

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky

Informační systém prodejny stavebnin  
Lukáš Karlík

Bakalářská práce  
2019

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Lukáš Karlík**  
Osobní číslo: **I16098**  
Studijní program: **B2646 Informační technologie**  
Studijní obor: **Informační technologie**  
Název tématu: **Informační systém prodejny stavebnin**  
Zadávající katedra: **Katedra informačních technologií**

### Z á s a d y p r o v y p r a c o v á n í :

Cílem této práce je vytvořit funkční aplikaci, která bude obsahovat nástroje sloužící pro zajištění činnosti prodejny stavebnin. Aplikace bude umožňovat tyto funkcionality: přístup uživatelů do systému podle oprávnění, evidenci výrobků, výrobců, zákazníků včetně jejich rezervací a nákupů, fakturace, upozornění zákazníků na slevy, odebírání/přidávání zboží na skladě a generování sestav dle volitelně zadaných kritérií. V teoretické části bude řešerše systémů, které se zabývají touto problematikou. Praktická část bude obsahovat popis použitých technologií, návrh databáze, ER diagram s využitím "Crow's Foot" notace entity-relationship a aplikační řešení. Pro vytvoření aplikace bude využit skriptovací jazyk PHP nebo JAVA a databáze MySQL nebo Oracle.

Rozsah grafických prací:

Rozsah pracovní zprávy: **min. 30 stran**

Forma zpracování bakalářské práce: **tištěná**

Seznam odborné literatury:

**LACKO, Luboslav. Oracle - Správa, programování a použití databázového systému. Brno: Computer Press a.s., 2007. 573 s. ISBN 978-80-251-1490-2**  
**GROFF, James R. a Paul N. WEINBERG. SQL kompletní průvodce. Brno: Computer Press a.s., 2005. 936 s. ISBN 80-251-0369-2**  
**NARAMORE, Elizabeth, Jason GERNER, Scouarnec YANN LE and Timothy BORONCZYK. PHP 6, MySQL, Apache: Vytváříme webové aplikace. Brno: Computer Press a.s., 2009. 816 s. EAN:9788025127674**  
**DRUSKA, P. CSS a XHTML - tvorba dokonalých webových stránek krok za krokem, Grada, 2006. 200 s. ISBN: 80-247-1382-9**

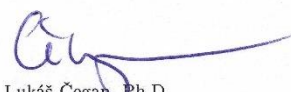
Vedoucí bakalářské práce: **Ing. Miloslav Macháček, Ph.D.**  
Katedra informačních technologií

Datum zadání bakalářské práce: **31. října 2018**

Termín odevzdání bakalářské práce: **12. května 2019**



Ing. Zdeněk Němec, Ph.D.  
děkan



Ing. Lukáš Čegan, Ph.D.  
pověřený vedením katedry

V Pardubicích dne 20. března 2018

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 10. 5. 2019

Lukáš Karlík

## **PODĚKOVÁNÍ**

Tímto bych chtěl poděkovat svému vedoucímu práce Ing. Miloslavu Macháčkovi, Ph.D. za cenné rady, pomoc a vstřícný přístup. Děkuji také svým rodičům a kamarádům za trpělivost a podporu během studia.

## **ANOTACE**

V bakalářské práci je rozebrána problematika informačního systému pro stavebniny. Tento systém má za cíl fungovat jako informační systém pro prodejnu stavebnin a zároveň sloužit jako rezervační systém pro zákazníky. Výsledná aplikace bude vystavěna na databázi s webovým rozhraním. Aplikace bude rozlišovat uživatele s různými oprávněními a na základě těchto oprávnění bude uživatelům zobrazovat rozdílný obsah a tím rozdílnou funkcionalitu.

## **KLÍČOVÁ SLOVA**

Databáze, informační systém, webová aplikace, PHP, MySQL, rezervační systém.

## **TITLE**

Information system of building materials shop.

## **ANNOTATION**

In this bachelor thesis we focus on problematics of informational system for building materials shop. This system will work as informational system for building materials store and will also serve as reservation system for customers. The final application will be built as database with web interface. Application will distinguish users with different authorizations and based on those, it will show different content and therefore different functionality.

## **KEYWORDS**

Database, information system, web application, PHP, MySQL, reservation system.

# OBSAH

Seznam obrázků .....	9
Seznam tabulek .....	10
Seznam zkratk .....	11
Úvod .....	12
<b>1 Teoretická část .....</b>	<b>13</b>
1.1 Informační systém.....	13
1.2 Návrh softwarových systémů.....	13
1.2.1 Byznys modelování.....	15
1.2.2 Specifikace požadavků .....	16
1.2.3 Analýza a návrh .....	17
1.2.4 Implementace .....	18
1.2.5 Testování a nasazení .....	19
1.3 Problematika systémů pro stavebniny .....	20
1.4 Požadavky na systém pro stavebniny .....	20
1.5 Rešerše konkurenčních řešení.....	21
1.6 Informační systém K2.....	22
1.6.1 Popis systému .....	23
1.6.2 Pořízení, nedostatky a zápory .....	23
1.7 Informační systém QI .....	23
1.7.1 Popis systému .....	24
1.7.2 Pořízení, nedostatky a zápory .....	25
1.8 Informační systémy HELIOS .....	25
1.8.1 Popis systému .....	26
1.8.2 Pořízení, nedostatky a zápory .....	26
1.9 Porovnání systémů .....	27
<b>2 Praktická část .....</b>	<b>28</b>
2.1 Návrh systému .....	28
2.1.1 Prostředí .....	28
2.1.2 Funkcionality .....	29
2.2 Analýza .....	30
2.2.1 Rich picture.....	30
2.2.2 Use case diagram .....	30
2.3 Použité technologie.....	31
2.3.1 Použitý software .....	32
2.4 Databáze.....	32
2.4.1 Model .....	32
2.4.2 Tabulky .....	34
2.4.3 Procedury .....	37
2.5 Aplikace .....	38
2.5.1 Popis vybraných stránek .....	39
2.5.2 Popis rolí .....	40
2.6 Instalační příručka.....	41

2.7	Uživatelská příručka .....	42
2.7.1	Klientská část .....	43
2.7.2	Zaměstnanecká část .....	43
<b>Závěr</b>	.....	<b>44</b>
<b>Použitá literatura</b>	.....	<b>45</b>
<b>Přílohy</b>	.....	<b>46</b>



## SEZNAM OBRÁZKŮ

Obrázek 1.1 – Informační systém K2 .....	22
Obrázek 1.2 – Informační systém QI.....	24
Obrázek 1.3 – Informační systém HELIOS .....	26
Obrázek 2.1 – Návrh prostředí.....	28
Obrázek 2.2 – Rich picture .....	30
Obrázek 2.3 – Use case diagram.....	31
Obrázek 2.4 – ER diagram databáze.....	33
Obrázek 2.5 – Stránka se sestavami.....	39
Obrázek 2.6 – Stránka s fakturou.....	40
Obrázek 2.7 – Struktura databáze v phpMyAdmin .....	42
Obrázek 2.8 – Ukázka různých menu .....	42

## SEZNAM TABULEK

Tabulka 1.1 – Porovnání vybraných systémů .....	27
Tabulka 2.1 – Tabulka logZamest .....	34
Tabulka 2.2 – Tabulka loginZam .....	34
Tabulka 2.3 – Tabulka zaměstnanec .....	35
Tabulka 2.4 – Tabulka kontakty .....	35
Tabulka 2.5 – Tabulka login .....	35
Tabulka 2.6 – Tabulka uživatel .....	36
Tabulka 2.7 – Tabulka faktury .....	36
Tabulka 2.8 – Tabulka objednávky .....	36
Tabulka 2.9 – Tabulka zboží .....	37
Tabulka 2.10 – Přihlašovací údaje .....	41

## SEZNAM ZKRATEK

PDF	Portable Document Format
UML	Unified Modeling Language
IS	Informační systém
PHP	PHP: Hypertext Preprocessor
CSS	Cascading Style Sheet
HTML	HyperText Markup Language
MySQL	My Structured Query Language
ER	Entity relationship
GDPR	General Data Protection Regulation
OSVČ	Osoba samostatně výdělečně činná
RFID	Radio Frequency Identification
PNG	Portable Network Graphics
MWB	MySQL Workbench Document
IDE	Integrated Development Environment
URL	Uniform Resource Locator

# ÚVOD

Hlavním cílem této bakalářské práce je vytvořit funkční informační systém pro prodejnu stavebnin. Tento systém slouží jako informační systém pro zaměstnance stavebnin a zároveň jako rezervační systém pro zákazníky. Systém je naprogramován jako webová aplikace. Je tak dostupný na co nejvíce různých platformách.

Systém v případě zaměstnanců slouží ke správě skladu stavebnin, spravování účtů a informací o zákaznících. Pro fungování skladu umožňuje vydávání rezervovaného zboží dle rezervace provedené zákazníkem. Dále dovoluje typické funkcionality pro sklad, jako přidávání nebo odebrání produktů a změnu množství naskladněných produktů. Systém navíc umožňuje generování různých sestav s informacemi o vydaných produktech, které se dají využít k získání statistik pro zlepšení zásobování, nebo výběru nového produktu.

Zákazníkům systém naopak umožňuje rezervaci požadovaného zboží. Každý zaregistrovaný zákazník, typicky firma nebo živnostník, dostane přístup k systému a jeho službám pro zákazníky. Hlavní funkcí je procházení katalogu produktů a následné rezervování produktů v požadovaném množství, které bude téměř ihned dostupné k vyzvednutí na prodejně. Zákazník má také možnost procházení historie svých nákupů a změny osobních nebo firemních údajů.

Teoretická část práce se nejdříve věnuje přiblížení významu informační systém. Následuje část, kde je popsán postup při vývoji softwaru. Dále jsou pak představeny konkurenční systémy. Tyto konkurenční systémy jsou následně popsány, zhodnoceny a porovnány. Hlavními aspekty porovnání systémů jsou přehlednost, porovnání funkcí a výběr dostupných platforem.

Praktická část se věnuje návrhu systému, počáteční analýze a popisu použitých technologií při vývoji systému. Následuje popis návrhu grafického prostředí použitého ve výsledném systému, popis databáze včetně ER diagramu a popis jednotlivých tabulek a propojení. Závěr práce se zabývá popisem výsledného systému a jednoduchou instalační a uživatelskou příručkou.

# 1 TEORETICKÁ ČÁST

Tato část práce je zaměřena na vysvětlení pojmu informační systém, vysvětlením vývoje softwarových systémů, představením a porovnáním podobných již existujících systémů.

## 1.1 Informační systém

Informační systémy jsou vývojová odpověď na zvyšování dostupnosti osobních počítačů a zavádění počítačů do firem v 80. letech minulého století. Před tímto nástupem techniky byla veškerá firemní data uchovávána pouze v papírové podobě. S těmito papíry se muselo složitě manipulovat a archivovat je. To zabíralo spoustu místa ve skladech a kancelářích a také velice zpomalovalo jakoukoli práci. Jakékoliv vyhledání konkrétních dat zabralo spoustu času přiděleným zaměstnancům. Digitalizace dat a záznamů pomohla v usnadnění a zrychlení práce.

Pojem informační systém je v informatice velice rozšířený a nedá se jednoznačně říci přesná definice. Každý autor pokládá za informační systém trochu něco jiného. V zásadě se dá říci, že se jedná o takový systém, který svému uživateli poskytuje určité informace v co nejpřehlednější formě. Dále také slouží ke správě dat obsažených v daném systému. Níže jsou uvedené některé vybrané definice.

„Informační systém (IS) lze chápat jako systém vzájemně propojených informací a procesů, které s těmito informacemi pracují. Přičemž pod pojmem procesy rozumíme funkce, které zpracovávají informace do systému vstupující a transformují je na informace ze systému vystupující. Zjednodušeně můžeme říci, že procesy jsou funkce zabezpečující sběr, přenos, uložení, zpracování a distribuci informací. Pod pojmem informace pak rozumíme data, která slouží zejména pro rozhodování a řízení v rozsáhlejších systémech.“ [1]

„Informační systém je systém, umožňující účelné uspořádání sběru, uchování, zpracování a poskytování informací. V reálném IS rozeznáme dvě složky. Ekosystém se skládá z uživatelů IS, investora IS a toho, kdo systém provozuje (user, funder, server). Ekosystém není pod kontrolou projektantů při návrhu IS. Endosystém se skládá z použitého hardware (media, zařízení), a software (algoritmy, datové struktury) a je plně pod kontrolou designéra IS.“ [2]

„Informační systém je soubor lidí, technických prostředků a metod (programů), zabezpečujících sběr, přenos, zpracování, uchování dat, za účelem prezentace informací pro potřeby uživatelů činných v systémech řízení.“ [3]

## 1.2 Návrh softwarových systémů

Jak již název napovídá návrh softwarových systémů se zabývá návrhem, vývojem a testováním systémů. Této problematice se týká celé vědecké odvětví nazývané softwarového inženýrství.

V tomto odvětví hraje svou roli spousta různých faktorů, které ovlivňují návrh a pozdější vývoj požadovaného systému. Nejdříve ale přesná definice softwarového inženýrství.

