



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA PODNIKATELSKÁ  
ÚSTAV INFORMATIKY

FACULTY OF BUSINESS AND MANAGEMENT  
INSTITUTE OF INFORMATICS

## MODUL PLÁNOVÁNÍ A ŘÍZENÍ VÝROBNÍCH ZDROJŮ

MODULE OF PLANNING AND MANAGEMENT OF PRODUCTION RESOURCES

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MICHAL KLAUDÍNÝ

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. JIŘÍ KŘÍŽ, Ph.D.

BRNO 2010

# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

**Klaudíny Michal**

---

Manažerská informatika (6209R021)

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách, Studijním a zkušebním řádem VUT v Brně a Směrnicí děkana pro realizaci bakalářských a magisterských studijních programů zadává bakalářskou práci s názvem:

**Modul plánování a řízení výrobních zdrojů**

v anglickém jazyce:

**Module of Planning and Management of Production Resources**

Pokyny pro vypracování:

Úvod

Vymezení problému a cíle práce

Teoretická východiska práce

Analýza problému a současné situace

Vlastní návrhy řešení, přínos návrhů řešení

Závěr

Seznam použité literatury

Přílohy

Seznam odborné literatury:

- DOSEDĚL, T. Počítačová bezpečnost a ochrana dat. 1. vydání. Brno: Computer Press, 2004. 190 s. ISBN 80-251-0106-1
- HOTEK, M. Microsoft SQL Server 2008: krok za krokem. 1.vydání. Brno: Computer Press, 2009. 488 s. ISBN 978-80-251-2466-6
- KOTLER, P. Marketing management. 1. vydání. Praha: Grada, 2007. 788 s. ISBN 978-80-247-1359-5
- PUŽMANOVÁ, R. Moderní komunikační sítě od A do Z. 2. aktualizované vydání. Brno: Computer Press, 2006. 430 s. ISBN 80-251-1278-0
- ŘEPA, V. Podnikové procesy. 2.rozšířené vydání. Praha: Grada, 2007. 281 s. ISBN 978-80-247-2252-8
- VOŘÍŠEK, J. Principy a modely řízení podnikové informatiky. 1. vydání. Praha: Oeconomica, 2008. 446 s. ISBN 978-80-245-1440-6

Vedoucí bakalářské práce: Ing. Jiří Kříž, Ph.D.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2009/2010.

L.S.

---

Ing. Jiří Kříž, Ph.D.  
Ředitel ústavu

---

doc. RNDr. Anna Putnová, Ph.D., MBA

V Brně, dne 16.05.2010

## **Anotácia**

Bakalárska práca je zameraná na návrh a vytvorenie systému na zefektívnenie riadenia a plánovania výrobných zdrojov konkrétnej firmy. Obsah práce tvorí krátke predstavenie firmy, potom nasleduje teoretická časť, v ktorej sú popísané použité programovacie technológie. V poslednej časti je navrhnutý a vypracovaný samotný systém a sú uvedené výhody, ktoré prinesie.

## **Kľúčové slová**

Visual C# 2008, Microsoft SQL Server 2008, .Net Framework 3.5, webová aplikácia, databáza

## **Annotation**

Bachelor's thesis is focused on suggestion and creation of system for effective management and planning of production resources of a real company. Thesis content is composed of short introduction of company, then follows theoretical part, in which are described used programming technologies. In last part is suggested and elaborated system itself and are presented advantages, that it will brings.

## **Key words**

Visual C# 2008, Microsoft SQL Server 2008, .Net Framework 3.5, web application, database

## **Bibliografická citácia**

KLAUDÍNÝ, M. *Modul plánovania a riadenia výrobných zdrojov*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2010. 58 s. Vedúci bakalárskej práce Ing. Jiří Kříž, Ph.D.

## Čestné prehlásenie

Prehlasujem, že predložená bakalárska práca je pôvodná a spracoval som ju samostatne.

Prehlasujem, že citácia použitých prameňov je úplná, že som v práci neporušil autorské práva (v zmysle zákona č. 121/200 Zb. O práve autorskom a o právach súvisiacich s právom autorským).

V Brne dňa

.....

Podpis

## **Pod'akovanie**

Rád by som sa poďakoval vedúcemu bakalárskej práce pánovi Ing. Jířimu Křížovi Ph.D., ktorý mi svojimi odbornými radami pomáhal pri spracovaní tejto témy. Pod'akovanie taktiež patrí firme CONTAL OK, spol. s r.o. a jej zamestnancom, ktorí mi poskytli dôležité informácie potrebné na tvorbu tejto práce.

# Obsah

ÚVOD.....	- 9 -
<b>1 VYMEDZENIE PROBLÉMU A CIEĽ PRÁCE.....</b>	<b>- 10 -</b>
1.1 VYMEDZENIE PROBLÉMU .....	- 10 -
1.2 CIEĽ PRÁCE .....	- 10 -
<b>2 ANALÝZA SÚČASNÉHO STAVU.....</b>	<b>- 11 -</b>
2.1 O FIRME.....	- 11 -
2.1.1 Všeobecné informácie o firme.....	- 11 -
2.1.2 História.....	- 11 -
2.1.3 Obchodná činnosť .....	- 12 -
2.1.4 SWOT analýza.....	- 12 -
2.2 IMPLEMENTOVANÉ SYSTÉMY .....	- 13 -
2.3 ZHODOTENIE ANALÝZY SÚČASNÉHO STAVU.....	- 15 -
2.3.1 Využívanie IT/IS v procese plánovania a riadenia vo výrobe .....	- 15 -
<b>3 TEORETICKÉ VÝCHODISKÁ .....</b>	<b>- 16 -</b>
3.1 RELAČNÝ DATABÁZOVÝ SYSTÉM .....	- 16 -
3.1.1 Databáza .....	- 16 -
3.1.2 Databázový systém .....	- 17 -
3.2 VÝVOJOVÉ PROSTREDIE .....	- 17 -
3.2.1 Microsoft Visual Studio 2008.....	- 17 -
3.2.2 Microsoft SQL Server Management Studio.....	- 18 -
3.3 PROGRAMOVANIE.....	- 18 -
3.3.1 Objektovo orientované programovanie .....	- 18 -
3.3.2 SQL .....	- 19 -
3.3.3 XHTML.....	- 19 -
3.3.4 HTML.....	- 19 -
3.3.5 XML.....	- 20 -
3.3.6 CSS.....	- 21 -
3.3.7 C#.....	- 21 -
3.4 TECHNOLÓGIE .....	- 22 -
3.4.1 .NET Framework 3.5.....	- 22 -
3.4.2 ASP.NET.....	- 23 -
3.4.3 ASP.NET Ajax.....	- 23 -
3.4.4 LINQ.....	- 24 -
3.4.5 ADO.NET.....	- 24 -
3.5 WEBOVÉ APLIKÁCIE .....	- 25 -



<b>4</b>	<b>VLASTNÝ NÁVRH RIEŠENIA, PRÍNOS NÁVRHU RIEŠENIA.....</b>	<b>- 26 -</b>
4.1	POŽIADAVKY ZO STRANY ZADÁVATEĽA .....	- 26 -
4.1.1	<i>Implementačné požiadavky</i> .....	- 26 -
4.1.2	<i>Funkčné požiadavky</i> .....	- 26 -
4.2	ANALÝZA POŽIADAVIEK .....	- 27 -
4.2.1	<i>Analýza implementačných požiadaviek</i> .....	- 27 -
4.2.2	<i>Analýza funkčných požiadaviek</i> .....	- 27 -
4.3	NÁVRH A TVORBA SYSTÉMU .....	- 28 -
4.3.1	<i>Základné funkcie systému</i> .....	- 28 -
4.3.2	<i>Stručný postup pri navrhovaní systému</i> .....	- 28 -
4.3.3	<i>Návrh databázy</i> .....	- 28 -
4.3.4	<i>Dátová štruktúra</i> .....	- 29 -
4.3.4.1	Tabuľky .....	- 29 -
4.3.4.2	SQL skript.....	- 33 -
4.3.4.3	E-R diagram.....	- 36 -
4.3.5	<i>Návrh užívateľského rozhrania</i> .....	- 37 -
4.3.6	<i>Tvorba užívateľského rozhrania</i> .....	- 40 -
4.3.6.1	Stručné zoznámenie s vývojovým prostredím Visual Web Developer 2008.....	- 40 -
4.3.6.2	Vytvorenie prepojenia s databázou a zobrazenie údajov z databázy .....	- 41 -
4.3.6.3	Editácia údajov cez užívateľské rozhranie .....	- 42 -
4.3.6.4	Zmazanie údajov cez užívateľské rozhranie.....	- 43 -
4.3.6.5	Vkladanie údajov cez užívateľské rozhranie .....	- 43 -
4.3.6.6	Dynamické zobrazovanie údajov.....	- 44 -
4.4	FINÁLNY VZHĽAD SYSTÉMU .....	- 45 -
4.4.1	<i>Prihlasovanie</i> .....	- 45 -
4.4.2	<i>Hlavné menu</i> .....	- 46 -
4.4.3	<i>Plánovanie</i> .....	- 47 -
4.5	PRÍNOSY NÁVRHU RIEŠENIA .....	- 48 -
	<b>ZÁVER.....</b>	<b>- 50 -</b>
	<b>ZOZNAM POUŽITÝCH ZDROJOV.....</b>	<b>- 51 -</b>
	<b>ZOZNAM OBRÁZKOV .....</b>	<b>- 53 -</b>
	<b>ZOZNAM TABULIEK.....</b>	<b>- 54 -</b>
	<b>PRÍLOHY .....</b>	<b>- 55 -</b>

## ÚVOD

V posledných rokoch sme svedkami toho ako informačné technológie a informačné systémy významnou mierou vstupujú do nášho každodenného života. Týka sa to nielen súkromných osôb, ale aj právnických osôb, čiže obchodných spoločností. Pre obchodné spoločnosti je dôležité zachovanie si konkurencieschopnosti a to v dnešnej dobe znamená implementáciu rôznych aplikácií na podporu firemných procesov. Význam takýchto aplikácií spočíva vo výraznom uľahčení a zefektívnení práce zamestnancov na rôznych funkčných pozíciách.

V súčasnosti sa v tejto oblasti dostávajú do popredia tzv. webové aplikácie. V čom spočíva rozdiel medzi štandardnou aplikáciou a webovou aplikáciou?

Štandardná aplikácia vyžaduje inštaláciu a údržbu programových prostriedkov nielen na strane servera, ale tiež na strane klientskeho počítača. Prístup k serveru je možný len z počítača s nainštalovanou klientskou aplikáciou. To znamená nutnosť inštalovať a udržiavať toľko klientských aplikácií, koľko klientských počítačov má prístup k serverovej aplikácii. Príkladom takýchto aplikácií sú ERP riešenia (vnútropodnikové softvérové informačné systémy) od spoločnosti SAP AG.

Webová aplikácia vyžaduje inštaláciu a údržbu programových prostriedkov len na strane servera. Prístup k serveru je možný z ľubovoľného počítača s pripojením na internet, ktorý je vybavený štandardným webovým prehliadačom. Jednou z najznámejších webových aplikácií je určite Gmail od spoločnosti Google, Inc..

Pre využitie webovej aplikácie sa rozhodla aj slovenská firma CONTAL OK, spol. s r.o. a to v oblasti podpory riadenia a plánovania výrobných zdrojov. Keďže rozsah takéhoto typu webovej aplikácie by mnohonásobne prevyšoval rozsah bakalárskej práce, bolo rozhodnuté, že postačí vytvorenie jedného užívateľského rozhrania s názvom „Manažér“, ktoré zachytáva základnú funkcionálnosť a podstatu celého systému.

# 1 VYMEDZENIE PROBLÉMU A CIEĽ PRÁCE

## 1.1 Vymedzenie problému

Malosériová zákazková výroba v súčasnom trhovom prostredí musí pružne reagovať na požiadavky zákazníka pri zachovaní minimálnych výrobných nákladov. Zabezpečiť takýto model činnosti vyžaduje prepracovaný systém riadenia výrobných procesov. Významnú úlohu v tomto kontexte má plánovanie a riadenie výrobných zdrojov. Pod pojmom výrobné zdroje sa chápe:

1. materiál - potrebný na výrobu výrobkov (základný, pomocný, polotovary, podčasti...)
2. pracovisko - fyzické pracovné miesto kde sa vykonávajú výrobné operácie a využívajú strojné zariadenia, nástroje, prípravky...
3. pracovník - vykonáva pridelené výrobné operácie na pridelenom pracovisku a spotrebováva materiál

Na základe plánu výroby a disponibilných výrobných zdrojov rozvrhuje manažér výroby (v ďalšom len manažér), v požadovanom časovom rámci, výrobné operácie. Výsledkom rozvrhovania je také priradenie výrobných operácií k pracoviskám a pracovníkom aby sa dosiahli požadované priebežné časy výroby jednotlivých výrobkov a súčasne boli efektívne využité ľudské a strojné kapacity.

## 1.2 Cieľ práce

Cieľom práce je návrh a vytvorenie systému plánovania a riadenia výrobných zdrojov, ktorý zefektívni vedúcemu pracovníkovi spracovávať objednávky a následne ich rozpracovať do úloh určených pre jednotlivých pracovníkov. Systém by mal taktiež pracovníkovi umožniť jednoduchý prehľad pridelených úloh a s nimi spojených výrobných postupov a potrebného materiálu.

