

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

DATABÁZE SKILLMATRIX

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Roman Vorel

BRNO 2007



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

DATABÁZE SKILLMATRIX

SKILLMATRIX DATABASE

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Roman Vorel

VEDOUČÍ PRÁCE

SUPERVISOR

Doc. Ing. Jaroslav Zendulka, CSc.

BRNO 2007

Abstrakt

Tato diplomová práce pojednává o vytvoření databázového informačního systému k doškolení zaměstnanců a výpočtu bonusů a jeho napojení na stávající personální systémy v prostředí společnosti Honeywell. V projektu je zohledněno řešení na úrovni mocného databázového serveru a maximální využití jeho schopností pro vývoj aplikace. Systém disponuje přístupovým oprávněním zahrnujícím několik uživatelských rolí a roli administrátora. Návrh systému je proveden v modelovacím jazyce UML. Implementace daného projektu je provedena na technologiích společnosti Microsoft. Systém je obecně implementován ve 3 vrstvách: uživatelské rozhraní, databázová komponenta, databáze. Uživatelské rozhraní je realizováno ve skriptovacím jazyce ASP.NET 2.0, za využití moderních technologií ASP.NET AJAX 1.0. Databázová komponenta je implementována za pomoci jazyka Visual .NET C# a knihoven ADO.NET. Databáze je pak realizována v prostředí SQL Serveru 2000 s využitím procedurálního jazyka Transact-SQL. Nedílnou součástí projektu je i testování na zkušebních datech a nasazení do ostrého provozu na intranetu společnosti Honeywell.

Klíčová slova

SQL Server, ADO.NET, ASP.NET, Visual C#, ASP.NET AJAX, ERP, UML.

Abstract

This work deals with creation of database information system for employee training and bonus calculation with direct connection on existing personal systems in Honeywell production company. The project makes provision for solution on dynamic database server and maximal usage of its potentiality for application development. The system dispose of access privileges containing several user roles and role of administrator. The system concept is realized in UML modeling language. The implementation is realized on Microsoft technologies. The system is generally implemented in 3 tiers: user interface, database component, data base model. The user interface is implemented in ASP.NET 2.0 with utilization of ASP.NET AJAX 1.0 technology. The database component is implemented using object oriented language Visual .NET C# and ADO.NET classes. Data base is relized on SQL Server 2000 in Transact-SQL language. The integral part of the project is also testing on sample data and application in live business on Honeywells company intranet.

Keywords

SQL Server, ADO.NET, ASP.NET, Visual C#, ASP.NET AJAX, ERP, UML.

Citace

Roman Vorel: Databáze SkillMatrix, diplomová práce, Brno, FIT VUT v Brně, 2007

Databáze SkillMatrix

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Doc. Ing. Jaroslava Zendulky, CSc.

Další informace mi poskytli pan Ing. Rostislav Pacas

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Jméno Příjmení
Datum

Poděkování

Chtěl bych poděkovat Doc. Ing. Jaroslavu Zendulkovi, CSc. za odborné vedení a konzultace při zpracování diplomové práce.

Dále bych rád poděkoval IT Manažerovi společnosti Honeywell panu Ing. Rostislavu Pacasovi a personálnímu oddělení společnosti za jejich pomoc a spolupráci na projektu o kterém pojednává tato diplomová práce.

© Roman Vorel, 2007.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

Obsah	5
1 Úvod.....	6
2 Požadavky na aplikaci.....	8
2.1 Prostředí společnosti	8
2.1.1 IT infrastruktura společnosti	8
2.1.2 Personální systémy.....	8
2.2 Neformální popis.....	9
2.3 Specifikace požadavků.....	10
2.3.1 Případy použití.....	11
2.3.2 Firemní procesy	17
2.3.3 Výpočet bonusů	19
2.3.4 Datový model.....	22
3 Návrh a implementace	26
3.1 Architektura systému.....	26
3.1.1 Použité technologie.....	27
3.2 Databáze.....	36
3.2.1 Diagram schématu databáze	36
3.2.2 Uložené procedury.....	40
3.2.3 Triggery	44
3.2.4 Pohledy	45
3.2.5 Komunikace s dílčími systémy	46
3.3 Aplikace	47
3.3.1 Adresářová struktura.....	47
3.3.2 Databázová komponenta.....	48
3.3.3 Přístupová oprávnění	52
3.3.4 Uživatelské rozhraní	54
4 Ukázka použití	56
4.1 Výpočet bonusů.....	56
4.2 Zadání skills	59
4.2.1 Zadání nového skill.....	59
4.2.2 Změna současného skill	60
5 Závěr	62
6 Literatura.....	63

1 Úvod

Cílem diplomové práce byl požadavek společnosti Honeywell na vytvoření databázové aplikace nazvané SkillMatrix pro doškolování zaměstnanců a výpočet bonusů.

Honeywell je nadnárodní společnost se sídlem v USA s ročním obratem okolo 25 miliard USD, výrobce a dodavatel různorodých technologií a služeb po celém světě. Zaměstnává na celém světě přibližně 125 000 lidí v 95 zemích. Akcie Honeywellu se obchodují na New Yorkské burze, stejně jako v Londýně, Chicagu či pacifických trzích. Jsou to jedny ze třiceti akcií, podle kterých je sestavován Down Jonesův průmyslový index.

Společnost Honeywell Brno o.z., která je zadavatelem projektu, vznikla v roce 2003 a dnes je jedním z významných výrobních závodů této společnosti v Evropě. V současnosti pracuje ve výrobním závodě 500 zaměstnanců. Ti se podílí na výrobě ventilů, termostatických hlavíc pro topná tělesa, ventilů do plynových kotlů a také osazování desek plošných spojů.

Systém SkillMatrix se měl stát jedním ze tří klíčových systémů, které takto velká, na výrobu orientovaná firma, bezesporu potřebuje.

Požadavek vedení společnosti Honeywell na takovýto systém plynul zejména z nejednotného a nekoncepčního stávajícího řešení v prostředí firmy a současného vydání nových procedur specifikujících nové firemní procesy a procedury pro doškolování a výpočet bonusů zaměstnanců.

Hlavními požadavky bylo napojit tento systém na stávající personální systémy. Jelikož tyto systémy obsahují podstatné informace o zaměstnancích, které se měly stát součástí systému SkillMatrix, je tento požadavek zcela logický. Toto propojení mělo přinést hlavní cíle, a to jednak přenos těchto klíčových dat do systému SkillMatrix, zajištění jejich synchronizace z důvodu aktuálnosti dat, minimalizace uživatelských vstupů a tedy časové náročnosti na uživatele, respektive správce systému.

Dalšími očekávanými a logickými přínosy řešení systému SkillMatrix bylo dodržení firemních procesů, snížení časové vytíženosti manažerů, personalistů a ostatních osob při realizaci těchto procesů, zabezpečení dat a centralizovaný víceúrovňový přístup.

První kapitola obsahuje stručné seznámení s problémem a celkový nástin toho, co se celá práce snaží realizovat.

Druhá kapitola přináší rozbor požadavků na systém SkillMatrix pro doškolování zaměstnanců a výpočet bonusů a jeho napojení na stávající personální systémy. Kromě specifikace samotných požadavků se zabývá analýzou případů použití, firemními procesy a v závěru analýzou datového modelu.

Třetí kapitola obsahuje návrh a vysvětlení implementace systému. Samotná implementace pak zahrnuje rozbor jednotlivých modulů a implementovaných komponent.

Čtvrtá kapitola prezentuje ukázky případů použití jednotlivých klíčových firemních procesů v prostředí aplikace.

Pátou kapitolou je závěr, který přináší zhodnocení práce a dosažených výsledků.

Šestou a poslední kapitolou je literatura, kde se nachází citace na publikace, z kterých tato práce čerpala.

2 Požadavky na aplikaci

Požadavky na aplikaci byly ustanoveny na základě konzultací s personálním oddělením a vedením společnosti. Při analýze byly též vytvořeny modely klíčových firemních procesů pomocí diagramů aktivity a případů použití v návaznosti na uživatelské role. Nedílnou součástí je i analýza personálních systémů, na které má být aplikace napojena.

2.1 Prostředí společnosti

2.1.1 IT infrastruktura společnosti

IT infrastruktura Honeywell Brno o.z. představuje serverová řešení postavená na technologiích Microsoft v podobě aplikačního a databázového serveru, file serveru a několika dalších serverů sloužících ostatním Honeywell závodům v Brně. Všechny tyto servery a celá počítačová síť LAN (local-area-network) jsou spravovány lokálním IT oddělením. Servery v podobě doménového řadiče a serverů enterprise aplikací, stejně tak jako WAN (wide-area network), jsou outsourcovány a nejsou součástí lokální IT infrastruktury, stejně tak jako nejsou ve správě lokálního IT.

Ve vztahu k návrhu aplikace byl klíčový především databázový a aplikační server:

- Databázový a aplikační server tvoří jedna serverová stanice na operačním systému Windows Server 2003
- Databázový server je realizován jako Microsoft SQL Server 2000, na kterém jsou umístěny všechny databázové aplikace
- Webový server je realizován jako Internet Information Services 6.0

2.1.2 Personální systémy

Jedná se o systémy, s nimiž bude aplikace SkillMatrix komunikovat.

Personální systémy představují klíčové informace o zaměstnancích a organizační struktuře každé společnosti.

Ve společnosti Honeywell Brno o.z. se jedná o Docházkový a Personální systém. Oba tyto systémy se měli stát zdroji dat pro aplikaci SkillMatrix.

Architektura a popis systémů:

1. Docházkový a přístupový systém Evidence 2000

- Řešení pro evidenci docházky a zabezpečený pohyb osob. Osoby jsou umístěny v organizačním schématu, které definuje firemní hierarchii. Každý zaměstnanec má svoji identifikační kartu, se kterou si na příslušném terminálu označuje příchody a

odchody, popřípadě volí důvody přerušování práce nebo důvod nepřítomnosti na pracovišti. Vyhodnocené údaje slouží jako podklady pro zpracování mezd v systému Target 2100, tyto data jsou přenášena na základě provedení exportu z aplikace do textového souboru.

- Dodavatel: Duha systém spol. s r.o.
- Struktura systému: Architektura klient-server pro databázové prostředí MS SQL 2000
- Struktura databáze: Standardní relační databázový model
- Požadovaná data pro SkillMatrix: dvě horní úrovně organizačního schématu (třetí nese SkillMatrix), přítomnost zaměstnanců na pracovišti, textový soubor s údaji o docházce, jako podklad pro zpracování a výpočet bonusů

2. Personální a mzdový systém Target 2100

- Řešení pro personální evidenci, výpočet mezd a vzdělávání. Obsahuje detailní informace o osobách a jejich pracovním zařazení, mzdách, absolvovaných školeních, dosaženém vzdělání atd. Výpočet mezd probíhá na základě importu dat s údaji o docházce z textového souboru systému Evidence 2000.
- Dodavatel: M-PRO spol. s r.o.
- Struktura systému: Architektura klient-server pro databázové prostředí MS SQL 2000
- Struktura databáze: Silně denormalizovaný databázový model
- Požadovaná data pro SkillMatrix: výčet osob pro SkillMatrix dle pracovního zařazení a aktuálnosti pracovního poměru, data o mzdě zaměstnanců pro výpočet bonusů

2.2 Neformální popis

Systém SkillMatrix je napojen na dva stávající personální systémy, ze kterých získává informace o organizační struktuře, personálních údajích osob, docházce, aktuálních mzdách a pracovním zařazením.

Přístup do aplikace je dvojúrovňový. Je definován přístup, jak k jednotlivým částem aplikace, tak skupinám z organizační struktury.

Objektivně definovaná organizační struktura, která se částečně přenesla ze současných personálních systémů, je definována následujícím způsobem. Jednotlivé osoby jsou přiřazeny na výroby a tyto výroby pak tvoří podskupinu vyšších organizačních celků s názvem divize. Divize zároveň tvoří vrchol organizační struktury. K výrobám jsou dále přiřazeny nejnižší jednotky – pracoviště. Právě na těchto pracovištích pak zaměstnanci získávají různé schopnosti (dále jen skills). Jednotlivá pracoviště, v rámci výrobního procesu, mají stanoveno procento složitosti a kategorii, která

definuje délky zapracování (dle složitosti dané operace) na jednotlivých úrovních (skills). Zaměstnanci jsou na jednotlivé úrovně skills přiřazováni Operation Managerem (dále jen OPS) ve spolupráci s Cell Leaderem a Interním školitelem. Ti zařazují jednotlivé zaměstnance, v rámci každého pracoviště, do jednotlivých úrovní (skills) a to následovně:

1. úroveň – zaškolení na novou operaci nebo skupinu operací (zaměstnanec nevykonává samostatnou práci),
2. úroveň – zaměstnanec vykonává práci pod dozorem,
3. úroveň – zaměstnanec vykonává práci samostatně,
4. úroveň – zaměstnanec vykonává práci samostatně a zaškoluje ostatní (Interní školitel / process owner).

Zařazení zaměstnance do úrovně 1 nebo 2 znamená, že zaměstnanci nepřísluší stanovené procento na daném pracovišti ve SkillMatrix.

Zařazení zaměstnance do úrovně 3 znamená, že zaměstnanci přísluší stanovené procento p na daném pracovišti ve SkillMatrix (SM).

Zařazení zaměstnance do úrovně 4 znamená, že zaměstnanci přísluší stanovené procento na daném pracovišti ve SkillMatrix, vynásobené 4.

V případě zařazení zaměstnance na více pracovišť se stanovená procenta ve SkillMatrix sčítají.

Další klíčovou funkcionalitou systému je výpočet výkonnostního bonusu. Jedná se tedy o spojení systému na doškolování zaměstnanců se systémem výkonnostního bonusu. Výkonnostní bonus je počítán vždy po ukončení měsíce, tedy po uzavření docházky do 10. dne nového měsíce, kdy musí být předány (vyexportovány) podklady mzdovému centru. Toto období se zpravidla nazývá mzdové období (ve vztahu k aplikaci ho nazýváme bonusovým obdobím). Podrobný popis principu výpočtu měsíčního bonusu, včetně požadavků na data, ze kterých má být počítán, je popsán v podkapitole 2.3.3 Výpočet bonusů.

Kromě dvou základních funkcionalit, kterými jsou skills a výpočet bonusů, systém disponuje vyhledáváním osob se zobrazováním osobních karet, které prezentují informace jako fotografie, zařazení osoby na výrobu, pracoviště na kterých má osoba nějaké skills, sumu procent složitosti, která bude použita pro výpočet bonusu a další. Tyto karty je pak možné tisknout a exportovat. Dále pak systém poskytuje správu pracovišť, která umožňuje vkládat a měnit současné nastavení jednotlivých pracovišť. Nedílnou součástí jsou i reporty a sumární report skill osob na jednotlivých pracovištích, report přítomnosti osob v práci dle úrovně skills na pracovišti a report bonusů dle období za všechny divize.

2.3 Specifikace požadavků

Na základě konzultací s personálním oddělením a vedením byly vytyčeny klíčové požadavky, které by měla aplikace splňovat:

- napojení na stávající personální systémy
- pravidelná aktualizace výrob a osob z personálních systémů
- zabezpečené přihlašování do aplikace splňující bezpečnostní požadavky společnosti Honeywell International
- správu uživatelů, rolí, a přístupových oprávnění
- evidenci a správu zaměstnanců dle organizačního schématu
- evidenci a správu schopností zaměstnanců na různých pracovištích dle organizačního schématu
- evidenci, správu a výpočet bonusů zaměstnanců vždy v daném mzdovém období a uchování historie spočítaných bonusů
- otevření a uzavření bonusového respektive mzdového období, ve kterém je možné editovat data pro výpočet bonusů zaměstnanců
- zajištění a využití vždy pouze aktuálních dat, v případě výpočtu bonusů využití dat pouze pro počítané mzdové období (jak z hlediska osob tak z hlediska schopností)
- správu pracovišť a délek zapracování dle organizačního schématu
- import dat zaměstnanců ze souboru
- reporty

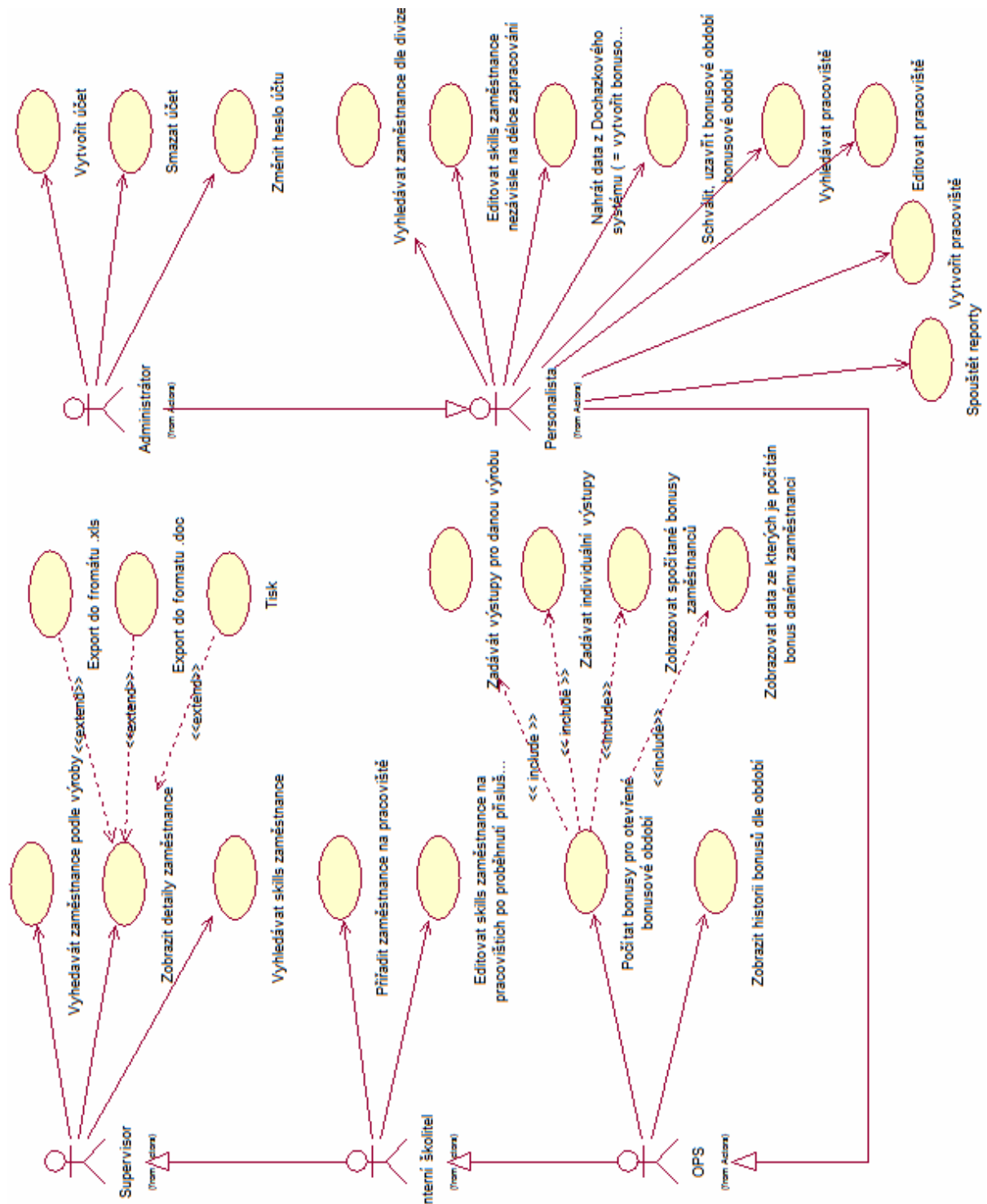
2.3.1 Případy použití

Specifikování klíčových požadavků se stalo podkladem pro vytvoření případů použití, jejich následné zakomponování do logických celků, respektive modulů a jejich asociaci vzhledem k jednotlivým uživatelským rolím.