„Softwarové inženýrství je inženýrská disciplína zabývající se praktickými problémy vývoje rozsáhlých softwarových systémů.“ [4]

Z této definice můžeme snadno vyčíst, že softwarové inženýrství se zabývá kompletní problematikou tvorby systému. Tedy od vytvoření požadavků, návrhu systému, řízení projektu, organizací dílčích prací až k samotnému testování a nasazení vyvíjeného systému do produkce. [4]

Návrh systémů ovlivňují dobré znalosti místa nasazení systému. Je nutné znát počítačové a softwarové vybavení stejně jako organizační strukturu a finanční možnosti firmy, pro kterou je software určen. K tomuto účelu se nejčastěji používají diagramy například v jazyce UML (Unified Modeling Language). [4]

Pro úspěšné zvládnutí vývoje požadovaného softwaru je také nutné mít zaměstnance schopné využít všech technik softwarového inženýrství pro správnou organizaci a komunikaci v jednotlivých týmech.

Softwarový proces lze brát jako základní stavební kámen tvorby systému. Jedná se o jakési kroky ve tvorbě systému. Tyto procesy jsou předem dané a při jejich dodržování dojde ke zvýšení šance na úspěšnou realizaci nového systému nebo úpravu toho současného. Další výhodou je větší předvídatelnost a čitelnost daných kroků. Tyto kroky se mohou vyvíjet a dále upravovat dle získaných zkušeností z předešlých prací. [4], [5]

Můžeme tedy říci, že nám softwarový proces dává přesný seznam činností, které se mají vykonat a zároveň říká jaká osoba a v jakém časovém horizontu má tuto činnost splnit. K tomuto se pojí pojmy: [5]

- „Role definují funkce, ve kterých vystupují lidé zapojení do softwarového procesu (analytik, programátor, tester, projektový manažér apod.). S každou rolí jsou spojeny artefakty, za které zodpovídá a aktivity, které má vykonávat. Jedna osoba může vystupovat ve více rolích a naopak jednu roli může zastávat více osob.“ [5]
- „Artefakt je označení pro entity, které jsou v rámci softwarového procesu vytvářeny, modifikovány nebo využívány. Může to být například textový dokument, zdrojový kód, komponenta apod. Mezi nejdůležitější artefakty patří modely.“ [5]
- „Aktivita je činnost prováděná pracovníky v jednotlivých rolích při realizování softwarového procesu. Jejich výsledkem jsou nově vytvořené nebo modifikované artefakty. S aktivitami bývají spojeny doporučení, jak při jejich provádění postupovat.“ [5]

- „Fáze softwarového procesu je časový interval, kterému odpovídá stanovený stupeň rozpracování (dokončení) softwarového díla.“ [5]
- „Milník stanovuje, které artefakty mají být v daném čase dokončeny. Slouží jako kontrolní bod umožňující rozhodnout, zda je možné přejít do další fáze procesu.“ [5]

Druhů softwarových procesů je nepřeborné množství, protože firmy zabývající se touto problematikou si velice často vytvoří nebo přizpůsobí softwarový proces tak, aby vyhovoval jejich konkrétnímu prostředí vývoje. Faktory mohou být velikost firmy neboli počet zaměstnanců, především jejich zkušenosti s vývojem softwaru, a nakonec také složitost vyvíjeného softwarového produktu. [5]

Nejčastěji se vychází z vodopádového modelu. Tento model rozděluje tvorbu produktu do čtyř částí: sběr a analýza požadavků, návrh samotného produktu, implementace a testování a nasazení produktu u zákazníka. Tento model má své slabiny, kdy se musí čekat na úplné dokončení předešlé části a doufat v její naprostou správnost. Od tohoto modelu se v současnosti ustupuje a používají se metody, které jsou více agilní a dokážou lépe reagovat na změny v zadání. Jedná se například o model iterativní, kdy se vyvíjí různé verze systému a funkce se postupně přidávají. [4], [5]

### 1.2.1 Byznys modelování

Význam byznys modelování je ve vytvoření společného jazyka, a tedy standardů pro tvorbu diagramů vyjadřujících vazby a činnosti oblasti řešení. K tomuto se využívá jazyk UML. Konkrétněji diagramy aktivit a diagramy tříd. [4]

Diagram aktivit slouží k zobrazení chování systému při konkrétních činnostech systému nebo operacích na základě pokynu uživatele. Zobrazuje podrobně veškeré kroky, které systém vykoná při plnění činností, a jejich pořadí. Tento diagram umí zobrazit rozhodování systému, zda danou činnost vykoná či nikoliv nebo jakým způsobem se vykoná je-li více možností. Umí též zobrazit místa vykonávaná v jeden okamžik, tedy souběžně. Navíc je zde možné zobrazit jaká entita vykonává jakou část aktivity. Tento diagram je vynikající nástroj k ujasnění jednotlivých činností a usnadňuje pozdější implementaci. [4], [5]

Diagram tříd, co se byznys modelování týče, zobrazuje pohled na strukturu projektu a na propojení strukturálních elementů. Jedná se o statický pohled na strukturu. Diagram je poskládaný ze tříd. Těmto třídám se nastavuje tzv. stereotyp. Ten nám slouží k upřesnění významu nebo funkce dané třídy. Tyto stereotypy jsou definované v jazyce UML, ale můžeme si i přidat vlastní. Dále zde řešíme asociace, četnost, role a asociční třídy. [4], [5]

- „Asociace říká, že jeden objekt má odkaz na druhý. Technicky to znamená, že mu může zaslat zprávu. Silnější formou tohoto vztahu jsou vztahy agregace a kompozice, které vyjadřují vztah celek-část, to je, že jeden objekt obsahuje jiný objekt. Rozdíl mezi nimi je ten, že v případě kompozice nemůže objekt, který vystupuje jako součást jiného vystupovat samostatně, to znamená, že jeho existence je spojena s existencí objektu, ve kterém je obsažen.“ [5]
- „Četnost (násobnost) zaznamenává kolik instancí asociované třídy se může napojit na jednu instanci třídy výchozí.“ [5]
- „Role umožňuje zaznamenat to, jak objekt chápe instance asociované třídy“ [5]
- „Asociační třída se zavádí v případě, že s některou asociací jsou spojeny další informace, popřípadě činnosti.“ [5]

Poslední důležitou věcí jsou vztahy mezi třídami tzv. generalizace. Toto má za cíl informovat, že jedna třída je potomek jiné třídy. Jedná se tedy o genericitu a dědičnost známou z programování. [5]

### 1.2.2 Specifikace požadavků

Hlavním cílem této činnosti je získání seznamu vlastností nebo též požadavků, které má vyvíjený systém splňovat. Tyto požadavky je nutné od zákazníka nejen získat, ale také je pravidelně aktualizovat či doplňovat pro případné změny v projektu. Požadavky musí být naprosto přesné a v této fázi vývoje nesmí docházet k chybám nebo k nedorozumění. Špatné sestavení požadavků bude velice pravděpodobně mít za následek selhání a finální produkt nebude odpovídat očekávání zákazníka, co se funkčnosti týče. [5]

Každý požadavek určuje jednu funkci, službu nebo požadavek na výkon či omezení hardwaru pro daný systém. Může tedy jít o popis funkcionality, typu platformy pro určený systém nebo výběr nástrojů pro vývoj. Jsou zde tedy popsány veškeré vlastnosti budoucího systému. Požadavky určují co, nikoliv jak. [5]

Požadavky se rozdělují na funkční a nefunkční. Funkční požadavky říkají, jak se má systém zachovat, popřípadě co dělat nemá, v situacích, do kterých se dostane. Tyto požadavky se určují pomocí diagramů případů užití. Nefunkční požadavky se zaměřují na pravidla pro správný chod systému v místě užívání. Jedná se například o nároky na hardware pro bezpečný běh systému nebo o pravidla dané zákony, jako povinné zálohování dat a GDPR. [5]

Jak bylo zmíněno výše, používají se zde diagramy případů užití (use case diagram). Tyto diagramy jsou složeny z aktérů a případů užití. Aktéři reprezentují entity, které k vyvíjenému systému přistupují a požadují nějaké služby. Entitami mohou být uživatelé nebo i další systémy



napojené na vyvíjený systém. Případy užití pak popisují úkony systému a podobu vzájemné komunikace mezi aktérem a systémem. V diagramu je tak znázorněno, jaký aktér má jaké případy užití. Zjednodušeně řečeno diagram říká, jaké služby můžou aktéři od systému vyžadovat. [4]

### 1.2.3 Analýza a návrh

V této části vývoje softwarového projektu využijeme především diagramy vytvořené v předešlých krocích. Hlavním cílem je zanalyzovat doposud získané informace, především tedy diagramy, ze kterých následně vhodně navrhneme podobu implementace. Návrh musí splňovat všechny stanovené požadavky stejně tak jako omezení. V ideálním případě by navržená implementace měla umožňovat zapracovat změny ve funkčních a nefunkčních požadavcích. [5]

Diagram tříd je zde tvořen třídami, které jsou navzájem spojené pomocí vazeb a tím znázorňují propojení jednotlivých částí systému. Každá třída obsahuje až čtyři části. Tyto části obsahují název třídy, výčet atributů, popis funkcionalit a zřídka používaný popis zodpovědností dané třídy. Ve třídě jsou znázorněny pomocí zkratk veškeré podrobnosti nutné ke správné implementaci. Pro atributy to je nastavení přístupu (public, private, protected), jejich název, datový typ a popřípadě výchozí hodnota. U funkcí to je opět nastavení přístupu, název funkce nebo metody, seznam parametrů s jejich datovými typy a návratová hodnota. Jednotlivé třídy jsou pak propojeny pomocí vazeb asociace (agregace a kompozice). Za pomocí těchto diagramů tříd je pak velice snadné strukturu naprogramovat, protože veškeré vazby a zákonitosti jsou zde popsány. [4], [5]

V analýze se snažíme co nejvíce porozumět zadanému problému. Také zde se vytvářejí diagramy, a to vylepšené diagramy tříd a sekvenční diagramy. Veškerá práce je zde se třídami a objekty. Třídy se berou z předešlých diagramů a přidávají se jim vlastnosti. Na třídy poté navazují objekty, což jsou instance třídy. Objekty zde dostávají specifikováno, jaké operace budou vykonávat a s kým a jak budou komunikovat, popřípadě jak budou reagovat na komunikaci jiného objektu. [4]

Podrobné znázornění vztahů tříd a objektů je zakresleno v sekvenčních diagramech, které popisují průběh komunikace v časovém oknu. V tomto diagramu jsou znázorněny vazby mezi objekty v závislosti na dané operaci, zobrazuje se také časová náročnost, cíle návratových hodnot a upřesnění typu komunikace (synchronní nebo asynchronní). [4]

Výsledkem těchto činností je návrh pro implementaci projektu. V podstatě se dá říci, že výsledné diagramy a zprávy se namapují na požadovanou architekturu výsledného softwaru.

Veškeré části se přetvoří na softwarové komponenty, které se pak používají ve výsledném projektu. V implementaci je pak tvořena architektura systému složená z komponentů, které se dělí na tři vrstvy: [4]

- „Uživatelské rozhraní umožňující komunikaci vyvíjeného systému s uživateli nebo jinými systémy.“ [4]
- „Distribuce sloužící k rozložení rozsáhlých aplikací na více navzájem propojených výpočetních uzlů.“ [4]
- „Persistence umožňující datový obsah reprezentovaný hodnotami atributů udržovaných objekty trvale ukládat tak, aby bylo umožněno jeho přetrvání i mimo dobu běhu programu.“ [4]

#### **1.2.4 Implementace**

Také zde ještě dochází k tvorbě diagramů, konkrétně diagramů komponent a diagramu nasazení. Ty zjednodušují práci programátorům tím, že se předešlé diagramy namapují na strukturu konkrétního programovacího jazyka. Po přechozích krocích přichází ke slovu samotné programování vyvíjeného systému. Diagramy jsou předány programátorům, kteří je začnou převádět do kódu požadovaného jazyka. [4]

Mapování probíhá na základě tabulky určené pro každý programovací jazyk. Tabulka má vždy dva sloupce, jeden pro elementy zakreslené v diagramech UML a druhý sloupec odpovídající jejich synonymu ve vybraném programovacím jazyce. Je zde možná jistá automatizace při převádění na kód pomocí programů, ale ty pouze vytvoří třídy nebo struktury s proměnnými. Metody a propojení má už na starosti programátor. [4]

Při samotném programování je třeba nejen vytvářet zdrojové kódy, ale také je nutné vytváření dalších dokumentů a dat. Jedná se o dokumentaci zdrojového kódu pro možné pozdější úpravy a opravy, uživatelskou dokumentaci sloužící jako návod pro ovládání systému, testovací data a testovací kódy pro průběžné testování funkčnosti systému.

