

Návrh informačního systému pro potravinářskou výrobu

Kryštof Knápek

Bakalářská práce
2023



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2022/2023

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Kryštof Knápek
Osobní číslo: A20517
Studijní program: B0613A140020 Softwarové inženýrství
Forma studia: Prezenční
Téma práce: Návrh informačního systému pro potravinářskou výrobu
Téma práce anglicky: Design of a Food Production Information System

Zásady pro vypracování

1. Vypracujte literární rešerši na dané téma.
2. Navrhněte a specifikujte požadavky na systém.
3. Proveďte funkční a datovou analýzu problémové oblasti.
4. Vytvořte model systému s využitím diagramů UML.
5. Realizujte prototyp aplikace pomocí drátěných modelů.

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. ARLOW, Jim a Ila NEUSTADT. UML 2 a unifikovaný proces vývoje aplikací: objektově orientovaná analýza a návrh prakticky. 2., aktualiz. a dopl. vyd. Brno: Computer Press, 2007, 567 s. ISBN 9788025115039.
2. FOWLER, Martin. Destilované UML. Praha: Grada, 2009. Knihovna programátora (Grada). ISBN 9788024720623.
3. KRAVAL, Ilja. Analytické modelování informačních systémů pomocí UML v praxi. 1. Brno: Object Consulting, 2010. ISBN 978-80-254-6986-6.
4. SOMMERVILLE, Ian. Softwarové inženýrství. Brno: Computer Press, 2013, 680 s. ISBN 9788025138267.
5. PILONE, Dan a Neil PITMAN. UML 2.0 in a nutshell. Sebastopol, Calif.: O'Reilly, c2005, xv, 216 s. ISBN 0596007957.

Vedoucí bakalářské práce:

Ing. Darina Bajusová

Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce:

2. prosince 2022

Termín odevzdání bakalářské práce:

26. května 2023



doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan

prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 7. prosince 2022

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 24.5.2023

Kryštof Knápek, v.r.
podpis studenta

ABSTRAKT

Cílem bakalářské práce je navrhnout potravinářský podnikový systém, který firmě usnadní plánování výroby (objednávky, správa receptů k přepočtu dávek, ...). Součástí práce bude zpracován přehled stávajících řešení, analýza požadavků, funkční a datová analýza zdokumentovaná vytvořením modelu s využitím diagramů UML a návrh prototypu aplikace realizovaný formou drátěných modelů.

Klíčová slova: návrh prototypu, potravinářský podnikový systém, diagramy UML, drátěné modely, analýza požadavků, funkční a datová analýza

ABSTRACT

The aim of the bachelor's thesis is to design a food industry enterprise system, which will facilitate production planning (orders, management of recipes for batch recalculations, etc.) for the company. The work will include a review of existing solutions, requirements analysis, functional and data analysis documented by creating a model using UML diagrams, and the design of a prototype application implemented in the form of wireframe models.

Keywords: prototype design, food business system, UML diagrams, wireframe models, requirements analysis, functional and data analysis

Chtěl bych poděkovat vedoucí bakalářské práce Ing. Darině Bajusové za všechny konzultace, rady a odborné vedení při vypracovávání práce.

Prohlašuji, že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST.....	10
1 INFORMAČNÍ SYSTÉM.....	11
1.1 SLOŽKY INFORMAČNÍHO SYSTÉMU	11
1.1.1 Lidé	11
1.1.2 Hardware	12
1.1.3 Software	12
1.1.4 Data	13
1.1.4.1 Data vs informace	13
1.1.5 Komunikační sítě	13
1.1.6 Zabezpečení.....	14
1.2 DRUHY INFORMAČNÍCH SYSTÉMŮ	14
1.2.1 Podnikové informační systémy	14
1.2.1.1 Executive Information Systems (EIS)	14
1.2.1.2 Enterprise Resource Planning (ERP).....	14
1.2.1.3 Customer Relationship Management (CRM)	15
1.2.1.4 Supply Chain Management (SCM).....	15
1.2.1.5 Manufacturing Execution Systems (MES)	15
1.2.2 Veřejné informační systémy	16
1.3 IS PRO PEKÁRNY NA ČESKÉM TRHU	16
1.3.1 Retail Systems (Evidence pekáren a cukráren)	16
1.3.2 MC Software (Odbyt a výroba pekárny).....	17
1.3.2.1 Číselníky	17
1.3.2.2 Expedice.....	18
1.3.2.3 Fakturace.....	18
1.3.2.4 Výroba	19
1.3.2.5 Sklady	19
1.3.3 Software PORS	19
1.3.4 COMPEX spol. s r.o.....	20
1.3.5 Shrnutí	21
2 UML.....	22
2.1 VÝZNAM SLOVA „UNIFIKOVANÝ“	22
2.2 POUŽITÍ UML	23
2.3 OBJEKTY V JAZYCE UML.....	24
2.4 STAVEBNÍ BLOKY JAZYKA UML	24
2.5 VĚCI.....	24
2.6 VZTAHY (RELACE)	25
2.7 DIAGRAMY	25
2.7.1 Diagramy chování	26
2.7.1.1 Diagram případu užití	26
2.7.1.2 Sekvenční diagram.....	27
2.7.2 Diagramy struktury	29
2.7.2.1 Diagram tříd.....	29
3 ENTITNĚ – RELAČNÍ DIAGRAM.....	34

4	POŽADAVKY	36
4.1	FUNKČNÍ POŽADAVKY	36
4.2	NEFUNKČNÍ POŽADAVKY	36
5	WIREFRAME	37
II	PRAKTICKÁ ČÁST	38
6	NÁVRH A SPECIFIKACE POŽADAVKŮ	39
6.1	NEFUNKČNÍ POŽADAVKY	39
6.2	FUNKČNÍ POŽADAVKY	40
6.2.1	Správa produktu	42
6.2.2	Správa zaměstnanců	42
6.2.3	Správa receptů	43
6.2.4	Správa surovin.....	44
6.2.5	Správa objednávek	44
6.2.6	Správa zákazníků	45
6.2.7	Správa faktur a dodacích listů	46
6.2.8	Správa skladu	47
6.2.9	Správa denní výroby	47
6.2.10	Správa přihlášení	47
7	NÁVRH DATABÁZE POMOCÍ ERD	49
7.1	ENTITY A JEJICH ATRIBUTY	50
7.2	VZTAHY	52
8	NÁVRH DIAGRAMU TŘÍD	55
8.1	TŘÍDY	56
8.1.1	Objednávka	56
8.1.2	Zákazník	56
8.1.3	Produkt	56
8.1.4	Kategorie	56
8.1.5	Faktura.....	56
8.1.6	Dodací_list	57
8.1.7	Produkt_Objednavka.....	57
8.1.8	Produkt_Výroba	57
8.1.9	Denní_Výroba	57
8.1.10	Recept.....	57
8.1.11	Surovina_Recept	57
8.1.12	Surovina	58
8.1.13	Sklad.....	58
8.1.14	Zaměstnanec.....	58
8.1.15	Vedoucí_Výroby	58
8.1.16	Majitel	58
8.1.17	Skladník.....	58
8.1.18	Pokladní.....	58
8.1.19	Účetní	58
8.1.20	Regál	59
8.1.21	Sklad.....	59
8.2	VÝČTOVÉ TYPY	59
8.2.1	Typ_alergenu.....	59
8.2.2	Typ_jednotky	59

8.2.3	Typ_směny	59
9	REALIZACE FUNKČNÍCH POŽADAVKŮ	60
9.1	AKTÉŘI.....	60
9.2	PŘÍPADY UŽITÍ.....	62
10	DIAGRAMY PROCESŮ	85
11	DRÁTĚNÉ MODEL Y	90
	ZÁVĚR	95
	SEZNAM POUŽITÉ LITERATURY	96
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	100
	SEZNAM OBRÁZKŮ	101
	SEZNAM TABULEK.....	102
	SEZNAM PŘÍLOH.....	103

ÚVOD

Tato bakalářská práce se zaměřuje na navržení potravinářského podnikového systému, jehož cílem je usnadnit firmám plánování výroby, včetně aspektů jako jsou objednávky, správa receptů a přepočítání dávek. Zásadní součástí této práce je zpracování přehledu stávajících řešení, podrobná analýza požadavků, funkční a datová analýza, a vytvoření modelů pomocí diagramů UML.

Cílem je vytvořit model, který poskytuje komplexní a integrované řešení pro řízení a optimalizaci procesů v rámci potravinářského podniku. Tato práce je výsledkem kombinace teoretického výzkumu a praktické aplikace získaných poznatků.

Konečným výsledkem bude návrh prototypu aplikace, který bude realizován formou drátěných modelů. Tyto modely poskytnou představu o tom, jak by mohl vypadat konečný systém, a umožní nám identifikovat potenciální problémy a vylepšení před samotnou fází implementace.

Výsledek této práce je jen první krok v cestě k vývoji efektivního a uživatelsky přívětivého informačního systému pro správu receptů v pekárenství. Práce je ukončena komplexním návrhem systému, ale je třeba pokračovat v dalších krocích, jako je implementace a testování.

I. TEORETICKÁ ČÁST

1 INFORMAČNÍ SYSTÉM

Informační systém představuje klíčový nástroj pro podporu rozhodování, koordinaci a řízení v organizaci. Jeho struktura se skládá z různých propojených částí, které slouží k shromažďování, zpracování, ukládání a distribuci informací. Mezi tyto části patří lidé, hardwarové prvky, softwarové aplikace, komunikační sítě a datové zdroje. Hlavními činnostmi informačního systému jsou sběr surových dat (vstup), jejich přeměna do smysluplné podoby (zpracování) a rozšiřování zpracovaných dat (výstup). Pro správné fungování informačního systému je rovněž klíčová zpětná vazba, která umožňuje hodnotit a upravovat vstupní informace.

Pro pochopení informačních systémů je nezbytné chápat problémy, které tyto systémy řeší, jejich strukturu, návrhové prvky a organizační procesy, které směřují k jejich řešení. Důležité oblasti znalostí zahrnují základní pojmy, informační technologie, aplikace v organizačním prostředí, vývojové metody a řízení. Moderní informační systémy využívají počítače a software, které jsou nezbytné pro jejich provoz, ale samy o sobě nedokážou vytvářet potřebné informace. Proto je důležité mít vhled do celkové historie a architektury daného systému. [1, 2]

1.1 Složky informačního systému

V této kapitole se zaměříme na hlavní složky informačního systému a jejich přínosy. Kapitola bude rozdělena do více podkapitol, které se jednotlivým složkám budou více věnovat.

1.1.1 Lidé

Při hodnocení informačních systémů je nutné zohlednit jak technické, tak lidské faktory. Lidé jsou pro fungování informačních systémů klíčoví, včetně role od uživatelské podpory až po CIO. I přes rostoucí automatizaci je stále potřeba odborníků, kteří rozumí datům, jako jsou obchodní analytici a specialisté na informační bezpečnost. Koncoví uživatelé, systémoví analytici, vývojáři softwaru a operátoři systémů jsou dalšími nezbytnými účastníky v procesu. Lidský faktor je nejdůležitějším prvkem informačních systémů, který proměňuje data a informace na užitečné poznatky pro porozumění minulosti a řízení budoucího chování. [3, 5]

1.1.2 Hardware

Hardware informačních systémů zahrnuje všechny fyzické, viditelné komponenty a nástroje používané pro vstup, zpracování a výstup dat. Tyto komponenty zahrnují systémové jednotky (například počítače a notebooky), interní zařízení a periferní zařízení (klávesnice, monitory, tiskárny, skenery), stejně jako paměťová média pro ukládání dat. Periferní zařízení mohou být k systémové jednotce připojena bezdrátově nebo kabelově a komunikují s jejími interními součástmi prostřednictvím nainstalovaného softwaru. Mezi hardwarové příklady patří počítače všech velikostí a tvarů, včetně mobilních zařízení, jako jsou chytré telefony, tablety a hodinky, které mohou shromažďovat, ukládat, zpřístupňovat a spravovat obrovské množství dat. [1, 2]

1.1.3 Software

Software je soubor abstraktních instrukcí, které řídí hardware, a bez kterého by počítač nemohl vykonávat žádné úlohy nebo funkce. Software může být uzavřený, což omezuje možnost uživatele měnit nebo upravovat kód, nebo otevřený, což umožňuje uživatelům kód měnit a upravovat. Software zahrnuje počítačové programy a postupy, které popisují, jak se systém ovládá. Tyto softwarové zdroje, včetně návodů a postupů pro zpracování informací, jsou nezbytné pro správné fungování všech informačních systémů, ať už jsou digitální nebo tradičnější.

1. **Systémový software:** Rozhraní mezi hardwarem a aplikačním softwarem zajišťuje systémový software, například operační systém (např. Microsoft Windows, Ubuntu Linux, Google Android, Apple iOS). Jeho úkolem je organizovat a spravovat hardwarovou část počítače.
2. **Aplikační software:** Uživatelé mohou používat aplikační software, jako jsou tabulky Google nebo aplikace Microsoft Outlook, k provádění konkrétních úkolů, například k manipulaci s tabulkami, vytváření dokumentů nebo navrhování webových stránek.
3. **Databázový software:** Databázový software umožňuje ukládat, zpracovávat a vyhledávat data organizovaným a efektivním způsobem. Příklady systémů, které provádějí správu databází, jsou MySQL, PostgreSQL, Oracle Database a Microsoft SQL Server.

4. **Middleware:** Software, známý jako middleware, umožňuje integraci a komunikaci mezi různými aplikacemi a systémy. Často zahrnuje technologie, jako jsou webové služby, rozhraní API a integrační nástroje.
5. **Bezpečnostní software:** Software používaný k ochraně informačních systémů a dat před neoprávněným přístupem, zneužitím a škodlivým softwarem se považuje za bezpečnostní software. Příkladem bezpečnostního softwaru jsou antivirové programy, firewally, systémy detekce narušení a šifrování. [1, 2, 3, 4, 5]

1.1.4 Data

Data jsou nehmotným základem informačních systémů a obsahují různé informace, které jsou cenné pro obchodní rozhodování. Databáze a datové sklady organizují a uchovávají data potřebná pro každodenní operace a pro informované rozhodování. Data mohou mít různé formy, včetně alfanumerických, textových, obrazových nebo zvukových. Technologie správy dat řídí organizaci dat ve fyzických úložištích. V éře "velkých dat" jsou databáze a datové sklady klíčové pro ukládání, zpracování a analýzu dat, která mohou odhalovat složité vztahy a podporovat rozhodování. Tyto zdroje dat, které se často spravují pomocí technologií jako jsou databáze a znalostní báze, uchovávají různé typy informací používaných ve znalostním managementu a expertních systémech. [3, 4, 5]

1.1.4.1 Data vs informace

Informace jsou relevantní, zpracované a uspořádané údaje. Naproti tomu data jsou nezpracovaná, neorganizovaná fakta, která potřebují další zpracování, aby měla smysl. Příkladem dat jsou výsledky testů žáků. Příkladem znalostí je třídní průměr odvozený z těchto dat. [2, 6]

1.1.5 Komunikační síť

Síť a komunikace, jako je internet, intranet a extranet, jsou klíčovými komponenty informačních systémů, které umožňují elektronické podnikání a obchodování. Propojují počítače a další zařízení prostřednictvím různých komunikačních médií a umožňují sdílení různých typů dat. Internet slouží jako globální technologická platforma pro vývoj nových produktů, služeb a obchodních modelů, a také pro interní komunikaci v rámci společností. Společnosti používají intranet a extranet ke koordinaci svých interních a externích aktivit. World Wide Web je online zdroj umožňující ukládání, vyhledávání, formátování a zobrazování dat v síti

propojených webových stránek. Sítě jsou zásadní pro informační systémy, protože umožňují přístup k datům uloženým v cloudu prostřednictvím telekomunikací. [1, 3, 4]

1.1.6 Zabezpečení

Obecné zabezpečení informačních systémů zahrnuje firewally, šifrování, zálohování, ověřování a další fyzickou a logickou ochranu dat a systémů. Jedná se o zásady a postupy na ochranu informačních systémů a dat před neoprávněným přístupem a možným zneužitím nebo poškozením. [1, 2]

1.2 Druhy informačních systémů

Tato podkapitola se zabývá jednotlivými druhy informačních systémů a jejich členěním do dvou kategorií. Jedna z nich je kategorie podnikových IS a druhou je kategorie veřejných IS.

1.2.1 Podnikové informační systémy

Podnikové informační systémy hrají důležitou roli v řízení a fungování organizace, protože umožňují efektivní sběr, zpracování a analýzu dat ve strategických a taktických rozhodovacích procesech. Tyto systémy lze dále rozdělit do několika kategorií.

1.2.1.1 *Executive Information Systems (EIS)*

Executive Information Systems (EIS) jsou systémy určené pro vrcholový management, které poskytují informace a analytické nástroje pro rozhodování. EIS umožňují efektivní sledování klíčových ukazatelů výkonnosti a podporují strategické rozhodování. Mezi klíčové funkce EIS patří přístup k širokému spektru podnikových informací, grafické a vizuální zobrazení dat pro identifikaci trendů a vzorců, analytické nástroje pro prediktivní modelování, sledování klíčových ukazatelů výkonnosti (KPI), podpora rozhodování poskytující relevantní data a analýzy pro porovnání různých scénářů, a schopnost personalizace a přizpůsobení se potřebám jednotlivých vedoucích pracovníků. [1, 2, 8]

1.2.1.2 *Enterprise Resource Planning (ERP)*

Systémy pro plánování podnikových zdrojů (ERP) kombinují data a operace z různých podnikových oblastí, jako je výroba, nákup, prodej, logistika a finance, čímž zvyšují efektivitu a kontrolu nad celkovým fungováním organizace. Klíčové funkce ERP systémů zahrnují integraci dat z různých podnikových zdrojů a oddělení, automatizaci procesů pro snížení chybovosti a zvýšení produktivity, plánování a sledování zdrojů pro zlepšení cash flow

a kontroly zásob, sledování výkonnosti pro podporu rychlého a informovaného rozhodování, a řízení dodavatelského řetězce pro lepší spolupráci s dodavateli a zákazníky a efektivnější logistiku. [1, 2, 7]

1.2.1.3 Customer Relationship Management (CRM)

Systémy pro řízení vztahů se zákazníky (CRM) podporují prodej, marketing a služby zákazníkům tím, že pomáhají společností spravovat a rozvíjet vztahy se zákazníky. CRM systémy shromažďují a analyzují údaje o zákaznících, umožňují společností lépe porozumět jejich potřebám a poskytovat jim personalizované produkty a služby. Klíčové funkce CRM systémů zahrnují ukládání a správu údajů o zákaznících, sledování a řízení marketingových a prodejních aktivit, poskytování analytických nástrojů pro rychlé a informované rozhodování a integraci s jinými systémy, jako jsou ERP a SCM, pro lepší spolupráci mezi odděleními. [1, 2, 7]

1.2.1.4 Supply Chain Management (SCM)

Systémy řízení dodavatelského řetězce (SCM) koordinují pohyb materiálů, informací a finančních prostředků mezi účastníky dodavatelského řetězce, včetně dodavatelů, výrobců, distributorů a zákazníků. Hlavními cíli SCM systémů je zlepšení spolupráce mezi partnery a optimalizace dodavatelského řetězce. SCM systémy poskytují nástroje pro předvídání poptávky, plánování výroby, nákup a distribuci výrobků. Řešení pro řízení vztahů s dodavateli, hodnocení dodavatelů, správu smluv a nákup. Systémy pro řízení výrobních procesů a zvyšování produkce. Nástroje pro řízení zásob, sledování dodávek a řízení skladů a poskytují platformu pro sdílení informací a spolupráci mezi partnery dodavatelského řetězce. [1, 2]

1.2.1.5 Manufacturing Execution Systems (MES)

Výrobní informační systémy (MES) se zaměřují na řízení a monitorování výrobních procesů v reálném čase, aby zvýšily efektivitu výroby, snížily chyby a zlepšily řízení výrobních zdrojů. Hlavní funkce MES systémů zahrnují plánování a řízení výroby, sledování materiálu a zásob pro efektivní řízení zásob a snižování plýtvání, kontrolu kvality pro zajištění dodržování standardů a výrobních postupů, sledování výkonnosti zařízení a obsluhy pro analýzu a zlepšení efektivity výroby, a dokumentaci a zpracování dat pro ukládání, zpracování a analýzu výrobních dat. [7, 8, 9]

1.2.2 Veřejné informační systémy

Podle anglického názvu (Public Information Systems) se jedná o systémy, které shromažďují, uchovávají a poskytují data pro veřejnost a komunity. Mezi hlavní příklady patří informační systémy používané místními úřady, veřejnými knihovnami a muzei. [10]

1.3 IS pro pekárny na českém trhu

1.3.1 Retail Systems (Evidence pekáren a cukráren)

Retail Systems poskytuje software pro pekárny a cukrárny, který je navržen tak, aby pokrýval široké spektrum operací, které jsou pro tyto podniky důležité. [11]

Hlavní funkce zahrnují:

1. Evidence až 1000 odběratelů, s možností rozdělení podle způsobu platby a přiřazení k různým distribučním okruhům.
2. Práce s více výrobními středisky, s daty tříděnými podle těchto středisek.
3. Definice polotovarů použitých pro výrobu a kalkulaci těchto výrobků.
4. Evidence až 1000 vlastních výrobků s neomezeným počtem prodejních cen přiřazených pro každého odběratele.
5. Kalkulace denního výrobního plánu a výpočet, jaké množství polotovarů je třeba vyrobit.
6. Tisk dodacích listů a dalších dokumentů pro denní plánování výroby.
7. Komplexní skladové hospodářství pro evidenci až 10 000 surovin a výrobků.

