

UNIVERZITA PARDUBICE

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

BAKALÁŘSKÁ PRÁCE

2018

Jiří Někvinda

Univerzita Pardubice

Fakulta elektrotechniky a informatiky

Optimální procedury pro řízení skladu

Jiří Nekvinda

Bakalářská práce

2018

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jiří Nekvinda**  
Osobní číslo: **I13288**  
Studijní program: **B2646 Informační technologie**  
Studijní obor: **Informační technologie**  
Název tématu: **Optimální procedury pro řízení skladu**  
Zadávací katedra: **Katedra informačních technologií**

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

Cílem této práce je vytvoření funkční aplikace, která bude umožňovat řízení pekárny - skladu a objednávek surovin podle aktuální poptávky výrobků a ceny.

Aplikace bude umožňovat:

Registraci uživatelů systému.

Přístup uživatelů do systému podle jejich práv.

Evidenci objednávaných surovin potřebných pro výrobu výrobků (druhy surovin, váha, cena).

Evidenci dodavatelských firem, které dané suroviny nabízí (název, adresa, četnost dodávek).

Evidenci výrobků, které společnost vyrábí (název, váha, cena, skladba surovin).

Zpracování objednávek surovin z pohledu minimalizace nákladů.

Upozornění na minimální zásobu suroviny na skladu.

Výpočet pojistné zásoby.

Aplikace statistické kontroly kvality (indexy způsobilosti, statistická přejímka).

Generování sestav dle volitelně zadaných kritérií (sledování poptávky v různé kalendářní dny a měsíce).

V úvodní části je nutno provést rešerši systémů, které se zabývají touto problematikou. Rešerši je nutné doplnit o porovnání s nově navrhovaným systémem, který bude předmětem této práce. Praktická část musí obsahovat analýzu navrhovaného řešení, která bude obsahovat popis použitých technologií, návrh databáze, ER diagram a aplikační řešení. Pro vytvoření bude využit skriptovací jazyk PHP nebo JAVA a databáze MySQL nebo Oracle.

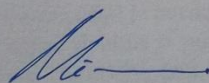
Rozsah grafických prací: 10  
Rozsah pracovní zprávy: 35  
Forma zpracování bakalářské práce: tištěná  
Seznam odborné literatury:

LINDA, B. Stochastické modely operačního výzkumu. Bratislava: Statis, 2004.  
ISBN 80-85659-33-6.

JABLONSKÝ, J. Operační výzkum. Praha: Professional Publishing, 2007. 323  
s. ISBN 978-80-86946-44-3.

Vedoucí bakalářské práce: Mgr. Jaroslav Marek, Ph.D.  
Katedra matematiky a fyziky

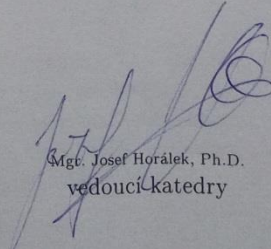
Datum zadání bakalářské práce: 31. října 2016  
Termín odevzdání bakalářské práce: 12. května 2017



Ing. Zdeněk Němec, Ph.D.  
děkan



L.S.



Mgr. Josef Horálek, Ph.D.  
vedoucí katedry

V Pardubicích dne 31. března 2017

## Prohlášení autora

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

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

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 9. 5. 2018

Jiří Někvianda

## **PODĚKOVÁNÍ**

Rád bych poděkoval vedoucímu své práce Mgr. Jaroslavovi Markovi, Ph.D. za velmi vstřícný přístup a cenné rady. Dále bych rád poděkoval rodině a přátelům za velkou podporu a motivaci.

## **ANOTACE**

Bakalářská práce se zaměřuje na vývoj funkční aplikace pro pekárnu, která umožní řízení, správu skladu a objednávek. V teoretické části jsou popsány typy architektur systémů, modely řízení skladu a existující podnikové systémy. Praktická část obsahuje analýzu navrhovaného řešení, popis použitých technologií, návrh databáze, ER diagram a aplikační příručku.

## **KLÍČOVÁ SLOVA**

Informační systém, pekárna, řízení, databáze, sklad

## **TITLE**

Optimal procedures for warehouse management

## **ANNOTATION**

The bachelor thesis focuses on the development of a functional application for a bakery, which facilitates management, warehouse management and ordering. The theoretical part describes system architecture types, warehouse management models and existing business systems. The practical part includes an analysis of the proposed solution, description of used technologies, database design, ER diagram and application manual.

## **KEYWORDS**

Information system, bakery, management, database, warehouse

# OBSAH

Úvod.....	12
1 Teoretická část .....	13
1.1 Podnikový informační systém.....	13
1.2 Technologické architektury.....	13
1.2.1 Dvouvrstvá architektura.....	13
1.2.2 Třívrstvá architektura.....	14
1.3 Modely a řízení zásob .....	14
1.3.1 Deterministické modely .....	15
1.3.2 Stochastické modely .....	18
1.4 Statistická kontrola kvality – Indexy způsobilosti a odhad podílu zmetků.....	19
1.4.1 Index $C_p$ .....	20
1.4.2 Index $C_{pk}$ .....	20
1.4.3 Index $C_{pm}$ .....	21
1.4.4 Odhad podílu zmetků.....	21
1.5 Analýza existujících řešení .....	22
1.5.1 SiStore.....	22
1.5.2 Altus Vario.....	23
1.5.3 IS RIS.....	25
1.5.4 Microsoft Dynamics NAV.....	26
1.5.5 Srovnání existujících řešení .....	26
2 Praktická Část .....	27
2.1 Návrh systému.....	27
2.2 Použité technologie .....	27
2.2.1 Java .....	27
2.2.2 NetBeans .....	27
2.2.3 MariaDB .....	27



2.2.4	XAMPP.....	28
2.3	Databáze.....	28
2.3.1	Logický model.....	28
2.3.2	Relační model.....	30
2.3.3	Tabulky.....	30
2.4	Uživatelské role.....	36
2.4.1	Správce systému.....	36
2.4.2	Operátor obchodu.....	36
2.4.3	Logistik.....	37
2.4.4	Operátor pekárny.....	38
2.4.5	Manager.....	38
2.5	Třídy v aplikaci.....	39
2.6	Matematické výpočty.....	40
2.6.1	Indexy způsobilosti.....	40
2.6.2	Nutriční hodnoty.....	41
2.6.3	Sklad a kontrola poptávky.....	42
2.7	Uživatelská příručka.....	43
2.7.1	Přihlášení.....	43
2.7.2	Registrace.....	44
2.7.3	IT.....	45
2.7.4	Pekárna.....	46
2.7.5	Výrobky.....	47
2.7.6	Sklad.....	48
2.8	Instalační příručka.....	49
3	Závěr.....	50
4	Použitá literatura.....	51
5	Přílohy.....	52

## SEZNAM ILUSTRACÍ A TABULEK

Obrázek 1 – Dvouvrstvá architektura. Zdroj: Vlastní.....	13
Obrázek 2 – Třívrstvá architektura. Zdroj: Vlastní.....	14
Obrázek 3 – Model I. Zdroj: Vlastní.....	15
Obrázek 4 – Model II. Zdroj: Vlastní .....	16
Obrázek 5 – Model III. Zdroj: Vlastní .....	17
Obrázek 6 – Model I. Zdroj: Vlastní.....	18
Obrázek 7 – Meze a cílová hodnota. Zdroj: Vlastní .....	19
Obrázek 8 – Charakteristiky a cílová hodnota. Zdroj: Vlastní .....	21
Obrázek 9 – Logický model. Zdroj: Vlastní .....	29
Obrázek 10 – Relační model. Zdroj: Vlastní .....	30
Obrázek 11 – Indexy způsobilosti. Zdroj: Vlastní .....	41
Obrázek 12 – Nutriční hodnoty. Zdroj: Vlastní .....	42
Obrázek 13 – Poptávka a stav skladu. Zdroj: Vlastní.....	43
Obrázek 14 – Přihlašovací obrazovka. Zdroj: Vlastní .....	44
Obrázek 15 – Registrace. Zdroj: Vlastní .....	45
Obrázek 16 – IT rozhraní. Zdroj: Vlastní .....	46
Obrázek 17 – Pekárna. Zdroj: Vlastní .....	47
Obrázek 18 – Výrobky. Zdroj: Vlastní .....	48
Obrázek 19 – Sklad. Zdroj: Vlastní .....	49
Tabulka 1 – Porovnání systémů. Zdroj: Vlastní .....	26
Tabulka 2 – Ukázka dat pro roli správce systému. Zdroj: Vlastní .....	36
Tabulka 3 – Ukázka dat pro roli operátor obchodu, výrobky. Zdroj: Vlastní .....	36
Tabulka 4 – Ukázka dat pro roli operátor obchodu, odběratelé. Zdroj: Vlastní .....	37
Tabulka 5 – Ukázka dat pro roli operátor obchodu, objednávka. Zdroj: Vlastní .....	37
Tabulka 6 – Ukázka dat pro roli logistik, dodavatelé. Zdroj: Vlastní .....	37
Tabulka 7 – Ukázka dat pro roli logistik, dodavatelé. Zdroj: Vlastní .....	38
Tabulka 8 – Ukázka dat pro roli oprátor pekárny, recept. Zdroj: Vlastní.....	38
Tabulka 9 – Ukázka dat pro roli manager. Zdroj: Vlastní .....	39

## **SEZNAM ZKRATEK A ZNAČEK**

ER	Entity-relationship
FIFO	First in first out
FEFO	First expired first out
SAP	Systems applications products
ČSN	Česká technická norma (dříve Československá státní norma)
VPN	Virtual private network
MS	Microsoft
SQL	Structured query language
ČNB	Česká národní banka
API	Application programming interface
PHP	Hypertext preprocessor
HTML	Hypertext markup language

## ÚVOD

Hlavním cílem této bakalářské práce je vytvořit funkční aplikaci, která bude sloužit k řízení pekárny, tudíž nemůže být bez úprav použita pro jiný druh firem. Aplikace umožňuje registraci uživatelů, evidenci surovin, dodavatelských firem, výrobků a objednávek. Mezi dílčí cíle patří vytvoření přehledného a uživatelsky přívětivého informačního systému. Ovládání by mělo být jednoduché a intuitivní bez nutnosti dlouhodobého školení.

Na začátku teoretické části práce je obecně popsán podnikový informační systém. Dále jsou představeny některé používané architektury informačních systémů. Další část popisuje modely používané pro řízení skladovaných zásob. Následující část popisuje metodiku statistické kontroly kvality a s tím související indexy způsobilosti. Poslední část obsahuje analýzu funkčních informačních systémů na trhu zabývajících se řízením firem a jejich shrnutí.

Praktická část obsahuje analýzu navrhovaného řešení a jeho implementaci. První část popisuje použité technologie navrhovaného informačního systému. Následující část je věnována přístupu do aplikace. Systém umožňuje registraci nových uživatelů a přístup do systému je omezen podle předem nastavených práv uživatelů. Další část je věnována popisu a návrhu databáze. Nachází se zde ER diagram sloužící pro znázornění ukládání dat do databáze a vzájemných vztahů mezi tabulkami. Následuje detailní popis všech tabulek, jejich klíčů a sloupců. Jsou zde také popsány všechny procedury vytvořené pro tuto aplikaci. Následující část popisuje jednotlivé třídy používané v aplikaci. Závěr praktické části je věnován uživatelské příručce a instalační příručce. Uživatelská příručka obsahuje detailní popis jednotlivých částí aplikace a slouží jako průvodce systémem. Instalační příručka popisuje všechny procedury nutné pro úspěšné zavedení systému.

# 1 TEORETICKÁ ČÁST

## 1.1 Podnikový informační systém

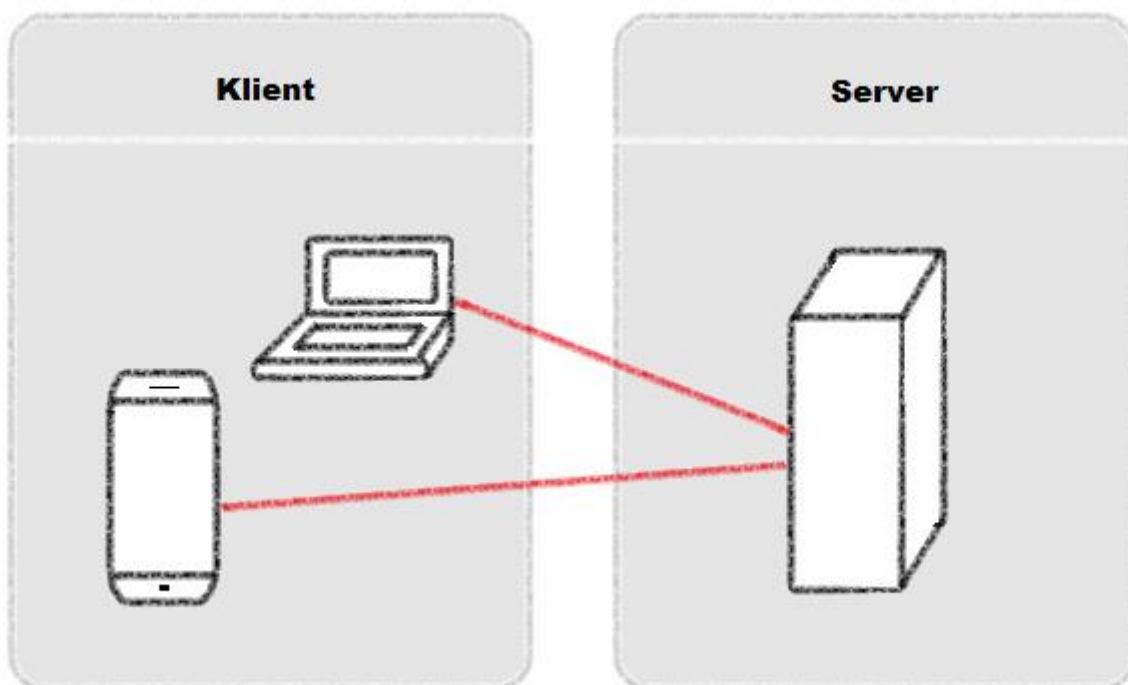
Informační systém je důležitý nástroj pro řízení firmy. Skládá se z hardwaru a softwaru a slouží pro sběr, zpracování a šíření dat o celém podniku. Díky dostupnosti těchto dat je možné efektivně řídit a plánovat firemní procesy.

## 1.2 Technologické architektury

V této podkapitole je popsány dva typy architektur, na kterých jsou dnešní informační systémy postaveny. Jako první je popsána dvouvrstvá architektura. Následně je popsána třívrstvá architektura, kterou dnes využívá mnoho aplikací.