Diagram komponent, který se v této fázi vývoje vytváří, slouží k zobrazení jednotlivých zdrojových souborů (komponent) a vazeb mezi nimi, které jsou popsány pomocí relací závislosti. Zde můžou být nakresleny diagramy pro zdrojové kódy, binární soubory a spustitelné soubory. Diagram nasazení naopak znázorňuje, kde v systému dané komponenty běží. Například tedy zda na počítači uživatele nebo na straně serveru, popřípadě na jakém serveru, pokud jich je v systému nainstalováno více. [4]

### 1.2.5 Testování a nasazení

Testování se provádí postupně během celého vývoje systému a má za cíl zaručit vysokou kvalitu výsledného systému, především úplnou správnost veškerých očekávaných funkcionalit. V rámci testování se zkusí především tři hlavní kritéria. Zda systém skutečně dělá to, pro co byl navrhnout, jestli pracuje bez pádů a je schopný dlouhodobě fungovat bez jakýchkoliv zásahů zvenčí a také zda má systém odpovídající výkonové parametry. Tato fáze vývoje je bez tvorby nových diagramů. [5]

Testy se provádějí během celého vývoje z jednoduchého, přitom důležitého důvodu a tím jsou peníze. Platí zde zásadní pravidlo, že čím dřív se na chybu přijde, tím menší budou náklady na její opravu. Z tohoto důvodu se po celou dobu vývoje vytvářejí nové testy, které jsou schopny otestovat současné vývojové fáze. [5]

Testy mají dva způsoby jejich vyhodnocení verifikaci a validaci. Verifikace kontroluje správnost vyvíjeného systému vůči modelu, který byl vytvořen ve fázi návrhu. Testuje, zda systém odpovídá tomu, co jsme navrhli a co od systému požadujeme. Jinak řečeno říká, jestli ve vývoji postupujeme správně. K verifikaci se používají testy jednotek, integrační testy a testy systému. Validace je kontrola ze strany zákazníka, která kontroluje správné chování a funkce systému dle jeho očekávání. Systém by měl fungovat tak, jak očekává zákazník, a právě validací získáme odpověď, jestli je systém správný. Jedná se například o akceptační testy. [4], [5]

Jednoduchý popis testů:

- Testy jednotek jsou zaměřeny na otestování nejmenších částí systému, tedy tříd, metod nebo funkcí. [5]
- Integrační testy zkoumají správnost fungování při napojení více jednotek dohromady, tedy testují jejich správnou komunikaci a návaznost. [5]
- Testy systému testují systém jako již hotový celek a zaměřují se na jeho funkčnost. Zkoumají bezpečnost, výkonnost a spolehlivost jeho běhu. [5]
- Akceptační testy jsou prováděny se zákazníkem a slouží k ověření správného fungování, dle jeho očekávání a požadavků. Tyto testy mohou probíhat přímo u zákazníka v rámci testování prvních zkušebních verzí produktu. [5]

Testovat systém lze pomocí dvou různých technik. V první technice zvané black box se testuje na základě znalostí zadávacích požadavků a tester jen zkouší, zda se systém chová dle očekávání. Druhá technika nazývaná white box už očekává úplnou znalost implementace a jejím cílem je vytvořit takové scénáře testování, aby byl použit každý řádek napsaného kódu. Tato druhá technika je více podrobná a také lépe otestuje kompletní systém. [5]

Nasazení produktu už je finále celého vývoje. Systém je nainstalován na místo určení a odzkoušen. Poté je vývoj dokončen, ale tím práce na systému nekončí. Ještě je nutné zabezpečit opravení případných chyb nalezených po nasazení systému a poskytovat uživatelskou podporu. V některých případech může docházet k dalšímu vývoji nových vylepšených verzí. [4]

### **1.3 Problematika systémů pro stavebniny**

V současné době je situace ve stavebnictví taková, že očekávanými uživateli systému budou starší věkové kategorie. Je to dáno velmi nízkou popularitou tohoto pracovního odvětví a neustále se snižujícím počtem nových vyučených zedníků, klempířů, pokrývačů a tesařů. Dle údajů Asociace malých a středních podniků a živnostníků ČR se počet vyučených zedníků mezi lety 2005 a 2018 snížil z 705 absolventů na 253. To je pokles o 64 %. U pokrývačů je situace ještě horší zde byl pokles ve stejném časovém období ze 74 na 20 absolventů, tedy pokles o 73 %. [6]

Z výše uvedených důvodů je tedy nutné přizpůsobit vývoj aplikace co největší jednoduchosti, protože je velký předpoklad, že uživatelé budou vyšší věkové skupiny a nebudou příliš zruční v ovládání počítače nebo chytrého telefonu. Proto je kladen důraz na přehlednost, stručnost a intuitivnost aplikace.

Aplikace dále musí podporovat co nejvíce platform, čemuž nejvíce vyhovuje právě webová aplikace. Takováto aplikace může běžet na všech platformách podporujících internetové připojení, což jsou v dnešní době skoro všechna osobní zařízení. Z důvodu velkého rozšíření chytrých telefonů je také nutné mít responzivní design, čehož se u webové aplikace docílí velice snadno. Vývoj systému pro stavebniny v tomto případě dává jasné výhody na stranu webové aplikace, protože není nutné vyvíjet více různých aplikací pro počítače a telefony.

Druhá část aplikace určená pro zaměstnance a její spravování má podobné nároky jako část určená pro běžné uživatele. Musí být přehledná a jednoduchá, aby se v ní zaměstnanci snadno vyznali a práce s ní byla rychlá. Ovšem i tak bude nutné zajistit školení pro zaměstnance pro rychlý a bezproblémový náběh systému na pobočce stavebnin.

### **1.4 Požadavky na systém pro stavebniny**

Aplikace je rozdělena na část pro zákazníky a pro zaměstnance stavebnin. Pro tyto dvě části platí některé společné požadavky a následně požadavky rozdílné z důvodu různých funkcionalit. V obou případech platí, že by grafické prostředí mělo být jednoduché, přehledné

a snadné k užívání. Ideálně tedy, aby byly ovládací prvky a prvky menu na stejném místě.

Požadavky pro zákaznickou část jsou následující:

- Registrace nových zákazníků (OSVČ, firmy, běžní občané)
- Přihlášení zákazníků s platným účtem
- Správa osobních údajů a změna hesla
- Procházení a filtrování nabízeného zboží
- Přidávání požadovaného zboží do košíku
- Správa košíku a rezervování v něm umístěného zboží
- Zobrazení aktivních rezervací daného zákazníka
- Zobrazení historie nakoupeného zboží daného zákazníka
- Možnost zobrazení a vytisknutí faktury

Zákazníkům, kteří ještě nemají účet nebo nejsou přihlášení, je dovoleno procházet zboží, ale pro další funkce je nutné se přihlásit nebo registrovat.

Požadavky pro zaměstnaneckou část mají více různých funkcionalit pro správu dat v systému. Nejdůležitější je zajistit, aby se k těmto funkcím mohli dostat pouze zaměstnanci s potřebným oprávněním. Zaměstnanci jsou rozděleni na různé úrovně oprávnění podle jejich pracovní pozice, a tedy i jiných požadavků od aplikace. Aplikace jim následně musí na základě oprávnění zobrazit různá menu a také zamezit přístupu do stránek s vyšším oprávněním.

Požadavky jsou pro zaměstnaneckou část aplikace tyto:

- Přidání účtu nového zaměstnance
- Správa zaměstnanců a zákazníků
- Správa zboží (přidání, odebrání, změna ceny, změna množství)
- Správa výrobců zboží
- Zobrazení všech aktivních rezervací včetně podrobného obsahu
- Zobrazení veškeré historie nákupů včetně podrobného obsahu
- Generování sestav s možností volby zobrazené obsahu
- Vydávání rezervací
- Fakturace a zobrazení faktury

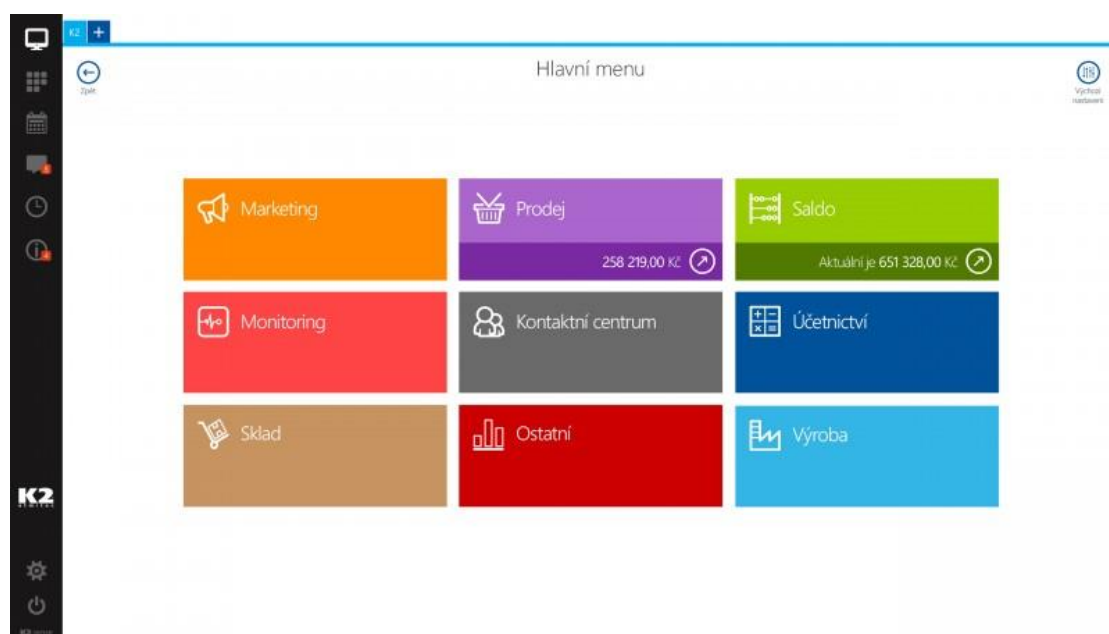
## **1.5 Rešerše konkurenčních řešení**

K zhodnocení navrhovaného systému byli vybráni tři zavedené informační systémy od firem s velkými zkušenostmi s těmito systémy, které na trhu působí už řadu let. Jednotlivé systémy budou nejdříve popsány a následně zhodnoceny jejich silné a slabé stránky. Veškeré porovnání

bude z pohledu uživatele a z dostupných informací ze stránek vývojářských firem. Je to z důvodu nedostupnosti těchto systémů běžnému člověku k jejich vyzkoušení a podrobnému analyzování. Zhodnocení je provedeno dle subjektivního názoru autora této práce a nemusí se shodovat s názorem jiné osoby.

## 1.6 Informační systém K2

První porovnávaný systém je od společnosti K2 atmitec s produktem Informační systém K2. Tato společnost je na trhu od roku 1991 a zabývá se vývojem nástrojů pro řízení firmy. Má zkušenosti jak s vývojem, tak s nasazováním produktů do firem různých velikostí. Nabízený produkt, jak firma uvádí, se na rozdíl od konkurenčních řešení neskládá z modulů, ale je tvořen jako celek. Ovšem i přes toto tvrzení můžeme zjistit, že systém je složený z komponentů různých funkcionalit, které se dají označit za moduly. Firma K2 nabízí vypracování projektu, instalaci systému v místě nasazení, migraci dat na nový systém a zaškolení zaměstnanců. [7]



Obrázek 1.1 – Informační systém K2<sup>1</sup>

Části systému K2 vhodné k porovnání s navrhovaným systémem jsou uživatelské rozhraní, řízení skladu, personalistika, obchod, manažerské vyhodnocování a e-shop. Systém K2 obsahuje ještě další části, ale ty už nejsou vhodné pro porovnávání. Tyto části ovšem nabízí větší provázání systému a také dodávají spoustu užitečných funkcionalit jako například monitoring systému.

<sup>1</sup> SystemOnLine. *SystemOnLine: S přehledem ve světě informačních technologií* [online]. [cit. 2019-03-26]. Dostupné z: <https://www.systemonline.cz/prehled-informacnich-systemu/cloud-computing-saas/informacni-system-k2-6.htm>

### **1.6.1 Popis systému**

Uživatelské rozhraní je moderní a přívětivé k uživateli. Nenachází se zde žádné šedé formuláře, ale barevné a líbivé stránky. Prostředí rozlišuje mezi potřebami různých rolí uživatelů. Řízení skladu umožňuje naprostý přehled o pohybu a počtu zboží umístěného na skladě. Sklad lze řídit online a je zde podpora integrace mobilních terminálů a čteček čárových kódů. Modul obsahuje nástroje k predikci a analýze pohybu zboží a tím usnadňuje rozhodování o množství dokoupeného zboží do skladu. [7]

Část systému pro personalistiku umožňuje uchovávat informace o zaměstnancích v databázi. Systém zde také eviduje kvalifikaci jednotlivých zaměstnanců a automaticky upozorní na potřebu přeškolení nebo obnovení kvalifikace. Bonusem pak je výpočet mezd a schopnost systému komunikovat s úřady v podobě všech potřebných tiskopisů. V manažerském vyhodnocování jsou nástroje pro generování sestav, grafů a statistik. [7]

Poslední částí je e-shop, kde jsou údaje o ceně nebo o množství přímo k dispozici a není nutná implementace programu pro propojení s informačním systémem. Systém K2 umožňuje blokaci ve skladu, tvorbu zakázek a různé platební brány. [7]

