnocení časového odhadu.
- Špatný návrh řešení
- Chybějící školení pracovníků
- Chybějící dokumentace systému
- Systém nezvýší produktivitu práce
- Systém nebude odpovídat zadání

### 3.4.2 Ohodnocení rizik skórovací metodou

*Tabulka 1: Klasifikační stupnice ohodnocení pravděpodobnosti a dopadu rizika [Vlastní zpracování]*

Pravděpodobnost (%)	Pravděpodobnost vzniku rizika	Hodnota dopadu	Dopad rizika na projekt
0–20	Téměř žádná	1–2	Minimální
21–40	Nízká	3–4	Méně významný
41–60	Pravděpodobná	5–6	Významný
61–80	Více pravděpodobná	7–8	Velmi významný
81–100	Vysoká pravděpodobnost	9–10	Kritický

Dle **tabulky 1** uvádějící klasifikaci pravděpodobnosti a dopadu rizik byly ohodnoceny identifikovaná rizika – viz **tabulka 2**. Kde **Číslo** je jednoznačná identifikace rizika, **P** je pravděpodobnost v procentním vyjádření, **D** je míra dopadu rizika, **H** představuje součin pravděpodobnosti a míry dopadu, tedy výslednou hodnotu rizika.

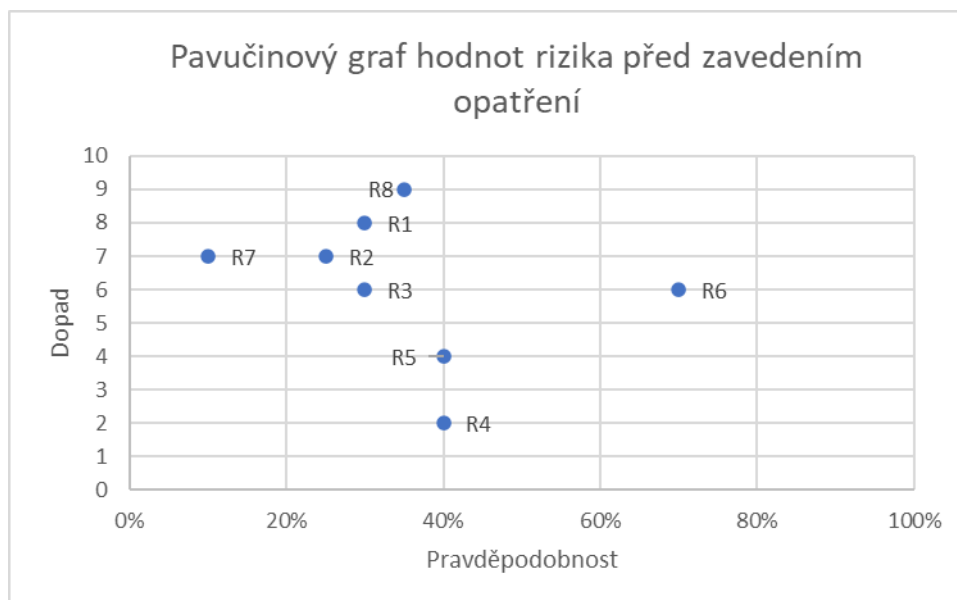


**Tabulka 2: Ohodnocení rizik [Vlastní zpracování]**

<b>Číslo</b>	<b>Hrozba</b>	<b>Scénář</b>	<b>P (%)</b>	<b>D</b>	<b>H</b>
R1	Špatně zvolená technologie vývoje.	Systém bude pomalý.	30	8	2,4
R2	Špatně navržené uživatelské rozhraní.	Systém nebude uživatelsky přívětivý.	25	7	1,75
R3	Podhodnocení časového odhadu.	Nastane zpoždění s čímž jsou vázány další náklady	30	6	1,8
R4	Špatný návrh řešení.	Chybějící funkce.	40	2	0,8
R5	Chybějící školení pracovníků.	Pracovníci nebudou umět ovládat systém.	40	4	1,6
R6	Chybějící dokumentace systému.	Problémy s pochopením systému a obtížné úpravy v kódu.	70	6	4,2
R7	Systém nezvýší produktivitu práce.	Ztráta peněz.	10	7	0,7
R8	Systém nebude odpovídat zadání.	Systém neplní požadovanou funkčnost.	35	9	3,15

### 3.4.3 Mapa rizik před opatřeními

Na **obrázku 16** je vyobrazen pomocí bodového grafu mapa jednotlivých rizik před opatřeními.



**Obrázek 16:** Pavučinový graf hodnot rizika před zavedením opatření [Vlastní zpracování]

Z grafu na **Obrázku 16** vyplývá, že z hlediska **pravděpodobnosti** je nejvíce ohrožující riziko R6 (chybějící dokumentace), riziko R5 (chybějícího školení pracovníků) a riziko R4 (špatného návrhu řešení).

Z hlediska **dopadu** jsou závažná: riziko R8 (systém nebude odpovídat zadání), riziko R1 (špatně zvolená technologie vývoje), riziko R2 (špatně navržené uživatelské rozhraní) a riziko R7 (systém nezvýší produktivitu práce).

Nejvíce ohrožujícími riziky podle **výsledné hodnoty rizika** je riziko R6 (chybějící dokumentace systému) a riziko R8 (systém nebude odpovídat zadání). Tato rizika jsou nejdůležitějšími kandidáty na návrh opatření ke snížení pravděpodobnosti či míry dopadu rizika.

### 3.4.4 Ohodnocení rizik po zavedení opatření

V **tabulce 3** jsou popsány návrhy ke snížení pravděpodobnosti identifikovaných rizik, také jsou uvedeny nová ohodnocení pravděpodobností (**Nová P**), míry dopadů (**Nový D**) a výsledné hodnoty rizik (**Nová H**) po zavedení opatření.

- **Chybějící dokumentace systému**

Pro snížení pravděpodobnosti výskytu tohoto rizika je navrženo vyčlenit speciální čas po implementaci, ve kterém se vytvoří dokumentace.

- **Systém nebude odpovídat zadání**

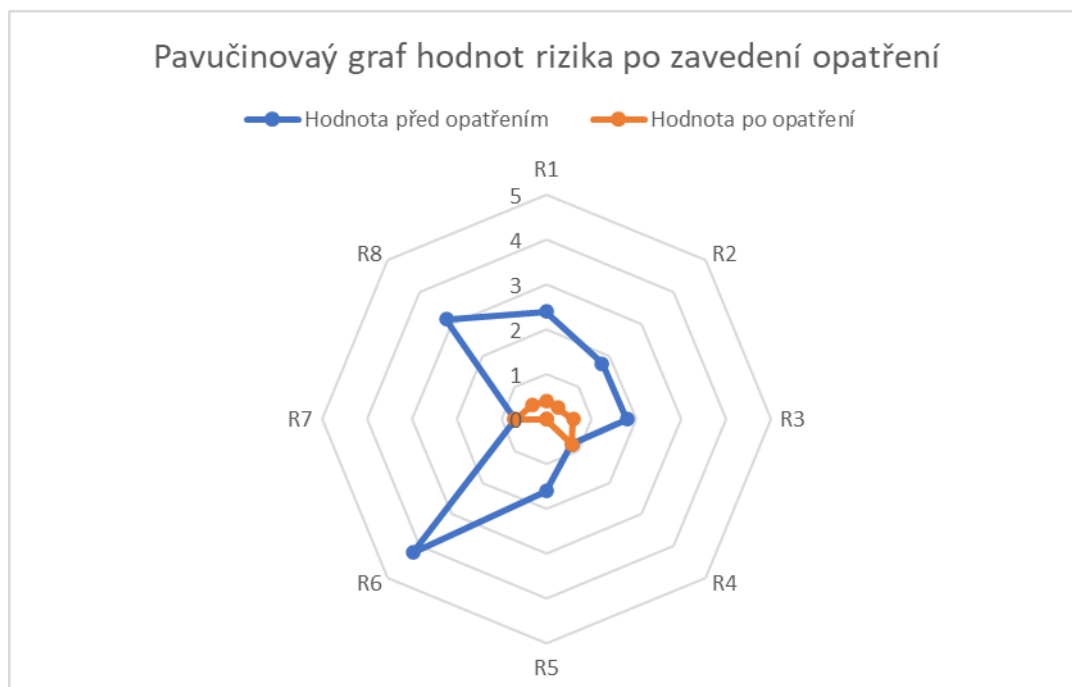
Při implementaci softwaru může snadno dojít k odchýlení se od zadání a záměru projektu. Proto existují moderní agilní praktiky vývoje, které stanovují tzv. sprinty, ve kterých jsou definovány přesně jaké nové funkce se naprogramují a také otestují. Tím dochází k inkrementálnímu vývoji, který lze plánovat dopředu s ohledem na původní zadání projektu.

*Tabulka 3: Návrh opatření [Vlastní zpracování]*

Číslo	Návrh opatření	Nová P (%)	Nový D	Nová H
R1	Vyčlenit čas na pečlivé zvážení dostupných technologií a vybrání nejlepší.	5	8	0,4
R2	Průběžné zapojování testerů s výsledným hodnocení UX.	5	7	0,35
R3	Snaha o přesnější časový odhad na základě minulých zkušeností a propočtů.	10	6	0,6
R4	Retence, funkce se případně doprogramují.	40	2	0,8
R5	Provést po implementaci školení pro všechny, jak systém používat.	0	4	0
R6	Po implementaci vyčlenit čas, pro vytvoření dokumentace.	0	6	0
R7	Retence.	10	7	0,7
R8	Implementovat pomocí praktik agilního vývoje s průběžnými kontrolami ohledně zadání.	5	9	0,45

### 3.4.5 Mapa rizik po opatření

Na **obrázku 17** je vyobrazena mapa rizik po zavedení opatření za pomoci pavučinového grafu.



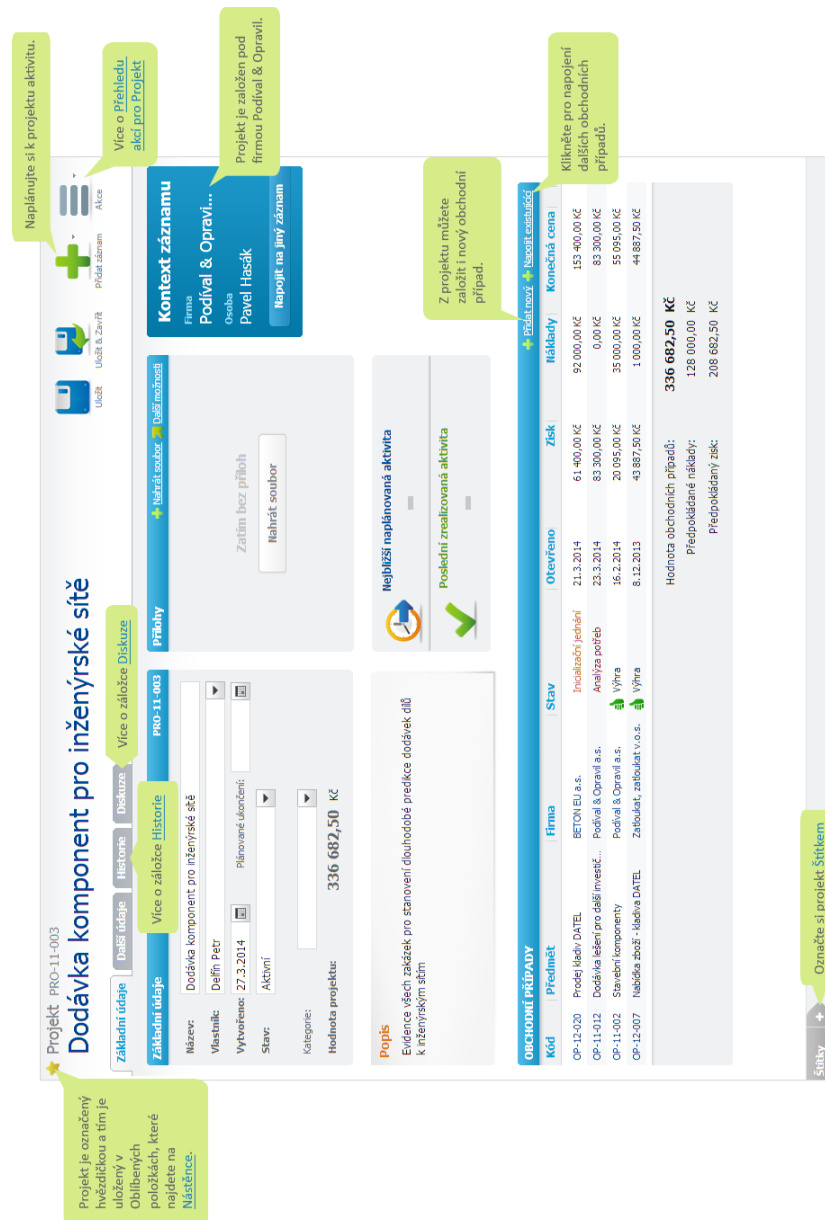
**Obrázek 17:** Pavučinový graf hodnot rizika po zavedení opatření [Vlastní zpracování]

Z pavučinového grafu zjišťujeme, že hodnoty po zavedení opatření byly výrazně sníženy. Proto navrhovaná opatření jsou žádoucí pro zajištění úspěchu této implementace.