### 1.2.1 Dvouvrstvá architektura

Dvouvrstvá architektura je také označována jako architektura klient-server. Klient zahrnuje uživatelské prostředí a všechny logické funkcionality. Server obsahuje relační databázi. Klient a server spolu vzájemně komunikují a předávají si data. Uživatelské dotazy klient překládá serveru a server vrací srozumitelné výsledky pro klienta, který je předkládá uživateli. Hlavní nevýhodou je skutečnost, že se s rostoucí aplikační částí zvyšují požadavky na hardware počítače nebo použitého zařízení. [6]

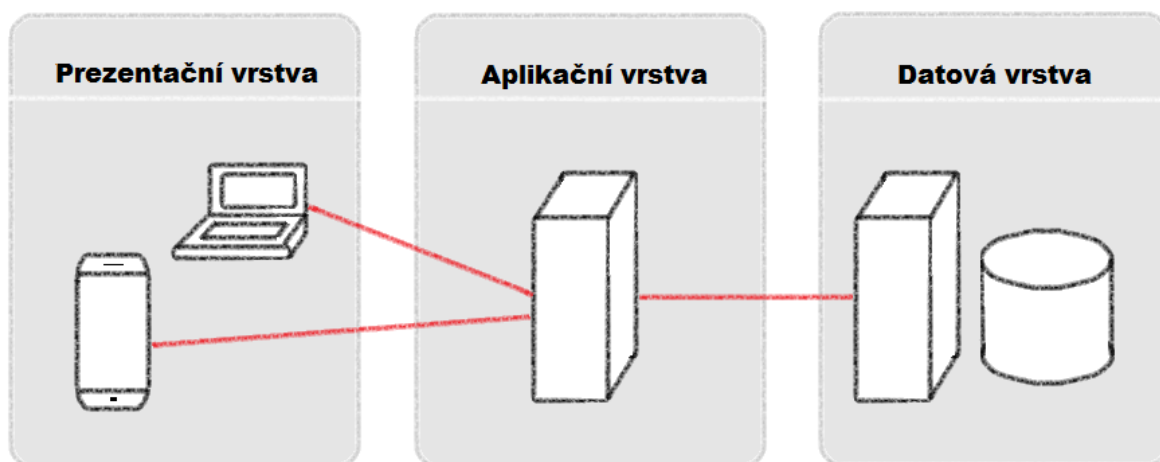


Obrázek 1 – Dvouvrstvá architektura. Zdroj: Vlastní

### 1.2.2 Třívrstvá architektura

Třívrstvá architektura v informatice znamená, že informační systém je složen z vrstev prezentační, aplikační a datové. Hlavní výhodou je, že jednotlivé vrstvy jsou oddělené a tudíž nezávislé na ostatních. V praxi ji využívá velké množství moderních aplikací, portálových řešení a webových stránek. Jedná se o moderní řešení robustních systémů. [7]

- Prezentační vrstva – Je označována jako viditelná vrstva. Slouží pro uživatele a stará se o vstupní požadavky a prezentaci výsledků. Je závislá na platformě, pro kterou byla vytvořena. [7]
- Aplikační vrstva – Je také nazývána jako aplikační server. Tato vrstva slouží pro výpočty a operace s vstupními a výstupními potřebami a daty. [7]
- Datová vrstva – Tato vrstva zajišťuje práci s daty a jejich řízení. Používá se například pro ukládání, výběr a zpracování. [7]



Obrázek 2 – Třívrstvá architektura. Zdroj: Vlastní

### 1.3 Modely a řízení zásob

Optimalizace zásob je velmi důležitá věc pro efektivní fungování podniku. V zásobách je uložena velká část firemního kapitálu. Díky modelům pro řízení zásob se sníží část vázaných peněz a tím se sníží i náklady. [1]

Jeden z hlavních ukazatelů pro řízení je charakteristika poptávky. Poptávka může být deterministická nebo stochastická. Deterministická poptávka znamená, že je poptávka za určitý časový úsek pevně daná. Naopak stochastická poptávka není pevně daná a její velikost lze jen odhadnout. [1]

Pro proces rozhodování je také důležité zamyslet se nad nedostatkem zásob, a jestli je tento stav v nějakém okamžiku přijatelný. Tento stav lze ovlivnit vytvořením pojistné zásoby. Pojistná zásoba by měla vynahradit neočekávaný výkyv dodávky zboží dodavatelem nebo neplánovanou změnu v objednávce. Například pokud bychom chtěli uspokojit poptávku na 95 %, lze ji vypočítat takto:

$$pz \geq z_{0.95} \times \sigma(f),$$

kde

$pz$  – pojistná zásoba,

$z_{0.95}$  – kvantil normálního rozdělení,

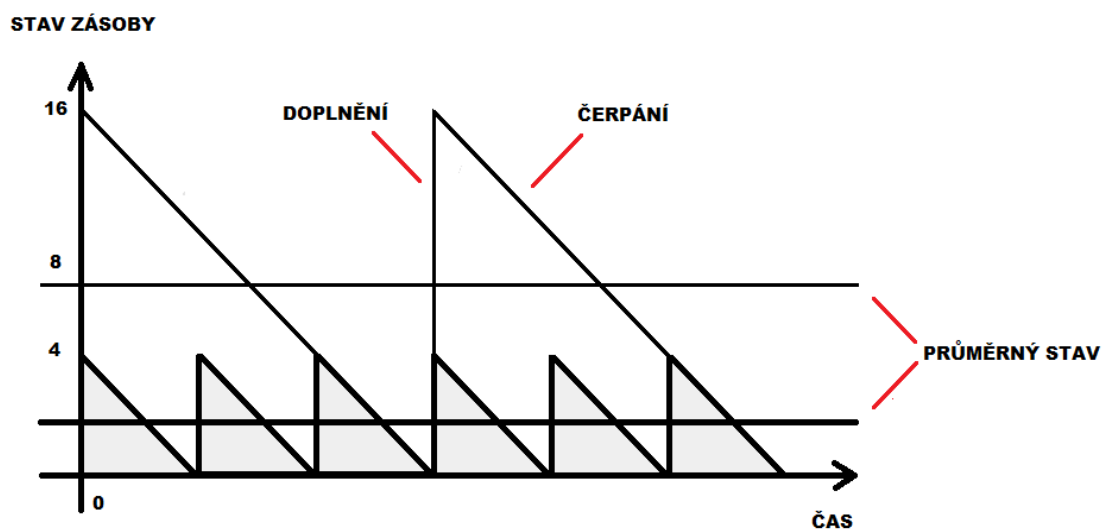
$\sigma(f)$  – směrodatná odchylka poptávky. Viz [1].

### 1.3.1 Deterministické modely

- **Model I. optimální velikost objednávky**

Tento model uvažuje několik předpokladů. Poptávka, dodací lhůta dodávek a jejich velikosti jsou v čase konstantní. Čerpání zásob je rovnoměrné. Cena objednávky není závislá na množstevních slevách. K doplnění skladových zásob dochází v jednom okamžiku a není povolen stav nedostatku zásoby. [1]

Obrázek 3 znázorňuje dvě různé možnosti doplňování zásob. První možnost značí velké objednávky s malou četností ale s velkými skladovacími náklady. Druhou možností je snížení velikosti objednávky s častějšími dodávkami a tím i nízkými skladovacími náklady. [1]

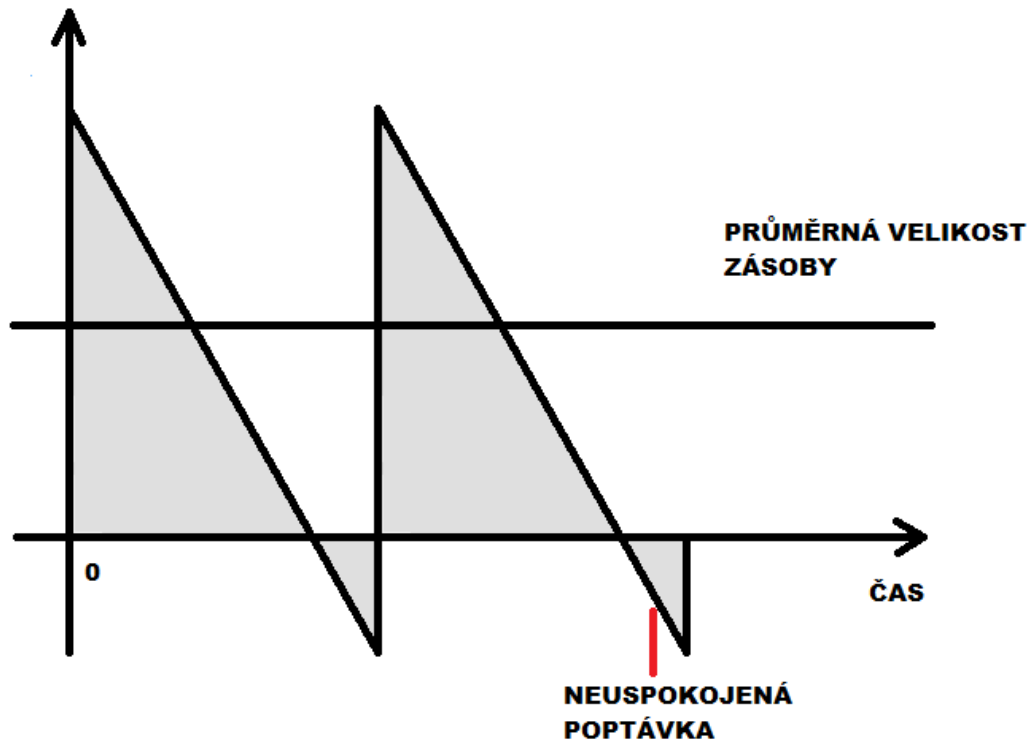


Obrázek 3 – Model I. Zdroj: Vlastní

- **Model II. přechodné neuspokojení poptávky**

Tento model se liší od předchozího modelu jen v předpokladu, že je povolen stav nedostatku zásoby. Znamená to, že poptávka může být dočasně neuspokojená. Zároveň se počítá s tím, že poptávka bude uspokojena s nejbližší dodávkou zásob. [1]

**STAV ZÁSoby**

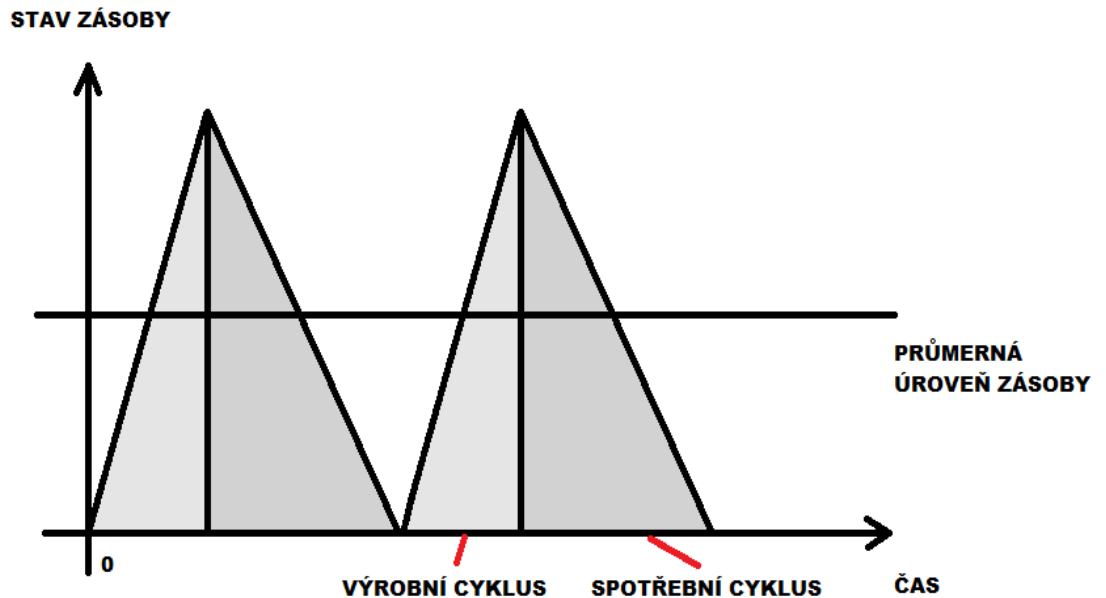


Obrázek 4 – Model II. Zdroj: Vlastní

- **Model III. produkční model**

Produkční model je podobný jako model I., ale liší se v doplnění skladových zásob. U tohoto modelu není dodávka jednorázová v jednom okamžiku, ale je rozdělena do výrobního a spotřebního cyklu. Při výrobním cyklu je sklad rovnoměrně doplňován, ale i vybírán. Ve spotřebním cyklu se jen vybírá zásoba ze skladu. Důležité je zde stanovit objem výrobní dodávky a interval mezi cykly. [1]





Obrázek 5 – Model III. Zdroj: Vlastní

Optimální objem výrobní dávky lze vypočítat ze vzorce:

$$q = \sqrt{\frac{2 \times Q \times c_2}{c_1}} \times \sqrt{\frac{p}{p-h}}$$

kde

$q$  – velikost dávky,

$Q$  – poptávka za celý rok,

$c_2$  – fixní náklady jedné dávky,

$c_1$  – skladovací náklady jedné jednotky za rok,

$p$  – intenzita produkce za zvolenou jednotku času,

$h$  – intenzita poptávky za jednotku času. Viz [1].