2.3.1.1 Use Case diagram

Use Case diagram ukazuje jednotlivé případy použití z hlediska uživatelských rolí. Generalizace jednotlivých rolí pak ukazuje, jak roste oprávnění provádět jednotlivé operace. Viz Obrázek 1: Use Case diagram

Obrázek 1: Use Case diagram



2.3.1.2 Role uživatelů

Systém disponuje několika uživatelskými rolemi s různou úrovní editace, a specifickou roli Administrátor.

Uživatelské role:

- Personalista

- OPS / Cell Leader
- Interní školitel
- Supervisor

Administrátorské role:

- Administrátor

2.3.1.3 Moduly a jejich popis

Modul správy účtů

Modul přístupný pouze uživatelům s rolí administrátor. Disponuje funkcemi pro vytváření účtů, mazání účtů a reset hesel.

Vytváření účtů

Umožňuje vytvořit účet, zvolit roli a přístup k osobám.

Mazání účtů

Umožňuje smazat účet dle uživatelského jména a tím zrušit přístup do aplikace.

Reset hesel

Umožňuje resetovat heslo libovolnému uživateli, dle uživatelského jména, při zajištění bezpečnosti hesla uživatele.

Modul správy pracovišť

Modul umožňuje spravovat jednotlivá pracoviště. Pracoviště jsou evidována dle jednotlivých výrobních.

Uživatel s přístupem do této sekce má možnost editovat pracoviště dle organizačního schématu, vkládat pracoviště do organizačního schématu a mazat pracoviště.

Vkládání pracoviště

Při vkládání nového pracoviště musí uživatel vybrat výrobu z organizačního schématu, na kterou si přeje vložit nové pracoviště, následně zadat název pracoviště, váhu a kategorii délek zapracování.

Editace pracoviště

Umožňuje editovat nastavení zvolného pracoviště a to především váhu a kategorii délek zapracování, do které má pracoviště spadat.

Mazání pracoviště

Umožňuje smazat pracoviště z organizačního schématu. Při smazání pracoviště jsou současně smazány všechny vazby na toto pracoviště.

Modul správy osob

Modul umožňující zobrazovat lidi z organizačního schématu, vyhledávat osoby, zobrazovat detaily osob, tisk a export detailů osoby do formátů .xls (Microsoft Office Excell) a .doc (Microsoft Office Word)

Vyhledávání osob

Kromě základního zvolení výroby a následného zobrazení osob na dané výrobě, může uživatel na zvolené výrobě vyhledávat osoby a to dle osobního čísla a dle jména, příjmení či jen části jména. Vždy jsou zobrazeny všechny osoby odpovídající hledanému výrazu.

Zobrazování detailů osoby

Uživatel má možnost si u každé osoby zobrazit její detaily. V těchto detailech je vidět fotka, osobní číslo, jméno a příjmení, výroba na které je osoba přiřazena, přítomnost osoby v práci (aktuální údaj z docházkového systému) a v neposlední řadě, přehled skills dané osoby na pracovištích kde již započalo zapracování.

Zobrazené detaily pak uživatel může tisknout a exportovat.

Modul správy skills

Modul slouží k přehledu skills osoby, přiřazování osob na zvolné pracoviště, respektive započetí tréninku na nějaký skill a také změnu skill na daném pracovišti v závislosti na informaci o probíhající a dobehlé době zapracování na daném pracovišti. K dispozici je také detail zvolného skill na daném pracovišti.

Zobrazení skills

Po vybrání osoby ze seznamu osob dle zvolné výroby jsou zobrazeny pracoviště na dané výrobě a skills osoby na těchto pracovištích s informací o tom zda dobehla či stále probíhá délka zapracování.

Editace skills

Standardně je možné měnit skill pouze po uplynutí délky zapracování, pouze uživatel s rolí personalista či administrátor má oprávnění měnit skill kdykoli, respektive nezávisle na délce zapracování. Při každé změně skill začne od data změny běžet odpovídající doba zapracování, na daném pracovišti, na vyšší skill.

Zobrazení detailu skills

U každého skill je možné si zobrazit jeho detaily s informacemi o datu přiřazení současného skill, hodnota současného skill, délce zapracování na vyšší skill, datu možnosti získání vyšší skill a hodnota vyššího skill.

Modul importu dat z docházkového systému

Umožňuje importovat data za dané období, ze souboru z docházkového systému, ve formátu .txt.

Současně s importem je třeba zadat fond pracovní doby pro odpovídající období.

Po importu souboru je se vytvoří bonusové období a je možné zadávat data pro bonusy pro dané období.

Modul výpočtu bonusů

Modul umožňuje, pokud je otevřené bonusové období, zadávat data pro výpočet bonusů. Jedná se o zadávání výstupů a auditu na pracovišti za dané období, a to buď jednotně pro celou výrobu, nebo individuálně přímo zvolené osobě.

Pokud nejsou zadány výstupy, provede se výpočet pouze u osob, které mají skills 4 respektive mají dle procedury a popsanych vzorců nárok na bonus za skill 4 i v případě nulového výstupu.

Po zadání výstupů jsou zobrazeny spočítané bonusy všech osob pro všechny skills pro zvolenou výrobu a dílčí důležité spočítané vzorce jako je celkový výstup (Output), HSEA, suma vah, tyto data jsou doplněna o data z docházkové systému (fond a odpracovaná doba) a v poslední řadě o data z personálního systému (základní mzda). Osoby, pro které platí odlišný výpočet bonusu tj. osoby se skills 4, jsou zvýrazněny oproti ostatním osobám. Ke každé osobě je pak možné si zobrazit detaily, kde jsou vypsána podrobná data ze kterých výpočet bonusů, dané osobě, proběhl.

Modul schvalování bonusového období

Modul umožňuje uzavřít otevřená mzdová období a tím ukončit možnost zadávání dat pro jejich výpočet. Po uzavření bonusového období jsou uloženy spočítané bonusy a jejich viditelnost je dále možná jen v historii bonusů.

Tímto je zajištěno, že v případě změny procesu výpočtu bonusů zůstane zachována historie a zároveň zajištěna nežádoucí možnost pozdní editace již spočítaných bonusů.

Modul historie spočítaných bonusů

Umožňuje pohled na historii vyplacených bonusů.

Modul Kiosek

Samostatný modul. Umožňuje na základě zadání kódu identifikační karty zobrazit kartu osoby se získanými skills (použití na informačních kioscích pro zaměstnance).

2.3.1.4 Přístupová práva

Každý uživatel se do aplikace musí přihlásit pod jedinečným uživatelským jménem a heslem. Toto heslo si může uživatel samostatně změnit. Heslo musí mít minimální délku šest znaků, alfanumerický charakter a je třeba použít alespoň jeden speciální znak.

Oddělenou částí je pak osobní přístup, který využívá zcela odlišný systém autentizace než zbytek systému. Tato část podléhá přihlášení na elektronickou identifikační kartu, kterou disponuje každý zaměstnanec.

Kromě samostatného přístupu do aplikace je přístup členěn do dvou částí a to na:

- Uživatelské role
- Přístup k osobám dle organizačního schématu

Uživatelské role

Role Administrátor

Má přístup do všech částí systému, s nejvyšším oprávněním editace.

Jednotlivé přístupy:

- Modul správy účtů
- Modul správy pracovišť
- Modul správy osob
- Modul správy skills
- Modul importu dat z docházkového systému
- Modul výpočtu bonusů
- Modul schvalování bonusového období
- Modul historie spočítaných bonusů

Role Personalista

Má přístup do všech částí systému, s nejvyšším oprávněním editace, kromě modulu pro správu přístupových oprávnění.

Jednotlivé přístupy:

- Modul správy pracovišť
- Modul správy osob
- Modul správy skills
- Modul importu dat z docházkového systému
- Modul výpočtu bonusů
- Modul schvalování bonusového období
- Modul historie spočítaných bonusů

Role OPS / Cell Leader

Jednotlivé přístupy:

- Modul správy osob
- Modul správy skills

- možnost editace skills pouze v závislosti na proběhlé délce zapracování
- Modul výpočtu bonusů
- Modul historie spočítaných bonusů

Role Interní školitel

Jednotlivé přístupy:

- Modul správy osob
- Modul správy skills
- možnost editace skills pouze v závislosti na proběhlé délce zapracování

Role Supervisor

Jednotlivé přístupy:

- Modul správy osob
- Modul správy skills
- přístup pouze pro čtení bez možnosti editace

Přístup k osobám dle organizačního schématu

Ke každé roli je možné přiřadit i přístup k osobám na základě přiřazení skupiny z vrcholu stromu organizační struktury, kterou je divize. Uživatel tak může mít, buď přístup k datům pouze jedné divize, nebo k datům všech divizí.

2.3.2 Firemní procesy

Pro klíčové firemní procesy, jež mají být v aplikaci nastaveny, jsou vytvořeny diagramy aktivity (v terminologii UML 2.0). Diagramy znázorňují interakce akcí a stavů vznikající v průběhu procesu z hlediska uživatelských rolí.

Diagram aktivity nastavování skills

Znázorňuje proces nastavování skills vzhledem k rolím OPS / Interní školitel a Personalista.

Uživatel v roli OPS a Interní školitel má oprávnění měnit skills pouze po proběhlé délce zapracování.

Pokud je potřeba změnit skill v době zapracovávání, musí být změna provedena uživatelem v roli

Personalista. Tím je zajištěno, jak dodržení příslušných délek zapracování, tak zároveň pokrytí

vyjímek, které mohou nastat (změna skill v probíhající době zapracování).

Viz Obrázek 2: Diagram aktivity skills

Obrázek 2: Diagram aktivity skills

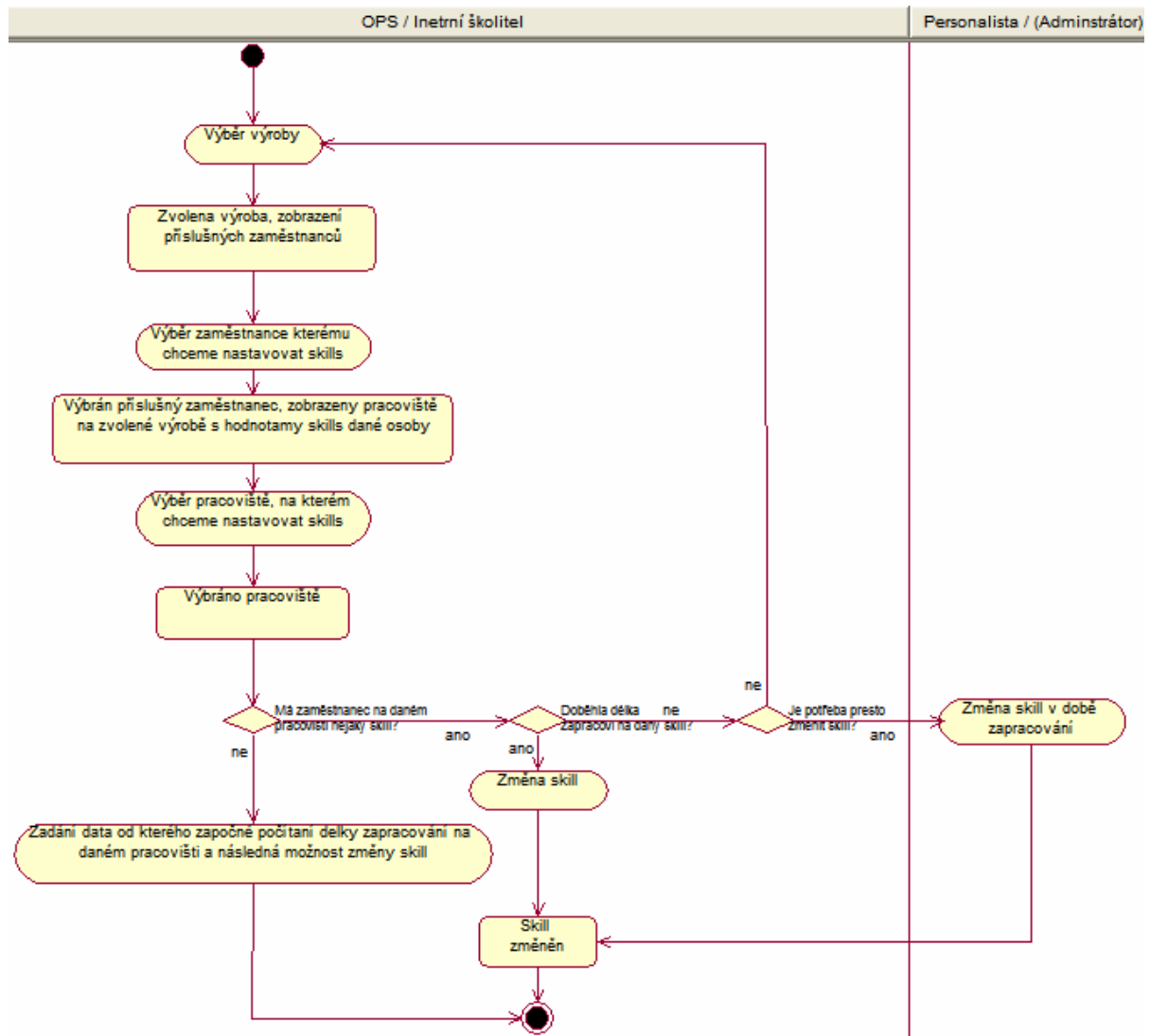
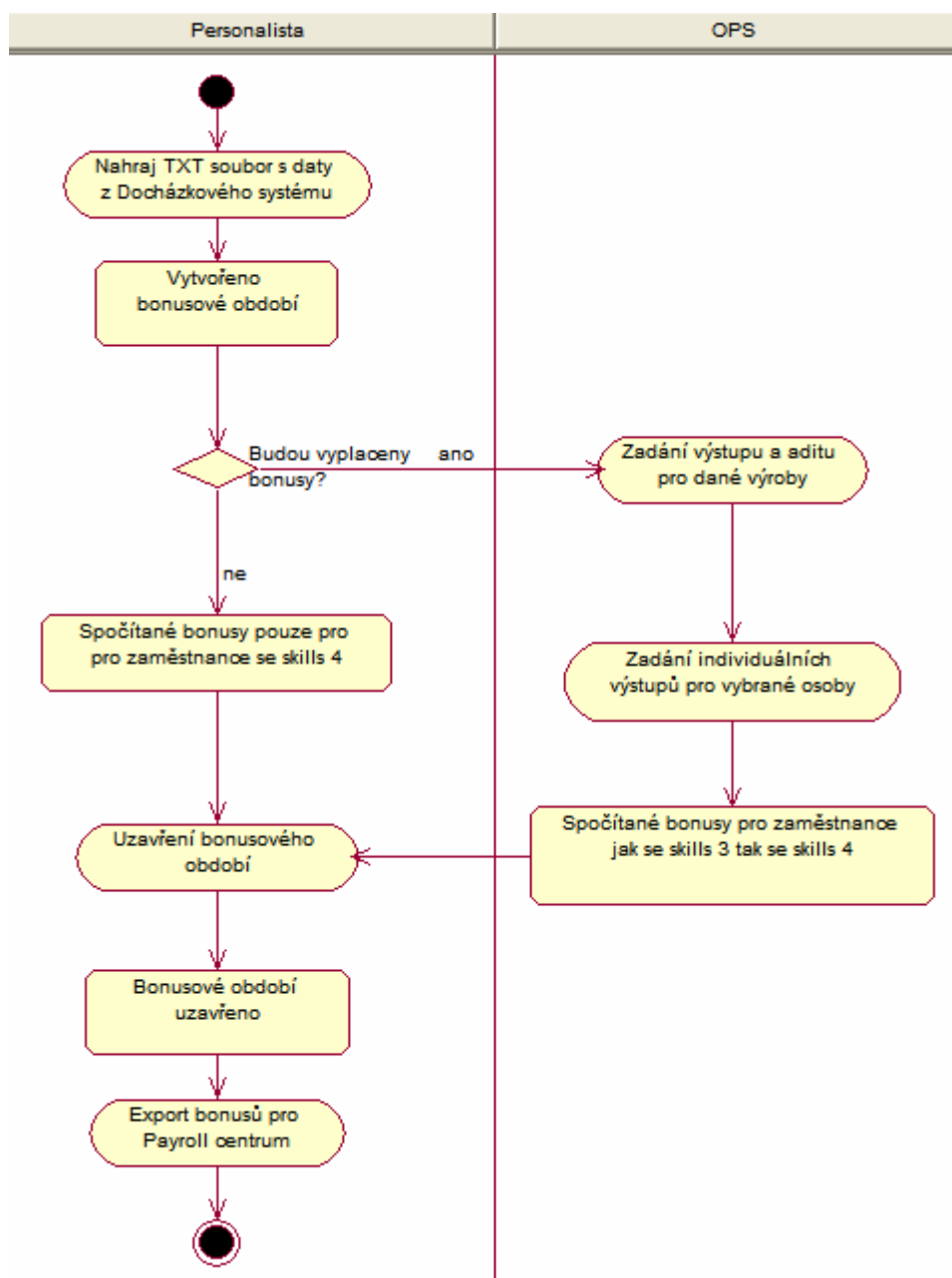


Diagram aktivity výpočtu bonusů

Znázorňuje princip výpočtu bonusů vzhledem k rolím OPS a Personalista. Uživatel v roli OPS je zodpovědný za zadání dat pro výpočet bonusů v otevřeném bonusovém období. Personalista pak za nahrání dat z docházkového systému, tedy vytvoření bonusového období a následně za jeho uzavření a export dat pro mzdové centrum.