## 3.5 Porovnání SB s ostatními produkty

### 3.5.1 Raynet

Jedná se o hotové řešení, které se platí měsíčním paušálem a to přesně 500 Kč za každého uživatele. Umožňuje správu zákazníků, obchodních případů a zdá se i projektů. Dále obsahuje reporty, kalkulace objednávek, správu dokumentů a kalendář. Toto řešení není úplně vhodné a v některých případech je i nedostačující pro firmu Mathesio [35]



Obrázek 18 : Vzhled detailu projektu Raynet [37]

### 3.5.2 Blue CRM

Firma Sysportal, s.r.o. nabízí stavbu CRM na míru. S tím, že na svých webových stránkách udává průměrnou cenu tohoto řešení okolo 160 000,-Kč. Je dost pravděpodobné, že by firma Mathesio tímto způsobem získala systém, jaký potřebuje. Jenže řešení, jenž nabízí SystemBaker poskytuje možnost si časem řešení dopravit podle toho, jak se procesy ve firmě mění. Taková úprava by byla u tohoto řešení ale zpoplatněna. [36]



**Obrázek 19: Detail Projektu – Blue CRM [36]**

### 3.5.3 OutSystems

Společnost OutSystems existuje již od roku 2001 a má rovněž svoji low-code platformu. Tato low-code platforma je hodně robustní a lze v ní provádět mnoho sofistikovaných aplikací. Platforma obsahuje mnoho funkcionalit, jenže jejich dokumentace neobsahuje mnoho příkladů užití. Firma spíše cílí na přímou komunikaci, ke zjištění, jak pracovat s těmito funkcionalitami. Pro začátečníky na platformě to může být demotivující a odrazující. [38]

Nicméně je velkým plusem, že má základní verzi tohoto low-code softwaru zdarma a rovněž nabízí i online training v platformě. Tedy tento systém by mohl být alternativou, pro implementaci vlastního CRM systému. [38]

### **3.5.4 Volba softwaru ke zpracování řešení**

Ke zpracování požadovaného CRM systému, byl zvolen software SystemBaker. Důvodem zvolení tohoto softwaru je primárně to, že se jedná o software dané firmy. Firma proto nebude mít žádné licenční náklady spojené s vývojem tohoto CRM. Software ještě není publikován. Proto se tak touto prací otestuje a poslouží i k opravě případných chyb. Práce pak bude také i jedním z reprezentativních produktů vytvořených za pomoci tohoto softwaru. [31]

Další výhodou, která přispěla k volbě softwaru je ta, že má velmi intuitivní ovládání a je u něj nejvyšší možná míra supportu v rámci interního týmu firmy. [31]

## 4 VLASTNÍ NÁVRH ŘEŠENÍ A PŘÍNOS NÁVRHU

V této části je návrh možné implementace požadovaného CRM systému firmou.

### 4.1 Případy užití ke splnění požadavků

- a) **Vidět a chápat spokojenost zákazníků u projektů, co narušuje tuto spokojenost a co děláme / co bychom měli dělat pro to, aby se zlepšila.**
  - Evidence zákazníků
  - Evidence projektů se sledováním spokojenosti zákazníka, problémů a ponaučení.
- b) **Vidět a chápat v rozumné struktuře základní informace ke konkrétnímu zákazníkovi / projektu a link na další zdroje.**
  - Možnost zobrazovat detail o každém zákazníkovi či projektu.
  - Detail obsahuje veškeré potřebné informace.
- c) **Možnost vytvářet a sledovat obchodní příležitosti, tedy nové potenciální projekty.**
  - Evidence obchodních příležitostí.
- d) **Rychlé zaučení nového pracovníka do business kontextu projektu, případně rychlá výměna projektového manažera.**
  - Detail projektu obsahuje relevantní informace, pro začlenění pracovníka do projektu.
- e) **Pomoc při plánování týdenních priorit.**
  - Evidence deadlinů, problémů a ponaučení.
  - Výpis všech deadlinů projektů řazených dle nejbližšího data.
  - Výpis všech aktuálních problémů a ponaučení z projektů.
- f) **Zjistit potenciální problémy okolo deadlinů u konkrétních projektů.**
  - Možnost evidovat u deadlinů informace jako důvod, název a závažnost.



## 4.2 Návrh databázové struktury

Bude třeba tabulek: Zákazník; Kontakt; Obchodní příležitost; Projekt; Deadline; Problém; Ponaučení; Cena; Sleva; Pracovník; Uživatel.

### 4.2.1 Tabulky

Všechny tabulky obsahují sloupeček pro unikátní identifikaci jednotlivých záznamů, tzv. Primární klíč.

#### 4.2.1.1 Zákazník

Záznamy v tabulce Zákazník představují jednotlivé zákazníky (firmy) a informace o nich. Mezi tyto informace patří: Název, Popis, Důležitost, Status, IČ, DIČ, Poznámka a Datum vytvoření a smazání. Datum smazání je zde kvůli tzv. „soft delete“ – tedy pokud se záznam v aplikaci smaže, tak se pouze nastaví tento atribut na aktuální čas a při výpisu se zobrazují jen ty záznamy, které mají Datum smazání s hodnotou NULL. Díky tomu se nevyhodí důležité informace o minulých zákaznících a je zde i možnost obnovení již smazaného zákazníka.

#### 4.2.1.2 Kontakt

Kontakt je v relaci se zákazníkem. Obsahuje informace o zaměstnancích, kteří zastupují zákazníka (firmu). Kontakt obsahuje sloupečky: Jméno, Telefon, E-mail, Poznámky a cizí klíč na Zákazníka, který slouží k identifikaci, ke kterému zákazníkovi daný kontakt náleží.

#### 4.2.1.3 Obchodní příležitost

Entita obchodních příležitostí obsahuje evidenci potenciálních nových projektů. Je zde relace na Zákazníka ve formě cizího klíče (ZakaznikId). V tabulce jsou sloupečky: Název, Popis, Stav, Cenový odhad, Odhad práce, Poznámky a Datумы vytvoření a smazání. I pro tuto tabulku je žádoucí, aby měla možnost Soft Delete a nepřicházelo se tak o informace.

#### **4.2.1.4 Projekt**

Entita Projekt je jednou z hlavních tabulek tohoto systému. Má relaci na Zákazníka ve formě cizího klíče (ZakaznikId). Obsahuje veškeré informace o jednotlivých produktech: Název; High Level Popis; Hlavní Use-Case; Detaily Projektu (link na zadávací dokumentaci na cloudovém úložišti); Frontend a Backend Technologie; Redmine; Postup ticketu v Redmine; Postup vývoje; Pravidelnost meetingů; Status; Spokojenost zákazníka; Tok komunikace a Datумы vytvoření a smazání. Projekt je rozhodně tabulkou, která musí podporovat Soft Delete. O tyto informace nechceme přijít a smazání dat nepřichází v úvahu.

#### **4.2.1.5 Deadline**

Záznamy v tabulce deadline slouží k evidenci jednotlivých deadlinů v projektech – vazba na Projekt (cizí klíč: ProjektId). Obsahuje Název, Závažnost, Datum soft deadlinu, Datum hard deadlinu a Důvod hard deadlinu.

#### **4.2.1.6 Problém**

Slouží k evidenci problémů, které nastaly u jednotlivých projektů. Je tedy jasné, že obsahuje vazbu na Projekt (cizí klíč: ProjektId). Sloupečky, které obsahuje jsou následující: Název, Závažnost, Detail, Důsledky, Datum vytvoření (Kdy se poprvé problém vyskytl), Vyřešen (pravdivostní hodnota).

#### **4.2.1.7 Ponaučení**

Jednoduchá tabulka navázaná na Projekt (cizí klíč: ProjektId), která umožňuje vidět, čím jsme se u projektu ponaučili. Obsahuje Název a Detail.

#### **4.2.1.8 Cena**

Tato tabulka se zaměřuje na informace k placení a cenové rozsahy jednotlivých projektů (vazba, cizí klíč: ProjektId). Obsahuje: Nám předpokládanou cenu – Dolní a Horní odhady; Zákazníkem předpokládanou cenu – Dolní a Horní odhad; Styl fakturace; Sjednanou hodinovku (hodinovou odměnu práce) a Reálnou cenu (může to být aktuální cena, či cena při uzavření projektu).

### 4.2.1.9 Sleva

















Taktéž velmi prostá tabulka, která uvádí nutnost poskytnutých slev zákazníkovi na projektu (vazba, cizí klíč: ProjektId). Je v ní uveden Název slevy a Detail, proč byla poskytnuta.

### 4.2.1.10 Pracovník

V této tabulce jsou vazby na Projekt a Uživatele. Dalo by se říct, že to je rozpisová M:N tabulka Projektu a Uživatele. Tedy 1 Projekt může mít N Uživatelů. A 1 Uživatel může mít N Projektů. S tím, že je v této tabulce uvedeno i specifikum vztahu uživatele na projektu – Role (Backend, Frontend, Tester, Projektový manažer, Devops).

### 4.2.1.11 Uživatel

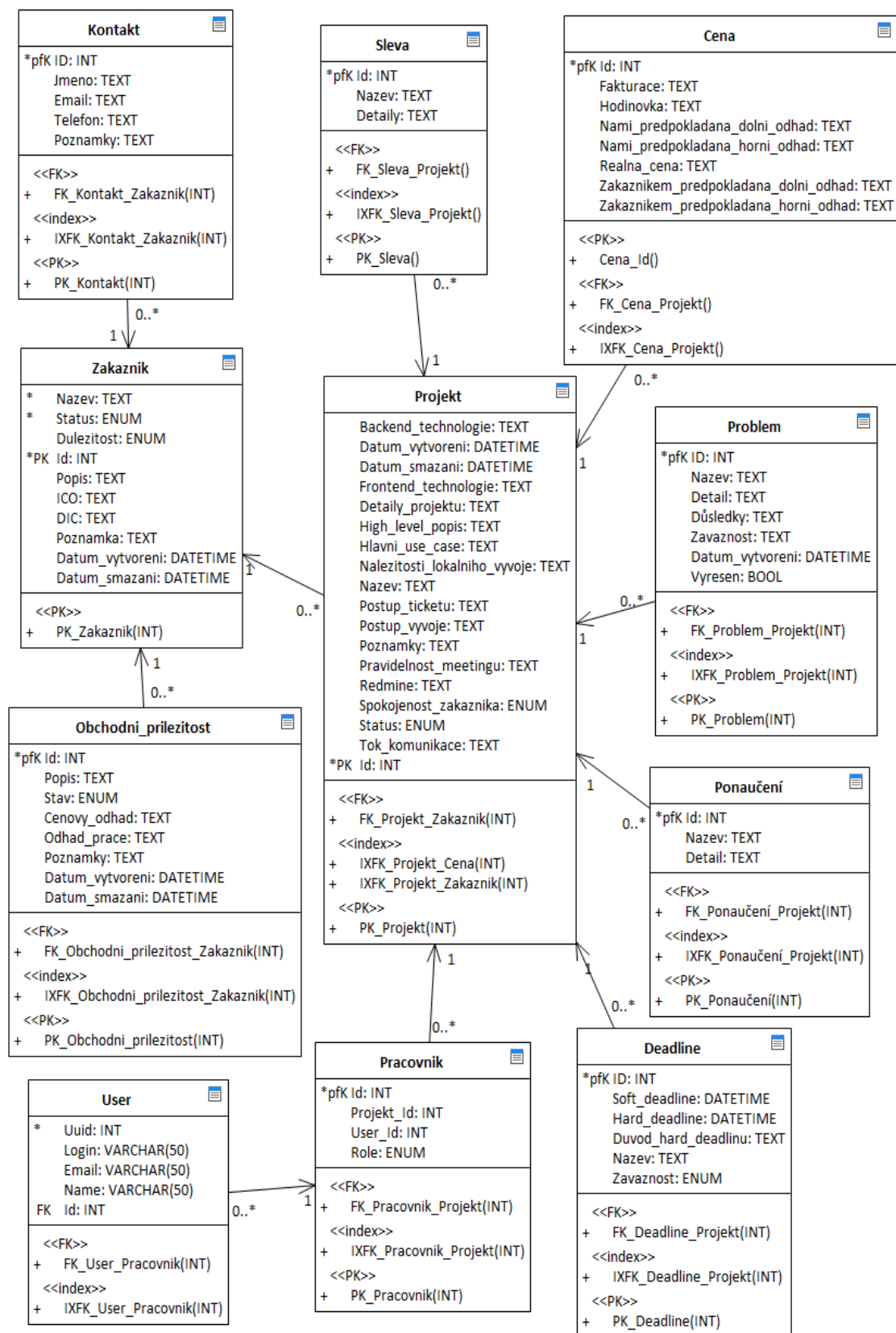
Tato tabulka je již vytvořena automaticky v rámci platformy SystemBaker, slouží jako vstup do platformy a identifikace jednotlivých uživatelů. Správa jednotlivých uživatelů se provádí v menu Admina – lze uživatele vytvořit, upravit a smazat (**Obrázek 21**).