## 2 ANALÝZA SÚČASNÉHO STAVU

### 2.1 O firme

#### 2.1.1 Všeobecné informácie o firme

**Názov:** CONTAL OK

**Sídlo:** Fr.Mráza 289/2

Žilina 010 01

Slovenská republika

**Právna forma:** spoločnosť s ručením obmedzeným

**Predmet podnikania:**

- automatizované spracovanie dát
- poskytovanie softvéru - predaj hotových programov na základe zmlúv s autormi
- sprostredkovanie obchodu a marketing
- veľkoobchod so spotrebnou a výpočtovou technikou a náhradnými dielmi
- poradenská činnosť v predmete podnikania
- reklamné a propagačné činnosti
- výroba kancelárskych strojov a počítačov (vrátane prístrojov na spracovanie dát)
- systémy pre zabezpečovanie objektov
- telekomunikácie (s výnimkou činnosti § 3 odst. 2 písm. h)) [13]

#### 2.1.2 História

*„Spoločnosť CONTAL OK, s.r.o. Žilina bola založená roku 1991 so 100% zahraničnou kapitálovou účasťou. Dnes má spoločnosť základný kapitál vo výške cca 1.000.000,- Sk v zložení 75% zahraničný vlastník, 25% slovenské subjekty.*

*V dobe založenia spoločnosti bol jej jedinou aktivitou vývoj programových prostriedkov pre oblasť riadenia a monitorovania pomalých dejov (riadenie výmenníkových staníc, kotolní, čističiek odpadových vôd a pod.) na báze personálnych*

počítačov. Postupom času rozšírila spoločnosť svoje aktivity aj o vývoj technických prostriedkov pre danú oblasť, t.j. vývoj riadiacich mikropočítačov na báze jednočipových mikroprocesorov, vývoj vstupno/výstupných modulov, komunikačných procesorov a rôznych špeciálnych elektronických zariadení šitých na mieru pre zákazníka, vrátane mechanických dielov.

V roku 1993 nastalo podstatné rozšírenie záberu spoločnosti, keď vývojové aktivity boli rozšírené o výrobnú činnosť v oblasti elektroniky. Dnešný rozsah výroby predstavuje približne 2500 dosiek plošných spojov mesačne, z tohto objemu je 95% určených na export. “ [17]

### **2.1.3 Obchodná činnosť**

Spoločnosť vyrába produkty s vlastným softvérovým a firmvérovým riešením vrátane výroby a montáže mechanických dielov, skriniek a následnej kompletáže a expedície. Dodáva komplexné riešenia v oblastiach: Prístupové a bezpečnostné systémy, Systémy vyhľadávania osôb – Paging, Dochádzkové systémy, Stravovacie systémy, Návštevne systémy a systém TELECALL.

Firma má obchodné aktivity na slovenskom a švédskom trhu. Na švédskom trhu firma predáva svoje produkty partnerskej firme CONTAL SECURITY AB. Podstatná časť produkcie a vývojových prác je exportovaná na škandinávsky trh.

### **2.1.4 SWOT analýza**

#### **Silné stránky:**

- pôsobnosť na domácom aj zahraničnom trhu
- vlastný vývoj a výroba hardvéru a softvéru
- komplexnosť vývoja (software, hardware, firmware, mechanické celky)
- schopnosť vyrábať jednoduché aj zložité výrobky v jednotkových až tisícových sériách
- vysoká profesionalita zamestnancov
- schopnosť navrhovať dosky plošných spojov a mechanické celky

**Slabé stránky:**

- absencia niektorých výrobných technológií spôsobuje závislosť od dodávateľov
- nedodržiavanie termínov dodávok a kvality zo strany dodávateľov
- kapacitné poddimenzovanie v oblasti vývoja a výroby za určitých hraničných okolností
- neschopnosť cenovo konkurovať veľkým spoločnostiam

**Príležitosti:**

- vznik legislatívy vyžadujúcej automatizované spracovanie dochádzky a jej prevod do miezd

**Hrozby:**

- prienik konkurencie (napr. z Číny) na slovenský trh s cenami, ktorým sa nedá konkurovať (hlavne v oblasti HW)
- hrozba zo strany konkurencie – pohltenie väčším subjektom, cieľené očierňovanie
- „vykupovanie mozgov“ – odchod zamestnancov z firmy na základe ponúknutia výhodnejších mzdových podmienok
- veľký problém môže byť strata strategického subdodávateľa z dôvodu napr. bankrotu v prípade ak nie sú alternatívy.

## 2.2 Implementované systémy

V súčasnosti firma využíva niekoľko systémov, ktoré zefektívňujú mnohé procesy. Niektoré z týchto systémov sú ich vlastným produktom.

### 1. Integrovaný systém MAGMA

- integruje účtovný systém, skladové hospodárstvo, MTZ (materialno-technické zabezpečenie) a technickú prípravu výroby

### 2. Personálny a mzdový systém

### 3. Dochádzkový systém Dochadzka 2000

- vlastný produkt

### 4. Prístupový a zabezpečovací systém Access 2000

- vlastný produkt

#### 5. IBM LOTUS NOTES 8

– aplikácia, ktorá pomáha organizáciám zlepšiť vnútornú kooperáciu, zvýšiť produktivitu a zefektívniť podnikové procesy. Nadstavba Q - Integra, ktorá predstavuje integrovaný softvérový produkt pre podporu riadenia kvality, rozhodovania a tímovú spoluprácu. V rámci tejto aplikácie majú aj svoj firemný e-mail.

## **2.3 Zhodotenie analýzy súčasného stavu**

V súčasnosti firma nedisponuje žiadnym informačným systémom prípadne aplikáciou, ktorá by robila podporu pre riadenie výroby. Celý proces zadávania a pridelovania prebieha nanajvýš neefektívnym spôsobom. Zadania úloh sa doručujú rôznymi spôsobmi a v rôznej forme, čo práve nepridáva na prehľadnosti tohto systému. Manažér v úseku výroby nemá ucelený prehľad o vyťažnosti svojich ľudí, či majú všetci zaplánované ďalšie úlohy, či náhodou nevznikajú prestoje a taktiež nie je ucelený prehľad o obsadenosti výrobných miest a strojných zariadení. Všetky tieto problémy vplyvajú jednak na efektivitu celého procesu výroby a zbytočne predlžujú proces rozplánovania úloh.

### ***2.3.1 Využívanie IT/IS v procese plánovania a riadenia vo výrobe***

Z hľadiska informačných technológií firma pre podporu procesu zadenovania úloh používa jeden z nástrojov kancelárskeho balíka Microsoft Office 2003. Spôsob uchovávanía informácií týkajúcich sa plánovania a riadenia prebieha prácnym zápisom do pracovných listov Microsoft Excelu. Jedná sa o zoznamy zákaziek, výrobných operácií, výrobkov, materiálov, zariadení a tak ďalej, nad ktorými žiaľ nie je možné vykonávať prípadné dotazovanie. Toto by bolo možné len v prípade, keby boli informácie uložené v štruktúrovanej databáze, ktorá v súčasnosti neexistuje.

Celý priebeh plánovania spočíva v zápise do pracovného listu: aká výrobná operácia sa bude vykonávať, kto ju bude vykonávať a čo na to bude potrebovať. V prípade akejkoľvek zmeny (napríklad zmena priority, počtu výrobkov, ...) je potrebné daného človeka kontaktovať buď osobne, telefonicky alebo elektronickou poštou, čo je značne neefektívne.



## 3 TEORETICKÉ VÝCHODISKÁ

### 3.1 Relačný databázový systém

Na vysvetlenie, čo vlastne relačný databázový systém predstavuje, je potrebné definovať pojem databáza, vysvetliť kľúčové termíny týkajúce sa tohto pojmu, definovať ich základne vzťahy a v neposlednom rade objasniť databázový systém ako taký.

#### 3.1.1 Databáza

Databáza predstavuje určitý súbor štruktúrovaných a usporiadaných údajov, ktorá je špecificky navrhnutá pre správu veľkého objemu dát. Spôsob ich uchovávanía umožňuje užívateľom ľahšiu správu a získavanie informácií. Serverové databázy sú navrhnuté tak, aby slúžili väčšiemu počtu užívateľov a umožnili spracovávať viacero požiadaviek súčasne.

Údaje v databázach sú uchovávané v tabuľkách (relácie, entity), ktoré sú tvorené riadkami (záznamy, usporiadané n-tice) a stĺpcami (polia, atribúty).

Kardinalita predstavuje jedinečnosť hodnôt v danom stĺpci tabuľky. Stupeň kardinality vyjadruje počet entít, ku ktorým sa dá prostredníctvom vzťahov priradiť iná entita inej tabuľky. Tieto vzťahy existujú vo všetkých databázach a dajú sa klasifikovať:

- vzťah 1:1 - pre každý riadok v tabuľke A existuje najviac jeden riadok v tabuľke B a naopak
- vzťah 1:M – pre každý riadok v tabuľke A existuje mnoho súvisiacich riadkov v tabuľke B
- vzťah M:M – pre každý riadok tabuľky A existuje mnoho súvisiacich riadkov v tabuľke B a naopak

Pre vytvorenie väzieb medzi tabuľkami je nutné definovať tzv. kľúče. Kľúč je jeden alebo viac atribútov relácie, ktoré sa používajú na identifikáciu riadku. Predovšetkým sa pracuje s dvoma typmi kľúčov – primárnym a cudzím kľúčom. Primárny kľúč je atribút alebo skupina atribútov, ktorých hodnoty jedinečným spôsobom identifikujú záznamy v tabuľke. Cudzí kľúč je určitý atribút, ktorý vytvára s primárnym kľúčom inej tabuľky väzbu medzi tabuľkami.

Normalizácia je proces minimalizácie až eliminácie duplicitných dát v relačných databázach. Normalizované návrhy by v ideálnom prípade mali udržiavať každú dátovú položku na jedinom mieste. Normalizácia zahŕňa niekoľko formálnych procesov tzv. normálnych foriem. Podľa niektorých zdrojov ich existuje 5 až 6, ale vo väčšine prípadov stačí aplikácia prvých troch:

- prvá normálna forma – požaduje, aby každý atribút obsahoval len atomické hodnoty, čo znamená, že hodnoty sú ďalej nedeliteľné
  - druhá normálna forma - požaduje, aby bola relácia v prvej normálnej forme a súčasne, aby každý neklúčový atribút bol na celom primárnom kľúči plne funkčne závislý
  - tretia normálna forma – požaduje, aby bola relácia v druhej normálnej forme a súčasne, aby všetky neklúčové atribúty boli navzájom nezávislé
- [3]

### **3.1.2 Databázový systém**

Databázový systém je program, ktorý umožňuje vytvárať a udržiavať databázy a pristupovať k nim. Ponúka možnosť písať dotazy nad databázou k vykonávaniu požadovaných akcií (pridávanie, úprava, mazanie údajov ... ). Určitým typom databázového systému je relačný databázový systém, ktorý uchováva údaje vo forme súvisiacich tabuliek. [3]

## **3.2 Vývojové prostredie**

Vývojové prostredie, označované aj ako IDE (Integrated Development Environment) predstavuje komplexný systém na vývoj softvérových aplikácií. V tejto práci sa využili prostredia Microsoft Visual Studio 2008 a Microsoft SQL Server Management Studio.

### **3.2.1 Microsoft Visual Studio 2008**

Microsoft Visual Studio 2008 je plne integrované vývojové prostredie, ktoré bolo navrhnuté tak, aby písanie kódu bolo čo najjednoduchšie. Visual Studio ponúka

veľmi dobre prepracované rozhranie pre prácu s viacerými dokumentami súčasne, umožňujúce robiť všetky operácie spojené s vývojom kódu. Ponúka viacero funkcií ako napríklad textový editor, editor návrhu vzhľadu, pomocné okná, možnosť prekladu v rámci prostredia, integrovaný ladiaci program, integrovaná nápoveda MSDN, prístup k iným programom. [5]

### **3.2.2 *Microsoft SQL Server Management Studio***

SQL Server Management Studio je nástroj poskytujúci všetky možnosti správy služieb systému SQL Server. Umožňuje vytvárať a spúšťať kód jazyka Transact-SQL (T-SQL), dotazovacieho jazyka MDX (Multidimensional Expression), rozšírenia DMX (Data Mining Extensions) a jazyka XMLA (XML for Analysis). [2]

## **3.3 Programovanie**

Programovanie je proces tvorby, písania, ladenia a testovania zdrojového kódu rôznych algoritmov a počítačových programov. Zdrojový kód predstavuje formuláciu postupu riešenia daného problému a je písaný v programovacích jazykoch, ktoré sú určitým komunikačným nástrojom medzi programátorom a počítačom.

### **3.3.1 *Objektovo orientované programovanie***

Objektovo orientované programovanie predstavuje programovanie pomocou objektov. Objekt je entita, ktorá má:

- stav – zahŕňa všetky vlastnosti objektu a ich hodnoty
- správanie - predstavuje konanie objektu pri zmenách stavu a aktivácií operácií
- identita - predstavuje jeho jednoznačnú identifikáciu

Objektovo orientovaný program sa vykonáva ako vzájomná interakcia objektov. Správanie sa objektov definuje jeho typ, ktorý sa označuje ako trieda (class). V zdrojových kódach sa najčastejšie definujú triedy a objekty predstavujú ich inštancie. Preto sa niekedy zdá, že sa jedná o programovanie pomocou tried.