Výhody:

1. Software může být použit na neomezeném počtu počítačů v rámci jedné organizace.
2. S nákupem produktu získáváte automaticky emailovou podporu zdarma na neomezenou dobu, s možností zakoupit telefonickou podporu.
3. Program je vyvíjen na moderních vývojových nástrojích firmy Microsoft, což zaručuje vysokou bezpečnost dat

1.3.2 MC Software (Odbyt a výroba pekárny)

MC Software také poskytuje program určený pro pekárny a cukrárny, který pokrývá celou problematiku evidence objednávek, expedice, fakturace, skladového hospodářství, výroby a dopravy.

Program je na trhu již od roku 1990 a od roku 2002 je k dispozici ve verzi pro Windows, což svědčí o jeho dlouhodobé spolehlivosti a vývoji.

Program převzal osvědčené postupy a řešení z původních verzí a přidal další funkce a možnosti, které poskytuje operační systém Windows.

Podrobnější popis jednotlivých funkcionalit:

1.3.2.1 Číselníky

Partneři se dělí na odběratele a dodavatele. Odběratele slouží k vystavování dodacích listů a faktur, zatímco dodavatelé slouží k příjmu surovin do skladu. Karta odběratele se dělí na fakturační a dodací místa. Informace o fakturačním místě pocházejí z obchodního rejstříku (adresa, IČO, DIČ), zatímco informace o dodacím místě určují jednotlivá místa, na která jsou zasílány dodací listy. Systém také podporuje různé způsoby platby, typy slev a nastavení faktur.

Seznamy výrobků obsahují seznam zboží, které se vyrábí. Informace o výrobku jsou rozděleny do dvou karet: prodejní a výrobní. Karta prodeje obsahuje základní informace o výrobku, jako je jeho název, hmotnost, základní velkoobchodní cena bez DPH, sazba DPH a informace o seskupení. Karta výroby obsahuje recepturu a další informace potřebné pro výrobu.

Výpisy surovin obsahují suroviny, které se sledují ve skladových zásobách a používají se ve výrobě. Karta surovin umožňuje zadat popis suroviny, jednotku množství (kusy, kilogramy atd.) a minimální množství na skladě. Cena suroviny použitá pro výpočet nákladů na suroviny u výrobků může být aktuální skladová cena, cena posledního nákupu nebo ručně zadaná cena.

1.3.2.2 Expedice

Příprava pro fakturaci slouží k uzavření dodacích listů v expediční části programu a jejich přesunu k vystavení faktur. Uzavřením dodacích listů pro fakturaci dojde k jejich uzamčení pro editaci v expediční části programu a pro další práci s nimi je nutné přejít do fakturační části programu. Při přesunu dodacích listů do fakturace dojde automaticky k vytvoření faktur za jednotlivé dodací listy a následně lze provádět další práci s fakturami.

V přípravě pro fakturaci lze dodací listy filtrovat podle různých kritérií (např. dle data, rozvozní linky, odběratele) a dále pak lze vybrat jednotlivé dodací listy pro přesun do fakturace. Pro přesun dodacích listů do fakturace je třeba nejprve provést uzavření dodacích listů pro fakturaci, což se provádí zadáním data uzavření (dodací listy s datem dodání do zadaného data uzavření budou přesunuty do fakturace). Po zadání data uzavření je možné provést kontrolu dodacích listů pro fakturaci – tato kontrola upozorní na případné nezaplacené hotově placené dodací listy nebo na případné neuhrazené vratky. Po provedení kontroly a odstranění případných nedostatků je možné provést uzavření dodacích listů pro fakturaci a jejich přesun do fakturace.

Po přesunu dodacích listů do fakturace je možné provést další práci s fakturami ve fakturační části programu - tisk faktur, zaslání faktur e-mailem, export faktur do účetního systému, atd.

1.3.2.3 Fakturace

Hromadná fakturace je proces, při kterém se faktury generují z uzavřených dodacích listů. Zadávají se základní údaje, jako je datum vystavení, datum odeslání, datum zdanitelného plnění, číslo faktury a text záhlaví. Jedním kliknutím se faktury vystaví a lze je hromadně vytisknout, a to i s firemním razítkem, pokud jsou vytištěny na laserové tiskárně. Tyto faktury lze zasílat e-mailem zákazníkům a exportovat do účetního programu, aby se předešlo ručnímu přepisování.

Tiskové sestavy pro fakturaci umožňují shromažďovat a analyzovat údaje o prodeji se zohledněním případných slev uplatněných na fakturované ceny zboží.

1.3.2.4 Výroba

Výrobní příkazy, které obsahují podrobné údaje o množství jednotlivých výrobků, jež mají být vyrobeny, lze zpracovat a vytisknout ze systému. Systém dokáže načítat množství z dodacích listů, spravovat zásoby a upravovat množství vyrobeného zboží. Výrobu lze denně uzavírat, přičemž systém může nastavit úpravu skladových zásob surovin na základě výroby a receptur.

Jako pomůcku při řízení výroby může systém tisknout přehledy výroby a bilance surovin a porovnávat skutečnou a normovanou spotřebu surovin.

Receptury, které jsou klíčové pro výpočet nákladů na výrobek a plánování výroby, lze definovat na základě množství nebo hmotnosti výrobku a mohou zahrnovat suroviny nebo polotovary z příslušných seznamů. V receptuře mohou být uvedeny hlavní suroviny, přičemž ostatní složky jsou vyčísleny v procentech těchto hlavních položek.

1.3.2.5 Sklady

Správa surovin na skladu zahrnuje zaznamenávání příjmů a výdejů, s možností sledování šarží. Příjmy a výdeje mohou pocházet z různých zdrojů a jsou evidovány v archivu skladových dokladů.

Stav skladu ukazuje aktuální množství surovin, s červeným zvýrazněním pro položky pod minimem.

Inventura slouží k zjištění rozdílu mezi evidovaným a skutečným stavem skladu, který je zaznamenán jako výdej do výroby. [12]

1.3.3 Software PORS

Společnost PORS software a.s. byla založena na začátku roku 1995 jako dceřiná společnost firmy PORS a.s. Je viděna jako pokračovatel tradice bývalého státního podniku Početnická a organizační služba Praha. Její hlavní oblastí působení je poskytování softwarových a IT služeb, včetně ekonomického software, maloobchodního pokladního systému, dodávek počítačů HP, počítačových sítí, servisních služeb, školení a účetních služeb. Jejím hlavním cílem je poskytnout fungující podnikovou agendu s kvalifikovaným personálem a dostatkem dostupných informací pro sledování a řízení vaší firmy. [35]

Společnost PORS poskytuje dva moduly, které jsou relevantní pro pekárny.

1. **modul Přímé dodávky zboží:** Tenhle modul se zaměřuje na sortimentní evidenci, účtování, dodavatelské slevy a zpracování elektronické fakturace. [13]
2. **modul Objednávky prodejen:** Tento modul se zabývá pořízením objednávek prodejen do velkoobchodu a zpracováním těchto objednávek. [14]

1.3.4 COMPEX spol. s r.o.

COMPEX, spol. s r.o. je česká firma, která působí v oboru vývoje ekonomického software, účetního poradenství a související obchodní činnosti. Firma byla založena 14. listopadu 1990 a v současné době má více než 30 let zkušeností v tuzemském podnikatelském prostředí. COMPEX je vývojářem vlastního aplikačního ERM software, který se zaměřuje na ekonomiku a účetnictví. Původně dodávali ekonomický systém FIS 2000 (verze pro DOS), který byl pravidelně aktualizován po dobu 10 let od roku 1991. V roce 2000 převzal nástupnictví ekonomický software FIS 3000, vyvinutý jako základní aplikace na platformě Windows.

FIS 3000 je modulární systém a je nabízen ve verzích pro podnikatele, příspěvkové i neziskové organizace. Software je propracovaný, ale zároveň snadno ovladatelný a snadno pochopitelný. Program je neustále aktualizován a vývojáři poskytují klientům bezplatnou poradenskou službu a pravidelně pořádají uživatelská školení. [15]

Seznam základních modulů:

1. Základní účetnictví
2. Faktury
3. Evidence majetku
4. Sklady
5. Objednávky
6. Nedokončená výroba
7. Manažer

1.3.5 Shrnutí

Z analýzy stávajících informačních systémů pro pekárny vyplývá, že tyto systémy neposkytují přímou podporu pro správu receptů. Na druhé straně excelují v jiných kritických oblastech podnikání, jako je řízení odběratelů, výrobních středisek, skladů a objednávek.

Proto se tato práce bude zaměřovat na navržení informačního systému, který bude sestaven tak, aby vyhovoval požadavkům spojených s řízením receptů a tím zaplnil existující mezeru na trhu.

2 UML

Unified Modelling Language (UML) je vizuální jazyk používaný k popisu a návrhu softwarových systémů, zejména objektově orientovaných systémů. UML usnadňuje účinnější a efektivnější návrhový dialog tím, že poskytuje vyšší úroveň abstrakce než běžné programovací jazyky. UML se skládá ze souboru grafických notací a používá se ke specifikaci, reprezentaci, konstrukci a dokumentaci komponent softwarového systému.

UML je otevřený standard, na který dohlíží skupina Object Management Group (OMG), a je kombinací několika objektově orientovaných grafických modelovacích jazyků populárních na konci 80. a začátku 90. let. UML byl vyvinut v roce 1997 a přinesl do této oblasti tolik potřebnou konzistenci. [17, 18]

Ačkoli je UML často spojován s objektově orientovanými softwarovými systémy, jeho rozšiřitelnost a přizpůsobivost vedly k tomu, že se používá i mimo ně. UML je navržen tak, aby se používal s modelovacími nástroji UML k vytváření diagramů, které jsou srozumitelné jak lidem, tak počítačům. Důležité je, že UML nepodporuje žádný modelovací přístup. Místo toho poskytuje vizuální jazyk pro vytváření modelů a lze jej použít se všemi existujícími metodikami. [17, 18]

UML zachycuje statickou strukturu a dynamické chování systému a poskytuje vícerozměrné porozumění systému. Jako obecný a komplexní modelovací jazyk pro diskrétní systémy není UML programovací jazyk, ale je podporován nástroji, které umožňují generování kódu a sestav. [17, 18]

2.1 Význam slova „Unifikovaný“

Pojem „Unified“ v kontextu Unified Modeling Language (UML) znamená širokou použitelnost a konzistenci napříč oblastmi:

1. **Metody a notace:** UML kombinuje široce přijímané pojmy z mnoha objektově orientovaných metod a poskytuje jasné definice, notace a terminologii UML vychází z historických metod a notací. Může přesně reprezentovat většinu současných modelů.
2. **Životní cyklus vývoje:** UML je bezproblémový a lze jej konzistentně aplikovat od fáze požadavků až po nasazení. Není třeba překládat, když se mezi jednotlivými fázemi používají stejné myšlenky a notace. To je velmi důležité pro inkrementální a iterativní vývoj.

3. **Oblasti použití:** UML může modelovat širokou škálu aplikačních oblastí, včetně velkých, komplexních systémů, systémů reálného času, distribuovaných systémů atd. Přestože UML má za cíl být všudypřítomným modelovacím jazykem, existují niky, kde jsou specializované jazyky výhodnější.
4. **Platformní a aplikační jazyky:** Implementace se může lišit, ale návrhová práce je ve všech prostředích stejná.
5. **Vývojové procesy:** UML je modelovací jazyk, nikoli konkrétní vývojový proces. Může tvořit základ většiny nových i stávajících vývojových procesů, stejně jako univerzální programovací jazyk může vyhovovat různým programovým paradigmatům.
6. **Vnitřní koncepce:** UML se pokouší spojit širokou škálu modelovacích konceptů, které lze použít v různých kontextech, což umožňuje hlubší pochopení a širší použití těchto konceptů. [17]

2.2 Použití UML

UML lze použít třemi způsoby, a to jako náčrt, plán nebo programovací jazyk.

1. **Kreslení konceptu:** Toto použití je nejběžnější: Vývojáři používají diagramy UML k popisu určitých aspektů systému. Aspekt dopředného inženýrství (diagramy UML jsou vytvořeny před napsáním kódu) nebo aspekt zpětného inženýrství (diagramy UML jsou vytvořeny z dříve napsaného kódu pro lepší pochopení). Pro usnadnění konverzace a porozumění se vytvářejí výběrové a neformální náčrty, které se zaměřují na klíčové součásti systému. [18]
2. **Kreslení detailních návrhů:** Na rozdíl od náčrtů se UML používá jako plán pro úplnost. Zde se vytváří komplexní návrh, který by měl obsahovat všechny podrobnosti, které programátor potřebuje k napsání kódu. Cílem je vytvořit komplexní model, který z programování udělá jednoduchý úkol. Systém může být plně nebo částečně popsán pomocí modelu na úrovni plánu. Plány poskytují podrobnou grafickou reprezentaci pro zpětné inženýrství kódu. [18]
3. **UML jako programovací jazyk:** V této aplikaci vývojář vytváří diagramy UML, které se přímo převádějí do spustitelného kódu. To vyžaduje specifické nástroje a velmi přesné vyjádření diagramů UML. V této souvislosti se často používá výraz Model Driven Architecture (MDA), další standard Object Management Group (OMG), který se snaží standardizovat použití UML jako programovacího jazyka. [18]

2.3 Objekty v jazyce UML

Základním předpokladem jazyka UML je, že software a další systémy lze reprezentovat jako síť vzájemně propojených objektů. Je plně propojen s objektově orientovanými softwarovými systémy a jazyky a lze jej aplikovat i na další aplikace, například obchodní procesy.

Model UML se skládá ze dvou aspektů:

1. **Statická struktura:** typy objektů a vazby mezi objekty důležité pro modelování systému jsou shrnuty ve statické struktuře.
2. **Dynamické chování:** dynamické chování popisuje, jak spolu objekty během své životnosti interagují, aby vytvořily požadovanou funkčnost systému.

Tyto dva aspekty modelu UML jsou vzájemně propojeny a bez jednoho nebo druhého nejsou úplné. [17]

2.4 Stavební bloky jazyka UML

UML se skládá ze tří základních stavebních bloků

1. **Věci:** představují vlastní modelovací materiály
2. **Vztahy:** Vztahy nebo také relace pomáhají „věci“ vzájemně propojit, Vztah tvoří konceptuální spojení mezi dvěma nebo více „věcmi“
3. **Diagramy:** představují soubor „věcí“, které reprezentují funkčnost softwarového systému a popisují, co bude systém realizovat (diagram na úrovni analýzy) nebo jak bude realizovat (diagram na úrovni návrhu) [17]

2.5 Věci

Věci nebo také prvky lze rozdělit v UML následovně:

1. **Strukturální prvky:** Jsou známy také jako „jména“ modelu UML, zahrnují uzly, třídy, rozhraní, spolupráce, případy užití, aktivní třídy, komponenty a uzly
2. **Prvky chování:** Tyhle prvky představují stavové stroje, interakce a aktivity, které fungují jako „slovesa“ modelu UML
3. **Seskupovací prvky:** Představují balíčky pro seskupování souvisejících modelovacích prvků do logicky konzistentních celků
4. **Anotační prvky:** Tyhle prvky představují poznámky, které lze připojit k modelu, například žluté lepicí papírky, pro zaznamenání improvizovaných informací [17]

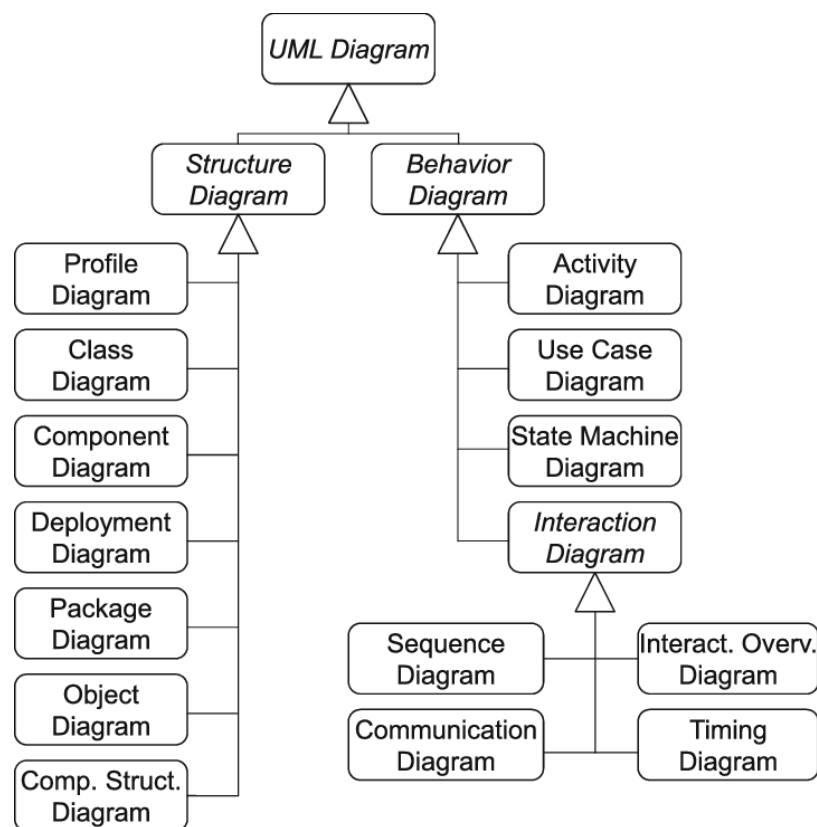
2.6 Vztahy (relace)

V modelech UML mohou vztahy ukazovat, jak spolu různé prvky souvisejí. Dobrým příkladem toho, jak jednotlivci vzájemně souvisí, je vztah v rodokmenu. Vztahy v modelech UML pomáhají pochopit důležité (sémantické) vazby mezi prvky. Mezi vztahy patří:

- závislost
- asociace
- agregace
- kompozice
- ochranná nádoba
- zobecnění
- realizace

2.7 Diagramy

Diagramy UML se používají k znázornění různých vlastností softwarového systému a pomáhají vývojářům pochopit jeho architekturu. V jazyce UML existuje mnoho druhů diagramů, z nichž každý se používá k popisu určité součásti softwarového systému.



Obrázek 1. Přehled všech UML diagramů [16]

2.7.1 Diagramy chování

Popisují dynamickou strukturu systému. Zobrazují, jak budou předměty interagovat, aby dosáhli požadovaného chování systému. [23]

2.7.1.1 Diagram případu užití

Diagram případu užití je grafický nástroj v rámci UML, který zachycuje interakce mezi uživateli a systémem. Slouží k popisu funkčních požadavků systému a zobrazuje, jak různí aktéři využívají systém k dosažení cílů. Jsou užitečným nástrojem pro analýzu a návrh systému z perspektivy uživatele. [26, 28]

Podle [26, 27] diagram případů užití se skládá z následujících prvků:

1. Aktéři

Aktér je vnější entita, která interaguje se systémem. Aktéři mohou být uživatelé, organizace nebo jiný systém. Aktéři mohou být jak uvnitř, tak mimo systém. Jsou zobrazeni jako postavičky nebo obdélníky s názvem uvnitř.

2. Případy užití

Případ užití reprezentuje konkrétní funkci nebo činnost, kterou systém poskytuje aktérům. Každý případ užití je sekvence akcí, která poskytuje měřitelnou hodnotu aktérovi a ukončuje se dosažením nějakého cíle. Případy užití jsou zobrazeny jako elipsy s názvem případu užití uvnitř. Případy užití mohou být spojeny s jedním nebo více aktéry.

3. Hranice

Hranice (anglicky 'boundary') slouží k vymezení hranic a rozsahu systému. Je reprezentována obdélníkem, který odděluje aktéry a případy užití, které patří do systému, od těch, které nepatří. Pomáhá organizovat a strukturovat případy užití a zlepšuje přehlednost diagramu.

4. Vztahy

Vztahy v diagramu případu užití vyjadřují interakce a vztahy mezi aktéry a případy užití.

1. **Asociace:** Jedná se o základní vztah, který je zobrazen jako čára mezi aktérem a případem užití. Tento vztah ukazuje, že aktér je zapojen do daného případu užití.