UUID	Uživatel	Jméno	E-mail	
183f8796-700c-446f-b30f-32561a91ec2	billy	Billy Earl	billy-earl@example.com	 
48541ee2-0f61-4919-990c-52178a265c87	Ferdinand	Ferdinand Developer	ferdinand-test@example.com	 
7d2d5d0f-93a0-4c12-a08a-dae3fa4c157a	pokus	Hokus Pokus	obrtik@mathesio.cz	 
9230d3ac-4512-457b-9837-ba6f6892f128	admin	Wade Wilson	developers@mathesio.com	 
bw2029a9-b09a-474f-853a-6ff6520ab398	tom	Tomáš Tomek	developers@mathesio.com	 
c791c8ff-156b-462f-907d-4927d46643d	Dominik	Dominik Moudry	dominik-moudry@example.test	 
d7e7f723-b86a-460f-87ea-faa3b22a8444	Bernard	Bernard Developer	obrtik@mathesio.cz	 
e196c92d-4485-44c1-92d3-78efe06d25d2	manager	Projektový manažer	josef.obrtik@vut.cz	 

Obrázek 20: Správa uživatelů v SystemBakeru [Vlastní zpracování]

## 4.2.2 Relační vztahy

Zákazník : Kontakt	1 : N	Zákazník může mít N kontaktů
Zákazník : Projekt	1 : N	Zákazník může mít N projektů
Zákazník : Obchodní příležitost	1 : N	Zákazník může mít N o. příležitostí
Projekt : Deadline	1 : N	Projekt může mít N deadlinů
Projekt : Problém	1 : N	Projekt může mít N problémů
Projekt : Ponaučení	1 : N	Projekt může mít N ponaučení
Projekt : Cena	1 : N	Projekt může mít N cen
Projekt : Sleva	1 : N	Projekt může mít N slev
Projekt : Pracovník	1 : N	Projekt může mít N pracovníků
Uživatel : Pracovník	1 : N	Uživatel může mít N pracovníků



Obrázek 21: Databázová struktura [Vlastní zdroj]

## **4.3 Případy užití podle entit**

V této části jsou zpracovány všechny případy užití, které je potřeba implementovat u daných datových objektů (entit). Kromě entity uživatele, tato entita je přímo spravována platformou. Lze proto přímo bez jakékoli vlastní implementace řešit přidávání, úpravu a odstraňování uživatelů.

### **4.3.1 Zákazník**

Vytvoření, Detail, Úprava, Soft smazání, Výpis všech, Výpis smazaných, Obnovení, Detail z pohledu zaměstnance, Výpis všech z pohledu zaměstnance

### **4.3.2 Kontakt**

Vytvoření, Smazání, Úprava, Výpis všech, Výpis podle zákazníka, Výpis podle zákazníka z pohledu zaměstnance

### **4.3.3 Obchodní příležitost**

Vytvoření, Úprava, Soft smazání, Detail, Detail z pohledu zaměstnance, Seznam smazaných, Obnovení smazané, Seznam všech, Seznam všech z pohledu zaměstnance, Seznam všech podle zákazníka

### **4.3.4 Projekt**

Vytvoření, Úprava, Detail, Soft smazání, Výpis všech, Výpis podle zákazníka, Detail z pohledu zaměstnance, Výpis všech z pohledu zaměstnance, Výpis smazaných, Obnovení smazaného projektu

### **4.3.5 Deadline**

Vytvoření, Úprava, Smazání, Výpis všech, Výpis všech z pohledu zaměstnance, Výpis všech podle projektu, Výpis všech podle projektu z pohledu zaměstnance

### 4.3.6 Problém

Vytvoření, Upravení, Výpis podle zákazníka, Výpis podle zákazníka z pohledu zaměstnance, Výpis podle projektu, Výpis podle projektu z pohledu zaměstnance, Výpis všech, Označit problém za vyřešený, Výpis všech problémů z pohledu zaměstnance

### 4.3.7 Ponaučení

Vytvoření, Úprava, Smazání, Výpis všech, Výpis všech z pohledu zaměstnance, Výpis všech podle projektu, Výpis všech podle projektu z pohledu zaměstnance

### 4.3.8 Cena

Vytvoření, Úprava, Smazání, Výpis podle projektu

### 4.3.9 Sleva

Vytvoření, Úprava, Smazání, Výpis podle projektu

### 4.3.10 Pracovník

Vytvoření vztahu pracovníka, Upravení vztahu, Smazání vztahu, Výpis pracovníků projektu, Výpis pracovníků projektu z pohledu zaměstnance

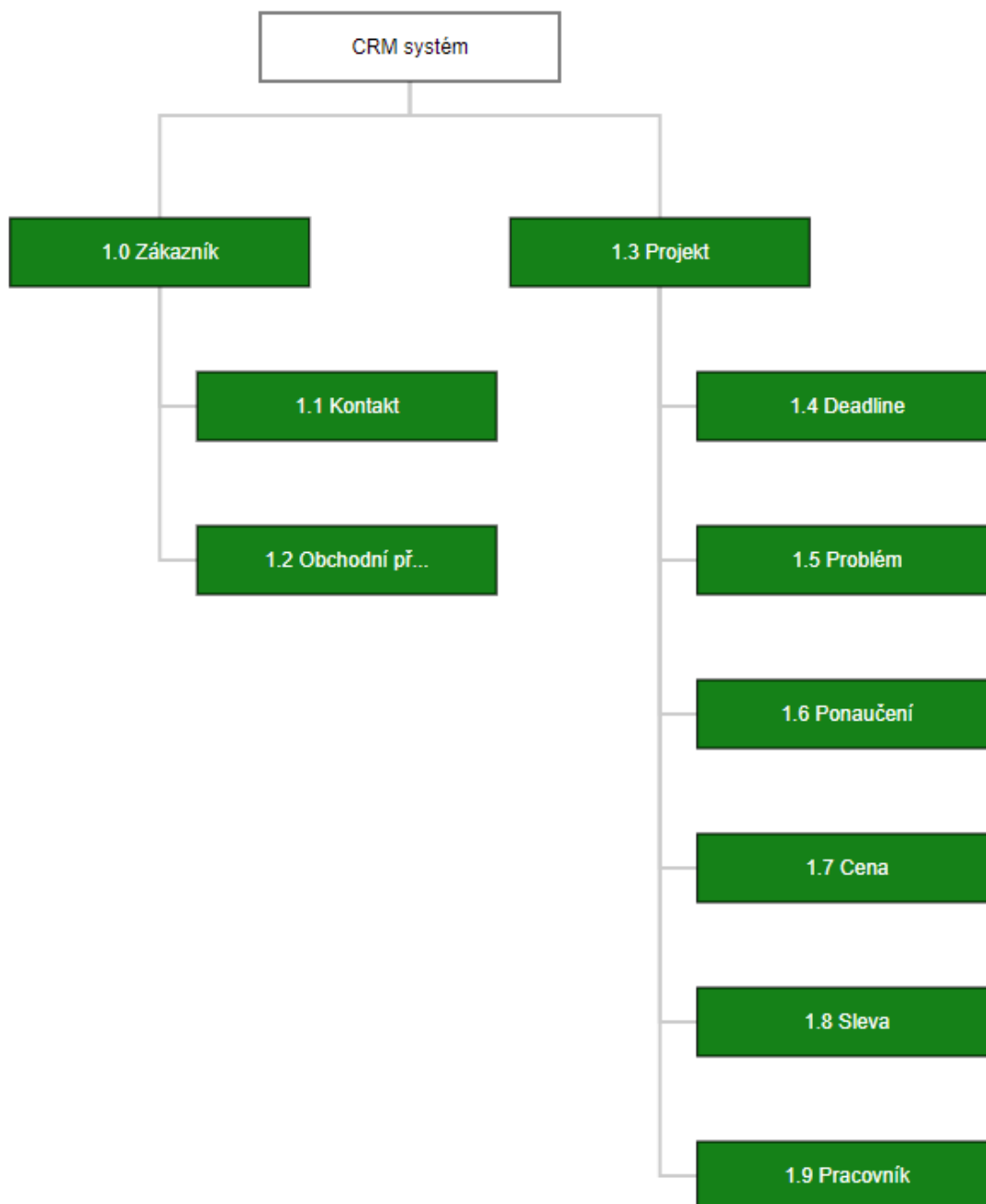
## 4.4 Role

V CRM bude jedna role – Zaměstnanec s tím, že bude mít pod-rolí Projektový manažer. Projektový manažer má práva admina, proto může cokoli editovat, zatímco zbylí spolupracovníci mohou pouze zobrazovat, přidávat a maximálně upravovat záznamy.

## 4.5 Work Breakdown Structure

Na **obrázku 23** je vyobrazena WBS, která označuje implementační body CRM systému, které je třeba splnit. **Prefix 1.X** znázorňuje postup implementace – tedy nejdříve se implementuje entita Zákazníka a její základní CRUD operace – Create (vy-

tvoření), Update (upravení), Read (čtení – výpisy dat) a Delete (smazání, v tomto případě soft) za pomoci procesů a obrazovek. Poté se na zákazníka napojí Kontakt, Obchodní příležitost a Projekt a rovněž se pro tyto entity vytvoří základní CRUD operace.



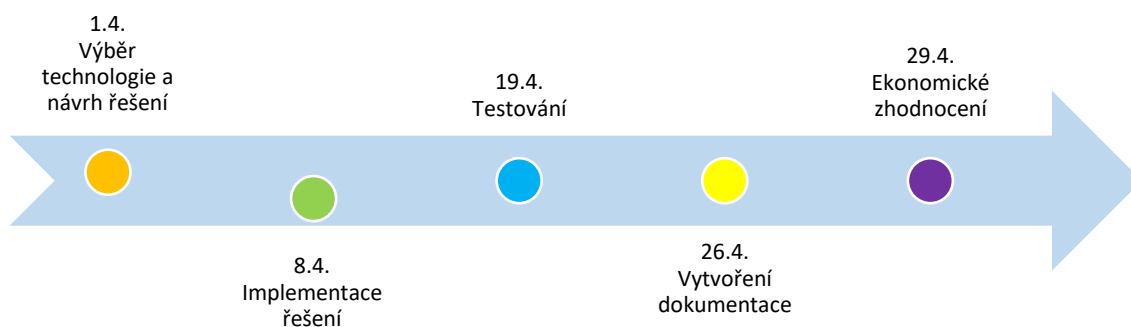
**Obrázek 22: WBS struktura pro implementaci [Vlastní zpracování]**



Poté se začnou vytvářet další entity závislé na Projektu, tedy Deadline, Problém, Ponaučení, Cena, Sleva, Pracovník. Pro všechny tyto entity se rovněž vytvoří základní CRUD operace.

## 4.6 Časová analýza

Časový odhad návrhu zpracování byl naplánován do několika bodů: viz časová osa na **Obrázku 20**. Krátkých časových úseků mezi jednotlivými body lze dosáhnout díky výběru svižné low-code platformy. Zpracování řešení proběhne v měsíci dubnu 2021, a budou se plnit tyto úkoly: Výběr technologie a návrh řešení, Implementace, Testování, Vytvoření dokumentace a vytvoření Ekonomického zhodnocení. Odhaduje se tedy, že zpracování nového CRM systému, bude hotovo v rámci jednoho měsíce.