Viz příloha Obrázek 3: Diagram aktivity bonusy

Obrázek 3: Diagram aktivity bonusu



2.3.3 Výpočet bonusů

Tato kapitola definuje klíčové informace a vstupní data pro výpočet bonusů.

Důležitým aspektem pro výpočet bonusu je váha (procentuální složitost), která musí být zadána pro každé jednotlivé pracoviště. Tato váha definuje složitost práce v porovnání s ostatními pracovišti. Jelikož objektivně nelze nastavit systém těchto vah v závislosti na délce zapracování, je třeba, aby každé pracoviště bylo zařazeno do kategorie délek zapracování. Tabulka 1 definuje čtyři kategorie složitosti práce, kam jsou všechny pracoviště zařazeny. Jednotlivé kategorie dle divize a složitosti práce (jedna až čtyři), definují vždy příslušné délky zapracování, které jsou nezbytné pro dosažení vyšší složitosti. Díky tomuto porovnání je zaručena objektivita vah pro každou operaci. Do

výkonnostního bonusu se však započítávají pouze operace, které zaměstnanec zná na 3 úroveň. Důvodem, proč se bonus započítává až od 3. úrovně je, že teprve zaměstnanec na třetí operaci je schopen samostatné práce bez nutnosti dohledu přímého nadřízeného.

Dalším kritériem pro výpočet výkonnostního (respektive měsíčního) bonusu, je výstup (Output) z každého pracoviště. Tento výstup nebo-li cíl definuje množství výrobků na směnu, které má být vyrobeno. Pokud daný operátor (výrobní dělník) není správně proškolen, není ani schopen tento cíl splnit. Z toho důvodu sledujeme množství správných výrobků na daném pracovišti versus cíl daného pracoviště.

Třetím kritériem ve výpočtu výkonnostního bonusu je audit bezpečnosti práce. Tento audit se provádí na týdenní bázi a kontroluje dodržování předpisů bezpečnosti práce a pořádek na pracovišti.

Čtvrtým kritériem je odpracování fondu pracovní doby. Toto kritérium je do výkonnostního bonusu započítáno z důvodu problémů s absencí.

Tabulka 1: Délky zapracování na jednotlivých úrovních operací v kategoriích 1–4

úroveň skills		1	2	3	4
kategorie (tarif)					
1	ENVI	2 dny	1 měs.	3 měs.	6 měs.
	CCV	2 týdny	2 měs.	4 měs.	8 měs.
	CCE	2 týdny	2 měs.	4 měs.	8 měs.
2	ENVI	2 týdny	2 měs.	4 měs.	6 měs.
	CCV	2 měs.	4 měs.	8 měs.	12 měs.
	CCE	2 měs.	4 měs.	8 měs.	12 měs.
3	ENVI	2 měs.	4 měs.	8 měs.	12 měs.
	CCV	3 měs.	5 měs.	8 měs.	16 měs.
	CCE	3 měs.	5 měs.	8 měs.	16 měs.
4*	ENVI	X	x	x	x
	CCV	6 měs.	8 měs.	12 měs.	18 měs.
	CCE	6 měs.	8 měs.	12 měs.	18 měs.

Suma procent složitosti (vah) (SM)

Pro skill úrovně 3:

$$SM = (\text{váha na daném pracovišti})$$

Pro skill úrovně 4:

$$SM 4 = (\text{váha na daném pracovišti}) * 4$$

Output

Výpočet poměru výstupu je prováděn za předcházející kalendářní měsíc pro jednotlivé výrobní divize podle následujícího vzorce:

$$O = \frac{\text{Počet kusů výrobků připravených k odeslání (odhlášených ze systému za období)}}{\text{výrobní plán dle HOS za období}}$$

Fond pracovní doby

Výpočet poměru plnění Fondu pracovní doby je prováděn za předcházející kalendářní měsíc pro jednotlivce podle následujícího vzorce:

$$PD = \frac{\text{počet skutečně odpracovaných hodin přesčasové práce}}{\text{fond pracovní doby}}$$

HSE5S audit

Bonus je počítán z průměrného měsíčního hodnocení HSE5S auditu příslušné výrobní divize / buňky. Rozsah hodnocení auditu je 0 až 10 bodů po kroku 0,1 bodu. Při počtu bodů rovných nebo menších než 5, je procento bonusu 0. Za každý další 0,1 bod nad 5 bodů (tzn. 5,1 a vyšší) je procento bonusu zvýšeno o 0,05%. Výpočet pro jednotlivé výrobní divize / buňky je prováděn podle následujícího vzorce.

$$HSE5S = (\text{průměrné měsíční hodnocení HSE5S auditu} - 5) * 0,5$$

Výpočet bonusu

Měsíční bonus pro zaměstnance zařazené do SkillMatrix úrovně 1. až 3.

Procento měsíčního bonusu je vypočteno podle následujícího vzorce:

$$\text{Měsíční bonus (MB)} = O * PD * 100 + SM + HSE5S - 90$$

V případě, že výše měsíčního bonusu podle výše uvedeného vzorce je 0 nebo v záporné hodnotě, zaměstnanec nemá nárok na výplatu měsíčního bonusu.

Měsíční bonus pro zaměstnance zařazené do SkillMatrix úrovně 4.

Pro zaměstnance zařazené do úrovně 4. platí výpočet bonusu uvedený výše a část měsíčního bonusu

$$\text{Část měsíčního bonusu (ČMB)} = SM4 * PD * 100$$

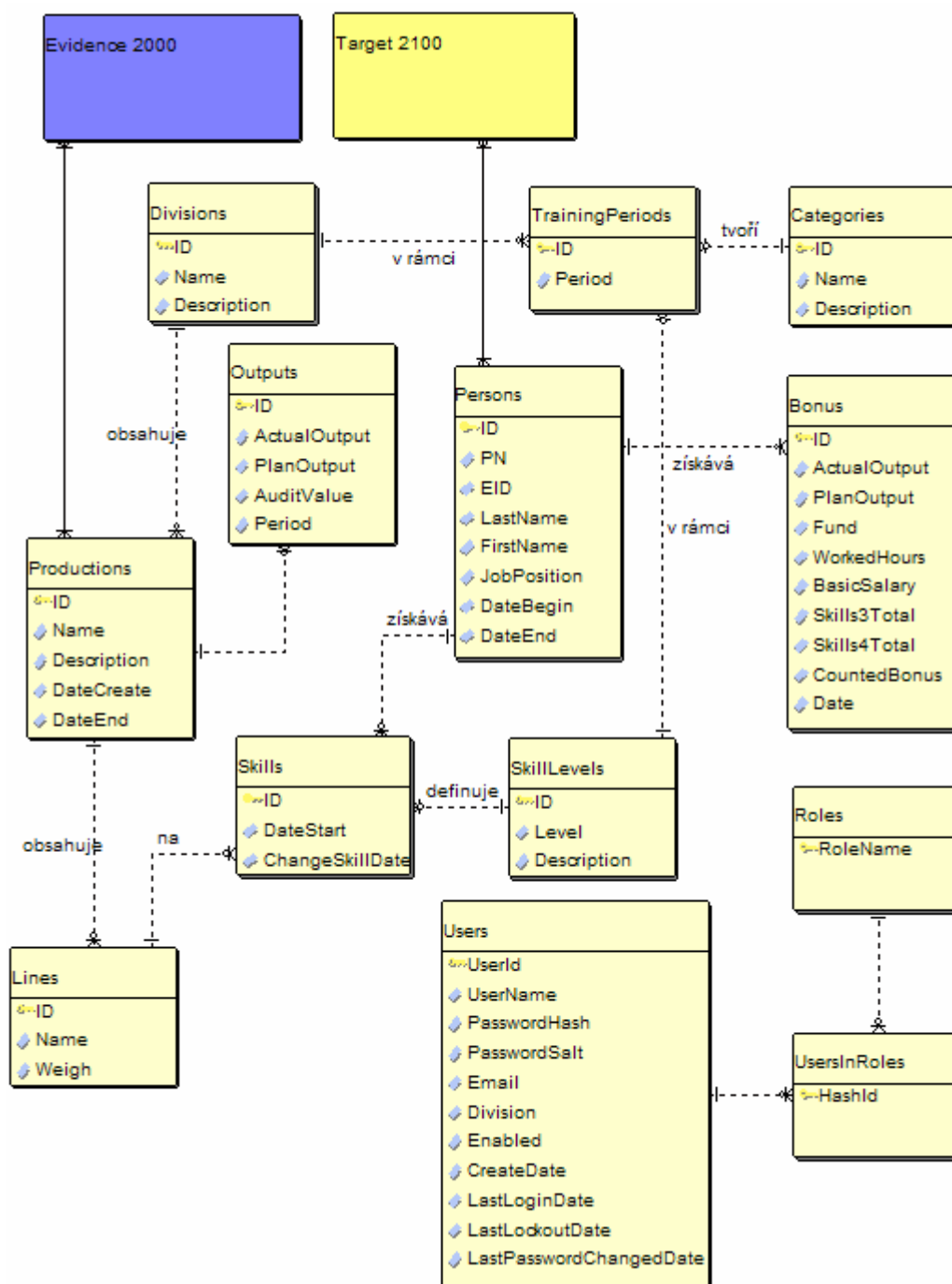
2.3.4 Datový model

2.3.4.1 ER diagram

ER diagram představuje reprezentaci hlavních entit a relací datového modelu. Relace v podobě plných čar naznačují, které entity bude třeba aktualizovat z jakého personálního systému.

Viz Obrázek 4: ER diagram

Obrázek 4: ER diagram



2.3.4.2 Atributy entit

Entita Divisions

- ID – Primární klíč
- Name – Název divize
- Description – Popis, případně delší název.

Entita Productions

- ID – Primární klíč
- Name – Název výroby.
- Description – Popis výroby, případně delší název.
- DateCreate – Datum vytvoření respektive počátku platnosti
- DateEnd – Datum konce platnosti

Entita Lines

- ID – Primární klíč, automaticky generovaný
- Name – Název pracoviště.
- Weigh – Váha, kterou může zaměstnanec získat, na daném pracovišti, po dosažení skill hodnoty 3.

Entita Categories

- ID – Primární klíč, automaticky generovaný.
- Level – Úroveň skill.
- Description – Popis dané úrovně.

Entita TrainingPeriods

- ID – Primární klíč, automaticky generovaný.
- Period – Délka zapracování na danou úroveň skill ve dnech.

Entita Persons

- ID – Primární klíč, automaticky generovaný
- PN – Osobní číslo osoby (jedinečné).
- EID – Jedinečný identifikátor osoby a zároveň přihlašovací jméno osoby do počítače.
- LastName – Příjmení.
- FirstName – Křestní jméno.
- JobPosition – Pracovní pozice osoby.
- DateBegin – Datum nástupu.

- DateEnd – Datum výstupu. Udává ukončení viditelnosti dané osoby, vždy od prvního dne následného měsíce.

Entita Skills

- ID - Primární klíč, automaticky generovaný.
- DateStart – Datum, které udává započetí zpracovávání na daném pracovišti.
- ChangeSkillDate – Datum změny vždy aktuální úrovně skill. Od tohoto data se dle dané úrovně skill počítá příslušná délka zpracování.

Entita SkillLevels

- ID – Primární klíč, automaticky generovaný.
- Level – Úroveň skill.
- Description – Popis dané úrovně.

Entita Outputs

- ID – Primární klíč, automaticky generovaný
- ActualOutput – Aktuální tedy skutečný výstup pro danou výrobu za dané období.
- PlanOutput – Plánovaný tedy předpokládaný výstup pro danou výrobu za dané období.
- AuditValue – Hodnota auditu na dané výrobě za dané období.
- Period – Bonusové období definované vždy ve tvaru rok, měsíc, den respektive XXXX-XX-01.

Entita Bonus

- ID – Primární klíč, automaticky generovaný
- PN – Osobní číslo osoby (jedinečné).
- ActualOutput – Aktuální respektive skutečný výstup z výroby za dané období. V případě individuálního zadání výstupu má hodnotu real jinak Null.
- PlanOutput – Plánovaný výstup z výroby za dané období. V případě individuálního zadání výstupu má hodnotu real jinak Null.
- Fund – Fond pracovní doby za dané období. Zadáván jednorázově hromadně, je možné ho individuálně měnit.
- WorkedHours – Celková odpracovaná doba za dané období, vypočtená z dat z docházkového systému (z naimportovaného souboru formátu .txt)
- BasicSalary – Základní mzda, pro dané období, z mzdového systému.
- Skills3Total – Suma vah pro skills zaměstnance rovno 3 pro dané období.
- Skills4Total – Suma vah pro skills zaměstnance rovno 4 pro dané období.
- CountedBonus – Spočítaný výsledný bonus za dané období.

- Date – Bonusové období definované vždy ve tvaru rok, měsíc, den respektive XXXX-XX-01

Entita Users

- UserId – Primární klíč, automaticky generovaný.
- UserName – Uživatelské jméno pro přihlášení do aplikace. Jedinečný identifikátor.
- PasswordHash – Zašifrované heslo pomocí SHA512 + salt.
- PasswordSalt – Náhodně vygenerovaný salt pro šifrování hesla.
- Email – Emailová adresa.
- Division – ID divize, kterou má uživatel vidět. Pokud je prázdná, vidí všechny divize.
- Enabled – Udává, zda je účet aktivní. Hodnota True nebo False.
- CreateDate – Datum vytvoření účtu.
- LastLoginDate – Poslední přihlášení do aplikace.
- LastLockoutDate – Datum posledního odhlášení z aplikace.
- LastPasswordChangedDate – Datum poslední změny hesla.

Entita Roles

- RoleName – Primární klíč, jedinečný identifikátor, název uživatelské role.

Entita UsersInRoles

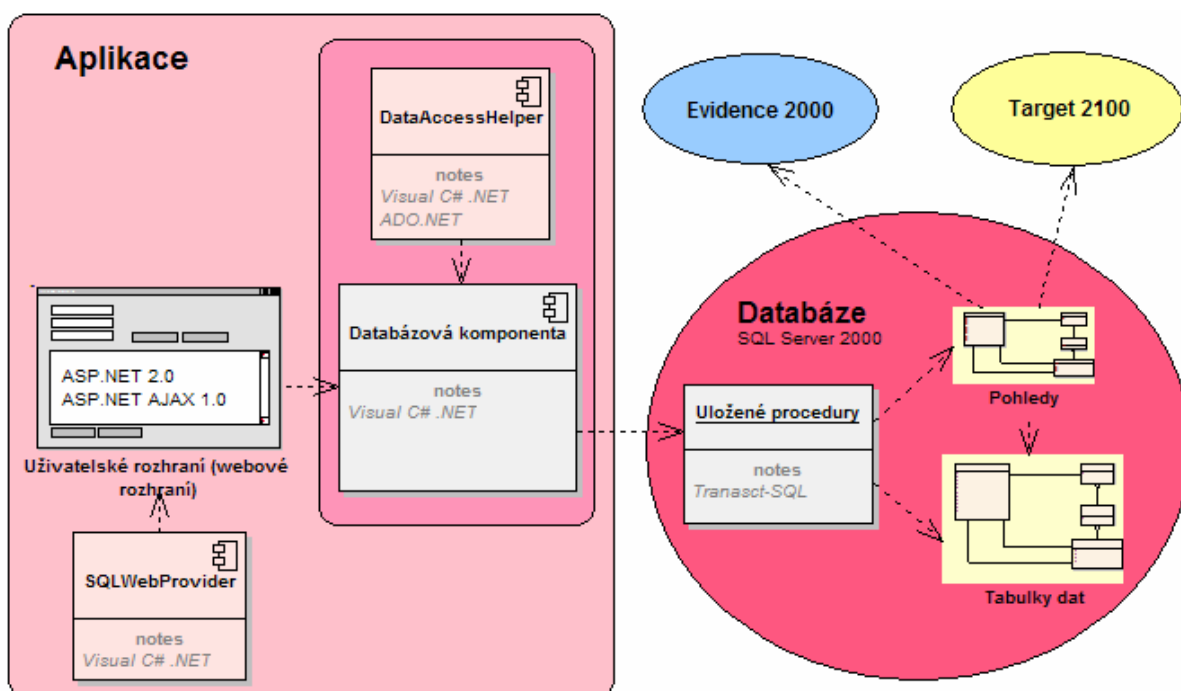
- HashId – Primární klíč, automaticky generovaný.

3 Návrh a implementace

3.1 Architektura systému

Celá architektura systému je postavena na technologiích Microsoft. Na obrázku 5 jsou vidět jednotlivé komponenty systému.

Obrázek 5: Architektura systému



Systém lze z hlediska architektury rozdělit na dva základní celky, aplikaci a databázi. Aplikace je implementována na platformě .NET Framework a databáze na platformě SQL Server.

Aplikaci lze pak dále rozdělit na databázovou komponentu, uživatelské rozhraní a přístupová práva.

Databázová komponenta obsahuje separátní třídy pro každou databázovou tabulku (nebo jejich logické propojení) a představuje prostředek komunikace mezi uživatelským rozhraním a databází. Každá takováto třída je pak implementována pomocí knihovny DataAccessHelper implementující jednotlivé knihovny ADO.NET.

Uživatelské rozhraní je pak implementováno v ASP.NET s využitím technologie AJAX. Kde jsou přidány knihovny nezbytné pro využití této technologie.

Přístupová oprávnění a tedy řízení přístupu do aplikace je definováno v samostatné knihovně SqlWebProviders.