- **Model IV. množstevní rabaty**

Při tomto modelu vycházíme z předpokladu, že cena objednávky zboží je závislá na velikosti objednávky. Dodavatel nabízí množstevní slevy a tím se sníží pořizovací a skladovací náklady. Skladovací náklady jsou často určovány procentem z nákupní ceny. Je tedy velice důležité spočítat si optimální velikost objednávky ze vzorce:

$$q = \sqrt{\frac{2 \times Q \times c_2}{c_1}}$$

kde

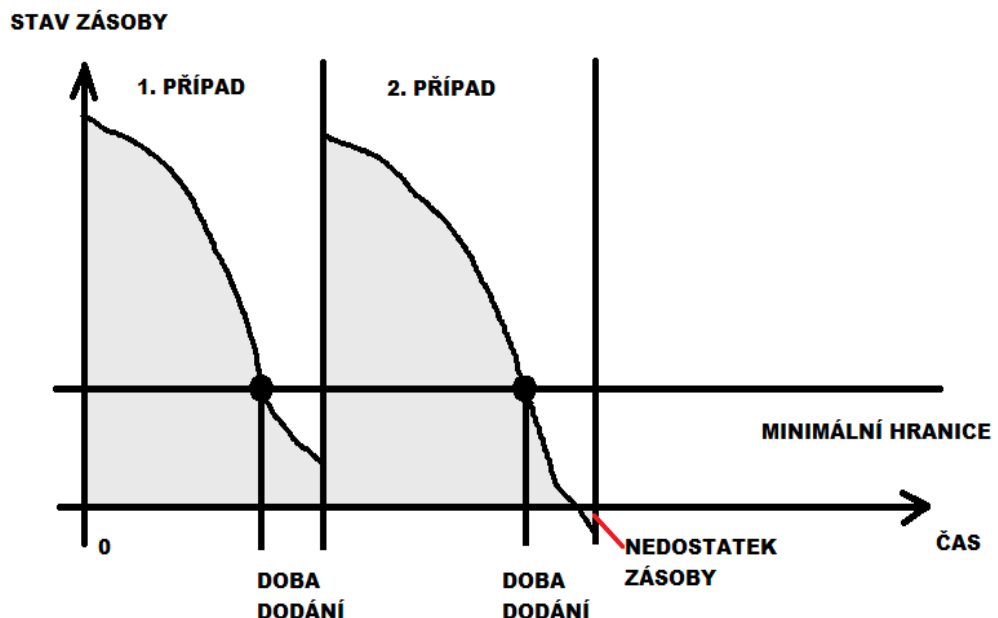
- $q$  – velikost objednávky,
- $Q$  – poptávka za celý rok,
- $c_2$  – náklady na jednu objednávku,
- $c_1$  – skladovací náklady. [1]

Optimální velikost objednávky lze pak poznat podle nejnižších nákladů. Viz [1].

### 1.3.2 Stochastické modely

- **Model I. stochastická spojitá poptávka**

U tohoto modelu je poptávkou náhodná veličina. Hlavní předpoklad u tohoto modelu je, že máme systém nepřetržitě sledující stav zásob a pro objednání zboží je nutno dosáhnout určitého minimálního stavu zboží. Období mezi objednáním a dodáním zboží je zde považováno za konstantní. Díky proměnlivé poptávce mohou nastat dvě možnosti. První případ, který může nastat, je situace, kdy nová dodávka dorazí na sklad a stav skladu je kladný. Nevznikne zde problém s plněním závazků. V druhém případě dochází k vyčerpání stavu na skladě a vznikne problém s plněním výrobních plánů. Pokud bychom se chtěli tomuto případu vyhnout, je nutné vytvořit pojistnou zásobu a upravit hranici minimálního stavu zboží. [1]



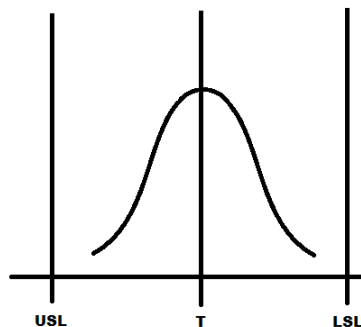
Obrázek 6 – Model I. Zdroj: Vlastní

- **Model II. optimalizace jednorázově vytvářené zásoby**

Tento model se používá pro vytvoření jednorázové zásoby na začátku určitého období. Dodatečné doplnění v průběhu není možné. Pro popis poptávky lze využít zkušeností z minulých období, neboť tento model se většinou používá u sezonního zboží. Jelikož u tohoto modelu není poptávka přesně určená, mohou díky tomu nastat tři případy. První případ nastane, pokud skutečná poptávka je v daném období nízká a zboží zůstane na skladě. Je pak rozumné zbylé zboží prodat ve slevě se ztrátou, než čekat na další období. Druhý případ nastává, když je skutečná poptávka za dané období vyšší. Vzniká zde ztráta z ušlého zisku. Třetí případ vznikne, pokud je skutečná poptávka rovna našemu předpokladu. Tato situace je spíše nereálná a v praxi se moc nevyskytuje. Nevznikají zde náklady ani ztráty. Viz [1].

#### 1.4 Statistická kontrola kvality – Indexy způsobilosti a odhad podílu zmetků

Pojem způsobilost označuje vlastnost, kdy například výrobní proces dosahuje stanovených měřítek kvality. U způsobilého výrobního procesu je pak velmi malá pravděpodobnost vzniku neshodného výrobku (jehož sledovaný znak kvality neleží v tolerančních mezích). Hlavními cíli hodnocení způsobilosti je určení hodnoty jakosti a míra proměnlivosti v okolí cílové hodnoty. Hodnocení je provedeno výpočtem vybraných indexů způsobilosti. Existuje více typů těchto indexů a pro spolehlivé výsledky je nutné splnit předpoklady pro jednotlivé typy.



Obrázek 7 – Meze a cílová hodnota. Zdroj: Vlastní

První rozlišení typů určuje charakteristika a to jestli je měřitelná. Pokud ano, lze další určení vyhodnotit podle typu rozdělení. Charakteristika je určena střední hodnotou  $\mu$  a směrodatnou odchylkou  $\sigma$ . Princip konstrukce je u všech typů stejný a je založen na

poměru mezi předepsanou přesností a skutečnou přesností výroby. Předepsaná přesnost je stanovena podle:

- horní hranicí (USL),
- dolní hranicí (LSL),
- cílovou hodnotou (T). [10]

#### 1.4.1 Index $C_p$

Index  $C_p$  je nejstarší a nejznámější index způsobilosti je definován jako:

$$C_p = \frac{USL - LSL}{6 \times \sigma}.$$

Tento vztah platí pouze pro normální rozdělení. Je tudíž nejdříve nutné vyhodnotit soubor dat a korektně určit jeho rozdělení. Obecně platí, že směrodatná odchylka  $\sigma$  není k dispozici a je ji tedy nutné nahradit výběrovou odchylkou  $s$ . Tímto nahrazením dostáváme odhad  $C_p$ . Převrácením hodnoty  $C_p$  získáme procentuální využití tolerančního intervalu. Při indexu  $C_p = 1$  leží  $\mu$  je ve středu tolerančních mezí a celkový počet 99,73 % výrobků je kvalitních. Norma ČSN ISO 8258 udává, že tato hodnota je nezpůsobilá. Pro způsobilost se tedy požaduje hodnota  $C_p > 1$ . V praxi je tato minimální hodnota stanovena na  $C_p = 1,33$ . [10]

#### 1.4.2 Index $C_{pk}$

Index  $C_{pk}$  vznikl kvůli požadavkům na obecnost hodnocení. Díky tomu umožňuje charakteristika sledovat jeho polohu v nastaveném tolerančním intervalu. V praxi může index vycházet i záporně. Značí to, že proces je mimo kontrolu. Pro výpočet při neomezeném horním intervalu se využívá tento vztah:

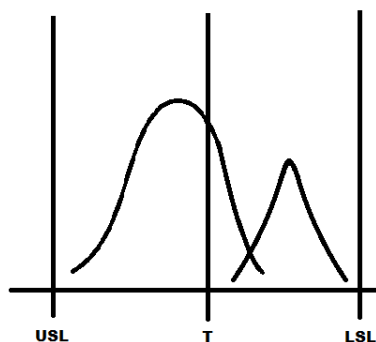
$$C_{pK} = C_{pL} = \frac{\mu - LSL}{3 \times \sigma}.$$

Pro výpočet při neomezeném dolním intervalu platí:

$$C_{pK} = C_{pU} = \frac{USL - \mu}{3 \times \sigma}.$$

Pro výpočet při oboustranně omezeného intervalu platí:

$$C_{pK} = \min\{C_{pL}, C_{pU}\}. [10]$$



Obrázek 8 – Charakteristiky a cílová hodnota. Zdroj: Vlastní

### 1.4.3 Index $C_{pm}$

Jedním z dalších indexů je  $C_{pm}$ . Index měří diferenci střední hodnoty a cílové hodnoty. Zohledňuje proměnlivost hodnot jakosti ale i míru dosažení optimální hodnoty. Měl by být používán pro případy, kdy cílová hodnota leží uprostřed tolerančního pole. [10]

$$C_{pm} = \frac{\min(USL-T, T-LSL)}{3 \times \sqrt{\sigma^2 + (\mu - T)^2}}$$

Pro výpočet při neomezené dolní toleranční mezi platí:

$$C_{pm} = \frac{USL - T}{3 \times \sqrt{\sigma^2 + (\mu - T)^2}}$$

Pro výpočet při neomezené horní toleranční mezi platí:

$$C_{pm} = \frac{T - LSL}{3 \times \sqrt{\sigma^2 + (\mu - T)^2}} \quad [10]$$

### 1.4.4 Odhad podílu zmetků

Podíl zmetků je často vyjádřen vzhledem k milionu kusů výrobků. Tento ukazatel má přímou souvislost s indexem způsobilosti. Díky této souvislosti lze stanovit přípustné procento zmetků výroby. Toto procento je označováno jako NC (non conform product). NC pro symetrickou toleranci je vyjádřeno jako:

$$NC = 2 \times \Phi(3C_p).$$

Pro nesymetrickou toleranci je vztah takto:

$$NC = \Phi(-3C_{pL}) + \Phi(-3C_{pU}),$$

kde

$\Phi$  – je distribuční funkce standardizovaného normálního rozdělení. [10]

## 1.5 Analýza existujících řešení

V této podkapitole budou analyzovány vybrané informační systémy, které se zabývají podobnou problematikou. Byli vybráni čtyři dodavatelé těchto systémů s různými použitými technologiemi. Jedná se o systémy SiStore, Altus Vario, IS RIS a MS Dynamics NAV.

### 1.5.1 SiStore

Jedná se o logistický informační systém od společnosti APSYS CZ s.r.o., která se zaměřuje na efektivně řízenou logistiku a výrobu. SiStore umožňuje on-line řízení skladů, procesů expedic, příjmů, kvality a lidí. Využívá mobilních terminálů s podporou čárových kódů. [2]

Tento systém je rozdělen na moduly:

- SiStore base

Tento modul zajišťuje základní procesy a ty jsou:

- tisk etiket,
  - sklady,
  - adresné příjmy a výdeje,
  - on-line kontroly,
  - přeskladňování mezi sklady,
  - hlídání minimálních a maximálních zásob,
  - šaržování,
  - FIFO a FEFO režim,
  - balící předpisy,
  - přebalování,
  - blokaci materiálu,
  - uvolňování materiálu,
  - a základní reporting. [2]
- Řízené skladování – Warehouse Management System naviguje operátora skladu na určité místo, kde je potřeba materiál naskladnit nebo vyzvednout. [2]
  - Odbyt a transport – SD modul. [2]
  - SCM Supply Chain Management – Slouží pro integraci dodavatelů. [2]
  - Early Warning System – Tento modul je systém včasného varování, který slouží pro monitorování nastavených hodnot. Při překročení nebo podkročení stavu je systémem kontaktován odpovědný pracovník a to pomocí emailu nebo sms. [2]
  - T.E.H. – Modul slouží pro nastavení doplňkové reakce vyvolané operací s terminálem. Velkou výhodou tohoto modulu je nastavení několika akcí na jednu událost. Konfigurace se provádí pomocí xml souboru v textovém editoru. Například je možno využít pro

odeslání elektronického dodacího listu zákazníkovi při ukončení nakládky a odjezdu auta. [2]

- SAP RFC connector – Tento modul je integrované rozhraní sloužící k obousměrné on-line komunikaci se systémem SAP ale může pracovat i v off-line režimu. Při výpadku se vytvoří synchronizační dávka, která je zpracována pro zprovoznění systému. [2]
- SiStore Exchange – Modul SiStore Exchange je univerzální systém sloužící pro výměnu dat mezi různými platformami a systémy. Zajišťuje kompletní výměny dat mezi serverem a cílovou stanicí. Využívá zálohovaný SiStore Exchange Server tudíž je nezávislý na dodavateli. Tato služba běží automaticky a nevyžaduje administraci. [2]

Hlavní výhody jsou:

- vysoká bezpečnost bez použití VPN,
- minimální nároky,
- nezávislá dávková výměna,
- komprimace a možnost šifrování dat. [2]

### **1.5.2 Altus Vario**

Altus Vario je informační systém od firmy Altus software s.r.o., který se zaměřuje na intuitivní a jednoduché ovládání. Jeho primární určení je pro firmy střední velikosti. Aplikace byla navržena v Microsoft Access a pro sdílení dat je využíván MS SQL Server. Tento systém lze rozšiřovat nebo doplňovat pomocí programátorského prostředí Visual Basic for Applications. Díky prostředí MS Office uživatel využívá programy Word a Excel. [3]

Systém se dělí na několik modulů:

- Katalog zboží – Modul slouží pro vytváření katalogových stromových struktur zboží a služeb. Produkty jsou děleny do kategorií a u každého produktu jsou zaznamenávány údaje podle potřeby uživatelů. [3]
- Nákup – Tento důležitý modul slouží pro řízení nákupu zboží. Nákup je provázán s přijatými objednávkami a nákupními strategiemi. Objednávání je založeno na plánu, který zahrnuje skladovou zásobu, dodací lhůty a požadované termíny dodání.
- Sklady – Pomocí toho modulu na správu skladu lze řídit libovolný počet skladů. Řízení zásob probíhá podle nastavených objednacích strategií. Modul je možno využívat samostatně nebo s propojením na e-shop. [3]
- CRM – Tento modul se stará o řízení vztahů k zákazníkům. Zahrnuje řízení obchodních kampaní, správu obchodních příležitostí a kategorizaci zákazníků. Centrálně eviduje

všechny dostupné údaje o zákaznících. Cílem je zvýšit loajalitu zákazníků a ziskovost podniku. [3]

- Prodej – Tento systém je určen pro prodej zboží a služeb a zároveň je provázán s CRM. Systém vytváří katalogy zboží a to dále může být řazeno do stromových kategorií. Prodejní cena je zapisována do katalogu nebo je možné vytvořit systém cenových kategorií a slev. [3]
- e-shop – Systém je možno propojit s internetovým obchodem nebo s maloobchodní prodejnou. Pomocí systému pro správu e-shopu je nabízen pro zákazníky katalog zboží. [3]
- Výroba – Tento modul slouží pro plánování a řízení výroby. Plánování je závislé na optimalizaci výrobních procesů, správného výpočtu potřeb suroviny a poskytování aktuálních informací. [3]
- Servis – Modul slouží pro správu údržby vlastních a cizích zařízení. Dále pro servisy nebo půjčovnu automobilů. [3]
- Služby – Tento systém řeší reklamace od zákazníků směrem k dodavatelům a zároveň nabízí i evidenci smluv. [3]
- Banka – Tento modul slouží pro automatickou práci s doklady. Přijaté a odeslané platby se automaticky propojí s fakturami. [3]
- Pokladna – Pomocí tohoto systému je možno pracovat s peněžní hotovostí. Slouží pro proplácení faktur nebo pro přímý prodej ze skladu. [3]
- Majetek – Tento modul slouží pro evidenci hmotného a nehmotného majetku firmy. U každého majetku se evidují plány odpisů, zhodnocení a úpravy cen. Vše je možné filtrovat podle různých období. [3]
- Kursovní lístek – V modulu se nachází evidence cizích měn s hodnotami jejich kursů. Data jsou automaticky načítána z kursovního lístku ČNB. [3]
- Účetnictví – Tento modul slouží pro vedení účetnictví. Legislativní správnost je zaručena pravidelnými aktualizacemi systému. [3]
- Mzdy – Pomocí tohoto systému je zaměstnanci vypočítávána mzda díky obsaženým funkcím s automatickým zaúčtováním do účetnictví. [3]
- Personalistika – Tento modul slouží hlavně pro evidenci všech informací o zaměstnancích. Je evidována jejich kvalifikace, pracovní hodnocení a osvědčení pro vykonávanou práci. [3]
- Komunikace – Modul slouží pro usnadnění komunikace mezi úřady, zákazníky a obchodními partnery. Využívá datovou schránku nebo evidenci přijaté a odeslané pošty. [3]