### **3.3.2 SQL**

SQL (Structured Query Language) je počítačový jazyk určený na definovanie a manipuláciu dát. V súčasnosti je najpoužívanejší programovací jazyk v relačných databázových systémoch, u ktorých sa stal štandardnou výbavou. Základné operácie jazyka SQL sú definícia dát ( pomocou príkazov CREATE, ALTER, DROP), manipulácia s datami (pomocou príkazov SELECT, INSERT, DELETE a UPDATE) a riadenie dát (pomocou príkazov REVOKE a GRANT). Microsoft SQL Server využíva určitú formu jazyka SQL a to už spomínaný T-SQL. Odlišnosť Transact-SQL od tradičného SQL spočíva v integrácii niektorého programovacieho jazyka ako napríklad Java, C#, C++. [2]

### **3.3.3 XHTML**

XHTML (Extensible Hypertext Markup Language) je značkový jazyk podobný HTML, ale s prísnejšou syntaxou. Vyvinutý a štandardizovaný bol konzorciom W3C v roku 2000. Dôvodom vývoja bola potreba zlúčenia HTML a XML. V súčasnosti existuje viacero verzií XHTML: XHTML 1.0 Strict, XHTML 1.0 Transitional a XHTML 1.0 Frameset.

### **3.3.4 HTML**

HTML (HyperText Markup Language) je jazyk určený na tvorbu webových stránok a zobrazovanie informácií vo webovom prehliadači. Dôraz je kladený na prezentáciu údajov. Je potrebné, aby jednotlivé časti, ktoré majú byť prezentované užívateľovi boli rozdelené do skupín s rovnakým významom (nadpis, zoznam, tabuľka, text, ...).

HTML kód je zložený z klasického textu a značiek (tzv. tagy). Značky sú vždy uzavreté do ostrých zátvoriek, ktoré majú za úlohu vymedziť, ohraničiť a oddeliť určitú časť textu.

#### **Rozdelenie značiek (tagov)**

Značky rozdeľujeme podľa niekoľkých aspektov.

Rozdelenie na párové a nepárové:

- *párové značky* – začínajú <x> a končia </x> ( x – predstavuje určitú značku napríklad *body*, *title*)
- *nepárové značky* – umiestňujú prvky v html dokumente, napríklad <br> značka slúžiaca na vytvorenie prázdneho riadku v dokumente

Rozdelenie podľa funkcie v dokumente:

- *Popisné značky (logické, sémantické)* – ich účelom je popis charakteru elementu alebo určenie významu textu; napríklad titulok (<title>), alebo adresa (<address>).
- *Štrukturálne značky* – ako je už z názvu jasné ich hlavnou úlohou je forma a štruktúra dokumentu. Jedná sa o nadpisy (<h1>,<h2>,<h3>,...) a odstavce (<p>).
- *Štylistické značky* – charakterizujú vzhľad elementu pri zobrazení. Napríklad zvýraznenie textu pomocou hrubého písma (<b>).

### **Základná štruktúra HTML dokumentu**

- *DTD (Document Type Definition)* – každý dokument by mal začínať definíciou typu. To znamená, že sa určí množina značiek, ktoré môžu byť použité v dokumente. Deklaruje sa na začiatku dokumentu: <!DOCTYPE... .
- *koreňový element html* – ohraničujú ho párové značky <html> a </html>. Predstavuje celý dokument, čiže ďalšie elementy sú obsiahnuté v ňom:
  - *hlavička dokumentu* – vymedzená je párovými značkami <head> a </head>. V tomto elemente sú definované metadata, názov dokumentu (title), kódovanie, popis, kľúčové slová, jazyk.
  - *telo dokumentu* – je ohraničené párovými značkami <body> a </body>. Tento element obsahuje text dokumentu. [16]

### **3.3.5 XML**

XML (Extensible Markup Language) sa využíva pre chod a správu procesov v aplikáciách a výmenu údajov medzi aplikáciami. Umožňuje jednoduché vytváranie konkrétnych značkovacích jazykov na rôzne účely. Taktiež bol vyvinutý konzorciom

W3C a predstavuje zjednodušenú podobu staršieho jazyka SGML (Standard Generalized Markup Language – univerzálny značkovací jazyk, ktorý umožňuje definovať značkovacie jazyky ako svoje podmnožiny).

### **Výhody XML**

- hierarchická štruktúra a jednoduchý spôsob kódovania
- široko prijímaný ako štandard so schopnosťou existencie na akejkolvek platforme
- používaný stále väčším počtom aplikácií a programovacích jazykov
- vytváranie štruktúrovaných dokumentov veľmi flexibilným spôsobom

### **3.3.6 CSS**

CSS (Cascading Style Sheets) je programovací jazyk na vizuálne formátovanie HTML, XHTML a XML dokumentov. CSS kód je možné presunúť do externých súborov, čím sa zmenší dátová veľkosť a jediným súborom je možné meniť celý vzhľad stránky. CSS zaručuje rovnaké vykresľovanie vo všetkých webových prehliadačoch, avšak nie vždy je jednoduchá optimalizácia kódu pre všetky prehliadače. CSS sa v HTML dokumente môže aplikovať viacerými spôsobmi:

- aplikácia na konkrétny element pomocou atribútu style
- aplikácia na celý dokument v hlavičke stránky pomocou párových značiek <style> a </style>
- pripojenie externého CSS dokumentu s nadefinovanými štýlmi pomocou elementu link

### **3.3.7 C#**

Spoločnosť Microsoft oficiálne popisuje jazyk C# ako „jednoduchý, moderný, objektovo orientovaný a typovo bezpečný programovací jazyk, ktorý je odvodený od jazykov C, C++ a Java“. Vyvinutý bol špeciálne pre technológiu .NET Framework. Cieľom pri vývoji bolo skĺbenie vyspelého jazyka ako C++ s možnosťou rýchleho programovania, ktoré bolo možné pri jazykoch ako Visual Basic a Delphi.

Syntakticky sa jazyk C# podobá jazykom C++ a Java do takej miery, že mnohé z kľúčových slov sú navlas rovnaké. Taktiež výpis kódu je nápadne podobný. Napriek

tomu výučba jazyka C# je omnoho jednoduchšia a jeho obtiažnosť sa vyrovná jazyku Java. Výhodou jazyka C# oproti spomínaným jazykom Java a C++ je štruktúra, ktorá lepšie vyhovuje moderným vývojovým prostrediam a poskytuje jednoduchosť používania jazyka, aká bola známa u jazyka Visual Basic. [8]

#### **Vlastnosti jazyka C#:**

- plný prístup ku knižnici základných tried .NET a takisto jednoduchá dostupnosť rozhrania Windows API (Application Programming Interface)
- pomocou jazyka C# sa dajú písať dynamické stránky ASP.NET a webové služby založené na XML
- plná podpora tried a objektovo orientovaného programovania
- konzistentná a vhodne definovaná sada základných nástrojov
- integrovaná podpora automatického generovania dokumentácie vo formáte XML
- možnosť označiť triedy a metódy užívateľsky definovanými atribútmi.

### **3.4 Technológie**

#### **3.4.1 .NET Framework 3.5**

Platforma .NET predstavuje aplikačné rozhranie (API) k programovaniu pre Windows. Vývojom .NET chcela spoločnosť Microsoft prísť s jednoduchou, ale napriek tomu pokročilou sadou jazykov, prostredí a vývojových nástrojov, ktoré vývojárom umožnia ľahkú tvorbu špičkových aplikácií. Tento ich cieľ sa im podarilo naplniť už v roku 2002, keď prišli s prvou verziou .NET Framework 1.0, ktorá sa stretla s veľkým úspechom. V roku 2005 bola predstavená verzia 2.0, ktorá bola považovaná za kľúčové vydanie tejto platformy. Súčasná verzia 3.5 síce neprinesla tak rozsiahle zmeny ako 2.0, ale je považovaná za významnú verziu, ktorá priniesla mnoho nových podstatných vlastností. Novinkou vo verzií 3.5 sú napríklad technológie ASP.NET Ajax a LINQ, ktorým sa budem venovať ďalej. [5]

#### **Výhody .NET**

- objektovo orientované programovanie

- jazyková nezávislosť - .NET obsahuje mnoho jazykov, ako napríklad Visual Basic, C#, J# alebo C++, ktoré sa prekladajú do spoločného sprostredkujúceho jazyka (Intermediate Language), čo umožňuje ich vzájomnú spoluprácu
- lepšia podpora dynamických webových stránok –pomocou novej technológie ASP.NET
- podpora webových služieb
- Visual Studio 2008 – integruje najlepšie funkcie jazykovo špecifických prostredí Visual Studia .NET 2002/2003/2005 a Visual Studia 6
- C# - nový objektovo orientovaný programovací jazyk určený na spoluprácu s .NET [5]

### **3.4.2 ASP.NET**

Technológia ASP (Active Server Pages) od spoločnosti Microsoft, ktorá je súčasťou platformy .NET Framework, umožňuje tvorbu dynamických webových dokumentov na webovom serveri po vyžiadaní protokolom HTTP<sup>1</sup>. Webová stránka ASP je v podstate súbor HTML s vloženými úsekmi kódu v jazyku JavaScript, ktoré sa vykonávajú na strane servera. Výhoda oproti podobným, už existujúcim, technológiám (PHP alebo ASP) spočíva v integrácii do systému .NET Framework, čím je zaručená podpora zo strany jazyka C#. Pomocou ASP sa dajú jednoducho vytvárať klientské aplikácie založené na prehliadači – tzv. webové aplikácie. [5]

### **3.4.3 ASP.NET Ajax**

Technológia Ajax (Asynchronous JavaScript and XML) predstavuje vylepšenie užívateľských rozhraní webových aplikácií prostredníctvom tzv. asynchrónnych spätných odosielaní a dynamickej manipulácie s webovou stránkou na strane klienta. V podstate to znamená, že umožňuje vývoj interaktívnych webových aplikácií, ktoré umožňujú meniť obsah bez potreby znovu načítania zo serveru. Aplikácie využívajúce túto technológiu poskytujú užívateľom oveľa komfortnejšie prostredie no nevyhnutnosťou pre ich bezproblémový chod sú modernejšie webové prehliadače.

---

<sup>1</sup> HTTP (Hypertext Transfer Protokol) – metóda prepravy informácií na webe



V minulosti bola potrebná inštalácia niekoľkých komponentov, aby bola možná tvorba stránok využívajúcich technológiu ASP.NET Ajax pomocou .Net Frameworku 2.0. Vo verzii .NET Framework 3.5 je podpora ASP.NET Ajax priamo zakomponovaná. Všetky nastavenia týkajúce sa Ajax-u sú umiestnené v súbore *Web.config*. [5]

### **3.4.4 LINQ**

Technológia LINQ (Language Integrated Query – dotazy integrované do jazyka) je novinkou od spoločnosti Microsoft uvedenou na trh spoločne s rozhraním .NET Framework 3.5 a s Visual Studiom 2008. Umožňuje revolučný spôsob práce s dátami. Prichádza s novou jednotnou koncepciou dotazovania nad rôznymi typmi dátových zdrojov.

#### **Výhody LINQ**

Ponúka spoločnú syntax pre dotazovanie akéhokoľvek dátového zdroja, to znamená, že rovnakým spôsobom sa vytvárajú dotazy nad dokumentom XML ako aj nad databázou SQL. Upevňuje prepojenie medzi relačnými dátami a objektovo orientovaným programovaním. Počas kompilácie sa zachytávajú mnohé chyby. Nápoveda IntelliSense<sup>1</sup> a podpora pre ladenie skraca dobu vývoja aplikácií.

### **3.4.5 ADO.NET**

Predstavuje knižnicu, ktorá je súčasťou platformy .NET Framework. Vytvorená bola v roku 2001 spoločnosťou Microsoft. Táto knižnica tvorí súbor tried, ktoré umožňujú programátorom pristupovať a pracovať s databázou. Väčšina profesionálnych aplikácií potrebuje k svojej funkčnosti spolupracovať s nejakým typom dátového. Túto spoluprácu umožňuje knižnica ADO.NET určená. [15]

---

<sup>1</sup> IntelliSense – funkcia, ktorá pomáha programátorovi generovať kód počas písania priamo v Code Editor

### **3.5 Webové aplikácie**

Webová aplikácia je aplikácia, ktorá je prístupná pomocou webového prehliadača cez lokálnu alebo globálnu sieť. Princíp fungovania tejto aplikácie spočíva v komunikácii klienta a servera. Na strane klienta sa nachádza webový prehliadač. Niekedy sa tomu hovorí aj tzv. tenký klient, pretože klient nepozná princípy a algoritmy aplikácie. Na strane servera je samotná aplikácia a databáza potrebných údajov. V súčasnosti sa webové aplikácie tešia obrovskej popularite, napríklad produkty od spoločnosti Google, Inc. (Calendar, Gmail, ...)