Uživatelského rozhraní

Platforma: Microsoft. .NET Framework 2.0

Technologie: ASP.NET 2.0 s využitím technologie ASP.NET AJAX 1.0 a ASP.NET AJAX Extensions

Vývojový nástroj: Microsoft Visual studio .NET 2005

Databázová komponenta (a knihovna DataAccessHelper)

Platforma: Microsoft. .NET Framework 2.0

Technologie: Microsoft Visual .NET C# s využitím knihoven ADO.NET

Vývojový nástroj: Microsoft Visual studio .NET 2005

Přístup do aplikace

Platforma: Microsoft .NET Framework 2.0 Membership and Roles API (SqlMembershipProvider a SqlRoleProvider)

Technologie: Microsoft Visual .NET C# s využitím knihoven ADO.NET

Vývojový nástroj: Microsoft Visual studio .NET 2005

Databáze

Platforma: Microsoft SQL Server 2000

Technologie: Transact-SQL

Vývojový nástroj: Microsoft SQL Server 2000 Enterprise Manager

3.1.1 Použité technologie

3.1.1.1 Platforma .NET Framework

Microsoft. NET Framework je nová počítačová platforma, která usnadňuje vývoj aplikací v distribuovaném prostředí sítě Internet. Platforma .NET Framework je navržena jako prostředí pro objektově orientované programování zajišťující zabezpečené provedení kódu a zamezující potížím s výkonem skriptovacích prostředí.

Platforma .NET Framework obsahuje dvě hlavní součásti: společný běhový jazykový modul a knihovnu tříd platformy .NET Framework. Technologie Microsoft ASP.NET je hostitelem běhového modulu, který poskytuje základní služby, jako jsou správa paměti, správa podprocesů a řízení vzdálené komunikace, a zároveň vynucuje přísné zabezpečení typů a jiné formy přesnosti kódu zajišťující zabezpečení a odolnost. Hlavním principem běhového modulu je správa kódu. Kód, který volá běhový modul, se nazývá spravovaný kód, zatímco kód, který běhový modul nevolá, se nazývá nespravovaný kód.

Knihovna tříd platformy .NET Framework je ucelený, objektově orientovaný soubor opakovaně použitelných typů, které mohou vývojáři použít k vytváření aplikací ASP.NET.

Funkce společného běhového jazykového modulu

Společný běhový jazykový modul (CLR) poskytuje správu paměti, provádění podprocesů, provádění kódu, ověřování zabezpečení kódu, kompilaci a jiné systémové služby. Tyto funkce jsou vlastní spravovanému kódu spuštěnému ve společném běhovém jazykovém modulu, jehož hostitelem je technologie ASP.NET. Spravovaným součástí je přidělen různý stupeň důvěryhodnosti podle řady faktorů, včetně původu (například Internet, rozlehlá síť nebo místní počítač). To znamená, že spravovaná součást může nebo také nemusí být schopná vykonávat citlivé funkce, přestože je používána ve stejné aktivní aplikaci.

Modul CLR vynucuje zabezpečení přístupu ke kódu. Aplikace mohou mít přidělený přístup k určitým prostředkům, ale nemají přístup k prostředkům mimo tato omezení.

Modul CLR dále vynucuje odolnost kódu implementací struktury přísného ověřování typu a kódu nazývané CTS (Common Type System). Systém CTS zajišťuje, aby spravovaný kód obsahoval vlastní popis, což znamená, že spravovaný kód může používat jiné typy a instance bez ohledu na jazyk kompilátoru použitý k vývoji softwaru.

Spravované prostředí modulu CLR navíc zamezuje mnoha častým problémům se softwarem. Modul CLR například automaticky zpracovává rozložení objektu, spravuje odkazy na objekty a nepoužívané objekty uvolňuje. Tato automatická správa paměti řeší dvě nejčastější chyby aplikací: nevrácenou paměť a neplatné odkazy do paměti. Modul CLR je navržen s cílem zvýšit výkon. Přestože modul CLR poskytuje mnoho standardních běhových služeb, není spravovaný kód nikdy interpretován. Kompilace JIT (Just-in-time) umožňuje, aby celý spravovaný kód pracoval v nativním strojovém jazyce systému, ve kterém je spouštěn. Další zvýšení výkonu je úlohou správce paměti.

Funkce součásti knihovna tříd platformy .NET Framework

Knihovna tříd platformy .NET Framework je souborem opakovaně použitelných typů, které jsou pevně integrovány do modulu CLR. Knihovna tříd je objektově orientovaná a nabízí typy, z kterých mohou aplikace odvozovat další funkce. Tato objektová orientace umožňuje snadné použití typů platformy .NET Framework, což zároveň zkracuje čas nutný k seznámení se s jejími novými funkcemi. Kromě toho mohou být do tříd platformy .NET Framework bezproblémově integrovány součásti jiných dodavatelů.

Typy platformy .NET Framework umožňují plnění řady programovacích úloh, jako správa řetězců, sběr dat, připojení k databázi a přístup k souborům. Kromě toho zahrnuje knihovna tříd typy podporující řadu specializovaných scénářů vývoje.

Grafické schéma platformy .NET Framework je vidět na obrázku 6.

Obrázek 6: Schéma platformy .NET Framework



3.1.1.2 Technologie ASP.NET

Technologie ASP.NET je jednotná platforma webových aplikací, která poskytuje služby nezbytné pro vytváření a zavádění podnikových webových aplikací. Technologie ASP.NET nabízí nový model programování a infrastruktury zajišťující bezpečnější, škálovatelnější a stabilnější aplikace, které mohou komunikovat s jakýmkoli prohlížečem nebo zařízením.

Technologie ASP.NET je součástí rozhraní Microsoft .NET Framework, počítačového prostředí, které zjednodušuje vývoj aplikací ve vysoce distribuovaném prostředí sítě Internet.

Technologie ASP.NET nabízí následující výhody:

- **Správa:** Technologie ASP.NET používá textový hierarchický systém konfigurace, který zjednodušuje použití nastavení v prostředí serveru a webových aplikací. Vzhledem k tomu, že informace týkající se konfigurace jsou uloženy jako prostý text, může být nové nastavení použito bez pomoci místních nástrojů správy. Jakékoli změny konfiguračních souborů budou automaticky rozpoznány a použity v aplikaci.
- **Zabezpečení:** Technologie ASP.NET poskytuje výchozí schémata autorizace a ověřování pro webové aplikace. Vývojáři mohou tyto schémata snadno doplňovat, odebírat nebo nahrazovat podle potřeb aplikace.
- **Snadné zavedení:** Aplikace technologie ASP.NET lze na server implementovat pouhým zkopírováním potřebných souborů. Není třeba server restartovat, a to ani v případě zavedení nebo nahrazení spuštěného zkompilevaného kódu.
- **Zvýšený výkon:** Technologie ASP.NET je zkompilevaný kód spuštěný na serveru. Na rozdíl od tradičního prostředí ASP (Active Server Pages) může technologie ASP.NET okamžitě používat výhody služeb časné vazby, kompilace JIT (just-in-time), nativní optimalizace a služeb mezipaměti pro zlepšení výkonu.
- **Flexibilní ukládání výstupu do mezipaměti:** Technologie ASP.NET může ukládat data stránek, části stránek nebo celé stránky do mezipaměti v závislosti na potřebách aplikace.

Položky uložené v mezipaměti mohou být závislé na souborech nebo na jiných položkách v mezipaměti nebo mohou být aktualizovány na základě zásad platnosti.

- Mezinárodní nastavení: Technologie ASP.NET vnitřně používá pro data požadavků a odpovědí kódování Unicode. Mezinárodní nastavení lze konfigurovat pro každý počítač, každý adresář a každou stránku.
- Podpora mobilních zařízení: Technologie ASP.NET podporuje jakýkoli prohlížeč v libovolném zařízení. Vývojáři používají ke komunikaci s novými mobilními zařízeními stejné techniky programování jako u tradičních prohlížečů stolních počítačů.
- Škálovatelnost a dostupnost: Technologie ASP.NET byla navržena tak, aby byla škálovatelná a aby její funkce vedly ke zlepšení výkonu v clusterových a víceprocesorových prostředích. Kromě toho jsou procesy pozorně sledovány a spravovány službou IIS (Internetová informační služba) a běhovým modulem ASP.NET. To znamená, že v případě nesprávné funkce některého procesu může být místo něj vytvořen jiný proces, což umožňuje udržovat schopnost aplikace zpracovávat požadavky.
- Sledování a ladění: Technologie ASP.NET poskytuje služby sledování, které lze povolit při ladění na úrovni aplikace i na úrovni stránky. Můžete si vybrat, zda chcete tyto informace zobrazit na stránce nebo na úrovni aplikace pomocí nástroje pro zobrazení sledování. Technologie ASP.NET podporuje místní a vzdálené ladění pomocí ladicích nástrojů rozhraní .NET Framework, a to během vývoje i při provozu aplikace. Pokud je aplikace uvedena do provozu, mohou být trasovací příkazy ponechány v provozním kódu, aniž by byl jakkoli ovlivněn výkon.
- Integrace s rozhraním .NET Framework: Vzhledem k tomu, že technologie ASP.NET je součástí rozhraní .NET Framework, je výkon a flexibilita celé platformy k dispozici webovým aplikacím. Knihovna tříd .NET a veškerá řešení zasílání zpráv a přístupu k datům jsou bez potíží dostupná z internetu. Technologie ASP.NET je nezávislá na jazyku, a vývojáři si proto mohou vybrat jazyk, který aplikaci nejlépe odpovídá. Kromě toho schopnost spolupráce společného běhového jazykového modulu zachovává existující investice do vývoje založeného na modelu COM.
- Kompatibilita s existujícími aplikacemi ASP: Technologie ASP a ASP.NET lze spustit souběžně na webovém serveru služby IIS, aniž by docházelo k vzájemnému rušení. Neexistuje možnost poškození existující aplikace ASP instalací technologie ASP.NET. Tato technologie zpracovává pouze soubory s příponou ASPX. Soubory s příponou ASP budou nadále zpracovány modulem ASP. Mějte však na paměti, že stránky ASP a ASP.NET nesdílejí stav relace a stav aplikace.

Technologie ASP.NET umožňuje použití dvou funkcí pro distribuované aplikace: webové formuláře a webové služby XML. Tyto dvě funkce jsou podporovány stejnou konfigurací a infrastrukturou ladění.

- Technologie webových formulářů umožňuje vytváření výkonných formulářových webových stránek. Stránky webových formulářů jsou opakovaně použitelné integrované nebo vlastní součásti, které zjednodušují kód stránky.
- Webové služby XML vytvořené pomocí technologie ASP.NET umožňují vzdálený přístup k serverům. Pomocí webových služeb XML lze programovacím rozhraním poskytnout data nebo obchodní logiku, které mohou dále obdržet a zpracovat klientské a serverové aplikace. Webové služby XML umožňují výměnu dat pomocí standardů, jako jsou například zasílání zpráv XML a přenos HTTP přes brány firewall ve scénářích klient/server a server/server. Webové služby XML mohou volat programy napsané v libovolném jazyce a pracující v jakémkoli operačním systému.

Webové aplikace ASP.NET

Aplikace platformy Microsoft ASP.NET je definována jako sada všech souborů a spustitelného kódu, které lze vyvolat z virtuálního adresáře a jeho podadresářů na webovém serveru. Může se jednat o webové stránky (soubory HTML), stránky webových formulářů (soubory ASPX), uživatelské ovládací prvky webových formulářů (soubory ASCX), webové služby XML (soubory ASMX), popisovače protokolu HTTP, moduly protokolu HTTP a další soubory (jako jsou obrázky a konfigurační soubory). Součástí aplikace ASP.NET jsou také všechna mapování skriptů přidružená k používané verzi platformy Microsoft .NET Framework.

Pokud je aplikace platformy ASP.NET vyvíjena pomocí sady Microsoft Visual Studio .NET, v podadresáři \Bin kořenového adresáře aplikace bude uložena knihovna DLL projektu. Všechny další knihovny DLL, které poskytují funkce využívané aplikací, musí být umístěny rovněž v adresáři \Bin, ať byla aplikace vyvinuta pomocí sady Visual Studio nebo nikoli. Mezi tyto knihovny patří zkompileované uživatelské serverové ovládací prvky technologie ASP.NET, obchodní logika nebo logika přístupu k databázím.

Vztah mezi aplikací platformy ASP.NET a Internetovou informační službou (IIS) je následující: služba IIS zpracovává všechny požadavky na publikovatelné soubory platformy ASP.NET pomocí modulu aspnet_isapi.dll (procesní model pro platformu ASP.NET). Služba IIS nicméně nezpracovává programový kód určený k provedení na straně serveru. Tuto činnost vykonává stroj ASP.NET a výstup následně předává službě IIS (nebo jinému webovému serveru), která obsluhuje požadavek klienta.

Rozložení webové aplikace ASP.NET

Aplikace ASP.NET musí být umístěna ve virtuálním adresáři služby IIS, který je označován jako kořenový adresář aplikace. Aplikace ASP.NET mohou obsahovat zkompileovaná sestavení (obvykle soubory DLL, které obsahují obchodní logiku), známý adresář vždy nazvaný \Bin, ve kterém je uložen předkompilovaný kód, nastavení konfigurace uložené v textovém souboru Web.config, který má srozumitelný formát, stránky, serverové ovládací prvky a webové služby XML.

Jakýkoli předkompilovaný kód, který není sdílen ostatními aplikacemi na serveru, musí být uložen v adresáři aplikace \Bin. Tento adresář je místní mezipaměť sestavení aplikace.

Soubory Web.config obsahují nastavení konfigurace platné pro celou aplikaci. Soubory Web.config jsou textovými soubory založenými na jazyku XML. To znamená, že k jejich vytvoření lze použít libovolný standardní textový editor nebo analyzátor jazyka XML a že mají srozumitelný formát. Pokud do kořenového adresáře aplikace neumístíte soubor Web.config, bude nastavení konfigurace určeno nastavením v souboru Machine.config, který je konfiguračním souborem pro celý server. Výchozí verze souboru Machine.config je nainstalována při instalaci platformy .NET Framework.

Řešení přístupových práv ve webové aplikaci v ASP.NET

K zabezpečení webových aplikací se nejčastěji používá způsob autentizace, který se v ASP.NET nazývá *Forms autentizace*. Jména a hesla uživatelů jsou uložena v rámci aplikace, zpravidla v relační databázi. Při přihlašování pak uživatel musí vyplnit formulář se jménem a heslem, na základě kterého se pak rozhoduje o jeho přihlášení. Pro správu uživatelů slouží služba Membership a pro správu rolí RoleManager.

Membership ASP.NET

- Řídí jména a hesla uživatelů
 - Deklarativní přes Web Site Admin Tool (součást Visual Studio .NET)
 - Programový přes Membership API
- Výrazně zjednodušuje formulářovou autentizaci
 - Poskytuje logiku pro validaci uživatelských jmen a hesel, zakládá účty atd.
 - Poskytuje datové úložiště pro ukládání uživatelských jmen a hesle, e-mail adres a dalších uživatelských dat
- Možné vytvořit vlastního „poskytovatele“ pro přístup k libovolnému zdroji dat

RoleManager ASP.NET

- Hotová bezpečnost postavená na rolích
 - Deklarativní přístup přes Web Site Admin Tool (součást Visual Studio .NET)
 - Programový přístup přes Roles API

- Zjednodušuje přidání bezpečnosti postavené na rolích do aplikací využívajících formulářovou autentizaci
 - Mapuje uživatele na role při každém dotazu
 - Poskytuje datové úložiště pro informace o rolích
- Možné vytvořit vlastního „poskytovatele“ pro přístup k libovolnému zdroji dat

Verze platformy ASP.NET

Verze platformy .NET Framework použitá aplikací ASP.NET je určena verzí souboru aspnet_isapi.dll, který je registrován prostřednictvím mapování skriptů služby IIS pro danou aplikaci. Mapování skriptů je trojstranné spojení přípony souboru (například ASPX), akce protokolu HTTP spojené s požadavkem (například GET) a knihovny DLL obsahující instrukce, které má webový server provést, přijme-li požadavek HTTP na soubor s danou příponou názvu. Mapování skriptů lze nastavit přímo pro kořenový adresář aplikace, který je virtuálním adresářem služby IIS, nebo lze povolit dědění mapování z nadřazených aplikací či webových serverů.

3.1.1.3 ASP.NET AJAX

AJAX se obvykle vykládá jako Asynchronous JavaScript and XML. Tento přístup se snaží od základů nabourat základní princip práce internetových serveru – tedy model request – response. Tradiční přístup je ten, že uživatel si vyžádá od serveru stránku – a to buď pomocí zadání internetové adresy, nebo nějakou akcí, kterou provede na jiné stránce (například kliknutím na tlačítko) a server stránku zašle. Během zpracování stažených dat a přípravě jejich zobrazení již může probíhat další získávání nových dat. JavaScript přes XMLHttpRequest požaduje po serveru nějaká data (obvykle XML), následně je za pomoci XSLT zpracuje a zobrazí v okně prohlížeče. Většina uživatelů si na to zvykla, ale zároveň vždy existovala místa, kde by se hodilo tento princip změnit. Například u aplikace kde máme velkou řadu ovládacích prvků na jedné stránce, právě v takové situaci by se hodilo, kdyby se obnovovala jen ta část stránky, respektive jen ten ovládací prvek, který je aktualizován.

AJAX je nástroj, který je spíše kombinací několika známých technologií, než novým počinem na poli webových technologií. Tyto technologie se vzájemně doplňují a dohromady působí velmi zajímavě. AJAX se skládá z těchto komponent:

- standardní prezentační vrstva založená na XHTML a CSS
- dynamické zobrazování a interakce za použití Document Object Model (DOM)
- výměna a manipulace s daty za použití XML a XSLT
- asynchronní výměna dat díky XMLHttpRequest
- a JavaScript, který všechny tyto komponenty spojuje v jeden fungující celek

AJAX je souhrné jméno pro celý princip – nechat provádět počítač klienta (jeho prohlížeč) asynchronní požadavky a naložit s nimi tak, aby nebylo potřeba obnovovat celou stránku. Jedinou možností, jak prohlížeč programovat, je JavaScript. A programování JavaScriptu je o dost jiné proti

ostatním programovacím jazykům – je to jazyk nepohodlný, jeho implementace se liší na prohlížečích, nedá se moc dobře ladit. Proto Microsoft vytvořil AJAX framework, který se snaží zjednodušit programování klientských skriptů. Díky AJAX frameworku nemusíte vůbec psát JavaScript – místo toho deklarujete třídy z frameworku způsobem velice podobným deklaraci serverových prvků ASP.NET. O interakci s klientem se postará ASP.NET a AJAX.