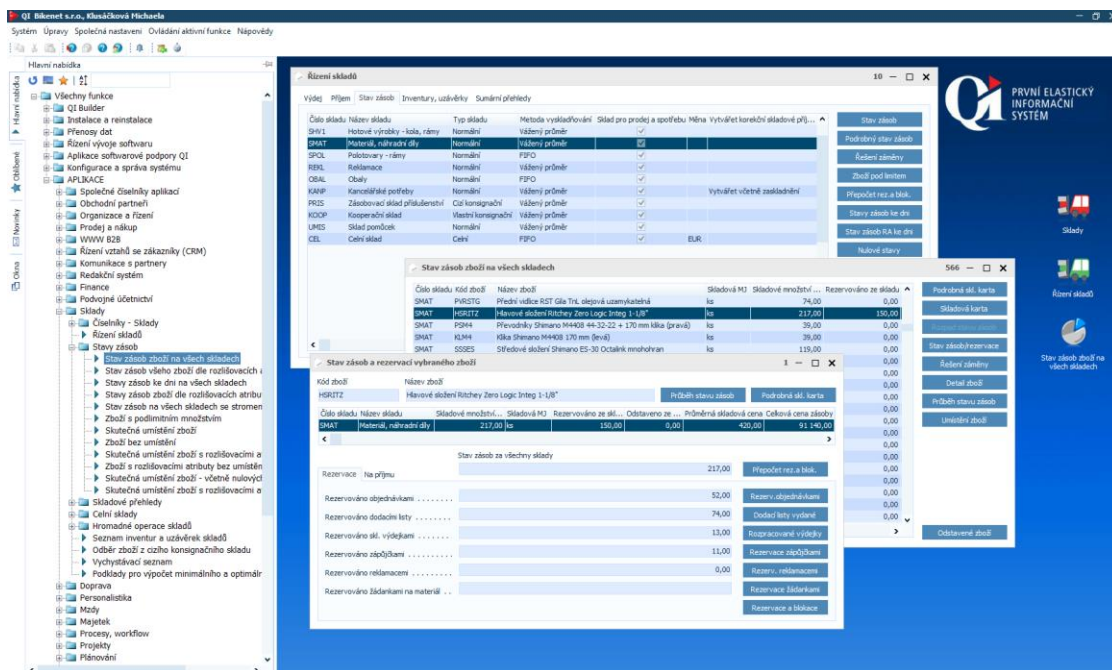
### **1.6.2 Pořízení, nedostatky a zápory**

Informační systém K2 je sofistikovaný, ale pro některé zákazníky zbytečně předimenzovaný, a tedy i zbytečně drahý. Největší nevýhoda je v jeho způsobu pořízení. Pořízení funguje tím způsobem, že zákazník nejdříve zaplatí za studii, instalaci systému do zákaznickovi firmy, konverzi dat a přeškolení. Za tyto služby se zaplatí jednorázově a následuje platba paušálem za provoz. Bohužel toto ještě není vše. Zákazník musí platit licenci na užívání systému K2 pro každého zaměstnance, který ho bude používat. Dále je také nutná instalace na počítače ve firmě zákazníka, což při jejich výměně může způsobovat problémy. [7]

## **1.7 Informační systém QI**

Druhý informační systém je od společnosti QI GROUP, která se na trhu s informačními systémy objevila v roce 2000. Zabývá se tvorbou systémů pro střední a velké firmy a má více než desítky referencí. Nabízený systém s názvem Informační systém QI může být tvořen až z 29 modulů. Zákazník tak má možnost vybrat si jen ty moduly, které skutečně využije. [8]

Implementace probíhá podobně jako u konkurentů, tedy od analýzy po nasazení a testování. K porovnání jsou vhodné moduly QI portál, systémové funkce, prodej a nákup, sklady, QI shop a personalistika. Firma QI k tomuto systémům dále nabízí například modul servis a údržba nebo také QI Helpdesk. Tyto dva poslední moduly jsou pro velké zákazníky doslova nutností. [8]



Obrázek 1.2 – Informační systém QI<sup>2</sup>

### 1.7.1 Popis systému

V tomto systému je možné mít pouze ty moduly, které zákazník skutečně využije. To má za následek pozitivní vliv na výslednou cenu systému. K systému samotnému, celý systém QI je uzavřený a postavený na QI platformě. Základem je QI portál sloužící jako grafické rozhraní, které není úplně zastaralé. Systémové funkce jsou další základní částí celého systému. Obsahují funkce pro rozdělení rolí dle pozice zaměstnance, jádro systému a služby pro export a import dat. Jedné se tedy o jakýsi modul pro správu celého systému. [8]

Modul prodej a nákup slouží k ukládání a tvorbě dat týkajících se obchodních transakcí. Umožňuje evidenci nákupů a zobrazování statistik prodejů stejně jako evidenci zákazníků. Zobrazuje podrobnosti ke zboží třeba šarži, sériové číslo a expiraci. Tento modul také podává informace o objednávkách, nákupech nebo fakturách. Modul sklady zobrazuje aktuální informace o množství zboží. To pomáhá pro zlepšení zásobování a minimalizaci nákupu zboží, které se neprodává. Obsahuje automatické upozornění na rezervované zboží, různé metody vyskladňování nebo evidenci zboží pomocí kódů. [8]

Díky modulu QI shop je možné, aby zákazníci nakupovali přímo z webové stránky. Modul má přímý přístup k datům ze systému QI. Je zde jednoduché a intuitivní ovládání a také funkce chytrého košíku, který pomáhá pravidelně nakupujícím uživatelům. V modulu personalistika je

<sup>2</sup> QI Group. *QI: CENTRÁLNÍ MOZEK FIRMY* [online]. [cit. 2019-03-26]. Dostupné z: <https://www.qi.cz/moduly/sklady/>



pak možné evidovat informace o zaměstnancích a jejich docházku jak v jednosměnném, tak i vícesměnném provozu. [8]

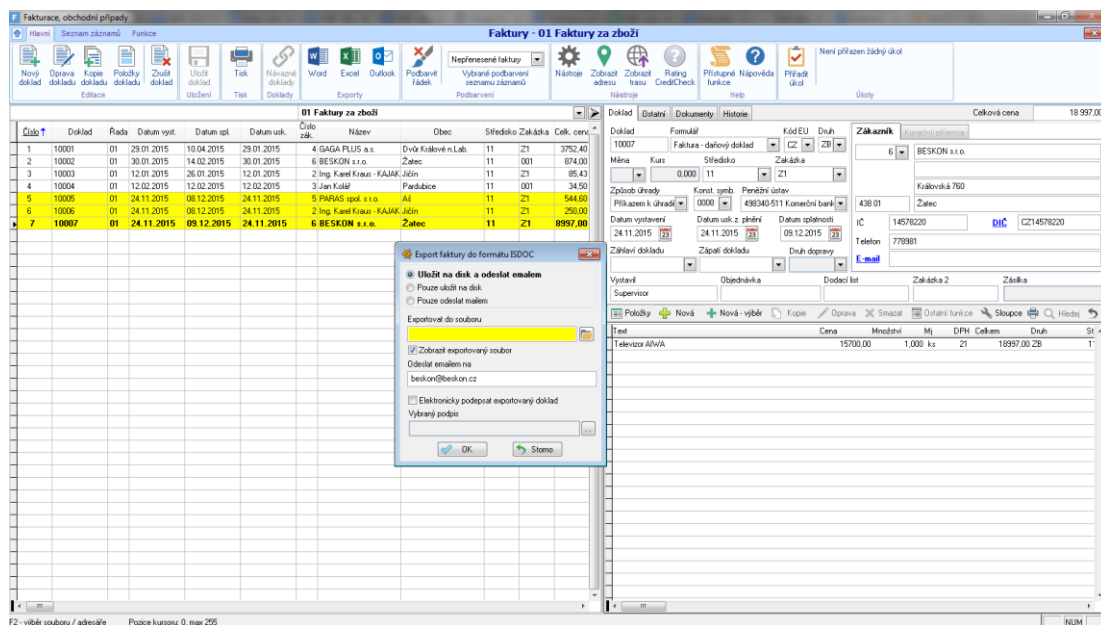
### **1.7.2 Pořízení, nedostatky a zápory**

Je nutné počítat s tím, že systém QI je potřeba instalovat na počítače jako speciální aplikaci. I zde se platí za analýzu a následnou implementaci, které doprovázející následné licenční poplatky, zde reprezentovány jako pronájem systému. Licence se u systému QI vztahuje na použité moduly a není tedy možné si tento systém koupit jednorázově. Grafické prostředí není příliš přehledné a nepatří ani k nejmodernějším. V případě pořízení modulu QI shop, je pak tento fakt znát i u výsledné webové stránky. Je možné pořídit pouze některé moduly, ale pro správné fungování systému je nutné jich pořídit velké množství, což se negativně zobrazí na výsledné ceně. [8]

## **1.8 Informační systémy HELIOS**

Třetím vybraným informačním systémem vhodným pro stavebniny je systém od firmy Asseco Solutions s názvem HELIOS. Tato firma se vývojem informačních systémů zabývá od roku 1990 a je největším vývojářem podnikového softwaru pro zdejší trh. Firma nabízí jak informační systém na míru, tak systém složený z modulů. [9]

Firma nabízí kompletní služby vývoje softwaru, nebo již zavedený a úspěšný systém HELIOS, který je vybrán k porovnání. Systém je dodáván ve verzích HELIOS Green, HELIOS Red, HELIOS Orange a HELIOS Easy. Tyto systémy se v zásadě se liší pouze vybranými základními moduly a zaměřením na různě velké klienty. Systém HELIOS je možné zakoupit již jako hotové řešení pro prodej stavebních materiálů, kde jsou moduly předem vybrané. Tento systém ale nemá webové rozhraní pro zákazníky, je tedy potřeba k systému přidat ve formě modulu e-Commerce. Pro porovnání jsou vybrány moduly sklad a logistika, nákup, prodej, lidské zdroje a marketing. [9]



Obrázek 1.3 – Informační systém HELIOS<sup>3</sup>

### 1.8.1 Popis systému

Zákazník má možnost vybrat si požadované moduly, které je možné přidat k vybrané verzi systému. Celý systém běží na platformě a jádru systému HELIOS. Modul pro sklad a logistiku umožňuje evidovat zboží a materiál na skladě. Umožňuje veškeré základní ovládání skladu. Systém využívá automatickou identifikaci zboží pomocí čárových kódů, RFID nebo podle šarží. Modul automatizuje procesy od objednání po vydání zboží. Modul pro prodej a nákup slouží ke komunikaci s odběrateli a dodavateli. Jsou zde evidováni všichni zákazníci a informace o nich. Další zásadní funkcí je vytváření nového zboží a tvorba ceny. [9]

V modulu lidské zdroje je možné evidovat zaměstnance a veškeré mzdové informace. U zaměstnanců je možné evidovat pracovní místo nebo školení, kterými prošli nebo teprve projdou. V modulu marketing jsou k dostání statistiky, grafy nebo podklady za účelem vhodného marketingu na vybrané zákazníky. Modul e-Commerce umožňuje propojení dat ze systému s webovou stránkou e-shopu. Samotný web je nutné realizovat externě. [9]

### 1.8.2 Pořízení, nedostatky a zápory

Na rozdíl od předešlých systému je tento systém možné zakoupit. Nejsou zde žádné licence na jednotlivé moduly a je nutné pouze platit za případnou podporu, školení nebo další moduly. Systém se zdá být výkonný, ale není příliš přehledný a nemá ani příliš moderní vzhled.

<sup>3</sup> Asseco Solutions. *Helios Asseco: HELIOS Red Velkoobchod* [online]. [cit. 2019-03-28]. Dostupné z: <https://www.helios.eu/produkty/helios-red/balicky-helios-red/helios-red-velkoobchod/>

Převažují zde šedé tabulky a okna. V případě využití modulu pro propojení s webovou stránkou je nutné vyhledat třetí stranu, která tyto stránky vytvoří a napojí na systém HELIOS, což bude mít negativní vliv na cenu systému jako celku. [9]

## 1.9 Porovnání systémů

Vybrané systémy byly popsány a zhodnoceny dle dostupných informací autorských firem. Systémy jsou velice zdařilé a jsou již několik let nasazeny u zákazníků. Díky modulům je lze přizpůsobit pro jakékoliv specifické požadavky zákazníků. Funkcionality jednotlivých modulů jsou dotaženy do každé maličkosti. Jednoznačně nejlepší systém se nedá s určitostí vybrat, vzhledem k různým požadavkům zákazníků, ale všechny vybrané systémy jsou si velice podobné. Největší rozdíl byl shledán ve formě pořízení každého systému. Podrobné srovnání viz tabulka 1 na další stránce.

Tabulka 1.1 – Porovnání vybraných systémů

	IS K2	IS QI	IS HELIOS
Moderní vzhled	Ano	Ano	Ne
Přehledný	Ano	Spíše ne	Spíše ne
Nutná instalace	Ano	Ano	Ano
Předplatné (za uživatele)	Ano	Ne	Ne
Předplatné (za moduly)	Ne	Ano	Ne
Jednorázový nákup IS	Ne	Ne	Ano
Placená podpora	Ano	Ano	Ano
Modul pro sklad	Ano	Ano	Ano
Čtečka čárových kódů	Ano	Ne	Ano
Modul pro personalistiku	Ano	Ano	Ano
Informace o školení	Ne	Ano	Ano
Modul pro prodej	Ano	Ano	Ano
Statistiky a grafy	Ano	Ano	Ano
Modul pro web	Ano	Ano	Ano
Web je součástí systému	Ano	Ano	Ne
Uživatelské role	Ano	Ano	Ano
Vhodný pro velké firmy	Ano	Ano	Ano
Vhodný pro malé firmy	Ne	Ne	Ano
Fórum k systému zdarma	Ne	Ne	Ano
Dostupné na mobilu	Ne	Ne	Ne

[7], [8], [9]

## 2 PRAKTICKÁ ČÁST

V praktické části bude popsán postup vývoje navrhovaného systému a také systém samotný. Začátek se věnuje procesu návrhu navrhovaného systému, kde je popsáno určení systému a také výběr prostředí a nutných funkcionalit. Následuje analýza projektu a popis navržených technologií. Nejdůležitější částí je popis modelu a tabulek navrhované databáze. Konec praktické části se zabývá popisem navrhovaného systému.