2. **Zahrnutí:** Zahrnutí (anglicky 'include') je vztah, kde jeden případ užití (zvaný základní) využívá funkcionalitu nebo chování jiného případu užití (zvaného zahrnutý). Zahrnutí se zobrazuje šipkou s nápisem '<<include>>' směřující od základního případu užití k zahrnutému.
3. **Rozšíření:** Rozšíření (anglicky 'extend') je vztah, kde jeden případ užití (zvaný rozšiřující) může doplnit chování jiného případu užití (zvaného rozšiřovaný) za určitých podmínek. Rozšíření se zobrazuje šipkou s nápisem '<<extend>>' směřující od rozšiřujícího případu užití k rozšiřovanému.
4. **Generalizace:** Generalizace je vztah, kde jeden aktér nebo případ užití dědí vlastnosti a chování jiného aktéra nebo případu užití. Tento vztah je zobrazen šipkou s prázdným hrotem směřující od odvozeného (specifičtějšího) elementu k základnímu (obecnějšímu) elementu.

2.7.1.2 Sekvenční diagram

Diagram sekvencí je typem interakčního diagramu, který se zaměřuje na výměnu zpráv mezi životními liniemi. Popisuje interakci tím, že zobrazuje sekvenci zpráv a jejich odpovídající výskyty na životních liniích. Diagram reprezentuje tok komunikace a pořadí operací v čase.

Hlavními komponenty, které obvykle najdeme v diagramu sekvencí UML, jsou životní linie, objekty v systému, aktivační boxy, zprávy a návratové zprávy, fragmenty.

Diagramy sekvencí ilustrují, jak jsou operace prováděny, a zobrazují zasílané zprávy a jejich časování. Poskytují strukturované zobrazení chování, které ukazuje pracovní postup, předávání zpráv a spolupráci mezi prvky pro dosažení požadovaného výsledku. Tyto diagramy zachycují tok informací a zodpovědností v rámci systému a jsou cenným nástrojem pro analýzu a modelování scénářů užití. Mohou také sloužit k vysvětlení sekvence kroků, které uživatelé a systém podnikají k dokončení konkrétních úkolů.

Závěrem lze říci, že diagramy sekvencí jsou důležité pro vizualizaci a porozumění komunikaci a chování prvků v systému, jejich interakcím v čase a toku informací mezi nimi. [24, 25]

Stručnější popis jednotlivých komponentů jsou čerpány z [24, 25]:

1. Objekty

Jsou zobrazeny jako obdélníky, které představují konkrétní nebo obecné entity v systému na vrcholu diagramu.

2. Aktéři

jsou typy objektu, které reprezentují role uživatelů v systému. Jsou to entity, které mohou komunikovat se systémem.

3. Životní linie

Vertikální přerušovaná čára, která se táhne od objektu dolů. Reprezentuje životnost objektu během interakce. Pomáhají vizuálně zobrazit tok informací a řízení v systému a jsou užitečné pro analýzu, návrh a dokumentaci.

4. Lifeline

Jsou to nezávislé připojitelné prvky a jednotliví účastníci interakce.

5. Aktivační box

Jedná se o tenký obdélník ležící na životní linii. Představuje čas, během kterého prvek provádí operaci. Horní a dolní část jsou zarovnané s časem zahájení a ukončení operace.

6. Zprávy

Jsou to interakce nebo komunikace mezi objekty jsou znázorněny šipkami mezi životními liniemi objektů. Šipky směřují od odesílatele k příjemci.

7. Návratové zprávy

Jedná se o odpovědi na zprávy. Jsou také zobrazeny šipkami, které mají přerušovanou čáru. Šipky jdou směrem zpět od příjemce k odesílateli.

8. Vlastní zpráva

Reprezentuje specifickou komunikaci mezi objekty. Je vyjádřena šipkou, která směřuje k téže linii života.

9. Fragment

Fragment sekvence usnadňuje tvorbu a údržbu přesných sekvenčních diagramů. Je reprezentován jako box, nazývaný kombinovaný fragment, který uzavírá části interakce v sekvenčním diagramu. Prvky fragmentu mohou v sekvenčním diagramu představovat iterativní nebo střídavé procesy. Obsahuje operátor, který označuje typ fragmentu. Typy fragmentů:

1. **Alt (alternativa):** Reprezentuje situaci s podmíněným chováním. Skládá se z podmínkového výrazu a větví, z nichž každá je spojena s konkrétní podmínkou. Pouze jedna větev je vykonána na základě splnění podmínky.

2. **Opt (Volitelné):** Popisuje chování, které může být provedeno nebo přeskočeno během vykonávání. Podmínkový výraz určuje, zda se dané volitelné chování provede.
3. **Loop (Smyčka):** Zobrazuje opakující se chování. Fragment smyčky obsahuje ochrannou podmínku, která určuje, zda se smyčka vykonává nebo ukončí. Podmínka může být jednoduchý nebo složitější výraz.

2.7.2 Diagramy struktury

Používají se k zobrazení toho, jak je systém sestaven. Zobrazují předměty a vztahy takovým způsobem, aby bylo vidět, co systém dělá. [23]

2.7.2.1 Diagram tříd

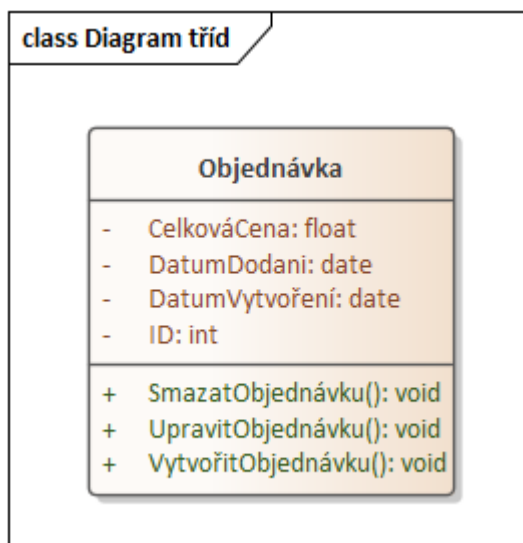
Diagramy tříd jsou základními diagramy v jazyce UML, které slouží k zachycení statických vztahů v softwaru a popisu, jak jsou věci strukturovány. Pomáhají při návrhu softwaru a rozhodování o vztazích mezi třídami, jako jsou odkazy mezi třídami nebo vlastnictví jedné třídy jinou. Tyto diagramy zobrazují fyzickou strukturu systému. [21]

Při vytváření objektově orientovaného modelu systému se diagramy tříd používají k zobrazení tříd a asociací mezi nimi. Třidu objektu lze chápat jako obecnou definici jednoho druhu objektu v systému. Asociace je propojení mezi třídami, které ukazuje existenci vztahu mezi nimi. Každá třída může mít určité znalosti o přidružených třídách. Při vývoji modelů v raných fázích softwarového inženýrství představují objekty reálného světa, například zákazník, recept nebo produkt. Při implementaci je pak nutné definovat další objekty pro zajištění požadované funkčnosti systému. Diagramy tříd se zaměřují na modelování objektů reálného světa v rámci tvorby požadavků a raného návrhu softwaru. [19, 20]

Třída v diagramu tříd symbolizuje specifické typy objektů, které mohou existovat v systému. Určuje také jejich vlastnosti a funkce, které tyto objekty mohou vykonávat. [21]

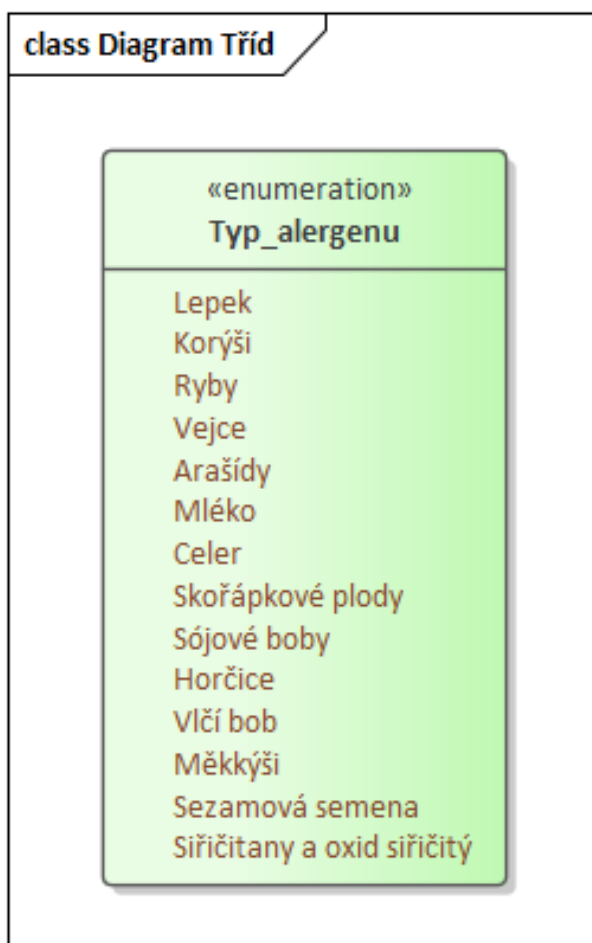
Podle [21] třída je rozdělena do tří částí:

1. **První část:** Obsahuje název třídy. Tato část je vždy nutná.
2. **Střední část:** Zde jsou vlastnosti třídy. Používá se k popisu vlastností třídy. Tato část je vyžadována pouze při popisu konkrétní instance třídy.
3. **Poslední část:** Obsahuje metody třídy. Metody popisují, jak třída interaguje s daty.



Obrázek 2. Ukázka třídy

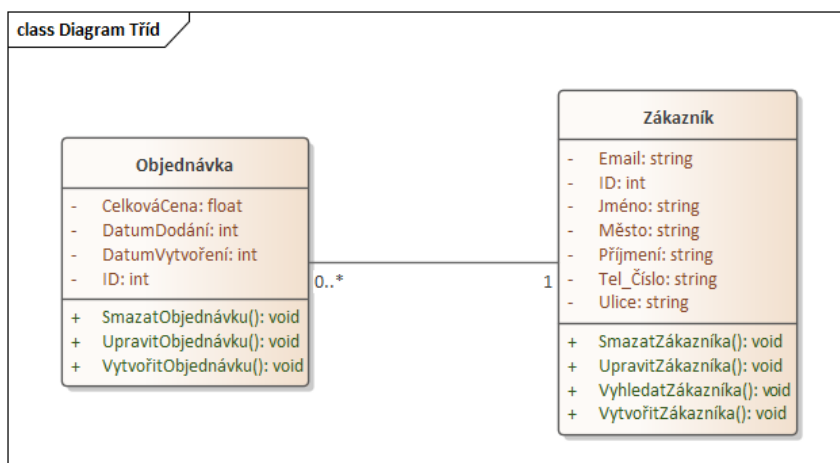
V diagramu tříd lze reprezentovat výčty (enumeration), které definují omezenou množinu konstantních hodnot, kterých může atribut nabývat. Výčty jsou zobrazeny jako oddělené boxy s názvem výčtu uvnitř. Výčty pomáhají přesněji specifikovat možné hodnoty atributů.



Obrázek 3. Ukázka enumeration

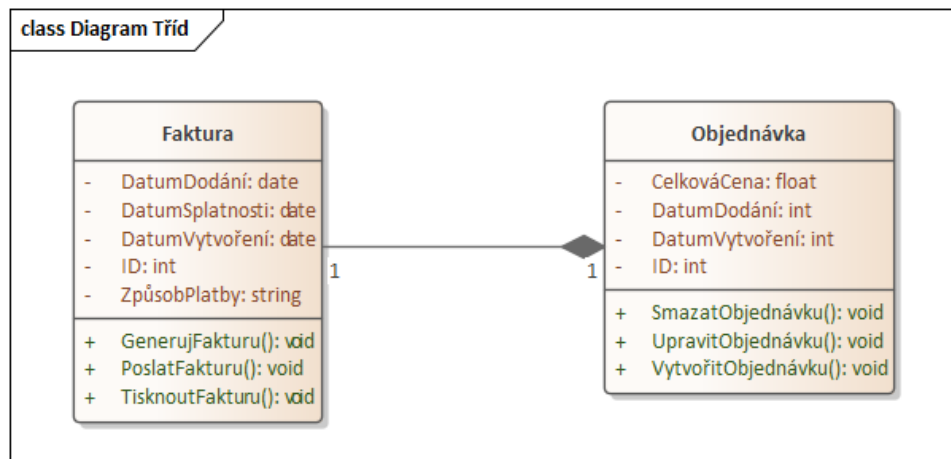
V diagramu tříd jsou znázorněny vztahy asociace a závislosti mezi třídami. Tyto vztahy slouží k ilustraci interakce a spolupráce mezi třídami v rámci systému. Vztahy mohou být jedním z následujících typů:

1. **Asociace:** Asociace je vztah mezi entitami, tj. spojuje objekty jedné entity s objekty jiné entity. Asociaci mezi dvěma třídami lze použít k přechodu od objektů jedné třídy k objektům druhé třídy a naopak. Je zcela přípustné, aby oba konce asociace ukazovaly na stejnou třídu, a na základě objektů stejné třídy se můžete odkazovat na jiné objekty stejné třídy. Asociace, která spojuje přesně dvě třídy, se nazývá binární asociace. Asociace, které spojují více než dvě třídy, jsou méně časté a nazývají se n-ární asociace. V diagramech jsou asociace graficky znázorněny pomocí čar spojujících stejné nebo různé třídy. Asociace se používají k zobrazení strukturních vztahů mezi entitami. [21]



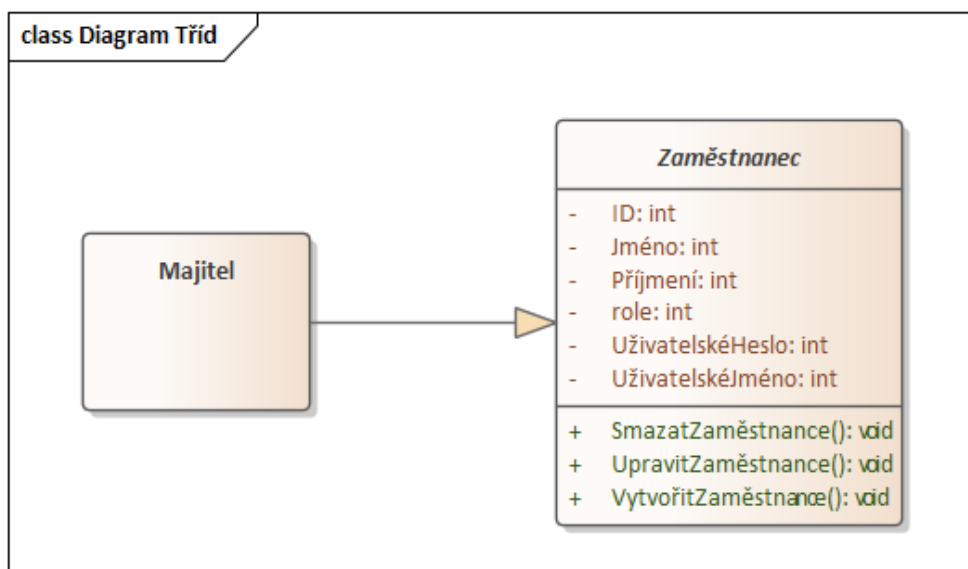
Obrázek 4. Ukázka asociace

2. **Agregace:** Asociace mezi dvěma třídami představuje strukturní vztah mezi rovnocennými prvky. V některých případech však může být žádoucí modelovat vztah "celek/část", kdy třída představuje větší celek a obsahuje menší části. Tento vztah se nazývá agregace a je to zvláštní typ vztahu reprezentovaný otevřeným kosočtvercem směřující k cílové nebo nadřazené třídě. [21]
3. **Kompozice:** Silnější agregací je kompozitní nebo složená agregace, která je reprezentovaná černým kosočtvercem směřující k cílové třídě. Používá se v případě, že složka může současně patřit nejvýše do jedné kompozice. Při odstranění hlavní části kompozice se odstraní i všechny její části. Z kompozice je však možné vymazat jednotlivé části, aniž by se vymazala celá kompozice. Kompozice mohou být tranzitivní, asymetrické vztahy a rekurzivní. [21]



Obrázek 5. Ukázka kompozice

4. **Generalizace:** Generalizace se používá k vyjádření dědičnosti. V diagramu je generalizace znázorněna šipkami vedoucími od konkrétního klasifikátoru k obecnějšímu klasifikátoru, které ukazují, že zdroj dědí vlastnosti cíle. [21]



Obrázek 6. Ukázka generalizace

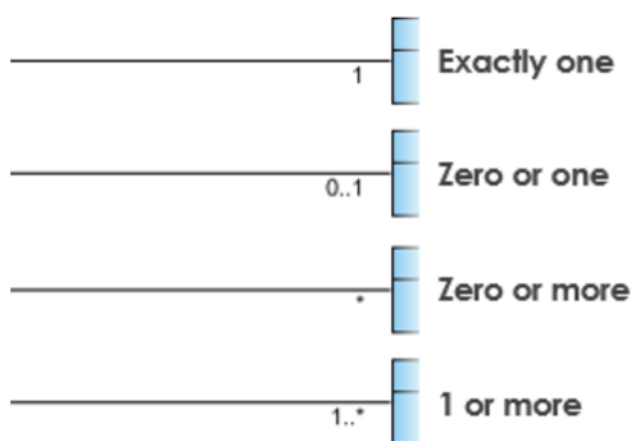
5. **Závislost:** Závislost je vztah mezi prvky, který vyjadřuje, jak změna specifikace jednoho prvku ovlivní jiný prvek, který jej používá. Schematicky je závislost znázorněna přerušovanou směřovanou čarou směřující k závislému prvku. Používá se k modelování závislostí mezi prvky v jazyce UML, například mezi třídami. Závislosti se používají k zobrazení toho, že jedna věc používá jinou. Například jedna třída používá jinou třídu jako argument pro operaci. Změna závislého prvku může ovlivnit chování použitého prvku nebo rozhraní. Závislosti lze použít mezi různými prvky, jako jsou třídy, rozhraní, balíčky a anotace. [21]

- 6. Realizace:** Realizace je vztah mezi prvky modelu, v němž jeden prvek implementuje specifikaci jiného prvku. Tento vztah se používá k vyjádření návaznosti, úplnosti a koordinace mezi specifikací a realizací. Schematicky je realizace znázorněna přerušovanou směrovou čarou a otevřeným hrotem šipky. Realizace se používají především mezi rozhraními a třídami nebo komponentami, kde rozhraní definuje smlouvu, kterou má třída nebo komponenta splnit. Realizace umožňují oddělení specifikace a implementace a také spojení mezi různými úrovněmi abstrakce v modelu systému. [21]

Násobnost

Asociace představuje strukturální vztah mezi objekty. V mnoha případech je důležité určit, kolik objektů může být prostřednictvím dané asociace propojeno. Tato informace se nazývá násobnost asociačních rolí a je reprezentována rozsahem hodnot nebo výrazem určujícím konkrétní hodnotu. Zadání násobnosti na jednom konci asociace nám říká, kolik objektů třídy na jedné straně musí mít odpovídající počet objektů třídy na straně druhé. Násobnost lze vyjádřit jako: [21]

- právě jeden (1)
- nula nebo jedna (0..1)
- Nula nebo víc (*)
- jeden nebo více (1..*)



Obrázek 7. Ukázka násobnosti [21]

3 ENTITNĚ – RELAČNÍ DIAGRAM

ERD graficky znázorňuje ER model. Jedná se o vývojové diagramy používané v softwarovém inženýrství, podnikových informačních systémech, vzdělávání a výzkumu pro návrh a ladění relačních databází. Tyto diagramy ukazují, jak spolu různé součásti systému souvisejí – lidé, objekty nebo koncepty. Pro znázornění objektů entit, vztahů a atributů se v ERD používají specifické symboly. Tyto diagramy slouží k identifikaci a zkoumání vztahů mezi entitami a také k vizualizaci struktury systému. Hlavním cílem ERD je nabídnout přehledný a srozumitelný pohled na objekty entit a jejich vazby, což usnadňuje návrh a pochopení databázových systémů. [29, 30]

Podle [29, 30] mezi základní komponenty ERD patří:

1. **Entita:** Entita je reprezentace objektu, který má v databázi význam. Může to být například osoba, věc nebo událost. V diagramu se entita zobrazuje jako obdélník.
2. **Atribut:** Atribut je charakteristika nebo vlastnost entity. Atributy mohou být například jméno, adresa nebo datum narození pro entitu "osoba". Atributy se v diagramu zobrazují jako kruhy nebo ovály, které jsou spojené s entitou.
3. **Vztah:** Vztah je asociace mezi dvěma nebo více entitami. Například, vztah mezi entitou "osoba" a "auto" může být "vlastní". Vztahy se v diagramu zobrazují jako diamanty nebo čáry spojující entity.
4. **Primární klíč:** Primární klíč je unikátní identifikátor entity. Žádné dvě entity nemohou mít stejný primární klíč. V diagramu se primární klíč zobrazuje podtržením atributu, který je primárním klíčem.
5. **Cizí klíč:** Cizí klíč je atribut v jedné entitě, který odkazuje na primární klíč jiné entity. Slouží k propojení entit. V diagramu se cizí klíč nezobrazuje explicitně, ale je implikován vztahem mezi entitami.
6. **Kardinalita:** Kardinalita označuje počet entit v jednom vztahu. Například, jedna osoba může vlastnit více aut, ale jedno auto může vlastnit jen jedna osoba. Kardinalita se v diagramu zobrazuje u vztahu a udává, kolik entit může být v daném vztahu.