ASP.NET 2.0 AJAX Extensions

ASP.NET 2.0 AJAX Extensions je hlavní součástí AJAXu která rozšiřuje serverové ovládací prvky frameworku ASP.NET 2.0 a to tak že poskytuje vývojářům s AJAX programovacím modelem stejnou funkcionalitu a prostředí jako tradiční ASP.NET programování. Vývojář tak díky této technologii nemusí příliš znát javascript a vystačí si s tradičním ASP.NET programováním.

ASP.NET AJAX Control Toolkit

ASP.NET AJAX Control Toolkit doplňuje jádro ASP.NET AJAX Extensions poskytnutím řady klientských ovládacích prvků a tzv. extenderů rozšiřujících stávající serverové prvky ASP.NET 2.0 o novou AJAX funkcionalitu.

3.1.1.4 Jazyk C#

Jazyk C# je nově vyvinutý jazyk pro Microsoft .NET. Je navržen pro maximální využití této rychle se rozvíjející platformy. Jedná se o silně objektově orientovaný jazyk vycházející z programovacích jazyků Java a C++, takže pokud nějaký z těchto jazyků znáte, nebude pro Vás C# velký problém. Stejně jako tyto jazyky je i C# case-sensitive, což znamená, že významově odlišuje velká a malá písmena ve výrazech. V tomto jazyce je realizováno 80% základních knihoven .NET frameworku. I přesto, že je koncipován hlavně pro psaní řízeného kódu, na jehož užití je platforma .NET postavena, lze jej v případě potřeby využít i pro tvorbu kódu neřízeného. Použití neřízeného kódu znamená, že běhové prostředí CLR neověřuje zda-li je napsaný kód bezpečný (například se neověřuje jinak vyžadovaná typová bezpečnost).

Hlavní vlastností jazyka C#, které můžeme při tvorbě aplikací použít:

- Třídy – základní stavební prvek při tvorbě objektově orientovaných aplikací obsahující akce (metody) a atributy
- Struktury – lze je chápat jako zjednodušené třídy, jejich užitím jsou nejčastěji popisovány vlastní datové struktury.
- Výčtové typy
- Vlastnosti – někdy označované jako chytré proměnné
- Pole a jejich „chytrá“ verze nazývaná indexery
- Zástupci – typově bezpečné ukazatele na funkce
- Události – druh zástupců sloužící ke zpracování asynchronních operací

3.1.1.5 Microsoft SQL Server

System řízení báze dat MS SQL Server je natolik robustní a spolehlivý, že tvoří databázovou vrstvu (data management tier, zadní vrstvu nebo back-end) aplikací podnikových informačních systémů strategického významu. Jeho nasazení jako databáze pro interaktivní internetové aplikace je také běžné.

Výhodu, že MS SQL obslouží bez problému provozní informační systém firmy i její interaktivní webovou prezentaci, lze spatřovat v jednoduchém propojení těchto aplikací.

Verze a licencování produktu

Aktuální verze na trhu je MS SQL 2005. Ve větších firmách, a to především díky požadavkům některých vývojově starších aplikací, stále převažuje verze MS SQL 2000. Verzí SQL server existuje celá řada. Plná komerční verze nese název Enterprise Edition. Vedle toho existuje Personal Edition, která však umožňuje pouze 8 současných aktivních připojení. Plně funkční databázové jádro (někdy také databázový stroj) MS SQL serveru, zvané MSDE (Microsoft Data Engine), je určeno pro vývoj a testování a může se volně šířit jako součást aplikací vyvinutých ve Visual Studiu .NET. Existuje tu však jisté omezení výkonu MSDE. Není však žádný problém vyměnit jádro za plnohodnotnou verzi a při tom není nutné změnit ani řádek kódu.

Požadavky na operační systém a hardware

Databázová platforma MS SQL je dostupná jen pro operační systémy Microsoft Windows. Na těch bez technologie NT, tj. Windows 95 / 98 / 98 Second Edition / Millenium Edition můžete provozovat MS SQL Server Personal Edition nebo jádro MSDE. Celá instalace Enterprise verze možná není. Lze využít jen možnosti připojení klienta. Rychlostně postačí procesor Pentium a vyšší nebo ekvivalentní s alespoň 512MB RAM operační paměti.

Pro plné využití je potřeba Windows NT 4 se Service Packem 5 a vyšším, Windows 2000 Professional (Windows 2000 Server je pro víceuživatelský server výhodnější), nebo Windows XP Professional (pro verzi Server platí to samé), případně Windows 2003 Server.

Utility, které jsou součástí MS SQL serveru

Každá z verzí, s výjimkou MSDE, je dodávána s užitečnými aplikacemi s grafickým uživatelským rozhraním usnadňujícím vývojáři jeho práci a tím i život. Je to samozřejmě klientská konzolová aplikace pro práci s databázemi pomocí SQL příkazu. Model ovládání je jasný: nejprve se k serveru připojí a pak do políčka vepisují SQL příkazy a sledují výsledky. Dále jsou k dispozici rozhraní pro import a export dat, což se hodí při přechodu z jiného nebo na jiný systém. Aplikace pro správu databází umožňují správcům všechny úkoly administrace serveru - správu rolí, uživatelů a jejich přístupových práv, pravidla pro zálohování a údržbu dat atd.

3.2 Databáze

Jak již bylo řečeno výše, databáze je jádrem aplikace a představuje z velké části realizaci firemních procesů v podobě uložených procedur a pohledů. Tato koncepce byla z jisté části vyžadována a to zejména z důvodu budoucí podpory či modifikace aplikace. Jelikož IT oddělení výrobních firem, kromě helpdesků a administrátorů sítě, tvoří spíše databázový specialista než aplikační programáři je tento požadavek zcela logický.

Veškerá komunikace, která probíhá s databází je zajištěna pomocí uložených procedur, optimalizovaných maximálně tak, aby v případě výběrových dotazů, vrátily vždy všechna potřebná data v již požadované formě.

Příkazy Update a Insert pracující s více jak jedním záznamem jsou zapouzdřeny do transakcí, aby byla vždy zachována konzistence dat.

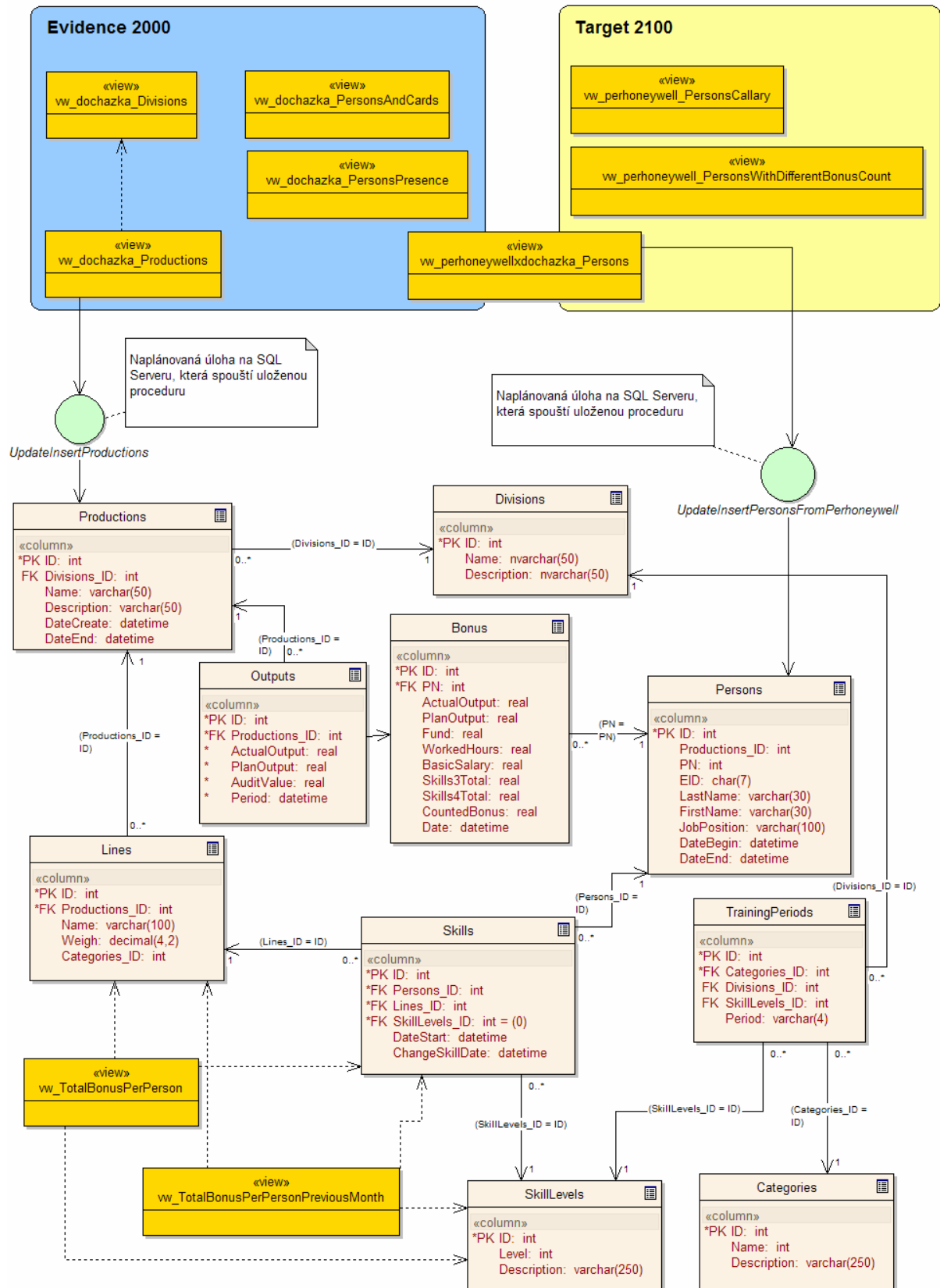
Klíčové asociace v jednotlivých tabulkách, které vznikají při jednotlivých operacích, jsou zajištěny pomocí triggerů.

Databáze představuje kromě samotného jádra aplikace také prostředek komunikace s dílčími systémy. Tato komunikace je zajištěna prostřednictvím pohledů uložených v databázi SkillMatrix, definujících která data je potřeba synchronizovat. Samotná synchronizace je následně provedena pomocí uložených procedur, které jsou navrženy jako naplánované úlohy na SQL Serveru.

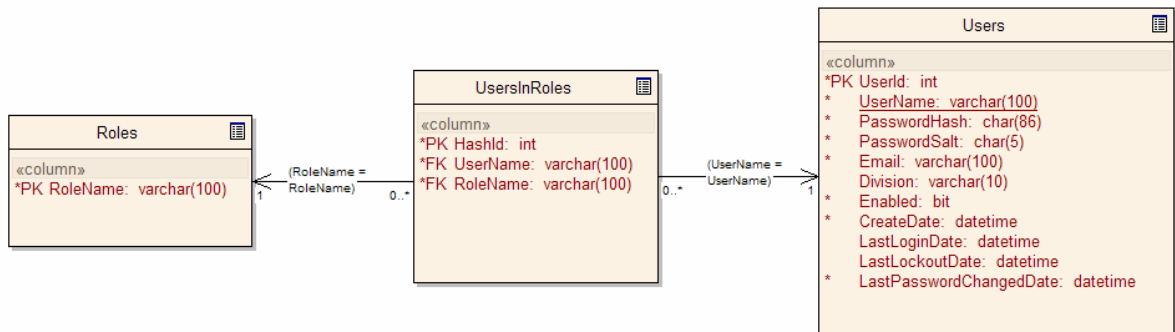
3.2.1 Diagram schématu databáze

Viz Obrázek 6: Diagram schématu DB hlavní části, Obrázek 7: Diagram schématu DB přístupových práv, Obrázek 8 Diagram schématu DB samostatných tabulek

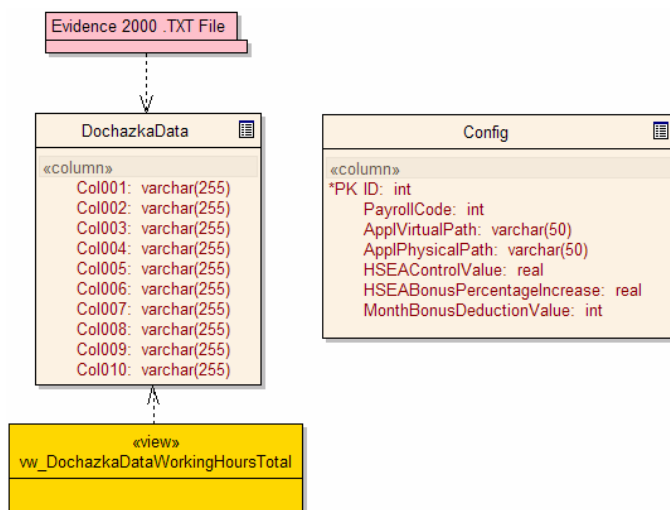
Obrázek 6: Diagram schématu DB hlavní části



Obrázek 7: Diagram schématu DB přístupových práv



Obrázek 8: Diagram schématu DB samostatných tabulek



Popis tabulek

Tabulka Divisions

Eviduje informace o divizích. Udává vrchol organizační struktury, který byl převzat z pohledu vw_dochazka_divisions.

Tabulka Productions

Eviduje informace o výroбах dle divizí, včetně data vzniku a data zániku. Tyto data udávají platnost výrob a slouží k uchování historie. Údaje v této tabulce mají původ v docházkovém systému a jsou aktualizovány skrze uloženou proceduru UpdInsProductions z pohledu vw_dochazka_productions. Aktualizací se myslí modifikace stávajících a vkládání nových dat.

Tabulka Lines

Eviduje informace o pracovištích na jednotlivých výroбах. Definuje příslušné váhy pracovišť a přiřazené kategorie délek zpracování

Tabulka Categories

Eviduje kategorie délek zapracování.

Tabulka TrainingPeriods

Eviduje jednotlivé délky zapracování dle kategorie, výše skill a divize.

Tabulka Persons

Eviduje informace o osobách, včetně data nástupu a data výstupu a přiřazení na danou výrobu. Tyto informace jsou aktualizovány skrze uloženou proceduru UpdInsPersonsFromPerhoneywell z pohledu vw_dochazkaxperhoneywell_persons.

Tabulka Skills

Eviduje informace o schopnostech respektive skills osob na jednotlivých pracovištích.

Tabulka SkillLevels

Eviduje úrovně skills.

Tabulka Outputs

Eviduje informace o výstupech z jednotlivých výroby za jednotlivá období.

Tabulka Bonus

Eviduje informace o spočítaných bonusech osob, stejně tak o vstupních hodnotách, ze kterých je bonus osobám počítán. Obsahuje hodnoty výstupů stejně jako tabulka Outputs. Jedná se však o individuální hodnoty, které jsou použity přednostně, jsou-li nastaveny.

Tabulka Users

Eviduje informace o uživateli.

Tabulka Roles

Eviduje informace o rolích.

Tabulka UsersInRoles

Eviduje informace o přiřazení uživatelů do jednotlivých rolí

Tabulka DochazkaData

Eviduje data naimportovaná z textového souboru z docházkového systému. Z tabulky Dochazka data jsou po importu vybrána konkrétní data definovaná v pohledu vw_DochazkaData, která jsou použita při výpočtu bonusů.

Tabulka Config

Eviduje nastavení aplikace a některé statické hodnoty použité při výpočtu bonusů. Tyto hodnoty jsou umístěny do tabulky Config zejména proto, že je zde jistý předpoklad jejich změny v budoucnosti a právě díky tomuto návrhu je není nutné měnit v několika uložených procedurách, ale pouze v tabulce Config.

3.2.2 Uložené procedury

3.2.2.1 Procedury výběru dat

GetClosedBonusPeriods ()

Bez vstupních parametrů, vrací seznam uzavřených bonusových období.

GetDivisions ()

Bez vstupních parametrů, vrací seznam divizí.

GetHistoryMonthBonusByProductionByPeriod (@ProductionID int, @Period datetime)

Na základě parametrů, id výroby a období, vrací seznam lidí a spočítaných bonusů za zvolené uzavřené bonusové období. Přičemž, jelikož se jedná již o spočítaný bonus, respektive uzavřené období, je informace o výši spočítaného bonusu brána ze sloupce CountedBonus.

GetLines ()

Bez vstupních parametrů, vrací seznam pracovišť.

GetLinesByProductions (@ProductionID int)

Rozšiřuje proceduru GetLines o vstupní parametr id výroby.

GetMonthBonusByPeriod (@Period datetime)

Spojuje informace z tabulek persons, bonus, productions, divisions, outputs a pohledu vw_perhoneywell_PersonsWithDifferentBonusCount (definuje osoby, které mohou mít skill 4). Na základě parametru období vrací spočítané bonusy v aktuálně otevřeném bonusovém období ve formě:

- BonusID – id bonusu
- PersonNumber – osobní číslo osoby
- PersonName – celé jméno osoby
- DivisionName – Název divize
- ProductionName – Název výroby na kterou osoba patří
- BasicSalary – Základní mzda pro dané období
- MonthBonus – Výše měsíčního bonusu v Kč

- BonusPercentage – Procento měsíčního bonusu ze základní mzdy v %
- Output – Celkový spočítaný výstup
- Fund – Fond pracovní doby za dané období
- WorkedHours – Celkový počet odpracovaných hodin.
- HSEA – Spočítaná hodnota HSEA
- Skills – Celková suma vah jednotlivých skills v % za minulý měsíc, respektive aktuální bonusové období.
- BonusCountType – Nabývá hodnot “Operator” a “InternalTrainer”, kde “InternalTrainer” definuje skupinu osob s možností získat skill 4, které jsou v aplikaci zvýrazněny.

GetMonthBonusByDivisionByPeriod (@DivisionID int, @Period datetime)

Rozšiřuje proceduru GetMonthBonusByPeriod v podobě vstupních parametrů na id divize a období.

GetMonthBonusByProductionByPeriod (@ProductionID int, @Period datetime)

Rozšiřuje proceduru GetMonthBonusByPeriod v podobě vstupních parametrů na id výroby a období.

GetMonthBonusReportForPayrollByPeriod (@Period datetime)

Report spočítaných bonusů v požadovaném formátu pro import do mzdového systému. Jedná se o osobní číslo osoby, číslo mzdové složky na kterou má být bonus vyplacen (ta je brána z tabulky Config), výše bonusu, jméno osoby a EID (Employee Identification number). Hodnota spočítaného bonusu je brána ze sloupce CountedBonus z tabulky Bonus.