#### **Výhody webových aplikácií**

- dostupnosť kdekoľvek tam, kde je prístup na internet bez potreby inštalácie softvéru
- aktualizácia prebieha jednoduchým spôsobom (keďže aplikácia beží na serveri, stačí, že ju prevádzkovateľ vykoná tam).

## **4 VLASTNÝ NÁVRH RIEŠENIA, PRÍNOS NÁVRHU RIEŠENIA**

Na úvod tejto sekcie bakalárskej práce budú definované požiadavky zo strany zadávateľa nasledované vypracovaným podrobným postupom návrhu riešenia. V ďalšej časti je uvedená samotná realizácia a na záver sú spomenuté výhody navrhnutého spôsobu riešenia.

### **4.1 Požiadavky zo strany zadávateľa**

Nižšie uvedené požiadavky boli konzultované a definované zamestnancami z oblasti výroby spoločnosti CONTAL OK, s.r.o., o ich skompletizovanie a naformulovanie sa postaral riaditeľ výroby.

#### ***4.1.1 Implementačné požiadavky***

Modul plánovania a riadenia výrobných zdrojov (MPRVZ) je samostatný modul vytvorený ako webová aplikácia spolupracujúca s Microsoft SQL Serverom.

Požadované SW prostredie a technológie:

- Operačné systémy Microsoft Windows XP / Vista
- Microsoft SQL databáza
- Microsoft SQL server
- ASP.NET
- webový prehliadač

#### ***4.1.2 Funkčné požiadavky***

- prehľadný a efektívny spôsob plánovania a riadenia výrobných zdrojov
- predefinované plánovacie formuláre priradovania výrobných operácií k pracoviskám a pracovníkom
- sledovanie rozpracovanosti z hľadiska časového a predmetného
- historické záznamy významných údajov

- prostredníctvom účelovo orientovaných užívateľských rozhraní „Manažér“, „Dohľad“ a „Pracovník“ fyzicky realizovaných na PC vyhodnocujú užívatelia poskytované údaje a vkladajú požadované údaje.

## **4.2 Analýza požiadaviek**

Po konzultácií požiadaviek s vedúcim bakalárskej práce mi bolo odporučené riešiť len časť celého systému z dôvodu, že celý požadovaný systém by výrazne prekročoval rozsah bakalárskej práce. Zhodli sme sa, že bude postačovať návrh a vytvorenie len jediného užívateľského rozhrania „Manažér“, ktoré bude zachytávať podstatu fungovania celého systému. Dopracovanie do finálnej podoby by mohlo byť obsahom diplomovej práce.

### ***4.2.1 Analýza implementačných požiadaviek***

Ako je už zo zoznamu SW prostredí a technológií jasné, firma vyžaduje výhradne použitie produktov spoločnosti Microsoft. Dôraz je na to kladený hlavne z toho dôvodu, že väčšina doteraz používaných systémov vo firme je fungujúcich na platforme od spoločnosti Microsoft, takže by nemali nastať problémy pri implementácii nového systému ako aj pri vzájomnej kompatibilite s ostatnými vo firme už existujúcimi systémami.

### ***4.2.2 Analýza funkčných požiadaviek***

Z funkčného hľadiska je najdôležitejšia samozrejme funkčnosť celého systému, aby vykonával to, čo bolo definované zadávateľom. Nemenej dôležitá je tzv. „user-friendly“ vlastnosť, to znamená jednoduchosť a intuitívnosť používania systému, bez potreby náročnejšieho školenia.

## **4.3 Návrh a tvorba systému**

### **4.3.1 Základné funkcie systému**

Hlavné funkcie, ktoré by mal systém vykonávať:

- prihlásenie užívateľa
- vytvorenie, editácia a zmazanie zákazky, výrobku, výrobných operácií zariadenia a materiálu
- rozplánovanie zákazky na jednotlivé výrobné operácie na základe výrobných postupov výrobkov obsiahnutých v zákazke
- pridelenie výrobných operácií jednotlivým zamestnancom a zariadeniam.
- náhľady na rôzne typy údajov

### **4.3.2 Stručný postup pri navrhovaní systému**

1. Návrh a vytvorenie databázy.
2. Naplnenie databázy testovacími údajmi pre záverečné overenie funkčnosti systému.
3. Návrh a vytvorenie užívateľského rozhrania vo Visual Studiu 2008.
4. Prepojenie databázy s užívateľským rozhraním.
5. Zfunkčnenie ovládacích prvkov za pomoci programovacieho jazyka C#.
6. Otestovanie funkčnosti.
7. Záverečná kontrola systému.

### **4.3.3 Návrh databázy**

Existencia databázy pri akejkoľvek webovej aplikácii je rozhodne nevyhnutná, či už z pohľadu čerpania, spracovávania alebo ukladania údajov. Taktiež pri tejto problematike je práca s údajmi nepostrádateľná, preto je potrebné navrhnúť vhodnú štruktúru databázy. Tým sa myslí vhodne stanoviť počet tabuliek, definovať atribúty obsiahnuté v nich, určiť primárne a cudzie kľúče a vytvoriť väzby medzi jednotlivými tabuľkami.

Databáza webovej aplikácie je vytvorená pre Microsoft SQL Server 2008 a jej tvorba prebiehala vo vývojovom prostredí Microsoft SQL Server Management Studio.

### 4.3.4 Dátová štruktúra

#### 4.3.4.1 Tabuľky

Základom celej databázy sú tabuľky „Zamestnanec“, „Material“, „Zariadenie“ a „Zakazka“, ktoré predstavujú 4 hlavné vstupy do systému. Údaje do tabuliek „Zamestnanec“ a „Material“ budú čerpané z personálneho a skladového systému.

Zamestnanec	
ID_zam	int identity (1,1) PK
meno	varchar (20)
priezvisko	varchar (25)
ID_vzd	integer FK
ID_poz	integer FK

Tabuľka č.1: Zamestnanec

Zakazka	
ID_zak	int identity (1,1) PK
nazov	varchar (100)
popis	varchar (max)
datum_pri	datetime
datum_kon	datetime
poznanky	varchar (max)

Tabuľka č.2: Zákazka

Zariadenie	
ID_zar	int identity (1,1) PK
ID_typ	integer FK
nazov	varchar (100)
popis	varchar (max)

Tabuľka č.3: Zariadenie

Material	
ID_mat	int identity (1,1) PK
nazov	varchar (100)
popis	varchar (max)

Tabuľka č.4: Materiál

Tabuľka „Vyrobna\_operacia“ predstavuje akési nepriame spojenie všetkých štyroch uvedených tabuliek. Nepriame z toho dôvodu, lebo medzi nimi sa nachádza ešte množstvo tabuliek, ktoré vznikli procesmi normalizácie a dekompozície. Pod výrobnou operáciou je myslená fáza výroby nejakého produktu, napríklad sa môže jednať o navrtanie uchytých otvorov do plastového dielu bezkontaktnéj čítačky čipových kariet. Významným atribútom tabuľky je doba trvania výrobnéj operácie. Tento atribút je nepostrádateľný v procese časového rozplánovania.

Vyrobna_operacia	
ID_ope	int identity (1,1) PK
nazov	varchar (100)
popis	varchar (max)
doba_trvania	time

Tabuľka č.5: Výrobná operácia

Ďalším typom tabuliek sú tzv. číselníky, ktoré predstavujú tabuľky s fixným počtom položiek pozostávajúcich z dvoch atribútov a to z identifikátoru a názvu. Pre ozrejmienie sa jedná o tabuľky: „Pozicia“, „Vzdelanie“ a „Kvalifikacia“.

Pozicia	
ID_poz	int identity (1,1) PK
nazov	varchar (50)

Vzdelanie	
ID_vzd	int identity (1,1) PK
nazov	varchar (50)

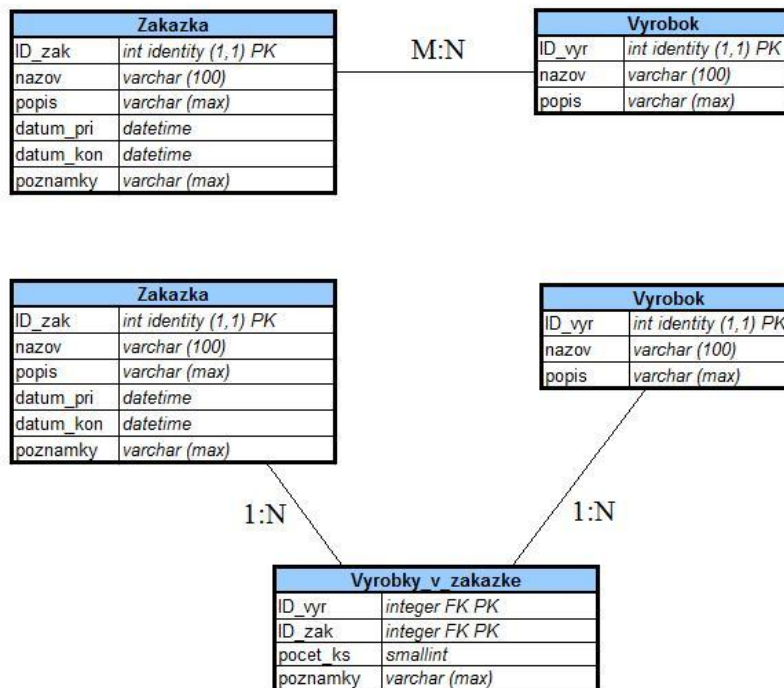
Kvalifikacia	
ID_kva	int identity (1,1) PK
nazov	varchar (50)

Tabuľka č.6: Pozícia

Tabuľka č.7: Vzdelanie

Tabuľka č.8: Kvalifikácia

Počas tvorby databázy vznikla na mnohých miestach medzi tabuľkami väzba M:N, ktorá je vo výslednej databázovej štruktúre nežiaduca a je potrebné ju dekomponovať. Dekompozíciou väzby M:N vznikla tretia tabuľka, ktorá je na pôvodné dve previazaná väzbami 1:N. Okrem iného obsahuje ich primárne kľúče, ktorých kombinácia tvorí primárny kľúč novovytvorenej entity. Proces dekompozície je ukazaný na názornom príklade z databázy:



OBRÁZOK Č. 1: DEKOMPOZÍCIA VÄZBY M:N

### Interpretácia:

*Vzťah M:N* – Jedna zakázka môže obsahovať N výrobkov a naopak jeden výrobok sa môže nachádzať v M zákazkách.

*Vzťah 1:N* – Jedna zákazka môže obsahovať N výrobkov v zákazke a jeden výrobok v zákazke sa nachádza práve v jednej konkrétnej zákazke.

Týmto procesom dekompozície vzniklo v databáze 8 nových tabuliek: „Material\_v\_operacii“, „Vyroby\_v\_zakazke“, „Vyrobný\_postup“, „Potrebne\_kval\_vo\_vyr\_operacii“, „Kvalifikacie\_zamestnanca“, „Typ\_zariadeni\_v\_operacii“, „Cinnosti\_zamestnanca“ a „Cinnosti\_zariadenia“.

Tabuľky „Cinnost“, „Cinnosti\_zamestnanca“ a „Cinnosti\_zariadenia“ vymedzujú vlastnú kategóriu tabuliek. Sú to tabuľky, ktorých obsah sa v porovnaní s ostatnými najdynamickejšie mení a sú veľmi dôležitými tabuľkami z pohľadu ukladania a manipulovania dát, týkajúcich sa zaeľovania úloh. Čo si pod tým predstaviť? V zásade „Vyrobná\_operacia“ a „Cinnost“ sú obsahovo veľmi príbuzné, až



na to, že „Vyrobna\_operacia“, je akási historicky nadobudnutá informácia. Jednoducho povedané, na výrobu výrobku treba vykonať nejaké množstvo výrobných operácií a tie trvajú nejakú dobu. Zato „Cinnost“ je akýsi určitý časový úsek začínajúci v konkrétny čas, počas ktorého sa danej výrobnej operácií venuje konkrétny zamestnanec, prípadne zariadenie. Táto tabuľka taktiež obsahuje čas ukončenia činnosti, ktorý v prvom rade slúži ako odozva od pracovníkov, či časová norma nie je príliš prísne nastavená alebo naopak, či nie je príliš benevolentná. Toto sa zistí porovnaním reálnej hodnoty (rozdiel medzi počiatočným a koncovým časom činnosti) a predpokladaným časom (doba trvania výrobných operácií vynásobená množstvom výrobkov). Vzťah týchto tabuliek je 1:N nakoľko jedna výrobná operácia sa môže vykonávať v N časových intervaloch a jedna konkrétna činnosť sa viaže na jedinú jednu výrobnú operáciu.

Cinnost	
ID_cin	<i>int identity (1,1) PK</i>
ID_ope	<i>integer FK PK</i>
cas_zac	<i>datetime</i>
poznamky	<i>varchar (max)</i>
cas_kon	<i>datetime</i>

Tabuľka č.9: Činnosť

Cinnosti_zariadenia	
ID_cin	<i>integer FK PK</i>
ID_zar	<i>integer FK PK</i>

Tabuľka č.10: Činnosť zariadenia

Cinnosti_zamestnanca	
ID_cin	<i>integer FK PK</i>
ID_zam	<i>integer FK PK</i>

Tabuľka č.11: Činnosť zamestnanca

Tabuľka „Typ\_zariadenia“ vytvára podmienky pre efektívnejšiu manipuláciu so zariadeniami v užívateľskom rozhraní a zachováva podobné štrukturovanie údajov ako vo vetve so zamestnancami.