- Řízení – Tento modul je určen pro řízení firmy. Umožňuje delegovat práci zaměstnancům a analyzovat firemní data. [3]
- Audit – Modul slouží pro kontrolu firmy a správu firemních dokumentů. Obsahuje deník událostí, kde je možné uchovat telefonáty a jednání. Dále obsahuje knihu jízd pro snadnou evidenci jízd firemních motorových vozidel. [3]

### 1.5.3 IS RIS

RIS je informační systém vyvíjen firmou Saul informační systémy s.r.o. Lze ho získat jako systém, který využívá dvouvrstvou nebo třívrstvou architekturu. Uživatelské rozhraní je vytvořeno ve vývojovém prostředí PowerBuilder a je určeno pro stanice s operačním systémem Windows. Pro ukládání dat je možné vybrat z těchto relačních databázových systémů:

- SYBASE,
- ORACLE,
- MS SQL 2000
- NEBO INFROMIX. [4]

Systém se dělí na několik modulů:

- Účetnictví – Tento modul slouží pro vedení účetnictví celé společnosti a obsahuje základní funkce jako je kniha závazků a pohledávek, pokladní knihu, bankovní operace, vnitřní doklady a další. [4]
- Manažerské řízení – Modul zpracovává data z ostatních částí systému pro finanční a obchodní manažery. Výstupem jsou statistiky a modifikovatelné přehledy. [4]
- Majetek – Tento modul slouží pro evidenci majetku firmy. Evidují se například vstupní ceny, plány účetních odpisů, stav a další charakteristiky. [4]
- Logistika a sklad – Pomocí toho modulu se vedou záznamy o zásobách na skladě a zároveň slouží pro řízení zásob. [4]
- Plánování a řízení výroby – Řízení výroby je v modulu určeno podle známého výrobního plánu. Výrobky jsou sdružovány do kusovníku dle výrobní dávky. [4]
- Zakázky
- Poptávky
- Správa dokumentů
- Mzdy
- Oborová řešení
- E-shop

### 1.5.4 Microsoft Dynamics NAV

NAV dříve Navision je informační software pro správu a řízení podniku od společnosti Microsoft určený pro malé a středně velké organizace. Společnost Microsoft koupila dánskou společnost Navision A/S a tím vznikla divize Microsoft Business Solutions. NAV funguje na třívrstvé architektuře klienta a uživatelské rozhraní je zaměřeno na pracovní role. Pro sdílení dat je využíván Microsoft SQL Server a podporované klientské operační systémy jsou typu Windows. Klíčové funkce jsou:

- správa ekonomiky firmy a účetnictví,
- řízení dodavatelského řetězce, výroby a provozu,
- marketing, obchod a servis,
- projektový management,
- business intelligence a reporting,
- správa měn. [5]

### 1.5.5 Srovnání existujících řešení

U všech řešení je informační systém rozdělen na podobné moduly, které obsahují podobné funkce pro informační systém podniku. Moduly lze snadno modifikovat a rozšiřovat podle přání zákazníků. Všechny systémy podporují mobilní terminály a čtečky čárových kódů.

Tabulka 1 – Porovnání systémů. Zdroj: Vlastní

	SiStore	Altus Vario	RIS	MS Dynamics NAV
MODULY	ANO	ANO	ANO	ANO
TERMINÁLY	ANO	ANO	ANO	ANO
DATABÁZE	–	MS SQL	ORACLE, MS SQL	MS SQL
OPERAČNÍ SYSTÉM	–	WINDOWS	WINDOWS	WINDOWS

## 2 PRAKTICKÁ ČÁST

### 2.1 Návrh systému

Navrhovaný systém je rozdělen na moduly, které jsou navrženy tak, aby obsáhly všechny zdrojová data. Část jich je možno vidět v Příloze A – *Zdrojová data*. Přístup do jednotlivých modulů je omezen jen pro uživatele s daným právem přístupu. Pro uchování jednotlivých uživatelů a všech dat potřebných pro aplikaci je využita relační databáze MySQL, která je umístěna na serveru. Aplikace je napsána v jazyce Java, který je zvolen z důvodu jednoduché přenositelnosti na jiné platformy.

### 2.2 Použité technologie

Tato podkapitola popisuje technologie použité při vývoji aplikace. Nejdříve je popsán použitý programovací jazyk a vývojové prostředí. Následuje zvolená relační databáze a server nutný pro odzkoušení a optimalizaci databáze.

#### 2.2.1 Java

Jazyk Java je objektově orientovaný programovací jazyk a byl představen firmou Sun Microsystems v roce 1995. Tento jazyk byl vyvinut pro potřeby, ve kterých jazyk C++ nebyl dostačující. Jde o jeden z nejpoužívanějších programovacích jazyků na světě. [11]

Na rozdíl od programovacího jazyku C++ je zde zjednodušená syntaxe a tím by měl být kód více čitelný. Zároveň díky své přenositelnosti je používán pro programy, které mají běžet na různých systémech. [11]

#### 2.2.2 NetBeans

NetBeans je celosvětově rozšířený projekt spadající pod licenci GNU GPL V2. Byl založen v roce 2000 firmou Sun Microsystems. Dnes se skládá z vývojového prostředí NetBeans IDE a z vývojové platformy NetBeans Platform. Samotné prostředí NetBeans IDE je vytvářeno v jazyce Java, ale podporuje prakticky jakýkoliv programovací jazyk jako třeba PHP, HTML a další jazyky. [8]

#### 2.2.3 MariaDB

MariaDB je relační databáze, spadající pod licenci svobodného softwaru GNU GPL. Vyvinuta byla vývojáři v roce 2009 jako alternativa k MySQL, kterou získala společnost Oracle. Dnes má již více než 12 milionů uživatelů. Používá ji například Google, Fedora a Wikipedia. API a protokoly MariaDB jsou kompatibilní s protokoly používanými MySQL.

To značí, že konektory, knihovny a aplikace, které pracovaly se serverem MySQL, by měly pracovat i s MariaDB. [12]

#### **2.2.4 XAMPP**

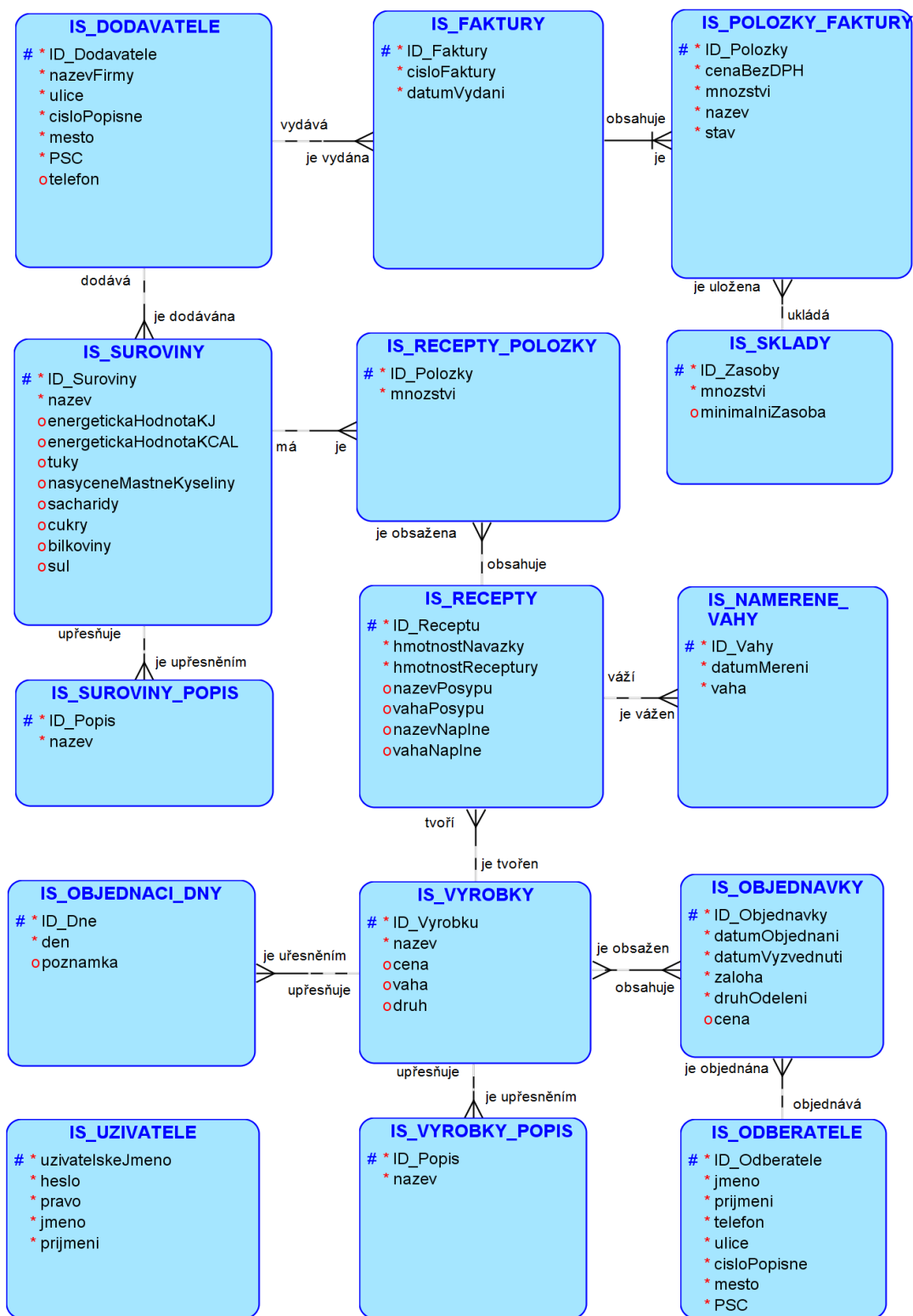
XAMPP je softwarový balíček nezávislý na hardwaru a softwaru. Balíček obsahuje volně dostupný software a to server Apache, databázi MySQL a interprety jazyků PHP a Perl. Od roku 2015 byla MySQL nahrazena MariaDB. XAMPP byl vyvinut jako nástroj pro návrháře a programátory, kteří potřebovali otestovat svojí práci bez přístupu k internetu. [9]

### **2.3 Databáze**

V této kapitole je popsána databázová implementace. V podkapitolách jsou nejdříve popsány a zobrazeny jednotlivé modely a následně se zde nachází popis všech tabulek vytvořených a zároveň používaných v této navržené aplikaci.

#### **2.3.1 Logický model**

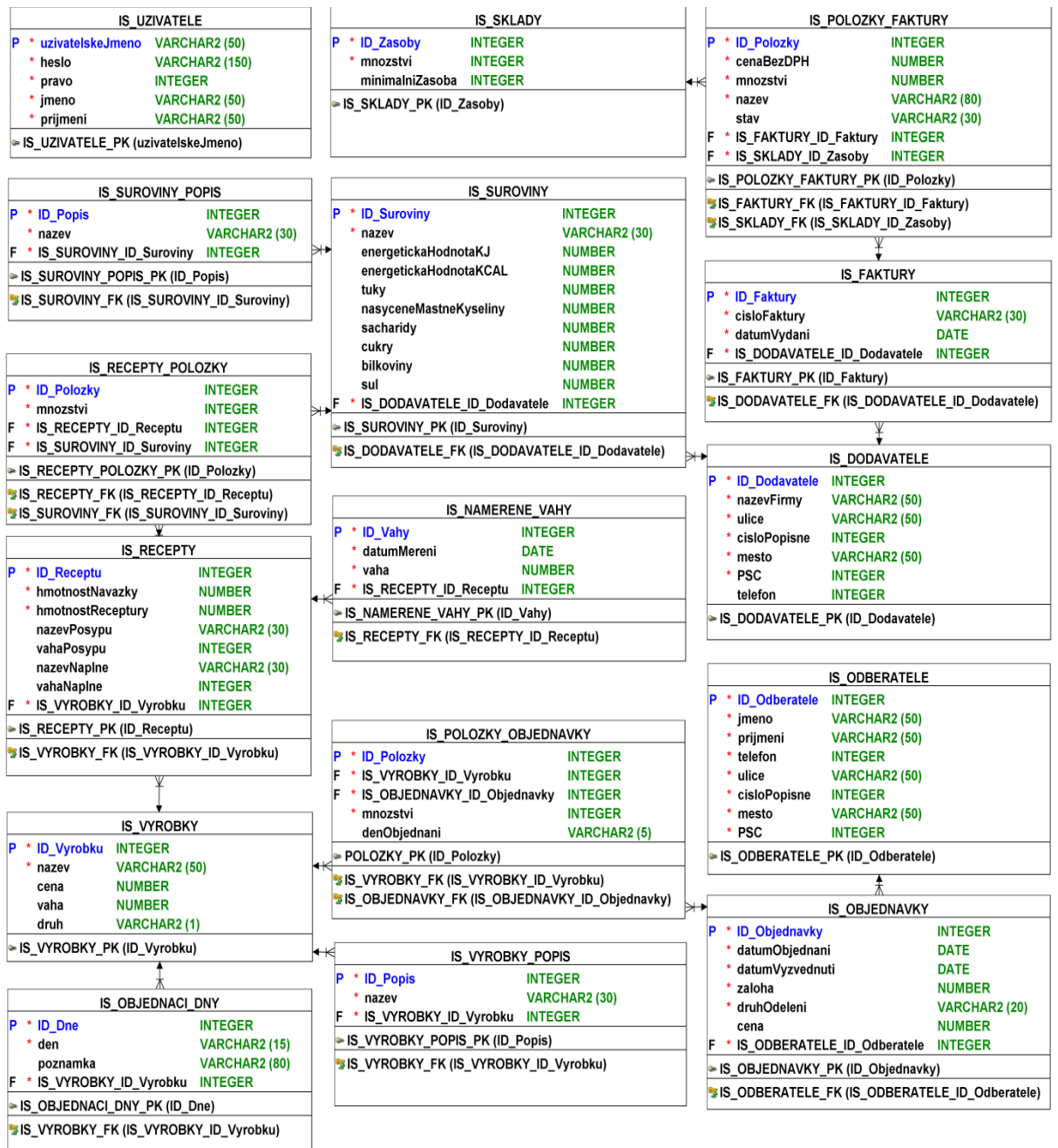
Na následujícím obrázku je zobrazen použitý návrh logického modelu pro ukládání dat do databáze. Model je složen z patnácti entit, tedy tabulek, které jsou kromě jedné tabulky vzájemně provázány. Tato tabulka nemá žádnou návaznost na ostatní a jedná se o tabulku IS\_UZIVATELE. Na obrázku 9 je dále vidět vazba typu M:N mezi tabulkami IS\_VYROBKY a IS\_OBJEDNAVKY, která bude odstraněna přidáním vazební tabulky.