Typy kardinalit:

1. **Kardinalita 1:1:** Tento typ kardinality znamená, že jedna instance jedné entity je spojena právě s jednou instancí jiné entity. Například, jeden produkt může mít pouze jeden recept a každý recept je přiřazen pouze jednomu produktu.

2. **Kardinalita 1:N:** Kardinalita 1:N znamená, že jedna instance jedné entity je spojena s nulou nebo více instancemi jiné entity. Například, jeden zákazník může mít více objednávek, ale každá objednávka je přiřazena pouze jednomu zákazníkovi.
3. **Kardinalita N:1:** Kardinalita N:1 je opakem kardinality 1:N. Znamená to, že více instancí jedné entity je spojeno s jednou instancí jiné entity. Například, více zaměstnanců může být přiřazeno k jedné pozici v rámci firmy.
4. **Kardinalita N:M:** Kardinalita N:M (mnoho k mnoha) znamená, že více instancí jedné entity je spojeno s více instancemi jiné entity. Tento typ kardinality se obvykle řeší pomocí prostředníka nebo pomocné tabulky. Například, vztah mezi "Produkt" a "Objednávka" může být mnoho k mnoha, protože jeden produkt může být ve více objednávkách a jedna objednávka může mít více produktů.

4 POŽADAVKY

Požadavky na informační systém jsou specifické instrukce, které určují funkcionality, vlastnosti a omezení systému. Tyto požadavky jsou klíčové pro design a správu systému. [31, 32]

4.1 Funkční požadavky

Jedná se o specifikace, které popisují očekávanou funkčnost a chování systému. Uvádějí činnosti, které musí systém provádět, povinnosti, které musí plnit a výsledky, které musí produkovat. Popisují akce, které může uživatel v systému provádět, a způsob, jakým systém na tyto akce reaguje. Funkční požadavky by měli být napsány co nejpřesněji. Jsou založeny na potřebách uživatelů nebo požadavcích zákazníka. [31, 32]

4.2 Nefunkční požadavky

Tyto požadavky definují kvalitativní aspekty, výkonnostní parametry nebo omezení systému. Nefunkční požadavky nezahrnují funkce systému, ale specifikují jeho vlastnosti a omezení, jako je bezpečnost, spolehlivost, efektivita a škálovatelnost. [31, 32]

5 WIREFRAME

Drátěný model je zkrácený náčrt nebo vizualizace rozložení a umístění jednotlivých prvků v rozhraní aplikace informačního systému. Jedná se o základní návrhový nástroj používaný při tvorbě webových stránek, informačních systémů nebo mobilních aplikací.

Drátěný model lze vytvořit buď ručně na papíře, nebo pomocí specializovaného softwaru pro návrh rozhraní. Zobrazení je obvykle černobílé a soustředí se pouze na organizaci a uspořádání prvků. Postrádá obrázky, barvy, typografii a složitou stylizaci. [33, 34]

Na vytvoření wireframe byla využita Figma, která je renomovaným nástrojem pro tvorbu uživatelských rozhraní a uživatelské zkušenosti. Díky své intuitivnosti a efektivitě umožňuje Figma navrhování, prototypování a sdílení projektů z jednoho místa. Figma je založena na cloudu a poskytuje možnost snadného sdílení a spolupráce na projektech.

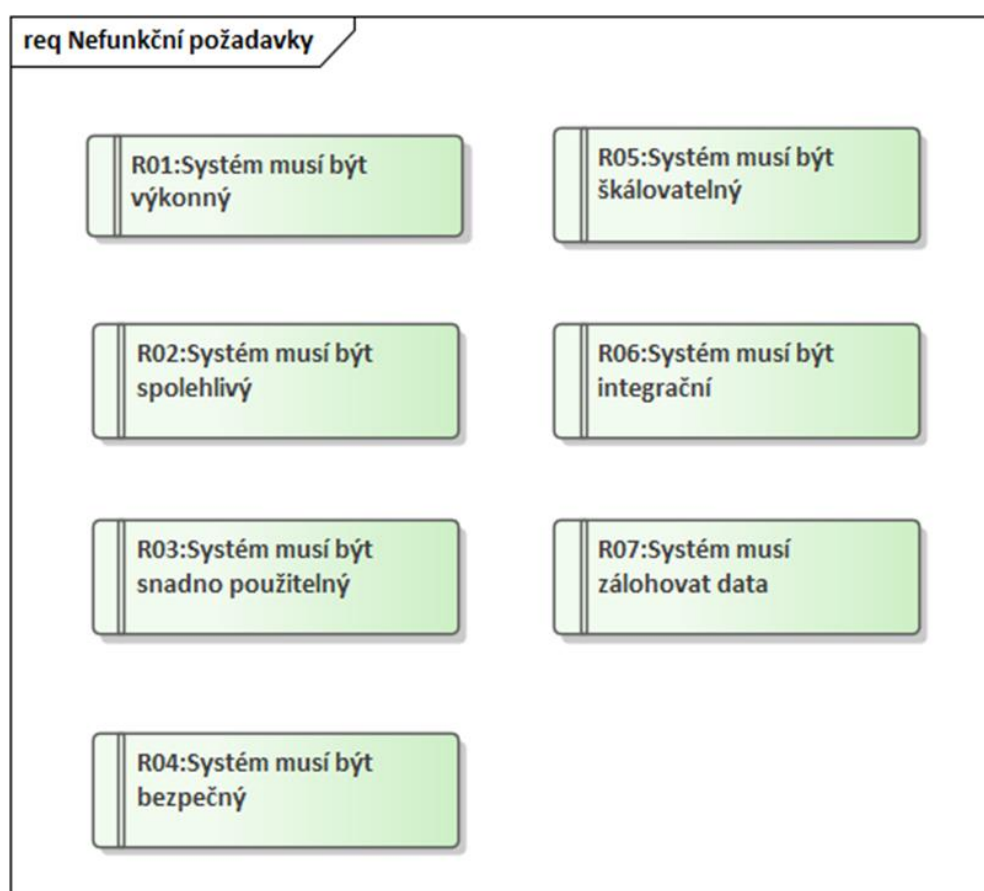
II. PRAKTICKÁ ČÁST

6 NÁVRH A SPECIFIKACE POŽADAVKŮ

Kapitola je zaměřena na klíčovou fázi vývoje informačního systému pro hospodářskou výrobu – návrh a specifikace požadavků. Tato fáze je nezbytná pro přesné a důkladné definování požadavků systému, které budou sloužit jako vodítko pro jeho následující vývoj. Navržené a popsané funkční a nefunkční požadavky byly vytvořeny pomocí jazyka UML.

6.1 Nefunkční požadavky

V této podkapitole jsou úspěšně navrženy a specifikovány nefunkční požadavky pro informační systém. Definování těchto požadavků umožňuje zajištění spolehlivého, výkonného, bezpečného a snadno ovladatelného systému, který bude schopen přizpůsobit se budoucím změnám. Tyto nefunkční požadavky představují důležitou součást celkového návrhu a jsou klíčové pro úspěšné nasazení a provoz informačního systému.

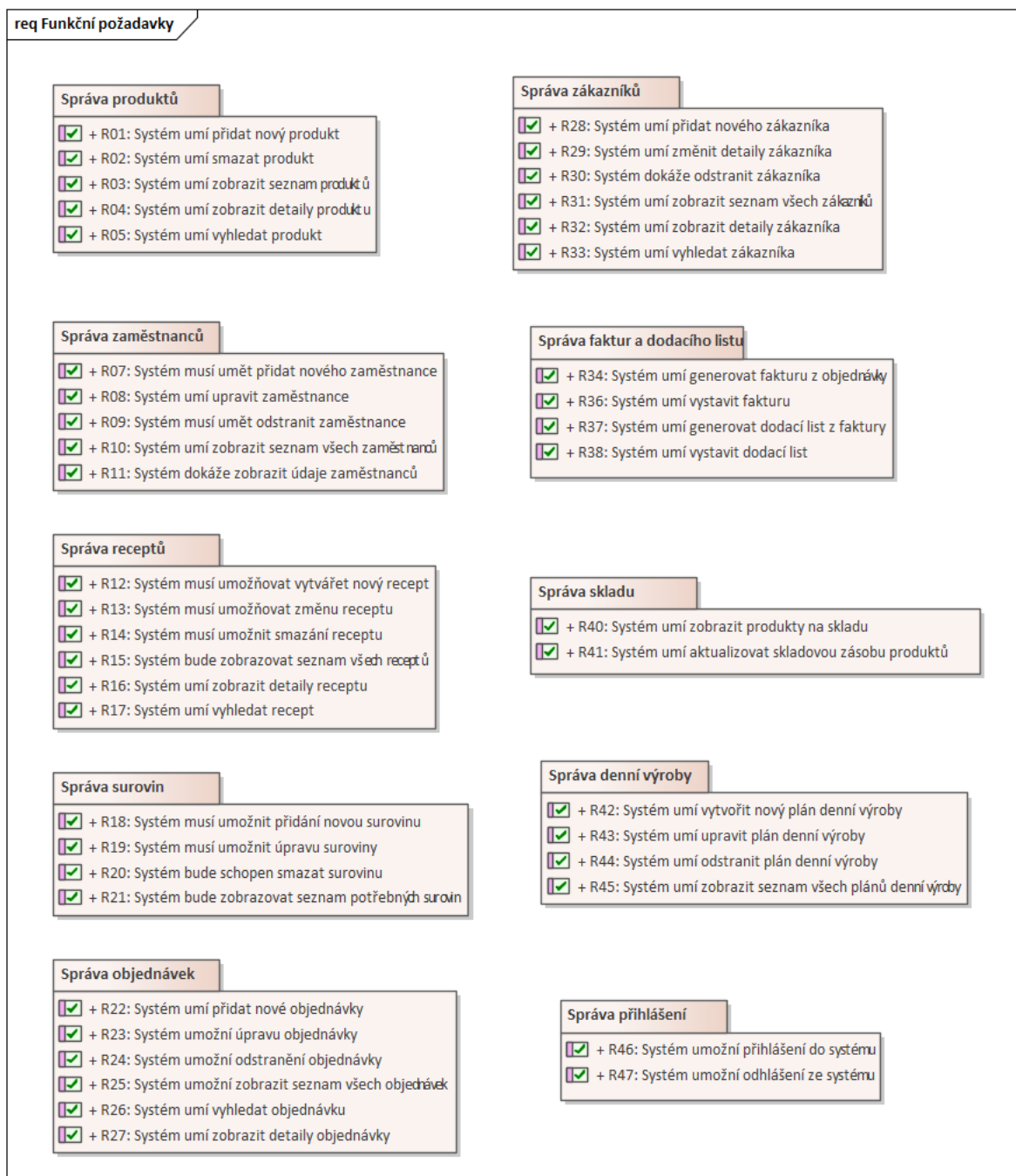


Obrázek 8. Navrhnuté nefunkční požadavky

- **R01: Systém musí být výkonný** – Systém musí být výkonný. To znamená, že musí být schopen rychle a efektivně zpracovávat požadavky a úlohy, které mu jsou přiděleny. To zahrnuje schopnost zvládnout velký objem dat a operací bez výrazného omezení rychlosti nebo kvality výstupu.
- **R02: Systém musí být spolehlivý** – Systém musí být spolehlivý. Tento požadavek se týká schopnosti systému fungovat bez poruch a selhání, a také schopnosti rychle se zotavit po případných chybách nebo výpadech.
- **R03: Systém musí být snadno použitelný** – Systém musí být snadno použitelný. Uživatelé by měli být schopni intuitivně navigovat systémem a využívat jeho funkce bez potřeby složitého školení nebo technických znalostí.
- **R04: Systém musí být bezpečný** – Systém musí být bezpečný. To znamená, že musí být schopen chránit citlivé informace a data uživatelů před neoprávněným přístupem, útoky nebo úniky.
- **R05: Systém musí být škálovatelný** – Systém musí být škálovatelný. To znamená, že by měl být navržen tak, aby mohl snadno zvládat růst v počtu uživatelů, dat nebo požadavků na zpracování bez potřeby výrazných změn v jeho architektuře nebo designu.
- **R06: Systém musí být integrační** – Systém musí být integrační. Musí být schopen snadno se integrovat s ostatními systémy a platformami, aby byla zajištěna jeho kompatibilita a interoperabilita.
- **R07: Systém musí zálohovat data** – Systém musí zálohovat data. To znamená, že musí existovat mechanismus pro pravidelné a spolehlivé zálohování dat, aby byla zajištěna jejich bezpečnost a dostupnost i v případě potenciálních chyb nebo selhání.

6.2 Funkční požadavky

Tato podkapitola je zaměřena na funkční požadavky spojené s různými aspekty správy informačního systému. Zde jsou specifikovány klíčové funkcionality, které systém musí poskytovat pro správu produktů, zaměstnanců, receptů, surovin, objednávek, zákazníků, faktur, dodacích listů, skladových zásob a plánů denní výroby. Cílem této podkapitoly je detailně popsat požadované funkční schopnosti a poskytnout směrnice pro návrh a jeho následnou implementaci.



Obrázek 8. Navrhnuté funkční požadavky

V této podkapitole jsou podrobně specifikovány funkční požadavky týkající se správy různých aspektů informačního systému. Tyto požadavky zahrnují schopnost systému přidávat, mazat, zobrazovat, upravovat a vyhledávat produkty, zaměstnance, recepty, suroviny, objednávky, zákazníky, faktury, dodací listy, skladové zásoby a plány denní výroby.

6.2.1 Správa produktu

- **R01: Systém umí přidat nový produkt** – Systém by měl poskytovat možnost přidání nového produktu do systému. Uživatel by měl být schopen zadat potřebné informace o novém produktu, jako je název, popis, kategorie, alergen, cenu. Po přidání by měl být nový produkt správně uložen v systému a dostupný pro další manipulaci.
- **R02: Systém umí smazat produkt** – Systém by měl uživateli poskytnout možnost smazat existující produkt ze systému. Po stlačení tlačítka 'smazat' by měl být produkt odstraněn z databáze.
- **R03: Systém umí zobrazit seznam produktů** – Systém by měl poskytovat funkci zobrazení seznamu všech dostupných produktů. Tento seznam by měl obsahovat klíčové informace o každém produktu, jako je název, cena nebo kategorie. Uživatel by měl mít možnost procházet tento seznam a rychle nalézt požadovaný produkt.
- **R04: Systém umí zobrazit detaily produktu** – Uživatel by měl mít možnost zobrazit podrobné informace o konkrétním produktu. To zahrnuje všechny relevantní údaje o produktu, jako je název, popis, cena, kategorie, dostupnost.
- **R05: Systém umí vyhledat produkt** – Systém by měl poskytovat možnost vyhledání konkrétního produktu na základě zadaných kritérií. Uživatel by měl mít možnost použít různé filtry a vyhledávací parametry, aby omezil výsledky a získal přesnější výsledky hledání.

6.2.2 Správa zaměstnanců

- **R07: Systém musí umět přidat nového zaměstnance** – Systém by měl umožnit přidání nového zaměstnance do databáze. Uživatel by měl být schopen zadat potřebné informace o zaměstnanci, jako je jméno, příjmení, datum narození, kontaktní údaje a další. Po přidání by měly být tyto informace uloženy v systému a dostupné pro další manipulaci.
- **R08: Systém umí upravit zaměstnance** – Uživatel by měl mít možnost upravit existující informace o zaměstnanci. Systém by měl poskytovat změnu údajů, jako je kontaktní číslo, adresa, pracovní pozice nebo oddělení. Po provedení změn by měly být aktualizované údaje uloženy v systému.
- **R09: Systém musí umět odstranit zaměstnance** – Systém by měl umožnit odstranění zaměstnance z databáze. Uživatel by měl mít možnost vybrat zaměstnance,

kterého chce odstranit, a potvrdit tuto akci. Po potvrzení by měl být zaměstnanec úspěšně odstraněn ze systému a jeho údaje by neměly být dále dostupné.

- **R10: Systém umí zobrazit seznam zaměstnanců** – Systém by měl poskytovat funkci zobrazení seznamu všech zaměstnanců uložených v databázi. Tento seznam by měl obsahovat základní informace o každém zaměstnanci, jako je jméno, příjmení, pracovní pozice a kontaktní údaje. Uživatel by měl mít možnost procházet tento seznam a rychle nalézt požadovaného zaměstnance.
- **R11: Systém dokáže zobrazit údaje zaměstnance** – Uživatel by měl mít možnost zobrazit podrobné údaje o konkrétním zaměstnanci. Systém by měl poskytovat možnost zobrazit informace, jako je jméno, příjmení, datum narození, pracovní pozice a kontaktní údaje.

6.2.3 Správa receptů

- **R12: Systém musí umožňovat vytvoření nového receptu** – Systém by měl umožnit uživatelům vytvářet nové recepty. Uživatel by měl mít možnost zadat potřebné informace o receptu, jako je název, seznam ingrediencí, postup a čas přípravy. Po vytvoření by měl být recept správně uložen v systému a dostupný pro další manipulaci.
- **R13: Systém musí umožňovat změnu receptu** – Uživatel by měl mít možnost provádět změny v existujících receptech. Systém by měl umožňovat úpravy názvu, ingrediencí, postupu nebo času přípravy. Po provedení změn by měl být recept aktualizován a správně uložen v systému.
- **R14: Systém musí umožnit smazání receptu** – Uživatel by měl mít možnost smazat existující recept ze systému. Systém by měl poskytovat prostředky pro vyhledání a označení receptu k odstranění. Po potvrzení by měl být recept úspěšně odstraněn ze systému a již by neměl být dostupný pro zobrazení nebo úpravy.
- **R15: Systém bude zobrazovat seznam všech receptů** – Systém by měl poskytovat možnost zobrazit seznam všech dostupných receptů. Tento seznam by měl obsahovat názvy receptů nebo čas přípravy, které umožní uživatelům rychle nalézt požadovaný recept.
- **R16: Systém umí zobrazit detaily receptu** – Uživatel by měl mít možnost zobrazit podrobné informace o konkrétním receptu. To zahrnuje název, seznam ingrediencí, postup přípravy, čas přípravy. Tímto způsobem si uživatelé mohou prohlížet podrobnosti receptů a získávat potřebné informace před samotnou přípravou produktu.

- **R17: Systém umí vyhledat recept** – Systém by měl poskytovat možnost vyhledání konkrétního receptu na základě zadaných kritérií. Uživatel by měl mít možnost vyhledat recept podle názvu produktu.

6.2.4 Správa surovin

- **R18: Systém musí umožnit přidání novou surovinu** – Systém by měl umožnit uživatelům přidat novou surovinu do databáze. Uživatel by měl být schopen zadat informace o surovině, jako je název, množství, jednotka. Po přidání by měla být nová surovina správně uložena k receptu.
- **R19: Systém musí umožnit úpravu suroviny** – Uživatel by měl mít možnost upravit existující surovinu. Systém by měl poskytovat možnost změny informací o surovině, jako je název, množství, jednotka. Po provedení změn by měly být aktualizované informace o surovině správně uloženy v systému.
- **R20: Systém bude schopen smazat surovinu** – Uživatel by měl mít možnost smazat existující surovinu ze systému. Po stlačení tlačítka smazat by měla být surovina odstraněna z receptu a databáze.
- **R21: Systém bude zobrazovat seznam potřebných surovin** – Systém by měl poskytovat funkci zobrazení seznamu potřebných surovin při náhledu receptu. Při prohlížení detailů konkrétního receptu by měl uživatel vidět seznam surovin, které jsou potřebné k jeho přípravě. Seznam by měl obsahovat informace o jednotlivých surovinách, jako je název, množství, jednotka. Tímto způsobem uživatel získá přehled o surovinách, které budou potřeba pro daný recept a bude mít možnost se na ně snadno odkázat při přípravě jídla.

6.2.5 Správa objednávek

- **R22: Systém umí přidat nové objednávky** – Systém by měl umožnit přidání nových objednávek. Uživatel by měl být schopen zadat informace o objednavce, jako je datum, zákazník, seznam produktů, množství, cena. Po přidání by měla být nová objednávka správně uložena v systému a dostupná pro další manipulaci.
- **R23: Systém umožní úpravu objednávky** – Uživatel by měl mít možnost upravit existující objednávky. Systém by měl poskytovat možnost změny informací o objednavce, jako je datum, zákazník, seznam produktů, množství, cena. Po provedení změn by měly být aktualizované informace o objednavce správně uloženy v systému.