Typ_zariadenia	
ID_typ	<i>int identity (1,1) PK</i>
nazov	<i>varchar (100)</i>
popis	<i>varchar (max)</i>

Tabuľka č.12: Typ zariadenia

Primárny kľúč väčšiny tabuliek má dátový typ nastavený ako *int identity (1,1)*, vďaka čomu sa zabezpečí vygenerovanie vždy unikátneho primárneho kľúča. Taktiež

vo väčšine tabuliek sa nachádza položka označená ako *poznámky*, ktorá vo finálnom systéme bude slúžiť na komentáre alebo prípadné dotazy.

#### 4.3.4.2 SQL skript

```
CREATE TABLE Vzdelanie
(ID_vzd int identity (1,1) PRIMARY KEY,
nazov varchar (50));
```

```
CREATE TABLE Pozicia
(ID_poz int identity (1,1) PRIMARY KEY,
nazov varchar (50));
```

```
CREATE TABLE Kvalifikacia
(ID_kva int identity (1,1) PRIMARY KEY,
nazov varchar (50));
```

```
CREATE TABLE Typ_zariadenia
(ID_typ int identity (1,1) PRIMARY KEY,
nazov varchar (100),
popis varchar (max));
```

```
CREATE TABLE Zariadenie
(ID_zar int identity (1,1) PRIMARY KEY,
ID_typ integer foreign key references Typ_zariadenia(ID_typ),
nazov varchar (100),
popis varchar (max));
```

```
CREATE TABLE Material
(ID_mat int identity (1,1) PRIMARY KEY,
nazov varchar (100),
popis varchar (max));
```

```
CREATE TABLE Vyrobok
(ID_vyr int identity (1,1) PRIMARY KEY,
nazov varchar (100),
popis varchar (max));
```

```
CREATE TABLE Vyrobná_operacia
(ID_ope int identity (1,1) PRIMARY KEY,
nazov varchar (100),
popis varchar (max),
doba_trvania time);
```

```
CREATE TABLE Zakazka
(ID_zak int identity (1,1) PRIMARY KEY,
nazov varchar (100),
popis varchar (max),
datum_pri datetime,
datum_kon datetime,
poznámky varchar (max),);
```

```
CREATE TABLE Zamestnanec
```

```

(ID_zam int identity (1,1) PRIMARY KEY,
meno varchar (20),
priezvisko varchar (25),
ID_vzd integer foreign key references Vzdelanie(ID_vzd),
ID_poz integer foreign key references Pozicia(ID_poz),);

CREATE TABLE Vyroby_v_zakazke
(ID_vyr integer foreign key references Vyrobov(ID_vyr),
ID_zak integer foreign key references Zakazka(ID_zak),
pocet_ks smallint,
poznanky varchar (max),
PRIMARY KEY (ID_vyr, ID_zak));

CREATE TABLE Material_v_operacii
(ID_mat integer foreign key references Material(ID_mat),
ID_ope integer foreign key references Vyrobn_a_operacia(ID_ope),
pocet_ks smallint,
poznanky varchar (max),
PRIMARY KEY (ID_mat, ID_ope));

CREATE TABLE Vyrobn_y_postup
(ID_vyr integer foreign key references Vyrobov(ID_vyr),
ID_ope integer foreign key references Vyrobn_a_operacia(ID_ope),
poznanky varchar (max),
PRIMARY KEY (ID_vyr, ID_ope));

CREATE TABLE Kvalifikacie_zamestnanca
(ID_kva integer foreign key references Kvalifikacia(ID_kva),
ID_zam integer foreign key references Zamestnanec(ID_zam),
poznanky varchar (max),
PRIMARY KEY (ID_kva, ID_zam));

CREATE TABLE Typ_zariadeni_v_operacii
(ID_typ integer foreign key references Typ_zariadenia(ID_typ),
ID_ope integer foreign key references Vyrobn_a_operacia(ID_ope),
poznanky varchar (max),
PRIMARY KEY (ID_typ, ID_ope));

CREATE TABLE Potrebne_kval_vo_vyr_operacii
(ID_kva integer foreign key references Kvalifikacia(ID_kva),
ID_ope integer foreign key references Vyrobn_a_operacia(ID_ope),
pocet smallint,
PRIMARY KEY (ID_kva, ID_ope));

CREATE TABLE Cinnost
(ID_cin int identity (1,1) PRIMARY KEY,
ID_ope integer foreign key references Vyrobn_a_operacia(ID_ope),
cas_zac datetime,
poznanky varchar (max),
cas_kon datetime);

CREATE TABLE Cinnosti_zamestnanca
(ID_cin integer foreign key references Cinnost(ID_cin),
ID_zam integer foreign key references Zamestnanec(ID_zam),
PRIMARY KEY (ID_cin, ID_zam));

CREATE TABLE Cinnosti_zariadenia
(ID_cin integer foreign key references Cinnost(ID_cin),
ID_zar integer foreign key references Zariadenie(ID_zar),

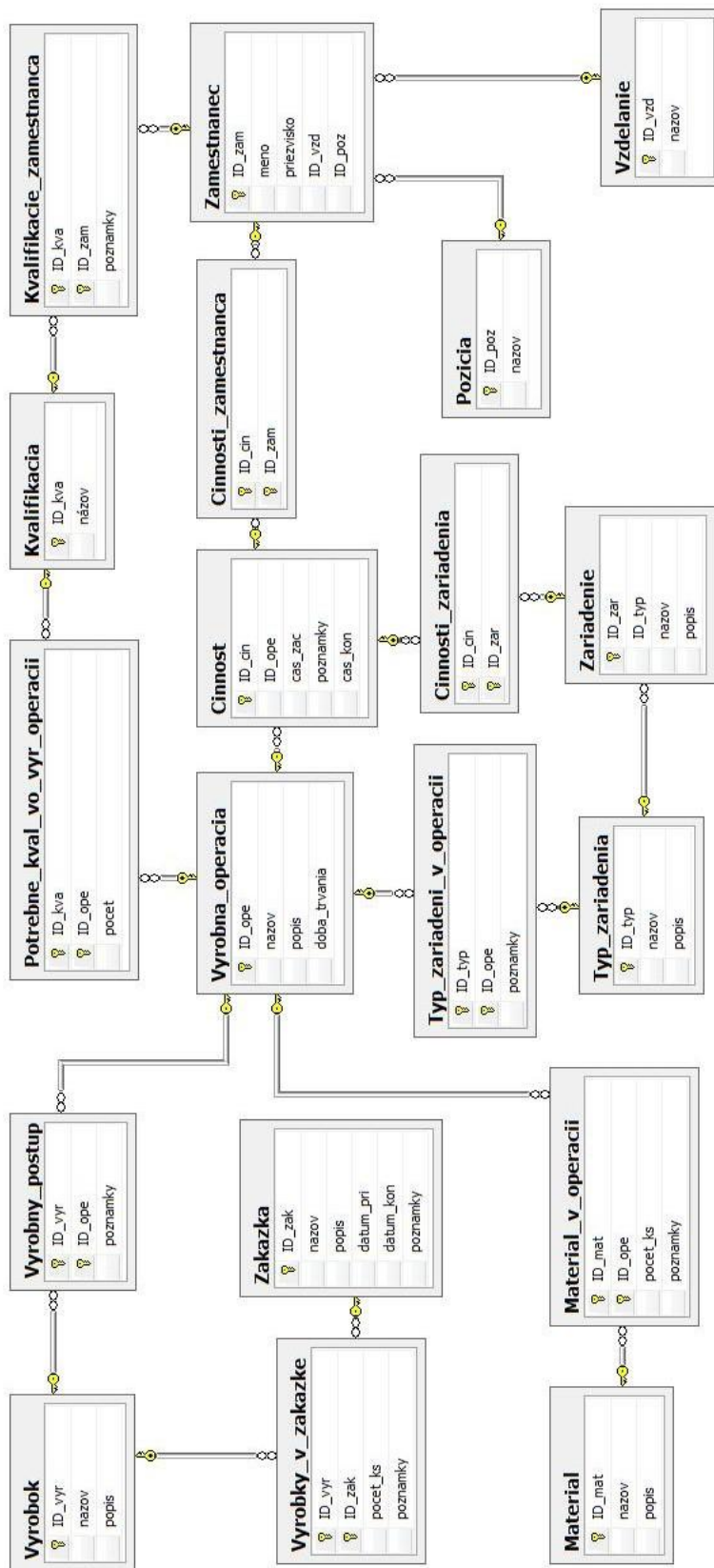
```

```
PRIMARY KEY (ID_cin, ID_zar));
```

Po samotnom vytvorení databázy bola naplnená testovacími údajmi, aby bolo možné odskúšať funkčnosť jednotlivých dotazov vytvorených pomocou užívateľského rozhrania.

Pre lepšie pochopenie jednotlivých entít bol vytvorený tzv. dátový list, ktorý vysvetľuje a špecifikuje jednotlivé entity a atribúty. Dátový list je priložený na záver práce ako príloha č.1.

### 4.3.4.3 E-R diagram



OBRÁZOK Č.2: ENTITNO-RELAČNÝ DIAGRAM

#### **4.3.5 Návrh užívateľského rozhrania**

Užívateľské rozhranie sa dá rozčleniť do niekoľkých sekcií, prvou takou sekciou, s ktorou sa užívateľ stretne pri spustení je „Prihlasovanie“. Tu sa užívateľ autentifikuje na základe prihlasovacieho mena a hesla. Po úspešnom prebehnutí prihlasovacieho procesu sa užívateľ dostane do sekcie „Hlavné menu“, ktoré je rozčlenené do niekoľkých polí. Rozloženie a popis obsahu jednotlivých polí sú uvedené nižšie. Taktiež sa tu nachádza odkaz na ďalšiu sekciu „Plánovanie“. Jedná sa o pravdepodobne najzložitejšiu časť celého systému. V tejto sekcií vidí manažér potrebné údaje o výrobných operáciách a vyťaženosť ľudí a zariadení. Dôležitosť tohto spoločného náhľadu spočíva v tom, aby boli všetky potrebné elementy na plánovanie viditeľné na jednej obrazovke bez potreby prepínania sa medzi oknami alebo sekciami, čo umožní manažérovi jednoduchým spôsobom plánovať prácu zamestnancom. Spôsob zobrazenia vyťaženosť jednotlivých zamestnancov a zariadení je riešený pomocou zoznamov s časovými údajmi.

## Sekcia „Hlavné menu“

Pole 1	
Pole 2	Pole 3
Pole 4	

OBRÁZOK Č.3: ŠTRUKTÚRA SEKcie „HLAVNÉ MENU“

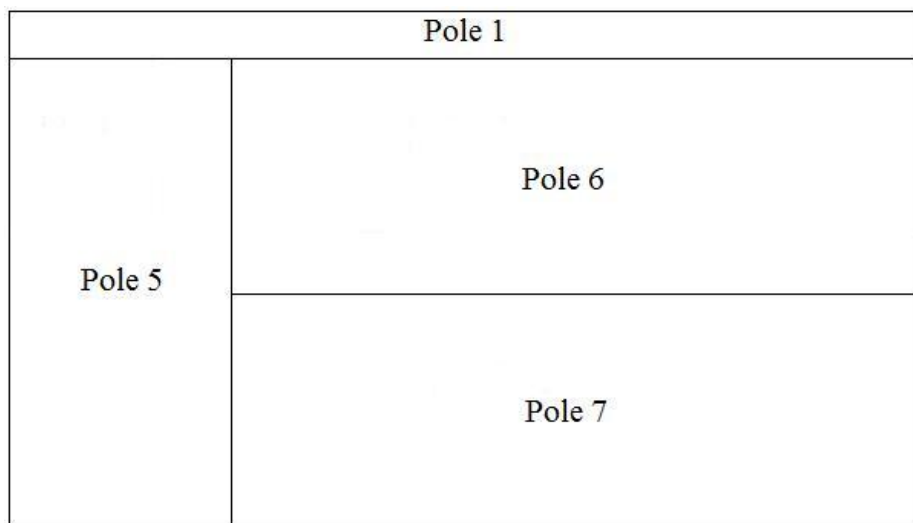
*Pole 1:* Predstavuje akési záhlavie webovej aplikácie. Obsahuje názov systému „MODUL PLÁNOVANIA A RIADENIA VÝROBNÝCH ZDROJOV“, prípadne názov sekcie, v ktorej sa užívateľ nachádza.

*Pole 2:* Obsahuje kalendár, aktuálny dátum a čas.

*Pole 3:* Obsahuje súbor záložiek (tzv. TabContainer): Zamestnanci, Výrobné operácie, Zákazky, Výrobky, Zariadenia, ktoré obsahujú zoznamy spomenutých položiek s možnosťou dostať sa k podrobnejším informáciám. Súčasťou tohto poľa sú taktiež odkazy na odhlásenie a na prechod do plánovacej sekcie.