GetOpenedBonusPeriods ()

Bez vstupních parametrů, vrací seznam otevřených bonusových období.

GetOutputsByProductionByPeriod (@ProductionID int, @Period datetime)

Na základě parametrů, id výroby a období, vrací nastavené výstupy (plánovaný a skutečný výstup) a hodnotu auditu z tabulky Outputs pro danou výrobu ve zvoleném období.

GetPersons ()

Bez vstupních parametrů, vrací seznam osob. Spojuje informace z tabulek persons a productions a pohledů vw_TotalBonusPerPerson (udává sumy vah za skills zaměstnance), vw_dochazka_PersonsPresence (udává údaje o aktuální přítomnosti osob v práci), vw_perhoneywell_PersonsWithDifferentBonusCount (definuje osoby, které mohou mít skill 4).

GetPersonsByLineBySkillLevelByStatus (@LineID int, @SkilllevelID int, @StatusID int)

Report, který na základě stanovení parametrů vyhledá osoby přítomné či nepřítomné v práci s požadovaným skill na daném pracovišti. Aktuální presence osob je získána z pohledu vw_dochazka_PersonsPresence.

GetPersonsByPersonName (@PersonName varchar)

Rozšiřuje proceduru GetPersons v podobě vstupních parametrů na jméno osoby.

GetPersonsByProduction (@ProductionID int)

Rozšiřuje proceduru GetPersons v podobě vstupních parametrů na id výroby.

GetPersonsDetailsByCardNumber (@CardID varchar)

Na základě parametru, kterým je číslo karty si z pohledu vw_dochazka_PersonsAndCards zjistí osobní číslo a dle něj zobrazí podrobnosti o osobě jako je fotografie, výroba, jméno, osobní číslo, suma vah na všech pracovištích a také přehled skills dané osoby.

GetPersonsDetailsByPerson (@PersonID int)

Na základě parametru id osoby, zobrazí podrobnosti o osobě jako je fotografie, výroba, jméno, osobní číslo, suma vah na všech pracovištích a také přehled skills dané osoby.

GetPersonsSkillsByCardNumber

Na základě parametru, kterým je číslo karty si z pohledu vw_dochazka_PersonsAndCards zjistí osobní číslo a dle něj zobrazí podrobnosti o skill osoby na jednotlivých pracovištích s informacemi o délkách zapracování.

GetProductions ()

Bez vstupních parametrů vrátí všechny výroby.

GetProductionsByDivision (@DivisionsID int)

Na základě id divize vypíše všechny výroby patřící pod danou divizi.

GetSkillDetailsBySkillID (@SkillID int)

Zobrazí podrobnosti o daném skill na základě id skill. Jedná se o výši aktuálního skill, datum jeho přiřazení, délku zapracování na vyšší skill, hodnotu vyššího skill (respektive skill, který osoba může získat po proběhlé délce zapracování) a datum možné změny na vyšší skill (dopočítaná hodnota ze sloupce ChangeSkillDate vzhledem k aktuálnímu datu).

GetSkillsByPerson (@PersonID int)

Na základě parametru id osoby vrátí skills dané osoby na pracovištích, kde již započala délka zpracování a váhu příslušnou danému pracovišti, pokud skill na daném pracovišti překročil hodnotu 3.

GetSkillsByPersonByProduction (@PersonID int, @ProductionsID int)

Spojuje informace z tabulek skills, skilllevels, lines, categories a trainingperiods a na základě vstupních parametrů zobrazí skills osoby na jednotlivých pracovištích s informacemi o započatých, dobehých a probíhajících délkách zpracování a příslušné váhy pokud hodnota skill překročila hodnotu 3.

GetTotalSkillsByPerson (@PersonID int)

Zobrazí celkovou sumu vah dle skills osoby.

GetDefaultViewMatrixByProductionID

Report ve formě pivot tabulky osob a jim příslušných skills na jednotlivých pracovištích dané výroby. Nejprve vytvoří dočasnou tabulku, vloží do ní informace o osobách, následně pomocí kurzoru prochází tabulku lines a dynamickými dotazy vytváří nejprve nové sloupce tabulky v podobě názvů pracovišť a pak do nich přiřazuje odpovídající hodnoty skills.

3.2.2.2 Procedurey modifikace a vkládání dat

InsDataIntoBonusTable

Sloučí data z pohledů vw_perhoneywell_PersonsCallary (aktuální základní mzda osoby), vw_DochazkaDataWorkingHoursTotal (počet odpracovaných hodin osoby), vw_TotalBonusPerPersonPreviousMonth (sumu vah z skills s platností k předešlému měsíci, tedy měsíci za který je bonus počítán), a tabulky Persons a vloží požadovaná data pro výpočet bonusů, za dané období, do tabulky Bonus.

InsFileIntoDochazkaDataTable

Provede bulk insert z textového souboru (vyexportovaného z docházkového systému) do tabulky DochazkaData.

InsLine (@ProductionID int, @LineName varchar, @Weigh decimal, @CategoryID int)

Na základě zadání výroby, na kterou má pracoviště patřit, nazvu pracoviště, váhy a kategorie délek zpracování vloží nové pracoviště na danou výrobu.

UpdCloseBonusPeriod (@Period)

Uzavře bonusové období dle předaného parametru, provede update do tabulky bonus, konkrétně sloupce CountedBonus, do kterého uloží již spočítané bonusy pro dané období. Tímto se uzavře, respektive schválí bonusové období a není ho možné dále editovat. Zároveň umožní export tohoto období pro Mzdové centrum. Přesouvá se do viditelnosti v historii.

Celý proces aktualizace je při tom zaimplementován do transakce.

UpdInsPersonsFromPerhoneywell

Provede aktualizaci stávajících, případně vložení nových osob z pohledu vw_perhoneywellxdochazka_Persons do tabulky Persons. Aktualizovány jsou údaje o jménu, příjmení, datu výstupu a přiřazení k výrobě. Vkládány jsou pak všechny sloupce tabulky.

Celý proces importu a aktualizace je při tom zaimplementován do transakce.

UpdInsProductionsFromDochazka

Provede aktualizaci stávajících, případně vložení nových výrob z pohledu vw_dochazka_Productions do tabulky Productions. Aktualizovány jsou údaje o názvu výroby a datu platnosti.

Celý proces importu a aktualizace je při tom zaimplementován do transakce.

UpdLines (@LineID int, @LineName varchar, @Weigh decimal, @CategoryID int)

Aktualizuje vlastnosti daného pracoviště jako je název pracoviště, váha a kategorie délek zapracování.

UpdOutputs (@OutputsID int, @ActualOutput real, @PlanOutput real, @AuditValue real)

Aktualizuje, nastaví výstupy dané výroby na základě parametrů.

UpdSetSkill

Aktualizuje hodnotu skill v tabulce skills. Pokud se jedná o započítání zapracování, tedy přiřazení osoby na dané pracoviště, je vloženo datum započítání a datum změny skill je nastaveno na identickou hodnotu a od něj se počítá příslušná délka zapracování. Pokud se jedná o aktualizaci skill je změněna hodnota skill a datum změny skill z kterého se následně počítá nová délka zapracování platná pro danou výši skill na daném pracovišti dle kategorie délek zapracování.

3.2.3 Triggery

Pomocí triggerů je zajištěn vznik klíčových asociací v tabulkách při modifikaci a vkládání dat. Triggery jsou použity pro zjednodušení operací importu dat ze souboru z docházkové systému (respektive z tabulky DochazkaData), dále pak při procesu aktualizace osob a v neposlední řadě při vložení nového pracoviště.

trgBonus_I

Trigger nad tabulkou Bonus, který se provede při vložení nového záznamu. Vytváří asociace v tabulce Outputs pro příslušné bonusové období a nastavuje hodnoty jednotlivých sloupců na defaultní hodnoty.

trgLines_I

Trigger nad tabulkou Lines, který se provede při vložení nového záznamu. Vytváří asociace nového pracoviště v tabulce skills se všemi osobami a nastavuje defaultní hodnotu.

trgPersons_I

Trigger nad tabulkou Persons, který se provede při vložení nového záznamu. Vytváří asociace nové osoby v tabulce skills se všemi pracovišti a nastavuje defaultní hodnotu.

3.2.4 Pohledy

Pohledy představují definici dat získávaných z jednotlivých personálních systémů a některá sumární data podstatná pro výpočet bonusů.

vw_dochazka_Divisions

Pohled nad databází Docházkového systému, tabulkou ggroups. Definuje vrchol organizační struktury, tedy divize, které se mají přenést do SkillMatrixu.

vw_dochazka_PersonsAndCards

Pohled nad databází Docházkového systému, tabulkami gcards, gpersons a gtie. Poskytuje informace o osobách a jim přiřazených identifikačních kartách a jejich kódech.

vw_dochazka_PersonsPresence

Pohled nad databází Docházkového systému, tabulkou gpersons. Poskytuje informace o aktuální přítomnosti, respektive nepřítomnosti osob v práci.

vw_dochazka_Productions

Pohled nad databází Docházkového systému, tabulkou ggroups. Definuje výroby, které budou dostupné v databázi SkillMatrix, dle vrcholu organizační struktury v pohledu vw_dochazka_Divisions.

vw_dochazkaDataWorkingHoursTotal

Pohled nad tabulkou DochazkaData v databázi SkillMatrix. Definuje mzdové složky s údaji o odpracovaných hodinách, které mají být použity pro výpočet bonusů.

vw_perhoneywell_PersonsCellary

Pohled nad databází Mzdového systému, tabulkou person. Poskytuje informace o aktuálních mzdách zaměstnanců.

V tabulce person jsou přitom vždy, k datu výpočtu bonusů, mzdy zaměstnanců platné k minulému měsíci. Aktuální mzda je v tabulce persons až po uzavření mzdového období, což je vždy až po uzavření bonusového období.

vw_perhoneywell_PersonsWithDifferentBonusCount

Pohled nad databází Mzdového systému, tabulkou persondef. Definuje osoby, které mohou mít skill 4 a tedy získat bonus i v případě že není vyplacen bonus pro skill 3 tedy jsou zadány nulové výstupy.

vw_perhoneywellxdochazka_Persons

Pohled nad databází Mzdového a Docházkového systému, tabulkami person, persondef a ggroups, gpersons a gtie. Definuje osoby určené pro SkillMatrix, jejich organizační zařazení na výrobu, datum nástupu, datum výstupu a další důležité informace.

vw_totalBonusPerPerson

Pohled nad tabulkami Skills, Lines, Persons a SkillLevels v databázi SkillMatrix. Poskytuje informace o sumách vah za skill 3 a skill 4 k aktuálnímu datu.

vw_totalBonusPerPersonPreviousMonth

Pohled nad tabulkami Skills, Lines, Persons a SkillLevels v databázi SkillMatrix. Poskytuje informace o sumách vah za skill 3 a skill 4 aktuálních k datu minulého měsíce. Tedy vždy měsíce, pro který je počítán bonus.

3.2.5 Komunikace s dílčími systémy

Veškerá data získávaná ze systémů Target 2100 a Evidence 2000 jsou definována jako pohledy uložené v databázi SkillMatrix. Použití dat je pak v moci uložených procedur.

Samotná aktualizace je vidět na Data diagramu, jedná se o:

Aktualizace výrob

Aktualizace dat probíhá na základě spuštění uložené procedury UpdateInsertProductions, která je naplánována jako naplánovaná úloha na SQL Serveru. Popis procedury viz podkapitola 4.1.2.2 UpdInsProductionsFromDochazka.

Aktualizace osob

Aktualizace dat probíhá na základě spuštění uložené procedury UpdateInsertPersons, která je naplánována jako naplánovaná úloha na SQL Serveru. Popis procedury viz podkapitola 4.1.2.2 UpdInsPersonsFromPerhoneywell.

3.3 Aplikace

3.3.1 Adresářová struktura

Adresáře definované ASP.NET

Samotný ASP.NET vymezuje některé názvy adresářů pro své použití. Jedná se o tyto adresáře:

- App_Browsers - umístění souborů .browser, využití pro adaptivní rendering (různé vykreslování pro různé prohlížeče)
- App_Code – sdílené třídy pro celou aplikaci, obsahuje jednotlivé objekty databázové komponenty a adaptory pro korektní renderování obrázků
- App_Data – nevyužito, používá se pro uložení souborů databáze, v našem případě je databáze přímo na SQL serveru
- App_Themes - adresář obsahující jednotlivé skiny, ikonky, css a obrázky
- Bin - zkompilevané assembly (AJAX, Membership Provider, DataAccessHelper)

Standardní adresáře

Adresáře tvořící jednotlivé moduly systému

Obsahují .aspx a .cs stránky reprezentující jednotlivé operace uvnitř modulů. Každý adresář obsahuje navíc soubor web.config, který definuje uživatelské role mající přístup do daného modulu.

- Bonus
- BonusHistory
- Home
- Persons
- Reports
- Skills
- Users
- Workplaces

Specifické adresáře

- Files – soubory ze systému Evidence 2000 importované do aplikace

- Kiosk – speciální adresář, který využívá jiný princip oprávnění, jedná se o osobní přístup, umožňuje přihlášení identifikační kartou a zobrazení osobních informací včetně dosažených skills

Soubory v kořenovém adresáři

Login.aspx

Přihlašovací stránka. Přístup anonymous.

Site.master

Definuje rozložení webové aplikace na nadpis, navigaci a obsah.

Web.config

Soubor obsahuje hlavní a centrální nastavení aplikace. Je rozdělen do několika částí:

- configSections – obsahuje nezbytná nastavení pro využití technologie ASP.NET AJAX
- appSettings – obsahuje ConnectionString, definující připojení k databázi
- system.web – obsahuje nastavení navigace (SiteMap), způsobu zabezpečení a přístupových oprávnění, včetně uživatelských rolí (Authentication, roleManager, membership), jednotlivé assemblies (pro využití technologie AJAX), httpHandlers, httpModuls (pro využití technologie AJAX)

Web.Sitemap

Mapuje jednotlivé aspx stránky do jednotlivých objektů navigace.

3.3.2 Databázová komponenta

Databázová komponenta představuje zapouzdření jednotlivých uložených procedur, v podobě metod, do logických celků (tříd) jež jsou implementovány s pomocí knihovny DataAccessHelper. Jednotlivé metody inicializují spojení s databází a za použití uložených databázových procedur provádějí operace nad databází (čtení dat, vkládání, editace, mazání). Jednotlivé třídy a metody jsou vidět na obrázku 9.

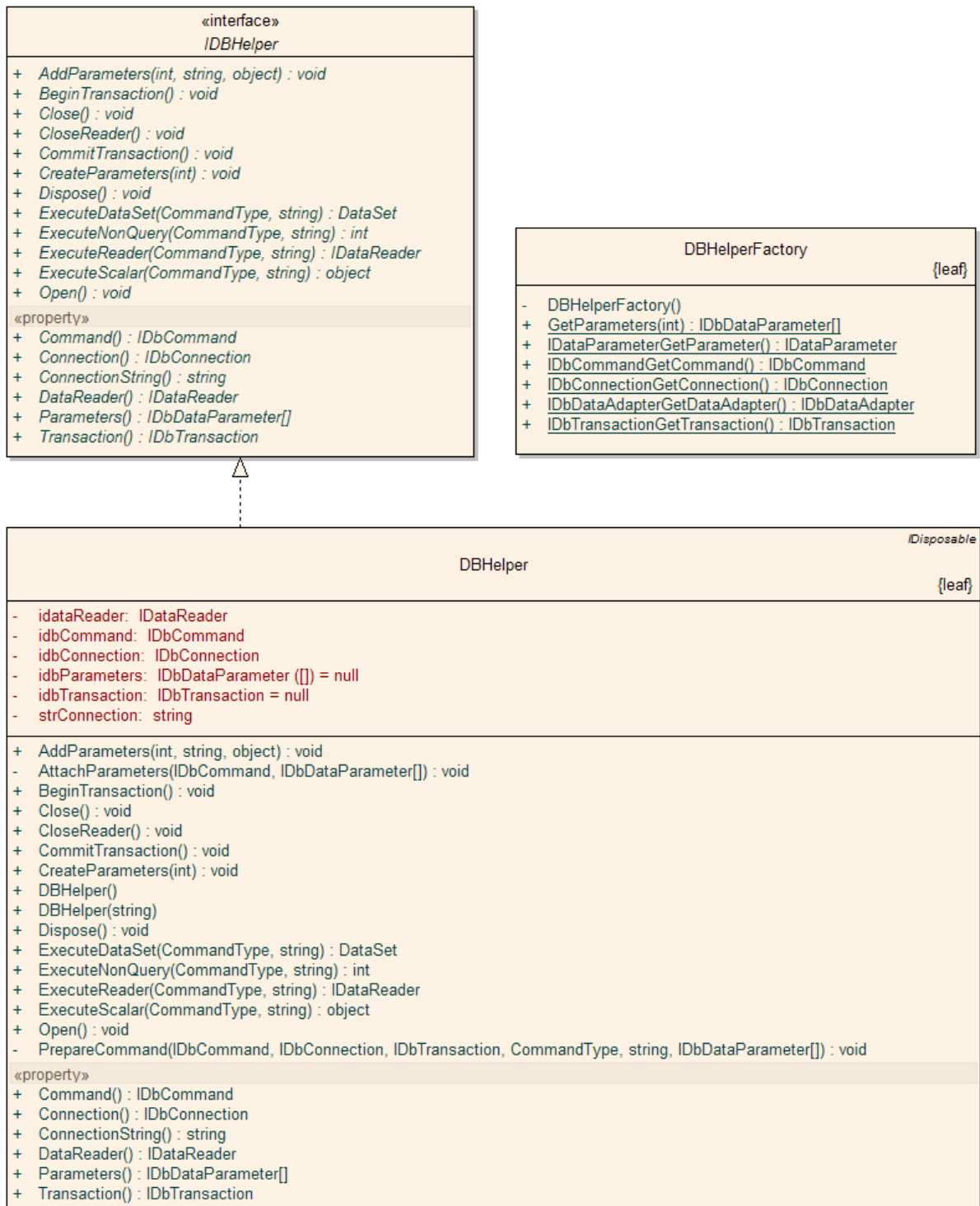
Obrázek 9: Diagram tříd databázové komponenty



3.3.2.1 DataAccessHelper

DataAccessHelper je knihovna napsaná v jazyce Visual.NET C# obsahující sérii ADO.NET tříd a rozhraní sloužících pro připojení k databázi SQL Serveru. Implementuje jednotlivé metody pro vytvoření připojení, uzavření připojení, provedení dotazu či uložené procedury, a to jak prostřednictvím metody DataReader tak DataSet. Jednotlivé třídy a rozhraní jsou vidět na obrázku 10.