*Obrázek 23: Časová linka návrhu zpracování [Vlastní zpracování]*

## 4.7 Implementace

V této části popisuji, jak jsem tento CRM systém implementoval. Z velké části se ale implementace opakuje, z tohoto důvodu bude lepší popsat pouze využití základní metodiky implementace, které jsem využíval. Z těchto metodik lze snadno získat představu, jak platforma funguje a jak se v ní implementuje. **Aplikace je implementována na adrese <https://sandbox.systembaker.com>**

**Přístupy do aplikace jsou následující:**

- **Projektový manažer: manager : demo** – (ve formátu login : heslo)
- **Zaměstnanec: bernard : demo**

## 4.7.1 Zákazník

Nejprve jsem vytvořil formulář napojený na datový objekt. DO se vytvoří automaticky při uložení formuláře, není potřeba nastavovat žádné datové typy. Všechny sloupce se nastaví automaticky podle vybraného typu inputu formuláře.



*Obrázek 24: Možnosti inputů [Vlastní zpracování]*

Postupoval jsem dle návrhu databáze, tedy vytvořil ve formuláři vstupy pro Název, Status, Důležitost, Popis, IČ, DIČ, Poznámku, Datum vytvoření a Datum smazání.

Téměř všechny atributy zákazníka jsem nastavil na textový vstup, kromě datumů a důležitosti se statusem. Status jsem nastavil jako výběr z možností: Aktivní, Neaktivní, Hlavní, Hlavní – neaktivní, Ztracený. Důležitost jsem nastavil taktéž jako výběr z možností, u této možnosti je možné nastavit hodnotu, která se ukládá do databáze a tzv. nadpis, který slouží jako zobrazovaná hodnota, př. hodnota 1 se uloží do databáze, ale v systému se zobrazuje jako Nejmenší. Tímto způsobem jsem seřadil a nastavil hodnoty 1 až 5 pro Nadpisy: Nejmenší, Malá, Střední, Velká a Největší, díky čemuž je lze snadno řadit při výpisu v tabulce – od nejdůležitějších po nejméně důležité.

Protože je požadováno, aby byl zákazník soft deletable, tak se ve formuláři vyskytuje i datum smazání, což není žádoucí, aby uživatel vyplňoval spolu s datem vytvoření. Z tohoto důvodu je nutno vstupní formulář uzpůsobit. Využil jsem tedy možnosti vytvořit kopii formuláře s názvem Zákazník – Input.

### Vytvoření nového zákazníka

**Název**

**Status**

**Důležitost**

**Popis**

**IČ**

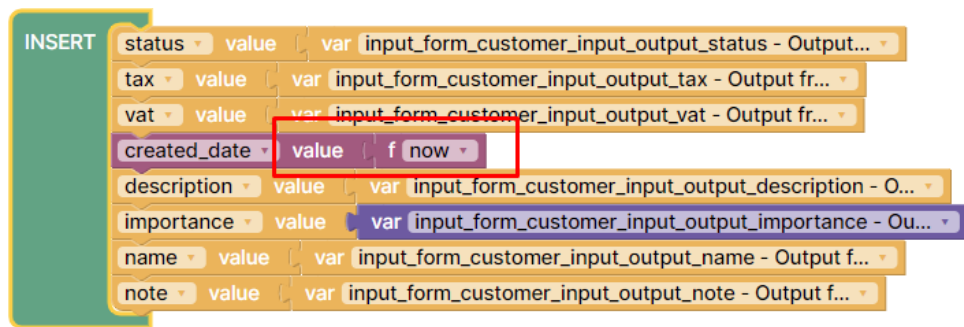
**DIČ**

**Poznámka**

**Odeslat**

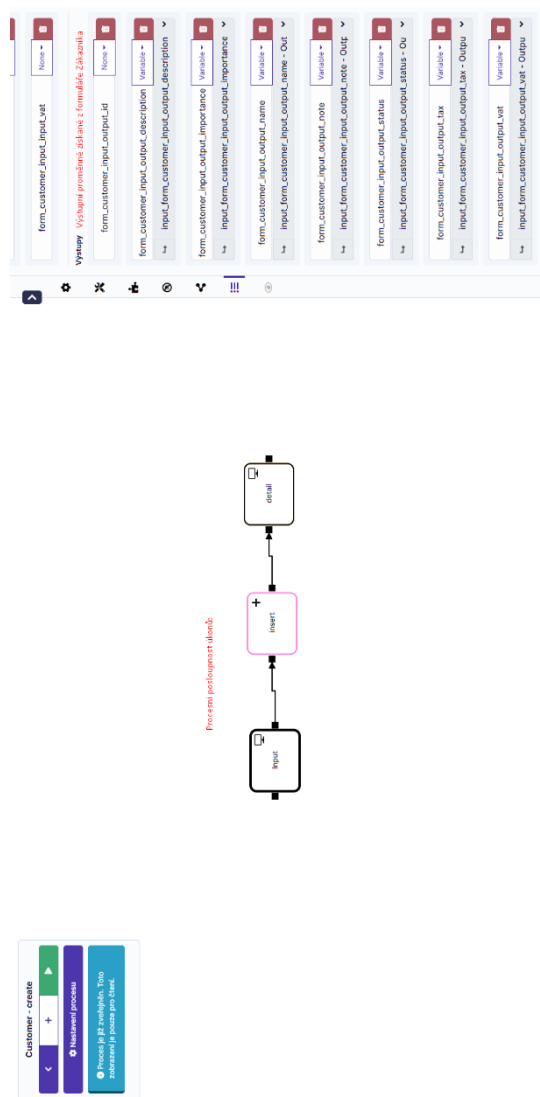
**Obrázek 25: Formulář pro vytvoření zákazníka [Vlastní zpracování]**

V duplikátu jsem zvolil možnost uložení hodnot do proměnných místo do datového objektu a odstranil nežádoucí vstupy. Tímto způsobem je možno zobrazit na obrazovce uživateli formulář, který má jen potřebná vstupní políčka a zadané hodnoty se mi uloží do proměnných, které následně v procesu vložím do datového objektu s tím, že nastavím u datum vytvoření hodnotu aktuálního data za pomoci funkce „now“, která je přímo dostupná v Blockly konfigurátoru vložení do DO.



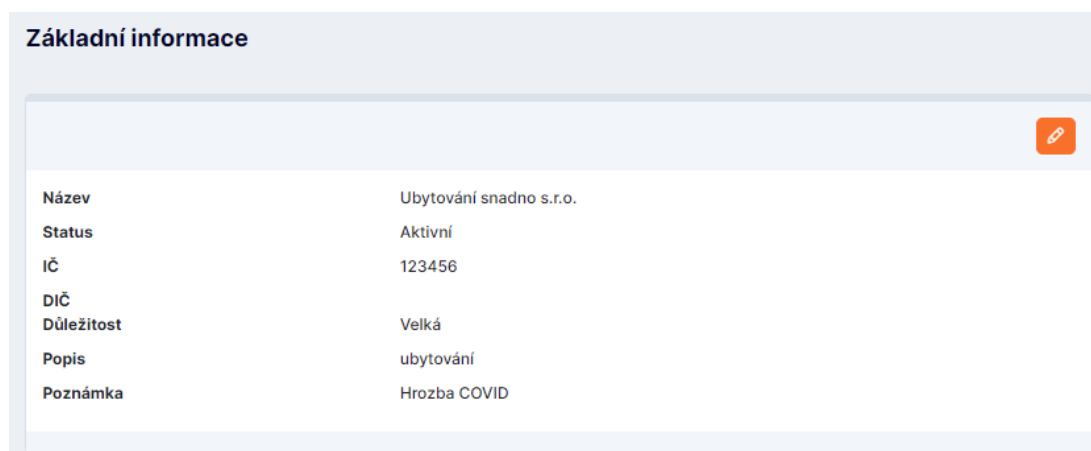
**Obrázek 26: Nastavení vložení do datového objektu v procesu [Vlastní zpracování]**

Proces vytvoření zákazníka (na obrázku 27) tedy obsahuje posloupnost těchto úkolů: Zobrazení obrazovky se vstupním formulářem pro zákazníka, po kliknutí na odeslat se zadané informace uloží do datového objektu (vložení na obrázku 26) a následně se zobrazí detail nově vloženého zákazníka.



**Obrázek 27: Proces vytvoření zákazníka [Vlastní zpracování]**

V obrazovce detailu jsem nastavil grid widget, který slouží pro zobrazení detailních informací vyfiltrovaného datového objektu podle Id, či umožňuje zobrazit i více záznamů z DO ve formě tabulky.



The screenshot shows a widget titled "Základní informace" (Basic Information). It contains a table with the following data:

Název	Ubytování snadno s.r.o.
Status	Aktivní
IČ	123456
DIČ	
Důležitost	Velká
Popis	ubytování
Poznámka	Hrozba COVID

In the top right corner of the table area, there is an orange edit icon (a pencil inside a square).

**Obrázek 28: Základní informace z detailu Zákazníka [Vlastní zpracování]**

V grid widgetu, je možno také přidávat tlačítka. Jak je vidět například na **obrázku 28** v pravé horní části widgetu. Těmto tlačítkům lze nastavit funkcionalita, buďto spuštění procesu, či editace nebo mazání. Při zvolení editace se ještě musí zvolit zase vstupní formulář – tedy Zákazník – input (**obrázek 29**) a je zde také možnost propsání aktuálně nastavených vlastností do formuláře. Na **obrázku 30** se zobrazeno mapování vstupů z formuláře na příslušné hodnoty daného záznamu v tabulce. Po odeslání se hodnoty upraví a není třeba ani využívat jakéhokoli procesu.

Vzhled Zdroj dat Sloupce Akce Výchozí ▾

**+ Akce**

Upravit f/v Formulář Vybrat Proces Upravit Smazat Pokračovat ✖

Vybrat formulář Editovat velikost okna formuláře

Zakaznik - input 5 Normální

**Vstupy**

description	Field ▾
description -	▾
importance	Field ▾
importance -	▾
name	Field ▾
name -	▾
note	Field ▾
note -	▾
status	Field ▾
status -	▾
tax	Field ▾
tax -	▾
vat	Field ▾
vat -	▾

**Obrázek 29: Definování akce Upravit – vstup [Vlastní zpracování]**

Vzhled Zdroj dat Sloupce Akce Výchozí ▾

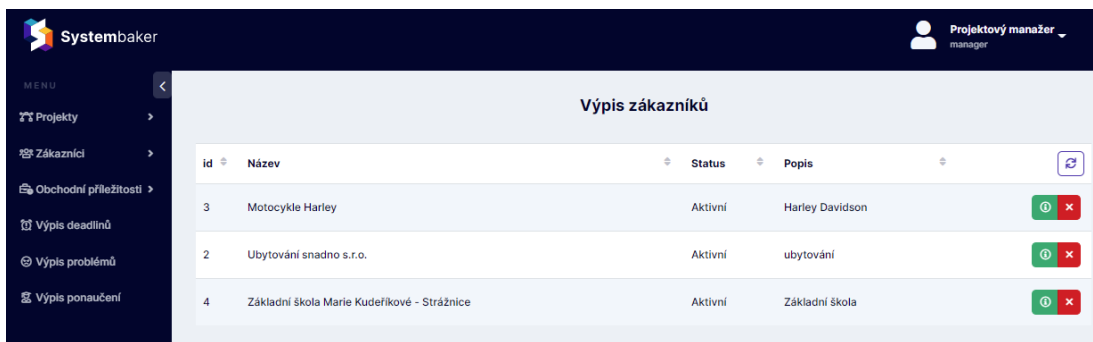
**+ Akce**







Upravit f/v Formulář Vybrat Proces Upravit Smazat Pokračovat ✖

id	None ▾
created_date	None ▾
deleted_date	None ▾
description	Form ▾
description - Output from field Description	▾
importance	Form ▾
importance - Output from field Importance	▾
name	Form ▾
name - Output from field Name	▾
note	Form ▾
note - Output from field Note	▾
status	Form ▾
status - Output from field Status	▾
tax	Form ▾
tax - Output from field Tax	▾
vat	Form ▾

**Obrázek 30: Definování akce Upravit - mapování vstupů [Vlastní zpracování]**

Pro výpis všech zákazníků je vytvořena obrazovka: Customer – list all, ve které jsou pomocí grid widgetu zobrazeny všechny záznamy z tabulky Zákazník, které splňují podmínku, že Datum smazání je NULL. V tabulce u každého záznamu jsou nastaveny a zobrazeny dvě tlačítka. Jedno pro detail daného zákazníka a druhé pro smazání. Tlačítko smazání je spouští proces: Customer – soft delete, který nastaví pro daný záznam (vyhledaný podle Id zákazníka) Datum smazání na aktuální časovou hodnotu.



id	Název	Status	Popis	
3	Motocykle Harley	Aktivní	Harley Davidson	 
2	Ubytování snadno s.r.o.	Aktivní	ubytování	 
4	Základní škola Marie Kudeřkové - Strážnice	Aktivní	Základní škola	 

**Obrázek 31: Výpis všech zákazníků [Vlastní zpracování]**

Zobrazení výpisu všech zákazníků na **obrázku 31** je zobrazení pro roli Projektového manažera, který má neomezená práva. Zbylým zaměstnancům bylo pomocí duplikace obrazovky zpracováno jiné zobrazení, ze kterého bylo odstraněno tlačítko smazání. Mazat může pouze projektový manažer.