Obrázek 9 – Logický model. Zdroj: Vlastní

## 2.3.2 Relační model

Na následujícím obrázku je zobrazen použitý návrh kompletního relačního modelu, obsahujícího šestnáct tabulek, z kterého byl vygenerován skript pro vytvoření tabulek v databázi.



Obrázek 10 – Relační model. Zdroj: Vlastní

## 2.3.3 Tabulky

Pro tento informační systém bylo navrženo a použito šestnáct tabulek, sloužících pro ukládání dat systému. Tato podkapitola popisuje všechny tabulky a jejich vazby pomocí cizích klíčů.

Tabulka IS\_UZIVATELE slouží k uchování zaregistrovaných uživatelů systému a obsahuje:

- *uzivatelskeJmeno*,
- *heslo*,
- *pravo*,
- *jmeno*,
- *prijmeni*.

Položka *uzivatelskeJmeno* je primární klíč. Sloupec *prava* obsahuje hodnoty, které určují role uživatelů. Do položky *heslo* je zašifrováno heslo metodou PBKDF2. V položkách *jmeno* a *prijmeni* se nachází jméno a příjmení zaměstnance. Tato tabulka neobsahuje žádný cizí klíč, jelikož nemá vazbu na žádnou tabulku.

Tabulka IS\_DODAVATELE slouží k uložení dodavatelů surovin. Tabulka obsahuje položky:

- *ID\_Dodavatele*,
- *nazevFirmy*,
- *ulice*,
- *cisloPopisne*,
- *mesto*,
- *PSC*,
- *telefon*.

Položka *ID\_Dodavatele* je umělý primární klíč. Sloupec *nazevFirmy* uchovává název firmy dodavatele. Položky *ulice*, *mesto*, *psc*, *telefon* doplňují informace o dodavatelích.

Tabulka IS\_FAKTURY slouží k uchovávání faktur. Tabulka obsahuje tyto atributy:

- *ID\_Faktury*,
- *cisloFaktury*,
- *datumVydani*,
- *ID\_Dodavatele*.

Položka *ID\_Faktury* je umělý primární klíč. Sloupec *cisloFaktury* uchovává číslo faktury dodavatele. Položka *datumVydani* obsahuje datum vydání faktury. *ID\_Dodavatele* je cizí klíč sloužící jako vazba mezi tabulkami.

Tabulka IS\_POLOZKY\_FAKTURY slouží k uchovávání položek faktur. Tabulka obsahuje tyto atributy:

- ID\_Polozky,
- cenaBezDPH,
- mnozstvi,
- nazev,
- stav,
- ID\_Faktury,
- ID\_Zasoby

Položka *ID\_Polozky* je umělý primární klíč. Sloupec *cenaBezDPH* uchovává cenu položky na faktuře dodavatele. Položka *mnozstvi* obsahuje počet dodaného zboží a sloupec *nazev* obsahuje druh suroviny. Atribut *stav* určuje, jestli dodaná položka byla přijata na sklad. *ID\_Faktury* a *ID\_Zasoby* jsou cizí klíče, které slouží jako vazba mezi tabulkami.

Tabulka IS\_SUROVINY uchovává suroviny a obsahuje:

- ID\_Suroviny,
- nazev,
- energetickaHodnotaKJ,
- energetickaHodnotaKCAL,
- tuky,
- nasyceneMastneKyseliny,
- sacharidy,
- cukry,
- bilkoviny,
- sul,
- ID\_Dodavatele.

Primárním klíčem je zde *ID\_Suroviny*. Cizí klíč je zde *ID\_Dodavatele* zastupující vazbu mezi tabulkami. Ostatní položky uchovávají podrobnosti o nutričních hodnotách jednotlivých surovin.

Tabulka IS\_SUROVINY\_POPIS slouží pro doplňující popis suroviny a obsahuje položky:

- ID\_Popis,
- nazev,
- ID\_Suroviny

Primárním umělým klíčem je zde položka *ID\_Popis*. Sloupec *nazev* slouží pro uchování detailů o surovině. Cizím klíčem je zde *ID\_Suroviny*.



Tabulka IS\_SKLADY slouží k uložení zboží na skladě a obsahuje položky:

- ID\_Zasoby,
- mnozstvi,
- minimalniZasoba.

Primární klíč je zde *ID\_Zasoby*. Další položka *mnozstvi* obsahuje množství dané položky na skladě. Dále atribut *minimalniZasoba* a obsahuje minimální množství pro řízení objednávek.

Tabulka IS\_VYROBKY slouží pro uložení výrobku ve firmě a obsahuje položky:

- ID\_Vyrobku,
- nazev,
- cena,
- vaha,
- druh.

Primárním umělým klíčem je zde položka *ID\_Vyrobku*. Sloupec *nazev*, *cena* a *vaha* slouží pro uchování detailů o výrobku. Sloupec *druh* rozděluje výrobky na cukrárenské a pekárenské.

Tabulka IS\_VYROBKY\_POPIS slouží jako číselník pro doplnění popisu výrobku a obsahuje položky:

- ID\_Popis,
- nazev,
- ID\_Vyrobku.

Primárním umělým klíčem je zde položka *ID\_Popis*. Sloupec *nazev* slouží pro uchování detailů o surovině. Cizím klíčem je zde *ID\_Vyrobku*.

Tabulka IS\_RECEPTY slouží pro uložení potřebných surovin na recept, hmotnosti jednotlivých navážek a obsahuje:

- ID\_Receptu,
- hmotnostNavazky,
- hmotnostReceptury,
- nazevPosypu,
- vahaPosypu,
- nazevNaplne,
- vahaNaplne
- ID\_Vyrobku.

Cizím klíčem je zde *ID\_Vyroбку* zastupující vazbu mezi tabulkami. Primárním klíčem je atribut *ID\_Receptu*. Položka *hmotnostNavazky* ukládá hmotnost navážených surovin. Sloupec *hmotnostReceptury* uchovává množství suroviny na daný recept. Položka *nazevPosypu* ukládá název posypové suroviny a *vahaPosypu* hmotnost posypu pro daný recept. Další sloupce ukládají to samé jako u posypu ale s náplní.

Tabulka *IS\_RECEPTY\_POLOZKY* slouží jako mezi tabulka, kvůli vazbě M:N mezi tabulkami *IS\_SUROVINY* a *IS\_RECEPTY* a obsahuje položky:

- *ID\_Polozky*,
- *mnozstvi*,
- *ID\_Receptu*,
- *ID\_Suroviny*.

*ID\_Polozky* je primárním umělým klíčem. *ID\_Receptu* a *ID\_Sroviny* jsou cizí klíče zastupující vazby mezi tabulkami.

Tabulka *IS\_NAMERENE\_VAHY* slouží pro uložení naměřené váhy hotových výrobků a obsahuje:

- *ID\_Vahy*,
- *datumMereni*,
- *vaha*,
- *ID\_Receptu*.

Sloupec *ID\_Vahy* je umělým primárním klíčem. Položka *ID\_Receptu* zastupuje vazbu mezi tabulkami. Sloupec *datumMereni* uchovává datum, kdy byl výrobek zvážen. Položka *vaha* uchovává váhu daného výrobku.

Tabulka *IS\_OBJEDNAVKY* slouží pro uložení objednávky zákazníků a obsahuje:

- *ID\_Objednavky*,
- *datumObjednani*,
- *datumVyzvednuti*,
- *zaloha*,
- *druhOddeleni*,
- *cena*,
- *ID\_Odberatele*.

Primárním umělým klíčem je zde *ID\_Objednavky*. Ostatní položky slouží pro uchování detailů o objednávkách. Cizím klíčem je zde *ID\_Odberatele*.

Tabulka IS\_POLOZKY\_OBJEDNAVKY slouží jako mezi tabulka, kvůli vazbě M:N mezi tabulkami IS\_VYROBKU A IS\_OBJEDNAVKY a obsahuje:

- ID\_Polozky,
- ID\_Vyrobku,
- ID\_Objednavky,
- mnozstvi,
- denObjednani.

*ID\_Polozky* je primárním umělým klíčem. *ID\_Vyrobku* a *ID\_Objednavky* jsou cizí klíče zastupující vazby mezi tabulkami.

Tabulka IS\_OBJEDNACI\_DNY slouží jako číselník dnů pro výrobky, které je možné objednat jen v některé dny a obsahuje:

- ID\_Dne,
- den,
- poznamka,
- ID\_Vyrobku.

Umělý primární klíč je zde *ID\_Dne*. Položka *den* uchovává dny, kdy je možno daný výrobek objednat. Sloupec *poznamka* je určen například pro svátky nebo pro nenadálé situace.

Tabulka IS\_ODBERATELE slouží pro evidenci zákazníka, který zadá objednávku a obsahuje:

- ID\_Odberatele,
- jmeno,
- prijmeni,
- telefon,
- ulice,
- cisloPopisne
- mesto
- PSC.

Primárním umělým klíčem je zde *ID\_Odberatele*. Ostatní položky slouží pro uchování detailů o odběrateli.

Jako celek tyto tabulky tvoří důležitou část informačního systému sloužící pro uchování všech potřebných dat. Uživatelé jednotlivých implementovaných modulů nemají přístup do celé databáze, ale jen do oblastí podle jejich pracovního zařazení.

## 2.4 Uživatelské role

Přístup, do informačního systému, je rozdělen pěti uživatelskými rolemi a to na *správce systému*, *operátor obchodu*, *logistik*, *operátor pekárny* a *manager*. Každá role zaručuje přístup do jiného modulu systému. Díky tomu je i omezen přístup k datům v databázi.

### 2.4.1 Správce systému

*Správce systému* je role pro technika nebo správce, který udržuje systém funkční a aktuální. Hlavním účelem je správa uživatelů a databáze. Jako jediný může existujícího uživatele smazat nebo mu změnit roli ale nemá oprávnění upravovat zadané informace od uživatelů jako třeba heslo a další. Uživatelé se mohou registrovat i bez asistence správce přes připravené rozhraní. Vytvořit nového uživatele s rolí *správce systému* může jen správce ze svého rozhraní.

Do databáze může přidávat data a to pouze do tabulky IS\_UZIVATELE a je mu zobrazena jen část atributů z tabulky IS\_UZIVATELE.

Tabulka 2 – Ukázka dat pro roli správce systému. Zdroj: Vlastní

uzivatelskeJmeno	pravo	jmeno	prijmeni
admin	4	já	já
hh	0	hh	hh
jirka	3	jirka	jj

### 2.4.2 Operátor obchodu

*Operátor obchodu* je role zaměstnance, který má přístup do sekce výrobků a objednávek. Uživatel s touto rolí má právo zadávat a upravovat výrobky, které je možné objednat. Má přístup k datům z tabulek IS\_VYROBKY a IS\_VYROBKY\_POPIS.

Tabulka 3 – Ukázka dat pro roli operátor obchodu, výrobky. Zdroj: Vlastní

nazev	popis_nazev	vaha	cena	druh
VÁNOČKA	VÁNOČNÍ	420	NULL	P
KOLEČKO	TŘENÉ	NULL	5.8	C

Zaměstnanec s touto rolí přístupu má zároveň na starost zadávání dat o odběratelích. Dále vytváří sestavy objednaného zboží pro cukrárenské a pekárenské výrobky. Díky tomu vzniká dvojí kontrola dostupných dat. Má tedy přístup do tabulek IS\_ODBERATELE, IS\_OBJEDNAVKY a IS\_POLOZKY\_OBJEDNAVKY.

**Tabulka 4 – Ukázka dat pro roli operátor obchodu, odběratelé. Zdroj: Vlastní**

<b>jmeno</b>	<b>prijmeni</b>	<b>telefon</b>	<b>ulice</b>	<b>cisloPopisne</b>	<b>mesto</b>	<b>PSC</b>
Jirí	Nekvinda	775123555	Dlouha	12	Jelenice	53624
Tomas	Jedno	606425888	Klidna	25	Vilemov	60025

**Tabulka 5 – Ukázka dat pro roli operátor obchodu, objednávka. Zdroj: Vlastní**

<b>Objednani</b>	<b>Vyzvednuti</b>	<b>zaloha</b>	<b>druhOdeleni</b>	<b>cena</b>	<b>nazevPolozky</b>	<b>mnozstvi</b>
1111-01-01	1111-01-01	0	C	0	CHLÉB	20
2018-04-03	2018-04-07	1000	P	2500	VÁNOČKA	20
2018-04-03	2018-04-07	1000	P	2500	KOLÁČKY	30

### 2.4.3 Logistik

Role *logistik* je určena pro zaměstnance starajícího se o sklad a zásoby. Zaměstnanec má právo zadávat nebo přidávat dodavatele. Přijímat faktury a následně naskladnit zboží. Zároveň vidí požadavky na suroviny za zvolené období pro výrobu. Pro jeho práci jsou potřeba tabulky IS\_SKLAD, IS\_DODAVATELE, IS\_FAKTURY a IS\_POLOZKY\_FAKTURY.

**Tabulka 6 – Ukázka dat pro roli logistik, dodavatelé. Zdroj: Vlastní**

<b>nazevFirmy</b>	<b>ulice</b>	<b>cisloPopisne</b>	<b>mesto</b>	<b>PSC</b>	<b>telefon</b>
ADAMEC ZBYNĚK	Štěpnická	1085	Uherské hradiště	68606	NULL
AGRI-M spol. s.r.o.	Mistřice	414	Mistřice	68712	NULL

**Tabulka 7 – Ukázka dat pro roli logistik, dodavatelé. Zdroj: Vlastní**

<b>cisloFaktury</b>	<b>datumVydani</b>	<b>nazev</b>	<b>mnozstvi</b>	<b>cenaBezDPH</b>
F4454	2018-04-24	Alidon	10	420
F4454	2018-04-24	Fajný chléb smes	10	300

#### **2.4.4 Operátor pekárny**

*Operátor pekárny* má přístup jen do oblasti spravující pekárnu. Jeho pracovní náplň spočívá v přidávání a upravování receptů. Dále je jeho prací zapisování naměřené hmotnosti výsledných výrobků z důvodů kontroly výrobního procesu a dodržení ČSN normy.