*Pole 4:* Predstavuje pracovnú plochu, na ktorej sa vykonávajú činnosti ako vytváranie a editácia, či už úloh alebo zákaziek alebo náhľad na pridelené úlohy jednotlivých zamestnancov.

## Sekcia „Plánovanie“



OBRÁZOK Č.4: ŠTRUKTÚRA SEKcie „PLÁNOVANIE“

*Pole 5:* Je možné rozdeliť na dve podčasti. Prvá podčasť obsahuje primárne zoznam zákaziek. Po zvolení požadovanej zákazky sa zobrazí zoznam výrobkov, ktoré zákazka obsahuje a po zvolení výrobku sa zobrazia potrebné výrobné operácie, čiže výrobný postup. V prípade, že manažér chce zaplánovať niektorú so zobrazených operácií, po jej zvolení sa zobrazí obsah polí: *Pole 6* a *Pole 7*. Druhá podčasť obsahuje elementy na vytvorenie nových činností a odkazy pre návrat do predchádzajúcej sekcie a na odhlásenie.

*Pole 6:* Obsahuje zoznam kvalifikovaných zamestnancov na zvolenú výrobnú operáciu. Ďalej je tu možnosť zobrazit' zaplánované činnosti jednotlivých zamestnancov zo zoznamu, ktoré je možné editovať (preplánovať) alebo odstrániť.

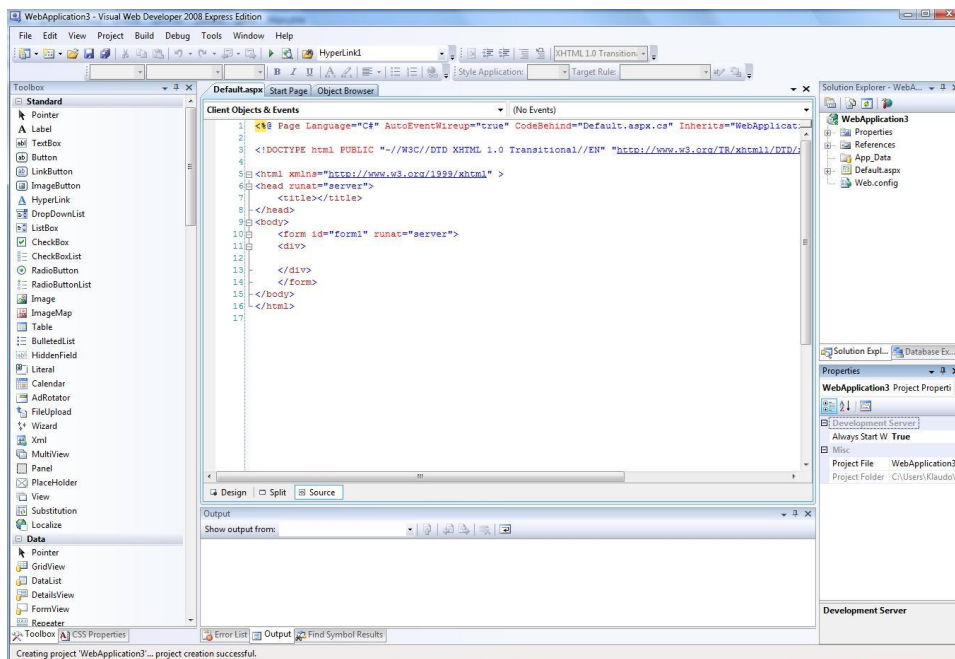
*Pole 7:* Obsah je podobný s *Pole 6*, s tým rozdielom, že sa tu pracuje so zariadeniami, ktoré vstupujú do výrobnéj operácie.



## 4.3.6 Tvorba užívateľského rozhrania

### 4.3.6.1 Stručné zoznámenie s vývojovým prostredím Visual Web Developer 2008

Visual Web Developer 2008 predstavuje jeden z nástrojov Visual Studio 2008, určený na vytváranie prevažne webových stránok a webových aplikácií. Po spustení tohto nástroja a vytvorení nového projektu sa zobrazí pracovné prostredie, ktoré je rozčlenené do niekoľkých sekcií. Úplne naľavo sa nachádza panel s elementami objektovo orientovaného programovania (*Toolbox*) s možnosťou prepuntia sa do druhej záložky, ktorá obsahuje nastavenia štýlovania a vzhľadu výsledného programu (*CSS Properties*). Pravá časť okna pozostáva z dvoch častí, vrchná časť obsahuje súborovú štruktúru a prepojenie s prípadnou databázou vytváraného produktu (*Solution Explorer* a *Database Explorer*) a spodná časť je venovaná úprave vlastností označeného elementu (*Properties*). Väčštinovú časť okna tvorí samotný programovací priestor s možnosťou pracovania v dizajnovej časti alebo časti so zdrojovým kódom.



OBRÁZOK Č.5: VISUAL WEB DEVELOPER 2008

Práca v prostredí Visual Web Developer je pomerne jednoduchá a intuitívna. Užívateľ po premyslení štruktúry budúceho programu jednoduchou technikou *Drag & Drop* presunie potrebné elementy z *Toolbox-u* do programovacieho priestoru, kde

vykoná potrebné prispôsobenie elementov danej problematike, zabezpečí zfunkčnenie a zadefinuje vzhľad a umiestnenie na stránke.

#### 4.3.6.2 Vytvorenie prepojenia s databázou a zobrazenie údajov z databázy

<u>ID_ope</u>	<u>nazov</u>	<u>doba_trvania</u>	
1	Výroba poriska	00:08:00	<a href="#">Select</a>
2	Výroba kovovej hlavy na kladivo	00:15:00	<a href="#">Select</a>
<b>3</b>	<b>Výroba kovovej hlavy na sekeru</b>	<b>00:16:00</b>	<a href="#">Select</a>
4	Navrtanie otvorov do plasového odliatku	00:02:00	<a href="#">Select</a>
5	Montáž elektronickej časti do plastového odliatku	00:03:00	<a href="#">Select</a>
6	Zmontovanie dosky plošných spojov	00:10:00	<a href="#">Select</a>

OBRÁZOK Č.6: ZOBRAZENIE ÚDAJOV TABUĽKY VÝROBNÁ OPERACIA S OZNAČENÝM TRETÍM ZÁZNAMOM

Na zobrazovanie väčšiny údajov z databázy v užívateľskom rozhraní systému sa používa element *GridView* v spolupráci s *SqlDataSource*. *GridView* predstavuje sofistikovaný objekt umožňujúci vykonávanie mnohých operácií s dátami. Dáta sú v ňom prezentované vo forme multifunkčnej tabuľky, v ktorej je možné zadefinovať mazanie, editáciu, vkladanie, selekciu položiek ale taktiež aj stránkovanie a zostupné/vzostupné zoradenie položiek podľa zvoleného atribútu. *SqlDataSource* slúži na definovanie „cesty“ k údajom a spôsob ich prezentácie. V podstate sa v ňom definuje lokalita a názov dátového zdroja a následne sa stanoví, ktoré atribúty majú byť prezentované v dátovom elemente, s ktorým je previazaný.

```

<asp:GridView ID="GridView5" runat="server" AllowPaging="True"
  AllowSorting="True" AutoGenerateColumns="False" DataKeyNames="ID_zar"
  DataSourceID="SqlDataSource5"
  onselectedindexchanged="GridView5_Select" BackColor="White"
  BorderColor="#999999" BorderStyle="None" BorderWidth="1px" CellPadding="3"
  GridLines="Vertical">
  <RowStyle BackColor="#EEEEEE" ForeColor="Black" />
  <Columns>
    <asp:BoundField DataField="ID_zar" HeaderText="ID_zar" InsertVisible="False"
      ReadOnly="True" SortExpression="ID_zar" />
    <asp:BoundField DataField="nazov" HeaderText="nazov" SortExpression="nazov" />
    <asp:CommandField ShowSelectButton="True" />
  </Columns>
  <FooterStyle BackColor="#CCCCCC" ForeColor="Black" />
  <PagerStyle BackColor="#999999" ForeColor="Black" HorizontalAlign="Center" />
  <selectedrowstyle backcolor="#008A8C" forecolor="White" Font-Bold="True" />
  <HeaderStyle BackColor="#000084" Font-Bold="True" ForeColor="White" />
  <AlternatingRowStyle BackColor="Gainsboro" />
</asp:GridView>
<asp:SqlDataSource ID="SqlDataSource5" runat="server"
  ConnectionString="<%%$ ConnectionStrings:pokus3ConnectionString %>"
  SelectCommand="SELECT [ID_zar], [nazov] FROM [Zariadenie]">
</asp:SqlDataSource>

```

OBRÁZOK Č.7: UKÁŽKA ZDROJOVÉHO KÓDU OBJEKTOV GRIDVIEW A SQLDATASOURCE

#### 4.3.6.3 Editácia údajov cez užívateľské rozhranie

ID mat	nazov	popis	
1	Drevený hranol		<a href="#">Edit</a>
2	Kovový blok		<a href="#">Edit</a>
3	Plastový odliatok		<a href="#">Edit</a>
4	Doska plošných spojov		<a href="#">Edit</a>
5	Súčiastky		<a href="#">Edit</a>
6	Šrouby		<a href="#">Edit</a>
7	Kovová rúra		<a href="#">Edit</a>
8	<input type="text" value="Matica M4"/>	<input type="text"/>	<a href="#">Update</a> <a href="#">Cancel</a>

OBRÁZOK Č.8: EDITÁCIA TABULKY MATERIÁL

Po stlačení odkazu *Edit* v požadovanom riadku sa príslušný riadok transformuje a vytvoria sa v ňom textové polia slúžiace na prepísanie aktuálnych údajov. Pre potvrdenie operácie je potrebné stlačiť *Update* a naopak pre jej ukončenie *Cancel*.

#### 4.3.6.4 Zmazanie údajov cez užívateľské rozhranie

ID_mat	nazov	popis	
1	Drevený hranol		<a href="#">Delete</a>
2	Kovový blok		<a href="#">Delete</a>
3	Plastový odliatok		<a href="#">Delete</a>
4	Doska plošných spojov		<a href="#">Delete</a>
5	Súčiastky		<a href="#">Delete</a>
6	Šrouby		<a href="#">Delete</a>
7	Kovová rúra		<a href="#">Delete</a>
8	Matica M4		<a href="#">Delete</a>

OBRÁZOK Č.9: ODSTRÁNENIE POLOŽKY Z TABULKY MATERIÁL

V prípade, že užívateľ má záujem zmazať nejakú položku stlačí *Delete* v príslušnom riadku, čo zabezpečí nielen odstránenie z elementu GridView ale aj odstránenie z databázy.

#### 4.3.6.5 Vkladanie údajov cez užívateľské rozhranie

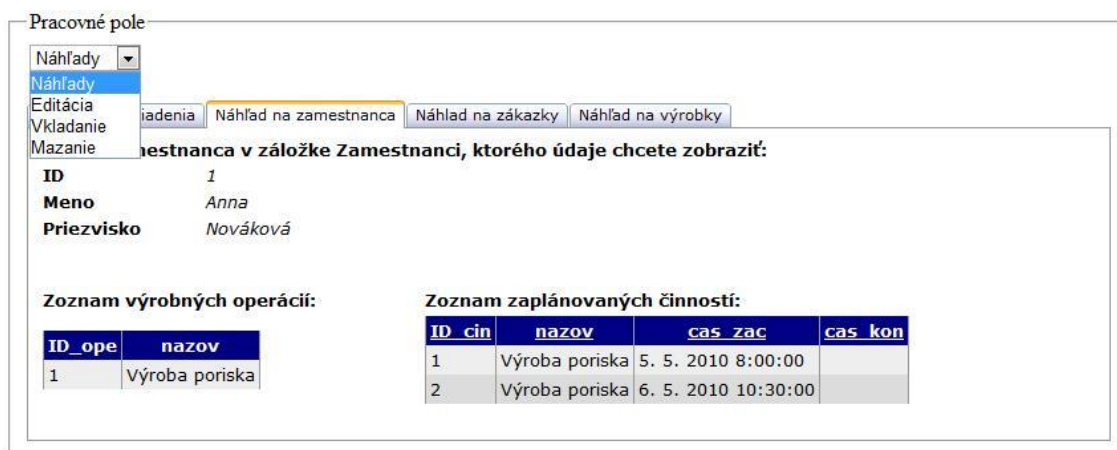
ID_mat	nazov	popis
1	Drevený hranol	
2	Kovový blok	
3	Plastový odliatok	
4	Doska plošných spojov	
5	Súčiastky	
6	Šrouby	
7	Kovová rúra	
8	Matica M4	
<input type="button" value="Vložiť"/> <input type="text"/> <input type="text"/>		

OBRÁZOK Č.10: VKLADANIE POLOŽKY Z TABULKY MATERIÁL

Súbor prvkov na vkladanie údajov do databázy je umiestnený v zápäťí (tzv. *Footer Template*) elementu GridView. Tento súbor je tvorený prevažne textovými poliami a potvrdzujúcim tlačidlom *Vložiť*. Užívateľ vypíše potrebné údaje do elementov *TextBox*, po potvrdení program automaticky vygeneruje unikátne ID a položku vloží do databázy.