Do detailu jsem postupně s následující implementací vložil výpisy relevantních informací k danému zákazníkovi (Projekty, Problémy, Kontakty, Obchodní příležitosti):

MENU

- 🏠 Projekty
- 👤 Zákazníci
- 📄 Obchodní příležitosti
- 🔍 Výpis desedlů
- 📧 Výpis problémů
- 📋 Výpis poučení

Projektový manažer

### Základní informace

Název	Ubytování snadno s.r.o.
Status	Aktivní
IČ	123456
DIČ	Velká
Důležitost	ubytování
Popis	Hrozba COVID
Poznámka	

### Projekty

Název	Status	Spokojenost zákazníka	High-level popis	Hlavní use-case
hotel SNADNO	Aktivní	Největší	Rezervační systém pro hotel v Brně.	Ubytování na pár nocí v Brně.

### Kontakty

Jméno	E-mail	Telefon	Poznámky
Johnny English	English-Johnny@matheslo.com	123456789	
Martin Moudrý	martin@moudry12.cz	720333222	

Přidat kontakt

### Obchodní příležitosti

Název	Stav	Popis
test 2	Diskutovaná interně	hah

### Problémy

Název	Závažnost	Detail	Důsledky	Projekt
Problémek	Najmenší	Malý	Nic	hotel SNADNO

Obrázek 32: Detail zákazníka [Vlastní zpracování]



## 4.7.2 Kontakt

Při vytváření kontaktu jsem opět volil vytvoření formuláře s DO. K vytvoření vazby na zákazníka jsem použil widget Select a nastavil jej na výběr z DO místo z na-  
definování hodnot.

The screenshot displays a configuration interface for a form titled "Kontakt". On the left, a dark blue sidebar contains the form name "Kontakt" and a "Uložit" button. The main area is divided into several sections:

- Výchozí hodnota:** A dropdown menu for "Proměnná" is set to "zakaznik - Input for field zakaznik". Below it, the "Výstupní proměnná" is set to "zakaznik".
- Specifické nastavení pro toto poličko:** This section includes:
  - Styl:** Set to "Classic".
  - Zdroj možnosti:** Set to "Datový objekt".
  - Datový objekt:** Set to "form\_customer".
  - Verze datového objektu:** Set to "Nejnovější".
  - Nadpis:** Set to "name".
  - Identifikátor:** Set to "id".
  - A "Podmínka filtrování" button is located at the bottom right of this section.

At the bottom of the main area, there is a "Poznámky" field, a "Zákazník" dropdown menu, and a "Submit" button.

Obrázek 33: Vytvoření vazby Kontaktu na Zákazníka [Vlastní zpracování]

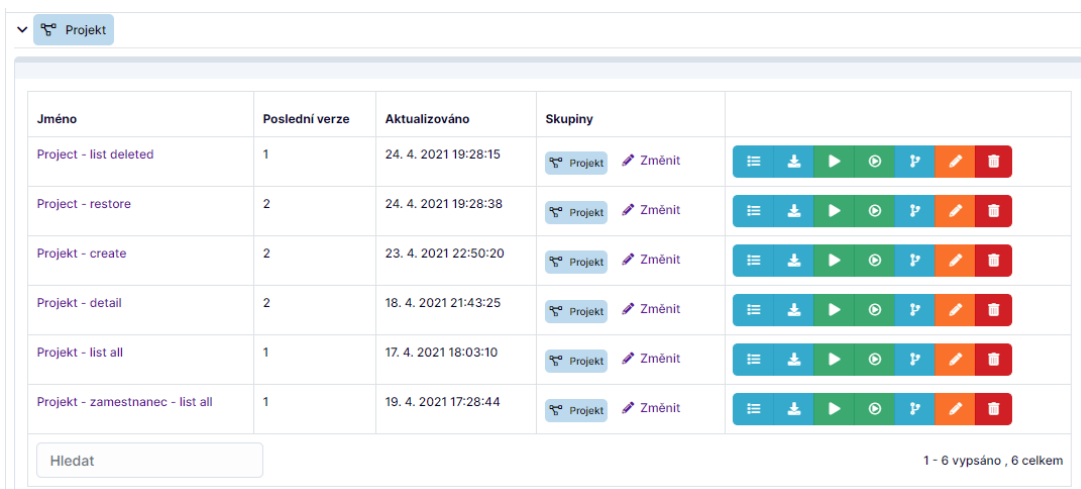
Platí podmínka pro filtrování, že u datového objektu zobrazují pouze Zákazníci, kteří mají Datum smazání NULlové, tedy nesmazané kontakty:

The screenshot displays a query builder interface. On the left, a sidebar titled "Podmínka filtrování" (Filter condition) contains various operators and functions: "Column", "Column -> Sorting Ascendant", "Column", "Text:", "f abs", "AND", "NOT", "+", and "var". The main workspace shows a query structure with a "WHERE" clause containing the condition "deleted\_date IS NULL" and an "ORDER BY" clause. On the right, a panel shows the "Datový objekt:" (Data object) set to "form\_customer", the "Verze datového objektu:" (Data object version) set to "Nejnovější" (Latest), and the "Identifikátor:" (Identifier) set to "id". A "Podmínka filtrování" button is also visible in this panel.

Obrázek 34: Podmínka filtrování pro výběr zákazníka [Vlastní zpracování]

## 4.7.3 Obchodní příležitost a Projekt

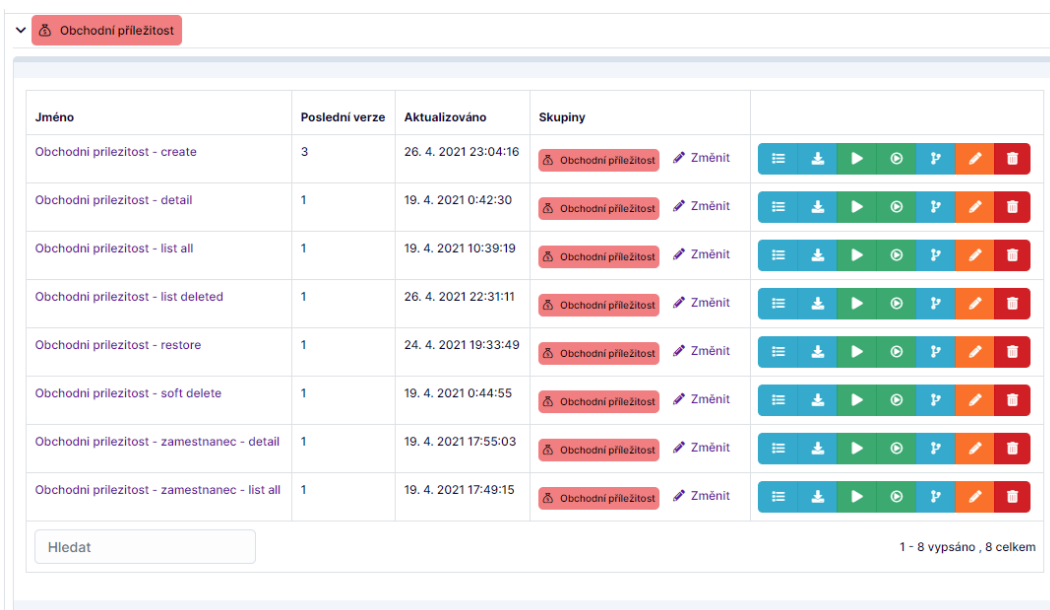
Obdobným způsobem jako u Zákazníka a Kontaktů jsou vytvořeny: formuláře s datovými objekty i bez nich pro získání vstupních dat od uživatelů, obrazovky se zobrazením detailů, vytváření nových záznamů, výpis smazaných (zde stačí obrátit podmínku – Datum smazání jako vyplněná hodnota [IS NOT NULL]), obnovení smazaných (nastavení Datum smazání zpět na nevyplněnou hodnotu [NULL]) a výpisu všech jak pro Projektového manažera, tak i pro zbylé zaměstnance. Obdobně jsou vytvořeny i procesy pro tabulky Obchodní příležitost a Projekt.



Jméno	Poslední verze	Aktualizováno	Skupiny	
Project - list deleted	1	24. 4. 2021 19:28:15	Projekt Změnit	[Icons]
Project - restore	2	24. 4. 2021 19:28:38	Projekt Změnit	[Icons]
Projekt - create	2	23. 4. 2021 22:50:20	Projekt Změnit	[Icons]
Projekt - detail	2	18. 4. 2021 21:43:25	Projekt Změnit	[Icons]
Projekt - list all	1	17. 4. 2021 18:03:10	Projekt Změnit	[Icons]
Projekt - zamestnanec - list all	1	19. 4. 2021 17:28:44	Projekt Změnit	[Icons]

Hledat 1 - 6 vypsanó , 6 celkem

Obrázek 35: Vytvořené procesy Projektu [Vlastní zpracování]



Jméno	Poslední verze	Aktualizováno	Skupiny	
Obchodni prilozitost - create	3	26. 4. 2021 23:04:16	Obchodni prilozitost Změnit	[Icons]
Obchodni prilozitost - detail	1	19. 4. 2021 0:42:30	Obchodni prilozitost Změnit	[Icons]
Obchodni prilozitost - list all	1	19. 4. 2021 10:39:19	Obchodni prilozitost Změnit	[Icons]
Obchodni prilozitost - list deleted	1	26. 4. 2021 22:31:11	Obchodni prilozitost Změnit	[Icons]
Obchodni prilozitost - restore	1	24. 4. 2021 19:33:49	Obchodni prilozitost Změnit	[Icons]
Obchodni prilozitost - soft delete	1	19. 4. 2021 0:44:55	Obchodni prilozitost Změnit	[Icons]
Obchodni prilozitost - zamestnanec - detail	1	19. 4. 2021 17:55:03	Obchodni prilozitost Změnit	[Icons]
Obchodni prilozitost - zamestnanec - list all	1	19. 4. 2021 17:49:15	Obchodni prilozitost Změnit	[Icons]

Hledat 1 - 8 vypsanó , 8 celkem

Obrázek 36: Vytvořené procesy Obchodní příležitosti [Vlastní zpracování]

V detailu projektu jsem rozdělil informace na základní informace a informace k vývoji. Pro obě části vytvořil samostatné vstupní formuláře a v zobrazení detailu projektu pro zaměstnance jsem dal možnost editace pouze informací spojených s vývojem, nikoli základních informací.

**Základní informace**

- Název: hotel SMADNO
- Zákazník: Ubytování snadno s.r.o.
- High-level popis: Rezervační systém pro hotel v Brně.
- Status: Aktivní
- Spokojenost zákazníka: Největší
- Hlavní use-case: Ubytování na pár nocí v Brně.
- Detaily projektu: cloud.google.com/mathesio-kuefialdno
- Tok komunikace: Developer → Project manager → Zákazník (s nazpět)
- Pravidelnost meetingů: Týdně
- Poznámky:

**Informace k vývoji**

- Backend technologie: .NET Core
- Frontend technologie: AngularJS
- Náležitosti lokálního vývoje: .NET 5, Rider, JetBrains
- Redmine: redmine.mathesio.com/room
- Postup ticketů: vytvořit, assignovat, plnit, review, uzavřít
- Postup vývoje: local → staging → produkce

**Kontakty zákazníka**

Jméno	E-mail	Telefon	Poznámky
Johnny English	English-Johnny@mathesio.com	123456789	
Martin Moury	martin@moury12.cz	720333222	

**Deadline**

Název	Zváznost	Soft deadline	Hard deadline	Čiudv hard deadline
Sprint 23	Velká	08. 07. 2021	16. 07. 2021	Je třeba ukázkou nové funkcionality zákazníkovi.
Sprint 24	Malá	16. 07. 2021	23. 07. 2021	


Obrázek 37: Detail projektu z pohledu zaměstnance [Vlastní zpracování]



## 4.7.4 Entity napojené na Projekt

Protože je práce v SystemBakeru velmi jednoduchá, tak se implementace následujících entit z většinové části opakuje. I pro Deadline, Problém, Ponaučení, Cena, Sleva a Pracovník je potřeba vytvořit vstupní formuláře, obrazovky s výpisy dat a procesy, které zaštiťují logiku a funkčnost tlačítek a víceméně celého CRM systému.