Má tedy přístup k datům z tabulek IS\_RECEPTY, IS\_RECEPTY\_POLOZKY a IS\_NAMERENE\_VAHY.

**Tabulka 8 – Ukázka dat pro roli oprátor pekárny, recept. Zdroj: Vlastní**

<b>Navazka</b>	<b>Receptura</b>	<b>mnozstvi</b>	<b>Nazev suroviny</b>	<b>nazev</b>
1040	1021.8	18000	Pšeničná mouka hladká polosvět	CHLÉB
1040	1021.8	7200	Žitná mouka chlebová	CHLÉB

#### **2.4.5 Manager**

Role *manager* je určena pro uživatele systému, který má na starosti hodnocení a analýzu výrobního procesu. Uživatel má přístup do modulu, kde jsou počítány indexy způsobilosti. V modulu se dále nachází grafické zobrazení distribuční funkce a histogram.

*Manager* nemůže do databáze přidávat žádná data a je mu zobrazen jen seznam výrobků z tabulky IS\_VYROBKY, IS\_VYROBKY\_POPIIS a seznam naměřených vah pro vybraný výrobek z tabulky IS\_.NAMERENE\_VAHY.

**Tabulka 9 – Ukázka dat pro roli manager. Zdroj: Vlastní**

ID_Vyrobtu	vaha	nazev	ID_Vyrobtu	datumMereni	Vaha
3	900	CHLÉB KVASOVÝ	5	2018-04-04	50
8	400	CHLÉB TOUSTOVÝ	5	2018-04-06	42
5	42	ROHLÍK TUKOVÝ	5	2018-04-02	45

## 2.5 Třídy v aplikaci

Java jakožto objektově orientovaný jazyk se skládá z objektů a tříd. Objekty vznikají, jako instance jednotlivých tříd uchovávají si svůj stav a lze nad nimi volat metody určené pro daný objekt. Třídy tedy obsahují atributy pro uchovávání stavu a metody. Tato podkapitola popisuje použité třídy v aplikaci, které jsou:

- Třída *MyConnector* – slouží pro nastavení a vytvoření spojení s databází. Obsahuje atributy nutné k vytvoření spojení a metody, které spojení navazují. Pro vytvoření spojení je nejdříve nutné nastavit URL adresu, port, přihlašovací jméno a heslo. Například pro nastavení spojení je implementována metoda *setUpConnection(String userName, String password)* se dvěma vstupními parametry.
- Třída *MyDBProperties* – zde se nacházejí dvě konstanty typu *String* a to pro přihlášení do databáze. Jedna obsahuje přihlašovací jméno a druhá heslo.
- Třída *PasswordStorage* – slouží pro zašifrování a ověření hesla. Pro zašifrování se používá metoda *PBKDF2*, která je vhodná proti útokům slovníkovou metodou nebo hrubou silou. Hlavní metody jsou *createHash(char[] password)* pro zašifrování a *verifyPassword(char[] password, String correctHash)* pro ověření hesla.
- Třída *Prvek* – slouží pro vytvoření objektu typu *Prvek* s atributy *nazev*, *pocet* a metodami určenými pro práci s těmito objekty. Důležitou metodou je zde metoda *compareTo(Prvek t)* sloužící pro porovnání objektů typu *Prvek*.
- Třída *FunkceGraf* – obsahuje metody pro vytvoření distribuční funkce a histogramu s pomocí knihoven *JFreeChart*. Nachází se zde například metoda pro výpočet četností z nashromážděných dat.

## 2.6 Matematické výpočty

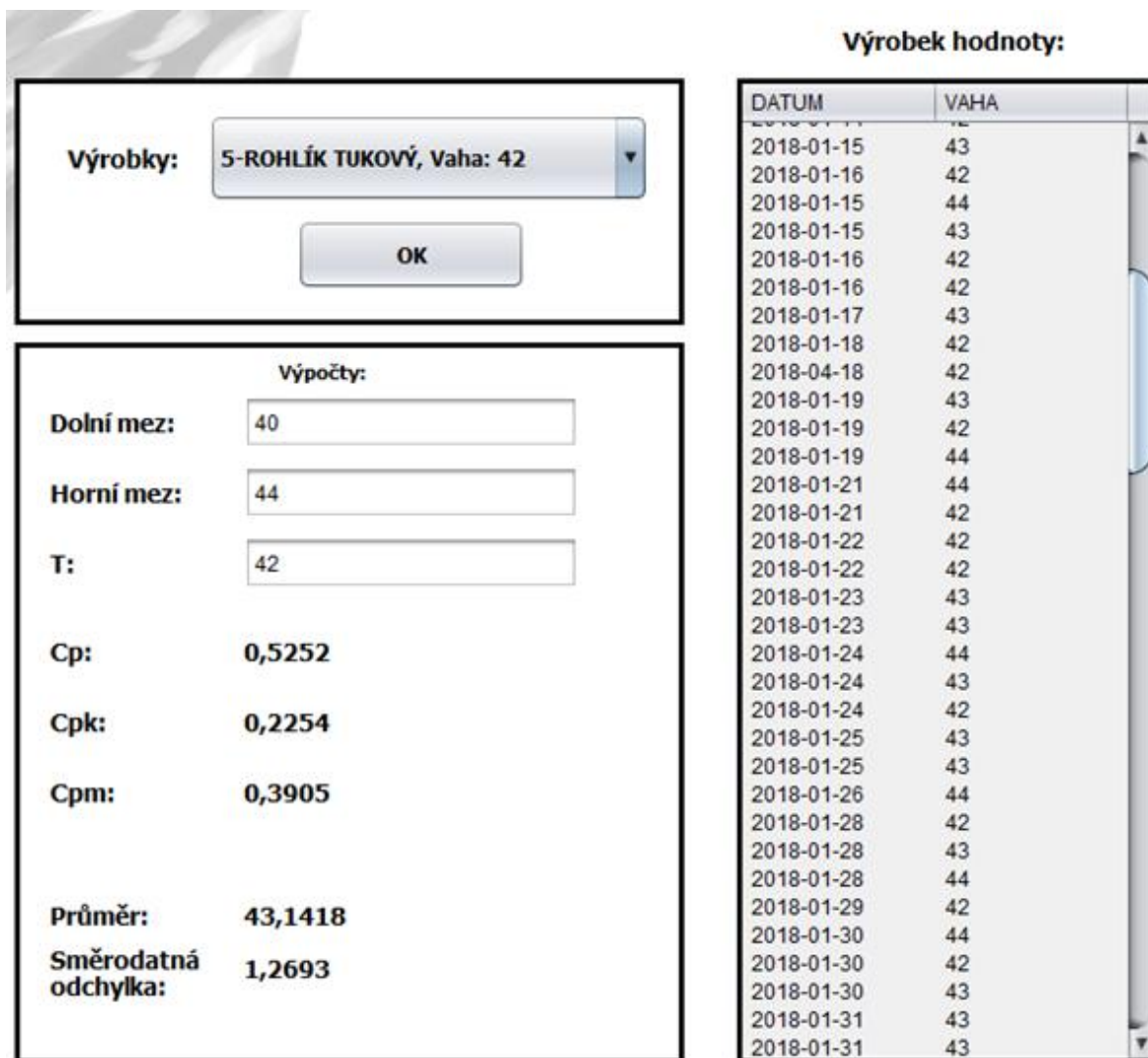
V této podkapitole jsou popsány zautomatizované výpočty prováděné programem s daty, které byly vloženy do aplikace prostřednictvím uživatelů spravující jednotlivé moduly. Pro znalost způsobilosti procesu a zjištění jakosti jsou zde počítány indexy způsobilosti s grafickým výstupem pro znázornění variability naměřených hodnot. Dále je popsán výpočet nutričních hodnot jednotlivých výrobků a nakonec následuje popis kontroly plnění závazků a tím i řízení skladových zásob.

### 2.6.1 Indexy způsobilosti

Aplikace zaznamenává data o zvážených hmotnostech jednotlivých výrobků. Pro každý výrobek je tedy možné v aplikaci vypočítat indexy způsobilosti, průměrnou hodnotu hmotnosti a směrodatnou odchylku. Zdrojový kód pro výpočty je uveden v Příloze B – *Zdrojový kód pro výpočet indexů způsobilosti, směrodatné odchylky a průměru.*

Vše je graficky vykresleno pomocí histogramu a distribuční funkce s využitím knihoven *jFreeChart*. Příklad grafu je v Příloze C – *Histogram a distribuční funkce.*





Obrázek 11 – Indexy způsobilosti. Zdroj: Vlastní

Z příkladu na horním obrázku lze vypočítat odhad podílu zmetků. Vypočet vypadá takto:

$$NC = \Phi(-3C_{pl}) + \Phi(-3C_{pu}) = 2\Phi(-3C_{pk}) = 2\Phi(-3 \times 0,2254) = 2\Phi(-0,6762) = 2(1 - \Phi(0,6762)) = 2(1 - 0,7485) = 0,503.$$

Při dané variabilitě výroby ( $\sigma=1,2693$ ) lze očekávat 50% výrobků mimo toleranční meze.

### 2.6.2 Nutriční hodnoty

V aplikaci se pro jednotlivé recepty vypočítávají nutriční hodnoty. V tabulce se nacházejí hodnoty jednotlivých surovin používané v receptu na daný výrobek. Výpočty jsou prováděny nejdříve sečtením jednotlivých sloupců a následně jsou tyto výsledky vynásobeny číslem 0,9 a to z důvodu snížení hmotnosti při pečení. Vztah je vyjádřen:

$$V = (\sum_1^n k_n) \times 0,9.$$

Příklad výsledku je možné si prohlédnout na Obrázek 12 – Nutriční hodnoty. Zdroj: Vlastní.

**Recept - VEKA TUKOVÁ**

ID	SUROVINA	MNOZSTVI V G	ENERGET. H...	ENERGET. H...	TUKY	NASY. MAST. ...	SACHARIDY	CUKRY	BILKOVINY	SUL
86	Pšeničná mo...	11250	1461	344	1	0	68	2	11	0
87	Margarín UNI...	225	2960	720	80	39	0	0	0	0
88	Margarín UNI...	225	2960	720	80	39	0	0	0	0
89	Sůl	225	0	0	0	0	0	0	0	94
90	Concord	35	2142	516	38	24	23	4	13	1
91	Voda	6300	0	0	0	0	0	0	0	0

**Obsah živin v g / 100g suroviny**

ENERGET. HODN...	ENERGET. HODN...	TUKY	NASY. MAST. KYSE...	SACHARIDY	CUKRY	BILKOVINY	SUL
8570.7	2070.0	179.1	91.8	81.9	5.4	21.6	85.5

Obrázek 12 – Nutriční hodnoty. Zdroj: Vlastní

Vrchní část obrázku zobrazuje seznam surovin, požadovaných pro daný recept, s jejich nutričními hodnotami. Spodní část zobrazuje výsledné nutriční hodnoty pro hotový výrobek.

### 2.6.3 Sklad a kontrola poptávky

Pro zvolenou poptávku aplikace umožňuje kontrolu skladu. Podle receptů k výrobkům je zobrazen seznam potřebných surovin. Pokud je na skladě dostatečné množství dané suroviny na pokrytí poptávky, je řádek ve skladu zvýrazněn zeleně a stav je změněn na *OK*. Sloupec *minimum* ve skladu zaručuje minimální skladovou zásobu.

SUROVINA	POCET	DEN	ZBOZI	MNOZSTVI ...	MINIMUM	STAV
Kmín	18.0000		29-Fajný c...	10	2	
UNI Aktiva Internation...	90.0000		131-Sůl	250	10	OK
Žitná mouka chlebová	800.0000					
Zárodečná kultura	240.0000					
Pšeničná mouka hla...	3600.0000					
Voda	3420.0000					
Sůl	150.0000					
FH Stabil	60.0000					
Voda	400.0000					
Žitná mouka chlebová	1440.0000					

Obrázek 13 – Poptávka a stav skladu. Zdroj: Vlastní

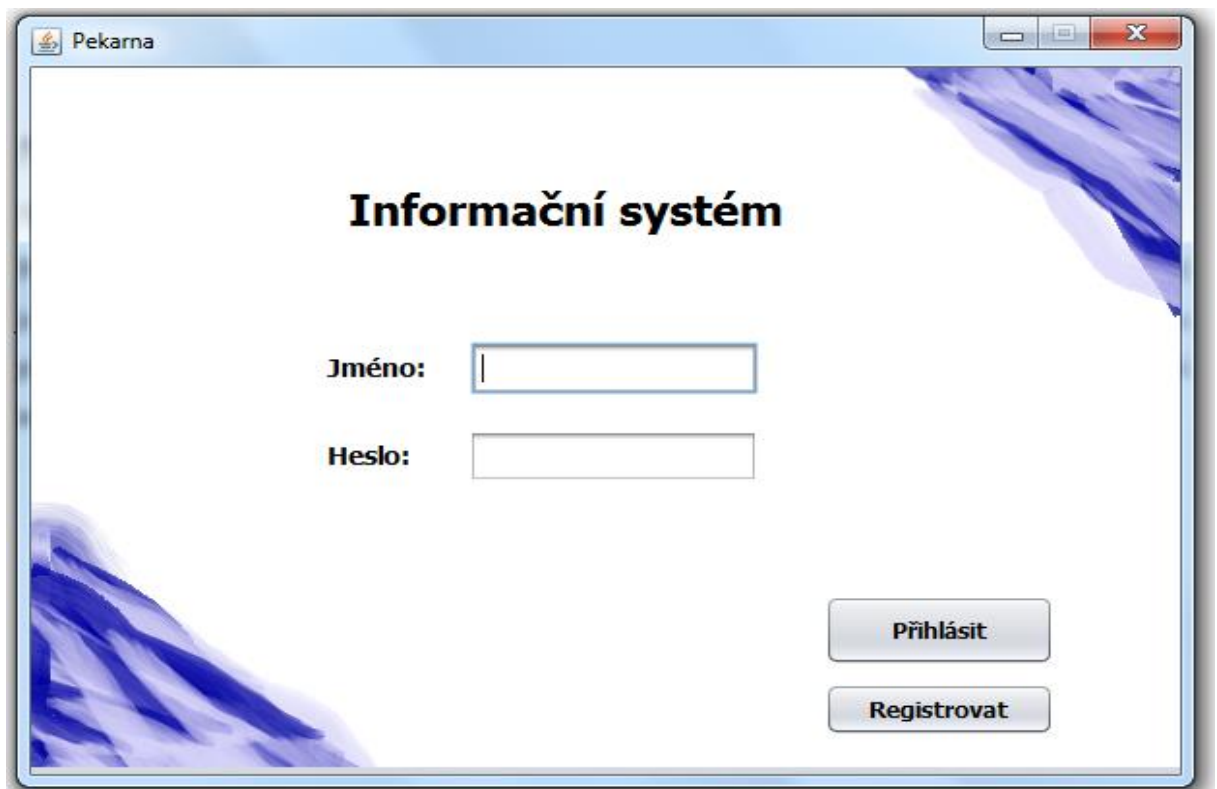
Tabulka vlevo zobrazuje poptávku po surovinách za zvolené období. V tabulce na pravé straně jsou zobrazeny suroviny na skladě a zároveň je zvýrazněn řádek s dostačujícím množstvím.