Obrázek 10: Diagram tříd DataAccessHelper



DBHelper

Implementuje jednotlivé metody založené na IDbHelper a využívá statické metody z DBHelperFactory .

Open ()

Vytvoří spojení dle ConnectionStringu a inicializuje prázdný command.

Close ()

Uzavře spojení

BeginTransaction ()

Nastaví, že se daný command bude provádět v transakci

CommitTransaction ()

Potvrdí provedené změny v databázi (Commit), v případě chyby vrátí změny v databázi do původního stavu (Rollback)

CreateParameters (int paramsCount)

Vytvoří požadovaný počet parametrů

AddParameters (int index, string paramName, object objValue)

Přiřadí hodnotu do vytvořeného parametru

PrepareCommand (IDbCommand command, IDbConnection connection, IDbTransaction transaction, CommandType commandType, string commandText, IDbDataParameter[]commandParameters)

Provede inicializaci příkazu (command) s daným připojením, typem příkazu, textem příkazu a požadovanými parametry

AttachParameters (IDbCommand command, IDbDataParameter[]commandParameters)

Přiřadí parametr do daného příkazu.

ExecuteReader (CommandType commandType, string commandText)

Metoda čtení dat. Využívá metodu PrepareCommand, provede daný příkaz a vrátí přečtený záznam.

ExecuteDataSet(CommandType commandType, string commandText)

Metoda čtení dat. Využívá metodu PrepareCommand, provede daný příkaz a vrátí celou množinu výsledků v podobě datasetu.

Inicializaci spojení, nastavení parametrů SQL příkazu, a rozběhnutí transakce

ExecuteScalar(CommandType commandType, string commandText)

Metoda čtení dat. Využívá metodu PrepareCommand, provede daný příkaz a vrátí první záznam z přečtené množiny záznamů.

ExecuteNonQuery(CommandType commandType, string commandText)

Využívá metodu PrepareCommand, provede daný příkaz a vrátí počet aktualizovaných případně vložených záznamů.

DBHelperFactory

Implementuje statické metody pro práci s SQL Serverem.

IDBHelper

Rozhraní, jež obsahuje definice jednotlivých metod implementovaných pomocí třídy DBHelper.

3.3.3 Přístupová oprávnění

Přístupová oprávnění do aplikace zajišťuje komponenta SqlWebProviders, která zjednodušuje a přepisuje vestavěné, velmi obecné API pro přihlašování v ASP.NET, především pak infrastrukturu pro přihlašování (Membership Provider) a přiřazování uživatelů do rolí (Role provider). Jednotlivé metody jsou přepsány (při zachování standardní funkcionality) tak, aby umožňovali komunikovat s databázovým modelem SkillMatrix.

3.3.3.1 SqlWebProviders

Na obrázku 11 je vidět diagram tříd přepsaného Membership a Role API.

SqlMembershipProvider

Přepisuje metody pro správu uživatelského účtu. Jedná se především o vytvoření účtu, reset hesla, smazání účtu a další.

SqlRoleProvider

Přepisuje metody pro správu rolí. Jedná se o vytvoření role, přiřazení účtu do role, smazání role a další.

Obrázek 11: Diagram tříd SqlWebProviders



3.3.3.2 Nastavení aplikace pro využití komponenty SqlWebProviders

K tomu, aby aplikace využívala SqlWebProviders, musí být v souboru web.config nedefinován způsob autentizace na “Forms” (tedy formulářovou autentizaci) a dále providery definující role a informace o uživateli v podobě “CustomizedRoleProvider“ a “CustomizedMembershipProvider“. Nastavení souboru web.config:

```

<!-- Nastavení způsobu autentizace na Forms-->
  <authentication mode="Forms" />
<!-- Zapnutí používání rolí -->
  <roleManager enabled="true"
    defaultProvider="CustomizedRoleProvider">
    <providers>
<!-- Nastavení RoleProvideru pro použití námy definovaného
SqlRoleProvideru z knihovny SqlWebProviders -->
      <add name="CustomizedRoleProvider"
        type="SqlWebProviders.SqlRoleProvider, SqlWebProviders"
        connectionString="SkillMatrix" />
    </providers>
  </roleManager>

```

```

<!-- Nastavení MembershipProvideru pro použití námy definovaného
SqlMembershipProvider z knihovny SqlWebProviders -->
<membership defaultProvider="CustomizedMembershipProvider">
  <providers>
    <add name="CustomizedMembershipProvider"
        type="SqlWebProviders.SqlMembershipProvider, SqlWebProviders"
        connectionStringName="SkillMatrix"
        requiresQuestionAndAnswer="false"
        requiresUniqueEmail="false"
        minRequiredPasswordLength="7"
        />
  </providers>
</membership>

```

3.3.4 Uživatelské rozhraní

Uživatelské rozhraní je vytvořeno v ASP.NET za využití technologie ASP.NET AJAX a specifické operace jsou doprogramovány v C# kódu. Rozhraní váže jednotlivé objekty z databázové komponenty na prvek ObjectDataSource, který je navázán jako zdroj dat na daný ovládací prvek.

Vázání dat v aspx stránce z databázové komponenty

Vázání dat je provedeno prostřednictvím prvku ObjectDataSource, který vytváří deklarativní spoj mezi ovládacími prvky webové stránky a databázovou komponentou.

Implementace technologie AJAX

Aby bylo možné využít technologii AJAX, včetně ovládacích prvků, v ASP.NET stránkách je třeba vložit do aplikace následující binární soubory:

- AjaxControlToolkit.dll
- AJAXExtensionsToolbox.dll
- System.Web.Extensions.Design.dll
- System.Web.Extensions.dll

Dále je třeba na každou stránku kde používáme AJAX vložit ovládací prvek „ScriptManager“.

Pokud chceme dosáhnout toho, aby aktualizace nějakého ovládacího prvku nevyvolala Postback celé stránky, začleníme tento ovládací prvek do prvku “UpdatePanel” z AJAX Control Toolkitu.

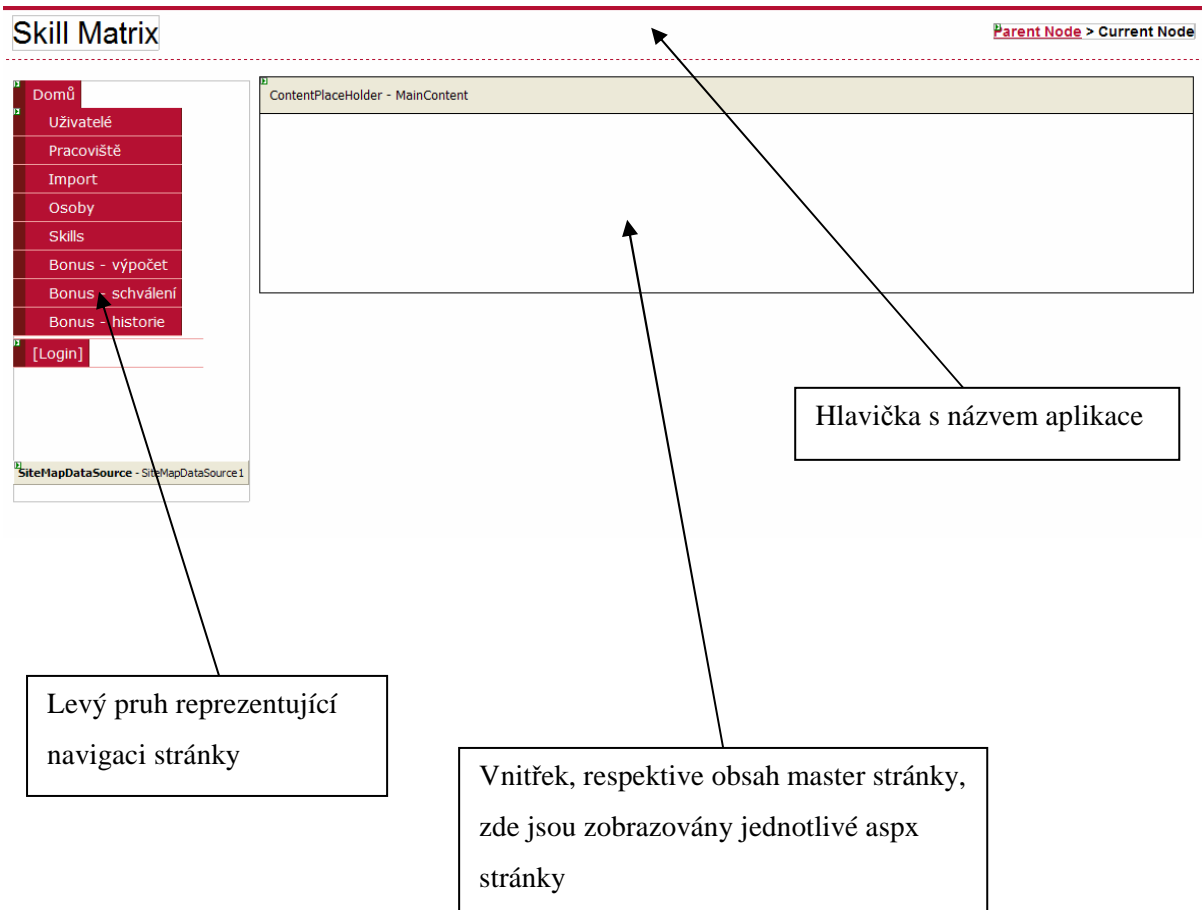
Pokud chceme nějaký ovládací prvek rozšířit o funkcionalitu z AJAX Extensions Toolboxu, je třeba na stránku vložit tento rozšiřující ovládací prvek a nastavit mu ID ovládacího prvku, který má rozšiřovat. Jakmile je toto nastaveno je příslušný prvek rozšířen o sérii nastavení umožňujících AJAX funkcionalitu.

Pokud však chceme ovládacímu prvku, rozšířeném o AJAX funkcionalitu předat nějakou hodnotu například z ovládacího prvku, který jej vyvolal, je to třeba ošetřit kódově.

Rozvržení uživatelského rozhraní

Rozložení uživatelského rozhraní je implementováno pomocí master pages. Master stránka představuje šablonu určující vzhled stránek, jež ji používají. Definuje hlavičku a levý pruh webové stránky. Jednotlivé aspx stránky pak doplňují “vnitřek“ master stránky k vnějším oblastem definovaných masterem. Šablona webové stránky v podobě master page je zobrazena na obrázku 12.

Obrázek 12: Master Page Skillmatrix



4 Ukázka použití

Pro ukázky použití jsou z důvodu ochrany osobních údajů použity fiktivní osoby a mzdy.

Přihlášení do aplikace: Titulní obrazovka

Před vstupem do systému je nezbytné se přihlásit.

Skill Matrix



4.1 Výpočet bonusů

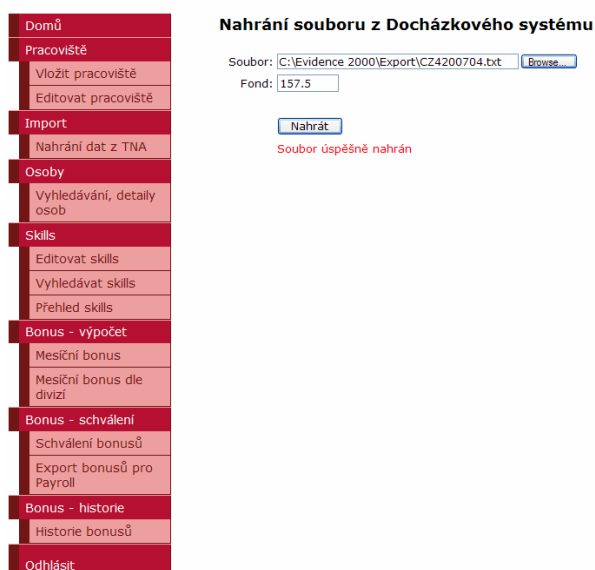
Ukázka znázorňuje celý proces výpočtu bonusů a podíl na tomto procesu z hlediska jednotlivých zodpovědností. Viz podkapitola 2.3.1 Diagram aktivity výpočtu bonusů.

Přihlášení Personalista: Modul Import > Nahrání dat

Nahrání souboru z docházkového systému (vytvoření bonusového období).

Skill Matrix

Import > Nahrání dat z TNA



Přihlášení OPS / Cell Leader: Modul Bonus – výpočet > Měsíční bonus

Zadání výstupů pro jednotlivé výroby v otevřeném bonusovém období.

Skill Matrix

Bonus - výpočet > Měsíční bonus

Domů

Osoby

Vyhledávání, detaily osob

Skills

Editovat skills

Vyhledávat skills

Přehled skills

Bonus - výpočet

Měsíční bonus

Měsíční bonus dle divízi

Bonus - historie

Histore bonusů

Odhlásit

Vypočet měsíčního bonusu

Divize:

Výroba:

Mzdové období:

Výstupy výroby

Skutečný výstup: 22354

Plánovaný výstup: 22786

Hodnota auditu: 7

Select	Pozice	ČC	Osoba	Divize	Výroba	Základní mzda	Měsíční bonus	Bonus%	Výstup	Fond	Odpracované hodiny	HSEA	Skills
		40925	Xxxxxxxxx Jarmila	CC Valves	Operator	12450 Kč	1956 Kč	16	0,98	157,5	157,5	0,36	7,25%
		40936	Xxxxxxxxx Jana	CC Valves	Operator	12450 Kč	1863 Kč	15	0,98	157,5	157,5	0,36	6,5%
		41024	Xxxxxxxxx Radmila	CC Valves	Operator	12450 Kč	1582 Kč	13	0,98	157,5	157,5	0,36	4,25%
		40727	Xxxxxxxxx Tomáš	CC Valves	Operator	12450 Kč	2392 Kč	19	0,98	157,5	157,5	0,36	10,75%
		40889	Xxxxxxxxx Martin	CC Valves	Operator	12450 Kč	1707 Kč	14	0,98	157,5	157,5	0,36	5,25%
		40924	Xxxxxxxxx Alfred	CC Valves	Operator	12450 Kč	1520 Kč	12	0,98	157,5	157,5	0,36	3,75%
		40570	Xxxxxxxxx Radko	CC Valves	Operator	12450 Kč	1894 Kč	15	0,98	157,5	157,5	0,36	6,75%
		40573	Xxxxxxxxx Gabriela	CC Valves	Operator	12450 Kč	1551 Kč	12	0,98	157,5	157,5	0,36	4%
		40641	Xxxxxxxxx Květoslav	CC Valves	Operator	12450 Kč	1551 Kč	12	0,98	157,5	157,5	0,36	4%
		40560	Xxxxxxxxx Eliška	CC Valves	Operator	12450 Kč	1738 Kč	14	0,98	157,5	157,5	0,36	5,5%
		40561	Xxxxxxxxx Ivo	CC Valves	Operator	12450 Kč	1769 Kč	14	0,98	157,5	157,5	0,36	5,75%
		40567	Xxxxxxxxx Kateřina	CC Valves	Operator	12450 Kč	2018 Kč	16	0,98	157,5	157,5	0,36	7,75%
		40113	Xxxxxxxxx Marie	CC Valves	Operator	12450 Kč	0 Kč	0	0,98	157,5	75	0,36	2%
		40380	Xxxxxxxxx Stanislava	CC Valves	Operator	12450 Kč	4384 Kč	35	0,98	157,5	157,5	0,36	26,75%
		40557	Xxxxxxxxx Libor	CC Valves	Operator	12450 Kč	2018 Kč	16	0,98	157,5	157,5	0,36	7,75%

Přihlášení OPS / Cell Leader: Modul Bonus – výpočet > Měsíční bonus

Zadání individuálních výstupů pro vybrané osoby v otevřeném bonusovém období (Modální okno s využitím technologie AJAX)

Domů

Osoby

Vyhledávání, detaily osob

Skills

Editovat skills

Vyhledávat skills

Přehled skills

Bonus - výpočet

Měsíční bonus

Měsíční bonus dle divízi

Bonus - historie

Histore bonusů

Odhlásit

Vypočet měsíčního bonusu

Divize:

Výroba:

Mzdové období:

Výstupy výroby

Skutečný výstup: 22354

Plánovaný výstup: 22786

Hodnota auditu: 7

Select	Pozice	ČC	Osoba	Divize	Výroba	Základní mzda	Měsíční bonus	Bonus%	Výstup	Fond	Odpracované hodiny	HSEA	Skills
		40925	Xxxxxxxxx Jarmila	CC Valves	Operator	12450 Kč	1956 Kč	16	0,98	157,5	157,5	0,36	7,25%
		40936	Xxxxxxxxx Jana	CC Valves	Operator	12450 Kč	1863 Kč	15	0,98	157,5	157,5	0,36	6,5%
		41024	Xxxxxxxxx Radmila	CC Valves	Operator	12450 Kč	1582 Kč	13	0,98	157,5	157,5	0,36	4,25%
		40727	Xxxxxxxxx Tomáš	CC Valves	Operator	12450 Kč	2392 Kč	19	0,98	157,5	157,5	0,36	10,75%
		40889	Xxxxxxxxx Martin	CC Valves	Operator	12450 Kč	1707 Kč	14	0,98	157,5	157,5	0,36	5,25%
		40924	Xxxxxxxxx Alfred	CC Valves	Operator	12450 Kč	1520 Kč	12	0,98	157,5	157,5	0,36	3,75%
		40570	Xxxxxxxxx Radko	CC Valves	Operator	12450 Kč	1894 Kč	15	0,98	157,5	157,5	0,36	6,75%
		40573	Xxxxxxxxx Gabriela	CC Valves	Operator	12450 Kč	1551 Kč	12	0,98	157,5	157,5	0,36	4%
		40641	Xxxxxxxxx Květoslav	CC Valves	Operator	12450 Kč	1551 Kč	12	0,98	157,5	157,5	0,36	4%
		40560	Xxxxxxxxx Eliška	CC Valves	Operator	12450 Kč	1738 Kč	14	0,98	157,5	157,5	0,36	5,5%
		40561	Xxxxxxxxx Ivo	CC Valves	Operator	12450 Kč	1769 Kč	14	0,98	157,5	157,5	0,36	5,75%
		40567	Xxxxxxxxx Kateřina	CC Valves	Operator	12450 Kč	2018 Kč	16	0,98	157,5	157,5	0,36	7,75%
		40113	Xxxxxxxxx Marie	CC Valves	Operator	12450 Kč	0 Kč	0	0,98	157,5	75	0,36	2%
		40380	Xxxxxxxxx Stanislava	CC Valves	Operator	12450 Kč	4384 Kč	35	0,98	157,5	157,5	0,36	26,75%
		40557	Xxxxxxxxx Libor	CC Valves	Operator	12450 Kč	2018 Kč	16	0,98	157,5	157,5	0,36	7,75%

Individuální data osoby

Skutečný výstup: 23458

Plánovaný výstup: 23894

Fond: 157,5

Příhlášení Personalista: Modul Bonus – schválení > Schválení bonusů

Schválení, respektive uzavření bonusového období. Uzavřením bonusového období, jsou uloženy spočítané hodnoty a dané období se přesune do Historie bonusů a není jej možné dále editovat.