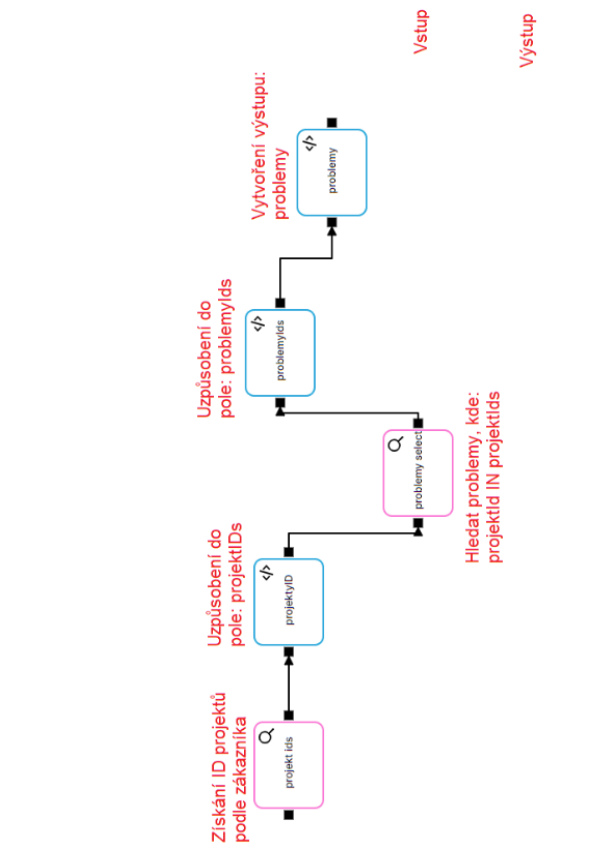
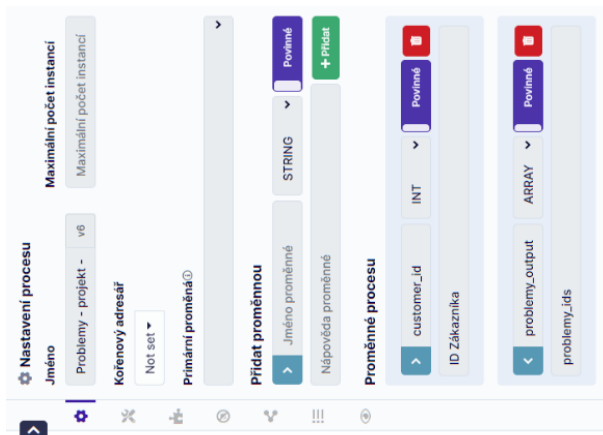
Na **obrázku 39** je zobrazen proces, který získává pole Id Problémů z Projektů určitého zákazníka.

Výstupem je tabulka zobrazující problémy daného zákazníka z různých projektů, kterou lze vidět na **obrázku 38**.



Název	Závažnost	Detail	Důsledky	Projekt	
Chybějící funkcionality	Střední	Zákazník chtěl funkcionality kalendáře nejbližších týdenních schůzek.	Ztráta důvěry ohledně dodržení termínů	Strážnice ZŠ - oznamuj	
Problémek	Nejmenší	Malý	Nic	hotel SNADNO	

*Obrázek 38: Problémy podle ZakaznikId [Vlastní zpracování]*

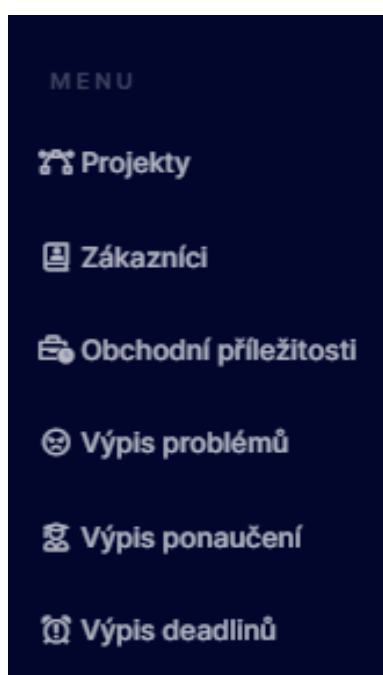


Obrázek 39: Proces na získání problémů podle ZakaznikId [Vlastní zpracování]

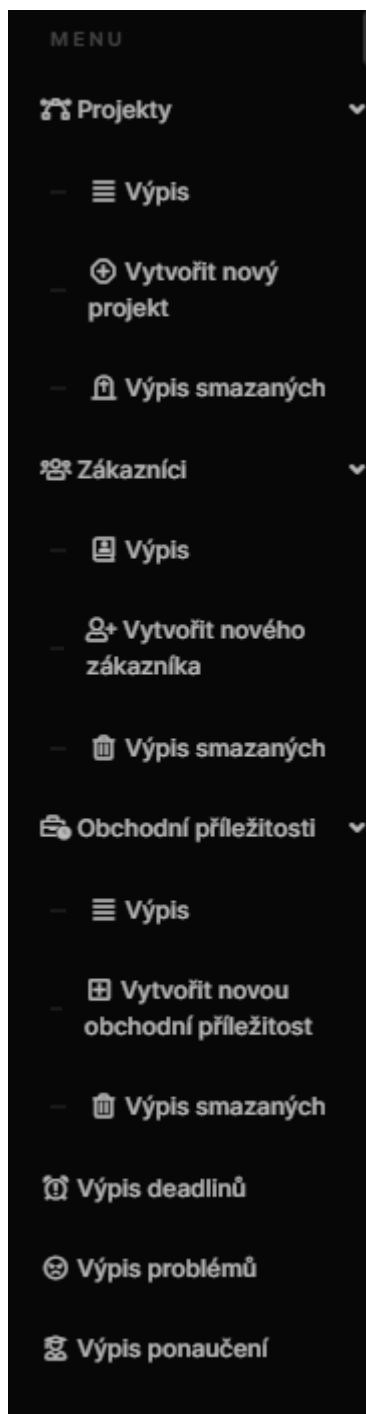
## 4.7.5 Vytvoření menu

Menu pro uživatele se vytváří pomocí Menu Creatoru – velmi jednoduše pomocí „drag & drop“, tedy stačí vybrat roli či uživatele, kterému je zamýšleno změnit menu a pak vybrat jednotlivé procesy (třeba zobrazující obrazovku s výpisem všech projektů) a přetáhnout je do pravé části v menu creatoru. Nové položce v menu je také možné pak nastavit i ikonku či přejmenovat. Menu creator je vyobrazen na **obrázku 42**.

Menu zaměstnance pak na **obrázku 40** a menu projektového manažera na **obrázku 41**.

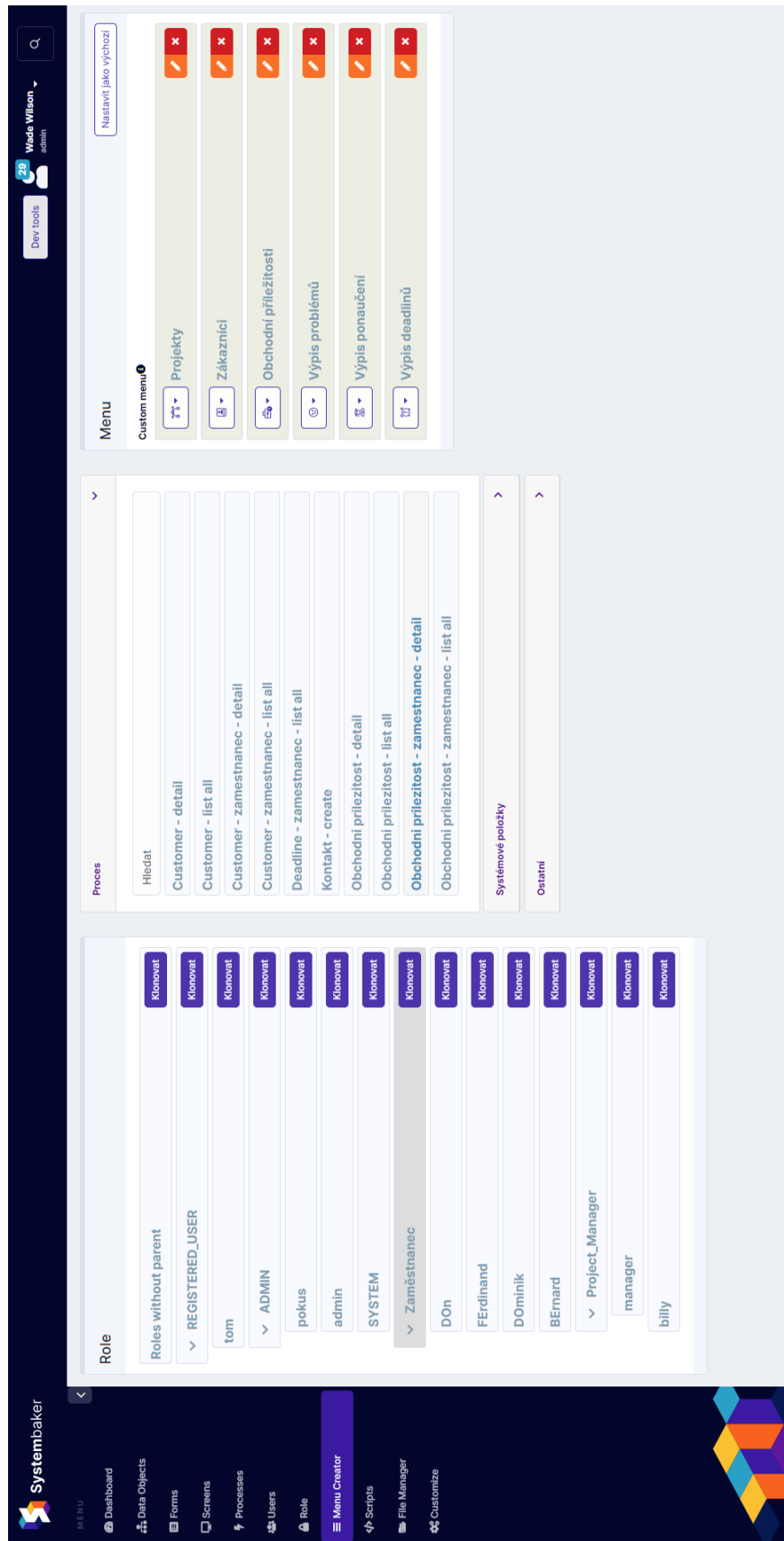


*Obrázek 40: Menu zaměstnance [Vlastní zpracování]*



*Obrázek 41: Menu Projektového manažera [Vlastní zpracování]*





Obrázek 42: Menu Creator - vytváření menu uživatelů [Vlastní zpracování]

## 4.7.6 Osobní zkušenost s vývojem na platformě SystemBaker

Pracovat s touto platformou bylo velmi zajímavou zkušeností, neboť se jedná o hodně moderní technologii. Sice je ještě stále v beta verzi a není vydána pro účely ostrého provozu, ale už teď se s ní dají vytvářet jakékoli aplikace. Díky této platformě, nemusíte být ani vývojář a můžete si vyrobit vlastní aplikaci. Podle mě je to velmi revoluční nástroj, který umí rapidně zrychlit vývoj aplikací. Velkou výhodou je, že kdykoli umožňuje modifikaci stávající implementace. Dojde-li ke změně procesů, velmi snadno se adaptuje stávající implementace na novou verzi.

Ač je to velmi robustní implementační nástroj, jako všechny low-code platformy, úplně bych jej nedoporučoval pro implementaci velmi specifických řešení. Zato je optimální pro informační podporu obchodních a administrativních procesů – tyto procesy se často rychle mění a jsou to primárně datově a procesně orientované systémy.