- **R24: Systém umožní odstranění objednávky** – Systém by měl umožnit odstranění existujících objednávek. Uživatel by měl mít možnost vybrat konkrétní objednávku a potvrdit její odstranění. Po potvrzení by měla být objednávka úspěšně odstraněna ze systému a již není dostupná pro další zobrazení nebo úpravy.
- **R25: Systém umožní zobrazit seznam všech objednávek** – Systém by měl poskytovat funkci zobrazení seznamu všech objednávek. Tento seznam by měl obsahovat informace o jednotlivých objednávkách, jako je číslo objednávky, datum, jméno zákazníka a další relevantní údaje. Uživatel by měl mít možnost procházet tento seznam a rychle nalézt požadovanou objednávku.
- **R26: Systém umí vyhledat objednávku** – Systém by měl poskytovat možnost vyhledat konkrétní objednávku na základě zadaných kritérií. Uživatel by měl mít možnost použít různé filtry a vyhledávací parametry, aby omezil výsledky a získal přesnější výsledky hledání.
- **R27: Systém umí zobrazit detaily objednávky** – Uživatel by měl mít možnost zobrazit podrobnosti o konkrétní objednávce. To zahrnuje informace o objednávce, jako je číslo objednávky, datum, seznam produktů, počet kusů, platební metoda.

6.2.6 Správa zákazníků

- **R28: Systém umí přidat nového zákazníka** – Systém by měl umožnit přidání nového zákazníka do databáze. Uživatel by měl být schopen zadat potřebné informace o zákazníkovi, jako je jméno, příjmení, kontaktní údaje a další relevantní informace. Po přidání by měl být nový zákazník správně uložen v systému a dostupný pro další manipulaci.
- **R29: Systém umí změnit detaily zákazníka** – Uživatel by měl mít možnost upravit existující informace o zákazníkovi. Systém by měl poskytovat prostředky pro změnu údajů, jako je kontaktní číslo, adresa nebo další relevantní informace. Po provedení změn by měly být aktualizované údaje o zákazníkovi správně uloženy v systému.
- **R30: Systém dokáže odstranit zákazníka** – Systém by měl umožnit odstranění existujícího zákazníka. Uživatel by měl mít možnost vybrat zákazníka, kterého chce odstranit, a potvrdit tuto akci. Po potvrzení by měl být zákazník úspěšně odstraněn ze systému a jeho údaje by neměly být dále dostupné.
- **R31: Systém umí zobrazit seznam všech zákazníků** – Systém by měl poskytovat funkci zobrazení seznamu všech zákazníků uložených v databázi. Tento seznam by

měl obsahovat základní informace o každém zákazníkovi, jako je jméno, příjmení, kontaktní údaje a další relevantní informace. Uživatel by měl mít možnost procházet tento seznam a rychle nalézt požadovaného zákazníka.

- **R32: Systém umí zobrazit detaily zákazníka** – Uživatel by měl mít možnost zobrazit podrobné informace o konkrétním zákazníkovi. To zahrnuje všechny relevantní údaje o zákazníkovi, jako je jméno, příjmení, adresa, kontaktní údaje a další atributy spojené s tímto zákazníkem.
- **R33: Systém umí vyhledat zákazníka** – Systém by měl poskytovat možnost vyhledat konkrétního zákazníka na základě zadaných kritérií. Uživatel by měl mít možnost použít různé filtry a vyhledávací parametry, aby omezil výsledky a získal přesnější výsledky hledání.

6.2.7 Správa faktur a dodacích listů

- **R34: Systém umí generovat fakturu z objednávky** – Systém by měl umožnit generování faktury na základě existující objednávky. Uživatel by měl mít možnost vybrat objednávku a prostřednictvím systému vytvořit odpovídající fakturu. Faktura by měla obsahovat potřebné informace o objednateli, jako jsou položky, množství, ceny, celková částka, identifikační údaje a další relevantní informace.
- **R36: Systém umí vystavit fakturu** – Systém by měl umožnit vystavení faktury pro danou objednávku. Uživatel by měl mít možnost vybrat, jestli chce odeslat fakturu zákazníkovi emailem nebo fakturu vytisknout a následně ji fyzicky předat zákazníkovi.
- **R37: Systém umí generovat dodací list z objednávky** – Systém by měl umožnit generování dodacího listu na základě existující objednávky. Uživatel by měl mít možnost vybrat objednávku a prostřednictvím systému vytvořit odpovídající dodací list. Dodací list by měl obsahovat informace o položkách, množství, identifikačních údajích, odesilatele a příjemce.
- **R38: Systém umí vystavit dodací list** – Systém by měl umožnit vystavení dodacího listu pro danou objednávku. Uživatel by měl mít možnost vybrat, jestli chce odeslat dodací list zákazníkovi emailem nebo dodací list vytisknout a následně jej fyzicky předat zákazníkovi.

6.2.8 Správa skladu

- **R40: Systém umí zobrazit produkty na skladu** – Systém by měl umožnit zobrazení seznamu produktů, které jsou momentálně uloženy na skladě. Tento seznam by měl obsahovat informace o jednotlivých produktech, jako je název, množství, dostupnost. Uživatel by měl mít možnost procházet tento seznam a získat přehled o skladové zásobě produktů.
- **R41: Systém umí aktualizovat skladovou zásobu produktů** – Systém by měl umožnit aktualizaci skladové zásoby produktů. Uživatel by měl mít možnost upravit množství produktů na skladě na základě příchodu nebo odebrání produktů.

6.2.9 Správa denní výroby

- **R42: Systém umí vytvořit nový plán denní výroby** – Systém by měl umožnit vytvoření nového plánu denní výroby. Uživatel by měl mít možnost zadat informace o plánu, jako je datum, předpokládané množství výroby a seznam produktů. Po vytvoření by měl být nový plán správně uložen v systému a dostupný pro další manipulaci.
- **R43: Systém umí upravit plán denní výroby** – Uživatel by měl mít možnost upravit existující plán denní výroby. Systém by měl poskytovat prostředky pro změnu informací o plánu, jako je datum, předpokládané množství výroby a seznam produktů. Po provedení změn by měl být plán aktualizován a správně uložen v systému.
- **R44: Systém umí odstranit plán denní výroby** – Systém by měl umožnit odstranění existujícího plánu denní výroby. Uživatel by měl mít možnost vybrat plán, který chce odstranit, a potvrdit tuto akci. Po potvrzení by měl být plán úspěšně odstraněn ze systému a již není dostupný pro další použití nebo zobrazení.
- **R45: Systém umí zobrazit seznam všech plánů denní výroby** – Systém by měl poskytovat funkci zobrazení seznamu všech plánů denní výroby. Tento seznam by měl obsahovat informace o jednotlivých plánech, jako je datum, předpokládané množství výroby a seznam produktů. Uživatel by měl mít možnost procházet tento seznam a rychle nalézt požadovaný plán.

6.2.10 Správa přihlášení

- **R46: Systém umožní přihlášení do systému** – Uživatel bude mít možnost přihlásit se do systému pomocí svých přihlašovacích údajů, jako je uživatelské jméno a heslo.

Systém ověří zadané údaje a pokud jsou správné, uživatel bude úspěšně přihlášen. Přihlášení do systému je klíčovou funkcionalitou, která umožňuje uživatelům získat přístup ke svým datům a provádět další operace v systému.

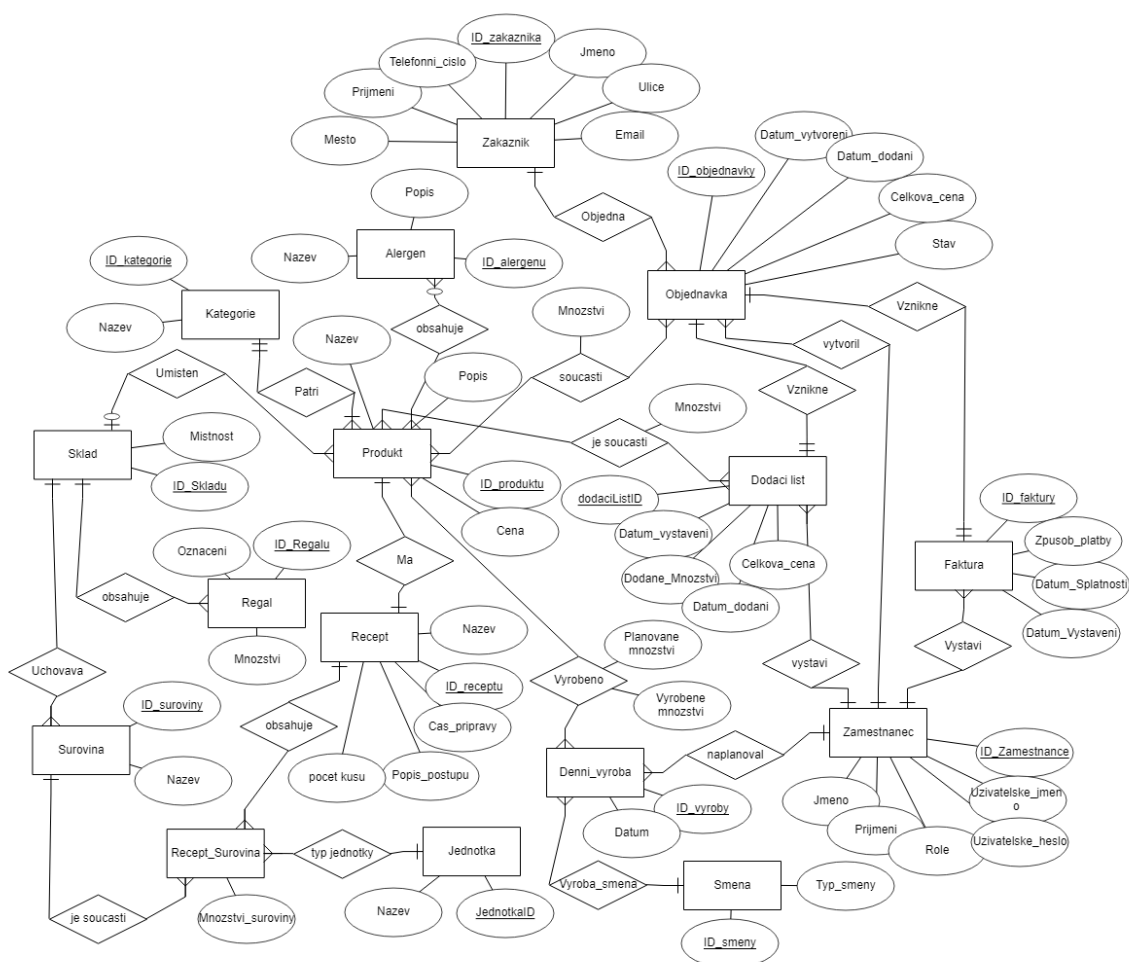
- **R47: Systém umožní odhlášení ze systému** – Uživatel, který je přihlášen do systému, bude mít možnost se odhlásit. Tímto požadavkem se zajistí, že uživatelé mohou ukončit svou relaci se systémem a bezpečně opustit aplikaci. Při odhlášení systém ukončí přístupová oprávnění uživatele a zabezpečí, aby nebylo možné provádět další operace bez přihlášení.

7 NÁVRH DATABÁZE POMOCÍ ERD

V této kapitole je představen návrh databáze pomocí Entitně-relačního diagramu (ERD), který byl vytvořen pomocí nástroje ERDplus. ERDplus je online nástroj pro vytváření a vizualizaci ER diagramů, který umožňuje snadné modelování a navrhování struktury databáze.

ERDplus poskytuje uživatelsky přívětivé rozhraní pro definování entit a jejich atributů, stejně jako pro vytváření vztahů mezi entitami. Nástroj umožňuje snadnou manipulaci s entitami, atributy a vztahy, a tak poskytuje flexibilitu při návrhu databáze.

Na základě navržených požadavků byl v ERDplus vytvořen ER diagram obsahující entity, jejich atributy a vztahy. Tyto informace byly pečlivě zachyceny a zobrazeny v diagramu, který slouží jako grafická reprezentace struktury databáze.



Obrázek 9. Návrh databáze pomocí ERD

7.1 Entity a jejich atributy

V rámci této podkapitoly byly navrženy a popsány základní entity, které budou tvořit informační systém pekárny. Tyto entity byly vybrány s cílem správně zachytit a modelovat různé aspekty provozu pekárny a vzájemné vztahy mezi nimi budou popsány v další podkapitole.

1. Zakazník

- **Popis:** Zákazník je entita, která reprezentuje konkrétní osobu nebo organizaci, která nakupuje produkty od pekárny.
- **Atributy:** ID, jméno, příjmení, adresa (ulice a město), telefonní číslo a emailovou adresu.

2. Objednávka

- **Popis:** Objednávka je entita, která reprezentuje konkrétní transakci mezi pekárnou a zákazníkem.
- **Atributy:** ID, datum vystavení, datum dodání, celková cena, stav

3. Produkt

- **Popis:** Produkt je entita, která reprezentuje konkrétní položku, kterou pekárna prodává.
- **Atributy:** ID, název, popis, cena

4. Alergen

- **Popis:** Alergen je entita, která reprezentuje konkrétní látku, která může u některých lidí způsobit alergickou reakci.
- **Atributy:** ID, název

5. Kategorie

- **Popis:** Kategorie je entita, která slouží k třídění produktů do různých skupin.
- **Atributy:** ID, název

6. Sklad

- **Popis:** Sklad je entita, která reprezentuje místo, kde jsou skladovány suroviny a produkty.
- **Atributy:** ID, regál, množství

7. Surovina

- **Popis:** Surovina je entita, která reprezentuje konkrétní surovinu, která se používá při výrobě produktů.
- **Atributy:** ID, název

8. Recept_Surovina

- **Popis:** Tato entita reprezentuje vztah mezi receptem a surovinou, určuje množství suroviny potřebné pro konkrétní recept.
- **Atributy:** Množství

9. Jednotka

- **Popis:** Jednotka je entita, která reprezentuje měrnou jednotku, která se používá při definování množství suroviny v receptu.
- **Atributy:** ID, název

10. Recept

- **Popis:** Recept je entita, která reprezentuje postup přípravy konkrétního produktu.
- **Atributy:** ID, název, čas přípravy, popis postupu, počet kusů

11. Denní_Vyroba

- **Popis:** Denní výroba je entita, která reprezentuje množství produktů vyrobených v konkrétním dni.
- **Atributy:** ID, datum

12. Smena

- **Popis:** Směna je entita, která reprezentuje konkrétní pracovní směnu (ranní, odpolední nebo večerní) během dne.
- **Atributy:** ID, typ směny

13. Zamestnanec

- **Popis:** Zaměstnanec je entita, která reprezentuje konkrétní osobu pracující v pekárně.
- **Atributy:** ID, jméno, příjmení, uživatelské jméno, uživatelské heslo, role

14. Faktura

- **Popis:** Faktura je entita, která reprezentuje fakturu vystavenou pro konkrétní objednávku.
- **Atributy:** ID, datum vystavení, datum splatnosti, způsob platby

15. Dodací_list

- **Popis:** Dodací list je entita, která reprezentuje dokument obsahující detaily o doručení konkrétní objednávky.
- **Atributy:** ID, datum vystavení, datum dodání, množství dodaného zboží, celková cena

7.2 Vztahy

V podkapitole "Vztahy" jsou popsány podrobně vztahy mezi entitami, které jsou vyobrazené na obrázku 9.

1. Produkt a Kategorie

Více produktů patří do jedné kategorie, zatímco jedna kategorie může mít více produktů. Kardinalita (N:1)

2. Produkt a Alergen

Jeden produkt může obsahovat nula nebo více alergenů a jeden alergen může být spojen s nula nebo více produkty. Kardinalita (N:M)

3. Objednávka a Zakazník

Každá objednávka je spojena s jedním zákazníkem, zatímco jeden zákazník může mít nula nebo více objednávek. Kardinalita (N:1)

4. Objednávka a Faktura

Každá objednávka je spojena právě s jednou fakturou a každá faktura je spojena právě s jednou objednávkou. Kardinalita (1:1)

5. Objednávka a Dodací_list

Každá objednávka je spojena právě s jedním dodacím listem a každý dodací list je spojen právě s jednou objednávkou. Kardinalita (1:1)

6. Objednávka a Produkt

Jedna objednávka může obsahovat více produktů a jeden produkt může být součástí více objednávek. Kardinalita (N:M)

7. Produkt a Sklad

Každý produkt je spojen právě s jedním skladem, zatímco jeden sklad může obsahovat více produktů. Kardinalita (N:1)

8. Produkt a Recept

Každý produkt je spojen právě s jedním receptem a každý recept je spojen právě s jedním produktem. Kardinalita (1:1)

9. Produkt a Denní_vyroba

Jeden produkt může být součástí více denních výrob a jedna denní výroba může obsahovat více produktů. Kardinalita (N:M)

10. Produkt a Dodací_list

Jeden produkt může být součástí více dodacích listů a jeden dodací list může obsahovat více produktů. Kardinalita (N:M)

11. Dodací_list a Zamestnanec

Každý dodací list je spojen právě s jedním zaměstnancem, zatímco jeden zaměstnanec může být spojen s nula nebo více dodacími listy. Kardinalita (N:1)

12. Zamestnanec a Objednavka

Každý zaměstnanec může být spojen s nula nebo více objednávkami, zatímco každá objednávka je spojena právě s jedním zaměstnancem. Kardinalita (1:N)

13. Zamestnanec a Denní_vyroba

Každý zaměstnanec může být spojen s nula nebo více denními výrobami, zatímco každá denní výroba je spojena právě s jedním zaměstnancem. Kardinalita (1:N)

14. Surovina a Sklad

Každá surovina je přiřazena právě jednomu skladu, zatímco jeden sklad může obsahovat nula nebo více surovin. Kardinalita (N:1)

15. Recept_surovina a Surovina

Každá položka v receptu surovin je spojena právě s jednou surovinou, zatímco jedna surovina může být součástí nula nebo více položek receptu surovin. Kardinalita (N:1)

16. Recept_surovina a Jednotka

Každá položka v receptu surovin je spojena právě s jednou jednotkou, zatímco jedna jednotka může být použita pro nula nebo více položek receptu surovin. Kardinalita (N:1)

17. Recept a Recept_surovina

Každý recept může obsahovat jednu nebo více položek receptu surovin, zatímco každá položka receptu surovin je spojena s jedním receptem. Kardinalita (1:N)

18. Faktura a Zamestnanec

Každá faktura je přiřazena právě jednomu zaměstnanci, zatímco jeden zaměstnanec může být spojen s nula nebo více fakturami. Kardinalita (N:1)

8.1 Třídy

Reprezentují typ entit a obsahují atributy a vztahy k jiným třídám.

8.1.1 Objednávka

Představuje objednávku, kterou vlastní zákazník.

- **Atributy** – ID (int), Celková_cena (float), Datum_dodání (date), Datum_vytvoření (date), Stav (boolean)
- **Operace** – SmazatObjednavku(), UpravitObjednavku(), VytvoritObjednavku()

8.1.2 Zákazník

Představuje zákazníka, který nakupuje v pekárně.

- **Atributy** – Email (string), ID (int), Jméno (string), Město (string), Příjmení (string), Tel_číslo (string), Ulice (string)
- **Operace** - SmazatZakaznika(), UpravitZakaznika(), VytvoritZakaznika()

8.1.3 Produkt

Představuje produkt, který je nabízen v pekárně.

- **Atributy** – Cena (float), ID (int), Kategorie (Typ_kategorie), Název (string), Popis (string)
- **Operace** – VytvoritProdukt(), SmazatProdukt(), UpravitProdukt(), PridatProdukt()

8.1.4 Kategorie

Představuje kategorii, do které patří produkt.

- **Atributy** – ID (int), Název (string)
- **Operace** - SmazatKategorii(), UpravitKategorii(), VytvoritKategorii()

8.1.5 Faktura

Představuje fakturu, která je vytvořena z objednávky.

- **Atributy** – Datum_dodání (date), Datum_splatnosti (date), Datum_vystavení (date), ID (int), Způsob_platby (string)
- **Operace** - OdeslatFakturu(), VytisknoutFakturu(), GenerujFakturu()

8.1.6 Dodací_list

Představuje dodací list, který je vytvořen z objednávky.

- **Atributy** – Celková_cena (float), Dodavatel (string), ID (int), Odběratel (string)
- **Operace** - OdeslatDodaciList(), VytisknoutDodaciList(), GenerujDodaciList()

8.1.7 Produkt_Objednavka

Představuje vazbu mezi produktem a objednávkou.

- **Atributy** – Množství (int)
- **Operace** - SnizitMnozstvi(), ZvysitMnozstvi(), ZjistitMnozstvi()

8.1.8 Produkt_Výroba

Představuje vazbu mezi produktem a výrobou.

- **Atributy** – PlánovanéMnožství (int), VyrobenéMnožství (int)
- **Operace** - SnizitMnozstvi(), ZvysitMnozstvi()

8.1.9 Denní_Výroba

Představuje plán denní výroby, kterou vytváří vedoucí výroby.