## 2.7 Uživatelská příručka

V této podkapitole je popsána uživatelská příručka a slouží jako základní přehled pro uživatele. Tento přehled by měl být dostačující pro pochopení jak pracovat v této aplikaci bez větších obtíží. Nejdříve je popsáno přihlášení a registrace uživatelů a funkce stejnojmenných rozhraní. Následně jsou popsány jednotlivé moduly jako *IT*, *Pekárna*, *Výrobky* a *Sklad*.

### 2.7.1 Přihlášení

Pro přístup do systému je nutné se přihlásit. Vyžadováno je uživatelské jméno a heslo. Pokud je heslo nebo uživatelské jméno špatně zadané, zobrazí se chybová zpráva. Při korektním přihlášení se uživatel dostane do části systému, která odpovídá jeho roli. Neregistrovaný uživatel má možnost se zaregistrovat. Jediná role, která takto nejde zaregistrovat je *správce systému*.



Obrázek 14 – Přihlašovací obrazovka. Zdroj: Vlastní

### 2.7.2 Registrace

Pro registraci je připraven formulář, kde je nutné zadat uživatelské jméno, jméno, příjmení, heslo a vybrat pracovní pozici. Pokud nejsou všechny kolonky vyplněny, data nelze uložit a objeví se chybová zpráva. Při správné registraci se okno zavře a objeví se přihlašovací formulář.

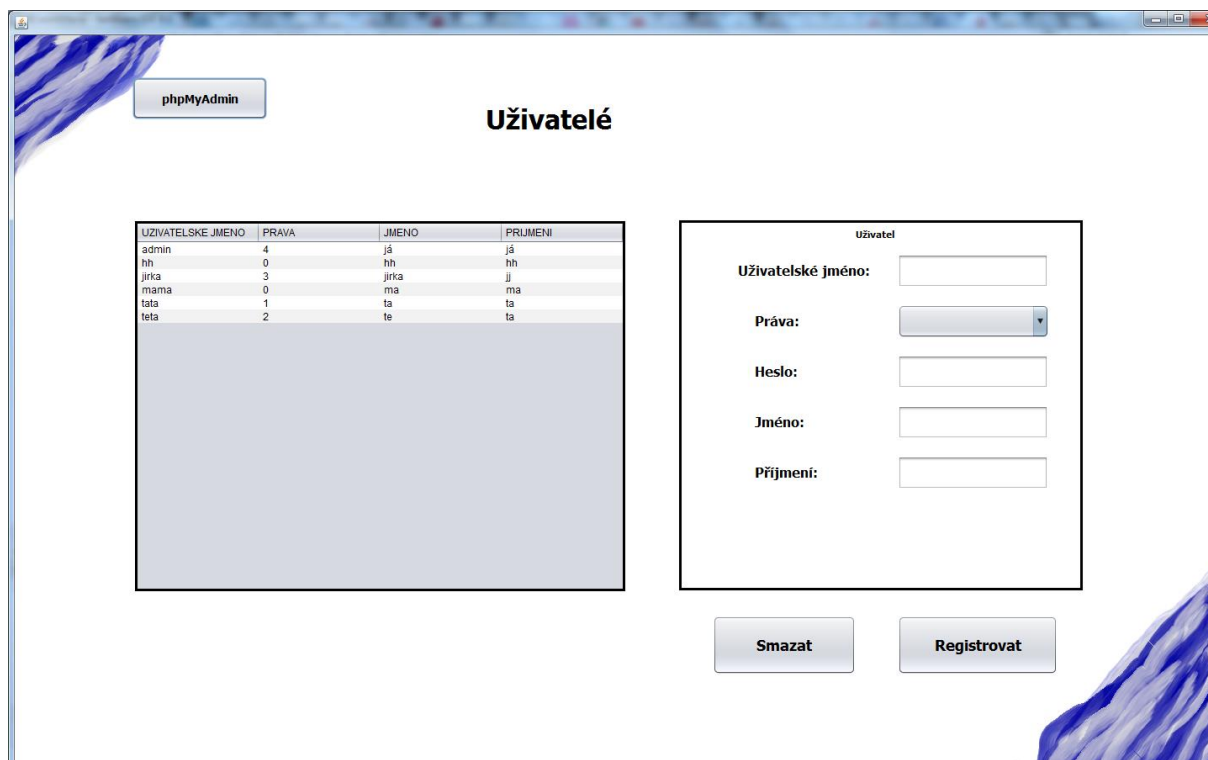
The image shows a software window titled "Registrace" (Registration). It contains the following fields and controls:

- Uživatelské jméno:** A text input field.
- Jméno:** A text input field.
- Příjmení:** A text input field.
- Heslo:** A text input field.
- Pozice:** A dropdown menu with "Logistik" selected.
- OK:** A button at the bottom center.

Obrázek 15 – Registrace. Zdroj: Vlastní

### 2.7.3 IT

Toto rozhraní je určené pro uživatele s rolí *správce systému*. Správce zde může upravovat databázi přes odkaz na phpMyAdmin. Dále může zaregistrovat nového uživatele. Další z možností je stávajícímu uživateli změnit zařazení do jiného oddělení. Správce však nemůže měnit uživatelské údaje, ale jen smazat zaregistrované uživatele.



Obrázek 16 – IT rozhraní. Zdroj: Vlastní

## 2.7.4 Pekárna

Tato obrazovka slouží pro uživatelskou roli *operátor pekárny*. Nachází se zde zobrazení všech výrobků. Výrobky je možné přidávat a upravovat. Výrobek se vybírá z tabulky kliknutím. Následně může být upraven nebo smazán. Pokud je výrobek vybrán, je možno přejít na další rozhraní a to tlačítky váha nebo recepty.

### Pekárna

ID	NAZEV	POPIS	HMOTNOST NA...	HMOTNOST DL...	POSYP	POSYP VAHA	NÁPLŇ	NÁPLŇ VÁHA
1	CHLĚB	KVASOVÝ	1040	1021.8				0
2	ROHLÍK	TUKOVÝ	52	55				0
3	CHLĚB	TOUSTOVÝ	450	481				0
4	VEKA	TUKOVÁ	360	365				0
5	VÁNOČKA	TUKOVÁ	340	356				0
6	ZÁVIN	MAKOVÝ	390	398			MAKOVÁ STAN...	0
7	ČESKÉ	BUCHTY	410	410			ANO	80
8	VÁNOČKA	VÁNOČNÍ	0	378		MANDLE PLÁTKY		0
9	VÁNOČKA	SYPANÁ	378	378		ARAŠÍDY		0
10	CHLĚB	FÍNSKÝ	520	521		SLUNEČNICE	5	0
11	CHLĚB	CIBULOVÝ	460	490				0
12	BAGETA	PREMIUM	180	178				0
13	CHLĚB	KMĚNOVÝ	1040	1042.6	KMĚN	1		0
13	CHLĚB	KRÁJENÝ	1040	1042.6	KMĚN	1		0
14	ZÁVIN	JABLEČNÝ	380	389			JABLKO	150
15	KOLÁČKY	POSVÍČENSKÉ	240	251			ANO	90
16	CHLĚB	PROBIO	570	586				0
17	VÁNOČKA	SVÁTEČNÍ	378	378		MANDLE PLÁTKY		0
18	VDOLEČEK	SVATEBNÍ	23	23			TVARHOVÁ	9
19	ZÁVIN	ORECHOVÝ	240	248			ORECHOVÁ	80
20	BAGETA	TUKOVÁ	210	217				0
21	CHLĚB	TOUSTOVÝ	450	470				0
21	CHLĚB	TRNAVÝ	450	470				0
22	CHLĚB	SLUNEČNICOVÝ	860	883	SLUNEČNICE	3		0
23	CHLĚB	KRUPÁK	580	633				0
24	KOLÁČ	SVATEBNÍ	48	43	hádká mouka, ...		TVARHOVÁ	18
25	VEČKA	CEREÁLNI	270	295				0
26	DUKÁTOVÉ	BUCHTIČKY	230	236				0
29	LISTOVÉ TESTO		10	0				0

Složení

3- CHLĚB KVASOVÝ

ID:

Hmotnost navážky:

Hmotnost dle receptu...:

Posyp:

Posyp váha:

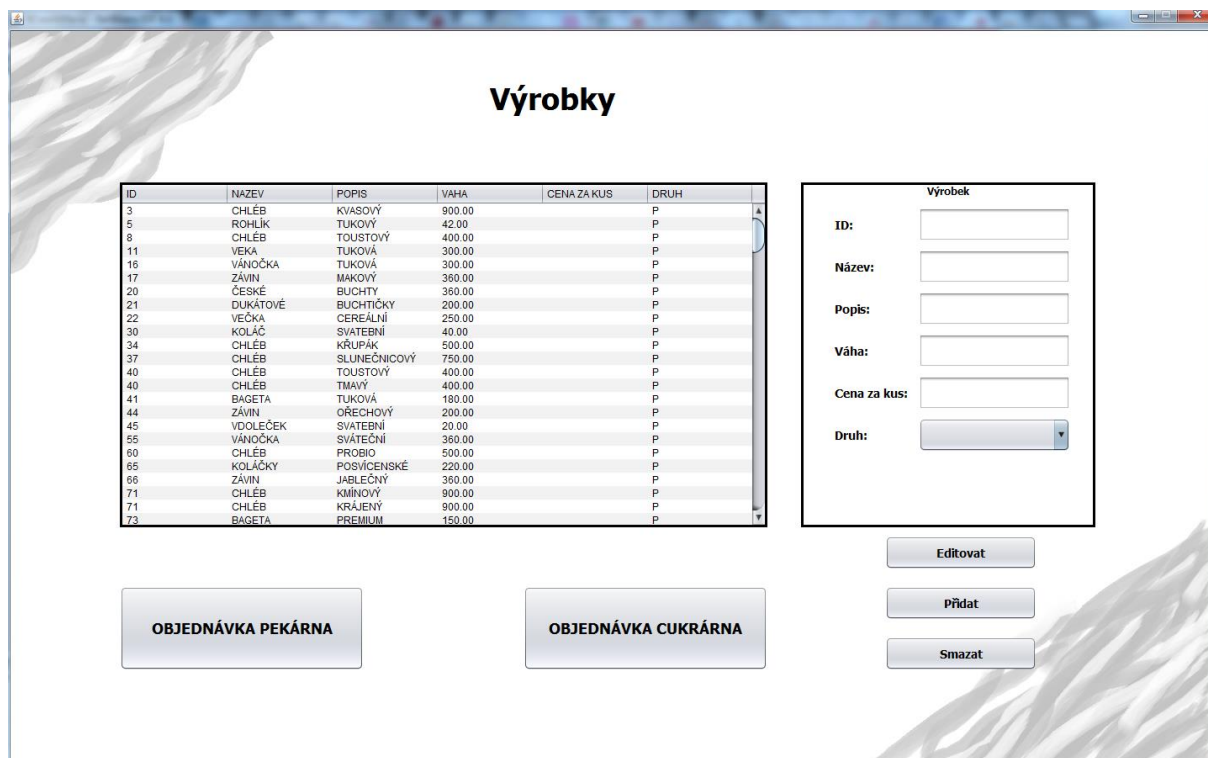
Náplň:

Náplň váha:

Obrázek 17 – Pekárna. Zdroj: Vlastní

## 2.7.5 Výrobky

Toto rozhraní je určeno pro uživatelskou roli *operátor obchodu*. Operátor má na starost správu všech výrobků a zároveň zde zadává objednávky pro pekárnu a cukrárnu. Tlačítko *objednávka pekárna* otevře rozhraní pro pekárenské objednávky a tlačítko *objednávka cukrárna* otevře rozhraní pro cukrárenské objednávky.