#### 4.3.6.6 Dynamické zobrazovanie údajov

Elementy *MultiView* a *TabContainer* umožňujú flexibilne meniť obsah zobrazovanej plochy, podľa potrieb užívateľa. Dôvody využitia týchto objektov sú jednoznačné: obmedzená zobrazovacia plocha prehliadača, potreba zobrazit' čo možno najväčšie množstvo údajov najkomfortnejším možným spôsobom v čo najmenšom množstve webových formulárov.

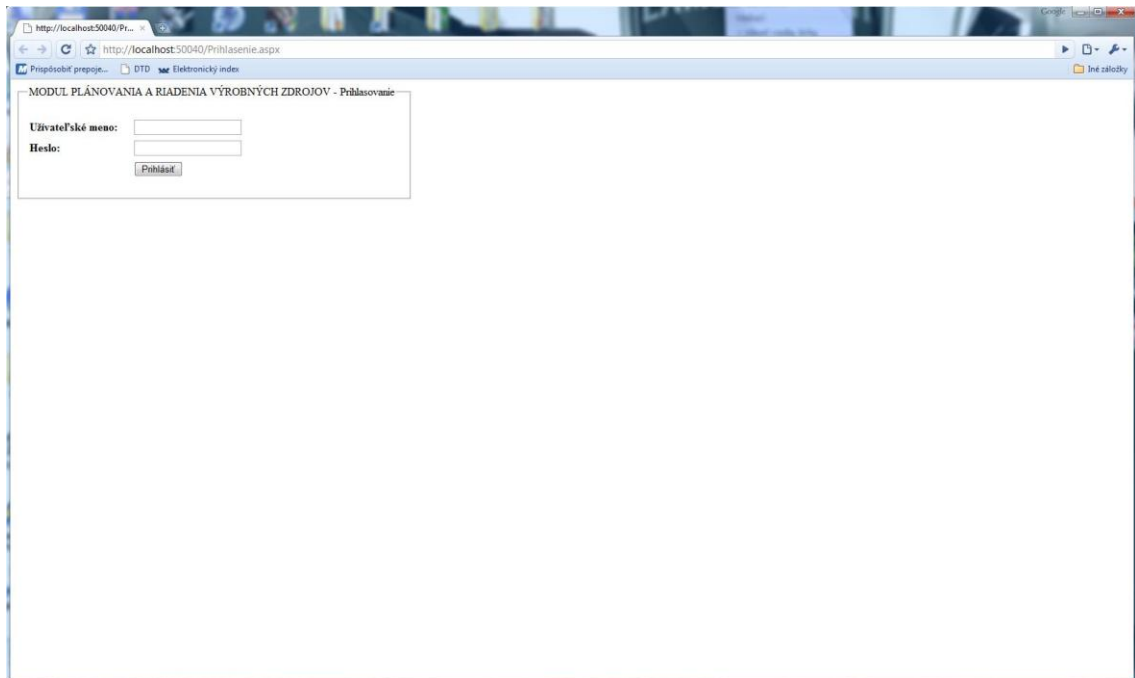


OBRAZOK Č.11: MULTIVIEW & TABCONTAINER

Spôsob fungovania *MultiView* spočíva v spolupráci rozbaľovacieho menu (*DropDownList*) a určitého počtu pohľadov (*Views*), ktorých obsah môže byť rôzny. Zobrazenie požadovaného pohľadu závisí od užívateľom zvolenej položky z menu. Na obrázku č.11 je zobrazený pohľad s názvom „Náhľady“, ktorého obsahom je tzv. *TabContainer*, ktorý umožňuje zobrazenie veľkého množstva údajov na relatívne malej ploche za pomoci záložiek. Prepínanie medzi záložkami užívateľ vykonáva stlačením na hlavičku požadovanej záložky.

## 4.4 Finálny vzhľad systému

### 4.4.1 Prihlasovanie



OBRÁZOK Č.12: PRIHLASOVANIE

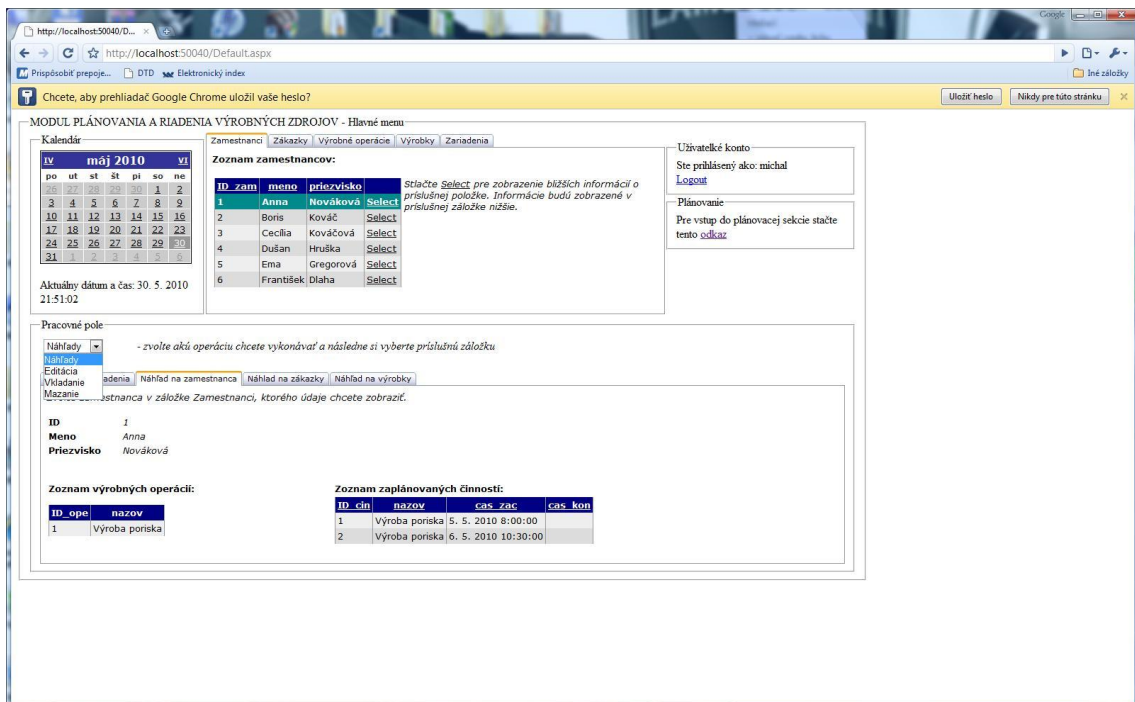
Predstavuje prístup do webovej aplikácie, ktorý je v súčasnom systéme nastavený na jediného užívateľa (manažéra). Po úspešnom prihlásení sa aktivuje sekcia Hlavné menu.

Užívateľské meno: *michal*

Heslo: *bakalarka*

Pozn.: Na prezentáciu webovej aplikácie bol použitý internetový prehliadač Google Chrome.

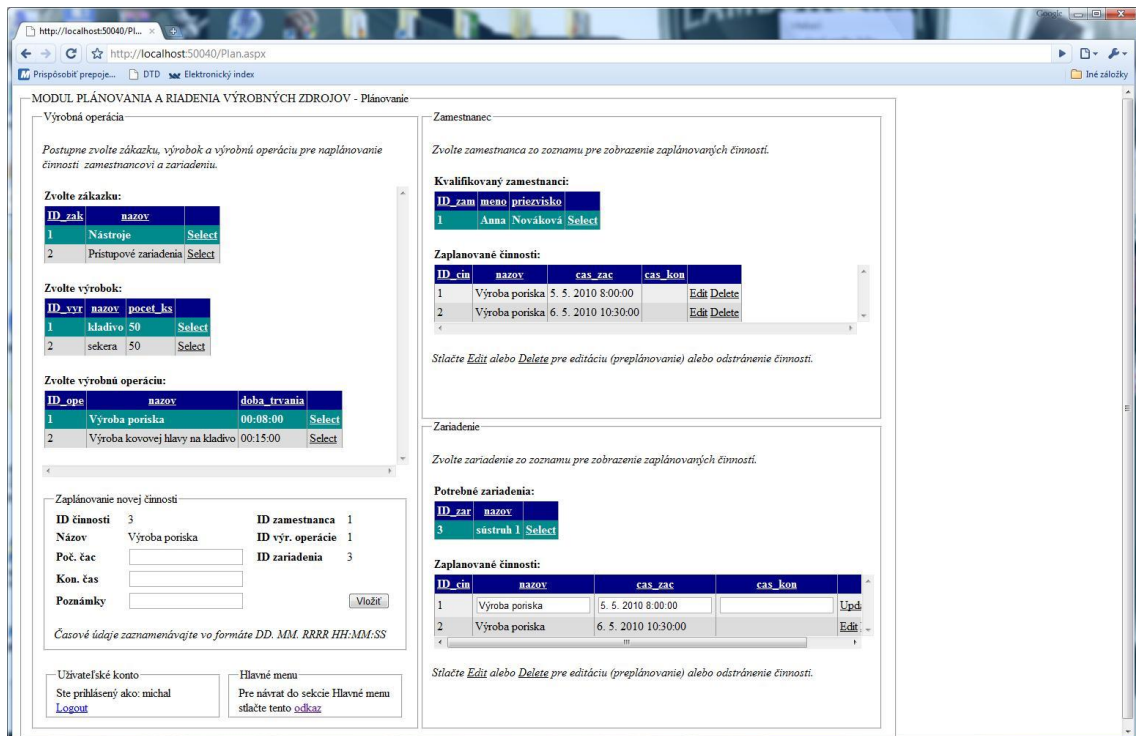
## 4.4.2 Hlavné menu



OBRÁZOK Č.13: HLAVNÉ MENU

Sekcia „Hlavné menu“ umožňuje manažérovi vykonávanie základných operácií (vkladanie, mazanie, editácia a náhľad) nad údajmi v databáze. Voľbu operácie umožňuje rozbaľovacie menu v ľavom hornom rohu *Pracovného poľa* a následne voľbou záložky má možnosť určiť nad akými údajmi chce danú operáciu uskutočniť.

## 4.4.3 Plánovanie



OBRÁZOK Č.14: PLÁNOVANIE

Sekcia „Plánovanie“ umožňuje manažérovi zapláňovať, preplánovať alebo odstrániť činnosť zamestnanca alebo zariadenia. Manažér postupne volí zákazku, výrobok a výrobnú operáciu. Následne sa mu zobrazí zoznam kvalifikovaných zamestnancov a zoznam zariadení, ktorým môže byť priradená zvolená výrobná operácia. Po výbere zamestnanca a zariadenia má manažér možnosť upraviť alebo odstrániť už pridelené činnosti. Na základe predchádzajúcich krokov sa postupne vyplní formulár nachádzajúci sa v poli s názvom *Zaplánovanie novej činnosti*. Tento formulár slúži na vytvorenie novej činnosti a pre jeho uloženie do databázy je nutné vyplnený formulár potvrdiť tlačidlom *Vložiť*.



## 4.5 Prínosy návrhu riešenia

Tento systém bol navrhnutý presne na mieru pre firmu CONTAL OK, čo znamená, že z funkčného hľadiska obsahuje len to, čo je potrebné pre danú firmu v oblasti riadenia výroby. Kúpa komerčného sofistikovaného systému by bola pre firmu ekonomicky neefektívna, nakoľko zaplatí aj za funkcie, ktoré ostanú nevyužité. Implementáciou takéhoto typu systému sa docieli hneď niekoľko prínosov:

1. Systém „šitý na mieru“ – predstavuje šikovného asistenta pri plánovaní, ktorý obsahuje skutočne len to, čo je nevyhnutné a potrebné. Docieli sa tým jednoduchosť a intuitívnosť pri používaní systému.
2. Efektivita plánovania a riadenia – zefektívnenie činností manažéra súvisiacich s procesom plánovania a riadenia výrobných zdrojov.
3. Spätná väzba z výrobného procesu o:
  - efektívnosti využívania pracovného fondu pracovníkmi
  - efektívnosti využívania strojných zariadení
  - správnosti časových noriem stanovených pre vykonávanie jednotlivých výrobných operácií
4. Webová aplikácia:
  - umožňuje výkon činnosti manažéra na ľubovoľnom mieste s prístupom na internet
  - jednoduchá údržba systému
5. Investičná úspora – firma nebude nútená investovať do komerčného riešenia, ktorého cena sa v súčasnosti pohybuje v stovkách až tisíckach eur. Tieto finančné prostriedky môžu byť investované efektívnejším spôsobom.

6. Zníženie mzdových nákladov – zvýšením efektivity vzniká aj časová úspora v činnostiach manažéra a v efektívnom využívaní pracovného fondu zamestnancami.

Spomínané prínosy sa viažu hlavne na finálny produkt, ale keďže užívateľské rozhranie „Manažér“ predstavuje základ celého systému, tak je možné tieto prínosy aplikovať aj len na túto časť.

## ZÁVER

Bakalárska práca bola venovaná vytvoreniu podporného modulu pre proces plánovania a riadenia výrobných zdrojov v úseku výroby spoločnosti CONTAL OK, spol. s r.o. na báze webovo orientovaného databázového systému. Zo štruktúrneho hľadiska je práca rozdelená do troch kapitol.