- **Atributy** – Datum (date), Směna (Typ_směny)
- **Operace** - NaplanovatVyrobu()

8.1.10 Recept

Představuje recept, podle kterého jsou vyrobeny produkty.

- **Atributy** – Čas_přípravy (int), ID (int), Název (řetězec), Popis_postupu (string)
- **Operace** - NastavitCasPripravy(), NastavitNazev(), SmazatRecept(), UpravitRecept(), VytvoritPostup(), VytvoritRecept()

8.1.11 Surovina_Recept

Představuje surovinu v receptu.

- **Atributy** – ID (int), Jednotka (Typ_jednotky), Množství (int), Název (string)
- **Operace** - PridatJednotku(), PridatMnozstvi(), PridatSurovinu(), SmazatSurovinu(), UpravitSurovinu()

8.1.12 Surovina

Představuje surovinu, z které jsou vyrobeny produkty.

- **Atributy** – ID (int), Název (string)
- **Operace** - PřidatSurovinu(), VyhledatSurovinu(), VytvořitSurovinu()

8.1.13 Sklad

Představuje sklad, ve kterém jsou uchovávané produkty.

- **Atributy** – Množství (int), Regal (string)
- **Operace** - SnižMnožství(), ZvyšMnožství(), ZískatMnožství()

8.1.14 Zaměstnanec

Jedná se o abstraktní třídu, která slouží jako základní kostra pro odvození konkrétních tříd zaměstnanců v pekárně.

- **Atributy** – ID (int), Jmeno (string), Pozice (string), Prijmeni (string), role (string), Uzivatelske_heslo (string), Uzivatelske_jmeno (string)
- **Operace** - SmazatZamestnance(), UpravitZamestnance(), VytvořitZamestnance()

8.1.15 Vedoucí_Výroby

Představuje potomka zaměstnance, který se stará o recepty a plánování výroby.

8.1.16 Majitel

Představuje majitele pekárny, který je potomkem vedoucího výroby. Zastupuje roli admina, který může v systému upravovat cokoliv – vytvářet nebo mazat zaměstnance, produkty nebo plány denní výroby.

8.1.17 Skladník

Představuje skladníka, který je potomkem zaměstnance a stará se o sklad a dodací listy.

8.1.18 Pokladní

Jedná se o pokladní, která je potomkem zaměstnance a stará se o zákazníky a objednávky.

8.1.19 Účetní

Jedná se účetní, který je potomkem zaměstnance a stará se o faktury.

8.1.20 Regál

Jedná se o třídu, která reprezentuje část ve skladu, kde jsou uchovávány produkty.

- **Atributy:** ID (int), množství (int), označení (string)
- **Operace:** ZнизMnozství(), ZískatMnozství(), ZvysMnozství()

8.1.21 Sklad

- **Atributy:** ID (int), Místnost (string)

8.2 Výčtové typy

Reprezentují konkrétní hodnoty, které tvoří omezenou množinu možností a jsou neměnné.

8.2.1 Typ_alergenu

- **Hodnoty** – Lepek, Koryši, Vejce, Ryby, Arašidy, Mléko, Skořápkové plody, Celer, Sójové boby, Hořčice, Sezamová semena, Měkkýši, Vlčí bob, Siřičitany, oxid siřičitý

8.2.2 Typ_jednotky

- **Hodnoty** – mg, g, dkg, kg, dcl, dl, l

8.2.3 Typ_směny

- **Hodnoty** – Ranní, Odpolední, Noční

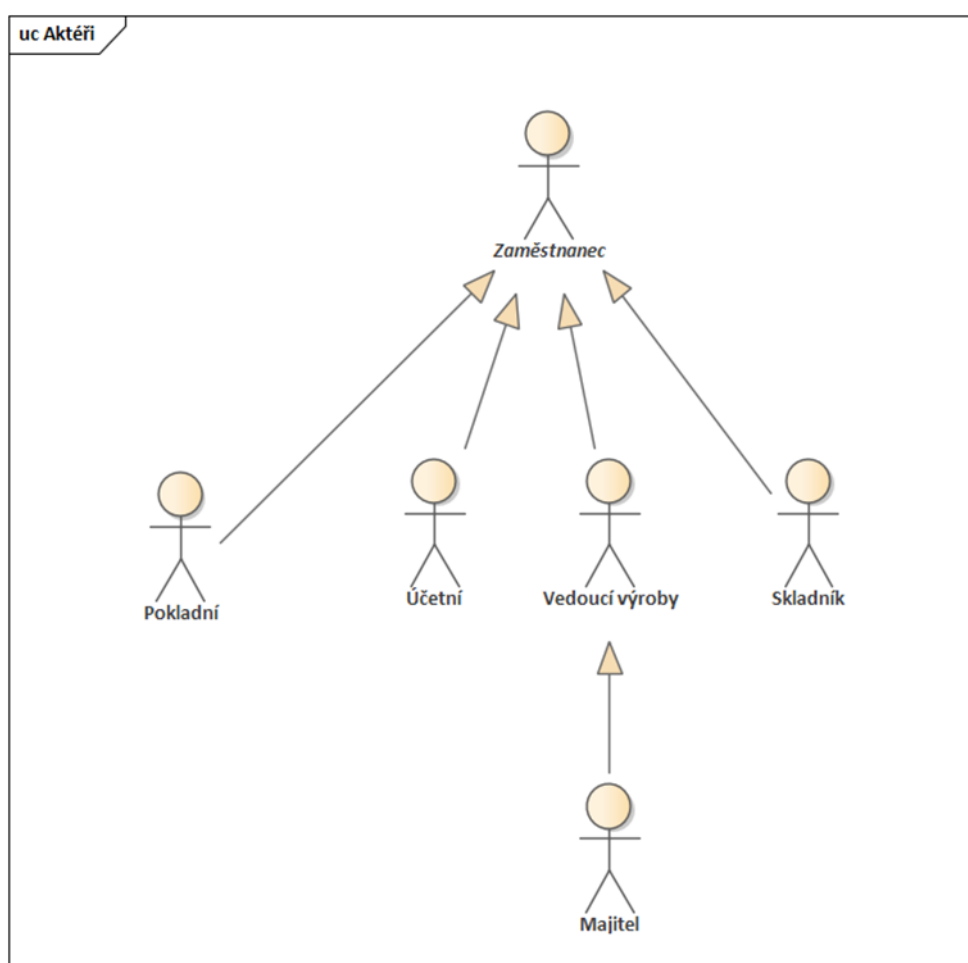
9 REALIZACE FUNKČNÍCH POŽADAVKŮ

V této kapitole se zaměříme na realizaci případů užití, což je klíčová fáze v modelování a analýze systému. Během této fáze budou identifikováni aktéři, kteří mají důležitou roli při komunikaci s navrhovaným informačním systémem, a také budou řešeny konkrétní případy užití, které popisují interakce mezi aktéry a systémem.

9.1 Aktéři

Aktéři jsou klíčovou součástí modelování a analýzy systému, přičemž jsou představováni různými rolmi nebo externími entitami, které komunikují se systémem. Lidé, jiný systém, hardware nebo i externí organizace mohou být aktéry.

V kontextu mé bakalářské práce je identifikováno několik aktérů, kteří mají specifické role a odpovědnosti v rámci navrhovaného informačního systému. Každý aktér je charakterizován svými individuálními požadavky, přístupy a cíli.



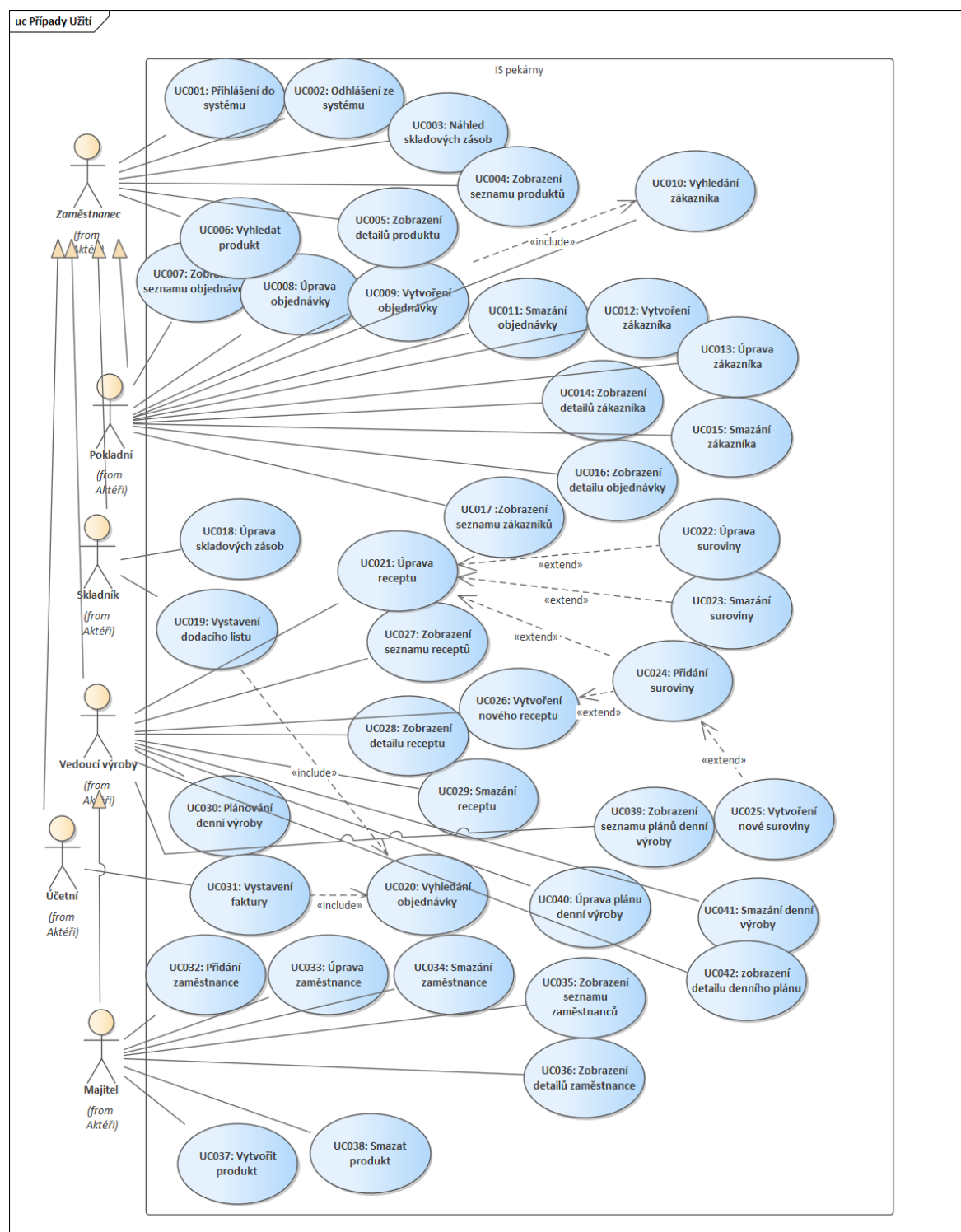
Obrázek 11. Aktéři

1. **Zaměstnanec:** Je obecným aktérem, který reprezentuje všechny zaměstnance pekárny, bez ohledu na jejich konkrétní roli. Zaměstnanec může mít přístup k informacím o produktech, skladových zásobách a provádět přihlášení a odhlášení ze systému.
2. **Pokladní:** Pokladní je zaměstnancem pekárny. Zabývá se objednávkami, zákazníky a produkty. Pokladní má přístup a oprávnění ke změnám, vytváření a mazání objednávek a zákazníků.
3. **Účetní:** Finanční aspekty pekárny a vystavování faktur řeší účetní, který je zaměstnancem pekárny.
4. **Skladník:** Surovinami a hotovými produkty se zabývá skladník, který je zaměstnancem pekárny zodpovědným za správu skladu a manipulaci.
5. **Vedoucí výroby:** Vedoucí výroby má zodpovědnost za manipulaci s recepty, surovinami a plánování denní výroby. Je zaměstnancem pekárny.
6. **Majitel:** Hlavním aktérem je majitel pekárny, který má zodpovědnost za celkové řízení a správu pekárny. Majitel disponuje širokými pravomocemi a oprávněními, včetně přidávání zaměstnanců, správy produktů a zobrazení informací o zaměstnancích.

9.2 Případy užití

V této části se věnujeme případům užití, které slouží k realizaci funkčních požadavků navrhovaného informačního systému. Případ užití představuje konkrétní scénář, ve kterém je popsána komunikace mezi aktéry a systémem a ukazuje, jak systém odpovídá na požadavky uživatelů. V rámci těchto případů užití analyzujeme a popisujeme konkrétní úkoly nebo akce, které aktéři provádějí v systému. Tímto způsobem zajišťujeme, že informační systém naplňuje funkční požadavky.

Hlavní případy užití jsou uvedeny v tabulkách pod obrázkem 12. Neuvedené případy užití se nachází v příloze (CD).



Obrázek 12. Diagram případu užití

Tabulka 1. Vytvoření objednávky

Název: Vytvoření objednávky		
ID: UC009		
Charakteristika: Popisuje vytvoření objednávky		
Primární aktér: Pokladní		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Pokladní má otevřený seznam objednávek.		
Výstupní podmínky: Systém vytvoří novou objednávku.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Pokladní	Klikne na tlačítko "Vytvořit objednávku".
2	Systém	Zobrazí prázdný formulář objednávky.
3	Pokladní	Pokračuje podle <include> Vyhledání zákazníka.
4	Pokladní	Doplní produkty a množství.
5	Pokladní	Stiskne tlačítko "Uložit".
6	Systém	Vytvoří novou objednávku.
7	-	Tímto případ užití končí.
Alternativní scénáře: UC009a – Tlačítko zrušit		

Tabulka 2. Zrušení vytvoření objednávky

Název – Alternativní scénář: Alternativní scénář: UC009a		
ID: UC009a		
Charakteristika: Zachycení alternativního toku případu užití		
Alternativní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Pokladní	Klikne na tlačítko "Zrušit".
2	Systém	Zavře formulář na vytvoření objednávky a vrátí pokladní na seznam objednávek.

Tabulka 3. Vyhledání zákazníka

Název: Vyhledání zákazníka		
ID: UC010		
Charakteristika: Případ užití popisuje vyhledávání zákazníka		
Primární aktér: Pokladní		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Pokladní má otevřený formulář pro vytvoření objednávky.		
Výstupní podmínky: Systém zobrazí hledaného zákazníka.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Pokladní	Do vyhledávacího pole zadá jméno a příjmení zákazníka.
2	Systém	Zobrazí hledaného zákazníka.
3	Pokladní	Vybere zákazníka.
4	Systém	Vyplní formulář údaji zákazníka.
5	-	Tímto případ užití končí.
Alternativní scénáře: UC010a – Zákazník neexistuje		

Tabulka 4. Zákazník neexistuje

Název – Alternativní scénář: Alternativní scénář: UC010a		
ID: UC010a		
Charakteristika: Zachycení alternativního toku případu užití		
Alternativní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Systém	Zobrazí hlášku, že hledaný zákazník neexistuje.

Tabulka 5. Smazání objednávky

Název: Smazání objednávky		
ID: UC011		
Charakteristika: Popisuje smazání objednávky		
Primární aktér: Pokladní		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Pokladní má otevřené detaily objednávky.		
Výstupní podmínky: Systém smaže objednávku.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Pokladní	Vybere jednu objednávku ze seznamu objednávek a klikne na tlačítko "Smazat objednávku".
2	Systém	Zobrazí hlášku "Opravdu chcete smazat objednávku?".
3	Pokladní	Klikne na tlačítko "Ano".
4	Systém	Odstraní objednávku.
5	-	Tímto případ užití končí.
Alternativní scénáře: UC011a – Tlačítko "Ne"		

Tabulka 6. Zrušení smazání objednávky

Název – Alternativní scénář: UC011a		
ID: UC011a		
Charakteristika: Zachycení alternativního toku případu užití		
Alternativní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Pokladní	Klikne na tlačítko "Ne".
2	Systém	Schová hlášku o smazání objednávky.

Tabulka 7. Vytvoření zákazníka

Název: Vytvoření zákazníka		
ID: UC012		
Charakteristika: Zachycení ukázkového případu užití		
Primární aktér: Pokladní		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Pokladní má zobrazený seznam zákazníků.		
Výstupní podmínky: Systém vytvoří zákazníka.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Pokladní	Klikne na tlačítko "Nový zákazník".
2	Systém	Zobrazí prázdný formulář zákazníka.
3	Pokladní	Vyplní formulář (jméno, příjmení, tel. číslo, email, ulice, město).
4	Pokladní	Klikne na tlačítko "Vytvořit zákazníka".
5	Systém	Vytvoří zákazníka.
6	-	Případ užití končí.
Alternativní scénáře: UC012a – Tlačítko zrušit		

Tabulka 8. Zrušení vytvoření zákazníka

Název – Alternativní scénář: UC012a		
ID: UC012a		
Charakteristika: Zachycení alternativního toku případu užití		
Alternativní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Pokladní	Klikne na tlačítko "Zrušit".
2	Systém	Zavře formulář pro vytvoření zákazníka.

Tabulka 9. Úprava skladových zásob

Název: Úprava skladových zásob		
ID: UC018		
Charakteristika: Zachycení ukázkového případu užití		
Primární aktér: Skladník		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Skladník má zobrazený seznam skladových zásob.		
Výstupní podmínky: Systém upraví zásoby.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Skladník	Vybere produkty.
2	Systém	Otevře editovací okno.
3	Skladník	Změní množství u produktů a zvolí "Uložit".
4	Systém	Uloží změny.
5	-	Tímto případ užití končí.
Alternativní scénáře: Nejsou		

Tabulka 10. Vystavení dodacího listu

Název: Vystavení dodacího listu		
ID: UC019		
Charakteristika: Tento případ užití popisuje vystavení dodacího listu		
Primární aktér: Skladník		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Skladník má otevřenou složku faktur.		
Výstupní podmínky: Systém vystaví dodací list.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Skladník	Vybere funkci "Vystavit dodací list".
2	Systém	Zobrazí prázdný formulář dodacího listu a pole pro hledání.
3	-	Dále se pokračuje podle «include» UC020: Vyhledání objednávky.
4	Skladník	Vybere odpovídající fakturu.
5	Systém	Vyplní dodací list údaji z faktury a zobrazí tlačítka "Uložit" a "Upravit".
6	Skladník	Klikne na tlačítko "Uložit".
7	Systém	Uloží dodací list a zobrazí možnosti "Odeslat" a "Tisk".
8	-	Pokud skladník zvolí "Odeslat":
9	Systém	Odešle dodací list zákazníkovi na email.
10	-	Pokud skladník zvolí "Tisk":
11	Systém	Odešle žádost tiskárně, která dodací list vytiskne.
12	-	Tímto případ užití končí.
Alternativní scénáře: UC019a – Upravit		

Tabulka 11. Úprava dodacího listu

Název – Alternativní scénář: UC019a		
ID: UC019a		
Charakteristika: Zachycení alternativního toku případu užití		
Alternativní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Skladník	Stiskne tlačítko "Upravit".
2	Systém	Zobrazí možnost úpravy produktů a jejich množství.
3	Skladník	Změní množství a klikne na "Uložit".

Tabulka 12. Vyhledání objednávky

Název: Vyhledání objednávky		
ID: UC020		
Charakteristika: Zachycení ukázkového případu užití		
Primární aktér: Skladník		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Skladník má otevřený formulář pro vystavení faktury.		
Výstupní podmínky: Systém vyhledá objednávku.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Skladník	Zvolí možnost vyhledat objednávku.
2	Systém	Zobrazí vyhledávací pole pro objednávku
3	Skladník	Zadá do vyhledávacího okna (číslo objednávky/jméno zákazníka/datum objednávky).
4	Systém	Vyhledá objednávky odpovídajícím zadaným informacím a zobrazí výsledky.
Alternativní scénáře: Nejsou		

Tabulka 13. Úprava receptu

Název: Úprava receptu		
ID: UC021		
Charakteristika: Zachycení ukázkového případu užití		
Primární aktér: Vedoucí výroby		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Vedoucí výroby má zobrazený detail receptu.		
Výstupní podmínky: Detaily receptu jsou změněny.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Vedoucí výroby	Klikne na tlačítko "Upravit".
2	System	Zobrazí formulář s detaily vybraného receptu.
3	Vedoucí výroby	Provádí úpravy v potřebných polích receptu.
4	-	Provede jedno z následujících: <extend> Úprava suroviny <extend> Smazání suroviny <extend> Přidání suroviny
5	Vedoucí výroby	Po provedení změn klikne na tlačítko "Uložit".
6	System	Změní detaily receptu.
7	-	Tímto případ užití končí.
Alternativní scénáře: Nejsou		

Tabulka 14. Úprava suroviny

Název: Úprava suroviny		
ID: UC022		
Charakteristika: Tento případ užití popisuje upravení suroviny.		
Primární aktér: Vedoucí výroby		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Podmínka, která musí být splněna před spuštěním případu užití.		
Výstupní podmínky: Podmínka, která musí být splněna po provedení scénáře.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Vedoucí výroby	Klikne na tlačítko "Upravit".
2	Systém	Zobrazí editové pole suroviny.
3	Vedoucí výroby	Upraví surovinu.
4	Vedoucí výroby	Klikne na tlačítko "Uložit".
5	Systém	Uloží změny suroviny.
6	-	Tímto případ užití končí.
Alternativní scénáře: Nejsou		

Tabulka 15. Přidání suroviny

Název: Přidání suroviny		
ID: UC024		
Charakteristika: Zachycení ukázkového případu užití		
Primární aktér: Vedoucí výroby		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Vedoucí výroby má otevřenou úpravu receptu.		
Výstupní podmínky: Systém přidá surovinu.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Vedoucí výroby	Do pole pro přidání suroviny vyplní množství, jednotku a název suroviny.
2	Vedoucí výroby	Stlačí tlačítko "Uložit".
3	Systém	Ověří, zda se surovina nachází v seznamu.
4	-	Surovina se nachází v seznamu.
5	Systém	Přidá surovinu do receptu.
6	-	Tímto případ užití končí.
Alternativní scénáře: UC024a – Surovina se nenachází v seznamu		

Tabulka 16. Surovina neexistuje

Název – Alternativní scénář: UC024a		
ID: UC024a		
Charakteristika: Zachycení alternativního toku případu užití		
Alternativní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Systém	Přejít na případ užití Vytvoření nové suroviny. «extend» UC025: Vytvoření nové suroviny
2	Systém	Přidá surovinu do receptu.