### 2.1 Návrh systému

Navrhovaný systém je inspirován porovnáváním systému a jsou z nich vzaty nejdůležitější funkcionality. Dále je zde snaha o vylepšení největších nevýhod porovnávaných systémů, tedy omezenost pro použité platformy, kdy je systém určen především pro počítače se systémem Windows. Z tohoto důvodu je kladen důraz na multiplatformní řešení systému v co nejjednodušší podobě. Vzhledem k těmto hlediskům bylo pro navrhovaný systém vybráno webové prostředí.

#### 2.1.1 Prostředí

Jak bylo zmíněno výše, pro systém bylo zvoleno webové prostředí z důvodu co možná největší kompatibility různých platform od počítačů po chytré telefony. Tento zvolený přístup také zjednoduší nasazení ve firmách a ovládání systému.



Obrázek 2.1 – Návrh prostředí

Stránky systému mají jednotný vzhled s ovládacími prvky umístěnými vždy na stejném místě. Tím je zaručeno snadné a intuitivní ovládání, které bude snadné si zapamatovat.

K navigování po stránce slouží menu umístěné v horní části stránky. Toto menu se mění dle role uživatele systému. Vedle menu je pak umístěn odkaz pro přihlášení do systému, pro klienty a zaměstnance. Pokud je uživatel již přihlášený, je na tomto místě umístěno tlačítko pro odhlášení.

### **2.1.2 Funkcionality**

Z představení a porovnání systémů vznikly funkcionality, které musí navrhovaný systém splňovat. Jde o funkce pro správu skladu, zaměstnanců, uživatelů (klientů), rezervací, historie nákupů a výpisu sestav.

Systém musí umožňovat běžné operace ve skladu, jako přidávání nového zboží, odebrání zboží, změnu jeho množství a také případnou úpravu informací o zboží. Dále musí systém umožňovat přidávat nové zaměstnance do systému s vygenerovaným přihlašovacím jménem a heslem, které si následně zaměstnanec změní. Dále musí systém umožňovat správu jak zaměstnanců, tak uživatelů. Jde o změnu jejich údajů, smazání starých účtů nebo změny zapomenutého hesla.

Pro běžné uživatele pak systém musí nabízet možnost jejich registrace, buď jako firma, OSVČ nebo soukromá osoba. Tito registrovaní uživatelé si pak budou moci rezervovat zboží, projít historii nákupů nebo spravovat vlastní účet. Užitečnou funkcí pak bude upozornění uživatelů na zlevnění zboží, které v poslední době nakupovali.

Systém musí dále provádět automatické činnosti od rezervování zboží uživatelem, přes vytvoření objednávky až po zarezervování požadovaného množství vybraného zboží. Tyto rezervace se pak budou zobrazovat příslušnému zaměstnanci, který si bude moci zobrazit podrobnosti rezervace, označit rezervaci za vydanou nebo rezervaci zrušit, pokud to bude situace vyžadovat.

Po vydání zboží systém automaticky přesune objednávku na stránku s nákupy, kde bude moci zaměstnanec s příslušným oprávněním každý nákup vyfakturovat a fakturu si buď vytisknout, nebo uložit. Uživatelům tato část systému bude zobrazovat pouze vlastní nákupy a umožní jim zobrazit si fakturu, pokud již byla vystavena.

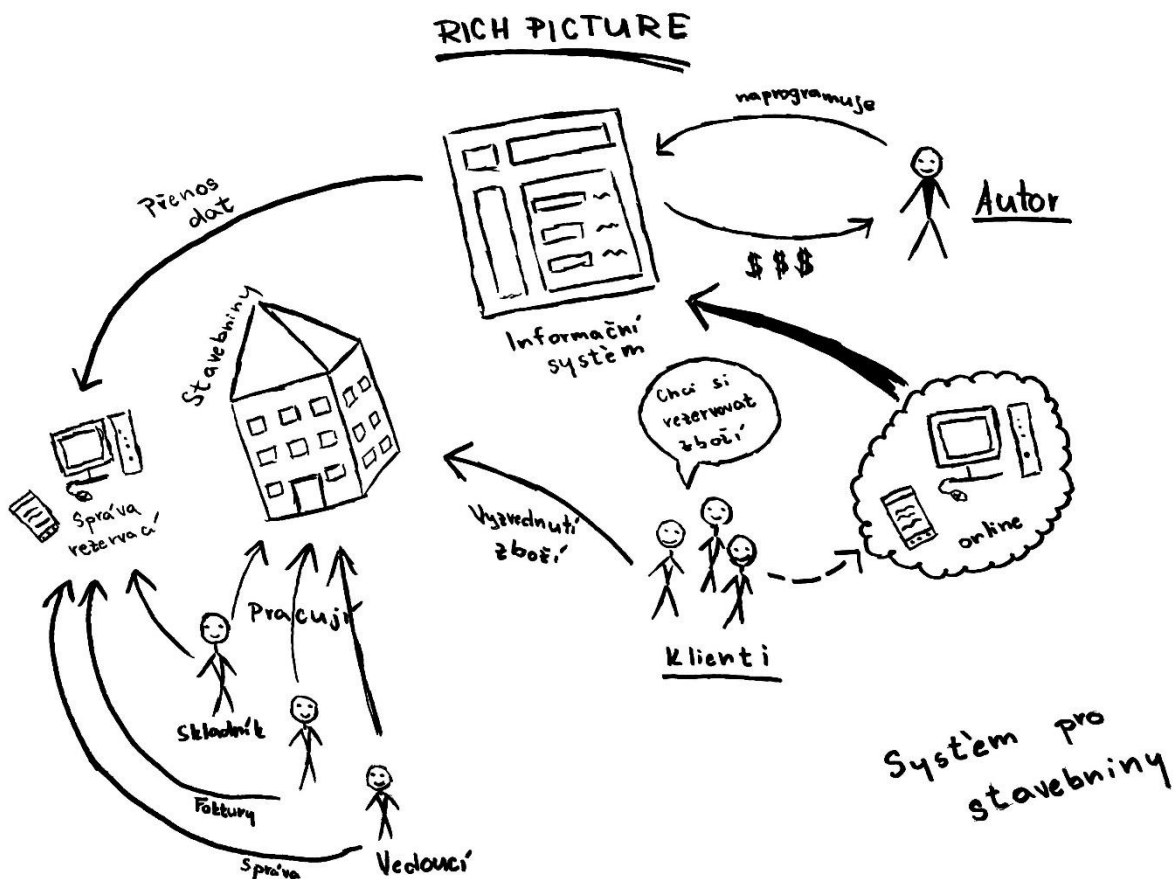
V poslední řadě navrhovaný systém umožní zaměstnancům zobrazení různých sestav prodejů s možností volby různých kritérií.

## 2.2 Analýza

Pro zdárné dokončení systému je nutná důkladná analýza. Pro potřeby této práce byl vytvořen rich picture k zobrazení situace a také use case diagram jednotlivých rolí v systému. Tyto dvě analýzy jsou přiloženy a popsány níže.

### 2.2.1 Rich picture

Rich picture je způsob zakreslení situace, která zobrazuje hlavní prvky vyvíjeného systému a vztahy mezi nimi. Zde má za cíl pomoci autorovi práce s vyjasněním vztahů v problematice informačního systému pro stavebniny. Hlavními prvky v tomto případě jsou informační systém, stavebniny, zaměstnanci, klienti a autor systému. Z obrázku 2.2 je pak jasné, jaké vztahy tyto jednotlivé prvky mají a jaké funkcionality se očekávají.



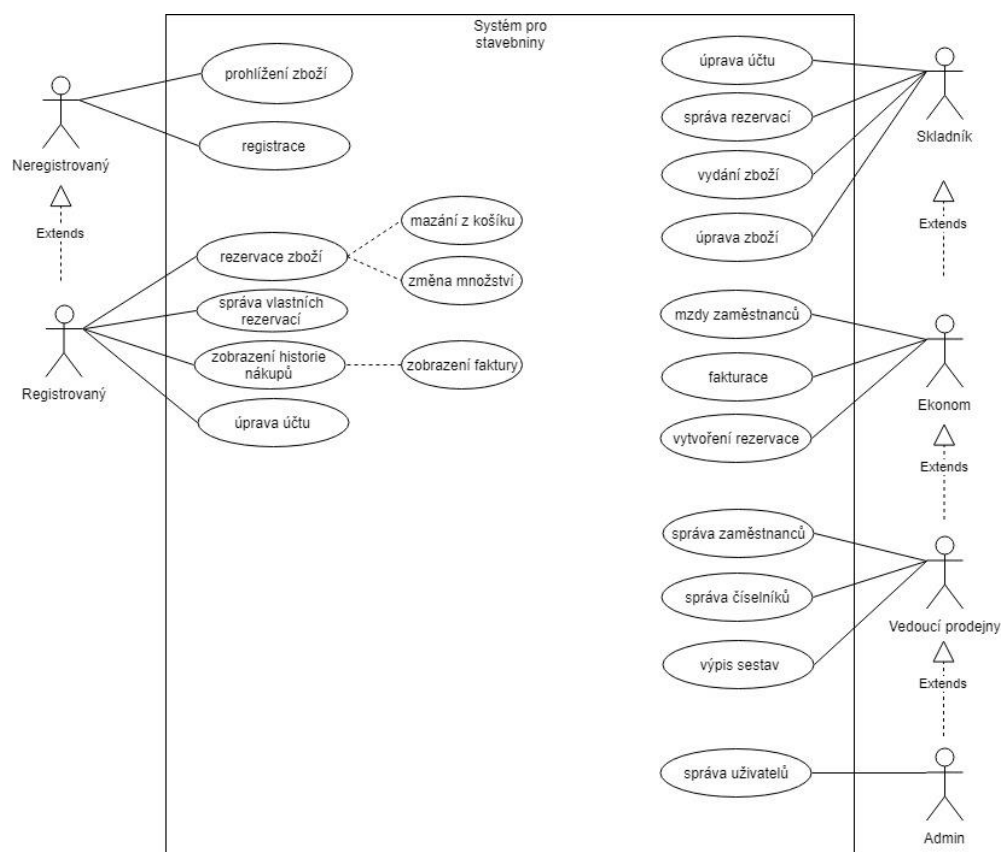
Obrázek 2.2 – Rich picture

### 2.2.2 Use case diagram

Další důležitou částí analýzy bylo vytvoření use case diagramu zobrazujícího jednotlivé entity používající systém a jejich funkcionality nebo nároky na systém. Mezi entity patří dvě role pro klienty (registrovaný a neregistrovaný) a dále role pro zaměstnance stavebnin, jako

administrátor, vedoucí prodejny, ekonom a skladník. Z diagramu je možné vyčíst hlavní funkce, které jednotlivé role budou v systému mít k dispozici.

Dále je zde vidět, jak jednotlivé entity závisí (dědí) na jiných entitách. Například u zaměstnanců je toto důležité z důvodu oddělení různých funkcí, kdy je nepřipustné, aby například skladník mohl měnit údaje klientských účtů. Podrobný popis funkcionalit je zobrazen v obrázku 2.3 pro možnost bližšího prozkoumání.



Obrázek 2.3 – Use case diagram

## 2.3 Použité technologie

Po rozhodnutí pro vytvoření systému jako webové aplikace je nutné vybrat vhodné technologie pro implementování. Ty jsou v rámci webových prostředí značně omezeny a oproti aplikacím určeným pro počítače zde není tak velký výběr různých programovacích jazyků. Pro navrhovaný systém byly vybrány nejrozšířenější technologie pro webové aplikace: HTML, PHP, CSS a jako databáze byla zvolena databáze MySQL.

HTML je značkovací jazyk a používá se k tvorbě webových stránek. Tvoří jakousi kostru požadované stránky. Skládá se z párových nebo nepárových tagů, které znázorňují jednotlivé elementy stránky. Tento jazyk vznikl v 90. letech minulého století a je vyvíjen do současnosti. Nejnovější verze je HTML 5.0. [10]

Přímo k HTML se vztahují kaskádové styly (CSS). Tento jazyk, v nejnovější verzi CSS3, slouží k nastýlování jednotlivých elementů z HTML nebo i jiných jazyků. Styly se zapisují buď do speciálního souboru, nebo přímo do těla HTML souboru. V kaskádových stylech se pomocí selektorů vybírají elementy, které je třeba upravit a pomocí příkazů s proměnnými atributy se nastaví požadovaný vzhled. Díky možnosti nastavení rozměrů pomocí relativních jednotek je možné snadno dosáhnout responzivního designu stránky. [10]

Skriptovací jazyk PHP má nejnovější verzi 7.3 a slouží k tvorbě dynamických webových stránek. Spolu s HTML pak tvoří základ stránky. Jazyk PHP běží na straně serveru a využívá se především k tvorbě dynamických dat na stránce, například při načítání dat z databáze nebo zobrazování různého obsahu pro různé uživatele. Výhodou je, že veškeré dění se provádí na serveru a k uživateli jde pouze výsledný kód stránky. [11]

Databáze MySQL je typ relační databáze, která v současné době patří k nejrozšířenějším. Tato databáze používá jazyk SQL podobně jako ostatní relační databáze a nově podporuje služby jako trigger, pohledy, funkce a procedury podobně jako Oracle databáze. Nejnovější verze této databáze pak je MySQL 8.0. [11], [11]

### **2.3.1 Použitý software**

Vývoj systému probíhal v IDE PhpStorm od společnosti JetBrains. Jedná se o nástroj přímo určený k tvorbě webových stránek a umí pracovat s jazyky PHP, HTML a JavaScript. Dalším programem byl XAMPP server, který sloužil jako lokální server pro běh systému. Posledním nástrojem byl MySQL Workbench, ve kterém byl vytvořen model databáze.