Výhodou SystemBakeru je také jeho intuitivnost. Byl tvořen s důrazem na uživatelskou přívětivost a velmi lehce se v něm dá orientovat. Cca za 5 hodin seznamování s platformou už můžete mít představu práce s platformou a můžete tvořit nejzákladnější aplikace. Odhaduji, že po cca 20 hodinovém seznámení se a procvičování práce na platformě SystemBaker se stáváte dostatečně zkušeným (znalým) k použití převážné většiny funkcí a možností platformy. S těmito znalostmi tak můžete implantovat nejrůznější aplikace.

Osobně jsem zkoušel i platformu OutSystems, ale ta mi nepřišla tak intuitivní. Po dvou hodinách jsem stále nebyl schopen zjistit, jak se napojují data k obrazovkám, či tlačítkům a procesům/funkcím.

## 4.7.7 Dosažení požadavků

Aplikace CRM systému, která je dostupná na adrese: <https://sandbox.systembaker.com> splňuje všechny požadavky majitele.

- Vidět a chápat spokojenost zákazníků u projektů, co narušuje tuto spokojenost a co děláme / co bychom měli dělat pro to, aby se zlepšila. – **V detailu projektu lze zobrazit aktuální spokojenost zákazníka a případné problémy projektu zhoršující spokojenost.**
- Vidět a chápat v rozumné struktuře základní informace ke konkrétnímu zákazníkovi / projektu a link na další zdroje. – **Všechny základní informace, lze zobrazit v detailech zákazníka/projektu, link na další zdroje je uložen v atributu: Detaily projektu.**
- Možnost vytvářet a sledovat obchodní příležitosti, tedy nové potenciální projekty. – **V systému existuje možnost vytvářet obchodní příležitosti a zobrazovat detaily ke sledování vývoje těchto nových obchodních příležitostí.**
- Rychlé zaučení nového pracovníka do business kontextu projektu, případně rychlá výměna projektového manažera. – **Díky přehledu všech možných informací o projektu či zákazníkovi (jejich detaily), deadlinech, problémech a ponaučeních lze snadno zjistit stav a business kontext.**
- Pomoc při plánování týdenních priorit. – **Pro zaměstnance je možnost zobrazení deadlinů seřazených podle nejbližšího data, také lze zobrazit seznam všech aktuálních problémů a tím zjistit, co je třeba udělat v následujících týdnech pro dosažení zlepšení.**
- Zjistit potenciální problémy okolo deadlinů u konkrétních projektů. – **V detailu projektu je výpis jeho deadlinů, kde se dají zjistit detaily ohledně daného deadlinu, případně je zde také zobrazení aktuálních problémů projektu, které mohou ovlivňovat tyto deadliny.**

#### 4.7.8 Možnosti do budoucna

Realizované řešení CRM systému v této diplomové práci je koncipováno tak, aby splňovalo nejzákladnější požadavky na funkcionalitu a možnost využití. Je zde tedy stále velký prostor pro vylepšení funkcionalit tohoto řešení. Např. možnost přetvořit Obchodní příležitost na Projekt, nebo zavedení emailových notifikací pro uživatele, a

další. Zmiňované notifikace je velmi snadné implementovat, neboť v procesu je komponenta notifikace, kde stačí nastavit email příjemce, šablonu obsahu a podmínku, kde se má e-mail poslat.

Tedy je co zlepšovat, ale to je právě **výhodou platformy**, protože dané modifikace a vylepšení CRM systému jsou velmi snadno implementovatelné a není vůbec problém si cokoli dopravit podle představ.

## 4.8 Ekonomické zhodnocení

Realizovat požadované řešení v platformě mi přibližně zabralo 35 hodin. S těmito hodinami kalkuluji v mém finančním odhadu implementační práce. Odhad nákladů je vyobrazen v **tabulce 4**. V tabulce jsou uvedeny alternativy z **kapitoly 3.5**.

**Tabulka 4: Odhad nákladů na implementaci [Vlastní zpracování]**

	Počáteční cena	Práce (h)	Práce * Mzda	Paušál (měsíc)	Celkem
Raynet	0 Kč	0	0 Kč	10 000 Kč	10 000 Kč / měs.
BlueCRM	160 000 Kč	0	0 Kč	0 Kč	160 000 Kč
OutSystems	0 Kč	60	30 000 Kč	0 Kč	30 000 Kč
Vlastnoručně	0 Kč	150	75 000 Kč	0 Kč	75 000 Kč
SystemBaker	0 Kč	35	17 500 Kč	0 Kč	17 500 Kč
Mzda (Kč/h)	500	Počet uživatelů	20		

Shrnu-li informace z odhadu nákladů implementace, vychází jako nejdražší řešení: BlueCRM – řešení na míru. Následuje zpracování pomocí platformy OutSystems, kdy se předpokládá delší doba na zaučení s platformou, z důvodu nižší intuitivnosti. V tomto případě se počítá s dvojnásobným časem implementace.

Nejlépe vychází platforma firmy SystemBaker, ve které se díky své intuitivnosti pro uživatele dokáže dosáhnout výsledku za neuvěřitelných 35 hodin, což při mzdové sazbě 500 Kč/h odpovídá částce 17 500 Kč.

Protože firma Mathesio se zaměřuje na vývoj softwaru na zakázku, je schopná si toto řešení vytvořit i vlastnoručně, toto řešení by přibližně zabralo 150 hodin práce. Při hodinové sazbě 500 Kč to vychází celkem na 75 000 Kč. Při stejné hodinové sazbě můžeme snadno porovnat vlastnoruční vývoj oproti vývoji v SystemBakeru. **System-baker je v tomto případě tedy zaokrouhleně 150h/35h = 4,3krát rychlejší a levnější než řešení klasickým vývojem.**

CRM řešení, které nabízí Raynet není pro firmu vůbec vhodné a s paušálním poplatkem 10 000 Kč (při 20 uživateli) se stane dražší vůči SystemBaker řešení přibližně za 50 dní, pro OutSystems jsou to 3 měsíce. Jenže při volbě, se často přihlíží ke klasickému projektovému trojimperativu – tedy Náklady, Čas, Kvalita. Přičemž SystemBaker jasně vítězí v kvalitě řešení na míru. Sice dostupnost řešení Raynet může být relativně okamžitá, ale bereme-li v potaz klasický 8 hodinový pracovní den (Monday – MD), máme implementaci řešení realizovanou za cca 4 a půl dne. Z logického hlediska by bylo určitě lepší si pár dnů počkat.

Z hlediska času a nákladů je tedy využití low-code platformy SystemBaker pro implementaci CRM systému **velmi vhodné řešení**, neboť již za jeden týden dokážete mít základní, a **hlavně funkční** aplikaci s náklady 17 500 Kč při hodinové mzdě 500 Kč/h.

# ZÁVĚR

Jsem rád, že jsem mohl tuto práci zpracovat, neboť mi umožnila získat aktuální pohled do firemního prostředí, ve kterém se aktuálně koná revoluce díky novým technologiím jako je třeba právě low-code platforma SystemBaker.

Platforma mi totiž umožnila svižně zareagovat na vymezený problém, kterému firma Mathesio čelila - nutnost vytvoření systému pro evidenci zákazníků, jejich spokojenosti a také evidenci projektů. Protože bychom měli také znát nástroje, se kterými pracujeme je správné si o nich udělat průzkum. K pochopení pozadí platformy mi tak posloužila teoretická část.

Znát svůj nástroj ale také plně nestačí, protože je třeba znát i s čím/kým mám pracovat, k tomu posloužila analytická část. V níž jsem se dozvěděl a ujednotil si informace o zákazníkovi, pro kterého mám navrhnout a implementovat požadovaný systém.

Následovala návrhová část, ve které jsem navrhl možné případy užití a databázi a pak je spolu spojil, také jsem si vytvořil návrh postupu implementace pomocí WBS.

V tomto bodě byly splněny všechny přípravné fáze a zbývalo se vrhnout do implementace. Při implementaci jsem postupoval podle návrhu WBS, tedy postupně implementoval dané entity a jejich funkcionality. Ve finální fázi implementace jsem pak vytvořil menu pro Zaměstnance a Projektové manažery. Kapitulu jsem zakončil osobním zhodnocením práce za pomoci tohoto revolučního nástroje a zmínil případné možnosti zlepšení implementace do budoucna, které lze snadno díky platformě SystemBaker učinit.

V poslední kapitole jsem se věnoval ekonomickému zhodnocení, ve kterém jsem zpracoval odhad nákladů a času potřebného k vytvoření tohoto CRM systému. Z výše uvedeného vyplývá, že SystemBaker exceluje z hlediska projektového trojimperativu jak na straně Času, tak i Nákladů, a dokonce i Kvality. V SystemBakeru lze implementovat systém za přibližně 35 hodin, což odpovídá ani ne jednomu pracovnímu týdnu. Ze strany nákladů se tak jedná o týdenní mzdu jednoho pracovníka, což je velmi pozitivní. A z posledního hlediska – kvality se jedná taky o skvělý výkon, neboť je systém plně přizpůsobitelný a splňuje všechny požadavky zadané na míru.

# SEZNAM POUŽITÉ LITERATURY

- [1] LAUDON, Ken a Jane P. LAUDON. *Management Information Systems: Managing the digital firm*. 12th Edition. New York: Prentice Hall, 2012. ISBN 978-01-357-9093-9.
- [2] Chapter 1: What Is an Information System? – Information Systems for Business and Beyond. *Pressbooks | The open book creation platform*. [online]. Canada, Montreal, Quebec: Book Oven, 2021 [cit. 2021-01-22]. Dostupné z: <https://bus206.pressbooks.com/chapter/chapter-1/>
- [3] HARDCASTLE, E. *Business Information Systems*. Ventus Publishing ApS, 2008. ISBN 978-87-7681-463-2
- [4] Hwswappuser.png. In: *Pressbooks | The open book creation platform*. [online]. Canada, Montreal, Quebec: Book Oven, 2021 [cit. 2021-01-22]. Dostupné z: <https://pressbooks.com/app/uploads/sites/10536/2012/07/hwswappuser.png>
- [5] GÁLA, L., J. POUR a Z. ŠEDIVÁ. *Podniková informatika*. 2. přeprac. a aktualiz. vyd. Praha: Grada Publishing, 2009. ISBN 978-80-247-2615-1
- [6] What is CRM? Definition & Beginner's Guide to CRM | CRM.org. *CRM Software Helping Humans. CRM Demystified & Much More | CRM.org* [online]. USA: CRM.ORG, 2021 [cit. 2021-01-22]. Dostupné z: <https://crm.org/crmland/what-is-a-crm>
- [7] Customer Relationship Management (CRM): What Is It? *The Balance Careers* [online]. New York: About, 2020 [cit. 2021-01-22]. Dostupné z: <https://www.thebalancecareers.com/what-is-crm-2917373>
- [8] Idzikowskiadam, A., Kurylo, P., Cyganiuk, J., & Ryczko, M. (2019). Customer Relationship Management (CRM) - Philosophy and its Significance for the Enterprise. *System Safety: Human, Technical Facility and Environment*, 1(1), 1004-1011.
- [9] Chapter 8: Business Processes – Information Systems for Business and Beyond. *Pressbooks | The open book creation platform*. [online]. Canada, Montreal, Quebec: Book Oven, 2021 [cit. 2021-01-22]. Dostupné z:

<https://bus206.pressbooks.com/chapter/chapter-9-info-systems-strategic-advantage/>