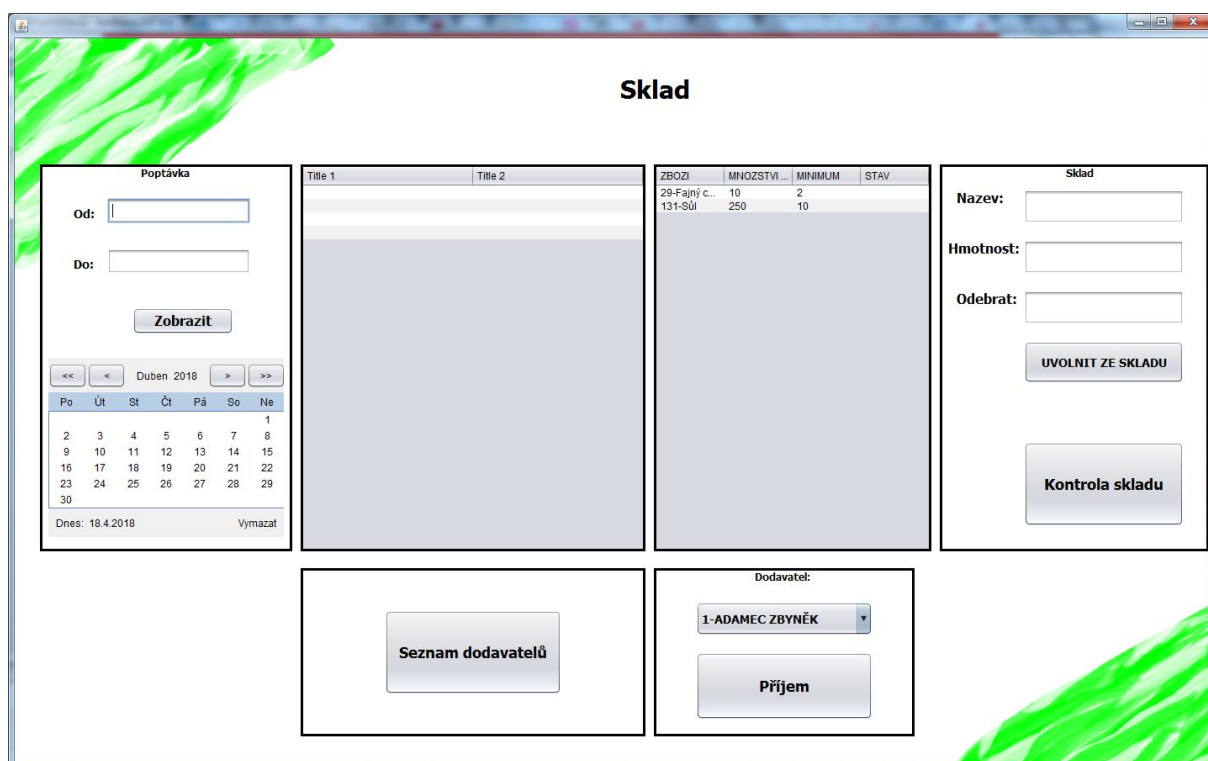


Obrázek 18 – Výrobky. Zdroj: Vlastní

## 2.7.6 Sklad

Tato obrazovka slouží pro uživatelskou roli *logistik*. V levé polovině se nachází tabulka s objednávkami. Tabulka se naplní daty po zadání data (od, do) a stisknutí tlačítka zobrazit. Pro přidání data do pole stačí vybrat datum v kalendáři a kliknout na požadované pole. V pravé polovině se nachází tabulka skladu a je naplněna automaticky. Tlačítko kontrola skladu slouží pro porovnání objednávek a skladu. Pokud je daná surovina na skladu a dostatečném množství, změní se v pravé tabulce sloupec stav na hodnotu *OK* a řádek se vybarví zeleně. Ostatní suroviny dostanou stav *low*. Tlačítko příjem otevře rozhraní pro příjem surovin od dodavatelů. Tlačítko dodavatelé otevře rozhraní s dodavateli.





Obrázek 19 – Sklad. Zdroj: Vlastní

## 2.8 Instalační příručka

Instalační příručka slouží jako návod pro zavedení aplikace do provozu. Na přiloženém CD se nachází spustitelný soubor aplikace, který je nutné zkopírovat na vhodné místo v počítači. Počítač musí být připojen do sítě, aby bylo možné navázat spojení se serverem. Databáze je umístěna na vlastním serveru. Pro obnovení smazané nebo poškozené databáze je nutné kontaktovat správce serveru, který vlastní potřebné SQL skripty.

### 3 ZÁVĚR

Díky analýze existujících softwarových řešení, která je umístěna v teoretické části bakalářské práce, bylo zjištěno, že na trhu existuje velký výběr informačních systémů pro řízení a správu firemních agend. Většina z těchto řešení je navržena jako modulární aplikace a potřebné rozšiřující moduly se musí dokoupit. Pro dosažení konkurenceschopnosti si myslím, že je důležité vytvořit uživatelsky přívětivou aplikaci usnadňující práci a zároveň dbát na grafický návrh pro odlišení od konkurence.

Tento informační systém byl navrhnout podle dat jen z jednoho podniku. Pro případné nasazení v jiných firmách s podobným zaměřením je nutné tuto aplikaci upravit podle potřeb konkrétního zákazníka. Tyto úpravy však nezaberou tolik času jako celý návrh databáze, aplikace a vývoj celého systému.

Pro tuto aplikaci by bylo možné vytvořit další moduly, které by rozšířily záběr tohoto informačního systému. Tyto nadstavby je však nejdříve nutné konzultovat se zadavatelem systému pro upřesnění požadovaných funkcionalit. Jedna z dalších úprav by spočívala v automatickém generování reportů, pro zvýšení efektivity a systematičtější správu firemních agend.

Proces vytváření závěrečné práce pro mě byl zajímavou zkušeností, díky které jsem si zdokonalil dosažené znalosti v oblasti programování, návrhu systému, návrhu databáze a s tím spojené využití jazyka SQL. Zároveň bylo přínosné vyzkoušet si analýzu dat, jakož to získávání a určení potřebných informací pro návrh databáze.

## 4 POUŽITÁ LITERATURA

- [1] JABLONSKÝ, Josef. *Operační výzkum: kvantitativní modely pro ekonomické rozhodování*. 3. vyd. Praha: Professional Publishing, 2007. ISBN 978-80-86946-44-3.
- [2] APSYS CZ, S.R.O. *SISTORE: Logistický informační systém* [online]. [cit. 2018-03-29]. Dostupné z: <http://www.sistore.eu/>
- [3] ALTUS SOFTWARE S.R.O. *Altus Vario: Software Altus Vario* [online]. [cit. 2018-03-29]. Dostupné z: <https://www.vario.cz/podrobny-popis/>
- [4] SAUL INFORMAČNÍ SYSTÉMY S.R.O. *Podnikový informační systém RIS* [online]. 2012 [cit. 2018-03-29]. Dostupné z: <http://www.saul.cz/erp-ris/>
- [5] NAVERTICA A.S. *NAVERTICA - Microsoft Dynamics NAV* [online]. 2018 [cit. 2018-03-29]. Dostupné z: <http://www.navertica.com/produkty/erp/microsoft-dynamics-nav/>
- [6] Architektura klient-server (Client–server model). In: ManagementMania.com [online]. Wilmington (DE) 2011-2018, 04.11.2016 [cit. 29.03.2018]. Dostupné z: <https://managementmania.com/cs/architektura-klient-server/>
- [7] Třívrstvá architektura (Three-tier architecture). In: ManagementMania.com [online]. Wilmington (DE) 2011-2018, 05.12.2015 [cit. 29.03.2018]. Dostupné z: <https://managementmania.com/cs/trivrstva-architektura-three-tier-architecture/>
- [8] NetBeans. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2018-03-29]. Dostupné z: <https://cs.wikipedia.org/wiki/NetBeans/>
- [9] XAMPP. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2018-03-29]. Dostupné z: <https://cs.wikipedia.org/wiki/XAMPP/>
- [10] TOŠENOVSKÝ, Josef. *Statistické metody pro zlepšování jakosti*. Ostrava: Montanex, 2000. ISBN 80-722-5040-x.
- [11] Java (programovací jazyk). In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2018-04-19]. Dostupné z: [https://cs.wikipedia.org/wiki/Java\\_\(programovac%C3%AD\\_jazyk\)/](https://cs.wikipedia.org/wiki/Java_(programovac%C3%AD_jazyk)/)
- [12] MariaDB. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2018-04-19]. Dostupné z: <https://cs.wikipedia.org/wiki/MariaDB/>

## 5 PŘÍLOHY

Příloha A – <i>Zdrojová data</i> .....	53
Příloha B – <i>Zdrojový kód pro výpočet indexů způsobilosti, směrodatné odchyly a průměru</i> .	56
Příloha C – <i>Histogram a distribuční funkce</i> .....	58

Příloha A – Zdrojová data

Třené koleč., rohlíček	..... ks .....	Kč	Větrník	..... ks .....	Kč
Linecké kolečko	..... ks .....	Kč	Griliášová trubička	..... ks .....	Kč
Rohlíček	..... ks .....	Kč	Meruň./hrušk. dezert	..... ks .....	Kč
Srdíčko linecké	..... ks .....	Kč	Čokol. trub. se šlehačkou	..... ks .....	Kč
Medáček	..... ks .....	Kč	Kominík	..... ks .....	Kč
Moka oválek	..... ks .....	Kč	Bolero řez	..... ks .....	Kč
Čokoládové kolečko	..... ks .....	Kč	Medová kulička	..... ks .....	Kč
Kokosový rohlíček	..... ks .....	Kč	Marokánka	..... ks .....	Kč
Laskonka	..... ks .....	Kč	Harlekýn	..... ks .....	Kč
Laskonka ořechová	..... ks .....	Kč	Kokosová kulička	..... ks .....	Kč
Kremšnýt	..... ks .....	Kč	Punčová koule	..... ks .....	Kč
Listová trubička	..... ks .....	Kč	Košíček punčový	..... ks .....	Kč
Šáteček	..... ks .....	Kč	Košíček ovocný	..... ks .....	Kč
Dia věneček se šlehačkou	..... ks .....	Kč	Pohádka	..... ks .....	Kč
Věneček	..... ks .....	Kč	Jahůdka	..... ks .....	Kč
Věneček s čokoládou	..... ks .....	Kč	Irský likér	..... ks .....	Kč
Kávový banánek	..... ks .....	Kč	Italské latté	..... ks .....	Kč
Kokoska	..... ks .....	Kč	Ovocná kostka	..... ks .....	Kč
Indián	..... ks .....	Kč	Ovocný mls	..... ks .....	Kč
Špička likérová	..... ks .....	Kč	Tropical	..... ks .....	Kč
Ořechový kornout	..... ks .....	Kč	Řez Míša/Tiramisu	..... ks .....	Kč
Kávové zrno	..... ks .....	Kč	Vídeňský košíček	..... ks .....	Kč
Koule arašídová	..... ks .....	Kč	Duo kostka	..... ks .....	Kč
Pařížský věneček	..... ks .....	Kč	Listové trubičky 150 g	..... ks .....	Kč
			Biskupský chlebíček 400 g	..... ks .....	Kč

			Pondělí	Úterý	Středa	Čtvrtek	Pátek	Sobota
č.	Název výrobku	g	ks	ks	ks	ks	ks	ks
94	Bageta tuková	90						—
41	Bageta tuková balená	180	—	—	—			—
73	Bageta Premium balená	150	—	—	—	—		—
11	Veka tuková	300						
75	Veka tuková krájená	300						
14	Kobliha ovocná	50						—
15	Kobliha s pudinkem	60		—			—	—
62	Kobliha nugeta	50						—
99	Smaženka jablková	50	—	—		—	—	—
51	Bavorský vdoleček	70	—		—	—	—	—
101	Loupák	45	—		—		—	—
52	Pletenec	100	—		—		—	—
18	Šáteček tvarohový	50						—
19	Květ makový	50						—
49	Květ ořechový	50						—
82	Šáteček povidlový	50						—
118	Buchta tvarohová	60	—		—		—	—
119	Buchta marmeláda	60	—		—		—	—
141	Buchta bramborová	180	—	—		—	—	—
59	Tvarohové pečivo - mák	50		—		—	—	—
59	Tvarohové pečivo - skořice	50		—		—	—	—
17	Závin makový/půlený	360/180	/	/	/	/	/	—
66	Závin jablkový	360						—

Název výrobku:	<b>CHLÉB KVASOVÝ</b>									
Hmotnost 1ks v g:	900									
Hmotnost navážky 1ks v g:	1040									
RECEPTURA:										
Surovina / Množství v g na ks	Pšeničná mouka hladká polosvětlá	Žitná mouka chlebová	Zárodečná kultura	Voda	Žitná mouka chlebová	FH stabil	UNI Aktiva International	Sól	Kmín	Voda
100	36 000	14 400	2 400	4 000	8 000	600	900	1 500	180	34 200
50	18 000	7 200	1 200	2 000	4 000	300	450	750	90	17 100
1	360	144	24	40	80	6	9	15	2	342
Hmotnost dle receptury:	1021,8									



## Příloha B – Zdrojový kód pro výpočet indexů způsobilosti, směrodatné odchylky a průměru

```
public double spocitejCpindex(double dMez, double hMez, double odchylka) {
    double indexCp = 0;
    indexCp = ((hMez - dMez) / (6 * odchylka));
    return indexCp;
}

public double spocitejCpk(double dMez, double hMez, double odchylka,
double m) {
    double indexCpk = 0;
    double indexCpkU = Double.MAX_VALUE;
    double indexCpkL = Double.MAX_VALUE;
    indexCpkU = ((hMez - m) / (3 * odchylka));
    indexCpkL = ((m - dMez) / (3 * odchylka));

    indexCpk = Math.min(indexCpkL, indexCpkU);
    return indexCpk;
}

public double spocitejCpkL(double dMez, double odchylka, double m) {
    double indexCpkL = 0;
    double indexCpkUp = Double.MAX_VALUE;
    double indexCpkLo = Double.MAX_VALUE;
    indexCpkLo = ((m - dMez) / (3 * odchylka));
    indexCpkL = Math.min(indexCpkLo, indexCpkUp);
    return indexCpkL;
}

public double spocitejCpkU(double hMez, double odchylka, double m) {
    double indexCpkU = 0;
    double indexCpkUp = Double.MAX_VALUE;
    double indexCpkLo = Double.MAX_VALUE;
    indexCpkUp = ((hMez - m) / (3 * odchylka));
    indexCpkU = Math.min(indexCpkLo, indexCpkUp);
    return indexCpkU;
}

public double spocitejCpml(double dMez, double odchylka, double t,
double m) {
    double indexCpml = 0;
    indexCpml = ((t - dMez) / (3 * Math.sqrt(Math.pow(odchylka, 2) +
(Math.pow(m - t, 2))))));
    return indexCpml;
}

public double spocitejCpmu(double hMez, double odchylka, double t,
double m) {
    double indexCpmu = 0;
    indexCpmu = ((hMez - t) / (3 * Math.sqrt(Math.pow(odchylka, 2) +
(Math.pow(m - t, 2))))));
    return indexCpmu;
}

public double spocitejCpmindex(double dMez, double hMez, double
odchylka, double t, double m) {
    double indexCpm = 0;
    double predVypocet = 0;
    predVypocet = (Math.pow(odchylka, 2) + (Math.pow(m - t, 2)));
    indexCpm = ((hMez - dMez) / (6 * Math.sqrt(predVypocet)));
    return indexCpm;
}
```



```

private double spocitejSmerodatnouOdchylku(double prumer) {
    Prvek get;
    double suma = 0;
    double pomocnyVypocet = 0;
    double vysledek = 0;
    int n;

    for (int i = 0; i < list1.size(); i++) {
        get = list1.get(i);
        suma = Math.pow((get.getPocet() - prumer), 2);
        pomocnyVypocet += suma;
    }
    n = list1.size();
    vysledek = Math.sqrt(pomocnyVypocet * (Math.pow(n, -1)));
    String text = String.format("%.4f", vysledek);
    jLabelOchylka.setText(text);
    return vysledek;
}

private double spocitejPrumer() {
    Prvek get;
    double prumer = 0;
    for (int i = 0; i < list1.size(); i++) {
        get = list1.get(i);
        prumer += get.getPocet();
    }
    String text = String.format("%.4f", prumer / list1.size());
    jLabelPrum.setText(text);
    return prumer / list1.size();
}

```

Příloha C – Histogram a distribuční funkce