Úvodná kapitola je venovaná analýze súčasného stavu danej problematiky a prezentácii firmy. Druhú kapitolu tvorí stručný teoretický základ použitých programových nástrojov, technológií a vývojových prostredí. Záverečná kapitola predstavuje samotný návrh, tvorbu systému a prínosy, ktoré implementácia systému prinesie.

V praxi tento systém reprezentuje webová aplikácia vytvorená pomocou vývojových prostredí Microsoft SQL Server Management Studio 2008 a Visual Studio 2008, v ktorých bola vytvorená databáza na uchovávanie dát a užívateľské rozhranie. Ako som už v úvode bakalárskej práce spomínal z dôvodu rozsahu práce bolo na podnet vedúceho riešené iba užívateľské rozhranie „Manažér“. Vytvorením tohto rozhrania, ktoré symbolizuje základnú kostru celého systému, boli dosiahnuté požadované ciele.

Nakoľko sa jedná len o základ systému v budúcnosti by mal byť dopracovaný do finálnej podoby. Znamenalo by to rozšírenie databázy o ďalšie položky, doplnenie rozhrania „Manažér“ o ďalšie funkcie a vytvorenie ďalších dvoch užívateľských rozhraní „Pracovník“ a „Dohľad“. Toto rozšírenie môže byť náplňou diplomovej práce a v prípade, že zadávateľ bude s výsledkom spokojný, dôjde aj k jeho následnej implementácii.

## ZOZNAM POUŽITÝCH ZDROJOV

### Knižné zdroje:

- [1] DOSEDĚL, T. *Počítačová bezpečnost a ochrana dat*. 1. vydání. Brno: Computer Press, 2004. 190 s. ISBN 80-251-0106-1
- [2] HOTEK, M. *Microsoft SQL Server 2008: krok za krokem*. 1. vydanie. Brno: Computer press, 2009. 488 s. ISBN 978-80-251-2466-6.
- [3] HUDDLESTON, J. a AGARWAL V.V. *Databáze v C# 2008*. Brno: Computer press, 2009. 424 s. ISBN 978-80-251-2309-6.
- [4] KOTLER, P. *Marketing management*. 1. vydání. Praha: Grada, 2007. 788 s. ISBN 978-80-247-1359-5
- [5] PROSISE, J. *Programování v Microsoft .NET*. Brno: Computer press, 2003. 736 s. ISBN 80-7226-879-1.
- [6] PUŽMANOVÁ, R. *Moderní komunikační sítě od A do Z*. 2. aktualizované vydání. Brno: Computer Press, 2006. 430 s. ISBN 80-251-1278-0
- [7] ŘEPA, V. *Podnikové procesy*. 2.rozšířené vydání. Praha: Grada, 2007. 281 s. ISBN 978-80-247-2252-8
- [8] SHARP, J. *Microsoft Visual C# 2008*. Brno: Computer press, 2008. 592 s. ISBN 978-80-251-2027-9.
- [9] VOŘÍŠEK, J. *Principy a modely řízení podnikové informatiky*. 1. vydání. Praha: Oeconomica, 2008. 446 s. ISBN 978-80-245-1440-6

### Elektronické zdroje:

- [10] *Ajax Control Toolkit Tutorials* [Online]. 2010 [cit. 2010-05-13].  
Dostupné na WWW: <[http://www.asp.net/ajaxlibrary/act\\_tutorials.ashx](http://www.asp.net/ajaxlibrary/act_tutorials.ashx)>.
- [11] *ASP.NET AJAX Control Toolkit* [Online]. 2010 [cit. 2010-05-13].  
Dostupné na WWW: <<http://www.asp.net/ajax/ajaxcontroltoolkit/Samples/Default.aspx>>.
- [12] *ASP.NET Quickstart Tutorial* [Online]. 2010 [cit. 2010-05-13].  
Dostupné na WWW: <<http://quickstart.aspnet.sk/QUICKSTARTV20/aspnet/default.aspx>>.

- [13] *Obchodný register* [Online]. 2009 [cit. 2009-12-07]. Dostupné na WWW: <<http://www.orsr.sk/vypis.asp?ID=1476&SID=5&P=1>>.
- [14] *Visual C# – stredisko vývojárov* [Online]. 2009 [cit. 2009-11-29]. Dostupné na WWW: <[http://msdn.microsoft.com/sk-sk/vcsharp/aa336766\(en-us\).aspx](http://msdn.microsoft.com/sk-sk/vcsharp/aa336766(en-us).aspx)>.
- [15] *Wikipédia – ADO.NET* [Online]. 2010 [cit. 2010-05-14]. Dostupné na WWW: <<http://sk.wikipedia.org/wiki/ADO.NET>>.
- [16] *Wikipédia – HTML* [Online]. 2010 [cit. 2010-05-24]. Dostupné na WWW: <<http://sk.wikipedia.org/wiki/HTML>>.

**Firemná literatúra:**

- [17] *O spoločnosti CONTAL OK*. Žilina: CONTAL OK, 2009. 1 s.

## ZOZNAM OBRÁZKOV

OBRÁZOK Č.1: DEKOMPOZÍCIA VÁZBY M:N.....	- 31 -
OBRÁZOK Č.2: ENTITNO-RELAČNÝ DIAGRAM .....	- 36 -
OBRÁZOK Č.3: ŠTRUKTÚRA SEKcie „HLAVNÉ MENU“ .....	- 38 -
OBRÁZOK Č.4: ŠTRUKTÚRA SEKcie „PLÁNOVANIE“ .....	- 39 -
OBRÁZOK Č.5: VISUAL WEB DEVELOPER 2008 .....	- 40 -
OBRÁZOK Č.6: ZOBRAZENIE ÚDAJOV TABULKY VÝROBNÁ OPERACIA S OZNAČENÝM TRETÍM ZÁZNAMOM ....	- 41 -
.....	
OBRÁZOK Č.7: UKÁŽKA ZDROJOVÉHO KÓDU OBJEKTOV GRIDVIEW A SQLDATASOURCE .....	- 42 -
OBRÁZOK Č.8: EDITÁCIA TABULKY MATERIÁL .....	- 42 -
OBRÁZOK Č.9: ODSTRÁNENIE POLOŽKY Z TABULKY MATERIÁL.....	- 43 -
OBRÁZOK Č.10: VKLADANIE POLOŽKY Z TABULKY MATERIÁL .....	- 43 -
OBRÁZOK Č.11: MULTIVIEW & TABCONTAINER.....	- 44 -
OBRÁZOK Č.12: PRIHLASOVANIE.....	- 45 -
OBRÁZOK Č.13: HLAVNÉ MENU.....	- 46 -
OBRÁZOK Č.14: PLÁNOVANIE .....	- 47 -

## ZOZNAM TABULIEK

TABUĽKA Č.1: ZAMESTNANEC.....	- 29 -
TABUĽKA Č.2: ZÁKAZKA .....	- 29 -
TABUĽKA Č.3: ZARIADENIE.....	- 29 -
TABUĽKA Č.4: MATERIÁL .....	- 29 -
TABUĽKA Č.5: VÝROBNÁ OPERÁCIA .....	- 30 -
TABUĽKA Č.6: POZÍCIA .....	- 30 -
TABUĽKA Č.7: VZDELANIE.....	- 30 -
TABUĽKA Č.8: KVALIFIKÁCIA .....	- 30 -
TABUĽKA Č.9: ČINNOSŤ .....	- 32 -
TABUĽKA Č.10: ČINNOSŤ ZARIADENIA.....	- 32 -
TABUĽKA Č.11: ČINNOSŤ ZAMESTNANCA .....	- 32 -
TABUĽKA Č.12: TYP ZARIADENIA .....	- 32 -

# PRÍLOHY

## Príloha č.1: Dátový list

Dátový list					
Tabuľka (Entita)	Atribút	Dátový typ	Integritné obmedzenia*	Hodnota	Popis
<b>Vzdelanie</b>					
	<b>ID_vzd</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
	<b>nazov</b>	<i>varchar (50)</i>			názvy jednotlivých typov vzdelania
<b>Pozicia</b>					
	<b>ID_poz</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
	<b>nazov</b>	<i>varchar (50)</i>			názvy jednotlivých typov pozícií
<b>Kvalifikacia</b>					
	<b>ID_kva</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
	<b>nazov</b>	<i>varchar (50)</i>			názvy jednotlivých typov kvalifikácií
<b>Vyrobok</b>					
	<b>ID_vyr</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
	<b>nazov</b>	<i>varchar (100)</i>			názov výrobku
	<b>popis</b>	<i>varchar (max)</i>		NULL	bližšie špecifikácie výrobku
<b>Typ_zariadenia</b>					
	<b>ID_typ</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
	<b>nazov</b>	<i>varchar (100)</i>			názov typu zariadenia
	<b>popis</b>	<i>varchar (max)</i>		NULL	bližšie špecifikácie typu zariadenia
<b>Zariadenie</b>					
	<b>ID_zar</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
	<b>ID_typ</b>	<i>integer</i>	FK		služi na prepojenie s tabuľkou Typ_zariadenia
	<b>nazov</b>	<i>varchar (100)</i>			názov zariadenia
	<b>popis</b>	<i>varchar (max)</i>		NULL	bližšie špecifikácie zariadenia



Material					
	<b>ID_mat</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
	<b>nazov</b>	<i>varchar (100)</i>			názov materiálu
	<b>popis</b>	<i>varchar (max)</i>		NULL	bližšie špecifikácie materiálu
Vyrobna_operacia					
	<b>ID_ope</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
	<b>nazov</b>	<i>varchar (100)</i>			názov výrobnnej operácie
	<b>popis</b>	<i>varchar (max)</i>		NULL	bližšie špecifikácie výrobnnej operácie
	<b>doba_trvania</b>	<i>time</i>			doba trvania výrobnnej operácie
Zakazka					
	<b>ID_zak</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
	<b>nazov</b>	<i>varchar (100)</i>			názov zákazky
	<b>popis</b>	<i>varchar (max)</i>			bližšie špecifikácie zákazky
	<b>datum_pri</b>	<i>datetime</i>			dátum prijatia zákazky
	<b>datum_kon</b>	<i>datetime</i>			dátum ukončenia zákazky
	<b>poznamky</b>	<i>varchar (max)</i>		NULL	priestor pre komentáre a dotazy
Zamestnanec					
	<b>ID_zam</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
	<b>meno</b>	<i>varchar (20)</i>			meno zamestnanca
	<b>priezvisko</b>	<i>varchar (25)</i>			priezvisko zamestnanca
	<b>ID_vzd</b>	<i>integer</i>	FK		slúži na prepojenie s tabuľkou Vzdelanie
	<b>ID_poz</b>	<i>integer</i>	FK		slúži na prepojenie s tabuľkou Pozícia
Vyroby_v_zakazke					
	<b>ID_vyr</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Vyrobov
	<b>ID_zak</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Zakazka
	<b>pocet_ks</b>	<i>smallint</i>			počet kusov výrobkov
	<b>poznamky</b>	<i>varchar (max)</i>		NULL	priestor pre komentáre a dotazy

Material_v_operacii					
	<b>ID_mat</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Material
	<b>ID_ope</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Vyrobná_operácia
	<b>pocet_ks</b>	<i>smallint</i>			počet kusov materiálu v danej výrobnej operácii
	<b>poznamky</b>	<i>varchar (max)</i>		NULL	priestor pre komentáre a dotazy
Vyrobný_postup					
	<b>ID_vyr</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Vyrobný_postup
	<b>ID_ope</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Vyrobná_operácia
	<b>poznamky</b>	<i>varchar (max)</i>		NULL	priestor pre komentáre a dotazy
Kvalifikácie_zamestnanca					
	<b>ID_kva</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Kvalifikácia
	<b>ID_zam</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Zamestnanec
	<b>poznamky</b>	<i>varchar (max)</i>		NULL	priestor pre komentáre a dotazy
Typ_zariadeni_v_operacii					
	<b>ID_typ</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Typ_zariadenia
	<b>ID_ope</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Vyrobná_operácia
	<b>poznamky</b>	<i>varchar (max)</i>		NULL	priestor pre komentáre a dotazy
Potrebne_kval_vo_vyr_operacii					
	<b>ID_kva</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Kvalifikácia
	<b>ID_ope</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou

				Vyrobná_operácia
<b>pocet</b>	<i>smallint</i>			počet ľudí potrebných vo výrobných operáciách s danou kvalifikáciou
<b>Cinnost</b>				
<b>ID_cin</b>	<i>int identity (1,1)</i>	PK		identifikátor tabuľky
<b>ID_ope</b>	<i>integer</i>	FK		služi na prepojenie s tabuľkou Vyrobná_operácia
<b>cas_zac</b>	<i>datetime</i>			počiatočný čas činnosti
<b>poznamky</b>	<i>varchar (max)</i>		NULL	priestor pre komentáre a dotazy
<b>cas_kon</b>	<i>datetime</i>			koncový čas činnosti
<b>Cinnosti_zariadenia</b>				
<b>ID_cin</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Cinnost
<b>ID_zar</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Zariadenie
<b>Cinnosti_zamestnanca</b>				
<b>ID_cin</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Cinnost
<b>ID_zam</b>	<i>integer</i>	PK, FK		identifikátor tabuľky, slúži na prepojenie s tabuľkou Zamestnanec

\* PK (Primary Key) – primárny kľúč, FK (Foreign Key) – cudzí kľúč