Tabulka 17. Vytvoření nové suroviny

Název: Vytvoření nové suroviny		
ID: UC025		
Charakteristika: Popisuje vytvoření nové suroviny		
Primární aktér: Vedoucí výroby		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Nezbytná surovina není nalezena v seznamu.		
Výstupní podmínky: Systém vytvoří novou surovinu.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Vedoucí vý- roby	Vyplní pole pro surovinu.
2	Systém	Nenajde surovinu v seznamu.
3	Vedoucí vý- roby	Klikne na tlačítko "Uložit".
4	Systém	Uloží novou surovinu.
5	-	Tímto případ užití končí.
Alternativní scénáře: Nejsou		

Tabulka 18. Vytvoření nového receptu

Název: Vytvoření nového receptu		
ID: UC026		
Charakteristika: Zachycení ukázkového případu užití		
Primární aktér: Vedoucí výroby		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Recept nemůže být vytvořen bez surovin. Vedoucí výroby má zobrazený seznam produktů		
Výstupní podmínky: Systém vytvoří nový recept		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Vedoucí výroby	Zvolí přidat nový recept u produktu.
2	Systém	Zobrazí prázdný formulář pro nový recept.
3	Vedoucí výroby	Vyplní formulář (název, čas přípravy, postup).
4	Vedoucí výroby	Přidává surovinu (název, jednotka, množství).
5	Systém	<extend> UC024 Přidání suroviny.
6	Vedoucí výroby	Zvolí uložit recept.
7	Systém	Uloží nový recept.
8	-	Tímto případ užití končí.
Alternativní scénáře: Nejsou		

Tabulka 19. Plánování denní výroby

Název: Plánování denní výroby		
ID: UC030		
Charakteristika: Případ užití popisuje plánování denní výroby.		
Primární aktér: Vedoucí výroby		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Vedoucí výroby otevřel rozhraní pro plánování denní výroby.		
Výstupní podmínky: Systém vytvořil plán denní výroby.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Vedoucí výroby	Otevře rozhraní pro plánování denní výroby.
2	Systém	Zobrazí kalendář a pole pro typ směny.
3	Vedoucí výroby	Zvolí den, pro který chce naplánovat výrobu a nastaví typ směny (ranní, odpolední, noční).
4	Systém	Zobrazí seznam produktů a jejich počet, který je potřeba vyrobit (systém také zohledňuje produkty na skladě).
5	Systém	Seřadí zobrazený seznam podle času přípravy.
6	Vedoucí výroby	Vybere produkty, které budou připravovat na dané směně.
7	Vedoucí výroby	Klikne na tlačítko "Uložit".
8	Systém	Vytvoří plán výroby a přidá seznam potřebných surovin a přepočte množství dle receptur.
9	Systém	Přidá o 10 % větší množství surovin pro případné chyby při přípravě.
10	-	Tímto případ užití končí.
Alternativní scénáře: Nejsou		

Tabulka 20. Vystavení faktury

Název: Vystavení faktury		
ID: UC031		
Charakteristika: Popisuje postup vystavení faktury		
Primární aktér: Účetní		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Nejsou		
Výstupní podmínky: Systém vystaví fakturu		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Účetní	Vybere složku "Faktury" z menu.
2	Systém	Zobrazí seznam dostupných funkcí ve složce "Faktury".
3	Účetní	Vybere možnost "Vystavit fakturu".
4	Systém	Zobrazí formulář pro vystavení faktury.
5	-	Pokračuje se podle <include> UC020: Vyhledání objednávky.
6	Účetní	Vybere odpovídající objednávku.
7	Systém	Načte a vyplní formulář údaji objednávky (detaily, položky, množství, cena, atd.).
8	Účetní	Zkontroluje detaily objednávky a stiskne tlačítko "Ok".
9	Systém	K formuláři zobrazí tlačítka "Odeslat" a "Tisk".
10	Účetní	Zvolí možnost "Odeslat":
11	Systém	Odešle fakturu zákazníkovi na email.
12	Účetní	Zvolí možnost "Tisk":
13	Systém	Odešle žádost tiskárně, aby vytiskla fakturu.
14	-	Tímto případ užití končí.
Alternativní scénáře: Nejsou		

Tabulka 21. Vyhledání objednávky

Název: Vyhledání objednávky		
ID: UC020		
Charakteristika: Případ užití popisuje hledání objednávky		
Primární aktér: Účetní		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Podmínka, která musí být splněna před spuštěním případu užití.		
Výstupní podmínky: Podmínka, která musí být splněna po provedení scénáře.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Účetní	Zvolí možnost vyhledat objednávku.
2	Systém	Zobrazí vyhledávací pole pro objednávku.
3	Účetní	Zadá do vyhledávacího okna (číslo objednávky/jméno zákazníka/datum objednávky).
4	Systém	Vyhledá objednávky odpovídajícím zadaným informacím a zobrazí výsledky.
Alternativní scénáře: UC001a – Alternativní scénář		

Tabulka 22. Přidání zaměstnance

Název: Přidání zaměstnance		
ID: UC032		
Charakteristika: Popis vytvoření zaměstnance		
Primární aktér: Majitel		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Admin/majitel má zobrazený seznam zaměstnanců.		
Výstupní podmínky: Systém přidá zaměstnance.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Majitel	Klikne na tlačítko "Přidat zaměstnance".
2	Systém	Zobrazí prázdný formulář pro zaměstnance.
3	Majitel	Vyplní formulář pro zaměstnance (jméno, příjmení, email, tel. číslo, město) a přidá mu roli.
4	Majitel	Klikne na tlačítko "Uložit".
5	Systém	Uloží nového zaměstnance.
6	-	Tímto případ užití končí.
Alternativní scénáře: Nejsou		

Tabulka 23. Vytvořit produkt

Název: Vytvořit produkt		
ID: UC037		
Charakteristika: Případ užití popisuje vytvoření produktu		
Primární aktér: Majitel		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Majitel má zobrazený seznam produktů.		
Výstupní podmínky: Systém vytvoří produkt.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Majitel	Stiskne tlačítko "Přidat".
2	Systém	Zobrazí formulář produktu.
3	Majitel	Vyplní název produktu, přidá obrázek a vybere kategorii, alergen.
4	Majitel	Klikne na tlačítko "Uložit".
5	Systém	Uloží produkt.
6	-	Tímto případ užití končí.
Alternativní scénáře: Nejsou		

Tabulka 24. Zobrazení seznamu plánů denní výroby

Název: Zobrazení seznamu plánů denní výroby		
ID: UC039		
Charakteristika: Zachycení ukázkového případu užití		
Primární aktér: Vedoucí výroby		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Nejsou		
Výstupní podmínky: Systém zobrazí seznam plánů denní výroby.		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Vedoucí vý- roby	Vybere složku "Denní výroby"
2	Systém	Zobrazí seznam plánů denní výroby (ID, den, typ směny, produkt, vyrobí, na skladě, čas přípravy, stav)
3	-	Tímto případ užití končí
Alternativní scénáře: Nejsou		

Tabulka 25. Úprava plánu denní výroby

Název: Úprava plánu denní výroby		
ID: UC040		
Charakteristika: Zachycení ukázkového případu užití		
Primární aktér: Vedoucí výroby		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Vedoucí výroby má otevřený plán denní výroby		
Výstupní podmínky: Vedoucí výroby změní plán denní výroby		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Vedoucí vý- roby	Stlačí "Upravit"
2	Systém	Systém zobrazí formulář, kde jsou následující informace (den, typ směny, produkty, počet kusů)
3	Vedoucí vý- roby	Zvolí "Uložit"
4	Systém	Systém upraví plán výroby, seznam potřebných surovin a přepočítá potřebné množství
5	-	Tímto případ užití končí
Alternativní scénáře: Nejsou		

Tabulka 26. Zobrazení detailu denního plánu

Název: Zobrazení detailu denního plánu		
ID: UC042		
Charakteristika: Zachycení ukázkového případu užití		
Primární aktér: Vedoucí výroby		
Vedlejší aktéři: Nejsou		
Vstupní podmínky: Vedoucí výroby má zobrazený seznam plánů denní výroby		
Výstupní podmínky: Systém zobrazí plán denní výroby		
Hlavní scénář:		
Krok	Aktér/Sys- tém	Popis
1	Vedoucí vý- roby	Rozklikne plán denní výroby
2	Systém	Systém zobrazí recepty produktů, které jsou v plánu výroby
3	Systém	Zobrazené recepty mají přepočítané množství surovin podle množství. Recepty mají i upravený čas přípravy podle množství
5	-	Tímto případ užití končí
Alternativní scénáře: Nejsou		

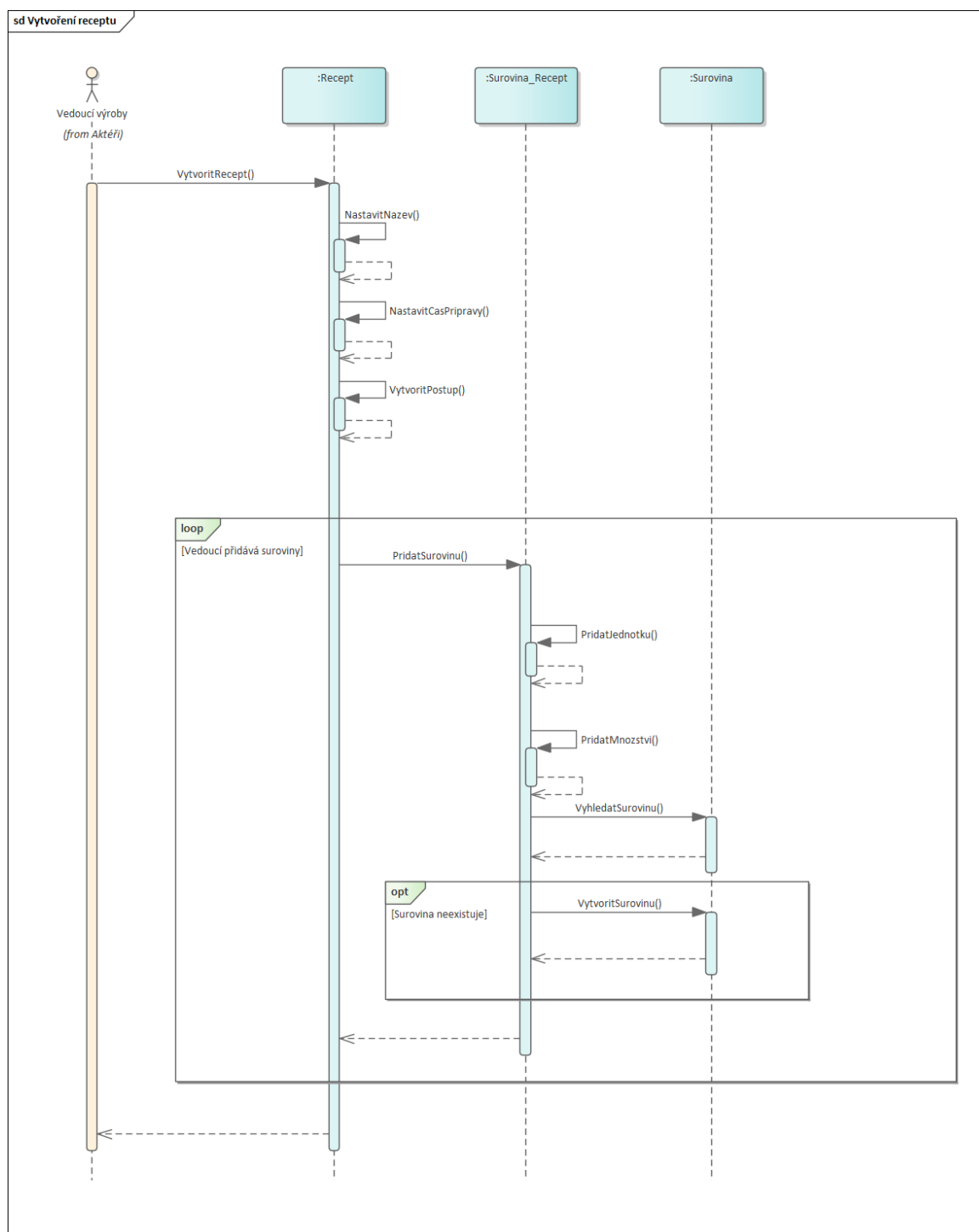
Obrázek 13. Matice vztahů

10 DIAGRAMY PROCESŮ

V této kapitole je proveden podrobný rozbor a vizualizace klíčových procesů spojených s návrhem informačního systému. Pro tento účel jsou použity sekvenční diagramy, které graficky představují komunikaci mezi různými aktéry a systémem v čase.

V první části této kapitoly je analyzován proces vytváření nových receptů. Proces je zde prezentován krok za krokem, tak jak je vnímán vedoucím výroby, a je ukázáno, jak systém reaguje na jednotlivé akce.

V druhé části je zkoumáno plánování denní výroby. Hlavní důraz je opět kladen na interakci mezi vedoucím výroby a systémem, tentokrát se zaměřením na koordinaci a plánování výrobních procesů.

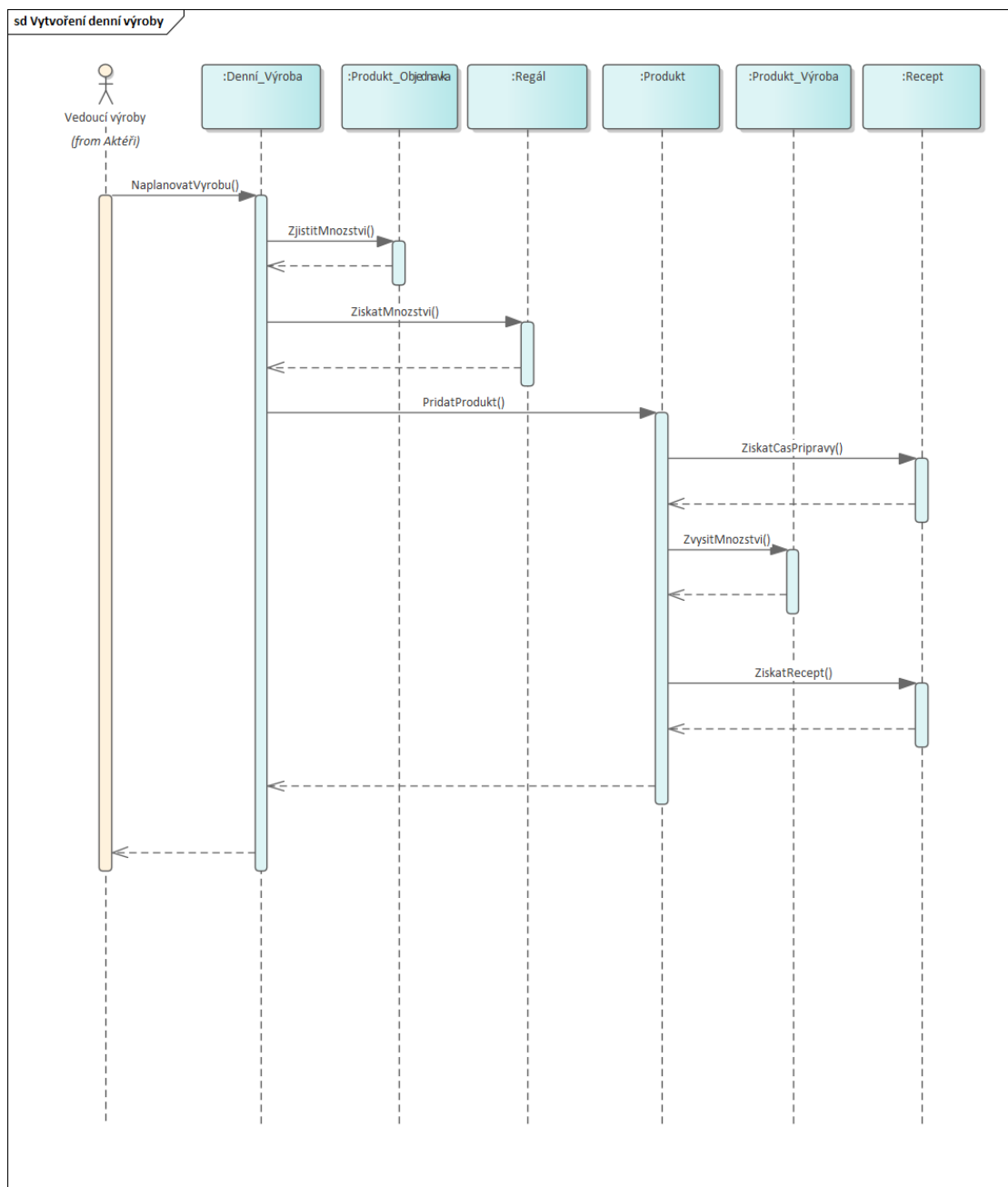


Obrázek 14. Sekvenční diagram vytvoření receptu

Proces vytváření nového receptu začíná, když vedoucí výrobu klikne na tlačítko Vytvořit recept. V systému, který si můžeme představit jako počítačovou aplikaci, je následně otevřen prázdný formulář pro recept. Vedoucí výroby do tohoto formuláře zadá název receptu, čas potřebný k jeho přípravě a podrobný postup přípravy.

Následuje fáze přidávání surovin do receptu, během které je pro každou surovinu zadán název, jednotka (například gramy, litry atd.) a množství. Pokud je některá surovina nová a v systému dosud neexistuje, bude automaticky vytvořena.

Po přidání všech surovin je recept uložen do systému a zobrazen v seznamu receptů.