## **2.4 Databáze**

Hlavní a nejdůležitější část navrhovaného systému je jeho databáze. Ta byla vytvářena na základě požadavků na výsledný systém. Hlavním úkolem databáze je vytvořit datovou základnu pro data potřebná v prodejně stavebnin, jako informace o zboží, účtech zaměstnanců a klientů a také rezervací a historie nákupů.

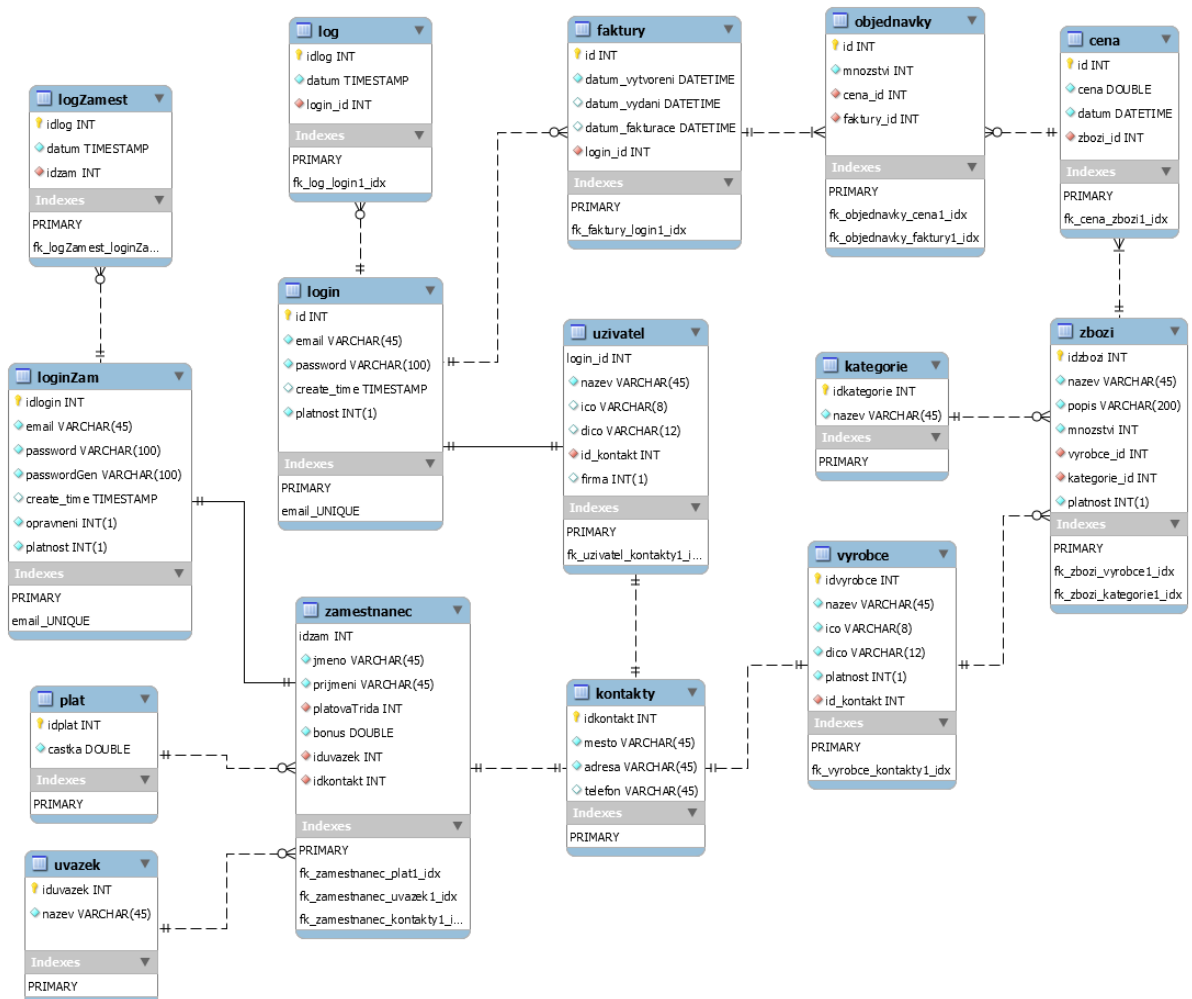
### **2.4.1 Model**

Model databáze byl vytvořen v programu MySQL Workbench. Model se skládá z 15 vzájemně propojených tabulek a je přiložen v příloze ve formátu PNG a MWB. Celý model se dá rozdělit na čtyři části starající se o data zboží (jde o tabulky zboží, cena, kategorie, výrobce), zaměstnanců (loginZam, zaměstnanec, plat, úvazek, logZamest), klientů (login, uživatel, log) a jejich nákupů nebo rezervací (faktury, objednávky).



Všechny tabulky kromě tabulek uživatel a zaměstnanec mají svůj vlastní primární klíč vygenerovaný pomocí funkce autoincrement v databázi MySQL. Tyto dvě zmíněné tabulky mají svůj primární klíč (primary key) propojený pomocí cizího klíče (foreign key) s primárním klíčem z tabulky login, respektive loginZam. Takto propojené tabulky tedy mají stejné primární klíče k zjednodušení práce s jejich daty.

Nejvíce zajímavou částí modelu je princip ukládání rezervací a historie nákupů. Slouží k tomu dvě tabulky: faktury a objednávky. Objednávky slouží jako jakési uložisko pro jednotlivé položky rezervace nebo nákupu a faktura tyto položky zastřešuje. O fakturách by se dalo říct, že to je jakýsi předek a objednávky jsou potomek. Jinak řečeno pomocí id faktury je možné v objednávkách najít jednotlivé položky. K rozpoznání, jestli jde o rezervaci nebo o nákup, slouží datum vydání, kdy u rezervace je tento sloupec prázdný. Po vydání zboží se pak vyplní příslušné datum a systém automaticky překlasifikuje rezervaci na nákup.



Obrázek 2.4 – ER diagram databáze

## 2.4.2 Tabulky

Databáze obsahuje celkem 15 tabulek, z toho tři jsou číselníky a zbytek obsahuje data systému. Sloupec platnost u všech záznamů znamená, zda je záznam platný nebo ne. Jinak řečeno, jestli je záznam smazaný nebo deaktivovaný, ale z důvodu zpětné komptability záznamů a údajů je v databázi stále uváděn.

Tabulka 2.1 – Tabulka logZamest

Klíč	Název sloupce	Datový typ
Primární klíč	idlog	int(11)
	datum	timestamp
Cizí klíč	idzam	int(11)

Tabulka logZamest slouží k uchování informací o aktivitě zaměstnanců. Do tabulky se automaticky zapisují časy přihlášení do systému pro daného zaměstnance. Ten je zde reprezentován pomocí cizího klíče z tabulky loginZam s kardinalitou 1:N.

Tabulka log je totožná s tabulkou logZamest, ale s tím rozdílem, že tato tabulka slouží k monitorování aktivity klientů. Sloupce jsou zde stejné, jen je zde cizí klíč na tabulku login také s kardinalitou 1:N.

Tabulka 2.2 – Tabulka loginZam

Klíč	Název sloupce	Datový typ
Primární klíč	idlogin	int(11)
Unikátní klíč	email	varchar(45)
	password	varchar(100)
	passwordGen	varchar(100)
	create_time	timestamp
	opraveni	int(1)
	platnost	int(1)

LoginZam slouží k uchování přihlašovacích údajů pro zaměstnance. Kromě svého ID a přihlašovacích údajů uchovává také heslo vygenerované systémem v nezašifrované podobě, toto prvotní heslo se po změně hesla zaměstnancem smaže. Dále jsou zde uloženy informace o úrovni oprávnění účtu a také zda je účet platný a nebyl již deaktivován. Tato tabulka je propojena kardinalitou 1:1 s tabulkou zaměstnanec.

Tabulka 2.3 – Tabulka zaměstnanec

Klíč	Název sloupce	Datový typ
Primární klíč, Cizí klíč	idzam	int(11)
	jmeno	varchar(45)
	prijmeni	varchar(45)
Cizí klíč	platovaTrida	int(11)
	bonus	double
Cizí klíč	iduvazek	int(11)
Cizí klíč	idkontakt	int(11)

Tabulka zaměstnanec používá jako svůj primární klíč cizí klíč z tabulky loginZam propojený s primárním klíčem tabulky loginZam. Propojené záznamy tak mají stejné primární klíče. Tabulka slouží k uchování informací o daném zaměstnanci. Informace o místě bydliště je v tabulce kontakty propojené s kardinalitou 1:1. Dále jsou zde napojeny dva číselníky, plat a úvazek, oba s kardinalitou 1:N.

Tabulka 2.4 – Tabulka kontakty

Klíč	Název sloupce	Datový typ
Primární klíč	idkontakt	int(11)
	mesto	varchar(45)
	adresa	varchar(45)
Cizí klíč	telefon	varchar(45)

Tabulka kontakty uchovává informace o bydlišti nebo sídle firmy a popřípadě i telefon. Jsou zde uloženy informace ze tří různých tabulek ze zaměstnanců, uživatelů a výrobců. Všechny tyto tabulky mají v kontaktech svoje údaje a všechny jsou napojené na tabulku kontakty s kardinalitou 1:1.

Tabulka plat je číselník obsahující pouze částku peněz, ze které jsou pak vytvořeny platové třídy v tabulce zaměstnanec. V tabulce úvazek je pak také číselník, ve kterém se uchovává název role v systému.

Tabulka 2.5 – Tabulka login

Klíč	Název sloupce	Datový typ
Primární klíč, Cizí klíč	idlogin	int(11)
Unikátní klíč	email	varchar(45)
	password	varchar(100)
	create_time	timestamp

	platnost	int(1)
--	----------	--------

Login je tabulka sloužící k uložení přihlašovacích údajů klientů. Je zde uloženo přihlašovací jméno ve formě emailu, zahashované heslo, čas vytvoření účtu a údaj o platnosti účtu. Tato tabulka je spojena s tabulkou faktury a s tabulkou uživatel. Kardinality zde jsou 1:N a 1:1.

Tabulka 2.6 – Tabulka uživatel

Klíč	Název sloupce	Datový typ
Primární klíč	login_id	int(11)
	nazev	varchar(45)
	ico	varchar(8)
	dico	varchar(12)
Cizí klíč	idkontakt	int(11)
	firma	int(1)

Uživatel uchovává informace o klientech. Kromě názvu subjektu je zde také uchováváno ičo a dič, pokud ho daný subjekt má, aby bylo možné rozeznat, zda jde o firmu nebo osobu, je zde sloupec umožňující uchovávat tuto informaci. Dále je zde spojení s tabulkou kontakty uchovávající kontaktní údaje na daný subjekt.

Tabulka 2.7 – Tabulka faktury

Klíč	Název sloupce	Datový typ
Primární klíč	id	int(11)
	datum_vytvoreni	datetime
	datum_vydani	datetime
	datum_fakturace	datetime
Cizí klíč	login_id	int(11)

Tabulka faktury slouží k uchovávání informací o rezervacích nebo o historii nákupu. Je napojena na tabulku login a na tabulku objednávky. Tabulka faktury pak obsahuje pouze tři sloupce s důležitými daty (vytvoření rezervace, vydání zboží, fakturace) a ID uživatele, který fakturu vytvořil. ID faktury slouží v tabulce objednávky k rozpoznání položek každé rezervace nebo nákupu.

Tabulka 2.8 – Tabulka objednávky

Klíč	Název sloupce	Datový typ
Primární klíč	id	int(11)
	mnozstvi	int(11)

Cizí klíč	cena_id	int(11)
Cizí klíč	faktury_id	int(11)

Objednávky obsahují jednotlivé položky každé faktury. Ty se rozlišují pomocí cizího klíče z tabulky faktury. Každý záznam obsahuje množství daného zboží, cizí klíč faktury a cizí klíč ukazující do tabulky cena, kde je pak samotná částka. To je z důvodu, aby si klient vyzvedl zboží za stejnou cenu, za kterou si ji rezervoval. Propojení je s tabulkou faktury 1:N a s tabulkou cena také 1:N.

Tabulka cena uchovává cenu zboží, datum změny a id zboží. Tato tabulka je pak dále propojená s tabulkou zboží s kardinalitou 1:N.