Skill Matrix

[Bonus - schválení](#) > Schválení bonusů

Domů
Pracoviště
Vložit pracoviště
Editovat pracoviště
Import
Nahrání dat z TNA
Osoby
Vyhledávání, detaily osob
Skills
Editovat skills
Vyhledávat skills
Přehled skills
Bonus - výpočet
Měsíční bonus
Měsíční bonus dle divízi
Bonus - schválení
Schválení bonusů
Export bonusů pro Payroll
Bonus - historie
Historie bonusů
Odhlásit

Uzavření bonusového období

Období: 2007 April

Příhlášení Personalista: Modul Bonus – schválení > Export bonusů pro Payroll

Export schválených bonusů ve formátu pro mzdové centrum. Jakmile je bonusové období uzavřeno, lze jej vyexportovat ve formátu pro mzdové centrum.

Skill Matrix

[Bonus - schválení](#) > Export bonusů pro Payroll

Domů
Pracoviště
Vložit pracoviště
Editovat pracoviště
Import
Nahrání dat z TNA
Osoby
Vyhledávání, detaily osob
Skills
Editovat skills
Vyhledávat skills
Přehled skills
Bonus - výpočet
Měsíční bonus
Měsíční bonus dle divízi
Bonus - schválení
Schválení bonusů
Export bonusů pro Payroll
Bonus - historie
Historie bonusů
Odhlásit

Export schválených bonusů pro Mzdové centrum

Mzdové období: 2007 April

Osobní číslo	Mzdová složka	Částka [KC]	Příjmení a jméno	EID
40010	3003	0	Xxxxxxxx Mojmir	E198810
40026	3003	0	Xxxxxxxx Tařana	E286279
40029	3003	0	Xxxxxxxx Věra	E198815
40039	3003	0	Xxxxxxxx Naděřda	E286288
40040	3003	0	Xxxxxxxx Ludmila	E286311
40045	3003	0	Xxxxxxxx Jana	E286451
40059	3003	2490	Xxxxxxxx Radek	E286230
40061	3003	0	Xxxxxxxx Aneřka	E286375
40062	3003	0	Xxxxxxxx Miluře	E286341
40069	3003	0	Xxxxxxxx Eva	E286515
40070	3003	0	Xxxxxxxx Hana	E286205
40072	3003	0	Xxxxxxxx Milada	E286415
40073	3003	0	Xxxxxxxx Miloř	E286235
40075	3003	0	Xxxxxxxx Jitka	E286336
40081	3003	0	Xxxxxxxx Hana	E286269
40082	3003	0	Xxxxxxxx Kateřina	E286504
40097	3003	0	Xxxxxxxx Věra	E286284
40106	3003	0	Xxxxxxxx Miluře	E286462
40111	3003	0	Xxxxxxxx Svatařa	E286391
40113	3003	0	Xxxxxxxx Marie	E286365
40117	3003	0	Xxxxxxxx Antonie	E286371
40127	3003	0	Xxxxxxxx Miroslav	E286281
40128	3003	0	Xxxxxxxx Aleř	E286511
40147	3003	0	Xxxxxxxx Alena	E286528
40181	3003	3984	Xxxxxxxx Lucie	E232624
40225	3003	0	Xxxxxxxx Hana	E286419
40234	3003	0	Xxxxxxxx Radomír	E286502
40235	3003	1992	Xxxxxxxx Jaroslav	E286525
40242	3003	0	Xxxxxxxx Radomira	E286435
40248	3003	1992	Xxxxxxxx Marie	E275423
40249	3003	0	Xxxxxxxx Ludmila	E286261

4.2 Zadání skills

Viz podkapitola 2.3.1 Diagram aktivity nastavování skills.

Přihlášení OPS / Cell Leader: Modul Skills > Editovat skills

Modrý graf znázorňuje probíhající délku zapracování, zelený pak proběhlou délku zapracování a tedy možnost změnit skill. V probíhající délce zapracování může skill změnit pouze personalista. Prázdné pole bez grafu pak znázorňuje, že na daném pracovišti osoba nemá žádný skill ani na něm nezačalo žádné zapracování.

Skill Matrix

[Skills > Editovat skills](#)

Domů

Osoby

Vyhledávání, detaily osob

Skills

Editovat skills

Vyhledávat skills

Přehled skills

Bonus - výpočet

Měsíční bonus

Měsíční bonus dle divizí

Bonus - historie

Historie bonusů

Odhlásit

Správa skills

Divize:

Výroba:

Osoba:

Pozice	OČ	Osoba
40380	Xxxxxxxx	Stanislava
40559	Xxxxxxxx	Lenka
40570	Xxxxxxxx	Radko
40561	Xxxxxxxx	Ivo
41024	Xxxxxxxx	Radmila
40922	Xxxxxxxx	Jana
40925	Xxxxxxxx	Jarmila
40641	Xxxxxxxx	Květoslav
40924	Xxxxxxxx	Alfred
40113	Xxxxxxxx	Marie
40727	Xxxxxxxx	Tomáš
40557	Xxxxxxxx	Libor
40573	Xxxxxxxx	Gabriela
40936	Xxxxxxxx	Jana
40560	Xxxxxxxx	Eliška

CC Valves

Pracoviště	Skill	Váha	Datum započeti
Aquarium / Armature m.	<div style="width: 100%; height: 10px; background-color: green;"></div>	0,00 %	2007-01-01
Big Oil	<div style="width: 100%; height: 10px; background-color: green;"></div>	1,00 %	2007-01-01
Finishing	<div style="width: 100%; height: 10px; background-color: green;"></div>	1,25 %	2007-01-01
Operator Rework	<div style="width: 100%; height: 10px; background-color: green;"></div>	1,50 %	2007-01-01
Operator Test	<div style="width: 100%; height: 10px; background-color: blue;"></div>	0,00 %	2007-01-01
Potting	<div style="width: 100%; height: 10px; background-color: blue;"></div>	0,00 %	2007-01-01
Small Oil	<div style="width: 100%; height: 10px; background-color: green;"></div>	1,75 %	2007-01-01

4.2.1 Zadání nového skill

Přihlášení OPS / Cell Leader: Modul Skills > Editovat skills

Je vybrána výroba, osoba a pracoviště (Potting). Zadání data započeti zapracování (pracovišti Potting)

Skill Matrix

[Skills > Editovat skills](#)

Domů

Osoby

Vyhledávání, detaily osob

Skills

Editovat skills

Vyhledávat skills

Přehled skills

Bonus - výpočet

Měsíční bonus

Měsíční bonus dle divizí

Bonus - historie

Historie bonusů

Odhlásit

Správa skills

Divize:

Výroba:

Osoba:

Pozice	OČ	Osoba
40380	Xxxxxxxx	Stanislava
40559	Xxxxxxxx	Lenka
40570	Xxxxxxxx	Radko
40561	Xxxxxxxx	Ivo
41024	Xxxxxxxx	Radmila
40922	Xxxxxxxx	Jana
40925	Xxxxxxxx	Jarmila
40641	Xxxxxxxx	Květoslav
40924	Xxxxxxxx	Alfred
40113	Xxxxxxxx	Marie
40727	Xxxxxxxx	Tomáš
40557	Xxxxxxxx	Libor
40573	Xxxxxxxx	Gabriela
40936	Xxxxxxxx	Jana
40560	Xxxxxxxx	Eliška

CC Valves

Pracoviště	Skill	Váha	Datum započeti
Aquarium / Armature m.	<div style="width: 100%; height: 10px; background-color: green;"></div>	0,00 %	2007-01-01
Big Oil	<div style="width: 100%; height: 10px; background-color: green;"></div>	1,00 %	2007-01-01
Finishing	<div style="width: 100%; height: 10px; background-color: green;"></div>	1,25 %	2007-01-01
Operator Rework	<div style="width: 100%; height: 10px; background-color: green;"></div>	1,50 %	2007-01-01
Operator Test	<div style="width: 100%; height: 10px; background-color: blue;"></div>	0,00 %	2007-01-01
Potting	<div style="width: 100%; height: 10px; background-color: blue;"></div>	0,00 %	<input type="text" value="2007-5-19"/>
Small Oil	<div style="width: 100%; height: 10px; background-color: green;"></div>	1,75 %	2007-01-01

květen 2007

po	úte	st	čt	pá	so	ne
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Přihlášení OPS / Cell Leader: Modul Skills > Editovat skills

Započatá délka zpracování na skill 1 (pracoviště Potting). Ihned po zadání data započetí zpracování na daném pracovišti, začne probíhat příslušná délka zpracování.

Skill Matrix

[Skills](#) > Editovat skills

Domů

Osoby

Vyhledávání, detaily osob

Skills

Editovat skills

Vyhledávat skills

Přehled skills

Bonus - výpočet

Měsíční bonus

Měsíční bonus dle divizi

Bonus - historie

Historie bonusů

Odhlásit

Správa skills

Divize: CC Valves

Výroba: Operator

Osoba:

POZICE	OČ	Osoba
	40380	Xxxxxxxx Stanislava
	40559	Xxxxxxxx Lenka
	40570	Xxxxxxxx Radko
	40561	Xxxxxxxx Ivo
	41024	Xxxxxxxx Radmila
	40922	Xxxxxxxx Jana
	40925	Xxxxxxxx Jarmila
	40641	Xxxxxxxx Květoslav
	40924	Xxxxxxxx Alfred
	40113	Xxxxxxxx Marie
	40727	Xxxxxxxx Tomáš
	40557	Xxxxxxxx Libor
	40573	Xxxxxxxx Gabriela
	40936	Xxxxxxxx Jana
	40560	Xxxxxxxx Eliška

CC Valves Operator

Pracoviště	Skill	Váha	Datum započetí
Aquarium / Armature m.		0,00 %	2007-01-01
Big Oil		1,00 %	2007-01-01
Finishing		1,25 %	2007-01-01
Operator Rework		1,50 %	2007-01-01
Operator Test		0,00 %	2007-01-01
Potting		0,00 %	2007-05-19
Small Oil		1,75 %	2007-01-01

4.2.2 Změna současného skill

Přihlášení OPS / Cell Leader: Modul Skills > Editovat skills

Je vybrána výroba, osoba a pracoviště (Operátor test). Změna skill, v probíhající době zpracování.

Pod tímto oprávněním není možné skill změnit. V době zpracování může skill změnit jen personalista.

Skill Matrix

[Skills](#) > Editovat skills

Domů

Osoby

Vyhledávání, detaily osob

Skills

Editovat skills

Vyhledávat skills

Přehled skills

Bonus - výpočet

Měsíční bonus

Měsíční bonus dle divizi

Bonus - historie

Historie bonusů

Odhlásit

Správa skills

Divize: CC Valves

Výroba: Operator

Osoba:

POZICE	OČ	Osoba
	40380	Xxxxxxxx Stanislava
	40559	Xxxxxxxx Lenka
	40570	Xxxxxxxx Radko
	40561	Xxxxxxxx Ivo
	41024	Xxxxxxxx Radmila
	40922	Xxxxxxxx Jana
	40925	Xxxxxxxx Jarmila
	40641	Xxxxxxxx Květoslav
	40924	Xxxxxxxx Alfred
	40113	Xxxxxxxx Marie
	40727	Xxxxxxxx Tomáš
	40557	Xxxxxxxx Libor
	40573	Xxxxxxxx Gabriela
	40936	Xxxxxxxx Jana
	40560	Xxxxxxxx Eliška

CC Valves Operator

Pracoviště	Skill	Váha	Datum započetí
Aquarium / Armature m.		0,00 %	2007-01-01
Big Oil		1,00 %	2007-01-01
Finishing		1,25 %	2007-01-01
Operator Rework		1,50 %	2007-01-01
Operator Test		0,00 %	
Potting		0,00 %	
Small Oil		1,75 %	2007-01-01

Přihlášení OPS / Cell Leader: Modul Skills > Editovat skills

Změna skill v době zpracování (modální okno s datem možné změny skill). Uživatel má možnost nahlédnout, kdy bude možné daný skill změnit v závislosti na délku zpracování. (ukázka technologie AJAX)

Skill Matrix Skills > Editovat skills

Domů

Osoby

Vyhledávání, detaily osob

Skills

Editovat skills

Vyhledávat skills

Přehled skills

Bonus - výpočet

Měsíční bonus

Měsíční bonus dle divízi

Bonus - historie

Historie bonusů

Odhlásit

Správa skills

Divize: CC Valves

Výroba: Operator

Osoba:

Posice	OČ	Osoba
40380	Xxxxxxxx	Stanislava
40559	Xxxxxxxx	Lenka
40570	Xxxxxxxx	Radko
40561	Xxxxxxxx	Ivo
41024	Xxxxxxxx	Radmila
40922	Xxxxxxxx	Jana
40925	Xxxxxxxx	Jarmila
40641	Xxxxxxxx	Květoslav
40924	Xxxxxxxx	Alfred
40113	Xxxxxxxx	Marie
40727	Xxxxxxxx	Tomáš
40557	Xxxxxxxx	Libor
40573	Xxxxxxxx	Gabriela
40936	Xxxxxxxx	Jana
40560	Xxxxxxxx	Eliška

CC Valves | Operator

Pracoviště	Skill	Váha	Datum započeti
Aquarium / Armature m.		0,00 %	2007-01-01
Big Oil		1,00 %	2007-01-01
Finishing		1,25 %	2007-01-01
Operator Rework		1,50 %	2007-01-01
Operator Test		0,00 %	2007-01-01
		0,00 %	
		1,75 %	2007-01-01

Detaily zvoleného skill

Současný skill:

Datum změny: 01.03.2007

Délka zpracování: 120 dní

Budoucí skill:

Datum změny: 29.06.2007

Cancel

Přihlášení Personalista: Modul Skills > Editovat skills

Změna skill v době zpracování. Role Personalista umožňuje změnit skill, pokud je potřeba, i v probíhající délce zpracování.

Skill Matrix Skills > Editovat skills

Domů

Pracoviště

Vložit pracoviště

Editovat pracoviště

Import

Nahrání dat z TNA

Osoby

Vyhledávání, detaily osob

Skills

Editovat skills

Vyhledávat skills

Přehled skills

Bonus - výpočet

Měsíční bonus

Měsíční bonus dle divízi

Bonus - schválení

Schválení bonusů

Export bonusů pro Payroll

Bonus - historie

Historie bonusů

Odhlásit

Správa skills

Divize: CC Valves

Výroba: Operator

Osoba:

Posice	OČ	Osoba
40380	Xxxxxxxx	Stanislava
40559	Xxxxxxxx	Lenka
40570	Xxxxxxxx	Radko
40561	Xxxxxxxx	Ivo
41024	Xxxxxxxx	Radmila
40922	Xxxxxxxx	Jana
40925	Xxxxxxxx	Jarmila
40641	Xxxxxxxx	Květoslav
40924	Xxxxxxxx	Alfred
40113	Xxxxxxxx	Marie
40727	Xxxxxxxx	Tomáš
40557	Xxxxxxxx	Libor
40573	Xxxxxxxx	Gabriela
40936	Xxxxxxxx	Jana
40560	Xxxxxxxx	Eliška

CC Valves | Operator

Pracoviště	Skill	Váha	Datum započeti
Aquarium / Armature m.		0,00 %	2007-01-01
Big Oil		1,00 %	2007-01-01
Finishing		1,25 %	2007-01-01
Operator Rework		1,50 %	2007-01-01
Operator Test		0,00 %	
	skill 2	0,00 %	
	Skill 0	0,00 %	
	Skill 1	0,00 %	
	Skill 2	1,75 %	2007-01-01
	Skill 3		
	Skill 4		

5 Závěr

Cílem této práce bylo splnit požadavky společnosti Honeywell, jedné z předních nadnárodních korporací v oblasti strojírenského průmyslu. Požadavkem bylo vytvoření centrálního systému na doškolování a výpočet bonusů zaměstnanců, pracujících ve výrobě a jeho napojení na stávající informační systémy.

Při analýze a návrhu bylo nutné důkladně prostudovat současnou IT infrastrukturu společnosti, zejména pak databázové modely personálních systémů Target 2100 a Evidence 2000, potenciálních zdrojů informací a dat pro systém SkillMatrix. Stejně tak bylo nutné se podrobně seznámit se stávajícími a nově vzniklými firemními procesy a procedurami souvisejícími s doškolováním zaměstnanců. Kromě těchto aspektů bylo žádoucí analyzovat nejnovější technologie návrhu a implementace webových aplikací v prostředí produktů a technologií firmy Microsoft.

Výsledkem tohoto zkoumání je pak robustní a uživatelsky komfortní aplikace, silně založená na databázovém modelu, implementující moderní technologie jako je ASP.NET AJAX a realizující rozhraní pro získávání a aktualizaci dat z klíčových informačních systémů společnosti.

V současné době je aplikace již úspěšně používána v prostředí společnosti Honeywell.

Kromě úspěšné realizace získal tento projekt cenu Bravo a stal se vzorem pro ostatní závody společnosti Honeywell po celé Evropě.

6 Literatura

- [1] Vieira, R.: SQL Server 2000 Programujeme profesionálně. Computer Press, a.s., 2001, 1206 s.
- [2] MacDonald, M., Szpuszta, M.: ASP.NET 2.0 a C# tvorba dynamických stránek profesionálně. Zoner Press, 1376 s.
- [3] Dokumentace pro vývojáře od firmy Microsoft. Dostupné z WWW: < <http://asp.net/>>.
- [4] Dokumentace pro vývojáře od firmy Microsoft. Dostupné z WWW: < <http://ajax.asp.net/>>.
- [5] Dokumentace pro vývojáře od firmy Microsoft. Dostupné z WWW: < <http://msdn2.microsoft.com/>>.