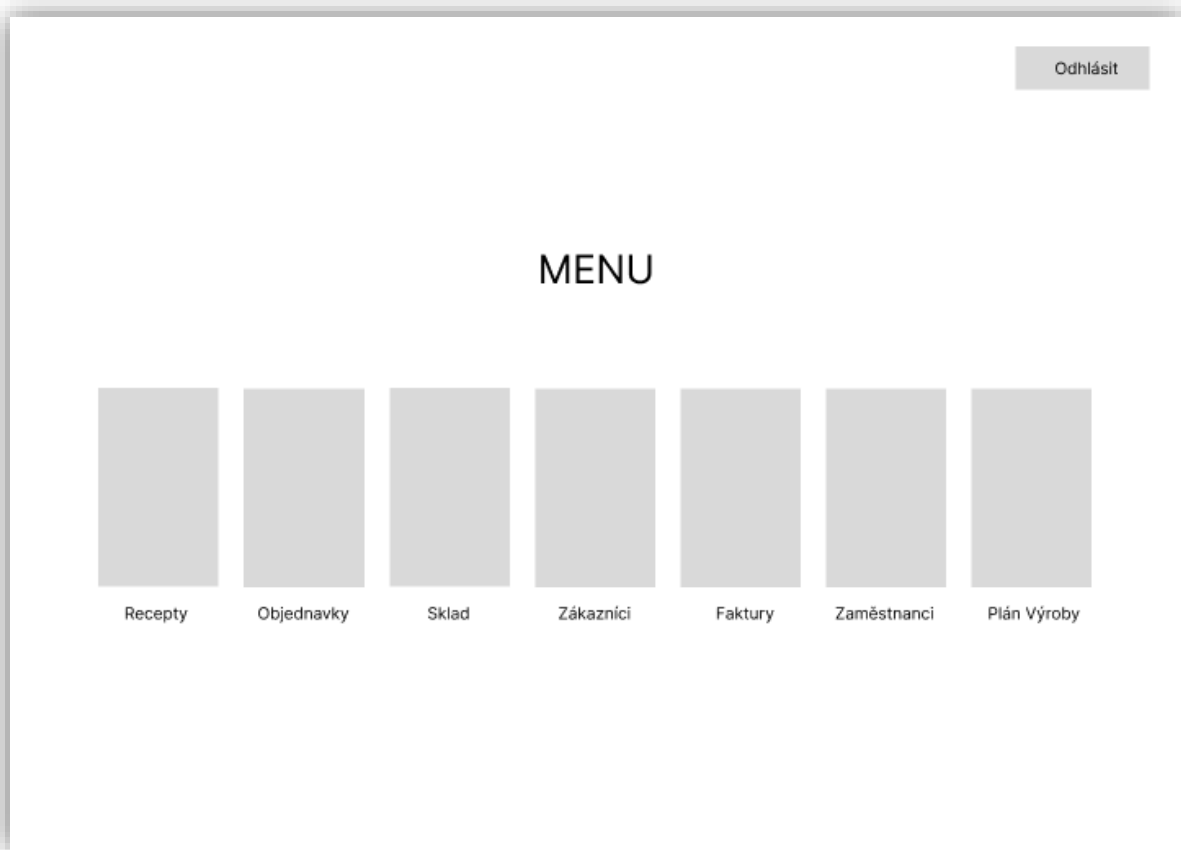
Obrázek 15. Sekvenční diagram vytvoření denní výroby

Celý proces vytváření denní výroby začíná, když vedoucí výroby oznámí systému potřebu naplánovat výrobu. Systém zobrazí kalendář, kde vedoucí výroby zvolí den, pro který chce vytvořit plán výroby. Poté vyznačí typ směny, jestli se jedná o směnu ranní, odpolední nebo noční. Po přijetí požadavku systém zobrazí z objednávek seznam produktů, potřebných pro výrobu a čas jejich přípravy. Výsledné číslo bude sníženo o počet produktů již vyrobených a uložených na skladě. Vedoucí výroby následně zvolí podle času přípravy produkty, které se budou vyrábět. Systém poté zobrazí recept vybraných produktů. Množství surovin v receptu již bude odpovídat zadanému počtu produktů a bude automaticky navýšeno o dalších

10 % pro případnou nepředvídatelnou chybu ve výrobě. Vedoucí výroby má v tomto okamžiku ještě možnost upravit celkové množství produktů pro výrobu (navýšit/snížit). Vedoucí výroby denní plán výroby uloží. Systém plán výroby zobrazí v seznamu denní výroby dle datumu.

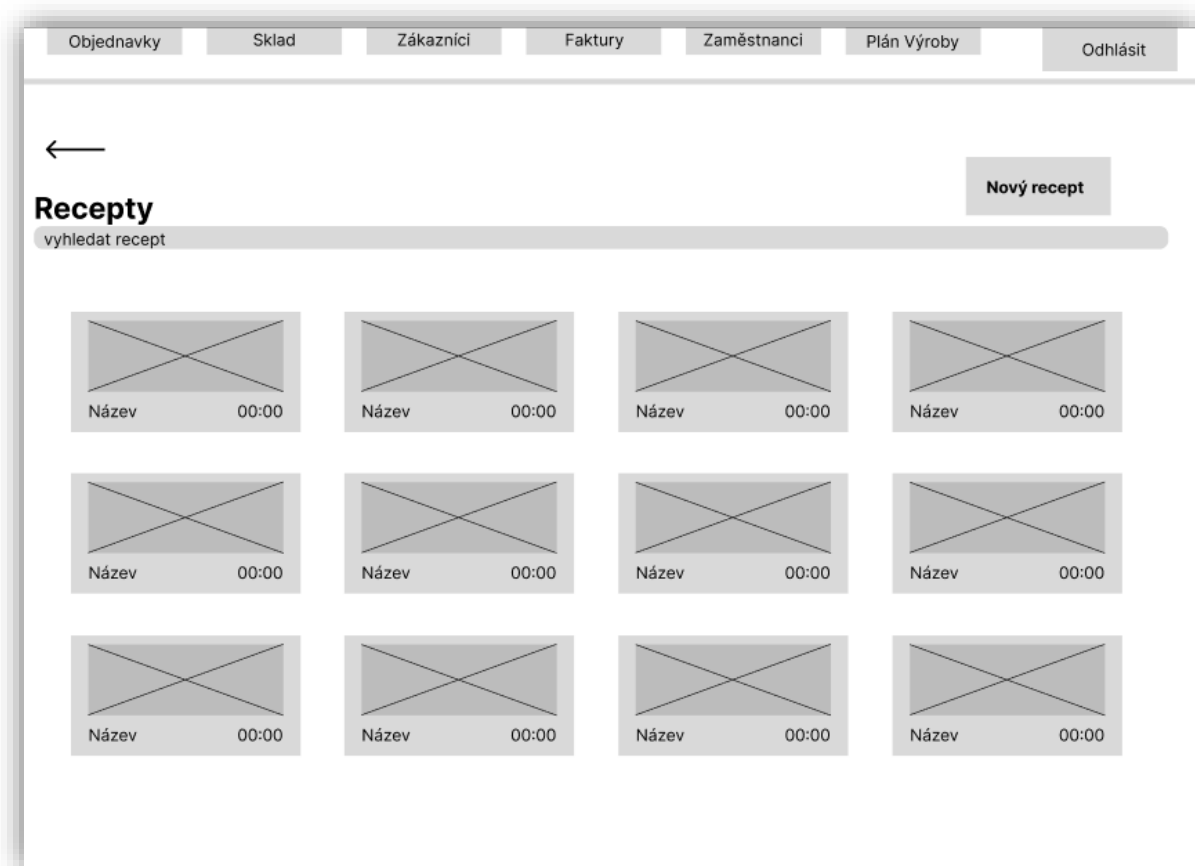
11 DRÁTĚNÉ MODELY

Ukázkové drátěné modely jsou popsány z pohledu role admina.



Obrázek 16. Menu informačního systému

Po přihlášení do informačního systému bude uživateli zobrazeno menu podle jeho přiřazené role. Pro roli účetní budou zobrazeny složky pro Objednavky a Faktury. Pro roli skladníka by to byla složka Skladu.



Obrázek 17. Seznam receptů

Po rozkliknutí složky Recepty je uživateli zobrazen seznam receptů. Může vytvořit nový recept, vyhledat recept.

[illegible]

Obrázek 18. Detail receptu

Po rozkliknutí receptu jsou jeho informace zobrazeny uživateli. Mezi tyto informace patří název receptu, obrázek produktu, čas přípravy, seznam surovin, popis postupu přípravy. Uživatel má možnost recept upravit nebo smazat. Přístup k receptu má pouze admin a vedoucí výroby.

Objednavky Sklad Zákazníci Faktury Zaměstnanci Plán Výroby Odhlásit

←

Nový recept

Název receptu Čas přípravy v minutách

0 Zrušit Uložit

Postup

Nahrát obrázek

text text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text text
text

Suroviny

Množství Jednotka Název

0 Přidat

počet jednotka surovina ×
počet jednotka surovina ×

Obrázek 19. Vytvoření nového receptu

Pokud uživatel stiskne tlačítko „Nový recept“, zobrazí se mu prázdný formulář pro vytvoření nového receptu. Formulář obsahuje pole pro název receptu, čas jeho přípravy v minutách, postup přípravy a sekci pro suroviny, která obsahuje pole pro množství, jednotku a název suroviny. Při vyplňování pole pro jednotku se otevře seznam s nabídkou jako je gram, kilogram, litr atd.

Při vyplňování pole pro název suroviny bude zadaný název vyhledán v databázi surovin. Pokud se nachází v databázi, tak uživatel bude moci vybrat zadanou surovinu z databáze. Pokud v databázi není, uživatel zadá celý název a klikne na „Přidat“, tak bude surovina přidána a zobrazena pod vyplňovacími poli.

Uživatel má možnost přidané suroviny zrušit. Po uložení receptu budou přidané suroviny, které nebyly v databázi, do ní automaticky uloženy a recept bude zobrazen v seznamu receptů.

Vytvořit recept se stejným názvem nebo bez názvu nebude možné.

Objednavky Sklad Zákazníci Faktury Zaměstnanci Plán Výroby Odhlásit

←

Úprava receptu

Název receptu Čas přípravy v minutách

Postup

text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text
text text text text text text text text text text text text text text text text text
text

Suroviny

Množství Jednotka Název

počet jednotka surovina
počet jednotka surovina

Obrázek 20. Úprava receptu

Formulář pro úpravu receptu je shodný s formulářem pro jeho vytvoření, s tím rozdílem, že formulář pro úpravu je vyplněný informacemi upravujícího receptu a vypsane suroviny mají dvě tlačítka navíc. Jedno je pro jejich úpravu a druhé pro jejich smazání.

ZÁVĚR

Cílem této bakalářské práce bylo navrhnout informační systém pro správu receptů v potravinářské výrobě, konkrétně v pekárenství.

V teoretické části byly popsány základní principy informačních systémů, jejich složky a typy. Dále byl popsán UML a jeho diagramy, požadavky a ERD. Součástí teoretické části byla také rešerše informačních systémů v pekárenství na českém trhu, což poskytlo přehled o existujících řešeních a umožnilo identifikovat oblasti pro možné zlepšení.

V praktické části byla vypracována specifikace požadavků na informační systém pro správu receptů, včetně funkčních a nefunkčních požadavků. Byla provedena funkční a datová analýza problémové oblasti na základě identifikace klíčových entit a procesů spojených s řízením receptů. Tyto informace byly následně zpracovány do modelů systému pomocí diagramů UML, včetně diagramů tříd a procesů.

Zvláštní důraz byl kladen na návrh uživatelského rozhraní, který byl realizován pomocí drátěných modelů. Tyto modely poskytují představu o tom, jak budou uživatelé komunikovat se systémem a jak bude systém prezentovat informace o receptech.

Výsledkem práce je komplexní návrh informačního systému pro správu receptů, který pokrývá všechny klíčové aspekty procesu od vytváření až po úpravu receptů pro jejich využití v pekárenské výrobě v reálném prostředí. Tento návrh však stále vyžaduje další práci s implementací a testováním, přesto již poskytuje pevný základ pro vývoj efektivního a uživatelsky přívětivého informačního systému pro správu receptů v pekárenském průmyslu.

SEZNAM POUŽITÉ LITERATURY

- [1] O'BRIEN, James A. a George M. MARAKAS. Introduction to information systems. 15th ed. New York: McGraw-Hill Irwin, c2010, xxxiii, 592 s. International student edition. ISBN 9780070167087.
- [2] LAUDON, Kenneth C. a Jane Price LAUDON. Management information systems: managing the digital firm. Sixteenth edition. Harlow, England: Pearson, [2020], 652 s. ISBN 978-1-292-29656-2.
- [3] Chapter 1: What Is an Information System? - Information Systems for Business and Beyond (2019). Home - Information Systems for Business and Beyond [online]. Dostupné z: <https://opentextbook.site/informationssystem2019/chapter/chapter-1-what-is-an-information-system-information-systems-introduction/#footnote-5-1>
- [4] What is information systems? Definition, uses, and examples | Zapier. Zapier | Automation that moves you forward [online]. Copyright ©. Dostupné z: <https://zapier.com/blog/what-is-information-systems/#what>
- [5] Resources of Information System - GeeksforGeeks. GeeksforGeeks | A computer science portal for geeks [online]. Dostupné z: <https://www.geeksforgeeks.org/resources-of-information-system/>
- [6] Difference between Information and Data - GeeksforGeeks. GeeksforGeeks | A computer science portal for geeks [online]. Dostupné z: <https://www.geeksforgeeks.org/difference-between-information-and-data/>
- [7] Informační systémy v kostce: ERP, CRM, implementace. Experten in der digitalen Welt | Rascasone [online]. Copyright ©. Dostupné z: <https://www.rascasone.com/cs/blog/informacni-systemy-erp-crm-implementace#typy-informacni-systemu>
- [8] Podnikové informační systémy (EIS) a jejich funkce. Experten in der digitalen Welt | Rascasone [online]. Copyright ©. Dostupné z: <https://www.rascasone.com/cs/blog/podnikove-informacni-systemy-eis>
- [9] MES systém (Manufacturing Execution System) | MEScenter. [online]. Dostupné z: <http://www.mescenter.org/cz/clanky/5-co-je-to-mes-system>
- [10] Typy informačních systémů | ADVIN.cz. Úspěšné weby a e-shopy, webové IS | ADVIN.cz [online]. Dostupné z: <https://www.advin.cz/novinky/93-typy-informacnich-systemu>

- [11] Smart retail solutions: pekárna [online]. 2010. Dostupné z: <http://retailsystems.cz/pekarna2005.htm>
- [12] Odbyt a výroba pekárny: software pro řízení pekárny [online]. 2018. Dostupné z: <http://www.mcsoftware.cz/pekw/>
- [13] Přímé dodávky zboží. Vítejte na stránce společnosti PORS software a.s. Chrudim [online]. Copyright © PORS. Dostupné z: <https://www.pors-sw.cz/prime-dodavky-zbozi.html>
- [14] Objednávky prodejen. Vítejte na stránce společnosti PORS software a.s. Chrudim [online]. Copyright © PORS. Dostupné z: <https://www.pors-sw.cz/objednavky-prodejen.html>
- [15] Fis 3000 [online]. Dostupné z: <http://www.compex.cz/index.php?menu=software>
- [16] Just a moment.... ResearchGate | Find and share research [online]. Dostupné z: https://www.researchgate.net/figure/Overview-of-UML-diagram-types_fig1_325747903
- [17] ARLOW, Jim a Ila NEUSTADT. UML 2 a unifikovaný proces vývoje aplikací: objektově orientovaná analýza a návrh prakticky. 2., aktualiz. a dopl. vyd. Brno: Computer Press, 2007. ISBN 9788025115039.
- [18] FOWLER, Martin. Destilované UML. Praha: Grada, 2009. Knihovna programátora (Grada). ISBN 9788024720623.
- [19] SOMMERVILLE, Ian. Softwarové inženýrství. Brno: Computer Press, 2013. ISBN 9788025138267.
- [20] PILONE, Dan a Neil PITMAN. UML 2.0 in a nutshell. Sebastopol: O'Reilly, 2005. ISBN 0596007957.
- [21] UML Class Diagram Tutorial. Ideal Modeling & Diagramming Tool for Agile Team Collaboration [online]. Dostupné z: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial/>
- [22] What is Class Diagram?. Ideal Modeling & Diagramming Tool for Agile Team Collaboration [online]. Dostupné z: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/>
- [23] Microsoft Corporation. Microsoft Corporation [online]. Dostupné z: <https://www.microsoft.com/cs-cz/microsoft-365/business-insights-ideas/resources/guide-to-uml-diagramming-and-database-modeling>

- [24] What is Sequence Diagram?. Ideal Modeling & Diagramming Tool for Agile Team Collaboration [online]. Dostupné z: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/>
- [25] Sequence Diagram | Enterprise Architect User Guide. UML modeling tools for Business, Software, Systems and Architecture [online]. Copyright © 2000. Dostupné z: https://sparxsystems.com/enterprise_architect_user_guide/16.0/modeling_languages/sequencediagram.html
- [26] What is Use Case Diagram?. Ideal Modeling & Diagramming Tool for Agile Team Collaboration [online]. Dostupné z: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/>
- [27] Use Case Diagram | Enterprise Architect User Guide. UML modeling tools for Business, Software, Systems and Architecture [online]. Copyright © 2000. Dostupné z: https://sparxsystems.com/enterprise_architect_user_guide/16.0/modeling_languages/usecasediagram.html
- [28] UML Use Case Diagram Tutorial | Lucidchart. [online]. Copyright ©. Dostupné z: <https://www.lucidchart.com/pages/uml-use-case-diagram>
- [29] What is Entity Relationship Diagram (ERD)?. Ideal Modeling & Diagramming Tool for Agile Team Collaboration [online]. Dostupné z: <https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/>
- [30] ER Diagram (ERD) - Definition & Overview | Lucidchart. [online]. Copyright ©. Dostupné z: <https://www.lucidchart.com/pages/er-diagrams>
- [31] What are Requirements?. Requirements.com - All About Requirements [online]. Copyright ©2005, 2006, 2008, 2009. Dostupné z: <https://requirements.com/Content/What-is/what-are-requirements>
- [32] Requirements Definition: What is it and how to apply it? | Complete Guide - Visure Solutions. Requirements Management ALM Tool - Visure Solutions [online]. Copyright © 2023 Visure Solutions, Inc. Dostupné z: <https://visuresolutions.com/blog/requirements-definition/>
- [33] What Is a Wireframe? + How to Create One | Coursera. Coursera | Degrees, Certificates, & Free Online Courses [online]. Copyright © 2023 Coursera Inc. All rights reserved. Dostupné z: <https://www.coursera.org/articles/wireframe>

- [34] What Is a Wireframe? Why You Should Start Using This UX Design Tool | Lucid-chart Blog. [online]. Copyright ©. Dostupné z: <https://www.lucid-chart.com/blog/what-is-a-wireframe>
- [35] Informace o firmě PORS software a.s.. Vítejte na stránce společnosti PORS software a.s. Chrudim [online]. Copyright © PORS. Dostupné z: <https://pors-sw.cz/ofirme.html>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

API	Application Programming Interface (rozhraní pro programování aplikací)
atd.	a tak dále
CRM	Customer Relationship Management (správa vztahů se zákazníky)
CIO	Chief Information Officer (ředitel informačních technologií)
DIČ	Daňové identifikační číslo (identifikační číslo pro účely daní)
DPH	Daň z přidané hodnoty (daňový systém)
EIS	Executive Information System (informační systém pro vedení)
ERP	Enterprise Resource Planning (plánování prostředků podniku)
ERM	Enterprise Risk Management (řízení rizik podniku)
IČO	Identifikační číslo osoby (identifikační číslo pro právnické osoby)
KPI	Key Performance Indicator (klíčový ukazatel výkonnosti)
MDA	Model Driven Architecture (architektura řízená modelem)
MES	Manufacturing Execution System (systém pro řízení výroby)
OMG	Object Management Group (organizace pro správu objektů)
SCM	Supply Chain Management (řízení dodavatelského řetězce)
UML	Unified Modeling Language (sjednocený modelovací jazyk)

SEZNAM OBRÁZKŮ

Obrázek 1. Přehled všech UML diagramů [16]	25
Obrázek 2. Ukázka třídy	30
Obrázek 3. Ukázka enumeration	30
Obrázek 4. Ukázka asociace	31
Obrázek 5. Ukázka kompozice	32
Obrázek 6. Ukázka generalizace	32
Obrázek 7. Ukázka násobnosti [21]	33
Obrázek 8. Navrhnuté funkční požadavky	41
Obrázek 9. Návrh databáze pomocí ERD	49
Obrázek 10. Diagram tříd	55
Obrázek 11. Aktéři	60
Obrázek 12. Diagram případu užití	63
Obrázek 13. Matice vztahů	84
Obrázek 14. Sekvenční diagram vytvoření receptu	86
Obrázek 15. Sekvenční diagram vytvoření denní výroby	88
Obrázek 16. Menu informačního systému	90
Obrázek 17. Seznam receptů	91
Obrázek 18. Detail receptu	92
Obrázek 19. Vytvoření nového receptu	93
Obrázek 20. Úprava receptu	94

SEZNAM TABULEK

Tabulka 1. Vytvoření objednávky	64
Tabulka 2. Zrušení vytvoření objednávky	64
Tabulka 3. Vyhledání zákazníka	65
Tabulka 4. Zákazník neexistuje	65
Tabulka 5. Smazání objednávky	66
Tabulka 6. Zrušení smazání objednávky	66
Tabulka 7. Vytvoření zákazníka	67
Tabulka 8. Zrušení vytvoření zákazníka	67
Tabulka 9. Úprava skladových zásob	68
Tabulka 10. Vystavení dodacího listu	69
Tabulka 11. Úprava dodacího listu	70
Tabulka 12. Vyhledání objednávky	70
Tabulka 13. Úprava receptu	71
Tabulka 14. Úprava suroviny	72
Tabulka 15. Přidání suroviny	73
Tabulka 16. Surovina neexistuje	73
Tabulka 17. Vytvoření nové suroviny	74
Tabulka 18. Vytvoření nového receptu	75
Tabulka 19. Plánování denní výroby	76
Tabulka 20. Vystavení faktury	77
Tabulka 21. Vyhledání objednávky	78
Tabulka 22. Přidání zaměstnance	79
Tabulka 23. Vytvořit produkt	80
Tabulka 24. Zobrazení seznamu plánů denní výroby	81
Tabulka 25. Úprava plánu denní výroby	82
Tabulka 26. Zobrazení detailu denního plánu	83

SEZNAM PŘÍLOH

P I Obsah CD

PŘÍLOHA P I: OBSAH CD

- Bakalářská práce
- Zdrojový soubor enterprise architect
- Zdrojový soubor erdplus
- Obrázek wireframe