Tabulka 2.9 – Tabulka zboží

Klíč	Název sloupce	Datový typ
Primární klíč	idzbozi	int(11)
	nazev	varchar(45)
	popis	varchar(200)
	mnozstvi	int(11)
Cizí klíč	vyrobce_id	int(11)
Cizí klíč	kategorie_id	int(11)
	platnost	int(1)

Zboží uchovává informace o zboží na skladě. Je napojeno na číselník kategorie a na tabulky výrobce. V této tabulce jsou uloženy veškeré důležité informace jako množství, popis, název nebo platnost. V tabulce kategorie jsou pak uloženy názvy kategorií.

Poslední tabulkou je tabulka výrobce s informacemi o různých výrobcích jako název, ičo, dič a kontaktními údaji v propojené tabulce kontakty.

### 2.4.3 Procedury

Díky nově podporovaným technologiím v MySQL, jako například procedura, je nyní možné některé náročnější práce s databází uložit jako proceduru do samotné databáze. Takto naprogramovaná a uložená činnost je pak vykonávána rychleji a efektivněji, než kdyby byla naprogramována například v jazyce PHP. [13]

V navrhovaném systému můžeme najít jednu proceduru, které se stará o mazání rezervací, které si klient po uplynutí určité doby nevyzvednul. Za předpokladu, že je v databázi uloženo mnoho propadlých rezervací je pak možno mazání vykonávat efektivněji. Další nespornou výhodou pak je, že procedura se dá zavolat ručně uživatelem ze systému nebo se může po

příslušném nastavení v MySQL automaticky volat v nastavený čas, například každý den o půlnoci, kdy je systém nevyužíván. [11], [13]

## 2.5 Aplikace

Výsledný systém je v adresářové struktuře rozdělen do složek pro zvýšení přehlednosti zdrojových souborů. První složka se nazývá body a jsou v ní PHP soubory obsahující menu pro jednotlivé role uživatelů. Další složkou je css a jak již název napovídá, obsahuje soubory s CSS kódy pro stylování stránek. Již zajímavější je složka funkce, která obsahuje dva soubory, ve kterých jsou uloženy veškeré PHP funkce celého systému. Jeden soubor je určen pro funkce klientské části systému a druhý pro zaměstnaneckou část systému. Další složka s názvem images pak uchovává veškeré obrázky používané v systému. Veškeré stránky určené pro zaměstnaneckou činnost pak jsou ve složce zaměstnanec a soubory stránek určených pro klienty pak jsou umístěny přímo v kořenovém adresáři.

Vstupní bod do systému je soubor index.php jako u všech webových stránek. Z této hlavní stránky je pak možné si vybrat další stránky. Kromě pár základních stránek je většina stránek dostupných pouze po přihlášení do systému za pomoci klientského nebo zaměstnaneckého účtu. Tyto stránky se samozřejmě dále dělí na stránky, kam mohou pouze zaměstnanci v závislosti na jejich oprávněních (pracovní pozici) v systému.

Ochrana systému je zaručena pomocí naprogramovaného ošetření veškerých vstupů, které se v systému nacházejí. Jde o vstupy v URL adrese stránky nebo v textových vstupech na stránkách systému, jako jsou pole pro login a heslo na stránce přihlášení do systému. Není tak možné systém napadnout pomocí techniky jakou je SQL injection. Další ochranou před zneužitím systému a jeho potenciálně citlivých dat, pak je automatické odhlášení uživatelů, kteří jsou nečinní více než 30 minut.

Celý systém má jednotný vzhled, ve kterém se mění hlavní ovládací menu umístěné na vršku každé stránky. Menu je měněno podle role přihlášeného uživatele. Pro navrhovaný systém jsou doporučeny prohlížeče Google Chrome nebo Microsoft Edge z důvodu jejich velké dostupnosti na zařízeních. Vzhled systému je pak možné i upravit pro případné zájemce o tento systém. Díky zvolené architektuře, kdy jsou veškeré funkce do stránek přidávány jako funkce je možné snadno a rychle měnit jejich vzhled.

## 2.5.1 Popis vybraných stránek

Nejzajímavějšími stránkami jsou úvodní stránka, stránka se sestavami a stránka zobrazující faktury. Úvodní stránka obsahuje několik dynamických prvků, které mají pouze estetický účel. Na hlavní stránce nalezneme základní informace o stavebninách, odkazy v boxech s nejdůležitějšími stránkami a na spodku stránky jsou pak kontakty. Hlavním smyslem této stránky je zaujmout případné nové zákazníky a přesměrovat uživatele na stránku pro přihlášení.

Stránka s výpisem sestav je rozdělena do několika sekcí, ve kterých jsou zobrazovány různé informace dle typu sekce a také dle zadání příslušných kritérií. Nejvíce nahoře jsou čtyři přednastavené sestavy. Dále jsou sekce pro sestavy všech uživatelů, vybraného uživatele, vybraného zboží a vybrané kategorie. V těchto kategoriích je pak možné měnit zadání sestavy a měnit časový horizont, ve kterém se hledají data. Sestavy jsou generovány vždy pod dané zadání v přehledných tabulkách. Této funkce může využít pouze vedoucí prodejny nebo administrátor, aby se snížilo riziko odcizení citlivých dat.

Přednastavené

\* Největší odběratelé \* Nejčastější přihlášení \* Nejvýdělečnější zboží \* Nejčastěji prodané zboží

Zobrazit

Výběr dle kritérií

Pro všechny uživatele

\* Obrat \* Nejvíce objednávek

Od 01.12.2018 Do 28.04.2019

Zobrazit

Pro vybraného uživatele

Výběr uživatele

\* Nejčastěji kupované zboží \* Obrat \* Info o objednávkách

Od 01.12.2018 Do 28.04.2019

Obrázek 2.5 – Stránka se sestavami

Na stránce faktury je možné si fakturu nejen zobrazit, ale díky správnému formátování je možné si fakturu i vytisknout. V doporučených prohlížečích je možné si fakturu uložit do PDF souboru díky jejich možnosti vytvoření PDF souboru. Stačí kliknout na tlačítko nad fakturou a v dialogovém okně prohlížeče změnit položku vybraná tiskárna, kde zvolíme volbu uložit do PDF.

Faktura				
Email: lukas.karlík@stavob.cz		Číslo účtu: 2345128912/3000		Faktura číslo: 2019017, Variabilní symbol: 2019017
Dodavatel: <b>Stavobliny</b> Obránců míru 26 Pardubice IČO: 65454654 DIČ: CZ65454654			Oděratel: <b>Lukas Novan</b> Testovací Praha1 IČO: 27153025 DIČ: CZ2610302525	
			Den splatnosti: 23. 05. 2019 Forma úhrady: FP Datum vystavení faktury: 23. 04. 2019 Datum uskutečnění zápisu: 23. 04. 2019	
Zboží	Množství	Cena bez DPH	s DPH 21%	Celkem Kč s DPH
Stěna keram. Driam: Classic Star	90ks	33,55,-	40,-	4020,-
Stěna keram. Tondach: Dioda 12	1000ks	43,45,-	51,-	51000,-
Spoj. mat.: Hajpl: Široká šerohenný	15ks	1,58,-	2,-	30,-
Vlákno: Hajpl: Kladivo	1ks	118,5,-	143,-	143,-
Spoj. mat.: Hajpl: Hřibák 25x10	10ks	2,37,-	3,-	30,-
Plach: Hajpl: Plach omyt	4ks	3842,1,-	4648,-	18460,-
Střech: Hajpl: Široká panská	7ks	128,4,-	155,-	800,-
			<b>CELKEM bez DPH:</b>	<b>63218 Kč</b>
			<b>CELKEM s DPH:</b>	<b>80020 Kč</b>
			<b>K ÚHRADĚ:</b>	<b>80020 Kč</b>

Obrázek 2.6 – Stránka s fakturou

## 2.5.2 Popis rolí

Role v systému se dají rozdělit do dvou základních skupin na role pro klienty a pro zaměstnance. Tyto role jsou již patrné v use case diagramu. Klienti mají přístup do systému buď v roli neregistrovaného uživatele, kdy jsou funkcionality prakticky nulové, nebo jako registrovaný uživatel. Zaměstnanecké role jsou pak skladník, ekonom, vedoucí a administrátor.

Neregistrovaný uživatel může v systému na hlavní stránku nebo si procházet zboží, ale pro další funkce se buď musí přihlásit, nebo se registrovat. K dispozici má jednoduché menu s odkazy na zboží a odkazy pro přihlášení do systému, kdy se předpokládá, že toto menu bude převážně sloužit jako zkratka na stránky pro přihlášení.

Registrovanému uživateli už systém umožňuje si zboží vkládat do košíku, navíc má také možnosti pro spravování svého účtu například změnu jména nebo změnu hesla. Samozřejmostí pak je umožnění zobrazení jednotlivých rezervací s možností zobrazení podrobností objednávky a zobrazení již vyzvednutých rezervací včetně podrobností a faktury.

V zaměstnanecké části má nejnižší oprávnění role skladníka. Ten má v systému pouze funkcionality potřebné pro jeho práci. Nejdůležitější je možnost úpravy zboží, kdy může přidávat nové zboží, nebo zboží z nabídky odebírat, nebo měnit množství vybraného produktu. Dále si pak může zobrazit aktuální objednávky od klientů, kde má možnost zobrazení podrobností, označit objednávku jako vydanou nebo i objednávku zrušit. Pro případné kontroly je umožněno i zobrazení již vydaných objednávek. Samozřejmostí je možnost správy údajů vlastního účtu.



Vyšší rolí v systému je role ekonom. V této roli je předpokládána osoba pracující v kanceláři vyřizující administrativu. Kromě funkcí stejných jako má skladních je tu navíc možnost editace mezd zaměstnanců, vyfakturování vydaných objednávek nebo možnost vytvoření objednávky pro klienta přítomného na pobočce.

Další rolí je vedoucí prodejny. Tato role má možnost úpravy veškerých dat v systému. Kromě stejných funkcí jako mají předešlé role v systému je tu navíc správa výrobců a kategorií zboží, nebo možnost správy zaměstnaneckých účtů dostupných v systému. Důležitou funkcionalitou je pak výpis sestav s různými informacemi o prodejkách nebo zákaznících.

Poslední rolí je pak administrátor, který má jak veškeré možnosti vedoucího, tak navíc má možnost spravovat klientské účty, což je důležité například při zapomenutí hesla.

## 2.6 Instalační příručka

Pro spuštění systému je zapotřebí několik kroků. Nejdůležitější je stáhnutí některého z web serverů, které jsou dostupné na internetu. Vývoj byl prováděn na programu XAMPP server verze 3.2.2. Proto je právě tento server nebo jeho novější verze doporučena ke stažení a návod se bude vztahovat k tomuto programu.

Po stažení a nainstalování serveru je potřeba najít složku s nainstalovaným serverem, ta je typicky na disku C. Poté je nutné zkopírovat složku „code“ z příložených souborů do složky „htdocs“ v adresáři serveru. Nyní je možné spustit XAMPP server a v jeho okně spustit služby Apache a MySQL pomocí tlačítek start. Dalším krokem je nahrání databáze na běžící server. To můžeme provést pomocí předinstalovaného nástroje phpMyAdmin na adrese <http://localhost/phpmyadmin/>. Zde na záložce import vybereme příložený soubor bc\_karlik.sql a naimportujeme databázi. Pokud byl import databáze úspěšný, bude struktura vypadat jako na obrázku 2.7.

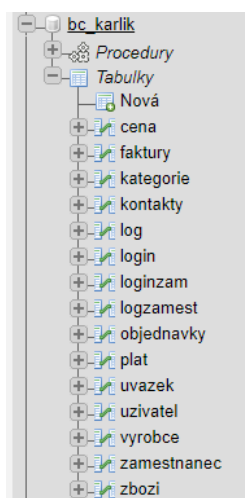
Nyní je systém dostupný na adrese <http://localhost/code/>. Při problémech s připojením systému k databázi je pak nutné zkontrolovat, zda se shodují jméno a přihlašovací údaje do databáze s údaji v souboru config.php.

Tabulka 2.10 – Přihlašovací údaje

Login	Heslo	Role	Poznámka
admin@admin.cz	123456	Administrátor	Přihlášení zaměstnanec
norvan@szn.cz	123456	Klient	Přihlášení uživatel

Přihlašovací údaje dalších zaměstnanců jsou na stránce editZamestnanec dostupné v horním menu administrátora v submenu účty. Zaměstnanci mají ještě vygenerované heslo od systému

pro první přihlášení, které je pro administrátora viditelné a zaměstnanec by si ho po prvním přihlášení měl změnit.



Obrázek 2.7 – Struktura databáze v phpMyAdmin

## 2.7 Uživatelská příručka

Veškeré funkcionality jsou dostupné přes horní menu, aby byl systém co nejpřehlednější a nejjednodušší na zapamatování a ovládání. To platí jak pro klientskou část, tak pro zaměstnaneckou část. Tlačítka pro stránky přihlášení nebo registraci jsou v horním menu nebo na úvodní stránce ve formě velkých tlačítek. Pokud je u položky menu znak plus jedná se o submenu a po najetí na tuto položku budou zobrazeny další položky.

Po přihlášení je vždy poslední položka tlačítko na odhlášení. Pokud je účet 30 minut neaktivní dojde k automatickému odhlášení. Na obrázku 2.8 jsou zobrazená jednotlivá menu. Od shora to jsou menu pro role neregistrovaný, registrovaný, skladník, ekonom a vedoucí s administrátorem.