- [10] What is Business Process Modeling Notation | Lucidchart. *Online Diagram Software & Visual Solution | Lucidchart* [online]. United States: Lucid Software, 2021 [cit. 2021-01-22]. Dostupné z: <https://www.lucidchart.com/pages/bpmn>
- [11] Order Fulfillment - BPI - The destination for everything process related. *BPI - The destination for everything process related* - [online]. Canada, Quebec: Business Process Incubator, 2021 [cit. 2021-01-23]. Dostupné z: <https://www.businessprocessincubator.com/content/order-fulfillment/>
- [12] *Unified Modeling Language (UML) description, UML diagram examples, tutorials and reference for all types of UML diagrams - use case diagrams, class, package, component, composite structure diagrams, deployments, activities, interactions, profiles, etc.* [online]. New Jersey: Kirill Fakhroudinov, 2020 [cit. 2021-01-23]. Dostupné z: <https://www.uml-diagrams.org>
- [13] Use-Case-Diagram.jpg. In: *Creately Blog - Diagramming Articles and Tips on How to Draw Diagrams* [online]. Australia, Melbourne: Cinergix Pty., 2021 [cit. 2021-01-23]. Dostupné z: <https://d3n817fwly711g.cloudfront.net/blog/wp-content/uploads/2012/02/Use-Case-Diagram.jpg>
- [14] What is a Flowchart | Lucidchart. *Online Diagram Software & Visual Solution | Lucidchart* [online]. United States: Lucid Software, 2021 [cit. 2021-01-23]. Dostupné z: <https://www.lucidchart.com/pages/what-is-a-flowchart-tutorial>
- [15] 0\*1AI2MenZ6duc6EcY.png. In: *A Comprehensive Guide to Flowchart with 50+ Examples | by Warren Lynch | Medium* [online]. New York: A Medium Corporation, 2019 [cit. 2021-01-23]. Dostupné z: [https://miro.medium.com/max/404/0\\*1AI2MenZ6duc6EcY.png](https://miro.medium.com/max/404/0*1AI2MenZ6duc6EcY.png)
- [16] What is a Data Flow Diagram | Lucidchart. *Online Diagram Software & Visual Solution | Lucidchart* [online]. United States: Lucid Software, 2021 [cit. 2021-01-23]. Dostupné z: <https://www.lucidchart.com/pages/data-flow-diagram>



- [17] *What is a Gantt Chart? Gantt Chart Software, Information, and History* [online]. United States: Gantt.com, 2021 [cit. 2021-01-23]. Dostupné z: <https://www.gantt.com>
- [18] What is a PERT Chart? | Definition, Examples, and Overview. *Product Roadmap Software | ProductPlan* [online]. California: ProductPlan, 2021 [cit. 2021-01-23]. Dostupné z: <https://www.productplan.com/glossary/pert-chart/>
- [19] Functional Block Diagram - an overview | ScienceDirect Topics. *ScienceDirect.com | Science, health and medical journals, full text articles and books*. [online]. Nizozemsko: Elsevier, 2021 [cit. 2021-01-23]. Dostupné z: <https://www.sciencedirect.com/topics/computer-science/functional-block-diagram>
- [20] *IDEF – Integrated DEFinition Methods (IDEF)* [online]. United States, Texas: Knowledge Based Systems, 2021 [cit. 2021-01-23]. Dostupné z: <https://www.idef.com>
- [21] MAHEY, Husan. *Robotic Process Automation with Automation Anywhere: Techniques to fuel business productivity and intelligent automation using RPA*. Birmingham: Packt Publishing, 2020. ISBN 978-1-83921-565-0.
- [22] What is RPA? | Robotic Process Automation Software | Automation Anywhere. *A Global Leader in Intelligent Automation & RPA | Automation Anywhere* [online]. USA, San Jose: Automation Anywhere, 2021 [cit. 2021-01-23]. Dostupné z: <https://www.automationanywhere.com/rpa/robotic-process-automation>
- [23] RPA Tutorial: What is Robotic Process Automation? Application. *Meet Guru99 - Free Training Tutorials & Video for IT Courses* [online]. Gujarat, India: Guru99 Tech Pvt, 2021 [cit. 2021-01-23]. Dostupné z: <https://www.guru99.com/robotic-process-automation-tutorial.html>
- [24] The Workflow Reference Model rmv1-16.html. *AIAI, University of Edinburgh* [online]. Edinburgh: The University of Edinburgh, 1994 [cit. 2021-01-23]. Dostupné z: <http://www.aiai.ed.ac.uk/project/wfmc/ARCHIVE/DOCS/refmodel/rmv1-16.html>

- [25] Služba ustanovení toku pracovních činností. *Evropská komise – oficiální internetové stránky* [online]. Brusel, Belgie: Evropská unie, 2021 [cit. 2021-01-23]. Dostupné z: <https://inspire.ec.europa.eu/metadata-codelist/SpatialDataServiceCategory/workflowEnactmentService>
- [26] What Is Low-Code? [2021 Update]. *OutSystems Blog* [online]. Linda-a-Velha, Portugal: OutSystems- Software em Rede, S.A., 2021 [cit. 2021-01-23]. Dostupné z: <https://www.outsystems.com/blog/posts/what-is-low-code/>
- [27] Benefits of Low-Code Development | Low-Code Basics. *Low-Code Basics - What is Low-Code? | Appian* [online]. Virginie, USA: Appian Corporation, 2021 [cit. 2021-01-23]. Dostupné z: <https://www.appian.com/low-code-basics/benefits/>
- [28] Chapter 10: Information Systems Development – Information Systems for Business and Beyond. *Pressbooks | The open book creation platform.* [online]. Canada, Montreal, Quebec: Book Oven, 2021 [cit. 2021-01-23]. Dostupné z: <https://bus206.pressbooks.com/chapter/chapter-10-information-systems-development/>
- [29] Veřejný rejstřík a Sběrka listin - Ministerstvo spravedlnosti České republiky. *Veřejný rejstřík a Sběrka listin - Ministerstvo spravedlnosti České republiky* [online]. Praha: Ministerstvo spravedlnosti České republiky, 2012 [cit. 2020-12-14]. Dostupné z: <https://or.justice.cz/ias/ui/rejstrik-firma.vysledky?subjektId=227505&typ=PLATNY>
- [30] *Mathesio.cz | Creative IT Solutions* [online]. Brno: Mathesio, 2017 [cit. 2020-12-14]. Dostupné z: <https://mathesio.cz>
- [31] Majitel firmy Mathesio s.r.o. – Ing. et Ing. Vojtěch Mates, Ph.D., *Interní záležitosti firmy Mathesio* [ústní sdělení, online]. Aplikace Discord, 18.3.2021
- [32] McKinsey 7S - ManagementMania.com. *Sociální síť pro business - ManagementMania.com* [online]. Delaware: MANAGEMENTMANIA.COM, 2015 [cit. 2021-03-19]. Dostupné z: <https://managementmania.com/cs/mckinsey-7s>

- [33] McKinseyho model 7S – Wikipedia. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2021 [cit. 2021-03-19]. Dostupné z: [https://cs.wikipedia.org/wiki/McKinseyho\\_model\\_7S](https://cs.wikipedia.org/wiki/McKinseyho_model_7S)
- [34] MVC (Model-View-Controller) Definition, 2018. The Tech Terms Computer Dictionary [online]. Wheaton: Sharpened Productions [cit. 2019-02-22]. Dostupné z: <https://techterms.com/definition/mvc>
- [35] CRM software pro řízení vztahu se zákazníky | RAYNET CRM [online]. Ostrava: RAYNET, 2021 [cit. 2021-4-27]. Dostupné z: <https://raynet.cz>
- [36] Hlavní stránka | Blue CRM cloudapp [online]. 2021: Sysportal, 2021 [cit. 2021-4-27]. Dostupné z: <http://www.bluecrm.cz>
- [37] Detailní záznam o Projektu – RAYNET Cloud CRM. RAYNET Cloud CRM [online]. Ostrava: RAYNET, 2021 [cit. 2021-4-27]. Dostupné z: <https://podpora.raynet.cz/hc/cs/articles/201748976-Detailn%C3%AD-z%C3%A1znam-o-Projektu>
- [38] Build Applications Fast, Right and For the Future | OutSystems [online]. Boston: OutSystems, 2021 [cit. 2021-4-27]. Dostupné z: <https://outsystems.com>
- [39] TICHÝ, Milík. Ovládání rizika: analýza a management. V Praze: C.H. Beck, 2006. Beckova edice ekonomie. ISBN 80-7179-415-5

# SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

API	Application Programming Interface
BP	Business Process
BPM	Business Process Management
BPM	Business Process Modeling
BPMS	Business Process Management Systém
BPR	Business Process Reengineering
CIO	Chief Information Officer
CRM	Customer Relationship Management
CRUD	Create, Read, Update, Delete
CMS	Content Management System
DevOps	Development Operations
DFD	Data Flow Diagram
DO	Datový objekt
MD	Man day – pracovní den
MVC	Model View Controller architektura
PERT	Program evaluation and review technique
QA	Quality Assurance
RPA	Robotic process automatization
SB	SystemBaker
WBS	Work Breakdown Structure
WFMC	Workflow Management Coalition
WFM(S)	Workflow Management Systém

# SEZNAM OBRÁZKŮ

Obrázek 1: Postup interakce uživatele s SW a HW [4] .....	12
Obrázek 2: Model procesu [5] .....	14
Obrázek 3: Příklad BPM notace [11] .....	20
Obrázek 4: Příklad Use Case UML Diagramu [13] .....	21
Obrázek 5: Příklad vývojového diagramu [15] .....	22
Obrázek 6: Úkoly, které lze automatizovat pomocí RPA [21] .....	25
Obrázek 7: Základní charakteristiky WFM systémů a souvislost mezi hlavními funkcemi [24] .....	28
Obrázek 8: Distribuce v rámci služby ustanovení workflow [24] .....	30
Obrázek 9: Obecná struktura Workflow produktu [24] .....	32
Obrázek 10: Detail Datového Objektu [Vlastní zpracování] .....	38
Obrázek 11: Editace Formuláře [Vlastní zpracování] .....	39
Obrázek 12: Editace Obrazovky [Vlastní zpracování] .....	40
Obrázek 13: Editace Procesu [Vlastní zpracování] .....	42
Obrázek 14: Ukázka Blockly při vyhledávání dat z DO [Vlastní zpracování] ..	43
Obrázek 15: Logo Mathesio s.r.o. [30] .....	44
Obrázek 16: Pavučinový graf hodnot rizika před zavedením opatření [Vlastní zpracování] .....	50
Obrázek 17: Pavučinový graf hodnot rizika po zavedení opatření [Vlastní zpracování] .....	52
Obrázek 18 : Vzhled detailu projektu Raynet [37] .....	53
Obrázek 19: Detail Projektu – Blue CRM [36] .....	54
Obrázek 21: Správa uživatelů v SystemBakeru [Vlastní zpracování] .....	59
Obrázek 22: Databázová struktura [Vlastní zdroj] .....	61
Obrázek 23: WBS struktura pro implementaci [Vlastní zpracování] .....	64
Obrázek 20: Časová linka návrhu zpracování [Vlastní zpracování] .....	65
Obrázek 24: Možnosti inputů [Vlastní zpracování] .....	66
Obrázek 25: Formulář pro vytvoření zákazníka [Vlastní zpracování] .....	67
Obrázek 26: Nastavení vložení do datového objektu v procesu [Vlastní zpracování] .....	68

Obrázek 27: Proces vytvoření zákazníka [Vlastní zpracování] .....	68
Obrázek 28: Základní informace z detailu Zákazníka [Vlastní zpracování].....	69
Obrázek 29: Definování akce Upravit – vstup [Vlastní zpracování] .....	70
Obrázek 30: Definování akce Upravit - mapování vstupů [Vlastní zpracování] .....	70
Obrázek 31: Výpis všech zákazníků [Vlastní zpracování] .....	71
Obrázek 32: Detail zákazníka [Vlastní zpracování].....	72
Obrázek 33: Vytvoření vazby Kontaktů na Zákazníka [Vlastní zpracování] ...	73
Obrázek 34: Podmínka filtrování pro výběr zákazníka [Vlastní zpracování] ...	74
Obrázek 35: Vytvořené procesy Projektů [Vlastní zpracování] .....	75
Obrázek 36: Vytvořené procesy Obchodní příležitosti [Vlastní zpracování] ...	75
Obrázek 37: Detail projektu z pohledu zaměstnance [Vlastní zpracování] .....	76
Obrázek 38: Problémy podle ZakaznikId [Vlastní zpracování].....	77
Obrázek 39: Proces na získání problémů podle ZakaznikId [Vlastní zpracování] .....	78
Obrázek 40: Menu zaměstnance [Vlastní zpracování].....	79
Obrázek 41: Menu Projektového manažera [Vlastní zpracování] .....	80
Obrázek 42: Menu Creator - vytváření menu uživatelů [Vlastní zpracování] ..	81

## SEZNAM TABULEK

Tabulka 1: Klasifikační stupnice ohodnocení pravděpodobnosti a dopadu rizika [Vlastní zpracování].....	48
Tabulka 2: Ohodnocení rizik [Vlastní zpracování].....	49
Tabulka 3: Návrh opatření [Vlastní zpracování].....	51
Tabulka 4: Odhad nákladů na implementaci [Vlastní zpracování].....	84