Obrázek 2.8 – Ukázka různých menu

### **2.7.1 Klientská část**

Po přihlášení je klient přesměrován na úvodní stránku, kde dostane pár základních informací a v případě zlevnění zboží i informaci o této změně. Pro rezervaci zboží je nutné jít na stránku s nabídkou zboží a po rozkliknutí daného zboží je možné vložit do košíku požadované množství a pak dále pokračovat ve vybírání zboží. Následně je nutné přejít na stránku košíku a potvrdit rezervaci vybraného zboží, čímž se v systému vytvoří objednávka.

Veškeré informace o rezervacích jsou dostupné přes submenu rezervace, kde si klient může zobrazit aktuální objednávky nebo již proběhlé nákupy. Na těchto stránkách je pak možné si zobrazit podrobnosti objednávek nebo si zobrazit faktury.

Pokud je potřeba si upravit jakékoliv údaje, je to možné přes odkaz v menu s názvem účet, kde si klient může změnit veškeré dříve zadané údaje.

### **2.7.2 Zaměstnanecká část**

Ovládání zaměstnanecké části je velice podobné jako ovládání v té klientské. Veškeré funkcionality dostupné pro danou roli jsou v horním menu a ovládání je vždy sjednocené nebo podobné. Většina stránek obsahuje tabulky s výpisy dat a na pravé straně jsou pak tlačítka pro další činnosti s daným záznamem. Může jít o editaci daného záznamu, zobrazení podrobností nebo smazání zboží z nabídky.

Jedna z výjimek je stránka pro generování sestav, kde je stránka rozdělena do několika pod sebou umístěných sekcí. Každá sekce pak generuje různé sestavy s volbou různých kritérií. Další výjimka pak je stránka pro editaci kategorií a výrobců, kde je všechno umístěno na jedné stránce. Obě tyto stránky jsou v submenu správa.

Pro editaci jakýchkoli účtů je nutné být přihlášen jako administrátor. Pak jsou v submenu účty dostupné stránky jak pro klienty, tak pro zaměstnance. Obě stránky vypisují účty do tabulky, ze které je pak možné vybrané účty editovat a v případě zaměstnanců kontrolovat jejich aktivitu. Pro přidání nového zaměstnance je tlačítko umístěno nad tabulkou se zaměstnaneckými účty. Po vytvoření nového zaměstnaneckého účtu je vygenerováno heslo, které je zobrazováno v nezašifrované podobě, aby bylo možné ho předat zaměstnanci k prvnímu přihlášení. Po změně hesla se u daného účtu žádné heslo nezobrazuje.

## ZÁVĚR

Cílem této bakalářské práce bylo vytvoření funkčního informačního systému pro prodejnu stavebnin. Vývoj byl založen na poznatcích z provedené rešerše podobných systémů dostupných na trhu. Z těchto poznatků pak vyvstaly požadavky na funkcionality systému a nedostatky stávajících systémů. Tyto nedostatky pak byly v navrhovaném systému vyřešeny.

Nejobtížnější částí vývoje bylo vytvoření vhodného databázového modelu, který by umožňoval ukládat a následně poskytovat veškeré potřebné informace. Tento model byl po nutné analýze úspěšně vytvořen a mohla být na jeho základě naimplementována webová aplikace. Webové prostředí bylo zvoleno tak, aby vyřešilo největší nedostatek porovnávaných systémů, které nejsou multiplatformní. Systém je naimplementován v jazyce PHP, který je možné spustit v každém internetovém prohlížeči, tím je zaručena dostupnost systému téměř na každém zařízení s přístupem na internet. Díky kaskádovým stylům je pak aplikace responzivní a nedělá jí problém zobrazení na různých typech zařízení.

Vývoj systému byl zdárně dokončen za dodržení vytyčených cílů a požadavků, kterými byly klientská část, zaměstnanecká část s podporou různých rolí, jednoduché ovládání a velká dostupnost na různých platformách. Tato bakalářská práce byla užitečná především k získání nových zkušeností s vytvářením databázových modelů, tvorbou SQL dotazů a webových aplikací. Navrhovaný systém je primárně určený pro potřeby bakalářské práce, ale mohl by být použitý i v ostrém nasazení, kde by ovšem byl zastíněn informačními systémy vytvořenými firmami, které jsou přeci jenom dokonalejší.

Do budoucna by bylo vhodné otestovat systém v náročném provozu pro optimalizaci SQL dotazů a nalezení případných chyb, které nebyly objeveny během vývoje. Systém je možné v budoucnu snadno rozšířit o další nové funkce, například o implementaci platební brány, grafů nebo věrnostních slev pro klienty.

## POUŽITÁ LITERATURA

- [1] ŠMÍD, Vladimír. Pojem informačního systému. *Management informačního systému* [online]. [cit. 2019-03-15]. Dostupné z: <http://www.fi.muni.cz/~smid/mis-infsys.htm>
- [2] HRONEK, Jiří. *Informační systémy* [online]. Olomouc, 2007 [cit. 2019-03-15]. Dostupné z: <http://phoenix.inf.upol.cz/esf/ucebni/infoSys.pdf>. Univerzita Palackého v Olomouci.
- [3] MOLNÁR, Zdeněk. *Podnikové informační systémy*. Vyd. 2., přeprac. Praha: Česká technika – nakladatelství ČVUT, 2009. ISBN 978-80-01-04380-6.
- [4] VONDRÁK, Ivo. *Úvod do softwarového inženýrství* [online]. 2002 [cit. 2019-03-15]. Dostupné z: [http://vondrak.cs.vsb.cz/download/Uvod\\_do\\_softwaroveho\\_inzenyrstvi.pdf](http://vondrak.cs.vsb.cz/download/Uvod_do_softwaroveho_inzenyrstvi.pdf)
- [5] SKLENÁŘ, Vladimír. *Softwarové inženýrství* [online]. Olomouc, 2007 [cit. 2019-03-15]. Dostupné z: <https://phoenix.inf.upol.cz/esf/ucebni/syspro.pdf>. Univerzita Palackého v Olomouci.
- [6] TÝM ASOCIACE MALÝCH A STŘEDNÍCH PODNIKŮ A ŽIVNOSTNÍKŮ ČR. Analýza řemeslných oborů. *Asociace malých a středních podniků a živnostníků ČR* [online]. únor 2019 [cit. 2019-03-20]. Dostupné z: [http://amsp.cz/wp-content/uploads/2019/02/Anal%C3%BDza-%C5%99emesla-2\\_2019.pdf](http://amsp.cz/wp-content/uploads/2019/02/Anal%C3%BDza-%C5%99emesla-2_2019.pdf)
- [7] K2 atmitec. *K2 atmitec: Informační systém K2: podnikový software pro úspěšné firmy* [online]. 2013 [cit. 2019-03-26]. Dostupné z: <https://www.k2.cz/cs>
- [8] QI Group. *QI: Informační systémy QI – CENTRÁLNÍ MOZEK FIRMY* [online]. 2001 [cit. 2019-03-26]. Dostupné z: <https://www.qi.cz/>
- [9] Asseco Solutions. *Helios Asseco: Informační systémy HELIOS pro všechna odvětví* [online]. 2019 [cit. 2019-03-28]. Dostupné z: <https://www.helios.eu/>
- [10] DRUSKA, Peter. *CSS a XHTML: tvorba dokonalých webových stránek krok za krokem*. Praha: Grada, 2006. Průvodce (Grada). ISBN 80-247-1382-9.
- [11] NARAMORE, Elizabeth, Jason GERNER, Scouarnec YANN LE and Timothy BORONCZYK. *PHP 6, MySQL, Apache: vytváříme webové aplikace*. Brno: Computer Press, 2009. ISBN 9788025127674.
- [12] GROFF, James R. a Paul N. WEINBERG. *SQL: kompletní průvodce*. Brno: CP Books, 2005. Programování (CP Books). ISBN 80-251-0369-2.
- [13] MySQL. *MySQL Documentation: The world's most popular open source database* [online]. 2016 [cit. 2019-04-22]. Dostupné z: <https://dev.mysql.com/doc/>

## **PŘÍLOHY**

Příloha A – Funkce pro přidání zboží do košíku .....	47
Příloha B – Funkce pro zobrazení náhledu zboží .....	48
Příloha C – Zdrojové kódy a dokumenty .....	50

## PŘÍLOHA A – FUNKCE PRO PŘIDÁNÍ ZBOŽÍ DO KOŠÍKU

```
function rezervace() {
    $db = spojeni();
    if (isset($_POST["sended"])) {
        if (empty($_POST["mnozstvi"])) {
            echo "<p class='hlaska'>Vypln formulář</p>";
        } else {
            if (!preg_match("/^[0-9]+$/", $_GET['id'])) {
                return;
            }
            $id = $_GET['id'];
            $mn = $_POST["mnozstvi"];

            $sql = "SELECT SUM(`objednavky`.`mnozstvi`) AS rez, `zbozi`.`mnozstvi`
FROM `objednavky` JOIN `faktury` ON `objednavky`.`faktury_id`=faktury.id JOIN `cena` ON
`cena`.`id`=`objednavky`.`cena_id` JOIN zbozi ON zbozi.idzbozi=cena.zbozi_id
WHERE `faktury`.`datum_vydani` IS NULL AND `cena`.`zbozi_id`={ $id}";

            if ($data = $db->query($sql)) {
                if ($data->num_rows > 0) {

                    $row = $data->fetch_assoc();
                    $mnozRez = $row['mnozstvi'] - $row['rez'];
                    if ($mnozRez >= $mn) {
                        if (!isset($_SESSION['kosik'])) {
                            $_SESSION['kosik'] = array();
                        }

                        if (array_key_exists($id, $_SESSION['kosik'])) {
                            $_SESSION['kosik'][$id]['mnozstvi'] = $mn;
                        } else {
                            $_SESSION['kosik'][$id];
                            $_SESSION['kosik'][$id]['mnozstvi'] = $mn;
                        }

                        header("Location:zbozi.php");
                    } else {
                        echo "<h1 class='nadpis_vedlejsi_stranka'>Zadané množství je velké.</h1>";
                    }
                }
            }
        }
    }
}
```

## PŘÍLOHA B – FUNKCE PRO ZOBRAZENÍ NÁHLEDU ZBOŽÍ

```
function nahledZbozi()
{
    $db = spojeni();
    if (isset($_GET['id'])) {
        if (!preg_match("/^[0-9]+$/", $_GET['id'])) {
            return;
        }

        $sql = "SELECT `zbozi`.`idzbozi` AS id, `zbozi`.`nazev`, (SELECT cena.cena FROM cena
        WHERE zbozi_id=`zbozi`.`idzbozi` ORDER BY cena.datum DESC LIMIT 1) AS cena ,
        `zbozi`.`mnozstvi`, `zbozi`.`popis`, `kategorie`.`nazev` AS kat, `vyrobce`.`nazev` AS vyr
        FROM `zbozi` JOIN `cena` ON `zbozi`.`idzbozi`=`cena`.`zbozi_id` JOIN `kategorie` ON
        `kategorie`.`idkategorie`=`zbozi`.`kategorie_id` JOIN `vyrobce` ON
        `vyrobce`.`idvyrobce`=`zbozi`.`vyrobce_id` WHERE `zbozi`.`idzbozi`={$_GET['id']} GROUP BY
        `zbozi`.`nazev`";

        if ($data = $db->query($sql)) {
            if ($data->num_rows > 0) {

                while ($row = $data->fetch_assoc()) {
                    echo "<div class='nahledZbozi'>";
                    echo "<div class='fotoNahled'>";

                    $nazevSlozky = "images/obrazkyZbozi/" . $row['id'];
                    if (is_dir($nazevSlozky)) {
                        $slozka = opendir($nazevSlozky);
                        for ($i = 0; $i < 1; $i++) {
                            $files = scandir($nazevSlozky);
                            $pocetSoub = count($files);
                            while ($nazevSouboru = readdir($slozka)) {
                                if ($nazevSouboru != "." && $nazevSouboru != ".." && $nazevSouboru !=
                                "nahledy" && $nazevSouboru != "thumbs.db" && $nazevSouboru != "Thumbs.db") {

                                    echo "<img src='$nazevSlozky/$nazevSouboru' alt=" class='zkft'>";
                                    break;
                                } else if ($pocetSoub <= 2) {

                                    $sodkaz = "images/1.jpg";
                                    echo "<img src='$sodkaz' alt=" class='zkft'>";
                                    break;
                                }
                            }
                        }
                    }
                }

                echo "</div>";
                $sqlCena = "SELECT SUM(`objednavky`.`mnozstvi`) AS rez
                FROM `objednavky` JOIN `faktury` ON `objednavky`.`faktury_id`=faktury.id JOIN
                `cena` ON `cena`.`id`=`objednavky`.`cena_id
                WHERE `faktury`.`datum_vydani` IS NULL AND `cena`.`zbozi_id`={$_GET['id']}";
                if ($dataR = $db->query($sqlCena)) {
                    if ($dataR->num_rows > 0) {
```





## **PŘÍLOHA C – ZDROJOVÉ KÓDY A DOKUMENTY**

K práci jsou přiloženy veškeré zdrojové kódy systému, databázový model, SQL příkazy pro vytvoření databáze s testovacími daty. Dále jsou přiloženy obrázky modelu databáze, use case diagramu a rich picture.